



ASFALIS TransServer

管理者マニュアル

2020年 8月

株式会社エリジョン

目次

1. 概要	2
1.1. ASFALIS TransServerとは	2
1.2. サーバおよびクライアントのシステム要件	2
1.3. 管理者とユーザの管理	4
1.4. 業務種別とオペレーションによるジョブ管理	5
2. クイックスタート	8
2.1. 管理者でのログイン	8
2.2. ユーザ管理	8
2.2.1. 新規ユーザ作成	8
2.3. シナリオ管理	10
2.3.1. シナリオ / コンポーネント	10
2.3.2. 新規シナリオ作成	12
2.3.3. サンプルシナリオ	13
2.4. サーバ設定	16
2.4.1. サーバ設定のエクスポート	17
2.5. メンテナンスモード	18
3. 運用設定	19
3.1. 業務種別管理	19
3.2. オペレーション	20
4. 権限管理を利用した運用設定	21
4.1. 既定のグループ	21
4.2. 権限の設定	21
4.2.1. 業務種別の権限	21
4.2.2. オペレーションの権限	22
4.2.3. シナリオの権限	23
4.2.4. ユーザ・グループの権限	23
5. その他の管理者用機能	25
5.1. ユーザ管理	25
5.2. グループ管理	25
5.2.1. グループ一覧	25
5.2.2. グループ作成・編集	26
5.2.3. グループツリー表示	27
5.3. 業務種別管理	28

5.4. オペレーション管理	28
5.4.1. オペレーションの一括作成	28
5.5. ジョブ管理	31
5.5.1. ジョブの基本操作	31
5.5.2. 時間帯指定によるジョブの実行 (夜間フラグ)	33
5.6. シナリオ管理	33
5.6.1. シナリオの無効化および取戻	33
5.6.2. シナリオ編集	34
5.6.3. シナリオのコピー作成	34
5.6.4. シナリオのグルーピング	34
5.6.5. シナリオに関する注意事項	37
5.7. コンポーネント管理	37
5.7.1. コンポーネントおよびパラメータの有効/無効の設定	37
5.7.2. コンポーネントテンプレート作成・編集	38
5.7.3. コンポーネントテンプレートのコピー	40
5.8. サーバ設定	40
5.8.1. 処理状況の監視に関わる機能	40
5.8.2. リソースフォルダ、共有フォルダが属するボリュームの空き容量監視	41
5.8.3. メール通知	42
5.9. 外部ファイルサーバ管理	46
5.9.1. 外部ファイルサーバー一覧	46
5.9.2. 外部ファイルサーバ登録・編集	46
5.9.3. ジョブ投入・シナリオでの利用	47
5.10. 操作ログ一覧	47
5.11. サービスモニタ	48
5.11.1. Slave Nodeの状態の確認	48
5.11.2. Slave Nodeの追加	49
5.11.3. Slave Nodeの削除	50
5.12. サーバ情報	50
6. バックアップ / リストア	52
6.1. データベースのバックアップ	52
6.1.1. 自動でバックアップファイルを作成する方法	52
6.1.2. 手動でバックアップファイルを作成する方法	53
6.2. データベースのリストア	56
6.2.1. rakeコマンドでリストアする方法	56

6.2.2. 手動でリストアする方法	56
7. 古いファイルの削除	60
7.1. ASFALIS TransServer導入マシン内のファイル	60
7.1.1. 削除対象ファイル	60
7.1.2. 設定手順	60
7.2. 共有フォルダ内のファイル	62
7.2.1. 削除対象ファイル	62
7.2.2. 設定手順	62
8. MIMEタイプと拡張子のカスタマイズ	63
8.1. MIMEタイプのカスタマイズ	63
8.1.1. MIMEタイプの新規作成	63
8.1.2. MIMEタイプの編集	63
8.2. 拡張子のカスタマイズ	64
8.2.1. 拡張子の新規作成	64
8.2.2. 拡張子の編集	66
8.2.3. 拡張子の設定例	66
9. カスタマイズ機能	67
9.1. 概要	67
9.2. カスタムコンポーネント	67
9.2.1. Slave Nodeで動作するカスタムコンポーネントの登録	67
9.2.2. ASFALIS TransServerへのパラメータ情報登録	69
9.2.3. 実行するSlave Nodeの指定	69
9.2.4. レポートファイルの ASFALIS TransServer への登録	69
9.2.5. 任意の情報の ASFALIS TransServer への登録	71
9.3. 規定のURL	73
9.4. ヘルプファイルの変更	73
9.5. SSLの使用	74
9.6. パラメータ中で展開される文字列	74
9.7. ASFALIS TransServerのチューニング	79
9.7.1. 概要	79
9.7.2. DBのチューニング	79
9.7.3. アプリケーションサーバ管理設定のチューニング	80
9.7.4. DBの最大同時接続数の設定	80
9.8. LDAP認証連携	80
9.8.1. 概要	80

9.8.2. LDAP認証連携を使用する	80
9.9. パラメータの拡張機能	81
9.9.1. 概要	81
9.9.2. 設定方法	81
9.9.3. フォーマット	81
9.9.4. 設定例	83
9.9.5. サポートされているパラメータ型	87
9.10. 統合 Windows 認証	87
9.10.1. Active Directory の設定	89
9.10.2. ASFALIS TransServer の設定	90
9.10.3. クライアントコンピュータの設定	93
9.10.4. トラブルシューティング	94
10. ユースケース	96
10.1. 抽出したファイルごとにシナリオを適用する	96
10.1.1. CATIA V5 to IGES変換の例	96
10.2. カスタムコンポーネントの実行結果を後続のコンポーネントで利用する	99
10.2.1. コンポーネントの実行結果をメールで通知する	99
10.2.2. コンポーネントの実行結果に基づいて別のシナリオの実行を制御する	100
10.3. コンポーネントの実行結果に基づくフロー制御	100
10.4. ENF Editorの出力する情報を後続のコンポーネントで利用する	101
10.5. コンポーネントを実行するSlave Nodeを指定する	103
11. トラブルシューティング	105
11.1. ASFALIS TransServerのログイン画面が表示されない	105
11.2. ファイルが存在しないというエラーが発生する	106
11.3. ASFALIS TransServerへのログインに失敗する	107
11.4. タスクマネージャーからのサービス再起動に失敗する	108
11.5. 各ログファイル/オプションの説明	108
12. 付録	109
12.1. メール送信コンポーネント	109
12.1.1. サンプルシナリオ	109
12.2. ジョブ停止コンポーネント	110
12.3. 機密情報削除ツール	110
12.3.1. 基本的な操作手順	111
12.3.2. 機密情報削除コマンドの試行（機密情報検証用ファイルの事前確認）	111
12.3.3. 機密情報削除コマンドの実行（データベースの更新）	112

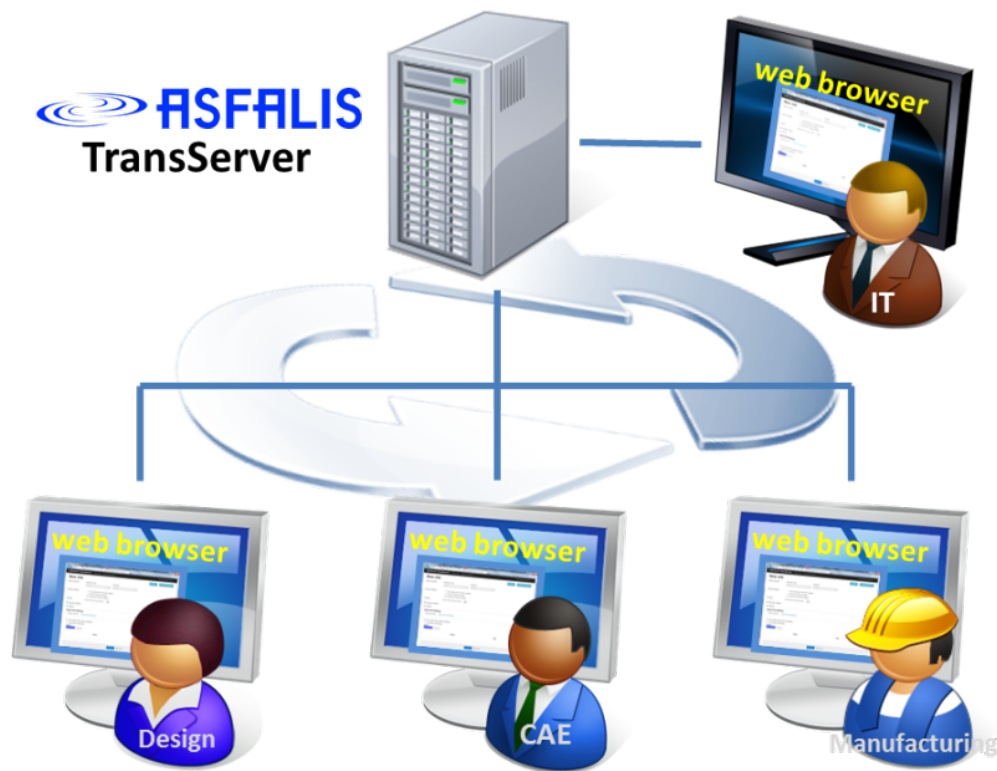
12.3.4. 注意事項	112
12.4. リソースフォルダの変更方法	113
12.5. 共有ネットワークフォルダの変更方法	113
12.6. インポート機能	114
12.6.1. ユーザ	114
12.6.2. グループ	118
12.6.3. 業務種別	121
12.6.4. シナリオ	126
12.6.5. コンポーネントテンプレート	135
12.6.6. カスタムコンポーネント	138
12.7. エクスポート機能	144
12.7.1. ユーザ	145
12.7.2. グループ	145
12.7.3. シナリオ	145
12.7.4. コンポーネントテンプレート	146
12.7.5. カスタムコンポーネント	147
12.8. ASFALIS TransServerに関連するプロセス	147
12.9. その他	148
12.9.1. ASFALIS TransServerへのパラメータ情報登録	148
12.9.2. 実行するSlave Nodeの指定	151
12.9.3. レポートファイルの登録	151

製品を使用する上での注意事項

本製品の管理者は、登録されたユーザの情報やそれぞれのアクセス権を、十分な注意を払って管理するようにしてください。これらの情報が適切に管理されていない場合、不正アクセスや情報漏洩が発生する危険性があります。また製品がインストールされた環境(コンピュータ本体、ネットワーク等)についても、修正プログラムの適用やファイアウォールの設定等、セキュリティ上のリスクを抑止するための運用をお願いいたします。

1. 概要

1.1. ASFALIS TransServerとは



ASFALIS TransServerは、ASFALISをマルチユーザで共有利用するためのWEBベースのクライアントサーバシステムです。本マニュアルはシステム管理者を対象として運用管理方法を説明しています。

ASFALIS TransServerはASFALIS Slave Nodeを利用して動作しますので、管理者はASFALIS Slave NodeとASFALIS TransServerを導入/設定する必要があります。導入の詳細についてはインストールガイドを参照してください。

本ドキュメントではトピックは以下のように分類されています。



注意) : この項目は必ず確認または設定しておいてください。



ヒント) : 操作、設定等に役立つ情報です。



情報) : 参考情報です。

1.2. サーバおよびクライアントのシステム要件

推奨動作環境

- サーバ(ASFALIS TransServer)

	平均同時アクセス数 3以下	平均同時アクセス数4～10
CPU	<ul style="list-style-type: none"> E5-24xx v2系, E5-26xx v2系 2.2GHz 4コア同等以上 E5-26xx v3系, E5-26xx v4系 2GHz 4コア同等以上 Xeon Silver 41xx系, Xeon Gold 51xx系 2GHz 4コア同等以上 	<ul style="list-style-type: none"> E5-24xx v2系, E5-26xx v2系 2.2GHz 8コア同等以上 E5-26xx v3系, E5-26xx v4系 2GHz 8コア同等以上 Xeon Silver 41xx系, Xeon Gold 51xx系 2GHz 8コア同等以上
メモリ	8GB以上	16GB以上
ネットワーク接続	1Gbps以上での接続を推奨	
アプリケーションサーバのインスタンス数*1	10	30
OS	Microsoft Windows Server 2012 RTM Microsoft Windows Server 2012 R2 RTM Microsoft Windows Server 2016	

*1 アプリケーションサーバのインスタンス数の変更方法は「[9.7.3, “アプリケーションサーバ管理設定のチューニング”](#)」をご確認ください。

平均同時アクセス数について

「平均同時アクセス数」とは同時かつ継続的にWebアクセスを繰り返しても応答性能を維持できるアクセス数を指します。「平均同時アクセス数」はアクティブなユーザの数に依存します。アクティブなユーザの数はユーザの業務内容や曜日・時間帯により異なると考えられます。想定される状況の中で最も頻度が高くなる時間帯を基準に見積もってください。なお、ここで『同時』とする基準は『3秒以内』としております。

注：この数字は単純にASFALIS TransServerに登録されているユーザの数とは異なります。

• クライアント

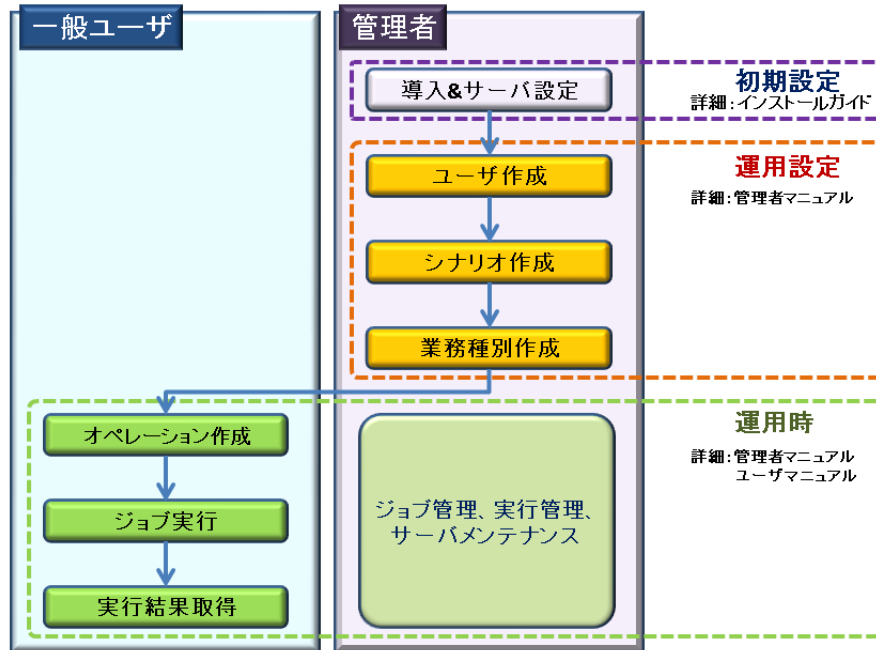
CPU	Core2 Duo E6600同等以上
メモリ	1GB以上
ネットワーク接続	100Mbps以上での接続を推奨
ブラウザ	Internet Explorer 11 Microsoft Edge Google Chrome

ドキュメントモードについて

Internet Explorer 11 を使用している場合であっても、ドキュメントモードが強制的に Internet Explorer 8 に変更されるような環境では、ASFALIS TransServer が正常に動作しない可能性があります。エンタープライズモードを利用している環境や、ドメインのポリシーでドキュメントモードを管理している環境ではこういった現象が発生することがありますのでご注意ください。

ASFALIS Slave Node側のシステム要件については、ASFALIS Slave Nodeのインストールガイドやシステム要件をご確認ください。

1.3. 管理者とユーザの管理



管理者が実行できること

- サーバ設定管理
ASFALIS TransServerの各種設定を変更することができます。
- ユーザ・グループ・権限管理
ユーザやグループの登録、変更、権限設定が可能です。
- シナリオ管理
実行する処理やパラメータを任意に設定した変換シナリオを作成できます。
変換シナリオ作成の他、編集、無効化および取戻ができます。
- 業務種別管理、オペレーション管理
業務種別とオペレーションの登録、設定変更、無効化および取戻が可能です。
- ジョブ・履歴管理
全ユーザのジョブを表示し、実行/停止、優先度変更、削除および取戻が可能です。
- ジョブ実行
ジョブを登録し、実行させることができます。

ユーザが実行できること

- オペレーション作成
既存の業務種別に属するオペレーションを作成することができます。

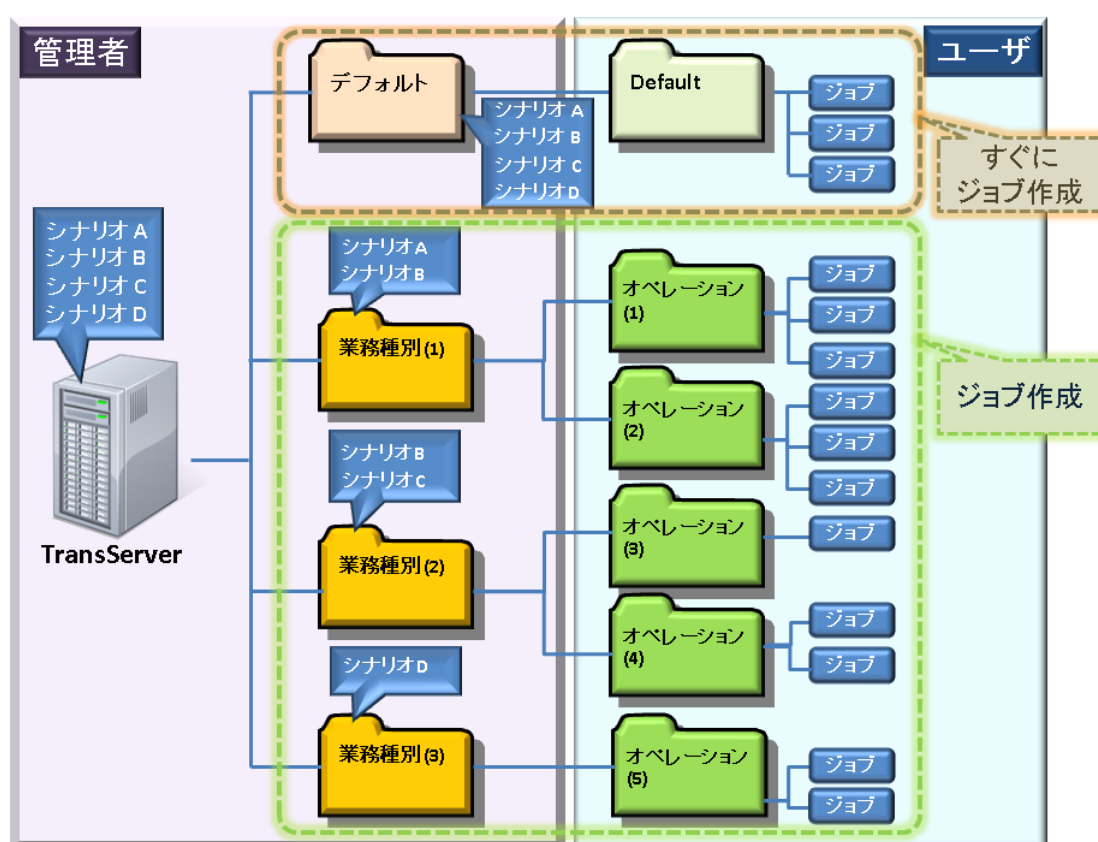
- シナリオのパラメータ変更（シナリオで許可されている場合）
管理者が作成した変換シナリオ内で許可されている場合は、ジョブ作成時に変換パラメータを変更することができます。
- ジョブ実行
ジョブを登録し、実行させることができます。



情報)

業務種別とオペレーションの関係については、「1.4, “業務種別とオペレーションによるジョブ管理”」をご参照ください。

1.4. 業務種別とオペレーションによるジョブ管理



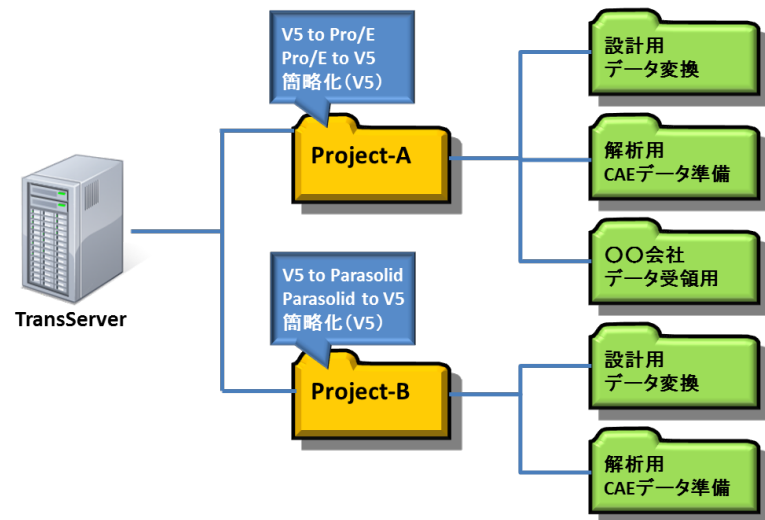
- ジョブ**
「ジョブ」は変換の単位です。それぞれのジョブには変換元の3次元データ、変換結果、ログ、変換の各段階の結果が含まれています。ジョブが失敗した場合は、ログや途中のプロセスが作成したファイルを確認することができます。
- 業務種別**
ジョブや実行結果を整理するために、ジョブを格納するフォルダに該当する機能が用意されています。「業務種別」はその第一階層のフォルダです。
「業務種別」は通常、部署やグループ別に作成/設定します。
- オペレーション**
ユーザは「業務種別」を選択し、その下の階層としてジョブを保存する「オペレーション」を作成することができます。

「オペレーション」は、プロジェクトや業務別に登録します。

。業務種別/オペレーションの使用例 (1)

目的	プロジェクトと変換用途によってジョブを管理します。
設定	プロジェクト毎の業務種別を作成します。各業務種別は、該当プロジェクト関係者が使用できるように設定します。各業務種別の下に用途別のオペレーションを作成します。

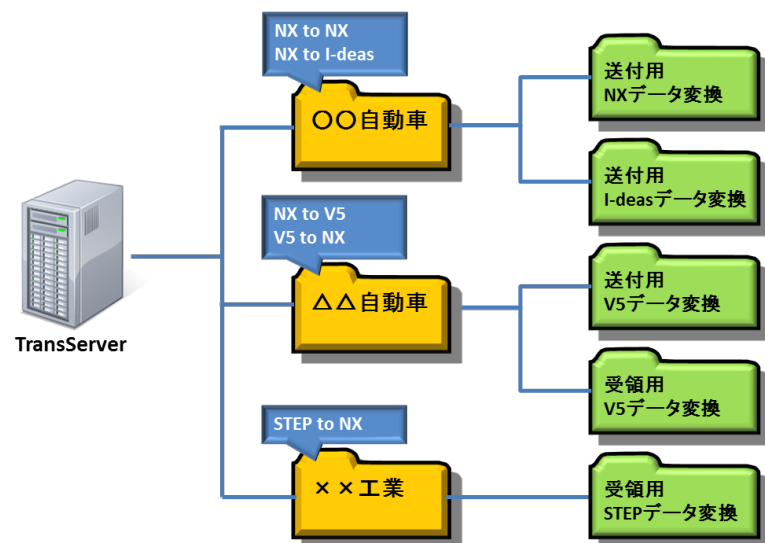
イメージ



。業務種別/オペレーションの使用例 (2)

目的	取引先と変換用途によってジョブを管理します。
設定	取引先毎の業務種別を作成します。各業務種別の下に用途別のオペレーションを作成します。

イメージ



・新規ジョブ

ユーザは二通りの方法でジョブを作成することができます。

「すぐにジョブ作成」では既定の業務種別、オペレーションである「デフォルト」および「Default」が使用されます。特定の業務種別、オペレーションに属さない暫定的なジョブを実行する際に使用します。

「ジョブ作成」では業務種別、オペレーションを指定してジョブを作成、実行します。

2. クイックスタート

本章では、最小限の設定でASFALIS TransServerを使用する方法を示します。全社的にASFALIS TransServerを使用する場合は「3. 運用設定」で示される詳細設定を行う必要があります。

2.1. 管理者でのログイン

インストール直後に既定で作成されている管理者ユーザのログイン名とパスワードは以下の通りです。

ログイン名	admin
パスワード	admin

2.2. ユーザ管理

2.2.1. 新規ユーザ作成

管理者は以下の手順でユーザを作成することができます。

1. [管理者機能] - [ユーザ管理]メニューを選択します。
2. [新規]ボタンを選択します。



3. ユーザ登録画面で必要項目を入力し[登録]ボタンを押します。

ユーザ登録

ユーザ情報

ログイン名 *	<input type="text"/>	タイムゾーン	(GMT+09:00) Osaka ▼
パスワード *	<input type="password"/>	言語優先度	<input checked="" type="radio"/> 日本語 <input type="radio"/> 英語 <input type="radio"/> ドイツ語
メールアドレス	<input type="text"/>	クライアント文字コード	ASCII ▼
メールオプション	<input checked="" type="checkbox"/> ジョブからのメールを受け取る <input checked="" type="checkbox"/> ジョブ失敗時に通知する		
可能な操作	<input checked="" type="checkbox"/> ログイン有効 <input type="checkbox"/> ユーザがメールアドレスを変更可		
		備考	<input type="text"/>

ユーザがファイル名に想定する文字コードを設定します。[詳細](#)

表示名

姓 *	<input type="text"/>	名前の表示形式	日本語
名 *	<input type="text"/>		<input checked="" type="radio"/> [(姓)(名)] <input type="radio"/> [(姓)(空白)(名)] <input type="radio"/> [(名)(空白)(姓)]

グループと権限

所属するグループ	ユーザ名/グループ名 NamedUsers ✓ Everyone ✓		
ユーザが所属するグループを指定して下さい。グループの持つ権限は所属するユーザに引き継がれます。			
対ユーザ・グループ権限	ユーザ名/グループ名	参照	ダウンロード許可
ユーザが他のユーザ・グループに対して可能な操作を設定します。 詳細			

[登録](#)

項目	説明
ログイン名	ASFALIS TransServerのログインに使用する名前を設定します。使用できる文字は、英大文字、英小文字、数字、アンダースコア、ハイフンです。先頭にハイフンを設定することはできません。同一のログイン名で複数のユーザを登録することはできません。ユーザ作成後にログイン名を変更することはできません。
パスワード	4文字以上の文字列を入力します。入力文字種の制限はありません。
メールアドレス	ユーザのメールアドレスを入力します。メールアドレスとして不正な文字列はエラーとなります。
メールオプション	<p>「ジョブからのメールを受け取る」 シナリオでメール送信が指定されている場合、ジョブ実行完了後、ASFALIS TransServerからメールが送信されます。</p> <p>「ジョブ失敗時に通知する」 投入したジョブが異常終了した場合にASFALIS TransServerからメールが送信されます。</p>

項目	説明
可能な操作	「ログイン有効」 アカウントの有効/無効を切り替えます。無効にすると該当ユーザはログインできなくなります。
	「ユーザがメールアドレスを変更可」 ユーザ自身によるメールアドレスの変更を可能にするかの設定です。
タイムゾーン	ローカルのタイムゾーンを設定します。 ログインしているユーザのタイムゾーンに従って時刻が表示されます。
言語優先度	上段に設定した言語が優先して使用されます。
クライアント文字コード	ユーザがファイル名に想定する文字コードを設定します。 アップロード・ダウンロードするファイルやzipファイル内に含まれているファイル名に非ASCII文字が含まれている場合、この設定に従ってファイル名が変換されます。
アクセストークン	Web APIの利用に用いるトークンです。詳細についてはカスタマイズ機能開発向けAPIマニュアルを参照してください。
表示名 (姓名)	ユーザの姓名を設定します。 255文字まで入力可能で入力文字種の制限はありません。
所属するグループ	ユーザが所属するグループを指定します。 グループの持つ権限は所属するユーザに引き継がれます。
対ユーザ・グループ権限	ユーザが他のユーザ・グループに対して可能な操作を設定します。 ここで設定する項目と所属するグループに許可されている操作が、ユーザの可能な操作になります。「参照」は他のユーザの情報を参照する権限、「ダウンロード許可」は指定したユーザ・グループに対して自分の持っている権限を委譲する権限です。

管理者およびユーザは、アカウント作成後も以下の表にある情報を変更することができます。

項目	管理者	一般ユーザ
ログイン名	×	×
パスワード	○	○
姓	○	×
名	○	×

2.3. シナリオ管理

2.3.1. シナリオ / コンポーネント

シナリオとは：

複数のコンポーネントを組み合わせて定義された処理フローをシナリオと呼びます。

コンポーネントとは：

ASFALIS TransServerで行われる処理の最小単位です。コンポーネントを組み合わせるシナリオを作成します。変換を行う「CADインポート」「CADエクスポート」「自動修正」、データ最適化/診断を行う「簡略化」「検証」などのASFALIS Componentsと、「PDQエラー処理制御」「メール送信」などASFALIS TransServerの動作を制御するコンポーネントがあります。

コンポーネント	説明
CADインポート	CADモデルや汎用フォーマットを変換しENFを出力します。 プルダウンで[CADインポート]を選択した後、右側に出現するプルダウンで入力ファイルの種類別のコンポーネントを指定します。 * 例 - [CADインポート][CATIA V5インポート] - [CADインポート][IGESインポート]
CADエクスポート	ENFを変換しCADモデルや汎用フォーマットファイルを出力します。 プルダウンで[CADエクスポート]を選択した後、右側に出現するプルダウンで出力ファイルの種類別のコンポーネントを指定します。 * 例 - [CADエクスポート][NXエクスポート] - [CADエクスポート][Parasolidエクスポート]
自動修正	ENFを、エクスポートするCAD向けに修正します。出力ファイルは修正後のENFです。 CAD to CADの変換は、基本的には「CADインポート → 自動修正 → CADエクスポート」の順で実行します。
簡略化	ENFを入力ファイルとし、形状簡略化処理を行います。出力ファイルは簡略化後のENFです。 サブコンポーネントとしてフィレット消去、穴の削除などの処理内容を選択します。サブコンポーネントを複数指定することも可能です。
ポリゴン化	ENFを入力ファイルとし、ポリゴン化処理を行います。出力ファイルはポリゴンを含むENFです。
ポリゴン最適化	ENFを入力ファイルとし、ポリゴン最適化処理を行います。出力ファイルはポリゴン最適化後のENFです。 サブコンポーネントとしてリメッシュ、平滑化などの処理内容を選択します。サブコンポーネントを複数指定することも可能です。
属性編集	ENFを入力ファイルとし、属性の編集を行います。出力ファイルは属性編集後のENFです。
アセンブリ編集	ENFを入力ファイルとし、アセンブリ構造の編集を行います。出力ファイルは編集後のENFです。
PDQ検証	ENFを入力ファイルとし、データ品質(PDQ)の検証を行います。PDQ検証結果を含むENFが出力されます。
干渉チェック	ENFを入力ファイルとし、アセンブリの干渉チェックを行いチェック結果としてCSVファイルが出力されます。出力ファイルはENFです。
CAD比較	2つのENFファイルを比較します。比較の結果ファイルは Model Viewerで高速表示可能なdvfx形式、もしくは CADdoctor、Model Viewerで読み込み可能なenf_diff形式のファイルで出力されます。
PDQエラー処理制御	PDQチェックの結果、指定したPDQ項目のエラー数が指定の数よりも多い場合に、ジョブを中断します。 PDQエラー処理制御を実行するには、直前に「検証」のコンポーネントを実行し、該当するPDQエラー項目を検証しておく必要があります。事前に検証した項目のみチェック対象とすることができます。

コンポーネント	説明
Windowsファイルサーバへ転送	Windowsの共有ファイルサーバに出力ファイルを転送することができます。事前に管理者機能の「外部ファイルサーバ管理」でファイルサーバを登録しておいてください。
FTPサーバへ転送	FTPサーバに出力ファイルを転送することができます。事前に管理者機能の「外部ファイルサーバ管理」でFTPサーバを登録しておいてください。
ENF編集	ENFを入力ファイルとし、ENFに含まれる情報の取得/ENFの編集を行います。出力ファイルは編集後のENFです。
メール送信	ジョブ完了時にメールを送信します。
出力ファイル名変更	入力されたファイルの名称を指定されたものに変更します。複数ファイル形式の場合、トップアセンブリのファイル名のみが変更されます。
待ち合わせ	直接接続されている2コンポーネントの処理が両方終わるのを待って後続のコンポーネントを実行します。
別のシナリオを実行	指定した別のシナリオを呼び出して実行します。
ジョブ停止	指定した条件が満たされた場合にジョブを異常終了します。

2.3.2. 新規シナリオ作成

1. [管理者機能] - [シナリオ管理]メニューを選択します。
2. [新規]ボタンを押します。



3. シナリオ名と権限を設定した上で、コンポーネントとパラメータを選択し、[登録]を実行します。

ASFALIS TransServer 新規ジョブ オペレーションモニタ 管理者機能 Japanese

シナリオ登録

シナリオ情報

シナリオ名 *

メモ

権限

使用可能ユーザ選択

ユーザ名/グループ名	参照/使用
NamedUsers	✓
ScenarioAdmins	✓

このシナリオを使えるようにするユーザ・グループを選択してください。

使用可能業務種別選択

選択なし

このシナリオを使えるようにする業務種別を選択してください。

コンポーネント設定

すべて開く すべて閉じる

コンポーネントを追加

コンポーネント 1 コンポーネントを選択してください

コンポーネントを追加

登録



注意)

シナリオを作成できるのはシナリオ管理権限をもつユーザ（AdministratorsグループまたはScenarioAdminsグループに属するユーザ）のみです。

2.3.3. サンプルシナリオ

1. IGES to Parasolid変換

1. シナリオ名を記入します。

シナリオ登録

シナリオ情報

シナリオ名 * 日本語

メモ

2. 「コンポーネント1」としてプルダウンから「CADインポート」「IGESインポート」を選択し、パラメータ「自動スティッチ」を設定します。

コンポーネント設定

コンポーネント 1 (iges2enf) CADインポート IGESインポート

入力ファイル選択
入力igesファイル: ジョブ入力 1

☐ コンポーネントの出力ファイルをシナリオの結果ファイルとする: 出力enfファイル
コンポーネントパラメータ設定

●スティッチトレランス (StitchTol) ☐ 設定する

●自動スティッチ (Stitch) ☒ 設定する

パラメータの説明 位相の異なるIGESファイルに対してAuto Stitchを行うかどうか制御する

パラメータ指定方法 ☐ あとでジョブ投入時に指定する
☒ 値を設定する

パラメータの固定値 ☐ Stitchを行わない。
☒ Stitchを行う。

パラメータ可視 ☐ ユーザに表示する

●任意パラメータ (オプション) (custom_parameters) ☐ 設定する

閉じる

ジョブ入力1

↓

IGESインポート (iges2enf)



ヒント)

パラメータ指定方法で「あとでジョブ投入時に指定する」を選択すると、ユーザがジョブ投入時にパラメータ設定を変更することができます。



注意)

各コンポーネントパラメータの設定ダイアログで設定可能なパラメータを任意パラメータで指定することは非推奨です。

[コンポーネントを追加]ボタンを押し、「コンポーネント2」としてプルダウンから「自動修正」を選択します。パラメータ「変換先CAD」を設定します。

コンポーネントを追加

コンポーネント 2 (dek) 自動修正

入力ファイル選択
入力enfファイル: iges2enf出力enfファイル

☐ コンポーネントの出力ファイルをシナリオの結果ファイルとする: 出力enfファイル
コンポーネントパラメータ設定

●変換先CAD (to_cad) ☒ 設定する

パラメータの説明 変換先のCADの種類を指定します。

パラメータ指定方法 ☐ あとでジョブ投入時に指定する
☒ 値を設定する

パラメータの固定値 ACISへ

パラメータ公開設定 ☒ ジョブ詳細画面で公開する

パラメータ可視 ☐ ユーザに表示する

閉じる

ジョブ入力1

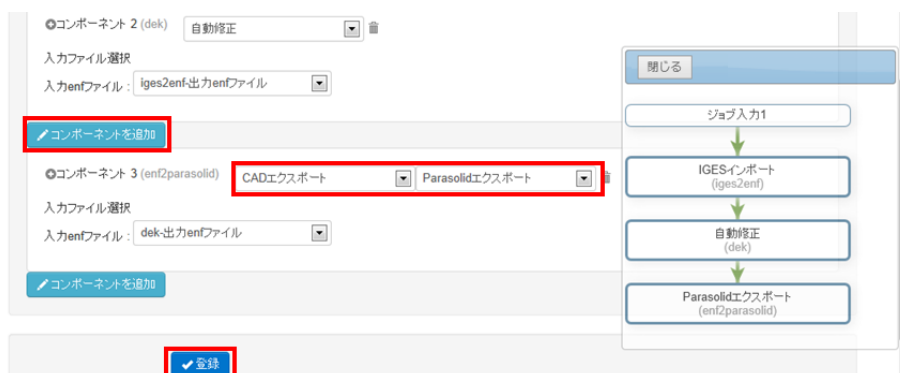
↓

IGESインポート (iges2enf)

↓

自動修正 (dek)

3. [コンポーネントを追加]ボタンを押し、「コンポーネント3」としてプルダウンから「CADエクスポート」「Parasolidエクスポート」を選択します。
4. [登録]ボタンを押すとシナリオの作成は完了です。



2. 2つのCATIA V5ファイルをENFに変換し、形状比較を実行

1. シナリオ名を記入します。
2. 「コンポーネント 1」としてプルダウンから「CADインポート」「CATIA V5インポート」を選択します。入力ファイル選択プルダウンが「ジョブ入力 1」となっていることを確認します。



3. [コンポーネントを追加]ボタンを押し、「コンポーネント 2」としてプルダウンから「CADインポート」「CATIA V5インポート」を選択します。入力ファイル選択プルダウンが「ジョブ入力 2」となっていることを確認します。



4. [コンポーネントを追加]ボタンを押し、「コンポーネント 3」としてプルダウンから「形状比較」を選択します。入力ファイル選択プルダウンの比較元ファイルが「catiaV52enf-出力enfファイル」、比較先ファイルが「catiaV52enf2-出力enfファイル」となっていることを確認します。



5. [登録]ボタンを押すとシナリオの作成は完了です。



ヒント)

シナリオ作成時と作成後一度も使用されていない状態で編集を行う場合、そのシナリオに対してコンポーネントの追加および削除が可能です。

コンポーネント

使用するコンポーネントを選択します。「コンポーネントの出力ファイルをシナリオの結果ファイルとする」にチェックを入れると、コンポーネントでの出力ファイルが、シナリオの出力ファイルとして定義されます。

シナリオの出力ファイルと定義されたファイルはジョブの一覧画面からダウンロードできるようになります。また、オペレーションに対して編集・削除権限の無いユーザも取得できるようになります。シナリオの出力として定義しない場合でも編集・削除権限があればジョブ詳細画面からダウンロードすることができます。権限についての詳細は「[4. 権限管理を利用した運用設定](#)」を参照してください。

パラメータ設定

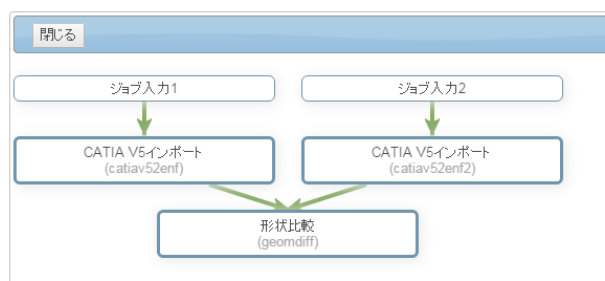
各コンポーネントで用意されたパラメータの使用有無や公開方法を設定します。パラメータ名の横の「設定する」チェックボックスをOnにするとパラメータが有効になります。

使用するパラメータのユーザへの公開方法の制御

- ジョブ投入時にユーザに指定させるか、シナリオで固定値を設定するか
 - ユーザに指定させる場合は、使用することを強制するか否か
 - シナリオ固定とする場合は、固定したパラメータをユーザに見せるか否か

シナリオフローチャート

作成中・編集中のシナリオのコンポーネントフローが、ブラウザ右中央に表示されます。形状比較などジョブ入力複数になるシナリオ、多数のコンポーネントからなるシナリオ、コンポーネント間の接続が複雑なシナリオを作成・編集する際に便利な機能です。「開く/閉じる」ボタンをクリックして、表示/非表示を切り替えることもできます。



2.4. サーバ設定

1. [管理者機能] - [サーバ設定]メニューを選択します。
2. 各種サーバ設定を行います。

ASFALIS TransServer 新規ジョブ オペレーションモニタ 管理者機能 Japanese

サーバ設定 検索

カテゴリ 表示件数 10

設定名, 説明	値
batch.client.address ASFALIS TransServerのバッチ機能(API)を実行するコンピュータのIPアドレスまたはホスト名を指定します。カンマ区切りで複数の値を設定することが出来ます。ASFALIS TransServerが接続元のIPアドレスを確認するのに使用するため、job.iniのSERVER_URLに指定するASFALIS TransServerのホスト名により、必要な指定内容が変わる場合があります。本機能の利用には、ASFALIS TransServer APIライセンスが必要です。 設定例: 127.0.0.1::1	127.0.0.1::1
batch.download.file_server バッチへのファイルダウンロードに使用するWindowsファイルサーバのIDを指定します。	
batch.jobstate.max_fetch_size JobStateバッチが実行された時に返却するジョブ数の上限を規定します。 設定例: 50	50



設定内容の詳細については、各設定と併せて表示される説明の内容を参照してください。

注意)

以下の2項目を必ず設定してください。



設定名	説明
system.default.timezone	ユーザ新規登録時に使われる既定のタイムゾーンを指定します。
web.common.server_base_address	ASFALIS TransServer利用者向けの起点となるURLを設定します。ここで指定したURLが本ドキュメントで記載している<ATS BaseURL>になります。

2.4.1. サーバ設定のエクスポート

サーバ設定の内容をJSON形式のファイルとしてエクスポートする場合には以下の手順で操作してください。エクスポートした情報を以前のものと比較することで、変更点を容易に確認することができます。またこの情報をサポート窓口に提示することで、より迅速に問題を解決できる可能性が高くなります。

1. [管理者機能] - [サーバ設定]メニューを選択します。
2. [エクスポート]を選択します。



ASFALIS TransServer

サーバ設定

すべてのカテゴリ ▼ 設定名, 説明 🔍

エクスポート 表示件数 10 ▼

設定名, 説明	値
batch.client.address ASFALIS TransServerのバッチ機能/APIを実行するコンピュータのIPアドレスまたはホスト名を指定します。カンマ区切りで複数の値を設定することができます。ASFALIS TransServerが接続元のIPアドレスを検証するのに使用するため、job.iniのSERVER_URLに指定するASFALIS TransServerのホスト名により、必要な指定内容が変わる場合があります。本機能の利用には、ASFALIS TransServer APIライセンスが必要です。 設定例: 127.0.0.1,::1	127.0.0.1,::1

またコマンドラインからサーバ設定をエクスポートすることもできます。ASFALIS TransServerが起動しない場合などにはこちらの方法を利用してください。

1. <ASFALIS TransServer 導入フォルダ>\k2dashboard\start-console.bat を実行します。
2. 起動したコマンドプロンプトで以下の rake コマンドを実行します(改行は不要です)。

```
bundle exec rake ats:server_configurations:export  
ATS_SERVER_CONFIGURATIONS_EXPORT_FILE=<出力ファイルパス>
```

2.5. メンテナンスモード

ASFALIS TransServerのメンテナンス時に、管理者権限を持つユーザのみがログインできる状態にすることができます。なお、管理者権限を持つユーザとは以下のいずれかのグループに所属しているユーザを指します。

- Administrators
- UserAdmins
- JobAdmins
- ScenarioAdmins

メンテナンスモードに入るには、サーバ設定でsystem.under_maintenanceにtrueを設定します。これにより、管理者権限を持たないユーザはログインできない状態となります。また、メンテナンスモードに入った時にログインしていた管理者権限を持たないユーザは強制的にログアウトされます。メンテナンス中はASFALIS TransServerの画面上部にメンテナンス中である旨が表示されます。メンテナンスモードを終了するにはsystem.under_maintenanceをfalseにしてください。

3. 運用設定

3.1. 業務種別管理

業務種別とは

ジョブや実行結果を整理するために、ジョブを格納するフォルダに該当する機能が用意されています。業務種別はその第一階層のフォルダで、グループや部署別に設定します。「1.4, “業務種別とオペレーションによるジョブ管理”」のイメージを参照してください。

1. [管理者機能] - [業務種別管理]メニューを選択します。
2. [新規]ボタンを押します。



3. 業務種別名を記入、シナリオを選択し、[登録]します。



業務種別では任意で「業務属性」を設定することができます。業務属性には関連プロジェクト名、送付先会社などを管理者が自由に設定することができ、オペレーション管理の補助情報として使用できます。設定した属性はオペレーションに引き継がれます。



業務種別では任意で「ジョブ優先度の上限」を設定することができます。優先度の上限を設定した場合、当該業務種別を使用して作成されたジョブではその上限よりも上位の優先度を選択することはできません。

1. キー名、ラベル名、説明を記入します。
2. 型、設定を選択し、設定値を入力し[登録]します。

オペレーション属性の定義

キー名 * ProjectName 削除

表示ラベル名 * プロジェクト名 ①

説明 * プロジェクト名を入力

型 テキスト

設定値 A203 ②

設定 ☒ 編集可能 ☒ ユーザ可視 ☐ 登録後編集可 ☐ 検索可

制約 制約設定

オペレーション属性追加

登録

3. 新規オペレーション作成時にオペレーション属性を参照または編集できます。

ASFALIS TransServer 新規ジョブ オペレーションモニタ 管理者機能 管理者

新規オペレーション作成

業務種別選択
for CAE

業務属性設定
オペレーション名

① プロジェクト名 A203 ②

3.2. オペレーション

オペレーションとは

オペレーションは業務種別の次の階層のフォルダです。プロジェクトや業務を登録します。業務種別との違いについては「1.4, “業務種別とオペレーションによるジョブ管理”」のイメージを参照してください。

1. オペレーションモニタから[新規オペレーション作成]ボタンを押します。

ASFALIS TransServer 新規ジョブ オペレーションモニタ 管理者機能 管理者

オペレーションモニタ

新規オペレーション作成 検索

すべて

アーカイブ 取戻し

表示件数 10

2. 業務種別とオペレーション名を設定し、[登録]を実行します。

ASFALIS TransServer 新規ジョブ オペレーションモニタ 管理者機能 Japanese

新規オペレーション作成

業務種別選択 Sample

業務属性設定
オペレーション名

4. 権限管理を利用した運用設定

4.1. 既定のグループ

グループ名	説明
UserAdmins	管理者メニューの以下の操作が可能なグループです。 * ユーザ管理 * グループ管理
JobAdmins	管理者メニューの以下の操作が可能なグループです。 * ジョブ管理
ScenarioAdmins	管理者メニューの以下の操作が可能なグループです。 * シナリオ管理 * 業務種別管理
Administrators	全ての操作が可能なグループです。
Everyone	全ユーザが所属するグループです。
NamedUsers	匿名ユーザ以外の全ユーザが所属するグループです。

4.2. 権限の設定

ASFALIS TransServerでは「業務種別」「オペレーション」「シナリオ」「ユーザ・グループ」に対して権限を設定することができます。

4.2.1. 業務種別の権限

業務種別を作成・編集する際、該当の業務種別から作成されたオペレーションに対して、指定されたユーザ・グループが持てる権限の上限を設定することができます。

この設定は特定の利用者が特定の業務種別を担当しているような場合に役に立ちます。このような場合、業務属性の権限設定は実際の利用者と業務の種類の関連を表すことになります。

ASFALIS TransServer 新規ジョブ オペレーションモニタ 管理者機能 管理者

業務種別登録

業務種別情報

業務種別名 ジョブ優先度の上限 低

シナリオ選択 選択なし

この業務種別で使用可能なシナリオを設定してください。

権限

権限設定の上限	ユーザ名/グループ名	参照	ダウンロード	更新削除
	JobAdmins	✓	✓	✓

この業務種別から作成したオペレーションに対してユーザ・グループが持てる権限の上限を設定します。

オペレーション属性の定義

4.2.2. オペレーションの権限

この権限設定項目は管理者だけでなく、オペレーションを作成するすべてのユーザが行うことになります。

オペレーションの権限には3段階あります。「参照」権限はオペレーションや含まれるジョブの詳細な情報を参照するための権限です。「ダウンロード」権限は含まれるジョブの結果ファイルをダウンロードするための権限です。「更新・削除」はジョブの作成、ジョブの中間ファイルとログファイルのダウンロード、オペレーションの情報更新、オペレーションの削除を行うための権限です。

オペレーションを作成・編集する際、「ダウンロード許可」権限を持っているユーザ・グループに対してアクセスレベルを指定することができます。

ASFALIS TransServer 新規ジョブ オペレーションモニタ 管理者機能 Japanese

新規オペレーション作成

業務種別選択 Sample

業務属性設定

オペレーション名

権限設定	ユーザ名/グループ名	参照	ダウンロード	更新/無効化
	Japanese	✓	✓	✓
	JobAdmins	✓	✓	✓

それぞれのユーザ・グループに与える権限を指定して下さい。 [詳細](#)

4.2.3. シナリオの権限

シナリオを作成・編集する際に、どのユーザ・グループに対してそのシナリオを利用できるようにするか設定することができます。この設定は特定のシナリオやコンポーネントの利用を制限したい場合に役に立ちます。初期設定ではNamedUsers (anonymousを除くすべてのユーザを含むグループ)がすべてのシナリオを利用可能になります。この振る舞いを変更したい場合にはサーバ設定の項目 "web.scenario.creation.default_permission_read_use_for_named_users.enabled" を参照してください。

ASFALIS TransServer 新規ジョブ オペレーションモニタ 管理者機能 管理者

シナリオ登録

シナリオ情報

シナリオ名 *

メモ

権限

使用可能ユーザ選択	ユーザ名/グループ名	参照/使用
<input checked="" type="checkbox"/>	ScenarioAdmins	✓
<input checked="" type="checkbox"/>	NamedUsers	✓

このシナリオを使えるようにするユーザ・グループを選択してください。

コンポーネント設定

4.2.4. ユーザ・グループの権限

ユーザ・グループを作成・編集する際、そのユーザ・グループが他のユーザ・グループに対して可能な操作を設定することができます。「参照」は他のユーザ・グループの情報を参照する権限、「ダウンロード許可」は指定したユーザ・グループに対して自分の持っている権限を委譲する権限です。

ユーザ登録

ユーザ情報

ログイン名 *	<input type="text"/>	タイムゾーン	(GMT+09:00) Osaka
パスワード *	<input type="password"/>	言語優先度	<input type="radio"/> 日本語
メールアドレス	<input type="text"/>		<input type="radio"/> 英語
メールオプション	<input checked="" type="checkbox"/> ジョブからのメールを受け取る <input checked="" type="checkbox"/> ジョブ失敗時に通知する		<input type="radio"/> ドイツ語
可能な操作	<input checked="" type="checkbox"/> ログイン有効 <input type="checkbox"/> ユーザがメールアドレスを変更可	クライアント文字コード	ASCII
			ユーザがファイル名に想定する文字コードを設定します。 詳細
		備考	<input type="text"/>

表示名

姓 *	<input type="text"/>	名前の表示形式	日本語
名 *	<input type="text"/>		<input checked="" type="radio"/> [(姓)(名)]
			<input type="radio"/> [(姓)(空白)(名)]
			<input type="radio"/> [(名)(空白)(姓)]

グループと権限

所属するグループ	<div>ユーザ名/グループ名</div> <div><div>NamedUsers</div><div>Everyone</div></div>
ユーザが所属するグループを指定して下さい。グループの持つ権限は所属するユーザに引き継がれます。	
対ユーザ・グループ権限	<div>ユーザ名/グループ名</div> <div>参照 ダウンロード許可</div>
ユーザが他のユーザ・グループに対して可能な操作を設定します。 詳細	



情報)

グループに含まれているユーザは、自分自身に与えられた権限に加えてグループに設定された権限を使用することができます。

5. その他の管理者用機能

5.1. ユーザ管理

管理者はユーザに関する以下の情報を変更可能です。

- ユーザ情報・表示名の変更
- ログイン権限の付与
- ユーザの削除と削除取消し

手順は以下の通りです。

1. [管理者機能] - [ユーザ管理]メニューを選択します。
2. 編集したいユーザのログイン名をクリックします。

ASFALIS TransServer 新規ジョブ オペレーションモニタ 管理者機能 Japanese

ユーザー一覧 新規 検索

ロック ロック解除 無効化 取戻 エクスポート 表示件数 10

ログイン名	メールアドレス	言語
admin	admin@example.com	日本語: 管理者 英語: Administrator ドイツ語: Administratoren
anonymous		日本語: 匿名ユーザ 英語: Anonymous ドイツ語: anonym

ヒント)

以下のアイコンからユーザ属性を変更できます。



ログイン有効にする		ログイン無効にする	
ユーザ削除		取戻	

ユーザー一覧には削除したユーザも表示されますので、リストから該当ユーザを再度ログイン可能に設定できます。



ヒント)

ユーザ情報をJSON形式でエクスポートすることができます。詳しくは「[12.7, “エクスポート機能”](#)」をご覧ください。

5.2. グループ管理

AdministratorsかUserAdminsに含まれているユーザは以下の画面を使用することができます。

5.2.1. グループ一覧

[管理者機能] - [グループ管理]メニューを選択することでグループ一覧を表示することができます。



既定のグループ"Administrators", "UserAdmins", "JobAdmins", "ScenarioAdmins", "NamedUsers", "Everyone"については編集・削除することができません。これらのグループにメンバーを追加する際にはユーザ作成・編集画面から操作を行ってください。



ヒント)

以下のアイコンからグループの情報を変更できます。

グループ作成・編集		ログイン無効にする	
取戻			

グループを無効化するには下記3つの条件を満たす必要があります。

- ルートグループに指定されていない
- 親グループに何も指定されていない
- メンバーに何も指定されていない

5.2.2. グループ作成・編集

グループを作成するにはグループ一覧画面から[新規]ボタンを選択します。グループを編集するにはグループ一覧画面からグループ名を選択します。



グループ名とルートグループに指定するかどうか、作成中のグループの親グループ、含まれるメンバーを選択します。グループが持つ他のユーザ・グループへの権限を設定します。

ASFALIS TransServer 新規ジョブ オペレーションモニタ 管理者機能 Japanese

グループ編集

グループ情報

グループ ID 42

グループ名

グループと権限

所属するグループ (Root)

ユーザ名/グループ名 -

このグループが所属する親グループを指定して下さい。親グループの持つ権限はこのグループに引き継がれます

メンバー

ユーザ名/グループ名 -

Japanese

このグループに含めるユーザ・グループを指定して下さい。このグループの持つ権限はメンバーのユーザ・グループに引き継がれます

権限設定

ユーザ名/グループ名	参照	ダウンロード許可
Administrators		
My Group		

グループに含まれるユーザが他のユーザ・グループに対して可能な操作を設定します。[詳細](#)

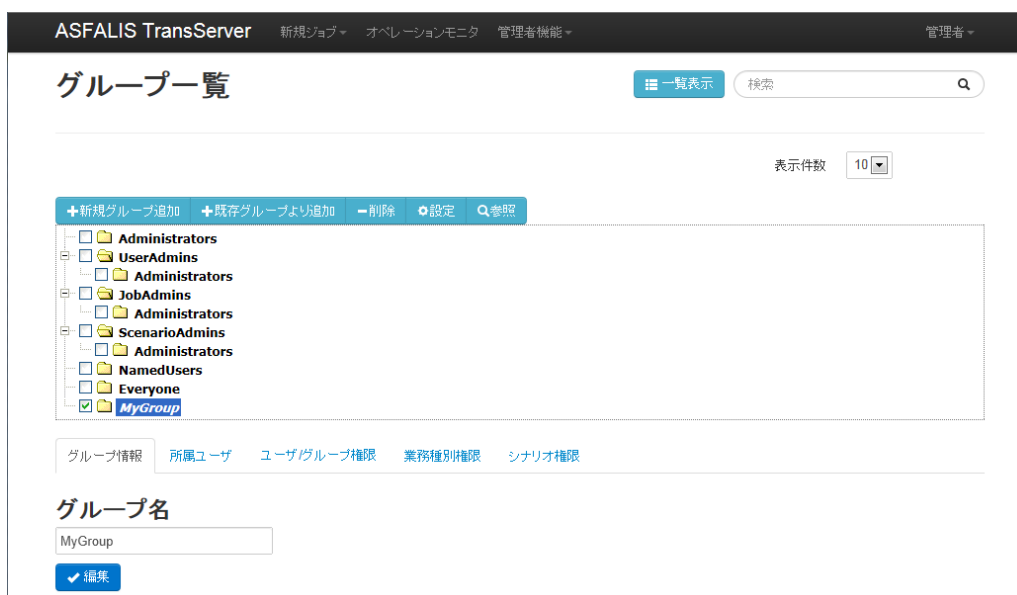
更新

5.2.3. グループツリー表示

グループの一覧表示の状態で[ツリー表示]ボタンを選択します。



ルートグループからはじまる親子関係をツリーで表現した構造が表示されます。選択したグループの情報を編集することができます。



5.3. 業務種別管理

「運用設定」の「3.1, “業務種別管理”」をご参照ください。

5.4. オペレーション管理

5.4.1. オペレーションの一括作成

既存のユーザ、グループごとのオペレーションを一括して作成することができます。対象となる業務種別のIDまたは識別名を指定してください。また、作成したいオペレーションの情報をまとめたファイルを作成してインポートすることもできます。手順は以下の通りです。

- 全ユーザに対するオペレーション一括作成

登録されているユーザのうち、既定で存在する admin および anonymous を除くすべてのユーザに対してオペレーションを一括で作成することができます。作成されるオペレーションの名称は "Private (user: <UserName>)" となります。<UserName>の部分にはシステムのデフォルト言語でのユーザの表示名が入ります。デフォルト言語での表示名が存在しない場合には、以降の優先言語で存在する名前が入ります。コマンドは以下の通りです。

```
bundle exec rake ats:job_lots:bulk_private_create TARGET=ALL_USERS
JOB_LOT_TEMPLATE_ID=1
```

または

```
bundle exec rake ats:job_lots:bulk_private_create TARGET=ALL_USERS
JOB_LOT_TEMPLATE_UNIQUE_ALIAS=job_lot_template_SAMPLE
```

- 全グループに対するオペレーション一括作成

登録されているグループのうち、既定で存在する Administrators、UserAdmins、JobAdmins、ScenarioAdmins、NamedUsers、Everyone を除くすべてのグループに対してオペレーションを一括で作成することができます。作成されるオペレーションの名称は Private (group: <GroupName>) となります。<GroupName>の部分にはシステムのデフォルト言語でのグループの表示名が入ります。デフォルト言語での表示名が存在しない場合には、以降の優先言語で存在する名前が入ります。コマンドは以下の通りです。

```
bundle exec rake ats:job_lots:bulk_private_create TARGET=ALL_GROUPS
JOB_LOT_TEMPLATE_ID=1
```

または

```
bundle exec rake ats:job_lots:bulk_private_create TARGET=ALL_GROUPS
JOB_LOT_TEMPLATE_UNIQUE_ALIAS=job_lot_template_SAMPLE
```

- ユーザ・グループを指定してのオペレーション一括作成

後述のフォーマットのファイルを作成して以下のコマンドを実行することにより、複数のオペレーションを一括で作成することができます。sample.txt の部分は実際のファイルの絶対パスで置き換えてください。

```
bundle exec rake ats:job_lots:bulk_private_create TARGET=FILE
FILE_PATH=sample.txt JOB_LOT_TEMPLATE_ID=1
```

または

```
bundle exec rake ats:job_lots:bulk_private_create TARGET=FILE
FILE_PATH=sample.txt JOB_LOT_TEMPLATE_UNIQUE_ALIAS=job_lot_template_SAMPLE
```

ファイルのフォーマットは以下の通りです。<tab>はタブで置き換えてください。

```
<specifier_key><tab><specifier_value><tab><job_lot_template_id_key><tab><job_lot_template_id><tab><name_locale1><tab><name_value1><tab><name_locale2><tab><name_value2>...
```

最後の "... " の部分には、ロケール<name_locale*>とそのロケールでの表示文字列<name_value*>の組み合わせ(*は1,2,...)をサポートされているロケールの数だけ設定することができます。

各項目の説明は以下の表の通りです。

項目名	説明	必須/任意
specifier_key	作成するオペレーションの対象となるユーザもしくはグループの指定方法を設定します。サポートされる値は以下の通りです。 <ul style="list-style-type: none"> • user_login_name • group_id • group_unique_alias 	必須
specifier_value	作成するオペレーションの対象となるユーザもしくはグループを"specifier_key"で指定した形式で設定します。	必須
job_lot_template_id_key	業務種別を指定する場合には"job_lot_template_id"を設定します。	任意
job_lot_template_id	"job_lot_template_id_key"を設定した場合には、業務種別IDの値を設定します。"specifier_value"で指定したユーザもしくはグループはこの業務種別に対して"更新/無効化"権限を持っている必要があります。	任意 (job_lot_template_id_keyを指定した場合には必須)
name_locale* (*は1,2,...)	オペレーション表示名のロケールを設定します。対応する"name_value*"に設定する値がここで設定するロケールでの表示名になります。サポートされる値は以下の通りです。 <ul style="list-style-type: none"> • en-US • ja-JP • de-DE • zh-CN 	任意
name_value* (*は1,2,...)	"name_locale*"で設定したロケールでの表示名を設定します。	任意 (name_locale*を指定した場合には必須)

"job_lot_template_id_key" と "job_lot_template_id" で業務種別を設定しない場合には、新たに業務種別が自動で作成されます。作成される業務種別の表示名は以下の通りとなります。

- ユーザに対するオペレーションの場合
"User <ユーザのデフォルト優先言語での表示名> Dedicated"
- グループに対するオペレーションの場合
"Group <グループのデフォルト優先言語での表示名> Dedicated"

"name_locale*" と "name_value*" (*は1,2,...) で表示名を指定しない場合には、以下の表示名が設定されます。

- ユーザに対するオペレーションの場合
"Private (User: <UserName>)"

<UserName> はシステムのデフォルト言語での表示名が入ります。デフォルト言語での表示名が存在しない場合には、以降の優先言語で存在する名前が入ります。

- グループに対するオペレーションの場合
"Private (Group: <GroupName>)"

<GroupName> はシステムのデフォルト言語での表示名が入ります。デフォルト言語での表示名が存在しない場合には、以降の優先言語で存在する名前が入ります。

ファイルの記載例

```
user_login_name user1 job_lot_template_id 3 ja-JP 検証
user_login_name user2
group_unique_alias group3 job_lot_template_id 4
group_id 12 job_lot_template_id 5 ja-JP 業務 de-DE Geschäft
```

オペレーションの一括作成でユーザごとに作成したオペレーションは、当該ユーザのスタートアップオペレーションに設定されます。

5.5. ジョブ管理

5.5.1. ジョブの基本操作

ジョブ管理画面から、以下のジョブ操作を実行することができます。

- ジョブの実行、停止
- ジョブの削除および取戻
- ジョブの優先度変更

手順は以下の通りです。

1. [管理者機能] - [ジョブ管理]メニューを選択します。
2. 編集したいジョブにチェックを入れ各操作のボタンを押します。

ASFALIS TransServer 新規ジョブ オペレーションモニタ 管理者機能

ジョブ管理

作成者 🔍

■ 中止 ○ 無効化 ◀ 取戻

表示件数 10 ▼

<input type="checkbox"/>	ジョブID	入力ファイル	登録日時	作成者	業務種別	オペレーション	シナリオ	ステータス
<input type="checkbox"/>	1	sample2.igs	07/21 18:36	管理者	デフォルト	デフォルト	IGES to Parasolid #1	異常終了



注意)
一度中止させたジョブを再開することはできません。再実行する必要があります。



ヒント)
サーバ設定の項目"web.common.admin_show_job_options"がFALSE(デフォルト)に設定されている場合、優先度の列は表示されません。



ヒント)
下記の条件を満たす場合、ジョブ詳細ページに[サポート]ボタンが表示されます。

- ユーザがAdministratorsまたはJobAdminsグループに属している
- 調査のためジョブの実行フォルダが取得できる状態にある
サーバ設定 scheduler.common.on_error_keep_directory または scheduler.common.keep_directory を使用して、実行フォルダを調査のために保存し、取得できるようにすることができます。詳細は各項目の説明を参照ください。



[サポート]ボタンが表示されているとき、以下の機能を使用することができます。

- 実行ディレクトリをダウンロード:
調査のため残されている実行ディレクトリをzipファイルとして全てダウンロードする機能です。
- 実行ディレクトリからモデルを除いてダウンロード:
調査のため残されている実行ディレクトリをzipファイルとしてダウンロードする機能です。サーバ設定 system.filename.model_pattern に該当する名前のファイルは対象から除かれます。
- 実行ディレクトリを削除する:
残されている実行ディレクトリを削除します。実行後、ダウンロードして取得することはできません。



ヒント)
ジョブ詳細ページの[自動リロード]ボタンをONにした場合、ジョブ詳細のステータスが自動で更新されます。更新間隔は以下のサーバ設定にて指定することができます(初期値: 5秒)。

web.common.auto_reload_interval_seconds



5.5.2. 時間帯指定によるジョブの実行 (夜間フラグ)

夜間フラグを設定すると特定の時間帯にだけジョブが実行されるようになります。設定手順は以下の通りです。

1. サーバ設定の項目"web.common.admin_show_job_options"をTRUEに設定します。
2. サーバ設定の項目"scheduler.common.nighttime_in_utc"にジョブの実行時間帯を設定します。
UTCで指定する必要がありますのでご注意ください。複数の時間帯を指定する場合はカンマで区切ってください。(例 : 18:00-23:00, 2:00-6:00)
3. [管理者機能] - [ジョブ管理]メニューを選択します。
4. 変更対象のジョブの[夜間フラグ]をチェックします。

ASFALIS TransServer 新規ジョブ オペレーションモニタ 管理者機能									
ジョブ管理									
<div> ■ 中止 🔄 無効化 🔍 取戻 + 優先度上昇 - 優先度低下 </div> <div>表示件数 10</div>									
<input type="checkbox"/> ジョブID	入力ファイル	登録日時	作成者	業務種別	オペレーション	シナリオ	ステータス	優先度	夜間フラグ
<input type="checkbox"/> 3593	Sample.prt	10/05 17:53	管理者			NXtoENF	未処理	低+	<input checked="" type="checkbox"/>



ヒント)

オペレーション詳細から一般ユーザが夜間フラグを設定できるようにするためには、併せて"web.common.show_job_options"をTRUEに設定します。

5.6. シナリオ管理

以下のシナリオ管理操作を実行できます。

- ・シナリオの無効化および取戻
- ・シナリオ編集
- ・シナリオのコピー作成
- ・シナリオのグルーピング

5.6.1. シナリオの無効化および取戻

1. [管理者機能] - [シナリオ管理]メニューを選択します。
2. シナリオを選択し[無効化]または[取戻]ボタンを押します。

ASFALIS TransServer 新規ジョブ オペレーションモニタ 管理者機能						
シナリオ一覧						
<div> 🔄 無効化 🔍 取戻 📄 インポート 📄 エクスポート </div> <div>表示件数 10</div>						
<input type="checkbox"/> シナリオID	シナリオ名	メモ	更新日時	作成者		
<input type="checkbox"/> 1	Sample Scenario		11/21 10:39	管理者 (admin)		
<input type="checkbox"/> 2	Sample Scenario 2		11/21 10:39	管理者 (admin)		

無効化および取戻はアイコンからの実行も可能です。

無効化		取戻	
-----	---	----	---



ヒント)

シナリオ情報をJSON形式でインポート・エクスポートすることができます。

詳しくは、[12.6.4, “シナリオ”](#)と[12.7.3, “シナリオ”](#)をご覧ください。

5.6.2. シナリオ編集

1. [管理者機能] - [シナリオ管理]メニューを選択します。
2. シナリオ名をクリックします。
3. シナリオ情報/権限/コンポーネント設定を編集し、[更新]ボタンを押します。



シナリオ編集

シナリオ情報

シナリオID 11

シナリオ名 * IGES to Parasolid

メモ

使用可能業務種別選択 選択なし
このシナリオをえるようにする業務種別を選択してください。

所属するシナリオグループ 選択なし

権限

使用可能ユーザ選択

ユーザ名/グループ名	参照/使用
NamedUsers	✓
ScenarioAdmins	✓

このシナリオをえるようにするユーザ・グループを選択してください。

コンポーネント設定

すべて開く すべて閉じる

コンポーネントを追加

コンポーネント 1 (iges2enf) CADインポート IGESインポート

コピー作成フロー:

```

graph TD
    A[コピー作成] --> B[ジョブ入力1]
    B --> C[IGESインポート (iges2enf)]
    C --> D[自動修正 (dek)]
    D --> E[Parasolidエクスポート (enf2parasolid)]
  
```

5.6.3. シナリオのコピー作成

シナリオ編集ページで[コピー作成]ボタンを押すとシナリオのコピーが作成され、別シナリオとして編集、登録することができます。

5.6.4. シナリオのグルーピング

サーバ設定で web.common.enable_scenario_grouping の値を TRUE に設定することで、シナリオのグルーピング機能が有効になります。有効にすると新規ジョブを作成する際、シナリオを選択する前にシナリオグループを選択することで、選択肢に表示されるシナリオをフィルタリングすることができます。

・フィルタリング前(すべてのシナリオを表示)

シナリオ設定

- ☒ 新規にシナリオを設定する
- ☐ 既存ジョブの設定を流用する
- ☐ 既存ジョブをパラメータを変えて再実行する

シナリオ選択

すべてのシナリオ

シナリオを選択してください

- シナリオを選択してください
- Check Invisible Parts
- IGES to STEP
- STEP to ENF
- test

・フィルタリング後

シナリオ設定

- ☒ 新規にシナリオを設定する
- ☐ 既存ジョブの設定を流用する
- ☐ 既存ジョブをパラメータを変えて再実行する

シナリオ選択

Scenario Group1

シナリオを選択してください

- シナリオを選択してください
- IGES to STEP
- test

また管理者グループまたはシナリオ管理者グループに所属するユーザは、以下の操作を実行することができます。

a. シナリオグループの新規作成、編集

シナリオグループを新規に作成することができます。シナリオグループ名と所属シナリオを設定します。またシナリオグループ一覧からグループ名を選択することで、既存シナリオグループのシナリオグループ名と所属シナリオを変更することもできます。

。シナリオグループの作成

シナリオ一覧

[新規](#) [シナリオグループ一覧](#) シナリオ名

[無効化](#) [戻る](#) [インポート](#) [エクスポート](#) 表示件数 10

シナリオグループ一覧

[新規](#) シナリオグループ名

表示件数 10

シナリオグループID	シナリオグループ名
1	Scenario Group1
2	Scenario Group2

。シナリオグループ名および所属シナリオの設定

シナリオグループ登録

シナリオグループ情報

シナリオグループ名 Scenario Group Name 日本語 +

所属シナリオ 選択

シナリオ名

test

NX to ENF

表示件数 10

- 。既存シナリオグループの編集

シナリオグループ編集

シナリオグループ情報

シナリオグループID 1

シナリオグループ名 Scenario Group1

所属シナリオ 選択

シナリオ名

test

NX to ENF

Check Invisible Parts

Check Invisible Part 2

STEP to ENF

ACIS to ENF

IGES to STEP

表示件数 10

b. シナリオグループの設定

シナリオを作成する際に、所属するシナリオグループを設定することができます。

シナリオ登録

シナリオ情報

シナリオ名 * Scenario Test

メモ

使用可能業務種別選択 選択なし

このシナリオをえるようにする業務種別を選択してください。

所属するシナリオグループ 選択なし

権限

使用可能ユーザ選択

このシナリオをえるようにするユーザ・グループを選択してください。

参照/使用

✓

✓

また既存シナリオの所属シナリオグループを変更したり新規に設定したりすることもできます。

シナリオ編集

コピー作成

開く

シナリオ情報

シナリオID 7

シナリオ名* IGES to STEP

メモ

使用可能業務種別選択 選択なし

このシナリオをえるようにする業務種別を選択してください。

所属するシナリオグループ

権限

使用可能ユーザ選択

参照/使用

このシナリオをえるようにするユーザグループを選択してください。

5.6.5. シナリオに関する注意事項

シナリオを作成、編集する際には、以下の点に注意してください。

- コンポーネントの追加または削除を行うことができるのは、新規シナリオ作成時および作成後一度も使用されていないシナリオを編集する場合のみです。
- 他のシナリオから呼び出されるシナリオ (子シナリオ) のパラメータを設定する際は「値を設定する」を選択し、「パラメータの固定値」を指定してください。具体例については[10.1.1, “CATIA V5 to IGES変換の例”](#)を参照してください。

5.7. コンポーネント管理

5.7.1. コンポーネントおよびパラメータの有効/無効の設定

コンポーネントおよびパラメータの有効/無効の設定を行うことができます。有効/無効の設定はシナリオ作成機能のみに影響し、ユーザのジョブ設定やジョブの実行時には影響しません。

コンポーネント・サブコンポーネント

- シナリオ新規作成時：

有効なコンポーネント・サブコンポーネントのみプルダウン中に表示されます。
- シナリオ編集時：

有効なコンポーネント・サブコンポーネントがプルダウン中に表示されます。

コンポーネント・サブコンポーネントが既に使われている場合には有効/無効のフラグによらずプルダウン中に表示されます。

コンポーネントのパラメータ

- シナリオ新規作成時：

有効なパラメータのみ表示されます。

- シナリオ編集時：
有効なパラメータが表示されます。既に使用されているパラメータは有効/無効のフラグによらず表示されます。

設定手順は以下の通りです。

- [管理者機能] - [コンポーネント管理]メニューを選択します。
- コンポーネント一覧ページでは"Enabled"のチェックボックスでコンポーネントの有効/無効を設定することができます。チェックボックスのチェックON/OFFを変更した時点で有効/無効の設定が保存されます。

ASFALIS TransServer 新規ジョブ オペレーションモニタ 管理者機能 Japanese					
コンポーネント一覧					
有効	エクスポート	内部名	種別	名前	説明
<input checked="" type="checkbox"/>	<input type="checkbox"/>	acis2enf	canvas	ACIS-インポート	ACISからENFへの変換
<input checked="" type="checkbox"/>	<input type="checkbox"/>	catiav42enf	canvas	CATIA V4インポート	CATIA V4からENFへの変換
<input checked="" type="checkbox"/>	<input type="checkbox"/>	catiav52enf	canvas	CATIA V5インポート	CATIA V5からENFへの変換
<input checked="" type="checkbox"/>	<input type="checkbox"/>	ideas2enf	canvas	NX I-deasインポート	NX I-deasからENFへの変換
<input checked="" type="checkbox"/>	<input type="checkbox"/>	iges2enf	canvas	IGES-インポート	IGESからENFへの変換
<input checked="" type="checkbox"/>	<input type="checkbox"/>	inv2enf	canvas	Inventor-インポート	InventorからENFへの変換
<input checked="" type="checkbox"/>	<input type="checkbox"/>	jt2enf	canvas	JTインポート	JTからENFへの変換

- コンポーネントの"内部名"をクリックするとコンポーネント詳細ページに移動します。コンポーネント詳細ページではサブコンポーネントおよびコンポーネントのパラメータの有効/無効を設定することができます。チェックボックスのチェックON/OFFを変更した時点で有効/無効の設定が保存されます。

Other Parameters

Enabled	Key	Name	Description	Value Type
<input checked="" type="checkbox"/>	CADVersion	CADバージョン指定	CADのバージョンを指定する	combobox
<input type="checkbox"/>	SlaveNodeLabel	スレーブノードのラベル指定	指定されたラベルを持つスレーブ...	text
<input type="checkbox"/>	ENFVersion	ENFバージョン	互換性のため出力するENFのバ...	pulldown
<input type="checkbox"/>	SlaveNodeName	スレーブノード指定	コンポーネントを実行するスレーブ...	text
<input checked="" type="checkbox"/>	Face	スキン変換	スキンを変換する	radiobutton
<input checked="" type="checkbox"/>	IsoCurve	単独カーブ変換	単独カーブを変換をする	radiobutton
<input type="checkbox"/>	FaceTransparency	透過度変換	フェースの透過度およびフィーチャ...	radiobutton
<input checked="" type="checkbox"/>	ConvertPartwise	省メモリ	省メモリモード	radiobutton
<input type="checkbox"/>	Axis	Axis変換	AxisSystemを変換する	radiobutton
<input checked="" type="checkbox"/>	XConvertSystemProperty	システム属性変換(Attribute)	システム定義属性を変換する	radiobutton
<input type="checkbox"/>	StructureMode	アセンブリ構造取得モード	アセンブリ構造取得モード	radiobutton

5.7.2. コンポーネントテンプレート作成・編集

シナリオ間で設定を共有するコンポーネントの情報を登録することができます。

- 「5.7.1, “コンポーネントおよびパラメータの有効/無効の設定”」に述べた方法でComponent Detail ページに移動し、[新規テンプレート]ボタンからコンポーネントテンプレートを登録します。

ASFALIS TransServer 新規ジョブ オペレーションモニタ 管理者機能 管理者					
コンポーネント詳細 - dek					
新規テンプレート					

ASFALIS TransServer 新規ジョブ オペレーションモニタ 管理者機能 管理者

コンポーネントテンプレート登録

テンプレート情報

テンプレート名 * convert 3D annotation

テンプレートの説明

パラメータ設定

- ☒ CADバージョン指定 (CADVersion) ☐ 設定する
- ☒ オカルス変換 (ConvertPartOccurrence) ☐ 設定する
- ☒ Note変換(Annotation) (XConvertNote) ☒ 設定する ON
- ☒ マテリアル属性変換(Attribute) (XConvertMaterial) ☒ 設定する ON
- ☒ ポリライン変換(Annotation) (XConvert3DAnnotationAsPolyline) ☒ 設定する ON

2. 登録したコンポーネントテンプレートはコンポーネント詳細ページのコンポーネントテンプレートに一覧表示されます。テンプレート名をクリックして内容を編集することができます。

ASFALIS TransServer 新規ジョブ オペレーションモニタ 管理者機能 管理者

コンポーネント詳細 - dek

新規テンプレート

コンポーネント情報

内部名
dek
名前
自動修正
説明
変換のため、三次元形状を自動で修正します。
種別
canvas

コンポーネントテンプレート

N/A

入出力パラメータ

キー	名前	入出力種別	値種別	MIME Type
inputfile	入力enfファイル	in	file	application/enf
outputfile	出力enfファイル	out	file	application/enf
logfile	テキストログファイル	log	textlog	
xmllogfile	XMLログファイル	log	xmllog	

その他のパラメータ

有効	キー	名前	説明	値種別
<input type="checkbox"/>	from_cad	変換元CAD	変換元のCADの種類を指定します。	radiobutton
<input checked="" type="checkbox"/>	to_cad	変換先CAD	変換先のCADの種類を指定します。	pulldown

3. シナリオ作成・編集の際、コンポーネントにコンポーネントテンプレートが存在する場合にはコンポーネントテンプレートを選択できるようになります。

コンポーネント設定

☒ すべて開く ☐ すべて閉じる

コンポーネントを追加

コンポーネント 1 (nx2enf) CADインポート NXインポート

コンポーネントを追加

テンプレートを選択してください
テンプレートを選択してください
convert 3D annotation



ヒント)

コンポーネントテンプレートを利用したシナリオが一度もジョブに使用されていない状態のとき

- コンポーネントテンプレートへのサブコンポーネントの追加および削除が可能です。
- シナリオで使用するコンポーネントテンプレートを変更することができます。

5.7.3. コンポーネントテンプレートのコピー

既存のテンプレートをコピーして新たにテンプレートを作成することができます。

1. 「5.7.1, “コンポーネントおよびパラメータの有効/無効の設定”」に述べた方法で“コンポーネント詳細”ページに移動します。
2. コンポーネントテンプレートの一覧に表示されているテンプレート中からコピーするテンプレート名を選択して“コンポーネントテンプレート編集”ページに移動します。



3. コンポーネントテンプレート編集"ページで [コピー作成] を選択します。



4. コピーされた内容を編集した上で [登録] を選択します。

5.8. サーバ設定

ASFALIS TransServer の動作を目的や要件に応じて変更するための各種設定を行うことができます。設定内容の詳細については、各設定と併せて表示される説明の内容を参照してください。

5.8.1. 処理状況の監視に関わる機能

ASFALIS TransServer では処理状況やボリューム容量を監視し、問題が発生した場合にメールで通知する機能があります。これらの機能については、必要に応じてサーバ設定から設定を変更することが

できます。

ボリューム容量の監視については、メールでの通知だけでなく操作画面にも警告メッセージを表示します。また、問題のある状態になった際には ASFALIS TransServer の処理を一部停止します。詳細は、「5.8.2, “リソースフォルダ、共有フォルダが属するボリュームの空き容量監視”」を参照してください。

監視対象	監視項目
リソースフォルダ・共有フォルダが属するボリューム	空き容量
ジョブ実行状況	失敗または強制終了
実行待ちのジョブ件数	実行待ちオーバーフロー
バックアップ	失敗
プロセスの実行状態 (システム監視)	プロセスの停止

5.8.2. リソースフォルダ、共有フォルダが属するボリュームの空き容量監視

リソースフォルダまたは共有フォルダが属するボリュームの空き容量が不足すると ASFALIS TransServer が正常に動作しなくなる可能性があります。このような状況を未然に防止するため、空き容量についてソフトリミットとハードリミットが設定されています。それぞれのリミットの詳細は以下の通りです。これらの項目の値を想定される使用状況に合わせて変更してください。

	メール通知	画面に 警告メッセージ	処理制限
ソフトリミット	○	○	-
ハードリミット	○	○	○

ソフトリミット

リソースフォルダまたは共有フォルダが属するボリュームの空き容量がこの値を下回ると、警告が管理者にメールで通知されるとともに ASFALIS TransServer の画面に警告メッセージが表示されます。ソフトリミットは以下の項目で指定することができます。

- リソースフォルダが属するボリュームの空き容量(既定値は 50GB) :
system.resource.soft_limit.size
- 共有フォルダが属するボリュームの空き容量(既定値は 50GB) :
system.shared_directory.soft_limit.size

ハードリミット

リソースフォルダまたは共有フォルダが属するボリュームの空き容量がこの値を下回ると、機能制限を通知するメールが管理者に送付されるとともに ASFALIS TransServer の画面に警告メッセージが表示されます。また各コンポーネントの処理が次に進まなくなり、ファイルセットの登録やジョブの投入も受け付けられなくなります。ハードリミットは以下の項目で指定することができます。

- リソースフォルダが属するボリュームの空き容量(既定値は 5GB) :

system.resource.hard_limit.size

- 共有フォルダが属するボリュームの空き容量(既定値は 5GB):

system.shared_directory.hard_limit.size

併せて web.login.hard_limit_authority.enabled を TRUE に設定すると、ボリュームの空き容量が上記の値を下回った時点で管理者以外のユーザは強制的にログアウトされます。

5.8.3. メール通知

メール通知機能を利用する場合には以下の項目を設定してください。これらの項目はサーバ設定の「カテゴリ」から「メール設定」を選択するとまとめて表示されます。

設定名	説明
scheduler.error_mail.sender	ASFALIS TransServer が送信するメールの送信者を指定します。(必須)
system.mail.enable_smtp_authentication	メール送信時に SMTP 認証を使用するか否かを設定します。使用する場合は "TRUE" を、使用しない場合は "FALSE" を指定します。(必須)
system.mail.enable_smtp_tls	メール送信時に SMTP over TLS を使用するかどうかを設定します。使用する場合は "TRUE" を、使用しない場合は "FALSE" を指定します。(必須)
system.mail.smtp_authentication_type	メール送信時に SMTP 認証を使用する場合は、認証の種類 (PLAIN、LOGIN、CRAM-MD5) を設定します。("system.mail.enable_smtp_authentication" に "TRUE" を指定した場合には必須)
system.mail.smtp_default_domain	メール送信時に使用するデフォルトのドメインを設定します。(任意)
system.mail.smtp_port	メール送信用の SMTP ポートを指定します。(必須)
system.mail.smtp_server	メール送信用の SMTP サーバを FQDN (完全修飾ドメイン名) で設定します。(必須)
system.mail.user_name	メール送信時に SMTP 認証で使用するユーザ名を設定します。("system.mail.enable_smtp_authentication" に "TRUE" を指定した場合には必須)
system.mail.user_password	メール送信時に SMTP 認証で使用するパスワードを設定します。("system.mail.enable_smtp_authentication" に "TRUE" を指定した場合には必須)

各種の問題が発生した場合にメールを受信したい場合には、以下の項目をそれぞれ設定してください。

- ジョブ実行に失敗した際にメールを受信したい場合

設定名	説明
scheduler.error_mail.body	ジョブ実行に失敗した際に送信するメールの本文を指定します。
scheduler.error_mail.recipient	ジョブ実行に失敗した際に送信するメールの受信者を指定します。

設定名	説明
scheduler.error_mail.subject	ジョブ実行に失敗した際に送信するメールの件名を指定します。

- 実行待ちオーバーフローについて警告メールを受信したい場合

設定名	説明
scheduler.overflow_mail.body_on_warning	実行待ちオーバーフロー警告メールの本文を指定します。
scheduler.overflow_mail.recipient	実行待ちオーバーフロー警告メールの受信者を指定します。
scheduler.overflow_mail.subject_on_warning	実行待ちオーバーフロー警告メールの件名を指定します。
scheduler.overflow_mail.body_on_resolved	実行待ちオーバーフロー状態の解消を通知するメールの本文を指定します。
scheduler.overflow_mail.subject_on_resolved	実行待ちオーバーフロー状態の解消を通知するメールの件名を指定します。
scheduler.overflow_mail.threshold	実行待ちのジョブがここで指定した件数以上見つかった場合に実行待ちオーバーフロー警告メールを送信します。

- ジョブが強制終了された際にメールを受信したい場合

設定名	説明
scheduler.terminate_mail.body	ジョブが強制終了された際に送信するメールの本文を指定します。
scheduler.terminate_mail.recipient	ジョブが強制終了された際に送信するメールの受信者を指定します。
scheduler.terminate_mail.subject	ジョブが強制終了された際に送信するメールの件名を指定します。

- バックアップに失敗した際にメールを受信したい場合

設定名	説明
system.backup.error_mail.body	バックアップに失敗した際に送信するメールの本文を指定します。
system.backup.error_mail.recipient	バックアップに失敗した際に送信するメールの受信者を指定します。
system.backup.error_mail.subject	バックアップに失敗した際に送信するメールの件名を指定します。

- システム監視で異常が発生した際にメールを受信したい場合

設定名	説明
system.monitor.error_mail_body	システム監視で異常が発生した際に送信するメールの本文を指定します。
system.monitor.error_mail_recipient	システム監視で異常が発生した際に送信するメールの受信者を指定します。
system.monitor.error_mail_subject	システム監視で異常が発生した際に送信するメールの件名を指定します。

システム監視では次のプロセスが停止した場合に異常が発生したと判断します。なお(*)の付いているプロセスについては、停止しても数秒後には再起動されるためあまり検知されることはありません。

- Apache
 - Nginx
 - Rails (*)
 - SlaveNode
 - MasterController
 - QueueController (*)
 - ScriptComponentWorker (*)
 - PerSlavenodeWorker (*)
- リソースフォルダが属するボリュームの空き容量がソフトリミット(*)を下回った際にメールを受信したい場合
 - *) ソフトリミットはsystem.resource.soft_limit.sizeで設定します。ソフトリミットの詳細については「[5.8.2, “リソースフォルダ、共有フォルダが属するボリュームの空き容量監視”](#)」を参照してください。
 - *) メールは system.monitor.error_mail_recipient で指定したアドレスに送信されます。

設定名	説明
system.resource.soft_limit.mail_body_on_warning	リソースフォルダが属するボリュームの空き容量がsystem.resource.soft_limit.size以下となった場合に、空き容量不足の警告を通知するメールの本文を指定します。
system.resource.soft_limit.mail_subject_on_warning	リソースフォルダが属するボリュームの空き容量がsystem.resource.soft_limit.size以下となった場合に、空き容量不足の警告を通知するメールの件名を指定します。
system.resource.soft_limit.mail_body_on_resolved	リソースフォルダが属するボリュームの空き容量がsystem.resource.soft_limit.sizeを上回った場合に、空き容量不足の解消を通知するメールの本文を指定します。
system.resource.soft_limit.mail_subject_on_resolved	リソースフォルダが属するボリュームの空き容量がsystem.resource.soft_limit.sizeを上回った場合に、空き容量不足の解消を通知するメールの件名を指定します。

- リソースフォルダが属するボリュームの空き容量がハードリミット(*)を下回った際にメールを受信したい場合

*) ハードリミットはsystem.resource.hard_limit.sizeで設定します。ハードリミットの詳細については「5.8.2, “リソースフォルダ、共有フォルダが属するボリュームの空き容量監視”」を参照してください。

*) メールは system.monitor.error_mail_recipient で指定したアドレスに送信されます。

設定名	説明
system.resource.hard_limit.mail_body_on_warning	リソースフォルダが属するボリュームの空き容量がsystem.resource.hard_limit.size以下となったことを通知するメールの本文を指定します。
system.resource.hard_limit.mail_subject_on_warning	リソースフォルダが属するボリュームの空き容量がsystem.resource.hard_limit.size以下となったことを通知するメールの件名を指定します。
system.resource.hard_limit.mail_body_on_resolved	リソースフォルダが属するボリュームの空き容量がsystem.resource.hard_limit.size以上になり、不足が解消したことを通知するメールの本文を指定します。
system.resource.hard_limit.mail_subject_on_resolved	リソースフォルダが属するボリュームの空き容量がsystem.resource.hard_limit.size以上になり、不足が解消したことを通知するメールの件名を指定します。

- 共有フォルダが属するボリュームの空き容量がソフトリミット(*)を下回った際にメールを受信したい場合

*) ソフトリミットはsystem.shared_directory.soft_limit.size で設定します。ソフトリミットの詳細については「5.8.2, “リソースフォルダ、共有フォルダが属するボリュームの空き容量監視”」を参照してください。

*) メールは system.monitor.error_mail_recipient で指定したアドレスに送信されます。

設定名	説明
system.shared_directory.soft_limit.mail_body_on_warning	共有フォルダが属するボリュームの空き容量がsystem.shared_directory.soft_limit.size以下となった場合に空き容量不足の警告を通知するメールの本文を指定します。
system.shared_directory.soft_limit.mail_subject_on_warning	共有フォルダが属するボリュームの空き容量がsystem.shared_directory.soft_limit.size以下となった場合に空き容量不足の警告を通知するメールの件名を指定します。
system.shared_directory.soft_limit.mail_body_on_resolved	共有フォルダが属するボリュームの空き容量がsystem.shared_directory.soft_limit.sizeを上回った場合に空き容量不足の解消を通知するメールの本文を指定します。
system.shared_directory.soft_limit.mail_subject_on_resolved	共有フォルダが属するボリュームの空き容量がsystem.shared_directory.soft_limit.sizeを上回った場合に空き容量不足の解消を通知するメールの件名を指定します。

- 共有フォルダが属するボリュームの空き容量がハードリミット(*)を下回った際にメールを受信したい場合

*) ハードリミットはsystem.shared_directory.hard_limit.size で設定します。ハードリミットの詳細については「5.8.2, “リソースフォルダ、共有フォルダが属するボリュームの空き容量監視”」を参照してください。

*) メールは system.monitor.error_mail_recipient で指定したアドレスに送信されます。

設定名	説明
system.shared_directory.hard_limit.mail_body_on_warning	共有フォルダが属するボリュームの空き容量がsystem.shared_directory.hard_limit.size以下となった場合に、機能の制限を通知するメールの本文を指定します。
system.shared_directory.hard_limit.mail_subject_on_warning	共有フォルダが属するボリュームの空き容量がsystem.shared_directory.hard_limit.size以下となった場合に、機能の制限を通知するメールの件名を指定します。
system.shared_directory.hard_limit.mail_body_on_resolved	共有フォルダが属するボリュームの空き容量がsystem.shared_directory.hard_limit.size以上になり、不足が解消したことを通知するメールの本文を指定します。
system.shared_directory.hard_limit.mail_subject_on_resolved	共有フォルダが属するボリュームの空き容量がsystem.shared_directory.hard_limit.size以上になり、不足が解消したことを通知するメールの件名を指定します。

5.9. 外部ファイルサーバ管理

ASFALIS TransServerでは新しいジョブを投入したり結果ファイルを取得したりする際、通常はウェブブラウザを経由して行います。しかし外部のファイルサーバに保存されたファイルからジョブを投入したり、結果ファイルを外部のファイルサーバに転送することもできます。本節では外部ファイルサーバの設定方法について説明します。

5.9.1. 外部ファイルサーバー一覧

[管理者機能] - [外部ファイルサーバ管理]を選択すると外部ファイルサーバの一覧画面が表示されます。

ASFALIS TransServer
新規ジョブ
オペレーションモニタ
管理者機能
管理者

外部ファイルサーバー一覧
新規

表示件数 10

ID	外部ファイルサーバ名	アドレス	プロトコル	使用するフォーマット	
1	Sample	\\sample\dir	Windowsファイル共有	ENFファイル、CATIA V5ファイル、NXファイル	⊗
2	Sample2	ftp://elysium.sample.co.jp	FTP	ENFファイル、IGESファイル、Parasolidファイル	⊗

ヒント)

以下のアイコンからサーバの情報を変更できます。

サーバを無効にする
⊗
取戻
⏪

5.9.2. 外部ファイルサーバ登録・編集

外部ファイルサーバー一覧ページで[新規]ボタンを選択すると外部ファイルサーバ登録ページが表示されます。それぞれの外部ファイルサーバ名を選択すると外部ファイルサーバの編集ページが表示されます。

The top screenshot shows the '外部ファイルサーバ登録' (Register External File Server) page. It has a header with 'ASFALIS TransServer' and navigation links. The main title is '外部ファイルサーバ登録' with a '接続テスト' (Test Connection) button. Below is a 'サーバ情報' (Server Information) section with fields for '外部ファイルサーバ名' (External File Server Name), '説明' (Description), and 'プロトコル' (Protocol) with a dropdown menu.

The bottom screenshot shows the '外部ファイルサーバ編集' (Edit External File Server) page. It has a similar header. The main title is '外部ファイルサーバ編集' with '接続テスト' and '削除' (Delete) buttons. Below is a 'サーバ情報' (Server Information) section with fields for '外部ファイルサーバ名', '説明', 'プロトコル', 'アドレス' (Address), and '文字コード' (Character Code). There are also instructions for each field.

利用可能な外部ファイルサーバの種類はWindowsファイル共有とFTPサーバです。



注意)

Windowsファイル共有を使用する場合、サーバプログラムの実行ユーザが追加の認証情報なしで使用できるフォルダを指定してください。



注意)

使用するフォーマットの指定を忘れないようにしてください。適切なフォーマットが指定されていないとファイル投入に使用することもファイル転送に使用することもできなくなります。

5.9.3. ジョブ投入・シナリオでの利用

すべてのユーザがジョブを投入する際、ファイルサーバが適切に設定されていれば利用することができます。シナリオで「Windowsファイルサーバへ転送」または「FTPサーバへ転送」コンポーネントを使用することで、結果ファイルを外部ファイルサーバに転送することができます。

5.10. 操作ログ一覧

シナリオ、ジョブ、ユーザ等に関する詳細情報を確認できます。

1. [管理者機能] - [操作ログ]メニューを選択します。
2. 操作種別、操作者、操作時間帯を設定し、[検索]を実行します。
3. 詳細を確認したい場合、🔍 アイコンをクリックします。

ASFALIS TransServer 新規ジョブ オペレーションモニタ 管理者機能 管理者

操作ログ一覧

操作種別: すべて

操作者: ユーザを選択してください

操作時間帯: ～から～まで YYYY/MM/DD hh:mm:ss から YYYY/MM/DD hh:mm:ss まで

検索 ダウンロード UTF-8

表示件数: 10

操作種別	データ	操作者	ホスト	操作日時
ログイン (108)		English	127.0.0.1	2015/10/07 09:55:53

5.11. サービスモニタ

Slave Nodeの状態やSlave Nodeで動作しているコンポーネントを確認することができます。またSlave Nodeの追加および削除を行うこともできます。

5.11.1. Slave Nodeの状態の確認

1. [管理者機能] - [サービスモニタ]メニューを選択します。
2. "ステータス"列でSlave Nodeの状態を確認します。

ASFALIS TransServer

サービスモニタ

変換サーバ

編集

ホスト:ポート	ステータス	
Server:59859	running	+

3. それぞれのSlave Nodeを選択して当該Slave Nodeで動作しているコンポーネントを確認します。



5.11.2. Slave Nodeの追加

1. [管理者機能] - [サービスモニタ]メニューを選択します。
2. [編集] を選択します。



3. [追加] を選択します。

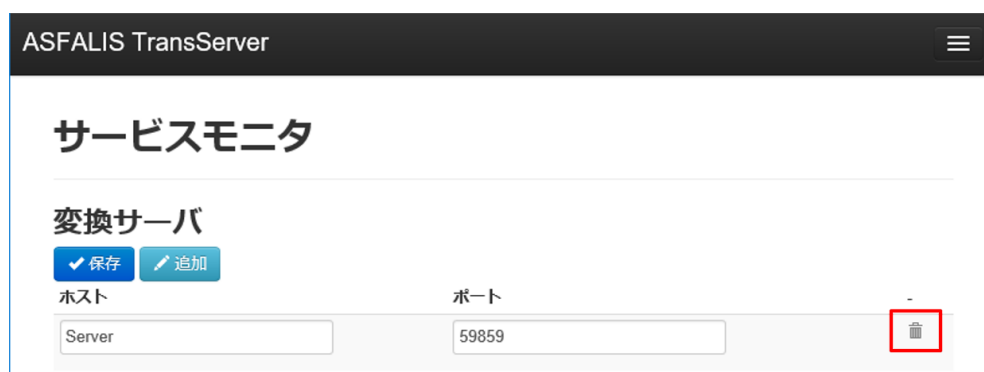


4. 追加するSlave Nodeのホストとポートを入力した上で [保存] を選択します。



5.11.3. Slave Nodeの削除

1. [管理者機能] - [サービスモニタ]メニューを選択します。
2. [編集] を選択します。
3. 削除するSlave Nodeの右に表示されているゴミ箱アイコンを選択します。



4. [保存] を選択します。



ヒント)

Slave Nodeの追加や削除は、上記の方法以外にスタートメニューの「Elysium TransServer」 > 「Slave Node設定」を選択した際に表示されるサーバ設定ダイアログからも実行することができます。詳細についてはインストールガイドをご参照ください。

5.12. サーバ情報

バージョン、データベース情報等を確認できます。

[管理者機能] - [サーバ情報]メニューを選択します。

サーバ情報

ASFALIS TransServer EX (Release 10.62)

インストールディレクトリ	C:\ELYSIUM\ASFALIS_TransServer\EX\k2dashboard
ポート	3000
K2Server バージョン	5.1.6
共有フォルダ	N:\share
ライセンスサーバ	TRUTH02-5093
最大クライアント数	10
APIライセンス	有効

データベース

アダプタタイプ	PostgreSQL
Host	localhost
Port	5432
Username	postgres
Database	trans_server_ex
Encoding	utf8
Charset	utf8
Reconnect	true
Pool	10
AllowConcurrency	true

Ruby

Ruby バージョン	2.1.6
Ruby パッチレベル	336
Ruby プラットフォーム	i386-mingw32
Rails バージョン	4.1.12

ASFALIS TransServer EX (Release 10.62)

6. バックアップ / リストア

ASFALIS TransServerを再インストールまたは更新する際は、あらかじめデータベースの情報をバックアップしておき、再インストールまたは更新後にリストアしてください。本章ではバックアップとリストアの手順を説明します。

6.1. データベースのバックアップ

6.1.1. 自動でバックアップファイルを作成する方法

定期的に自動でバックアップを取得する機能です。設定手順は以下のとおりです。

1. [管理者機能] - [サーバ設定]を開き以下の項目を設定してください。

system.backup.backup_directory にディレクトリを設定すると、自動バックアップが行われるようになります。

設定名	説明
system.backup.backup_directory	バックアップファイルを作成するディレクトリを指定します。この項目にパスが設定されると自動バックアップが行われるようになります。
system.backup.backup_duration	バックアップ間隔を秒で設定します。実施時刻で指定する場合はsystem.backup.backup_cronを設定し、本項目には空文字列を設定します。(*1)
system.backup.backup_cron	バックアップの実施時刻をcron形式で設定します。実施間隔で指定する場合はsystem.backup.backup_durationを設定し、本項目には空文字列を設定します。(*1)
system.backup.backup_target	バックアップの対象を指定します。 全てのファイルをバックアップ対象とする場合はAllを指定します。モデルやログを除く場合はExcept models and logs を指定します。
system.backup.backup_retention_count	バックアップの保持数を指定します。バックアップ完了後にこの値を超えた場合、古いバックアップが削除されます。0または負の数を指定した場合、バックアップを無制限に保持します。
system.backup.error_mail_recipient	バックアップに失敗した際に送信するメールの受信者を指定します。
system.backup.error_mail_subject	バックアップに失敗した際に送信するメールの件名を指定します。
system.backup.error_mail_body	バックアップに失敗した際に送信するメールの本文を指定します。

(*1) system.backup.backup_durationとsystem.backup.backup_cronどちらかの指定が必要

ヒント)

cron形式とは：

cron形式は4つのスペースで区切られた5つの日時欄から成ります。

バックアップ/古いファイルの削除は指定された日時ごとに実行されます。

日時欄は以下の通り指定します。



欄	可能な値
分	0 - 59
時間	0 - 23
日	1 - 31
月	1 - 12
曜日	0 - 6 (0は日曜日、6は土曜日)

アスタリスク(*)は可能なすべての値を意味します。ハイフンを用いて値の範囲を、カンマを用いて複数の値を指定することもできます。

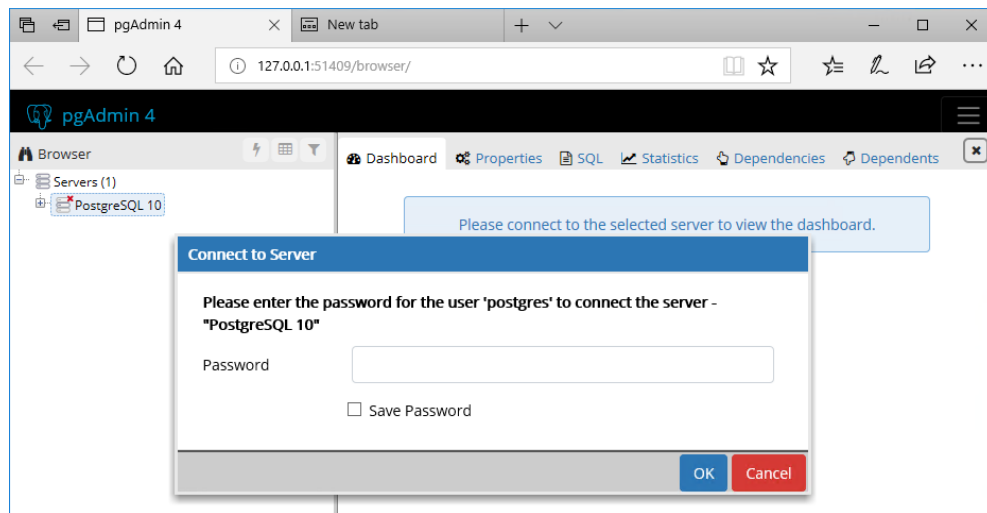
設定例)

cron形式	実行開始日時
30 5 * * *	毎日5:30
0 0 1 * *	毎月1日の0:00
0 0-12 * * 6	毎週土曜日の0時-12時に一時間おき
0,30 * * * *	毎時0分、30分

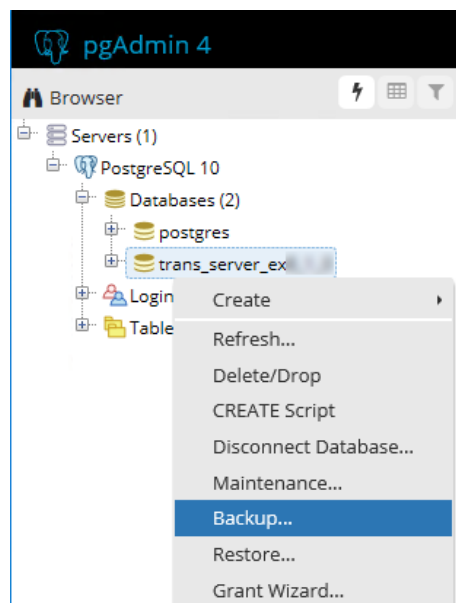
6.1.2. 手動でバックアップファイルを作成する方法

データベース、リストア用のkeyファイル、プライベートリソースフォルダをバックアップします。

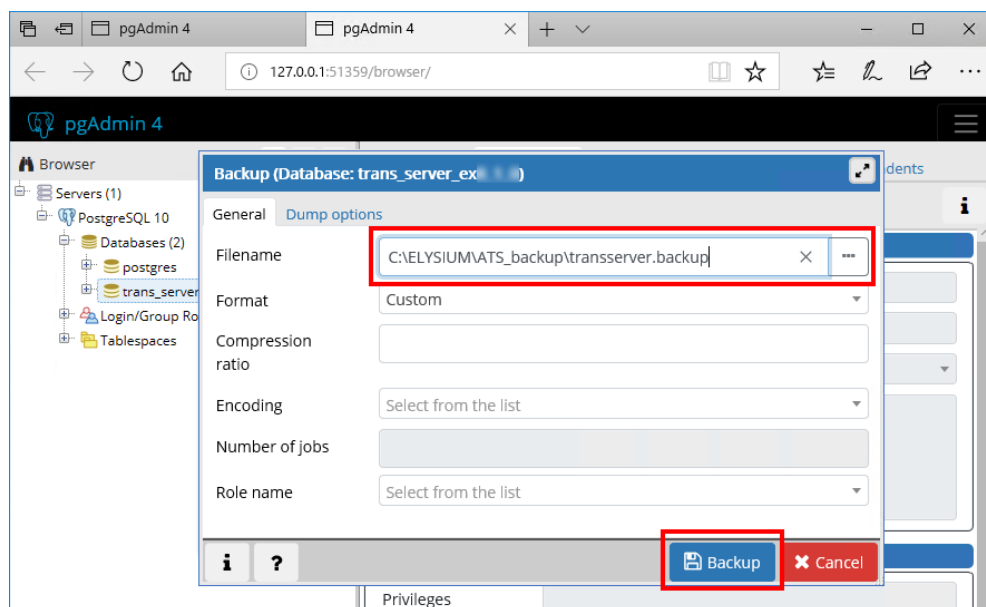
1. データベースを起動します。Windowsのスタートメニューから[PostgreSQL 10.*] - [pgAdmin 4]を選択します。
2. オブジェクトブラウザからPostgreSQL 10.* (初期設定の場合、localhost:5432)をダブルクリックし、出現したダイアログにパスワードを入力します。



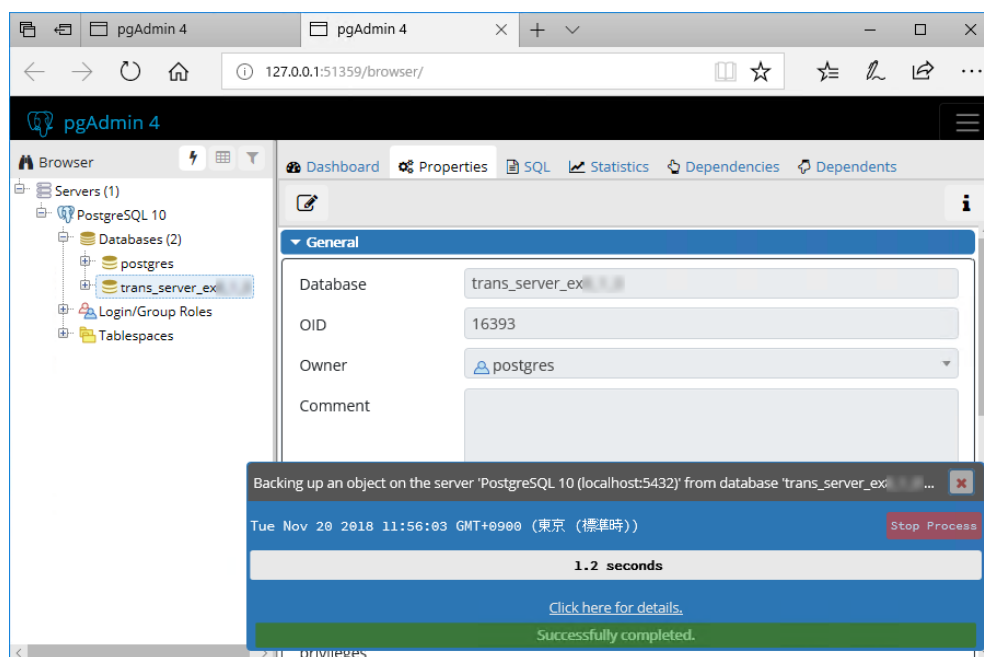
3. データベースを開き、作成したデータベース名を右クリックして[Backup]を選択します。



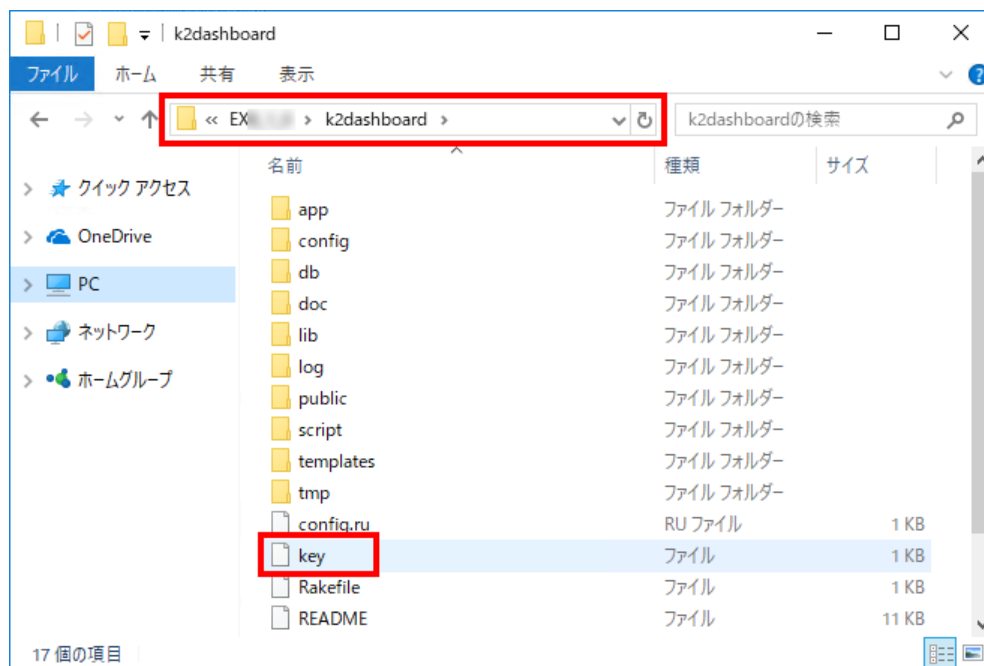
4. バックアップファイルの名前と保存場所を入力し[Backup]をクリックします。



5. バックアップ実行後、"Successfully completed." と表示されていることを確認してからダイアログ右上の[X]ボタンを押します。



6. < ASFALIS TransServer導入フォルダ>\k2dashboard の下にある、keyというファイルを他のフォルダにコピーしバックアップをとります。



7. 以下の場所にある設定ファイル(k2server.ini)を開きます。
< ASFALIS TransServer導入フォルダ>\config\k2server.ini
8. [K2server]セクションに記述されている"PrivateResourcePath"の指すフォルダのバックアップを取ります。

6.2. データベースのリストア

6.2.1. rakeコマンドでリストアする方法

半自動にてデータベースをリストアすることができます。以下の手順は「[6.1.1, “自動でバックアップファイルを作成する方法”](#)」に記載した手順にてバックアップを採取したことを前提としています。

1. 他のコンピューターにリストアする場合はバックアップ元のバックアップフォルダをリストアしたい環境へコピーします。
2. <ASFALIS TransServer導入フォルダ>\k2dashboard\start-console.bat を実行します。
3. 起動したコマンドプロンプトで以下のrakeコマンドを実行します。

```
bundle exec rake ats:restore ATS_BACKUP_FOLDER=<Path to ATS Backup Folder>
```

- <Path to ATS Backup Folder> : バックアップフォルダへの絶対パスを指定してください。
例 : \\path\to\ats_backup\20160705_023206_backup_d5c868f2

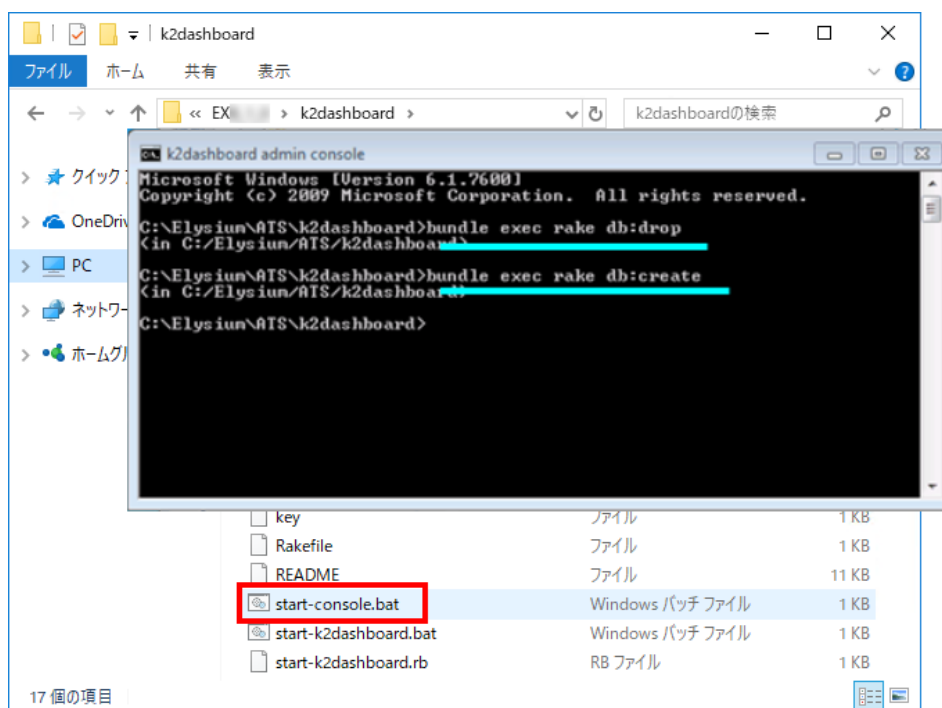
6.2.2. 手動でリストアする方法

手動でASFALIS TransServerデータベースのリストアを行う際は以下の点に注意してください。

- 新規導入環境にリストアする場合、まず現在のデータベースを初期化する必要があります。
- リストアすると現在のデータベースは見られなくなります。
- 元のマシンとは別のマシンへデータをリストアする場合、バックアップに含まれるプライベートリソースフォルダをリストア先のマシンに配置して、そのフォルダへのパスをk2server.iniの"PrivateResourcePath"に設定します。

リストア手順

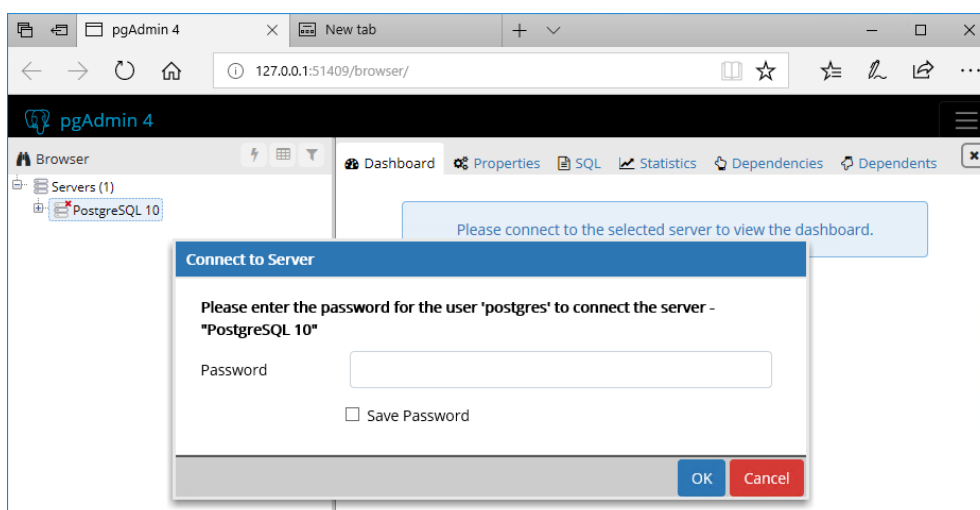
1. 最初に、リストアするデータベースを作成します。
 - i. <ASFALIS TransServer導入フォルダ>\k2dashboard\start-console.bat を実行します。
 - ii. "bundle exec rake db:drop"を実行します。
 - iii. "bundle exec rake db:create"を実行します。



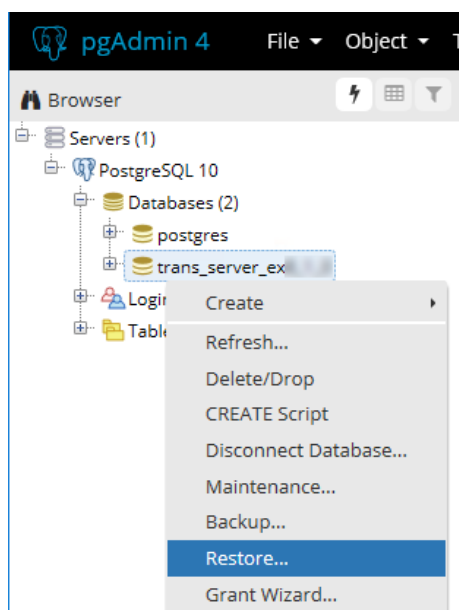
注意)

データベースを初期化するには、pgAdminを終了させておく必要があります。pgAdminが起動している場合、"db:drop"が実行できません。"db:drop"を実行した際にデータベースが見えなくなることを確認してください。

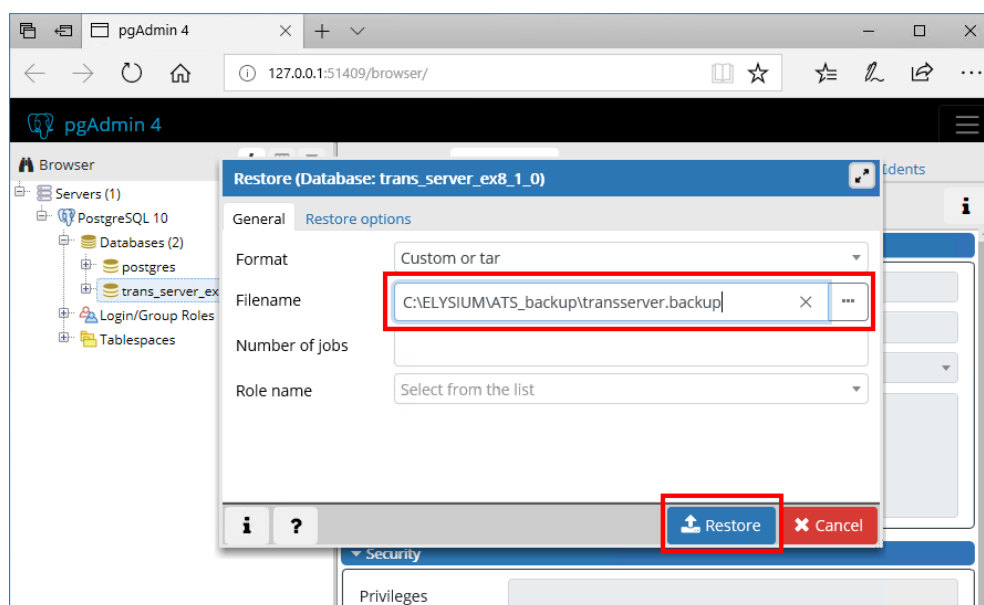
2. データベースを起動します。Windowsのスタートメニューから[PostgreSQL 10.*] - [pgAdmin 4]を選択します。
3. オブジェクトブラウザからPostgreSQL 10.* (初期設定の場合、localhost:5432)をダブルクリックし、出現したダイアログにパスワードを入力します。



4. データベースを開き、リストアップしたいデータベース名を右クリックし[Restore]を選択します。



5. バックアップファイルを選択し[Restore]をクリックします。



6. リストア実行後、"Successfully completed." と表示されていることを確認してからダイアログ右上の[X]ボタンを押します。



ヒント)

エラーが出力された場合はデータベースのリストアをキャンセルし、手順1の初期化が正しく行われていたか、再確認してください。

7. リストアしたデータベースを選択し、vaultsテーブルを開きます。
8. vaultsテーブルのデータを更新します。ats_version列がリストア対象の ASFALIS TransServer のバージョンになっている2つのレコードのpath列を以下のとおり変更し、上書き保存してください。
- 以下の例は、リストア対象の ASFALIS TransServer がEX8.1.0で、PrivateResourcePathを C:\Elysium\ASFALIS_TransServer\resource に設定した場合です。

is_protected列の値	path列に書き換える値
true	<PrivateResourcePathの設定値>\<ASFALIS TransServer のバージョン>\protected (例) C:\Elysium\ASFALIS_TransServer\resource\EX8.1.0\protected
false	<PrivateResourcePathの設定値>\<ASFALIS TransServer のバージョン>\unprotected (例) C:\Elysium\ASFALIS_TransServer\resource\EX8.1.0\unprotected



PrivateResourcePath にはローカルドライブのパスか UNC パスを指定するようにしてください。ネットワークドライブを含むパス (例 N:\transserver\Resources) を指定した場合、ASFALIS TransServer が正常に動作しない場合がありますので指定しないでください。

9. <ASFALIS TransServer導入フォルダ>\k2dashboard の下にあるkeyファイルを、バックアップしておいたkeyファイルと置換します。

7. 古いファイルの削除

ASFALIS TransServerでは、以前の変換結果等のファイルを自動的に削除し、ディスクの空き容量を増やすことが可能です。

7.1. ASFALIS TransServer導入マシン内のファイル

7.1.1. 削除対象ファイル

以下のファイルが削除対象となります。

- 入力ファイル
- 出力ファイル
- 各コンポーネントのテキストログファイル
- 各コンポーネントのXMLログファイル
- 各コンポーネントの出力ファイル

7.1.2. 設定手順

[管理者機能] - [サーバ設定]を開き、以下の項目を設定してください。

設定名	説明
system.resource.clean_up_duration	古いファイル削除の実施間隔を秒で設定します。実施時刻で指定する場合はsystem.resource.clean_up_cronを設定し、本項目には空文字列を設定します。(*1)
system.resource.clean_up_window_time	古いファイルを削除する際、実際にファイルを削除するまでの猶予期間を秒で設定します。
system.resource.limit_total_size	保持するファイルの合計サイズを設定します(単位: byte)。システム内に保持しているファイルが指定されたサイズを超えた場合、古いデータから削除します。
system.resource.min_lifetime	ファイルの最低保持期限を秒で設定します。この期間内のファイルはsystem.resource.limit_total_sizeで指定されたサイズを超えても削除対象になりません。また、この期間を過ぎてもsystem.resource.limit_total_sizeで指定されたサイズを超えなければ削除対象になりません。
system.resource.clean_up_cron	古いファイル削除の実施時刻をcron形式で設定します。実施間隔で指定する場合はsystem.resource.clean_up_durationを設定し、本項目には空文字列を設定します。(*1)

(*1) system.resource.clean_up_durationとsystem.resource.clean_up_cronのどちらかの指定が必要です。

システム内に保持しているファイルが、system.resource.limit_total_sizeで指定された容量を超えた場合に、ファイルが作成されてからsystem.resource.min_lifetimeで指定された時間を超えたファイルを削除候補とします。削除候補となったファイルは、system.resource.clean_up_window_timeで

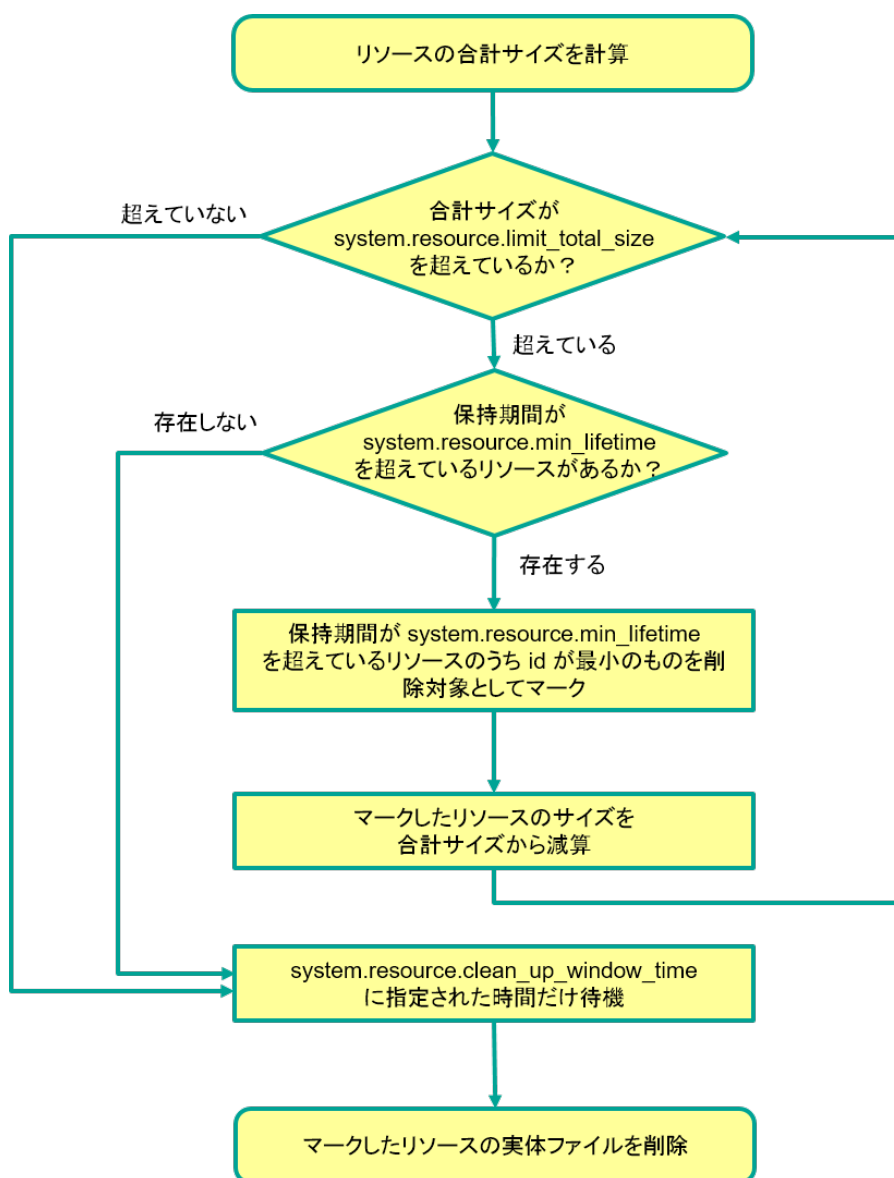
指定した時間後に削除されます。一連のファイル削除処理はsystem.resource.clean_up_durationで指定した間隔で、または system.resource.clean_up_cronで指定された日時 to 実行されます。



ヒント)

cron形式については、「[6.1.1, “自動でバックアップファイルを作成する方法”](#)」の「ヒント) cron形式とは」をご参照ください。

以下のフローチャートはファイル削除の流れを示したものです。ここでのリソースとは「[7.1.1, “削除対象ファイル”](#)」に記載した削除対象となるファイルの総称です。



また、ファイル削除はコマンドからも実行可能です。<ATS Install Directory>\k2dashboard\start-console.bat を実行してコンソールを起動し、以下のコマンドを実行します。

```
bundle exec rake ats:resources:cleanup
```

なお、JOB_ID_LIST オプションを使用することにより、指定した ID のジョブに関連するファイルだ

けを削除することもできます。コマンドの例は以下の通りです。

```
bundle exec rake ats:resources:cleanup JOB_ID_LIST="1,2,3"
```

この例では、ID が 1、2、3 のジョブに関連するファイルが削除されます。



このオプションを使用した場合、system.resource.clean_up_window_time で指定した期間が経過していても指定されたファイルは削除されます。

7.2. 共有フォルダ内のファイル

7.2.1. 削除対象ファイル

共有フォルダ内の作業フォルダが削除対象となります。

7.2.2. 設定手順

[管理者機能] - [サーバ設定]を開き、以下の項目を設定してください。

設定名	説明
scheduler.common.on_error_keep_directory	ジョブの処理中にエラーが起こった場合、作業用ディレクトリを保持するかどうかを設定します。エラーが起きたジョブの作業用ディレクトリの保持期間はscheduler.common.failed_directory_clean_up_durationに準じます。
scheduler.common.keep_directory	作業用ディレクトリを保持するかどうかを設定します。本設定がTRUEの時、常に作業用ディレクトリが保持され、正常終了のジョブ詳細画面からも作業用ディレクトリをダウンロードできます。正常終了したジョブの作業用ディレクトリの保持期間はscheduler.common.succeeded_directory_clean_up_durationに準じます。
scheduler.common.failed_directory_clean_up_duration	失敗したジョブの作業フォルダを削除する間隔を秒で指定します。
scheduler.common.succeeded_directory_clean_up_duration	成功したジョブの作業フォルダを削除する間隔を秒で指定します。

8. MIMEタイプと拡張子のカスタマイズ

ASFALIS TransServerでは、関連するCADなどのアプリケーション毎にファイルの種別をMIMEタイプとして管理し、コンポーネントの入力と出力が結び付けられるかを判定する際などに使用しています。カスタムコンポーネントで使用するMIMEタイプを作成したり、MIMEタイプと拡張子の関連をカスタマイズしたりすることができます。

8.1. MIMEタイプのカスタマイズ

Administratorsグループのユーザとしてログインしている状態で <ASFALIS TransServer のアドレス>/mime_types にアクセスすることでMIMEタイプの一覧を確認することができます。

ASFALIS TransServer 新規ジョブ オペレーションモニタ 管理者機能			
Mime-Type List			
新規			
Id	Name	Order	Is customized
1	application/enf	100	false
2	application/acis	110	false
3	application/catia4	140	false
4	application/catia5	150	false
5	application/icad	155	false
6	application/ideas	160	false

既存のMIMEタイプに対し拡張子の追加、変更を行う場合、各MIMEタイプ名をクリックします。

8.1.1. MIMEタイプの新規作成

右上の「新規」ボタンをクリックすると新規作成画面に移動します。Name と Order を設定し「登録」ボタンをクリックすると、新しいMIMEタイプが作成され、一覧画面に戻ります。

8.1.2. MIMEタイプの編集

各MIMEタイプ名のリンクをクリックすると編集画面に移動します。

Edit Mime-Type

Name

application/catia5

Order

150

更新

File-Type List

Id	Name	Regular Expression of Extension	Basic Regular Expression of Extension	Is dependent	
8	application/catia5/catpart	CATPart	CATPart	false	
9	application/catia5/catproduct	CATProduct	CATProduct	false	
10	application/catia5/cgr	cgr	cgr	true	
11	application/catia5/model	model	model	true	

新規

編集画面では、以下の作業を実行することができます。

- Nameの変更
- Orderの変更
- 関連する拡張子一覧の確認、削除
- 拡張子の作成/編集画面への遷移



注意

一度拡張子を削除すると、同じ拡張子を再度作成する以外に元に戻す方法はありませんのでご注意ください。

8.2. 拡張子のカスタマイズ

8.2.1. 拡張子の新規作成

MIMEタイプ編集画面右下の「新規」ボタンをクリックすることで、各MIMEタイプに関連付いた拡張子の新規作成画面に移動します。

ASFALIS TransServer
新規ジョブ
オペレーションモニタ
管理者機能
管理者

New File-Type

Name
Regular Expression of Extension
Basic Regular Expression of Extension
is_version_extension
☐
is_dependent
☐
select_child_ruby_code
header_test_ruby_code

Related File-Type List

ID	Name	Regular expression of extension	Basic regular expression	Is parent	Exclude as top
8	application/catia5/catpart	CATPart	CATPart	<input checked="" type="checkbox"/>	<input type="checkbox"/>
9	application/catia5/catproduct	CATProduct	CATProduct	<input checked="" type="checkbox"/>	<input type="checkbox"/>
10	application/catia5/cgr	cgr	cgr	<input type="checkbox"/>	<input type="checkbox"/>
11	application/catia5/model	model	model	<input type="checkbox"/>	<input type="checkbox"/>

新規作成画面では、以下を設定することができます。具体例は「[8.2.3, “拡張子の設定例”](#)」に記載があります。

- Name : 拡張子名
- Regular Expression of Extension : 拡張正規表現
- Basic Regular Expression of Extension : 基本正規表現
- Is_version_extension : 拡張子の後ろに付く数字が存在すればTRUE (ex. .prt.1)
- Is_dependent : この拡張子のファイルが必ずトップアセンブリにならない場合はTRUE(*1) の設定が可能(*2)

(関連付いているMIMEタイプの持つ拡張子に対して)

- Is parent : 編集中の拡張子に対して親になり得る場合TRUE
- Exclude as top : 親の拡張子のファイルがある時に、編集中の拡張子は必ずトップアセンブリにならない場合TRUE (*3)

(*1) MIMEタイプに関連付く拡張子のうち少なくとも1つはFALSEである必要があります。

(*2) select_child_ruby_code, header_test_ruby_code は変更しないでください。

(*3) Is parentがTRUEの時のみ指定可能です。

8.2.2. 拡張子の編集

MIMEタイプ編集画面の拡張子名のリンクをクリックすることで、各拡張子の編集画面に移動します。各項目は新規作成画面と同一です。

8.2.3. 拡張子の設定例

- i. Aは 子要素にA,B,Cを持つ ことができる
- ii. Bは 単体でパートとして機能 する
- iii. Cは Aに対して依存関係 にある
- iv. A,Cが共に存在するアセンブリでは Cはトップアセンブリにならない

という拡張子A,B,Cを作成する手順を以下に示します。括弧内のローマ数字は上記の下線部の条件に対応します。

1. 新規作成画面を開き、Aを作成する。全てのチェックボックスはオフにしておく。
2. Aの編集画面を開き、Aに対するIs parentにチェックを入れる(i)。
3. 新規作成画面を開き、Bを作成する。
is_dependent はチェック無し(ii)とし、Aに対するis_parentにチェックを入れる(i)。
4. 新規作成画面を開き、Cを作成する。
Is_dependentにチェックを入れ(iii)、Aに対するis_parent、Exclude as top にチェックを入れる(i, iv)。

手順は以上となります。

9. カスタマイズ機能

9.1. 概要

本章では、ASFALIS TransServerの管理者向け機能の一部として提供されているカスタマイズ機能について説明します。

用語の定義

- ネットワーク上のブラウザから接続を許可している URL、つまり ASFALIS TransServerの外部公開URL を <ATS Base URL> と表現します。
(例: <http://hostname:3000>)
- ASFALIS TransServerの導入ディレクトリを <ATS Install Directory> と表現します。
(例: C:\ELYSIUM\transserver)
- ASFALIS TransServerの設定ファイルサンプルが配置されているディレクトリを<ATS Sample Directory> と表現します。
(例: C:\ELYSIUM\transserver\k2dashboard\public\samples)

9.2. カスタムコンポーネント

任意の実行ファイルを、カスタムコンポーネントとしてASFALIS TransServerに組み込み、実行させることができます。カスタムコンポーネントを動作させるには以下の2点が必要です。

- Slave Node上で動作する実行ファイルをカスタムコンポーネントとして登録する
- ASFALIS TransServerにコンポーネントとそのパラメータ情報を登録する

カスタムコンポーネントに登録したパラメータは通常のコンポーネントのパラメータと同様に無効化することができます。パラメータの内容を変更したい場合には、登録した情報を編集する代わりに新たにパラメータを追加して不要になったパラメータを無効化してください。カスタムコンポーネント自体も通常のコンポーネントと同様に無効化することができます。

9.2.1. Slave Nodeで動作するカスタムコンポーネントの登録

Slave Nodeを動作させているマシンにカスタムコンポーネントを登録します。カスタムコンポーネントは、<カスタムコンポーネント名>.cmp ファイルを、以下のフォルダに配置することで利用できるようになります。

- <ASFALIS Component Install Directory>\data\custom\

サンプルは以下のフォルダにあります。

- <ATS Sample Directory>\custom_component

cmpファイルはiniファイル形式の設定ファイルです。下記の情報を記述します。

- Execute セクション

Executable	実行可能ファイルのパスを指定します。(必須)
Argument_(数字)	実行可能ファイルに渡す引数を指定します。引数の順番は0から始まる数字を使って指定します。(任意)
Cancel	ONを指定すると強制終了、タイムアウトが有効になります。強制終了やタイムアウトの際には実行中のプロセスに停止シグナルが送信されます。(任意)
Asfalismodule	ASFALISモジュールの配置フォルダを指定します。(任意, CancelがONの場合には必須)

- Output セクション

outputfile	指定されたファイルをカスタムコンポーネントの出力ファイルとして扱います。パラメータ名は変更可能ですがTransServerへ登録する情報と一致している必要があります。(outputfile/outputfile_filenameどちらかが必須)
outputfile_filename	出力が複数ファイルになる場合はこちらを指定します。トップアセンブリとしたいファイルを1行目に、その他のファイルを2行目以降に1行ずつ記述したテキストファイルをカスタムコンポーネントで出力します。ファイルはworkフォルダからの相対パスで指定します。テキスト内の文字列は、書かれた通りの文字列がファイルパスとして解釈されます。末尾のスペースやダブルクォーテーションが含まれていると、ファイルパスとして解釈できずファイルが見つからないエラーの原因となります。パラメータ名は変更可能ですがASFALIS TransServerへ登録する出力ファイルパラメータに_filenameを付けたものである必要があります。(outputfile/outputfile_filenameどちらかが必須)
reportfile_filename	レポートファイルの登録機能を使用する場合に指定します。詳しくは「 9.2.2, “ASFALIS TransServerへのパラメータ情報登録” 」を参照してください。(任意)
logfile	指定されたファイルをカスタムコンポーネントのログファイルとして扱います。パラメータ名は変更可能ですがTransServerへ登録する情報と一致している必要があります。(任意)

- \$(WORKSPACE) はSlave Node上でのワークフォルダのパスに展開されます。ASFALIS TransServerはコンポーネント実行毎にパスが一意となるワークフォルダを作成します。同フォルダは結果ファイルをシステムに受け渡しする以外に、一時的なファイルなどを置くために使用することができます。
- \$(ADC_INSTDIR) はSlave Nodeのインストールフォルダのパスに展開されます。
- {変数名} は該当するパラメータの情報が展開されます。入力ファイルやファイル型のパラメータの場合にはファイルパスに、それ以外のパラメータの場合には値を示す文字列の情報になります。
- 実行可能ファイルにはバッチファイル(.batファイル)を使用することができます。その場合、バッチファイル内でコマンドライン引数を参照する際には%~1の形式を使用してください。%1の形式を使用すると意図しないダブルクォーテーション等が含まれる場合があります。

9.2.2. ASFALIS TransServerへのパラメータ情報登録

カスタムコンポーネントのパラメータ情報はJSON形式のファイルとして作成してDBに登録します。ファイルはRFC4627準拠の形式で、文字コードはUTF-8を使用してください。

インポートするには <ATS Install Directory>\k2dashboard\start-console.bat を実行してコンソールを起動し、作成したファイルを引数として以下のコマンドを実行してください(改行は不要です)。

```
bundle exec rake ats:components:import
ATS_COMPONENTS_IMPORT_FILE=C:\path\to\components.json
```

ファイルのフォーマットおよびサンプルについては「[12.6.6, “カスタムコンポーネント”](#)」を参照してください。また、登録が終わったら必ずASFALIS TransServerを再起動してください。

9.2.3. 実行するSlave Nodeの指定

シナリオを定義する際に、併せてカスタムコンポーネントを実行するSlave Nodeを指定したい場合には、カスタムコンポーネントに対して前提となる設定を実施しておく必要があります。作成したJSONファイルに以下のオブジェクトを追記してください。通常のASFALISコンポーネントと同様に、SlaveNodeNameパラメータに対して直接Slave Nodeを指定して実行させることができます。SlaveNodeLabelパラメータでの指定はできません。

```
{
  "key_name": "SlaveNodeName",
  "name": {
    "en-US": "Specific SlaveNode",
    "ja-JP": "SlaveNode 指定"
  },
  "description": {
    "en-US": "Specify machine name and port number of the Slave Node where Component is executed; input \"(machine_name):(port_number)\"",
    "ja-JP": "コンポーネントを実行するSlave Nodeのマシン名とポート番号を指定します。\" (マシン名) : (ポート番号) \"を入力。"
  },
  "value_type": "text"
}
```

9.2.4. レポートファイルの ASFALIS TransServer への登録

ブラウザ上で表示可能な形式のファイル群(HTML、XML等)をカスタムコンポーネントで生成すれば、ASFALIS TransServerに登録することができます。登録したファイル群はジョブ詳細画面上で閲覧することができます。以降ではこのファイル群をレポートファイルと呼びます。

以下にレポートファイルを登録するための手順を示します。
なおサンプルは以下のフォルダにあります。

<ATS Sample Directory>\custom_component_reportfile

1. cmp ファイルの作成

最初にcmpファイルを作成し、Slave Node上に配置します。cmpファイルについては「[9.2.1, “Slave Nodeで動作するカスタムコンポーネントの登録”](#)」を参照してください。

cmpファイルの中で、レポートファイルとして登録したいファイル群を指定するファイル（以降report_file.txtと呼びます）へのパスをreportfile_filenameの値として指定します。report_file.txtの詳細については、b. を参照してください。

2. カスタムコンポーネントの設定

カスタムコンポーネントについては、report_file.txtで指定したファイル群をワークフォルダ内の指定されたパスに出力するように設定します。またcmpファイルのreportfile_filenameで指定したパスに対してカスタムコンポーネントがreport_file.txtを出力するように設定します。

report_file.txtには、ルートとなるファイルのパスを1行目に、それ以外のファイルのパスを2行目以降に1行ずつワークフォルダからの相対パスで記載してください。

レポートファイルとして登録したいファイル群を指定するファイル（report_file.txt）のサンプル

```
index.html
sample.css
sample.js
images/A1.png
images/A2.jpeg
```

*) このファイルに空行がある場合、レポートが正しく表示されません。

なおここではパラメータ名が "reportfile_filename" となっていますが、変更することができません。変更する場合には、後述のルールに従ってファイル名を指定してください。

- 前半部分：パラメータ情報ファイル(JSON形式)内のレポートに関連するオブジェクトに設定されている key_name の値を指定します。
- 後半部分：_filename を指定します。

3. ASFALIS TransServerへのパラメータ情報登録

JSON形式のファイルを用いてカスタムコンポーネントのパラメータ情報をDBに登録します。登録方法については「[9.2.2, “ASFALIS TransServerへのパラメータ情報登録”](#)」を参照してください。レポートファイルに関連するパラメータは以下の例に従ってオブジェクトを指定してください。上に記載したサンプルフォルダに、以下の内容を含むJSON形式のファイル(components.json)を配置しています。


```
{
  "key_name": "reportfile",
  "name": {
    "ja-JP": "レポートの名前をここに設定します",
    "en-US": " Replace this string with the name of the report"
  },
  "description": {
    "ja-JP": "レポートの説明をここに記載します",
    "en-US": " Replace this string with the description of the report "
  },
  "value_type": "markup_report",
  "mime_type": "application/octet-stream"
}
```



注意)

ユーザの入力に依存する文字列をレポートに埋め込む場合、適切な処理を行う必要があります。もし適切な処理を行っていない場合、悪意ある攻撃者が例えばモデルの属性などにスクリプトを埋め込むことにより、任意のユーザがASFALIS TransServer上でレポートを表示したタイミングでこのスクリプトが実行されます(XSS脆弱性)。スクリプトの内容によってはASFALIS TransServer内の情報にアクセスできてしまいます。ASFALIS TransServerでスクリプトの実行は停止されません。カスタムコンポーネントが、レポートに埋め込む文字列を安全な文字列にエスケープし、ブラウザによってスクリプトとして解釈されないようにしてください。

9.2.5. 任意の情報の ASFALIS TransServer への登録

任意の情報をJSON形式で出力することによりASFALIS TransServerに登録することができます。登録した情報は「9.6, “パラメータ中で展開される文字列”」の機能を利用することで後段のコンポーネントのパラメータに設定することができます。この機能を利用するための手順を以下に示します。

1. 実行ファイルの準備

登録したい情報をJSON形式のファイルとして共有フォルダ上のワークフォルダに書き出すように、実行ファイルに設定します。JSONの詳細については、以下のRFCを参照してください。

<https://www.ietf.org/rfc/rfc4627.txt>

JSONファイルは共有フォルダ上のワークフォルダに書き出されるように設定してください。

JSONファイルの例

```
{
  " ProcessResult ": "WARNING: some errors occurred."
}
```

上記のJSONファイルを出力した場合、後続のコンポーネントでは以下の様にして値を参照するこ

とができます。

設定例 : Process Result: #{component.custom_component1.properties["ProcessResult"]}

参照結果 : Process Result: WARNING: some errors occurred.

値を参照するための設定方法については「[9.6, “パラメータ中で展開される文字列”](#)」に詳しく記載しています。

2. cmp ファイルの作成とJSONファイルのパスの指定

準備した実行ファイルに関する情報を記述したcmpファイルを作成します。cmpファイルに記述する内容については「[9.2.1, “Slave Nodeで動作するカスタムコンポーネントの登録”](#)」を参照してください。その上でoutputセクションに実行ファイルが書き出すJSONファイルのパスを指定します。以下は、property.jsonという名称のJSONファイルをワークフォルダに出力する場合の設定例です。

```
propertyfile=$(WORKSPACE)\property.json
```

*) "propertyfile"の部分には任意の名称を設定することができます。

なおサンプルは以下のフォルダにあります。

<ATS Sample Directory>\custom_component_properties

上記のサンプルでは入力されたENFファイルをそのまま出力し、併せてコンポーネントパラメータに設定されている値をJSONファイルとして書き出します。

3. cmpファイルのSlave Nodeへの配置

作成したcmpファイルをSlave Node上に配置します。配置先は以下のフォルダになります。

<ASFALIS Component Install Directory>\data\custom

配置後に「ノード設定」からカスタムコンポーネントを"使用するコンポーネント"に登録した上で、Slave Nodeを再起動してください。

4. ASFALIS TransServerへのパラメータ情報登録

登録する情報に対応するパラメータの定義を含むJSONファイルを作成し、ASFALIS TransServerのDBに登録します。JSONファイルには以下の例の様に"log_parameters"セクションに設定を追加してください("..."の部分は前後の記載を省略しています)。登録方法については「[9.2.3, “実行するSlave Nodeの指定”](#)」を参照してください。

```

...
  "log_parameters": [
...
    {
      "key_name": "propertyfile",
      "name": {
        "ja-JP": "出力属性ファイル",
        "en-US": "output attribute file"
      },
      "description": {
        "ja-JP": "登録したい情報を格納したファイルです。",
        "en-US": " Output from Custom Component A, use as parameter for
Component B."
      },
      "value_type": "json",
    },
...
  ]
...

```

*) key_name" に指定する値はcmpファイルで設定した名称と一致している必要があります。

9.3. 規定のURL

ASFALIS TransServerが生成したJob IDを<Job ID>、Operation ID を<Operation ID>と表すと、オペレーションとジョブの情報を確認するページのURLは以下のようになります。

オペレーションモニタ : <ATS Base URL>/

オペレーション詳細 : <ATS Base URL>/job_lots/<OperationID>

ジョブ一覧 : <ATS Base URL>/jobs

ジョブ詳細 : <ATS Base URL>/jobs/<Job ID>

Operation ID はオペレーションモニタ、オペレーション詳細ページ等から、Job IDはオペレーション詳細、ジョブ一覧ページ等から取得することができます。

9.4. ヘルプファイルの変更

以下2つのサーバ設定項目で、一般ユーザ向けのヘルプファイル、管理者ユーザ向けのヘルプファイルを変更することができます。

- system.url.admin_tutorial : 管理者向けヘルプファイルを指定します。
- system.url.user_tutorial : 一般ユーザ向けヘルプファイルを設定します。

9.5. SSLの使用

ASFALIS TransServerでは外部公開URLに使用するプロトコルをHTTPSにすることができます。
設定ファイル <ATS Install Directory>\config\k2server.ini の K2dashboard セクションにあるSSL
をOnに設定すると、外部公開URLは下記のとおりになります。

[https://\(サーバ名\):\(SSLListenPortで指定したポート番号\)/](https://(サーバ名):(SSLListenPortで指定したポート番号)/)



注意)

<ATS Install Directory>\Apache(version)\Apache2以下に、k2server.iniの設定項目に合わせて鍵ファイルを配置する必要があります。設定変更にはASFALIS TransServerの再起動が必要です。

9.6. パラメータ中で展開される文字列

ASFALIS TransServerではジョブ実行の際、コンポーネントで使用するパラメータに特定の文字列を記入することで、パラメータの値を動的に設定することができます。これは例えば「ファイル変換後にメール送信コンポーネントを利用するシナリオでメールの本文に変換したファイル名を含めたい」場合などに有用な機能です。

以下の文字列を#{ と } ではさんでパラメータに記述します。なお、時間に関連するものについては文字列の組み合わせで様々な出力形式が指定できるため別表に記載しています。"コンポーネントインスタンス名"等を実際の文字列に置き換える際「"」(ダブルクォーテーション)は不要です。

component."コンポーネントインスタンス名".param."IO以外のパラメータ"	パラメータの値が展開されます。
component."コンポーネントインスタンス名".param."IOパラメータ"	コンポーネント入出力のファイル名が展開されます。
component."コンポーネントインスタンス名".param."IOパラメータ".strip_extension	コンポーネント入出力のファイル名から拡張子を取り除いたものが展開されます。
component."コンポーネントインスタンス名".status@"ロケール名"	コンポーネントのステータスが指定したロケールで展開されます。
component."コンポーネントインスタンス名".owner	コンポーネントが実行されたSlave Nodeが展開されます。
component."コンポーネントインスタンス名".properties.ExtendedErrors	コンポーネント実行時に出力された拡張エラーの情報がJSON形式で展開されます。 *拡張エラーの詳細についてはXMLログのマニュアルにある「Errors」の項を参照してください。
component."コンポーネントインスタンス名".properties.ExtendedErrors["拡張エラーに割り振られる連番"]["errorType"]	JSON形式で展開された拡張エラーに含まれる情報のうち、指定したJSONのオブジェクト表記に該当するものが展開されます。 *"errorType"は固定です。
component."コンポーネントインスタンス名".properties.ExtendedErrors.jsonpath(JSONPath形式のクエリ文字列)	JSON形式で展開された拡張エラーに含まれる情報のうち、指定したJSONPath形式のクエリに該当するものが展開されます。

component."コンポーネントインスタンス名".properties.ExtendedErrors.count	コンポーネント実行時に出力された拡張エラーの件数が展開されます。
component."コンポーネントインスタンス名".properties["rc"]	コンポーネント実行時に出力された拡張エラーが展開されます。
job.id	ジョブのIDが展開されます。
job.input_file_set	ジョブ入力のファイル名が展開されます。ジョブが複数ある場合は、最初のジョブの情報が出力されます。
job.input_file_set.strip_extension	ジョブ入力のファイル名から拡張子を取り除いたものが展開されます。ジョブが複数ある場合は、最初のジョブの情報が出力されます。
job.input_file_sets	すべてのジョブ入力のファイル名がカンマ区切りで展開されます。
job.input_file_sets.strip_extension	すべてのジョブ入力のファイル名から拡張子を取り除いたものがカンマ区切りで展開されます。
job.input_file_sets[n]	ジョブ入力配列のn番地(ジョブ入力1なら0番地 → job.input_file_sets[0])に該当するファイル名が展開されます。
job.input_file_sets[n].strip_extension	ジョブ入力配列のn番地(ジョブ入力1なら0番地 → job.input_file_sets[0])に該当するファイル名から拡張子を取り除いたものが展開されます。
job.uploaded_filesize_sum	ジョブ入力のアップロード時のファイルサイズの合計値が展開されます(単位: byte)。
job.extracted_filesize_sum	ジョブ入力の展開後のファイルサイズの合計値が展開されます(単位: byte)。
job.unique_name	ジョブの識別名が展開されます。
job.description	ジョブのメモが展開されます。
job.priority@"ロケール名"	ジョブの優先度が指定したロケールで展開されます。
job.is_nightly	ジョブの夜間フラグがONであればtrueが、OFFであればfalseが展開されます。
job.is_erased	ジョブが無効化されていればtrueが、そうでなければfalseが展開されます。
job.login_name	ジョブを投入したユーザのログイン名が展開されます。
job.create_user.login_name	ジョブを投入したユーザのログイン名が展開されます。
job.create_user.display_name@"ロケール名"	ジョブを投入したユーザの表示名が指定したロケールで展開されます。
job.url_to_download_results	ジョブの結果ファイルを一括でダウンロードするためのURLが展開されます。
job_lot.id	オペレーションのIDが展開されます。
job_lot.unique_name	オペレーションの識別名が展開されます。
job_lot.name	オペレーション名が展開されます。

job_lot.login_name	オペレーション作成者のログイン名が展開されます。
job_lot.create_user.login_name	オペレーション作成者のログイン名が展開されます。
job_lot.create_user.display_name @"ロケール名"	オペレーション作成者の表示名が指定したロケールで展開されます。
job_lot.property."オペレーション 属性のキー"	オペレーション属性の値が展開されます。
scenario.id	シナリオのIDが展開されます。
scenario.name@"ロケール名"	指定したロケールでのシナリオ名が展開されます。
scenario.unique_name	シナリオの識別名が展開されます。
serverconfig."サーバ設定名"	サーバ設定の値が展開されます。
job_queue.waiting_job_count	キューに残っているジョブ数が展開されます。

「コンポーネントインスタンス名」にはシナリオ編集やジョブ詳細ページに現れる、シナリオ内でコンポーネントを特定する名前を記述します。例): parasolid2enf

各コンポーネントのパラメータ名はシナリオ登録・編集画面で確認できるほか、<ATS Base URL>/components/のページから情報を一覧することができます。オペレーション属性のキーには業務種別管理で登録した属性のキーを指定します。ロケール名には en-US、ja-JP、de-DE、zh-CN のいずれかを指定します。

注)

コンポーネントの出力ファイルパラメータは、そのコンポーネント以降のコンポーネントでのみ利用可能です。これは、出力ファイルがそのコンポーネントの実行完了時点で決定するためです。またファイル型パラメータに指定されたファイルについては、そのファイルの実体もワークフォルダにコピーされます。

例えば customcomponent→mail というシナリオにおいて

```
#\{component.customcomponent.param.outputfile}
```

という記法は、mailコンポーネントのパラメータに対しては利用可能ですが、customcomponentコンポーネントのパラメータに対しては利用できません。

注)

「別のシナリオを実行」コンポーネントを利用する場合に、呼ばれる側のシナリオ(子シナリオ)から呼んだ側のシナリオ(親シナリオ)で設定されるパラメータを参照したい場合には、以下のように設定してください。+ parent_job以下の指定方法は、job以下の指定方法と同じです。

```
#\{job.parent_job....}
```

使用例(1)：親シナリオのジョブIDを子シナリオで参照したい場合

```
#{job.parent_job.id}
```

使用例(2)：親シナリオで実行した形状比較の実行結果を子シナリオで参照したい場合

```
#{job.parent_job.component.geomdiff.properties["rc"]}
```

注)

ファイルサーバを指定するパラメータ（例:「Windowsファイルサーバへ転送」コンポーネントの「ファイルコピー先(result_base_directory)」パラメータ）が展開される場合、選択したファイルサーバとディレクトリを接続したパスが返却されます。例えば、\\fileserver\address というアドレスで登録されたファイルサーバ以下に selected\directory というサブディレクトリの構造がある場合、このパラメータの設定時に選択したディレクトリによって、展開されるパスが変わります。

```
#{component.copy_to_cifs.param.result_base_directory}
-> \\fileserver\address (ファイルサーバのルートディレクトリが選択されている場合)
-> \\fileserver\address\selected\directory\.
(サブディレクトリ以下が選択されている場合)
```

コンポーネントインスタンスの時間に関連するパラメータについては以下の規則で指定することができます。

```
component."コンポーネントインスタンス名".<タイミング識別子>(<タイムゾーン>).format(
<形式>)
```

*) ".format(<形式>)"は省略することができます。

関数等の詳細は以下の通りです。

<タイミング識別子>

started_at	処理開始日時を指定します。
ended_at	処理終了日時を指定します。

<タイムゾーン>

指定なし	日時(UTC)が %Y%m%d-%H%M%S の形式で展開されます。
地域名	指定したタイムゾーンにおける日時が %Y/%m/%d %H:%M:%S (地域名) の形式で展開されます。
localtime	ASFALIS TransServer 実行環境のタイムゾーンにおける日時が %Y/%m/%d %H:%M:%S の形式で展開されます。

<形式>

path	タイミング識別子の出力が %Y%m%d-%H%M%S の形式で展開されます。
pretty_print	タイミング識別子の出力が %Y/%m/%d %H:%M:%S の形式で展開されます。

使用例 : nx2enfのコンポーネントインスタンスが日本時間2016/10/1 19:00:15に開始し、日本時間2016/10/1 20:00:30 に終了した場合

```
component.nx2enf.started_at(localtime) -> 2016/10/01 19:00:15
component.nx2enf.started_at(localtime).format(path) -> 20161001-190015
component.nx2enf.ended_at -> 20161001-110030
component.nx2enf.ended_at(Tokyo).format(pretty_print) -> 2016/10/01 20:00:30
(Tokyo)
```

ジョブの時間に関連するパラメータについては以下の規則で指定することができます。

```
job. <タイミング識別子>(<タイムゾーン>).format(<形式>)
```

*) ".format(<形式>)"は省略することができます。

関数等の詳細は以下の通りです。

<タイミング識別子>

created_at	ジョブの作成日時を指定します。
updated_at	ジョブの最終更新日時を指定します。
started_at	ジョブの開始日時を指定します。

<タイムゾーン>

指定なし	日時(UTC)が %Y%m%d-%H%M%S の形式で展開されます。
地域名	指定したタイムゾーンにおける日時が %Y/%m/%d %H:%M:%S (地域名) の形式で展開されます。
localtime	ASFALIS TransServer 実行環境のタイムゾーンにおける日時が %Y/%m/%d %H:%M:%S の形式で展開されます。

<形式>

path	タイミング識別子の出力が %Y%m%d-%H%M%S の形式で展開されます。
pretty_print	タイミング識別子の出力が %Y/%m/%d %H:%M:%S の形式で展開されます。

使用例) 2016/10/31 21:00:45に作成されたジョブが2016/10/31 21:05:10に更新され、2016/10/31 21:15:00に開始されて終了した場合(すべて日本時間)


```

job.created_at(localtime) -> 2016/10/31 21:00:45
job.updated_at(localtime).format(path) -> 20161031-210510
job.started_at -> 20161031-121500

```

なおこれらの文字列はジョブの実行中に展開されるため、ジョブの終了時刻は取得することができません。代わりに特定のコンポーネントの終了時刻をジョブの終了時刻とみなす場合には下記の通り指定してください。

```
component.(component instance name).ended_at
```

9.7. ASFALIS TransServerのチューニング

9.7.1. 概要

ほとんどの場合ASFALIS TransServer はデフォルトの設定で使うことができますが、多数のユーザが同時に使用したり、多数のジョブやファイルを登録したりするとパフォーマンスが低下する場合があります。このようなとき、快適にASFALIS TransServer を利用するために設定を変更する方法について記述します。

9.7.2. DBのチューニング

ASFALIS TransServerではPostgreSQLを使用しています。データ量が多く、負荷の高い環境ではPostgreSQLの設定を変更することでパフォーマンスを改善できる場合があります。

- 設定箇所
postgresql.conf (PostgreSQLの設定ファイル)
- 設定内容

項目名	デフォルト値	指定する値の例
shared_buffers	32MB	512MB
temp_buffers	8MB	80MB
work_mem	1MB	32MB
maintenance_work_mem	16MB	256MB
wal_buffers	64kB	512kB
effective_cache_size	128MB	1024MB
deadlock_timeout	1s	10s

内容の詳細についてはPostgreSQLのドキュメントを参照してください。

postgresql.conf の内容を変更した場合にはコントロールパネルのサービス一覧からPostgreSQLの

サービスを選択し、再起動を行ってください。

- サービス名の例 : postgresql-x64-10 - PostgreSQL Server 10

9.7.3. アプリケーションサーバ管理設定のチューニング

ASFALIS TransServerに対する同時アクセス数が多いことでレスポンス性能が低下している場合、アプリケーションサーバの起動数の調整によりパフォーマンスを改善できる場合があります。

- **設定箇所**
ファイル : <ATS Install Directory>\config\k2server.ini
K2dashboard セクションの BalancerMemberNumの値
- **設定内容**
想定する同時アクセス数の3 倍程度を目安に値を変更します。

k2server.ini編集後、ASFALIS TransServer を再起動してください。

9.7.4. DBの最大同時接続数の設定

「9.7.3, “アプリケーションサーバ管理設定のチューニング”」で指定するアプリケーションサーバの起動数や、ASFALIS TransServerが使用するSlave Nodeの数に応じてPostgreSQLの最大同時接続数を調整する必要があります。設定方法については「9.7.2, “DBのチューニング”」を参照してください。

- **設定箇所**
postgresql.confのmax_connectionsパラメータ
- **設定内容**
A : アプリケーションサーバの起動数（デフォルト値は3）
B : 使用するSlave Node数
として、最大同時接続数には少なくとも100または $(10 * A) + (5 * B) + 50$ のうち大きい方の値を設定します。

9.8. LDAP認証連携

9.8.1. 概要

ASFALIS TransServerはユーザ認証のためにLDAPサーバと連携することができます。LDAPサーバにユーザのパスワード認証を委譲することができます。LDAP認証連携機能を使用するには、下記の項目が必要です。

- 認証の委譲を受け付ける適切なLDAPサーバ
- ユーザ情報を検索することのできるアカウント

9.8.2. LDAP認証連携を使用する

LDAP認証連携機能を使用するにはサーバ設定 web.login.sso_ldap.enabled をTRUEに設定し、併せてユーザ編集画面から“LDAP連携を有効にする”というチェックボックスを選択して更新します。この

チェックボックスはサーバ設定 `web.login.sso_ldap.enabled` が TRUE のときのみ現れます。詳細は下記のサーバ設定項目を参照してください。

- `web.login.sso_ldap.enabled`
- `web.login.sso_ldap.search_account_name_type`
- `web.login.sso_ldap.connection_user_dn`
- `web.login.sso_ldap.connection_user_password`
- `web.login.sso_ldap.server_host`
- `web.login.sso_ldap.server_port`
- `web.login.sso_ldap.search_base_dn`

9.9. パラメータの拡張機能

9.9.1. 概要

任意パラメータのキー毎に入力欄を設けたり、あらかじめ指定した項目に基づく選択肢を設けたりするなど、コンポーネントパラメータの指定方法をカスタマイズすることができます。パラメータのテキスト入力が多いシナリオで、入力ミスを防ぎたい場合などに有用な機能です。

9.9.2. 設定方法

コンポーネントパラメータの「パラメータの指定方法」から「拡張機能」(*)を選択すると、「拡張機能の設定」というテキストエリアが表示されます。次に示すフォーマットにしたがって、JSON形式 (RFC4627準拠) で設定値を指定してください。

*内部的にはパラメータの指定方法が「あとでジョブ投入時に指定する」となっているものとして扱われます。

9.9.3. フォーマット

階層	ノード名	説明	型	必須
1	<code>format</code>	-	object	○
2	<code>type</code>	<ul style="list-style-type: none"> • <code>multiple</code>: 任意パラメータのようにキーと値で複数指定できるパラメータ。(multi も可) • <code>single</code>: それ以外のパラメータ 	string	○
2	<code>parameters</code>	各キーのパラメータ設定の拡張に用いる情報を持つオブジェクトを配列で指定します。(type が <code>multiple</code> または <code>multi</code> の場合)	array of object	*1
3	<code>(parameter)</code>	-	object	-
4	<code>key</code>	各パラメータのキーを指定します。	string	-

階層	ノード名	説明	型	必須
4	type	<ul style="list-style-type: none"> • select: プルダウン形式 • radio: ラジオボタン形式 • text: テキストボックス形式 	string	○
4	value	<p>ジョブ作成画面におけるこのキーのパラメータの初期値を指定します。</p> <ul style="list-style-type: none"> • type が select, radio の場合 value と同値の項目が options に存在する場合は、その項目が選択された状態になります。value と同値の項目が options に存在しない場合、または value の指定がない場合は、一番最初の項目が選択された状態になります。 • type が text の場合 value に指定した値がテキストボックスに反映されます。 	string	-
4	options	選択肢を配列で指定します。type が select の場合のみ適用されます。	array of string	*3
5	(option)	選択肢の値のみを指定する場合は、直接 string で指定します。	string or object	*3
6	value	選択肢の値を指定します。	string	*3
6	description	<p>選択肢の説明文を指定します。</p> <ul style="list-style-type: none"> • type が select の場合 空文字列を指定した場合、または指定を省略した場合、value に該当する選択肢があればその選択肢の説明文が適用されます。 • type が radio の場合 指定されている場合、空文字列であってもその値で既存の選択肢の説明文が上書きされます。 指定を省略した場合、value に該当する選択肢があれば、その選択肢の説明文が適用されます。 	string	-
2	parameter	パラメータ設定の拡張に用いる情報を持つオブジェクトを指定します。key は指定しません。(type に single を指定した場合)	object	*2
3	type	(parameter) の type と同じ	string	○

階層	ノード名	説明	型	必須
3	value	(parameter) の value と同じ	string	-
3	options	(parameter) の options と同じ	array of string	*3
4	(option)	(option)と同じ	string or object	*3
5	value	(option) の value と同じ	string	*3
5	description	(option) の description と同じ	string	-

*1 format の type が multiple の場合に必須

*2 format の type が single の場合に必須

*3 parameter の type が select, radio の場合に必須

9.9.4. 設定例

- サンプル1

enfeditor コンポーネントの custom_parameters について、各キーの設定方法を以下のようにカスタマイズする。

- use_asmedit
 - 入力形式: プルダウン
 - 初期値: 0
 - 選択肢: 0,1
- PartialAssemblyExtraction
 - 入力形式: ラジオボタン
 - 初期値: 1
 - 選択肢: 0,1
- ExtractByRegularExpression
 - 入力形式: プルダウン
 - 初期値: 0
(value の指定がないので一番最初の選択肢が初期値となる)
 - 選択肢: 0,1
- extraction_target
 - 入力形式: テキスト
 - 初期値: \\path\to\extractlist.xml

```
{
  "format": {
    "type": "multiple",
    "parameters": [
      {
        "key": "use_asmedit",
        "type": "select",
        "value": "0",
        "options": [
          "0",
          "1"
        ]
      },
      {
        "key": "PartialAssemblyExtraction",
        "type": "radio",
        "value": "1",
        "options": [
          "0",
          "1"
        ]
      },
      {
        "key": "ExtractByRegularExpression",
        "type": "select",
        "options": [
          "0",
          "1"
        ]
      },
      {
        "key": "extraction_target",
        "type": "text",
        "value": "\\path\\to\\extractlist.xml"
      }
    ]
  }
}
```

- サンプル2

copy_to_cifs の allow_overwrite について、設定方法を以下のようにカスタマイズする。

- 入力形式: ラジオボタン
- 初期値: 3
- 選択肢

設定値	選択肢のラベル	備考
0	0: 上書きしない・警告なし（変更済み）	*1
1	1:	*2
2	2: 上書きを許可する。上書きを行っても警告を付与しない。	*3
3	3: 上書きを許可する。上書きを行った場合は警告を付与する。	*3
4	4:	*4

*1 description 指定時、指定した内容がラベルが上書きされる。

*2 description 指定時、指定した内容が空文字列の場合もラベルが上書きされる。

*3 description 省略時、設定値に該当する選択肢があれば、その選択肢の説明文がラベルに反映される。

*4 description 省略時、設定値に該当する選択肢がなければ、設定値がラベルに反映される。

```
{
  "format": {
    "type": "single",
    "parameter": {
      "type": "radio",
      "value": "3",
      "options": [
        {
          "value": "0",
          "description": "上書きしない・警告なし（変更済み）"
        },
        {
          "value": "1",
          "description": ""
        },
        {
          "value": "2"
        },
        "3",
        "4"
      ]
    }
  }
}
```

• サンプル3

copy_to_cifs の allow_overwrite について、設定方法を以下のようにカスタマイズする。

- 入力形式: プルダウン形式
- 初期値: 3

。選択肢

設定値	選択肢のラベル	備考
0	上書きしない・警告なし（変更済み）	*1
1	上書きを行わない。ただし、同名のファイルが既に存在する場合は警告を付与する。	*2
2	上書きを許可する。上書きを行っても警告を付与しない。	*2
3	上書きを許可する。上書きを行った場合は警告を付与する。	*2
4	4	*3

*1 description 指定時、指定した内容でラベルが上書きされる。

*2 description が空文字列または省略時、設定値に該当する選択肢があれば、その選択肢の説明文がラベルに反映される。

*3 description が空文字列または省略時、設定値に該当する選択肢がなければ、設定値がラベルに反映される。

```
{
  "format": {
    "type": "single",
    "parameter": {
      "type": "select",
      "value": "3",
      "options": [
        {
          "value": "0",
          "description": "上書きしない・警告なし（変更済み）"
        },
        {
          "value": "1",
          "description": ""
        },
        {
          "value": "2"
        },
        "3",
        "4"
      ]
    }
  }
}
```


9.9.5. サポートされているパラメータ型

パラメータの拡張機能は、以下のパラメータ型に対して有効です。設定値に制約のあるパラメータについては、設定の仕方を併記しています。

- number
- text
- textarea
 - 指定する type によってパラメータの反映のされ方が変わります。
 - multiple (または multi) 指定時は、各キーが「キー=値」の形式に変換されてパラメータに反映されます。
 - single 指定時は、指定または選択した値がそのままパラメータに反映されます。
- radiobutton
 - 「コンポーネント詳細 > コンポーネントパラメータ設定 > パラメータの選択肢」の「パラメータの値」に記載されている値が設定値となるようにしてください。
- boolean
 - "0" または "1" が設定値となるようにしてください。
- bitflag
 - 「コンポーネント詳細 > コンポーネントパラメータ設定 > パラメータの選択肢」の「パラメータの値」に記載されている値の総和が設定値となるようにしてください。
- combobox
 - 「コンポーネント詳細 > コンポーネントパラメータ設定 > パラメータの選択肢」の「パラメータの値」に記載されている値が設定値となるようにしてください。
- pulldown
 - 「コンポーネント詳細 > コンポーネントパラメータ設定 > パラメータの選択肢」の「パラメータの値」に記載されている値が設定値となるようにしてください。
- scenario
 - シナリオの ID が設定値となるようにしてください。

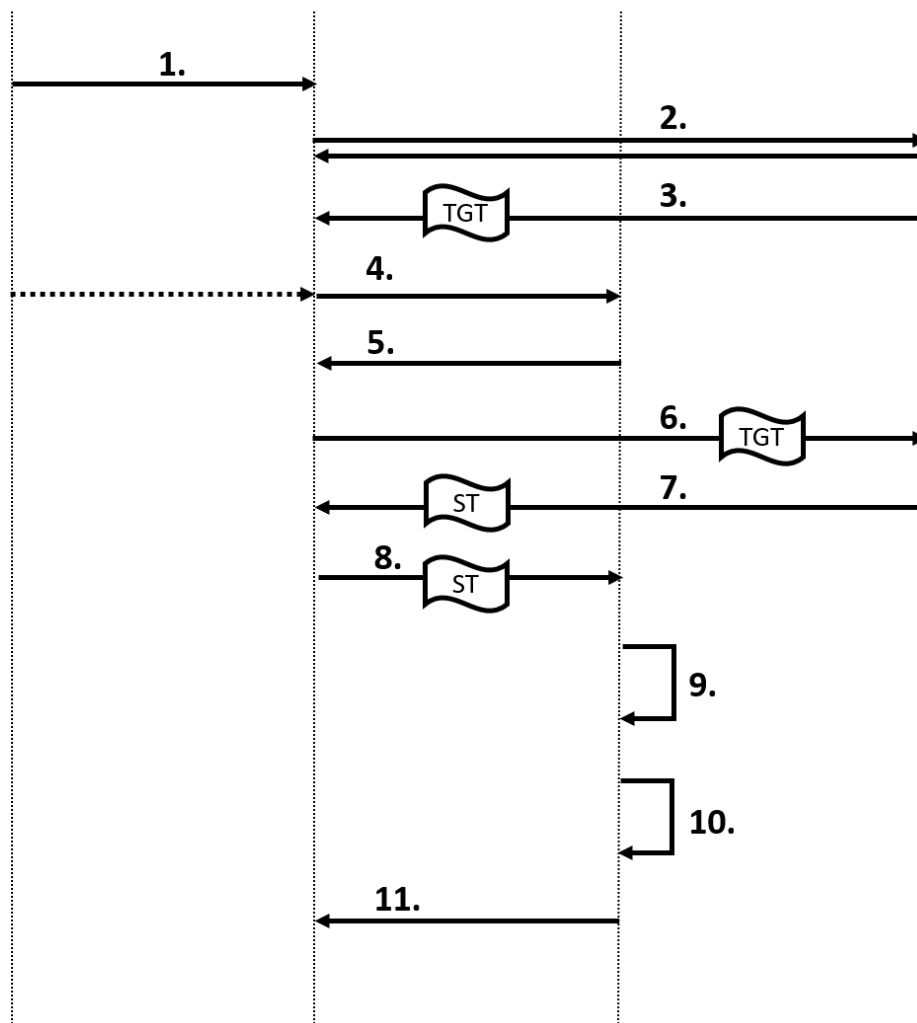
9.10. 統合 Windows 認証

Active Directory と通信することで、現在 Windows にログイン中のユーザが明示的な認証作業を行うことなく ASFALIS TransServer へログインできるように設定することができます。本機能を使用するためには、対象の環境に Active Directory が導入されている必要があります。認証の流れは以下の通りです。

1. ユーザがクライアントコンピュータにログインを試行
2. クライアントコンピュータはドメインコントローラと通信してユーザを認証
3. ログインが成功するとドメインコントローラがクライアントコンピュータに Ticket-Granting

Ticket (以下、TGT) を配布

4. ユーザはクライアントコンピュータから ASFALIS TransServer へのアクセスを試行
5. ASFALIS TransServer が認証要求を応答
6. クライアントコンピュータが TGT をドメインコントローラに提示
7. ドメインコントローラが TGT の有効性を確認して ASFALIS TransServer 用の Service Ticket (以下、ST) をクライアントコンピュータに提供
8. クライアントコンピュータが ST を ASFALIS TransServer に提示
9. ASFALIS TransServer が提示された ST の有効性を確認
10. ASFALIS TransServer が ST に含まれるユーザ名に基づいて ASFALIS TransServer に登録されたユーザを取得 (マッピング) し、マッピングされたユーザの有効性を確認
11. ASFALIS TransServer がクライアントコンピュータに対してログイン後の画面を応答



TGT の詳細については、以下のサイトをご参照ください。

Ticket-Granting Tickets

<https://docs.microsoft.com/ja-jp/windows/win32/secauthn/ticket-granting-tickets>



また ASFALIS TransServer は統合 Windows 認証を実装するため、SPNEGO を使用しています。SPNEGO の詳細については以下をご参照ください。

SPNEGO Token Handshake by Using HTTP Headers

[https://docs.microsoft.com/en-us/previous-versions/ms995330\(v=msdn.10\)#spnego-token-handshake-by-using-http-headers](https://docs.microsoft.com/en-us/previous-versions/ms995330(v=msdn.10)#spnego-token-handshake-by-using-http-headers)

本機能を利用する場合には以下の設定が必要となります。

- Active Directory の設定
- ASFALIS TransServer の設定
- クライアントコンピュータの設定

9.10.1. Active Directory の設定

本手順は Active Directory の管理者が実行してください。

次節以降に記載するコマンド例は、以下の条件で設定することを想定して記載しています。実行する際には適宜対象の環境に合わせて読み替えてください。

- ASFALIS TransServer の動作するコンピュータのホスト名: transserver
- 認証連携用に作成するユーザのユーザ名: transserver-user
- 認証連携用に作成するユーザのパスワード: password
- Kerberos のレルム: AD.EXAMPLE.COM
- サービスプリンシパル名: HTTP/transserver.ad.example.com

認証連携用ユーザの作成

ASFALIS TransServer と Active Directory が通信する際に利用するドメインユーザを新規に作成してください。この際には以下の点に注意してください。

- "ユーザーは次回ログオン時にパスワードの変更が必要" のチェックを外してください。
- 対話的にログオンするユーザではないため、なるべく長く複雑なパスワードを設定してください。またドメインのポリシーに反しない限り、"パスワードを無期限にする" もチェックしておくことを推奨します。

Kerberos keytab ファイルの作成 (サービスプリンシパル名の登録)

以下の例を参考に ktpass コマンドを実行して Kerberos keytab ファイルを作成し、作成されたファイルを ASFALIS TransServer の管理者に配布してください。コマンドは改行を含まず 1 行で実行して

ください。このコマンドを実行することで、ファイルの作成と併せてサービスプリンシパル名がドメインに登録されます。

```
ktpass -princ HTTP/transserver.ad.example.com@AD.EXAMPLE.COM -mapuser transserver-user@AD.EXAMPLE.COM -pass password -crypto ALL -ptype KRB5_NT_PRINCIPAL -out C:\temp\kerberos\transserver.keytab
```

各引数の説明は以下の通りです。

- princ: サービスプリンシパル名 (書式は以下の通りです)

HTTP/{ASFALIS TransServer の完全修飾ドメイン名}@{Kerberos のレルム}

- mapuser: ASFALIS TransServer と Active Directory が通信する際に利用するドメインユーザ名 (9.10.1.1, “認証連携用ユーザの作成” で作成したユーザのユーザ名を指定してください)
- pass: ASFALIS TransServer と Active Directory が通信する際に利用するドメインユーザのパスワード (9.10.1.1, “認証連携用ユーザの作成” で作成したユーザのパスワードを指定してください)
- crypto: Kerberos keytab ファイルに格納するキーの暗号化形式 ("ALL" を指定してください)
- ptype: プリンシパルの形式 ("KRB5_NT_PRINCIPAL" を指定してください)
- out: 作成する Kerberos keytab ファイルのファイル名とファイル出力先のパス

ktpass コマンドの詳細については、以下の URL を参照してください。

ktpass

<https://docs.microsoft.com/ja-jp/windows-server/administration/windows-commands/ktpass>



全ての設定が終了したら、以下の内容を ASFALIS TransServer の管理者に連絡してください。

- Kerberos のレルム
- サービスプリンシパル名
- ドメインコントローラの完全修飾ドメイン名

9.10.2. ASFALIS TransServer の設定

本手順は ASFALIS TransServer の管理者が実行してください。



以下の手順は、既に ASFALIS TransServer がインストール済みであるものとして記載しています。インストール未実施の場合には設定前に実施してください。

Kerberos 構成ファイルの作成

ASFALIS TransServer の動作するコンピュータの C:\Windows に、"krb5.ini" という名称のファイル

を作成してください。ファイルの内容は以下の通りです。

```
[libdefaults]
    default_realm = AD.EXAMPLE.COM

# The following krb5.conf variables are only for MIT Kerberos.
    kdc_timesync = 1
    ccache_type = 4
    forwardable = true
    proxiable = true

# The following libdefaults parameters are only for Heimdal Kerberos.
    fcc-mit-ticketflags = true

[realms]
    AD.EXAMPLE.COM = {
        kdc = controller.ad.example.com
        admin_server = controller.ad.example.com
        default_domain = ad.example.com
    }

[domain_realm]
    .ad.example.com = AD.EXAMPLE.COM
    ad.example.com = AD.EXAMPLE.COM
```

上記は設定例です。環境に合わせて以下の部分を変更してください。

- AD.EXAMPLE.COM: Kerberosのレルム
- controller.ad.example.com: ドメインコントローラの完全修飾ドメイン名
- ad.example.com: ドメイン名

keytab ファイルの配置

Active Directory の管理者から配布された .keytab ファイルを以下のパスに配置してください。

- <ATS Install Directory>\Apache\<バージョン>\Apache24\conf

ユーザの作成

統合 Windows 認証を用いて ASFALIS TransServer にログインすることを許可するユーザのアカウントを ASFALIS TransServer に登録してください。登録は「ユーザ管理」メニューから行うか、ユーザのインポート機能を用いて行います。

ユーザのインポート機能については、「ASFALIS TransServer 管理者マニュアル」の「付録」-「インポート機能」-「ユーザ」を参照してください。大量のユーザを登録する場合にはインポート機能が便利です。

登録したユーザをどの Active Directory ユーザに対応させるかについては、任意のマッピングルールを定義することができます。特に要件がない限り、Active Directory ユーザと同じユーザ名を指定す

ることを推奨します。

設定ファイルの編集

ASFALIS TransServer の設定ファイルを編集します。編集対象のファイルは以下の通りです。

- <ATS Install Directory>\config\k2server.ini

k2server.ini に含まれる Kerberos 認証関連の項目を以下の通り変更してください。

- Krb: "On" にします。
- KrbAuthName: ユーザに提示されるパスワードダイアログに表示する文字列を指定します。パスワードダイアログはパスワード認証を許可している場合にのみ表示されます。文字列の内容に問題がなければ編集する必要はありません。
- KrbAuthRealms: Kerberos のレルムを設定します。
- KrbServiceName: ASFALIS TransServer のサービスプリンシパル名を設定します。
- Krb5Keytab: .keytab ファイルを以下のフォルダからの相対パスで指定します。
<ATS Install Directory>\Apache<バージョン>\Apache24
- KrbMethodK5Passwd: ドメインに参加していないコンピュータからドメインのユーザーを用いて認証することを許可する場合は "On" に、許可しない場合は "Off" にします。"Off" にした場合、ドメインに参加していないコンピュータから ASFALIS TransServer にログインすることができなくなります。



編集を開始する前に ASFALIS TransServer を停止してください。

以下は設定例です。

```
Krb=On
KrbAuthName=TransServer authentication
KrbAuthRealms=AD.EXAMPLE.COM
KrbServiceName=HTTP/transserver.ad.example.com
Krb5Keytab=conf/transserver.keytab
KrbMethodK5Passwd=On
```

データベースの編集

以下の手順で ASFALIS TransServer のデータベースを編集してください。

1. pgAdmin 4 を起動します。
2. [PostgreSQL 10] - [Databases] - [trans_server_ex_***] - [Schemas] - [public] - [Tables] - [server_configurations] を選択します。
※ *** は ASFALIS TransServer のバージョン番号です。
3. テーブル内の以下の項目を編集します。
 - web.login.sso_reverse_proxy.enabled: TRUEに設定します。

- web.login.sso_reverse_proxy.login_name_mapping: ドメインユーザの名前を ASFALIS TransServer に登録されたユーザのログイン名にマッピングする Ruby のコードを記述します。

以下は、{任意のユーザ名}@AD.EXAMPLE.COM を {任意のユーザ名} にマッピングするための設定例です。

```
{ |req|
  orig_user_name = req['X-Forwarded-User'].downcase
  if orig_user_name =~ /(.)@ad\.example\.com$/i
    $1
  end
}
```

4. ASFALIS TransServer を起動します。



本手順を実施するとドメインユーザに関連付けられていないユーザで ASFALIS TransServer にログインすることができなくなります。問題が発生した場合には、"web.login.sso_reverse_proxy.enabled" の値を "FALSE" に戻してください。

9.10.3. クライアントコンピュータの設定

以下の手順に従って、クライアントコンピュータ (ASFALIS TransServer へアクセスを行うコンピュータ) で設定を実施してください。

1. クライアントコンピュータ上で Internet Explorer を起動します。
2. [インターネットオプション] を開きます。
3. [セキュリティ] タブを開きます。
4. "ローカルイントラネット" を選択し、[サイト] を選択します。
5. [ローカルイントラネット] ウィンドウで [詳細設定] を選択します。
6. "この Web サイトをゾーンに追加する" の下にある欄に、以下の URL を入力して [追加] をクリックします。
http://<ASFALIS TransServer の動作するコンピュータの完全修飾ドメイン名>
設定例: http://transserver.ad.example.com
7. [閉じる] を選択します。
8. [ローカルイントラネット] ウィンドウで [OK] を選択します。
9. [インターネットオプション] ウィンドウで [OK] を選択します。

手順は以上です。クライアントコンピュータから ASFALIS TransServer にアクセスし、認証画面が表示されることなく ASFALIS TransServer のコンソールが表示されることを確認します。

なお上記の設定を行った後であれば、Chrome を使用して ASFALIS TransServer にアクセスすることもできます。



本設定は ASFALIS TransServer にアクセスするすべてのコンピュータで実施してください。



ASFALIS TransServer の動作するコンピュータ上でブラウザを起動してアクセスした場合、認証が正常に動作しません。アクセスは必ず別のコンピュータから実行してください。現時点においてこれは製品としての制限事項となります。

9.10.4. トラブルシューティング

a. Internal Server Error が表示される

ASFALIS TransServer にアクセスする際に指定した URL が、サービスプリンシパル名を登録する際に用いたコンピュータ名の形式と一致していない場合に本エラーが発生することがあります。例えばサービスプリンシパル名には "transserver.ad.example.com" という完全修飾ドメイン名を用いている状態で、アクセス先 URL に "http://transserver:3000/" や "http://192.168.20.20:3000/" 等を指定した場合が考えられます。この場合、正しい URL は "http://transserver.ad.example.com:3000/" となります。サービスプリンシパル名に含まれるコンピュータ名の形式を用いて URL を指定するようにしてください。ドメインに登録されているサービスプリンシパル名は以下のコマンドで確認することができます。

```
setspn -L <認証連携用に作成したユーザのユーザ名>
```

なお、ASFALIS TransServer の動作するコンピュータ上でブラウザを起動してアクセスした場合にも Internal Server Error が表示されることがあります。

b. ブラウザの認証画面が表示される

ASFALIS TransServer にアクセスした際にブラウザの認証画面が表示される場合、以下のような原因が考えられます。

- クライアントコンピュータがドメインに参加していない。
- ユーザがクライアントコンピュータに対してドメインユーザの権限でログオンしていない。
- クライアントコンピュータで [9.10.3, “クライアントコンピュータの設定”](#) の内容を実施していない。
- ASFALIS TransServer の動作するコンピュータ上でブラウザを起動してアクセスしている。

c. 上記以外の問題が発生する

原因を切り分ける際、Apache のログから有用な情報が得られる場合があります。Apache のログは以下にあります。

- <ATS Install Directory>\log\Apache\k3_web.log.{YYYYDDMM}

上記のログに ASFALIS TransServer へのアクセスを行ったドメインユーザの名前が含まれている場合、ASFALIS TransServer の "server_configurations" に設定した内容に問題がある可能性があります。"web.login.sso_reverse_proxy.login_name_mapping" に設定したマッピングの内容を見直してください。

また上記のログにドメインユーザの名前が含まれていなかった場合、ASFALIS TransServer がド

メインユーザの情報を取得できていません。この場合には ktpass コマンド実行時の引数、.keytab ファイルの配置状態、"krb5.ini" の内容、"k2server.ini" ファイルの内容等を見直してください。

10. ユースケース

10.1. 抽出したファイルごとにシナリオを適用する

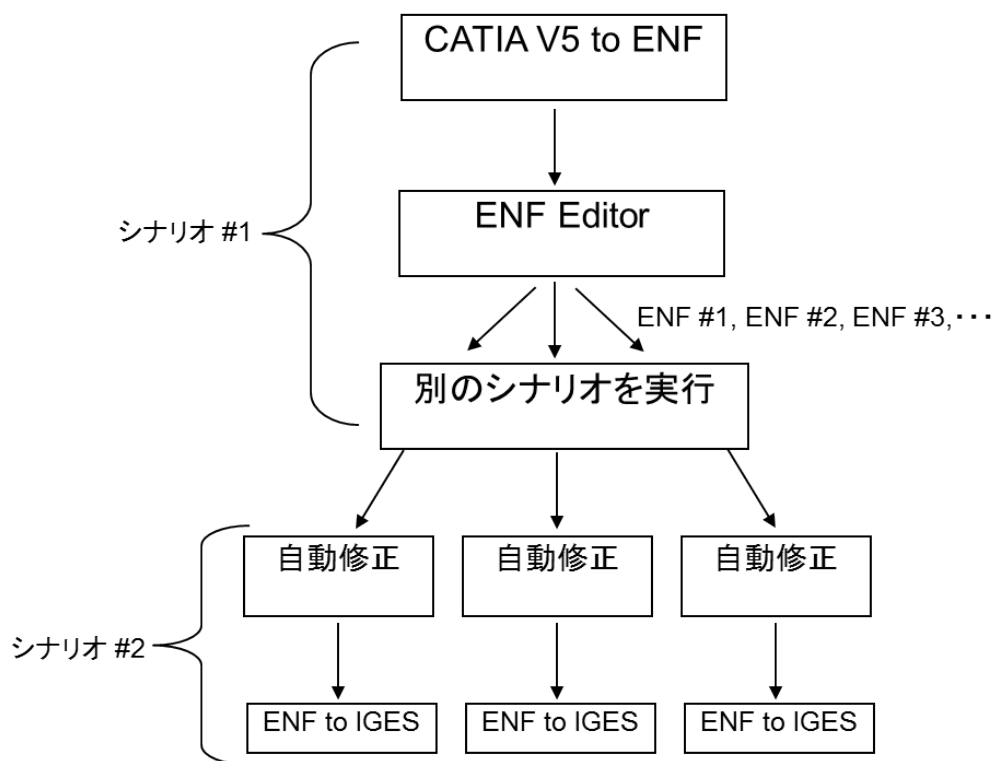
コンポーネント「ENF Editor」と「別のシナリオを実行」を使用することで、以下の条件を満たす変換を1つのシナリオにて実行することが可能です。

- 変換対象モデルから必要な部分を抜き出すなどして複数のENFファイルを出力
- それらの複数のENFファイルに対してある一つのシナリオをまとめて適用

以下に例を示します。

10.1.1. CATIA V5 to IGES変換の例

例として以下のようなシナリオによる変換を考えます。



この"シナリオ#1"は以下のようなフローとなっています。

1. CATIA V5 to ENFにてENFを出力
2. ENF Editorにて必要な部分を抽出し、複数ファイル(ENF)を出力
3. それぞれのENFに対して"シナリオ#2" (自動修正+ENF to IGES)を適用

シナリオの設定例は以下の通りです。

●コンポーネント 1 (catia52enf) CATIA V5インポート

入力ファイル選択
 入力catia52enfファイル: ジョブ入力 1

●コンポーネント 2 (enfeditor) ENF編集

入力ファイル選択
 入力enfファイル: catia52enf-出力enfファイル

☐ コンポーネントの出力ファイルをシナリオの結果ファイルとする: 出力enfファイル
☒ コンポーネントの出力ファイルをシナリオの結果ファイルとする: 抽出ENF

コンポーネントパラメータ設定

●ENF 編集用Rubyスクリプトファイル(アップロード) (scriptfile_upload) ☐ 設定する
 ●ENF 編集用Rubyスクリプトファイル (scriptfile) ☐ 設定する
 ●パラメータファイルリスト (parameterfile_list) ☐ 設定する
 ●任意パラメータ (custom_parameters) ☒ 設定する

パラメータの説明 任意のパラメータを設定します。キーと値は =(イコール) で区切って下さい。

パラメータ指定方法 ☐ あとでジョブ投入時に指定する
☒ 値を設定する

パラメータの固定値

```
use_asmedit=1
UseRegexpInNameList=1
PartialAssemblyExtraction=1
extraction_target=C:\work\temp\extr
actlist_sample01.xml
```

パラメータ可視 ☒ ユーザに表示する

- ENF Editorのパラメータとしては、以下の任意パラメータを設定してください。

```
use_asmedit=1
ExtractByRegularExpression=1
PartialAssemblyExtraction=1
extraction_target=C:\work\temp\extractlist_sample01.xml (抽出対象XMLファイル)
```

抽出対象XMLファイルの例は以下の通りです。

```
<?xml version="1.0" encoding="Shift_JIS" ?>
<AssemblyExtractionList>
  <AssemblyExtractionRule type="name" keep_position="yes" >
    <Part name="sample01-1" />
  </AssemblyExtractionRule>
  <AssemblyExtractionRule type="name" keep_position="yes" >
    <Part name="sample01-2" />
  </AssemblyExtractionRule>
</AssemblyExtractionList>
```

*この例は"sample01-1"、"sample01-2"というパート名のパートを抽出するためのXMLファイルです。

●コンポーネント 3 (execute_another_scenario) 別のシナリオを実行 ▼

入力ファイル選択

入力ファイル **enfeditor-抽出ENF** ▼

☐ コンポーネントの出力ファイルをシナリオの結果ファイルとする：出力ファイル

コンポーネントパラメータ設定 ⊕ ⊖ ✓

●実行する条件 (execute_condition) ☐ 設定する

●実行シナリオ (another_scenario) ☒ 設定する

パラメータの説明 実行するシナリオを指定します。

パラメータ指定方法 ☐ あとでジョブ投入時に指定する
☒ 値を設定する

パラメータの固定値 **ENF to IGES #1** ▼

パラメータ可視 ☒ ユーザに表示する

●子ジョブの実行結果を待ち受ける (wait_until_child_jobs_end) ☐ 設定する

●ジョブ投入指示ファイル (job_entry_direction_file) ☐ 設定する

- コンポーネント「別のシナリオを実行」の入力ファイルとしては "enfeditor-抽出ENF" を選択してください。
- 上記の例も含め「別のシナリオを実行」で呼び出すシナリオのパラメータについて「値を設定する」を選択し、パラメータの固定値を指定してください。

このシナリオに対してジョブを投入すると、ジョブ情報ページにて以下のような変換結果を確認できます。

ジョブ情報

ステータス 正常終了
登録日時 08/03 23:57
オペレーション デフォルト
シナリオ名 V5 + EE + another scenario #1
入力ファイル sample01.CATProduct
作成者 Toshiya Sakai
メモ
ジョブ識別名 job_85_20160803_145756

コンポーネントステータス

出力ファイルダウンロード ログダウンロード すべて開く すべて閉じる

<input type="checkbox"/>	詳細	出力ファイル	ログ
<input type="checkbox"/> CATIA V5インポート catiaV52enf	CATIA V5からENFへの変換 ステータス 正常終了 (TRUTH01:59859) メッセージ	sample01.enf	sample01.log sample01_log.xml
<input type="checkbox"/> ENF編集 enfeditor	Rubyスクリプトを用いてENFを編集します。 ステータス 正常終了 (TRUTH01:59859) メッセージ ●コンポーネントパラメータ 任意パラメータ use_asmedit=1 UseRegexpInNameList=1 PartialAssemblyExtraction=1 extraction_target=C:\work\templextractlist_sample01.xml	sample01.enf sample01-1.enf sample01-2.enf	sample01.log sample01_log.xml
<input type="checkbox"/> 別のシナリオを実行 execute_another_scenario	指定した別のシナリオを実行します。 ステータス 正常終了 メッセージ ●コンポーネントパラメータ 実行シナリオ 22	sample01-1.enf	

●子ジョブ一覧

入力ファイル

表示件数

ジョブID	入力ファイル	登録日時	作成者	業務種別	オペレーション	シナリオ	ステータス
87	sample01-2.enf	08/03 23:58	Toshiya Sakai	デフォルト	デフォルト	ENF to IGES #1	正常終了
86	sample01-1.enf	08/03 23:58	Toshiya Sakai	デフォルト	デフォルト	ENF to IGES #1	正常終了

この例では、「ENF編集」にて2ファイルのENFが出力され、「別のシナリオを実行」にてそれぞれ

のENFに対して"ENF to IGES #1"というシナリオが適用されていることが分かります。

このような形で、一つのシナリオ内で複数ファイル(ENF)を抽出しそれぞれのENFに対してある一つのシナリオをまとめて適用することができます。

10.2. カスタムコンポーネントの実行結果を後続のコンポーネントで利用する

カスタムコンポーネントの実行結果を後続のコンポーネントのパラメータとして利用することができます。具体例は以下の通りです。

10.2.1. コンポーネントの実行結果をメールで通知する

処理の実行結果を示す文字列をカスタムコンポーネントで登録し、登録した文字列を後続の「メール送信」コンポーネントで利用します。

1. 処理の結果に基づいて以下のようなJSONファイルを出力する実行ファイルを用意します。

処理成功時:

```
{
  "processResult": "INFORMATION: Process succeeded."
}
```

処理失敗時:

```
{
  "processResult": "WARNING: some errors occurred."
}
```

2. 出力されたJSONファイルを登録するための設定を行います。手順については「[9.2.5, “任意の情報の ASFALIS TransServer への登録”](#)」を参照してください。
3. 「メール送信」コンポーネントの定義で本文に以下の内容を設定します

```
Process Result: #{component.custom_component1.properties["processResult"]}
```

*) "custom_component1" の部分には実行結果を登録したカスタムコンポーネントの名称を設定してください。詳細は「[9.6, “パラメータ中で展開される文字列”](#)」を参照してください。

上記の通り設定した場合には処理の実行結果に応じて以下のようなメッセージを含むメールが送信されます。

処理成功時:

Process Result: INFORMATION: Process succeeded.

処理失敗時:

Process Result: WARNING: some errors occurred.

10.2.2. コンポーネントの実行結果に基づいて別のシナリオの実行を制御する

処理の実行結果に含まれるエラーの数がしきい値を超えている場合にのみ「別のシナリオを実行」コンポーネントを実行します。

1. 処理の結果として出力されたエラーの数を以下のようなJSONファイルとして出力する実行ファイルを用意します。以下は例であり、"5" についてはエラーの数に応じて変わるようにしてください。

```
{  
  "errorCount": 5  
}
```

2. 出力されたJSONファイルを登録するための設定を行います。手順については「[9.2.5, “任意の情報の ASFALIS TransServer への登録”](#)」を参照してください。
3. 「別のシナリオを実行」コンポーネントの"実行する条件"に以下の通り設定します。

```
execute_if #{component.custom_component2.properties["errorCount"]} > 0
```

*"custom_component2" の部分には実行結果を登録したカスタムコンポーネントの名称を設定してください。詳細は「[9.6, “パラメータ中で展開される文字列”](#)」を参照してください。

上記の通り設定した場合、エラーの数に応じて以下の様に動作が変わります。

- エラーの数が0だった場合(execute_if 0 > 0):
しきい値として0が設定されているためエラーの数はしきい値以下と判定され「別のシナリオを実行」コンポーネントは実行されません。
- エラーの数が3だった場合(execute_if 3 > 0):
しきい値として0が設定されているためエラーの数はしきい値以上と判定され「別のシナリオを実行」コンポーネントが実行されます。

10.3. コンポーネントの実行結果に基づくフロー制御

ASFALIS TransServer の各コンポーネントを実行した結果に基づいて、別シナリオの実行要否もしくはコンポーネントの停止要否を判断することができます。「別のシナリオを実行」コンポーネントもしくは「ジョブ停止」コンポーネントの「実行する条件」パラメータに、以下の内容を設定してください。

```
execute_if #{component."コンポーネントインスタンス名".properties["rc"]} "演算子"
"エラーコード"
```

"演算子" には一般的な算術演算子("==" や "!=" 等)を指定してください。具体的な使用例は以下の通りです。

- 例1 形状比較が 1202 で終了した場合にのみ実行する

```
execute_if #{component.geomdiff.properties["rc"]} == 1202
```

- 例2 NXエクスポートが 0 で終了した場合にのみ実行する

```
execute_if #{component.enf2nx.properties["rc"]} == 0
```

- 例3 NXエクスポートが 0 以外で終了した場合に実行する

```
execute_if #{component.enf2nx.properties["rc"]} != 0
```

なお拡張エラーコード(1000以上のリターンコード)を利用する際は、該当するコンポーネントについてSlave Node側で設定を行う必要があります。利用する場合にはSlave Nodeの『ノードの設定』に以下の通り設定してください。

リターンコード	設定する内容
1001	EXTEND_ERROR_INFO 1もしくはEXTEND_ERROR_INFO 2
1002	EXTEND_ERROR_INFO 1もしくはEXTEND_ERROR_INFO 2
1003	EXTEND_ERROR_INFO 1もしくはEXTEND_ERROR_INFO 2
1011	EXTEND_ERROR_INFO 2
1012	EXTEND_ERROR_INFO 2
1021	EXTEND_ERROR_INFO 2
1202	EXTEND_ERROR_INFO 2

リターンコード 1201 を利用する場合には、PDQ CheckerでOptionalReturnCode=2 を設定してください。

10.4. ENF Editorの出力する情報を後続のコンポーネントで利用する

「ENF Editor」コンポーネントで任意の情報をASFALIS TransServerに登録しておき、登録した情報

を後続のコンポーネントで利用することができます。「ENF Editor」コンポーネントから呼び出すスクリプトに、メインの処理に加えて以下の処理を追加することで情報を登録することができます。

- a. 必要な情報を含むハッシュを作成する。
- b. ハッシュをJSONファイルとしてワークフォルダに書き出す。
- c. 出力したJSONファイルの絶対パスを propertyfile キーに指定する。

スクリプトのサンプルを以下に示します。

```
require "rubygems"
require "enfeditor"
require "json"
class MyEnfEditor
  def initialize
  end
  def execute(argv)
    EnfEditor::EESession.open(argv) do |session|
      input_enf = session.parameter['inputfile']
      output_enf = session.parameter['outputfile']
      session.read_enf(input_enf)

      # ハッシュの作成
      exported_hash = {
        "Property1" => 1,
        "Property2" => {
          "Property2-1" => "2-1"
        }
      }

      # JSONファイルの書き出し
      json_path = "property.json"
      File.open(json_path, "w") do |f|
        f.write JSON.generate(exported_hash)
      end

      # JSONファイルの絶対パスを propertyfile キーに指定
      session.xmllog.log_additional_output_filename(File.expand_path(json_path), 'propertyfile')
      session.write_enf(output_enf)
    end
  end
end

z = MyEnfEditor.new
z.execute(ARGV)
```

上記のスクリプトを「ENF Editor」コンポーネントから実行した場合、ワークフォルダに property.json というファイルが出力され、その内容がASFALIS TransServerに登録されます。登録された情報は、以下の様にして後続のコンポーネントで呼び出すことができます。

設定例 1 :

```
#{component.enfeditor.properties["Property1"]}
```

上に挙げたサンプルスクリプトを実行した場合、このように設定すると "1" が返されます。

設定例 2 :

```
#{component.enfeditor.properties["Property2"]["Property2-1"]}
```

上に挙げたサンプルスクリプトを実行した場合、このように設定すると "2-1" が返されます。

10.5. コンポーネントを実行するSlave Nodeを指定する

各コンポーネントを実行するSlave Nodeを明示的に指定することができます。例えばシナリオによって処理を実行するSlave Nodeを分けたい場合などに有効です。設定方法は以下の通りです。

1. [管理者機能] - [コンポーネント管理] を選択します。
2. コンポーネント一覧から実行するSlave Nodeを指定したいコンポーネントを選択します。
3. "その他のパラメータ" にある SlaveNodeName と SlaveNodeLabel のいずれかもしくは両方をチェックします。
4. [管理者機能] - [シナリオ管理] を選択します。
5. 既存のシナリオを選択するか、新規にシナリオを作成します。
6. 実行するSlave Nodeを指定したいコンポーネントについて SlaveNodeName もしくは SlaveNodeLabel の値を設定します。

●スレーブノード指定 (SlaveNodeName) ☒ 設定する

パラメータの説明 コンポーネントを実行するスレーブノードのマシン名とポート番号を指定します。"(マシン名):(ポート番号)"を入力。

パラメータ指定方法 ☐ あとでジョブ投入時に指定する
☒ 値を設定する

パラメータの固定値

パラメータ可視 ☐ ユーザに表示する

●スレーブノードのラベル指定 (SlaveNodeLabel) ☐ 設定する

パラメータの説明 指定されたラベルを持つスレーブノードでコンポーネントを実行します。

パラメータ指定方法 ☐ あとでジョブ投入時に指定する
☒ 値を設定する

パラメータの固定値

パラメータ可視 ☐ ユーザに表示する

SlaveNodeName を指定した場合には、名前とポートが一致する特定のSlave Nodeでコンポーネント

が実行されます。SlaveNodeLabel を用いる場合には、複数のSlave Nodeをひとつのグループとして扱い、そのグループに属するSlave Nodeのいずれかでコンポーネントが実行されます。

11. トラブルシューティング

11.1. ASFALIS TransServerのログイン画面が表示されない

発生原因(1)

ASFALIS TransServerの起動に成功しているがファイアウォールの設定に不備がある

対策(1)

サーバを起動したPCからは接続に成功するが他のクライアントPCから接続に失敗する場合、ファイアウォールの設定を確認してください。

発生原因(2)

既に起動しているアプリケーションとポートが重複しているためASFALIS TransServerの起動に失敗している

対策(2)

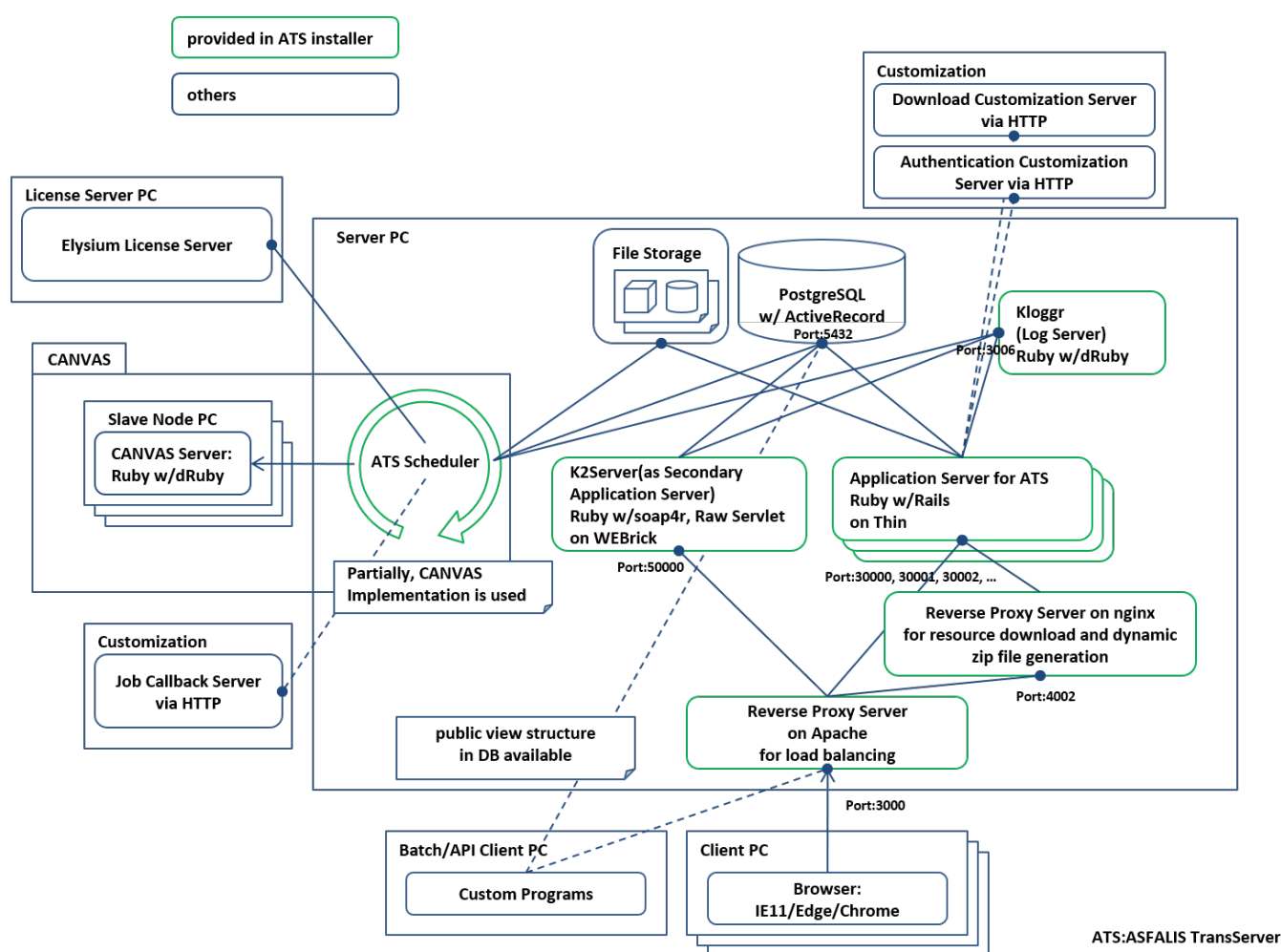
<ASFALIS TransServer導入フォルダ>\config\k2server.ini に記述されているポート番号を変更してください。k2server.ini に設定されている既定のポート番号については次の表を参照してください。

プロセス	ポート番号 (既定値)	設定箇所
Apache (リバースプロキシ)	3000	[K2server] セクションのListenPort
Nginx (負荷分散、zip作成)	4002	[Nginx] セクションのListenPort
Rails (アプリケーションサーバ)	30000, 30001, 30002	[K2dashboard]セクションのBalancerMemberStartPort (BalancerMemberNumに指定された数だけポートを使用)
KLoggr (ログ記録)	3006	[Kloggr] セクションの ListenPort
K2server (統計情報記録)	50000	[K2server] セクションの BalancerMemberStartPort
MasterController (スケジューラ監視)	3011	[K2scheduler] セクションの MasterControllerPort(*)
QueueController (ジョブのキュー管理)	3012	[K2scheduler] セクションの QueueControllerPort(*)
MiscWorker (バックアップ等)	3013	[K2scheduler] セクションの MiscWorkerPort(*)
ScriptComponentWorker (スクリプト実行)	3100, 3101	[K2scheduler] セクションの ScriptComponentWorkerPortStart(*) (ScriptComponentWorkerNumに指定された数だけポートを使用)

プロセス	ポート番号 (既定値)	設定箇所
PerSlavenodeWorker (Slave Nodeとの通信)	3200, ...	[K2scheduler] セクションの PerSlavenodeWorkerPortStart(*) (Slave Nodeの個数だけポートを使用)

(*) : 初期設定状態ではこの項目は存在しません。既定値が使用されます。明示的に指定する際には [K2scheduler] セクションを作成した上で追加してください。

上記の表にあるプロセス間の関係は以下の図の通りです。MasterController 以降の項目はいずれも ATS Scheduler の中で動作しています。



11.2. ファイルが存在しないというエラーが発生する

共有ネットワークフォルダ上にファイルが存在しないというエラーが発生するものの実際にはファイルが存在している、という問題が発生する場合があります。この現象の発生原因と対策は以下の通りです。

*この問題は同じ条件下でも発生する場合と発生しない場合があります。

発生原因

SMBバージョン2を利用していることで共有ネットワークフォルダ上のファイルのキャッシュが不正な状態になっている。

共有ネットワークフォルダはWindowsファイル共有の仕組みを利用しています。Windowsファイル共有の利用するプロトコル(SMB)にはバージョンが1から3まであり、バージョン2を利用している場合に、ファイルのキャッシュが不正な状態になることで本問題が発生することがあります。

対策

この問題はSMBバージョン3を利用することで回避することができます。SMBバージョン3を利用するためには、ファイルサーバと、共有ネットワークフォルダを利用する側 (ASFALIS TransServerおよびSlave Node) の両方がバージョン3に対応している必要があります。

11.3. ASFALIS TransServerへのログインに失敗する

理由と対策は以下の通りです。

(ア) アカウント情報が不正

- ログイン名、パスワードなどが一致していない場合、ログインに失敗します。正しいログイン名、パスワードを確認の上、再度ログイン操作を行ってください。

(イ) 該当ユーザが既にログインしている

- デフォルトでは同一ユーザが複数セッションを使用することはできません。他のPCから同一ユーザがログインしている場合はログインに失敗します。他のPCでログアウトしてから再度ログイン操作を行ってください。
- 以下のサーバ設定で、同一ユーザが複数セッションを使用できるよう設定することが可能です。

```
web.login.enable_multiple_login
```

同一ユーザが複数箇所から接続できるよう、複数のセッションを使用することを許容するかどうかを設定します。同時に使用できるセッション数はASFALIS TransServerのライセンスによって決められたクライアントの接続数までに制限されます。

(ウ) 同時接続可能数を超えている

- ASFALIS TransServer はご購入頂いたライセンス数分までしか同時にログインして使用することができません。該当するエラーメッセージが表示された場合には他のユーザがログアウトしてから再度ログイン操作を行ってください。

(エ) ユーザのログインが無効になっている

- アカウントが無効になっている旨のメッセージが表示された場合、そのユーザはログインできないように設定されています。ログインできるようにするにはユーザ編集ページからログイン可能に設定してください。

(オ) keyファイルを削除してしまった

- <ASFALIS TransServer導入フォルダ>\k2dashboard\key に作成されるファイルを誤って削除してしまうと、ASFALIS TransServerに登録されたパスワード情報を使用してログインできなくなりますのでご注意ください。

(カ) サーバ・クライアント間で時刻が合っていない

- サーバ・クライアント間で時刻が合っていない場合、ログイン処理が不安定になる場合があります。時刻が合っていることをご確認ください。

11.4. タスクマネージャーからのサービス再起動に失敗する

タスクマネージャーの [サービス] タブからASFALIS TransServerのサービスを再起動した際、サービスの停止に時間がかかることで、サービスの状態が「停止」から「開始中」に切り替わらない場合があります。この場合、停止状態のサービスを手動で起動してください。なおサービス管理ツールから再起動を行った場合には、サービスの再起動が正しく行われます。

11.5. 各ログファイル/オプションの説明

- <ASFALIS TransServer導入フォルダ>/log/install_k2.log

ASFALIS TransServerインストール時のログファイルです。
インストールに問題が起きた場合にはこのファイルを参照してください。

- <ASFALIS TransServer導入フォルダ>/log/transserver.log

ASFALIS TransServerのログファイルです。
ASFALIS TransServer使用中に、個別のSlave Node以外で問題が起きた場合にはこのファイルを参照してください。

- <ASFALIS Component導入フォルダ>/log/slave_node_(ポート番号).log

Slave Nodeでの処理についてのログファイルです。
ASFALIS TransServerでは変換や最適化等の負荷が高い処理をSlave Nodeに委譲します。
Slave Node上に委譲した処理で問題が起きた場合にはこのファイルを参照してください。

- ジョブ詳細画面に表示されるログファイル

コンポーネントの実行が成功したとき、またコンポーネントの実行に失敗した場合でもログファイルを検出することができた場合には、ログファイルを画面から参照することができます。この画面では、投入ファイルがマルチバイトの文字列を含んでいた場合、その部分が?に変換されて表示されることがあります。

- ジョブ詳細画面に表示される「サポート」オプション

実行に失敗したジョブの実行フォルダが残されている場合、JobAdminsグループのユーザはジョブ詳細画面を表示すると「サポート」という追加機能が使用できます。
ベンダ向けにモデルファイルや詳細な再現情報を送付する場合にご利用ください。

12. 付録

12.1. メール送信コンポーネント

メール送信コンポーネントを使用することで、ジョブ実行の完了をメールで通知することができます。送付するメール文面には、ジョブ情報を記載することも可能です。「2.4, “サーバ設定”」を参照し、事前にメール送信に必要なサーバ設定を行ってください。

12.1.1. サンプルシナリオ

- コンポーネント1 CADインポート (IGESインポート)
- コンポーネント2 自動修正
- コンポーネント3 CADエクスポート (STEPエクスポート)

上記のシナリオで、変換実行後にメール送信する場合には、コンポーネント4としてメール送信を追加したシナリオを作成します。

The screenshot shows the configuration interface for the Email component (Component 4). The interface includes several sections:

- Component 1 (iges2enf):** CADインポート, IGESインポート
- Component 2 (dek):** 自動修正
- Component 3 (enf2step):** CADエクスポート, STEPエクスポート
- Component 4 (mail):** メール送信

Below the component selection, there is a section for "コンポーネントパラメータ設定" (Component Parameter Settings) for Component 4. The settings include:

- 送信元 (from) ☒ 設定する
- 宛先 (to) ☐ 設定する
- 宛先(ユーザ選択) (to_by_user) ☐ 設定する
- ジョブ投入者に送付する (to_job_user) ☐ 設定する
- Cc (cc) ☐ 設定する
- Cc(ユーザ選択) (cc_by_user) ☐ 設定する
- 件名 (subject) ☒ 設定する
- ヘッダ (body_header) ☐ 設定する
- 本文 (body) ☒ 設定する
- フッタ (body_footer) ☐ 設定する
- 添付ファイル (attached_files) ☐ 設定する

件名、ヘッダ、本文およびフッタには以下の変数を使用することが可能です。

入力ファイル名	<code>#{component.(component instance name).param.inputfile}</code> 例) <code>#{component.iges2enf.param.inputfile}</code>
出力ファイル名	<code>#{component.(component instance name).param.outputfile}</code> 例) <code>#{component.enf2step.param.outputfile}</code>
実行シナリオ名	<code>#{scenario.name@"ロケール名"}</code>

ジョブ詳細ページのURL	<code>#{serverconfig.web.common.server_base_address}/jobs/#{job.id}</code>
--------------	--

(component instance name)はコンポーネント名の横に表示されている文字列に置き換えて記載してください。



ロケール名には en-US、ja-JP、de-DE、zh-CNのいずれかを指定します。

12.2. ジョブ停止コンポーネント

ジョブ停止コンポーネントを使用することで設定された条件が満たされた場合にジョブを異常終了にすることができます。パラメータとして"実行する条件"を定義するようになっているので、以下の例の様に `execute_if` に続く形で条件を指定します。

- 例 1

```
execute_if #{component.enfchecker.pdq["G-SU-IS"].count} > 0
```

enfcheckerコンポーネントでPDQエラー「曲面の自己干渉」が見つかった場合に動作します。

- 例 2

```
execute_if #{component.dek.properties.ExtendedErrors.count} > 0
```

dekコンポーネントで拡張エラーが出力された場合に動作します。

12.3. 機密情報削除ツール

「機密情報削除ツール」は、ASFALIS TransServerのデータベースに対して機密性の高い情報を削除するツールです。

ASFALIS TransServerにて発生した問題への技術サポートの際、その解決のためにASFALIS TransServerが使用しているデータベースに含まれる情報が役立つ機会があります。しかし、情報セキュリティ上の理由からそのままデータベース情報を技術サポート窓口へ提供することは困難な場合もあります。

このツールを使用することで、ASFALIS TransServerのデータベースの機密性を排除することができ、技術サポート時に有効な情報として開発元や代理店へのデータベース情報送付が可能となります。

12.3.1. 基本的な操作手順

基本的には、以下の手順に沿って機密性を排除したASFALIS TransServerのデータベースを準備することができます。

1. データベースのバックアップ
管理者マニュアル 参照
2. 機密情報削除コマンド を試行
データベース更新前にどの情報が削除されるか事前確認可
3. 機密情報削除コマンド を実行
機密性を排除した形でデータベースを更新
4. データベースのバックアップ
管理者マニュアル 参照。このデータを開発元などへ送付
5. データベースのリストア
管理者マニュアル参照。必要に応じて1.でバックアップしたデータから元の環境を復元



【重要】

実行するとデータベースが直接更新されるため、機密情報削除コマンドは本番環境に対しては実行しないようにしてください。このコマンドについては、本番環境からデータベースをバックアップ/リストアした別の環境で実行するようにしてください。

上記の手順 2.や手順3.で挙げた「機密情報削除コマンド」の使用方法詳細について、次節で説明します。

12.3.2. 機密情報削除コマンドの試行（機密情報検証用ファイルの事前確認）

1. 機密情報削除コマンドを試行する際は、<ATS Install Directory>\k2dashboard\start-console.bat を実行してコンソールを起動し、以下のコマンドを実行します。この操作はASFALIS TransServer が停止している時に行ってください。

```
>bundle exec rake support:filter_private_data FILTER_CONFIG_PATH="<ATS Install Directory>\config\filter_private_data_config.rb"
```

- filter_private_data_config.rb は<ATS Install Directory>\configフォルダ内に配置されています。FILTER_CONFIG_PATHは各環境に合わせて設定してください。
- 2. 上記コマンドの実行後、<ATS Install Directory>\k2dashboard\内に「機密情報検証用ファイル」(*.csv)が出力されます。
機密情報検証用ファイルとは、どの情報が実際に削除されるか、そのまま残るか等を確認できるファイルです。機密情報検証用ファイルとしては以下の3種類のCSVファイルが出力されます。
 - i. フィルタリスト(filter_private_data_detail_filtered.csv)
置換された文字列

ii. ホワイトリスト(filter_private_data_detail_white.csv)

置換されなかった文字列で機密データを含まないとされるもの。ほとんどの場合このファイルの内容を確認する必要はありません

iii. グレーリスト(filter_private_data_detail_gray.csv)

置換されなかった文字列で機密データを含む可能性があるもの

グレーリストに機密情報が含まれている場合、必要に応じて設定ファイル

(filter_private_data_config.rb) を編集して、該当するテーブルの該当するカラムをフィルタする情報のリストに加える等した上で、再度本ツールを実行してください。

手作業でデータを編集する場合には、pgAdmin 4等のツールを使用して機密情報にあたる情報を直接編集してください。データベース内のテーブル名とIDはグレーリストに含まれています。

12.3.3. 機密情報削除コマンドの実行（データベースの更新）

1. 機密情報削除コマンドを実行する際は、<ATS Install Directory>\k2dashboard\start-console.bat を実行してコンソールを起動し、以下のコマンドを実行します。この操作はASFALIS TransServer が停止している時に行ってください。

```
>bundle exec rake support:filter_private_data FILTER_CONFIG_PATH="<ATS Install Directory>\config\filter_private_data_config.rb" DRY_RUN=NO
```

- filter_private_data_config.rb は<ATS Install Directory>\configフォルダ内に配置されています。FILTER_CONFIG_PATHは各環境に合わせて設定してください。
2. 上記コマンドの実行後、<ATS Install Directory>\k2dashboard\内に「機密情報検証用ファイル」(*.csv)が出力されます。その3種類のファイルの内容通りに実際のデータベースも更新されます。

データベースが直接更新されるため、この状態でASFALIS TransServerを実際に起動することもできます。

- デフォルトユーザ admin のパスワードは admin に書き換えられます。

12.3.4. 注意事項

以下の点にご留意ください。

- 本ツールはデータベースの内容を変更しますが、CADモデル等のファイルには変更を加えません。
- データベースには真偽値、整数値、文字列、日付などの情報が含まれています。本ツールではデータベースに含まれる文字列の情報のみを変更します。
- 既定の設定では、コンポーネントに与えるパラメータのキー、値はサポートに必要な情報であるため、フィルタされません。メールコンポーネント等を使用している場合にはご注意ください。
- 既定の設定では、ASFALIS TransServerに接続するための情報を含むASFALIS TransServerのサーバ設定項目については、パスワードに類するものを除いてフィルタされません。また、Slave Nodeを特定するための「マシン名+ポート番号」の情報もフィルタされません。一方、ASFALIS

TransServerに接続するクライアントの情報（IPアドレス等）はフィルタされます。

12.4. リソースフォルダの変更方法

ディスクフルなどの理由により、インストール時に指定したリソースフォルダのパスを変更したい場合には、以下の手順で設定を変更してください。

1. ASFALIS TransServerを停止します。
2. <ASFALIS TransServerの導入フォルダ>\k2dashboard\start-console.bat を実行してコンソールを起動し、以下のコマンドを実行してリソースフォルダを変更先にコピーします。

```
>bundle exec rake ats:resources:copy  
RESOURCE_PATH=(リソースフォルダに指定するフォルダのフルパス)
```



リソースフォルダのパスにはローカルドライブのパスか UNC パスを指定するようにしてください。ネットワークドライブを含むパス (例 N:\(transserver\resources) を指定した場合、ASFALIS TransServer が正常に動作しない場合がありますので指定しないでください。

3. 変更前のリソースフォルダを削除します。
4. テキストエディタで以下のファイルを開き、[K2server]セクションにあるPrivateResourcePathの値を、リソースフォルダに指定するフォルダのフルパスに変更します。 <ASFALIS TransServerの導入フォルダ>\config\k2server.ini
5. ASFALIS TransServerを開始します。

12.5. 共有ネットワークフォルダの変更方法

インストール時に指定した共有ネットワークフォルダのパスを変更したい場合には、以下の手順で設定を変更してください。

1. ASFALIS TransServerを停止します。
2. テキストエディタで以下のファイルを開きます。
<ASFALIS TransServerの導入フォルダ>\set_ruby_env.bat
3. "@set SHARED_FOLDER="に続く部分の内容を変更します。
4. 変更内容を保存してテキストエディタを終了します。
5. ASFALIS TransServerを開始します。

共有ネットワークフォルダを設定する際、以下の点に注意してください。

- データ処理時の出力フォルダやワークフォルダは共有ネットワークフォルダの下に作成されるため、共有ネットワークフォルダには空き容量の大きなドライブを割り当てる必要があります。
- 共有ネットワークフォルダにはUNCパス(\\で始まるパス)を設定できません。

- ドライブのルートフォルダ("C:\"など)を共有ネットワークフォルダにする事はできません。

12.6. インポート機能

ASFALIS TransServer内で取り扱う各情報をJSON形式で入力する機能で、コンソールを操作して利用します。一部は管理機能の画面からも利用することができます。

別のASFALIS TransServerでエクスポートしたJSONファイルを経由して情報を導入したり、外部システムのユーザ・組織情報から作成したJSONファイルを経由して情報を同期したりすることができます。



注意)

インポートするJSONファイルは、RFC4627準拠の形式で、文字コードはUTF-8を使用して記述してください。また、コンソールからの操作はASFALIS TransServer が停止している時に行ってください。

12.6.1. ユーザ

後述する仕様のJSON形式でファイルを作成し、下記のコマンドを実行することで、ASFALIS TransServerのユーザの登録および更新が可能です。

```
bundle exec rake ats:users:import ATS_USERS_IMPORT_FILE=C:\path\to\users.json
```

JSONファイルは以下の形式で指定してください。

階層	ノード名	説明	型	必須
1	format_version	1 を指定します。	int	○
1	users	-	array of object	○
2	(user)	-	object	-
3	login_name	ログイン名を記述します。login_nameが一致するユーザ情報がDB上に存在しない場合、ユーザ情報の新規登録処理を行います。一致するユーザがDB上に存在する場合、ユーザ情報の更新処理を行います。	string	○
3	default	同一JSONドキュメント内にあるデフォルト設定の参照先を指定します。この設定はユーザ情報の新規作成時のみ有効です。更新時には無視されます。	object	-
3	password	パスワードを記述します。指定しない場合ランダムなパスワードが設定されます。このパスワードを取得する方法はありません。	string	-

階層	ノード名	説明	型	必須
3	name	name以下にはロケール名をキーとしてさらにネストした first_name, family_name, name_format_typeを記述します。 name_format_typeには "family_first", "family_space_first", "first_space_family" が指定可能です。それぞれのロケール名について、first_name, family_name, name_format_typeの3項目が全て指定されていることが必要です。	object	○
3	email	メールアドレスを指定します。デフォルト値は空で固定です。	string	-
3	is_mail_component_mail_recipient	ユーザが投入したジョブが完了した時にメールコンポーネントが送信するメールを受け取る場合はtrueを指定します。デフォルト設定が有効です。	boolean	-
3	is_error_mail_recipient	ユーザが投入したジョブが失敗した時に、ユーザにメールで通知を行う場合はtrueを指定します。デフォルト設定が有効です。	boolean	-
3	is_login_enable	ログインを有効にする場合はtrueを指定します。デフォルト設定が有効です。	boolean	-
3	is_editable_email	ユーザが自身のメールアドレスを変更できるようにする場合はtrueを指定します。デフォルト設定が有効です。	boolean	-
3	permissions	対ユーザ・グループ権限を指定します。 インポートしたユーザ自身に対するPermit Download権限は自動で割り当てられます。	array of object	-
4	(permission)	-	object	-
5	permission_code	指定されたユーザ・グループに対する権限の種別をコード指定します。 1:Read / 2: Permit Download	int	○
5	login_name	対象ユーザのlogin_nameを指定します。	array of string	*1a
5	group_id	対象グループをIDで指定します。	array of int	*1b
5	group_unique_alias	対象グループをグループ識別名で指定します。	array of string	*1c
3	groups	ユーザの所属グループを、グループ識別名で指定します。	array of object	- (*2a)
3	groups_to_add	当該ユーザをメンバーに追加したいグループを、グループ識別名で指定します。	array of object	- (*2b)
3	groups_to_remove	当該ユーザをメンバーから削除したいグループを、グループ識別名で指定します。	array of object	- (*2b)
3	memo	備考を指定します。デフォルト設定が有効です。	string	-
3	time_zone	タイムゾーン名を指定します。デフォルト設定が有効です。	string	-

階層	ノード名	説明	型	必須
3	language_priority	ユーザごとに利用する言語の優先度をロケール名の配列で順番に指定します。デフォルト設定が有効です。	array of string	-
3	character_code	クライアント文字コード名を指定します。デフォルト設定が有効です。	string	-
3	is_enable_ldap	LDAP認証連携機能を利用する場合はtrueを指定します。	boolean	-
3	ldap_account_name		string	-
3	external_owner	外部システム名を指定します。	string	-
1	user_defaults	新規登録時にデフォルト値として使用する値を指定します。	array of object	-
2	(user_default)	-	object	-
3	\$id	デフォルト値のセットのIDを指定します。Userオブジェクトからはこの名前で参照されます。	string	○
3	is_mail_component_mail_recipient	-	boolean	-
3	is_error_mail_recipient	-	boolean	-
3	is_login_enable	-	boolean	-
3	is_editable_email	-	boolean	-
3	memo	-	string	-
3	time_zone	-	string	-
3	language_priority	-	array of string	-
3	character_code	-	string	-

*1): *1a, *1b, *1cいずれか必須

*2): *2a, *2bの同時指定は不可。ただしgroups_to_addとgroups_to_removeの同時指定は可

以下はサンプルです。

```
{
  "format_version": 1,
  "users": [
    {
      "login_name": "test_user_1",
      "password": "test_user_1",
      "default": {
        "$ref": "user_default_1"
      }
    },
  ],
}
```

```

    "external_owner":
      "LDAP://dc1.example.com/CN=User_1,DC=example,DC=com",
    "name": {
      "ja-JP": {
        "first_name": "テスト用",
        "family_name": "ユーザ",
        "name_format_type": "first_space_family"
      },
      "en-US": {
        "first_name": "for test",
        "family_name": "User",
        "name_format_type": "family_space_first"
      }
    },
    "groups": [
      "group_alias_1",
      "group_alias_2"
    ],
    "language_priority": [
      "ja-JP",
      "en-US",
      "de-DE"
    ],
    "email": "test@example.com",
    "is_login_enable": true,
    "is_editable_email": true,
    "is_error_mail_recipient": true,
    "is_mail_component_mail_recipient": true,
    "time_zone": "Tokyo",
    "character_code": "Shift_JIS"
  },
  {
    "login_name": "test_user_2",
    "default": {
      "$ref": "user_default_1"
    },
    "external_owner":
      "LDAP://dc1.example.com/CN=User_2,DC=example,DC=com",
    "name": {
      "ja-JP": {
        "first_name": "テスト用",
        "family_name": "ユーザ 2",
        "name_format_type": "first_space_family"
      },
      "en-US": {
        "first_name": "for test 2",
        "family_name": "User",
        "name_format_type": "family_space_first"
      }
    }
  }

```

```

    },
    "groups_to_add": [
        "group_alias_1"
    ],
    "groups_to_remove": [
        "group_alias_2"
    ]
  }
],
"user_defaults": [
  {
    "$id": "user_default_1",
    "language_priority": [
      "ja-JP",
      "en-US",
      "de-DE"
    ],
    "is_login_enable": false,
    "is_editable_email": true,
    "is_error_mail_recipient": true,
    "time_zone": "Tokyo",
    "character_code": "Shift_JIS"
  }
]
}

```

12.6.2. グループ

後述する仕様のJSON形式でファイルを作成し、下記のコマンドを実行することで、ASFALIS TransServerのグループの登録および更新が可能です。

```
bundle exec rake ats:groups:import ATS_GROUPS_IMPORT_FILE=C:\path\to\groups.json
```

JSONファイルは以下の形式で指定してください。

階層	ノード名	説明	型	必須
1	format_version	1を指定します。	int	○
1	groups	-	array of object	○
2	(group)	-	object	-
3	unique_alias	グループの識別名を指定します。	string	○
3	external_owner	外部システム名を指定します。	string	-

階層	ノード名	説明	型	必須
3	name	グループの名前を指定します。 登録時は最低1つのロケール名の名前を指定してください。	object	-
3	permissions	対ユーザ・グループ権限を指定します。 インポートしたグループ自身に対する Permit Download権限は自動で割り 当てられます。	array of object	-
4	(permission)	-	object	-
5	permission_code	指定されたユーザ・グループに対する権限の 種別をコード指定します。 1:Read / 2: Permit Download	int	○
5	login_name	対象ユーザのlogin_nameを指定します。	array of string	*1a
5	group_id	対象グループをIDで指定します。	array of int	*1b
5	group_unique_alias	対象グループをグループ識別名で指定します。 。	array of string	*1c
4	\$(locale_name)	-	string	*L
3	child_groups	グループのメンバーとなるグループを、グル ープ識別名で指定します。	array of string	- (*2a)
3	child_groups_to_ad d	当該グループのメンバーとして追加したいグル ープを、グループ識別名で指定します。	array of object	- (*2b)
3	child_groups_to_re move	当該グループのメンバーから削除したい所属 グループを、グループ識別名で指定します。	array of object	- (*2b)
3	child_users	当該グループのメンバーとなるユーザを、ユ ーザのログイン名で指定します。	array of string	- (*3a)
3	child_users_to_add	当該グループのメンバーに追加したいグルー プを、ユーザのログイン名で指定します。	array of object	- (*3b)
3	child_users_to_rem ove	当該グループのメンバーから削除したいユー ザを、ユーザのログイン名で指定します。	array of object	- (*3b)
3	is_root	グループをRootにする場合はtrueを指定し ます。	boolean	-

*1): *1a, *1b, *1cいずれか必須

*2): *2a, *2bの同時指定は不可。ただしchild_groups_to_addとchild_groups_to_removeの同時指定は可

*3): *3a, *3bの同時指定は不可。ただしchild_users_to_addとchild_users_to_removeの同時指定は可

*L): 少なくとも1つのロケールが必須

※※ \$(locale_name) : ja-JP、en-US、de-DE、zh-CNのいずれかのロケール名

以下はサンプルです。

```
{
```

```
"format_version": 1,
"groups": [
  {
    "unique_alias": "Group_1",
    "external_owner":
      "LDAP://dc1.example.com/CN=Group_1,DC=example,DC=com",
    "permissions": [
      {
        "login_name": [
          "user_1"
        ],
        "group_unique_alias": [
          "ScenarioAdmins",
          "NamedUsers"
        ],
        "permission_code": 1
      }
    ],
    "child_groups": [
      "Group2"
    ],
    "child_users_to_add": [
      "user_name_1"
    ],
    "child_users_to_remove": [
      "user_name_2"
    ],
    "name": {
      "ja-JP": "グループ1",
      "en-US": "Group_1"
    }
  },
  {
    "unique_alias": "Group_2",
    "external_owner":
      "LDAP://dc1.example.com/CN=Group_2,DC=example,DC=com",
    "child_groups": [
      "Group_3"
    ],
    "name": {
      "ja-JP": "グループ2",
      "en-US": "Group_2"
    }
  },
  {
    "unique_alias": "Group_3",
    "external_owner":
      "LDAP://dc1.example.com/CN=Group_3,DC=example,DC=com",
    "child_groups_to_add": [
```

```

    "group_alias_1"
  ],
  "child_groups_to_remove": [
    "group_alias_2"
  ],
  "child_users": [
    "user_1"
  ],
  "name": {
    "ja-JP": "グループ3",
    "en-US": "Group_3"
  }
}
]
}

```

12.6.3. 業務種別

後述する仕様のJSON形式でファイルを作成し、下記のコマンドを実行することで、ASFALIS TransServer の業務種別の登録および更新が可能です。

```

bundle exec rake ats:job_lot_templates:import
ATS_JOB_LOT_TEMPLATES_IMPORT_FILE=C:\path\to\job_lot_templates.json

```

JSONファイルは以下の形式で指定してください。

階層	ノード名	説明	型	必須
1	format_version	1を指定します。	int	○
1	job_lot_templates	-	array of object	○
2	(job_lot_template)	-	object	-
3	id	既存の業務種別を更新する場合は、更新する業務種別のIDを指定します。業務種別を新規に登録する場合は、この項目を指定しないでください。	int	更
3	name	業務種別の名前を指定します。登録時は最低1つのロケール名の名前を指定します。	object	登
4	\$(locale_name)	-	string	*L
3	priority	1: 低、2: 中、3: 高、4: 最高、省略時: 低	int	-
3	login_name	業務種別の作成者のlogin_nameを指定します。	string	○
3	external_owner	外部システム名を指定します。	string	-
3	is_erased	業務種別を削除する時trueを指定します	boolean	-

階層	ノード名	説明	型	必須
3	scenarios	業務種別で使用可能なシナリオを指定します。省略時は変更しません。	array of object	-
4	(scenario)	-	object	-
5	id	シナリオをIDで指定します。	int	*1a
5	unique_alias	シナリオを識別子で指定します。	string	*1b
3	permissions	ユーザ・グループに対し業務種別から作られるオペレーションに割り当てられる権限の最大値を指定します。 JobAdminsには自動的にEdit/Delete権限が割り当てられます。	array of object	-
4	(permission)	-	object	-
5	permission_code	作成するオペレーションに対する権限の種別をコード指定します。 1: Read、2: Download、3: Edit/Delete	int	○
5	login_name	対象ユーザのlogin_nameを指定します。	array of string	*2a
5	group_id	対象グループをIDで指定します。	array of int	*2b
5	group_unique_alias	対象グループをグループ識別名で指定します。	array of string	*2c
3	job_lot_property_definitions	業務種別に割り当てる業務属性定義を指定します。	array of object	-
4	(job_lot_property_definition)	-	object	-
5	key_name	業務種別内で一意に定まる文字列を指定します。見つかった場合は該当する業務属性を更新します。見つからなかった場合は新規登録します。	string	○
5	label	業務属性のラベルを指定します。登録時は最低1つのロケール名の説明を指定してください。	object	登
6	\$(locale_name)	-	string	*L
5	description	業務属性の説明を指定します。登録時は最低1つのロケール名の説明を指定してください。	object	登
6	\$(locale_name)	-	string	*L
5	value_constraint	値の制約を指定します。	string	-
5	value_type	業務属性の型を指定します。登録時に指定必須です。更新時には指定しないでください。型は変更不可です。	string	登
5	value	業務属性の値を指定します。ファイル型の場合は、アップロードするファイルをフルパスで指定します。	string	登

階層	ノード名	説明	型	必須
5	is_editable	業務属性の『編集可』を有効にする場合にtrueを指定します。業務属性をオペレーション作成時に編集できるようになります。指定がない場合はデフォルトでfalseが指定されます。	boolean	-
5	is_visible	業務属性の『ユーザ可視』を有効にする場合はtrueを指定します。業務属性をオペレーション作成時に閲覧できるようになります。指定がない場合はデフォルトでfalseが指定されます。	boolean	-
5	is_always_editable	業務属性の『登録後編集可』を有効にする場合はtrueを指定します。業務属性をオペレーション詳細画面で編集できるようになります。指定がない場合はデフォルトでfalseが指定されます。	boolean	-
5	is_searchable	業務属性の『検索可』の状態を指定します。trueの場合はオペレーション詳細検索時にこの業務属性を検索条件として利用できます。指定がない場合はデフォルトでfalseが指定されます。	boolean	-
5	values	業務属性の値の選択肢を指定します。ラジオボタン型・プルダウン型のみ登録時必須です。それ以外の型の属性には指定しないでください。更新時は登録済のデータも併せて指定してください。指定されない選択肢は削除されます。	array of object	*3
6	(value)	-	object	-
7	value	業務属性の値の選択肢に対応する属性値を指定します。	string	○
7	description	業務属性の値の選択肢の説明を指定します。登録時は最低1つのロケール名の説明を指定してください。	object	登
8	\$(locale_name)	-	string	*L
5	subwindow_url	サブウィンドウのURLを指定します。サブウィンドウ型のみ登録可能です。それ以外の型では指定しないでください。	string	-

*1): *1a, *1b いずれか必須

*2): *2a, *2b, *2c いずれか必須

*3): ラジオボタン・プルダウン型の登録時のみ必須

*L): 少なくとも1つのロケールが必須

※※ \$(locale_name): ja-JP、en-US、de-DE、zh-CNのいずれかのロケール名

以下はサンプルです。

```
{
  "format_version": 1,
  "job_lot_templates": [
    {
```

```

"name": {
  "ja-JP": "業務種別登録テスト",
  "en-US": "registering Operation Type test"
},
"external_owner": "CN=JobLotTemplate_1,DC=example,DC=com",
"priority": 4,
"login_name": "admin",
"is_erased": false,
"scenarios": [
  {
    "id": 500000
  }
],
"permissions": [
  {
    "group_unique_alias": [
      "Everyone"
    ],
    "permission_code": 1
  }
],
"job_lot_property_definitions": [
  {
    "key_name": "text1",
    "value_type": "text",
    "label": {
      "ja-JP": "テキスト1",
      "en-US": "Text1"
    },
    "description": {
      "ja-JP": "これはテキスト1です",
      "en-US": "This is Text1."
    },
    "value": "aaa",
    "is_editable": true,
    "is_visible": true,
    "is_always_editable": false,
    "is_searchable": true,
    "value_constraint": "input_not_blank"
  },
  {
    "key_name": "subwindow1",
    "value_type": "subwindow",
    "label": {
      "ja-JP": "サブウィンドウ1",
      "en-US": "Subwindow1"
    },
    "description": {
      "ja-JP": "これはサブウィンドウ1です",

```

```

        "en-US": "This is Subwindow1."
    },
    "value": "subsubsub",
    "is_editable": false,
    "is_visible": true,
    "is_always_editable": false,
    "is_searchable": true,
    "subwindow_url":
"http://localhost:3000/samples/clientAPI_sample/subwindow_page.html"
    },
    {
        "key_name": "radio1",
        "value_type": "radiobutton",
        "label": {
            "ja-JP": "ラジオ1",
            "en-US": "Radio1"
        },
        "description": {
            "ja-JP": "これはラジオボタン1です",
            "en-US": "This is Radiobutton1."
        },
        "value": "2",
        "is_editable": true,
        "is_visible": true,
        "is_always_editable": true,
        "is_searchable": true,
        "values": [
            {
                "value": "1",
                "description": {
                    "ja-JP": "これはいちです",
                    "en-US": "This is one"
                }
            },
            {
                "value": "2",
                "description": {
                    "ja-JP": "これはにです",
                    "en-US": "This is two"
                }
            },
            {
                "value": "3",
                "description": {
                    "ja-JP": "これはさんです",
                    "en-US": "This is three"
                }
            }
        ]
    }
]

```

```

    }
  ]
}
]
}

```

12.6.4. シナリオ

後述する仕様のJSON形式でファイルを作成し、下記のコマンドを実行するか、シナリオ一覧画面の『インポート』ボタンからファイルをアップロードすることで、ASFALIS TransServer のシナリオの登録および更新ができます。

```

bundle exec rake ats:scenarios:import
ATS_SCENARIOS_IMPORT_FILE=C:\path\to\scenarios.json

```

シナリオ内で利用するコンポーネントテンプレート・カスタムコンポーネントも同時にインポート可能です。

JSONファイルは以下の形式で指定してください

階層	ノード名	説明	型	必須
1	format_version	1を指定します。	int	○
1	scenarios	-	array of object	○
2	(scenario)	-	object	-
3	id	更新時のみ指定します。既存のシナリオを更新する場合は更新対象のシナリオIDを指定してください。	int	*1
3	unique_alias	シナリオの識別名を指定します。更新時は更新対象のシナリオ識別名を指定してください。新規登録時は既存のシナリオと重複しない識別名を指定してください。指定がない場合は、重複しないランダムな識別名で新規登録されます。	string	*1
3	name	シナリオの名前を指定します。登録時は最低1つのロケール名の名前を指定してください。	object	-
4	\$(locale_name)	-	string	*L
3	description	シナリオの説明文を指定します。	object	-
4	\$(locale_name)	-	string	-
3	login_name	シナリオの作成者をユーザのログイン名で指定します。	string	○
3	external_owner	外部システム名を入力します。	string	-

階層	ノード名	説明	型	必須
3	is_erased	シナリオを削除済みにする場合はtrueを指定します。指定がない場合はfalseがデフォルトで指定されます。	boolean	-
3	permissions	-	array of object	-
4	(permission)	-	-	-
5	permission_code	0または1を指定します。1を指定すると、後述の対象ユーザ・グループに対してシナリオの参照/使用権限が有効になります。	int	*2
5	login_name	対象ユーザをlogin_nameの配列で指定します。	array of string	-
6	(login_name)	-	string	-
5	group_id	対象グループをグループIDの配列で指定します。	array of int	-
6	(group_id)	-	int	-
5	group_unique_alias	対象グループをグループ識別名の配列で指定します。	array of string	-
6	(group_unique_aliases)	-	string	-
3	job_lot_templates	シナリオを使用可能な業務種別を指定します。	object	-
4	id	対象業務種別をIDの配列で指定します。	array of int	-
4	unique_alias	対象業務種別を識別名の配列で指定します。	array of string	-
3	component_instances	-	array of object	○
4	(component_instance)	-	object	-
5	name	コンポーネントの名前を指定します。同種のコンポーネントが存在する場合は、上からiges2enf, iges2enf2, iges2enf3…のように命名します。	string	○
5	component_internal_name	コンポーネントの内部名を指定します。こちらは同様のコンポーネントが存在しても、同じものを指定してください。	string	○
5	template_unique_aliases	コンポーネントパラメータを使用する場合に、使用するコンポーネントテンプレートの識別名を指定します。後述のparametersでパラメータを設定する場合は指定しないでください。	string	-
5	parameters	コンポーネントのそれぞれのパラメータをオブジェクトの配列で指定します。何も指定しない場合は空にしておきます。	array of object	*3
6	(parameter)	-	object	-

階層	ノード名	説明	型	必須
7	key_name	パラメータのキー名を指定します。	string	*4
7	value	パラメータの値を指定します。指定する値によって型が変化します。	-	*4 *6
7	is_mandatory	『ジョブ投入時に設定必須』を有効にする場合はtrueを指定します。	boolean	*4
7	is_editable	『あとでジョブ投入時に指定』を有効にする場合はtrueを指定します。	boolean	*4
7	is_visible	『ユーザに表示する』を有効にする場合はtrueを指定します。	boolean	*4
5	sub_component_instances	サブコンポーネントを指定します。	array of object	*5
6	(sub_component_instance)	-	object	-
7	sub_process_name	サブプロセス名を指定します。サブコンポーネント名は『コンポーネント名_サブプロセス名』で構成されており、後半のサブプロセス名のみを記述してください。	string	*5
7	parameters	サブコンポーネントのそれぞれのパラメータを、ハッシュの配列で指定します。何も指定しない場合は空にしておきます。	array of object	*5
8	(parameter)	-	object	-
9	key_name	パラメータのキー名を指定します。	string	*5
9	value	パラメータの値を指定します。指定する値によって型が変化します。	-	*5 *6
9	file_path	ファイル型のパラメータを指定します。指定する値によって型が変化します。	-	*5 *6
9	is_mandatory	『ジョブ投入時に設定必須』を有効にする場合は、trueを指定します。	boolean	*5
9	is_editable	『あとでジョブ投入時に指定』を有効にする場合は、trueを指定します。	boolean	*5
9	is_visible	『ユーザに表示する』を有効にする場合は、trueを指定します。	boolean	*5
3	component_instance_relations	コンポーネントの接続情報を指定します。	array of object	○
4	(component_instance_relation)	-	object	-
5	source_component_instance	接続元のコンポーネント(component_instance)のnameを指定します。	string	○
5	source_component_parameter	接続元のコンポーネントの入出力パラメータのうち、IO TypeがoutであるKeyを指定します。入出力パラメータはコンポーネント詳細ページにて確認してください。	string	○

階層	ノード名	説明	型	必須
5	target_component_instance	接続先のコンポーネント(component_instance)のnameを指定します。	string	○
5	target_component_parameter	接続先のコンポーネントの入出力パラメータのうち、IO TypeがinであるKeyを指定します。入出力パラメータはコンポーネント詳細ページにて確認してください。	string	○
1	component_templates	「 12.7.4, “コンポーネントテンプレート” 」を参照。	array of object	-
1	components	「 12.6.6, “カスタムコンポーネント” 」を参照。	array of object	-

*1): いずれも指定がなければ、ランダムな識別名でシナリオが新規作成されます。

*2): permissions指定時必須。

*3): 必須パラメータを持つコンポーネントの場合は空を指定することができません。

*4): parameters指定時必須。

*5): 簡略化、ポリゴン最適化指定時に必須。

*6): ファイルを扱うパラメータにはfile_pathを、それ以外はvalueを指定します。

*L): 少なくとも1つのロケールが必須。

※※ \$(locale_name): ja-JP、en-US、de-DE、zh-CNのいずれかのロケール名

パラメータの指定例は以下の通りです。

- テキスト

valueに設定値を指定します。『パラメータの制約』に従って、数値または文字列を指定してください。

```
{
  "key_name": "body",
  "value": "message in body",
  ...
},
{
  "key_name": "StitchTol",
  "value": 2,
  ...
}
```

- プルダウン、コンボボックス

valueに設定項目ごとに決められたパラメータを指定します。

```
{
  "key_name": "to_cad",
  "value": "iges",
  ...
},
{
  "key_name": "CADVersion",
  "value": "B23",
  ...
}
```

- ラジオボタン
選択肢に割り当てられている値をvalueに指定します。

```
{
  "key_name": "CleanSelfIntersectingLoop",
  "value": 2,
  ...
}
```

- 真偽値
valueにtrueまたはfalseを指定します。指定しなかった場合はfalseを指定した場合と同じ動作になります。

```
{
  "key_name": "to_job_user",
  "value": true,
  ...
}
```

- ビットフラグ
各チェックボックスに割り振られている数値のうち、チェックされている値の総和をvalueに指定します。

```
{
  "key_name": "RemoveDuplicateSurface",
  "value": 7,
  ...
}
```

- 単数パラメータ（転送先指定）
転送先の外部ファイルサーバを指定する場合はファイルサーバのIDとパスを指定します。ファイルはピリオドで指定した箇所に転送されます。

```
{
  "key_name": "result_base_directory",
  "value": {
    "file_server": "1",
    "path": "dir/subdir/."
  },
  ...
}
```

- 複数パラメータ（例: 「Windowsファイルサーバへ転送」の「コピーするファイルのキー」）
パラメータの値が複数ある場合は、パラメータを配列にして指定します。

```
{
  "key_name": "additional_file_keys",
  "value": ["key1", "key2"],
  ...
}
```

- 単数ファイル型パラメータ
ファイル型のパラメータは、file_pathにJSONファイルがあるフォルダからの相対パスを指定します。ブラウザからアップロードする場合は、フォルダ構造を維持したままzipファイルにしてアップロードします。

```
{
  "key_name": "mapfile",
  "file_path": "./dir/mapfile_sample.xml",
  ...
}
```

- 複数ファイル型パラメータ
複数ファイルを指定可能なパラメータは、file_pathにJSONファイルがあるフォルダからの相対パスを配列にして指定します。ブラウザからアップロードする場合は、フォルダ構造を維持したままzipファイルにしてアップロードします。

```
{
  "key_name": "attached_files",
  "file_path": [
    "./dir/01/test1.txt",
    "./dir/02/test1.txt"
  ],
  ...
}
```

- キー付ファイル型パラメータ

キー付ファイル型のパラメータは、キー名とJSONファイルがあるフォルダからの相対パスからなるオブジェクトを配列にして指定します。UIからアップロードする場合は、フォルダ構造を維持したままzipファイルにしてアップロードします。

```
{
  "key_name": "parameterfile_list",
  "value": [
    {
      "key_name" : "key1",
      "file_path" : "./dir/key1/test1.txt"
    },
    {
      "key_name" : "key2",
      "file_path" : "./dir/key2/test1.txt"
    }
  ],
  ...
}
```

- パスワード型パラメータ

パスワードは暗号化して保存されているため、インポートするパラメータには暗号化されたパスワードとsaltを指定します。

```
{
  "key_name": "password",
  "value": "2ff9ad2828bc4e2fdaee413e559a22b9",
  "salt": "U13MF9r4",
  ...
}
```

以下は「CATIAV5インポート-Windowsファイルサーバへ転送-メール送信」のシナリオをインポートするためのサンプルです。

```
{
  "format_version": 1,
  "scenarios": [
    {
      "unique_alias": "Scenario_1",
      "is_erased": false,
      "external_owner": "CN=Scenario_1,DC=example,DC=com",
      "name": {
        "ja-JP": "シナリオ1",
        "en-US": "Scenario_1"
      },
    },
  ],
}
```

```

"description": {
  "ja-JP": "これはシナリオ1です",
  "en-US": "This is Scenario_1"
},
"permissions": [
  {
    "login_name": [
      "user_1"
    ],
    "group_unique_alias": [
      "ScenarioAdmins",
      "NamedUsers"
    ],
    "permission_code": 1
  }
],
"component_instances": [
  {
    "name": "cativ52enf",
    "component_internal_name": "cativ52enf",
    "parameters": [
      {
        "key_name": "ConvertPartwise",
        "value": "0",
        "is_mandatory": true,
        "is_editable": false,
        "is_visible": true
      }
    ]
  },
  {
    "name": "copy_to_cifs",
    "component_internal_name": "copy_to_cifs",
    "parameters": [
      {
        "key_name": "result_base_directory",
        "value": {
          "file_server": "1",
          "path": "."
        },
        "is_mandatory": true,
        "is_editable": true,
        "is_visible": true
      }
    ]
  },
  {
    "name": "mail",
    "component_internal_name": "mail",

```

```

      "parameters": [
        {
          "key_name": "from",
          "value": "test@mail.com",
          "is_mandatory": true,
          "is_editable": true,
          "is_visible": true
        },
        {
          "key_name": "body",
          "value": "body",
          "is_mandatory": true,
          "is_editable": true,
          "is_visible": true
        },
        {
          "key_name": "subject",
          "value": "subject",
          "is_mandatory": true,
          "is_editable": true,
          "is_visible": true
        }
      ]
    },
    ],
    "component_instance_relations": [
      {
        "source_component_instance": null,
        "source_component_parameter": null,
        "target_component_instance": "catia52enf",
        "target_component_parameter": "inputfile"
      },
      {
        "source_component_instance": "catia52enf",
        "source_component_parameter": "outputfile",
        "target_component_instance": "copy_to_cifs",
        "target_component_parameter": "inputfile"
      },
      {
        "source_component_instance": "copy_to_cifs",
        "source_component_parameter": "outputfile",
        "target_component_instance": "mail",
        "target_component_parameter": "inputfile"
      },
      {
        "source_component_instance": "mail",
        "source_component_parameter": "outputfile",
        "target_component_instance": null,
        "target_component_parameter": null
      }
    ]
  }
}

```



```

    }
  ]
}
],
"component_templates": [
  *省略可。詳しくはコンポーネントテンプレートの項をご覧ください。
],
"components": [
  *省略可。詳しくはカスタムコンポーネントの項をご覧ください。
]
}

```

12.6.5. コンポーネントテンプレート

後述する仕様のJSON形式でファイルを作成し下記のコマンドを実行すること
で、ASFALIS TransServer のコンポーネントテンプレートの登録および更新ができます。

```

bundle exec rake ats:component_templates:import
ATS_COMPONENT_TEMPLATES_IMPORT_FILE=C:\path\to\component_templates.json

```

JSONファイルは以下の形式で指定してください。

階層	ノード名	説明	型	必須
1	format_version	1を指定します。	int	○
1	component_templates	-	array of object	○
2	(component_template)	-	object	-
3	unique_alias	コンポーネントテンプレートの識別名を指定します。	string	○
3	external_owner	外部システム名を入力します。	string	-
3	name	コンポーネントテンプレートの名前を指定します。登録時は最低1つのロケール名の名前を指定してください。	object	-
4	\$(locale_name)	-	string	*L
3	description	コンポーネントテンプレートの説明文を指定します。	object	-
4	\$(locale_name)	-	string	*L
3	component_internal_name	コンポーネントの内部名を指定します。	string	○
3	parameters	コンポーネントのそれぞれのパラメータをオブジェクトの配列で指定します。何も指定しない場合は空にしておきます。	array of object	*1

階層	ノード名	説明	型	必須
4	(parameter)	-	object	-
5	key_name	パラメータのキー名を指定します。	string	*1
5	value	パラメータの値を指定します。指定する値によって型が変化します。	-	*1
5	is_mandatory	『ジョブ投入時に設定必須』を有効にする場合はtrueを指定します。	boolean	*1
5	is_editable	『あとでジョブ投入時に指定』を有効にする場合はtrueを指定します。	boolean	*1
5	is_visible	『ユーザに表示する』を有効にする場合はtrueを指定します。	boolean	*1
3	sub_component_instances	サブコンポーネントを指定します。	array of object	*2
4	(sub_component_instance)	-	object	-
5	sub_process_name	サブプロセス名を指定します。サブコンポーネント名は『コンポーネント名_サブプロセス名』で構成されており、後半のサブプロセス名のみを記述してください。	string	*2
5	parameters	サブコンポーネントのそれぞれのパラメータを、ハッシュの配列で指定します。何も指定しない場合は空にしておきます。	array of object	*2
6	(parameter)	-	object	-
7	key_name	パラメータのキー名を指定します。	string	*2
7	value	パラメータの値を指定します。指定する値によって型が変化します。	-	*2
7	is_mandatory	『ジョブ投入時に設定必須』を有効にする場合は、trueを指定します。	Boolean	*2
7	is_editable	『あとでジョブ投入時に指定』を有効にする場合は、trueを指定します。	Boolean	*2
7	is_visible	『ユーザに表示する』を有効にする場合は、trueを指定します。	Boolean	*2

*1): コンポーネントに設定が必須なパラメータが含まれている場合は空を指定することができません。

*2): 簡略化、ポリゴン最適化指定時に必須。

*L): 少なくとも1つのロケールが必須。

※※ \$(locale_name): ja-JP、en-US、de-DE、zh-CNのいずれかのロケール名

以下はサンプルです。

```
{
  "format_version": 1,
  "component_templates": [
    {
```

```

    "unique_alias": "ComponentTemplate_1",
    "external_owner": "CN=ComponentTemplate_1,DC=example,DC=com",
    "name": {
      "ja-JP": "コンポーネントテンプレート1",
      "en-US": "ComponentTemplate_1"
    },
    "description": {
      "ja-JP": "これはコンポーネントテンプレート1です",
      "en-US": "This is ComponentTemplate_1"
    },
    "component_internal_name": "mail",
    "parameters": [
      {
        "key_name": "from",
        "value": "",
        "is_mandatory": true,
        "is_editable": true,
        "is_visible": true
      },
      {
        "key_name": "to",
        "value": "",
        "is_mandatory": false,
        "is_editable": true,
        "is_visible": true
      },
      {
        "key_name": "subject",
        "value": "",
        "is_mandatory": true,
        "is_editable": true,
        "is_visible": true
      },
      {
        "key_name": "body",
        "value": "test message",
        "is_mandatory": true,
        "is_editable": true,
        "is_visible": true
      }
    ]
  },
  {
    "unique_alias": "ComponentTemplate_2",
    "external_owner": "CN=ComponentTemplate_2,DC=example,DC=com",
    "name": {
      "ja-JP": "コンポーネントテンプレート2",
      "en-US": "ComponentTemplate_2"
    },
  },

```

```

    "description": {
      "ja-JP": "これはコンポーネントテンプレート2です",
      "en-US": "This is ComponentTemplate_2"
    },
    "component_internal_name": "simplify",
    "parameters": [],
    "sub_component_instances": [
      {
        "sub_process_name": "removeroundhole",
        "parameters": [
          {
            "key_name": "AvoidInterferenceWithinPart",
            "value": "1",
            "is_mandatory": false,
            "is_editable": true,
            "is_visible": true
          },
          {
            "key_name": "HolePenetrationType",
            "value": "1",
            "is_mandatory": false,
            "is_editable": true,
            "is_visible": true
          }
        ]
      }
    ]
  }
}

```

12.6.6. カスタムコンポーネント

後述する仕様のJSON形式でファイルを作成し下記のコマンドを実行することで、ASFALIS TransServer のカスタムコンポーネントの登録および更新ができます。

```

bundle exec rake ats:components:import
ATS_COMPONENTS_IMPORT_FILE=C:\path\to\components.json

```

JSONファイルは以下の形式で指定してください。

階層	ノード名	説明	型	必須
1	format_version	1を指定します。	int	○
1	components	-	array of object	-
2	(component)	-	object	-

階層	ノード名	説明	型	必須
3	internal_name	カスタムコンポーネントの内部名を指定します。	string	○
3	external_owner	外部システム名を指定します。	string	-
3	execution_type	スクリプトコンポーネントの場合は2を指定します。それ以外は1を指定します。	int	○
3	component_type_id	コンポーネントがCADを書き出す場合は1を、中間ファイルを読み込んでCADを出力する場合は2を指定します。どちらでもない場合はNULLを指定します。	int	-
3	order	コンポーネントを並べて表示する際の表示順序を指定します。指定しない場合は、全コンポーネント中のorderの最大値+1が指定されます。	int	-
3	is_enabled	コンポーネントを無効にしておきたい場合はfalseを指定します。指定がない場合はデフォルトでtrueが指定されます。	boolean	-
3	name	コンポーネントの名前を指定します。登録時は最低1つのロケール名の名前を指定してください。	object	-
4	\$(locale_name)	-	string	*L
3	description	コンポーネントの説明文を指定します。	object	-
4	\$(locale_name)	-	string	-
3	input_parameters	コンポーネントへの入力を設定します。	array of object	-
4	(input_parameter)	-	object	-
5	key_name	パラメータのキー名を指定します。	string	○
5	name	パラメータの名前を指定します。登録時は最低1つのロケール名の名前を指定してください。	object	-
6	\$(locale_name)	-	string	*L
5	description	パラメータの説明文を指定します。	object	-
6	\$(locale_name)	-	string	-
5	is_mandatory	指定必須パラメータにする場合はtrueを指定します。指定がない場合はデフォルトでfalseが指定されます。	boolean	-
5	mime_type	パラメータのMIME_TYPEを指定します。指定しない場合はデフォルトでNULLが指定されます。	string	-
5	value_constraint_id	パラメータに対して、制約を設ける場合は、制約のIDを指定します。指定しない場合はデフォルトでNULLが指定されます。	int	-

階層	ノード名	説明	型	必須
5	value_type	期待されるパラメータの種類を指定します。指定できるパラメータはnumber, text, textarea, radiobutton, file, pulldownです。	string	-
5	values	パラメータ設定値に関する情報を配列で指定します。	array of object	-
6	(value)	-	object	-
7	value	パラメータの設定に用いる設定値を指定します。	array of object	○
7	description	パラメータ設定値に関する説明を指定します。	object	-
8	\$(locale_name)	-	string	*L
7	is_enabled	コンポーネントパラメータを無効化する場合はfalseを指定します。指定がない場合は、デフォルトでtrueが指定されます。	boolean	-

*L): 少なくとも1つのロケールが必須

※※ \$(locale_name): ja-JP、en-US、de-DE、zh-CNのいずれかのロケール名

下記はサンプルです。

```
{
  "format_version": 1,
  "components": [
    {
      "internal_name": "CustomComponent_1",
      "execution_type": 3,
      "component_type_id": null,
      "order": 111,
      "is_enabled": true,
      "external_owner": "CN=CustomComponent_1,DC=example,DC=com",
      "name": {
        "ja-JP": "カスタムコンポーネント1",
        "en-US": "CustomComponent_1"
      },
      "description": {
        "ja-JP": "これはカスタムコンポーネント1です",
        "en-US": "This is CustomComponent_1"
      },
      "input_parameters": [
        {
          "key_name": "input_file",
          "name": {
            "en-US": "File of translation",
            "ja-JP": "変換対象モデルファイル"
          }
        }
      ]
    }
  ]
}
```

```

        "description": {
            "en-US": "File of translation",
            "ja-JP": "変換対象モデルファイル"
        },
        "value_type": "file",
        "is_mandatory": true,
        "mime_type": "application/octet-stream"
    },
],
"output_parameters": [
    {
        "key_name": "output_file",
        "name": {
            "en-US": "Executing result file",
            "ja-JP": "変換後モデルファイル"
        },
        "description": {
            "en-US": "Executing result file",
            "ja-JP": "変換後モデルファイル"
        },
        "value_type": "file",
        "is_mandatory": true,
        "mime_type": "application/octet-stream"
    },
],
"log_parameters": [
    {
        "key_name": "log_file",
        "name": {
            "en-US": "Log File",
            "ja-JP": "変換ログファイル"
        },
        "description": {
            "en-US": "Log File",
            "ja-JP": "変換ログファイル"
        },
        "value_type": "file",
        "is_mandatory": true
    },
    {
        "key_name": "status_file",
        "name": {
            "en-US": "Status File",
            "ja-JP": "ステータス出力ファイル"
        },
        "description": {
            "en-US": "Status File",
            "ja-JP": "ステータス出力ファイル"
        },
    },

```

```

        "value_type": "file",
        "is_mandatory": true
    }
],
"component_parameters": [
    {
        "key_name": "output_file_name",
        "name": {
            "en-US": "Executing result file name",
            "ja-JP": "変換後モデルファイル名"
        },
        "description": {
            "en-US": "Specify executing result file name",
            "ja-JP": "変換後モデルファイル名を指定します"
        },
        "value_type": "text",
        "is_mandatory": true
    },
    {
        "key_name": "input_format_name",
        "name": {
            "en-US": "Input format name",
            "ja-JP": "入力フォーマット名"
        },
        "description": {
            "en-US": "Specify input format name",
            "ja-JP": "入力フォーマット名を指定します"
        },
        "value_type": "text"
    },
    {
        "key_name": "SlaveNodeName",
        "name": {
            "en-US": "SlaveNode Name",
            "ja-JP": "Slave Node指定"
        },
        "description": {
            "en-US": "Specify machine name and port number of the Slave
Node where Component is executed; input \"(machine_name):(port_number)\",
            "ja-JP": "コンポーネントを実行するSlave
Nodeのマシン名とポート番号を指定します。\" (マシン名) : (ポート番号) \"を入力。"
        },
        "value_type": "text"
    },
    {
        "key_name": "output_format_name",
        "name": {
            "en-US": "Output format name",
            "ja-JP": "出力フォーマット名"

```



```

    },
    "description": {
      "en-US": "Specify output format name",
      "ja-JP": "出力フォーマット名を指定します"
    },
    "value_type": "pulldown",
    "is_customized": true,
    "values": [
      {
        "value": "application/proe",
        "description": {
          "en-US": "proe",
          "ja-JP": "proe"
        },
        "is_enabled": true
      },
      {
        "value": "application/inventor",
        "description": {
          "en-US": "inv",
          "ja-JP": "inv"
        },
        "is_enabled": true
      }
    ]
  },
  {
    "key_name": "exc_opt_param1",
    "name": {
      "en-US": "PARAM1",
      "ja-JP": "PARAM1"
    },
    "description": {
      "en-US": "description of PARAM1",
      "ja-JP": "PARAM1の説明です"
    },
    "value_type": "text"
  },
  {
    "key_name": "exc_opt_param2",
    "name": {
      "en-US": "PARAM2",
      "ja-JP": "PARAM2"
    },
    "description": {
      "en-US": "description of PARAM2",
      "ja-JP": "PARAM2の説明です"
    },
    "value_type": "text"
  }

```

```

    },
    {
      "key_name": "exc_opt_param3",
      "name": {
        "en-US": "PARAM3",
        "ja-JP": "PARAM3"
      },
      "description": {
        "en-US": "description of PARAM3",
        "ja-JP": "PARAM3の説明です"
      },
      "value_type": "text"
    },
    {
      "key_name": "custom_param1",
      "name": {
        "en-US": "CUSTOM PARAM1",
        "ja-JP": "CUSTOM PARAM1"
      },
      "description": {
        "en-US": "description of CUSTOM PARAM1",
        "ja-JP": "CUSTOM PARAM1の説明です"
      },
      "value_type": "text"
    },
    {
      "key_name": "custom_param2",
      "name": {
        "en-US": "CUSTOM PARAM2",
        "ja-JP": "CUSTOM PARAM2"
      },
      "description": {
        "en-US": "description of CUSTOM PARAM2",
        "ja-JP": "CUSTOM PARAM2の説明です"
      },
      "value_type": "text"
    }
  ]
}
]
}

```

12.7. エクスポート機能

ASFALIS TransServer内で取り扱う各情報をJSON形式で出力する機能です。コンソールを操作して利用します。一部は管理機能の画面からも利用することができます。エクスポートされるファイルはインポート機能で用いるJSON形式と同じ形式です。

12.7.1. ユーザ

下記のコマンドを実行することで、ASFALIS TransServerに登録されている全てのユーザ情報と言語優先度のデフォルト値が指定したパスのファイルにJSON形式でエクスポートされます(改行は不要です)。

```
bundle exec rake ats:users:export ATS_USERS_EXPORT_FILE=C:\path\to\users.json  
INCLUDE_GROUP=TRUE
```

INCLUDE_GROUPの指定は任意です。指定しない場合、INCLUDE_GROUP=FALSEとして扱います。また、ユーザー一覧画面からエクスポートしたいユーザにチェックを付けて『エクスポート』ボタンを押すと、ブラウザを通じてJSON形式のエクスポートファイルをダウンロードすることができます。

12.7.2. グループ

下記のコマンドを実行することで、ASFALIS TransServerに登録されている全てのグループが、指定したパスのファイルにJSON形式でエクスポートされます(改行は不要です)。

```
bundle exec rake ats:groups:export ATS_GROUPS_EXPORT_FILE=C:\path\to\groups.json
```

なお、デフォルトグループ (Administrators, ScenarioAdminsなど) はエクスポートの対象となりません。

12.7.3. シナリオ

下記のコマンドを実行することで、ASFALIS TransServerに登録されている全てのシナリオが、指定したパスのファイルにJSON形式でエクスポートされます(改行は不要です)。

```
bundle exec rake ats:scenarios:export  
ATS_SCENARIOS_EXPORT_FILE=C:\path\to\scenarios.json
```

引数にID_LISTやUNIQUE_ALIAS_LISTを指定することで、エクスポートするシナリオをIDや識別名で絞り込むことができます。これらの引数は両方同時に指定することもできます。

```
bundle exec rake ats:scenarios:export  
ATS_SCENARIOS_EXPORT_FILE=C:\path\to\scenarios.json ID_LIST="1,2,3"  
bundle exec rake ats:scenarios:export  
ATS_SCENARIOS_EXPORT_FILE=C:\path\to\scenarios.json  
UNIQUE_ALIAS_LIST="Scenario_1,Scenario_2"
```

また、シナリオ一覧画面からエクスポートしたいシナリオにチェックを付けて『エクスポート』ボタンを押すと、ブラウザを通じてJSON形式のエクスポートファイルをダウンロードすることができます。

エクスポートするシナリオに、ファイルをアップロードして指定しているパラメータがある場合はエクスポートと同時に、シナリオ識別名、コンポーネント内部名、パラメータ名に応じてk2dashboard以下にフォルダが生成され、その中にアップロードされているファイルが保存されます。

- 例)：シナリオ（識別名:Scenario1）で、ENF編集（enfeditor）のENF編集用Rubyスクリプトファイル（アップロード）（scriptfile_upload）に、sample.rbがアップロードされている場合。

```
k2dashboard\Scenario1_enfeditor_scriptfile_upload\sample.rb
```

シナリオをエクスポートする際に、登録用にエクスポートするか更新用にエクスポートするかを指定することができます。

- ブラウザからエクスポート: シナリオ一覧画面の[エクスポート]ボタンのプルダウンメニューから選択します。

シナリオ一覧



コマンドからエクスポート: デフォルトは登録用エクスポートです。更新用エクスポートを実行する場合は UPDATE_MODE=true を指定してコマンドを実行します。



ヒント)

登録用エクスポートは、新しいASFALIS TransServer環境にシナリオを移行する際に便利です。

更新用エクスポートは、既存のシナリオを一括で更新する際に便利です。

12.7.4. コンポーネントテンプレート

下記のコマンドを実行することで、ASFALIS TransServerに登録されている全てのコンポーネントテンプレートが、指定したパスのファイルにJSON形式でエクスポートされます(改行は不要です)。

```
bundle exec rake ats:component_templates:export
ATS_COMPONENT_TEMPLATES_EXPORT_FILE=C:\path\to\component_templates.json
```

引数にUNIQUE_ALIAS_LISTを指定することで、エクスポートするコンポーネントテンプレートを識別名で絞り込むことができます。

```
bundle exec rake ats:component_templates:export
ATS_COMPONENT_TEMPLATES_EXPORT_FILE=C:\path\to\component_templates.json
UNIQUE_ALIAS_LIST="ComponentTemplate_1,ComponentTemplate_2"
```

エクスポートするテンプレートに、ファイルをアップロードして指定しているパラメータがある場合は、エクスポートと同時に、テンプレート識別名、コンポーネント内部名、パラメータ名に応じたk2dashboard以下にフォルダが生成され、その中にアップロードされているファイルが保存されます。

例): テンプレート (識別名:Template1) で、ENF編集 (enfeditor) のENF編集用Rubyスクリプトファイル (アップロード) (scriptfile_upload) に、sample.rbがアップロードされている場合。

```
k2dashboard\Template1_enfeditor_scriptfile_upload\sample.rb
```

12.7.5. カスタムコンポーネント

エクスポートを実行する前に、コンポーネント一覧からエクスポートに含めたいコンポーネントにチェックを付けておきます。

前述の操作でチェックを付けたカスタムコンポーネントは、以下のコマンドを実行することで指定したパスのファイルにJSON形式でエクスポートされます(改行は不要です)。

```
bundle exec rake ats:components:export
ATS_COMPONENTS_EXPORT_FILE=C:\path\to\components.json
```

引数にINTERNAL_NAME_LISTを指定することで、エクスポートするカスタムコンポーネントを内部名で絞り込むことができます。

```
bundle exec rake ats:components:export
ATS_COMPONENTS_EXPORT_FILE=C:\path\to\components.json
INTERNAL_NAME_LIST="CustomComponent_1, CustomComponent_2"
```

12.8. ASFALIS TransServerに関連するプロセス

ASFALIS TransServerでは、内部的に複数のruby.exeプロセスを起動しています。それぞれのプロセスは異なる役割を担っています。詳細は以下の通りです。

役割	説明	Ruby.exeプロセスの数
Rails	アプリケーションサーバ	Railsの数x3
MasterController	各プロセスの監視および管理	2
QueueController	タスク割り振り状況の管理	2

役割	説明	Ruby.exeプロセスの数
MiscWorker	バックアップ・不要ファイル削除等	2
ScriptComponent	スクリプトコンポーネントの実行	4
PerSlavenodeWorker	Slave Nodeとの通信	Slave Nodeの数 x2
KLoggr	ログの記録	2
K2server	利用統計等	1
ASFALIS TransServer 起動スクリプト	ASFALIS TransServerを起動する ために実行されるスクリプト	2

例えば既定の設定においてRailsは3インスタンス起動されるため、Ruby.exeは9プロセス起動します。ASFALIS TransServerに登録されたSlave Nodeの数も3であった場合、ruby.exeの数は以下の通りとなります。

Rails	3x3=9
MasterController	2
QueueController	2
MiscWorker	2
ScriptComponent	4
PerSlavenodeWorker	3x2=6
KLoggr	2
K2server	1
ASFALIS TransServer起動スクリプト	2
合計	30

また負荷の高い環境などでより多くのRailsやSlave Nodeを設定した場合には、それらの数に応じて起動されるruby.exeの数も増えることになります。ただしこれらのプロセスは必要であるために起動されているものであり、プロセスの数が多いことがシステムに対してなんらかの悪影響を及ぼすことはありません。

12.9. その他

パラメータの登録についてはEX6.1以前と同様にCSV形式のファイルを利用して実施することも可能です。以下に手順を記載します。ただし、現在はJSON形式のファイルによる登録方法を推奨しております。

12.9.1. ASFALIS TransServerへのパラメータ情報登録

CSV形式のファイルでデータを登録する場合、CSVファイルの文字コードはBOM付きのUTF-8である必要があります。作成するファイルは以下の3種類です。

- components.csv
- abstract_component_parameter_relations.csv
- component_parameters.csv

ファイルのサンプルは以下のフォルダにあります。

- <ATS Sample Directory>\custom_component

作成したファイルは以下のフォルダに配置してください。

- <ATS Install Directory>\k2dashboard\db\seed\custom\<任意のフォルダ名>\

配置したら <ATS Install Directory>\k2dashboard\start-console.bat を実行してコンソールを起動し、以下のコマンドを実行してパラメータをインポートします。

```
bundle exec rake db:load_csv CSV_FOLDER=db/seed/custom/<任意のフォルダ名>/
```

注) この操作はASFALIS TransServerが停止している時に行ってください。

各ファイルのフォーマットは以下の通りです。

- components.csv

\$internal_name	カスタムコンポーネント名を指定します。シナリオ登録・編集ページに表示されます。ジョブ作成・詳細ページに表示される文字列は、この項目に必要なに応じてサフィックスを追加したものです。
has_sub_components	FALSE を指定します。
execution_type	1 を指定します。
&component_type_id#ComponentType@internal_name>id	空文字列を指定します。
order	任意の数値を指定します。シナリオ登録・編集時にコンポーネント一覧を表示する際の順序が変わります。値が小さいほうが先に表示されます。
name@"ロケール名"	コンポーネント名を指定したロケールの言語で記述します。シナリオ作成・編集、ジョブ作成・詳細ページで表示されます。
description@"ロケール名"	コンポーネントの説明を指定したロケールの言語で記述します。ジョブ作成(ポップアップ)・詳細ページで表示されます。
has_through_io	FALSEを指定します。

*) : ロケール名には en-US, ja-JP, de-DE, zh-CNのいずれかを指定します。

- abstract_component_parameter_relations.csv

\$component_parameter_id# ComponentParameter@internal_name>id	一意になる内部名を指定します。
component_type	Component を指定します。
component_id#'component_type'@internal_name>id	components.csvに指定したカスタムコンポーネント名を指定します。

• component_parameters.csv

\$internal_name	一意になる内部名を指定します。画面には表示されません。
key_name	パラメータの内部キー名を指定します。cmpファイルで指定したものと同一文字列を指定します。
io_type	<p>1-4のいずれかの数値を指定します。</p> <ol style="list-style-type: none"> コンポーネントの入力ファイルの場合に指定します。本パラメータが以下に影響することになります。 <ul style="list-style-type: none"> シナリオ作成・編集画面でコンポーネントの接続可能性の判定 ジョブ投入画面で選択するファイルに対するフィルタ コンポーネントの出力ファイルの場合に指定します。本パラメータが以下に影響することになります。 <ul style="list-style-type: none"> シナリオ作成・編集画面でコンポーネントの接続可能性の判定 ジョブ詳細画面の出力ファイル欄と一括ダウンロード対象の制御 実行するジョブの情報としてコンポーネントに与えることができるパラメータの場合に指定します。 <ul style="list-style-type: none"> シナリオ登録・編集画面で値を設定するパラメータとして表示されます。 コンポーネントのログファイルの場合に指定します。 <ul style="list-style-type: none"> ジョブ詳細画面でログファイル欄と一括ダウンロード対象の制御
value_type	<p>io_typeの値に従って指定します。 io_typeが1, 2, または4のとき、8を指定します。 io_typeが3のとき、テキストのパラメータを作成するには2を、任意のファイルをパラメータとして利用するには8を指定します。</p>
is_mandatory	<p>io_typeが3のとき、必須パラメータとする場合にはTRUE、そうでない場合にはFALSEを指定します。必須パラメータ(TRUE)とした場合、シナリオ作成・編集画面で「設定する」チェックボックスがOnで固定になります。</p>

order	任意の数値を指定します。パラメータを表示する際の順序が変わります。値が小さいほうが先に表示されます。
is_multiple	FALSEを指定します。
&mime_type_id#MimeType @name>id	value_typeが8のとき、ファイルの種類が何かを登録します。 <ATS Install Directory>\k2dashboard\db\seed\mime_types.csvの\$nameカラムに指定されている値から選択します。value_typeが8(file)のとき、ファイルの種類を指定します。io_typeが1,2のとき、コンポーネントの接続可能性の判定に使用されます。
&value_constraint_id#Value Constraint @internal_name>id	io_typeが3のとき、パラメータに与える制約を指定します。 <ATS Install Directory>\k2dashboard\db\seed\value_constraints.csvの\$internal_nameカラムに指定されている値から選択します。制約を設定しない場合、空文字列を指定します。
name@"ロケール名"	パラメータ名を指定したロケールの言語で記述します。シナリオ登録・編集、ジョブ作成・詳細ページで表示されます。
description@"ロケール名"	パラメータの説明を指定したロケールの言語で記述します。シナリオ登録・編集、ジョブ作成・詳細ページで表示されます。

*) : ロケール名には en-US、ja-JP、de-DE、zh-CNのいずれかを指定します。

12.9.2. 実行するSlave Nodeの指定

カスタムコンポーネントを実行するSlave Nodeを指定する場合、component_parameters.csv 内にkey_name がSlaveNodeName のパラメータを作成します。通常のASFALISコンポーネントと同様に、SlaveNodeName パラメータで直接Slave Nodeを指定して実行させることができます。サンプルは以下のフォルダにあります。

- <ATS Sample Directory>\custom_component_fixed_slavenode

12.9.3. レポートファイルの登録

CSV形式のファイルを利用してカスタムコンポーネントでレポートファイルを登録するための手順を示します。サンプルは以下のフォルダにあります。

- <ATS Sample Directory>\custom_component_reportfile

cmpファイルを作成して登録する手順については「[9.2.4, “レポートファイルの ASFALIS TransServer への登録”](#)」の a. と同様となります。カスタムコンポーネントのパラメータ情報をDBに登録する方法は「[12.9.1, “ASFALIS TransServerへのパラメータ情報登録”](#)」を参照してください。ただしreportfileに固有のパラメータは下記に従って登録してください。

- component_parameters.csv

\$internal_name	一意になる内部名を指定します。画面には表示されません。
key_name	reportfile を指定します。 この値は変更可能ですが、report_file.txtのパスを参照するキー(デフォルトではreportfile_filename)が、この値に_filenameを付けたものである必要があります。

io_type	4 を指定します。
value_type	21 を指定します。
is_mandatory	FALSEを指定します。
is_multiple	FALSEを指定します。
&mime_type_id#MimeType @name>id	application/octet-stream を指定します。
&value_constraint_id#Value Constraint @internal_name>id	空文字列を指定します。

- abstract_component_parameter_relations.csv

「[12.9.1, “ASFALIS TransServerへのパラメータ情報登録”](#)」に従い、reportfileに対応する行を記述します。

本コンテンツに関わる著作権は株式会社エリジオンもしくは原権利者に帰属しています。
著作権者の承諾なしに無断で改変、複製、転載、再配布、転送、公衆送信、販売、貸与などの
行為をすることは禁じられています。