



Comitato MAG

MAG Metadati Amministrativi e Gestionali

Manuale Utente

Versione 2.0.1

a cura di Elena Pierazzo

Ultimo aggiornamento: 8 marzo 2006

ICCU

Indice

1. Introduzione.....	6
1. Alle origini di MAG.....	6
2. Metadati e Oggetti Digitali.....	7
3. Lo Schema MAG.....	8
4. Interazione con standard internazionali.....	9
1. Dublin Core.....	10
2. NISO e MIX.....	10
3. METS.....	11
4. MPEG-21.....	11
2. Introduzione a XML.....	13
1. Cos'è XML.....	13
2. Sintassi XML.....	14
1. Elementi.....	14
2. Attributi.....	16
3. Commenti.....	17
3. Namespaces.....	17
4. Sintassi degli schema W3C.....	18
1. Un esempio: il file bibliografia.xml e il suo schema.....	18
2. Tipi complessi.....	21
3. Occorrenze degli elementi e degli attributi.....	22
4. Tipi semplici.....	23
5. Tipi anonimi.....	24
6. Tipi complessi derivati da tipi semplici.....	25
7. Scelte fra più elementi.....	26
8. Le annotazioni.....	26
9. Target namespace e oggetti locali non qualificati o qualificati.....	27
10. Schemi suddivisi in più file e importazione di altri schemi.....	28
3. Struttura dello schema e dei documenti MAG.....	32
1. Struttura e componenti dello schema MAG.....	32
1. Il file metadigit.xsd.....	32
2. I tipi MAG: il file metatype.xsd, audio.xsd e video.xsd.....	33
3. xlink e il tipo complesso link.....	33
2. Struttura dei documenti MAG.....	34
1. L'elemento root.....	34
2. Sezioni obbligatorie.....	36
3. Sezioni opzionali.....	37
4. Sezione GEN.....	38
1. L'istituzione e il progetto.....	38
2. Accessibilità e status dell'oggetto digitale.....	39
3. Caratteristiche tecniche di gruppi omogenei di oggetti.....	41
1. Gruppi di immagini.....	42
2. Gruppi di tracce audio.....	43

3. Gruppi di stream video.....	45
5. Sezione BIB.....	47
1. La descrizione bibliografica: il set di elementi Dublin Core.....	48
1. Il dc:identifier.....	48
2. Gli altri elementi Dublin Core.....	50
2. La proprietà dell'oggetto analogico.....	53
3. Sistemi di catalogazioni specializzate.....	54
4. Pubblicazioni seriali e unità componenti.....	55
1. Pubblicazioni seriali.....	55
2. Unità componenti.....	57
6. Sezione STRU.....	59
1. Il numero di sequenza e la sua denominazione.....	60
2. Il contenuto di un livello strutturale.....	61
1. La struttura interna di una risorsa.....	61
2. Record di spoglio. Spogli nudi e spogli vestiti.....	63
3. Ricongiungimento di parti separate di un oggetto unitario.....	66
4. Collegamenti fra oggetti digitali affini.....	67
1. Ricongiungimento all'interno dell'unico record MAG.....	68
2. Ricongiungimento con record di spoglio.....	69
3. Ricongiungimento con record di spoglio di secondo livello.....	71
5. La descrizione di element.....	74
7. Sezione IMG.....	77
1. Identificazione, denominazione e uso delle immagini.....	78
2. Tipo di scansione.....	80
3. Localizzazione, integrità e grandezza del file.....	81
4. Le dimensioni dell'immagine digitale.....	81
5. Principali caratteristiche tecniche dell'immagine.....	82
1. La risoluzione delle immagini: misurazioni obsolete.....	84
6. Il formato delle immagini.....	84
7. La scansione dell'oggetto.....	85
8. La creazione del file.....	87
9. Il target.....	87
10. Formati alternativi dell'immagine.....	88
11. Lo schema NISO.....	89
8. Sezione AUDIO.....	97
1. Identificazione e denominazione delle tracce audio.....	98
2. La descrizione tecnica della traccia audio.....	98
1. Uso, localizzazione, integrità e grandezza del file.....	100
2. Le dimensioni della traccia audio.....	101
3. Principali caratteristiche tecniche della traccia audio.....	102
4. Il formato delle tracce audio.....	104
5. La trascrizione dell'oggetto.....	106
1. Gli strumenti di digitalizzazione.....	109
2. Dati tecnici della trascrizione: valori di sintesi e valori globali.....	110
6. La creazione del file.....	113

9. Sezione VIDEO.....	115
1. Identificazione e denominazione degli stream video.....	116
2. La descrizione tecnica della traccia video.....	116
1. Uso, localizzazione, integrità e grandezza del file.....	118
2. Le dimensioni stream video.....	119
3. Principali caratteristiche tecniche dello stream video.....	120
4. Il formato degli stream video.....	123
5. La creazione dello stream video digitale.....	125
1. Gli strumenti di digitalizzazione.....	128
2. Dati tecnici della trascrizione: valori di sintesi e valori globali.....	129
6. La creazione del file.....	131
10. Sezione OCR.....	132
1. Identificazione, denominazione e uso dei file di testo.....	133
2. Localizzazione, integrità, fonte e grandezza del file di testo.....	133
3. Il formato dei file di testo ocr.....	134
4. Il software di riconoscimento ottico.....	135
5. La creazione del file.....	135
11. Sezione DOC.....	136
1. Identificazione, denominazione e uso dei file di testo DOC.....	137
2. Localizzazione, integrità e grandezza del file di testo.....	137
3. Il formato dei file di testo born digital.....	138
4. La creazione del file.....	138
12. Sezione DIS.....	140

Il presente documento

Il presente documento costituisce il *Manuale d'Uso* per lo standard MAG di cui il *Reference* costituisce la versione sintetica.

Qualsiasi discrepanza esistente fra quanto documentato dal presente *Manuale d'Uso* e lo schema MAG deve essere risolto dando priorità a quanto imposto dallo schema. In caso di discrepanza fra il *Manuale d'Uso* e il *Reference* deve essere risolto dando priorità a quanto espresso dal *Manuale*.

Il presente manuale è stato prodotto da un documento XML codificato secondo le direttive [TEI](#), in particolare tramite l'adozione della DTD [TEI Lite](#)¹ cui sono stati applicati due fogli di stile XSL per la produzione della versione HTML e PDF.

1. BURNARD L., SPERBERG-MCQUEEN C. M. (1995), *TEI Lite: An Introduction to Text Encoding for Interchange (TEI U5)*, <http://www.tei-c.org/Lite/index.html>

1. Introduzione

1.1. Alle origini di MAG

Con l'acronimo MAG - Metadati Amministrativi e Gestionali - viene proposto uno **application profile**² che ha l'obiettivo di fornire le specifiche formali per la fase di raccolta, di trasferimento e disseminazione dei metadati e dei dati digitali nei rispettivi archivi. Lo Schema MAG è realizzato e mantenuto dal Comitato Mag.

L'ICCU, tra le numerose attività relative all'utilizzo dei metadati, ha costituito nel 2000, un Gruppo di studio sugli standard e le applicazioni di metadati nei beni culturali a cui hanno partecipato rappresentanti di biblioteche, musei e archivi. Il Gruppo è nato con l'obiettivo di coordinare a livello nazionale le implementazioni di metadati nei progetti di digitalizzazione nei diversi settori dei beni culturali e di raccordare le iniziative italiane con quelle europee e internazionali.

Il Gruppo di studio, che ha operato nel periodo 2000 – 2002, si è suddiviso al suo interno in due sottogruppi:

- Sottogruppo Metadati descrittivi
- Sottogruppo Metadati amministrativi gestionali - MAG

Da luglio 2003 si è formalmente costituito un gruppo di lavoro permanente – il Comitato Mag³ – che prosegue le attività del Gruppo di studio sugli standard e le applicazioni di metadati nei beni culturali con particolare riferimento alle attività connesse alla diffusione ed evoluzione del set di Metadati Amministrativi Gestionali (MAG).

Gli obiettivi del Comitato MAG sono:

- diffusione dello Schema MAG
- mantenimento ed evoluzione dello schema
- produzione di manuali e linee guida
- assistenza agli implementatori
- formazione e promozione
- rapporti con altri progetti e agenzie (progetti europei, Dublin Core, etc.)

Le attività del Comitato MAG sono condotte da una Segreteria tecnica. La Segreteria tecnica ha il compito di curare l'istruttoria delle richieste degli implementatori, fornendo loro assistenza, e di coordinare le attività di aggiornamento ed estensione dello schema.

La segreteria ha avviato le seguenti attività:

- elaborazione del **Manuale** per l'utilizzo dello Schema MAG
- produzione di documentazione di supporto sia a carattere generale che specifico per le diverse tipologie di materiale sottoforma di **Raccomandazioni e Linee guida**

2. «We define application profiles as schemas which consist of data elements drawn from one or more namespaces, combined together by implementors, and optimised for a particular local application» (R. Heery and M. Patel. *Application profiles: mixing and matching metadata schemas*, <http://www.ariadne.ac.uk/issue25/app-profiles/>).

3. Il Comitato MAG è costituito da: Francesco Baldi (Discoteca di Stato); Giovanni Bergamin (Biblioteca Nazionale Centrale di Firenze); Gianfranco Crupi (Università degli Studi La Sapienza di Roma); Gloria Cirocchi, Simona Gatta (Biblioteca della Camera dei Deputati); Cristina Magliano, Patrizia Martini (ICCU); Maurizio Messina (Biblioteca Marciana di Venezia).

- produzione di un registro di link a schemi, modelli e reference tecnici di riferimento.

1.2. Metadati e Oggetti Digitali

Il **Manuale MAG** costituisce uno strumento di supporto alle attività di impianto e di gestione di basi dati di oggetti digitali che formano una **collezione digitale**. Tali oggetti possono essere il prodotto di lavori di digitalizzazione di originali analogici o essere digitali nativi, e possono avere molteplici tipologie: immagini statiche, documenti in formato testo, suono, documenti audiovisivi, e insieme possono generare collezioni digitali multimediali.

L'ambito di più immediata applicazione del MAG è dato dai progetti che fanno riferimento alla **Biblioteca Digitale Italiana (BDI)**, qui intesa estensivamente e genericamente come insieme delle attività di documentazione digitale che intendono condividere determinati standard e possono essere in questo senso certificate.

In tali attività i metadati rivestono un'importanza crescente, tanto da venire considerati parte costituente della definizione stessa di oggetto digitale⁴: una risorsa digitale è inseparabilmente composta dal contenuto informativo (una sequenza di bit) e da una serie di informazioni (metadati) tali da rendere quella sequenza di bit significativa, individuabile, e accessibile per la fruizione, l'archiviazione, la conservazione, la disseminazione e le altre operazioni gestionali.

Nell'ambito dei progetti di digitalizzazione un'accurata definizione dei metadati associati agli oggetti che compongono una collezione digitale costituisce il discrimine fra i progetti orientati alla mera produzione di **dati** e quelli orientati all'erogazione di **servizi**. I primi effettuano scansioni digitali per lo più con l'obiettivo di ridurre l'uso dei documenti analogici originali o di realizzare copie da cui trarre riproduzioni, e si sono spesso concretizzati nella produzione di quantità elevate di supporti poco agevoli da maneggiare, come i CD-ROM, DVD, ecc. I secondi tengono conto dell'esigenza, e si assumono la responsabilità, di certificare l'integrità dei contenuti informativi, di conservarli e di mantenere inalterata nel lungo periodo la loro accessibilità da parte di una determinata comunità di utenti. Sia che siano prevalentemente orientati alla conservazione permanente dei contenuti informativi, sia che privilegino l'accessibilità, i progetti che prevedono un uso coerente dei metadati sono in grado, in una parola, di garantire la qualità dell'informazione digitale e di porsi in una posizione più favorevole nel sollecitare i finanziamenti necessari a sostenere nel lungo periodo il proprio funzionamento. Esiste infatti un nesso molto forte fra qualità complessiva dei contenuti e sostenibilità dei progetti.

La funzione normalizzatrice dei contenuti digitali svolta dai metadati è inoltre condizione per l'industrializzazione dei processi di digitalizzazione, e contribuisce a creare un mercato di prodotti e servizi in questo settore.

La creazione e l'organizzazione di metadati è da sempre al centro delle attività delle istituzioni della memoria (archivi, biblioteche, musei), e diverse sono le proposte per la loro tassonomia; ai fini del **Manuale MAG** si è presa a riferimento quella di Wendler, che ha il vantaggio della semplicità e della chiarezza⁵. I metadati possono dunque essere distinti in tre

4. Si veda ad es. la definizione di **Oggetto Digitale** in California Digital Library. *Digital Object Standard: Metadata, Content and Encoding*, May 18, 2001: "Un oggetto digitale è definito [...] come un qualcosa (es. un'immagine, una registrazione audio, un documento testuale) che è stato codificato in modo digitale e integrato con metadati tali da supportarne l'individuazione, l'uso e l'immagazzinamento".

5. R. Wendler. *LDI Update: Metadata in the Library*. In: «Library Notes», n. 1286 (1999), p. 4-5.

categorie funzionali:

- **DESCRITTIVI:** per l'identificazione ed il recupero degli oggetti digitali; sono costituiti da descrizioni normalizzate dei documenti fonte (o dei documenti digitali nativi), risiedono generalmente nelle basi dati dei sistemi di Information Retrieval all'esterno degli archivi degli oggetti digitali, e sono collegati a questi ultimi tramite appositi link;
- **AMMINISTRATIVI e GESTIONALI:** per le svariate operazioni di gestione degli oggetti digitali all'interno dell'archivio;
- **STRUTTURALI:** per descrivere la struttura interna dei documenti (es. introduzione, capitoli, indice di un libro) e gestire le relazioni fra le varie parti componenti degli oggetti digitali.

Nel mondo della documentazione cartacea (o genericamente analogica) un ruolo preponderante è stato assunto dai metadati descrittivi, mentre i metadati gestionali, quali ad esempio il numero d'inventario o la segnatura in una scheda di catalogo di biblioteca, hanno suscitato, data anche la specificità e l'evidenza della loro funzione, minore interesse.

Nel contesto delle biblioteche digitali invece, data la labilità dell'informazione, sono i metadati amministrativi e gestionali (MAG), più che quelli descrittivi, ad assumere un'importanza preponderante. I MAG rendono infatti più affidabile e sicura l'informazione digitale documentando le modalità di generazione, immissione, archiviazione e manutenzione degli oggetti digitali nel sistema di gestione dell'archivio digitale. In particolare lo schema MAG fornisce delle specifiche formali per le fasi di raccolta e archiviazione dei metadati e fornisce elementi per:⁶

- identificare in maniera univoca gli oggetti digitali
- certificare l'autenticità e l'integrità dei contenuti
- documentare la catena di custodia degli oggetti digitali
- documentare i processi tecnici eseguiti per la conservazione permanente degli oggetti digitali
- informare sulle condizioni e i diritti di accesso agli oggetti digitali da parte degli utenti finali.

Si tratta di informazioni fondamentali ai fini del mantenimento e dell'accessibilità nel lungo periodo dell'"eredità culturale digitale" delle varie comunità. In questo senso i MAG si prestano particolarmente bene ad essere utilizzati all'interno di modelli logico-funzionali dell'archivio degli oggetti digitali fortemente orientati alla conservazione permanente, come l'**Open Archival Information System (OAIS)**⁷, divenuto nel 2003 lo standard ISO 14721.

1.3. Lo Schema MAG

Da un punto di vista logico l'archivio dei metadati gestionali che MAG contribuisce ad alimentare risulta distinto da una parte dall'archivio bibliografico, che, ad esempio nelle biblioteche, si presenta all'utente nella forma di OPAC, e dall'altro dall'archivio degli oggetti

6. Si veda: OCLC/RLG Working Group on Preservation Metadata. *Preservation Metadata for Digital Objects: a Review of the State of the Art*, a White Paper, January 2001, p. 2. http://www.oclc.org/research/pmwg/presmeta_wp.pdf

7. Si veda: Consultative Committee for Space Data Systems. *Reference Model for an Open Archival Information System (OAIS)*, CCSDS 650.0-B-1, Blue Book, January 2002 <http://www.ccsds.org/documents/650x0b1.pdf>

digitali, che costituiscono il contenuto informativo.

Occorre specificare quanto segue:

- più che per particolari tipologie di documenti fonte (es. periodici, musica a stampa o manoscritta, carte geografiche, fotografie etc.), il set di metadati MAG viene definito per tipologie di oggetti digitali (es. immagini statiche, testi prodotti con tecnologia OCR, suono, audiovisivi etc.);
- i MAG sono espressi nella sintassi XML;
- i MAG si presentano come uno standard aperto, documentato, liberamente disponibile e del tutto indipendente da specifiche piattaforme hardware e software, al fine di favorirne un impiego generalizzato;
- di norma i MAG vanno associati alla risorsa digitale nel suo complesso, e comprendono informazioni generali sul progetto e sul metodo di digitalizzazione ed altre informazioni relative all'unità elementare dell'archivio digitale, cioè al singolo oggetto digitale;
- i MAG comprendono una quantità limitata di metadati descrittivi, ereditati dagli applicativi di catalogazione ed espressi in formato Dublin Core (p. 10). Di norma tali metadati descrittivi sono funzionali a operazioni gestionali interne all'archivio degli oggetti digitali;
- di norma, con poche eccezioni, i MAG vanno raccolti o prodotti con procedure automatizzate nel corso del processo di digitalizzazione o trattamento (post-processing) dell'oggetto digitale.

Lo schema risulta composto di diverse sezioni, utilizzabili a seconda del contenuto digitale e dell'impiego dello stesso:

- **GEN**: informazioni generali sul progetto e sul tipo di digitalizzazione
- **BIB**: metadati descrittivi dell'oggetto analogico
- **STRU**: metadati strutturali dell'oggetto digitale
- **IMG**: metadati specifici per le immagini fisse
- **OCR**: metadati specifici relativi al riconoscimento ottico del testo
- **DOC**: metadati specifici per oggetti digitali in formato testo che possono essere **derivati** o **born digital**
- **AUDIO**: metadati specifici per file audio
- **VIDEO**: metadati specifici per file video
- **DIS**: metadati specifici per la distribuzione di oggetti digitali.

Lo schema è aperto, ma si assume che il cuore dei MAG sia quello indicato nel presente Manuale. Utilizzando lo schema è possibile produrre per ogni oggetto digitale un file guida standard che raccoglie tutte le informazioni sull'oggetto medesimo e contiene la mappa di tutti i file generati contestualmente alla digitalizzazione. Si ritiene che ogni sistema o processo di digitalizzazione sia in grado di attrezzarsi per produrre file di questo tipo.

1.4. Interazione con standard internazionali

Essendo un **application profile**, MAG interagisce e interopera con diversi standard internazionali di codifica di metadati. In alcuni casi, infatti, MAG assume altri schemi di codifica (Dublin Core e NISO), in altri casi, invece, può essere trasformato in formati diversi (METS e MPEG-21).

1.4.1. Dublin Core

Sviluppato dall'DCMI (**Dublin Core Metadata Initiative**), il set di marcatori Dublin Core è forse il più diffuso standard di metadati a livello internazionale. Il Dublin Core (da qui in avanti: DC) deve il suo successo da un lato alla sua semplicità (è costituito di solo quindici elementi), dall'altro alla sua estrema flessibilità.

La versione aggiornata del formato DC si caratterizza per: semplicità di utilizzo in quanto si rivolge sia a non catalogatori che a specialisti; interoperabilità semantica stabilendo una comune rete di dati concordati nel loro significato e valore; promozione di uno strumento utile per una infrastruttura a livello internazionale; flessibilità, in quanto permette di integrare e sviluppare la struttura dei dati con significati semantici diversi ed appropriati al contesto di applicazione. Inoltre esso vuole proporsi come una alternativa per il materiale digitale a formati più elaborati di catalogazione presenti nel mondo bibliotecario (ad es. i formati di registrazione MARC). Queste stesse caratteristiche, nella prospettiva della biblioteca digitale lo rendono uno dei formati applicabili alla descrizione di oggetti in differenti tipologie di supporti compreso quello elettronico, a beneficio delle varie comunità che interagiscono continuamente (biblioteche, mondo editoriale, produttori, autori, etc.).⁸

Il sito web dell'organizzazione (<http://dublincore.org/>) è ricco di informazioni e di suggerimenti per l'applicazione dello standard. Da parte sua l'ICCU ha curato la traduzione italiana delle specifiche descrittive dello standard (<http://www.iccu.sbn.it/dublinco.html>).

Lo schema MAG importa il set di elementi Dublin Core tramite il namespace convenzionale `dc:` e li utilizza per la descrizione dell'oggetto analogico alla base della digitalizzazione; per esempio, nel caso della digitalizzazione di un volume, gli elementi DC si occuperanno di registrare i metadati descrittivi relativi al volume cartaceo. Nel caso, invece, di un documento **born digital** – per il quale si veda la sezione DOC (p. 136) –, si occuperà di descrivere la **natura** dell'oggetto bibliografico, facendo semmai riferimento a un eventuale fonte bibliografica del documento tramite l'elemento `<dc:source>`.

1.4.2. NISO e MIX

Il **NISO Technical Metadata for Digital Still Images Standards Committee** ha sviluppato delle linee guida per la creazione di metadati amministrativi e gestionali relativi alle immagini statiche. Tali linee guida sono contenute nel **Data Dictionary – Technical Metadata for Digital Still Images**, il cui obiettivo è così espresso:

The purpose of this data dictionary is to define a standard set of metadata elements for digital images. Standardizing the information allows users to develop, exchange, and interpret digital image files. The dictionary has been designed to facilitate interoperability between systems, services, and software as well as to support the long-term management of and continuing access to digital image collections.⁹

A differenza del DC, lo standard NISO non comprende uno schema di codifica XML, ma si limita a fornire una serie di linee guida per la creazione di linguaggi di markup basati su di esse. Attualmente lo standard non è ancora stato rilasciato formalmente e viene distribuito

8. <http://www.iccu.sbn.it/dublinco.html> .

9. *Data Dictionary – Technical Metadata for Digital Still Images*, <http://www.niso.org/pdfs/DataDict.pdf> p. 1.

in **Trial Use**.¹⁰

Lo schema MAG assume lo standard NISO, sviluppando un linguaggio di markup definito NISO-MAG, incluso al proprio interno tramite un namespace (p. 17) il cui prefisso distintivo è `niso:`. Tale schema, distribuito insieme allo schema MAG, si basa su una precedente versione del **Data Dictionary NISO**, il **Working Draft** del 2000. È stato infatti deciso dal Comitato MAG di attendere una versione stabile di tale standard prima di procedere all'aggiornamento dello schema di codifica; per approfondimenti si veda il paragrafo Lo schema NISO (p.89).

La Library of Congress insieme allo MARC Standards Office ha a sua volta creato uno schema di codifica basato sulle linee guida del **Data Dictionary NISO**, dando vita allo schema di codifica **NISO MIX**, attualmente ancora allo stato di **draft**¹¹. Una futura interoperabilità fra lo schema NISO sviluppato all'interno del progetto MAG e NISO MIX è certamente auspicabile e verrà presa in considerazione non appena tale schema raggiungerà uno status che garantisce una maggior stabilità e durata.

1.4.3. METS

Lo schema METS, sviluppato dalla Library of Congress,¹² è uno standard per codificare metadati descrittivi, amministrativi e strutturali relativi a oggetti all'interno di una biblioteca digitale, esattamente come MAG, ma che, a differenza di MAG, si propone più come “schema contenitore” che come risposta o punto di riferimento per la registrazione di metadati. METS, infatti, può integrare al suo interno diversi schemi di codifica non predeterminati, mentre dà ben poche indicazioni su come e cosa si debba codificare.

Particolarmente vaghe sono le indicazioni circa i metadati tecnici. Per le immagini statiche METS raccomanda, infatti, l'adozione di NISO MIX che, come sappiamo, è al momento ancora in status di **draft**, mentre per la codifica di audio e video non fornisce alcuna indicazione.

Detto questo, METS è sicuramente uno schema con molte potenzialità e che ha incontrato l'interesse di numerose istituzioni in tutto il mondo. È per tale motivo che è stato deciso di pensare a un meccanismo per il quale MAG e METS possano interagire. In particolare, il Comitato MAG ha elaborato un applicativo (basato su un foglio di stile XSLT) in grado di trasformare un file MAG in un file METS, in modo da offrire agli utenti MAG la possibilità di usufruire MAG per le proprie esigenze interne e, in generale, per coordinarsi a progetti che adottano il medesimo schema di codifica; allo stesso tempo però il sistema di conversione garantirà l'interscambiabilità e la condivisione dei dati a livello internazionale.

Si noti che in ogni caso MAG può essere usato come estensione di METS¹³.

1.4.4. MPEG-21

L'ultimo nato fra gli schemi di codifica dei metadati è l'MPEG-21 (ISO/IEC 21000-N):

MPEG-21 aims at defining a normative open framework for multimedia delivery and consumption for

10. Si veda *Data Dictionary—Technical Metadata for Digital Still Images*, 2002, reperibile al sito http://www.niso.org/standards/resources/Z39_87_trial_use.pdf

11. <http://www.loc.gov/standards/mix/>

12. Si veda <http://www.loc.gov/standards/mets/>.

13. Si veda, per esempio, l'application profile realizzato dalla Biblioteca Provinciale di Campobasso che integra MAG all'interno di uno schema METS: <http://web-serv.provincia.campobasso.it/biblioteca/digitale/>

use by all the players in the delivery and consumption chain. This open framework will provide content creators, producers, distributors and service providers with equal opportunities in the MPEG-21 enabled open market.¹⁴

Il cuore dell'MPEG-21 è il concetto di Digital Item. I **digital item** sono oggetti digitali strutturati che includono una rappresentazione standard, un'identificativo e dei metadati. Più concretamente, un Digital Item è costituito di una combinazione di risorse (come uno stream video, tracce audio, immagini statiche, ecc.), metadati (come descrittori, identificativi, ecc.) e strutture (che descrivono le relazioni che intercorrono fra le risorse).

MPEG-21 si occupa di due aspetti fondamentali: la definizione dei requisiti tecnici fondamentali degli oggetti digitali e la possibilità di interazione da parte dell'utente con i medesimi oggetti. Il secondo di questi aspetti investe direttamente i metadati, grazie ai quali è infatti possibile interagire con l'oggetto digitale. A questo proposito, la parte 2 dello standard (ISO/IEC 21000-2:2003) contiene le specifiche di uno schema di codifica per dichiarare la struttura e le caratteristiche dei Digital Item, denominato **DIDL (Digital Item Declaration Language)**, le cui maggiori caratteristiche sono la flessibilità e l'interoperabilità.

È in fase di studio la possibilità di realizzare un tool analogo a quello realizzato per METS per convertire i file MAG in file DIDL oltre a delle linee guida che consentano l'interoperabilità dei formati.

14. <http://www.chiariglione.org/mpeg/standards/mpeg-21/mpeg-21.htm>

2. Introduzione a XML

2.1. Cos'è XML

XML (acronimo per **eXtensible Markup Language**) è un linguaggio di markup sviluppato dal W3C (**World Wide Web Consortium**) nel 1999 con lo scopo di standardizzare la creazione di applicazioni per il Web e per l'interscambio dei dati. Il linguaggio XML ha trovato una sua ideale applicazione sia nel settore dell'editoria digitale che in quello della conservazione e nell'interscambio dei dati.

Un linguaggio di markup è costituito da un insieme di marche o etichette (dette anche **tag**) che servono ad annotare e/o descrivere una caratteristica o la natura di un determinato testo o di un dato. XML consente di creare marche personalizzate che possono descrivere in modo semplice e intuitivo per l'utente finale così come per il computer ogni caratteristica del dato considerato.

La sua semplicità di utilizzo ne ha determinato il grande successo. Infatti, pur essendo un linguaggio molto giovane, è già alla base di numerosissime applicazioni e su XML sono ormai basati diversi progetti di archiviazione e gestione di dati, testuali e non. Inoltre, essendo garantito e sviluppato dal W3C, XML fornisce buone garanzie circa la sua durata e la possibilità di conservare i dati nel lungo periodo.

L'operazione di annotazione dei dati tramite le marche XML si definisce normalmente **marcatura** oppure **codifica**. Poiché la possibilità di creare linguaggi personalizzati può portare alla proliferazione di una babele di diversi schemi di codifica e quindi portare alla mancata condivisione dei dati, molte istituzioni hanno creato i propri schemi di codifica che hanno poi proposto alla comunità internazionale. È quindi possibile adottare uno di questi schemi di codifica e poi codificare i propri dati seguendone le direttive e le specifiche.

Pur essendo altamente flessibile e personalizzabile, il linguaggio XML prevede tre successivi livelli di correttezza, di cui solo il primo è obbligatorio. Un documento codificato in XML può infatti essere:

- **ben formato**: quando ottempera alle specifiche della sintassi XML; è verificato automaticamente dal computer grazie ad analizzatori sintattici (detti **parser**);
- **valido**: quando rispetta i vincoli dello schema di codifica prescelto, nell'eventualità che ne adotti uno; è verificato automaticamente dal computer grazie a speciali analizzatori sintattici (detti **parser validanti**). Un documento valido è anche ben formato; il contrario, invece, non è sempre vero.
- **semanticamente coerente**: quando il documento rispetta la semantica dello schema di codifica prescelto; è verificato dall'uomo.

Se paragoniamo l'insieme dei tag usati per descrivere la natura di un particolare oggetto a un vocabolario con una determinata ortografia (documento ben formato), allora possiamo paragonare uno schema di codifica a una grammatica che definisce i rapporti reciproci dei vari elementi che compongono il lessico del linguaggio, vale a dire la sintassi (documenti validi). Questi due livelli possono essere controllati e verificati direttamente dal computer a cui è possibile insegnare le "regole" di ortografia e di sintassi. Il compito di usare ortografia e sintassi in modo da produrre documenti semanticamente coerenti spetta invece all'uomo. Non c'è infatti modo di spiegare a un computer che è sbagliato marcare il titolo di

un volume come se fosse il titolo di un articolo. Un computer, infatti, si limiterà a controllare che un titolo segua il nome dell'eventuale autore e che preceda le note tipografiche; il resto è compito del codificatore o del revisore della codifica.

I paragrafi che seguono spiegheranno le regole per ottenere documenti XML ben formati e validi, mentre i capitoli successivi spiegheranno la semantica dello schema di codifica MAG. Nei paragrafi che seguono le spiegazioni sono volutamente semplificate in modo da consentire all'utente di comprendere i primi rudimenti della sintassi XML al fine di seguire meglio le spiegazioni e i vincoli relativi a ciascun elemento.

2.2. Sintassi XML

2.2.1. Elementi

L'unità base di XML è l'elemento. L'elemento è un'unità di significato considerata in quanto componente strutturale di un documento. Per esempio, in una descrizione bibliografica, il nome dell'autore costituirà un elemento, così come il titolo e ciascuna delle componenti delle note tipografiche.

Un documento XML può essere descritto come una sequenza organizzata di elementi, tutti racchiusi da un **elemento radice** (o elemento **root**).

Diversi tipi di elementi assumono diversi nomi. Il nome di un elemento è detto **generic identifier** o **GI**.

Per marcare un elemento è necessario introdurre una particolare sequenza di caratteri per segnare l'inizio dell'elemento (detto **start tag**) e un'altra sequenza di caratteri per segnare la fine (detto **end tag**).

Un elemento può essere una stringa di testo circondata da marcatori (tag) come nell'esempio:

```
<elemento>Questo è un elemento</elemento>
```

Qui la codifica `<elemento>` indica l'inizio dell'elemento, e la codifica `</elemento>` ne indica la fine. Ogni elemento è contraddistinto da un tag di inizio, formato dal nome dell'elemento racchiuso fra parentesi angolate (`<NomeElemento>`) e da un tag di chiusura, formato dal nome dell'elemento preceduto da uno slash e racchiuso fra parentesi angolate (`</nomeElemento>`). È importante ricordare che XML è case sensitive, cioè distingue le maiuscole dalle minuscole, per cui scrivere `<NomeElemento>` non è equivalente a `<NOMEELEMENTO>` o a `<nomeelemento>`.

Un elemento è composto sia dai tag che lo delimitano (**start tag** e **end tag**) che dal suo contenuto. Nel caso appena visto il contenuto dell'elemento è solo testo.

Un elemento può anche essere vuoto, il che significa che potrebbe non avere alcun contenuto (per esempio potrebbe semplicemente rappresentare un collegamento a un altro file o a un'immagine). La sintassi di un elemento vuoto può assumere due forme:

- nome dell'elemento seguito da uno slash, racchiuso da parentesi angolate:
`<elementoVuoto />`
- start tag seguito da end tag non inframmezzati da nessun carattere, nemmeno da uno spazio: `<elementoVuoto> </elementoVuoto>`.

Un elemento può essere usato più volte all'interno dello stesso documento; tecnicamente si dice che ogni volta che viene usato, l'elemento è istanziato nel documento.¹⁵

Un documento XML può consistere in una serie di elementi l'uno dopo l'altro (racchiusi dall'elemento radice); ad esempio:

```
<radice>
  <titolo> Questo è un elemento titolo</titolo>
  <p>Questo è un paragrafo</p>
  <nota>Questa è una nota</nota>
</radice>
```

Oppure, ed è il caso più frequente, gli elementi possono essere annidati gli uni dentro gli altri; per esempio:

```
<radice>
  <capitolo>Questo è un capitolo,
    <titolo> Questo è un elemento titolo</titolo>
    <p>Questo è un paragrafo</p>
  </capitolo>
</radice>
```

Le relazioni tra gli elementi in XML sono essenzialmente relazioni gerarchiche e ordinali. XML si basa su una struttura fortemente gerarchica in cui non esistono sovrapposizioni tra elementi. Questo significa che la struttura astratta di un documento XML deve essere rappresentabile mediante un grafo ad albero (simile a un albero genealogico) in cui a ciascun nodo corrisponde un elemento e a ogni ramo verso il basso una relazione di inclusione.

Vediamo ora un esempio di una citazione bibliografica:

```
<citazione>
  <autore>Carlo Dionisotti</autore>
  <titolo>Geografia e storia della letteratura italiana</titolo>
  <editore>Einaudi</editore>
  <data>1967</data>
  <soggetto/>
</citazione>
```

Così come si trova, l'esempio precedente è un documento XML ben-formato. Per raggiungere questo stato il documento ha dovuto sottostare alle seguenti regole:

- deve esistere un singolo elemento (compreso fra uno start e un end tag) che racchiuda l'intero documento: tale elemento assume il nome di elemento root (<citazione> nel

15. Se si adotta un particolare schema di codifica, sarà compito di tale schema stabilire quante volte l'elemento potrà essere usato e in quali circostanze.

- nostro caso) o elemento radice;
- tutti gli elementi devono essere racchiusi all'interno dell'elemento root o all'interno di elementi a loro volta racchiusi nell'elemento root; gli elementi non possono intrecciarsi (**overlapping**) gli uni agli altri. Vale a dire che se l'elemento B è viene aperto all'interno dell'elemento A, B deve essere chiuso prima che si chiuda A;
- il tag che indica l'inizio e la fine dell'elemento deve essere sempre presente;
- Gli elementi vuoti devono essere correttamente chiusi o tramite uno slash all'interno dell'unico tag (<oggetto/>), oppure senza introdurre alcun carattere fra lo start tag e l'end tag.

Per definire i rapporti reciproci fra gli elementi, si usa normalmente la metafora dei rapporti di parentela. Quando un elemento è contenuto direttamente dentro un altro, si dice che quell'elemento è **figlio** dell'altro. Nel nostro esempio l'elemento `autore` è figlio dell'elemento `citazione`; analogamente, `citazione` è **genitore** di `autore`. Se un elemento è figlio di un figlio di un altro elemento, si dice che ne è il **discendente**, mentre l'elemento da cui discende è detto **antenato**. Supponendo che l'elemento `autore` abbia due figli `nome` e `cognome`, potremmo dire che `cognome` è discendente di `citazione` e che `citazione` è l'antenato di `cognome`.

2.2.2. Attributi

Gli attributi servono a specificare meglio le caratteristiche degli elementi e quindi dei dati. Gli attributi sono costituiti da un nome e da un valore legato al nome da un segno di uguale e racchiuso fra virgolette secondo la sintassi:

```
nome="valore"
```

Tramite gli attributi è possibile per esempio descrivere un particolare status di un elemento o dare maggiori informazioni relativamente all'elemento stesso. Per esempio possiamo dire se una particolare citazione riguarda una monografia oppure il contributo di una miscellanea:

```
<citazione tipo="monografia">  
  <autore>Carlo Dionisotti</autore>  
  <titolo>Geografia e storia della letteratura italiana</titolo>  
  <editore>Einaudi</editore>  
  <data>1967</data>  
  <oggetto/>  
</citazione>
```

oppure potremmo usare gli attributi per dire in che data è stato generato un particolare dato e da chi.

L'attributo deve essere dichiarato all'interno dello start tag (non può invece essere inserito nell'end tag) o in un elemento vuoto.

È importante ricordare che XML richiede per essere ben formato che i valori degli attributi siano necessariamente racchiusi fra virgolette semplici o doppie, a scelta, con

l'obbligo della coerenza, vale a dire che un valore può essere racchiuso fra virgolette semplici o doppie, e non una semplice e una doppia

2.2.3. Commenti

A volte può essere utile utilizzare commenti nei file XML, brani che vengono inseriti arbitrariamente dal codificatore allo scopo di annotare una caratteristica non strutturale o per ricordare qualcosa. I commenti XML sono identici ai commenti HTML. È possibile usare i commenti per includere note esplicative nei documenti che verranno ignorate da un parser XML. I commenti possono apparire ovunque in un documento, ma non all'interno dei tag. Come in HTML iniziano con la stringa `<!--` e finiscono con `-->`, come nell'esempio:

```
<citazione tipo="monografia">
  <autore>Carlo Dionisotti</autore>
  <titolo>Geografia e storia della letteratura italiana</titolo>
  <editore>Einaudi</editore>
  <data>1967</data>
  <!-- attenzione: ricordarsi di inserire il soggetto -->
  <soggetto/>
</citazione>
```

2.3. Namespaces

A volte è necessario poter usare dentro al proprio documento XML alcuni elementi che hanno una particolare provenienza, vale a dire che sono definiti in un qualche schema di codifica. XML fornisce uno strumento per andare incontro a questa esigenza chiamato **namespace** (la traduzione italiana – spazio nome – non è quasi mai usata).

Un namespace è costituito da un prefisso distintivo. Per usare un namespace è necessario dichiararlo all'interno di un qualsiasi elemento di un documento XML, usando un attributo standard disponibile per tutti gli elementi XML, qualsiasi essi siano. L'attributo si chiama `xmlns` (abbreviazione di XML NameSpace) e deve essere seguito da un due punti e dal prefisso distintivo del namespace da adottare.

Poniamo, per esempio, di voler usare alcuni degli elementi definiti dallo schema Dublin Core per codificare la citazione bibliografica dell'esempio visto in precedenza. Per farlo basterà usare il namespace `dc` nel modo che segue:

```
<citazione xmlns:dc="http://purl.org/dc/elements/1.1/">
  <dc:creator>Carlo Dionisotti</dc:creator>
  <dc:title>Geografia e storia della letteratura italiana</dc:title>
  <dc:publisher>Einaudi</dc:publisher>
  <dc:date>1967</dc:date>
  <dc:subject/>
</citazione>
```

La dichiarazione di un namespace avviene grazie a un attributo standard speciale,

previsto dalle specifiche XML, `xmlns:prefisso` (nel nostro esempio `xmlns:dc`) il cui valore è dato dalla URL dove si trova la documentazione dello schema cui il namespace fa riferimento (nel nostro esempio `http://purl.org/dc/elements/1.1/`).

Si noti che un namespace può essere dichiarato in un qualsiasi luogo del documento XML, ma comunque prima che tale namespace venga usato per la prima volta. È comunque buona norma dichiararlo all'interno dell'elemento root.

2.4. Sintassi degli schema W3C

Attualmente ci sono diversi modi per dichiarare uno schema di codifica, vale a dire una sorta di grammatica che definisca quali elementi sarà possibile usare all'interno di un documento XML e in che modo tali elementi si potranno relazionare fra di loro. Tale grammatica infatti dovrà definire quali elementi potranno essere contenuti da altri elementi, quante volte si potranno usare tali elementi, e se potranno o meno avere degli attributi, e se sì, quali potranno essere e se dovranno o meno avere un certo contenuto.

Il modo più conosciuto per dichiarare uno schema di codifica è attraverso una Document Type Definition (o DTD), una sintassi ereditata direttamente dal linguaggio SGML,¹⁶ da cui l'XML deriva. La DTD è costituita da una sintassi estremamente semplice e facile, ma che presenta notevoli limitazioni, soprattutto per quanto riguarda la definizione delle diverse tipologie di elementi e delle occorrenze degli stessi.

Recentemente a questo metodo tradizionale si sono aggiunti gli schema (da leggersi all'inglese, **schima**), una raccomandazione del W3C contenuta in nuce fin dalla prima versione del linguaggio XML del 1999. Esistono attualmente diversi tipi di schema che, con una sintassi diversificata, giungono più o meno gli stessi risultati e che rappresentano un superamento dei limiti posti dalle DTD. I più usati sono sicuramente Relax NG¹⁷ e gli schemi W3C¹⁸.

Lo schema di codifica MAG adotta le specifiche W3C la cui sintassi di base verrà di seguito brevemente presentata. Tale presentazione segue da vicino la documentazione fornita dal W3C (<http://www.w3.org/TR/xmlschema-0/>) allo scopo di permettere agevolmente ai lettori interessati di approfondire la loro conoscenza in materia sfruttando le nozioni apprese in questo manuale.

2.4.1. Un esempio: il file `bibliografia.xml` e il suo schema

Per cominciare l'esposizione degli schema W3C, prendiamo come esempio un documento XML che descrive una semplice bibliografia.

```
<?xml version="1.0" encoding="UTF-8"?>
<bibliografia>
  <citazione id="cit1">
    <autore tipo="contributo">I. BISCEGLIA BONOMI</autore>
```

16. **Standard Generalised Markup Language**, uno standard formale (ISO 8879:1986) da cui derivano in modo più meno diretto i maggiori linguaggi di markup come XML, appunto, e HTML.

17. Si veda: J. Clark e M. Makoto. *Relax NG Specification*, December 2001, <http://www.oasis-open.org/committees/relax-ng/spec-20011203.html> .

18. Si veda l'ampia documentazione fornita dal W3C relativamente alla sintassi degli schema: <http://www.w3.org/TR/xmlschema-0/> .

```
<titolo>La grammatica di Pierfrancesco Giambullari: saggio di un'analisi delle forme
verbalì del fiorentino vivo</titolo>
<volume>
  <titolo>Rinascimento: aspetti e problemi attuali</titolo>
  <noteTipografiche tipo="miscellanea">
    <luogoEd>Firenze</luogoEd>
    <editore>Olschki</editore>
    <annoEd>1982</annoEd>
  </noteTipografiche>
  <pag>231-42</pag>
</volume>
</citazione>
<citazione id="cit2">
  <autore tipo="articolo">L. SERVOLINI</autore>
  <titolo>Glorie dell'arte tipografica italiana. Francesco Marcolini da Forlì</titolo>
  <rivista>
    <nome>Accademie e biblioteche d'Italia</nome>
    <annata>XV 1940</annata>
    <pag>15-21</pag>
  </rivista>
</citazione>
<citazione id="cit3">
  <autore tipo="monografia">C. DI FILIPPO BAREGGI</autore>
  <titolo>Il mestiere di scrivere. Lavoro intellettuale e mercato librario a Venezia
nel Cinquecento</titolo>
  <noteTipografiche>
    <luogoEd>Roma</luogoEd>
    <editore>Bulzoni</editore>
    <annoEd>1988</annoEd>
  </noteTipografiche>
</citazione>
</bibliografia>
```

La bibliografia è costituita da un elemento principale bibliografia, e di alcuni sottoelementi come titolo, autore, rivista, ecc. Alcuni di questi, a loro volta contengono ulteriori sottoelementi e, eventualmente, attributi; altri contengono solo testo. Gli elementi che contengono sottoelementi o attributi sono di **tipo complesso**, mentre gli elementi che non contengono alcun sottoelemento o attributi sono di **tipo semplice**.

Lo schema che rende valido il precedente documento è il seguente:

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:annotation>
    <xsd:documentation xml:lang="it"> Schema per la creazione di bibliografie
  </xsd:documentation>
</xsd:annotation>
```

```
<xsd:element name="bibliografia">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="citazione" type="Citazioni" maxOccurs="unbounded" />
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

```
<xsd:complexType name="Citazioni">
  <xsd:sequence>
    <xsd:element name="autore" minOccurs="0">
      <xsd:complexType>
        <xsd:simpleContent>
          <xsd:extension base="xsd:string">
            <xsd:attribute name="tipo" use="required">
              <xsd:simpleType>
                <xsd:restriction base="xsd:string">
                  <xsd:enumeration value="contributo" />
                  <xsd:enumeration value="monografia" />
                  <xsd:enumeration value="articolo" />
                </xsd:restriction>
              </xsd:simpleType>
            </xsd:attribute>
          </xsd:extension>
        </xsd:simpleContent>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="titolo" type="xsd:string" />
    <xsd:choice>
      <xsd:element name="noteTipografiche" type="NTipo" />
      <xsd:element name="rivista" type="Periodici" />
      <xsd:element name="volume" type="Miscellanea" />
    </xsd:choice>
  </xsd:sequence>
  <xsd:attribute name="id" type="xsd:ID" use="required" />
</xsd:complexType>
```

```
<xsd:complexType name="NTipo">
  <xsd:sequence>
    <xsd:element name="luogoEd" type="xsd:string" />
    <xsd:element name="editore" type="xsd:string" />
    <xsd:element name="annoEd" type="xsd:gYear" />
  </xsd:sequence>
  <xsd:attribute name="tipo" use="optional" default="monografia">
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:enumeration value="monografia" />
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:attribute>
</xsd:complexType>
```

```

        <xsd:enumeration value="miscellanea"/>
    </xsd:restriction>
</xsd:simpleType>
</xsd:attribute>
</xsd:complexType>

<xsd:complexType name="Periodici">
    <xsd:sequence>
        <xsd:element name="nome" type="xsd:string"/>
        <xsd:element name="annata" type="xsd:string"/>
        <xsd:element ref="pag"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="Miscellanea">
    <xsd:sequence>
        <xsd:element name="titolo" type="xsd:string"/>
        <xsd:element name="noteTipografiche" type="NTipo"/>
        <xsd:element ref="pag"/>
    </xsd:sequence>
</xsd:complexType>

<xsd:element name="pag" type="paginazione"/>

<xsd:simpleType name="paginazione">
    <xsd:restriction base="xsd:string">
        <xsd:pattern value="\d{1,4}-\d{1,4}"/>
    </xsd:restriction>
</xsd:simpleType>

</xsd:schema>

```

Lo schema bibliografia è costituito da un elemento `xsd:schema` e da una serie di sottoelementi, di cui i più importanti sono `xsd:element`, `xsd:complexType` e `xsd:simpleType`. Ciascuno degli elementi dello schema ha il prefisso `xsd:` che è associato con il namespace degli XML Schema grazie alla dichiarazione iniziale contenuta all'interno dell'elemento root `xmlns:xsd="http://www.w3.org/2001/XMLSchema"`. Il prefisso `xsd:` è un prefisso convenzionale per denotare gli schema W3C, ma in alternativa avrebbe potuto essere scelto un qualsiasi altro prefisso. L'associazione del prefisso aiuta a distinguere quali elementi e tipi sono definiti all'interno degli schema W3C e quali invece sono definiti dall'utente.

2.4.2. Tipi complessi

I tipi complessi, che possono contenere elementi e attributi, sono definiti grazie all'elemento `xsd:complexType`, mentre gli elementi sono definiti grazie all'elemento `xsd:element`. Per definire un elemento che ne contiene altri è necessario:

- dichiarare un `xsd:complexType` cui si dà un determinato nome attraverso l'attributo `name`
- dichiarare un elemento dichiarando (attraverso l'attributo `type`) che il suo tipo è quello definito da tale `xsd:complexType`

```
<xsd:element name="noteTipografiche" type="NTipo"/>

<xsd:complexType name="NTipo">
  <xsd:sequence>
    <xsd:element name="luogoEd" type="xsd:string"/>
    <xsd:element name="editore" type="xsd:string"/>
    <xsd:element name="annoEd" type="xsd:gYear"/>
  </xsd:sequence>
  <xsd:attribute name="tipo" use="optional" default="monografia"/>
</xsd:complexType>
```

Nell'esempio abbiamo definito un elemento `e`, tramite l'attributo `name`, abbiamo stabilito che si chiama `noteTipografiche`; grazie all'attributo `type` abbiamo detto che tale elemento sarà di tipo `NTipo`. Successivamente (ma l'ordine è indifferente) abbiamo definito il tipo complesso `NTipo` e abbiamo detto che tutti gli elementi di tipo `NTipo` avranno al loro interno una sequenza di tre elementi che si chiamano rispettivamente `luogoEd`, `editore` e `annoEd` che dovranno comparire tutti obbligatoriamente nell'ordine specificato grazie all'elemento `xsd:sequence`. Tutti gli elementi di tipo `NTipo`, inoltre, potranno avere un attributo opzionale che si chiama `tipo`. Abbiamo anche stabilito che il contenuto degli elementi `luogoEd` e `editore` sarà costituito da una semplice stringa di caratteri, usando il tipo `xsd:string`, viceversa il contenuto di `annoEd` è stato definito come `xsd:gYear` e cioè potrà contenere solo l'indicazione di un anno.

A volte è necessario usare più volte un elemento già definito altrove come nell'esempio:

```
<xsd:element ref="pag"/>
```

La dichiarazione fa riferimento a un elemento `pag` che è già stato definito da qualche altra parte nello schema. In generale l'attributo `ref` dovrebbe far riferimento a un **elemento globale**, vale a dire un elemento che è stato definito direttamente come figlio dell'elemento `xsd:schema` piuttosto che come figlio di un `complexType`.

La dichiarazione di elementi globali deve essere fatto con estrema cautela perché ciascun elemento globale, a rigore, potrebbe essere usato come elemento `root`, con la conseguente anarchia dello schema di codifica.

2.4.3. Occorrenze degli elementi e degli attributi

L'elemento `citazione` potrà essere ripetuto un numero indeterminato di volte grazie al valore (`unbounded`) dell'attributo `maxOccurs`, ma dovrà essere usato almeno una volta. Non avrebbe infatti senso una bibliografia senza almeno una citazione al suo interno. Se avessimo

voluto renderlo opzionale, avremmo dovuto dichiarare l'attributo `minOccurs` con valore 0, come difatti avviene per l'elemento `autore`.

```
<xsd:element name="citazione" type="Citazioni" maxOccurs="unbounded"/>
```

```
<xsd:element name="autore" minOccurs="0">
```

Il valore di default degli attributi `minOccurs` e `maxOccurs` è 1, il che significa che tutti gli elementi che vengono dichiarati devono necessariamente essere usati una e una sola volta, a meno che non venga espresso un valore diverso. Il valore di `maxOccurs` può essere espresso, oltre che con il termine `unbounded` anche con un valore numerico tipo "16" o "54".

Gli attributi possono essere usati una sola volta all'interno dello stesso elemento, ma è possibile dichiarare se il loro uso è obbligatorio o opzionale, e, nel caso siano opzionali, se esista un valore di default. L'attributo `tipo` del tipo complesso `NTipo` è definito, grazie all'attributo `use` come opzionale; se avessimo voluto che fosse obbligatorio avremmo dovuto usare il termine `required`:

```
<xsd:attribute name="tipo" use="optional" default="monografia"/>
```

Lo stesso attributo ha anche un valore di default, vale a dire `monografia` dichiarato grazie all'attributo `default` che non avremmo potuto usare se avessimo dichiarato che l'attributo era `required`.

2.4.4. Tipi semplici

Esistono due tipologie di tipi semplici: quelli definiti direttamente dal W3C come parte dello schema di codifica, e quelli definiti dall'utente. Fanno parte dei tipi definiti dal W3C quelli che appartengono al namespace `xsd:`. Nel nostro esempio troviamo:

- `xsd:string`: una qualsiasi stringa di caratteri, compresi spazi e segni di punteggiatura
- `xsd:gYear`: un anno solare
- `xsd:ID`: un identificatore univoco che deve necessariamente cominciare con una lettera (non un numero) e non può contenere spazi o segni di punteggiatura ad eccezione del trattino, del trattino basso e del punto.

Altri tipi particolarmente comuni sono `xsd:date`, che definisce una data in un particolare formato standardizzazione, e `xsd:anyURI` che contiene una URI (ULR, URN) che punta a un particolare oggetto digitale.¹⁹

È possibile definire nuovi tipi semplici derivandoli da un tipo semplice esistente, vale a dire definendo un sottoinsieme di possibili valore fra quelli potenzialmente possibili. Per definire un tipo semplice si usa l'elemento `simpleType`, mentre per definire un sottoinsieme di valori fra quelli previsti da un tipo semplice si usa `xsd:restriction`.

19. Per gli altri tipi vedi <http://www.w3.org/TR/xmlschema-2/>.

```
<xsd:simpleType name="paginazione">  
  <xsd:restriction base="xsd:string">  
    <xsd:pattern value="\d{1,4}-\d{1,4}"/>  
  </xsd:restriction>  
</xsd:simpleType>
```

Nell'esempio abbiamo creato un nuovo tipo semplice chiamato `paginazione` definito come sottoinsieme del tipo semplice `xsd:string`. Abbiamo ipotizzato che la paginazione di un contributo o di un articolo assumerà sempre la forma `paginaIniziale-paginaFinale`; per poter controllare che i dati immessi all'interno di questo elemento seguano sempre tale modello abbiamo usato l'elemento `xsd:pattern` il cui attributo `value` descrive il formato secondo una sintassi particolare (desunta dalle cosiddette **espressioni regolari**)²⁰. Nel nostro esempio, la sintassi proposta significa che il tipo `paginazione` sarà costituito da un numero variabile di caratteri (da uno a quattro), seguiti da un trattino e da un altro blocco di caratteri (da uno a quattro).

Un'altra tipologia di restrizioni è quella definita dall'elemento `xsd:enumeration` che serve ad elencare la rosa dei possibili valori. Tale possibilità è stata usata per definire i possibili valore dell'attributo `tipo`:

```
<xsd:simpleType>  
  <xsd:restriction base="xsd:string">  
    <xsd:enumeration value="monografia"/>  
    <xsd:enumeration value="miscellanea"/>  
  </xsd:restriction>  
</xsd:simpleType>
```

2.4.5. Tipi anonimi

Se si presume che un tipo semplice o un tipo complesso sarà usato solo una volta, è possibile definire **tipi anonimi** (o **non-nominati**), tanto semplici che complessi, come figli o discendenti di un altro elemento, evitando così di moltiplicare i nomi che devono essere univoci per evitare ambiguità.

```
<xsd:complexType name="NTipo">  
  <xsd:sequence>  
    <xsd:element name="luogoEd" type="xsd:string"/>  
    <xsd:element name="editore" type="xsd:string"/>  
    <xsd:element name="annoEd" type="xsd:gYear"/>  
  </xsd:sequence>  
  <xsd:attribute name="tipo" use="optional" default="monografia">  
    <xsd:simpleType>
```

20. Si tratta di un liguaggio che serve a rappresentare in modo formalizzato una qualsiasi sequenza di caratteri. Le espressioni regolari usate dagli schemi W3C derivano in massima parte da quelle usate all'interno del linguaggio di programmazione Perl.


```

    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="monografia"/>
      <xsd:enumeration value="miscellanea"/>
    </xsd:restriction>
  </xsd:simpleType>
</xsd:attribute>
</xsd:complexType>

```

Nell'esempio, i valori dell'attributo `tipo` sono definiti grazie a un `xsd:simpleType` privo di nome, figlio dell'elemento `xsd:attribute`.

2.4.6. Tipi complessi derivati da tipi semplici

Si consideri ora l'elemento `autore` del documento `bibliografia.xml`

```
<autore tipo="articolo">L. SERVOLINI</autore>
```

Come si vede il suo contenuto è solo testo e quindi dovrebbe essere di tipo semplice, ma `autore` ha anche un attributo che quindi lo porta a essere un tipo complesso. Per poter definire un tale elemento all'interno dello schema, sarà necessario estendere un tipo semplice in modo tale da fargli contenere un attributo:

```

<xsd:element name="autore" minOccurs="0">
  <xsd:complexType>
    <xsd:simpleContent>
      <xsd:extension base="xsd:string">
        <xsd:attribute name="tipo" use="required">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:enumeration value="contributo"/>
              <xsd:enumeration value="monografia"/>
              <xsd:enumeration value="articolo"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:attribute>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>
</xsd:element>

```

Come si vede, è stato definito prima un tipo complesso anonimo come figlio dell'elemento `xsd:element`; al suo interno si è inserito l'elemento `xsd:simpleContent` contenente a sua volta l'elemento `xsd:extension` il cui attributo `base` definisce quale tipo semplice dovrà essere usato per il contenuto dell'elemento (`xsd:string`, nel nostro caso). All'interno di `xsd:extension` abbiamo usato l'elemento `xsd:attribute` il cui valore è

definito a sua volta come un tipo semplice costituito da una enumerazione di valori.

2.4.7. Scelte fra più elementi

Talvolta potrebbe essere utile specificare che il contenuto di un elemento potrà essere di un tipo oppure di un altro a seconda dei casi. Per esempio nelle citazioni il contenuto cambia notevolmente a seconda se si tratti della citazione di una monografia, dell'articolo di un periodico oppure del contributo di una rivista. In questi casi si possono creare delle scelte attraverso l'elemento `xsd:choice`:

```
<xsd:complexType name="Citazioni">
  <xsd:sequence>
    <xsd:element name="autore" minOccurs="0"/>
    <xsd:element name="titolo" type="xsd:string"/>
    <xsd:choice>
      <xsd:element name="noteTipografiche" type="NTipo"/>
      <xsd:element name="rivista" type="Periodici"/>
      <xsd:element name="volume" type="Miscellanea"/>
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>
```

La dichiarazione `choice` significa che solo una delle opzioni offerte potrà essere scelta di volta in volta, poiché una citazione potrebbe riguardare una monografia (nel qual caso si userebbe l'opzione data dall'elemento `noteTipografiche`) o l'articolo di un periodico (nel qual caso si userebbe l'opzione `rivista`) o il contributo di un volume miscelaneo (nel qual caso si userebbe l'opzione `volume`), ma mai due di queste opzioni insieme.

2.4.8. Le annotazioni

Gli schemi W3C comprendono alcuni elementi specializzati per inserire annotazioni all'interno degli schemi in modo da facilitare la comprensione degli stessi sia da parte degli utenti umani, sia da parte degli elaboratori automatici che potranno usare tali annotazioni per creare documentazioni automatiche.

Nella bibliografia usata come esempio è stata inserita una breve descrizione dell'obiettivo per il quale è stato elaborato lo schema; tale descrizione è stata inserita all'interno dell'elemento `xsd:documentation` a sua volta contenuto dentro a `xsd:annotation`. È stato inoltre usato l'attributo standard XML `xml:lang` (come da raccomandazione W3C) per dichiarare in quale lingua è scritta la documentazione

```
<xsd:annotation>
  <xsd:documentation xml:lang="it"> Schema per la creazione di bibliografie
</xsd:documentation>
</xsd:annotation>
```

L'elemento `xsd:annotation` può essere usato in qualsiasi punto dello schema. È naturalmente possibile usare dei commenti, ma `xsd:annotation` è consigliato qualora il

commento da inserire serve a dare informazioni circa lo schema e il suo utilizzo.

```
<xsd:annotation>
  <xsd:documentation xml:lang="it"> Schema per la creazione di bibliografie
    <!-- ricordarsi di spiegare meglio questo punto -->
  </xsd:documentation>
</xsd:annotation>
```

2.4.9. Target namespace e oggetti locali non qualificati o qualificati

Uno schema può essere visto come un vocabolario di tipi e dichiarazioni di elementi appartenenti a un particolare namespace chiamato **target namespace**. La dichiarazione di un target namespace consentirà a un processore XML di sapere quale schema usare per validare i documenti che vengono prodotti sulla base di tale schema.

Vediamo ora una nuova versione dello schema `biblio.xsd` nel quale viene esplicitamente dichiarato un target namespace e dove si dice che comunque non è necessario che gli elementi e gli attributi appartenenti a tale namespace dichiarino esplicitamente la loro appartenenza, non devono cioè essere **qualificati**

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns="http://www.iccu.sbn.it/biblio.xsd"
  targetNamespace="http://www.iccu.sbn.it/biblio.xsd"
  elementFormDefault="unqualified"
  attributeFormDefault="unqualified">

<!-- qui lo schema -->

</xsd:schema>
```

Come si vede, all'elemento `xsd:schema` sono stati aggiunti alcuni attributi:

- `xmlns` senza alcun prefisso di namespace, il cui valore è costituito dalla URI del namespace di riferimento
- `targetNamespace` fornisce la URI del target namespace
- `elementFormDefault` dice che gli elementi appartenenti al target namespace non devono dichiarare la loro appartenenza tramite un prefisso
- `attributeFormDefault` dice che gli attributi appartenenti al target namespace non devono dichiarare la loro appartenenza tramite un prefisso

Quando si creano dei documenti basati su tale schema, l'elemento root dovrà assumere la seguente forma:

```
<?xml version="1.0" encoding="UTF-8"?>
<bibliografia xmlns="http://www.iccu.sbn.it/biblio.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
xsi:schemaLocation="http://www.iccu.sbn.it/biblio.xsd">
```

```
<!-- qui il documento -->
```

```
</bibliografia>
```

A volte può essere necessario qualificare gli elementi e/o gli attributi del target namespace (per esempio: qualora si voglia diffondere i propri documenti all'esterno della propria istituzione). In tal caso sarà necessario dichiarare il valore "qualified" per gli elementi e/o gli attributi; il risultato sarà che tutti gli elementi (normalmente gli attributi vengono di rado qualificati) dovranno avere necessariamente il prefisso distintivo del namespace; sarà inoltre necessario far seguire l'attributo `xmlns` dal prefisso del target namespace (es: `xmlns:bib=""`).

```
<bib:bibliografia xmlns:bib="http://www.iccu.sbn.it/biblio.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.iccu.sbn.it/biblio.xsd ">
  <bib:citazione id="cit1">
    <bib:autore tipo="monografia">I. BISCEGLIA BONOMI</bib:autore>

    <!-- ecc. -->

  </bib:citazione>

  <!-- ecc. -->
</bibliografia>
```

2.4.10. Schemi suddivisi in più file e importazione di altri schemi

A volte può essere utile suddividere uno schema su più file, specie nel caso di schemi particolarmente complessi o lunghi. In questi casi si può usare l'elemento `xsd:include` il cui attributo `schemaLocation` indica la URI del file in cui è conservata la parte dello schema da includere in quel punto.

```
<xsd:include schemaLocation="biblio2.xsd"/>
```

Supponiamo invece di voler usare all'interno del nostro schema alcuni elementi provenienti dallo schema Dublin Core e affiancarli alla nostra descrizione bibliografica, mantenendo l'indicazione della loro provenienza. In questo caso è necessario importare sia lo schema che il namespace tramite l'elemento `xsd:import`.

Tornando al nostro esempio, per prima cosa sarà necessario dichiarare il namespace del Dublin Core all'interno dell'elemento `root`:

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
            xmlns="http://www.iccu.sbn.it/biblio.xsd"
            targetNamespace="http://www.iccu.sbn.it/biblio.xsd"
            xmlns:dc="http://purl.org/dc/elements/1.1/"
            elementFormDefault="qualified"
            attributeFormDefault="unqualified" >

    <!-- ecc. -->

<xsd:schema>
```

Successivamente bisogna importare lo schema DC:

```
<xsd:import namespace="http://purl.org/dc/elements/1.1/"
            schemaLocation="./simpledc20020312.xsd"/>
```

A questo punto sarà possibile usare gli elementi Dublin Core all'interno dello schema. Per farlo, modificheremo il contenuto dell'elemento <citazione> per inserire un nuovo elemento opzionale che chiameremo dc di tipo DC_Type, poi dichiareremo il tipo complesso DC_Type al cui interno dichiareremo gli elementi Dublin Core.

```
<xsd:complexType name="Citazioni">
  <xsd:sequence>
    <xsd:element name="autore" minOccurs="0">
      <xsd:complexType>
        <xsd:simpleContent>
          <xsd:extension base="xsd:string">
            <xsd:attribute name="tipo" use="required">
              <xsd:simpleType>
                <xsd:restriction base="xsd:string">
                  <xsd:enumeration value="contributo"/>
                  <xsd:enumeration value="monografia"/>
                  <xsd:enumeration value="articolo"/>
                </xsd:restriction>
              </xsd:simpleType>
            </xsd:attribute>
          </xsd:extension>
        </xsd:simpleContent>
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="titolo" type="xsd:string" />
    <xsd:choice>
      <xsd:element name="noteTipografiche" type="Typo"/>
      <xsd:element name="rivista" type="Periodici"/>
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>
```

```
        <xsd:element name="volume" type="Miscellanea"/>
    </xsd:choice>
< -- modifica -->
    <xsd:element name="dc" type="DC_Type" maxOccurs="unbounded"/>
</xsd:sequence>
    <xsd:attribute name="id" type="xsd:ID" use="required"/>
</xsd:complexType>

<xsd:complexType name="DC_Type">
    <xsd:sequence>
        <xsd:element ref="dc:identifier" maxOccurs="unbounded"/>
        <xsd:element ref="dc:title" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="dc:creator" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="dc:publisher" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="dc:subject" minOccurs="0" maxOccurs="unbounded"/>
        <!-- altri elementi Dublin Core -->
    </xsd:sequence>
</xsd:complexType>
```

In virtù di tali modifiche sarà possibile usare nei documenti XML basati sullo schema `biblio.xsd` gli elementi Dublin Core all'interno dell'elemento `dc`:

```
<bibliografia xmlns="http://http://www.iccu.sbn.it/biblio.xsd"
              xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              xsi:schemaLocation="http://www.iccu.sbn.it/biblio.xsd"
              xmlns:dc="http://purl.org/dc/elements/1.1/">
  <citazione id="cit1">
    <autore tipo="contributo">I. BISCEGLIA BONOMI</autore>
    <titolo>La grammatica di Pierfrancesco Giambullari: saggio di un'analisi delle forme
      verbali del fiorentino vivo</titolo>
    <volume>
      <titolo>Rinascimento: aspetti e problemi attuali</titolo>
      <noteTipografiche tipo="miscellanea">
        <luogoEd>Firenze</luogoEd>
        <editore>Olschki</editore>
        <annoEd>1982</annoEd>
      </noteTipografiche>
      <pag>231-42</pag>
    </volume>
    <dc>
      <dc:title>La grammatica di Pierfrancesco Giambullari: saggio di un'analisi delle
        forme verbali del fiorentino vivo</dc:title>
      <dc:creator>I. BISCEGLIA BONOMI</dc:creator>
    </dc>
  </citazione>
</bibliografia>
```

L'importazione di un altro schema di codifica fa di `biblio.xsd` un **application profile**, vale a dire uno schema costituito da elementi recuperati da uno o più namespace combinati insieme dagli implementatori e ottimizzati per un particolare applicazione²¹.

21. R. Heery and M. Patel. *Application profiles: mixing and matching metadata schemas*,
<http://http://www.ariadne.ac.uk/issue25/app-profiles/>

3. Struttura dello schema e dei documenti MAG

3.1. Struttura e componenti dello schema MAG

Lo schema MAG è un **application profile**, vale a dire che integra elementi provenienti da più namespace. Nel dettaglio, lo schema è composto di quattro file (`metadigit.xsd`, `metatype.xsd`, `audio.xsd` e `video.xsd`), collegati gli uni agli altri mediante a un meccanismo di inclusione, e di quattro diversi namespace:

- **mag**: il target namespace
- **niso**: schema che traduce le linee guida del Data Dictionary NISO; è stato realizzato dal Comitato MAG. Il file che lo contiene (`niso-mag.xsd`) è distribuito assieme alle altre componenti dello schema MAG
- **dc**: schema Dublin Core; il file che lo contiene è distribuito direttamente dal Dublin Core Metadata Initiative (DCMI) e, per comodità, assieme alle altre componenti dello schema MAG
- **xlink**: schema W3C comprendente una serie di attributi che consentono di creare collegamenti fra diverse risorse

Lo schema prevede che gli elementi siano qualificati, a differenza degli attributi che sono non qualificati.

3.1.1. Il file `metadigit.xsd`

Il file `metadigit.xsd` è il file principale dello schema MAG. Esso contiene principalmente la dichiarazione dell'elemento root dello schema MAG, vale a dire `metadigit`, l'unico elemento globale definito all'interno dello schema, il cui contenuto è dichiarato grazie a un tipo complesso anonimo.

```
<xsd:element name="metadigit">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="gen" type="gen"/>
      <xsd:element name="bib" type="bib"/>
      <xsd:element name="stru" type="stru" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="img" type="img" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="audio" type="audio" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="video" type="video" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="ocr" type="ocr" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="doc" type="doc" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="dis" type="dis" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="version" type="xsd:string" default="2.0.1"/>
  </xsd:complexType>
</xsd:element>
```


Tutti i tipi, semplici e complessi, sono invece definiti nel file `metatype.xsd`, connesso a `metadigit.xsd` grazie a un'inclusione.

```
<xsd:include schemaLocation="./metatype.xsd"/>
```

Sempre all'interno del file troviamo una serie di **univocity constraint**, ossia dei vincoli di univocità posti in luoghi chiave dello schema e che verranno illustrati più avanti.

Il file comprende inoltre una documentazione dettagliata che testimonia le innovazioni introdotte nelle varie versioni del file

3.1.2. I tipi MAG: il file `metatype.xsd`, `audio.xsd` e `video.xsd`

I tipi MAG, semplici o complessi, sono tutti definiti nei file ancillari:

- **`metatype.xsd`** : definisce i tipi di uso generalizzato e relativi alle sezioni IMG, DOC, OCR e DIS.
- **`audio.xsd`** : definisce i tipi usati dai file audio documentati nella sezione AUDIO (p.97)
- **`video.xsd`** : definisce i tipi usati dai file video documentati nella sezione VIDEO (p.115).

Il file `metatype.xsd`, inoltre, importa i namespace usati nello schema: Dublin Core (`dc:`) e NISO (`niso:`); `xlink` (`xlink:`), invece, viene abilitato, ma non importato.

3.1.3. `xlink` e il tipo complesso `link`

Lo schema MAG importa e utilizza il namespace `xlink`, uno schema del W3C che definisce una serie di attributi utili per localizzare una risorsa nella rete. Tale namespace è importato all'interno dello schema (nel file `metatype`) grazie a un tipo complesso:

```
<xsd:complexType name="link">
  <xsd:attributeGroup ref="xlink:simpleLink"/>
  <xsd:attribute name="Location">
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:enumeration value="URN"/>
        <xsd:enumeration value="URL"/>
        <xsd:enumeration value="URI"/>
        <xsd:enumeration value="PURL"/>
        <xsd:enumeration value="HANDLE"/>
        <xsd:enumeration value="DOI"/>
        <xsd:enumeration value="OTHER"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:attribute>
</xsd:complexType>
```

Gli attributi appartenenti a `xlink` (tutti opzionali) sono:

- **`href`** : di tipo `anyURI`, localizza una risorsa remota

- **type** : definisce il tipo di link operato. Ha il valore fisso di `simple`
- **role** : di tipo `anyURI`, descrive il ruolo che la risorsa remota ha all'interno del link
- **arcrole** : di tipo `anyURI`, descrive il ruolo che la risorsa remota ha all'interno dell'arco (locale-remoto) del link.
- **title** : di tipo `xsd:string`, descrive il titolo o il significato della risorsa cui si fa riferimento
- **show** : descrive il modo di mostrare la risorsa. I valori possibili sono:
 - **new** : la risorsa si apre in una nuova finestra
 - **replace** : la risorsa si apre al posto della finestra attuale
 - **embed** : la risorsa si apre all'interno della finestra attuale
 - **other** : altro
 - **none** : la risorsa non viene mostrata
- **actuate** : dà informazioni relativamente a quando mostrare la risorsa. I valori possibili sono:
 - **onLoad** : la risorsa viene aperta all'apertura della pagina
 - **onRequest** : la risorsa viene aperta su richiesta dell'utente
 - **other** : altro
 - **none** : la risorsa non viene aperta

Il tipo complesso `link` contiene anche la definizione di un ulteriore attributo `Location` che specifica il tipo di link definito da `xlink:href`. I valori possibili sono:

- **URN**: *Uniform Resource Name*;
- **URL**: *Uniform Resource Locator*;
- **URI**: *Uniform Resource Identifier* (sui primi tre tipi di identificatori di risorsa, si veda la panoramica informativa del W3C <http://www.w3.org/Addressing/>)
- **PURL**: Persistent URL, sviluppato dalla OCLC, Online Computer Library Center (si veda il sito PURL <http://purl.oclc.org/>).
- **HANDLE**: tipologia di riferimenti definiti secondo il sistema **Handle** della CNRI, Corporation for National Research Initiatives (si veda il sito dell'Handle System <http://www.handle.net/>).
- **DOI**: Digital Object Identifier (si veda il sito della Doi Foundation <http://www.doi.org/>).
- **OTHER**: altro.

3.2. Struttura dei documenti MAG

3.2.1. L'elemento root

L'elemento root dello schema MAG è `<metadigit>` (o `<mag:metadigit>` a seconda che si decida o meno di usare elementi qualificati). Nello start tag debbono essere richiamati tutti gli attributi necessari ad abilitare i numerosi namespace usati dallo schema MAG, come nell'esempio:

```
<mag:metadigit xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:niso="http://www.niso.org/pdfs/DataDict.pdf"
  xmlns:xlink="http://www.w3.org/TR/xlink"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:mag="http://www.iccu.sbn.it/metaAG1.pdf"
  xsi:schemaLocation="http://www.iccu.sbn.it/metaAG1.pdf metadigit.xsd"
  version="2.0.1">

  <!-- qui il documento -->

</mag:metadigit>
```

Per l'elemento è definito un unico attributo opzionale:

- **version** : indica la versione dello schema MAG. Il valore di default è 2.0.1

Si noti che l'esempio presenta un documento MAG qualificato, che normalmente si usa solo per la distribuzione del documento all'esterno dell'istituzione; in ogni caso lo schema distribuito prevede che i documenti presenti elementi qualificati e attributi non qualificati (file metadigit.xsd):

```
<xsd:schema xmlns="http://www.iccu.sbn.it/metaAG1.pdf"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://www.iccu.sbn.it/metaAG1.pdf"
  elementFormDefault="qualified" attributeFormDefault="unqualified">

  <!-- qui lo schema -->

</xsd:schema>
```

L'elemento può contenere i seguenti elementi corrispondenti ciascuno ad altrettante sezioni:

- **<gen>** : contiene informazioni generali sul progetto e sul tipo di digitalizzazione; l'elemento è obbligatorio e non ripetibile
- **<bib>** : contiene metadati descrittivi dell'oggetto analogico o, nel caso dei documenti *born digital*, relativamente al documento stesso; l'elemento è obbligatorio e non ripetibile
- **<stru>** : contiene metadati strutturali dell'oggetto digitale; l'elemento è opzionale e ripetibile
- **** : contiene metadati specifici per le immagini fisse; l'elemento è opzionale e ripetibile
- **<audio>** : contiene metadati specifici per file audio; l'elemento è opzionale e ripetibile
- **<video>** : contiene metadati specifici per file video; l'elemento è opzionale e ripetibile
- **<ocr>** : contiene metadati specifici relativi al riconoscimento ottico del testo; l'elemento

- è opzionale e ripetibile
- **<doc>** : metadati specifici per oggetti digitali in formato testo che possono essere **derivati** o **born digital**; l'elemento è opzionale e ripetibile
 - **<dis>** : contiene metadati specifici per la distribuzione di oggetti digitali; l'elemento è opzionale e non ripetibile.

3.2.2. Sezioni obbligatorie

Un documento MAG per essere valido deve necessariamente contenere le sezioni marcate dagli elementi `<gen>` e `<bib>`. Queste sezioni infatti contengono fondamentali informazioni circa l'istituzione che opera la digitalizzazione, il progetto di digitalizzazione, lo stato dell'oggetto digitale e il codice identificativo dell'oggetto stesso.

Volendo semplificare al massimo, a rigore un documento MAG potrebbe assumere la seguente forma:

```
<metadigit xmlns:dc="http://purl.org/dc/elements/1.1/"
          xmlns:niso="http://www.niso.org/pdfs/DataDict.pdf"
          xmlns:xlink="http://www.w3.org/TR/xlink"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xmlns:mag="http://www.iccu.sbn.it/metaAG1.pdf"
          xsi:schemaLocation="http://www.iccu.sbn.it/metaAG1.pdf metadigit.xsd">
  <gen>
    <stprog>www.mioprogetto.it</stprog>
    <agency>Qui l'istituzione promotrice della digitalizzazione</agency>
    <access_rights>Diritti di accesso all'oggetto analogico </access_rights>
      <!-- valori possibili: 0 = uso interno; 1 = uso pubblico -->
    <completeness>Completezza della digitalizzazione dell'oggetto analogico</completeness>
      <!-- valori possibili: 0 = completa; 1 = incompleta -->
  </gen>
  <bib level="a"> <!-- valori di level: a=analitico; c=raccolta; m=monografia:
    s=pubblicazione in serie -->
    <dc:identifier>codice identificativo dell'oggetto</dc:identifier>
  </bib>
</metadigit>
```

È evidente che un file cosiffatto ha poco senso dal momento che non contiene che pochissime informazioni relativamente all'oggetto digitale in questione (di cui, però, non si sa nemmeno la natura), ma questo dimostra l'estrema flessibilità dello schema, pensato per adattarsi a molteplici esigenze e livelli di accuratezza.

È infatti necessario distinguere ciò che è obbligatorio da un punto di vista tecnico (ciò che rende valido un file XML) e quello che è necessario affinché la registrazione dei metadati possa assolvere ai compiti per i quali è stata realizzata. Per esempio, in un processo di digitalizzazione un'informazione fondamentale dovrebbe essere per lo meno l'URI dell'oggetto digitale (dove è conservato) e se si tratta di un'immagine, di un video o di un documento di testo, anche se questo non è strettamente obbligatorio.

3.2.3. Sezioni opzionali

Le sezioni marcate dagli elementi <stru>, , <audio>, <video>, <ocr>, <doc> e <dis> sono opzionali nel senso visto prima: a seconda del tipo di oggetto digitale da descrivere e delle esigenze del singolo progetto possono essere più o meno necessarie ed essenziali.

4. Sezione GEN

L'elemento <gen> è il primo figlio dell'elemento root <metadigit> ed è obbligatorio. Esso contiene una serie di elementi figli che contengono informazioni relative all'istituzione responsabile del progetto di digitalizzazione, al progetto stesso, alla completezza o integrità del file, all'accessibilità dell'oggetto (o gli oggetti) descritto nella sezione BIB.

L'elemento, inoltre, può contenere informazioni tecniche condivise da più oggetti descritti dal documento MAG. L'elemento non è ripetibile.

Per l'elemento sono definiti due attributi opzionali:

- **creation** : la data di creazione della sezione
- **last_update** : la data dell'ultimo aggiornamento della sezione

Entrambi gli attributi sono di tipo `xsd:dateTime`, vale a dire che la data deve essere necessariamente espressa nel seguente formato: Anno (4 cifre) - Mese (2 cifre) - Giorno (2 cifre) T Ora (2 cifre) : Minuti (2 cifre) : Secondi (2 cifre) 0, più formalmente, `aaaa-mm-ggThh:mm:ss`; per esempio:

```
<mag:gen creation="2005-08-04T13:00:00" last_update="2005-08-04T13:00:00">
```

L'elemento <gen> può contenere:

- **<stprog>** : indicazione del progetto di digitalizzazione; l'elemento è obbligatorio e non ripetibile
- **<collection>** : riferimento alla collezione di cui la risorsa digitale farà parte; l'elemento è opzionale e non ripetibile
- **<agency>** : agenzia responsabile del processo di digitalizzazione; l'elemento è obbligatorio e non ripetibile
- **<access_rights>** : condizioni di accesso all'oggetto descritto nella sezione BIB; l'elemento è obbligatorio e non ripetibile
- **<completeness>** : completezza della digitalizzazione; l'elemento è obbligatorio e non ripetibile
- **<img_group>** : caratteristiche comuni a gruppi omogenei di immagini; l'elemento è opzionale e ripetibile
- **<audio_group>** : caratteristiche comuni a gruppi omogenei di file audio; l'elemento è opzionale e ripetibile
- **<video_group>** : caratteristiche comuni a gruppi omogenei di file video; l'elemento è opzionale e ripetibile.

4.1. L'istituzione e il progetto

Il primo gruppo di elementi che sono contenuti dentro <gen>, in gran parte obbligatori, servono per dare informazioni circa l'istituzione che promuove il progetto di digitalizzazione e il progetto stesso. Tali elementi sono:

- **<stprog>** : contiene la URI dove è possibile trovare la documentazione relativa la progetto di digitalizzazione. Tipicamente si tratta della pagina web in cui sono specificate le scelte relative alla digitalizzazione del progetto; in alternativa si suggerisce di puntare alla home page dell'istituzione responsabile del progetto. Il suo contenuto è `xsd:anyURI`. L'elemento è obbligatorio, non ripetibile e non sono definiti attributi.
- **<collection>** : contiene la URI (tipicamente l'indirizzo di una pagina web) di un documento in cui viene specificata la collezione cui fa parte la risorsa o le risorse digitalizzate. Il suo contenuto è `xsd:anyURI`. L'elemento è opzionale, non ripetibile e non sono definiti attributi.
- **<agency>** : contiene il nome dell'istituzione responsabile del progetto di digitalizzazione. Il suo contenuto è `xsd:string`, ma si raccomanda di usare la sintassi UNIMARC definita per il campo 801, cioè `cod. paese (due caratteri):codice Agenzia per intero`, per esempio: `IT:BNCF`. In alternativa è possibile usare una sigla riconosciuta, per esempio dall'Anagrafe biblioteche italiane: <http://anagrafe.iccu.sbn.it/>, per esempio: `IT:VE0049` o `IT:RM1316`. L'elemento è obbligatorio, non ripetibile e non sono definiti attributi.

Per esempio:

```
<gen creation="2005-08-04T13:00:00" last_update="2005-08-04T13:00:00">
  <stprog>http://marciana.venezia.sbn.it/admv.htm</stprog>
  <agency>IT:VE0049</agency>
  <!-- omissis -->
</gen>
```

Gli elementi sono così formalmente definiti (file `metatype.xsd`):

```
<xsd:complexType name="gen">
  <xsd:sequence>
    <xsd:element name="stprog" type="xsd:anyURI"/>
    <xsd:element name="collection" type="xsd:anyURI" minOccurs="0"/>
    <xsd:element name="agency" type="xsd:string"/>
    <!-- omissis -->
  </xsd:sequence>
  <xsd:attribute name="creation" type="xsd:dateTime" use="optional"/>
  <xsd:attribute name="last_update" type="xsd:dateTime" use="optional"/>
</xsd:complexType>
```

4.2. Accessibilità e status dell'oggetto digitale

Un secondo gruppo di elementi definisce lo status dell'oggetto e ne dichiara il livello di accessibilità.

- **<access_rights>** : dichiara le condizioni di accessibilità dell'oggetto descritto nella sezione BIB. Il suo contenuto deve assumere uno dei seguenti valori:

- 0 : uso riservato all'interno dell'istituzione
- 1 : uso pubblico

L'elemento è obbligatorio, non ripetibile e non sono definiti attributi.

- **<completeness>** : dichiara la completezza della digitalizzazione. Il suo contenuto deve assumere uno dei seguenti valori:

- 0 : digitalizzazione completa
- 1 : digitalizzazione incompleta

L'elemento è obbligatorio, non ripetibile e non sono definiti attributi.

L'esempio che segue riguarda un oggetto digitale completamente digitalizzato il cui accesso è libero

```
<gen>
<!-- omissis -->
    <access_rights>1</access_rights>
    <completeness>0</completeness>
</gen>
```

Gli elementi sono così formalmente definiti (file `metatype.xsd`):

```
<xsd:complexType name="gen">
  <xsd:sequence>
    <!-- omissis -->
    <xsd:element name="access_rights" type="access_rights"/>
    <xsd:element name="completeness" type="completeness"/>
    <!-- omissis -->
  </xsd:sequence>
  <!-- omissis -->
</xsd:complexType>

<xsd:simpleType name="access_rights">
  <xsd:restriction base="xsd:integer">
    <xsd:enumeration value="0">
      <xsd:annotation>
        <xsd:documentation>uso riservato all'interno
        dell'istituzione</xsd:documentation>
      </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="1">
      <xsd:annotation>
        <xsd:documentation>uso pubblico</xsd:documentation>
      </xsd:annotation>
    </xsd:enumeration>
  </xsd:restriction>
```



```

</xsd:simpleType>

<xsd:simpleType name="completeness">
  <xsd:restriction base="xsd:integer">
    <xsd:enumeration value="0">
      <xsd:annotation>
        <xsd:documentation>digitalizzazione
completa</xsd:documentation>
      </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="1">
      <xsd:annotation>
        <xsd:documentation>digitalizzazione
parziale</xsd:documentation>
      </xsd:annotation>
    </xsd:enumeration>
  </xsd:restriction>
</xsd:simpleType>

```

4.3. Caratteristiche tecniche di gruppi omogenei di oggetti

La digitalizzazione di un oggetto analogico (per esempio un volume) può dar luogo a molti oggetti digitali (per esempio le immagini che riproducono ogni pagina del volume). Normalmente tali oggetti condividono un certo numero di caratteristiche, specie se la digitalizzazione è stata compiuta nello stesso momento. In questi casi è normale descrivere tutte le immagini ottenute dalla digitalizzazione di un singolo oggetto analogico all'interno di un unico documento MAG.

Per esempio se digitalizziamo le pagine di un volume usando il medesimo scanner e, una volta settata la risoluzione, il formato di output e una serie di altri parametri, tutte le immagini ottenute dalla scansione delle pagine avranno caratteristiche tecniche comuni. Lo stesso discorso vale per tracce audio e per stream video.

Al fine di non dover ripetere all'interno dello stesso documento MAG le medesime informazioni, è possibile inserirle una volta sola all'interno dell'elemento <gen> e successivamente fare richiamare tali definizioni ogni volta che si descrive un nuovo oggetto digitale. Gli elementi definiti a tale scopo sono tre: <img_group>, <audio_group> e <video_group>. Tali elementi sono così formalmente definiti (file metatype.xsd):

```

<xsd:complexType name="gen">
  <xsd:sequence>
    <!-- omissis -->
    <xsd:element name="img_group" type="img_group" minOccurs="0"
      maxOccurs="unbounded"/>
    <xsd:element name="audio_group" type="audio_group" minOccurs="0"
      maxOccurs="unbounded"/>
    <xsd:element name="video_group" type="video_group" minOccurs="0"

```

```
maxOccurs="unbounded" />  
</xsd:sequence>  
<!-- omissis -->  
</xsd:complexType>
```

4.3.1. Gruppi di immagini

Le caratteristiche comuni a un gruppo omogeneo immagini possono essere definite all'interno dell'elemento `<img_group>`. L'elemento è opzionale, ripetibile e ha un attributo obbligatorio:

- **ID** : l'attributo è di tipo `xsd:ID`, vale a dire che contiene un identificatore univoco che consentirà poi di richiamare l'intera sezione. Tale identificatore dovrà necessariamente cominciare con una lettera (non un numero), non dovrà contenere spazi al suo interno né segni di punteggiatura diversi dal punto, il trattino e il trattino basso.

L'elemento `<img_group>` può contenere i seguenti elementi:

- **<image_metrics>** : contiene le caratteristiche tecniche principali dell'immagine, definite secondo lo standard NISO. L'elemento è obbligatorio e non ripetibile. Per maggiori dettagli sull'elemento si veda la trattazione contenuta nella sezione IMG (p.82).
- **<ppi>** : indica la risoluzione dell'immagine espressa in **ppi, pixel per inch**, risoluzione spaziale in entrambe le direzioni orizzontale e verticale (pixel quadrato). L'elemento è opzionale e non ripetibile. Per maggiori dettagli sull'elemento si veda la trattazione contenuta nella sezione IMG (p.84).
- **<dpi>** : indica la risoluzione dell'immagine espressa in **dpi, dot per inch**, risoluzione spaziale in entrambe le direzioni orizzontale e verticale (pixel quadrato). L'elemento è opzionale e non ripetibile. Per maggiori dettagli sull'elemento si veda la trattazione contenuta nella sezione IMG (p.84).
- **<format>** : dichiara la tipologia dell'immagine e la modalità di compressione. L'elemento è obbligatorio e non ripetibile. Per maggiori dettagli sull'elemento si veda la trattazione contenuta nella sezione IMG (p.84).
- **<scanning>** : dichiara le modalità di svolgimento della scansione. L'elemento è opzionale e non ripetibile. Per maggiori dettagli sull'elemento si veda la trattazione contenuta nella sezione IMG (p.85).

Per esempio:

```
<img_group ID="BNM3-000017">  
  <image_metrics>  
    <niso:samplingfrequencyunit>2</niso:samplingfrequencyunit>  
    <niso:samplingfrequencyplane>1</niso:samplingfrequencyplane>  
    <niso:photometricinterpretation>RGB</niso:photometricinterpretation>  
    <niso:bitpersample>8,8,8</niso:bitpersample>  
  </image_metrics>  
  <format>
```

```
<niso:name>TIF</niso:name>
<niso:mime>image/tiff</niso:mime>
<niso:compression>Uncompressed</niso:compression>
</format>
<scanning>
  <niso:scanningagency>IT:VE0049</niso:scanningagency>
  <niso:scanningsystem>
    <niso:scanner_manufacturer>Zeutschel</niso:scanner_manufacturer>
    <niso:scanner_model>OS6000</niso:scanner_model>
    <niso:capture_software>OmniScan 8.01</niso:capture_software>
  </niso:scanningsystem>
</scanning>
</img_group>
```

L'elemento è così formalmente definito (file `metatype.xsd`):

```
<xsd:complexType name="img_group">
  <xsd:sequence>
    <xsd:element name="image_metrics" type="niso:spatialmetrics"/>
    <xsd:element name="ppi" type="xsd:positiveInteger" minOccurs="0"/>
    <xsd:element name="dpi" type="xsd:positiveInteger" minOccurs="0"/>
    <xsd:element name="format" type="niso:format"/>
    <xsd:element name="scanning" type="niso:image_creation" minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="ID" type="xsd:ID"/>
</xsd:complexType>
```

4.3.2. Gruppi di tracce audio

Le caratteristiche comuni a un gruppo omogeneo tracce audio possono essere definite all'interno dell'elemento `<audio_group>`. L'elemento è opzionale, ripetibile e ha un attributo obbligatorio:

- **ID** : l'attributo è di tipo `xsd:ID`, vale a dire che contiene un identificatore univoco che consentirà poi di richiamare l'intera sezione. Tale identificatore dovrà necessariamente cominciare con una lettera (non un numero), non dovrà contenere spazi al suo interno né segni di punteggiatura diversi dal punto, il trattino e il trattino basso.

L'elemento `<audio_group>` può contenere i seguenti elementi:

- **<audio_metrics>** : definisce le caratteristiche tecniche della traccia audio digitale. L'elemento è obbligatorio e non ripetibile. Per maggiori dettagli sull'elemento si veda la trattazione contenuta nella sezione AUDIO (p.97).
- **<format>** : definisce la tipologia della traccia audio digitale e le modalità di compressione. L'elemento è obbligatorio e non ripetibile. Per maggiori dettagli sull'elemento si veda la trattazione contenuta nella sezione AUDIO (p.104).

- **<transcription>** : descrive le modalità della trascrizione digitale della traccia audio. L'elemento è opzionale e non ripetibile. Per maggiori dettagli sull'elemento si veda la trattazione contenuta nella sezione AUDIO (p.106).

Per esempio:

```
<audio_group ID="aGroupIntra">
  <audio_metrics>
    <samplingfrequency>48</samplingfrequency>
    <bitpersample>24</bitpersample>
    <bitrate>256</bitrate>
  </audio_metrics>
  <format>
    <name>mp3</name>
    <mime>audio/mp3</mime>
    <compression>MPEG-1 layer 3</compression>
    <channel_configuration>Joint stereo</channel_configuration>
  </format>
  <transcription>
    <sourcetype>disco (78 gg.)</sourcetype>
    <transcriptionagency>Discoteca di Stato - Museo dell'Audiovisivo</transcriptionagency>
    <transcriptiondate>2005-12-28T19:22:48</transcriptiondate>
    <devicesource>giradischi Technics doppio braccio, testina Stanton</devicesource>
    <transcriptionchain>
      <device_description Type="convertitore A/D 24/48" Unique_identifier="AD-16X"
        Comments="dispositivo acquistato nel 2004"/>
      <device_manufacturer>Apogee</device_manufacturer>
      <device_model Model="Rosetta" Serial_Number="AD-16X"/>
      <capture_software>Analogue Audio Ingestion</capture_software>
      <device_settings>48Khz, double arms</device_settings>
    </transcriptionchain>
  </transcription>
</audio_group>
```

L'elemento è così formalmente definito (file metatype.xsd):

```
<xsd:complexType name="audio_group">
  <xsd:sequence>
    <xsd:element name="audio_metrics" type="audio_spatialmetrics"/>
    <xsd:element name="format" type="audio_format"/>
    <xsd:element name="transcription" type="audio_creation" minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="ID" type="xsd:ID"/>
</xsd:complexType>
```

4.3.3. Gruppi di stream video

Le caratteristiche comuni a un gruppo omogeneo stream video possono essere definite all'interno dell'elemento `<video_group>`. L'elemento è opzionale, ripetibile e ha un attributo obbligatorio:

- **ID** : l'attributo è di tipo `xsd:ID`, vale a dire che contiene un identificatore univoco che consentirà poi di richiamare l'intera sezione. Tale identificatore dovrà necessariamente cominciare con una lettera (non un numero), non dovrà contenere spazi al suo interno né segni di punteggiatura diversi dal punto, il trattino e il trattino basso.

L'elemento `<video_group>` può contenere i seguenti elementi:

- **<video_metrics>** : definisce le caratteristiche tecniche dello stream video digitale. L'elemento è obbligatorio e non ripetibile. Per maggiori dettagli sull'elemento si veda la trattazione contenuta nella sezione VIDEO (p.120).
- **<format>** : definisce la tipologia dello stream video e le modalità di compressione. L'elemento è obbligatorio e non ripetibile. Per maggiori dettagli sull'elemento si veda la trattazione contenuta nella sezione VIDEO (p.123)
- **<digitisation>** : descrive il processo di trascrizione del contenuto video. L'elemento è opzionale e non ripetibile. Per maggiori dettagli sull'elemento si veda la trattazione contenuta nella sezione VIDEO (p.125)

Per esempio:

```
<video_group ID="vGroupIntra">
  <video_metrics>
    <videosize>720x576</videosize>
    <aspectratio>4:3</aspectratio>
    <framerate>25</framerate>
  </video_metrics>
  <format>
    <name>avi</name>
    <mime>video/avi</mime>
    <videofformat>PAL</videofformat>
    <encode>interlaced</encode>
    <streamtype>Uncompressed</streamtype>
    <codec>digital video</codec>
  </format>
  <digitisation>
    <sourcetype>casseta Betacam SP</sourcetype>
    <transcriptionagency>Discoteca di Stato - Museo dell'Audiovisivo</transcriptionagency>
    <devicesource>Sony Betacam SP</devicesource>
    <transcriptionchain>
      <device_description Type="convertitore video A/D" Unique_identifier="DVrex"
        Comments="dispositivo acquistato nel 2005"/>
      <device_manufacturer Manufacturer="Canopus"/>
    </transcriptionchain>
  </digitisation>
</video_group>
```

```
<device_model Model="DVrex" />
<capture_software>Video editing program </capture_software>
<device_settings>DV compression</device_settings>
</transcriptionchain>
</digitisation>
</video_group>
```

L'elemento è così formalmente definito (file metatype.xsd):

```
<xsd:complexType name="video_group">
  <xsd:sequence>
    <xsd:element name="video_metrics" type="video_spatialmetrics"/>
    <xsd:element name="format" type="video_format"/>
    <xsd:element name="digitisation" type="video_creation" minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="ID" type="xsd:ID"/>
</xsd:complexType>
```

5. Sezione BIB

L'elemento `<bib>` è il secondo figlio dell'elemento root `<metadigit>` ed è obbligatorio. Esso contiene una serie di elementi figli che raccolgono metadati descrittivi relativamente all'oggetto analogico digitalizzato o, nel caso di documenti **born digital**, relativamente al documento stesso. L'elemento non è ripetibile.

Per l'elemento è definito un attributo obbligatorio:

- **level** : indica il livello della descrizione bibliografica. Il suo valore deve essere scelto fra i seguenti:
 - **a**: spoglio
 - **m**: monografia
 - **s**: seriale
 - **c**: raccolta prodotta dall'istituzione

```
<bib level="m">
  <!-- qui metadati descrittivi relativi a una monografia -->
</bib>
```

L'elemento `<bib>` può contenere:

- **Set di elementi Dublin Core** (`<dc:*>`): raccoglie i metadati descrittivi dell'oggetto analogico alla base della digitalizzazione; tutti gli elementi sono opzionali (tranne `<dc:identifier>` che è obbligatorio) e ripetibili.
- **<holdings>** : raccoglie le informazioni relative all'Istituzione che possiede l'oggetto analogico. L'elemento è opzionale e ripetibile
- **<local_bib>** : viene inteso come contenitore per il trasporto di informazioni specialistiche raccolte durante il processo di digitalizzazione. L'elemento è opzionale e ripetibile
- **<piece>** : contiene dati relativi a un'unità fisica componente di un'unità superiore (es.: fascicolo di un seriale, parte di una unità bibliografica). L'elemento è opzionale e non ripetibile.

L'elemento `<bib>` è così formalmente dichiarato (file `metatype.xsd`):

```
<xsd:complexType name="bib">
  <xsd:sequence>
    <!-- omissis -->
  </xsd:sequence>
  <xsd:attribute name="level" type="bibliographic_level" use="required"/>
</xsd:complexType>
```

5.1. La descrizione bibliografica: il set di elementi Dublin Core

Lo schema MAG importa il set di elementi Dublin Core tramite il namespace convenzionale `dc:` e li utilizza per la descrizione dell'oggetto analogico alla base della digitalizzazione; per esempio, nel caso della digitalizzazione di un volume, gli elementi DC si occuperanno di registrare i metadati descrittivi relativi al volume cartaceo. Nel caso, invece, di un documento **born digital** – per il quale si veda la sezione DOC (p. 136) –, si occuperà di descrivere la **natura** dell'oggetto bibliografico, facendo semmai riferimento a un eventuale fonte bibliografica del documento tramite l'elemento `<dc:source>`

Il tag set Dublin Core (DC) rappresenta uno degli standard più diffusi al mondo per la codifica dei metadati descrittivi. È costituito di 15 elementi, tutti opzionali che si devono presentare nell'ordine in cui sono illustrati.

5.1.1. Il `<dc:identifier>`

Il primo degli elementi Dublin Core è il `<dc:identifier>` che contiene un identificatore univoco di un record descrittivo nell'ambito di un dato contesto. Di solito si usa un identificatore di un record bibliografico (opportunamente normalizzato) appartenente a un qualche schema di catalogazione (per es. SBN, Library of Congress).

Il `<dc:identifier>`, tuttavia, non va confuso con la segnatura dell'oggetto analogico o con la sua classificazione catalografica. Si tratta infatti di un **codice identificativo** che serve per fare riferimento in modo univoco a un dato oggetto; come tale pertanto, non dovrebbe contenere al suo interno alcuno spazio o altro carattere dotato di significato speciale. Nel caso in cui si voglia comunque usare segnature o sigle catalografiche, poiché un identificatore meccanico deve sottostare a regole particolari, è comunque necessario normalizzare tale segnatura applicando le cosiddette **URI Escaping Techniques**.

In un URI (**Uniform Resource Identifier**) alcuni caratteri hanno infatti un significato particolare e quindi per usarli al di fuori del loro significato è necessario impiegare una codifica particolare. Nel dettaglio, per forzare il sistema ad accettare un carattere dotato di un particolare significato senza tale significato è necessario introdurre un escape, vale a dire il simbolo di percento `%` seguito dalla codifica esadecimale (composta di due cifre) del carattere stesso. Per esempio lo spazio ha il significato speciale di "fine di un URI", se vogliamo invece che non venga considerato in questo modo, dovremo sostituirlo con `%20`. Un altro carattere che ha un significato particolare è il segno di slash `/` che significa "gerarchicamente inferiore" e che può essere forzato grazie alla codifica `%2F`.

I seguenti caratteri sono riservati e devono essere codificati con un escape:

/	%2F
?	%3F
#	%23
[e]	%5B e %5D
;	%3B
:	%3A

@	%40
&	%26
=	%3D
+	%2B
\$	%24
,	%2C
<	%3C
>	%3E
%	%25
"	%22
{ e }	%7B e %7D
	%7C
\	%5C
^	%5E
`	%60
(spazio)	%20

Sono invece utilizzabili i seguenti caratteri:

- numeri
- lettere maiuscole e minuscole
- segni di punteggiatura quali (separati da |) - | _ | . | ! | ~ | * | ' | (|)

```
<dc:identifier>A0023jii145</dc:identifier>
```

```
ms. 12364 -->
```

```
<dc:identifier>ms_12364</dc:identifier> oppure
```

```
<dc:identifier>ms.%2012364</dc:identifier>
```

L'elemento è ripetibile.

Nel caso di `<dc:identifier>` plurimi, nella versione 1.5 di MAG era consentito differenziarli tramite l'utilizzo dell'attributo `xsi:type`. Tale soluzione, però, poneva complessi problemi di validazione. In questa versione, nel caso si vogliano inserire più `<dc:identifier>`, si propone l'utilizzo di un identificatore standardizzato da porre nel contenuto dell'elemento, vale a dire lo schema **URI info** che serve a referenziare tramite una URI gli asset riconosciuti che, pur avendo un identificatore pubblico, non possono essere dereferenziati a partire dalla stessa URI (ad esempio, non si possono presentare nella forma `http://CFI0342793`). Per poter usare tale sistema, è necessario registrare preventivamente un **namespace** al sito <http://info-uri.info/>. Ulteriori informazioni sullo schema **URI info** possono essere lette al sito <http://info-uri.info/registry/docs/misc/faq.html> oppure al sito <http://www.loc.gov/standards/uri/info.html#openurl>

<dc:identifier>info:sbn/CFI0342793</dc:identifier>

<dc:identifier>info:bni/2004-778</dc:identifier>

5.1.2. Gli altri elementi Dublin Core

- <dc:title> : il titolo della risorsa, o, più formalmente, il nome attraverso il quale la risorsa è conosciuta.
Per **risorsa**, qui e altrove, si intende l'oggetto analogico o, nel caso dei documenti *born digital*, l'oggetto per il quale si creano metadati di tipo MAG.
- <dc:creator> : autore della risorsa, o più formalmente il nome dell'entità che ha la responsabilità principale della produzione del contenuto della risorsa
- <dc:publisher> : entità responsabile della produzione della risorsa (per esempio la casa editrice presso la quale è stato pubblicato l'oggetto analogico).
- <dc:subject> : argomento della risorsa
- <dc:description> : una spiegazione del contenuto della risorsa. Una descrizione può includere (ma non è limitata solo a): un riassunto analitico, un indice, un riferimento al contenuto di una rappresentazione grafica o un testo libero del contenuto
- <dc:contributor> : un'entità responsabile della produzione di un contributo al contenuto della risorsa (per esempio il nome del curatore, del traduttore, dell'illustratore).
- <dc:date> : una data associata a un evento del ciclo di vita della risorsa. Normalmente <dc:date> è associata alla creazione o alla disponibilità della risorsa. Si consiglia di usare un numero di 8 caratteri nella forma YYYY-MM-DD come definita in <http://www.w3.org/TR/NOTE-datetime> , un profilo dello standard ISO 8601. Secondo tale schema la data 1994-11-05 corrisponde al 5 novembre 1994. Sono possibili molti altri schemi, ma se vengono utilizzati, devono poter essere identificati in modo univoco
- <dc:type> : contiene la natura o il genere del contenuto della risorsa. Si consiglia di utilizzare la codifica UNIMARC Guida Pos. 6 nella traduzione italiana. In particolare si suggerisce di usare i seguenti valori:
 - Testo a stampa
 - Manoscritto
 - Musica a stampa
 - Musica manoscritta
 - Cartografia a stampa
 - Cartografia manoscritta
 - Materiale video
 - Registrazione sonora non musicale
 - Registrazione sonora musicale
 - Materiale grafico
 - Risorsa elettronica
 - Materiale multimediale
 - Oggetto a tre dimensioni

Nel caso di documenti **born digital** si consiglia invece l'uso delle tipologie raccomandate dal Dublin Core Type Vocabulary, consultabile al sito <http://dublincore.org/documents/dcmi-type-vocabulary/> e che comprende i seguenti valori:

- Collection

- Dataset
 - Event
 - Image
 - Interactive Resource
 - Moving Image
 - Physical Object
 - Service
 - Software
 - Sound
 - Still Image
 - Text
- .
- **<dc:format>** : la manifestazione fisica della risorsa. Normalmente può includere il tipo di supporto o le dimensioni della risorsa.
 - **<dc:source>** : riferimento a una risorsa dalla quale è derivata la risorsa in oggetto. Campo di rara applicazione nei progetti di scansione digitale, mentre è applicabile nel caso di documenti **born digital**.
 - **<dc:language>** : la lingua del contenuto intellettuale della risorsa analogica. Per i valori dell'elemento Language si raccomanda di utilizzare quanto definito dal RFC 1766 (<http://www.ietf.org/rfc/rfc1766.txt>) che include un codice di lingua su due caratteri (derivato dallo standard ISO 639 consultabile al sito <http://209.41.82.30/cover/iso639a.html>), seguito opzionalmente da un codice di paese su due caratteri (derivato dallo standard ISO 3166 consultabile al sito <http://209.41.82.30/cover/country3166.html>). Ad esempio, "en" per l'inglese, "fr" per il francese, "it" per l'italiano o "en-uk" per l'inglese usato nel Regno Unito
 - **<dc:relation>** : riferimento a una risorsa correlata. Si raccomanda di far riferimento alla risorsa per mezzo di una sequenza di caratteri alfabetici o numerici in conformità ad un sistema di identificazione formalmente definito.
 - **<dc:coverage>** : estensione o scopo del contenuto della risorsa. Normalmente l'elemento include la localizzazione spaziale (il nome di un luogo o le coordinate geografiche), il periodo temporale (l'indicazione di un periodo, una data o un range di date) o una giurisdizione (ad esempio il nome di un'entità amministrativa). Si raccomanda di selezionare un valore da un vocabolario controllato (ad esempio il Thesaurus of Geographic Names [TGN] consultabile al sito http://www.getty.edu/research/conducting_research/vocabularies/tgn/) e, se possibile, di utilizzare i nomi di luogo o i periodi di tempo piuttosto che identificatori numerici come serie di coordinate o range di date.
 - **<dc:rights>** : informazione sui diritti esercitati sulla risorsa. Normalmente contiene un'indicazione di gestione dei diritti sulla risorsa, o un riferimento a un servizio che fornisce questa informazione. L'informazione sui diritti spesso comprende i diritti di proprietà intellettuale Intellectual Property Rights (IPR), Copyright e vari diritti di proprietà. Se l'elemento è assente, non si può fare alcuna ipotesi sullo stato di questi o altri diritti in riferimento alla risorsa.

Si vedano i seguenti esempi d'uso, il primo relativo a un documento digitalizzato, il secondo, invece, relativo a un documento **born digital**:

```
<dc:identifier>ARM0001415</dc:identifier>  
<dc:title>Scarlatty. / Libro 5. / Año de 1753</dc:title>  
<dc:creator>Scarlatti, Domenico</dc:creator>
```

```
<dc:date>1753</dc:date>

<dc:identifier>bibit:cibit:200310082125</dc:identifier>
<dc:title>Commedia</dc:title>
<dc:creator>Alighieri, Dante</dc:creator>
<dc:publisher>Roma : Biblioteca Italiana</dc:publisher>
<dc:subject>851.1 - POESIA ITALIANA, PRIMO PERIODO FINO AL 1375</dc:subject>
<dc:subject>Poesia</dc:subject>
<dc:subject>300</dc:subject>
<dc:date>2003</dc:date>
<dc:type>text</dc:type>
<dc:format>electronic - 711 Kb</dc:format>
<dc:source>Alighieri, Dante. Le opere / DanteAlighieri ; a cura di Societa
dantesca italiana ; a cura di Petrocchi, Giorgio - Milano ; [poi] Firenze : Mondadori ;
[poi] Le Lettere , 1994</dc:source>
<dc:source>IT\ICCU\RAV\0236189</dc:source>
<dc:language>it</dc:language>
```

Maggiori informazioni circa lo schema DC possono essere reperite direttamente al sito della Dublin Core Metadata Initiative <http://dublincore.org/index.shtml> . L'ICCU fornisce anche una traduzione italiana dello schema²².

Gli elementi del set Dublin Core sono formalmente dichiarati all'interno dello schema MAG come segue (file metaype.xsd):

```
<xsd:complexType name="bib">
  <xsd:sequence>
    <xsd:element ref="dc:identifier" maxOccurs="unbounded"/>
    <xsd:element ref="dc:title" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element ref="dc:creator" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element ref="dc:publisher" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element ref="dc:subject" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element ref="dc:description" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element ref="dc:contributor" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element ref="dc:date" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element ref="dc:type" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element ref="dc:format" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element ref="dc:source" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element ref="dc:language" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element ref="dc:relation" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element ref="dc:coverage" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element ref="dc:rights" minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
  <!-- omissis -->
</xsd:complexType>
```

22. Consultabile al sito <http://www.iccu.sbn.it/dublinco.html> .

5.2. La proprietà dell'oggetto analogico

Al gruppo degli elementi DC, seguono una serie di elementi che servono a localizzare l'oggetto analogico all'interno di una particolare istituzione. Tali elementi sono racchiusi dall'elemento `<holdings>`. L'elemento è opzionale e ripetibile per contemplare la possibilità che un oggetto analogico possa essere posseduto da più istituzioni (è il caso, per esempio, di una rivista le cui annate non sono integralmente possedute da un'unica biblioteca). Per l'elemento è definito un unico attributo opzionale:

- **ID** : di tipo `xsd:ID`, serve a definire un identificatore univoco all'interno del record MAG cui è possibile fare riferimento da altri luoghi del medesimo record. L'attributo trova la sua utilità qualora vi sia la necessità di dichiarare diversi `<holdings>`.

Il contenuto dell'elemento `<holdings>` è di tipo `xsd:sequence`, vale a dire che è costituito da una sequenza di elementi. Nello specifico, l'elemento `<holdings>` contiene i seguenti elementi, tutti opzionali:

- **<library>** : contiene il nome dell'istituzione proprietaria dell'oggetto analogico o di parte dell'oggetto analogico. Di tipo `xsd:string`, è opzionale e non ripetibile.
- **<inventory_number>** : contiene il numero di inventario attribuito all'oggetto analogico dall'istituzione che lo possiede. Di tipo `xsd:string`, è opzionale e non ripetibile.
- **<shelfmark>** : contiene la collocazione dell'oggetto digitale all'interno del catalogo dell'istituzione che lo possiede. Di tipo `xsd:string`, è opzionale e ripetibile. Per l'elemento è definito un attributo:
 - **type** : si usa per definire il tipo di collocazione nel caso di collocazioni plurime, per esempio quando si vuole registrare una collocazione antica e una moderna. L'attributo è opzionale e il suo contenuto è `xsd:string`.

Si vedano i seguenti esempi d'uso dell'elemento `<holdings>`:

```
<holdings ID="A0">
  <library>Biblioteca nazionale Marciana, Venezia, Italia</library>
  <shelfmark>It.IV,205(=9776)</shelfmark>
</holdings>
```

```
<holdings ID="PO_02">
  <library>Biblioteca nazionale Marciana, Venezia, Italia</library>
  <shelfmark>CII.4.*</shelfmark>
</holdings>
```

```
<holdings>
  <library>Biblioteca comunale Augusta, Perugia, Italia (BAP)</library>
```

```
<inventory_number>206625</inventory_number>  
<shelfmark>ms 1094</shelfmark>  
</holdings>
```

L'elemento `<holdings>` è così formalmente definito (file `metatype.xsd`):

```
<xsd:element name="holdings" minOccurs="0" maxOccurs="unbounded">  
  <xsd:complexType>  
    <xsd:sequence>  
      <xsd:element name="library" type="xsd:string" minOccurs="0"/>  
      <xsd:element name="inventory_number" type="xsd:string" minOccurs="0"/>  
      <xsd:element name="shelfmark" minOccurs="0" maxOccurs="unbounded">  
        <xsd:complexType mixed="true">  
          <xsd:attribute name="type" type="xsd:string"/>  
        </xsd:complexType>  
      </xsd:element>  
    </xsd:sequence>  
    <xsd:attribute name="ID" type="xsd:ID"/>  
  </xsd:complexType>  
</xsd:element>
```

5.3. Sistemi di catalogazioni specializzate

Alcuni progetti di digitalizzazione che hanno adottato MAG come standard per la raccolta dei metadati amministrativi e gestionali, hanno messo in evidenza la necessità di dotare lo schema di alcuni elementi per la raccolta di particolari informazioni specialistiche relativamente all'oggetto analogico raccolte durante il processo di digitalizzazione. Tali informazioni non potevano essere agevolmente codificate all'interno del set Dublin Core poiché la scelta di non avvalersi degli elementi Dublin Core qualificati rendevano difficilmente identificabili tali contenuti. È stato perciò creato l'elemento `<local_bib>` di tipo `xsd:sequence`, per il quale non sono definiti attributi. L'elemento è opzionale così pure come gli elementi ivi contenuti:

- `<geo_coord>` : di tipo `xsd:string`, contiene le coordinate geografiche relative a una carta o a una mappa. L'elemento è opzionale e ripetibile. Non sono definiti attributi.
- `<not_date>` : di tipo `xsd:string`, contiene la data di notifica relativa a un bando o a un editto. L'elemento è opzionale e ripetibile. Non sono definiti attributi.

L'elemento `<local_bib>` è così formalmente definito (file `metatype.xsd`):

```
<xsd:element name="local_bib" minOccurs="0" maxOccurs="unbounded">  
  <xsd:complexType>  
    <xsd:sequence>  
      <xsd:element name="geo_coord" minOccurs="0" maxOccurs="unbounded"/>  
      <xsd:element name="not_date" minOccurs="0" maxOccurs="unbounded"/>  
    </xsd:sequence>  
</xsd:element>
```

```
</xsd:complexType>  
</xsd:element>
```

5.4. Pubblicazioni seriali e unità componenti

Pubblicazioni seriali e unità componenti di opere più vaste possono essere minuziosamente descritte. Tali informazioni sono raccolte dall'elemento <piece>, di tipo `xsd:choice`, vale a dire che può avere due contenuti diversi a seconda che contenga dati relativi a una pubblicazione seriale (per esempio il fascicolo di una rivista) o all'unità componente di un'opera più vasta (per esempio il singolo volume di un'enciclopedia). L'elemento è opzionale e non ripetibile; non sono definiti attributi.

5.4.1. Pubblicazioni seriali

Nel caso di pubblicazioni seriali l'elemento <piece> assume il seguente contenuto:

- **<year>** : di tipo `xsd:string`, contiene l'annata di copertura editoriale di una pubblicazione seriale nella forma in cui si trova sulla pubblicazione stessa; per esempio 1913-1914 o anche 1987. L'elemento è obbligatorio qualora si scelga l'opzione periodici, anche se formalmente opzionale, e non ripetibile. Non sono definiti attributi.
- **<issue>** : di tipo `xsd:string`, contiene gli estremi identificatori di un fascicolo di una pubblicazione seriale nella forma in cui si trova sulla pubblicazione stessa; per esempio n.° 8. L'elemento è obbligatorio qualora si scelga l'opzione periodici, anche se formalmente opzionale, e non ripetibile. Non sono definiti attributi.
- **<stpiece_per>** : il campo permette di registrare in una forma normalizzata il riferimento a un fascicolo di un periodico; questo sia per poter scambiare i dati, sia per poter ordinare in modo automatico i vari record. Il campo <stpiece_per> è opzionale, non ripetibile e non è inteso a sostituire le informazioni contenute negli altri campi di <piece>. Formalmente è definito come restrizione del tipo `xsd:string`, essendo il suo contenuto regolato da una complessa espressione regolare. La sintassi utilizzata per la normalizzazione è quella dello standard SICI (ANSI/NISO Z39.56) per i segmenti **Chronology**, **Enumeration** e **Supplements and Indexes** http://www.niso.org/standards/standard_detail.cfm?std_id=530 . Sinteticamente il risultato si presenta come

```
(cronologia)livello_numerazione:livello_numerazione:livello_numerazione  
:livello_numerazione.
```

Le regole per la sua creazione sono descritte ai punti 6.3.2, 6.3.3 e 6.3.4 del SICI e vengono qui di seguito richiamate.

- **1. Cronologia:**

Tutte le date sono registrate numericamente fra parentesi tonde secondo il formato YYYYMMDD (YYYY = anno, MM = mese, DD = giorno). Si usano solo i livelli applicabili. Per esempio, se la data da registrare non ha mese e giorno o stagione, si registra solo l'anno. I codici da adottare sono:

Codici di cronologia

- 01 = gennaio
- 02 = febbraio
- 03 = marzo
- 04 = aprile
- 05 = maggio
- 06 = giugno
- 07 = luglio
- 08 = agosto
- 09 = settembre
- 10 = ottobre
- 11 = novembre
- 12 = dicembre
- 21 = Primavera
- 22 = Estate
- 23 = Autunno
- 24 = Inverno
- 31 = primo quarto
- 32 = secondo quarto
- 33 = terzo quarto
- 34 = quarto quarto

Es: La Repubblica 23 gennaio 2005 -> (20050123)

Airone febbraio 2003 -> (200302)

Renaissance Quarterly, 2° quarto 2004 -> (200432)

- **2. Cronologia combinata:**

Si usa la barra “/” nella cronologia per una combinazione di date o di parti di data
Esempi:

(199312/199401)20:2 per Dicembre 93 / Gennaio 94 volume 20, n. 2

(119021/22)17:3/4 per Primavera Estate 1990 volume 17 numero 3/4

Se manca la cronologia la punteggiatura “()” è comunque obbligatoria.

- **3. Numerazione:**

La numerazione identifica il fascicolo. Vi possono essere più livelli (con un massimo di 4) separati da due punti “:” Può essere omessa - senza segnalare l’omissione - se il fascicolo non presenta questa tipologia di informazioni. Per esempio:

La Repubblica 23 gennaio 2005, anno 24, n. 23 --> (20050123)24:23

Airone febbraio 2003, anno 18 n. 2 --> (200302)18:2

Renaissance Quarterly, 2° quarto 2004, anno 36 parte 2 --> (200432)36:2.

- **4. Numerazione combinata e numerazione continua:**

Nella numerazione si può usare la barra “/” per una numerazione combinata.

(119021/22)17:3/4 per Primavera Estate 1990 volume 17 numero 3/4.

Se vi sono due tipologie di numerazione si preferisce quella regolare volume:numero e in questo caso non si tiene conto della numerazione continua dei

fascicoli.

Vol 21, n. 13 (fasc 389) 23 giugno 1995 viene codificato in (11950623)21:13
e non si tiene conto di "fasc 389"

Se un periodico presenta solo una numerazione progressiva dei fascicoli senza alcuna indicazione cronologica e senza alcuna indicazione di fascicolo il risultato sarà, ad esempio, () 454 per indicare il fascicolo numero 454.

- **5. Supplementi e indici:**

Un supplemento al fascicolo si indica con il "+"

(19950910)+ per Supplemento del 10 settembre 1995 non riferito a un particolare numero

(198408)21:8+ per Supplemento al Volume 21 numero 8, Agosto 1984

Gli indici si indicano con "*"

Indice dell'annata 1990 --> (1990)*

Tuttavia se l'indice esce come fascicolo numerato entro la numerazione regolare del periodico, l'indice è trattato come un regolare fascicolo.

5.4.2. Unità componenti

Nel caso di pubblicazioni seriali l'elemento <piece> assume il seguente contenuto:

- <part_number> : numero di unità componente. Per esempio: 2, IV, 4.5. L'elemento è obbligatorio nel caso di unità componenti, anche se formalmente opzionale, e non ripetibile.
- <part_name> : nome/titolo di una unità componente. Per esempio: Volume II; Parte III, Tomo 2. L'elemento è obbligatorio nel caso di unità componenti, anche se formalmente opzionale, e non ripetibile.
- <stpiece_vol> : forma normalizzata del riferimento a una parte di una unità componente. Formalmente è definito come restrizione del tipo `xsd:string`, essendo il suo contenuto regolato da un'espressione regolare. La sintassi da adottare è la seguente: `volume:parte:parte`, volume può avere fino a 3 cifre, parte fino a quattro; entrambe le sezioni parte sono opzionali.

Volume 3, parte 2, tomo 1 --> 3:2:1

Volume 47 --> 47

Volume 2, parte 1 --> 2:1

L'elemento è opzionale e non ripetibile.

L'elemento <piece> è così formalmente definito (file `metatype.xsd`):

```
<xsd:element name="piece" minOccurs="0">  
  <xsd:complexType>
```

```
<xsd:choice minOccurs="0">
  <xsd:sequence>
    <xsd:element name="year" type="xsd:string"/>
    <xsd:element name="issue" type="xsd:string"/>
    <xsd:element name="stpiece_per" type="SICI" minOccurs="0"/>
  </xsd:sequence>
  <xsd:sequence>
    <xsd:element name="part_number" type="xsd:positiveInteger"/>
    <xsd:element name="part_name" type="xsd:string"/>
    <xsd:element name="stpiece_vol" type="BICI" minOccurs="0"/>
  </xsd:sequence>
</xsd:choice>
</xsd:complexType>
</xsd:element>
```

I tipi SICI e BICI che regolano i contenuti rispettivamente di <stpiece_per> e <stpiece_vol> sono così definiti:

```
<xsd:simpleType name="SICI">
  <xsd:restriction base="xsd:string">
    <xsd:pattern
value="\((\d{4}/\d{4})?((\d{2})/(\d{2}|\d{6}))?((\d{2})/\d{2})?
    )???\)((\+|\*)?|(\d{1,4}:(\d{1,4})/\d{1,4})?(\+|\*)?)"
    </xsd:restriction>
  </xsd:simpleType>

<xsd:simpleType name="BICI">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="\d{1,3}(:\d{1,4}(:\d{1,4})?)?"/>
  </xsd:restriction>
</xsd:simpleType>
```

6. Sezione STRU

L'elemento `<stru>` è il terzo figlio dell'elemento root `<metadigit>`; l'elemento è opzionale, ripetibile e nidificabile. Esso contiene informazioni circa la struttura logica del documento digitalizzato. Tramite le informazioni codificate in questa sezione del record MAG è infatti possibile documentare la suddivisione interna di un documento (capitoli di un libro, articoli di una rivista), realizzare record MAG radice e record MAG di spoglio oppure collegare diversi MAG fra loro.

La sezione STRU trova la sua tipica utilizzazione nei seguenti tre casi:

- quando si ritenga opportuno mettere in evidenza le eventuali partizioni interne di un oggetto digitale (es. capitoli di un libro);
- nel caso di spogli, laddove le partizioni spogliate (es. articoli di una rivista) siano discrete autonomamente e come tali possano avere una propria sezione BIB;
- per far riferimento ad altri oggetti MAG correlati non appartenenti alla stessa tipologia, per esempio nel caso di un cofanetto contenente CD musicali e un fascicolo di testo.

Non ha invece senso nel caso di un oggetto unitario, privo di strutture interne.

L'elemento `<stru>` è opzionale, ripetibile e ricorsivo, nel senso che è possibile innestare un numero indeterminato di elementi `<stru>` gli uni dentro agli altri, al fine di documentare l'eventuale articolazione interna di un documento. Il suo contenuto è di tipo `xsd:sequence` e può contenere i seguenti elementi:

- `<sequence_number>` : contiene un numero progressivo che identifica un particolare `<stru>`. L'elemento è opzionale e non ripetibile
- `<nomenclature>` : contiene il nome o la descrizione di una particolare struttura. L'elemento è opzionale e non ripetibile
- `<element>` : opzionale e ripetibile, contiene:
 - un collegamento fra una particolare struttura e un contenuto precisato nelle sezioni IMG, OCR, DOC, AUDIO o VIDEO descritto all'interno del medesimo record MAG
 - un collegamento fra una particolare struttura e un contenuto precisato nelle sezioni IMG, OCR, DOC, AUDIO o VIDEO descritto all'interno di altri record MAG.
- `<stru>` : contiene un ulteriore livello strutturale gerarchicamente subordinato rispetto all'elemento genitore. L'elemento è opzionale e ripetibile.

Per l'elemento `<stru>` sono definiti tre attributi, tutti opzionali. Il loro uso è deprecato in favore dell'uso degli elementi contenuti nella sezione. Tali attributi sono stati mantenuti per garantire la compatibilità della presente versione rispetto alle versioni precedenti.

- `descr` : di tipo `xsd:string`, contiene il nome o la descrizione di una particolare struttura. Al posto di tale attributo è preferibile usare l'elemento `<nomenclature>`
- `start` : di tipo `xsd:string`, contiene il numero di sequenza che segna l'inizio del range di contenuti (precisati nelle sezioni IMG, OCR, DOC, AUDIO o VIDEO) da collegare. Al posto di tale attributo è preferibile usare l'elemento `<start>` contenuto all'interno di `<element>`.

- **stop** : di tipo `xsd:string`, contiene il numero di sequenza che segna la fine del range di contenuti (precisati nelle sezioni IMG, OCR, DOC, AUDIO o VIDEO) da collegare. Al posto di tale attributo è preferibile usare l'elemento `<stop>` contenuto all'interno di `<element>`.

L'elemento `<stru>` è così formalmente definito (file `metatype.xsd`).

```
<xsd:complexType name="stru">
  <xsd:annotation>
    <xsd:documentation>il generico elemento STRU, ripetibile e nidificabile, è
reso univoco dall'elemento dc:identifier contenuto in BIB e dalla concatenazione dei
sequence_number dei diversi livelli di nidificazione degli STRU. Il "livello" è fornito
implicitamente dalla nidificazione degli STRU.</xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="sequence_number" type="xsd:positiveInteger" minOccurs="0"/>
    <xsd:element name="nomenclature" type="xsd:string" minOccurs="0"/>
    <xsd:element name="element" type="stru_element" minOccurs="0"
      maxOccurs="unbounded"/>
    <xsd:element name="stru" type="stru" minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
  <xsd:attribute name="descr" type="xsd:string" use="optional"/>
  <xsd:attribute name="start" type="xsd:positiveInteger" use="optional"/>
  <xsd:attribute name="stop" type="xsd:positiveInteger" use="optional"/>
</xsd:complexType>
```

6.1. Il numero di sequenza e la sua denominazione

Ogni livello strutturale può essere identificato da un numero di sequenza e può assumere una particolare denominazione, in accordo con gli standard di progetto.

Il primo componente di un elemento `<stru>` è il suo numero di sequenza (`<sequence_number>`) che serve a rendere univoco l'elemento stesso e verrà utilizzato per fare riferimento a tale `<stru>`. L'elemento `<sequence_number>` è opzionale (per essere compatibile con le versioni precedenti di MAG, ma in realtà fondamentale), non ripetibile ed è definito come `xsd:positiveInteger`, vale a dire che deve essere costituito da un numero positivo.

Proprio per la sua natura di identificativo di un particolare livello strutturale, sul contenuto dell'elemento è posto un vincolo di univocità contenuto nel file `metadigit.xsd`:

```
<xsd:unique name="unigstru">
  <xsd:selector xpath="stru"/>
  <xsd:field xpath="sequence_number"/>
</xsd:unique>
```

La denominazione del livello strutturale è contenuta dall'elemento `<nomenclature>`. L'elemento, opzionale e non ripetibile, ha un contenuto di tipo `xsd:string`, vale a dire che può contenere una qualsiasi sequenza di caratteri. Il formato che tale denominazione può

assumere dipende dagli standard di progetto che possono prevedere di riproporre i titoli dei vari livelli strutturali del documento analogico, oppure di adottare denominazioni standardizzate.

Si veda il seguente esempio relativo a un volume di sonate di Domenico Scarlatti

```
<stru>
  <sequence_number>001</sequence_number>
  <nomenclature>Sonata 1</nomenclature>
  <!-- omissis -->
</stru>
<stru>
  <sequence_number>002</sequence_number>
  <nomenclature>Sonata 2</nomenclature>
  <!-- omissis -->
</stru>
```

6.2. Il contenuto di un livello strutturale

Il contenuto di un livello strutturale viene inserito all'interno di `<element>`. L'elemento permette di individuare prima la risorsa di riferimento tramite l'elemento `<dc:identifier>` di un altro record MAG, oppure di localizzare un particolare file contenente un record MAG privo di `<dc:identifier>` tramite l'elemento `<file>`, di dichiarare la denominazione di tale contenuto tramite l'elemento `<nomenclature>`, di far riferimento a una particolare risorsa tramite `<resource>`, infine di definire il range di attribuzione tramite gli elementi `<start>` e `<stop>`.

Gli esempi che seguono vogliono illustrare alcuni usi tipici di `<element>` e di conseguenza di `<stru>` di cui `<element>` è il cuore.

6.2.1. La struttura interna di una risorsa

Come detto preliminarmente, una sezione STRU trova la sua prima occasione di utilizzo nel dare conto della partizione interna di una risorsa. Si supponga di digitalizzare *I Promessi Sposi* composto, come è noto, di un'Introduzione e di trentotto capitoli; tale struttura può essere descritta nel modo che segue:

```
<stru>
  <sequence_number>001</sequence_number>
  <nomenclature>Introduzione</nomenclature>
  <element>
    <resource>img</resource>
    <start sequence_number="001" />
    <stop sequence_number="004" />
  </element>
</stru>
<stru>
  <sequence_number>002</sequence_number>
  <nomenclature>Capitolo I</nomenclature>
```

```
<element>
<start sequence_number="005"/>
<stop sequence_number="015"/>
</element>
</stru>
<stru>
<sequence_number>003</sequence_number>
<nomenclature>Capitolo II</nomenclature>
<element>
<start sequence_number="016"/>
<stop sequence_number="024"/>
</element>
</stru>
<!-- altre stru -->
```

Come si vede, per ciascuna partizione strutturale del romanzo è stato usato un elemento `<stru>`, dotato di un `<sequence_number>` progressivo e di un elemento `<nomenclature>` che contiene il titolo della partizione considerata. All'interno di `<element>` abbiamo usato l'elemento `<resource>` per indicare che il tipo di contenuti cui si far riferimento è un'immagine statica solo nel primo caso, poiché nel caso di immagini, l'elemento `<resource>` può anche essere omesso visto che `IMG` è il valore di default. Ogni `<element>` contiene al suo interno la coppia di elementi `<start>` e `<stop>` i cui attributi `sequence_number` fanno riferimento all'elemento `<sequence_number>` della sezione `` (p. 78).

L'elemento `<stru>` può essere nidificato. Vediamo infatti l'esempio della `<stru>` del presente manuale:

```
<stru>
  <sequence_number>1</sequence_number>
  <nomenclature>Introduzione</nomenclature>
  <stru>
    <sequence_number>1</sequence_number>
    <nomenclature>Alle origini di MAG</nomenclature>
  </stru>
  <stru>
    <sequence_number>2</sequence_number>
    <nomenclature>Metadati e Oggetti digitali</nomenclature>
  </stru>
  <stru>
    <sequence_number>3</sequence_number>
    <nomenclature>Lo schema MAG</nomenclature>
  </stru>
  <stru>
    <sequence_number>4</sequence_number>
    <nomenclature>Interazione con standard internazionali</nomenclature>
  <stru>
```

```

        <sequence_number>1</sequence_number>
        <nomenclature>Dublin Core</nomenclature>
    </stru>
    <stru>
        <sequence_number>2</sequence_number>
        <nomenclature>NISO e MIX</nomenclature>
    </stru>
</stru>
</stru>

```

6.2.2. Record di spoglio. Spogli nudi e spogli vestiti.

La sezione STRU trova un suo impiego ideale nel caso della realizzazione di record di spoglio. Si supponga di aver realizzato un record MAG relativo a un volume di sonate di Domenico Scarlatti. Tale record è identificabile grazie a <dc:identifier>ARM0001415</dc:identifier>; al suo interno saranno documentate e descritte anche tutte le immagini relative alla scansione. Si supponga poi di voler realizzare un record di spoglio relativo a ciascuna sonata. Il record madre e quindi con <bib level="m"> (p. 47) si presenterà come segue:

```

<metadigit>
  <gen creation="2005-08-04T13:00:00" last_update="2005-08-04T13:00:00">
    <stprog>http://marciana.venezia.sbn.it/admv.htm</stprog>
    <agency>IT:VE0049</agency>
    <access_rights>1</access_rights>
    <completeness>0</completeness>
  </gen>
  <bib level="m">
    <dc:identifier>ARM0001415</dc:identifier>
    <dc:title>Scarlatti. / Libro 5. / Año de 1753</dc:title>
    <dc:creator>Scarlatti, Domenico</dc:creator>
    <dc:date>1753</dc:date>
    <holdings>
      <library>Biblioteca nazionale Marciana, Venezia, Italia</library>
      <shelfmark>It.IV,205(=9776)</shelfmark>
    </holdings>
  </bib>
  <img>
    <sequence_number>001</sequence_number>
    <!-- altri elementi -->
  </img>
  <img>
    <sequence_number>002</sequence_number>
    <!-- altri elementi -->
  </img>
  <img>

```

```
<sequence_number>003</sequence_number>
<!-- altri elementi -->
</img>
<!-- altre img -->
</metadigit>
```

Ciascun record di spoglio - cioè <bib level="a"> (p. 47) - si presenterà come segue²³:

```
<metadigit>
  <gen creation="2005-08-04T13:00:00" last_update="2005-08-04T13:00:00">
    <stprog>http://marciana.venezia.sbn.it/admv.htm</stprog>
    <agency>IT:VE0049</agency>
    <access_rights>1</access_rights>
    <completeness>0</completeness>
  </gen>
  <bib level="a">
    <dc:identifier>ARM0001457</dc:identifier>
    <dc:title>Sonata / 1</dc:title>
    <dc:creator>Scarlatti, Domenico</dc:creator>
    <dc:date>1753</dc:date>
    <holdings ID="A0">
      <library>Biblioteca nazionale Marciana, Venezia, Italia</library>
      <shelfmark>It.IV,205(=9776)</shelfmark>
    </holdings>
  </bib>
  <stru>
    <sequence_number>001</sequence_number>
    <element>
      <dc:identifier>ARM0001415</dc:identifier>
      <start sequence_number="001"/>
      <stop sequence_number="003"/>
    </element>
  </stru>
</metadigit>
```

Come si vede, la scheda di spoglio è dotata della propria sezione BIB e quindi anche di un proprio <dc:identifier>. Nella sezione STRU, invece, all'interno di <element> si fa riferimento alla scheda madre tramite <dc:identifier>ARM0001415</dc:identifier>; gli attributi sequence_number di <start> e <stop> fanno infatti riferimento alla sezione IMG del record madre. Si noti che l'elemento <dc:identifier> all'interno di <element> si usa esclusivamente nel caso in cui i contenuti definiti da <resource> siano descritti in un record MAG diverso; se tali contenuti sono descritti all'interno del medesimo record MAG, il <dc:identifier> si omette.

23. L'esempio riflette una distribuzione non ridondante di dati fra record madre e record di spoglio; nulla vieta, però, di duplicare le informazioni e di inserire le stru in ambedue i livelli.

L'esempio appena visto, così come si presenta, documenta uno **spoglio nudo**, vale a dire che i contenuti multimediali (le immagini, nel nostro caso) sono descritti all'interno del record madre, mentre i record di spoglio si limitano a puntare al record madre tramite il `<dc:identifier>` all'interno di `<element>`.

È tuttavia possibile anche realizzare i cosiddetti **spogli vestiti**; in questo caso la scheda madre si limiterà a dare conto della struttura interna e potrà al limite contenere la documentazione relativa alle immagini comuni all'intera monografia (copertina, pagine preliminari, indice, ecc.) mentre la descrizione delle immagini relative a ciascuna partizione strutturale sarà contenuta all'interno del record di spoglio. Si veda l'esempio di una scheda di uno spoglio vestito:

```
<metadigit>
  <gen creation="2005-08-04T13:00:00" last_update="2005-08-04T13:00:00">
    <stprog>http://marciana.venezia.sbn.it/admv.htm</stprog>
    <agency>IT:VE0049</agency>
    <access_rights>1</access_rights>
    <completeness>0</completeness>
  </gen>
  <bib level="a">
    <dc:identifier>ARM0001457</dc:identifier>
    <dc:title>Sonata / 1</dc:title>
    <dc:creator>Scarlatti, Domenico</dc:creator>
    <dc:date>1753</dc:date>
    <holdings ID="A0">
      <library>Biblioteca nazionale Marciana, Venezia, Italia</library>
      <shelfmark>It.IV,205(=9776)</shelfmark>
    </holdings>
  </bib>
  <img>
    <sequence_number>001</sequence_number>
    <!-- altri elementi -->
  </img>
  <img>
    <sequence_number>002</sequence_number>
    <!-- altri elementi -->
  </img>
  <img>
    <sequence_number>003</sequence_number>
    <!-- altri elementi -->
  </img>
  <!-- altre img -->
</metadigit>
```

In questo caso, per esplicitare comunque il legame fra la scheda madre e le schede di spoglio, all'interno di `<element>` della scheda madre si dovrà inserire il `<dc:identifier>` della scheda di spoglio visto che i contenuti (le immagini) non sono descritti al proprio interno:

```
<metadigit>
  <gen creation="2005-08-04T13:00:00" last_update="2005-08-04T13:00:00">
    <stprog>http://marciana.venezia.sbn.it/admv.htm</stprog>
    <agency>IT:VE0049</agency>
    <access_rights>1</access_rights>
    <completeness>0</completeness>
  </gen>
  <bib level="m">
    <dc:identifier>ARM0001415</dc:identifier>
    <dc:title>Scarlatty. / Libro 5. / Año de 1753</dc:title>
    <dc:creator>Scarlatti, Domenico</dc:creator>
    <dc:date>1753</dc:date>
    <holdings>
      <library>Biblioteca nazionale Marciana, Venezia, Italia</library>
      <shelfmark>It.IV,205(=9776)</shelfmark>
    </holdings>
  </bib>
  <stru>
    <sequence_number>001</sequence_number>
    <nomenclature>Sonata 1</nomenclature>
    <element>
      <dc:identifier>ARM0001457</dc:identifier>
      <start sequence_number="001"/>
      <stop sequence_number="003"/>
    </element>
  </stru>
  <!-- altre stru -->
</metadigit>
```

6.2.3. Ricongiungimento di parti separate di un oggetto unitario

Supponiamo di avere un oggetto digitale logicamente unitario, ma fisicamente diviso in diversi oggetti analogici, per esempio l'articolo di un periodico diviso in due diversi fascicoli. Tale realtà può essere descritta come segue:

```
<stru>
  <sequence_number>1</sequence_number>
  <nomenclature>Titolo Articolo</nomenclature>
  <element num="1">
    <nomenclature>Titolo Articolo, Parte 1</nomenclature>
    <dc:identifier>IT4560987</dc:identifier>
    <piece>
      <year>1999</year>
      <issue>2</issue>
    </piece>
  </element>
</stru>
```

```
<resource>img</resource>
<start sequence_number="089" />
<stop sequence_number="105" />
</element>
<element num="2">
  <nomenclature>Titolo Articolo, Parte 2</nomenclature>
  <dc:identifier>IT4560987</dc:identifier>
  <piece>
    <year>1999</year>
    <issue>3</issue>
  </piece>
  <start sequence_number="003" />
  <stop sequence_number="069" />
</element>
</stru>
```

Come si vede all'interno di <stru> sono stati inseriti due <element> che fanno riferimento ai contenuti di due diversi fascicoli, individuati dall'elemento <piece>; per dichiarare la consequenzialità degli <element> è stato usato l'attributo num. Questa soluzione presuppone che il tipo di risorsa (dichiarata da <resource>) sia il medesimo e infatti l'elemento <resource> è stato omesso nel secondo <element> (<resource> avrebbe potuto essere omesso anche nel primo caso in quanto facente riferimento alla sezione di default IMG, ma qui è stato inserito per chiarezza). Anche in questo caso il valore degli attributi sequence_number degli elementi <start> e <stop> fanno riferimento al record MAG individuato dal <dc:identifier>. Per individuare esattamente il seriale è stato usato l'elemento <piece> in modo del tutto analogo rispetto a quello visto all'interno della sezione <bib> (p. 55).

6.2.4. Collegamenti fra oggetti digitali affini

Supponiamo di digitalizzare un CD musicale che contenga anche un fascicolo cartaceo con i testi delle canzoni presenti nel CD. Supponiamo che fra gli scopi del progetto di digitalizzazione ci sia anche quello di collegare fra di loro le immagini ricavate dalla scansione del fascicolo di accompagnamento e le tracce audio ricavate dalla digitalizzazione del CD, in modo che a ogni brano corrisponda il testo della canzone.

Tale obiettivo può essere realizzato in tre diversi modi:

- un unico record MAG contenente sia l'esplicitazione della struttura che i dati dei contenuti digitali, audio e testo;
- un record MAG madre contenente i dati relativi a testo e tracce audio e una serie di record di spoglio che colleghino fra loro i dati relativi alla digitalizzazione del testo e delle tracce audio
- una record MAG madre priva di qualsiasi informazioni circa la struttura e il multimediale, una serie di spogli relativi alle tracce audio e un'altra serie di spogli relativi ai testi delle canzoni presenti nel fascicolo di accompagnamento, più un'ulteriore serie di spogli che colleghino gli spogli dell'audio con gli spogli del testo.

Vediamo ora la l'esemplificazione relativa a ciascuna di queste possibilità.

6.2.4.1. Ricongiungimento all'interno dell'unico record MAG

L'esempio è relativo a un unico record MAG che contiene al suo interno i dati relativi alla digitalizzazione del fascicolo di accompagnamento e alla digitalizzazione delle tracce audio; la sezione STRU realizza pure il ricongiungimento fra le immagini che riproducono il testo di ciascuna canzone e la relativa traccia audio:

```
<metadigit version="2.0.1">
  <gen>
    <stprog>http://biblioteca/mioprogetto.html</stprog>
    <agency>IT:AA000</agency>
    <access_rights>1</access_rights>
    <completeness>0</completeness>
  </gen>
  <bib level="m">
    <dc:identifier>BIB01</dc:identifier>
    <dc:title>Buon Compleanno Elvis</dc:title>
    <dc:creator>Ligabue</dc:creator>
    <dc:type>Registrazione sonora musicale</dc:type>
    <dc:type>Testo a stampa</dc:type>
  </bib>
  <stru>
    <sequence_number>1</sequence_number>
    <nomenclature>Certe Notti</nomenclature>
    <element>
      <nomenclature>Certe notti (testo)</nomenclature>
      <resource>img</resource>
      <start sequence_number="002" />
      <stop sequence_number="003" />
    </element>
    <element>
      <nomenclature>Certe notti (audio)</nomenclature>
      <resource>audio</resource>
      <start sequence_number="005" />
      <stop sequence_number="005" />
    </element>
  </stru>
  <img>
    <sequence_number>001</sequence_number>
    <nomenclature>Copertina</nomenclature>
  </img>
  <img>
    <sequence_number>002</sequence_number>
    <nomenclature>Pagina 2</nomenclature>
  </img>
  <img>
    <sequence_number>003</sequence_number>
    <nomenclature>Pagina 3</nomenclature>
  </img>
</metadigit>
```

```

</img>
<!-- altre img -->
<audio>
  <sequence_number>001</sequence_number>
  <nomenclature>Traccia 1</nomenclature>
</audio>
<!-- altre tracce audio -->
<audio>
  <sequence_number>005</sequence_number>
  <nomenclature>Traccia 5</nomenclature>
</audio>
</metadigit>

```

Nell'esempio vediamo che un unico elemento `<stru>` fa riferimento alla canzone *Certe Notti*; è stato poi inserito un `<element>` relativo alle immagini che riproducono il testo della canzone, e un `<element>` relativo alla traccia audio. Si noti che il valore degli attributi `sequence_number` degli elementi `<start>` e `<stop>` fanno riferimento alle sezioni `` e `<audio>` individuate dall'elemento `<resource>`.

6.2.4.2. Ricongiungimento con record di spoglio

L'esempio è relativo a un record MAG che raccoglie i dati delle digitalizzazioni e descrive separatamente il fascicolo di accompagnamento e il CD audio. Il collegamento fra testo della canzone e traccia audio viene realizzato grazie a una scheda di spoglio.

Il record madre:

```

<metadigit version="2.0.1">
  <gen>
    <stprog>http://biblioteca/mioprogetto.html</stprog>
    <agency>IT:AA000</agency>
    <access_rights>1</access_rights>
    <completeness>0</completeness>
  </gen>
  <bib level="m">
    <dc:identifier>BIB01</dc:identifier>
    <dc:title>Buon Compleanno Elvis</dc:title>
    <dc:creator>Ligabue</dc:creator>
    <dc:type>Registrazione sonora musicale</dc:type>
    <dc:type>Testo a stampa</dc:type>
  </bib>
  <stru>
    <sequence_number>1</sequence_number>
    <nomenclature>Buon Compleanno Elvis - Fascicolo</nomenclature>
    <stru>
      <sequence_number>1</sequence_number>
      <nomenclature>Certe notti</nomenclature>
    <element>
      <resource>img</resource>

```

```
<start sequence_number="002" />
<stop sequence_number="003" />
</element>
</stru>
<!-- altre canzoni testo-->
</stru>
<stru>
  <sequence_number>2</sequence_number>
  <nomenclature>Buon Compleanno Elvis - CD Audio</nomenclature>
  <stru>
    <sequence_number>1</sequence_number>
    <nomenclature>Certe notti</nomenclature>
    <element>
      <resource>audio</resource>
      <start sequence_number="005" />
      <stop sequence_number="005" />
    </element>
  </stru>
  <!-- altre canzoni audio -->
</stru>
<img>
  <sequence_number>001</sequence_number>
  <nomenclature>Copertina</nomenclature>
</img>
<img>
  <sequence_number>002</sequence_number>
  <nomenclature>Pagina 2</nomenclature>
</img>
<img>
  <sequence_number>003</sequence_number>
  <nomenclature>Pagina 3</nomenclature>
</img>
<!-- altre img -->
<audio>
  <sequence_number>001</sequence_number>
  <nomenclature>Traccia 1</nomenclature>
</audio>
<!-- altre tracce audio -->
<audio>
  <sequence_number>005</sequence_number>
  <nomenclature>Traccia 5</nomenclature>
</audio>
</metadigit>
```

Come si vede il record madre contiene i dati relativi alle digitalizzazioni del CD e del fascicolo, mentre la sezione STRU li descrive separatamente.

Si veda ora il record di spoglio che collega la traccia audio con il relativo testo della

canzone:

```
<metadigit version="2.0.1">
  <gen>
    <stprog>http://biblioteca/mioprogetto.html</stprog>
    <agency>IT:AA000</agency>
    <access_rights>1</access_rights>
    <completeness>0</completeness>
  </gen>
  <bib level="a">
    <dc:identifier>BIB01_TA</dc:identifier>
    <dc:title>Certe notti</dc:title>
    <dc:creator>Ligabue</dc:creator>
    <dc:type>Registrazione sonora musicale</dc:type>
    <dc:type>Testo a stampa</dc:type>
  </bib>
  <stru>
    <sequence_number>1</sequence_number>
    <nomenclature>Certe notti</nomenclature>
  <element>
    <nomenclature>Certe Notti (testo)</nomenclature>
    <dc:identifier>BIB01</dc:identifier>
    <resource>img</resource>
    <start sequence_number="002" />
    <stop sequence_number="003" />
  </element>
  <element>
    <nomenclature>Certe Notti (testo)</nomenclature>
    <dc:identifier>BIB01</dc:identifier>
    <resource>audio</resource>
    <start sequence_number="005" />
    <stop sequence_number="005" />
  </element>
</stru>
</metadigit>
```

Nell'esempio vediamo che un unico elemento `<stru>` fa riferimento alla canzone *Certe Notti*; è stato poi inserito un `<element>` relativo alle immagini che riproducono il testo della canzone, e un `<element>` relativo alla traccia audio. Si noti inoltre che in questo caso il valore degli attributi `sequence_number` degli elementi `<start>` e `<stop>` fanno riferimento alle sezioni `` e `<audio>` della scheda madre, individuata da `<dc:identifier>BIB01</dc:identifier>`.

6.2.4.3. Ricongiungimento con record di spoglio di secondo livello

L'ultimo caso presenta un'architettura più complessa che prevede tre livelli:

- scheda madre priva della documentazione relativa alla digitalizzazione di CD e fascicolo

- (fatta eccezione per la copertina del CD) e priva della sezione STRU.
- schede di spoglio relative a ciascuna traccia audio e a ciascuna testo contenuto nel fascicolo; la scheda contiene anche i dati relativi alla digitalizzazione (spoglio vestito)
 - schede di spoglio che collegano ciascuna traccia audio con il testo della canzone.

Si veda per prima la scheda madre:

```
<metadigit version="2.0.1">
  <gen>
    <stprog>http://biblioteca/mioprogetto.html</stprog>
    <agency>IT:AA000</agency>
    <access_rights>1</access_rights>
    <completeness>0</completeness>
  </gen>
  <bib level="m">
    <dc:identifier>BIB01</dc:identifier>
    <dc:title>Buon Compleanno Elvis</dc:title>
    <dc:creator>Ligabue</dc:creator>
    <dc:type>Registrazione sonora musicale</dc:type>
    <dc:type>Testo a stampa</dc:type>
  </bib>
  <img>
    <sequence_number>001</sequence_number>
    <nomenclature>Copertina</nomenclature>
  </img>
</metadigit>
```

Queste invece le due schede di spoglio relative alla canzone *Certe notti*, audio e testo:

```
<metadigit version="2.0.1">
  <gen>
    <stprog>http://biblioteca/mioprogetto.html</stprog>
    <agency>IT:AA000</agency>
    <access_rights>1</access_rights>
    <completeness>0</completeness>
  </gen>
  <bib level="a">
    <dc:identifier>BIB01_A</dc:identifier>
    <dc:title>Certe notti (audio)</dc:title>
    <dc:creator>Ligabue</dc:creator>
    <dc:type>Registrazione sonora musicale</dc:type>
  </bib>
  <audio>
    <sequence_number>005</sequence_number>
    <nomenclature>Traccia 5</nomenclature>
  </audio>
</metadigit>
```



```
<metadigit version="2.0.1">
  <gen>
    <stprog>http://biblioteca/mioprogetto.html</stprog>
    <agency>IT:AA000</agency>
    <access_rights>1</access_rights>
    <completeness>0</completeness>
  </gen>
  <bib level="a">
    <dc:identifier>BIB01_T</dc:identifier>
    <dc:title>Certe notti (testo)</dc:title>
    <dc:creator>Ligabue</dc:creator>
    <dc:type>Testo a stampa</dc:type>
  </bib>
  <img>
    <sequence_number>002</sequence_number>
    <nomenclature>Pagina 2</nomenclature>
  </img>
  <img>
    <sequence_number>003</sequence_number>
    <nomenclature>Pagina 3</nomenclature>
  </img>
</metadigit>
```

Infine la scheda di spoglio che collega audio e testo:

```
<metadigit version="2.0.1">
  <gen>
    <stprog>http://biblioteca/mioprogetto.html</stprog>
    <agency>IT:AA000</agency>
    <access_rights>1</access_rights>
    <completeness>0</completeness>
  </gen>
  <bib level="a">
    <dc:identifier>BIB01_TA</dc:identifier>
    <dc:title>Certe notti</dc:title>
    <dc:creator>Ligabue</dc:creator>
    <dc:type>Registrazione sonora musicale</dc:type>
    <dc:type>Testo a stampa</dc:type>
  </bib>
  <stru>
    <sequence_number>1</sequence_number>
    <nomenclature>Certe notti</nomenclature>
  <element>
    <nomenclature>Certe Notti (testo)</nomenclature>
```

```
<dc:identifier>BIB01_T</dc:identifier>
<resource>img</resource>
<start sequence_number="002" />
<stop sequence_number="003" />
</element>
<element>
  <nomenclature>Certe Notti (testo)</nomenclature>
  <dc:identifier>BIB01_A</dc:identifier>
  <resource>audio</resource>
  <start sequence_number="005" />
  <stop sequence_number="005" />
</element>
</stru>
</metadigit>
```

Nell'esempio vediamo che l'ultimo record contiene un unico elemento `<stru>` che fa riferimento alla canzone *Certe Notti*; è stato poi inserito un `<element>` relativo alle immagini che riproducono il testo della canzone, e un `<element>` relativo alla traccia audio. Si noti inoltre che in questo caso il valore degli attributi `sequence_number` degli elementi `<start>` e `<stop>` fanno riferimento alle sezioni `` e `<audio>` delle due schede di spoglio (individuate dai diversi `<dc:identifier>`).

6.2.5. La descrizione di `<element>`

Riassumendo, `<element>` è opzionale e ripetibile. Per l'elemento sono definiti i seguenti attributi:

- **num** : serve a dichiarare l'ordine degli `<element>` nel caso di un oggetto digitale logicamente unitario ma fisicamente diviso in più oggetti analogici dello stesso tipo. È di tipo `xsd:positiveInteger` (vale dire che può contenere solo numeri positivi) ed è opzionale
- **descr** : mantenuto solo per compatibilità con le precedenti versioni MAG, contiene la denominazione di uno `<stru>`. Il suo uso è deprecato in favore dell'elemento `<nomenclature>`. L'attributo è di tipo `xsd:string` ed è opzionale.

L'elemento `<element>` è di tipo `xsd:sequence` e può contenere:

- **<nomenclature>** : descrive l'elemento. Di tipo `xsd:string`, è opzionale e non ripetibile. Non sono definiti attributi.
- **<file>** : si usa quando si vuole puntare a un record MAG privo di `<dc:identifier>`, oppure senza ricorrere all'uso di `<dc:identifier>` e `<piece>`. È di tipo `link`, vale a dire che è un elemento vuoto che supporta attributi definiti dal namespace `xlink` (p. 33). L'elemento non è ripetibile e opzionale; il suo uso è sconsigliato.
- **<dc:identifier>** : identificatore della risorsa cui fa riferimento il livello strutturale, definito all'interno della sezione `<bib>` (p. 48) di un altro record MAG. Se i contenuti digitali definiti da `<resource>` sono descritti all'interno del medesimo record, si omette. L'elemento è opzionale e non ripetibile

- **<piece>** : dichiara l'unità fisica di una pubblicazione seriale; l'elemento è descritto all'interno della sezione <bib> (p. 55). L'elemento è opzionale e non ripetibile
- **<resource>** : dichiara il tipo di risorsa digitale a cui fa riferimento il livello strutturale. Di tipo `resource_type`, può avere i seguenti valori: `img`, `audio`, `video`, `doc`, `ocr`. Se omesso il riferimento si intende alla sezione . L'elemento è opzionale e non ripetibile
- **<start>** : è un elemento vuoto. Permette di individuare l'inizio dell'intervallo di attribuzione dei contenuti definiti da <resource>, all'interno del documento indicato da <dc:identifier> o all'interno del medesimo record nel caso <dc:identifier> sia assente. L'elemento è opzionale e non ripetibile. Per l'elemento sono definiti due attributi:
 - **sequence_number**: di tipo `xsd:positiveInteger`, è obbligatorio. Individua un singolo contenuto, nell'ambito della sezione definita da <resource>
 - **offset**: di tipo `xsd:time` è opzionale. Fornisce un marcatore temporale all'interno di un flusso (applicabile solo ad `audio` e `video`).
- **<stop>** : è un elemento vuoto. Permette di individuare la fine dell'intervallo di attribuzione dei contenuti definiti da <resource>, all'interno del documento indicato da <dc:identifier> o all'interno del medesimo record nel caso <dc:identifier> sia assente. L'elemento è opzionale e non ripetibile. Per l'elemento sono definiti due attributi:
 - **sequence_number**: di tipo `xsd:positiveInteger`, è obbligatorio. Individua un singolo contenuto, nell'ambito della sezione definita da <resource>
 - **offset**: di tipo `xsd:time` è opzionale. Fornisce un marcatore temporale all'interno di un flusso (applicabile solo ad `audio` e `video`).

L'elemento <element> è così formalmente definito (file `metatype.xsd`):

```
<xsd:complexType name="stru_element">
  <xsd:sequence>
    <xsd:element name="nomenclature" minOccurs="0"/>
    <xsd:element name="file" type="link" minOccurs="0"/>
    <xsd:element ref="dc:identifier" minOccurs="0"/>
    <xsd:element name="piece" minOccurs="0">
      <xsd:complexType>
        <xsd:choice minOccurs="0">
          <xsd:sequence>
            <xsd:element name="year" type="xsd:string"/>
            <xsd:element name="issue" type="xsd:string"/>
            <xsd:element name="stpiece_per" type="SICI"/>
          </xsd:sequence>
          <xsd:sequence>
            <xsd:element name="part_number" type="xsd:positiveInteger"/>
            <xsd:element name="part_name" type="xsd:string"/>
            <xsd:element name="stpiece_vol" type="BICI"/>
          </xsd:sequence>
        </xsd:choice>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
```

```
        </xsd:sequence>
      </xsd:choice>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="resource" type="resource_type" default="img" minOccurs="0">
    <xsd:annotation>
      <xsd:documentation>se omissso, vale IMG</xsd:documentation>
    </xsd:annotation>
  </xsd:element>
  <xsd:element name="start" minOccurs="0">
    <xsd:complexType>
      <xsd:attribute name="sequence_number" type="xsd:positiveInteger" use="required"/>
      <xsd:attribute name="offset" type="xsd:time" use="optional"/>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="stop" minOccurs="0">
    <xsd:annotation>
      <xsd:documentation>se omissso, coincide con START</xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:attribute name="sequence_number" type="xsd:positiveInteger" use="required"/>
      <xsd:attribute name="offset" type="xsd:time" use="optional"/>
    </xsd:complexType>
  </xsd:element>
</xsd:sequence>
<xsd:attribute name="descr" type="xsd:string" use="optional"/>
<xsd:attribute name="num" type="xsd:positiveInteger" use="optional"/>
</xsd:complexType>
```

Il tipo complesso resource_type richiamato dall'elemento <resource> è così formalmente descritto (file metatype.xsd):

```
<xsd:simpleType name="resource_type">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="img"/>
    <xsd:enumeration value="audio"/>
    <xsd:enumeration value="video"/>
    <xsd:enumeration value="ocr"/>
    <xsd:enumeration value="doc"/>
  </xsd:restriction>
</xsd:simpleType>
```

7. Sezione IMG

La sezione IMG raccoglie i metadati amministrativi e gestionali relativi alle immagini statiche. Alcuni di questi dati, in realtà, possono essere raccolti direttamente all'interno della sezione GEN, grazie all'elemento `<img_group>`, il cui contenuto verrà tuttavia trattato in questa sezione per omogeneità tematica.

La sezione IMG utilizza il namespace `niso`: che fa riferimento a uno schema che traduce le linee guida del Data Dictionary NISO. Tale schema è stato realizzato dal Comitato MAG e verrà quindi qui interamente documentato di volta in volta nei successivi paragrafi e complessivamente nel paragrafo **Lo schema Niso** (p. 89).

La sezione IMG è costituita di una sequenza di elementi ``, uno per ciascuna immagine digitale descritta da MAG. L'elemento è opzionale e ripetibile. Il suo contenuto è di tipo `xsd:sequence`, e può contenere i seguenti elementi:

- `<sequence_number>` : contiene il numero di sequenza identificativo dell'immagine. Obbligatorio e non ripetibile
- `<nomenclature>` : contiene la denominazione o titolo dell'immagine. Obbligatorio e non ripetibile
- `<usage>` : definisce l'ambito d'uso dell'immagine in relazione agli standard di progetto. Opzionale e ripetibile
- `<side>` : indica se l'immagine acquisita comprende una o due pagine del libro. Opzionale e non ripetibile
- `<scale>` : indica la presenza di una scala millimetrica in fase di digitalizzazione. Opzionale e non ripetibile
- `<file>` : localizza il file contenente l'immagine. Obbligatorio e non ripetibile
- `<md5>` : contiene l'impronta del file. Obbligatorio e non ripetibile
- `<filesize>` : fornisce la dimensione del file contenente l'immagine in byte. Opzionale e non ripetibile
- `<image_dimensions>` : definisce le dimensioni dell'immagine digitale. Obbligatorio e non ripetibile
- `<image_metrics>` : fornisce le principali caratteristiche tecniche dell'immagine secondo lo standard NISO. Opzionale e non ripetibile
- `<ppi>` : risoluzione dell'immagine espressa in ppi. Opzionale e non ripetibile
- `<dpi>` : risoluzione dell'immagine espressa in dpi. Opzionale e non ripetibile
- `<format>` : dichiara formato dell'immagine secondo lo standard NISO. Opzionale e non ripetibile
- `<scanning>` : registra le modalità di scansione dell'immagine. Opzionale e non ripetibile
- `<datetimecreated>` : dichiara la data e l'ora di creazione dell'immagine. Opzionale e non ripetibile
- `<target>` : indica la presenza di un target (scala cromatica) durante scansione dell'immagine secondo lo standard NISO. Opzionale e ripetibile
- `<altimg>` : contiene la descrizione di un eventuale altro formato della medesima immagine. Opzionale e ripetibile
- `<note>` : eventuali annotazioni all'immagine. Opzionale e non ripetibile. L'elemento è presente in ogni sezione; è di tipo `xsd:string` e può contenere qualsiasi tipo di annotazione.

Per l'elemento sono inoltre definiti i seguenti attributi:

- **imggroupID** : di tipo `xsd:IDREF` contiene un riferimento all'attributo `ID` dell'elemento `<img_group>` (p. 42). Tale attributo consente di collegare un `` con le caratteristiche tecniche definite globalmente da `<img_group>`. L'attributo è opzionale; qualora non sia usato si assume che le caratteristiche tecniche dell'immagine non siano state altrove descritte e quindi l'elemento `<image_metrics>` deve ritenersi obbligatorio, così come `<format>` e `<scanning>`.
- **holdingsID** : di tipo `xsd:IDREF` contiene un riferimento all'attributo `ID` dell'elemento `<holdings>` (p. 53) e serve a definire a quale istituzione appartiene l'oggetto analogico digitalizzato. L'attributo è opzionale.

L'elemento `` è così formalmente definito (file `metatype.xsd`):

```
<xsd:complexType name="img">
  <xsd:sequence>
    <xsd:element name="sequence_number" type="xsd:positiveInteger"/>
    <xsd:element name="nomenclature" type="xsd:string"/>
    <xsd:element name="usage" type="usages" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="side" type="side" minOccurs="0"/>
    <xsd:element name="scale" type="millimetric_scale" minOccurs="0"/>
    <xsd:element name="file" type="link"/>
    <xsd:element name="md5" type="niso:checksum"/>
    <xsd:element name="filesize" type="xsd:positiveInteger" minOccurs="0"/>
    <xsd:element name="image_dimensions" type="niso:dimensions"/>
    <xsd:element name="image_metrics" type="niso:spatialmetrics" minOccurs="0"/>
    <xsd:element name="ppi" type="xsd:positiveInteger" minOccurs="0"/>
    <xsd:element name="dpi" type="xsd:positiveInteger" minOccurs="0"/>
    <xsd:element name="format" type="niso:format" minOccurs="0"/>
    <xsd:element name="scanning" type="niso:image_creation" minOccurs="0"/>
    <xsd:element name="datetimecreated" type="xsd:dateTime" minOccurs="0"/>
    <xsd:element name="target" type="niso:targetdata" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="altimg" minOccurs="0" maxOccurs="unbounded">
      <xsd:complexType>
        <!-- omissis -->
      </xsd:complexType>
    </xsd:element>
    <xsd:element name="note" type="xsd:string" minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="imggroupID" type="xsd:IDREF" use="optional"/>
  <xsd:attribute name="holdingsID" type="xsd:IDREF" use="optional"/>
</xsd:complexType>
```

7.1. Identificazione, denominazione e uso delle immagini

Ciascuna immagine descritta all'interno di MAG viene identificata univocamente grazie all'elemento `<sequence_number>`. Il suo contenuto è di tipo `xsd:positiveInteger`,

vale a dire che è costituito di un numero positivo. Dato il suo compito di identificatore, sul contenuto di <sequence_number> è stato posto un vincolo di univocità (file metadigit.xsd):

```
<xsd:unique name="uniquimg">
  <xsd:selector xpath="img"/>
  <xsd:field xpath="sequence_number"/>
</xsd:unique>
```

L'elemento è obbligatorio e non ripetibile. Il valore dell'elemento è puntato dagli attributi sequence_number degli elementi <start> e <stop> della sezione <stru> (p. 61). Per esempio:

```
<stru>
  <sequence_number>1</sequence_number>
  <nomenclature>Introduzione</nomenclature>
  <element>
    <resource>img</resource>
    <start sequence_number="002"/>
    <stop sequence_number="003"/>
  </element>
</stru>
<img>
  <sequence_number>002</sequence_number>
  <nomenclature>Pagina 2</nomenclature>
</img>
<img>
  <sequence_number>003</sequence_number>
  <nomenclature>Pagina 3</nomenclature>
</img>
```

A ciascuna immagine deve inoltre essere attribuita una denominazione, per esempio Pagina 1, Carta 2v, ecc. Tale denominazione viene codificata dall'elemento <nomenclature>. L'elemento è di tipo xsd:string; si consiglia comunque di definire una nomenclatura controllata negli standard di progetto. L'elemento è obbligatorio e non ripetibile

Dello stesso oggetto digitale (tipicamente un foglio di carta) possono essere tratte più immagini digitali, più o meno definite, in diversi formati, ognuna delle quali con una diversa finalità. È infatti usuale creare immagini di alta qualità per l'archiviazione interna e immagini di qualità più limitata per la diffusione esterna. La finalità dell'immagine digitale viene registrata dall'elemento <usage>. L'elemento è di tipo xsd:string; al fine di favorire la portabilità dei dati, si consiglia tuttavia di adottare le seguenti due tassonomie (adottate dai maggiori progetti di digitalizzazione italiani), la prima relativa alle modalità d'uso, la seconda al possesso del copyright da parte dell'istituzione:

- 1 : master
- 2 : alta risoluzione
- 3 : bassa risoluzione
- 4 : preview

e

- **a** : il repository non ha il copyright dell'oggetto digitale
- **b** : il repository ha il copyright dell'oggetto digitale

L'elemento è opzionale e ripetibile.

7.2. Tipo di scansione

Esattamente come per la fotocopiatura, la scansione di un oggetto analogico può procedere in vario modo, è possibile infatti procedere per una pagina alla volta oppure per pagine affiancate. Tale informazione può essere registrata grazie all'elemento opzionale e non ripetibile `<side>`, per il quale è definito un tipo semplice specializzato denominato a sua volta `side`. Tale tipo è definito come restrizione di `xsd:string` ed è costituito dall'enumerazione dei seguenti valori:

- **left** : l'immagine contiene la digitalizzazione della pagina sinistra di un volume o di un fascicolo
- **right** : l'immagine contiene la digitalizzazione della pagina destra di un volume o di un fascicolo
- **double** : l'immagine contiene la digitalizzazione di una doppia pagina di un volume o di un fascicolo
- **part** : l'immagine contiene la digitalizzazione parziale dell'oggetto analogico fonte.

Il tipo è formalmente definito come segue (file `metatype.xsd`):

```
<xsd:simpleType name="side">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="left"/>
    <xsd:enumeration value="right"/>
    <xsd:enumeration value="double"/>
    <xsd:enumeration value="part"/>
  </xsd:restriction>
</xsd:simpleType>
```

Durante la scansione è possibile impiegare una scala millimetrica da affiancare all'oggetto sottoposto a scansione in modo da ricostruire le dimensioni dell'originale partendo dalla sua riproduzione digitale. L'informazione può essere registrata grazie all'elemento opzionale e non ripetibile `<scale>` per il quale è definito un tipo semplice specializzato denominato `millimetric_scale`. Tale tipo è definito come restrizione di `xsd:string` ed è costituito dall'enumerazione dei seguenti valori:

- **0** : non è presente alcuna scala millimetrica
- **1** : è presente una scala millimetrica

Il tipo è formalmente definito come segue (file `metatype.xsd`):

```
<xsd:simpleType name="millimetric_scale">
```



```
<xsd:restriction base="xsd:integer">
  <xsd:enumeration value="0"/>
  <xsd:enumeration value="1"/>
</xsd:restriction>
</xsd:simpleType>
```

7.3. Localizzazione, integrità e grandezza del file

L'elemento `<file>` consente di localizzare il file che contiene l'immagine digitale. È di tipo `link`, vale a dire che è un elemento vuoto che supporta attributi definiti dal namespace `xlink` (p. 33). L'elemento è obbligatorio e non ripetibile.

L'integrità del contenuto digitale è verificata grazie alla sua impronta digitale, registrata dall'elemento `<md5>`, un codice standard di 32 caratteri che viene rilevato automaticamente grazie all'impiego di appositi applicativi. Le regole per il rilevamento dell'impronta devono essere definite localmente, così come i momenti per il rilievo stesso (prima del momento del deposito, al momento del deposito, o in entrambi i momenti). Si tratta di una raccomandazione NISO e come tale il tipo specializzato che governa il contenuto dell'elemento appartiene al namespace `niso` ed è denominato `niso:checksum` (p. 89). Tale tipo è definito come restrizione di `xsd:string` che limita la lunghezza massima della stringa a 32 caratteri.

La grandezza del file (che va espressa in byte) è registrata dall'elemento `<filesize>`. L'elemento è di tipo `xsd:Integer` (un numero positivo), è opzionale e non ripetibile. Anche l'elemento `<filesize>` è una raccomandazione NISO (Cfr. [Data Dictionary](#), p. 13).

Si veda il seguente esempio relativo all'uso degli elementi visti finora della sezione IMG (che presuppone l'uso dell'elemento `<img_group>` come testimoniato dalla presenza dell'attributo `imggroupID`):

```
<img imggroupID="IG1">
  <sequence_number>1</sequence_number>
  <nomenclature>dorso</nomenclature>
  <usage>2</usage>
  <scale>0</scale>
  <file Location="URL" xlink:href="Mus_1094/IG1/Mus_1094_001_00_dorso.jpg"/>
  <md5>ec6cdf613640e1e212ac91dd65aefb21</md5>
  <filesize>32248860</filesize>
  <!-- omissis -->
</img>
```

7.4. Le dimensioni dell'immagine digitale

Le dimensioni dell'immagini digitale sono codificate grazie all'elemento `<image_dimensions>`, per il quale è definito un tipo complesso specializzato, `niso:dimensions`, appartenente al namespace `niso` (p. 89). L'elemento `<image_dimensions>` è obbligatorio e non ripetibile. Il tipo `niso:dimensions` è definito come `xsd:sequence` e comprende i seguenti elementi:

- **<niso:imagelength>** : contiene la lunghezza dell'immagine, vale a dire la dimensione verticale espressa in pixel. L'elemento è obbligatorio e non ripetibile. Raccomandazione NISO ([Data Dictionary](#) , p. 23)
- **<niso:imagewidth>** : contiene la larghezza dell'immagine, vale a dire la dimensione orizzontale espressa in pixel. L'elemento è obbligatorio e non ripetibile. Raccomandazione NISO ([Data Dictionary](#) , p. 22)
- **<niso:source_xdimension>** : contiene la larghezza (dimensione orizzontale) dell'oggetto analogico digitalizzato espresso in pollici (inches). L'elemento è opzionale e non ripetibile. Raccomandazione NISO ([Data Dictionary](#) , pp. 25-26)
- **<niso:source_ydimension>** : contiene la lunghezza (dimensione verticale) dell'oggetto analogico digitalizzato espresso in pollici (inches). L'elemento è opzionale e non ripetibile. Raccomandazione NISO ([Data Dictionary](#) , p. 26)

Si veda il seguente esempio d'uso di <image_dimensions>:

```
<image_dimensions>
  <niso:imagelength>906</niso:imagelength>
  <niso:imagewidth>743</niso:imagewidth>
  <niso:source_xdimension>10.319445</niso:source_xdimension>
  <niso:source_ydimension>12.583333</niso:source_ydimension>
</image_dimensions>
```

7.5. Principali caratteristiche tecniche dell'immagine

Le principali caratteristiche tecniche dell'immagine sono contenute nell'elemento <image_metrics> e sono codificate secondo lo standard NISO. L'elemento è, oltre che dentro , può essere usato dentro <img_group> (p. 42); è formalmente opzionale, ma in pratica deve sempre essere usato, o dentro o dentro <img_group>. Per <img_metrics> è definito un tipo specializzato appartenente al namespace niso (p. 89) denominato niso:spatialmetrics. Tale tipo è di tipo xsd:sequence e contiene sei elementi:

- **<niso:samplingfrequencyunit>** : obbligatorio e non ripetibile, definisce l'unità di misura usata per il contenuto degli elementi <niso:xsamplingfrequency> e <niso:ysamplingfrequency>. Per l'elemento è definito un tipo semplice specializzato, anch'esso contenuto nel file niso-mag.xsd, denominato niso:samplingfrequencyunittype. Tale tipo è definito come restrizione di xsd:string ed è costituito dall'enumerazione dei seguenti valori:
 - 1 : nessuna unità di misura definita
 - 2 : inch, pollice
 - 3 : centimetro
- **<niso:samplingfrequencyplane>** : obbligatorio e non ripetibile, dichiara il piano focale del campionamento. Per l'elemento è definito un tipo semplice specializzato, anch'esso contenuto nel file niso-mag.xsd, denominato niso:samplingfrequencyplanetype. Tale tipo è definito come restrizione di xsd:string ed è costituito dall'enumerazione dei seguenti valori:

- **1** : camera/scanner focal plane, quando non sono definite le dimensioni dell'oggetto che si sta digitalizzando (per es. quando si riproduce con una fotocamera)
- **2** : object plane, quando l'oggetto e la riproduzione hanno la stessa dimensione (per es. quando si riproduce con uno scanner)
- **3** : source object plane, quando la dimensione della riproduzione è maggiore dell'oggetto originale (per es. quando si riproduce da un microfilm)

- **<niso:xsamplingfrequency>** : opzionale (ma obbligatorio se applicabile) e non ripetibile, contiene la frequenza di campionamento nella direzione orizzontale, presente in alternativa a **<ppi>** e **<dpi>** (p. 84) (elementi obsoleti), con **<niso:samplingfrequencyunit>** = 2 (inch) o 3 (centimetro); con 1 il campo è nullo. Il suo contenuto è di tipo **xsd:positiveInteger**, vale a dire un numero positivo
- **<niso:ysamplingfrequency>** : opzionale (ma obbligatorio se applicabile) e non ripetibile, contiene la frequenza di campionamento nella direzione verticale, presente in alternativa a **<ppi>** e **<dpi>** (p. 84) (elementi obsoleti), con **<niso:samplingfrequencyunit>** = 2 (inch) o 3 (centimetro); con 1 il campo è nullo. Il suo contenuto è di tipo **xsd:positiveInteger**, vale a dire un numero positivo
- **<niso:photometricinterpretation>** : obbligatorio e non ripetibile, definisce l'interpretazione fotometrica dei bit del campione. Per l'elemento è definito un tipo semplice specializzato, anch'esso contenuto nel file **niso-mag.xsd**, denominato **niso:photometricinterpretationtype**. Tale tipo è definito come restrizione di **xsd:string** ed è costituito dall'enumerazione dei seguenti valori:
 - WhiteIsZero
 - BlackIsZero
 - RGB
 - Palette color
 - Transparency Mask
 - CMYK
 - YcbCr
 - CIELab

- **<niso:bitpersample>** : obbligatorio e non ripetibile, definisce il numero di bit per ciascun campione, esplicitando il rapporto profondità/colore. Per l'elemento è definito un tipo semplice specializzato, anch'esso contenuto nel file **niso-mag.xsd**, denominato **niso:bitpersampletype**. Tale tipo è definito come restrizione di **xsd:string** ed è costituito dall'enumerazione dei seguenti valori:
 - **1** : bitonale, bianco e nero
 - **4** : scala di grigi a 4 bit
 - **8** : scala di grigi o gamma di 256 colori a 8 bit
 - **8,8,8** : colori RGB a 24 bit
 - **16,16,16** : per immagini TIFF o HDR a 48 bit
 - **8,8,8,8** : CMYK a 32 bit

Si veda ora il seguente esempio d'uso dell'elemento **<image_metrics>**:

```
<image_metrics>
  <niso:samplingfrequencyunit>2</niso:samplingfrequencyunit>
  <niso:samplingfrequencyplane>3</niso:samplingfrequencyplane>
  <niso:xsamplingfrequency>300</niso:xsamplingfrequency>
  <niso:ysamplingfrequency>300</niso:ysamplingfrequency>
  <niso:photometricinterpretation>RGB</niso:photometricinterpretation>
  <niso:bitpersample>8,8,8</niso:bitpersample>
</image_metrics>
```

7.5.1. La risoluzione delle immagini: misurazioni obsolete

Lo schema MAG contiene la coppia di elementi `<ppi>` e `<dpi>`, opzionali e non ripetibili, che consentono di registrare la risoluzione spaziale delle immagini, in entrambe le direzioni, orizzontale e verticale, di un inch (pollice) quadrato. Le due categorie misurano rispettivamente:

- `<ppi>` : pixel per inch, cioè il numero di pixel presenti per ogni pollice quadrato
- `<dpi>` : dots per inch, cioè il numero di punti presenti per ogni pollice quadrato. Si applica propriamente agli output (testo, immagini) prodotti dalle stampanti e non alle immagini memorizzate su supporto digitale; tale misurazione è tuttavia normalmente utilizzata anche all'interno di progetti di digitalizzazione.

Tali misurazioni sono sconsigliate in quanto obsolete e imprecise; sono conservate solo per garantire la compatibilità con le versioni precedenti di MAG. Si consiglia, invece di usare `<niso:xsamplingfrequency>` e `<niso:ysamplingfrequency>` all'interno di `<image_metrics>`. L'uso di `<dpi>` e `<ppi>` di fatto equivalgono a `<niso:xsamplingfrequency>` e `<niso:ysamplingfrequency>` con valori uguali e con `<niso:samplingfrequencyunit> = 2`.

Entrambi gli elementi sono definiti come `xsd:positiveInteger` e sono opzionali. Entrambi gli elementi possono essere inclusi anche dentro `<img_group>`.

7.6. Il formato delle immagini

Il formato delle immagini (tipologia e modalità di compressione) è gestito dall'elemento `<format>` ed è codificato secondo lo standard NISO. L'elemento, oltre che dentro ``, può essere usato dentro `<img_group>` (p. 42); è formalmente opzionale, ma in pratica deve sempre essere usato, o dentro `` o dentro `<img_group>`. Per `<format>` è definito un tipo specializzato appartenente al namespace `niso` (p. 89) denominato `niso:format` che presenta una sequenza di tre elementi:

- `<niso:name>` : obbligatorio e non ripetibile, contiene il formato dell'immagine. È di tipo `xsd:string`, si consiglia di usare valori come `JPG`, `GIF`, `TIF`, `PDF` ecc. Si raccomanda di usare valori formati da tre caratteri. Una sintassi alternativa può essere adottata per i formati che codificano il numero di revisione nel file header: `[formato file][numero di revisione]`, per esempio: `TIFF/EP 1.0.0.0`.

- **<niso:mime>** : obbligatorio e non ripetibile, contiene il tipo mime dell'immagine. Per l'elemento è definito un tipo semplice specializzato, anch'esso contenuto nel file `niso-mag.xsd`, denominato `niso:img_mimetype`. Tale tipo è definito come restrizione di `xsd:string` ed è costituito dall'enumerazione dei seguenti valori (si noti la presenza del formato PDF, che per sua natura si presta sia per contenuti testuali – e come tale il valore è previsto fra i mime type delle sezioni OCR e DOC –, sia per veicolare immagini):
 - `image/jpeg`
 - `image/tiff`
 - `image/gif`
 - `image/png`
 - `image/vnd.djvu`
 - `application/pdf`

- **<niso:compression>** : obbligatorio e non ripetibile, dichiara il tipo di compressione applicato all'immagine. Per l'elemento è definito un tipo semplice specializzato, anch'esso contenuto nel file `niso-mag.xsd`, denominato `niso:compressiontype`. Tale tipo è definito come restrizione di `xsd:string` ed è costituito dall'enumerazione dei seguenti valori:
 - `Uncompressed`
 - `CCITT 1D`
 - `CCITT Group 3`
 - `CCITT Group 4`
 - `LZW`
 - `JPG`
 - `PNG`
 - `DJVU`

Si veda un esempio di uso dell'elemento `<format>`:

```
<format>
  <niso:name>JPG</niso:name>
  <niso:mime>image/jpeg</niso:mime>
  <niso:compression>JPG</niso:compression>
</format>
```

7.7. La scansione dell'oggetto

Le modalità della scansione dell'oggetto sono codificate dell'elemento `<scanning>` secondo lo standard NISO. L'elemento è, oltre che dentro ``, può essere usato dentro `<img_group>` (p. 42); è formalmente opzionale, ma in pratica deve sempre essere usato, o dentro `` o dentro `<img_group>`. Per `<scanning>` è definito un tipo specializzato appartenente al namespace `niso` (p. 89) denominato `niso:image_creation`. È di tipo `xsd:sequence` e contiene quattro elementi tutti opzionali; si consiglia comunque di usarne

almeno uno. Gli esempi dei valori sono desunte dalla normativa ICCD per la catalogazione delle fotografie: <http://www.iccd.beniculturali.it/download/schedaf.pdf> :

- **<niso:sourcetype>** : opzionale e non ripetibile, descrive le caratteristiche fisiche del supporto analogico di partenza. Di tipo `xsd:string`, si suggerisce comunque di adottare uno dei seguenti valori:
 - **negativo** : per immagini fotografiche i cui valori tonali risultino invertiti rispetto a quelli del soggetto raffigurato e che permettono di produrre un numero illimitato di "positivi"
 - **positivo** : per immagini fotografiche, ottenute da "negativi", i cui valori tonali corrispondano a quelli del soggetto raffigurato; sono da considerarsi "positivi" anche i prodotti ottenuti da matrici virtuali attraverso stampanti, plotter, etc.
 - **diapositiva** : per immagini fotografiche positive realizzate su supporti trasparenti e visibili per trasparenza o per proiezione
 - **unicum** : per immagini fotografiche "uniche", ottenute cioè senza mediazione di "negativi" e che, a loro volta, non possono essere utilizzate come "matrici"; sono da considerarsi "unicum", ad esempio, dagherrotipi, ambrotipi, ferrotipi, polaroid ed inoltre prodotti unici ottenuti con procedimenti elettronici analogico-digitali, come fax o fotocopie
 - **fotografia virtuale** : per "matrici virtuali", cioè per immagini latenti memorizzate su memorie di massa analogiche, analogico-digitali e digitali
 - **vario: .../...** : per oggetti complessi e/o composti costituiti da elementi appartenenti a categorie diverse. Es.: vario: positivo/unicum; vario: unicum/positivo/fotografia virtuale
- **<niso:scanningagency>** : opzionale e non ripetibile, contiene il nome della persona, società o ente produttore dell'immagine digitale, cioè dell'entità che ha realizzato la scansione. È di tipo `xsd:string`. Se assente, si assume che la scansione sia stata effettuata all'interno dell'istituzione responsabile del progetto di digitalizzazione.
- **<niso:devicesource>** : opzionale e non ripetibile, descrive la tipologia dell'apparecchiatura di scansione, per esempio "scanner", "fotocamera digitale", "videocamera". È di tipo `xsd:string`.
- **<niso:scanningsystem>** : opzionale e non ripetibile, descrive il dispositivo usato per la scansione attraverso tre diversi elementi definiti grazie a un tipo complesso anonimo:
 - **<niso:scanner_manufacturer>** : obbligatorio e non ripetibile, contiene il nome del produttore del dispositivo. È di tipo `xsd:string`.
 - **<niso:scanner_model>** : obbligatorio e non ripetibile, contiene la marca e il modello dell'apparecchiatura di acquisizione. È di tipo `xsd:string`.
 - **<niso:capture_software>** : obbligatorio e non ripetibile, contiene il nome del software usato per l'acquisizione dell'immagine. È di tipo `xsd:string`.

Si veda ora un esempio di uso dell'elemento `<scanning>`

`<scanning>`

```
<niso:scanningagency>nome azienda</niso:scanningagency>
<niso:devicesource>scanner</niso:devicesource>
<niso:scanningsystem>
  <niso:scanner_manufacturer>Zeutschel</niso:scanner_manufacturer>
  <niso:scanner_model>OS10000 TT</niso:scanner_model>
  <niso:capture_software>OmniScan 10.01</niso:capture_software>
</niso:scanningsystem>
</scanning>
```

7.8. La creazione del file

L'elemento `<datetimecreated>` registra la data e l'ora di creazione del file digitale. L'elemento è obbligatorio e non ripetibile; è di tipo `xsd:dateTime`, vale a dire che assume la forma `YYYY-MM-DDThh:mm:ss:mmm` di cui si vedano le specificazioni nella sezione GEN (p. 38).

Per esempio:

```
<datetimecreated>2005-04-13T02:01:52</datetimecreated>
```

7.9. Il target

L'eventuale presenza, la tipologia e le modalità d'utilizzo di un target (o scala cromatica) durante la scansione dell'oggetto analogico è identificata dalla sezione codificata dall'elemento `<target>`, secondo lo standard NISO. L'elemento è opzionale e non ripetibile. Per `<target>` è definito un tipo specializzato appartenente al namespace `niso` (p. 89) denominato `niso:targetdata`. Tale tipo è di tipo `xsd:sequence` e contiene cinque elementi:

- **<niso:targetType>** : opzionale e non ripetibile, dichiara se il target è interno o esterno. Per l'elemento è definito un tipo semplice specializzato, anch'esso contenuto nel file `niso-mag.xsd`, denominato `niso:targettype`. Tale tipo è definito come restrizione di `xsd:string` ed è costituito dall'enumerazione dei seguenti valori:
 - 0 il target è esterno
 - 1 il target è interno
- **<niso:targetID>** : obbligatorio e non ripetibile, identifica il nome del target, produttore o organizzazione, il numero della versione o il media. È di tipo `xsd:string`.
- **<niso:imageData>** : opzionale e non ripetibile, identifica il path dell'immagine digitale che funge da target esterno. È di tipo `xsd:anyURI`. Si usa solo se `<niso:targetType>` è uguale a 0 (esterno).
- **<niso:performanceData>** : opzionale e non ripetibile, identifica il path del file che contiene i dati dell'immagine performance relativa al target identificato da `<niso:targetID>`. È di tipo `xsd:anyURI`.
- **<niso:profiles>** : opzionale e non ripetibile, identifica il path del file che contiene il profilo dei colori ICC o un altro profilo di gestione. È di tipo `xsd:anyURI`.

7.10. Formati alternativi dell'immagine

L'elemento `<altimg>` codifica la presenza di eventuali formati alternativi rispetto a quello considerato master (la politica d'uso è a carico della singola istituzione). La struttura dell'elemento ripropone in modo semplificato quello di `` e infatti contiene:

- `<usage>` : opzionale e ripetibile; vd. par. **Identificazione, denominazione e uso delle immagini** (p. 78)
- `<file>` : obbligatorio e non ripetibile; vd. par. **Localizzazione, integrità e grandezza del file** (p. 81)
- `<md5>` : obbligatorio e non ripetibile; vd. par. **Localizzazione, integrità e grandezza del file** (p. 81)
- `<filesize>` : opzionale e non ripetibile; vd. par. **Localizzazione, integrità e grandezza del file** (p. 81).
- `<image_dimensions>` : obbligatorio e non ripetibile; vd. par. **Le dimensioni dell'immagine digitale** (p. 81)
- `<image_metrics>` : opzionale, se le caratteristiche sono descritte in `<img_group>` (p. 42), e non ripetibile. Vd. par. **Principali caratteristiche tecniche dell'immagine** (p. 82)
- `<ppi>` : opzionale e non ripetibile, sconsigliato; vd. par. **La risoluzione delle immagini: misurazioni obsolete** (p. 84)
- `<dpi>` : opzionale e non ripetibile, sconsigliato; vd. par. **La risoluzione delle immagini: misurazioni obsolete** (p. 84)
- `<format>` : opzionale, se le caratteristiche sono descritte in `<img_group>` (p. 42), e non ripetibile. Vd. par. **Il formato delle immagini** (p. 84)
- `<scanning>` : opzionale, se le caratteristiche sono descritte in `<img_group>` (p. 42), e non ripetibile. Vd. par. **La scansione dell'oggetto** (p. 85)
- `<datetimecreated>` : opzionale e non ripetibile; vd. par. **La creazione del file** (p. 87). Se assente si assume che coincida con quello relativo all'immagine principale.

Come per `` è definito l'attributo opzionale `imggroupID` che fa riferimento all'attributo `ID` di `<img_group>` che consente di collegare un `<altimg>` con le caratteristiche tecniche definite globalmente da `<img_group>`.

Vediamo ora un esempio d'uso dell'elemento `<altimg>`:

```
<altimg imggroupID="IG2">
  <usage>1</usage>
  <file Location="URL" xlink:href="/AFR/1.A.5/300/0001.JPG" />
  <md5>685c472fa3e39e2335f2e5816b9309a7</md5>
  <filesize>312959</filesize>
  <image_dimensions>
    <niso:imagelength>2681</niso:imagelength>
    <niso:imagewidth>682</niso:imagewidth>
  </image_dimensions>
  <datetimecreated>2005-10-11T10:37:14</datetimecreated>
</altimg>
```

L'elemento è formalmente definito come segue (file `metatype.xsd`):


```
<xsd:element name="altimg" minOccurs="0" maxOccurs="unbounded">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="usage" type="usages" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="file" type="link"/>
      <xsd:element name="md5" type="niso:checksum"/>
      <xsd:element name="filesize" type="xsd:positiveInteger" minOccurs="0"/>
      <xsd:element name="image_dimensions" type="niso:dimensions"/>
      <xsd:element name="image_metrics" type="niso:spatialmetrics" minOccurs="0"/>
      <xsd:element name="ppi" type="xsd:positiveInteger" minOccurs="0"/>
      <xsd:element name="dpi" type="xsd:positiveInteger" minOccurs="0"/>
      <xsd:element name="format" type="niso:format" minOccurs="0"/>
      <xsd:element name="scanning" type="niso:image_creation" minOccurs="0"/>
      <xsd:element name="datetimecreated" type="xsd:dateTime" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="imggroupID" type="xsd:IDREF" use="optional"/>
  </xsd:complexType>
</xsd:element>
```

7.11. Lo schema NISO

Lo schema NISO si basa sulla versione **NISO NISO Draft Standard. Data Dictionary – Technical Metadata for Digital Still Images** (nel presente manuale semplicemente **Data Dictionary**), distribuito con lo status di **Working Draft 1.0** del 5 luglio 2000²⁴. Dello standard NISO esiste anche una versione più recente: **NISOZ39.87-2002. Data Dictionary – Technical Metadata for Digital Still Images** distribuito con lo status di **Draft Standard for Trial Use**, del 31 Dicembre 2003²⁵.

Il Comitato MAG ha tuttavia ritenuto di non aggiornare per il momento lo schema `niso-mag.xsd`, poiché anche la nuova versione si presenta come **working draft** e quindi in versione non stabile. Quando il NISO Standard Comitee rilascerà una versione stabile dello standard, il Comitato MAG valuterà l'opportunità di adeguare lo schema a tale nuova versione.

Lo schema NISO è contenuto nel file `niso-mag.xsd` e contiene la definizione di dieci `simpleType` e sei `complexType`. Anche se lo standard NISO è stato concepito per la descrizione delle caratteristiche tecniche delle immagini digitali, lo schema `niso-mag.xsd` include anche specifiche tecniche relative a documenti **text-oriented** (riconoscibili grazie al prefisso `doc`).

Tipi Semplici

- **checksum** : restrizione di `xsd:string`, limita la lunghezza massima della stringa a 32 caratteri (cfr. [Data Dictionary](#) , p. 13). È usato dall'elemento `md5` delle sezioni IMG, AUDIO, VIDEO, OCR e DOC. È formalmente definito come segue:

24. reperibile al sito <http://www.niso.org/pdfs/DataDict.pdf> .

25. reperibile al sito http://www.niso.org/standards/resources/Z39_87_trial_use.pdf .

```
<xsd:simpleType name="checksum">  
  <xsd:restriction base="xsd:string">  
    <xsd:length value="32"/>  
  </xsd:restriction>  
</xsd:simpleType>
```

- **img_mimetype** : restrizione di `xsd:string`, è costituito dall'enumerazione di valori (cfr. [Data Dictionary](#) , p. 7). È usato dal tipo complesso `format` per l'elemento `<mime>`. È formalmente definito come segue:

```
<xsd:simpleType name="img_mimetype">  
  <xsd:restriction base="xsd:string">  
    <xsd:enumeration value="image/gif"/>  
    <xsd:enumeration value="image/jpeg"/>  
    <xsd:enumeration value="image/tiff"/>  
    <xsd:enumeration value="image/png"/>  
    <xsd:enumeration value="image/vnd.djvu"/>  
    <xsd:enumeration value="application/pdf"/>  
  </xsd:restriction>  
</xsd:simpleType>
```

- **compressiontype** : restrizione di `xsd:string`, è costituito dall'enumerazione di valori (cfr. [Data Dictionary](#) , p. 8). È usato dal tipo complesso `format` per l'elemento `<compression>`. È formalmente definito come segue:

```
<xsd:simpleType name="compressiontype">  
  <xsd:restriction base="xsd:string">  
    <xsd:enumeration value="Uncompressed"/>  
    <xsd:enumeration value="CCITT 1D"/>  
    <xsd:enumeration value="CCITT Group 3"/>  
    <xsd:enumeration value="CCITT Group 4"/>  
    <xsd:enumeration value="LZW"/>  
    <xsd:enumeration value="JPG"/>  
    <xsd:enumeration value="PNG"/>  
    <xsd:enumeration value="DJVU"/>  
  </xsd:restriction>  
</xsd:simpleType>
```

- **photometricinterpretationtype** : restrizione di `xsd:string`, è costituito dall'enumerazione di valori (cfr. [Data Dictionary](#) , p. 8). È usato dal tipo complesso `spatialmetrics` per l'elemento `<photometricinterpretation>`. È formalmente definito come segue:

```
<xsd:simpleType name="photometricinterpretationtype">  
  <xsd:restriction base="xsd:string">
```

```

<xsd:enumeration value="WhiteIsZero"/>
<xsd:enumeration value="BlackIsZero"/>
<xsd:enumeration value="RGB"/>
<xsd:enumeration value="Palette color"/>
<xsd:enumeration value="Transparency Mask"/>
<xsd:enumeration value="CMYK"/>
<xsd:enumeration value="YCbCr"/>
<xsd:enumeration value="CIELab"/>
</xsd:restriction>
</xsd:simpleType>

```

- **samplingfrequencyunittype** : restrizione di xsd:string, è costituito dall'enumerazione di 1, 2 e 3 ([Data Dictionary](#) , p. 24). È usato dal tipo complesso spatialmetrics per l'elemento <samplingfrequencyunit>. È formalmente definito come segue:

```

<xsd:simpleType name="samplingfrequencyunittype">
  <xsd:restriction base="xsd:integer">
    <xsd:enumeration value="1">
      <xsd:annotation>
        <xsd:documentation>no absolute unit of measurement</xsd:documentation>
      </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="2">
      <xsd:annotation>
        <xsd:documentation>inch</xsd:documentation>
      </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="3">
      <xsd:annotation>
        <xsd:documentation>centimeter</xsd:documentation>
      </xsd:annotation>
    </xsd:enumeration>
  </xsd:restriction>
</xsd:simpleType>

```

- **samplingfrequencyplanetype** : restrizione di xsd:string, è costituito dall'enumerazione di 1, 2 e 3 ([Data Dictionary](#) , p. 25). È usato dal tipo complesso spatialmetrics per l'elemento <samplingfrequencyplane>. È formalmente definito come segue:

```

<xsd:simpleType name="samplingfrequencyplanetype">
  <xsd:restriction base="xsd:integer">
    <xsd:enumeration value="1">
      <xsd:annotation>
        <xsd:documentation>camera/scanner focal plane</xsd:documentation>
      </xsd:annotation>
    </xsd:enumeration>
  </xsd:restriction>
</xsd:simpleType>

```

```
</xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="2">
  <xsd:annotation>
    <xsd:documentation>object plane</xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="3">
  <xsd:annotation>
    <xsd:documentation>source object plane</xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>
</xsd:restriction>
</xsd:simpleType>
```

- **bitpersamplotype** : restrizione di `xsd:string`, è costituito dall'enumerazione di valori (cfr. [Data Dictionary](#) , p. 27). È usato dal tipo complesso `spatialmetrics` per l'elemento `<bitpersample>`. È formalmente definito come segue:

```
<xsd:simpleType name="bitpersamplotype">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="1">
      <xsd:annotation>
        <xsd:documentation>bitonal</xsd:documentation>
      </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="4">
      <xsd:annotation>
        <xsd:documentation>4-bit grey</xsd:documentation>
      </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="8">
      <xsd:annotation>
        <xsd:documentation>8-bit grey or palette</xsd:documentation>
      </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="8,8,8">
      <xsd:annotation>
        <xsd:documentation>RGB</xsd:documentation>
      </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="16,16,16">
      <xsd:annotation>
        <xsd:documentation>TIFF, HDR</xsd:documentation>
      </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="8,8,8,8">
      <xsd:annotation>
```

```
        <xsd:documentation>CMYK</xsd:documentation>
      </xsd:annotation>
    </xsd:enumeration>
  </xsd:restriction>
</xsd:simpleType>
```

- **targettype** : restrizione di `xsd:string`, è costituito dall'enumerazione di valori (cfr. [Data Dictionary](#) , p. 32). È usato dal tipo complesso `targetdata` per l'elemento `<targetType>`. È formalmente definito come segue:

```
<xsd:simpleType name="targettype">
  <xsd:restriction base="xsd:integer">
    <xsd:enumeration value="0">
      <xsd:annotation>
        <xsd:documentation>external</xsd:documentation>
      </xsd:annotation>
    </xsd:enumeration>
    <xsd:enumeration value="1">
      <xsd:annotation>
        <xsd:documentation>internal</xsd:documentation>
      </xsd:annotation>
    </xsd:enumeration>
  </xsd:restriction>
</xsd:simpleType>
```

- **doc_mimetype** : restrizione di `xsd:string`, è costituito dall'enumerazione di valori. È usato dal tipo complesso `docFormat` per l'elemento `<mime>`. È formalmente definito come segue:

```
<xsd:simpleType name="doc_mimetype">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="text/plain"/>
    <xsd:enumeration value="text/xml"/>
    <xsd:enumeration value="text/html"/>
    <xsd:enumeration value="text/rtf"/>
    <xsd:enumeration value="application/msword"/>
    <xsd:enumeration value="application/pdf"/>
  </xsd:restriction>
</xsd:simpleType>
```

- **doc_compressiontype** : restrizione di `xsd:string`, è costituito dall'enumerazione di valori. È usato dal tipo complesso `docFormat` per l'elemento `<compression>`. È formalmente definito come segue:

```
<xsd:simpleType name="doc_compressiontype">
  <xsd:restriction base="xsd:string">
```

```
<xsd:enumeration value="Uncompressed" />
<xsd:enumeration value="ZIP" />
<xsd:enumeration value="RAR" />
<xsd:enumeration value="GZ" />
</xsd:restriction>
</xsd:simpleType>
```

Tipi Complessi

- **format** : `xsd:sequence`, contiene tre elementi che gestiscono il formato e il tipo di compressione delle immagini; cfr. [Data Dictionary](#) , pp. 7-8. È usato dall'elemento `img/format` e da `gen/img_group/format` e richiama i seguenti tipi semplici:

- `img_mimetype`
- `compressiontype`

È formalmente definito come segue:

```
<xsd:complexType name="format">
  <xsd:sequence>
    <xsd:element name="name" type="xsd:string" />
    <xsd:element name="mime" type="img_mimetype" />
    <xsd:element name="compression" type="compressiontype" minOccurs="0" />
  </xsd:sequence>
</xsd:complexType>
```

- **image_creation** : `xsd:sequence`, contiene tre elementi e tre subelementi, che nel complesso descrivono il modo in cui l'immagine digitale è stata realizzata e include marca e modello dell'apparecchiatura, oltre al software utilizzato; cfr. [Data Dictionary](#) , pp. 16-18. È usato dall'elemento `img/scanning` e da `gen/img_group/scanning`. È formalmente definito come segue:

```
<xsd:complexType name="image_creation">
  <xsd:sequence>
    <xsd:element name="sourcetype" type="xsd:string" minOccurs="0" />
    <xsd:element name="scanningagency" minOccurs="0" />
    <xsd:element name="devicesource" type="xsd:string" minOccurs="0" />
    <xsd:element name="scanningsystem" minOccurs="0">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="scanner_manufacturer" type="xsd:string" />
          <xsd:element name="scanner_model" type="xsd:string" />
          <xsd:element name="capture_software" type="xsd:string" />
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
```

```

        </xsd:complexType>
    </xsd:element>
</xsd:sequence>
</xsd:complexType>

```

- **dimensions** : `xsd:sequence`, contiene quattro elementi che nel complesso specificano le dimensioni verticali e orizzontali dell'immagine digitale espresse in pixel (`<imagelenght>` e `<imagewidth>`, obbligatori) e le dimensioni orizzontali e verticali dell'oggetto analogico digitalizzato espresse in inches (`<source_xdimension>` e `<source_ydimension>`, opzionali); cfr. [Data Dictionary](#) , p. 22. È usato dall'elemento `img/image_dimensions`. È formalmente definito come segue:

```

<xsd:complexType name="dimensions">
  <xsd:sequence>
    <xsd:element name="imagelength" type="xsd:positiveInteger"/>
    <xsd:element name="imagewidth" type="xsd:positiveInteger"/>
    <xsd:element name="source_xdimension" type="xsd:double" minOccurs="0"/>
    <xsd:element name="source_ydimension" type="xsd:double" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

```

- **spatialmetrics** : `xsd:sequence`, contiene sei elementi che specificano le principali caratteristiche tecniche delle immagini digitali. In particolare si definisce la frequenza di campionamento dell'immagine (`<samplingfrequencyunit>`, `<xsamplingfrequency>` e `<ysamplingfrequency>`), il piano di focalizzazione (`<samplingfrequencyplane>`), l'interpretazione fotometrica (`<photometricinterpretation>`) e il numero di bit per campione (`<bitpersample>`); cfr. [Data Dictionary](#) , pp. 23-26. È usato dall'elemento `img/image_metrics` e da `gen/img_group/image_metrics` e richiama i seguenti tipi semplici:

- `samplingfrequencyunittype`
- `samplingfrequencyplanetype`
- `photometricinterpretationtype`
- `bitpersampletype`

È formalmente definito come segue:

```

<xsd:complexType name="spatialmetrics">
  <xsd:sequence>
    <xsd:element name="samplingfrequencyunit" type="samplingfrequencyunittype"/>
    <xsd:element name="samplingfrequencyplane" type="samplingfrequencyplanetype"/>
    <xsd:element name="xsamplingfrequency" type="xsd:positiveInteger" minOccurs="0"/>
    <xsd:element name="ysamplingfrequency" type="xsd:positiveInteger" minOccurs="0"/>
    <xsd:element name="photometricinterpretation" type="photometricinterpretationtype"/>
    <xsd:element name="bitpersample" type="bitpersampletype"/>
  </xsd:sequence>
</xsd:complexType>

```

- **targetdata** : `xsd:sequence`, contiene cinque elementi che descrivono la tipologia e le modalità di utilizzo di un target in fase di acquisizione dell'immagine digitale; cfr. [Data Dictionary](#) , pp. 30-33. È usato dall'elemento `img/target` e richiama il tipo semplice `targettype`. È formalmente definito come segue:

```
<xsd:complexType name="targetdata">
  <xsd:sequence>
    <xsd:element name="targetType" type="targettype" minOccurs="0"/>
    <xsd:element name="targetID" type="xsd:string"/>
    <xsd:element name="imageData" type="xsd:anyURI" minOccurs="0"/>
    <xsd:element name="performanceData" type="xsd:anyURI" minOccurs="0"/>
    <xsd:element name="profiles" type="xsd:anyURI" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

- **docFormat** : `xsd:sequence`, contiene tre elementi che gestiscono il formato e il tipo di compressione dei documenti text-oriented. È usato dall'elemento `doc/format` e da `ocr/format` e richiama i seguenti tipi semplici:

- `doc_mimetype`
- `doc_compressiontype`

È formalmente definito come segue:

```
<xsd:complexType name="docFormat">
  <xsd:annotation>
    <xsd:documentation>Nonostante lo schema NISO sia nato per descrivere le
      caratteristiche tecniche di file contenenti immagini, per ragioni di
      omogenità si è scelto di usare il medesimo formato anche per
      file text-oriented</xsd:documentation>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="name" type="xsd:string"/>
    <xsd:element name="mime" type="doc_mimetype"/>
    <xsd:element name="compression" type="doc_compressiontype" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```


8. Sezione AUDIO

La sezione AUDIO raccoglie i metadati amministrativi e gestionali relativi ai file audio. Alcuni di questi dati, in realtà possono essere raccolti direttamente all'interno della sezione GEN grazie all'elemento `<audio_group>` il cui contenuto verrà tuttavia trattato in questa sezione per omogeneità tematica.

La sezione AUDIO è costituita di una sequenza di elementi `<audio>`, uno per ciascuna traccia audio descritta da MAG. L'elemento è opzionale e ripetibile. Il suo contenuto è di tipo `xsd:sequence`, e può contenere i seguenti elementi:

- `<sequence_number>` : contiene il numero di sequenza identificativo della traccia audio. Obbligatorio e non ripetibile
- `<nomenclature>` : contiene la denominazione o titolo della traccia audio. Obbligatorio e non ripetibile.
- `<proxies>` : descrizione della traccia audio digitale individuata da `<sequence_number>`. Obbligatorio e ripetibile.
- `<note>` : eventuali annotazioni. Opzionale e non ripetibile

Per l'elemento sono inoltre definiti i seguenti attributi:

- `audiogroupID` : di tipo `xsd:IDREF` contiene un riferimento all'attributo `ID` dell'elemento `<audio_group>` (p. 43). Tale attributo consente di collegare un elemento `<audio>` con le caratteristiche tecniche definite globalmente da `<audio_group>`. L'attributo è opzionale; se dell'oggetto analogico fonte sono stati tratti più formati, l'attributo va utilizzato all'interno dell'elemento `<proxies>` (p. 98) che descrive ciascuno dei formati realizzati. Qualora l'attributo non sia usato né all'interno di `<audio>` né all'interno dell'elemento `<proxies>` si assume che le caratteristiche tecniche della traccia audio digitale **non siano state altrove descritte** e quindi l'elemento `<audio_metrics>` deve ritenersi obbligatorio, così come `<format>` e `<transcription>` (tutti all'interno dell'elemento `<proxies>`).
- `holdingsID` : di tipo `xsd:IDREF` contiene un riferimento all'attributo `ID` dell'elemento `<holdings>` (p. 53) e serve a definire a quale istituzione appartiene l'oggetto analogico digitalizzato. L'attributo è opzionale.

L'elemento `<audio>` è così formalmente definito (file `metatype.xsd`):

```
<xsd:complexType name="audio">
  <xsd:sequence>
    <xsd:element name="sequence_number" type="xsd:positiveInteger"/>
    <xsd:element name="nomenclature" type="xsd:string"/>
    <xsd:element name="proxies" type="audioproxy" maxOccurs="unbounded"/>
    <xsd:element name="note" type="xsd:string" minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="holdingsID" type="xsd:IDREF" use="optional"/>
  <xsd:attribute name="audiogroupID" type="xsd:IDREF" use="optional"/>
</xsd:complexType>
```

8.1. Identificazione e denominazione delle tracce audio

Ciascuna traccia audio descritta all'interno di MAG viene identificata univocamente grazie all'elemento `<sequence_number>`. Il suo contenuto è di tipo `xsd:positiveInteger`, vale a dire che è costituito di un numero positivo. Dato il suo compito di identificatore, sul contenuto di `<sequence_number>` è stato posto un vincolo di univocità (file `metadigit.xsd`):

```
<xsd:unique name="uniquaudio">
  <xsd:selector xpath="audio"/>
  <xsd:field xpath="sequence_number"/>
</xsd:unique>
```

L'elemento è obbligatorio e non ripetibile. Il valore dell'elemento è puntato dagli attributi `sequence_number` degli elementi `<start>` e `<stop>` della sezione `<stru>` (p. 61). Per esempio:

```
<stru>
  <sequence_number>1</sequence_number>
  <nomenclature>Traccia 1</nomenclature>
  <element>
    <resource>audio</resource>
    <start sequence_number="001"/>
    <stop sequence_number="001"/>
  </element>
</stru>
<audio>
  <sequence_number>001</sequence_number>
  <nomenclature>Traccia 1</nomenclature>
</audio>
```

A ciascuna traccia audio deve inoltre essere attribuita una denominazione, per esempio Ouverture, Certe Notti, Canzone 1 ecc. Tale denominazione viene codificata dall'elemento `<nomenclature>`. L'elemento è di tipo `xsd:string`; si consiglia comunque di definire una nomenclatura controllata negli standard di progetto. L'elemento è obbligatorio e non ripetibile.

8.2. La descrizione tecnica della traccia audio

La descrizione della traccia audio è contenuta tutta all'interno dell'elemento `<proxies>`. L'elemento è obbligatorio e ripetibile in quanto dello stesso oggetto analogico (disco, CD, ecc) è possibile trarre più versioni da usare in ambiti diversi. È di tipo `xsd:sequence` e può contenere:

- **<usage>** : definisce l'ambito d'uso della traccia audio in relazione agli standard di progetto. Opzionale e ripetibile
- **<file>** : localizza il file contenente la traccia audio. Obbligatorio e non ripetibile

- `<md5>` : contiene l'impronta del file. Obbligatorio e non ripetibile
- `<filesize>` : fornisce la dimensione del file contenente la traccia audio in byte. Opzionale e non ripetibile
- `<audio_dimensions>` : definisce le dimensioni della traccia audio e non ripetibile
- `<audio_metrics>` : fornisce le principali caratteristiche tecniche della traccia audio. Opzionale e non ripetibile
- `<format>` : dichiara il formato della traccia audio. Opzionale e non ripetibile
- `<transcription>` : descrive il tipo di trascrizione della traccia audio. Opzionale e non ripetibile
- `<datetimecreated>` : dichiara la data e l'ora della creazione della traccia audio digitale. Opzionale e non ripetibile

Per l'elemento è definito un solo attributo:

- **audiogroupID** : di tipo `xsd:IDREF` contiene un riferimento all'attributo `ID` dell'elemento `<audio_group>` (p. 43). Tale attributo consente di collegare un `<proxies>` con le caratteristiche tecniche definite globalmente da `<audio_group>`. L'attributo è opzionale e si usa quando dallo stesso oggetto analogico sono stati tratti più formati, descritti ciascuno da un proprio elemento `<proxies>`. Qualora dall'oggetto analogico sia stato tratto un solo oggetto digitale, l'attributo dovrebbe essere usato all'interno dell'elemento `<audio>`. L'uso è ovviamente subordinato all'impiego dell'elemento `<audio_group>` all'interno della sezione GEN. Qualora l'attributo non sia usato né all'interno di `<audio>` né all'interno dell'elemento `<proxies>` si assume che le caratteristiche tecniche della traccia audio digitale non siano state altrove descritte e quindi l'elemento `<audio_metrics>` deve ritenersi obbligatorio, così come `<format>` e `<transcription>`. Si danno quindi tre possibilità:
 - **l'attributo audiogroupID è usato all'interno di <audio>** : le caratteristiche tecniche della traccia audio sono definite all'interno dell'elemento `<audio_group>` della sezione GEN. Esiste un solo elemento `<proxies>` all'interno di `<audio>` in quanto dell'oggetto analogico è stata tratta un'unica versione
 - **l'attributo audiogroupID è usato all'interno di <proxies>** : le caratteristiche tecniche della traccia audio sono definite all'interno dell'elemento `<audio_group>` della sezione GEN. Esistono più elementi `<proxies>` all'interno di `<audio>`, ognuno dei quali descrive un formato alternativo del medesimo oggetto analogico. In questo caso si assume che esistano più `<img_group>`
 - **l'attributo audiogroupID non è usato né all'interno di <proxies> né all'interno di <audio>** : le caratteristiche tecniche della traccia audio digitale non sono state altrove descritte e quindi gli elementi `<audio_metrics>`, `<format>` e `<transcription>` figli di `<proxies>`, formalmente opzionali, debbono ritenersi obbligatori.

Per esempio

```
<gen>  
<!-- omissis -->
```

```
<audio_group ID="aGroupIntra">
  <!-- omissis -->
</audio_group>
<audio_group ID="aGroupInter">
  <!-- omissis -->
</audio_group>
</gen>
<!-- omissis -->
<audio>
  <sequence_number>1</sequence_number>
  <nomenclature>Di sera, dove andare?</nomenclature>
  <proxies audiogroupID="aGroupIntra">
    <!-- omissis -->
  </proxies>
  <proxies audiogroupID="aGroupInter">
    <!-- omissis -->
  </proxies>
</audio>
```

Il tipo complesso `audioproxies` che descrive l'elemento `<proxies>` è così formalmente definito (file `metatype.xsd`):

```
<xsd:complexType name="audioproxy">
  <xsd:sequence>
    <xsd:element name="usage" type="usages" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="file" type="link"/>
    <xsd:element name="md5" type="niso:checksum"/>
    <xsd:element name="filesize" type="xsd:unsignedLong" minOccurs="0"/>
    <xsd:element name="audio_dimensions" type="audio_dimensions"/>
    <xsd:element name="audio_metrics" type="audio_spatialmetrics" minOccurs="0"/>
    <xsd:element name="format" type="audio_format" minOccurs="0"/>
    <xsd:element name="transcription" type="audio_creation" minOccurs="0"/>
    <xsd:element name="datetimecreated" type="xsd:dateTime" minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="audiogroupID" type="xsd:IDREF" use="optional"/>
</xsd:complexType>
```

8.2.1. Uso, localizzazione, integrità e grandezza del file

Dello stesso oggetto digitale (tipicamente disco in vinile, un nastro, un CD) possono essere tratte più tracce audio digitali, di qualità più o meno elevata, in diversi formati, ognuna delle quali con una diversa finalità. È infatti usuale creare tracce audio di alta qualità per l'archiviazione interna e tracce di qualità più limitata per la diffusione esterna. La finalità della traccia digitale viene registrata dall'elemento `<usage>`. L'elemento è di tipo `xsd:string`; al fine di favorire la portabilità dei dati, si consiglia tuttavia di adottare le seguenti due

tassonomie (adottate dai maggiori progetti di digitalizzazione italiani), la prima relativa alle modalità d'uso, la seconda al possesso del copyright da parte dell'istituzione:

- 1 : master
- 2 : alta qualità
- 3 : bassa qualità
- 4 : preview

e

- a : il repository non ha il copyright dell'oggetto digitale
- b : il repository ha il copyright dell'oggetto digitale

L'elemento è opzionale e ripetibile.

L'elemento `<file>` consente di localizzare il file che contiene la traccia audio digitale. È di tipo `link`, vale a dire che è un elemento vuoto che supporta attributi definiti dal namespace `xlink` (p. 33). L'elemento è obbligatorio e non ripetibile.

L'integrità del contenuto digitale è verificata grazie alla sua impronta digitale, registrata dall'elemento `<md5>`, un codice standard di 32 caratteri che viene rilevato automaticamente grazie all'impiego di appositi applicativi. Le regole per il rilevamento dell'impronta devono essere definite localmente, così come i momenti per il rilievo stesso (prima del momento del deposito, al momento del deposito, o in entrambi i momenti). Si tratta di una raccomandazione NISO e come tale il tipo specializzato che governa il contenuto dell'elemento appartiene al namespace `niso` ed è denominato `niso:checksum` (p. 89). Tale tipo è definito come restrizione di `xsd:string` che limita la lunghezza massima della stringa a 32 caratteri.

La grandezza del file (che va espressa in byte) è registrata dall'elemento `<filesize>`. L'elemento è di tipo `xsd:unsignedLong` (un numero non negativo il cui valore massimo è 18446744073709551615; si veda la definizione fornita dal W3C al sito <http://www.w3.org/TR/2004/REC-xmlschema-2-20041028/datatypes.html>), è opzionale e non ripetibile. Anche l'elemento `<filesize>` è una raccomandazione NISO (Cfr. [Data Dictionary](#), p. 13).

```
<usage>2</usage>
<file Location="URL" xlink:href="14290/6385_A_audio.0.equalized.256.mp3"/>
<md5>54081682d349e7037e0eab9757afe3e3</md5>
<filesize>6304512</filesize>
```

8.2.2. Le dimensioni della traccia audio

Le dimensioni della traccia audio sono codificate dall'elemento obbligatorio e non ripetibile `<audio_dimensions>`; poiché la dimensione di una traccia audio è data esclusivamente dalla sua durata, l'elemento `<audio_dimensions>` contiene un solo elemento `<duration>` (anch'esso obbligatorio e non ripetibile) di tipo `xsd:time`.

Il tipo complesso `audio_dimensions` che regola il contenuto dell'elemento `<audio_dimensions>` è formalmente definito come segue (file `audio-mag-xsd:`)

```
<xsd:complexType name="audio_dimensions">
  <xsd:sequence>
    <xsd:element name="duration" type="xsd:time"/>
  </xsd:sequence>
</xsd:complexType>
```

8.2.3. Principali caratteristiche tecniche della traccia audio

Le principali caratteristiche tecniche della traccia audio sono contenute nell'elemento `<audio_metrics>`. L'elemento, oltre che dentro `<proxies>`, può essere usato dentro `<audio_group>` (p. 43); è formalmente opzionale, ma in pratica deve sempre essere usato, o dentro `<proxies>` o dentro `<audio_group>`. Per `<audio_metrics>` è definito un tipo specializzato denominato `audio_spatialmetrics`. Tale tipo è di tipo `xsd:sequence` e contiene:

- **<samplingfrequency>** : descrive la frequenza di campionamento della traccia audio digitale espressa in KHz; l'elemento è obbligatorio e non ripetibile. Per l'elemento è definito un tipo semplice specializzato denominato `samplingfrequencytype`; tale tipo è una restrizione del tipo `xsd:float` (contenuto variabile, si veda <http://www.w3.org/TR/2004/REC-xmlschema-2-20041028/datatypes.html>) e ammette i seguenti valori (separati da |): 8 | 11.025 | 12 | 16 | 22.05 | 24 | 32 | 44.1 | 48 | 96. Il tipo `samplingfrequencytype` è formalmente definito come segue (file `audio-mag.xsd`):

```
<xsd:simpleType name="samplingfrequencytype">
  <xsd:annotation>
    <xsd:documentation>espressa in KHz</xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:float">
    <xsd:enumeration value="8"/>
    <xsd:enumeration value="11.025"/>
    <xsd:enumeration value="12"/>
    <xsd:enumeration value="16"/>
    <xsd:enumeration value="22.05"/>
    <xsd:enumeration value="24"/>
    <xsd:enumeration value="32"/>
    <xsd:enumeration value="44.1"/>
    <xsd:enumeration value="48"/>
    <xsd:enumeration value="96"/>
  </xsd:restriction>
</xsd:simpleType>
```

- **<bitpersample>** : indica il numero di bit di cui è composto il singolo campione; l'elemento è alternativo a `<bitrate>`, uno dei due elementi deve essere necessariamente presente; non ripetibile. Per l'elemento è definito un tipo semplice specializzato denominato `bitpersamplotype`, restrizione del tipo `xsd:integer` che contiene i

seguenti valori (separati da |): 8 | 16 | 24. Il tipo `bitpersampletype` è formalmente definito come segue (file `audio-mag.xsd`):

```
<xsd:simpleType name="bitpersampletype">
  <xsd:restriction base="xsd:integer">
    <xsd:enumeration value="8"/>
    <xsd:enumeration value="16"/>
    <xsd:enumeration value="24"/>
  </xsd:restriction>
</xsd:simpleType>
```

- **<bitrate>** : esprime i kbit per secondo (kbps) del campione considerato; l'elemento è alternativo a `<bitpersample>`, uno dei due elementi deve essere necessariamente presente; non ripetibile. Per l'elemento è definito un tipo semplice specializzato denominato `bitratetype`, restrizione del tipo `xsd:integer` che contiene i seguenti valori (separati da |): 24 | 32 | 48 | 56 | 64 | 96 | 128 | 160 | 192 | 256 | 320 | 384. Il tipo `bitratetype` è formalmente definito come segue (file `audio-mag.xsd`):

```
<xsd:simpleType name="bitratetype">
  <xsd:annotation>
    <xsd:documentation>espressa in Kbps</xsd:documentation>
  </xsd:annotation>
  <xsd:restriction base="xsd:integer">
    <xsd:enumeration value="24"/>
    <xsd:enumeration value="32"/>
    <xsd:enumeration value="48"/>
    <xsd:enumeration value="56"/>
    <xsd:enumeration value="64"/>
    <xsd:enumeration value="96"/>
    <xsd:enumeration value="128"/>
    <xsd:enumeration value="160"/>
    <xsd:enumeration value="192"/>
    <xsd:enumeration value="256"/>
    <xsd:enumeration value="320"/>
    <xsd:enumeration value="384"/>
  </xsd:restriction>
</xsd:simpleType>
```

Per esempio:

```
<audio_metrics>
  <samplingfrequency>48</samplingfrequency>
  <bitpersample>24</bitpersample>
</audio_metrics>
```

Il tipo complesso `audio_spatialmetrics` che regola il contenuto dell'elemento `<audio_metrics>` è così formalmente definito (file `audio-mag.xsd`):

```
<xsd:complexType name="audio_spatialmetrics">
  <xsd:sequence>
    <xsd:element name="samplingfrequency" type="samplingfrequencytype"/>
    <xsd:choice>
      <xsd:element name="bitpersample" type="bitpersampletype"/>
      <xsd:element name="bitrate" type="bitratetype"/>
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>
```

8.2.4. Il formato delle tracce audio

Il formato delle tracce audio (tipologia e modalità di compressione) è gestito dall'elemento `<format>`. L'elemento, oltre che dentro `<proxies>`, può essere usato dentro `<audio_group>` (p. 43); è formalmente opzionale, ma in pratica deve sempre essere usato, o dentro `<proxies>` o dentro `<audio_group>`. Per `<format>` è definito un tipo specializzato denominato `audio_format`. Tale tipo è di tipo `xsd:sequence` e contiene quattro elementi:

- **<name>** : obbligatorio e non ripetibile, contiene il formato della traccia audio. È di tipo `xsd:string`, si consiglia di usare valori come WAV, MP3, ecc.
- **<mime>** : obbligatorio e non ripetibile, contiene il tipo mime della traccia audio. Per l'elemento è definito un tipo semplice specializzato, anch'esso contenuto nel file `audio-mag.xsd`, denominato `audio_mimetype`. Tale tipo è definito come restrizione di `xsd:string` ed è costituito dall'enumerazione dei seguenti valori:
 - audio/wav
 - audio/mpeg
 - audio/mpg
 - audio/mp3
 - audio/x-mpeg
 - audio/midi
 - audio/x-realaudio

Il tipo `audio_mimetype` è formalmente definito come segue (file `audio-mag.xsd`):

```
<xsd:simpleType name="audio_mimetype">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="audio/wav"/>
    <xsd:enumeration value="audio/mpeg"/>
    <xsd:enumeration value="audio/mpg"/>
    <xsd:enumeration value="audio/mp3"/>
  </xsd:restriction>
</xsd:simpleType>
```



```
<xsd:enumeration value="audio/x-mpeg"/>
<xsd:enumeration value="audio/midi"/>
<xsd:enumeration value="audio/x-realaudio"/>
</xsd:restriction>
</xsd:simpleType>
```

- **<compression>** : obbligatorio e non ripetibile, dichiara il tipo di compressione applicato al file audio. Per l'elemento è definito un tipo semplice specializzato, anch'esso contenuto nel file audio-mag.xsd, denominato `compressiontype`. Tale tipo è definito come restrizione di `xsd:string` ed è costituito dall'enumerazione dei seguenti valori:

- Uncompressed
- Linear PCM
- MPEG-1 layer 1
- MPEG-1 layer 2
- MPEG-1 layer 3
- AC3
- Dolby
- DTC

Il tipo `compressiontype` è formalmente definito come segue (file audio-mag.xsd):

```
<xsd:simpleType name="compressiontype">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="Uncompressed"/>
    <xsd:enumeration value="Linear PCM"/>
    <xsd:enumeration value="MPEG-1 layer 1"/>
    <xsd:enumeration value="MPEG-1 layer 2"/>
    <xsd:enumeration value="MPEG-1 layer 3"/>
    <xsd:enumeration value="AC3"/>
    <xsd:enumeration value="Dolby"/>
    <xsd:enumeration value="DTS"/>
  </xsd:restriction>
</xsd:simpleType>
```

- **<channel_configuration>** : obbligatorio e non ripetibile, definisce lo schema di configurazione dei canali audio. Per l'elemento è definito un tipo semplice specializzato, anch'esso contenuto nel file audio-mag.xsd, denominato `channelsconfigurationtype`. Tale tipo è definito come restrizione di `xsd:string` ed è costituito dall'enumerazione dei seguenti valori:

- Mono
- Dual mono
- Joint stereo
- Stereo
- 2 ch

- 4 ch
- 5.1 ch
- 6.1 ch

Il tipo `channelsconfigurationtype` è formalmente definito come segue (file `audio-mag.xsd`):

```
<xsd:simpleType name="channelsconfigurationtype">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="Mono"/>
    <xsd:enumeration value="Dual mono"/>
    <xsd:enumeration value="Joint stereo"/>
    <xsd:enumeration value="Stereo"/>
    <xsd:enumeration value="2 ch"/>
    <xsd:enumeration value="4 ch"/>
    <xsd:enumeration value="5.1 ch"/>
    <xsd:enumeration value="6.1 ch"/>
  </xsd:restriction>
</xsd:simpleType>
```

Per esempio:

```
<format>
  <name>mp3</name>
  <mime>audio/mp3</mime>
  <compression>MPEG-1 layer 3</compression>
  <channel_configuration>Joint stereo</channel_configuration>
</format>
```

Il tipo complesso `audio_format` che regola il contenuto dell'elemento `<format>` è così formalmente definito (file `audio-mag.xsd`):

```
<xsd:complexType name="audio_format">
  <xsd:sequence>
    <xsd:element name="name" type="xsd:string"/>
    <xsd:element name="mime" type="audio_mimetype"/>
    <xsd:element name="compression" type="compressiontype" minOccurs="0"/>
    <xsd:element name="channel_configuration" type="channelsconfigurationtype"
minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

8.2.5. La trascrizione dell'oggetto

Le modalità della trascrizione della traccia audio digitale sono descritte dall'elemento `<transcription>`. Oltre che dentro `<proxies>` può essere usato all'interno di `<audio_group>` (p. 43); per tale ragione è formalmente opzionale, ma in realtà deve essere usato o all'interno di `<audio_group>` o dentro `<proxies>`. L'elemento non è ripetibile ed è di tipo `xsd:sequence`; può contenere i seguenti elementi:

- **`<sourcetype>`** : obbligatorio e non ripetibile, descrive il tipo di fonte analogica utilizzata per l'acquisizione della traccia audio digitale; è di tipo `xsd:string` ma si consiglia tuttavia di adottare la seguente tassonomia desunta dal *Catalogo Gestionale della Discoteca di Stato*:
 - Registrazione sonora
 - Disco
 - Nastro (bobina aperta)
 - Nastro (cassetta)
 - Nastro (cartuccia)
 - Registrazione su filo
 - Cilindro
 - Rullo di tamburo
 - Film (sonoro)
 - Compact Disc
 - DAT
 - Disco (33 gg.)
 - Disco (45 gg.)
 - Disco (78 gg.)
 - Altro
- **`<transcriptionagency>`** : opzionale, contiene il nome dell'istituzione che ha effettuato la digitalizzazione della traccia audio; è utile nel caso la digitalizzazione non sia stata effettuata direttamente dall'istituzione responsabile del processo di digitalizzazione registrato elemento `<agency>` (p. 38). L'elemento non è ripetibile ed è di tipo `xsd:string`.
- **`<transcriptiondate>`** : opzionale, contiene la data in cui è stata effettuata la digitalizzazione della traccia audio; è utile nel caso in cui la data della digitalizzazione non coincida con quella della creazione del record MAG. Non ripetibile, è di tipo `xsd:dateTime`.
- **`<devicesource>`** : opzionale ma raccomandato, descrive la tipologia dell'apparecchiatura usata per realizzare la digitalizzazione. Non ripetibile, è di tipo `xsd:string`, ma per ragioni di omogeneità si consiglia comunque di adottare, ove possibile, la seguente nomenclatura: `microfono` | `recorder` | `giradischi`.
- **`<transcriptionchain>`** : opzionale e ripetibile, descrive gli strumenti usati nel processo di digitalizzazione. Ripetibile, è di tipo `xsd:sequence` e contiene cinque elementi
 - `<device_description>`
 - `<device_manufacturer>`
 - `<device_model>`
 - `<capture_software>`

- <device_setting>

Per i dettagli si veda il par. *Gli strumenti di digitalizzazione* (p. 109)

- <transcriptionsummary> : opzionale e ripetibile, raccoglie i dati misurati durante il processo di digitalizzazione **per i valori di sintesi**. Permette di definire nomi, tipi e valori delle grandezze fisiche misurate, consentendone una nidificazione gerarchica (ad esempio valori per canale). Supporta due formati alternativi: 1. con gli elementi <grouping> e <transcriptionsummary> si possono definire dei raggruppamenti di tipologie di dati; 2. Con gli elementi <data_description>, <data_unit> e <data_value> si definiscono le misure considerate. Per i dettagli si veda il par. *Dati tecnici della trascrizione* (p. 110).
- <transcriptiondata> : opzionale e ripetibile, raccoglie le sequenze di dati misurate durante il processo di digitalizzazione. Permette di definire nomi, tipi e valori delle grandezze fisiche misurate, consentendone una nidificazione gerarchica (ad esempio valori per canale). Supporta due formati alternativi: 1. con gli elementi <grouping> e <transcriptiondata> si possono definire dei raggruppamenti di tipologie di dati; 2. Con gli elementi <data_description>, <data_unit>, <interval> e <data_value> si definiscono le misure considerate. *Dati tecnici della trascrizione* (p.110).

Si veda il seguente esempio:

```
<transcription>
  <sourcetype>disco (78 gg.)</sourcetype>
  <transcriptionagency>Discoteca di Stato - Museo dell'Audiovisivo</transcriptionagency>
  <transcriptiondate>2005-12-28T19:22:48</transcriptiondate>
  <devicesource>giradischi Technics doppio braccio, testina Stanton</devicesource>
  <transcriptionchain>
    <!-- omissis -->
  </transcriptionchain>
</transcription>
```

Il tipo complesso `audio_creation` che regola il contenuto dell'elemento `<transcription>` è formalmente definito come segue (file `audio-mag.xsd`):

```
<xsd:complexType name="audio_creation">
  <xsd:sequence>
    <xsd:element name="sourcetype" type="xsd:string"/>
    <xsd:element name="transcriptionagency" minOccurs="0"/>
    <xsd:element name="transcriptiondate" type="xsd:dateTime" minOccurs="0"/>
    <xsd:element name="devicesource" type="xsd:string" minOccurs="0"/>
    <xsd:element name="transcriptionchain" minOccurs="0" maxOccurs="unbounded">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="device_description">
            <xsd:complexType>
              <xsd:simpleContent>
```

```
<xsd:extension base="xsd:string">
  <xsd:attribute name="Type" type="xsd:string" use="required"/>
  <xsd:attribute name="Unique_identifier" type="xsd:string" use="optional"/>
  <xsd:attribute name="Comments" type="xsd:string" use="optional"/>
</xsd:extension>
</xsd:simpleContent>
</xsd:complexType>
</xsd:element>
<xsd:element name="device_manufacturer" type="xsd:string"/>
<xsd:element name="device_model">
  <xsd:complexType>
    <xsd:simpleContent>
      <xsd:extension base="xsd:string">
        <xsd:attribute name="Model" type="xsd:string" use="required"/>
        <xsd:attribute name="Serial_Number" type="xsd:string" use="optional"/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>
</xsd:element>
<xsd:element name="capture_software" type="xsd:string" minOccurs="0"/>
<xsd:element name="device_settings" type="xsd:string" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="transcriptionsummary" type="audio_transcriptionsummarytype"
  minOccurs="0" maxOccurs="unbounded"/>
<xsd:element name="transcriptiondata" type="audio_transcriptiondatatype"
  minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
```

8.2.5.1. Gli strumenti di digitalizzazione

L'elemento `<transcriptionchain>` consente di descrivere accuratamente gli strumenti hardware e software utilizzati dal processo di digitalizzazione. L'elemento contiene cinque elementi:

- **<device_description>** : obbligatorio e non ripetibile, contiene le generalità sul tipo di dispositivo audio utilizzato per la digitalizzazione: Per l'elemento sono definiti tre attributi:
 - **Type** : obbligatorio e di tipo `xsd:string` descrive il tipo di dispositivo
 - **Unique_identifier** : opzionale e di tipo `xsd:string`, contiene un codice identificativo del dispositivo
 - **Comments** : opzionale e di tipo `xsd:string`, contiene ogni eventuale commento circa il dispositivo
- **<device_manufacturer>** : obbligatorio e non ripetibile, contiene il nome del produttore

- del dispositivo usato per la trascrizione; è di tipo `xsd:string`
- `<device_model>` : obbligatorio e non ripetibile, contiene il nome del modello del dispositivo usato per la trascrizione. Per l'elemento sono definiti due attributi:
 - `Model` : obbligatorio e di tipo `xsd:string` contiene il nome del modello del dispositivo
 - `Serial_Number` : opzionale e di tipo `xsd:string`, contiene il numero seriale del dispositivo
- `<capture_software>` : opzionale e non ripetibile, contiene il nome del software utilizzato per l'acquisizione della traccia digitale; è di tipo `xsd:string`.
- `<device_setting>` : opzionale e non ripetibile, descrive le impostazioni usate dal dispositivo di acquisizione.

Si veda il seguente esempio:

```
<transcriptionchain>
  <device_description Type="convertitore A/D 24/48" Unique_identifier="AD-16X"
    Comments="dispositivo acquistato nel 2004" />
  <device_manufacturer>Apogee</device_manufacturer>
  <device_model Model="Rosetta" Serial_Number="AD-16X"/>
  <capture_software>Analogue Audio Ingestion</capture_software>
  <device_settings>48Khz, double arms</device_settings>
</transcriptionchain>
```

8.2.5.2. Dati tecnici della trascrizione: valori di sintesi e valori globali

Gli elementi `<transcriptionsummary>` e `<transcriptiondata>` raccolgono i dati tecnici relativi alla trascrizione, il primo raccoglie i dati di sintesi, il secondo i dati assoluti del file digitale. Entrambi gli elementi possono essere usati per definire nomi, tipi e valori delle grandezze fisiche misurate, consentendone una nidificazione gerarchica. Entrambi gli elementi supportano due formati alternativi (`xsd:choice`), uno per definire raggruppamenti di tipologie di dati, l'altro per raccogliere effettivamente i dati.

Vediamo ora il dettaglio dei due elementi.

- `<transcriptionsummary>` : raccoglie i dati misurati durante il processo di digitalizzazione **per i valori di sintesi**. Supporta due formati alternativi:
 - con gli elementi `<grouping>` (obbligatorio) e `<transcriptionsummary>` (opzionale) si possono definire dei raggruppamenti di tipologie di dati. L'elemento `<grouping>` di tipo `xsd:string` contiene l'etichetta della tipologia dei dati considerati.
 - le misure considerate si definiscono grazie agli elementi `<data_description>` (di tipo `xsd:string` e obbligatorio) che contiene la descrizione del tipo di dati, `<data_unit>` (opzionale di tipo `xsd:string`) che contiene l'unità di misura per i dati considerati, e `<data_value>` (obbligatorio, di tipo `xsd:float`) che contiene il valore dei dati considerati.

Ma si veda un esempio di utilizzo dell'elemento:

```
<transcriptionsummary>
  <grouping>Distorsione analogica</grouping>
  <transcriptionsummary>
    <data_description>intermodularizzazione</data_description>
    <data_unit>percentuale</data_unit>
    <data_value>0.1</data_value>
  </transcriptionsummary>
</transcriptionsummary>
```

Si veda anche il seguente esempio:

```
<transcriptionsummary>
  <grouping>dinamica</grouping>
  <transcriptionsummary>
    <data_description>peak</data_description>
    <data_unit>dB</data_unit>
    <data_value>-9</data_value>
  </transcriptionsummary>
  <transcriptionsummary>
    <data_description>dynamic</data_description>
    <data_unit>dB</data_unit>
    <data_value>50</data_value>
  </transcriptionsummary>
  <transcriptionsummary>
    <data_description>SNR</data_description>
    <data_unit>dB</data_unit>
    <data_value>24.5</data_value>
  </transcriptionsummary>
</transcriptionsummary>

<transcriptionsummary>
  <data_description>clicks</data_description>
  <data_unit>Samples per million</data_unit>
  <data_value>0.42</data_value>
</transcriptionsummary>
```

Il tipo complesso `audio_transcriptionssummarytype` che regola il contenuto dell'elemento `<transcriptionsummary>` è formalmente definito come segue (file `audio-mag.xsd`):

```
<xsd:complexType name="audio_transcriptionssummarytype">
  <xsd:choice>
    <xsd:sequence>
```

```
<xsd:element name="grouping" type="xsd:string"/>
<xsd:element name="transcriptionsummary"
  type="audio_transcriptionsummarytype" minOccurs="0"
  maxOccurs="unbounded"/>
</xsd:sequence>
<xsd:sequence>
  <xsd:element name="data_description" type="xsd:string"/>
  <xsd:element name="data_unit" type="xsd:string" minOccurs="0"/>
  <xsd:element name="data_value" type="xsd:float"/>
</xsd:sequence>
</xsd:choice>
</xsd:complexType>
```

- **<transcriptiondata>** : raccoglie i dati misurati durante il processo di digitalizzazione. Supporta due formati alternativi:
 - con gli elementi <grouping> e <transcriptiondata> si possono definire dei raggruppamenti di tipologie di dati. L'elemento <grouping> contiene l'etichetta della tipologia dei dati considerati.
 - le misure considerate si definiscono grazie agli elementi <data_description> (di tipo `xsd:string` e obbligatorio) che contiene la descrizione del tipo di dati, <data_unit> (opzionale di tipo `xsd:string`) che contiene l'unità di misura per i dati considerati, <interval> che definisce l'intervallo di tempo entro il quale i valori sono misurati, e <data_value> (obbligatorio, di tipo `xsd:float`) che contiene il valore dei dati considerati. L'elemento <interval> supporta due attributi obbligatori di tipo `xsd:time`, `start` e `stop` che definiscono rispettivamente l'inizio e la fine dell'intervallo di misurazione dei dati considerati.

Ma si veda un esempio di utilizzo dell'elemento:

```
<transcriptiondata>
  <grouping>Distorsione del dominio digitale</grouping>
  <transcriptiondata>
    <data_description>tempo di saturazione</data_description>
    <data_unit>percentuale</data_unit>
    <interval start="00:00:01" stop="00:00:03"/>
    <data_value>0.3</data_value>
  </transcriptiondata>
</transcriptiondata>
```

Si veda anche il seguente esempio:

```
<transcriptiondata>
  <grouping>segmentazione</grouping>
  <transcriptiondata>
```



```
<data_description>silence</data_description>
<data_unit>millisecondi</data_unit>
<interval start="00:00:00.000" stop="00:00:03.000" />
<data_value>3000</data_value>
</transcriptiondata>
<transcriptiondata>
  <data_description>segnale</data_description>
  <data_unit>millisecondi</data_unit>
  <interval start="00:00:03.000" stop="00:03:17.000" />
  <data_value>197000</data_value>
</transcriptiondata>
</transcriptiondata>
```

Il tipo complesso `audio_transcriptiondatatype` che regola il contenuto dell'elemento `<transcriptiondata>` è formalmente definito come segue (file `audio-mag.xsd`):

```
<xsd:complexType name="audio_transcriptiondatatype">
  <xsd:choice>
    <xsd:sequence>
      <xsd:element name="grouping" type="xsd:string" />
      <xsd:element name="transcriptiondata"
        type="audio_transcriptiondatatype" minOccurs="0"
        maxOccurs="unbounded" />
    </xsd:sequence>
    <xsd:sequence>
      <xsd:element name="data_description" type="xsd:string" />
      <xsd:element name="data_unit" type="xsd:string" minOccurs="0" />
      <xsd:element name="interval" minOccurs="0">
        <xsd:complexType>
          <xsd:attribute name="start" type="xsd:time"
            use="required" />
          <xsd:attribute name="stop" type="xsd:time"
            use="required" />
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="data_value" type="xsd:float"
        maxOccurs="unbounded" />
    </xsd:sequence>
  </xsd:choice>
</xsd:complexType>
```

8.2.6. La creazione del file

Il momento (data e ora) della creazione del file digitale (che può non coincidere con la data e il momento della trascrizione digitale, per esempio nel caso di correzioni apportate al

file) viene registrata dell'elemento `<datetimecreated>`. L'elemento è di tipo `xsd:dateTime`, è opzionale e non ripetibile.

```
<datetimecreated>2005-12-29T10:42:16</datetimecreated>
```

9. Sezione VIDEO

La sezione VIDEO raccoglie i metadati amministrativi e gestionali relativi ai file video. Alcuni di questi dati, in realtà, possono essere raccolti direttamente all'interno della sezione GEN grazie all'elemento `<video_group>` il cui contenuto verrà tuttavia trattato in questa sezione per omogeneità tematica.

La sezione VIDEO è costituita di una sequenza di elementi `<video>`, uno per ciascuno stream video descritto da MAG. L'elemento è opzionale e ripetibile. Il suo contenuto è di tipo `xsd:sequence`, e può contenere i seguenti elementi:

- `<sequence_number>` : contiene il numero di sequenza identificativo dello stream video. Obbligatorio e non ripetibile
- `<nomenclature>` : contiene la denominazione o titolo dello stream video. Obbligatorio e non ripetibile.
- `<proxies>` : descrizione dello stream video digitale individuato da `<sequence_number>`. Obbligatorio e ripetibile.
- `<note>` : eventuali annotazioni. Opzionale e non ripetibile.

Per l'elemento sono inoltre definiti i seguenti attributi:

- `videogroupID` : di tipo `xsd:IDREF` contiene un riferimento all'attributo `ID` dell'elemento `<video_group>` (p. 45). Tale attributo consente di collegare un `<video>` con le caratteristiche tecniche definite globalmente da `<video_group>`. L'attributo è opzionale; se dell'oggetto analogico fonte sono stati tratti più formati, l'attributo va utilizzato all'interno dell'elemento `<proxies>` (p. 116) che descrive ciascuno dei formati realizzati. Qualora l'attributo non sia usato né all'interno di `<video>` né all'interno dell'elemento `<proxies>` si assume che le caratteristiche tecniche dello stream video digitale **non siano state altrove descritte** e quindi l'elemento `<video_metrics>` deve ritenersi obbligatorio, così come `<format>` e `<digitisation>` (tutti all'interno dell'elemento `<proxies>`).
- `holdingsID` : di tipo `xsd:IDREF` contiene un riferimento all'attributo `ID` dell'elemento `<holdings>` (p. 53) e serve a definire a quale istituzione appartiene l'oggetto analogico digitalizzato. L'attributo è opzionale.

L'elemento `<video>` è così formalmente definito (file `metatype.xsd`):

```
<xsd:complexType name="video">
  <xsd:sequence>
    <xsd:element name="sequence_number" type="xsd:positiveInteger"/>
    <xsd:element name="nomenclature" type="xsd:string"/>
    <xsd:element name="proxies" type="videoproxy" maxOccurs="unbounded"/>
    <xsd:element name="note" type="xsd:string" minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="holdingsID" type="xsd:IDREF" use="optional"/>
  <xsd:attribute name="videogroupID" type="xsd:IDREF" use="optional"/>
</xsd:complexType>
```

9.1. Identificazione e denominazione degli stream video

Ciascuno stream video descritto all'interno di MAG viene identificato univocamente grazie all'elemento `<sequence_number>`. Il suo contenuto è di tipo `xsd:positiveInteger`, vale a dire che è costituito di un numero positivo. Dato il suo compito di identificatore, sul contenuto di `<sequence_number>` è stato posto un vincolo di univocità (file `metadigit.xsd`):

```
<xsd:unique name="uniquvideo">
  <xsd:selector xpath="video"/>
  <xsd:field xpath="sequence_number"/>
</xsd:unique>
```

L'elemento è obbligatorio e non ripetibile. Il valore dell'elemento è puntato dagli attributi `sequence_number` degli elementi `<start>` e `<stop>` della sezione `<stru>` (p. 61).

```
<stru>
  <sequence_number>1</sequence_number>
  <nomenclature>Stream 1</nomenclature>
  <element>
    <resource>video</resource>
    <start sequence_number="001"/>
    <stop sequence_number="001"/>
  </element>
</stru>
<video>
  <sequence_number>001</sequence_number>
  <nomenclature>Stream 1</nomenclature>
</video>
```

A ciascuno stream video deve inoltre essere attribuita una denominazione, per esempio *Canzonissima '52, Fiorella Mannoia live, ecc.* Tale denominazione viene codificata dall'elemento `<nomenclature>`. L'elemento è di tipo `xsd:string`; si consiglia comunque di definire una nomenclatura controllata negli standard di progetto. L'elemento è obbligatorio e non ripetibile.

9.2. La descrizione tecnica della traccia video

La descrizione dello stream video è contenuta tutta all'interno dell'elemento `<proxies>`. L'elemento è obbligatorio e ripetibile in quanto dello stesso oggetto analogico (VHS, DVD, ecc) è possibile trarre più versioni da usare in ambiti diversi. È di tipo `xsd:sequence` e può contenere:

- **<usage>** : definisce l'ambito d'uso dello stream video in relazione agli standard di progetto. Opzionale e ripetibile
- **<file>** : localizza il file contenente lo stream video. Obbligatorio e non ripetibile
- **<md5>** : contiene l'impronta del file. Obbligatorio e non ripetibile

- **<filesize>** : fornisce la dimensione del file contenente lo stream video in byte. Opzionale e non ripetibile
- **<video_dimensions>** : definisce le dimensioni dello stream video. Obbligatorio e non ripetibile
- **<video_metrics>** : fornisce le principali caratteristiche tecniche dello stream video. Opzionale e non ripetibile
- **<format>** : dichiara il formato dello stream video. Opzionale e non ripetibile
- **<digitisation>** : descrive il tipo di trascrizione dello stream video. Opzionale e non ripetibile
- **<datetimecreated>** : dichiara la data e l'ora della creazione dello stream video digitale. Opzionale e non ripetibile

Per l'elemento è definito un solo attributo:

- **videogroupID** : di tipo `xsd:IDREF` contiene un riferimento all'attributo `ID` dell'elemento `<video_group>` (p. 45). Tale attributo consente di collegare un `<proxies>` con le caratteristiche tecniche definite globalmente da `<video_group>`. L'attributo è opzionale e si usa quando dallo stesso oggetto analogico sono stati tratti più formati, descritti ciascuno da un proprio elemento `<proxies>`. Qualora dall'oggetto analogico sia stato tratto un solo oggetto digitale, l'attributo dovrebbe essere usato all'interno dell'elemento `<video>`. L'uso è ovviamente subordinato all'impiego dell'elemento `<video_group>` all'interno della sezione GEN. Qualora l'attributo non sia usato né all'interno di `<video>` né all'interno dell'elemento `<proxies>` si assume che le caratteristiche tecniche della traccia audio digitale non siano state altrove descritte e quindi l'elemento `<video_metrics>` deve ritenersi obbligatorio, così come `<format>` e `<digitisation>`. Si danno quindi tre possibilità:
 - **l'attributo `videogroupID` è usato all'interno di `<video>`** : le caratteristiche tecniche dello stream video sono definite all'interno dell'elemento `<video_group>` della sezione GEN. Esiste un solo elemento `<proxies>` all'interno di `<video>` in quanto dell'oggetto analogico è stata tratta un'unica versione
 - **l'attributo `videogroupID` è usato all'interno di `<proxies>`** : le caratteristiche tecniche dello stream video sono definite all'interno dell'elemento `<video_group>` della sezione GEN. Esistono più elementi `<proxies>` all'interno di `<video>`, ognuno dei quali descrive un formato alternativo del medesimo oggetto analogico. Si assume che esistano più `<video_group>`.
 - **l'attributo `videogroupID` non è usato né all'interno di `<proxies>` né all'interno di `<video>`** : le caratteristiche tecniche dello stream video digitale non sono state altrove descritte e quindi gli elementi `<video_metrics>`, `<format>` e `<digitisation>` figli di `<proxies>`, formalmente opzionali, debbono ritenersi obbligatori.

Per esempio:

```
<gen>  
<!-- omissis -->
```

```
<video_group ID="vGroupIntra">
  <!-- omissis -->
</video_group>
<video_group ID="vGroupInter">
  <!-- omissis -->
</video_group>
</gen>
<!-- omissis -->
<video>
  <sequence_number>1</sequence_number>
  <nomenclature>Aida - Atto 1°</nomenclature>
  <proxies videogroupID="vGroupIntra">
    <!-- omissis -->
  </proxies>
  <proxies videogroupID="vGroupInter">
    <!-- omissis -->
  </proxies>
</video>
```

Il tipo complesso `videoproxies` che descrive l'elemento `<proxies>` è così formalmente definito (file `metatype.xsd`):

```
<xsd:complexType name="videoproxy">
  <xsd:sequence>
    <xsd:element name="usage" type="usages" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="file" type="link"/>
    <xsd:element name="md5" type="niso:checksum"/>
    <xsd:element name="filesize" type="xsd:unsignedLong" minOccurs="0"/>
    <xsd:element name="video_dimensions" type="video_dimensions"/>
    <xsd:element name="video_metrics" type="video_spatialmetrics" minOccurs="0"/>
    <xsd:element name="format" type="video_format" minOccurs="0"/>
    <xsd:element name="digitisation" type="video_creation" minOccurs="0"/>
    <xsd:element name="datetimecreated" type="xsd:dateTime" minOccurs="0"/>
  </xsd:sequence>
  <xsd:attribute name="videogroupID" type="xsd:IDREF" use="optional"/>
</xsd:complexType>
```

9.2.1. Uso, localizzazione, integrità e grandezza del file

Dello stesso oggetto digitale (tipicamente una cassetta VHS, una bobina) possono essere tratti più stream video digitali, di qualità più o meno elevata, in diversi formati, ognuno dei quali con una diversa finalità. È infatti usuale creare file video di alta qualità per l'archiviazione interna e file di qualità più limitata per la diffusione esterna. La finalità dello stream video viene registrato dall'elemento `<usage>`. L'elemento è di tipo `xsd:string`; al fine di favorire la portabilità dei dati, si consiglia tuttavia di adottare le seguenti due tassonomie

(adottate dai maggiori progetti di digitalizzazione italiani), la prima relativa alle modalità d'uso, la seconda al possesso del copyright da parte dell'istituzione:

- 1 : master
- 2 : alta qualità
- 3 : bassa qualità
- 4 : preview

e

- a : il repository non ha il copyright dell'oggetto digitale
- b : il repository ha il copyright dell'oggetto digitale

L'elemento è opzionale e ripetibile.

L'elemento `<file>` consente di localizzare il file che contiene lo stream video. È di tipo `link`, vale a dire che è un elemento vuoto che supporta attributi definiti dal namespace `xlink` (p. 33). L'elemento è obbligatorio e non ripetibile.

L'integrità del contenuto digitale è verificata grazie alla sua impronta digitale, registrata dall'elemento `<md5>`, un codice standard di 32 caratteri che viene rilevato automaticamente grazie all'impiego di appositi applicativi. Le regole per il rilevamento dell'impronta devono essere definite localmente, così come i momenti per il rilievo stesso (prima del momento del deposito, al momento del deposito, o in entrambi i momenti). Si tratta di una raccomandazione NISO e come tale il tipo specializzato che governa il contenuto dell'elemento appartiene al namespace `niso` ed è denominato `niso:checksum` (p. 89). Tale tipo è definito come restrizione di `xsd:string` che limita la lunghezza massima della stringa a 32 caratteri.

La grandezza del file (che va espressa in byte) è registrata dall'elemento `<filesize>`. L'elemento è di tipo `xsd:unsignedLong` (un numero non negativo il cui valore massimo è 18446744073709551615; si veda la definizione fornita dal W3C al sito <http://www.w3.org/TR/2004/REC-xmlschema-2-20041028/datatypes.html>), è opzionale e non ripetibile. Anche l'elemento `<filesize>` è una raccomandazione NISO (Cfr. [Data Dictionary](#), p. 13).

Si veda l'esempio:

```
<usage>2</usage>
<file Location="URL" xlink:href="15641/211835_1_video.avi"/>
<md5>25189f63903b242a96908e350817a714</md5>
<filesize>6014976</filesize>
```

9.2.2. Le dimensioni stream video

Le dimensioni dello stream sono codificate dall'elemento obbligatorio e non ripetibile `<video_dimensions>`; poiché la dimensione di uno stream video è data esclusivamente dalla sua durata, l'elemento `<video_dimensions>` contiene un solo elemento `<duration>` (anch'esso obbligatorio e non ripetibile) di tipo `xsd:time`. Per esempio:

```
<video_dimensions>
  <duration>00:36:59</duration>
</video_dimensions>
```

Il tipo complesso `video_dimensions` che regola il contenuto dell'elemento `<video_dimensions>` è formalmente definito come segue (file `video-mag-xsd`):

```
<xsd:complexType name="video_dimensions">
  <xsd:sequence>
    <xsd:element name="duration" type="xsd:time"/>
  </xsd:sequence>
</xsd:complexType>
```

9.2.3. Principali caratteristiche tecniche dello stream video

Le principali caratteristiche tecniche dello stream video sono contenute nell'elemento `<video_metrics>`. L'elemento, oltre che dentro `<proxies>`, può essere usato dentro `<video_group>` (p. 45); è formalmente opzionale, ma in pratica deve sempre essere usato, o dentro `<proxies>` o dentro `<video_group>`. Per `<video_metrics>` è definito un tipo specializzato denominato `video_spatialmetrics`. Tale tipo è di tipo `xsd:sequence` e contiene:

- `<videosize>` : contiene la definizione del frame misurato in pixel. l'elemento è obbligatorio e non ripetibile. Per l'elemento è definito un tipo semplice specializzato denominato `videosizetype`; tale tipo è una restrizione del tipo `xsd:string` e ammette i seguenti valori:
 - **160x120** : applicabile per il formato Real Video
 - **176x144** : applicabile per il formato Windows Media
 - **192x144** : applicabile per la diffusione via internet
 - **280x180** : applicabile per il formato RealVideo
 - **320x240** : applicabile per il formato RealVideo
 - **352x288** : applicabile per il formato VHS, VCD
 - **360x288** : applicabile per il formato Windows Media
 - **384x288** : applicabile per il formato multimedia
 - **480x576** : applicabile per il formato SVCD
 - **720x576** : applicabile per il formato DVD

Il tipo `videosizetype` è formalmente definito come segue (file `video-mag.xsd`):

```
<xsd:simpleType name="videosizetype">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="160x120">
      <xsd:annotation>
        <xsd:documentation>RealVideo</xsd:documentation>
      </xsd:annotation>
    </xsd:enumeration>
  </xsd:restriction>
</xsd:simpleType>
```



```
</xsd:enumeration>
<xsd:enumeration value="176x144">
  <xsd:annotation>
    <xsd:documentation>Windows Media</xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="192x144">
  <xsd:annotation>
    <xsd:documentation>internet</xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="280x180">
  <xsd:annotation>
    <xsd:documentation>RealVideo</xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="320x240">
  <xsd:annotation>
    <xsd:documentation>RealVideo</xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="352x288">
  <xsd:annotation>
    <xsd:documentation>VHS, VCD</xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="360x288">
  <xsd:annotation>
    <xsd:documentation>Windows Media</xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="384x288">
  <xsd:annotation>
    <xsd:documentation>multimedia</xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="480x576">
  <xsd:annotation>
    <xsd:documentation>SVCD</xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>
<xsd:enumeration value="720x576">
  <xsd:annotation>
    <xsd:documentation>DVD</xsd:documentation>
  </xsd:annotation>
</xsd:enumeration>
</xsd:restriction>
</xsd:simpleType>
```

- **<aspectratio>** : contiene il formato dell'immagine (larghezza:altezza). L'elemento è opzionale e non ripetibile. Per l'elemento è definito un tipo semplice specializzato denominato `aspectratio`; tale tipo è una restrizione del tipo `xsd:string` e ammette i seguenti valori (separati da |): 1:1 | 4:3 | 16:9 | 2.11:1. Il tipo `aspectratio` è formalmente definito come segue (file `video-mag.xsd`):

```
<xsd:simpleType name="aspectratio">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="1:1"/>
    <xsd:enumeration value="4:3"/>
    <xsd:enumeration value="16:9"/>
    <xsd:enumeration value="2.11:1"/>
  </xsd:restriction>
</xsd:simpleType>
```

- **<framerate>** : contiene il numero di quadri al secondo del flusso video. L'elemento è obbligatorio e non ripetibile. Per l'elemento è definito un tipo semplice specializzato denominato `framerate`; tale tipo è una restrizione del tipo `xsd:float` (si veda <http://www.w3.org/TR/2004/REC-xmlschema-2-20041028/datatypes.html>) e ammette i seguenti valori (separati da |): 23.976 | 24 | 25 | 29.97 | 30 | 50 | 59.94 | 60. Il tipo `framerate` è formalmente definito come segue (file `video-mag.xsd`):

```
<xsd:simpleType name="framerate">
  <xsd:restriction base="xsd:float">
    <xsd:enumeration value="23.976"/>
    <xsd:enumeration value="24"/>
    <xsd:enumeration value="25"/>
    <xsd:enumeration value="29.97"/>
    <xsd:enumeration value="30"/>
    <xsd:enumeration value="50"/>
    <xsd:enumeration value="59.94"/>
    <xsd:enumeration value="60"/>
  </xsd:restriction>
</xsd:simpleType>
```

Si veda l'esempio:

```
<video_metrics>
  <videosize>720x576</videosize>
  <aspectratio>4:3</aspectratio>
  <framerate>25</framerate>
</video_metrics>
```

Il tipo complesso `video_spatialmetrics` che regola il contenuto dell'elemento `<video_metrics>` è così formalmente definito (file `video-mag.xsd`):

```
<xsd:complexType name="video_spatialmetrics">
  <xsd:sequence>
    <xsd:element name="videosize" type="videosizetype"/>
    <xsd:element name="aspectratio" type="aspectratiotype" minOccurs="0"/>
    <xsd:element name="framerate" type="frameratetype"/>
  </xsd:sequence>
</xsd:complexType>
```

9.2.4. Il formato degli stream video

Il formato degli stream video (tipologia e modalità di compressione) è gestito dall'elemento `<format>`. L'elemento, oltre che dentro `<proxies>`, può essere usato dentro `<video_group>` (p. 45); è formalmente opzionale, ma in pratica deve sempre essere usato, o dentro `<proxies>` o dentro `<video_group>`. Per `<format>` è definito un tipo specializzato denominato `video_format`. Tale tipo è di tipo `xsd:sequence` e contiene quattro elementi:

- **<name>** : obbligatorio e non ripetibile, contiene il formato dello stream video. È di tipo `xsd:string`, ma si consiglia di usare valori standardizzati come WMV, MPEG, ASF, ecc.
- **<mime>** : obbligatorio e non ripetibile, contiene il tipo mime dello stream video. Per l'elemento è definito un tipo semplice specializzato, anch'esso contenuto nel file `video-mag.xsd`, denominato `video_mimetype`. Tale tipo è definito come restrizione di `xsd:string` ed è costituito dall'enumerazione dei seguenti valori:

- `video/x-ms-asf`
- `video/avi`
- `video/mpeg`
- `video/vnd.rn-realvideo`
- `video/wmv`

Il tipo `video_mimetype` è formalmente definito come segue (file `video-mag.xsd`):

```
<xsd:simpleType name="mimetype">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="video/x-ms-asf"/>
    <xsd:enumeration value="video/avi"/>
    <xsd:enumeration value="video/mpeg"/>
    <xsd:enumeration value="video/vnd.rn-realvideo"/>
    <xsd:enumeration value="video/wmv"/>
  </xsd:restriction>
</xsd:simpleType>
```

- **<videofORMAT>** : obbligatorio e non ripetibile, contiene il tipo di standard video adottato. Per l'elemento è definito un tipo semplice specializzato, anch'esso contenuto nel file `video-mag.xsd`, denominato `videoformattype`. Tale tipo è definito come restrizione di `xsd:string` ed è costituito dall'enumerazione dei seguenti valori (separati da |): `Component` | `NTSC` | `PAL` | `SECAM` | `Unspecified`. Il tipo `videoformattype` è formalmente definito come segue (file `video-mag.xsd`):

```
<xsd:simpleType name="videoformattype">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="Component"/>
    <xsd:enumeration value="NTSC"/>
    <xsd:enumeration value="PAL"/>
    <xsd:enumeration value="SECAM"/>
    <xsd:enumeration value="Unspecified"/>
  </xsd:restriction>
</xsd:simpleType>
```

- **<enCODE>** : opzionale e non ripetibile, contiene il tipo di encode adottato. Per l'elemento è definito un tipo semplice specializzato, anch'esso contenuto nel file `video-mag.xsd`, denominato `encodetype`. Tale tipo è definito come restrizione di `xsd:string` ed è costituito dall'enumerazione dei seguenti valori (separati da |): `interlaced` | `non-interlaced`. Il tipo `encodetype` è formalmente definito come segue (file `video-mag.xsd`):

```
<xsd:simpleType name="encodetype">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="interlaced"/>
    <xsd:enumeration value="non-interlaced"/>
  </xsd:restriction>
</xsd:simpleType>
```

- **<streamTYPE>** : opzionale e non ripetibile, contiene tipo di compressione e incapsulamento dello stream video. Per l'elemento è definito un tipo semplice specializzato, anch'esso contenuto nel file `video-mag.xsd`, denominato `streamtypetype`. Tale tipo è definito come restrizione di `xsd:string` ed è costituito dall'enumerazione dei seguenti valori (separati da |): `Uncompressed` | `MPEG-1` | `MPEG-2` | `MPEG-4`. Il tipo `streamtypetype` è formalmente definito come segue (file `video-mag.xsd`):

```
<xsd:simpleType name="streamtypetype">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="Uncompressed"/>
    <xsd:enumeration value="MPEG-1"/>
    <xsd:enumeration value="MPEG-2"/>
    <xsd:enumeration value="MPEG-4"/>
  </xsd:restriction>
</xsd:simpleType>
```

```
</xsd:restriction>  
</xsd:simpleType>
```

- **<codec>** : opzionale e non ripetibile, descrive il tipo di codifica del contenuto video. L'elemento è di tipo `xsd:string`, si consiglia tuttavia di utilizzare un vocabolario controllato del tipo DIVX, XVID.

Si veda l'esempio:

```
<format>  
  <name>avi</name>  
  <mime>video/avi</mime>  
  <videofORMAT>PAL</videofORMAT>  
  <encode>interlaced</encode>  
  <streamtype>Uncompressed</streamtype>  
  <codec>digital video</codec>  
</format>
```

Il tipo complesso `video_format` che regola il contenuto dell'elemento `<format>` è così formalmente definito (file `video-mag.xsd`):

```
<xsd:complexType name="video_format">  
  <xsd:sequence>  
    <xsd:element name="name" type="xsd:string"/>  
    <xsd:element name="mime" type="mimetype"/>  
    <xsd:element name="videofORMAT" type="videofORMATtype"/>  
    <xsd:element name="encode" type="encodetype" minOccurs="0"/>  
    <xsd:element name="streamtype" type="streamtypetype" minOccurs="0"/>  
    <xsd:element name="codec" type="xsd:string" minOccurs="0"/>  
  </xsd:sequence>  
</xsd:complexType>
```

9.2.5. La creazione dello stream video digitale

Le modalità della creazione dello stream video digitale sono descritte dall'elemento `<digitisation>`. Oltre che dentro `<proxies>`, può essere usato all'interno di `<video_group>` (p. 45); per tale ragione è formalmente opzionale, ma in realtà deve essere usato o all'interno di `<video_group>` o dentro `<proxies>`. L'elemento non è ripetibile, è di tipo `xsd:sequence` e può contenere i seguenti elementi:

- **<sourcetype>** : opzionale e non ripetibile, descrive il tipo di fonte analogica utilizzata per l'acquisizione dello stream video; è di tipo `xsd:string` ma si consiglia tuttavia di adottare la seguente tassonomia desunta dal *Catalogo Gestionale della Discoteca di Stato*:

- Filmato
 - Videoregistrazione
 - Videocartuccia
 - Videodisco
 - Videocassetta
 - Bobina video
 - Videoregistrazione elettronica (EVR)
 - Materiale da proiettare
 - Diapositiva, set di diapositive, stereografo
 - DVD
 - Altro
- **<transcriptionagency>** : opzionale, contiene il nome dell'istituzione che ha effettuato la digitalizzazione della traccia audio; è utile nel caso la digitalizzazione non sia stata effettuata direttamente dall'istituzione responsabile del processo di digitalizzazione registrato elemento <agency> (p. 38). L'elemento non è ripetibile ed è di tipo xsd:string.
- **<devicesource>** : opzionale ma raccomandato, descrive la tipologia dell'apparecchiatura usata per realizzare la digitalizzazione. Non ripetibile, è di tipo xsd:string, ma per ragioni di omogeneità si consiglia comunque di adottare, ove possibile, la seguente nomenclatura:
- dispositivo di lettura video
 - videoregistratore
 - scheda di acquisizione video
 - dispositivo telecinema

Si veda l'esempio:

```
<digitisation>
  <sourcetype>Betacam SP</sourcetype>
  <transcriptionagency>Discoteca di Stato - Museo dell'Audiovisivo</transcriptionagency>
  <devicesource>Sony Betacam SP</devicesource>
  <transcriptionchain>
    <!-- omissis -->
  </transcriptionchain>
</digitisation>
```

- **<transcriptionchain>** : opzionale, descrive gli strumenti usati nel processo di digitalizzazione. Ripetibile, è di tipo xsd:sequence e contiene cinque elementi:
- <device_description>
 - <device_manufacturer>
 - <device_model>
 - <capture_software>
 - <device_setting>

Per i dettagli si veda il par. *Gli strumenti di digitalizzazione* (p. 128).

- **<transcriptionsummary>** : opzionale e ripetibile, raccoglie i dati misurati durante il processo di digitalizzazione **per i valori di sintesi**. Permette di definire nomi, tipi e valori delle grandezze fisiche misurate, consentendone una nidificazione gerarchica (ad esempio valori per canale). Supporta due formati alternativi: 1. con gli elementi <grouping> e <transcriptionsummary> si possono definire dei raggruppamenti di tipologie di dati; 2. Con gli elementi <data_description>, <data_unit> e <data_value> si definiscono le misure considerate. Per i dettagli si veda il par. *Dati tecnici della trascrizione* (p. 129).
- **<transcriptiondata>** : opzionale e ripetibile, raccoglie le sequenze di dati misurate durante il processo di digitalizzazione. Permette di definire nomi, tipi e valori delle grandezze fisiche misurate, consentendone una nidificazione gerarchica. Supporta due formati alternativi: 1. con gli elementi <grouping> e <transcriptiondata> si possono definire dei raggruppamenti di tipologie di dati; 2. Con gli elementi <data_description>, <data_unit>, <interval> e <data_value> si definiscono le misure considerate. *Dati tecnici della trascrizione* (p.129).

Il tipo complesso `video_creation` che regola il contenuto dell'elemento <digitisation> è formalmente definito come segue (file `video-mag.xsd`):

```
<xsd:complexType name="video_creation">
  <xsd:sequence>
    <xsd:element name="sourcetype" type="xsd:string" minOccurs="0"/>
    <xsd:element name="transcriptionagency" minOccurs="0"/>
    <xsd:element name="devicesource" type="xsd:string" minOccurs="0"/>
    <xsd:element name="transcriptionchain" minOccurs="0" maxOccurs="unbounded">
      <xsd:complexType>
        <xsd:sequence>
          <xsd:element name="device_description">
            <xsd:complexType>
              <xsd:simpleContent>
                <xsd:extension base="xsd:string">
                  <xsd:attribute name="Type" type="xsd:string" use="required"/>
                  <xsd:attribute name="Unique_identifier" type="xsd:string" use="optional"/>
                  <xsd:attribute name="Comments" type="xsd:string" use="optional"/>
                </xsd:extension>
              </xsd:simpleContent>
            </xsd:complexType>
          </xsd:element>
          <xsd:element name="device_manufacturer">
            <xsd:complexType>
              <xsd:simpleContent>
                <xsd:extension base="xsd:string">
                  <xsd:attribute name="Manufacturer" type="xsd:string" use="required"/>
                </xsd:extension>
              </xsd:simpleContent>
            </xsd:complexType>
          </xsd:element>
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  </xsd:sequence>
</xsd:complexType>
```

```
</xsd:element>
<xsd:element name="device_model">
  <xsd:complexType>
    <xsd:simpleContent>
      <xsd:extension base="xsd:string">
        <xsd:attribute name="Model" type="xsd:string" use="required"/>
        <xsd:attribute name="Serial_Number" type="xsd:string" use="optional"/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>
</xsd:element>
<xsd:element name="capture_software" type="xsd:string" minOccurs="0"/>
<xsd:element name="device_settings" type="xsd:string" minOccurs="0"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="transcriptionsummary" type="transcriptionsummarytype"
  minOccurs="0" maxOccurs="unbounded"/>
<xsd:element name="transcriptiondata" type="transcriptiondatatype"
  minOccurs="0" maxOccurs="unbounded"/>
</xsd:sequence>
</xsd:complexType>
```

9.2.5.1. Gli strumenti di digitalizzazione

L'elemento `<transcriptionchain>` consente di descrivere accuratamente gli strumenti hardware e software utilizzati dal processo di digitalizzazione. L'elemento contiene cinque elementi:

- **<device_description>** : obbligatorio e non ripetibile, contiene le generalità sul tipo di dispositivo video utilizzato per la digitalizzazione: Per l'elemento sono definiti tre attributi:
 - **Type** : obbligatorio e di tipo `xsd:string` descrive il tipo di dispositivo
 - **Unique_identifier** : opzionale e di tipo `xsd:string`, contiene un codice identificativo del dispositivo
 - **Comments** : opzionale e di tipo `xsd:string`, contiene ogni eventuale commento circa il dispositivo
- **<device_manufacturer>** : obbligatorio e non ripetibile, contiene il nome del produttore del dispositivo usato per la trascrizione; è di tipo `xsd:string`
- **<device_model>** : obbligatorio e non ripetibile, contiene il nome del modello del dispositivo usato per la trascrizione. Per l'elemento sono definiti due attributi:
 - **Model** : obbligatorio e di tipo `xsd:string` contiene il nome del modello del dispositivo
 - **Serial_Number** : opzionale e di tipo `xsd:string`, contiene il numero seriale del

dispositivo

- **<capture_software>** : opzionale e non ripetibile, contiene il nome del software utilizzato per l'acquisizione dello stream video; è di tipo `xsd:string`.
- **<device_setting>** : opzionale e non ripetibile, descrive le impostazioni usate dal dispositivo di acquisizione.

Per esempio:

```
<transcriptionchain>
  <device_description Type="convertitore video A/D" Unique_identifier="DVrex"
    Comments="dispositivo acquistato nel 2005"/>
  <device_manufacturer>Canopus</device_manufacturer>
  <device_model Model="DVrex"/>
  <capture_software>Video editing program</capture_software>
  <device_settings>DV compression</device_settings>
</transcriptionchain>
```

9.2.5.2. Dati tecnici della trascrizione: valori di sintesi e valori globali

Gli elementi `<transcriptionsummary>` e `<transcriptiondata>` raccolgono i dati tecnici relativi alla trascrizione, il primo raccoglie i dati di sintesi, il secondo i dati assoluti del file digitale. Entrambi gli elementi possono essere usati per definire nomi, tipi e valori delle grandezze fisiche misurate, consentendone una nidificazione gerarchica (ad esempio valori per canale). Entrambi gli elementi supportano due formati alternativi (`xsd:choice`), uno per definire raggruppamenti di tipologie di dati, l'altro per raccogliere effettivamente i dati.

Vediamo ora il dettaglio dei due elementi.

- **<transcriptionsummary>** : raccoglie i dati misurati durante il processo di digitalizzazione **per i valori di sintesi**. Supporta due formati alternativi:
 - con gli elementi `<grouping>` (obbligatorio) e `<transcriptionsummary>` (opzionale) si possono definire dei raggruppamenti di tipologie di dati. L'elemento `<grouping>` di tipo `xsd:string` contiene l'etichetta della tipologia dei dati considerati.
 - le misure considerate si definiscono grazie agli elementi `<data_description>` (di tipo `xsd:string` e obbligatorio) che contiene la descrizione del tipo di dati, `<data_unit>` (opzionale di tipo `xsd:string`) che contiene l'unità di misura per i dati considerati, e `<data_value>` (obbligatorio, di tipo `xsd:float`) che contiene il valore dei dati considerati.

Si veda il seguente esempio:

```
<transcriptionsummary>
  <data_description>drop frames</data_description>
  <data_unit>frames</data_unit>
```

```
<data_value>5</data_value>
</transcriptionsummary>
<transcriptionsummary>
  <data_description>SNR</data_description>
  <data_unit>dB</data_unit>
  <data_value>24.5</data_value>
</transcriptionsummary>
```

Il tipo complesso `transcriptionsummarytype` che regola il contenuto dell'elemento `<transcriptionsummary>` è formalmente definito come segue (file `video-mag.xsd`):

```
<xsd:complexType name="video_transcriptionsummarytype">
  <xsd:choice>
    <xsd:sequence>
      <xsd:element name="grouping" type="xsd:string"/>
      <xsd:element name="transcriptionsummary"
        type="video_transcriptionsummarytype" minOccurs="0"
        maxOccurs="unbounded"/>
    </xsd:sequence>
    <xsd:sequence>
      <xsd:element name="data_description" type="xsd:string"/>
      <xsd:element name="data_unit" type="xsd:string" minOccurs="0"/>
      <xsd:element name="data_value" type="xsd:float"/>
    </xsd:sequence>
  </xsd:choice>
</xsd:complexType>
```

- **<transcriptiondata>** : raccoglie i dati misurati durante il processo di digitalizzazione. Supporta due formati alternativi:
 - con gli elementi `<grouping>` e `<transcriptiondata>` si possono definire dei raggruppamenti di tipologie di dati. L'elemento `<grouping>` contiene l'etichetta della tipologia dei dati considerati.
 - le misure considerate si definiscono grazie agli elementi `<data_description>` (di tipo `xsd:string` e obbligatorio) che contiene la descrizione del tipo di dati, `<data_unit>` (opzionale di tipo `xsd:string`) che contiene l'unità di misura per i dati considerati, `<interval>` che definisce l'intervallo di tempo entro il quale i valori sono misurati, e `<data_value>` (obbligatorio, di tipo `xsd:float`) che contiene il valore dei dati considerati. L'elemento `<interval>` supporta due attributi obbligatori di tipo `xsd:time`, `start` e `stop` che definiscono rispettivamente l'inizio e la fine dell'intervallo di misurazione dei dati considerati.

Il tipo complesso `transcriptiondatatype` che regola il contenuto dell'elemento `<transcriptiondata>` è formalmente definito come segue (file `video-mag.xsd`):

```
<xsd:complexType name="video_transcriptiondatatype">
  <xsd:choice>
    <xsd:sequence>
      <xsd:element name="grouping" type="xsd:string"/>
      <xsd:element name="transcriptiondata"
        type="video_transcriptiondatatype" minOccurs="0"
        maxOccurs="unbounded"/>
    </xsd:sequence>
    <xsd:sequence>
      <xsd:element name="data_description" type="xsd:string"/>
      <xsd:element name="data_unit" type="xsd:string" minOccurs="0"/>
      <xsd:element name="interval" minOccurs="0">
        <xsd:complexType>
          <xsd:attribute name="start" type="xsd:time"
            use="required"/>
          <xsd:attribute name="stop" type="xsd:time"
            use="required"/>
        </xsd:complexType>
      </xsd:element>
      <xsd:element name="data_value" type="xsd:float"
        maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:choice>
</xsd:complexType>
```

9.2.6. La creazione del file

Il momento (data e ora) della creazione del file digitale (che può non coincidere con la data e il momento della trascrizione digitale, per esempio nel caso di correzioni apportate al file) viene registrata dell'elemento `<datetimecreated>`. L'elemento è di tipo `xsd:dateTime`, è opzionale e non ripetibile. Per esempio:

```
<datetimecreated>2005-12-28T19:22:42</datetimecreated>
```

10. Sezione OCR

La sezione OCR raccoglie i metadati amministrativi e gestionali relativi a file di testo ottenuti mediante riconoscimento ottico automatico del contenuto. I file classificabili all'interno di tale sezione si differenziano da quelli descritti dalla sezione DOC perché non presuppongono alcun intervento correttivo manuale e quindi qualsiasi intervento editoriale è assente.

La sezione OCR utilizza il namespace `niso`: che fa riferimento allo schema che traduce le linee guida del Data Dictionary NISO. Anche se lo standard NISO è stato concepito per la descrizione delle caratteristiche tecniche delle immagini digitali, lo schema `niso-mag.xsd` include anche specifiche tecniche relative a documenti **text-oriented** (riconoscibili grazie al prefisso `doc`). Tale schema è stato realizzato dal Comitato MAG ed è interamente documentato nel paragrafo **Lo schema Niso** (p. 89).

La sezione OCR è costituita di una sequenza di elementi `<ocr>`, uno per ciascun file di testo descritto da MAG. L'elemento è opzionale e ripetibile. Il suo contenuto è di tipo `xsd:sequence`, e può contenere i seguenti elementi:

- `<sequence_number>` : contiene il numero di sequenza identificativo del file di testo. Obbligatorio e non ripetibile
- `<nomenclature>` : contiene la denominazione o titolo del file di testo. Obbligatorio e non ripetibile.
- `<usage>` : definisce l'ambito d'uso del file di testo in relazione agli standard di progetto. Opzionale e ripetibile
- `<file>` : localizza il file contenente il file ocr. Obbligatorio e non ripetibile
- `<md5>` : contiene l'impronta del file. Obbligatorio e non ripetibile
- `<source>` : contiene il link al file sorgente, cioè l'immagine da cui è stata tratto il file OCR. Obbligatorio e non ripetibile
- `<filesize>` : fornisce la dimensione del file di testo in byte. Opzionale e non ripetibile
- `<format>` : dichiara il formato del file di testo. Obbligatorio e non ripetibile
- `<software_ocr>` : dichiara il tipo di software usato per il riconoscimento ottico automatico del contenuto. Opzionale e non ripetibile
- `<datetimecreated>` : dichiara la data e l'ora della creazione del file di testo. Opzionale e non ripetibile
- `<note>` : eventuali annotazioni. Opzionale e non ripetibile

Per l'elemento è definito un solo attributo:

- `holdingsID` : di tipo `xsd:IDREF` contiene un riferimento all'attributo `ID` dell'elemento `<holdings>` (p. 53) e serve a definire a quale istituzione appartiene l'oggetto analogico digitalizzato. L'attributo è opzionale.

L'elemento `<ocr>` è così formalmente definito (file `metatype.xsd`):

```
<xsd:complexType name="ocr">
  <xsd:sequence>
    <xsd:element name="sequence_number" type="xsd:positiveInteger"/>
    <xsd:element name="nomenclature" type="xsd:string"/>
  </xsd:sequence>
</xsd:complexType>
```

```
<xsd:element name="usage" type="usages" minOccurs="0" maxOccurs="unbounded" />
<xsd:element name="file" type="link" />
<xsd:element name="md5" type="niso:checksum" />
<xsd:element name="source" type="link" />
<xsd:element name="filesize" type="xsd:positiveInteger" minOccurs="0" />
<xsd:element name="format" type="niso:docFormat" />
<xsd:element name="software_ocr" type="xsd:string" minOccurs="0" />
<xsd:element name="datetimecreated" type="xsd:dateTime" minOccurs="0" />
<xsd:element name="note" type="xsd:string" minOccurs="0" />
</xsd:sequence>
<xsd:attribute name="holdingsID" type="xsd:IDREF" use="optional" />
</xsd:complexType>
```

10.1. Identificazione, denominazione e uso dei file di testo

Ciascun file di testo descritto all'interno di MAG viene identificato univocamente grazie all'elemento `<sequence_number>`. Il suo contenuto è di tipo `xsd:positiveInteger`, vale a dire che è costituito di un numero positivo. Dato il suo compito di identificatore, sul contenuto di `<sequence_number>` è stato posto un vincolo di univocità (file `metadigit.xsd`):

```
<xsd:unique name="uniquocr">
  <xsd:selector xpath="ocr" />
  <xsd:field xpath="sequence_number" />
</xsd:unique>
```

L'elemento è obbligatorio e non ripetibile. Il valore dell'elemento è puntato dagli attributi `sequence_number` degli elementi `<start>` e `<stop>` della sezione `<stru>` (p. 61).

Al file ocr deve inoltre essere attribuita una denominazione, per esempio Pagina 1, Capitolo 1, ecc. Tale denominazione viene codificata dall'elemento `<nomenclature>`. L'elemento è di tipo `xsd:string`; si consiglia comunque di definire una nomenclatura controllata negli standard di progetto. L'elemento è obbligatorio e non ripetibile.

La finalità del file di testo viene registrata dall'elemento `<usage>`. L'elemento è di tipo `xsd:string`; al fine di favorire la portabilità dei dati, si consiglia tuttavia di adottare la seguente tassonomia relativa al possesso del copyright da parte dell'istituzione:

- **a** : il repository non ha il copyright dell'oggetto digitale
- **b** : il repository ha il copyright dell'oggetto digitale

L'elemento è opzionale e ripetibile.

10.2. Localizzazione, integrità, fonte e grandezza del file di testo

L'elemento `<file>` consente di localizzare il file di testo ocr. È di tipo `link`, vale a dire che è un elemento vuoto che supporta attributi definiti dal namespace `xlink` (p. 33). L'elemento è obbligatorio e non ripetibile.

L'integrità del contenuto digitale è verificata grazie alla sua impronta digitale,

registrata dall'elemento `<md5>`, un codice standard di 32 caratteri che viene rilevato automaticamente grazie all'impiego di appositi applicativi. Le regole per il rilevamento dell'impronta devono essere definite localmente, così come i momenti per il rilievo stesso (prima del momento del deposito, al momento del deposito, o in entrambi i momenti). Si tratta di una raccomandazione NISO e come tale il tipo specializzato che governa il contenuto dell'elemento appartiene al namespace `niso` ed è denominato `niso:checksum` (p. 89). Tale tipo è definito come restrizione di `xsd:string` che limita la lunghezza massima della stringa a 32 caratteri.

Il file da cui è stato tratto il file di testo (tipicamente un'immagine digitale) è localizzato dall'elemento `<source>`. È di tipo `link`, vale a dire che è un elemento vuoto che supporta attributi definiti dal namespace `xlink` (p. 33). L'elemento è obbligatorio e non ripetibile.

La grandezza del file (che va espressa in byte) è registrata dall'elemento `<filesize>`. L'elemento è di tipo `xsd:Integer` (un numero positivo), è opzionale e non ripetibile. Anche l'elemento `<filesize>` è una raccomandazione NISO (Cfr. [Data Dictionary](#), p. 13).

10.3. Il formato dei file di testo ocr

Il formato dei file di testo ocr (tipologia e modalità di compressione) è gestito dall'elemento `<format>` ed è codificato secondo lo standard NISO. Per `<format>` è definito un tipo specializzato appartenente al namespace `niso` (p. 89) denominato `niso:docFormat`. Tale tipo è di tipo `xsd:sequence` e contiene tre elementi:

- **`<niso:name>`** : obbligatorio e non ripetibile, contiene il formato del file testo. È di tipo `xsd:string`, si consiglia di usare valori come `TXT` ecc. Si raccomanda di usare valori formati da tre caratteri.
- **`<niso:mime>`** : obbligatorio e non ripetibile, contiene il tipo mime del file di testo. Per l'elemento è definito un tipo semplice specializzato, anch'esso contenuto nel file `niso-mag.xsd`, denominato `niso:doc_mimetype`. Tale tipo è definito come restrizione di `xsd:string` ed è costituito dall'enumerazione dei seguenti valori:
 - `text/plain`
 - `text/xml`
 - `text/html`
 - `text/rtf`
 - `application/msword`
 - `application/pdf`
- **`<niso:compression>`** : obbligatorio e non ripetibile, dichiara il tipo di compressione applicato al file di testo. La compressione viene in genere applicata per migliorare l'archiviazione dei file di testo all'interno delle singole repository e per la fase di DIP. Per l'elemento è definito un tipo semplice specializzato, anch'esso contenuto nel file `niso-mag.xsd`, denominato `niso:doc_compressiontype`. Tale tipo è definito come restrizione di `xsd:string` ed è costituito dall'enumerazione dei seguenti valori:
 - `Uncompressed`
 - `ZIP`
 - `RAR`

- GZ

10.4. Il software di riconoscimento ottico

Il tipo di software usato per il riconoscimento ottico del contenuto del file è documentato dall'elemento `<software_ocr>`; l'elemento è opzionale e non ripetibile. È di tipo `xsd:string` e serve per registrare il nome e la versione del software usato.

10.5. La creazione del file

L'elemento `<datetimecreated>` registra la data e l'ora di creazione del file digitale. L'elemento è opzionale e non ripetibile; è di tipo `xsd:dateTime`, vale a dire che assume la forma `YYYY-MM-DDThh:mm:ss.mmm` di cui si vedano le specificazioni nella sezione GEN (p. 38).

Per esempio:

```
<datetimecreated>2005-04-13T02:01:52</datetimecreated>
```

11. Sezione DOC

La sezione DOC raccoglie i metadati amministrativi e gestionali relativi a file di testo **born digital**. Analogamente la sezione DOC descrive anche file frutto di ocr cui è stato applicato un controllo editoriale manuale successivo e pertanto i file classificabili all'interno di tale sezione si differenziano da quelli descritti dalla sezione OCR (dai quali possono derivare) perché presuppongono un intervento editoriale.

La sezione DOC utilizza il namespace `niso`: che fa riferimento allo schema che traduce le linee guida del Data Dictionary NISO. Tale schema è stato realizzato dal Comitato MAG ed è interamente documentato nel paragrafo **Lo schema Niso** (p. 89).

La sezione DOC è costituita di una sequenza di elementi `<doc>`, uno per ciascun file di testo descritto da MAG. L'elemento è opzionale e ripetibile. Il suo contenuto è di tipo `xsd:sequence`, e può contenere i seguenti elementi:

- `<sequence_number>` : contiene il numero di sequenza identificativo del file di testo. Obbligatorio e non ripetibile
- `<nomenclature>` : contiene la denominazione o titolo del file di testo. Obbligatorio e non ripetibile.
- `<usage>` : definisce l'ambito d'uso del file di testo in relazione agli standard di progetto. Opzionale e ripetibile
- `<file>` : localizza il file contenente il documento *born digital*. Obbligatorio e non ripetibile
- `<md5>` : contiene l'impronta del file. Obbligatorio e non ripetibile
- `<filesize>` : fornisce la dimensione del file di testo in byte. Opzionale e non ripetibile
- `<format>` : dichiara il formato del file di testo. Opzionale e non ripetibile
- `<datetimecreated>` : dichiara la data e l'ora della creazione del file di testo. Opzionale e non ripetibile
- `<note>` : eventuali annotazioni. Opzionale e non ripetibile.

Per l'elemento è definito un solo attributo:

- **holdingsID** : di tipo `xsd:IDREF` contiene un riferimento all'attributo `ID` dell'elemento `<holdings>` (p. 53) e serve a definire a quale istituzione appartiene l'oggetto analogico digitalizzato. L'attributo è opzionale.

L'elemento `<doc>` è così formalmente definito (file `metatype.xsd`):

```
<xsd:complexType name="doc">
  <xsd:sequence>
    <xsd:element name="sequence_number" type="xsd:positiveInteger"/>
    <xsd:element name="nomenclature" type="xsd:string"/>
    <xsd:element name="usage" type="usages" minOccurs="0" maxOccurs="unbounded"/>
    <xsd:element name="file" type="link"/>
    <xsd:element name="md5" type="niso:checksum"/>
    <xsd:element name="filesize" type="xsd:positiveInteger" minOccurs="0"/>
    <xsd:element name="format" type="niso:docFormat" minOccurs="0"/>
    <xsd:element name="datetimecreated" type="xsd:dateTime" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```



```
<xsd:element name="note" type="xsd:string" minOccurs="0"/>
</xsd:sequence>
<xsd:attribute name="holdingsID" type="xsd:IDREF" use="optional"/>
</xsd:complexType>
```

11.1. Identificazione, denominazione e uso dei file di testo DOC

Ciascun file di testo descritto all'interno di MAG viene identificato univocamente grazie all'elemento `<sequence_number>`. Il suo contenuto è di tipo `xsd:positiveInteger`, vale a dire che è costituito di un numero positivo. Dato il suo compito di identificatore, sul contenuto di `<sequence_number>` è stato posto un vincolo di univocità (file `metadigit.xsd`):

```
<xsd:unique name="uniqudoc">
  <xsd:selector xpath="doc"/>
  <xsd:field xpath="sequence_number"/>
</xsd:unique>
```

L'elemento è obbligatorio e non ripetibile. Il valore dell'elemento è puntato dagli attributi `sequence_number` degli elementi `<start>` e `<stop>` della sezione `<stru>` (p. 61).

Al file `doc` deve inoltre essere attribuita una denominazione, per esempio `Pagina 1`, `Capitolo 1`, ecc. Tale denominazione viene codificata dall'elemento `<nomenclature>`. L'elemento è di tipo `xsd:string`; si consiglia comunque di definire una nomenclatura controllata negli standard di progetto. L'elemento è obbligatorio e non ripetibile.

La finalità del file di testo viene registrata dall'elemento `<usage>`. L'elemento è di tipo `xsd:string`; al fine di favorire la portabilità dei dati, si consiglia tuttavia di adottare la seguente tassonomia relativa al possesso del copyright da parte dell'istituzione:

- **a** : il repository non ha il copyright dell'oggetto digitale
- **b** : il repository ha il copyright dell'oggetto digitale

L'elemento è opzionale e ripetibile.

11.2. Localizzazione, integrità e grandezza del file di testo

L'elemento `<file>` consente di localizzare il file di testo `born digital`. È di tipo `link`, vale a dire che è un elemento vuoto che supporta attributi definiti dal namespace `xlink` (p. 33). L'elemento è obbligatorio e non ripetibile.

L'integrità del contenuto digitale è verificata grazie alla sua impronta digitale, registrata dall'elemento `<md5>`, un codice standard di 32 caratteri che viene rilevato automaticamente grazie all'impiego di appositi applicativi. Le regole per il rilevamento dell'impronta devono essere definite localmente, così come i momenti per il rilievo stesso (prima del momento del deposito, al momento del deposito, o in entrambi i momenti). Si tratta di una raccomandazione NISO e come tale il tipo specializzato che governa il contenuto dell'elemento appartiene al namespace `niso` ed è denominato `niso:checksum` (p. 89). Tale tipo è definito come restrizione di `xsd:string` che limita la lunghezza massima della stringa a 32 caratteri.

La grandezza del file (che va espressa in byte) è registrata dell'elemento `<filesize>`. L'elemento è di tipo `xsd:Integer` (un numero positivo), è opzionale e non ripetibile. Anche l'elemento `<filesize>` è una raccomandazione NISO (Cfr. [Data Dictionary](#), p. 13).

11.3. Il formato dei file di testo born digital

Il formato dei file di testo (tipologia e modalità di compressione) è gestito dall'elemento `<format>` ed è codificato secondo lo standard NISO. Anche se lo standard NISO è stato concepito per la descrizione delle caratteristiche tecniche delle immagini digitali, lo schema `niso-mag.xsd` include anche specifiche tecniche relative a documenti **text-oriented** (riconoscibili grazie al prefisso `doc`). Per `<format>` è definito un tipo specializzato appartenente al namespace `niso` (p. 89) denominato `niso:docFormat`. Tale tipo è di tipo `xsd:sequence` e contiene tre elementi:

- **`<niso:name>`** : obbligatorio e non ripetibile, contiene il formato del file testo. È di tipo `xsd:string`, si consiglia di usare valori come `TXT` ecc. Si raccomanda di usare valori formati da tre caratteri.
- **`<niso:mime>`** : obbligatorio e non ripetibile, contiene il tipo mime del file di testo. Per l'elemento è definito un tipo semplice specializzato, anch'esso contenuto nel file `niso-mag.xsd`, denominato `niso:doc_mimetype`. Tale tipo è definito come restrizione di `xsd:string` ed è costituito dall'enumerazione dei seguenti valori:
 - `text/plain`
 - `text/xml`
 - `text/html`
 - `text/rtf`
 - `application/msword`
 - `application/pdf`
- **`<niso:compression>`** : obbligatorio e non ripetibile, dichiara il tipo di compressione applicato al file di testo. La compressione viene in genere applicata per migliorare l'archiviazione dei file di testo all'interno delle singole repository e per la fase di DIP. Per l'elemento è definito un tipo semplice specializzato, anch'esso contenuto nel file `niso-mag.xsd`, denominato `niso:doc_compressiontype`. Tale tipo è definito come restrizione di `xsd:string` ed è costituito dall'enumerazione dei seguenti valori:
 - `Uncompressed`
 - `ZIP`
 - `RAR`
 - `GZ`

11.4. La creazione del file

L'elemento `<datetimecreated>` registra la data e l'ora di creazione del file digitale. L'elemento è opzionale e non ripetibile; è di tipo `xsd:dateTime`, vale a dire che assume la forma `YYYY-MM-DDThh:mm:ss.mmm` di cui si vedano le specificazioni nella sezione GEN (p.

38).

Per esempio:

<datetimecreated>2005-04-13T02:01:52</datetimecreated>

12. Sezione DIS

La sezione DIS si usa nella fase di DIP (Dissemination Information Package) del modello OAI PMH per la disseminazione degli oggetti digitali e contiene informazioni circa la fruibilità dell'oggetto digitale.

Si usa per precisare condizioni particolari di accesso all'oggetto digitale o per annunciare il thumbnail di un'immagine. L'uso tipico della sezione è all'interno del protocollo OAI PMH nel quale il data provider comunica al service provider le condizioni per l'uso dell'oggetto digitale. L'uso della sezione non va confuso con quanto dichiarato dall'elemento `<usage>` che riguarda la gestione dei contenuti digitali nella repository interna; la sezione DIS fa invece riferimento all'esistenza di condizioni particolari (anche commerciali) di distribuzione all'esterno dell'istituzione.

La sezione DIS è costituita di un unico elemento `<dis>`, opzionale e non ripetibile. Il suo contenuto è di tipo `xsd:sequence`, e può contenere un solo elemento:

- `<dis_item>` : opzionale e ripetibile, descrive ogni singolo oggetto digitale distribuito sotto particolari condizioni dall'istituzione proprietaria dell'oggetto digitale (in altre parole la sezione non si usa se gli oggetti sono di libero uso). È di tipo `sequence` ma al suo interno si trova anche un `xsd:choice` e può contenere i seguenti elementi:
 - `<file>` : obbligatorio, localizza il file che contiene l'oggetto digitale da distribuire. È di tipo `link`, vale a dire che è un elemento vuoto che supporta attributi definiti dal namespace `xlink` (p. 33). L'elemento non è ripetibile
 - `<preview>` : contiene l'estratto del file multimediale disponibile per la disseminazione e si usa in alternativa a `<available>`. Per l'elemento è definito un tipo semplice specializzato denominato `preview`. Tale tipo è definito come restrizione di `xsd:string` ed è costituito dall'enumerazione dei seguenti valori:
 - `thumbnail` 'francobollo', applicabile nel caso di immagini.
 - `sample` 'riassunto' applicabile nel caso di altro tipo di oggetti digitale (tracce audio, stream video, documenti di testo).

Il tipo semplice `preview` che regola il contenuto dell'elemento `<preview>` è formalmente definito come segue (file `metatype.xsd`):

```
<xsd:simpleType name="preview">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="thumbnail"/>
    <xsd:enumeration value="sample"/>
  </xsd:restriction>
</xsd:simpleType>
```

L'elemento è alternativo all'elemento `<available>`: nel caso in cui l'elemento `<dis_item>` contenga l'elemento `<preview>` significa che l'oggetto è disponibile solo come anteprima.

- `<available>` : contiene un indirizzo di rete dove sono disponibili le condizioni di

accessibilità del file multimediale disponibile per la disseminazione; tali condizioni riguardano tipicamente diritti o prezzo. Di tipo `xsd:anyURI`, non è ripetibile e si usa in alternativa a `<preview>`: nel caso in cui l'elemento `<dis_item>` contenga l'elemento `<available>` significa che l'oggetto digitale è disponibile alle condizioni indicate all'indirizzo di rete precisato dall'elemento.

L'elemento `<dis_item>` è definito formalmente come segue (file `metatype.xsd`):

```
<xsd:complexType name="dis_item">
  <xsd:sequence>
    <xsd:element name="file" type="link"/>
    <xsd:choice minOccurs="0">
      <xsd:element name="preview" type="preview"/>
      <xsd:element name="available" type="xsd:anyURI"/>
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>
```

Si veda il seguente esempio d'uso della sezione DIS; si noti che il record presenta gli elementi qualificati, come è consigliabile fare nel caso di distribuzione del record al di fuori dell'istituzione:

```
<mag:metadigit version="2.0.1">
  <mag:gen creation="2004-04-09T14:18:15.000000"
    last_update="2004-10-28T11:00:30.000000">
    <mag:stprog>http://www.bncf.firenze.sbn.it/fondobertini.htm</mag:stprog>
    <mag:agency>IT:BNCF</mag:agency>
    <mag:access_rights>1</mag:access_rights>
    <mag:completeness>1</mag:completeness>
  </mag:gen>
  <mag:bib level="m">
    <dc:identifier>info:sbn/CFI0482250</dc:identifier>
    <dc:title>14 lithographies originales de Pablo Picasso</dc:title>
    <dc:creator>Picasso, Pablo</dc:creator>
    <dc:publisher>[S. l.: s. n.], Mourlot frères</dc:publisher>
    <dc:date>1954</dc:date>
    <dc:language>fre</dc:language>
  </mag:bib>
  <mag:img>
    <mag:sequence_number>1</mag:sequence_number>
    <mag:nomenclature>Frontespizio: r</mag:nomenclature>
    <mag:file Location="URL"
      xlink:href="http://bncf.firenze.sbn.it/Img?idr=BNCF0002970919"/>
    <mag:md5>02F8D88D2C3220AD1F0491F6E3B906F5</mag:md5>
    <mag:filesize>32248860</mag:filesize>
    <mag:image_dimensions>
      <niso:imagelength>3752</niso:imagelength>
```

```
<niso:imagewidth>2865</niso:imagewidth>
</mag:image_dimensions>
<mag:dpi>400</mag:dpi>
<mag:format>
  <niso:name>TIFF</niso:name>
  <niso:mime>image/tiff</niso:mime>
  <niso:compression>Uncompressed</niso:compression>
</mag:format>
<mag:altimg>
  <mag:file Location="URL"
xlink:href="http://bncf.firenze.sbn.it/Img?idr=BNCF0002970627"/>
  <mag:md5>f4a191e3dc4b952270ffc774eaebc99a</mag:md5>
  <mag:filesize>168581</mag:filesize>
  <mag:image_dimensions>
    <niso:imagelength>1876</niso:imagelength>
    <niso:imagewidth>1433</niso:imagewidth>
  </mag:image_dimensions>
  <mag:dpi>200</mag:dpi>
  <mag:format>
    <niso:name>JPEG</niso:name>
    <niso:mime>image/jpeg</niso:mime>
    <niso:compression>JPG</niso:compression>
  </mag:format>
</mag:altimg>
</mag:img>
<mag:dis>
  <mag:dis_item>
    <mag:file Location="URL"
xlink:href="http://bncf.firenze.sbn.it/Img?idr=BNCF0002970627"/>
    <mag:preview>sample</mag:preview>
  </mag:dis_item>
</mag:dis>
</mag:metadigit>
```

