

# ConMas Generator 操作説明書

---

# 改訂履歴

| 日付         | 記載ページ          | 改訂内容  |
|------------|----------------|---|
| --         | --             | 初版を発行しました。  |
| 2015.02.09 | P.39 置換文字列について | 置換文字列のシステム日付の指定方法にパラメータを追加                          |
| 2015.10.19 | P43            | サーバーOS実行時の注意事項を追記                                   |
|            | P44            | ジョブ実行ログ 項目説明 を追記                                    |
| 2017.04.12 | P41            | 置換文字列“ {ConMasWatcherFile}”が“保存ファイル名”でも利用可能であることを追記 |
| 2018.12.03 | P8             | プロキシ設定を追記   |
|            | P16            | フォルダー監視ジョブ リトライ設定を追記                                |
| 2018.12.26 | P45            | サーバーOS実行時の注意事項に注意対象OSを追記                            |
| 2019.09.04 | P49～           | ご利用上の注意を追記  |
| 2022.01.26 | P7             | 「WebAPI」接続URLを追記                                    |

# 目次

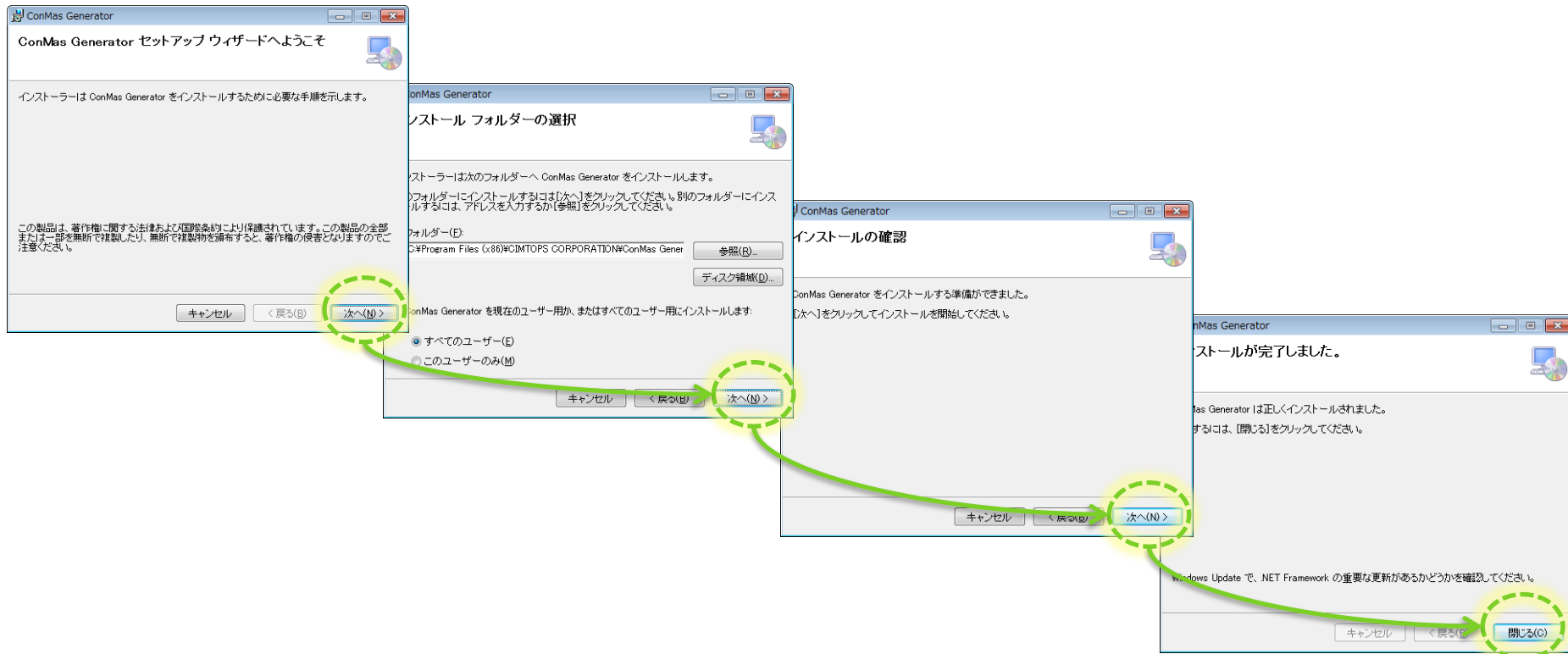
- ConMas Generatorインストール
- 初期設定
- プロキシ設定
- ConMasGeneratorジョブについて
- ジョブ設定画面説明
- フォルダー監視ジョブ設定
- フォルダー監視ジョブ リトライ設定
- フォルダー監視サービスについて
- スケジュールジョブ設定
- スケジュールジョブについて
- 置換文字列について
- ジョブ実行ログ参照
- ジョブ実行ログ 項目説明
- メール設定
- ファイルログについて

# ConMas Generatorインストール

インストールメディア内の以下のファイルを実行します。

ConMasGeneratorSetup.msi

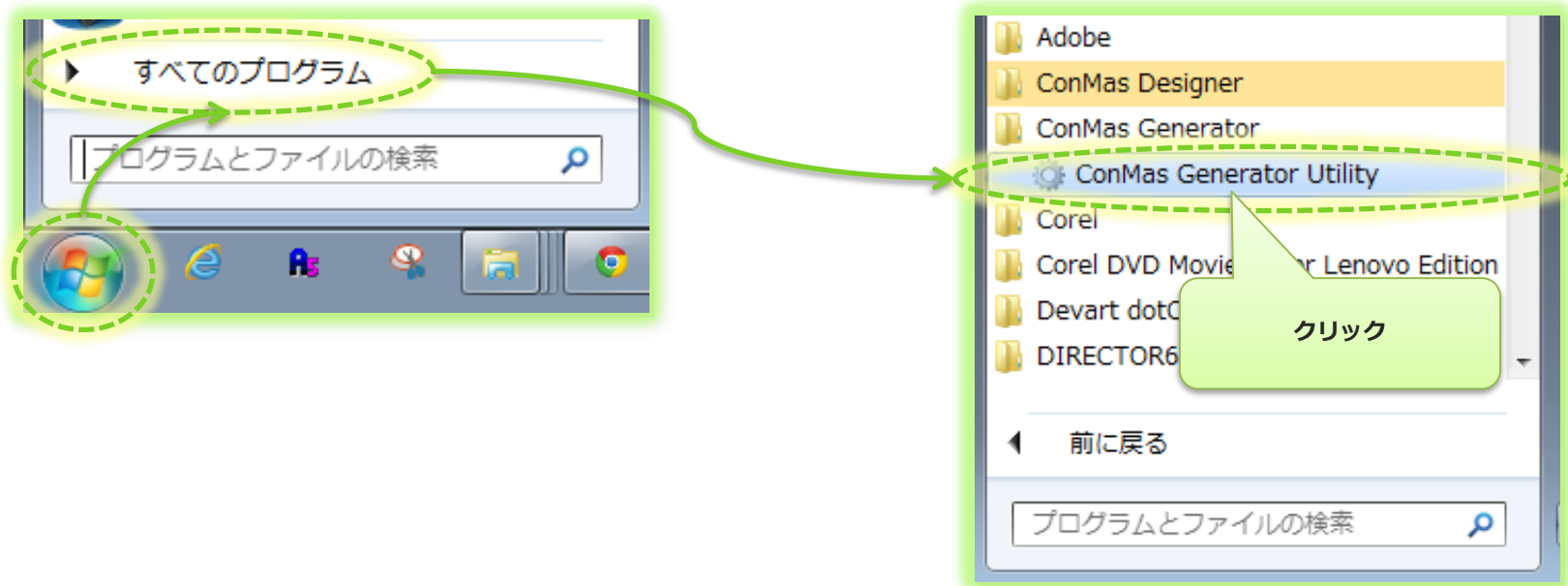
※旧バージョンは予めアンインストールしてください。



# 初期設定

Generatorの初期設定を行います。

1. Windowsスタートメニュー→すべてのプログラム→ConMas Generator  
→ConMas Generator Utilityをクリックします。



【補足】

ConMas Generator Utilityの起動には、管理者権限が必要です。

# 初期設定

Generatorの初期設定を行います。

2. 「基本設定」タブを選択し、各項目の設定を行います。  
「保存」ボタンを押下すると、設定は完了します。  
※各項目の設定値については、次ページで説明します。

The screenshot shows the 'ConMas Generator [Version:1.0.0.0]' application window. The '基本設定' (Basic Settings) tab is selected and highlighted with a dashed green circle. A green arrow points from this tab to the '保存' (Save) button, which is also highlighted with a dashed green circle and a callout bubble saying 'クリックして保存' (Click to save). The main configuration area is enclosed in a dashed green box with a callout bubble saying '編集' (Edit). The settings are as follows:

|                                 |  |       |      |
|---------------------------------|--|-------|------|
| ConMasサーバーURL                   | http://localhost/ConMasAPI/Rests/APIExecute.aspx |       |      |
| ConMasユーザー                      | user   |       |      |
| ConMasパスワード                     |  |       |      |
| ローカル管理者ユーザー                     | Administrator                                    |       |      |
| ローカル管理者パスワード                    |  |       |      |
| ジョブ実行ログ保存日数                     | 7  |       |      |
| Expect100Continue               | <input type="checkbox"/>                         |       |      |
| Timeout                         | 1800000  |       |      |
| <input type="checkbox"/> プロキシ設定 |  |       |      |
| IPアドレス                          | 192.168.10.81                                    | ポート   | 8080 |
| ユーザー                            |  | パスワード |      |

# 初期設定

Generatorの初期設定を行います。

## 3. 設定値説明

- ConMasサーバーURL

外部連携オプションより提供された「WebAPI」接続URLを設定します。

オンプレミス版 `http(s)://{ホスト名}/ConMasAPI/Rests/APIExecute.aspx`

クラウド版 `https://{ホスト名}/ConMasAPIXXXXXX/_Rests/APIExecute.aspx`

- ConMasユーザー

ConMasサーバーに接続出来るユーザー名を設定します。

- ConMasパスワード

ConMasユーザーのパスワードを設定します。

- ローカル管理者ユーザー

Generatorインストール端末の管理者ユーザを設定します。

- ローカル管理者パスワード

ローカル管理者ユーザーのパスワードを入力します。

- ジョブ実行ログ保存日数

ジョブ実行ログの保存日数を設定します。

- Expect100Continue

ご利用のネットワーク環境に合わせて、100-Continue動作を設定します。

※Expect 100-Continue 機能の詳細については、IETF RFC 2616 Section 10.1.1を参照ください。

- Timeout設定

HTTP要求タイムアウト値をミリ秒で設定します。

# 初期設定 プロキシ設定

プロキシ設定を行う場合は、プロキシ設定にチェックします。

プロキシ設定

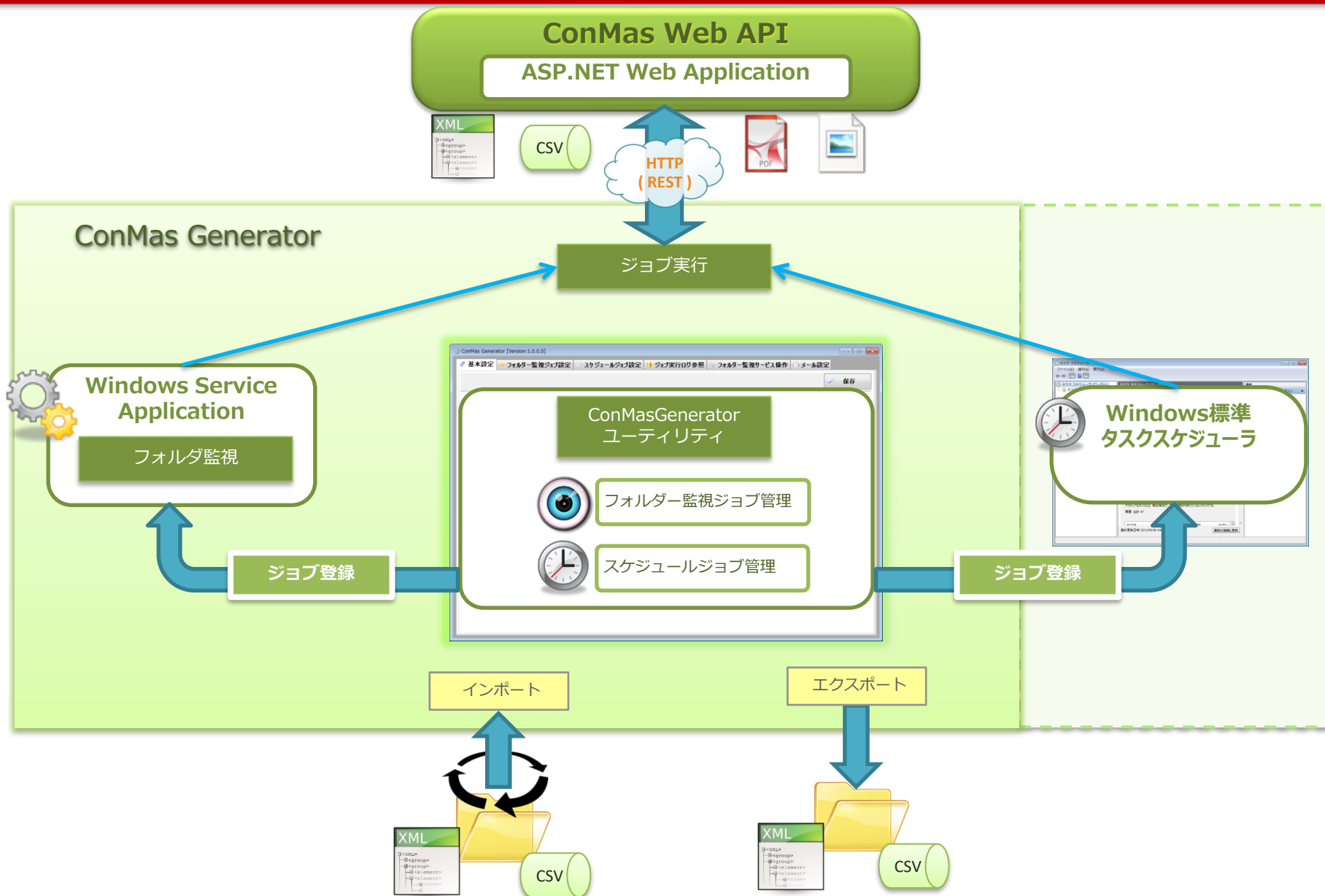
IPアドレス  ポート

ユーザー  パスワード

- IPアドレス  
プロキシサーバーのIPアドレス
- ポート  
プロキシサーバーのポート番号
- ユーザー、パスワード  
ユーザー認証時のユーザーID、パスワード



# ConMas Generator ジョブについて



# ConMas Generatorジョブについて

## 1. ジョブとは

任意の処理をひとまとめにした単位。

配下に複数のプロセスを持つ事が可能です。

※プロセスについては、次ページの“4. プロセス”を参照ください。

## 2. ジョブの種類

ジョブには、“フォルダー監視ジョブ”と“スケジュールジョブ”2種類があります。

### ◆フォルダー監視ジョブ

指定フォルダを監視し、フォルダ内にファイルが追加されたタイミングでジョブを実行します。

### ◆スケジュールジョブ

ジョブが定期的に実行します。

## 3. ジョブ構造

ジョブは以下の階層で管理されます。



# ConMas Generatorジョブについて

## 4. プロセス

ジョブに属します。

配下に複数のコマンドを持つ事が可能です。

プロセスタイプを決定させる階層。

### ◆プロセスタイプ

- WebAPI (REST処理)
- ローカル実行ファイル起動 (ローカルEXE実行処理)

## 5. コマンド

プロセスに属します。

決定されたプロセスタイプに対応したコマンドを設定できます。

### ◆コマンド

プロセスタイプ : WebAPI (REST処理)

- REST処理に必要な URL、POSTパラメータ設定を持ちます。

プロセスタイプ : ローカル実行ファイル起動 (ローカルEXE実行処理)

- 実行ファイル (EXE) パス、パラメータ設定を持ちます。

### 【補足】

WebAPI処理のサーバーセッション保持について

1プロセス配下の各コマンド間は、セッションが保持されます。

# ジョブ設定画面説明

## ジョブ設定画面の説明

現在の編集を破棄して、  
編集前の状態に戻します。

ジョブ・プロセス・コマンドの  
追加・削除  
※同様の操作が右クリックでも可能です。

ジョブ設定の保存

The screenshot shows a software interface for job settings. At the top, there are several tabs: 'ジョブ設定', 'フォルダ監視設定', 'スケジュールジョブ設定', 'ジョブ実行ログ参照', 'フォルダ監視サービス操作', and 'メール設定'. Below the tabs is a toolbar with icons for undo, add, delete, and up/down arrows. A tree view on the left shows a hierarchy: 'フォルダ監視ジョブ' > 'フォルダ監視ジョブ01' > '自動帳票作成 (CSV)' > 'Login' > '[CSV]AutoGenerate' > 'Logout'. A large central area is designated for editing job processes and commands. A '保存' (Save) button is located in the top right corner. Callouts provide detailed instructions for each of these elements.

ジョブ・プロセス・コマンドの  
実行順番変更

保存

ジョブ設定をツリー  
表示します。

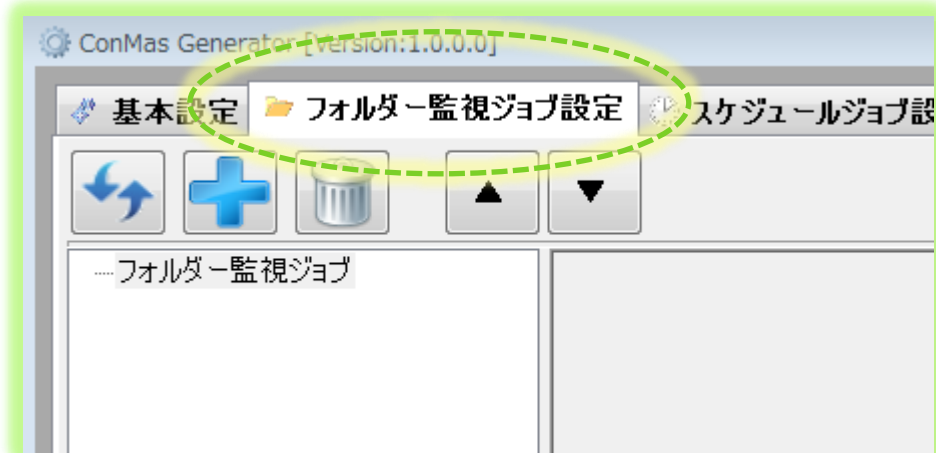
トップ  
→ジョブ  
→プロセス  
→コマンド  
の階層で表示されます。

ジョブ・プロセス・コマンド  
編集領域

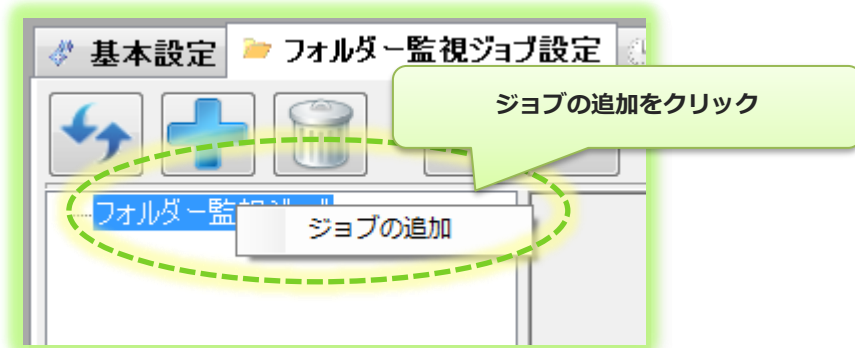
ジョブ設定ツリーで選択されたノードに対応した  
編集ビューが表示されます。

# フォルダー監視ジョブ設定

1. 「フォルダ監視ジョブ」タブを選択します。

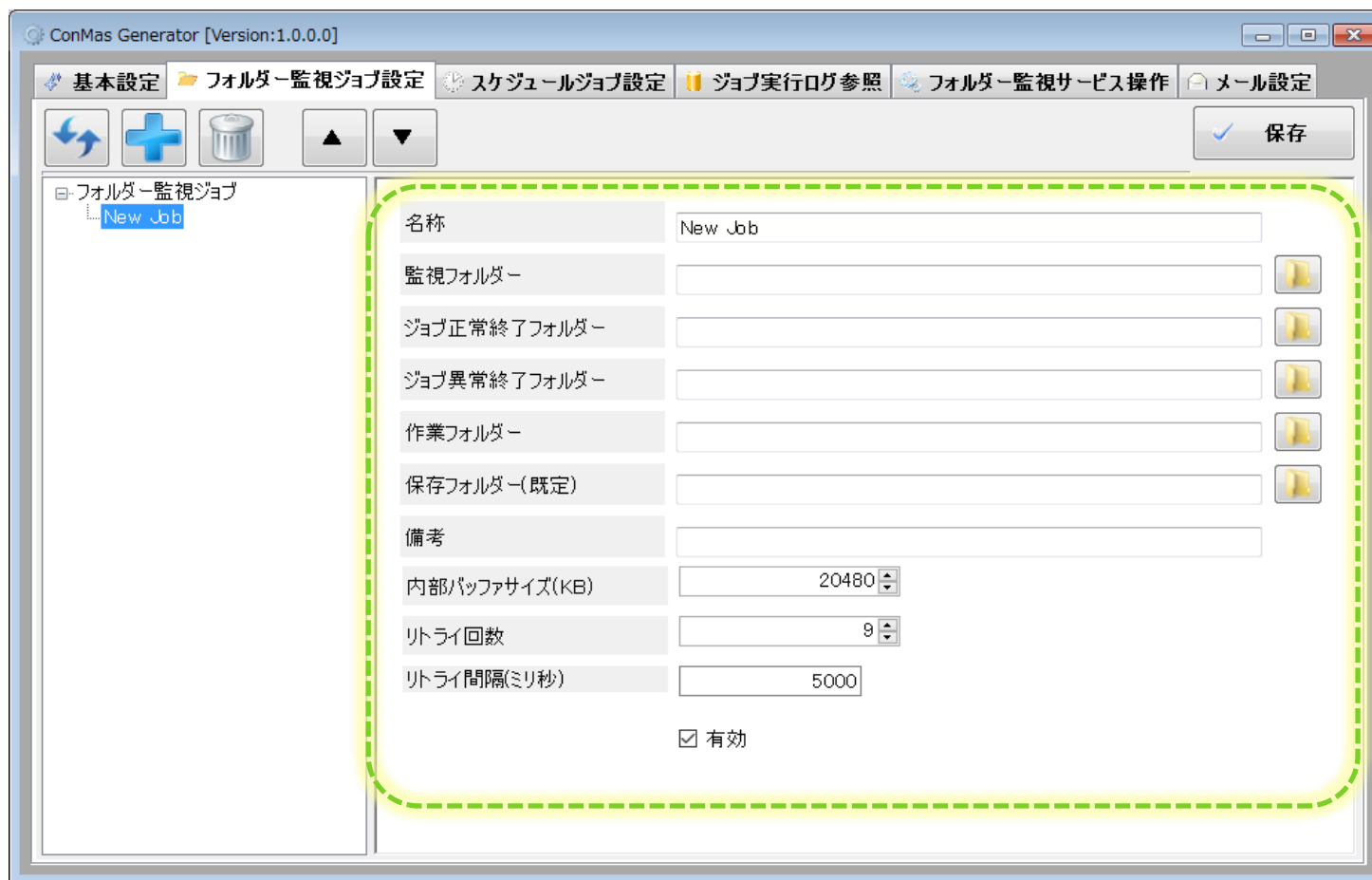


2. ジョブを追加します。  
"フォルダ監視ジョブ"ノード上で右クリック→ジョブの追加 をクリック



# フォルダー監視ジョブ設定

3. ジョブを編集します。  
画面右部にジョブ編集ビューが表示されます。  
※各編集項目については、次ページにて説明します。



# フォルダー監視ジョブ設定

## ジョブ編集項目説明

※青文字は入力必須

- 名称  
ジョブ名称を設定します。
- 監視フォルダー  
監視する対象のフォルダーを設定します。  
※サブフォルダは監視対象外となります。
- ジョブ正常終了フォルダー  
正常にジョブ実行された場合に、トリガーとなったファイルが移動するフォルダーを設定します。
- ジョブ異常終了フォルダー  
ジョブ実行中に何らかの異常が発生した場合に、トリガーとなったファイルが移動するフォルダーを設定します。
- 作業フォルダー  
トリガーとなったファイルをジョブ実行中の間のみ、保管させるフォルダーを設定します。
- 保存フォルダー（既定）  
WebApi処理を行った際に、結果を保存するフォルダーを設定します。  
※コマンド単位で、保存先は変更可能です。
- 備考  
ジョブに対する備考を設定します。
- 有効  
ジョブを無効にしたい場合は、チェックを外します。

# フォルダー監視ジョブ設定

## フォルダー監視ジョブ設定時の注意事項

フォルダ監視ジョブは、監視フォルダにファイルが配置されるとファイルを移動してジョブ実行を行います。複数ジョブで同じフォルダを監視すると、配置したファイルを奪い合うような動作になりエラーが発生します。安定利用するためには、1つのジョブで1つのフォルダを監視するように設定してください。



# フォルダー監視ジョブ リトライ設定

## 4. リトライ設定

フォルダー監視はファイル書き込み中でも動作しますが、書き込み中のためエラーが発生しジョブが強制終了します。

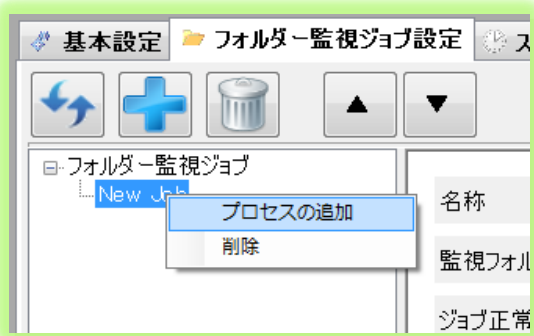
これを回避するのがリトライ設定です

※書き込み中の間、リトライすることを書き込みの終了を待ちます

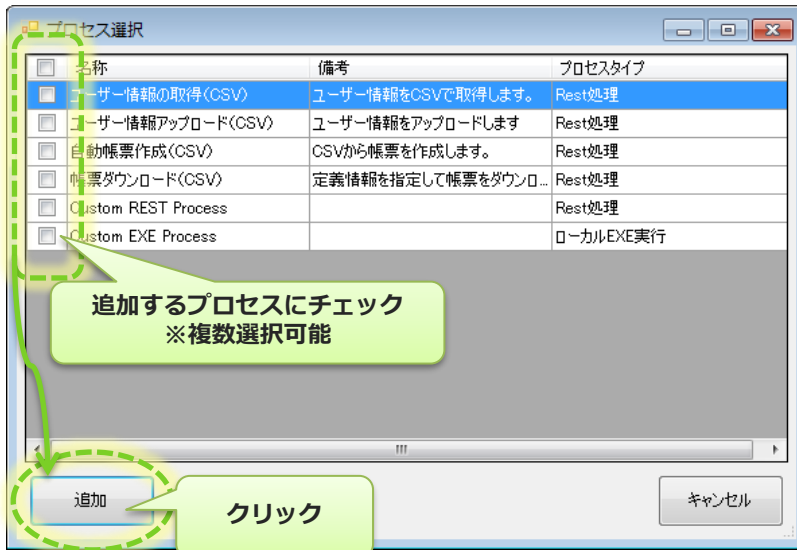
- 内部バッファサイズ (KB)  
内部バッファサイズは4096～65536 (キロバイト) で設定します  
大きいほど数多くのファイルを監視できます
- リトライ回数  
書き込み中エラーの場合、本設定分ジョブをリトライします
- リトライ間隔 (ミリ秒)  
書き込み中エラーの場合にリトライする間隔をミリ秒で設定します

# フォルダー監視ジョブ設定

5. プロセスを追加します。  
“ジョブノード”上で右クリック→“プロセスの追加”をクリック



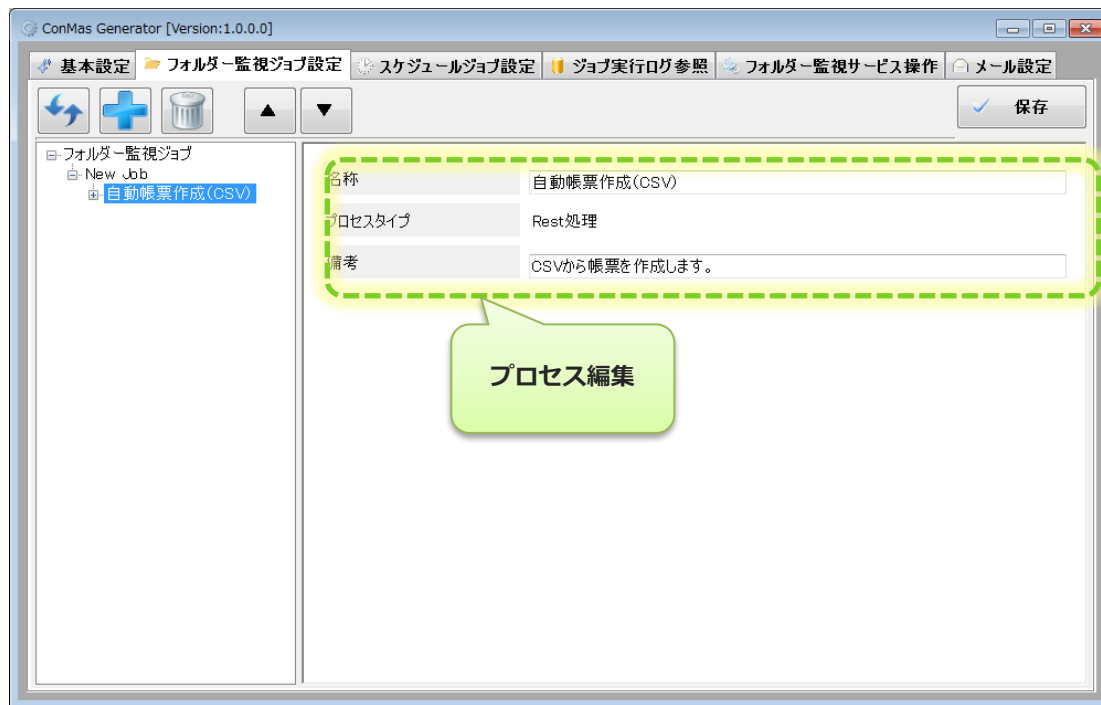
6. プロセスを選択します。  
プロセス選択画面から追加したいプロセスを選択し、「追加」を押下



# フォルダー監視ジョブ設定

7. 必要に応じてプロセスを編集します。

先ほど追加したプロセスをツリービューより選択すると画面右部にプロセス編集ビューに切り替わります。



8. プロセス編集項目説明 ※青文字は入力必須

- 名称

プロセス名称を設定します。

- 備考

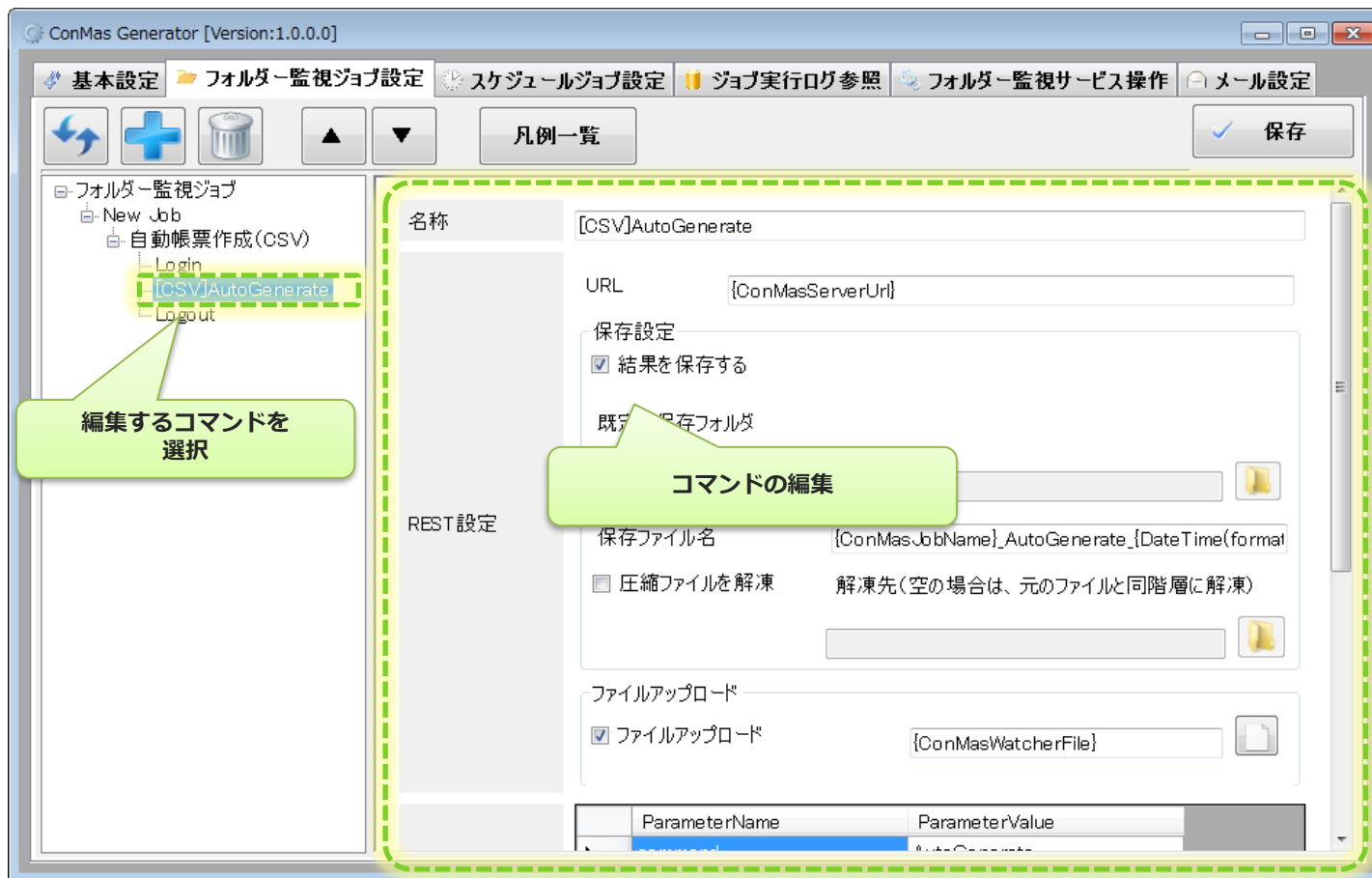
プロセスの備考を設定します。

# フォルダー監視ジョブ設定

9. コマンドを編集します。

ツリービュー内よりプロセスノードを開いて編集するコマンドを選択します。

※コマンドの編集項目については、次ページにて説明します。



# フォルダー監視ジョブ設定

## 10. コマンド編集項目説明 ※青文字は入力必須

### ◆プロセスタイプ : WebAPI (REST処理)

- 名称

コマンド名称を設定します。

- URL

リクエスト先URLを設定します。

※ConMas外部連携WebAPIの場合は、“{ConMasServerUrl}”と記述します。

- 保存設定

- 結果を保存する

結果を保存する場合はチェックします。

- 保存フォルダ変更

保存フォルダを変更する場合はチェックして、保存フォルダを設定します。

チェックなしの場合はジョブの「保存フォルダ (既定)」となります。

- 保存ファイル名

保存ファイル名を設定します。

※置換文字の割り当てが可能です。詳しくは「置換文字列について」を参照してください。

- 圧縮ファイルを解凍

結果が圧縮ファイルの場合で、即解凍したい場合にチェックします。

解凍先フォルダを指定する事も出来ます。

# フォルダー監視ジョブ設定

## 10. コマンド編集項目説明

### ◆プロセスタイプ : WebAPI (REST処理)

- ファイルアップロード

ファイルをアップロードしたい場合はチェックし、アップロードファイルを指定します。

※“{ConMasWatcherFile}”を指定すると、トリガーとなったファイルが対象となります。

- パラメータ設定

REST処理に必要なパラメータを設定します。

※置換文字の割り当てが可能です。詳しくは「置換文字列について」を参照してください。

### ◆プロセスタイプ : ローカル実行ファイル起動 (EXE処理)

- 名称

コマンド名称を設定します。

- 実行ファイルパス

実行ファイルパスを設定します。

※フルパス指定

- 待機設定

実行ファイルの終了を待つ場合は、チェックします。

- パラメータ設定

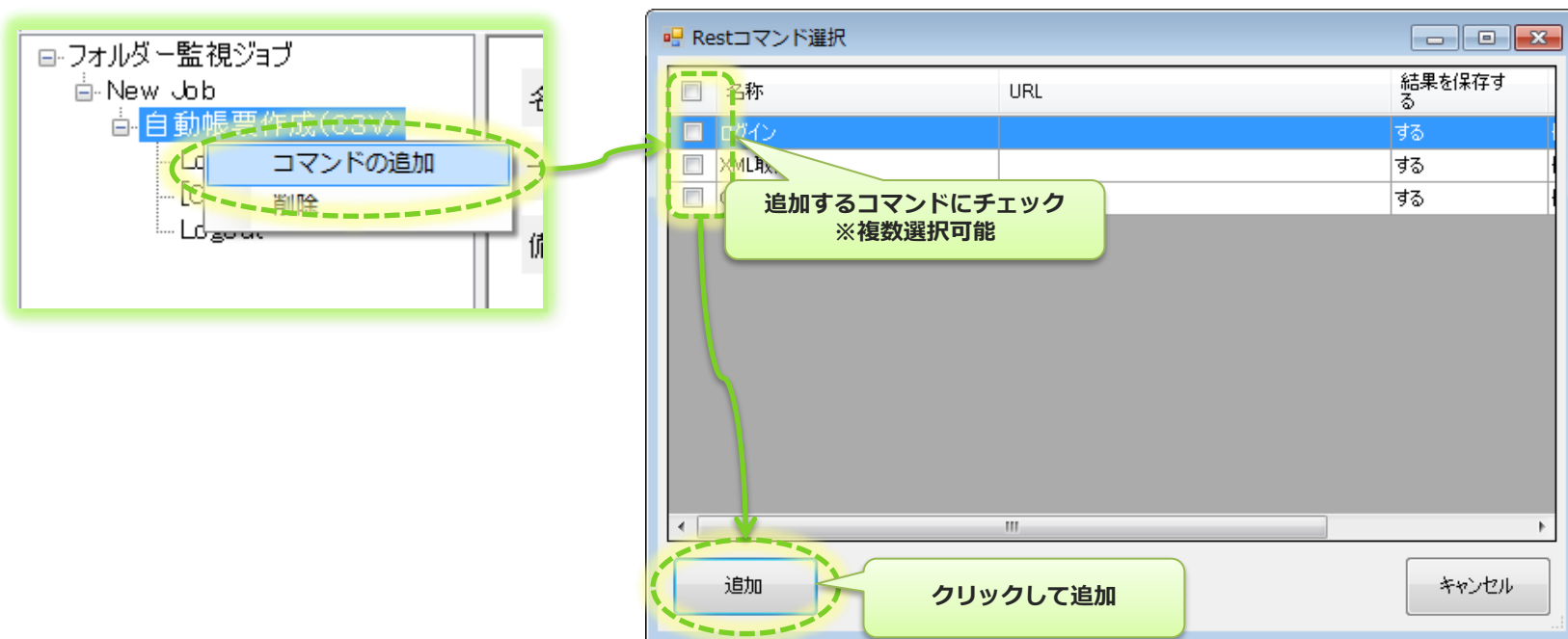
実行ファイルにパラメータを渡す場合に記述します。

※“{ConMasWatcherFile}”を指定すると、トリガーとなったファイルパスを取得する事が出来ます。

# フォルダー監視ジョブ設定

## 1 1. コマンドを追加します。

コマンドを追加したい場合は、ツリービューにて追加するプロセス上で右クリック→“コマンドの追加”をクリックします。

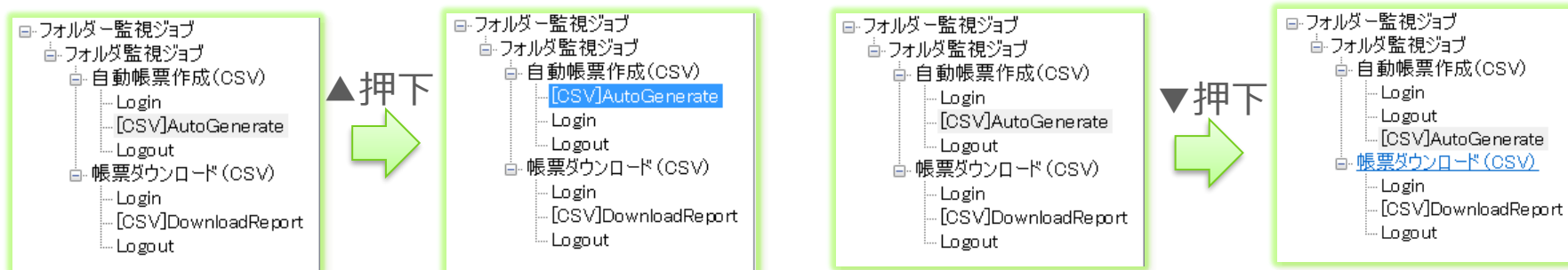


# フォルダー監視ジョブ設定

1 2. プロセス、コマンドの順番を変更します。(ジョブも可能)  
順番をしたいプロセス又はコマンドをツリービュー上で選択



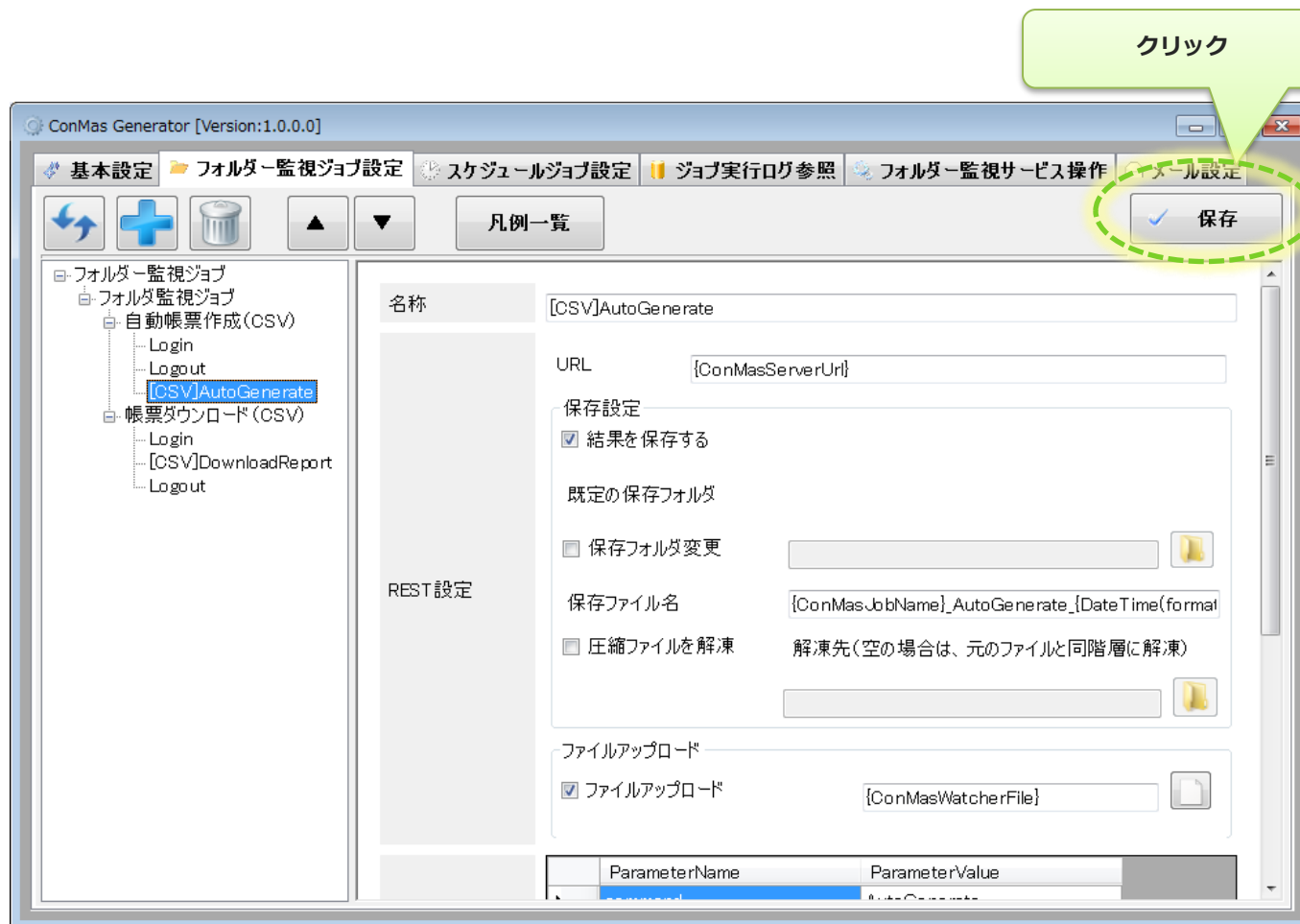
▲、▼ボタンを押下してノード移動





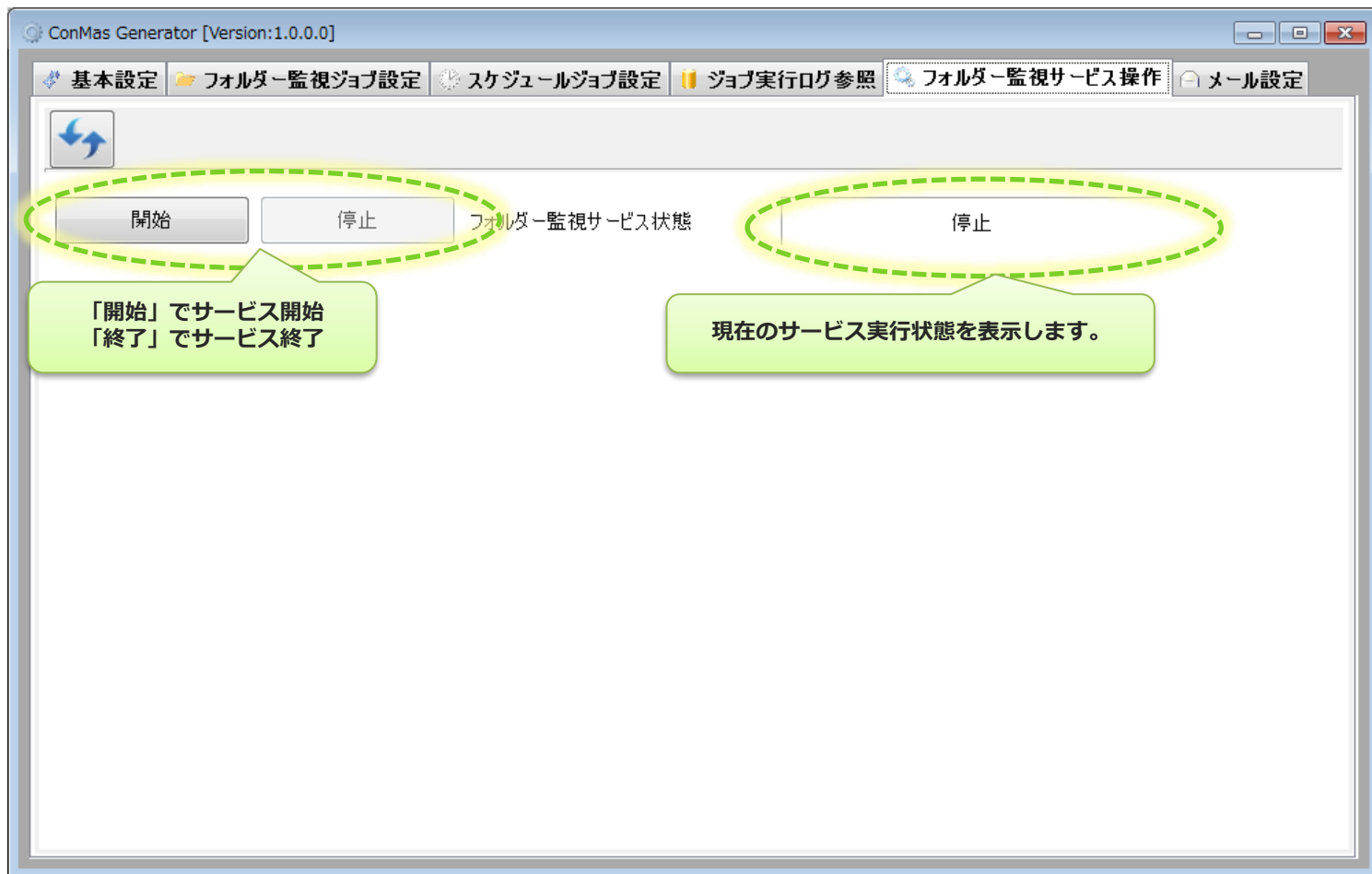
# フォルダー監視ジョブ設定

- 1 3. 保存して、フォルダー監視ジョブを開始します。  
画面右上の「保存」ボタンを押下すると、画面で設定されたジョブの内容でフォルダー監視が開始されます。



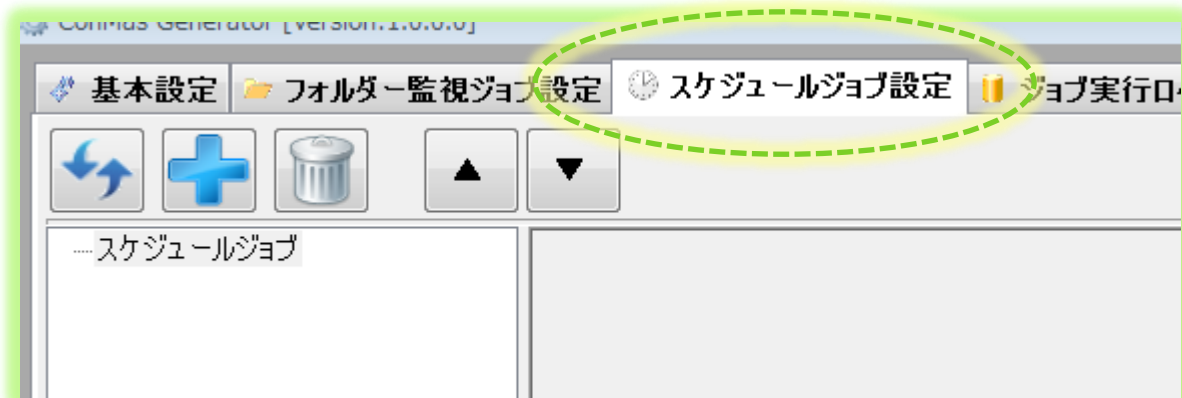
# フォルダー監視サービスについて

フォルダー監視はWindowsサービスにて行っています。  
サービスを停止したい場合は、「フォルダー監視サービス操作」タブを選択し  
「停止」ボタンを押下する事で停止します。



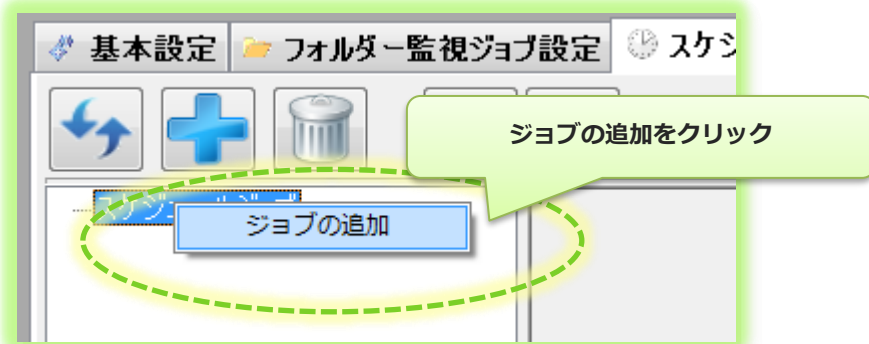
# スケジュールジョブ設定

1. 「スケジュールジョブ設定」タブを選択します。



2. ジョブを追加します。

“スケジュールジョブ”ノード上で右クリック→ジョブの追加 をクリック



# スケジュールジョブ設定

3. ジョブを編集します。  
画面右部にジョブ編集ビューが表示されます。  
※各編集項目については、次ページにて説明します。

ConMas Generator [Version:1.0.0.0]

基本設定 | フォルダ-監視ジョブ設定 | **スケジュールジョブ設定** | ジョブ実行ログ参照 | フォルダ-監視サービス操作 | メール設定

保存

スケジュールジョブ  
New Job

名称: New Job

保存フォルダ-(既定): [ ]

備考: [ ]

作業フォルダ: [ ]

設定

一回のみ  毎日  毎週  毎月

開始日時: 2012/09/05 16 : 37

間隔(日): 1

繰り返し間隔(分): [ ]

有効

有効期限: 2012/09/05 16 : 37

# スケジュールジョブ設定

## 4. ジョブ編集項目説明

### ※青文字は入力必須

- 名称  
ジョブ名称を設定します。
- 作業フォルダー  
アップロードしたファイルの移動させる場合の規定フォルダーを設定します。  
※アップロードしない場合でも必ず指定する必要があります。
- 保存フォルダー（既定）  
WebApi処理を行った際に、結果を保存するフォルダーを設定します。  
※コマンド単位で、保存先は変更可能です。
- 備考  
ジョブに対する備考を設定します。

# スケジュールジョブ設定

## 4. ジョブ編集項目説明

※青文字は入力必須

- 設定

- スケジュールタイプ

- 一回のみ

指定した開始日時に一回だけジョブを実行します。

- 毎日

指定した間隔（日）毎、指定した開始日時の時刻にジョブを実行します。

（繰り返し設定にて、分単位のジョブ実行も可能です。）

- 毎週

指定した週毎、指定した開始日時の時刻にジョブを実行します。

- 毎月

指定した月日毎、指定した開始日時の時刻にジョブを実行します。

- 有効

ジョブを無効にしたい場合は、チェックを外します。

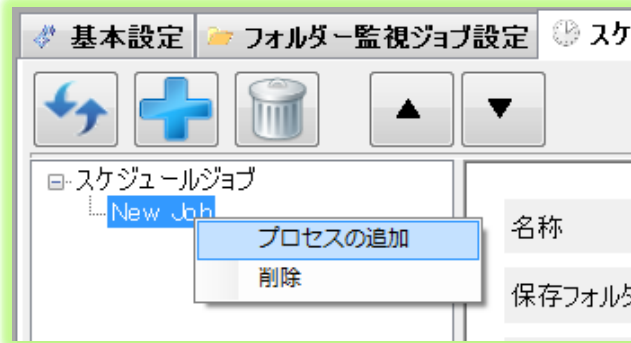
- 有効期限

ジョブの有効期限を指定したい場合は、設定します。

# スケジュールジョブ設定

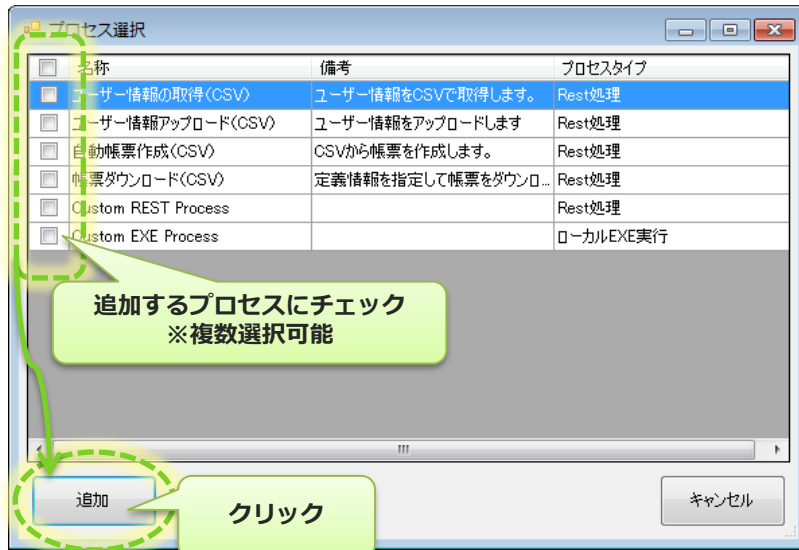
5. プロセスを追加します。

“ジョブノード”上で右クリック→“プロセスの追加”をクリック



6. プロセスを選択します。

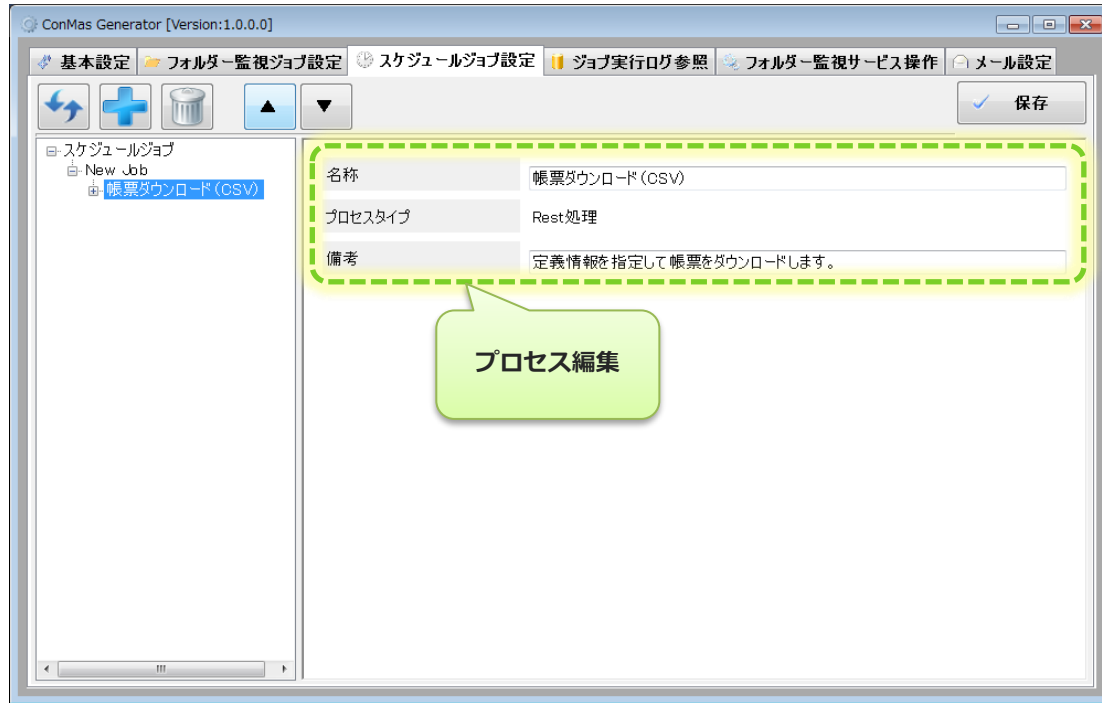
プロセス選択画面から追加したいプロセスを選択し、「追加」を押下



# スケジュールジョブ設定

7. 必要に応じてプロセスを編集します。

先ほど追加したプロセスをツリービューより選択すると画面右部にプロセス編集ビューに切り替わります。



8. プロセス編集項目説明 ※青文字は入力必須

- 名称

プロセス名称を設定します。

- 備考

プロセスの備考を設定します。



# スケジュールジョブ設定

9. コマンドを編集します。

ツリービュー内よりプロセスノードを開いて編集するコマンドを選択します。

※コマンドの編集項目については、次ページにて説明します。



# スケジュールジョブ設定

## 10. コマンド編集項目説明 ※青文字は入力必須

### ◆プロセスタイプ : WebAPI (REST処理)

- 名称

コマンド名称を設定します。

- URL

リクエスト先URLを設定します。

※ConMas外部連携WebAPIの場合は、“{ConMasServerUrl}”と記述します。

- 保存設定

- 結果を保存する

結果を保存する場合はチェックします。

- 保存フォルダ変更

保存フォルダを変更する場合はチェックして、保存フォルダを設定します。

チェックなしの場合はジョブの「保存フォルダ (既定)」となります。

- 保存ファイル名

保存ファイル名を設定します。

※置換文字の割り当てが可能です。詳しくは「置換文字列について」を参照してください。

- 圧縮ファイルを解凍

結果が圧縮ファイルの場合で、即解凍したい場合にチェックします。

解凍先フォルダを指定する事も出来ます。

# スケジュールジョブ設定

## 10. コマンド編集項目説明

### ◆プロセスタイプ : WebAPI (REST処理)

- ファイルアップロード

ファイルをアップロードしたい場合はチェックし、アップロードファイルを指定します。

※アップロードした後に対象ファイルを移動させたい場合は、以下のアップロード後の動作設定を行います。

- アップロード後の動作

- 何もしない

アップロードしたファイルは、そのままの場所に残ります。

- 作業フォルダへファイル移動

ジョブで設定した作業フォルダへ移動します。

- 指定フォルダへファイル移動

指定フォルダへ移動します。

- パラメータ設定

REST処理に必要なパラメータを設定します。

※置換文字の割り当てが可能です。詳しくは「置換文字列について」を参照してください。

# スケジュールジョブ設定

## 10. コマンド編集項目説明

### ◆プロセスタイプ：ローカル実行ファイル起動（EXE処理）

- 名称

コマンド名称を設定します。

- 実行ファイルパス

実行ファイルパスを設定します。

※フルパス指定

- 待機設定

実行ファイルの終了を待つ場合は、チェックします。

- パラメータ設定

実行ファイルにパラメータを渡す場合に記述します。

# スケジュールジョブ設定

1. コマンドを追加します。  
コマンドを追加したい場合は、ツリービューにて追加するプロセス上で右クリック→“コマンドの追加”をクリックします。

スケジュールジョブ

- New Job
  - 帳票ダウンロード (CSV) (右クリックメニュー)
    - コマンドの追加
    - 削除
  - Log in
  - [CSV]
  - Logout

| 名称                                       | URL | 結果を保存する |
|--|-----|---------|
| <input checked="" type="checkbox"/> ログイン |     | する      |
| <input type="checkbox"/> XML取            |     | する      |
| <input type="checkbox"/>                 |     | する      |

追加するコマンドにチェック  
※複数選択可能

追加

クリックして追加

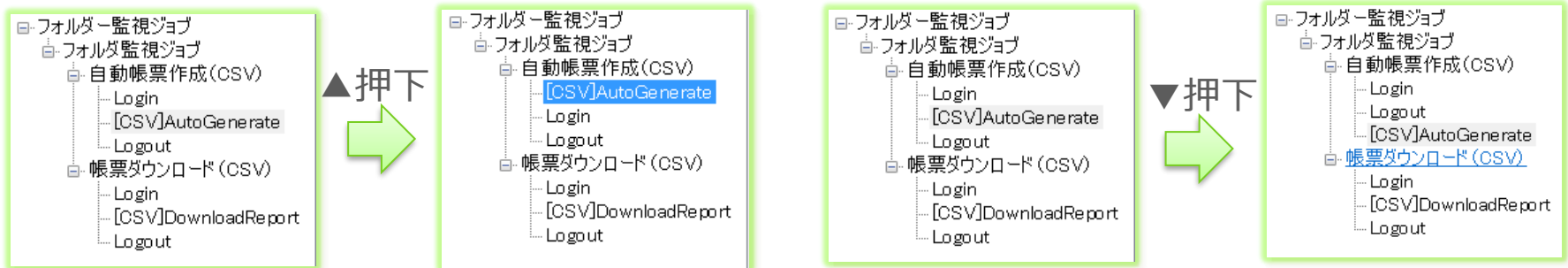
キャンセル

# スケジュールジョブ設定

1 2. プロセス、コマンドの順番を変更します。(ジョブも可能)  
順番をしたいプロセス又はコマンドをツリービュー上で選択

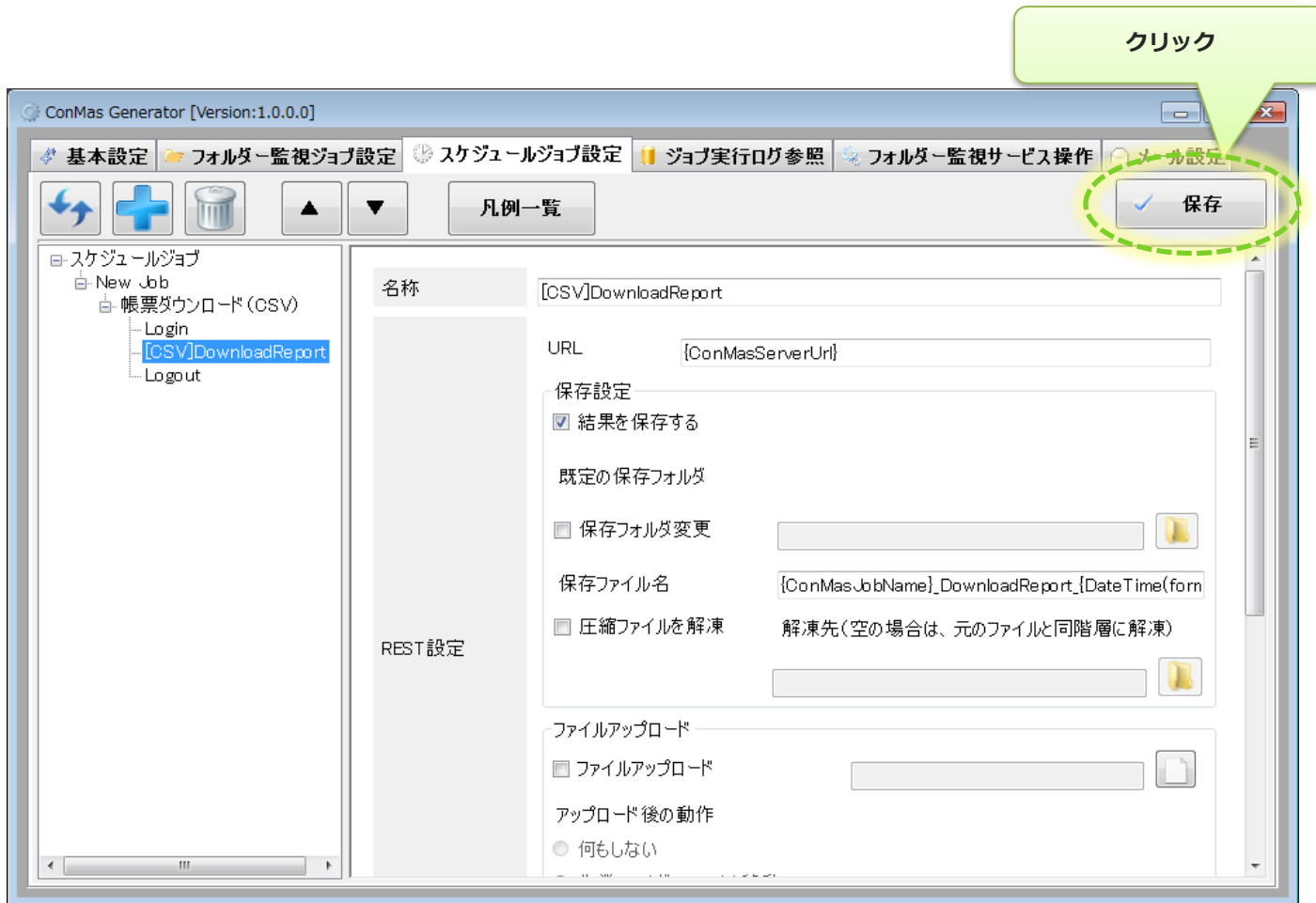


▲、▼ボタンを押下してノード移動



# スケジュールジョブ設定

- 1 3. 保存して、スケジュールジョブを開始します。  
画面右上の「保存」ボタンを押下すると、画面で設定されたジョブの内容でスケジュールジョブが開始されます。



# スケジュールジョブについて

スケジュールジョブの実行は、Windows標準のタスクスケジューラを使用しています。  
保存すると、タスクスケジューラにてジョブのスケジュール情報を参照する事が可能です。

The image shows two overlapping windows. The top window is 'ConMas Generator [Version:1.0.0.0]' with the 'スケジュールジョブ設定' (Schedule Job Settings) tab active. A '保存' (Save) button is highlighted with a green dashed circle. A green callout bubble points from this button to the 'タスク スケジューラ' (Task Scheduler) window below. The Task Scheduler window shows a list of tasks in the 'タスク スケジューラ ライブラリ' (Task Scheduler Library). The task 'ComMasG...' is selected and highlighted in blue. Below the list, the task's properties are visible, including the name 'ComMasGenerator Job\_New Job\_787b89c9-c8cc-481c-958a-d9b842823d46', creator 'sameru', and security options.

タスクスケジューラに登録します。

| 名前            | 状態   | トリガー  | 次回の実行時刻         |
|---------------|------|---|-----------------|
| Adobe Flas... | 準備完了 | 毎日 9:24 に起動 - トリガーされた後、1 日間の間 1 時間 ごとに繰り返します。 | 2012/09/05 16:2 |
| ComMasG...    | 準備完了 | 2012/09/04 以降毎週日曜日、金曜日、13:15 に起動              |                 |
| DiskUpdate    | 準備完了 | 任意のユーザーがワークステーションのロックを解除したとき                  |                 |
| GoogleUpd...  | 準備完了 | 複数のトリガーの定義                                    | 2012/09/06 1:12 |
| GoogleUpd...  | 準備完了 | 毎日 1:12 に起動 - トリガーされた後、1 日間の間 1 時間 ごとに繰り返します。 | 2012/09/05 16:1 |
| MCP           | 準備完了 | 任意のユーザーのログイン時                                 |                 |
| Microsoft_... | 準備完了 | タスクの作成または変更時                                  |                 |
| PMTask        | 準備完了 | コンピューターのアイドル時                                 |                 |



# 置換文字列について

コマンド編集にて、いくつかの設定項目で置換文字列を使用してシステム固有の値を自動設定する事が出来ます。

## ◆“URL” で使用できる置換文字列

| 置換文字列             | 置換される値             |
|-------------------|--------------------|
| {ConMasServerUrl} | 基本情報のConMasサーバーURL |

The screenshot shows a web-based configuration interface for a job. On the left is a tree view of jobs under 'スケジュールジョブ', with 'Login' selected. The main area shows the configuration for the 'Login' job. The 'URL' field is highlighted with a dashed green box and contains the placeholder '{ConMasServerUrl}'. Other fields include '名称' (Login) and a '保存設定' section with a checked '結果を保存する' option. A '保存' button is visible in the top right.

## ◆“ファイルアップロード” で使用できる置換文字列

※フォルダ監視ジョブ限定

| 置換文字列               | 置換される値            |
|---------------------|-------------------|
| {ConMasWatcherFile} | トリガーとなったファイルのフルパス |

The screenshot shows the configuration for a file upload job. The 'ファイルアップロード' checkbox is checked. The file path field contains the placeholder '{ConMasWatcherFile}' and has a file icon to its right.

# 置換文字列について

コマンド編集にて、いくつかの設定項目で置換文字列を使用してシステム固有の値を自動設定する事が出来ます。

◆“保存ファイル名” で使用できる置換文字列

| 置換文字列   | 置換される値   |
|---|--|
| {ConMasJobName}   | ジョブ名称  |
| {ConMasProcessIndex}                                    | プロセスの実行連番 (0~)   |
| {ConMasProcessName}                                     | プロセス名称   |
| {ConMasCommandIndex}                                    | コマンドの実行連番 (0~)   |
| {ConMasCommandName}                                     | コマンド名称   |
| {DateTime(format=yyyyMMdd,offsetType=M,offsetValue=-2)} | システム日付 (フォーマット指定(yyyyMMdd))<br>offsetType → Y:年、M:月、D:日 のいずれかを指定<br>offsetValue → offsetTypeで指定した識別子からの増減を指定する |

保存ファイル名

{ConMasJobName}\_Logout\_{DateTime(format=yyyyMMdd,offsetType=M,offsetValue=-2)}

例 : {DateTime(format=yyyyMMdd,offsetType=M,offsetValue=-2)}

→ システム日付から2か月前を指定する

# 置換文字列について

コマンド編集にて、いくつかの設定項目で置換文字列を使用してシステム固有の値を自動設定する事が出来ます。

◆RESTコマンド “パラメータ設定”の “ParameterVaule”で利用できる置換文字列

| 置換文字列             | 置換される値             |
|-------------------|--------------------|
| {ConMasServerUrl} | 基本設定のConMasサーバーURL |
| {ConMasUser}      | 基本設定のConMasユーザー    |
| {ConMasPassword}  | 基本設定のConMasパスワード   |

|         |   |               |                  |
|---------|---|---------------|------------------|
| パラメータ設定 |   | ParameterName | ParameterValue   |
|         | ▶ | command       | Login            |
|         |   | user          | {ConMasUser}     |
|         |   | password      | {ConMasPassword} |
|         | * |               |                  |

# 置換文字列について

コマンド編集にて、いくつかの設定項目で置換文字列を使用してシステム固有の値を自動設定する事が出来ます。

- ◆ 実行ファイルコマンド “パラメータ設定”、保存ファイル名で利用できる置換文字列  
※フォルダ監視ジョブのみ

| 置換文字列               | 置換される値            |
|---------------------|-------------------|
| {ConMasWatcherFile} | トリガーとなったファイルのフルパス |

パラメータ設定

# 置換文字列について

コマンド編集にて、いくつかの設定項目で置換文字列を使用してシステム固有の値を自動設定する事が出来ます。

## 【補足】

画面上部の「凡例一覧」を押下すると、凡例一覧が表示されます。  
置換文字列ボタンを押下すると、クリップボードへコピーします。

クリック

凡例一覧

保存

凡例一覧

|                               |  |
|-------------------------------|--|
| {ConMas.JobId}                | 保存ファイル名に指定可能。“JOBID”に置換されます。                             |
| {ConMas.JobName}              | 保存ファイル名に指定可能。“JOB名称”に置換されます。                             |
| {ConMas.ProcessIndex}         | 保存ファイル名に指定可能。“コマンド実行順序番号”に置換されます。                        |
| {ConMas.ProcessName}          | 保存ファイル名に指定可能。“コマンド名称”に置換されます。                            |
| {ConMas.CommandIndex}         | 保存ファイル名に指定可能。フォルダ監視ジョブのアップロードファイル名に指定可能。監視ファイルパスに置換されます。 |
| {ConMas.CommandName}          | 保存ファイル名に指定可能。システム日付で置換されます。〈日付フォーマットは変更可能〉               |
| {ConMas.WatcherFile}          | 保存ファイル名に指定可能。RESTパラメータに指定可能。基本設定の“ConMasサーバーURL”に置換されます。 |
| {DateTime(format="yyyyMMdd")} | RESTパラメータに指定可能。基本設定の“ConMasユーザー”に置換されます。                 |
| {ConMas.ServerUrl}            | RESTパラメータに指定可能。基本設定の“ConMasパスワード”に置換されます。                |
| {ConMas.User}                 |  |
| {ConMas.Password}             |  |

クリックすると、置換文字列をクリップボードにコピーします。

# ジョブ実行ログ

ジョブの実行ログを参照します。

最新の状態を取得します。

表示レコード件数を指定します。

ジョブ名称でLIKE検索を行います。

| DATE_TIME     | JOB_NAME   | JOB_TYPE | PROCESS_INC | PROCESS_NAI | PROCESS_TYF | COMMAND_IN | COMMAND_NA    | COMMAND       |
|---------------|------------|----------|-------------|-------------|-------------|------------|---------------|---------------|
| 2012-09-05... | スケジュールジ... | 1        | 0           | 帳票ダウンロ...   | 0           | 2          | Logout        | http://localh |
| 2012-09-05... | スケジュールジ... | 1        | 0           | 帳票ダウンロ...   | 0           | 1          | [CSV]Downl... | http://localh |
| 2012-09-05... | スケジュールジ... | 1        | 0           | 帳票ダウンロ...   | 0           | 0          | Login         | http://localh |

## 【補足】

COMMAND\_RESULTの値が 0、1 以外の場合は、何らかの異常があった状態です。  
COMMAND\_MESSAGEの内容を確認してください。

※実行したREST処理内、EXE内部で発生したエラーについては、Generatorでは判定出来ませんので、ご了承ください。

## 【注意事項】

下記のサーバーOSではジョブ実行ログ参照画面は表示できません。

- ・ Windows Server 2008
- ・ Windows Server 2008R2
- ・ Windows Server 2012
- ・ Windows Server 2012R2
- ・ Windows Server 2016

以下の場所にファイルログを出力していますのでこちらを参照してください。

出力先：C:\¥ConMas¥ConMasGenerator¥Files¥logs¥rootConMasJob.日付.log

# ジョブ実行ログ 項目説明

- **DATE\_TIME**  
ジョブの実行日時
- **JOB\_NAME**  
Generatorで登録したJOB名称
- **JOB\_TYPE**  
0:フォルダー監視ジョブ  
1:スケジュールジョブ（毎日）  
2:スケジュールジョブ（毎週）  
3:スケジュールジョブ（毎月）  
4:スケジュールジョブ（一回のみ）
- **PROCESS\_INDEX**  
実行中プロセスの実行順序（0スタート）
- **PROCESS\_NAME**  
実行中プロセス名称
- **PROCESS\_TYPE**  
0:REST処理実行（カスタムRESTプロセス）  
1:ローカル実行ファイルの実行（カスタムEXEプロセス）
- **COMMAND\_INDEX**  
実行中コマンドの実行順序（0スタート）
- **COMMAND\_NAME**  
実行中コマンド名称
- **COMMAND**  
実行プログラム（実行コマンド）
- **COMMAND\_RESULT**  
コマンドの結果コード 0:実行成功 1:実行失敗
- **COMMAND\_RESULT\_MESSEGE**  
コマンドの結果メッセージ COMMAND\_RESULT=0の場合はNormalExit 1の場合はエラー内容
- **JOB\_ID**  
JOBの一意名称ID

# メール設定

メール設定を行う事で、エラーメールを送信できます。

クリックして保存

エラーメールを送信する

SMTPサーバー

ポート

送信者アドレス

パスワード

パスワード(確認用)

SSL設定  SSL暗号化する

| 送信先アドレス   |
|-----------|
| ▶ address |
| *         |

送信先アドレス

- ・ **エラーメールを送信する**  
チェックすると、エラーメールを送信します。
- ・ **SMTPサーバ**  
メール転送サーバを設定します。
- ・ **ポート**  
ポート番号を設定します。
- ・ **送信者アドレス (アカウント)**  
送信者アドレスを設定します。
- ・ **パスワード**  
送信者アドレスのパスワードを設定します。
- ・ **SSL**  
SSL暗号化する場合はチェックします。
- ・ **送信先アドレス**  
送信するアドレスを設定します。(複数設定可能)



# ファイルログについて

以下の場所にConMas Generatorのプログラムログを出力します。  
何か問題が発生した場合の対処などにご利用下さい。

出力先 : C:¥ConMas¥ConMasGenerator¥Files¥logs

**ConMasGenerator.日付.log**

⇒フォルダ監視サービスの実行ログ

**rootConMasJob.日付.log**

⇒ジョブ実行ログ

**ConMasGeneratorUtility.日付.log**

⇒画面UI（ユーティリティ）の実行ログ

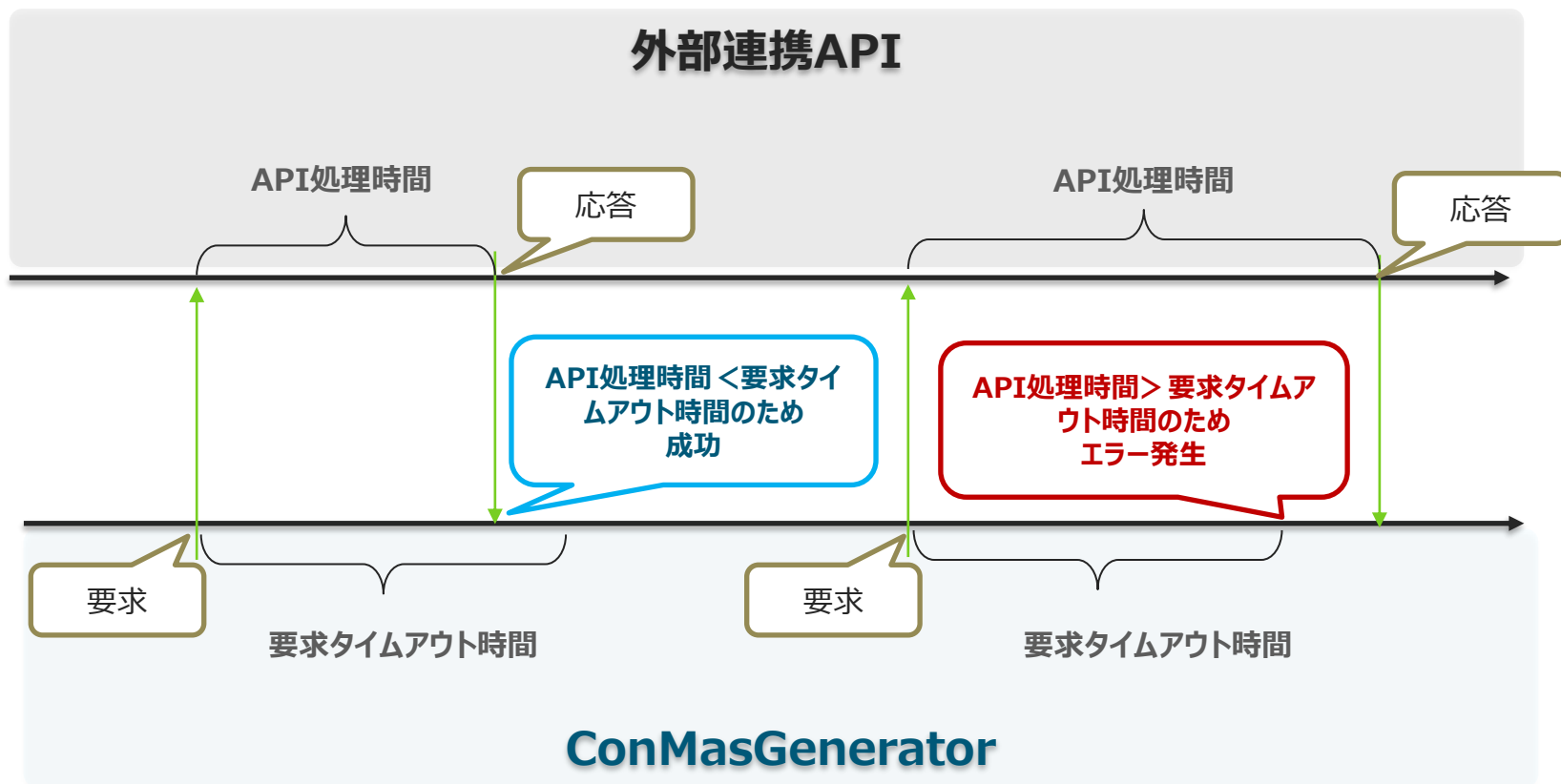
# ご利用上の注意

**ConMas GeneratorはWebAPIやローカルEXEをコールするクライアントツールです。  
簡単にAPIのコールができる反面、細かな制御はできません。  
以下を参照の上で、使用をご検討ください。**

- 要求タイムアウト
- セッションタイムアウト
- APIの処理結果とGeneratorの正常、異常判定の違い

# 要求タイムアウト

外部連携APIとはサーバーサイドで実行されるRest形式のWebAPIです。  
ConMasGeneratorはAPI、EXEをキック（要求して応答をうける）するクライアントツールとなります。  
APIはサーバーサイドで動作しているため、GeneratorはAPI要求から応答までの間、処理を待ちます。  
この待つ時間には制限があり、この制限時間を超えることを要求タイムアウトと呼びます。  
※要求タイムアウトが発生するとGeneratorはエラーとなりジョブは失敗します。



# 要求タイムアウトの回避

要求タイムアウト回避には以下の2つがあります。

- ConMasGeneratorの要求タイムアウト時間を引き延ばす
- APIの処理時間を考慮したコール内容にする（推奨）

例)

自動帳票作成APIを実行する

- 1 帳票あたりAPI処理時間が 5 s
- 要求タイムアウト 60s

上記の場合、一回の要求でXML（CSV）に記載する帳票数は要求タイムアウト内である60sを超えないよう調整する。

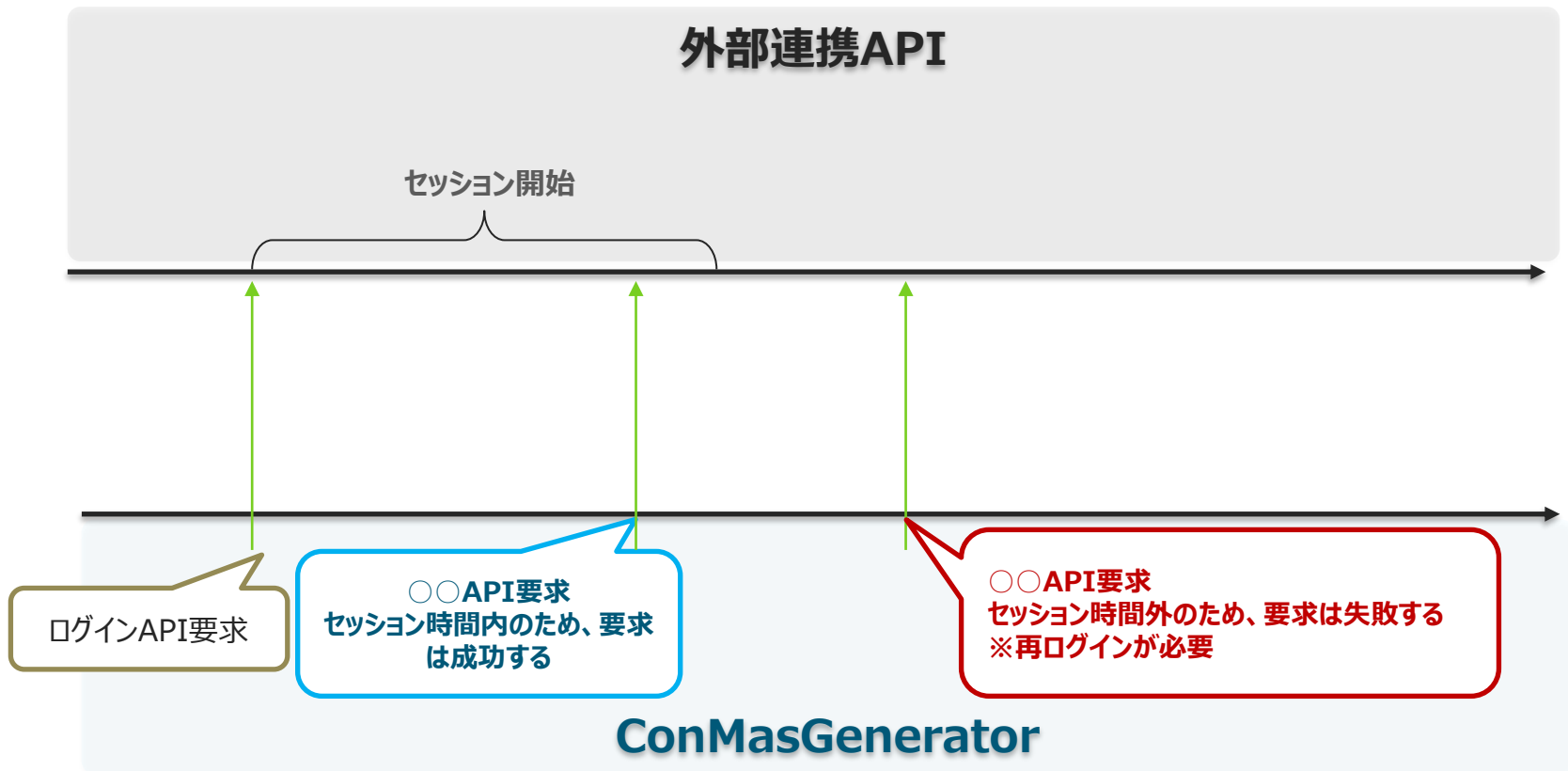
上記例では、「要求タイムアウト 60s > 1帳票あたりのAPI処理時間 5s × 帳票数」

帳票数 = 11帳票以下にする

※APIの処理時間はサーバー状態により変動するため、余裕を持った設定にすることを推奨します

# セッションタイムアウト

外部連携APIのコールが許可されているのはセッションが維持できる時間までです。  
セッションは時間によってタイムアウトし、その後は再度ログインAPIのコールが必要となります。



# セッションタイムアウト レスポンス

セッション切れはログインAPI以外のすべてのAPIが以下レスポンスを返します。

```
<conmas>  
  <error>  
    <code>5</code>  
    <remarks>  
      <remark>…</remark>  
    </remarks>  
  </error>  
</conmas>
```

この値は保存ファイルの中身を確認することで分かります。

※保存ファイル名がXXXXXX.zipであっても、この場合はXMLファイルです。

# セッションタイムアウトの対策

## ■ 対策 1

- ・保存ファイルを確認するプログラムを開発し、カスタムEXEを利用して判定、再ログインする

## ■ 対策 2

- ・APIコール直前にログインAPIをコールすることで発生率は大きく下がります。

## ■ セッションタイムアウトにおけるConMasGenerator制限事項 再ログインは行われません。

セッションタイムアウトの場合はセッションタイムアウトレスポンスが保存ファイルに格納されジョブは正常終了となります。

# APIの処理結果とGeneratorの正常、異常判定の違い

APIの処理結果は保存ファイルとして格納されます。

Generatorでは本ファイルの中身までは確認せず、保存フォルダーに保存ファイルとして格納したら“正常”と判断します。

“異常”はGenerator上でAPIのコールに失敗、保存ファイルが受け取れない（要求タイムアウト）により発生します。

以下例)

- ・セッションタイムアウト
- ・APIの要求パラメーター不正
- ・各APIでの例外

など

※細かなエラー内容は外部連携 WebAPI I/F仕様書を参照

このため、各APIの結果は保存フォルダー内の保存ファイルを確認する必要があります。

**GeneratorはAPIの成否は判定しないツールです。**

**このため、要求パラメーターの間違いなどが無いよう、ジョブ実行を必ず実施し保存ファイルを確認してご利用ください。**



# 改修履歴

| 日付         | Version | 修正内容                              |
|------------|---------|-----------------------------------|
| 2019.09.05 | 1.0.9   | プロキシのユーザー、パスワードが逆転している不具合を修正しました。 |
|            |         |                                   |
|            |         |                                   |
|            |         |                                   |
|            |         |                                   |