L'AGENZIA DELLE ACCISE, DOGANE E MONOPOLI

PROGETTO WEB SERVICES DOGANE

"SERVIZIO ACQUISIZIONE INFORMAZIONI INTEROPERABILITÀ"

DOGANE EMANIFEST E GESTIONE TEMPORANEA CUSTODIA

Ambiente di Addestramento

Manuale operativo - versione del 18 Novembre 2022

Sommario

1. GENERALITÀ

1.1. Canali di comunicazione dei sistemi

I Web Services sono esposti da SOGEI utilizzando gli standard più diffusi (SOAP, REST) e sono fruibili attraverso canali di comunicazione sicuri data la sensibilità dei dati scambiati.

La cooperazione tra ente interessato e SOGEI avviene attraverso un canale https bilanciato (certificato client e server).

L'autenticazione necessita di un certificato (cosiddetto Certificato di autenticazione – CA Dogane) rilasciato agli utenti registrati che ne facciano opportuna richiesta. I meccanismi di autenticazione e autorizzazione sono descritti in dettaglio nel paragrafo "Modalità di accreditamento".

I messaggi XML, ove previsto, vengono firmati dal client e trasmessi sfruttando il messaggio SOAP. Gli XML di cui sopra devono essere creati seguendo schemi XSD, rispettandone il contenuto e tutti i vincoli di obbligatorietà e molteplicità. Pertanto durante la fase di ricezione del messaggio, oltre alla verifica della firma che serve a preservarne l'integrità, viene fatta una validazione rispetto allo schema XSD, tesa a controllare formalmente il contenuto del messaggio.

2. SOA DOMINIO ESTERNO

2.1. Descrizione dell'oggetto di interscambio

Rispetto alla specificità del servizio erogato è rilasciato uno specifico tracciato dati XSD che contiene tutti i campi necessari alla sottomissione delle richieste di elaborazione ed alla gestione dei messaggi di ritorno. I campi utilizzati da un servizio web possono essere di input o di output. I campi di input obbligatori per ciascun servizio sono riprodotti nei documenti allegati nella sezione riguardante il servizio. I restanti campi, cioè quelli del DTO esclusi quelli di input, sono campi di output e in generale, ma non sempre, sono riempiti alla risposta dal servizio web invocato.

2.2. Modalità di accreditamento

Per usufruire dei servizi offerti, l'operatore economico interessato deve:

- dotarsi di credenziali SPID (Sistema Pubblico di Identità Digitale) strettamente di livello 2 e di Persona Fisica (anche ad uso professionale) o di una CNS (Carta Nazionale dei Servizi) di Persona Fisica o di una CIE (Carta d'Identità Elettronica) per l'accesso al portale istituzionale PUDM (Portale Unico delle Dogane e dei Monopoli) dell'Agenzia.
- richiedere, tramite la funzione "Mio Profilo" del MAU (Modello Autorizzativo Unico),

l'autorizzazione connessa all'attività svolta.

Le credenziali SPID di livello 2 permettono l'accesso ai servizi con nome utente e password insieme ad un codice temporaneo che viene inviato all'utente mediante sms o con app mobile.
Per ulteriori informazioni sull'ottenimento delle credenziali SPID e CNS si rimanda ai rispettivi fornitori del servizio di Identity Management.
Per ottenere l'**autorizzazione** all'utilizzo dei servizi offerti da ADM, l'operatore economico deve effettuare l'accesso all'**Area riservata** del PUDM (www.adm.gov.it), selezionando la tab SPID o CNS sulla pagina di login proposta. A valle della fase di autenticazione, dovrà quindi accedere alla funzione "**Mio Profilo**", disponibile tra i **Servizi online.**
In tale fase, l'operatore economico dovrà individuare il "**Gestore**", Persona Fisica a cui il soggetto giuridico - che ha titolo ad utilizzare i servizi digitali - conferisce delega per l'attribuzione e la gestione delle autorizzazioni. Il "Gestore", ricevuta la delega, attribuisce le autorizzazioni ai vari servizi secondo le necessità operative dell'operatore economico.

L'individuazione del Gestore non è necessaria nel caso in cui l'operatore economico sia una ditta individuale e le autorizzazioni siano gestite direttamente dal titolare.

Per le **trasmissioni dei Manifesti delle Merci in Arrivo (MMA)** e **in Partenza (MMP)** è stato definito sul MAU un apposito profilo con le seguenti caratteristiche:

Nome applicazione/servizio	Presentazione Manifesti Merci
Codice	dlr_emanifest
Categoria	Dogane
Descrizione autorizzazione	Consente l'invio dei Manifesti delle merci e di consultare i relativi esiti elaborativi
Necessita approvazione	NO
Tipologia di utenza	Persone fisiche e persone giuridiche
Livello di Autenticazione	Consistente(SPID/CNS)
Tipologia di interazione	System To System, User To System

Tabella 1 - Caratteristiche del profilo Emanifest definito sul MAU

Per la firma **dei Manifesti delle Merci in Arrivo (MMA)** e **in Partenza (MMP)** è stato definito sul MAU un apposito profilo con le seguenti caratteristiche:

Nome applicazione/servizio	Firma Manifesti Merci
Codice	dlr_emanifest_firma

Categoria	Dogane
Descrizione autorizzazione	Consente di firmare digitalmente i Manifesti delle merci
Necessita approvazione	SI
Tipologia di utenza	Persone fisiche e persone giuridiche
Livello di Autenticazione	Consistente(SPID/CNS)
Tipologia di interazione	System To System, User To System

Tabella 1 - Caratteristiche del profilo Emanifest Firma definito sul MAU

Il dichiarante dovrà delegare il profilo **dlr_ emanifest _firma** ad una persona fisica, che sarà così abilitata alla firma digitale dei dati predisposti in formato xml.

Per la gestione di Partite di Temporanea Custodia (TC) è stato definito sul MAU un apposito profilo con le seguenti caratteristiche:

Nome applicazione/servizio	Gestori TC Servizio Presentazione Partite Di Temporanea Custodia
Codice	dlr_gestoritc
Categoria	Dogane
Descrizione autorizzazione	Consente l'invio delle Partite di Temporanea Custodia (Gestori TC) e di consultare i relativi esiti elaborativi
Necessita approvazione	NO
Tipologia di utenza	Persone fisiche e persone giuridiche
Livello di Autenticazione	Consistente(SPID/CNS)
Tipologia di interazione	System To System, User To System

Tabella 2 - Caratteristiche del profilo Gestione Temporanea Custodia definito sul MAU

Per la firma di Partite di Temporanea Custodia (TC) è stato definito sul MAU un apposito

profilo con le seguenti caratteristiche:

Nome applicazione/servizio	Firma Gestori TC
Codice	dlr_gestoritc_firma
Categoria	Dogane
Descrizione autorizzazione	Consente di firmare digitalmente i messaggi dei gestori di temporanea custodia
Necessita approvazione	SI
Tipologia di utenza	Persone fisiche e persone giuridiche
Livello di Autenticazione	Consistente(SPID/CNS)
Tipologia di interazione	System To System, User To System

Tabella 2 - Caratteristiche del profilo di firma per Gestione Temporanea Custodia definito sul MAU

Il dichiarante dovrà delegare il profilo **dlr_ gestoritc _firma** ad una persona fisica, che sarà così abilitata alla firma digitale dei dati predisposti in formato xml.

Le istruzioni di dettaglio sono disponibili, come di consueto, nell'assistenza on line alla voce, "Come fare per" → "Utilizzare le Altre applicazioni doganali" → "Mio profilo", dove è possibile reperire ulteriori informazioni riguardanti la figura del "Gestore" e le funzionalità disponibili (attribuzioni di autorizzazioni, deleghe, revoche) nonché alla voce Altri servizi"
→ "Nuovo modello autorizzativo: Gestione autorizzazioni".

È di riferimento, per le modalità di accesso sopra rappresentate, la nota prot. n. 104198/RU del 14 settembre 2017 - "Nuovo Modello Autorizzativo e modalità per l'accesso ai servizi digitali disponibili sul Portale Nazionale", e seguenti, cui si rimanda per completezza.

Gli operatori economici, oltre che dotarsi delle credenziali SPID o CNS, dovranno richiedere l'autorizzazione al servizio "Gestione certificati" (od eventualmente delegarlo), che permette di accedere alla linea di lavoro Area Riservata > Servizi online > Interattivi > Gestione Certificati, ove sono presenti le istruzioni per generare:

- il Certificato di autenticazione di addestramento, da utilizzare se l'utente dovrà operare in ambiente di addestramento;

- il Certificato di autenticazione di produzione, da utilizzare se l'utente dovrà operare in ambiente reale.

Dal momento in cui l'operatore è già in possesso di un certificato di autenticazione precedentemente rilasciato, è possibile usufruire dei servizi per cui è stato abilitato.

Nell'ambito della sicurezza e delle modalità di accreditamento descritte, l'accesso ai servizi cooperativi si articola in due fasi ben distinte, **autenticazione** ed **autorizzazione** così come già avviene per l'accesso ai servizi web on-line; in particolare:

1. autenticazione utente: l'accesso ai Web Services è consentito ai soli utenti in possesso di uno specifico "**Certificato di Autenticazione**" rilasciato dall'Agenzia delle Dogane;

2. autorizzazione utente: l'utilizzo dello specifico servizio è sottoposto al preventivo controllo di **autorizzazione** del singolo utente richiedente.

La fase di autenticazione utente inizia con il riconoscimento del Certificato. Superata l'autenticazione il certificato viene sottoposto al controllo tramite l'invocazione di appositi servizi che ne verificano il titolare ed il firmatario. A questo punto scatta la fase di autorizzazione utente. Tramite il controllo delle autorizzazioni è possibile stabilire se l'utenza è abilitata ad effettuare l'operazione richiesta.

2.3. Modalità di Firma dei Messaggi XML

Per la modalità di firma digitale dei messaggi XML - il DPCM 22 febbraio 2013, articolo 63 comma 3 - Codifica firma XAdES descrive le caratteristiche delle applicazioni di generazione della firma XML. I certificati di firma sono rilasciati dai certificatori accreditati secondo quanto definito nella Deliberazione CNIPA n. 45 del 21 maggio 2009. La deliberazione prescrive (art. 21, comma 16) che "Ai sensi del comma 8, sono altresì riconosciuti il formato di busta crittografica e di firma descritti nei documenti ETSI TS 101 903 – XAdES (versione 1.4.1) e ETSI TS 102 904 (versione 1.1.1)". L'art. 9 della Deliberazione prescrive che "L'elemento KeyInfo, opzionale nella specifica RFC 3275, deve essere sempre presente nella busta crittografica". La specifica ETSI TS 101 903 prescrive che possa essere usato l'elemento KeyInfo ovvero il SigningCertificate.

Visto quanto disposto al sopra citato art. 21 della deliberazione, considerata l'esigenza di salvaguardare la validità delle firme XML generate con strumenti forniti da certificatori accreditati in altri Stati membri dell'Unione, si chiarisce che, fermo restando il rispetto della citata specifica ETSI, l'assenza dell'elemento KeyInfo non ha come conseguenza l'invalidità della firma XAdES.

Delle tre tipologie di firma XML citate nella deliberazione è necessario che il client di firma generi firme digitali di tipo XAdES-BES enveloped.

Il messaggio XML trasferito come byte[] deve essere firmato con XML Digital Signature e deve inoltre soddisfare i seguenti requisiti tecnici:

- La firma XML è di tipo Enveloped dove l'elemento caratterizzante la firma digitale **ds:Signature** sarà posto come ultimo elemento della radice della struttura XML. Tale documento viene firmato digitalmente tramite l'utilizzo di chiavi e relativo certificato di firma a disposizione dell'operatore;

- uso obbligatorio dell'attributo **Id** per i tag **<ds:Signature>** e **<ds:SignatureValue>**.

Per il certificato di firma digitale occorre avvalersi di un Prestatore di servizi fiduciari indicato da lista AGID ed europea, presente ai seguenti link:

https://www.agid.gov.it/it/piattaforme/firma-elettronica-qualificata/prestatoriservizi-fiduciari-qualificati

http://tlbrowser.tsl.website/tools/index.jsp

I certificati di firma rilasciati dai Prestatori di servizi fiduciari qualificati devono essere FEQ eIDAS.

Nella sola fase di sperimentazione da effettuare nell'ambiente di addestramento sarà possibile utilizzare un Certificato di Firma, denominato "Certificato di Firma UNICO ADM", generato dalla CA "NON qualificata" dell'Agenzia Dogane e Monopoli, che potrà essere scaricato dall'applicazione Gestione certificati, disponibile nell'Area riservata del Portale ADM.
Nell'ambiente di addestramento sarà sempre possibile utilizzare i certificati di firma FEQ eidas. Si fa altresì presente che successivamente ad una prima fase di test, l'utente dovrà assolutamente procedere con un test mediante certificato di firma FEQ eidas, ai fini della conformità di integrazione rispetto a quanto offerto in ambiente di esercizio, dove saranno accettati solo Certificati di firma FEQ eidas.

2.4 Creazione Profilo di firma

Il dichiarante dovrà delegare un profilo di firma ad una persona fisica e questa sarà abilitata a firmare le dichiarazioni.

3. IL SERVIZIO PER I MANIFESTI DELLE MERCI

3.1. Tipologia di dichiarazioni accolte

Il sistema di accoglienza consente all'Operatore Economico di trasmettere i dati relativi ai **Manifesti delle Merci in arrivo (MMA)** e ai **Manifesti delle Merci in Partenza (MMP)**, mettendo a disposizione un insieme di funzionalità esposte tramite il Web Service "**EManifestService",** descritto nei paragrafi successivi.

L'operatore economico deve inviare al sistema le dichiarazioni MMA/MMP in formato XML, adottando il nuovo tracciato dati relativo all'**eManifest,** composto da:

- un datagroup relativo ai dati di testata (dati generali);
- N datagroup relativi ai dati di spedizione;
- Per ogni spedizione, da M datagroup relativi ai cargo_item.

La fase di acquisizione/convalida di un flusso eManifest può avvenire secondo i seguenti scenari:

- flusso XML contenente la sola sezione dei dati generali: AIDA genera il protocollo del manifesto;

- flusso XML contenente la sezione dei dati generali insieme alle sezioni delle spedizioni e dei cargo_item: AIDA genera il protocollo del manifesto;

- e dopo l'invio del flusso contenente la sezione dei dati generali:

    - flusso XML contenente soltanto le sezioni delle spedizioni e dei cargo_item;

    - flusso XML contenente la sola sezione dei cargo item;

    - flusso XML integrativo pervenuto dopo la convalida;

    - flusso XML di convalida del manifesto,

il cui utilizzo è differenziato per i diversi attori che partecipano al processo:

- **carrier**: è il vettore responsabile del manifesto;
- **spedizioniere**: agente incaricato dal vettore alla trasmissione del manifesto;
- **subagente**: costituisce una porzione specifica del manifesto;

3.2. Attori e ruoli

L'apertura e la convalida del manifesto sono a cura esclusiva del carrier o, se presente, dello spedizioniere incaricato dal carrier.

Al momento dell'apertura del manifesto, il carrier o lo spedizioniere definiscono l'eventuale lista dei subagenti che da quel momento possono concorrere alla formazione progressiva del manifesto; i subagenti possono integrare il manifesto solo con ulteriori spedizioni e cargo_item.

Se lo spedizioniere compila i dati delle spedizioni e dei cargo item, questi ultimi sono sempre attribuiti al carrier.

Al momento dell'apertura, il sistema assegna un protocollo al manifesto.

Il carrier/spedizioniere comunica il protocollo e il numero di pratica utilizzato (LRN – Local Reference Number) ai subagenti, ove presenti, per consentire ad essi la trasmissione di integrazioni successive.

Ogni subagente è svincolato rispetto agli altri nella numerazione delle spedizioni e dei cargo_item; il sistema restituisce gli esiti dell'acquisizione, in modalità esclusiva, a ciascun subagente.

Anche dopo la convalida del manifesto possono essere effettuate ulteriori integrazioni.

In figura sono riportati i ruoli dei diversi attori che contribuiscono a gestire un manifesto:

Ruoli	Apertura/integrazioni manifesto	Rettifica	Annullamento	Consultazione	Convalida Manifesto
Carrier	effettua apertura manifesto/ trasmette integrazioni manifesto	SI, rettifica solo i propri dati	SI, annulla anche i dati dei subagenti se non presente lo spedizioniere	SI, consulta solo i propri dati	SI
Spedizioniere	effettua apertura manifesto/ trasmette integrazioni manifesto	SI, rettifica solo i propri dati	SI, annulla anche i dati del carrier e dei subagenti	SI, consulta solo i propri dati	SI
Subagente	trasmette integrazioni manifesto	SI, rettifica solo i propri dati	SI, annulla solo i propri dati	SI, consulta solo i propri dati	NO

Tabella 3 – Ruoli e operazioni consentite sul manifesto

3.3. Il servizio EManifest

Il servizio "**EManifestService"** del tipo EJB – WS, mette a disposizione le seguenti operazioni relative all'accoglienza dei **Manifesti delle Merci in arrivo (MMA)** e dei **Manifesti delle Merci in Partenza (MMP)**:

- submission (registrazione del Manifesto);
- amendment (rettifica di un componente del Manifesto);
- invalidation (cancellazione globale o di un elemento del Manifesto);
- updating (integrazione del Manifesto);
- validation (convalida del Manifesto);
- summaryResult (lista delle partite o degli MRN non imbarcabili).

Nei paragrafi successivi sono descritti i dati di interscambio, le operazioni ed i parametri di input/output per ogni operazione.

3.4. Endpoint

In ambiente di prova il servizio è esposto con il seguente endpoint:

https://interoptest.adm.gov.it/EManifestServiceWeb/services/EManifest

3.5. Metodo Process

Il metodo *process* permette l'elaborazione dei **Manifesti delle Merci in arrivo (MMA) e Manifesti delle Merci in Partenza (MMP)**.
Per ogni elaborazione effettuata verrà indicata l'operazione che è stata innescata con i relativi dati di input (Richiesta) e di output (Risposta). Ogni operazione è identificata mediante un *serviceId*.

Il servizio del tipo EJB - WS, avrà la seguente operazione esposta:

- Risposta *process* (Richiesta input) ed i seguenti parametri:

Metodo	Input	Output
process	Richiesta	Risposta

Tabella 4 - Descrizione metodo process

I dati in input relativi al tracciato "Richiesta" sono descritti in dettaglio nell'allegato tecnico.

Il tipo di dati in output "Risposta" descritto in dettaglio nell'allegato tecnico, contiene i seguenti elementi:

- *IUT*: identificativo univoco transazione;
- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione della richiesta, più propriamente descritto nel paragrafo 9.7;
- *data*: in questo elemento sarà presente la risposta codificata in base64Binary (opzionale);
- *data di registrazione:* data in cui il messaggio è pervenuto al sistema di accoglienza.

3.6. Le operazioni

3.6.1. Registrazione del Manifesto

L'operazione "**submission**" (registrazione) consente la registrazione di un Manifesto MMA o MMP.

Per invocare il servizio è necessario creare un messaggio SOAP di fruizione che deve contenere nel tipo di dati in input "Richiesta" i seguenti campi obbligatori:

- serviceId
- data
  - xmlList

- dichiarante

Segue la descrizione dei campi:

- *serviceId*: indica il tipo di operazione da eseguire. Necessario per effettuare il dispatching verso i servizi richiesti. Nel caso specifico della suddetta operazione va indicato: "**submission**";

- *data*: rappresenta una collezione di oggetti contenenti:

  - xmlList: contiene il messaggio xml valido per l'XSD descritto in dettaglio nell'allegato tecnico (EMANIFEST_INPUT.xsd) firmato digitalmente secondo le indicazioni espresse nel paragrafo 2.3 e codificato in base64binary;

  - dichiarante: l'identificazione dell'utente dichiarante (codice fiscale del soggetto per il quale si sta inviando il messaggio).

Una volta inviato il messaggio, il servizio restituirà in output un oggetto "Risposta" descritto in dettaglio nell'allegato tecnico che contiene i seguenti elementi:

- *IUT*: identificativo univoco transazione;

- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione più propriamente descritto nel paragrafo 9.7;

- *data*: in questo elemento sarà presente la risposta codificata in base64Binary (opzionale);

- *data di registrazione:* data in cui il messaggio è pervenuto al sistema di accoglienza.

Il manifesto viene registrato dal carrier/spedizioniere consumando il servizio EManifestService con **ITATipoOperazioneType = S** (modalità Submission)

Il flusso XML può contenere solo i dati generali ovvero i dati generali ed elementi di spedizione/cargo item.

All'atto della ricezione del flusso XML il sistema predispone un esito di risposta di primo livello costituito da:

a) una codifica di errore (il flusso è respinto);

b) il protocollo assegnato all'eManifest seguito:

   - nel caso di MMA, dalla lista delle partite generate dal sistema (MRN+Item e registro A3/PF associato) senza le informazioni di sicurezza;

- nel caso di MMP, dalla lista degli MRN+ITEM non autorizzati all'imbarco in quanto rischiosi

3.6.2. Rettifica del Manifesto

L'operazione "**amendment**", del tipo EJB - WS, consente di effettuare la rettifica di un Manifesto MMA o MMP.

Per invocare il servizio è necessario creare un messaggio SOAP di fruizione che deve contenere nel tipo di dati in input "Richiesta" i seguenti campi obbligatori:

- serviceId
- data
  - xmlList
  - dichiarante

Segue la descrizione dei campi:

- *serviceId*: indica il tipo di operazione da eseguire. Necessario per effettuare il dispatching verso i servizi richiesti. Nel caso specifico della suddetta operazione va indicato: "**amendment**";

- *data*: rappresenta una collezione di oggetti contenenti:

  - xmlList: contiene il messaggio XML valido per l'XSD descritto in dettaglio nell'allegato tecnico (EMANIFEST_INPUT.xsd) firmato digitalmente secondo le indicazioni espresse nel paragrafo 2.3 e codificato in base64binary;

  - dichiarante: l'identificazione dell'utente dichiarante (codice fiscale del soggetto per il quale si sta inviando il messaggio).

Una volta inviato il messaggio, il servizio restituirà in output un oggetto "Risposta" descritto in dettaglio nell'allegato tecnico che contiene i seguenti elementi:

- *IUT*: identificativo univoco transazione;

- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione più propriamente descritto nel paragrafo 9.7;

- *data*: in questo elemento sarà presente la risposta codificata in base64Binary (opzionale);

- *data di registrazione:* data in cui il messaggio è pervenuto al sistema di accoglienza.

Il manifesto viene rettificato dal carrier/spedizioniere/subagente consumando il servizio EManifestService con **ITATipoOperazioneType = A** (modalità Amendment).

Lo spedizioniere, il carrier e i subagenti possono rettificare solo i dati di propria pertinenza.

Sono contemplate 3 tipologie di rettifica:

- rettifica dei dati generali dell'eManifest;
- rettifica dei dati di una delle N spedizioni componenti;
- rettifica dei dati di uno degli M cargo_item di una spedizione

Il servizio consente di indicare più richieste di rettifica associate a diversi elementi di diversi manifesti.

La rettifica di un elemento del manifesto successiva alla convalida necessita della convalida in AIDA da parte dell'ufficio competente, quindi viene restituito un primo esito di 'presa in carico della richiesta'.

A seguito dell'avvenuto monitoraggio da parte dell'ufficio doganale, AIDA genera un secondo livello di esito che indica se la rettifica è stata accettata o rifiutata.

3.6.3. Cancellazione globale o di un elemento del Manifesto

L'operazione "**invalidation**" (cancellazione) del tipo EJB - WS, consente di effettuare l'annullamento di un Manifesto MMA o MMP inviato in precedenza.

Per invocare il servizio è necessario creare un messaggio SOAP di fruizione che deve contenere nel tipo di dati in input "Richiesta" i seguenti campi obbligatori:

- serviceId
- data
  - xmlList
  - dichiarante

Segue la descrizione dei campi:

- *serviceId*: indica il tipo di operazione da eseguire. Necessario per effettuare il dispatching verso i servizi richiesti. Nel caso specifico della suddetta operazione va indicato: "**invalidation**";
- *data*: rappresenta una collezione di oggetti contenenti:
  - xmlList: contiene il messaggio xml valido per l'XSD descritto in dettaglio nell'allegato tecnico (EMANIFEST_INPUT.xsd) firmato digitalmente

secondo le indicazioni espresse nel paragrafo 2.3 e codificato in base64binary;

- dichiarante: l'identificazione dell'utente dichiarante (codice fiscale del soggetto per il quale si sta inviando il messaggio).

Una volta inviato il messaggio, il servizio restituirà in output un oggetto "Risposta" descritto in dettaglio nell'allegato tecnico che contiene i seguenti elementi:

- *IUT*: identificativo univoco transazione;
- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione più propriamente descritto nel paragrafo 9.7;
- *data*: in questo elemento sarà presente la risposta codificata in base64Binary (opzionale);
- *data di registrazione:* data in cui il messaggio è pervenuto al sistema di accoglienza.

Il manifesto viene cancellato globalmente solo dal carrier (se non presente lo spedizioniere) o dallo spedizioniere, mentre gli elementi del manifesto possono essere cancellati, secondo competenza, dal carrier/spedizioniere/subagente.

Il servizio EManifestService viene utilizzato con **ITATipoOperazioneType = I** (modalità Invalidation).

Si può procedere in alternativa con:

- l'annullamento globale dell'eManifest;
- l'annullamento puntuale di una spedizione (comporta l'annullamento di tutti i cargo_item collegati);
- l'annullamento puntuale di cargo_item collegato ad una specifica spedizione

All'atto della ricezione del flusso XML il sistema predispone in output una risposta che riporta il protocollo del manifesto cancellato globalmente ovvero gli estremi dell'elemento annullato.

3.6.4. Integrazione del Manifesto

L'operazione "**updating**" (integrazione) consente l'integrazione di un Manifesto MMA o MMP.

Per invocare il servizio, viene creato un messaggio SOAP di fruizione che deve contenere nel tipo di dati in input "Richiesta" i seguenti campi obbligatori:

- serviceId
- data
    - xmlList
    - dichiarante

Segue la descrizione dei campi:

- *serviceId*: indica il tipo di operazione da eseguire. Necessario per effettuare il dispatching verso i servizi richiesti. Nel caso specifico della suddetta operazione va indicato: "**updating**";

- *data*: rappresenta una collezione di oggetti contenenti:

    - xmlList: contiene il messaggio xml valido per l'XSD descritto in dettaglio nell'allegato tecnico (EMANIFEST_INPUT.xsd) firmato digitalmente secondo le indicazioni espresse nel paragrafo 2.3 e codificato in base64binary;
    - dichiarante: l'identificazione dell'utente dichiarante (codice fiscale del soggetto per il quale si sta inviando il messaggio).

Una volta inviato il messaggio, il servizio restituirà in output un oggetto "Risposta" descritto in dettaglio nell'allegato tecnico che contiene i seguenti elementi:

- *IUT*: identificativo univoco transazione;
- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione più propriamente descritto nel paragrafo 9.7;

- *data*: in questo elemento sarà presente la risposta codificata in base64Binary (opzionale);

- *data di registrazione:* data in cui il messaggio è pervenuto al sistema di accoglienza.

Il manifesto viene integrato dal carrier/spedizioniere/subagente consumando il servizio EManifestService con **ITATipoOperazioneType = U** (modalità Updating).

Il flusso di input può contenere elementi di spedizione (ciascuna contenente almeno un cargo item) o elementi di soli cargo item.

All'atto della ricezione del flusso, il sistema predispone una risposta di primo livello costituita da una codifica di errore se il flusso è respinto, ovvero in caso di esito positivo:

- nel caso di MMA, una lista delle partite generate dal sistema (MRN+Item e registro A3/PF associato) senza le informazioni su eventuali controlli di sicurezza (safety & security);

- nel caso di MMP, la lista degli MRN+ITEM non autorizzati all'imbarco in quanto rischiosi.

Nel caso di integrazioni pervenute dopo la convalida, il manifesto assume lo stato di 'parzialmente convalidato'.
La convalida di tali integrazioni, a carico dell'ufficio doganale mediante applicazione web in AIDA, produce un aggiornamento della risposta di primo livello.

3.6.5. Convalida del Manifesto

L'operazione "**validation**" (validazione) consente di effettuare la convalida di un MMA/MMP, per un determinato Codice Manifesto (protocollo e-Manifest), da parte del responsabile (il ricorso a tale servizio deve essere inteso come operazione di fall-back, in quanto la convalida del manifesto nel sistema reingegnerizzato avviene mediante colloquio con il sistema informativo della Capitaneria di Porto).

Per invocare il servizio, viene creato un messaggio SOAP di fruizione che deve contenere nel tipo di dati in input "Richiesta" i seguenti campi obbligatori:

- serviceId
- data
  - xmlList
  - dichiarante

Segue la descrizione dei campi:

- *serviceId*: indica il tipo di operazione da eseguire. Necessario per effettuare il dispatching verso i servizi richiesti. Nel caso specifico della suddetta operazione va indicato: "**validation**";

- *data*: rappresenta una collezione di oggetti contenenti:

  - xmlList: contiene il messaggio xml valido per l'XSD descritto in dettaglio nell'allegato tecnico (EMANIFEST_INPUT.xsd) firmato digitalmente secondo le indicazioni espresse nel paragrafo 2.3 e codificato in base64binary;
  - dichiarante: l'identificazione dell'utente dichiarante (codice fiscale del soggetto per il quale si sta inviando il messaggio).

Una volta inviato il messaggio, il servizio restituirà in output un oggetto "Risposta" descritto in dettaglio nell'allegato tecnico che contiene i seguenti elementi:

- *IUT: identificativo univoco transazione;*
- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione più propriamente descritto nel paragrafo 9.7;

- *data*: in questo elemento sarà presente la risposta codificata in base64Binary (opzionale);

- *data di registrazione:* data in cui il messaggio è pervenuto al sistema di accoglienza.

Il manifesto viene convalidato dal carrier/spedizioniere consumando il servizio EManifestService con **ITATipoOperazioneType = V** (modalità Validation).
All'atto della ricezione del flusso, il sistema predispone una risposta di primo livello costituita da una codifica di errore se il flusso è respinto, ovvero in caso di esito positivo

- nel caso di MMA, dal protocollo del manifesto seguito dalla lista delle partite generate dal sistema (MRN+ITEM e registro A3/PF associato) con le informazioni relative alla necessità un eventuale controllo sicurezza (safety & security);
- nel caso di MMP, dal protocollo del manifesto seguito dalla lista degli MRN+ITEM cancellati dal sistema in quanto rischiosi.

L'apposizione dell'esito del controllo Safety & Security, a carico dell'ufficio doganale mediante applicazione web in AIDA, produce un aggiornamento della risposta di primo livello con la notifica di sicurezza opportuna, per le merci contenute nei manifesti in arrivo e/o partenza.

3.6.6. Lista delle partite e degli MRN non imbarcabili

L'operazione "**summaryResult**" (esito riepilogativo) consente di inviare la richiesta di un esito riepilogativo, per un determinato Codice Manifesto (protocollo e-Manifest), che consente di recuperare la lista di partite A3 generate, nel caso MMA, o la lista degli MRN non imbarcabili, autorizzati all'imbarco o cancellati da sistema, nel caso MMP.
Il servizio viene consumato indicando **ITATipoOperazioneType = SR**.

Per invocare il servizio, viene creato un messaggio SOAP di fruizione che deve contenere nel tipo di dati in input "Richiesta" i seguenti campi obbligatori:

- serviceId
- data
  - xmlList
  - dichiarante

Segue la descrizione dei campi:

- *serviceId*: indica il tipo di operazione da eseguire. Necessario per effettuare il dispatching verso i servizi richiesti. Nel caso specifico della suddetta operazione va indicato: "**summaryResult**";

- *data*: rappresenta una collezione di oggetti contenenti:
  - xmlList: contiene il messaggio xml valido per l'XSD descritto in dettaglio

nell'allegato tecnico (EMANIFEST_OUTPUT.xsd) firmato digitalmente secondo le indicazioni espresse nel paragrafo 2.3 e codificato in base64binary;
- *dichiarante:* l'identificazione dell'utente dichiarante (codice fiscale del soggetto per il quale si sta inviando il messaggio).

Una volta inviato il messaggio, il servizio restituirà in output un oggetto "Risposta" descritto in dettaglio nell'allegato tecnico che contiene i seguenti elementi:

- *IUT*: identificativo univoco transazione;
- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione più propriamente descritto nel paragrafo 9.7;
- *data*: in questo elemento sarà presente la risposta codificata in base64Binary (opzionale);
- *data di registrazione:* data in cui il messaggio è pervenuto al sistema di accoglienza.

Il carrier, lo spedizioniere e il subagente, quindi, possono utilizzare tale servizio in modalità SummaryResult.
L'esito riepilogativo può rappresentare un esito di primo livello o successivo, in funzione dell'avvenuta convalida o meno del manifesto.

4. I SERVIZI PER LA TEMPORANEA CUSTODIA

4.1. Tipologia di dichiarazioni accolte

Il sistema di accoglienza consente all'Operatore Economico di trasmettere e gestire le **Partite di Temporanea Custodia (TC)**, mettendo a disposizione un insieme di funzionalità esposte tramite Web Services, implementate per:

- registrare o modificare partite di temporanea custodia non iscritte a manifesto (TEMPORARYSTORAGEService);
- interrogare le merci che entrano/escono in/da un magazzino di temporanea custodia (CHECKService);
- richiedere il trasferimento della merce da un magazzino ad un altro o interrogare lo stato della richiesta di trasferimento (STORAGETRANSFERService);
- comunicare l'ingresso della merce nel magazzino di temporanea custodia (CNTINSTORAGEService);
- comunicare le discrepanze rilevate sulla merce in ingresso o in uscita al/dal magazzino di temporanea custodia (DISCREPANCYService).

4.2. Attori e ruoli

I servizi possono essere fruiti dai seguenti attori:

- Operatore economico (detentore della merce o suo rappresentante);
- Gestore del magazzino di temporanea custodia;
- Handler che opera nei nodi logistici portuali o di destinazione, ossia negli impianti in cui sono movimentate le merci.

secondo quanto indicato dallo schema seguente:

	TEMPORARY STORAGE	**CHECK**	**STORAGE TRANSFER**	**CNTIN STORAGE**	**DISCREPANCY**
Operatori economici	SI	NO	NO	NO	NO
Gestori T.C.	NO	SI	SI	SI	SI
Handler	NO	SI, solo in modalità Incoming	NO	NO	NO

Tabella 5 - Fruitori dei servizi

Di seguito vengono descritti tutti i servizi riguardanti la temporanea custodia con le relative operazioni

4.3. Il servizio TEMPORARYSTORAGE (Temporanea Custodia)

Il servizio "**TEMPORARYSTORAGEService**", del tipo EJB – WS, mette a disposizione le

seguenti operazioni relative al servizio di **acquisizione delle schede partita di Temporanea Custodia**:

- submission (registrazione delle schede partita di temporanea custodia non iscritte a Manifesto);
- amendment (rettifica delle schede partite di temporanea custodia non iscritte a Manifesto);

Di seguito sono descritti i dati di interscambio ed i parametri di input/output per ogni operazione elencata.

4.3.1. Endpoint

In ambiente di prova l'endpoint con cui il servizio è esposto è:

https://interoptest.adm.gov.it/TEMPORARYSTORAGEServiceWeb/services/TEMPORARYSTORAGEService

4.3.2. Metodo Process

Il metodo process permette l'elaborazione delle operazioni relative al servizio di acquisizione delle schede partita di Temporanea Custodia.
Per ogni elaborazione effettuata verrà indicata l'operazione che è stata innescata con i relativi dati di input (Richiesta) e di output (Risposta). Ogni operazione è identificata mediante un serviceId.

Il servizio del tipo EJB - WS ha la seguente operazione esposta:

- Risposta *process* (Richiesta input) e i seguenti parametri:

Metodo	Input	Output
process	Richiesta	Risposta

Tabella 6 - Descrizione metodo process

I dati in input relativi al tracciato "Richiesta" sono descritti in dettaglio nell'allegato tecnico.

Il tipo di dati in output "Risposta" descritto in dettaglio nell'allegato tecnico, contiene i seguenti elementi:

- *IUT*: identificativo univoco transazione;

- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione della richiesta, più propriamente descritto nel paragrafo 9.7;

- *data*: in questo elemento sarà presente la risposta codificata in base64Binary (opzionale);

- *data di registrazione:* data in cui il messaggio è pervenuto al sistema di accoglienza.

4.3.3. Le operazioni

Il servizio, se utilizzato in modalità "submission" consente agli operatori economici di acquisire una dichiarazione di temporanea custodia ovvero, se utilizzato in modalità "amendment", consente di rettificare una dichiarazione di temporanea custodia precedentemente acquisita.

4.3.3.1. Registrazione delle Schede Partita di Temporanea Custodia

L'operazione "**submission**" (registrazione) consente l'acquisizione di una scheda partita di temporanea custodia.

Per invocare il servizio, viene creato un messaggio SOAP di fruizione che deve contenere nel tipo di dati in input "Richiesta" i seguenti campi obbligatori:

- serviceId
- data
    - xmlList
    - dichiarante

Segue la descrizione dei campi:

- *serviceId*: indica il tipo di operazione da eseguire. Necessario per effettuare il dispatching verso i servizi richiesti. Nel caso specifico della suddetta operazione va indicato: "**submission**";

- *data*: rappresenta una collezione di oggetti contenenti:

    - *xmlList:* contiene il messaggio xml valido per l'XSD descritto in dettaglio nell'allegato tecnico (TEMPORARY_STORAGE_INPUT.xsd) firmato digitalmente secondo le indicazioni espresse nel paragrafo 2.3 e codificato in base64binary;

    - *dichiarante:* l'identificazione dell'utente dichiarante (codice fiscale del soggetto per il quale si sta inviando il messaggio).

Una volta inviato il messaggio, il servizio restituirà in output un oggetto "Risposta" descritto in dettaglio nell'allegato tecnico che contiene i seguenti elementi:

- *IUT*: identificativo univoco transazione;
- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione più propriamente descritto nel paragrafo 9.7;
- *data*: in questo elemento sarà presente la risposta codificata in base64Binary (opzionale);
- *data di registrazione:* data in cui il messaggio è pervenuto al sistema di accoglienza.

L'operazione **submission** consente di registrare a sistema le schede partita di temporanea custodia non iscritte a manifesto e può essere fruita impostando opportunamente il campo **ITATipoOperazioneType = S.**

I dati di input del servizio sono i seguenti:

- **Dati di testata,** fra i quali è presente il ITATipoOperazioneType che indica la modalità in cui si intende consumare il servizio;

I dati di testata sono seguiti da:

- **Body di acquisizione di temporanea custodia,** ossia la lista dei dati delle dichiarazioni da acquisire.

L'output è costituito da una lista di elementi che riportano:

- **esito:** stato di elaborazione (positivo/negativo);
- **descrizione errore** (solo se esito negativo)**;**
- **estremi della partita** generata/rettificata.

4.3.3.2. Rettifica delle Schede Partita di Temporanea Custodia

L'operazione "**amendment**" (rettifica) consente di effettuare la rettifica di una scheda partita di temporanea custodia.

Per invocare il servizio, viene creato un messaggio SOAP di fruizione che deve contenere nel tipo di dati in input "Richiesta" i seguenti campi obbligatori:

- serviceId
- data
  - xmlList
  - dichiarante

Segue la descrizione dei campi:

- *serviceId*: indica il tipo di operazione da eseguire. Necessario per effettuare il

dispatching verso i servizi richiesti. Nel caso specifico della suddetta operazione va indicato: "**amendment**";

- *data*: rappresenta una collezione di oggetti contenenti:
    - *xmlList:* contiene il messaggio xml valido per l'XSD descritto in dettaglio nell'allegato tecnico (TEMPORARY_STORAGE_INPUT.xsd) firmato digitalmente secondo le indicazioni espresse nel paragrafo 2.3 e codificato in base64binary;
    - *dichiarante*: l'identificazione dell'utente dichiarante (codice fiscale del soggetto per il quale si sta inviando il messaggio).

Una volta inviato il messaggio, il servizio restituirà in output un oggetto "Risposta" descritto in dettaglio nell'allegato tecnico che contiene i seguenti elementi:

- *IUT*: identificativo univoco transazione;
- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione più propriamente descritto nel paragrafo 9.7;
- *data*: in questo elemento sarà presente la risposta codificata in base64Binary (opzionale);
- *data di registrazione:* data in cui il messaggio è pervenuto al sistema di accoglienza.

L'operazione **amendment** consente quindi di rettificare le schede partita di temporanea custodia non iscritte a manifesto e può essere fruita impostando opportunamente il campo **ITATipoOperazioneType = A**.

I dati di input del servizio sono i seguenti:

- **Dati di testata,** fra i quali è presente il ITATipoOperazioneType che indica la modalità in cui si intende consumare il servizio;

I dati di testata sono seguiti da:

- **Body di rettifica di temporanea custodia,** ovvero la lista dei dati delle partite da rettificare.

L'output è costituito da una lista di elementi che riportano:

- **esito:** stato di elaborazione (positivo/negativo);
- **descrizione errore** (solo se esito negativo)**;**
- **estremi della partita** generata/rettificata.

4.4. Il servizio DISCREPANCY (Discrepanze sulle merci)

Il servizio "**DISCREPANCYService**", del tipo EJB – WS, mette a disposizione la seguente operazione per l'acquisizione di un messaggio XML relativo alle discrepanze per le merci in entrata/uscita:

- submission (registrazione della lista delle discrepanze riscontrate sulle merci in ingresso/uscita nei magazzini).

Di seguito sono descritti i dati di interscambio ed i parametri di input/output dell'operazione indicata.

4.4.1. Endpoint

In ambiente di prova l'endpoint con cui il servizio è esposto è:

https://interoptest.adm.gov.it/DISCREPANCYServiceWeb/services/DISCREPANCYService

4.4.2. Metodo Process

Il metodo process permette l'elaborazione dei messaggi XML relativi alle discrepanze per le merci in entrata/uscita.
Per ogni elaborazione effettuata viene indicata l'operazione che è stata innescata con i relativi dati di input (Richiesta) e di output (Risposta). Ogni operazione è identificata mediante un serviceId.

Il servizio del tipo EJB - WS ha la seguente operazione esposta:

- Risposta *process* (Richiesta input) Ed i seguenti parametri:

Metodo	Input	Output
process	Richiesta	Risposta

Tabella 7 - Descrizione metodo process

I dati in input relativi al tracciato "Richiesta" sono descritti in dettaglio nell'allegato tecnico.

Il tipo di dati in output "Risposta" descritto in dettaglio nell'allegato tecnico, contiene i

seguenti elementi:

- *IUT*: identificativo univoco transazione;
- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione della richiesta, più propriamente descritto nel paragrafo 9.7;
- *data*: in questo elemento è presente la risposta codificata in base64Binary (opzionale);
- *data di registrazione:* data in cui il messaggio è pervenuto al sistema di accoglienza.

4.4.3. Registrazione della lista delle discrepanze

L'operazione "**submission**" (registrazione) consente l'acquisizione di un XML relativo alle discrepanze.

Per invocare il servizio, viene creato un messaggio SOAP di fruizione che deve contenere nel tipo di dati in input "Richiesta" i seguenti campi obbligatori:

- serviceId
- data
  - xmlList
  - dichiarante

Segue la descrizione dei campi:

- *serviceId*: indica il tipo di operazione da eseguire. Necessario per effettuare il dispatching verso i servizi richiesti. Nel caso specifico della suddetta operazione va indicato: "**submission**";
- *data*: rappresenta una collezione di oggetti contenenti:
  - *xmlList:* contiene il messaggio xml valido per l'XSD descritto in dettaglio nell'allegato tecnico (DISCREPANCY_INPUT.xsd) firmato digitalmente secondo le indicazioni espresse nel paragrafo 2.3 e codificato in base64binary;
  - *dichiarante*: l'identificazione dell'utente dichiarante (codice fiscale del soggetto per il quale si sta inviando il messaggio).

Una volta inviato il messaggio, il servizio restituisce in output un oggetto "Risposta" descritto in dettaglio nell'allegato tecnico che contiene i seguenti elementi:

- *IUT*: identificativo univoco transazione;
- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione più propriamente descritto nel paragrafo 9.7;

- *data*: in questo elemento è presente la risposta codificata in base64Binary (opzionale);

- *data di registrazione:* data in cui il messaggio è pervenuto al sistema di accoglienza.

Il servizio è utilizzato dai gestori di Temporanea Custodia per comunicare all'ufficio doganale di competenza la lista delle discrepanze riscontrate sulle merci in ingresso/uscita nei magazzini di competenza rispetto ai dati indicati sui manifesti in arrivo o partenza, sulle dichiarazioni di temporanea custodia e sulle richieste di trasferimento di magazzino.

La comunicazione della lista delle discrepanze avviene mediante un opportuno popolamento dei BODY di ingresso/uscita.

Il servizio restituisce un output che riporta un elenco di elementi costituito da:

- Progressivo discrepanza;
- Esito della segnalazione della discrepanza (positivo/negativo);
- Se l'esito è positivo viene rappresentata l'istanza della discrepanza segnalata in input, altrimenti la descrizione dell'errore.

4.5. Il servizio STORAGE TRANSFER (Interrogazione Movimentazione delle Merci)

Il servizio "**STORAGETRANSFERService**", del tipo EJB –WS, mette a disposizione le seguenti operazioni relative alla gestione dei trasferimenti di magazzino:

- authorizationRequest (richiesta autorizzazione al trasferimento della merce dal magazzino);
- authorizationsStatus (interrogazione dello stato dell'autorizzazione al trasferimento precedentemente richiesta).
- authorizationTransferIOT (richiesta di trasferimento di merce indicando IOT di un boooking)

Di seguito sono descritti i dati di interscambio ed i parametri di input/output dell'operazione indicata.

4.5.1. Endpoint

In ambiente di prova l'endpoint con cui il servizio è esposto è:

https://interoptest.adm.gov.it/STORAGETRANSFERServiceWeb/services/STORAGETRANSFERService

4.5.2. Metodo Process

Il metodo process permette l'elaborazione dei messaggi XML relativi alla gestione dei trasferimenti di magazzino.
Per ogni elaborazione effettuata viene indicata l'operazione che è stata innescata con i relativi dati di input (Richiesta) e di output (Risposta). Ogni operazione è identificata mediante un serviceId.

Il servizio del tipo EJB - WS ha la seguente operazione esposta:

- Risposta *process* (Richiesta input) e i seguenti parametri:

Metodo	Input	Output
process	Richiesta	Risposta

Tabella 8 - Descrizione metodo process

I dati in input relativi al tracciato "Richiesta" sono descritti in dettaglio nell'allegato tecnico.

Il tipo di dati in output "Risposta", descritto in dettaglio nell'allegato tecnico, contiene i seguenti elementi:

- *IUT*: identificativo univoco transazione;
- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione della richiesta, più propriamente descritto nel paragrafo 9.7;
- *data*: in questo elemento è presente la risposta codificata in base64Binary (opzionale);
- *data di registrazione:* data in cui il messaggio è pervenuto al sistema di accoglienza.

4.5.3. Le operazioni

Il servizio è utilizzato dal gestore di temporanea custodia per richiedere all'ufficio doganale di competenza l'autorizzazione a movimentare la merce dal proprio magazzino ad altro magazzino di destinazione, ovvero per interrogare lo stato delle richieste di trasferimento precedentemente effettuate.

4.5.3.1. Richiesta di Autorizzazione al Trasferimento della Merce

L'operazione "**authorizationRequest**" (richiesta autorizzazione) consente l'acquisizione di un messaggio xml relativo ad una richiesta di autorizzazione al trasferimento della merce.

Per invocare il servizio, viene creato un messaggio SOAP di fruizione che deve contenere nel tipo di dati in input “Richiesta” i seguenti campi obbligatori:

- serviceId
- data
  - xmlList
  - dichiarante

Segue la descrizione dei campi:

- *serviceId*: indica il tipo di operazione da eseguire. Necessario per effettuare il dispatching verso i servizi richiesti. Nel caso specifico della suddetta operazione va indicato: “**authorizationRequest**”;

- *data*: rappresenta una collezione di oggetti contenenti:

  - *xmlList:* contiene il messaggio xml valido per l’XSD descritto in dettaglio nell’allegato tecnico (STORAGE_TRANSFER_INPUT.xsd) firmato digitalmente secondo le indicazioni espresse nel paragrafo 2.3 e codificato in base64binary;

  - *dichiarante*: l’identificazione dell’utente dichiarante (codice fiscale del soggetto per il quale si sta inviando il messaggio).

Una volta inviato il messaggio, il servizio restituisce in output un oggetto “Risposta” descritto in dettaglio nell’allegato tecnico che contiene i seguenti elementi:

- *IUT*: identificativo univoco transazione;

- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione più propriamente descritto nel paragrafo 9.7;

- *data*: in questo elemento è presente la risposta codificata in base64Binary (opzionale);

- *data di registrazione:* data in cui il messaggio è pervenuto al sistema di accoglienza.

L’operazione **authorizationRequest** consente al gestore T.C. di richiedere il trasferimento della merce dal proprio magazzino di competenza ad un altro magazzino e può essere fruita impostando opportunamente il campo **ITATipoOperazioneType = R**.

In modalità **authorizationRequest**, l’input è rappresentato da una lista di elementi contenenti gli estremi delle partite da trasferire, nonché il magazzino di partenza e di destinazione.

L’output è costituito dai seguenti elementi:

- la richiesta effettuata;

- l'esito dell'elaborazione (positivo/negativo);
- una sezione dedicata ad evidenziare gli estremi della nuova partita generata nel magazzino di destinazione, a fronte dell'autorizzazione al trasferimento concessa dal funzionario doganale.

4.5.3.2. Verifica dello Stato di Avanzamento di una Autorizzazione

L'operazione "**authorizationsStatus**" (interrogazione stato autorizzazione) consente di effettuare la verifica dello stato di avanzamento delle autorizzazioni al trasferimento della merce.

Per invocare il servizio, viene creato un messaggio SOAP di fruizione che deve contenere nel tipo di dati in input "Richiesta" i seguenti campi obbligatori:

- serviceId
- data
  - xmlList
  - dichiarante

Segue la descrizione dei campi:

- *serviceId*: indica il tipo di operazione da eseguire. Necessario per effettuare il dispatching verso i servizi richiesti. Nel caso specifico della suddetta operazione va indicato: "**authorizationsStatus**";

- *data*: rappresenta una collezione di oggetti contenenti:

  - *xmlList:* contiene il messaggio xml valido per l'XSD descritto in dettaglio nell'allegato tecnico (STORAGE_TRANSFER_INPUT.xsd) firmato digitalmente secondo le indicazioni espresse nel paragrafo 2.3 e codificato in base64binary;

  - *dichiarante*: l'identificazione dell'utente dichiarante (codice fiscale del soggetto per il quale si sta inviando il messaggio).

Una volta inviato il messaggio, il servizio restituirà in output un oggetto "Risposta" descritto in dettaglio nell'allegato tecnico che contiene i seguenti elementi:

- *IUT*: identificativo univoco transazione;

- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione più propriamente descritto nel paragrafo 9.7;

- *data*: in questo elemento sarà presente la risposta codificata in base64Binary (opzionale);

- *data di registrazione:* data in cui il messaggio è pervenuto al sistema di

accoglienza.

L'operazione **authorizationStatus** consente al gestore T.C. di interrogare lo stato dell'autorizzazione del trasferimento precedentemente richiesto e può essere fruita impostando opportunamente il campo **ITATipoOperazioneType = S**.

In modalità **authorizationStatus**, l'input è rappresentato:

- dagli estremi della partita per cui è stata richiesta l'autorizzazione al trasferimento e la data in cui è stata effettuata la richiesta;
- da un range di date per le quali si ricercano le richieste di trasferimento effettuate.

L'output è costituito dai seguenti elementi:

- estremi della richiesta effettuata;
- elenco di elementi differenziati per esito (positivo o negativo)

  Se negativo viene riportata una descrizione di errore, se positivo viene riportato un dettaglio della richiesta effettuata, gli estremi della partita oggetto del trasferimento, ed eventuale data di autorizzazione concessa dal funzionario.

4.5.3.3. Verifica dello Stato di Avanzamento di una Autorizzazione

L'operazione **"authorizationTransferIOT"** (richiesta di trasferimento tramite IOT) consente di trasferire la merce precedentemente indicata in un Booking e aggregata tramite IOT.

Per invocare il servizio, viene creato un messaggio SOAP di fruizione che deve contenere nel tipo di dati in input "Richiesta" i seguenti campi obbligatori:

- serviceId
- data
  - xmlList
  - dichiarante

Segue la descrizione dei campi:

- *serviceId*: indica il tipo di operazione da eseguire. Necessario per effettuare il dispatching verso i servizi richiesti. Nel caso specifico della suddetta operazione va indicato: **"authorizationTransferIOT"**;

- *data*: rappresenta una collezione di oggetti contenenti:

  - *xmlList:* contiene il messaggio xml valido per l'XSD descritto in dettaglio nell'allegato tecnico (STORAGE_TRANSFER_INPUT.xsd) firmato digitalmente secondo le indicazioni espresse nel paragrafo 2.3 e codificato in base64binary;

- *dichiarante*: l'identificazione dell'utente dichiarante (codice fiscale del soggetto per il quale si sta inviando il messaggio).

Una volta inviato il messaggio, il servizio restituirà in output un oggetto "Risposta" descritto in dettaglio nell'allegato tecnico che contiene i seguenti elementi:

- *IUT*: identificativo univoco transazione;
- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione più propriamente descritto nel paragrafo 9.7;
- *data*: in questo elemento sarà presente la risposta codificata in base64Binary (opzionale);
- *data di registrazione:* data in cui il messaggio è pervenuto al sistema di accoglienza.

L'operazione **authorizationTransferIOT** consente al gestore T.C. di trasferire uno o più IOT, corrispondenti ad aggregati di partite e può essere fruita impostando opportunamente il campo **ITATipoOperazioneType = Q**.

In modalità **authorizationTransferIOT**, l'input è rappresentato:

- da un progressivo richiesta autorizzazione;
- da un codice IOT (16 caratteri).

L'output è costituito dai seguenti elementi:

- estremi della richiesta effettuata;
- elenco di elementi differenziati per esito (positivo o negativo)

  Se negativo viene riportata una descrizione di errore, se positivo viene riportato un dettaglio della richiesta effettuata, gli estremi della partita oggetto del trasferimento, ed eventuale data di autorizzazione concessa dal funzionario.

4.6. Il servizio CNTINSTORAGE (Container in magazzino)

Il servizio "**CNTINSTORAGEService**", del tipo EJB – WS, mette a disposizione le seguenti operazioni per l'acquisizione di un messaggio XML relativo alla comunicazione dell'arrivo della merce, soggetta a trasferimento di magazzino o a corridoio controllato, nel magazzino di competenza:

- endTransfer (trasferimento magazzino);
- endCorridor (trasferimento corridoio).

Di seguito sono descritti i dati di interscambio ed i parametri di input/output per ogni operazione elencata.

4.6.1. Endpoint

In ambiente di prova l'endpoint con cui il servizio è esposto è:

https://interoptest.adm.gov.it/CNTINSTORAGEServiceWeb/services/CNTINSTORAGEService

4.6.2. Metodo Process

Il metodo process permette di comunicare l'arrivo della merce soggetta a trasferimento di magazzino o a corridoio controllato, nel magazzino di competenza. Ogni operazione è identificata mediante un serviceId.
Per ogni elaborazione effettuata viene indicata l'operazione che è stata innescata con i relativi dati di input (Richiesta) e di output (Risposta).

Il servizio del tipo EJB - WS ha la seguente operazione esposta:

- Risposta *process* (Richiesta input) e i seguenti parametri:

Metodo	Input	Output
process	Richiesta	Risposta

Tabella 9 - Descrizione metodo process

I dati in input relativi al tracciato "Richiesta" sono descritti in dettaglio nell'allegato tecnico.

Il tipo di dati in output "Risposta" descritto in dettaglio nell'allegato tecnico, contiene i seguenti elementi:

- *IUT*: identificativo univoco transazione;
- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione della richiesta, più propriamente descritto nel paragrafo 9.7;
- *data*: in questo elemento è presente la risposta codificata in base64Binary (opzionale);
- *data di registrazione:* data in cui il messaggio è pervenuto al sistema di accoglienza.

4.6.3. Le operazioni

Il servizio è utilizzato dal gestore di temporanea custodia per comunicare l'arrivo della merce, soggetta a trasferimento di magazzino o a corridoio controllato, nel magazzino di competenza.

4.6.3.1. Comunicazione Avvenuto Ingresso di un Container senza corridoi

L'operazione "**endTransfer**" (trasferimento magazzino) consente l'acquisizione di un messaggio xml relativo ad una comunicazione di avvenuto ingresso di merce interessata al trasferimento di magazzino.

Per invocare il servizio, viene creato un messaggio SOAP di fruizione che deve contenere nel tipo di dati in input "Richiesta" i seguenti campi obbligatori:

- serviceId
- data
  - xmlList
  - dichiarante

Segue la descrizione dei campi:

- *serviceId*: indica il tipo di operazione da eseguire. Necessario per effettuare il dispatching verso i servizi richiesti. Nel caso specifico della suddetta operazione va indicato: "**endTransfer**";

- *data*: rappresenta una collezione di oggetti contenenti:

  - xmlList: contiene il messaggio xml valido per l'XSD descritto in dettaglio nell'allegato tecnico (CNTINSTORAGE_INPUT.xsd) firmato digitalmente secondo le indicazioni espresse nel paragrafo 2.3 e codificato in base64binary;

  - *dichiarante*: l'identificazione dell'utente dichiarante (codice fiscale del soggetto per il quale si sta inviando il messaggio).

Una volta inviato il messaggio, il servizio restituirà in output un oggetto "Risposta" descritto in dettaglio nell'allegato tecnico che contiene i seguenti elementi:

- *IUT*: identificativo univoco transazione;
- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione più propriamente descritto nel paragrafo 9.7;

- *data*: in questo elemento è presente la risposta codificata in base64Binary

(opzionale);

- *data di registrazione:* data in cui il messaggio è pervenuto al sistema di accoglienza.

L'operazione **endTransfer** consente al gestore T.C. di comunicare l'elenco della merce interessata al trasferimento di magazzino e può essere fruita impostando opportunamente il campo **ITATipoOperazioneType = T**.

L'output è costituito dai seguenti elementi:

- esito dell'elaborazione (positivo/negativo), ed eventuale descrizione di errore;
- estremi della merce indicata in input.

4.6.3.2. Comunicazione Avvenuto Ingresso di un Container con i Corridoi

L'operazione "**endCorridor**" (trasferimento corridoio) consente l'acquisizione di un messaggio xml relativo ad una comunicazione dell'elenco dei container coinvolti in un corridoio controllato.

Per invocare il servizio, viene creato un messaggio SOAP di fruizione che deve contenere nel tipo di dati in input "Richiesta" i seguenti campi obbligatori:

- serviceId
- data
  - xmlList
  - dichiarante

Segue la descrizione dei campi:

- *serviceId*: indica il tipo di operazione da eseguire. Necessario per effettuare il dispatching verso i servizi richiesti. Nel caso specifico della suddetta operazione va indicato: "**endCorridor**";

- *data*: rappresenta una collezione di oggetti contenenti:

  - *xmlList*: contiene il messaggio xml valido per l'XSD descritto in dettaglio nell'allegato tecnico (CNTINSTORAGE_INPUT.xsd) firmato digitalmente secondo le indicazioni espresse nel paragrafo 2.3 e codificato in base64binary;

  - *dichiarante*: l'identificazione dell'utente dichiarante (codice fiscale del soggetto per il quale si sta inviando il messaggio).

Una volta inviato il messaggio, il servizio restituisce in output un oggetto "Risposta" descritto in dettaglio nell'allegato tecnico che contiene i seguenti elementi:

- *IUT*: identificativo univoco transazione;
- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione più propriamente descritto nel paragrafo 9.7;
- *data*: in questo elemento è presente la risposta codificata in base64Binary (opzionale);
- *data di registrazione:* data in cui il messaggio è pervenuto al sistema di accoglienza.

L'operazione **endCorridor** consente al gestore T.C. di comunicare l'elenco dei container coinvolti in un corridoio controllato e può essere fruita impostando opportunamente il campo **ITATipoOperazioneType = C**.

L'output è costituito dai seguenti elementi:

- esito dell'elaborazione (positivo/negativo), ed eventuale descrizione di errore;
- estremi del container, ufficio/magazzino di partenza, ufficio/magazzino di arrivo, estremi partita di destinazione.

4.7. Il servizio CHECKService (Interrogazione merci in ingresso e in uscita magazzino)

Il servizio "**CHECKService**", del tipo EJB – WS, mette a disposizione le seguenti operazioni relative al servizio di richiesta di consultazione della merce in arrivo o in partenza di competenza:

- incoming (interrogazione merci in ingresso);
- outgoing (interrogazione merci in uscita).

Di seguito sono descritti i dati di interscambio ed i parametri di input/output per ogni operazione elencata.

4.7.1. Endpoint

In ambiente di prova l'endpoint con cui il servizio è esposto è:

https://interoptest.adm.gov.it/CHECKServiceWeb/services/CHECKService

4.7.2. Metodo Process

Il metodo process permette l'elaborazione della richiesta di consultazione della merce in arrivo o in partenza di competenza.
Per ogni elaborazione effettuata viene indicata l'operazione che è stata innescata con i relativi dati di input (Richiesta) e di output (Risposta). Ogni operazione è identificata mediante un serviceId.

Il servizio del tipo EJB - WS, avrà la seguente operazione esposta:

- Risposta *process* (Richiesta input) e i seguenti parametri:

Metodo	Input	Output
process	Richiesta	Risposta

Tabella 10 - Descrizione metodo process

I dati in input relativi al tracciato "Richiesta" sono descritti in dettaglio nell'allegato tecnico.

Il tipo di dati in output "Risposta" descritto in dettaglio nell'allegato tecnico, contiene i seguenti elementi:

- *IUT*: identificativo univoco transazione;
- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione della richiesta, più propriamente descritto nel paragrafo 9.7;
- *data*: in questo elemento è presente la risposta codificata in base64Binary (opzionale);
- *data di registrazione:* data in cui il messaggio è pervenuto al sistema di accoglienza.

4.7.3. Le operazioni

Il servizio è utilizzato dal gestore di temporanea custodia per interrogare le merci in ingresso e in uscita nel/dal magazzino di competenza.

L'handler portuale/destinazione può utilizzare il servizio solo per interrogare le merci in ingresso nel nodo logistico di competenza.

4.7.3.1. Consultazione delle Informazioni sulla Merce in Ingresso

L'operazione "**incoming**" (interrogazione merci in ingresso) consente di effettuare una richiesta di consultazione delle informazioni sulla merce in ingresso di propria competenza.

Per invocare il servizio, viene creato un messaggio SOAP di fruizione che deve contenere nel tipo di dati in input “Richiesta” i seguenti campi obbligatori:

- serviceId
- data
  - xmlList
  - dichiarante

Segue la descrizione dei campi:

- *serviceId*: indica il tipo di operazione da eseguire. Necessario per effettuare il dispatching verso i servizi richiesti. Nel caso specifico della suddetta operazione va indicato: “**incoming**”;

- *data*: rappresenta una collezione di oggetti contenenti:

  - *xmlList*: contiene il messaggio xml valido per l’XSD descritto in dettaglio nell’allegato tecnico (CHECK_INPUT.xsd) firmato digitalmente secondo le indicazioni espresse nel paragrafo 2.3 e codificato in base64binary;

  - *dichiarante*: l’identificazione dell’utente dichiarante (codice fiscale del soggetto per il quale si sta inviando il messaggio).

Una volta inviato il messaggio, il servizio restituisce in output un oggetto “Risposta” descritto in dettaglio nell’allegato tecnico che contiene i seguenti elementi:

- *IUT*: identificativo univoco transazione;

- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione più propriamente descritto nel paragrafo 9.7;

- *data*: in questo elemento sarà presente la risposta codificata in base64Binary (opzionale);

- *data di registrazione:* data in cui il messaggio è pervenuto al sistema di accoglienza.

L’operazione **incoming** consente al gestore T.C. di interrogare le merci in ingresso e può essere fruita impostando opportunamente il campo **ITATipoOperazioneType = I**.

I dati di input del servizio sono:

- **Dati di testata,** contenente le generalità del fruitore del servizio**;**
- **ITATipoOperazioneType,** che indica la modalità in cui si intende consumare il servizio;

- **Sezione incoming:** consente di impostare i parametri per interrogare le merci in ingresso (tra i parametri si evidenziano la data di registrazione e di scadenza);

La risposta **incoming** contiene una prima sezione in cui sono riportati tutti i parametri di input indicati nella sezione incoming, e una seconda sezione che riporta l'esito negativo o positivo della ricerca. Solo nel caso di esito positivo il servizio restituisce la lista della merce in ingresso.

4.7.3.2. Consultazione delle Informazioni sulla Merce in Uscita

L'operazione "**outgoing**" (interrogazione merci in uscita) consente di effettuare una richiesta di consultazione delle informazioni sulla merce in uscita di propria competenza.

Per invocare il servizio, viene creato un messaggio SOAP di fruizione che deve contenere nel tipo di dati in input "Richiesta" i seguenti campi obbligatori:

- serviceId
- data
  - xmlList
  - dichiarante

Segue la descrizione dei campi:

- *serviceId*: indica il tipo di operazione da eseguire. Necessario per effettuare il dispatching verso i servizi richiesti. Nel caso specifico della suddetta operazione va indicato: "**outgoing**";

- data: rappresenta una collezione di oggetti contenenti:

  - *xmlList*: contiene il messaggio xml valido per l'XSD descritto in dettaglio nell'allegato tecnico (CHECK_INPUT.xsd) firmato digitalmente secondo le indicazioni espresse nel paragrafo 2.3 e codificato in base64binary;

  - *dichiarante*: l'identificazione dell'utente dichiarante (codice fiscale del soggetto per il quale si sta inviando il messaggio).

Una volta inviato il messaggio, il servizio restituisce in output un oggetto "Risposta" descritto in dettaglio nell'allegato tecnico che contiene i seguenti elementi:

- *IUT*: identificativo univoco transazione;

- *esito*: con il codice e la descrizione del messaggio che indica lo stato di elaborazione più propriamente descritto nel paragrafo 9.7;

- *data*: in questo elemento è presente la risposta codificata in base64Binary (opzionale);

- *data di registrazione: data in cui il messaggio è pervenuto al sistema di accoglienza.*

L'operazione **outgoing** consente al gestore T.C. di interrogare le merci in uscita e può essere fruita impostando opportunamente il campo **ITATipoOperazioneType = O**.

I dati di input del servizio sono:

- **Dati di testata,** contenente le generalità del fruitore del servizio**;**
- **ITATipoOperazioneType,** che indica la modalità in cui si intende consumare il servizio;
- **Sezione outgoing**: consente di impostare i parametri per interrogare le merci in uscita (tra i parametri si evidenzia la data di registrazione della partita).

La risposta **outgoing** contiene una prima sezione in cui sono riportati tutti i parametri indicati nella sezione outgoing, e una seconda sezione che riporta l'esito negativo o positivo della ricerca. Ovviamente, solo nel caso di esito positivo il servizio restituisce la lista della merce in uscita.

5. CODICI ERRORE/SEGNALAZIONE

I controlli effettuati dalle procedure di back-end del servizio possono restituire, all'interno dell'elemento data, uno o più codici di Errore o Segnalazione In caso di Codice Esito generale uguale a 198 – "Elaborazione KO: con esito" o 200 "Elaborazione OK: completata con esito finale".

La tabella contenente la descrizione dei codici di errore è pubblicata sul sito istituzionale dell'Agenzia delle Dogane e dei Monopoli.

6. SERVIZIO INTEROPRSERVICE - CONTROLLO DELLO STATO

Per favorire l'integrazione di sistema è disponibile un WebService REST che consente, dato uno IUT (identificativo univoco transazione), di controllare lo stato di accoglienza o di elaborazione relativo all'operazione per cui è stato generato quello specifico IUT.

Al paragrafo "9.8 Allegati tecnici" di questo documento sono riportate le informazioni riguardanti le api Open Api e Swagger, utili a generare i client.

È possibile generare in modo automatizzato un client in diversi linguaggi di programmazione attraverso i tools messi a disposizione dal sito online per mezzo della documentazione fornita in allegato al servizio e nel paragrafo:
"9.8. Documentazione Open Api del servizio REST InteropRService".

Un esempio di invocazione REST è la seguente:

Curl Request

```
curl -X GET --header 'Accept: application/json'
'https://interop.adm.gov.it/InteropRServiceWeb/servic
es/InteropRService/selezionaStato/20180426M4000000013'
```

Request URL

```
https://interop.adm.gov.it/InteropRServiceWeb/service
s/InteropRService/selezionaStato/20180426M4000000013
```

Response Body: 20

Response Code: 200

Response Headers

```
{
   "x-powered-by": "Servlet/3.0", "content-type": "application/json", "content-
   language": "it-IT", "transfer-encoding": "chunked",
   "date": "Fri, 07 Jul 2017 10:12:33 GMT"
}
```

In questo esempio è stato richiesto lo stato per lo iut: *20180426M4000000013*

La risposta in Response Code “200” indica che la chiamata è avvenuta con successo.

La risposta in Response Body “20” indica che lo stato della richiesta per lo IUT indicato ha codice “20”, che, come descritto nella tabella di decodifica, corrisponde alla descrizione: “Input Acquisito a sistema”.

Questo esempio di invocazione del servizio può essere valido anche come esempio in ambiente reale, basta cambiare l’endpoint nella “Request URL” come descritto nel paragrafo successivo.

6.1. Endpoint in Ambiente di Prova

In ambiente di prova il servizio viene esposto con il seguente endpoint:

https://interoptest.adm.gov.it/InteropRServiceWeb/services/InteropRService

Installando il certificato di autenticazione nel Browser è possibile consultare la documentazione on line agli indirizzi:

https://interoptest.adm.gov.it/InteropRServiceWeb/services/InteropRService/api/InteropRService.json

https://interoptest.adm.gov.it/InteropRServiceWeb/services/InteropRService/api/InteropRService.yaml

https://interoptest.adm.gov.it/InteropRServiceWeb/services/InteropRService/api

6.2. Endpoint In Ambiente Reale

In ambiente reale l'endpoint con cui il servizio viene esposto è:

https://interop.adm.gov.it/InteropRServiceWeb/services/InteropRService

Installando il certificato di autenticazione nel Browser è possibile consultare la documentazione on line agli indirizzi:

https://interop.adm.gov.it/InteropRServiceWeb/services/InteropRService/api/InteropRService.json

https://interop.adm.gov.it/InteropRServiceWeb/services/InteropRService/api/InteropRService.yaml

https://interop.adm.gov.it/InteropRServiceWeb/services/InteropRService/api

7. SERVIZIO INTEROPSERVICE - RECUPERO DELL'ESITO

Viene messo a disposizione un Web Service SOAP che permette di recuperare tramite lo IUT l'esito codificato in bytearray nel campo data nell'oggetto di Risposta, qualora sia previsto e prodotto dai servizi descritti nei paragrafi 3 e 4.
Il file di esito disponibile al recupero è sigillato elettronicamente (con firma digitale), secondo lo standard XAdES-BES enveloped con riferimento alle regole tecniche definite dalla DELIBERAZIONE N. 45 DEL 21 MAGGIO 2009, secondo il regolamento UE n° 910/2014 – eIDAS.

L'intestatario del certificato di firma usato nelle operazioni è l'Agenzia delle Dogane e dei Monopoli.

Il servizio del tipo EJB - WS, avrà la seguente operazione esposta: risposta recuperaEsito (String IUT) con i seguenti parametri:

Metodo	Input	Output
recuperaEsito	IUT	Risposta

Tabella 11 - Descrizione del metodo recuperaEsito

7.1. Endpoint

In ambiente di prova il servizio è esposto con il seguente endpoint:

https://interoptest.adm.gov.it/InteropServiceWEB/services/InteropService

In ambiente reale l'endpoint con cui il servizio è esposto è:

https://interop.adm.gov.it/InteropServiceWEB/services/InteropService

8. CODICI DI STATO O DI ERRORE DI RISPOSTA AI WEB SERVICES

Codice	Descrizione dello stato o dell'errore
0	Servizio non disponibile
1	La verifica della firma è fallita
2	Il certificato utilizzato per la firma non è valido
3	L'Autorità di certificazione non è ritenuta sicura
4	La verifica dell'integrità del messaggio è fallita
5	Messaggio non firmato
7	CA verifica certificato: fallita
9	Service ID non esistente
10	Verifica xsd: fallita
11	Errore in accodamento richiesta
12	Richiesta non ancora elaborata
13	Condizioni xsd violate
14	Utente non autorizzato
15	Dati di input non validi

Codice	Descrizione dello stato o dell'errore
16	Certificato autenticazione non valido
18	Firmatario non autorizzato
20	Acquisito a sistema
50	In elaborazione
51	In elaborazione: controllo sostanziale superato
197	Elaborazione KO: senza esito
198	Elaborazione KO: con esito
199	Elaborazione OK: completata senza esito finale
200	Elaborazione OK: completata con esito finale

Tabella 12 - Codici di stato o di errore dei Web Services

9. ALLEGATI TECNICI

9.1. Documentazione Tracciati Dati di Input EManifestService

Viene riportata di seguito la struttura dello schema **definitorio.xsd** del servizio EManifestService

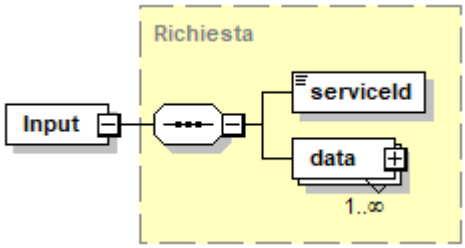
schema location: **definitorio.xsd**

attributeFormDefault:

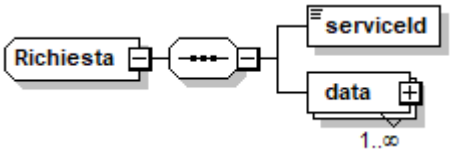
elementFormDefault: **qualified**
targetNamespace: **http://emanifest.domest.sogei.it**

Elements: Input
Complex types: Richiesta

element **Input**

Diagram	
Namespace	http://emanifest.domest.sogei.it
Type	Richiesta
Properties	content complex
Children	serviceId data
Source	<xs:element name="Input" type="Richiesta"/>

complexType **Richiesta**

diagram	
namespace	http://emanifest.domest.sogei.it

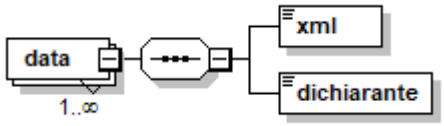
children	serviceId data
used by	element Input
source	<xs:complexType name="Richiesta"> <xs:sequence> <xs:element name="serviceId"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:enumeration value="submission"/> <xs:enumeration value="amendment"/> <xs:enumeration value="invalidation"/> <xs:enumeration value="updating"/> <xs:enumeration value="validation"/> <xs:enumeration value="summaryResult"/> </xs:restriction> </xs:simpleType> </xs:element> <xs:element name="data" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element name="xml" type="xs:base64Binary"/> <xs:element name="dichiarante"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:pattern value="([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}|[0-9]{11})"/> </xs:restriction> </xs:simpleType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType>

element **Richiesta/serviceId**

diagram	
namespace	http://emanifest.domest.sogei.it
type	restriction of **xs:string**
properties	content simple
facets	Kind / Value / Annotation enumeration submission enumeration amendment enumeration invalidation enumeration updating enumeration validation enumeration summaryResult

source	<xs:element name="serviceId"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:enumeration value="submission"/> <xs:enumeration value="amendment"/> <xs:enumeration value="invalidation"/> <xs:enumeration value="updating"/> <xs:enumeration value="validation"/> <xs:enumeration value="summaryResult"/> </xs:restriction> </xs:simpleType> </xs:element>

element **Richiesta/data**

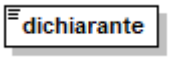
diagram	
namespace	http://emanifest.domest.sogei.it
properties	minOcc 1 maxOcc unbounded content complex
children	xml dichiarante
source	<xs:element name="data" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element name="xml" type="xs:base64Binary"/> <xs:element name="dichiarante"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:pattern value="([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}|[0-9]{11})"/> </xs:restriction> </xs:simpleType> </xs:element> </xs:sequence> </xs:complexType> </xs:element>

element **Richiesta/data/xml**

diagram	
namespace	http://emanifest.domest.sogei.it
type	**xs:base64Binary**
properties	content simple

source	<xs:element name="xml" type="xs:base64Binary"/>

element **Richiesta/data/dichiarante**

diagram	
namespace	http://emanifest.domest.sogei.it
type	restriction of **xs:string**
properties	content simple
facets	Kind: pattern Value: ([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}|[0-9]{11}) Annotation:
source	<xs:element name="dichiarante"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:pattern value="([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}|[0-9]{11})"/> </xs:restriction> </xs:simpleType> </xs:element>

9.2. Documentazione Tracciati Dati di Input servizio TEMPORARYSTORAGEService

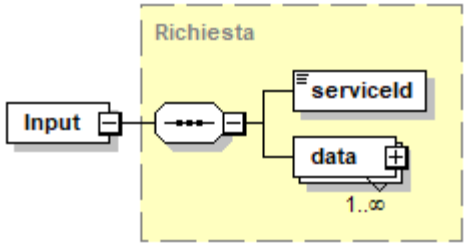
Viene riportata di seguito la struttura dello schema **definitorio.xsd** del servizio TEMPORARYSTORAGEService

schema location: **definitorio.xsd**

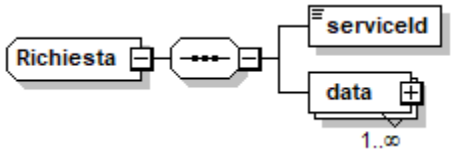
attributeFormDefault:
elementFormDefault: **qualified**
targetNamespace: **http://temporarystorageservice.domest.sogei.it**

Elements	Complex types
Input	Richiesta

element **Input**

diagram	
namespace	http://temporarystorageservice.domest.sogei.it
type	Richiesta
properties	content complex
children	serviceId data
source	<xs:element name="Input" type="Richiesta"/>

complexType **Richiesta**

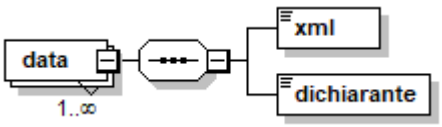
diagram	
namespace	http://temporarystorageservice.domest.sogei.it
children	serviceId data
used by	element Input
source	<xs:complexType name="Richiesta"> <xs:sequence> <xs:element name="serviceId"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:enumeration value="submission"/> <xs:enumeration value="amendment"/> </xs:restriction> </xs:simpleType> </xs:element> <xs:element name="data" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element name="xml" type="xs:base64Binary"/> <xs:element name="dichiarante"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:pattern value="([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}|[0-9]{11})"/> </xs:restriction> </xs:simpleType>

```
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
 </xs:sequence>
</xs:complexType>
```

element **Richiesta/serviceId**

diagram	
namespace	http://temporarystorageservice.domest.sogei.it
type	restriction of **xs:string**
properties	content simple
facets	Kind / Value / Annotation enumeration submission enumeration amendment
source	<xs:element name="serviceId"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:enumeration value="submission"/> <xs:enumeration value="amendment"/> </xs:restriction> </xs:simpleType> </xs:element>

element **Richiesta/data**

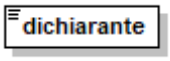
diagram	
namespace	http://temporarystorageservice.domest.sogei.it
properties	minOcc 1 maxOcc unbounded content complex
children	xml dichiarante
source	<xs:element name="data" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element name="xml" type="xs:base64Binary"/> <xs:element name="dichiarante"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:pattern value="([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}|[0-9]{11})"/> </xs:restriction>

```
        </xs:simpleType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

element **Richiesta/data/xml**

diagram	
namespace	http://temporarystorageservice.domest.sogei.it
type	**xs:base64Binary**
properties	content simple
source	<xs:element name="xml" type="xs:base64Binary"/>

element **Richiesta/data/dichiarante**

diagram	
namespace	http://temporarystorageservice.domest.sogei.it
type	restriction of **xs:string**
properties	content simple
facets	Kind: pattern — Value: ([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}|[0-9]{11}) — Annotation:
source	<xs:element name="dichiarante"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:pattern value="([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}|[0-9]{11})"/> </xs:restriction> </xs:simpleType> </xs:element>

9.3. Documentazione Tracciati Dati di Input DISCREPANCYService

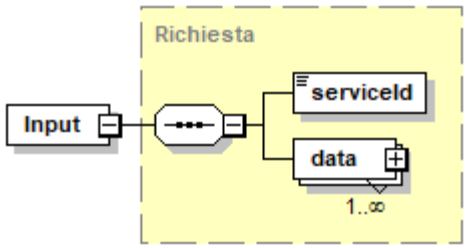
Viene riportata di seguito la struttura dello schema **definitorio.xsd** del servizio DISCREPANCYService

schema location: **definitorio.xsd**
attributeFormDefault:

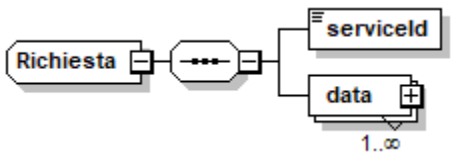
elementFormDefault: **qualified**
targetNamespace: **http://discrepancyservice.domest.sogei.it**

Elements	Complex types
Input	Richiesta

element **Input**

diagram	
namespace	http://discrepancyservice.domest.sogei.it
type	Richiesta
properties	content complex
children	serviceId data
source	<xs:element name="Input" type="Richiesta"/>

complexType **Richiesta**

diagram	
namespace	http://discrepancyservice.domest.sogei.it
children	serviceId data
used by	element Input
source	<xs:complexType name="Richiesta"> <xs:sequence> <xs:element name="serviceId"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:enumeration value="submission"/> </xs:restriction> </xs:simpleType> </xs:element> <xs:element name="data" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element name="xml" type="xs:base64Binary"/>

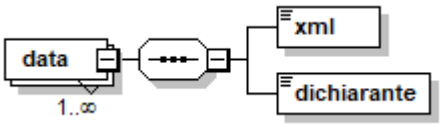
```
        <xs:element name="dichiarante">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:pattern value="([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}|[0-9]{11})"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
 </xs:sequence>
</xs:complexType>
```

element **Richiesta/serviceId**

diagram	
namespace	http://discrepancyservice.domest.sogei.it
type	restriction of **xs:string**
properties	content simple
facets	Kind: enumeration — Value: submission — Annotation

source	`<xs:element name="serviceId">`
	`<xs:simpleType>`
	`<xs:restriction base="xs:string">`
	`<xs:enumeration value="submission"/>`
	`</xs:restriction>`
	`</xs:simpleType>`
	`</xs:element>`

element **Richiesta/data**

diagram	
namespace	http://discrepancyservice.domest.sogei.it
properties	minOcc 1 maxOcc unbounded content complex
children	xml dichiarante

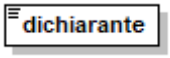
source	`<xs:element name="data" maxOccurs="unbounded">`
	`<xs:complexType>`
	`<xs:sequence>`
	`<xs:element name="xml" type="xs:base64Binary"/>`
	`<xs:element name="dichiarante">`

```
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:pattern value="([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-
9]{3}[A-Z]{1}|[0-9]{11})"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

element **Richiesta/data/xml**

diagram	
namespace	http://discrepancyservice.domest.sogei.it
type	**xs:base64Binary**
properties	content simple
source	<xs:element name="xml" type="xs:base64Binary"/>

element **Richiesta/data/dichiarante**

diagram	
namespace	http://discrepancyservice.domest.sogei.it
type	restriction of **xs:string**
properties	content simple
facets	Kind: pattern — Value: ([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}|[0-9]{11}) — Annotation
source	<xs:element name="dichiarante"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:pattern value="([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}|[0-9]{11})"/> </xs:restriction> </xs:simpleType> </xs:element>

9.4. Documentazione Tracciati Dati di Input STORAGETRANSFERService

Viene riportata di seguito la struttura dello schema **definitorio.xsd** del servizio

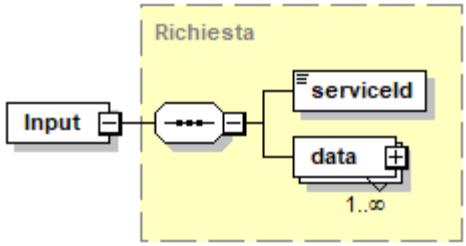
STORAGETRANSFERService

schema location: **definitorio.xsd**
attributeFormDefault:

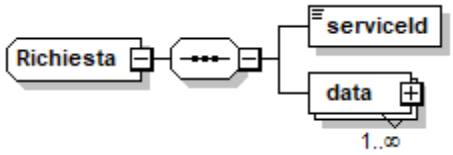
elementFormDefault: **qualified**
targetNamespace: **http://storagetransferservice.domest.sogei.it**

Elements	Complex types
Input	Richiesta

element **Input**

diagram	
namespace	http://storagetransferservice.domest.sogei.it
type	Richiesta
properties	content complex
children	serviceId data
source	<xs:element name="Input" type="Richiesta"/>

complexType **Richiesta**

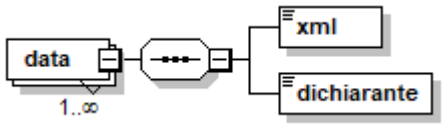
diagram	
namespace	http://storagetransferservice.domest.sogei.it
children	serviceId data
used by	element Input
source	<xs:complexType name="Richiesta"> <xs:sequence> <xs:element name="serviceId"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:enumeration value="authorizationRequest"/>

```
        <xs:enumeration value="authorizationsStatus"/>
        <xs:enumeration value="authorizationTransferIOT"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <xs:element name="data" maxOccurs="unbounded">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="xml" type="xs:base64Binary"/>
        <xs:element name="dichiarante">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:pattern value="([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-
Z]{1}[0-9]{3}[A-Z]{1}|[0-9]{11})"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
 </xs:sequence>
</xs:complexType>
```

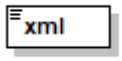
element **Richiesta/serviceId**

diagram	
namespace	http://storagetransferservice.domest.sogei.it
type	restriction of **xs:string**
properties	content simple
facets	Kind / Value / Annotation enumeration authorizationRequest enumeration authorizationsStatus enumeration authorizationTransferIOT
source	<xs:element name="serviceId"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:enumeration value="authorizationRequest"/> <xs:enumeration value="authorizationsStatus"/> <xs:enumeration value="authorizationTransferIOT"/> </xs:restriction> </xs:simpleType> </xs:element>

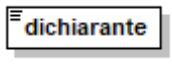
element **Richiesta/data**

diagram	
namespace	http://storagetransferservice.domest.sogei.it
properties	minOcc 1 maxOcc unbounded content complex
children	xml dichiarante
source	<xs:element name="data" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element name="xml" type="xs:base64Binary"/> <xs:element name="dichiarante"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:pattern value="([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}|[0-9]{11})"/> </xs:restriction> </xs:simpleType> </xs:element> </xs:sequence> </xs:complexType> </xs:element>

element **Richiesta/data/xml**

diagram	
namespace	http://storagetransferservice.domest.sogei.it
type	**xs:base64Binary**
properties	content simple
source	<xs:element name="xml" type="xs:base64Binary"/>

lement **Richiesta/data/dichiarante**

diagram	
namespace	http://storagetransferservice.domest.sogei.it
type	restriction of **xs:string**
properties	content simple
facets	Kind: pattern Value: ([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}|[0-9]{11}) Annotation

source	<xs:element name="dichiarante"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:pattern value="([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}|[0-9]{11})"/> </xs:restriction> </xs:simpleType> </xs:element>

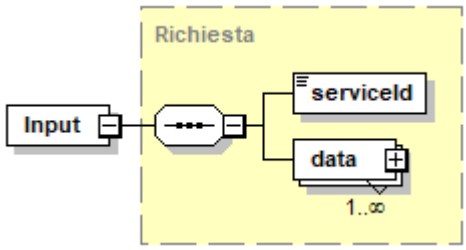
9.5. Documentazione Tracciati Dati di Input CNTINSTORAGEService

Viene riportata di seguito la struttura dello schema **definitorio.xsd** del servizio CNTINSTORAGEService

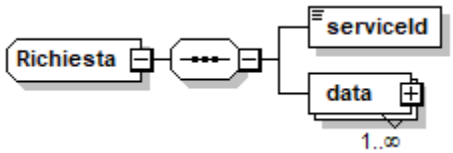
schema location: **definitorio.xsd**
attributeFormDefault:
elementFormDefault: **qualified**
targetNamespace: **http://cntinstorageservice.domest.sogei.it**

Elements: Input
Complex types: Richiesta

element **Input**

diagram	
namespace	http://cntinstorageservice.domest.sogei.it
type	Richiesta
properties	content complex
children	serviceId data
source	<xs:element name="Input" type="Richiesta"/>

complexType **Richiesta**

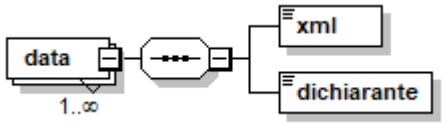
diagram	
namespace	http://cntinstorageservice.domest.sogei.it
children	serviceId data
used by	element Input
source	<xs:complexType name="Richiesta"> <xs:sequence> <xs:element name="serviceId"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:enumeration value="endTransfer"/> <xs:enumeration value="endCorridor"/> </xs:restriction> </xs:simpleType> </xs:element> <xs:element name="data" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element name="xml" type="xs:base64Binary"/> <xs:element name="dichiarante"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:pattern value="([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}|[0-9]{11})"/> </xs:restriction> </xs:simpleType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType>

element **Richiesta/serviceId**

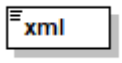
diagram	
namespace	http://cntinstorageservice.domest.sogei.it
type	restriction of **xs:string**
properties	content simple
facets	Kind / Value / Annotation enumeration endTransfer enumeration endCorridor

source	<xs:element name="serviceId"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:enumeration value="endTransfer"/> <xs:enumeration value="endCorridor"/> </xs:restriction> </xs:simpleType> </xs:element>

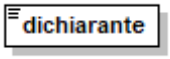
element **Richiesta/data**

diagram	
namespace	http://cntinstorageservice.domest.sogei.it
properties	minOcc 1 maxOcc unbounded content complex
children	xml dichiarante
source	<xs:element name="data" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element name="xml" type="xs:base64Binary"/> <xs:element name="dichiarante"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:pattern value="([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}|[0-9]{11})"/> </xs:restriction> </xs:simpleType> </xs:element> </xs:sequence> </xs:complexType> </xs:element>

element **Richiesta/data/xml**

diagram	
namespace	http://cntinstorageservice.domest.sogei.it
type	**xs:base64Binary**
properties	content simple
source	<xs:element name="xml" type="xs:base64Binary"/>

element **Richiesta/data/dichiarante**

diagram	
namespace	http://cntinstorageservice.domest.sogei.it
type	restriction of **xs:string**
properties	content simple
facets	Kind: pattern; Value: ([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}[0-9]{11}); Annotation
source	<xs:element name="dichiarante"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:pattern value="([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}[0-9]{11})"/> </xs:restriction> </xs:simpleType> </xs:element>

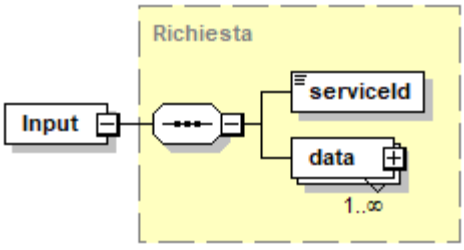
9.6. Documentazione Tracciati Dati di Input CHECKService

Viene riportata di seguito la struttura dello schema **definitorio.xsd** del servizio CHECKService

schema location: **definitorio.xsd**
attributeFormDefault:
elementFormDefault: **qualified**
targetNamespace: **http://checkservice.domest.sogei.it**

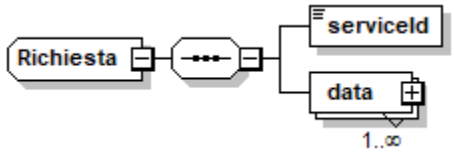
Elements	Complex types
Input	Richiesta

element **Input**

diagram	
namespace	http://checkservice.domest.sogei.it
type	Richiesta
properties	content complex
children	serviceId data

source	<xs:element name="Input" type="Richiesta"/>

complexType **Richiesta**

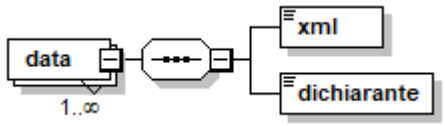
diagram	
namespace	http://checkservice.domest.sogei.it
children	serviceId data
used by	element Input
source	<xs:complexType name="Richiesta"> <xs:sequence> <xs:element name="serviceId"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:enumeration value="incoming"/> <xs:enumeration value="outgoing"/> </xs:restriction> </xs:simpleType> </xs:element> <xs:element name="data" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element name="xml" type="xs:base64Binary"/> <xs:element name="dichiarante"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:pattern value="([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}|[0-9]{11})"/> </xs:restriction> </xs:simpleType> </xs:element> </xs:sequence> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType>

element **Richiesta/serviceId**

diagram	
namespace	http://checkservice.domest.sogei.it
type	restriction of **xs:string**

properties	content simple
facets	Kind / Value / Annotation enumeration incoming enumeration outgoing
source	<xs:element name="serviceId"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:enumeration value="incoming"/> <xs:enumeration value="outgoing"/> </xs:restriction> </xs:simpleType> </xs:element>

element **Richiesta/data**

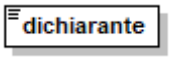
diagram	
namespace	http://checkservice.domest.sogei.it
properties	minOcc 1 maxOcc unbounded content complex
children	xml dichiarante
source	<xs:element name="data" maxOccurs="unbounded"> <xs:complexType> <xs:sequence> <xs:element name="xml" type="xs:base64Binary"/> <xs:element name="dichiarante"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:pattern value="([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}|[0-9]{11})"/> </xs:restriction> </xs:simpleType> </xs:element> </xs:sequence> </xs:complexType> </xs:element>

element **Richiesta/data/xml**

diagram	
namespace	http://checkservice.domest.sogei.it

type	**xs:base64Binary**
properties	content simple
source	<xs:element name="xml" type="xs:base64Binary"/>

element **Richiesta/data/dichiarante**

diagram	
namespace	http://checkservice.domest.sogei.it
type	restriction of **xs:string**
properties	content simple
facets	Kind: pattern — Value: ([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}[0-9]{11}) — Annotation
source	<xs:element name="dichiarante"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:pattern value="([A-Z]{6}[0-9]{2}[A-Z]{1}[0-9]{2}[A-Z]{1}[0-9]{3}[A-Z]{1}[0-9]{11})"/> </xs:restriction> </xs:simpleType> </xs:element>

9.7. Documentazione Tracciati Dati di Esito

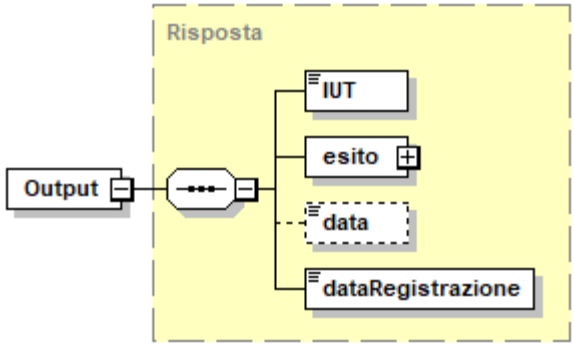
Viene riportata di seguito la struttura dello schema **esitoServizi.xsd**.
Tale schema è univoco per tutti i servizi citati in questo documento.

Schema **esitoServizi.xsd**

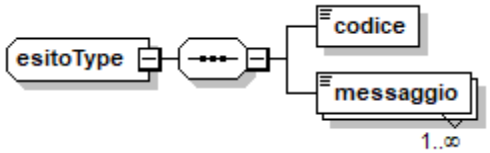
schema location: esitoServizi.xsd
attributeFormDefault:
elementFormDefault: **qualified**
targetNamespace: **http://ws.sogei.it/output/**

Elements	Complex types
Output	esitoType
	Risposta

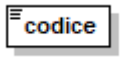
element **Output**

diagram	
namespace	http://ws.sogei.it/output/
type	Risposta
properties	content complex
children	IUT esito data dataRegistrazione
source	<xs:element name="Output" type="Risposta"/>

complexType **esitoType**

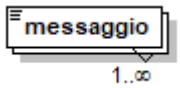
diagram	
namespace	http://ws.sogei.it/output/
children	codice messaggio
used by	element Risposta/esito
source	<xs:complexType name="esitoType"> <xs:sequence> <xs:element name="codice" type="xs:string"/> <xs:element name="messaggio" type="xs:string" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType>

element **esitoType/codice**

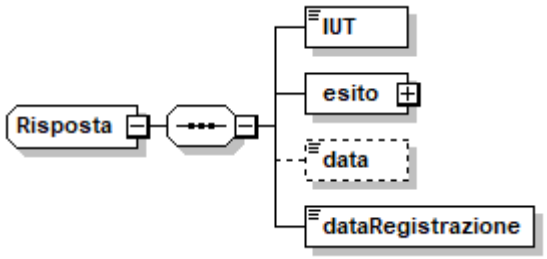
diagram	
namespace	http://ws.sogei.it/output/
type	**xs:string**

properties	content simple
source	<xs:element name="codice" type="xs:string"/>

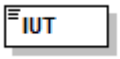
element **esitoType/messaggio**

diagram	
namespace	http://ws.sogei.it/output/
type	**xs:string**
properties	minOcc 1 maxOcc unbounded content simple
source	<xs:element name="messaggio" type="xs:string" maxOccurs="unbounded"/>

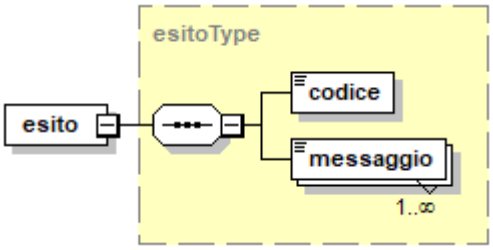
complexType **Risposta**

diagram	
namespace	http://ws.sogei.it/output/
children	IUT esito data dataRegistrazione
used by	element Output
source	<xs:complexType name="Risposta"> <xs:sequence> <xs:element name="IUT"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:maxLength value="20"/> </xs:restriction> </xs:simpleType> </xs:element> <xs:element name="esito" type="esitoType"/> <xs:element name="data" type="xs:base64Binary" minOccurs="0"/> <xs:element name="dataRegistrazione" type="xs:date"/> </xs:sequence> </xs:complexType>

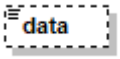
element **Risposta/IUT**

diagram	
namespace	http://ws.sogei.it/output/
type	restriction of **xs:string**
properties	content simple
facets	Kind Value Annotation maxLength 20
source	<xs:element name="IUT"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:maxLength value="20"/> </xs:restriction> </xs:simpleType> </xs:element>

element **Risposta/esito**

diagram	
namespace	http://ws.sogei.it/output/
type	esitoType
properties	content complex
children	codice messaggio
source	<xs:element name="esito" type="esitoType"/>

element **Risposta/data**

diagram	
namespace	http://ws.sogei.it/output/
type	**xs:base64Binary**
properties	minOcc 0 maxOcc 1 content simple
source	<xs:element name="data" type="xs:base64Binary" minOccurs="0"/>

element **Risposta/dataRegistrazione**

diagram	
namespace	http://ws.sogei.it/output/
type	**xs:date**
properties	content simple
source	<xs:element name="dataRegistrazione" type="xs:date"/>

9.8. Documentazione Open Api Del Servizio Rest InteropRService

Si riportano di seguito le informazioni utili per la generazione di un client che permetta l'invocazione del Web Service REST per il controllo dello stato:

- ***INFORMAZIONI SULLA VERSIONE***

  Versione: 1.0.2

- ***SCHEMA URI***

  BasePath: /InteropRServiceWeb/services

- ***TAGS***

  InteropRService

- ***OPERAZIONI***

  selezionaStato

SELEZIONA STATO

- Method: GET
- Endpoint (prova): **https://interoptest.adm.gov.it**
- Endpoint (reale): **https://interop.adm.gov.it**
- Resource:

  /InteropRServiceWeb/services/InteropRService/selezionaStato/{iut}

Descrizione

Il servizio restituisce lo stato di accoglienza o di elaborazione relativo all'operazione

per cui è stato generato uno specifico IUT.

Parametri

Tipo	Nome	Descrizione	Schema
Path	**iut** Obbligatorio	IUT di cui si vuole recuperare lo stato	string

Tabella 13 - Parametri chiamata REST

Risposte

Codice HTTP	Descrizione	Schema
200	Il codice indicante lo stato	Nessun commento
403	Accesso negato	Nessun commento
404	Nessuno stato trovato relativo al codice IUT in input	Nessun commento
406	Dati input errati	Nessun commento
500	Errore interno	Nessun commento

Tabella 14 - Codici HTTP di risposta alla chiamata REST

- **Esempio di richiesta http**

https://interoptest.adm.gov.it/InteropRServiceWeb/services/InteropRService/selezionaStato/20180426M4000000013