

ESP間連携プロトコル 仕様書

目次

1.1	ESP 間連携プロトコル	1
1.1.1	通信プロトコル	1
1.1.2	メッセージ構造	1
1.1.3	ESP 間認証.....	1
1.2	ESP 間連携プロトコルの基本メッセージ	2
1.2.1	SOAP 基本メッセージ	2
1.2.2	メッセージの種類.....	2
1.3	SOAP ボディ共通要素.....	3
1.3.1	各種要素の名前空間.....	3
1.3.2	要素記述例	3
1.3.3	SOAP ボディのルート要素	3
1.3.4	SOAP ボディの共通要素	5
1.4	各種メッセージによるボディ要素の設定	8
1.4.1	Forward メッセージ.....	8
1.4.2	Notify メッセージ.....	11
	付録1	13

1 ESP 間連携プロトコル

ESP間連携プロトコル (Federated Resource Transfer and Transform Protocol) は、ESPを跨ったファイルの転送およびフォーマット変換を行う。

ESP間連携プロトコルは、ESPとESPの間で使うことを想定している。

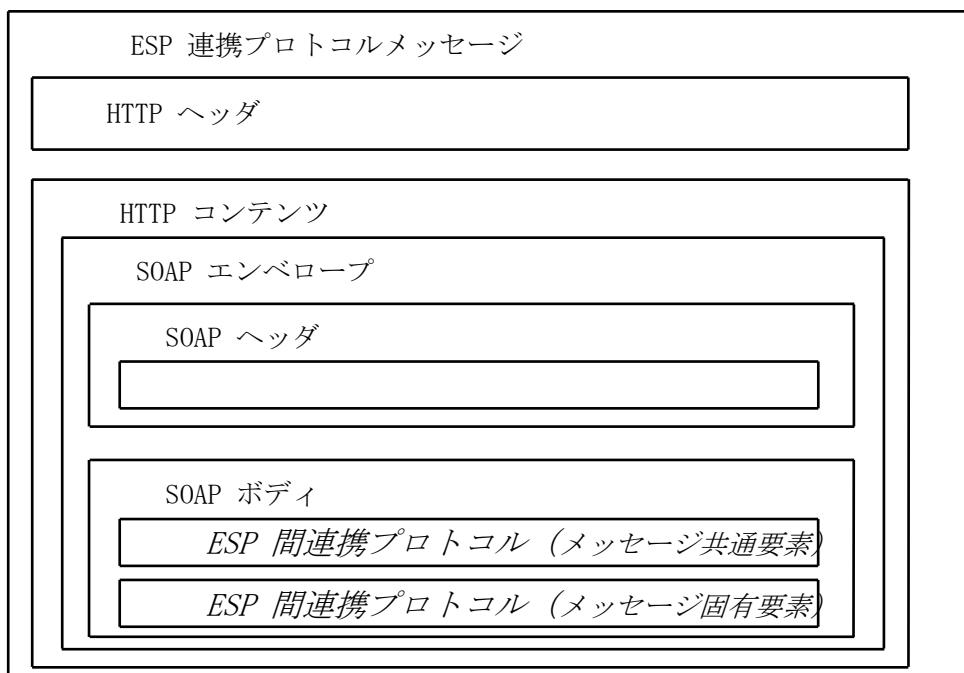
1.1 ESP 間連携プロトコル

1.1.1 通信プロトコル

ESP間でデータを転送する仕組みとしてSOAP 1.2 (Simple Object Access Protocol)を用い、SOAPを転送するプロトコルとしては、HTTPS 1.1 (Hypertext Transfer Protocol Secure)を用いる。

SOAP のメッセージは、本仕様で規定した形式とする。

1.1.2 メッセージ構造



1.1.3 ESP 間認証

ESP間連携プロトコルは独自の認証機能を有しないため、HTTPS 1.1のクライアント認証を用いることを必須とする。サーバー認証は任意とする。

1.2 ESP 間連携プロトコルの基本メッセージ

1.2.1 SOAP 基本メッセージ

```
<s:Envelope xmlns:s=http://www.w3.org/2003/05/soap-envelop
xmlns:frftp="http://frftp.jp/2018/05/">
  <s:Header>
  </s:Header>
  <s:Body>
    ESP 間連携プロトコルメッセージボディ共通要素
    ESP 間連携プロトコルメッセージボディ固有要素
  </s:Body>
</s:Envelope>
```

1.2.1.1 SOAP ヘッダ

ESP間連携プロトコルでは使用しないため、タグから記述しない。

1.2.1.2 SOAP ボディ

全てのメッセージに対して SOAP ボディ内に送信/応答メッセージ共通の要素を持つ。更に、共通の要素の他に SOAP メッセージの種類により、固有の要素を持つ。

1.2.2 メッセージの種類

下記記述内容は、メッセージの内容を示す。

項番	メッセージ	概要	SOAP メソッド
1	Forward	ESP から別 ESP にファイルを送信する。	Forward
2	Notify	受信側 ESP から送信元 ESP に対して状態変化を通知する。	Notify

多くのESPは、ユーザー企業や他ESPからファイル受信しても、宛先のユーザー企業に能動的にファイルを送信しないで、ユーザー企業がファイル取得を行うのを待っている。

この場合、Forwardを行ったESPは最終ユーザー企業のファイル取得完了を知ることが出来ない。

そこで、受信したファイルの状態が、「ファイル取得待ち」から「ファイル取得済」に状態が変化した時などに、Notifyメッセージを使って能動的に送信元ESPに状態変化を通知する。

1.3 SOAP ボディ共通要素

1.3.1 各種要素の名前空間

メッセージの要素を構成する上でのネームスペースの略称を記載する。

項番	名前空間接頭子	URI	備考
1	frttp	http://frttp.jp/2018/05/	ESP 間連携
2	ds	http://www.w3.org/2000/09/xmldsig#	XML 署名

1.3.2 要素記述例

```

<s:Envelope xmlns:s=http://www.w3.org/2003/05/soap-envelope
  xmlns:frttp=" http://frttp.jp/2018/05/">
  <s:Body>
    < ESP間連携ボディルート要素>
      ESP間連携プロトコルメッセージボディ共通要素
      ESP間連携プロトコルメッセージボディ固有要素
    </ ESP間連携ボディルート要素>
  </s:Body>
</s:Envelope>

```

1.3.3 SOAP ボディのルート要素

1.3.3.1 ルート要素項目

メッセージによりボディのESP 間連携ルート要素が以下となる。

項番	メッセージ	送受信	ルート要素
1	Forward	送信時	ForwardRequest
2		応答時	ForwardResponse
3	Notify	送信時	NotifyRequest
4		応答時	NotifyResponse

1.3.3.2 要素記述例

```
<s:Envelope xmlns:s=http://www.w3.org/2003/05/soap-envelope
  xmlns:frftp="http://frftp.jp/2018/05/">
  <s:Body>
    <frftp:ForwardRequest>
      ESP 間連携プロトコルメッセージボディ共通要素
      ESP 間連携プロトコルメッセージボディ固有要素
    </frftp:ForwardRequest>
  </s:Body>
</s:Envelope>
```

メッセージにより、ForwardRequest、ForwardResponse、NotifyRequest、NotifyResponse に置き換わる。

1.3.4 SOAP ボディの共通要素

1.3.4.1 共通要素項目

SOAP ボディの共通要素は、SOAPボディのルート要素の直下に記述される。

項番	ボディ要素	名前空間 接頭子	値		
			メッセージ	送信時 URI	応答時 URI
1	Action	frttp	Forward	ForwardRequest	ForwardResponse
			Notify	NotifyRequest	NotifyResponse
2	To @format	frttp	送信先の連携アドレスを記述する。 (連携アドレスは「3.3.4.3 連携アドレス」参照) 送信時：メッセージ送信先の連携アドレスを記述する。 応答時：メッセージ送信時と同じ連携アドレスをそのまま記述する。 ファイルのフォーマットを記述する From 要素の属性。 フォーマット形式は付録1：フォーマット一覧 参照		
3	From @format	frttp	送信時：メッセージ送信元の連携アドレスを記述する。 応答時：メッセージ送信時と同じ連携アドレスをそのまま記述する。 ファイルのフォーマットを記述する From 要素の属性。 フォーマット形式は付録1：フォーマット一覧 参照		
4	MessageID	frttp	送信メッセージの一意性を保持するための識別子。 重複送信判定等に使用する。 UUID を生成し、以下のフォーマットで記述する。 urn:uuid:XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX 送信時：メッセージ送信元で作成した UUID を記述する。 応答時：メッセージ送信先で作成した UUID を記述する。		

共通要素項目は上記の順に記述する。

1.3.4.2 要素記述例

```

<s:Envelope xmlns:s=http://www.w3.org/2003/05/soap-envelope xmlns:frttp="
http://frttp.jp/2018/05/">
  <s:Body>
    < ESP間連携ボディルート要素>
      <frttp:Action>ForwardRequest</frttp:Action>
      <frttp:To>xxx@aaa</frttp:To>
      < frttp:From>yyy@bbb</frttp:From>
      < frttp:MessageID>urn:uuid:XXXXXXXX-XXXX- ... XXXXXXXX</frttp:MessageID>
      ESP間連携プロトコルメッセージボディ固有要素
    </ ESP間連携ボディルート要素>
  </s:Body>
</s:Envelope>
  
```

1.3.4.3 連携アドレス

連携アドレスは RFC5322 で規定されている addr-spec の形式（電子メールアドレスと同等の形式）とするが、連携アドレスに空白、タブ、改行を含めてはならない。

連携アドレスは以下の構成となる。

username@domain

- username: ローカルの ESP のユーザーアドレス
- domain: グローバルに一意になる ESP アドレス（ドメイン）

ドメインによってESPが特定されるものとする。

※今回使用可能な連携アドレスについては、付録2：連携アドレス一覧 参照

※補足

連携アドレスは RFC5322 で規定されている addr-spec の形式とするが、RFC5322 の規定される addr-spec の内、本仕様では以下の形式だけ扱える。

特に空白、タブ、改行を含めない。

```
addr-spec      = local-part "@" domain
local-part     = dot-atom
domain         = dot-atom / domain-literal
domain-literal = "[" *( dtext) "]"

dtext          = %d33-90 /           ; "[", "]", "¥" を除く印刷可能な
                %d94-126 /         ; US-ASCII 文字

dot-atom-text  = 1*atext *("." 1*atext)
dot-atom       = dot-atom-text

atext          = ALPHA / DIGIT /     ; Printable US-ASCII
                "!" / "#" /         ; specials を除く文字で
                "$" / "%" /         ; atom に使用される
                "&" / "' " /
                "*" / "+" /
                "-" / "/" /
                "=" / "?" /
                "^" / "_ " /
                "`" / "{" /
                "|" / "}" /
```

1.3.4.4 SOAP 通信のエラー発生時のボディ要素

SOAP通信レベルのエラーなどのシステムエラー発生時はSOAP Faultにエラー状況を格納し、返却する。

ESP間連携のメッセージの整合性チェック等のエラーは通常のレスポンス形式で返却し、SOAP Faultは使用しない。

1.3.4.5 SOAP Fault 記述例

```
<s:Envelope xmlns:s=http://www.w3.org/2003/05/soap-envelope
  xmlns:frttp=" http://frttp.jp/2018/05/">
  <s:Body>
    <s:Fault>
      <s:Code>
        <s:Value>[Code]</s:Value>
        <s:Subcode>
          <s:Value>[Subcode]</s:Value>
        </s:Subcode>
      </s:Code>
      <s:Reason>
        <s:Text xml:lang="en">[Reason]</s:Text> </s:Reason>
      <s:Detail>
        [Detail]
        ...
      </s:Detail>
    </s:Fault>
  </s:Body>
</s:Envelope>
```

1.3.4.6 署名方法

本節の記載は省略する。

1.4 各種メッセージによるボディ要素の設定

1.4.1 Forward メッセージ

1.4.1.1 送信メッセージ（送信側）

ボディの固有要素ルート要素（ForwardRequest）に以下の項目を作成する。

項番	ボディ要素	名前空間接頭子	値
1	Data @format @compress	frttp	送信するファイルをBase64 でエンコードしたデータ ファイルのフォーマットを記述するData要素の属性 フォーマット形式は付録1：フォーマット一覧 参照 ファイルを Zip 圧縮（※）してから Base64 でエン コードしている場合、指定する 属性値は、application/zip 固定
2	Signature	ds	発信者が ESP のユーザーであることを証明するための 署名 XML Signature Syntax and Processing Version 1.1 に従って記述する。

共通要素項目に引き続き上記の順に記述する。

※ファイルのZip圧縮に関する注意事項

- ① Zip圧縮はファイル単一とすること。（階層構造で圧縮しないこと）
- ② 圧縮するファイルにパスワードをかけないこと。

上記注意事項に違反する場合、3.4.1.7応答処理中のエラー処理の Protokol仕様以外のエラーとして連携される。

1.4.1.2 応答メッセージ（受信側）

ボディの固有要素ルート要素（ForwardResponse）に以下の項目を作成する。

項番	ボディ要素	名前空間接頭子	値
1	Result	frttp	true:正常に受信処理が出来た場合 false: 受信処理がエラーになった場合
2	Reason	frttp	エラー理由
3	Detail	frttp	エラー詳細情報

共通要素項目に引き続き上記の順に記述する。

1.4.1.3 応答処理 受信時整合性チェック

以下の条件に当てはまる場合、Result タグの値をfalseとし、Reasonタグに下記の理由を入れて返答する。

項番	条件	理由
1	Action が「Forward」と一致しない場合	Unexpected Action
2	To の連携アドレスの形式が誤っている場合	Illegal To Address
3	From の連携アドレスの形式が誤っている場合	Illegal From Address

1.4.1.4 応答処理 変換機能

ESPが変換機能を持つ場合、以下の処理を行う。

(1) 整合性チェック

以下の条件に当てはまる場合、Result タグの値をfalseとし、Reasonタグに下記の理由を入れてForwardResponseを送信する。

項番	条件	理由
1	Dataのformat属性のフォーマット形式とToのformat属性のフォーマット形式が異なり、かつ、Dataのformat属性のフォーマット形式からToのformat属性のフォーマット形式への変換のマッピング定義を持たない場合	Inconvertible Format
2	Fromのformat属性のフォーマット形式とDataのformat属性のフォーマット形式が異なる場合	Mismatching between formats of From and Data
3	格納されているファイルがDataのformat属性のフォーマット形式に合っていない場合 (変換処理中にファイルのフォーマットエラー等が発生した場合を含む)	Mismatching between formats of Data and Content

(2) 変換処理

Dataのformat属性のフォーマット形式からToのformat属性のフォーマット形式への変換のマッピング定義に従い、ファイル形式を変換する。

Dataのformat属性のフォーマット形式をToのformat属性のフォーマット形式に変更する。

変換をすると署名のData要素が変わるため、新たな署名に変更する。

1.4.1.5 応答処理 中継機能

ESPが中継機能を持ち、To の連携アドレスのドメインが自 ESP ではない場合、以下の処理を行う。

(1) 整合性チェック

以下の条件に当てはまる場合、Result タグの値をfalseとし、Reasonタグに下記の理由を入れてForwardResponseを送信する。

項番	条件	理由
1	他ESPを特定できない場合	Unknown Address for Relaying

(2) 中継処理

To の連携アドレスのドメインからESPを特定し、Forward メッセージを送信する。
送信先から受信したForwardResponseをそのまま返却としてForwardしてきたESPに送信する。

但し、他ESPが認証エラー等のSOAP Faultを返却した場合は、返却されたFault情報をそのままFaultとして返答する。

1.4.1.6 応答処理 受信機能

ESPが中継機能を持たない場合、または、To の連携アドレスのドメインが自 ESPの場合、以下の処理を行う。

(1) 整合性チェック

以下の条件に当てはまる場合、Result タグの値をfalseとし、Reasonタグに下記の理由を入れてForwardResponseを送信する。

項番	条件	理由
1	To の連携アドレスのドメインが自 ESPではなく、かつ、自ESP が中継機能を持たない場合	Unknown Domain of Address
2	To の連携アドレスが自ESP のドメインだが、自ESPの登録ユーザー（企業）に該当しない場合	Unknown User of Address
3	To のformat属性が指定されていて自ESPが受け付けられないフォーマットの場合	Unhandled Format
4	他ESPから受信したメッセージのMessageID と重複する場合	Duplicated MessageID of Message

(2) 受信処理

To の連携アドレスが自 ESP のユーザー企業宛ての場合、メッセージを格納し、ユーザー企業が取得可能な状態にする。

Result 要素に true を設定し、ForwardResponseを送信する。

受信処理中にエラーが発生した場合、Result 要素に false を設定し、Reasonタグに理由「その他エラー」を入れてForwardResponseを送信する。

1.4.1.7 応答処理中のエラー処理

障害解析等に使うエラー詳細情報を記述する場合はForwardResponse 要素のDetail タグに記述する。

上記に規定するチェック以外のエラーが検出された場合 (※)、Result タグの値をfalseとし、Reasonタグに理由「Other ERROR」、Detail タグにエラー詳細情報を入れてForwardResponseを送信する。

※共同利用システム側で検知したプロトコル仕様以外のエラーの連携方法について以下の値を設定し、ForwardResponseを送信する。

実証実験時のエラーの解析には事務局に問合せをしていただき、詳細を返答する。

【設定値】

- Result タグの値 : false
- Reasonタグの値 : Other ERROR
- Detail タグの値 : エラー詳細情報

1.4.2 Notify メッセージ

1.4.2.1 送信メッセージ（送信側）

ボディの固有要素ルート要素（NotifyRequest）に以下項目を作成する。

項番	ボディ要素	名前空間接頭子	値
1	MessegeID	frttp	Forward で送信された際の MessegeID を記述する
2	Status	frttp	Received: 受信企業受信済み 受信企業が取得済とした場合

共通要素項目に引き続き上記の順に記述する。

1.4.2.2 応答メッセージ（受信側）

ボディの固有要素ルート要素（NotifyResponse）に以下項目を作成する。

項番	ボディ要素	名前空間接頭子	値
1	Result	frttp	true: 正常に受信処理が出来た場合 false: 受信処理がエラーになった場合
2	Reason	frttp	エラー理由
3	Detail	frttp	エラー詳細情報

共通要素項目に引き続き上記の順に記述する。

1.4.2.3 送信メッセージ発出タイミング

以下の事象が発生した時に、Forward してきた ESPに Statusに下記の状態を入れた NotifyRequestを送信する。

項番	事象	状態
1	受信企業がESP内で受信確認したことを認識した場合	Received

1.4.2.4 応答処理 受信時整合性チェック

以下の条件に当てはまる場合、Result タグの値をfalseとし、Reasonタグに下記の理由を入れてNotifyResponseを送信する。

項番	条件	理由
1	Actionが「Notify」と一致しない場合	Unexpected Action
2	To の連携アドレスの形式が誤っている場合	Illegal To Address
3	From の連携アドレスの形式が誤っている場合	Illegal From Address

1.4.2.5 応答処理 中継機能

ESPが中継機能を持ち、To の連携アドレスのドメインが自 ESP ではない場合、以下の処理を行う。

(1) 整合性チェック

以下の条件に当てはまる場合、Result タグの値をfalseとし、Reasonタグに下記の理由を入れてNotifyResponseを送信する。

項番	条件	理由
1	他ESPを特定できない場合	Unknown Address for Relaying

(2) 中継処理

To の連携アドレスのドメインからESPを特定し、NotifyRequest を送信する。

送信先から受信したNotifyResponseをそのまま返却としてNotifyしてきたESPに送信する。

但し、他ESPが認証エラー等のSOAP Faultを返却した場合は、返却されたFault情報をそのままFaultとして返答する。

1.4.2.6 応答処理 受信機能

ESPが中継機能を持たない場合、または、To の連携アドレスのドメインが自 ESPの場合、以下の処理を行う。

(1) 整合性チェック

以下の条件に当てはまる場合、Result タグの値をfalseとし、Reasonタグに下記の理由を入れてNotifyResponseを送信する。

項番	条件	理由
1	To の連携アドレスのドメインが自 ESP ではなく、かつ、自ESP が中継機能を持たない場合	Unknown Address
2	自ESPからForwardで送信済のMessageID の値の中に、受信したMessageID の値と一致するものが無い場合	Unknown MessageID

(2) 受信処理

MessageIDから特定するForwardした当該のメッセージの状態を受信企業取得済に変更する。

Result 要素にtrueを設定し、NotifyResponseを送信する。

受信処理中にエラーが発生した場合、Result 要素に false を設定し、Reasonタグに理由「その他エラー」を入れてNotifyResponseを送信する。

1.4.2.7 応答処理中のエラー処理

障害解析等に使うエラー詳細情報を記述する場合はForwardResponse 要素のDetail タグに記述する。

上記に規定するチェック以外エラーが検出された場合、Result タグの値をfalseとし、Reasonタグに理由「Other ERROR」、Detail タグにエラー詳細情報を入れてForwardResponseを送信する。

付録1 : WSDL

WSDL 1.1 に基づいて本仕様を記述した WSDL 例を以下に示す。

```
<!-- FRFTP WSDL Version 0.2 -->
<definitions
  ns:wsam="http://www.w3.org/2007/05/addressing/metadata"
  xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
  xmlns:tns="http://frftp.jp/2018/05/"
  xmlns="http://schemas.xmlsoap.org/wsdl/"
  targetNamespace="http://frftp.jp/2018/05/"
  name="FrttpService">

  <types>
    <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" version="0.9"
      xmlns="http://frftp.jp/2018/05/"
      targetNamespace="http://frftp.jp/2018/05/"
      elementFormDefault="qualified">
      <xs:element name="ForwardRequest" type="ForwardRequestType"/>
      <xs:element name="ForwardResponse" type="ForwardResponseType"/>
      <xs:element name="NotifyRequest" type="NotifyRequestType"/>
      <xs:element name="NotifyResponse" type="NotifyResponseType"/>
      <xs:simpleType name="ActionType">
        <xs:restriction base="xs:string">
          <xs:enumeration value="Forward"/>
          <xs:enumeration value="Notify"/>
        </xs:restriction>
      </xs:simpleType>
      <xs:complexType name="AddressType">
        <xs:simpleContent>
          <xs:extension base="xs:string">
            <xs:attribute name="format" type="xs:string"/>
          </xs:extension>
        </xs:simpleContent>
      </xs:complexType>
      <xs:simpleType name="CompressType">
        <xs:restriction base="xs:string">
          <xs:enumeration value="application/zip"/>
        </xs:restriction>
      </xs:simpleType>
      <xs:complexType name="DataType">
        <xs:simpleContent>
          <xs:extension base="xs:base64Binary">
            <xs:attribute name="format" type="xs:string"/>
            <xs:attribute name="compress" type="CompressType"/>
          </xs:extension>
        </xs:simpleContent>
      </xs:complexType>
      <xs:complexType name="SignatureType">
        <xs:sequence>
          <xs:any namespace="##any" processContents="lax" maxOccurs="unbounded"/>
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="ForwardRequestType">
```

```

<xs:sequence>
  <xs:element name="Action" type="ActionType"/>
  <xs:element name="To" type="AddressType"/>
  <xs:element name="From" type="AddressType"/>
  <xs:element name="MessageID" type="xs:anyURI"/>
  <xs:element name="Data" type="DataType"/>
  <xs:element name="Signature" type="SignatureType" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
<xs:complexType name="ForwardResponseType">
  <xs:sequence>
    <xs:element name="Action" type="ActionType"/>
    <xs:element name="To" type="AddressType"/>
    <xs:element name="From" type="AddressType"/>
    <xs:element name="MessageID" type="xs:anyURI"/>
    <xs:element name="Result" type="xs:boolean"/>
    <xs:element name="Reason" type="xs:string" minOccurs="0"/>
    <xs:element name="Detail" type="xs:string" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="NotifyRequestType">
  <xs:sequence>
    <xs:element name="Action" type="ActionType"/>
    <xs:element name="To" type="AddressType"/>
    <xs:element name="From" type="AddressType"/>
    <xs:element name="MessageID" type="xs:anyURI"/>
    <xs:element name="Status" type="xs:string"/>
    <xs:element name="Reason" type="xs:string" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
<xs:complexType name="NotifyResponseType">
  <xs:sequence>
    <xs:element name="Action" type="ActionType"/>
    <xs:element name="To" type="AddressType"/>
    <xs:element name="From" type="AddressType"/>
    <xs:element name="MessageID" type="xs:anyURI"/>
    <xs:element name="Result" type="xs:boolean"/>
    <xs:element name="Reason" type="xs:string" minOccurs="0"/>
    <xs:element name="Detail" type="xs:string" minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
</xs:schema>
</types>

<message name="Forward">
  <part name="parameters" element="tns:ForwardRequest"/>
</message>
<message name="ForwardResponse">
  <part name="parameters" element="tns:ForwardResponse"/>
</message>
<message name="Notify">
  <part name="parameters" element="tns:NotifyRequest"/>
</message>
<message name="NotifyResponse">
  <part name="parameters" element="tns:NotifyResponse"/>
</message>

```



```

<portType name="frttp">
  <operation name="Forward">
    <input wsam:Action="http://frttp.jp/2018/05/ForwardRequest" message="tns:Forward"/>
    <output wsam:Action="http://frttp.jp/2018/05/ForwardResponse" message="tns:ForwardResponse"/>
  </operation>
  <operation name="Notify">
    <input wsam:Action="http://frttp.jp/2018/05/NotifyRequest" message="tns:Notify"/>
    <output wsam:Action="http://frttp.jp/2018/05/NotifyResponse" message="tns:NotifyResponse"/>
  </operation>
</portType>

<binding name="frttpPortBinding" type="tns:frttp">
  <soap12:binding transport="http://schemas.xmlsoap.org/soap/http" style="document"/>
  <operation name="Forward">
    <soap12:operation soapAction=""/>
    <input>
      <soap12:body use="literal"/>
    </input>
    <output>
      <soap12:body use="literal"/>
    </output>
  </operation>
  <operation name="Notify">
    <soap12:operation soapAction=""/>
    <input>
      <soap12:body use="literal"/>
    </input>
    <output>
      <soap12:body use="literal"/>
    </output>
  </operation>
</binding>

<service name="FrttService">
  <port name="frttpPort" binding="tns:frttpPortBinding">
    <soap12:address location="http://localhost:8880/frttp"/>
  </port>
</service>

</definitions>

```

※ドメイン確定後、再度展開する。

最後に

本仕様書に記載のESP間連携プロトコルは、中小企業庁様の平成29年度「中小企業・小規模事業者決済情報管理支援事業」に係る実証プロジェクトにおいて考案されたものであり、当事業に参画された企業様に公開された「共同利用システム接続仕様書」のうち、今のESP間連携プロトコルに該当する部分のみを抜粋し、不要な文言を削除したものとなっております。

中小企業庁様、株式会社NTTデータ経営研究所様、当事業に参画された企業の皆様には、心より感謝申し上げます。

現在は、中小企業共通EDIの普及推進を目的に結成された「つなぐITコンソーシアム」にてこれを引き継ぎ、普及に向けた取り組みを行っております。

本仕様書に関するお問い合わせ先

つなぐITコンソーシアム会員

株式会社エクス 稲葉 秀嗣

電話 06-7711-1161

E-mail inaba@xeex.co.jp

Copyright (c) 2021 つなぐITコンソーシアム.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

Copyright (C) 2021年 つなぐITコンソーシアム.

この文書を、フリーソフトウェア財団発行のGNUフリー文書利用許諾契約書(バージョン1.2かそれ以降から一つを選択)が定める条件の下で複製、頒布、あるいは改変することを許可する。変更不可部分、表カバーテキスト、裏カバーテキストは存在しない。この利用許諾契約書の複製物は「GNUフリー文書利用許諾契約書」という章に含まれている。