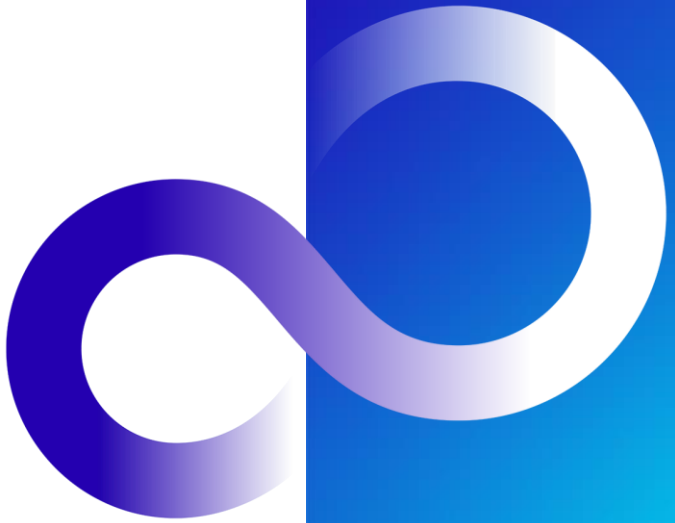
基本設計書
参加者識別子管理機能等

2024/02/14



1 機能概要

コード	機能	説明	要件	I/F	共通
DEX-1C	来歴管理	・コネクタ機能を介して収受されるデータの収受、供与の来歴を記録管理するAPIを提供する ・DDP送信時・受信時などの来歴情報を管理する	・データ交換（コネクタ）機能と連携し、DDP提供時、及びDDP受領時に来歴を登録する ・DDP送信時、受信時などの来歴を登録する ・来歴取得依頼に応じて、産業用データ連携基盤参加者に関わる来歴を返却する	・来歴登録 ・来歴確認(来歴ID検索) ・来歴確認(eventkey検索)	IDP連携（認可）
DEX-2B	電子証明書管理	・分野間データ連携基盤として参加機関のTLS証明書を管理し、コネクタ機能を介したデータ送受信時の参加機関の真正性を担保するために活用する	・システム外で外部機関（発行局）から発行されたTLS証明書をDBに登録する ・コネクタからの要求に対し、TLS証明書を返却する	・電子証明書登録 ・電子証明書取得 ・電子証明書削除	
DEX-2C	電子署名管理	・分野間データ連携基盤上で授受されるデータセット等のオブジェクトの真正性、完全性を証明するための電子署名管理機能を提供する	・分野間データ連携基盤の参加機関向けの自己証明書を作成する ・自己証明書を利用し、Data e-TRUST内でオブジェクトと署名紐づける(トラストシールIDを発行) ・Data e-TRUSTを利用し、トラストシールIDとオブジェクトから真正性、完全性を検証する	・署名用証明書作成 ・シール作成 ・シール検証	

FUJITSU

1-1 共通機能

1-1-1 共通機能 IDP連携 ①機能概要

FUJITSU

機能概要

対象API実行時にIDP(OpenIdConnect準拠)から発行されたアクセストークンを検証する機能を提供する
認証についてはコネクタ、分野間連携サービスなどの前段のサービスで完了しているものとし、
アクセストークンの受渡しがない場合に、IDPへの認証の問合せ(リダイレクトによる認証画面の表示など)は実行しない

対象機能

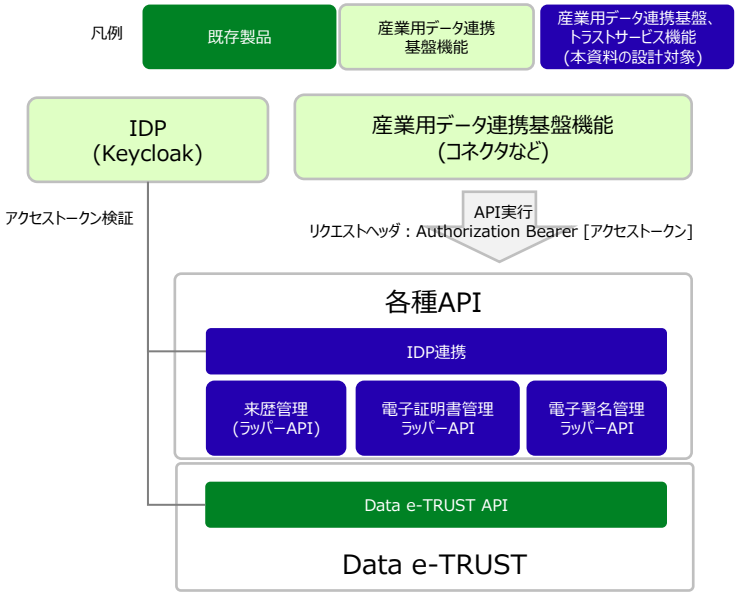
来歴管理
電子証明書管理
電子署名管理
Data e-TRUST

方式

・API実行
対象API実行時に、リクエストヘッダ(Authorization Bearer)に
指定されたアクセストークンを検証する

設定例（リクエストヘッダ）
Authorization Bearer [アクセストークン]

・アクセストークン検証
keycloak-connectorなどOSSを利用しAPIにアクセス時にアクセストークン検証を行う



1-1-1 共通機能 IDP連携 ②アクセストークン検証

FUJITSU

アクセストークン検証

WebAPIの認可としてBearer-Onlyでのアクセスを前提としたアクセストークン検証を行う
以下のOSSライブラリを利用し、アクセストークンのアクティブ検証、Authorization Bearerヘッダの検証を行う

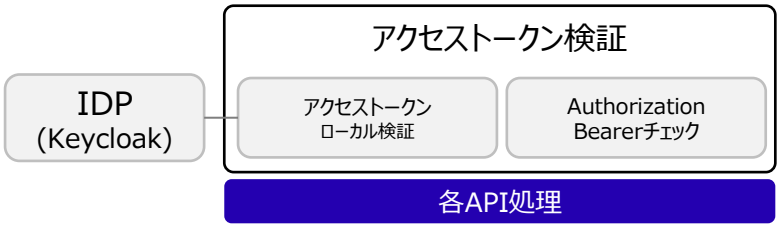
OSSライブラリ：Keycloak Node.js Adaptor(keycloak-connect)
https://www.npmjs.com/package/keycloak-connect

検証方式　　：ローカル検証
アクセストークン形式　：JWS(ジョーズ) (署名付きJWT(ジョット))

keycloak-connect指定情報（例）

Keycloak.js (リソースサーバで指定するKeyCloak接続のための設定情報)

```
{
   "realm": "demo-api",
   "resource": "sample-rest-api",
   "auth-server-url": http://172.17.0.2:8080/auth,
   “bearer-only”: true,
   “realm-public-key”: “xxxxxxxxxxxxxxxxxxxxxxx”
}
```



1-1-1 共通機能 IDP連携 ③Data e-TRUST連携

FUJITSU

Data e-TRUSTでのアクセストークン検証

ラッパーAPIと同様、WebAPIの認可としてBearer-Onlyでのアクセスを前提としたアクセストークン検証を行う
Data e-TRUSTでは、外部IDPより公開鍵を取得し、ローカル検証にてトークンのアクティブ検証を行う

検証方式　　：ローカル検証 (Keycloakのjwks_uri(証明書エンドポイント)から公開鍵を取得し、トークンのアクティブ検証を実施)
アクセストークン形式　：JWS(ジョーズ) (署名付きJWT(ジョット))

※初回接続時に公開鍵を取得し、オンメモリで保管。2回目以降はメモリ上の公開鍵を利用する

FUJITSU

1-2 各種機能

1-2-1 来歴管理 ①機能概要

FUJITSU

機能概要

コネクタなどのフロントサービスで発生するデータの収受、供与の来歴を記録管理するためのAPIを提供する
今回、SIP2での開発物を流用するため、分野間連携サービス、コネクタ機能とCADDE/来歴管理をつなぐラッパーAPIとして機能提供する
具体的には、来歴管理として、履歴登録、来歴確認APIを提供する

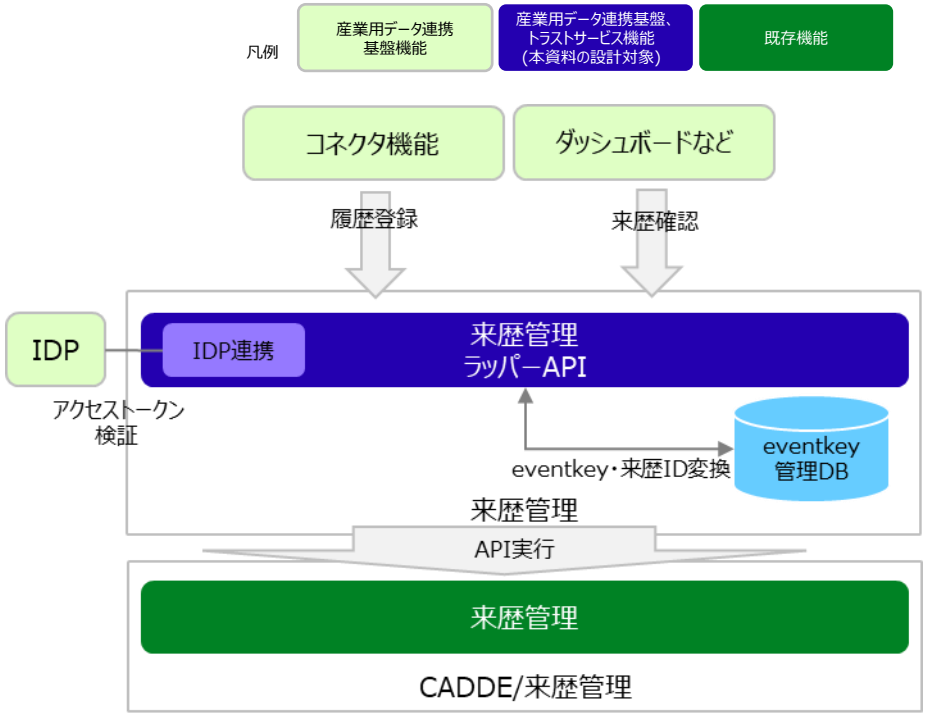
対象機能

来歴管理

提供API

以下、APIを提供する

No.	ラッパーAPI I/F		API概要	連携機能
1	履歴登録API		来歴管理に送信・受信履歴を登録する。 登録された履歴に対し、来歴ID(イベントID)を発行し、返却する。 ※CADDE/履歴登録APIと同等の機能を提供する	コネクタ機能、IDP連携 履歴登録API(CADDE/来歴管理)
2	来歴確認API	来歴ID検索	登録済みの履歴に対して、来歴ID(イベントID)を指定することで、その履歴を基点とした一連の来歴を取得する ※CADDE/来歴確認APIと同等の機能を提供する	ダッシュボードなど、IDP連携 来歴確認API(CADDE/来歴管理)
		eventkey検索	登録済みの履歴に対して、eventkey(任意のキー)を指定することで、eventkeyに紐づく履歴を起点とした一連の来歴を取得する	ダッシュボードなど、IDP連携 来歴確認API(CADDE/来歴管理)



1-2-1 来歴管理 ②ラッパー化対象範囲

FUJITSU

来歴管理ラッパーAPIから、以下のCADDE/来歴管理のAPIを呼び出す

※実装対象 〇：実装対象、×：実証対象外、－：テスト対象外

No.	実装対象	CADDE/来歴管理		対象シーン	概要	呼び元	備考
1	－	履歴登録API	eventwithhash	データ原本情報登録	・データ提供者から原本のデータを受け取ったデータのハッシュ値を計算して来歴IDを含む履歴情報を生成・登録し、カタログに付与する履歴情報の来歴IDを返す	データカタログ作成支援ツール	来歴管理システム設計書v1.4
2	－			二次データ原本情報登録	・二次データ提供者からデータと連結先の来歴IDを受け取った、データのハッシュ値を計算して履歴情報を生成・登録し、カタログに付与する履歴情報の来歴IDを返す	データカタログ作成支援ツール	来歴管理システム設計書v1.4
3	〇			送信履歴登録	・データ提供者側からデータ送信における履歴登録の要求を受け、送信履歴を登録する	コネクタ	来歴管理システム設計書v1.4
4	〇			受信履歴登録	・データ受領者側からデータ受信における履歴登録の要求を受け、受信履歴を登録する	コネクタ	来歴管理システム設計書v1.4
5	－			データ加工履歴登録	・データ受領者あるいはデータ提供者においてデータ加工に伴うデータ加工履歴を登録する	コネクタ	来歴管理システム設計書v1.4
6	〇	来歴確認API	lineage/{event_ID}	来歴確認	・データ受領者からカタログ記載の履歴情報の来歴IDを受け取った、来歴情報を返す ・データ取得後にデータ受領者から履歴情報の来歴IDを受け取った、来歴情報を返す	コネクタ	来歴管理システム設計書v1.4
7	×	履歴検索API	searchevents	履歴ID検索	・データ提供者から検索キーと値を受け取った、対応する来歴IDのリストを返す ※OSS/Apache CouchDBを利用した試験的に実装された全量検索機能 運用上来歴確認APIで要件を満たすと考え、履歴検査のAPIは性能面や運用性などに課題があるため、今回、ラッパー提供の対象外	コネクタ	来歴管理システム設計書v1.4
8	×	ユーザ管理API	adduser	ユーザ情報登録	・ユーザ情報を登録する	来歴管理API	来歴管理システム設計書v1.4
9	×		deluser	ユーザ情報削除	・ユーザ情報を削除する	来歴管理API	来歴管理システム設計書v1.4
10	×		user/enroll	セッションID取得	・管理用セッションIDを取得する	来歴管理API	来歴管理システム設計書v1.4

1-2-1 来歴管理 ③データモデル

FUJITSU

データモデル図
50_来歴管理システム設計書v1.4(※) 4.1 データモデル図を参照のこと

履歴項目
50_来歴管理システム設計書v1.4(※) 4.2 履歴項目を参照のこと

履歴の登録情報
50_来歴管理システム設計書v1.4(※) 4.3 履歴の登録情報を参照のこと

履歴間の関係性
50_来歴管理システム設計書v1.4(※) 4.4 履歴間の関係性設定を参照のこと

データ来歴構成
50_来歴管理システム設計書v1.4(※) 4.5 データ来歴の構成を参照のこと

※CADDE-sip/documentsでの公開情報
格納場所パス：doc/2/50_V4_ユースケース基本設計書/70_V4_設計書_来歴管理システム設計書

1-2-1 来歴管理 ④対象シーン別API利用方法

FUJITSU

履歴登録APIでは、対象シーンに応じてHTTPリクエストBodyパラメータを切り替えることで各シーンでの来歴を登録する
cdlpreviouseventsに、存在しない来歴IDが指定された場合、エラーを返却する
必要に応じて、オプション／ユーザ定義情報として任意のKey／Valueを追加する

1) 送信履歴登録

```
{
  "cdldatamodelversion":"2.0",
  “cdleventtype”:“Sent”,          // イベントタイプ（必須）：送信履歴登録を示す"Sent"を入力
  “eventkey”:”DDP-ID-001”,        // 検索の起点とするイベントキー（オプション／ユーザ定義）
  “dataprovider”:“AAA-BBB”,       // データ送信者（提供者）ID: コネクタ管理のデータ提供者IDを入力 （オプション／ユーザー定義）
  “datauser”:“CCC-DDD”,           // データ受信者（受領者）ID: コネクタ管理のデータ受領者IDを入力 （オプション／ユーザー定義）
  …
  "cdlpreviousevents": []
}
```

2) 受信履歴登録

```
{
  "cdldatamodelversion":"2.0",
  "cdleventtype":"Received",      // イベントタイプ（必須）：受信履歴登録を示す"Received"を入力
  “dataprovider”:“AAA-BBB”,       // データ送信者（提供者）ID: コネクタ管理のデータ提供者IDを入力 （オプション／ユーザー定義）
  “datauser”:“CCC-DDD”,           // データ受信者（受領者）ID: コネクタ管理のデータ受領者IDを入力 （オプション／ユーザー定義）
  …
  "cdlpreviousevents": [
     "aaaa-bbbb-cccc-dddd",       // 前段の来歴ID(イベントID)を入力。
  ]
}
```

1-2-1 来歴管理 ⑤来歴確認

FUJITSU

データ受領者およびデータ提供者が、参照する履歴から過去の履歴を確認できる（例:取得したデータの履歴を確認したい）機能を要し、あるいはそれに続く履歴（例:提供したデータのその後の交換履歴を確認したい）を確認できる機能を提供する

1) 来歴ID検索
登録された来歴を管理するID(来歴ID/イベントID)で特定の履歴を指定し、それを起点とした一連の履歴（前方あるいは後方）を取得する

2) eventkey検索
ダッシュボードなど来歴IDを展開しないシステム用に類推可能なeventkey(任意文字列)を指定することで来歴検索可能な仕組みを提供する
履歴登録時に指定したeventkeyをルートの履歴として指定し、それを起点とした一連の履歴（前方あるいは後方）を取得する
eventkeyに紐づくルートの履歴を起点とし、来歴情報を取得するため、同じeventkeyで異なる送受信の履歴登録した場合、全ての送受信の来歴が返却されるものとする

※eventkeyについて
eventkeyは任意の値を指定可能とするが、システムで一意の値とし、この値と紐づくルートの履歴は1つのみとする

1-2-1 来歴管理 ⑥eventkeyによる来歴管理方式

来歴確認時にDDPやユーザIDなどを利用して作成された任意のeventkeyを指定させ、ダッシュボードなどからの来歴データの検索を容易にする

■想定ユースケース

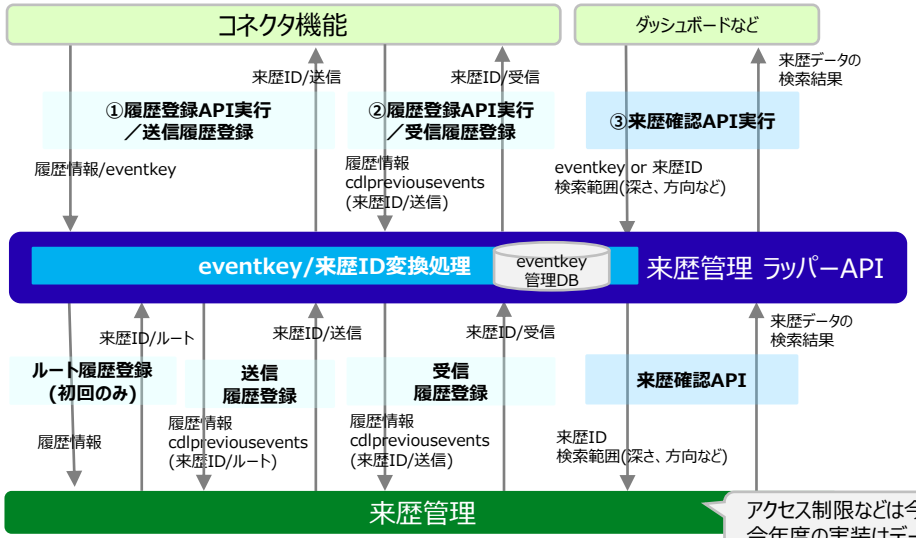
① 履歴登録API実行/送信履歴登録：履歴情報にeventkeyを指定し、送信履歴を登録する
来歴管理ラッパーAPIは、eventkey管理DBに存在しないeventkeyが指定された場合(初回)、ルート履歴を登録し、eventkey管理DBにeventkeyと来歴ID/ルートをセットで保管する
来歴管理ラッパーAPIは、eventkey管理DBにeventkeyが存在する場合、セットで保管された来歴ID/ルートを取得する
来歴管理ラッパーAPIは、cdlpreviousevents に来歴ID/ルートを指定し、送信履歴を登録する

② 履歴登録API実行/受信履歴登録：履歴情報と①で返却された来歴IDをcdlpreviousevents に指定し、送信履歴を登録する(※eventkeyを指定しないこと)
来歴管理ラッパーAPIは、eventkeyが未指定の場合、ルート履歴の登録をせず、指定された履歴情報やcdlpreviousevents から履歴登録を実行する

③ 来歴確認API実行：eventkey、または来歴IDを指定し、来歴データの検索結果を取得する(完全一致のみ)
来歴管理ラッパーAPIは、URIパラメータにeventkeyが指定された場合、 eventkey管理DBに一致する来歴IDを取得し、検索範囲情報と共に来歴を検索する

eventkeyは以下の値を想定する
「DDPの格納先URL(リソースURL)」

履歴登録、来歴確認APIのユースケース



来歴確認APIでの取得データ（検索条件のeventkeyとして「DDP001」を指定した場合）

取得データ例（来歴データの検索結果）

履歴情報(ルート)
dataprovider:A社、datauser:B社
来歴ID：history000
eventkey:DDP001
…
cdlpreviousevents：なし

履歴情報(送信履歴)
dataprovider:A社、datauser:B社
来歴ID：history001
eventkey:DDP001
eventdate:2023/10/01
…
cdlpreviousevents：history000

履歴情報(受信履歴)
dataprovider:A社、datauser:B社
来歴ID：history002
…
cdlpreviousevents：history001

格納データ

履歴情報(ルート)
dataprovider:A社、datauser:C社
来歴ID：history100
eventkey:DDP001-C社
…
cdlpreviousevents：なし

履歴情報(送信履歴)
dataprovider:A社、datauser:C社
来歴ID：history011
eventkey:DDP001-C社
…
cdlpreviousevents：history100

履歴情報(受信履歴)
dataprovider:A社、datauser:C社
来歴ID：history012
…
cdlpreviousevents：history011

アクセス制限などは今年度、協議している内容で、履歴の検索などに利用する属性含め別途協議が必要という認識
今年度の実装はデータ授受ごとにデータを取得する方向性を確立するためにシステムでやり取りするDDP ID相当のキーでの検索を想定

1-2-2 電子証明書管理 ①機能概要

機能概要

分野間データ連携サービスからシステム外の第三者機関から発行されたTLS証明書を登録する機能、コネクタからの指定により登録したTLS証明書を返却する機能を提供する。具体的には、システム外で発行されたTLS証明書をDBに登録し、指定されたTLS証明書を取得/削除する機能を提供する

対象機能

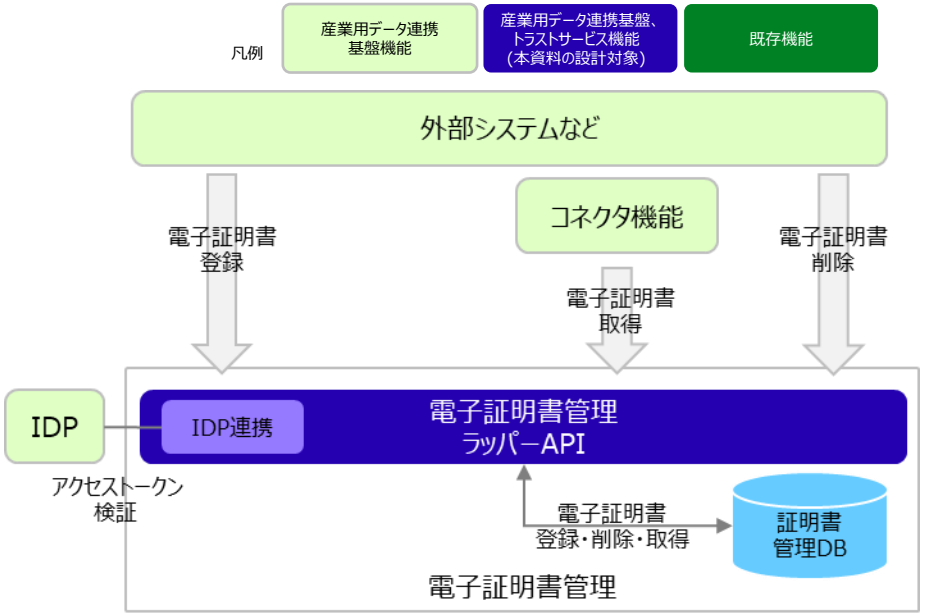
電子証明書管理

提供API

以下APIを提供する

No.	ラッパーAPI I/F	API概要	連携機能
1	電子証明書登録API	システム外部で発行した 電子証明書(String)を登録する	IDP連携
2	電子証明書取得API	指定した電子証明書を取得する (Stringを返却する)	コネクタ機能、IDP連携
3	電子証明書削除API	指定した電子証明書を削除する	IDP連携

リクエストで複数値を処理するのが容易なStringを返却する。



1-2-2 電子証明書管理 ②証明書想定

FUJITSU

証明書として、以下のX509 V3を想定し、証明書を管理する
最終的には、PEM(Privacy-Enhanced mail)形式として、DERでエンコードしたバイト列をBase64エンコードしたテキストの上下をヘッダ(-----BEGIN{label}-----)、フッタ(-----END{label}-----)で囲んだファイルを想定する

参考 X509 v3の証明書フォーマット

No.	項目		名称	型	備考
1	Certificate (証明書全体)	TBSCertificate	証明書の内容	TBSCertificate	証明書の内容
2		signatureAlgorithm	署名アルゴリズム	AlgorithmIdentifier	
3		signatreValue	署名	BIT STRING	

No.	項目	名称		型	備考
1	TBSCertificate (証明書の内容)	Version	バージョン	EXPLICIT Version DEFAULT v1	
2		serialNumber	シリアル番号	CertificateSerialNumber	
3		signature	署名アルゴリズム	AlgorithmIdentifier	
4		issuer	発行者	Name	自己証明書の場合、subjectと同じ
5		validity	有効期限	Validity	
6		subject	主体者	Name	
7		subjectPublicKeyInfo	公開鍵	SubjectPublicKeyInfo	
8		issuerUniqueKey	発行者識別子	IMPLICIT UniqueIdentifier OPTIONAL, -- If present, version MUST be v2 or v3	
9		subjectUniqueKey	主体者識別子		
10		extensitons	拡張	IMPLICT Extensions OPTIONAL, -- If present, version MUST b3 v3	マルチドメイン証明書の場合、 subject alternative namesはここで指定

1-2-2 電子証明書管理 ③管理単位と方式

FUJITSU

証明書管理単位

発行社(Issuer)と主体者(Subject)の組合せで証明書を管理する
APIパラメータとしては、発行者、主体者として企業の識別子（企業ID）を指定する

方式

証明書登録は既存の証明書があっても上書きせずに新規で証明書を登録する
今後、第三者機関からの証明書発行となることを想定し、エビデンスとして発行した証明書は全て残す
証明書取得は上記組合せの最新情報を提供する

証明書削除はDBの容量を鑑み、証明書IDから証明書を削除する機能を提供する

また、今回、証明書の発行を行わないため、証明書失効機能は提供しない

1-2-2 電子証明書管理 ④証明書登録

FUJITSU

証明書登録

証明書登録APIで前述のPEM形式の証明書(文字列)を登録する。将来的に証明書発行機能との連携も鑑み、
証明書のファイル情報に加え、以下のようなX509 V3のsubject情報をDBに格納できるようにしておく
さらに、マルチドメイン証明書の作成も見据え、Subject Alternative Names(SAN)の枠も用意する

※証明書情報として格納するが、以下の情報を元に証明書作成はしない

No.	項目	名称		型	備考
1	TBSCertificate/ subject	CN	コモンネーム	www.dataex.co.jp	
2		OU	部門名	Sales Department	
3		O	組織名	de limited	
4		L	市町村名	Kawasaki City	自己証明書の場合、subjectと同じ
5		ST	都道府県名	Kanagawa	
6		C	国名	JP	
7		emailAddress	組織識別子	detarou@mailaddress.co.jp	
8	TBSCertificate/ subject alternative names(san)	CN	コモンネーム	www.dataex.co.jp	
9		OU	部門名	Sales Department	
10		O	組織名	de limited	
11		L	市町村名	Kawasaki City	
12		ST	都道府県名	Kanagawa	
13		C	国名	JP	
14		emailAddress	組織識別子	detarou@mailaddress.co.jp	

1-2-3 電子署名管理 ①機能概要

FUJITSU

機能概要

Data e-TRUSTのトラストシール機能を利用し、DDPに対しトラストシールを付与・検証する機能を提供することで、分野間データ連携基盤上で授受されるデータセット等のオブジェクトの真正性、完全性を証明する。

具体的には、事前にデータ提供者向けに自己証明書を作成し、データ送信時にDDPに対し作成した証明書から作成したトラストシールを付与する。必要に応じてDDP内の検証対象データ(ハッシュ値)とシールID(トラストシールの識別子)からリモート署名検証を行う

対象機能

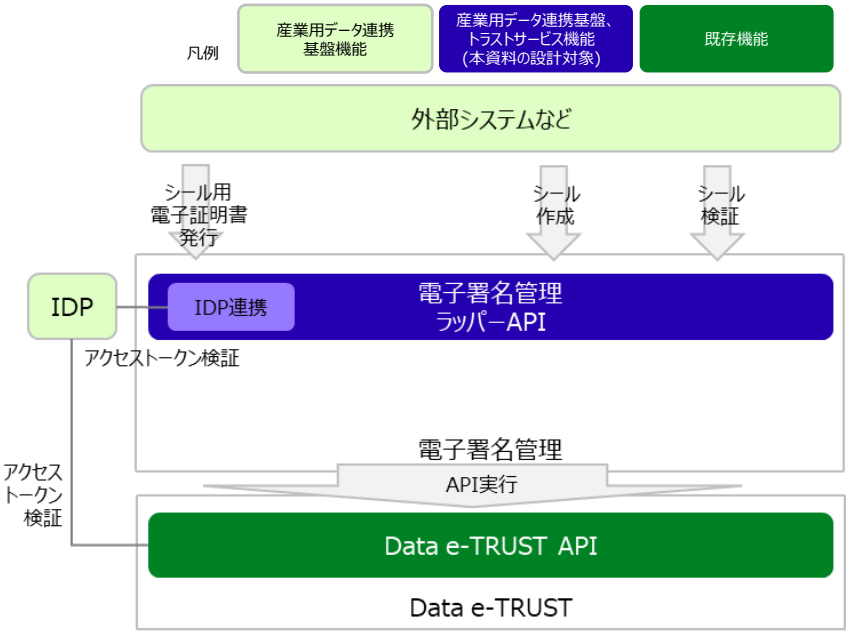
電子署名管理

提供API

以下、APIを提供する

No.	ラッパーAPI I/F	API概要	連携機能
1	署名用 証明書作成API	トラストシールを利用するデータ提供者向けに自己証明書を発行する	外部システム、IDP連携 Data e-TRUST API 証明書作成 Data e-TRUST API 証明書送付
2	シール作成API	指定したオブジェクトのハッシュ値に対し、自己証明書を付与し管理する。トラストシールIDを返却する	外部システム、IDP連携 Data e-TRUST API 受領済証明書取得 Data e-TRUST API トラストシール作成
3	シール検証API	指定したオブジェクトのハッシュ値とトラストシールIDを元にシール検証を行う	外部システム、IDP連携 Data e-TRUST API トラストシール検証

シール作成API
ファイルStreamを受け取ってハッシュ化すると時間を要するので、通信負荷を下げるためにハッシュ値を受領する方式にする

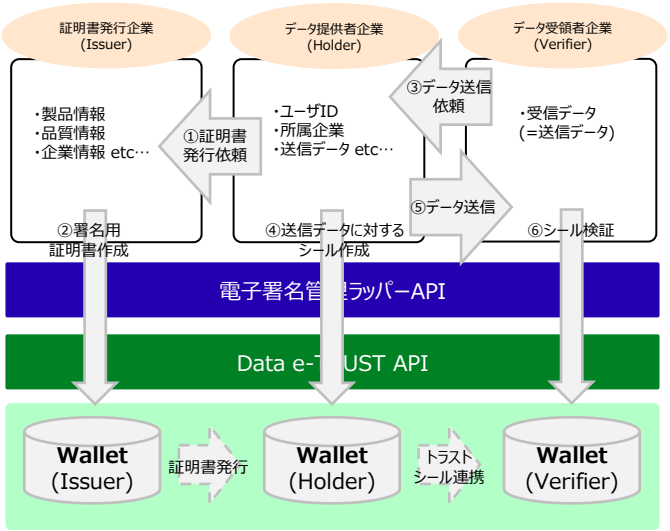


1-2-3 電子署名管理 ②方式

方式：証明書作成からシール付与の流れ

◇事前作業
①Holder　：送信データの真正性を担保するために証明書発行企業(Issuer)に
　　データ提供者/企業(Holder)向けに署名用証明書を発行を依頼する
②Issuer　：依頼に応じて署名用の証明書を発行する

◇データ送信/受信時
③Verifier　：データ受領者/企業(Verifier)がHolderにデータ送信を依頼する
④Holder　：事前に発行された証明書を利用し送信データ(検証対象データ)
　　にシールを付与する
　　(Holderはシール付与時にトラストシールIDを受け取る)
⑤Holder　：送信データと、トラストシールIDをVerifierに送信する
⑥Verifier　：受信したデータ、トラストシールIDからシールの真正性を検証する



1-2-3 電子署名管理 ③署名用証明書

FUJITSU

署名用証明書

署名用証明書として、下記の情報を管理する
証明書情報に加え、Data e-TRUST独自の項目として証明書の種別管理として「Kind」、
証明対象に対する付随情報を入力可能な「Attribute」枠を有します

電子署名法施行規則6条4項 最長5年（通常2～3年）
長期署名：10年（危始化前にタイムスタンプ付与しなおす方式）
⇒一般的な3年に合わせる
(2023.06.29 https://www.cloudsign.jp/media/electronic-signature-deadline/)

署名用証明書 構成

(自己)証明書情報
Validity　：有効期間
　L not_before　：開始日時
　L not_after　：終了日時
Issure　：発行者
Holder　：被証明者(主体者)

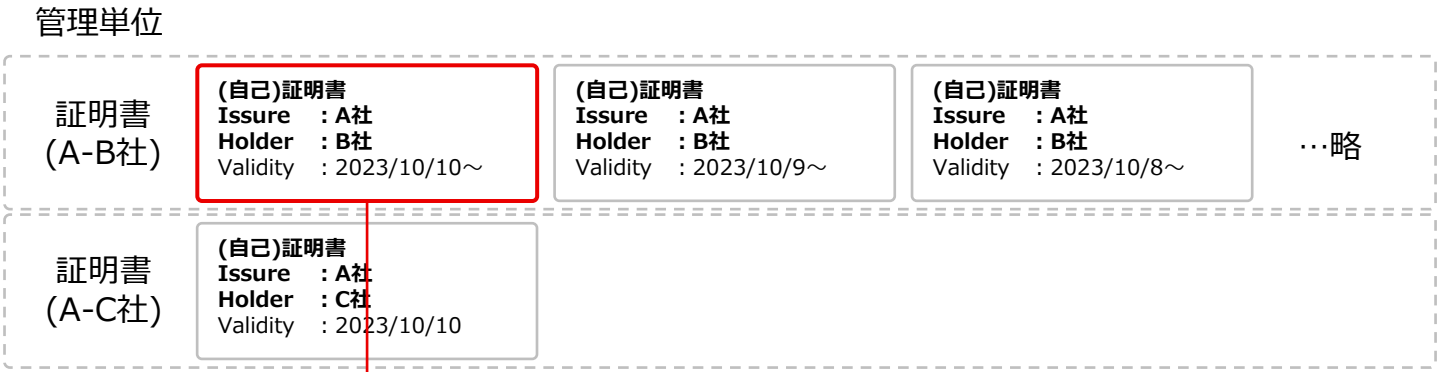
付帯情報（Data e-TRUST 独自管理項目）
Kind　：証明書種別 (任意文字列)
Attribute　：証明書属性情報 (任意のオブジェクト配列)

No	API 設定 項目	項目	設定値	設定例	デフォルト値
1	–	Validity (Required)	署名用証明書作成API実行日から3年で自動指定	–	–
2	○	Issuer (Required)	署名用証明書作成APIに指定する証明書の発行者となる企業ID	DSA	–
3	○	Holder (Required)	署名用証明書作成APIで指定する証明書の主体者の企業ID	Fujitsu	–
4	○	Kind (Option)	署名用証明書作成APIで指定する証明書の種別	BudgetMeeting	Null
5	○	Attribute (Option)	署名用証明書作成APIで指定する任意の文字列	{ “ApprovalDate”:”2023/10/10 10:00:00” “Meeting”: “2023th Approval Meeting” }	Null

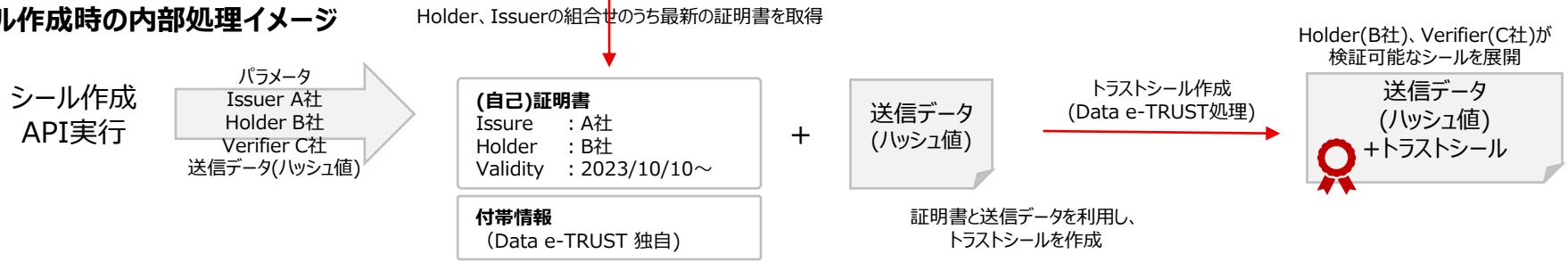
1-2-3 電子署名管理 ③証明書管理単位とシール作成

証明書は、**発行者のIssuer、主体者のHolderの組合せで管理**する
同じ組み合わせで証明書を発行した場合、新規で発行した証明書を有効にする
Holderがシール作成API実行時にパラメータで指定したIssuerから発行された最新の証明書を取得し、トラストシールを作成する

証明書管理イメージ



シール作成時の内部処理イメージ



1-2-3 電子署名管理 ④想定ユースケース

FUJITSU

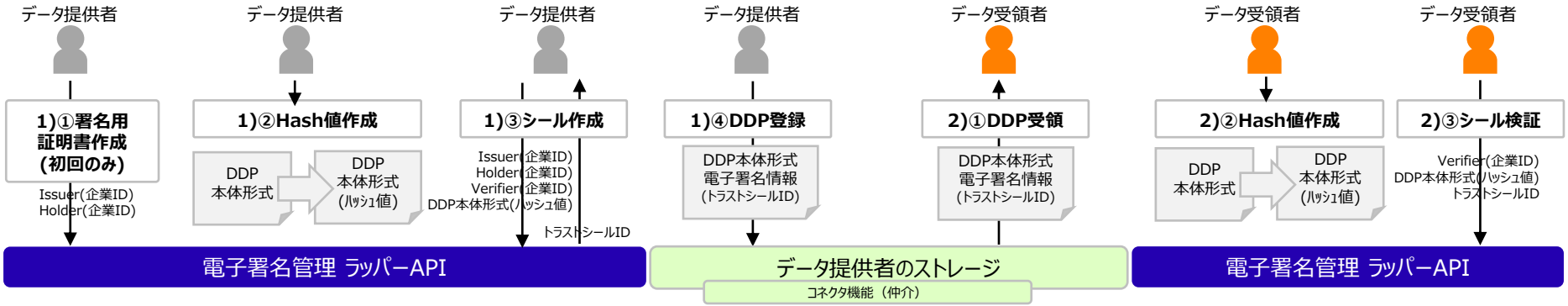
[補足）DDPファイルと電子署名の関係性（今年度の想定）]
今回、JarやZIPファイルなどの圧縮ファイルとしてDDP本体形式を内包するDDPを提供する
DDP本体形式をハッシュ化し、トラストシールIDはファイル名、あるいは、圧縮ファイル内に保持することを想定する
ファイル例
・Jarファイル（ファイル名：任意の場合。内部：DDP本体形式(ハッシュ化対象)、メタ領域(トラストシールID)）
・ZIPファイル（ファイル名：トラストシールIDの場合。内部：DDP本体形式(ハッシュ化対象)） など

今回、システム外での実行を想定し、以下のユースケースとする

1) 事前準備
- ①データ提供者はIssuer、Holderの企業IDを指定し、署名用証明書APIを実行することで署名用の証明書を発行(初回のみ)
  - ※Issuer/Holderで同じ企業IDの指定も可能
- ②データ提供者は、DDP本体形式からハッシュ値を生成
- ③データ提供者は、Issuer、Holder、Verifierの企業IDを指定し、シール作成APIを実行することでトラストシールIDを取得
  - ※Verifierの企業IDはシール作成時点で決められない場合は、システム全体で共通するダミー企業IDを指定することも可能
- ④データ提供者は、トラストシールIDをDDPの電子署名情報領域に格納し、DDPをデータ提供者のストレージに登録する

2) データ送信／シール検証
- ①データ受領者はデータ提供者からDDPを受領する
- ②データ受領者は、DDP本体形式からハッシュ値を生成
- ③データ受領者は、DDPの電子署名情報領域のトラストシールID、
  DDP本体形式から生成したHash値を指定し、シール検証APIを実行することでデータの真正性を確認する

・DDP構造（2023/10時点での想定）
<DDP> ::= <DDP本体形式> [<電子署名情報>]
（※<DDP本体形式> ::= <DDP識別情報> [<約定>] (<データセット付帯情報> <データセット>)+)
⇒今年度は、DDP本体形式をハッシュ化、電子署名情報にトラストシールIDを格納する想定



FUJITSU