

個人信用情報照会システム

Seiko Trust 個信サービス

HULFT Square 連携

概要資料

第 1.0 版

2025 年 11 月 17 日

セイコーソリューションズ株式会社

はじめに

本資料は、弊社が提供する「Seiko Trust 個信サービス」と、株式会社セゾンテクノロジーが提供する「HULFT Square」との連携に関する概要資料となります。

弊社は、両サービス間での技術的な接続確認および運用上の有効性を確認することを目的として技術検証を行いました。本資料では、技術検証内容に基づいた「HULFT Square」と「Seiko Trust 個信サービス」の連携方法についてご紹介いたします。

「Seiko Trust 個信サービス」とは

「Seiko Trust 個信サービス」（以下、「個信サービス」）は、お客様環境と個人信用情報機関（CIC・JICC・KSC）をセキュアに一括接続し、個人向けローンにおける審査業務のサポートをするクラウドサービスです。金融機関やノンバンク 170 社以上の導入実績をもつ、個人信用情報照会用パッケージソフトウェア「L-CRIP（エルクリップ）」の長年の開発ノウハウを活かし、個信業務の省力化・効率化を実現します。

「HULFT Square」とは

「[HULFT Square](#)」は株式会社セゾンテクノロジーが提供する日本発 iPaaS（Integration Platform as a Service）です。

オンプレミスとクラウド、社内外に分散した様々なデータを安全・確実に、一元的に連携・活用できる環境を提供します。拡張性のあるシステム間連携をユーザーフレンドリーな操作性によりノーコード開発を実現します。企業内に分散するデータ活用と業務効率化を推進します。

連携の概要と効果

個信サービスと「HULFT Square」との連携では、「HULFT Square」が用意する各種コネクタ・データ連携機能と個信サービスが用意するインターフェースを使用します。「HULFT Square」を使って個信サービスにデータを登録し、個信サービスからデータを取得することで、個人信用情報照会の効率化を実現します。

本書で使用する用語

本書では、「HULFT Square」が用意する画面上での操作・設定を扱っています。そのため、「HULFT Square」固有の用語を使用している場合があります。

<「HULFT Square」で使用される主な用語と概念>

以下の URL をご参照ください。

<https://www.hulft.com/help/ja-jp/HULFTSquare/Content/TOP/Glossary/glossary.htm>

<個信サービスに関連した主な用語と概念>

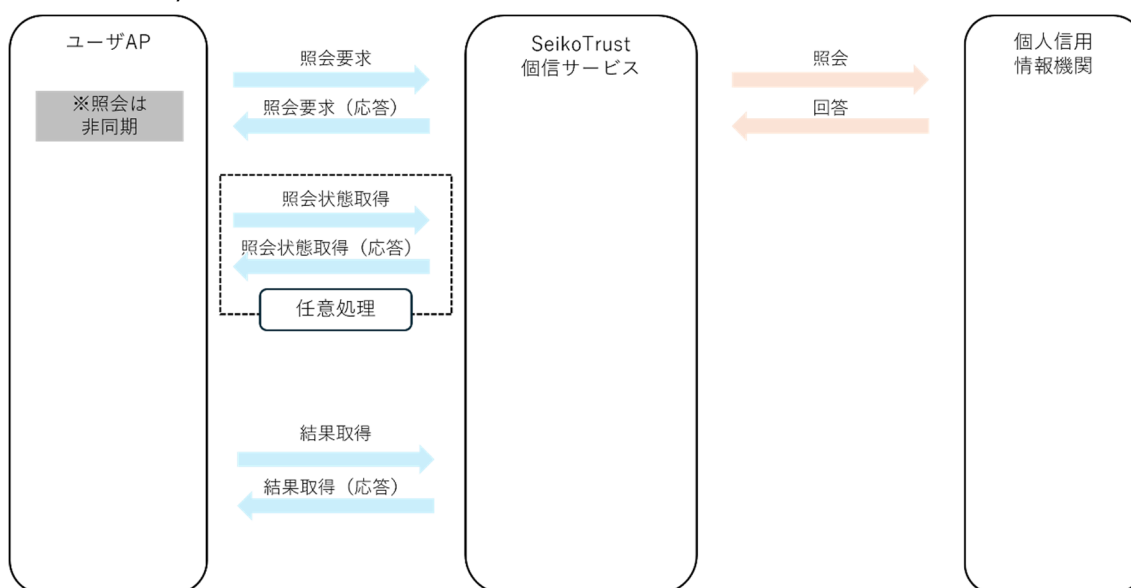
下表のとおりです。

用語・概念	説明
照会	個人信用情報機関に申込者の個人信用情報を確認するための依頼を行うこと。
回答	個人信用情報機関への依頼（照会）に基づいて取得することができる個人信用情報。照会結果。
ユーザ AP	個信サービスからみたアクセス元のシステム。本書では、HULFT Square。
案件番号	ユーザが設定する一意の番号。個信サービス内の管理用番号として使用する。既に個信サービスに登録されている案件番号を設定した場合、エラーとなる。

個信サービスのインターフェース

個信サービスは、ユーザ AP とのインターフェースに、REST API 形式の JSON でのインターフェースを採用しています。

ユーザ AP と個信サービスは、サービスを利用するユーザ AP をクライアント、個信サービスをサーバとするサーバ/クライアント構成となります。



「照会要求」と「結果取得」は非同期になり、個人信用情報機関からの回答状況は、照会状態取得にて確認する必要があります。個人信用情報機関からの回答が来ない段階で、ユーザ AP から結果取得を行うと、「処理中」のステータスが返却されます。

個信サービス（照会機能）で提供するインターフェースは次のとおりです。

No.	大項目	項目	物理名	内容
1	要求	照会要求	requestInquiry	個人信用情報機関に照会を行うためのインターフェース
2	要求	照会要求	requestInquiryKscOwnInfo	KSC の自社提出情報照会を行うためのインターフェース ※本書の対象外となります
3	要求	照会要求	requestInquiryKscKanpoPersonal	KSC の官報情報（個人）照会を行うためのインターフェース ※本書の対象外となります
4	監視	照会状態取得	checkInquiryStatus	照会要求の照会状況（正常完了、照会中、エラーなど）を確認するためのインターフェース
5	結果取得	照会結果取得	getJudgement	個信サービス独自仕様で、判定した結果をサマリーしたデータを取得するためのインターフェース

動作概要

1. 全体の概要

利用ユーザからの入力情報（照会要求）を CSV ファイルと仮定し、次の流れを検証対象としております。

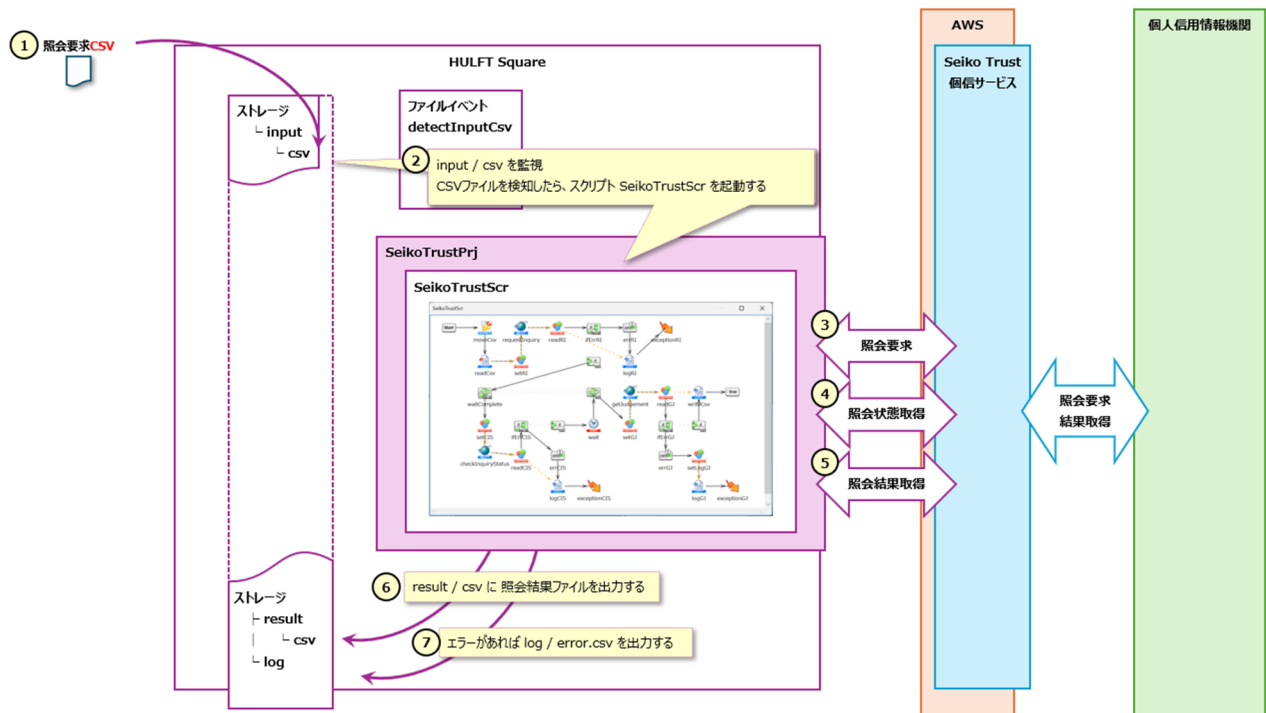
- ・ HULFT Square 上に格納された照会要求ファイル（CSV）から、個信サービスの API 形式に変換
- ・ 個信サービスへ照会を依頼
- ・ 個信サービスから個人情報情報機関へ照会
- ・ 個人情報情報機関から回答を取得
- ・ 回答結果を HULFT Square 上に CSV ファイルで格納

照会要求ファイル、照会結果ファイルの格納方法やレイアウトは、ご利用される業務内容（システム要件）に合わせる必要があります。本資料は HULFT Square のスクリプトで構成の参考として参照下さい。

※本検証では、個人情報情報機関の代用として、個信サービス上に設置したシミュレータを使用しています。個信サービスからシミュレータに対して照会を行うと、シミュレータ上に個信サービスから取得可能な回答用のデータが生成されます。

動作時に使用するファイルとディレクトリは次のとおりです。

No.	ディレクトリ	説明	補足
1	/input/csv/(任意名).csv	照会要求 CSV ファイルの配置場所	拡張子が「.csv」であれば、ファイル名は任意
2	/result/csv/(案件番号).csv	照会結果ファイルの出力場所	結果出力ファイルは、自動的に「(案件番号).csv」となる
3	/log.error.csv	エラーログの出力場所	固定名



2. ファイルイベントについて

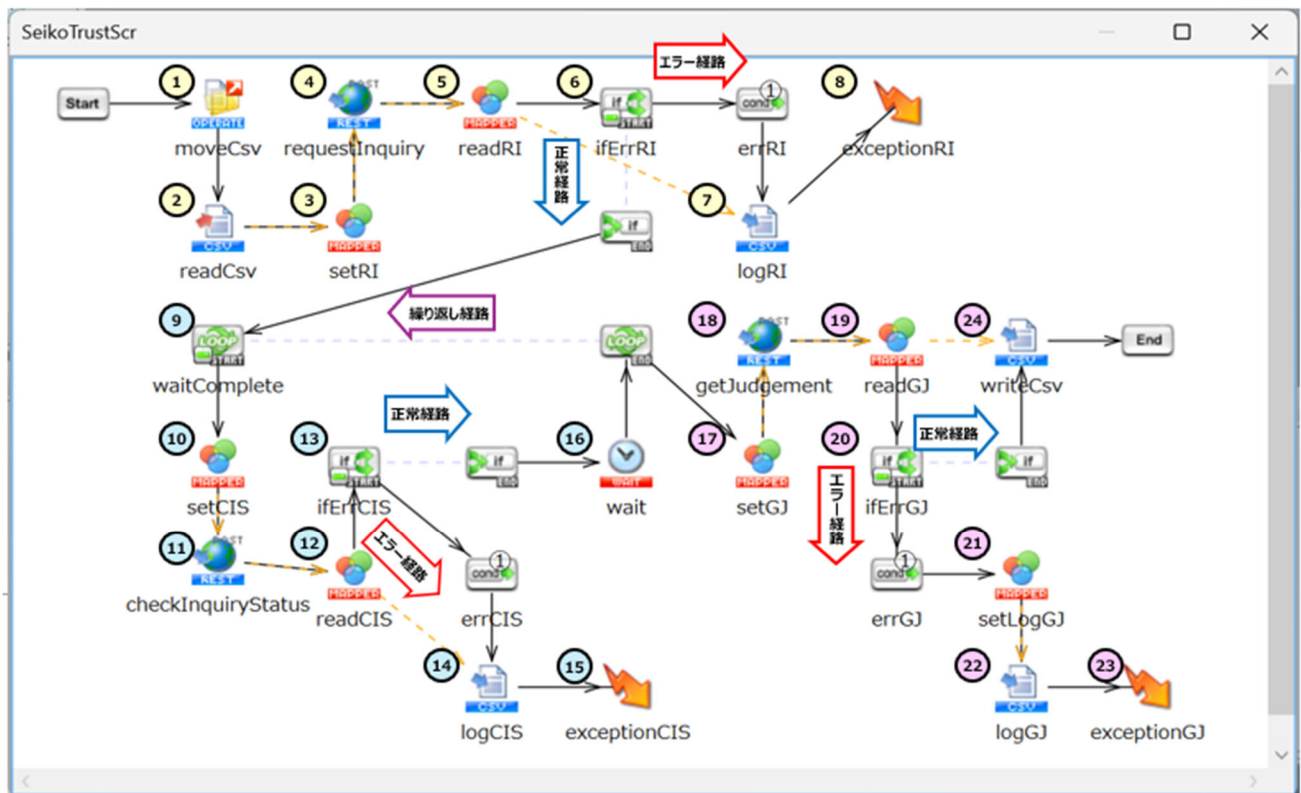
[input/csv ディレクトリ] を監視するファイルイベント [detectInputCsv] を作成します。

[detectInputCsv] はファイルが配置される都度、スクリプト [SeikoTrustScr] を起動します。

スクリプトはファイルを検知した時点で起動し、複数のスクリプトは並列に動作します。同じ時間帯に複数の処理を実行可能です。



3. スクリプトの内容



No.	名前	動作
1	moveCsv	input/csv/XX.csv を work に移動する
2	readCsv	work/XX.csv を読み込む
3	setRI	requestInquiry のため、リクエスト JSON を準備する
4	requestInquiry	個信サービスに、APIrequestInquiry を送信、レスポンスを受信する
5	readRI	requestInquiry のレスポンス JSON を読み込む
6	ifErrRI-errRI	requestInquiry にエラーがあれば 7 へ進める
7	logRI	requestInquiry のエラーをログ出力する
8	exceptionRI	フローをエラー終了にする
9	waitComplete	11:checkInquiryStatus の結果が取得完了になるまで繰り返すためのループ定義
10	setCIS	checkInquiryStatus のため、リクエスト JSON を準備する
11	checkInquiryStatus	個信サービスに、APIcheckInquiryStatus を送信、レスポンスを受信する
12	readCIS	checkInquiryStatus のレスポンス JSON を読み込む
13	ifErrCIS-errCIS	checkInquiryStatus にエラーがあれば 14 へ進める
14	logCIS	checkInquiryStatus のエラーをログ出力する
15	exceptionCIS	フローをエラー終了にする
16	Wait	1 秒間、待機する

(前頁続き)

No.	名前	動作
17	setGJ	getJudgement のため、リクエスト JSON を準備する
18	getJudgement	個信サービスに、APIgetJudgement を送信、レスポンスを受信する
19	readGJ	getJudgement のレスポンス JSON を読み込む
20	ifErrGJ-errGJ	getJudgement にエラーがあれば 21 へ進める
21	setLogGJ	ログ出力のため、getJudgement のレスポンス JSON を読み込む
22	logGJ	getJudgement のエラーをログ出力する
23	exceptionGJ	フローをエラー終了にする
24	writeCSV	getJudgement のレスポンスより、取得結果 CSV を出力する