

A2A
Specifiche Web Services
SARA

Contenuti

CONTENUTI	2
1 INTRODUZIONE	3
2 UPLOAD SEGMENTATO	4
2.1 RICHIESTA UPLOAD SEGMENTATO.....	4
2.1.1 <i>Input del WS</i>	4
2.1.2 <i>Output del WS</i>	5
2.2 UPLOAD SEGMENTATO.....	6
2.2.1 <i>Input del WS</i>	6
2.2.2 <i>Output del WS</i>	7
2.2.3 <i>Considerazioni</i>	8
3 APPENDICE	9
3.1 APPENDICE A – WSDL.....	9
3.1.1 <i>WSDL A2ABankITService</i>	9
3.2 APPENDICE B – GESTIONE E TIPI DI ERRORE	17

1 INTRODUZIONE

Di seguito saranno riportate le specifiche per l'utilizzo dei WS necessari al sistema software A2A. In particolare, tale documento descrive le specifiche per il servizio di UPLOAD SEGMENTATO da utilizzarsi nell'ambito della rilevazione SARA.

Tale servizio è suddiviso in due funzionalità:

- la prima (Richiesta upload segmentato) permette all'utente di richiedere l'autorizzazione ad effettuare un upload;
- la seconda invece permette di inviare i segmenti relativi all'upload da effettuare.

Entrambe le richieste devono contenere le necessarie informazioni di autenticazione.

Quando l'utente deve eseguire l'upload di un file, invia una richiesta indicando i parametri necessari all'invio. Nel caso in cui l'invio non fosse possibile, il sistema A2A restituisce un apposito messaggio di errore/avviso all'utente indicando la motivazione della mancata presa in carico dell'operazione; nel caso in cui l'invio viene autorizzato, viene restituito all'utente:

- token di transazione: generato dinamicamente e valido per un tempo pari a 5 minuti
- dimensione massima del chunk che può essere inviato
- indicazione se il file che si sta inviando è stato precedentemente inviato senza terminazione
- stream offset del file (0 se si sta inviando un file "nuovo")

Nel caso in cui lo stesso file sia stato precedentemente inviato senza però completarlo, e quindi il servizio di richiesta restituisce l'informazione di file esistente, in occasione dell'invio del primo chunk da appendere è possibile specificare se si vuole sovrascrivere il file o se si vuole continuare l'append dello stream. Nel primo caso il servizio si comporta come se il file fosse nuovo, elimina il file esistente creandone uno nuovo e imposta come stream offset un valore pari a 0; nel secondo caso appende al file esistente lo stream a partire dall'offset indicato.

Con queste informazioni, l'utente può preparare l'input per l'invio del file; a seconda della dimensione del file possono essere eseguite più invocazioni del servizio di invio file dove cambierà il dato relativo al frammento che si sta inviando e il token di autorizzazione. Se durante l'invio di un segmento si verifica almeno un errore, il WS restituisce uno o più messaggi di errore; se, invece, l'invio avviene correttamente, il WS restituisce all'utente l'offset del file dal quale prendere il prossimo segmento e un nuovo token, con le stesse proprietà indicate prima. Al termine dell'invio dell'ultimo segmento il servizio indica all'utente che il file è completato.

Entrambi i servizi verificano che l'utente oltre ad essere correttamente autenticato secondo le specifiche previste dall'infrastruttura A2A sia anche autorizzato ad operare per la survey indicata (i.e. SARA).

La modalità di autenticazione prevista avviene tramite il meccanismo dello UsernameToken (standard WS-Security), che deve contenere le credenziali di accesso utilizzate anche per l'accesso al portale.

Per richiamare le suddette funzionalità il WS (A2ABankITService) espone due metodi:

```
requestChunkedUpload(A2ARequestChunkedUploadInput input),  
chunkedUploadData(A2AChunkedUploadInput input)
```

2 UPLOAD SEGMENTATO

Di seguito sono riportate le informazioni di dettaglio delle due funzionalità utilizzate per l'upload segmentato.

2.1 Richiesta upload segmentato

L'operation del WS da utilizzare per invocare la funzionalità è `requestChunkedUpload`.

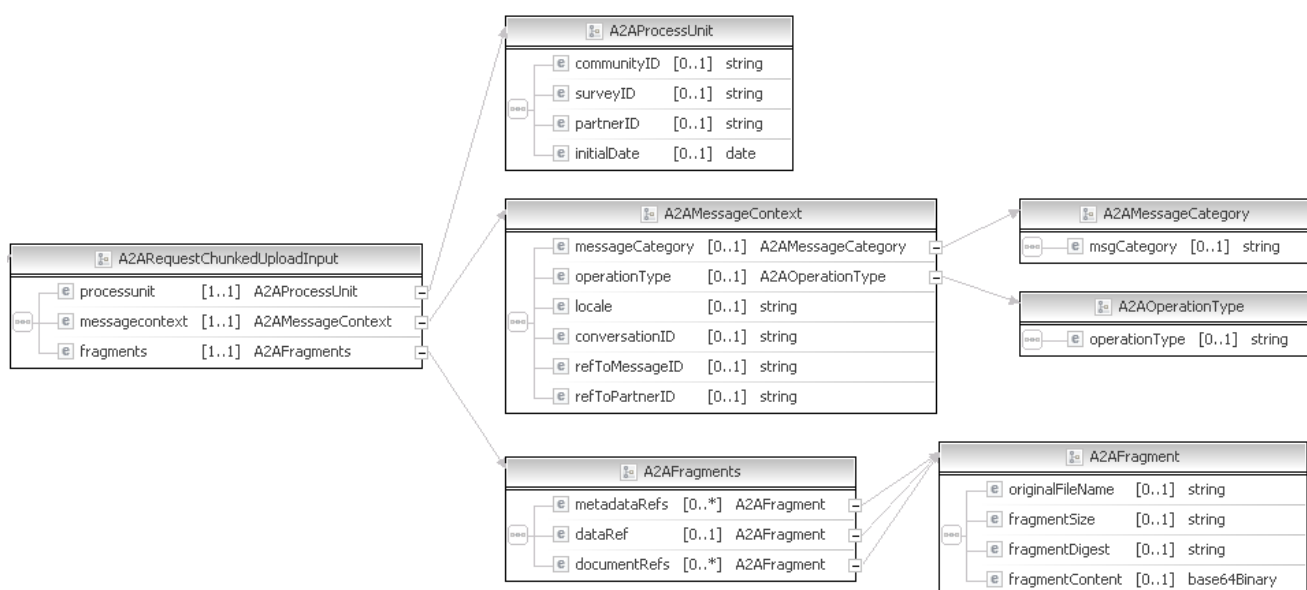
La funzionalità riceve le informazioni riguardante la *process unit*, il *message context* e il file da inviare e verifica se l'operazione può essere eseguita; a riguardo vengono effettuati i seguenti controlli:

- Un messaggio di diagnostico può essere inviato solo se non ci sono altri diagnostici in lavorazione o consegne non protocollate.
- Un messaggio di consegna può essere inviato solo se non ci sono altre consegne non protocollate.
- La dimensione massima accettata per il file da inviare è pari a 300MB. File che superano tale soglia devono essere inviati in modalità compressa.

Per ulteriori dettagli sulla composizione degli oggetti di input si rimanda ai paragrafi successivi.

2.1.1 Input del WS

La struttura dei dati `A2ARequestChunkedUploadInput` input del WS è illustrata di seguito:



PROPRIETA'	STRUTTURA DATI	OBBLIGATORIO
processUnit	A2AProcessUnit	SI
messageContext	A2AMessageContext	SI
fragments	A2AFragments	SI

Di seguito la modalità di compilazione dell'input, limitatamente ai parametri obbligatori, per l'inoltro di segnalazioni SARA:

- processUnit
 - communityID: deve essere impostato al valore **UIF**
 - surveyID: deve essere impostato al valore **SARA** o **SARV** in funzione di quale rilevazione si stia inoltrando (nuovo schema segnaletico o vecchio).
 - partnerID: deve essere impostato pari al codice del segnalante
 - initialDate: deve essere impostato pari alla data di riferimento per cui si sta effettuando l'invio

- messageContext
 - messageCategory: deve essere impostato al valore **SEND**
 - operationType: deve essere valorizzato con **PRODUCTION** se si tratta di una consegna, con **DIAGNOSTIC** nel caso di richiesta di diagnostico
 - locale: deve essere impostato al valore **DEFAULT**
 - conversationID: valorizzato con un identificativo dello scambio definito dal client

- fragments
 - dataRef
 - originalFileName: nome originale del file
 - fragmentSize: dimensione totale del file che si sta inviando

2.1.2 Output del WS

Il WS restituisce in output la struttura `A2AResultChunkedUploadOutput` illustrata di seguito:



PROPRIETA'	DESCRIZIONE	STRUTTURA DATI
<code>requestAccepted</code>	Flag richiesta accettata	boolean
<code>requestAcceptedInfo</code>	Info richiesta accettata	<code>A2AResultChunkedUploadAccepted</code>
<code>requestRefusedInfo</code>	Info richiesta rifiutata	<code>A2AResultChunkedUploadRefused</code>
<code>originalFileName</code>	Nome originale del file	Stringa di caratteri
<code>processUnit</code>	Info flusso dati	<code>A2AResultChunkedUploadProcessUnit</code>

Le informazioni di interesse relativamente all'output ricevuto sono le seguenti:

- requestAccepted: valorizzata a **true** se la richiesta è accettata, a **false** altrimenti
- requestAcceptedInfo: valorizzato quando `requestAccepted` vale true e contiene

- o **token**: chiave identificativa della transazione da utilizzare come input nella sequenza di upload successiva
 - o **chunkMaxSize**: dimensione massima ammissibile per i segmenti da inoltrare
 - o **fileExists**: flag che indica se per il file indicato è già presente un precedente upload non completato
 - o **streamOffset**: valorizzato con il valore dell'offset da cui inoltrare il segmento successivo (valorizzato a 0 nel caso in cui per il file indicato non siano presenti precedenti upload non completati)
- **requestRefusedInfo**: valorizzato se requestAccepted vale false e contiene la lista di errori (codice e descrizione) verificatisi nell'elaborazione della richiesta

2.2 Upload segmentato

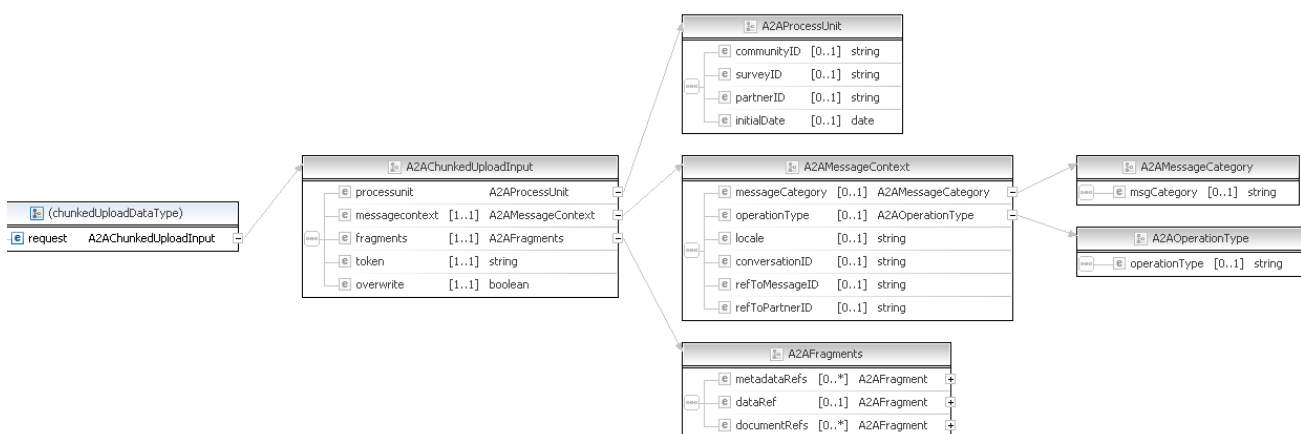
L'operation del WS da utilizzare per invocare la funzionalità è `chunkedUploadData`.

La funzionalità permette di inviare file suddivisi in segmenti la cui dimensione massima è fissata; qualora il file abbia una dimensione minore di quella di un singolo segmento, il file viene inviato tramite un'unica invocazione del servizio.

I parametri necessari per l'invocazione sono quelli utili a identificare il flusso (process unit e message context) e le informazioni riguardanti il segmento del file; nel caso di presenza di un precedente upload non completato del medesimo file, è necessario specificare inoltre se questo deve essere sovrascritto o se il segmento che si sta inviando è in *append* alla porzione di file precedentemente inoltrato.

2.2.1 Input del WS

La struttura dei dati `A2AChunkedUploadInput` input del WS è illustrata di seguito:



PROPRIETA'	DESCRIZIONE	STRUTTURA DATI	OBBLIGATORIO
processUnit	Process Unit	A2AProcessUnit	SI
messageContext	Message context	A2AMessageContext	SI
fragments	Frammenti file	A2AFragments	SI

token	Token di autorizzazione	Stringa di caratteri	SI
overwrite	Overwrite file esistente	boolean	SI

Di seguito la modalità di compilazione dell'input, limitatamente ai parametri obbligatori, per l'inoltro di segnalazioni SARA:

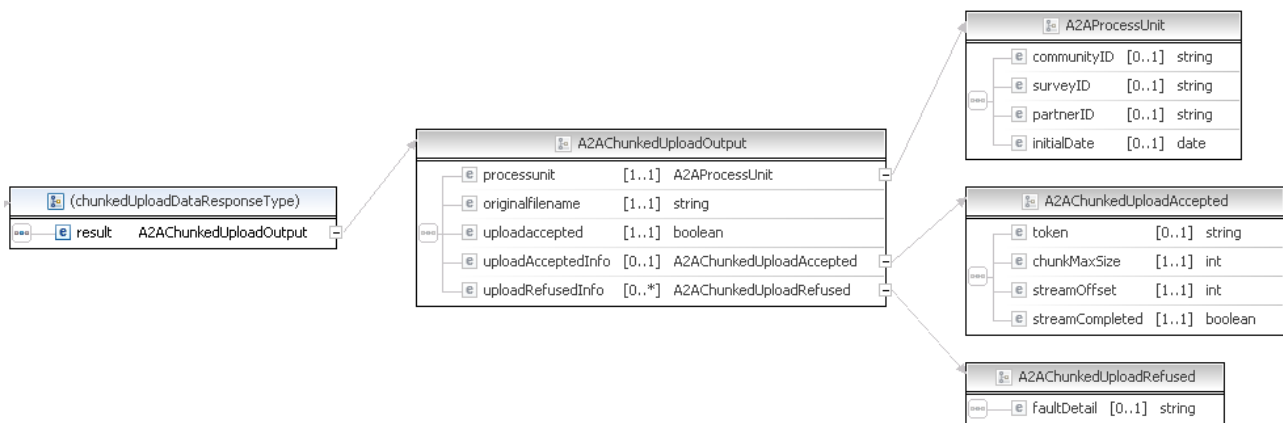
- processUnit: deve essere valorizzata analogamente all'operation di richiesta.
- messageContext: deve essere valorizzata analogamente all'operation di richiesta.
- fragments
 - o dataRef
 - originalFileName: nome originale del file
 - fragmentSize: dimensione del segmento del file che si sta inviando
 - fragmentContent: stream di byte del segmento del file
- token: deve essere valorizzato con il token ricevuto in output dalla invocazione precedente.
- overwrite: flag booleano che indica se il segmento che si sta inviando è in *append* alla porzione di file precedentemente inoltrato

Considerazioni sui parametri di input:

- il token di transazione da utilizzarsi è sempre quello ricevuto nella risposta dell'ultima operazione effettuata. Se si sta inviando il primo segmento si deve utilizzare il token ricevuto nella richiesta di upload; per i segmenti successivi bisogna invece utilizzare il token ricevuto come risposta all'invio del segmento precedente.
- L'informazione overwrite sarà presa in considerazione solo in caso di primo invio del segmento, in seguito all'invocazione del servizio di richiesta upload. Se è impostato a true il file verrà eliminato, creato uno nuovo e il segmento inviato verrà appeso al file a partire dall'offset 0; se viene impostato a false, il segmento verrà appeso a partire dall'offset stream corrente.

2.2.2 Output del WS

Il WS restituisce in output la struttura `A2AChunkedUploadOutput` illustrata di seguito:



PROPRIETA'	DESCRIZIONE	STRUTTURA DATI
uploadAccepted	Upload accettato	boolean
uploadAcceptedInfo	Info upload accettato	A2AChunkedUploadAccepted
uploadRefusedInfo	Info upload rifiutato	A2AChunkedUploadRefused
originalFileName	Nome del file	Stringa di caratteri
processUnit	Process unit	A2AProcessUnit

Le informazioni di interesse relativamente all'output ricevuto sono le seguenti:

- uploadAccepted: valorizzata a **true** se l'upload del semgmento è accettato, a **false** altrimenti
- uploadAcceptedInfo: valorizzato quando uploadAccepted vale true e contiene
 - o token: chiave identificativa della transazione da utilizzare come input nella sequenza di upload successiva
 - o chunkMaxSize: dimensione massima ammissibile per i segmenti da inoltrare
 - o streamOffset: valorizzato con il valore dell'offset da cui inoltrare il segmento successivo
 - o streamCompleted: flag boolean che indica se il file è stato completamente inviato
- requestRefusedInfo: valorizzato se uploadAccepted vale false e contiene la lista di errori (codice e descrizione) verificatisi nell'elaborazione della richiesta

2.2.3 Considerazioni

In caso di interruzione dell'upload, è possibile effettuare il recovery dello stesso: il funzionamento corretto prevede che l'utente richiami inizialmente la funzionalità requestChunkedUpload in modo da ottenere le informazioni corrette per continuare l'upload precedente e, successivamente, inviare il segmento corretto.

3 APPENDICE

Il Web Service è raggiungibile nei due ambienti previsti: Produzione (ambiente ufficiale) e Collaudo (ambiente di collaudo con i segnalanti).

L'endpoint per i due ambienti è il seguente:

per l'ambiente di Produzione

<https://a2a.bancaditalia.it/infostat/CollectionServices>

Per l'ambiente di Collaudo

<https://certa2a.bancaditalia.it/infostat/CollectionServices>

3.1 APPENDICE A – WSDL

Di seguito viene riportato il file WSDL generato alla creazione del Web Service.

Lo stesso è raggiungibile utilizzando gli indirizzi:

- ambiente di produzione: <https://a2a.bancaditalia.it/infostat/CollectionServices?wsdl>
- ambiente di collaudo: <https://certa2a.bancaditalia.it/infostat/CollectionServices?wsdl>

Il WSDL, corredato da tutti i file XSD necessari alla sua completa definizione, è comunque disponibile come archivio compresso sul sito Banca d'Italia nella sezione Unità di informazione finanziaria > Comunicazioni al pubblico > Revisione del sistema di invio delle segnalazioni antiriciclaggio aggregate.

3.1.1 WSDL A2ABankITService

```
<?xml version="1.0" encoding="UTF-8"?><wsdl:definitions name="A2ABankIT"
targetNamespace="http://nrd.a2a.bankitalia.it"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://nrd.a2a.bankitalia.it"
xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <wsdl:types>
    <xsd:schema targetNamespace="http://nrd.a2a.bankitalia.it">
      <xsd:include
schemaLocation="A2ARequestChunkedUploadOutput.xsd"/>
      <xsd:include
schemaLocation="A2ARequestChunkedUploadInput.xsd"/>
      <xsd:include schemaLocation="A2AChunkedUploadOutput.xsd"/>
      <xsd:include schemaLocation="A2AChunkedUploadInput.xsd"/>
      <xsd:include schemaLocation="A2AAgreementCalendarRes.xsd"/>
      <xsd:include schemaLocation="A2AAgreementCalendarReq.xsd"/>
      <xsd:include schemaLocation="A2ASurveyAndPartnerInfoRes.xsd"/>
      <xsd:include schemaLocation="A2ASurveyAndPartnerInfoReq.xsd"/>
      <xsd:include schemaLocation="A2AFault.xsd"/>
      <xsd:include schemaLocation="A2AFragment2DownloadRes.xsd"/>
      <xsd:include schemaLocation="A2AFragment2DownloadReq.xsd"/>
      <xsd:include schemaLocation="A2AMessageCriteriaInput.xsd"/>
      <xsd:include schemaLocation="A2AMessageResultOutput.xsd"/>
      <xsd:include schemaLocation="A2AResult.xsd"/>
      <xsd:include schemaLocation="A2AFragments.xsd"/>
      <xsd:include schemaLocation="A2AMessageContext.xsd"/>
    </xsd:schema>
  </wsdl:types>
</wsdl:definitions>
```

```

        <xsd:include schemaLocation="A2AProcessUnit.xsd"/>
        <xsd:element name="uploadData">
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:element name="processUnit"
nillable="true" type="tns:A2AProcessUnit"/>
                    <xsd:element name="messageContext"
nillable="true" type="tns:A2AMessageContext"/>
                    <xsd:element name="fragments"
nillable="true" type="tns:A2AFragments"/>
                </xsd:sequence>
            </xsd:complexType>
        </xsd:element>
        <xsd:element name="uploadDataResponse">
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:element name="result" nillable="true"
type="tns:A2AResult"/>
                </xsd:sequence>
            </xsd:complexType>
        </xsd:element>
        <xsd:element name="getMessages">
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:element name="inputCriteria"
nillable="true" type="tns:A2AMessageCriteriaInput"/>
                </xsd:sequence>
            </xsd:complexType>
        </xsd:element>
        <xsd:element name="getMessagesResponse">
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:element name="result" nillable="true"
type="tns:A2AMessageResultOutput"/>
                </xsd:sequence>
            </xsd:complexType>
        </xsd:element>
        <xsd:element name="downloadFragment">
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:element name="processUnit"
nillable="true" type="tns:A2AProcessUnit"/>
                    <xsd:element name="fragmentReq"
nillable="true" type="tns:A2AFragment2DownloadReq"/>
                    <xsd:element name="payloadType"
nillable="true" type="xsd:string"/>
                </xsd:sequence>
            </xsd:complexType>
        </xsd:element>
        <xsd:element name="downloadFragmentResponse">
            <xsd:complexType>
                <xsd:sequence>
                    <xsd:element name="fragmentRes"
nillable="true" type="tns:A2AFragment2DownloadRes"/>
                </xsd:sequence>
            </xsd:complexType>
        </xsd:element>
        <xsd:element name="getSurveyAndPartnerInfo">
            <xsd:complexType>
                <xsd:sequence>

```

```

                                <xsd:element name="request" nillable="true"
type="tns:A2ASurveyAndPartnerInfoReq"/>
                                </xsd:sequence>
                                </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="getSurveyAndPartnerInfoResponse">
                                <xsd:complexType>
                                <xsd:sequence>
                                <xsd:element name="response" nillable="true"
type="tns:A2ASurveyAndPartnerInfoRes"/>
                                </xsd:sequence>
                                </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="A2AFault_element" nillable="true"
type="tns:A2AFault"/>
                                <xsd:element name="A2AFault_element2" nillable="true"
type="tns:A2AFault"/>
                                <xsd:element name="getAgreementCalendar">
                                <xsd:complexType>
                                <xsd:sequence>
                                <xsd:element name="request" nillable="true"
type="tns:A2AAgreementCalendarReq"/>
                                </xsd:sequence>
                                </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="getAgreementCalendarResponse">
                                <xsd:complexType>
                                <xsd:sequence>
                                <xsd:element name="response" nillable="true"
type="tns:A2AAgreementCalendarRes"/>
                                </xsd:sequence>
                                </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="chunkedUploadData">
                                <xsd:complexType>
                                <xsd:sequence>
                                <xsd:element name="request"
type="tns:A2AChunkedUploadInput"/>
                                </xsd:sequence>
                                </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="chunkedUploadDataResponse">
                                <xsd:complexType>
                                <xsd:sequence>
                                <xsd:element name="result"
type="tns:A2AChunkedUploadOutput"/>
                                </xsd:sequence>
                                </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="requestChunkedUpload">
                                <xsd:complexType>
                                <xsd:sequence>
                                <xsd:element name="request"
type="tns:A2ARequestChunkedUploadInput"/>
                                </xsd:sequence>
                                </xsd:complexType>
                                </xsd:element>
                                <xsd:element name="requestChunkedUploadResponse">

```

```

        <xsd:complexType>
            <xsd:sequence>

                <xsd:element name="result"
type="tns:A2ARequestChunkedUploadOutput"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
    <xsd:element name="hello">
        <xsd:complexType>
            <xsd:sequence>

                <xsd:element name="in" type="xsd:string"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
    <xsd:element name="helloResponse">
        <xsd:complexType>
            <xsd:sequence>

                <xsd:element name="out" type="xsd:string"/>
            </xsd:sequence>
        </xsd:complexType>
    </xsd:element>
    <xsd:element name="hello_fault">
        <xsd:complexType>
            <xsd:sequence>

                <xsd:element name="hello_fault"
type="xsd:string">
            </xsd:element>
        </xsd:sequence>
    </xsd:complexType>
</xsd:element>
</xsd:schema>
</wsdl:types>
<wsdl:message name="uploadDataRequestMsg">
    <wsdl:part element="tns:uploadData" name="uploadDataParameters"/>
</wsdl:message>
<wsdl:message name="uploadDataResponseMsg">
    <wsdl:part element="tns:uploadDataResponse"
name="uploadDataResult"/>
</wsdl:message>
<wsdl:message name="getMessagesRequestMsg">
    <wsdl:part element="tns:getMessages" name="getMessagesParameters"/>
</wsdl:message>
<wsdl:message name="getMessagesResponseMsg">
    <wsdl:part element="tns:getMessagesResponse"
name="getMessagesResult"/>
</wsdl:message>
<wsdl:message name="downloadFragmentRequestMsg">
    <wsdl:part element="tns:downloadFragment"
name="downloadFragmentParameters"/>
</wsdl:message>
<wsdl:message name="downloadFragmentResponseMsg">
    <wsdl:part element="tns:downloadFragmentResponse"
name="downloadFragmentResult"/>
</wsdl:message>
<wsdl:message name="uploadData_faultMsg">
    <wsdl:part element="tns:A2AFault_element2" name="fault"/>
</wsdl:message>

```

```

<wsdl:message name="getMessages_faultMsg">
  <wsdl:part element="tns:A2AFault_element" name="fault"/>
</wsdl:message>
<wsdl:message name="downloadFragment_faultMsg">
  <wsdl:part element="tns:A2AFault_element" name="fault"/>
</wsdl:message>
<wsdl:message name="getSurveyAndPartnerInfoRequestMsg">
  <wsdl:part element="tns:getSurveyAndPartnerInfo"
name="getSurveyAndPartnerInfoParameters"/>
</wsdl:message>
<wsdl:message name="getSurveyAndPartnerInfoResponseMsg">
  <wsdl:part element="tns:getSurveyAndPartnerInfoResponse"
name="getSurveyAndPartnerInfoResult"/>
</wsdl:message>
<wsdl:message name="getSurveyAndPartnerInfo_faultMsg">
  <wsdl:part element="tns:A2AFault_element" name="fault"/>
</wsdl:message>
<wsdl:message name="getAgreementCalendarRequestMsg">
  <wsdl:part element="tns:getAgreementCalendar"
name="getAgreementCalendarParameters"/>
</wsdl:message>
<wsdl:message name="getAgreementCalendarResponseMsg">
  <wsdl:part element="tns:getAgreementCalendarResponse"
name="getAgreementCalendarResult"/>
</wsdl:message>
<wsdl:message name="getAgreementCalendar_faultMsg">
  <wsdl:part element="tns:A2AFault_element" name="fault"/>
</wsdl:message>
<wsdl:message name="chunkedUploadDataRequestMsg">
  <wsdl:part element="tns:chunkedUploadData"
name="chunkedUploadInput"/>
</wsdl:message>
<wsdl:message name="chunkedUploadDataResponseMsg">
  <wsdl:part element="tns:chunkedUploadDataResponse"
name="chunkedUploadOutput">
  </wsdl:part>
</wsdl:message>
<wsdl:message name="chunkedUploadData_faultMsg">
  <wsdl:part element="tns:A2AFault_element2" name="fault"/>
</wsdl:message>
<wsdl:message name="requestChunkedUploadRequestMsg">
  <wsdl:part element="tns:requestChunkedUpload"
name="requestChunkedUploadInput">
  </wsdl:part>
</wsdl:message>
<wsdl:message name="requestChunkedUploadResponseMsg">
  <wsdl:part element="tns:requestChunkedUploadResponse"
name="requestChunkedUploadOutput"/>
</wsdl:message>
<wsdl:message name="requestChunkedUpload_faultMsg">
  <wsdl:part element="tns:A2AFault_element2" name="fault"/>
</wsdl:message>
<wsdl:message name="helloRequest">
  <wsdl:part element="tns:hello" name="request"/>
</wsdl:message>
<wsdl:message name="helloResponse">
  <wsdl:part element="tns:helloResponse" name="response"/>
</wsdl:message>
<wsdl:message name="hello_faultMsg">
  <wsdl:part element="tns:A2AFault_element" name="fault"/>
</wsdl:message>

```

```

    <wsdl:portType name="A2ABankIT">
      <wsdl:operation name="uploadData">
        <wsdl:input message="tns:uploadDataRequestMsg"
name="uploadDataRequest"/>
        <wsdl:output message="tns:uploadDataResponseMsg"
name="uploadDataResponse"/>
        <wsdl:fault message="tns:uploadData_faultMsg" name="fault"/>
      </wsdl:operation>
      <wsdl:operation name="getMessages">
        <wsdl:input message="tns:getMessagesRequestMsg"
name="getMessagesRequest"/>
        <wsdl:output message="tns:getMessagesResponseMsg"
name="getMessagesResponse"/>
        <wsdl:fault message="tns:getMessages_faultMsg" name="fault"/>
      </wsdl:operation>
      <wsdl:operation name="downloadFragment">
        <wsdl:input message="tns:downloadFragmentRequestMsg"
name="downloadFragmentRequest"/>
        <wsdl:output message="tns:downloadFragmentResponseMsg"
name="downloadFragmentResponse"/>
        <wsdl:fault message="tns:downloadFragment_faultMsg"
name="fault"/>
      </wsdl:operation>
      <wsdl:operation name="getSurveyAndPartnerInfo">
        <wsdl:input message="tns:getSurveyAndPartnerInfoRequestMsg"
name="getSurveyAndPartnerInfoRequest"/>
        <wsdl:output message="tns:getSurveyAndPartnerInfoResponseMsg"
name="getSurveyAndPartnerInfoResponse"/>
        <wsdl:fault message="tns:getSurveyAndPartnerInfo_faultMsg"
name="fault"/>
      </wsdl:operation>
      <wsdl:operation name="getAgreementCalendar">
        <wsdl:input message="tns:getAgreementCalendarRequestMsg"
name="getAgreementCalendarRequest"/>
        <wsdl:output message="tns:getAgreementCalendarResponseMsg"
name="getAgreementCalendarResponse"/>
        <wsdl:fault message="tns:getAgreementCalendar_faultMsg"
name="fault"/>
      </wsdl:operation>
      <wsdl:operation name="chunkedUploadData">
        <wsdl:input message="tns:chunkedUploadDataRequestMsg"
name="chunkedUploadDataRequest"/>
        <wsdl:output message="tns:chunkedUploadDataResponseMsg"
name="chunkedUploadDataResponse"/>
        <wsdl:fault message="tns:chunkedUploadData_faultMsg"
name="fault"/>
      </wsdl:operation>
      <wsdl:operation name="requestChunkedUpload">
        <wsdl:input message="tns:requestChunkedUploadRequestMsg"
name="requestChunkedUploadRequest"/>
        <wsdl:output message="tns:requestChunkedUploadResponseMsg"
name="requestChunkedUploadResponse"/>
        <wsdl:fault message="tns:requestChunkedUpload_faultMsg"
name="fault"/>
      </wsdl:operation>
    </wsdl:portType>
    <wsdl:binding name="A2ABankITBinding" type="tns:A2ABankIT">
      <soap:binding style="document"
transport="http://schemas.xmlsoap.org/soap/http"/>
      <wsdl:operation name="uploadData">

```

```

        <soap:operation
soapAction="http://nrd.a2a.bankitalia.it/uploadData"/>
        <wsdl:input name="uploadDataRequest">
            <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output name="uploadDataResponse">
            <soap:body use="literal"/>
        </wsdl:output>
        <wsdl:fault name="fault">
            <soap:fault name="fault" use="literal"/>
        </wsdl:fault>
    </wsdl:operation>
    <wsdl:operation name="getMessages">
        <soap:operation
soapAction="http://nrd.a2a.bankitalia.it/getMessages"/>
        <wsdl:input name="getMessagesRequest">
            <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output name="getMessagesResponse">
            <soap:body use="literal"/>
        </wsdl:output>
        <wsdl:fault name="fault">
            <soap:fault name="fault" use="literal"/>
        </wsdl:fault>
    </wsdl:operation>
    <wsdl:operation name="downloadFragment">
        <soap:operation
soapAction="http://nrd.a2a.bankitalia.it/downloadFragment"/>
        <wsdl:input name="downloadFragmentRequest">
            <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output name="downloadFragmentResponse">
            <soap:body use="literal"/>
        </wsdl:output>
        <wsdl:fault name="fault">
            <soap:fault name="fault" use="literal"/>
        </wsdl:fault>
    </wsdl:operation>
    <wsdl:operation name="getSurveyAndPartnerInfo">
        <soap:operation
soapAction="http://nrd.a2a.bankitalia.it/getSurveyAndPartnerInfo"/>
        <wsdl:input name="getSurveyAndPartnerInfoRequest">
            <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output name="getSurveyAndPartnerInfoResponse">
            <soap:body use="literal"/>
        </wsdl:output>
        <wsdl:fault name="fault">
            <soap:fault name="fault" use="literal"/>
        </wsdl:fault>
    </wsdl:operation>
    <wsdl:operation name="getAgreementCalendar">
        <soap:operation
soapAction="http://nrd.a2a.bankitalia.it/getAgreementCalendar"/>
        <wsdl:input name="getAgreementCalendarRequest">
            <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output name="getAgreementCalendarResponse">
            <soap:body use="literal"/>
        </wsdl:output>
        <wsdl:fault name="fault">

```

```

        <soap:fault name="fault" use="literal"/>
    </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="chunkedUploadData">
    <soap:operation
soapAction="http://nrd.a2a.bankitalia.it/chunkedUploadData"/>
    <wsdl:input name="chunkedUploadDataRequest">
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output name="chunkedUploadDataResponse">
        <soap:body use="literal"/>
    </wsdl:output>
    <wsdl:fault name="fault">
        <soap:fault name="fault" use="literal"/>
    </wsdl:fault>
</wsdl:operation>
<wsdl:operation name="requestChunkedUpload">
    <soap:operation
soapAction="http://nrd.a2a.bankitalia.it/requestChunkedUpload"/>
    <wsdl:input name="requestChunkedUploadRequest">
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output name="requestChunkedUploadResponse">
        <soap:body use="literal"/>
    </wsdl:output>
    <wsdl:fault name="fault">
        <soap:fault name="fault" use="literal"/>
    </wsdl:fault>
</wsdl:operation>
</wsdl:binding>
<wsdl:service name="A2ABankITService">
    <wsdl:port binding="tns:A2ABankITBinding" name="A2ABankITPort">
        <soap:address
location="http://localhost:9081/InfostatA2A/A2ABankITService"/>
    </wsdl:port>
</wsdl:service>
</wsdl:definitions>
</wsdl:definitions>

```


3.2 APPENDICE B – GESTIONE E TIPI DI ERRORE

Nei casi di errore il messaggio di fault generato dal servizio utilizza un oggetto di tipo `A2ARequestChunkedUploadRefused`. Tale struttura è composta dalle proprietà `codice`, `descrizione` e `dettaglio`.

Di seguito è riportata la tabella riguardante gli errori previsti dalle funzionalità dei WS:

CODICE	DESCRIZIONE	NOTE
-1	Errore generico	
0	E' in elaborazione una richiesta di tipo #0. Attendere il termine dell'elaborazione prima di inviare una nuova richiesta.	#0 può assumere il valore CONSEGNA o CONSEGNA/DIAGNOSTICO
1	Tipo file #0 non ammesso per la survey #1.	#0 e #1 sono valorizzati con i valori presi dal file di properties relativo alle survey
2	Il file ha una dimensione superiore a quella ammessa per la survey #0 pari a #1 bytes.	#0 e #1 sono valorizzati con i valori presi dal file di properties relativo alle survey
3	Token non valido	
4	Token mancante	
5	Token scaduto	
6	Parametro obbligatorio non presente	E' aggiunto tra i dettagli il parametro mancante: TOKEN, PROCESS_UNIT, COMMUNITY_ID, PARTNER_ID, SURVEY_ID, INITIAL_DATE, MESSAGE_CONTEXT, OPERATION_TYPE, MESSAGE_CATEGORY, FRAGMENTS, FRAGMENTS_DATAREF, FRAGMENTS_DATAREF_ORIGINALFILENAME, FRAGMENTS_DATAREF_FILESIZE, FRAGMENTS_DATAREF_DATAHANDLER
7	Utente non autorizzato	
8	Dimensione del segmento superiore a quella ammessa	La dimensione ammessa del segmento viene restituita in fase di richiesta dell'upload
9	Dimensione del segmento errata	
10	Data contabile non valida	
11	Dimensione del file superata	Verifica se lo stream che si vuole appendere al file supera la dimensione totale prevista del file
12	Dimensione del file errata	
13	Operation type non ammessa (PRODUCTION/DIAGNOSTIC)	Sono ammessi solo i valori PRODUCTION e DIAGNOSTIC
14	Agreement non trovato. Verificare i dati forniti	
15	Aggiornamento informazioni upload non eseguito. Verificare i dati forniti o contattare l'assistenza	
16	Creazione informazioni upload non eseguito. Verificare i dati forniti o contattare l'assistenza	
17	Il messaggio non e' stato consegnato. Contattare l'assistenza	