

PGED-ACTA-STE011-ACARIS Modulo “Servizi Applicativi”

ACTA Archive Interoperability Services

Specifica dei Object Services

Versione <9.0>

VERIFICHE E APPROVAZIONI

VERSIONE	REDAZIONE		CONTROLLO APPROVAZIONE		AUTORIZZAZIONE EMISSIONE	
	NOME	DATA	NOME	DATA	NOME	DATA
1.0	G.PANETTA	16/09/2009	PEIRA	16/09/2009	GOLA	16/09/2009
2.0	GARBOLINO	29/09/2009	PEIRA	29/09/2009	GOLA	29/09/2009
3.0	GARBOLINO	16/12/2009	PEIRA	16/12/2009	GOLA	16/12/2009

4.0	GARBOLINO	17/02/2010	PEIRA	17/02/2010	GOLA	17/02/2010
5.0	MARANDO GARBOLINO				CERONI	04/11/10
6.0	MARANDO GARBOLINO				CERONI	16/11/10
7	MARANDO	01/02/11			CERONI	01/02/11
8.0	MARANDO	03/12/2012			CERONI	04/12/2012
9	MARANDO	24/01/13			MARANDO	24/01/13

STATO DELLE VARIAZIONI

VERSIONE	PARAGRAFO O PAGINA	DESCRIZIONE DELLA VARIAZIONE
V01	Tutto il documento	Versione iniziale del documento
V02	Par 2.5	Aggiunti Codici errore per Crea Sottofascicolo
V03	Par 2.5	Aggiunto Codice errore per Controllo esistenza Folder Parent SER-E130
V04	Par 2.5 Tutto il documento	Eliminato paragrafo Codici di errore / messaggi restituiti Modificati i riferimenti dei codici errore al nuovo documento Glossario messaggi PGED-ACTA-STE17-ACARISv01_Glossario Messaggi.xls
V05	Par 2.4	Aggiunta descrizione dei servizi query, updateProperties, getContentStream, setContentStream, addRenditionStream, getProperties
V06	Par 2.4.5 Operazione: updateProperties	Aggiunta descrizione dei controlli da eseguire per non fare l'update delle properties di cabio stato
V07	Operazione getContentStream	Modifica effettuata in risposta a issue n. 36 Completata enum relativa alle tipologie di content stream. Specificato in modo più approfondito il funzionamento del servizio Modificata la cardinalità dell'output
V08	Operazione moveDocument	Creato relativo servizio
V09	Operazione moveDocument	Aggiunta gestione verifica smistamento Aggiunta possibilità di utilizzare l'operazione per allegare il documento da spostare ad altro documento esistente Introdotta la verifica che nel contenitore target non esista già una classificazione per il documento da spostare

INDICE

1. Introduzione	6
1.1 Scopo del documento.....	6
1.2 Riferimenti.....	6
2. Servizio Object Service	7
2.1 Obiettivi	7
2.2 Modalità di richiamo	7
2.3 Riferimento Oggetti Domanin Model ACARIS.....	7
2.4 Descrizione dell'interfaccia del servizio	7
2.4.1 Operazione: createDocument	7
2.4.1.1 Descrizione dell'operazione.....	8
2.4.1.2 Interfaccia di richiamo (input)	8
2.4.1.3 Interfaccia di output	8
2.4.1.4 Logica di Business	9
2.4.2 Operazione: createAssociativeDocument	9
2.4.2.1 Descrizione dell'operazione.....	9
2.4.2.2 Interfaccia di richiamo (input)	9
2.4.2.3 Interfaccia di output	10
2.4.2.4 Logica di Business	10
2.4.3 Operazione: createFolder	11
2.4.3.1 Descrizione dell'operazione.....	11
2.4.3.2 Interfaccia di richiamo (input)	11
2.4.3.3 Interfaccia di output	12
2.4.3.4 Logica di Business	12
2.4.4 Operazione: createRelationship	13
2.4.4.1 Descrizione dell'operazione.....	13
2.4.4.2 Interfaccia di richiamo (input)	13
2.4.4.3 Interfaccia di output	14
2.4.4.4 Logica di Business	14
2.4.5 Operazione: updateProperties	14
2.4.5.1 Descrizione dell'operazione.....	14
2.4.5.2 Interfaccia di richiamo (input)	15
2.4.5.3 Interfaccia di output	15
2.4.5.4 Logiche di Business	15
2.4.6 Operazione query	17
2.4.6.1 Descrizione dell'operazione.....	17
2.4.6.2 Interfaccia di richiamo (input)	17
2.4.6.3 Interfaccia di output	18
2.4.6.4 Logiche di Business	18
2.4.7 Operazione getContentStream	18
2.4.7.1 Descrizione dell'operazione.....	18
2.4.7.2 Interfaccia di richiamo (input)	18

2.4.7.3 Interfaccia di output	19
2.4.7.4 Logiche di Business	19
2.4.8 Operazione setContentStream	19
2.4.8.1 Descrizione dell'operazione.....	19
2.4.8.2 Interfaccia di richiamo (input)	20
2.4.8.3 Interfaccia di output	20
2.4.8.4 Logiche di Business	20
2.4.9 Operazione addRenditionStream	20
2.4.9.1 Descrizione dell'operazione.....	20
2.4.9.2 Interfaccia di richiamo (input)	20
2.4.9.3 Interfaccia di output	21
2.4.9.4 Logiche di Business	21
2.4.10 Operazione getProperties.....	21
2.4.10.1 Descrizione dell'operazione.....	21
2.4.10.2 Interfaccia di richiamo (input)	22
2.4.10.3 Interfaccia di output	22
2.4.10.4 Logiche di Business	22
2.4.11 Operazione moveDocument.....	23
2.4.11.1 Descrizione dell'operazione.....	23
2.4.11.2 Interfaccia di richiamo (input)	23
2.4.11.3 Interfaccia di output	23
2.4.11.4 Logiche di Business	23
2.5 Governance	25
2.5.1 Requisiti di sicurezza	25

1. Introduzione

1.1 Scopo del documento

Questo documento specifica tutte le operazioni possibili sugli oggetti del repository ACTA.

1.2 Riferimenti

- [A1] PGED-ACTA-SRS-v14_ Requisiti funzionali.doc
- [A2] PGED-ACTA-SRS-ACARIS-V01.doc
- [A3] PGED-ACTA-STE00-ACARIS-V01-DomainModel.doc
- [A4] PGED-ACTA-UC-ACARISDOC01-V01_Crea_DocumentoArchivistico.doc
- [A5]
- [A6]
- [A7] PGED-ACTA-UC-ACARISREL01-V01_Crea_DocumentComposition.doc
- [A8] PGED-ACTA-UC-ACARISFOLD02-V01_Crea_DocFisico.doc
- [A9] PGED-ACTA-UC-ACARISFOLD003-V01_Crea_fascicolo.doc
- [A10] PGED-ACTA-Architettura_V02.doc
- [A11] PGED-ACTA-UC-ACARISASS01-V01_Crea_Documento_Classificato.doc
- [A12] PGED-ACTA-UC-ACARISASS02-V01_Crea_Documento_Elettronico.doc
- [A13] PGED-ACTA-STE17-ACARISv01_Glossario Messaggi.xls

2. Servizio *Object Service*

2.1 Obiettivi

L'insieme delle operazioni possibili nell'ambito degli Object Services permettono la creazione di oggetti di tipo **Documento**, **Fascicolo** e **Relationship** in un repository.

2.2 Modalità di richiamo

Per quanto riguarda il richiamo dei servizi avviene in modalità sincrona.

Essi sono esposti tramite il servizio di infrastruttura CSI (PA/PD) e Web Services (SOAP), si rimanda al documento di architettura [A10].

2.3 Riferimento Oggetti Domanin Model ACARIS

Nel seguito i riferimenti agli oggetti del domain data model degli ACTA Services, come descritto in [A3], sono effettuati per mezzo dei nomi definiti negli schema di riferimento ed in particolare:

- `complexType`

```
<xs:complexType name="ObjectTypeId">  
  <xs:complexContent>  
    <xs:sequence>  
      <xs:element name="PropertyId" type="PropertyType"/>  
      ...  
    </xs:sequence>  
  </xs:complexContent>  
</xs:complexType>
```
- `simpleType`

```
<xs:simpleType name="PropertyType">  
  ...  
</xs:simpleType>
```

Il nome utilizzato è quello specificato per l'attributo `name` dei tipi complessi o semplici definiti negli schema ACARIS-Core.xsd o ACARIS-Messaging.xsd. Può trattarsi a seconda dei casi del nome della classe dell'oggetto del modello (attributo `ObjectTypeId`) oppure il nome della property (attributo `PropertyId`) oppure il tipo della property (attributo `PropertyType`).

2.4 Descrizione dell'interfaccia del servizio

2.4.1 Operazione: *createDocument*

2.4.1.1 Descrizione dell'operazione

Crea un oggetto documento del tipo specificato e aggiunge il documento in un folder.

Se è specificato un Folder ed il tipo di Documento non è fra quelli permessi per questo Folder viene sollevata una *AcarisException* con messaggio SER-E902 [A13] se l' *objectType* non accetta un *contentStream* viene sollevata una *AcarisException* con messaggio SER-E903 [A13].

2.4.1.2 Interfaccia di richiamo (input)

- ✓ ID repositoryId: Repository Id
 - ✓ ID typeId: Tipo Documento
 - ✓ ID principal: Identificativo dell' attore che esegue l'operazione
 - ✓ Collection properties : Elenco delle properties che caratterizzano il documento.
 - ✓ ID folderId: Folder padre per il documento che si vuole creare
 - ✓ ContentStream contentStream
-
- **typeId**
I tipi di oggetto ammessi sono:
 - ✓ *ContenutoFisicoPropertiesType*
 - ✓ *ClipsMetallicaPropertiesType*
 - ✓ *GruppoAllegatiPropertiesType*
 - **principal**
Identifica univocamente l'attore che vuole operare sul repository, questo deve essere presente su ACTA ed autorizzato ad operare sui contenuti, viene verificata l'autorizzazione alla creazione dello specifico oggetto Document richiesto (abilitazione al corrispettivo CdU in ACTA).
 - **collection properties**
Per le properties fare riferimento al Data Model rif. [A3] al capitolo Document Object Type.
 - **folderId**
Identifica l'oggetto in cui viene creato il nuovo documento, deve essere verificato che l'oggetto che si vuole creare possa essere inserito nel Parent folder (documento di *objectConstraints*)
 - **contentStream**
E' il file fisico (nel caso di inserimento di un documento elettronico) è solo previsto per un l'oggetto di tipo *ContenutoFisicoPropertiesType*

2.4.1.3 Interfaccia di output

- ✓ ID objectId: Identificativo del document creato

2.4.1.4 Logica di Business

Nella tabella successiva sono indicati i casi d'uso che descrivono i controlli che il sistema deve effettuare a livello di 'logica di business' dipendentemente dal **typeId** specificato nella chiamata del servizio.

TypeId	Caso d'uso
ContenutoFisicoPropertiesType	Il sistema esegue i controlli indicati nel caso d'uso Crea_ContentutoFisico rif.[A6]
ClipsMetallicaPropertiesType	Da implementare
GruppoAllegatiPropertiesType	Da implementare

2.4.2 Operazione: *createAssociativeDocument*

2.4.2.1 Descrizione dell'operazione

Crea un oggetto documento del tipo specificato e aggiunge il documento in un folder creando anche l'oggetto che permette l'associazione tra il documento ed il folder specificato.

Se vengono specificati un Folder un tipo di Documento ed un tipo di Associazione non permessi il sistema genera una AcarisException con messaggio SER-E905 [A13] se l'objectType non accetta un contentStream viene sollevata una AcarisException con messaggio SER-E903 [A13] .

2.4.2.2 Interfaccia di richiamo (input)

- ✓ ID repositoryId: Repository Id
- ✓ ID typeId: Tipo Documento
- ✓ ID AssociationTypeID: Tipo Associazione
- ✓ ID principal: Identificativo dell' attore che esegue l'operazione
- ✓ Collection propertiesDocumento : Elenco delle properties che caratterizzano il documento.
- ✓ Collection propertiesAssociazione : Elenco delle properties che caratterizzano il documento.
- ✓ ID folderId: Folder padre per il documento che si vuole creare
- ✓ ContentStream contentStream

▪ **typeId**

I tipi di oggetto ammessi sono:

- ✓ ContenutoFisicoPropertiesType

- ✓ DocumentoDBPropertiesType (2a Release)
- ✓ DocumentoRegistroPropertiesType (2a Release)
- ✓ DocumentoSemplicePropertiesType

▪ **AssociationTypeId**

I tipi di oggetto ammessi sono:

- ✓ ClassificazionePropertiesType
- ✓ DocumentoFisicoPropertiesType

▪ **principal**

Identifica univocamente l'attore che vuole operare sul repository, questo deve essere presente su ACTA ed autorizzato ad operare sui contenuti, viene verificata l'autorizzazione alla creazione dello specifico oggetto Document richiesto (abilitazione al corrispettivo CdU in ACTA).

▪ **collection properties**

Per le properties fare riferimento al Data Model rif. [A3] al capitolo Document Object Type.

▪ **folderId**

Identifica l'oggetto in cui viene creato il nuovo documento, deve essere verificato che l'oggetto che si vuole creare possa essere inserito nel Parent folder (documento di objectConstraints)

▪ **contentStream**

E' il file fisico (nel caso di inserimento di un documento elettronico) è solo previsto per un l'oggetto di tipo ContenutoFisicoPropertiesType

2.4.2.3 Interfaccia di output

- ✓ ID objectId: Identificativo del document creato

2.4.2.4 Logica di Business

Combinazioni permesse fra oggetti:

TypeId	AssociationTypeId	FolderId <i>La classe dell'oggetto deve essere del tipo ...</i>
DocumentoSemplicePropertiesType DocumentoDBPropertiesType DocumentoRegistroPropertiesType	ClassificazionePropertiesType	DossierPropertiesType SottofascicoloPropertiesType FascicoloRealeEreditatoPropertiesType FascicoloRealeLiberoPropertiesType FascicoloRealeContinuoPropertiesType FascicoloRealeLegislaturaPropertiesType FascicoloRealeAnnualePropertiesType VolumeSerieTipDocPropertiesType SerieTipologicaDocumentiPropertiesType VolumeSottoFascPropertiesType

		VolumeFascPropertiesType
ContenutoFisicoPropertiesType	DocumentoFisicoPropertiesType	DocumentPropertiesType (è ammessa solo l'istanza denominata DocumentRoot in [A3] para 2.4.1)

Se viene indicata una combinazione non permessa il sistema genera una AcarisException con il messaggio **SER-E905** [A13].

Nella tabella successiva sono indicati i casi d'uso che descrivono i controlli che il sistema deve effettuare a livello di 'logica di business' dipendentemente dal **typeId** specificato nella chiamata del servizio.

TypeId	Caso d'uso
ContenutoFisicoPropertiesType	Il sistema esegue i controlli indicati nel caso d'uso Crea_DocumentoElettronico rif.[A12]
DocumentoSemplicePropertiesType DocumentoDBPropertiesType DocumentoRegistroPropertiesType	Il sistema esegue i controlli indicati nel caso d'uso Crea_DocumentoClassificato rif.[A11]

2.4.3 Operazione: *createFolder*

2.4.3.1 Descrizione dell'operazione

Crea un oggetto Folder del tipo specificato.

Se il tipo di Folder che deve essere creato non è uno fra quelli ammessi ("Allowed_Child_Object_Types") per il Folder padre viene sollevata una AcarisException **SER-E900** [A13].

Non si può creare il Root folder con questo servizio.

Gli oggetti di tipo Folder 'creabili' sono descritti nel documento SRS Servizi rif.[A2] per ognuno di questi oggetti vengono di seguito descritte le regole di business per la loro creazione.

2.4.3.2 Interfaccia di richiamo (input)

- ✓ ID repositoryId: Identificativo del repository
- ✓ ID typeId: Tipo Folder
- ✓ ID principal: Identificativo dell' attore che esegue l'operazione
- ✓ Collection properties: Elenco delle properties che caratterizzano il folder.
- ✓ ID folderId: Folder padre per il folder che si vuole creare.

▪ **typeID**

I Tipi Folder ammessi sono:

- ✓ Dossier
- ✓ Sottofascicolo
- ✓ Fascicolo reale ereditato
- ✓ Fascicolo reale libero
- ✓ Fascicolo reale continuo
- ✓ Fascicolo reale legislatura
- ✓ Fascicolo reale annuale
- ✓ Fascicolo temporaneo
- ✓ Titolare (2a Release)
- ✓ Voce (2a Release)
- ✓ Serie di dossier (2a Release)
- ✓ Serie tipologica di documenti (2a Release)
- ✓ Serie di fascicoli (2a Release)
- ✓ Volume di fascicoli (2a Release)
- ✓ Volume serie fascicoli (2a Release)
- ✓ Volume serie tipologica documenti (2a Release)
- ✓ Volume sottofascicoli (2a Release)

▪ **Principal**

Identifica univocamente l'attore che vuole operare sul repository, questo deve essere presente su ACTA ed autorizzato ad operare sui contenuti, viene verificata l'autorizzazione alla creazione dello specifico oggetto Folder richiesto (abilitazione al corrispettivo CdU in ACTA).

▪ **collection properties**

Per le properties fare riferimento al Data Model rif. [A3] al capitolo Folder Object Type.

▪ **folderId**

Identifica l'oggetto in cui viene creato il nuovo folder, deve essere verificato che l'oggetto che si vuole creare possa essere inserito nel Parent folder.

2.4.3.3 Interfaccia di output

- ✓ ID objectId: Identificativo del folder creato

2.4.3.4 Logica di Business

Nella tabella successiva sono indicati i casi d'uso che descrivono i controlli che il sistema deve effettuare a livello di 'logica di business' dipendentemente dal *typeId* specificato nella chiamata del servizio.

TypeId	Caso d'uso
DossierPropertiesType	Da implementare

SottofascicoloPropertiesType	
FascicoloRealeEreditatoPropertiesType	Il sistema esegue i controlli indicati nel caso d'uso Crea Fascicolo rif. [A9]
FascicoloRealeLiberoPropertiesType	
FascicoloRealeContinuoPropertiesType	
FascicoloRealeLegislaturaPropertiesType	
FascicoloRealeAnnualePropertiesType	
FascicoloTemporaneoPropertiesType	Da implementare
DocumentoFisicoPropertiesType	Il sistema esegue i controlli indicati nel caso d'uso Crea Documento Fisico rif. [A8]

Oggetti non creabili nella prima release:

TitolarioPropertiesType	n.a
VocePropertiesType	n.a
SerieDossierPropertiesType	n.a
SerieTipologicaDocumentiPropertiesType	n.a
SerieFascicoliPropertiesType	n.a
VolumeFascicoliPropertiesType	n.a
VolumeSerieFascicoliPropertiesType	n.a
VolumeSerieTipologicaDocumentiPropertiesType	n.a
VolumeSottofascicoliPropertiesType	n.a

2.4.4 Operazione: *createRelationship*

2.4.4.1 Descrizione dell'operazione

Crea un oggetto Relationship del tipo specificato.

Se il source object o il target object non sono trovati viene sollevata una AcarisException SER-E903 [A13].

Se il TypeID del source o dell' oggetto Target non è uno degli ObjectTypes permessi nel Tipo di Relationship viene sollevata una AcarisException SER-E904 [A13].

Gli oggetti di tipo Relationship 'creabili' sono descritti nel documento SRS Servizi rif.[A2] per ognuno di questi oggetti vengono di seguito descritte le regole di business per la loro creazione.

2.4.4.2 Interfaccia di richiamo (input)

- ✓ ID repositoryId: Repository Id
- ✓ ID typeId: Tipo Relationship
- ✓ Collection properties: Elenco delle properties che caratterizzano la relazione.
- ✓ ID sourceObjectId : Identificativo oggetto sorgente della relazione

- ✓ ID targetObjectId : Identificativo oggetto destinazione della relazione

- **typeID**

I Tipi Relationship ammessi sono

- ✓ DocumentAssociationPropertiesType
- ✓ HistoryModificheTecnichePropertiesType
- ✓ DocumentCompositionPropertiesType
- ✓ HistoryVecchieVersioniPropertiesType

- **collection properties**

Per le properties fare riferimento al Data Model rif. [A3] al capitolo Relationship Object Type.

- **sourceObjectId**

Identifica l'oggetto sorgente per la nuova relazione da creare, deve essere verificato che l'oggetto sia tra quelli permessi per la creazione della relazione.

- **targetObjectId**

Identifica l'oggetto destinazione per la nuova relazione da creare, deve essere verificato che l'oggetto sia tra quelli permessi per la creazione della relazione.

2.4.4.3 Interfaccia di output

- ✓ ID objectId: Identificativo della relationship creata

2.4.4.4 Logica di Business

Nella tabella successiva sono indicati i casi d'uso che descrivono i controlli che il sistema deve effettuare a livello di 'logica di business' dipendentemente dal **typeId** specificato nella chiamata del servizio.

TypeId	Caso d'uso
DocumentAssociationPropertiesType	Non da implementare
HistoryModificheTecnichePropertiesType	Da implementare
DocumentCompositionPropertiesType	Crea_DocumentComposition rif.[A7]
HistoryVecchieVersioniPropertiesType	Da implementare

2.4.5 Operazione: **updateProperties**

2.4.5.1 Descrizione dell'operazione

L'operazione consente l'aggiornamento delle properties di un oggetto specifico.

La tipologia dell'oggetto da modificare viene stabilita facendo riferimento a quanto specificato nel parametro del tipo PropertyType.

Vengono modificate solo le properties espressamente elencate nel parametro del tipo PropertyType[].

La politica adottata per l'aggiornamento è quella della completa sovrascrittura dei valori esistenti nel repository.

Il servizio di `updateProperties`, non deve essere utilizzato per aggiornare i campi che modificano lo stato dell'entità, ci riferiamo, ad esempio, a quei metadati che comportano l'apertura e chiusura di una struttura aggregativa.

In questi casi, la modifica viene gestita tramite appositi e indipendenti casi d'uso.

Per evitare che questi metadati vengano comunque modificati tramite `updateProperties`, anche tramite involontaria impostazione a true dell'apposito flag, il sistema effettua una verifica puntuale che il metadato non sia presente nell'apposita lista di metadati delicati.

2.4.5.2 Interfaccia di richiamo (input)

- `repositoryId`
- `objectId`
- `principalId`
- `changeToken`
- `properties`

I parametri del tipo `RepositoryIdType` e `PrincipalIdType` non presentano particolarità rispetto agli altri servizi.

ObjectIdType objectId

Rappresenta l'identificatore dell'oggetto da modificare: tipicamente selezionato in precedente operazione di navigazione o query

ChangeTokenType changeToken

Contiene le informazioni necessarie per la verifica dell'accesso in concorrenza.

Valorizzato automaticamente dal sistema nelle operazioni `getProperties` e `query` nei casi di `EnumPropertyFilter.ALL` e `EnumPropertyFilter.LIST`.

PropertyType[] properties

Elenco dei metadati da aggiornare. Il sistema sovrascrive i soli metadati specificati nell'elenco. La stessa politica viene adottata nel caso di metadati multivalore

2.4.5.3 Interfaccia di output

SimpleResponseType

Oggetto complesso costituito dall'identificatore dell'oggetto modificato e dal `changeToken` relativo alla modifica effettuata

2.4.5.4 Logiche di Business

Nella prima consegna viene effettuata la modifica delle `properties` del documento semplice.

Nelle successive versioni del documento verranno gestite le modifiche delle altre tipologie di oggetti aggiornabili tramite servizi `acaris`.

Nella tabella di seguito, vengono elencati tutti gli oggetti che possono essere Aggiornati in GCO e che dovranno essere sviluppati.

Typeld	
DocumentoDBPropertiesType	Consegnato
DocumentoRegistroPropertiesType	
DocumentoSemplicePropertiesType	
FascicoloTemporaneoPropertiesType	Da Implementare
SottofascicoloPropertiesType	Da Implementare
DossierPropertiesType	Da Implementare
VolumeSerieFascicoliPropertiesType	Da Implementare
VolumeSerieTipologicaDocumentiPropertiesType	
VolumeFascicoliPropertiesType	
VolumeSottofascicoliPropertiesType	Da Implementare
SerieDossierPropertiesType	
SerieTipologicaDocumentiPropertiesType	
SerieFascicoliPropertiesType	Da Implementare
FascicoloRealeEreditatoPropertiesType	
FascicoloRealeLiberoPropertiesType	
FascicoloRealeLegislaturaPropertiesType	
FascicoloRealeAnnualePropertiesType	
FascicoloRealeContinuoPropertiesType	
AnnotazioniPropertiesType	Da Implementare

2.4.6 Operazione *query*

2.4.6.1 Descrizione dell'operazione

Consente di effettuare delle query sul modulo funzionale Acaris Archive.

Per un approfondimento si veda l'apposita sezione in allegato tecnico.

2.4.6.2 Interfaccia di richiamo (input)

- repositoryId
- principalId
- target
- filter
- criteria
- maxItems
- skipCount

I parametri del tipo RepositoryIdType e PrincipalIdType non presentano particolarità rispetto agli altri servizi.

QueryableObjectType target

Rappresenta l'entità interrogabile. Può essere un'entità fisica oppure una vista dinamica.

PropertyFilterType filter

Permette di specificare l'elenco delle properties che devono essere restituite per ogni oggetto individuato nel sistema.

QueryConditionType[] criteria

Elenco di criteri (in questa prima versione concatenate in AND) di selezione.

Integer maxItems**Integer skipCount**

Vengono utilizzati per gestire la paginazione dei risultati.

2.4.6.3 Interfaccia di output

2.4.6.4 Logiche di Business

Per un approfondimento in merito a interfaccia di input, parametri, interfaccia di output si veda l'apposita sezione dell'allegato tecnico di Acta.

2.4.7 Operazione *getContentStream*

2.4.7.1 Descrizione dell'operazione

L'operazione serve a recuperare un content stream di un contenuto fisico del tipo specificato e collegato ad un documento fisico.

Ad un documento fisico possono corrispondere n contenuti fisici il cui tipo deve rientrare in un'elencazione che accanto al documento primario prevede: firma digitale, marca temporale, rendition engine e rendition document.

Per ognuna delle tipologie è previsto un solo content stream salvo nel caso di firma e marca: in questo caso il sistema restituisce una collection.

[AcarisContentStreamType\[\]](#) [getContentStream](#)([ObjectIdType](#) repositoryId, [ObjectIdType](#) documentId, [PrincipalIdType](#) principalId, [EnumStreamId](#) streamId)

2.4.7.2 Interfaccia di richiamo (input)

- repositoryId
- principalId
- documentId
- streamId

I parametri del tipo RepositoryIdType e PrincipalIdType non presentano particolarità rispetto agli altri servizi.

ObjectIdType documentId

Rappresenta l'identificatore del documento restituito tramite precedente operazione di navigazione o query del sistema. Si tratta dell'identificatore del documento fisico.

EnumStreamId streamId

Permette di specificare se il documento di cui si richiede lo stream è il documento primario oppure uno dei documenti collegati. Segue la specifica nell'apposito schema:

```
<xs:simpleType name="enumStreamId">
  <xs:restriction base="common:string">
    <xs:enumeration value="primary" />
    <xs:enumeration value="signature" />
    <xs:enumeration value="timestamp" />
    <xs:enumeration value="renditionEngine" />
    <xs:enumeration value="renditionDocument" />
  </xs:restriction>
</xs:simpleType>
```

2.4.7.3 Interfaccia di output

Viene restituita una collection di `AcarisContentStreamType` contenente lo stream ed alcuni metadati associati. L'oggetto viene così definito nell'apposito schema:

```
<!-- Content Stream -->
<xs:complexType name="acarisContentStreamType">
  <xs:sequence>
    <xs:element name="length" type="common:contentStreamLengthType" minOccurs="0" />
    <xs:element name="mimeType" type="common:enumMimeTypeType" minOccurs="0" />
    <xs:element name="filename" type="common:contentStreamFilenameType" minOccurs="0" />
    <!-- <xs:element name="uri" type="xs:anyURI" minOccurs="0" /> -->
    <xs:element name="streamMTOM" type="common:base64Binary"
xmime:expectedContentTypes="*/*" />
    <xs:element name="stream" type="common:base64Binary" />
    <xs:element name="nodeUID" type="common:string" minOccurs="0" />
  </xs:sequence>
</xs:complexType>
```

2.4.7.4 Logiche di Business

Le logiche dell'operazione sono descritte in PGED-ACTA-UC-ACARISGCS01-V01_Get_ContentStream.doc

2.4.8 Operazione *setContentStream*

2.4.8.1 Descrizione dell'operazione

L'operazione imposta (crea o sostituisce) il content-stream di uno specifico oggetto di tipo documento.

Se il documento ha settato i valori Rappresentazione digitale = NO e Cartaceo = SI (placeholder), non deve essere permesso l'aggiunta di uno stream.

2.4.8.2 Interfaccia di richiamo (input)

- repositoryId
- principalId
- documentId
- changeToken
- contentstream

I parametri del tipo RepositoryIdType e PrincipalIdType non presentano particolarità rispetto agli altri servizi.

ObjectIdType documentId

Rappresenta l'identificatore del documento restituito tramite precedente operazione di navigazione o query del sistema.

contentStream

E' il file fisico che si vuole caricare

ChangeTokenType changeToken

Contiene le informazioni necessarie per la verifica dell'accesso in concorrenza.
Valorizzato automaticamente dal sistema.

2.4.8.3 Interfaccia di output

SimpleResponseType

Oggetto complesso costituito dall'identificatore dell'oggetto modificato e dal changeToken relativo alla modifica effettuata

2.4.8.4 Logiche di Business

2.4.9 Operazione *addRenditionStream*

2.4.9.1 Descrizione dell'operazione

L'operazione imposta uno specifico rendition stream, sia esso un foglio di stile, o il risultato dell'applicazione dello stesso foglio di stile al doc xml inserito come documento.

Se il documento ha settato i valori Rappresentazione digitale = NO e Cartaceo = SI (placeholder), non deve essere permesso l'aggiunta di uno stream.

2.4.9.2 Interfaccia di richiamo (input)

- repositoryId
- principalId
- documentId
- changeToken
- streamId
- contentstream

I parametri del tipo RepositoryIdType e PrincipalIdType non presentano particolarità rispetto agli altri servizi.

ObjectIdType documentId

Rappresenta l'identificatore del documento restituito tramite precedente operazione di navigazione o query del sistema.

streamId

E' la tipologia di Rendition file che si vuole caricare.

- PrimaryStream = il foglio di stile o Rendition Engine e dovrà essere salvato come definito nell'apposita entità definita dal repository.
- renditionPDFstream = il file pdf, risultato dell'utilizzo del foglio di stile, dovrà essere salvato come definito nell'apposita entità definita dal repository.
- renditionHTMLstream = il file html, risultato dell'utilizzo del foglio di stile, dovrà essere salvato come definito nell'apposita entità definita dal repository.

contentStream

E' il file fisico che si vuole caricare

ChangeTokenType changeToken

Contiene le informazioni necessarie per la verifica dell'accesso in concorrenza.

Valorizzato automaticamente dal sistema.

2.4.9.3 Interfaccia di output**SimpleResponseType**

Oggetto complesso costituito dall'identificatore dell'oggetto modificato e dal changeToken relativo alla modifica effettuata

2.4.9.4 Logiche di Business**2.4.10 Operazione *getProperties*****2.4.10.1 Descrizione dell'operazione**

Operazione di interrogazione puntuale del Titolare, questo servizio permette di richiedere il dettaglio delle properties dell'oggetto specificato.

Il servizio segue le logiche relative agli oggetti presenti su ActaWeb

TypeId	
DocumentoDBPropertiesType	Consegnato
DocumentoRegistroPropertiesType	
DocumentoSemplicePropertiesType	
FascicoloTemporaneoPropertiesType	Consegnato
SottofascicoloPropertiesType	Consegnato
DossierPropertiesType	Consegnato
VolumeSerieFascicoliPropertiesType	Consegnato
VolumeSerieTipologicaDocumentiPropertiesType	
VolumeFascicoliPropertiesType	
VolumeSottofascicoliPropertiesType	Consegnato
SerieDossierPropertiesType	
SerieTipologicaDocumentiPropertiesType	
SerieFascicoliPropertiesType	Consegnato
FascicoloRealeEreditatoPropertiesType	
FascicoloRealeLiberoPropertiesType	
FascicoloRealeLegislaturaPropertiesType	
FascicoloRealeAnnualePropertiesType	
FascicoloRealeContinuoPropertiesType	
AnnotazioniPropertiesType	Da Implementare

2.4.10.2 Interfaccia di richiamo (input)

- **ObjectIdType repositoryId:** Repository e quindi ente di riferimento.
- **ObjectIdType objectId:** Nodo dell'albero dell'organigramma da interrogare.
- **PrincipalIdType principalId:** Utente che richiede il servizio (se non specificato si possono interrogare solo le AOO o l'Ente).
- **PropertyFilterType filter:** Proprietà da valorizzare per il nodo interrogato.

2.4.10.3 Interfaccia di output

- **ObjectIdType objectId:** Nodo dell'albero dell'organigramma.
- **<Array> PropertyType properties:** Proprietà valorizzate per il nodo.

2.4.10.4 Logiche di Business

2.4.11 **Operazione moveDocument**

2.4.11.1 Descrizione dell'operazione

Consente di effettuare lo spostamento di un documento (con gli eventuali allegati), da una struttura aggregativa di partenza ad un'altra di destinazione.

2.4.11.2 Interfaccia di richiamo (input)

- repositoryId
- principalId
- associativeObjectId
- sourceFolderId
- destinationFolderId
- associativeProperties

I parametri del tipo RepositoryIdType e PrincipalIdType non presentano particolarità rispetto agli altri servizi.

ObjectIdType associativeObjectId

Rappresenta l'identificatore della classificazione del documento che si vuole spostare.

ObjectIdType sourceFolderId

E' l'identificatore della struttura aggregativa di partenza.

L'operazione è consentita applicando le logiche previste da acta per l'operazione di taglia. E' necessario essere adeguatamente profilati e avere le corrette autorizzazioni sui contenuti: acl

ObjectIdType destinationFolderId

E' l'identificatore della struttura aggregativa di destinazione.

L'operazione è consentita applicando le logiche previste da acta per l'operazione di incolla. E' necessario essere adeguatamente profilati e avere le corrette autorizzazioni sui contenuti: acl

PropertiesType associativeProperties

Sottoclasse di properties (MoveDocumentPropertiesType) contenente alcune istruzioni per l'esecuzione dell'operazione.

Le properties sono le seguenti:

offlineMoveRequest che serve per indicare al sistema se si vuole chiedere la prenotazione dell'esecuzione via batch dell'operazione qualora il documento da spostare abbia un numero di allegati superiore a quelli previsti

idSmistamentoType che serve per consentire al sistema di effettuare i controlli necessari quando il documento da spostare è collegato a uno smistamento

2.4.11.3 Interfaccia di output

ObjectIdType

Identificatore della nuova classificazione creata oppure, nel caso di prenotazione dell'esecuzione offline, della prenotazione della richiesta registrata dal sistema.

2.4.11.4 Logiche di Business

Si richiamano le logiche previste da acta per quanto riguarda il completamento delle operazioni di taglia e incolla di documenti.

Se il documento ha degli allegati, questi ultimi devono essere spostati insieme al documento principale.

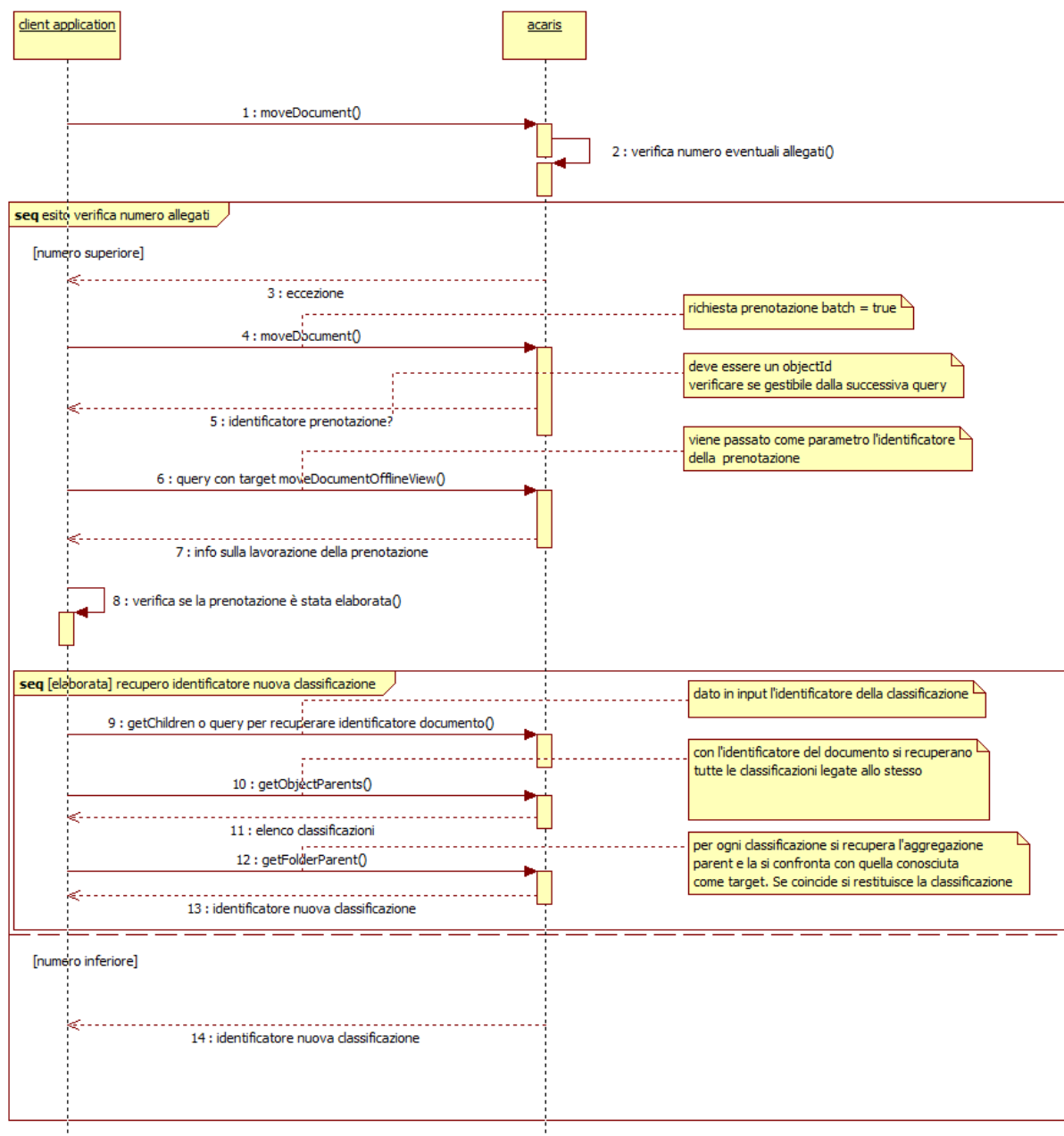
Il singolo allegato non può essere spostato indipendente dal documento principale.

Nel caso di documento con allegati, il sistema verifica che il numero degli stessi non superi quello previsto per l'ente. In quest'ultimo caso, il sistema solleva eccezione invitando ad effettuare una nuova invocazione del servizio richiedendo però la prenotazione dell'esecuzione offline dello spostamento.

Il sistema, restituisce, in luogo della nuova classificazione, come normalmente fa quando non si rientra in questo scenario, l'identificatore della prenotazione effettuata. A questo punto, per recuperare l'identificatore della nuova classificazione, l'applicazione client dovrà effettuare i seguenti passaggi:

1. verificare che la richiesta sia stata elaborata: mediante l'invocazione di query con target *moveDocumentOfflineView* e criterio di ricerca *objectIdRichiestaPrenotazione*
2. recuperare l'identificatore della nuova classificazione mediante un'operazione di navigazione a partire dall'identificatore del documento legato alla classificazione originaria e avendo come parent folder la nuova struttura aggregativa. Per quell'identificatore il sistema dovrà trovare una sola classificazione che è quella di interesse del fruitore

Segue un diagramma che illustra gli scenari sopra descritti.



Il parametro associativeProperties deve essere valorizzato come segue:

null (nessuna valorizzazione) nei casi di spostamento “normale”

MoveDocumentPropertiesType quando si deve gestire lo spostamento di un documento con allegati e/o quando il documento da spostare è legato a una registrazione di protocollo e quest'ultima è stata smistata; in quest'ultimo caso la classe in oggetto serve per veicolare l'identificatore dello smistamento da utilizzare (ce ne potrebbero infatti essere diversi).

Viene introdotta la possibilità di utilizzare l'operazione anche per effettuare lo spostamento di un documento (che non deve avere allegati) come allegato di un altro documento di cui si indica la classificazione.

Il sistema effettua i controlli opportuni sollevando se necessario eccezione e veicolando, a questo riguardo, i seguenti numeri di errore: SER-E168, SER-E169, SER-E170.

Non è consentito spostare un documento in una struttura aggregativa nella quale, per lo stesso documento, esiste già una classificazione. In caso di violazione il sistema solleva eccezione con il seguente codice di errore: SER -E171

2.5 Governance

2.5.1 Requisiti di sicurezza

I requisiti di qualità sono comuni all'intero sistema e sono elencati nel documento di requisiti funzionali (cfr. riferimento [A1]).