NESTOR: A Formal Model for Digital Archives

Nicola Ferro and Gianmaria Silvello

Department of Information Engineering, University of Padua.
Via Gradenigo 6/B, 35131, Padua, Italy.

Abstract

Archives are an extremely valuable part of our cultural heritage since they represent the trace of the activities of a physical or juridical person in the course of their business. Despite their importance, the models and technologies that have been developed over the past two decades in the Digital Library (DL) field have not been specifically tailored to archives. This is especially true when it comes to formal and foundational frameworks, as the Streams, Structures, Spaces, Scenarios, Societies (5S) model is.

Therefore, we propose an innovative formal model, called NEsted SeTs for Object hieRarchies (NESTOR), for archives, explicitly built around the concepts of context and hierarchy which play a central role in the archival realm. NESTOR is composed of two set-based data models: the Nested Sets Model (NS-M) and the Inverse Nested Sets Model (INS-M) that express the hierarchical relationships between objects throughout the inclusion property between sets. We formally study the properties of these models and prove their equivalence with the notion of hierarchy entailed by the archives.

We then use NESTOR to extend the 5S model in order to take into account the specific features of the archives and to tailor the notion of digital library accordingly. This offers the possibility of opening up the full wealth of DL methods and technologies to archives. We demonstrate the impact of NESTOR on this problem through three example use cases.

Keywords:
foundation, digital archive, digital library, hierarchy, set-based model, application, 5S model, OAI-PMH, OAI-ORE, linked data, annotation, libraries, archives and museums (LAM)

1. Introduction

Over the past two decades, digital libraries have been steadily evolving and have been shaping the way in which people and institutions access and interact with our cultural heritage, study, and learn [17, 18, 43–45, 69, 73, 106]. Nowadays, their reach goes far beyond the realm of traditional libraries and also encompasses other kinds of cultural heritage institutions, such as archives and museums. Nevertheless, these institutions are quite different from several points-of-view: they have different internal organizations and traditions; their resources are different in nature, structure, and descriptions; and their users have different information needs which call for different access methods to resources.

Archives are not simply constituted by a series of objects that have been accumulated and filed with the passing of time – as usually happens with libraries that collect, for example, individual published books, journals, and serials. Instead, archives represent the trace of the activities

of a physical or juridical person in the course of their business which is preserved because of their continued value.

To this end, archives keep the *context* in which their records have been created and the network of relationships between them in order to preserve their informative content and provide understandable and useful information over time [47]. The fundamental characteristic of archives resides in their *hierarchical organization*. This expresses the *context* – i.e. the relationships and dependencies between the records of the archive – by using what is called the *archival bond* and it distinguishes archives from other objects in the realm of cultural heritage – e.g. books – which in general are perceived as individual, repeatable and unrelated entities [104]. Archives are in fact made up of series which, in turn, can be organized in sub-series formed of archival units, such as files, registers and so on. These archival units have a homogeneous nature and can, in turn, be divided into subunits containing items such as letters, reports, contracts, testaments, photographs, drawings and so on [57].

Digital libraries benefit from the existence of sophisticated formal models, such as the Streams, Structures, Spaces, Scenarios, Societies (5S) model [44, 48, 49], which allow us to formally describe them and to prove their properties and features. Despite the importance of archives, so far there has been no attempt to develop a dedicated formal model, built around their peculiar constituents, such as the notion of *archival bond*. Nor can we exploit the 5S model as it is for archives because, as we will discuss later on, it needs some kind of extension and tailoring.

In this article we highlight the central role of formal models for the digital library, because integration and cooperation between these models can turn into a factual interoperability between the different facets of DL, including their community, methodology and technology. In this context a model for archives is sorely needed to formally define their characteristics and to prove that general digital library methods and technologies can be embodied in this field and respect archival practice.

Therefore, we propose an innovative formal model for archives built around the notion of *archival bond* and *hierarchy*. The proposed model, called NEsted SeTs for Object hieRarchies (NESTOR), is based on the idea of expressing the hierarchical relationships between objects through the inclusion property between sets, in place of the binary relation between nodes exploited by the tree [14].

Then we exploit NESTOR to formally extend the 5S model to define a *digital archive* as a specific case of digital library able to take into consideration the peculiar features of archives. This defines an actual bridge between these two formal models which: (i) allows archives to exist and interact with other realities (i.e. libraries and museums); (ii) provides archives the possibility of exploiting the full wealth of digital library technologies and methods; and, (iii) enables integrated access to heterogeneous contents.

As concrete accounts of this and as substantial examples of their application, we apply NESTOR and the extended 5S model to three typical scenarios for digital archives and overcome well-known issues in the field. The first is called "detaching the archives" which is the case of interoperability between digital archives where we formally exploit the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) to give a concrete account of how digital library technologies can be adopted with archives. The second scenario is called "unchaining the archives" which shows how the archives modeled with NESTOR can form compound digital objects made available as Linked Open Data (LOD) [55] on the Web adopting Open Archives Initiative Object Reuse and Exchange (OAI-ORE) as a working framework. Finally, the third scenario is called "socializing the archives" which describes how NESTOR together with the Flexible Annotation Semantic Tool (FAST) [7] can enhance the role of annotations in the archives by helping

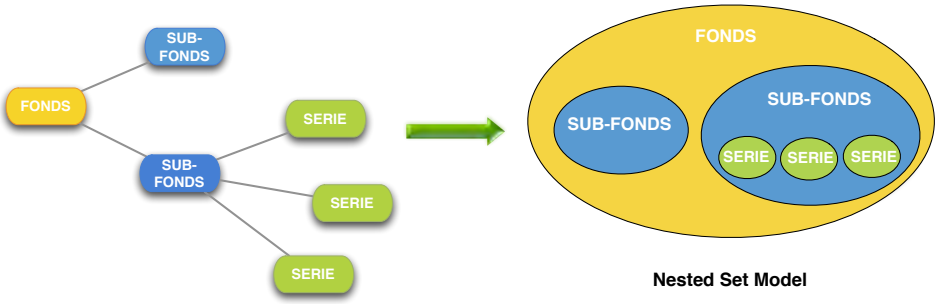


Figure 1: An archive modeled by means of the NS-M.

both archivists and end-users in the description and interpretation of archival resources.

The paper is organized as follows. Section 2 provides an intuitive overview of the principles underlying the two set data models composing NESTOR (i.e. the NS-M and INS-M) and a presentation of the main contributions of this work. In Section 3 we provide some background on archives, formal models for digital libraries and discuss the related work about nested sets methodologies. In Section 4 we formally present NESTOR along with its properties. Section 5 shows the equivalence between NESTOR and the archival trees. In Section 6 we introduce our extension to the 5S model via NESTOR and in Sections 7-9 we apply NESTOR and this extension to three case studies. We draw conclusions and point to future work in Section 10. In Appendix A we report all the proofs of the properties and theorems presented in Sections 4 and 5.

2. NESTOR: Overview and Contributions

2.1. Intuitive Overview of the Model

The set data models composing NESTOR are well-suited for archival practice; indeed, the idea of "set" shapes the concept of archival division which is a "container" comprising distinct elements that have some properties in common. If we consider the Chinese boxes metaphor, a hierarchy is composed of a sequence of boxes contained one inside the other; if we look at an archive from the physical point-of-view, we can see that it resembles the Chinese boxes structure as there are boxes, folders, sheets, etc. contained one inside the other.

Nested sets are closer to this view of reality than trees are. Indeed, although archival practice commonly considers archives as trees, a tree is actually a higher level abstraction than the nested sets as it only focuses on structural relationships. Indeed, NESTOR comprises both the structure and the content of the archive, where the inclusion relationships represent the structure and the elements belonging to the sets represent the content.

To illustrate the basic ideas behind NESTOR, let us consider an archive composed of six divisions: a fonds, two sub-fonds, and three series.

As shown in Figure 1, the first model composing NESTOR – i.e. the Nested Sets Model (NS-M) – adopts a bottom-up approach: (i) each set corresponds to an archival division; (ii) the innermost sets are the leaves of the hierarchy, e.g. the series; (iii) you create supersets as you climb up the hierarchy, e.g. the sub-fonds and fonds. In general, in Figure 1 we can see that each node of the archival tree is mapped into a set, where child nodes become *proper subsets* of the

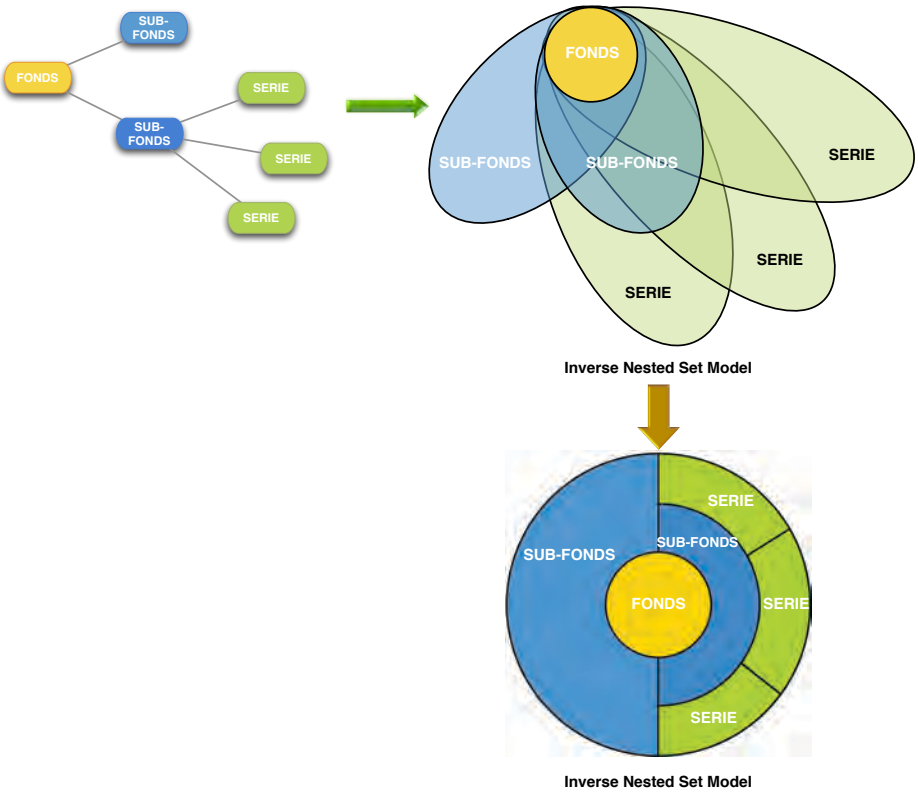


Figure 2: An archive modeled by means of the INS-M.

set created from the parent node. Every set is a subset of at least one set; the set corresponding to the tree root is the only set without any supersets and every set in the hierarchy is a subset of the root set. The leaves are sets with no subsets. The tree structure is maintained thanks to the nested organization and the relationships between the sets are expressed by the set inclusion order. Even the disjunction between two sets brings information; indeed, the disjunction of two sets means that these belong to two different branches of the same archival hierarchy.

As shown in Figure 2, the second model composing NESTOR – i.e. Inverse Nested Sets Model (INS-M) – adopts a top-down approach: (i) each set corresponds to an archival division; (ii) the innermost set is the root of the hierarchy, i.e. the fonds; (iii) you create supersets as you climb down the hierarchy, e.g. sub-fonds and then series. We can say that a tree is mapped into the INS-M by transforming each node into a set, where each parent node becomes a subset of the sets created from its children. The set created from the tree's root is the only set with no subsets and the root set is a proper subset of all the sets in the hierarchy. The leaves are the sets with no supersets and they are sets containing all the sets created from the nodes composing the tree path from a leaf to the root. An important aspect of INS-M is that the intersection of every couple of sets obtained from two nodes is always a set representing a node in the tree. The intersection of all the sets in the INS-M is the set mapped from the root of the tree.

Unfortunately, the representation of the INS-M by means of the Euler-Venn diagrams (adopted for the NS-M) is not very expressive and can be confusing for the reader [10] – see Figure 2. Nevertheless, we can exploit the "*DocBall representation*" [26] – see bottom of Figure 2 – which is composed of a set of circular sectors arranged in concentric rings. In the context of NESTOR a circular ring has to be seen as a set containing objects, where the outer rings are supersets of the inner rings. Each ring represents a level of the hierarchy with the center (level 0) representing the root. In a ring, the circular sectors represent the nodes in the corresponding level. Therefore, the fonds is represented by the inner ring at level 0 of the DocBall. At level 1 we find the direct supersets of the fonds which are the sub-fonds; both these sets are represented as circular sectors comprising the inner circle. With this representation a subset is presented in a ring within the set including it. Indeed, we can see that the fonds is included by all the other sets. If the intersection of two or more sets is empty, then these sets have no common circular sector in the inner rings of the DocBall.

From this description we can see that the INS-M can be associated to the top-down descriptive activity and the NS-M to the bottom-up one. The top-down descriptive activity is followed by the archivist when s/he has to describe an archive for which s/he knows the structure in advance. For instance, the archivist knows that there is a fonds divided into three sub-fonds and so on and so forth; in this case the activity is to describe these archival divisions and the documents they contain. We call this top-down because in this case the archivist knows *a priori* how to divide the documents (i.e. elements) into the archival divisions (i.e. sets). The bottom-up description activity works the other way around; the archivist starts to study the documents and s/he decides how to put them together in order to form an archival division, thus the archival hierarchy is built from the bottom. We call it bottom-up approach because in this case dividing the documents into archival divisions is an iterative process: the archivist starts from the whole set of documents (i.e. the fonds) and s/he defines the subsets (i.e. subfonds, series, etc.) by construction, analysing the documents one by one.

2.2. Contributions to the field of Digital Libraries

In the context of Libraries, Archives, and Museums (LAM) unifying a variety of organizational settings and provide more integrated access to their contents is an aspect of utmost importance. Indeed, LAM collect, manage and share digital contents; although the type of materials may differ, and professional practices vary, LAM share an overlapping set of functions. Fulfilling these functions in "collaboration rather than isolation creates a win-win for users and institutions" [109]. The convergence between libraries, archives and museums has been a topic of much discussion in the digital library community, but the emerging similarities between these three types of cultural heritage institutions are not yet evident in the proposed formal models, developed systems, and education of professionals [98, 99].

In particular, there are no state-of-the-art formal models for archives and this has prevented them from being fully integrated in digital library communities, methodologies and technologies. The definition of the set data models and their properties we give in Section 4 proves that the nested sets idea can be formalized as a proper data model that can be exploited to represent and manage archival hierarchies. Indeed, we show that it is possible to represent a hierarchical organization by means of the sets and then represent the objects belonging to the sets and formally establish relations between them.

The formalization of NESTOR settles a common ground for dealing with hierarchies open to existing models, solutions and technologies; it exploits and enhances the state of the art in the

fields of digital library, thus providing a further level of expressiveness and a theoretical environment that can be exploited for the definition of innovative systems, functionalities and services. Furthermore, as will also emerge later on, the nested sets models have several advantages over trees while remaining semantically equivalent, and even though they are well known in the field from an intuitive point-of-view, they have not been formalized before.

We exploit the formal basis provided by NESTOR to extend the widely-known 5S Model [49] in order to explicitly enclose the archives and their constraints in the reality it intends to model. Afterwards, we exploit a main feature of NESTOR which is the separation between the structural and the content aspects of the entities represented within the set data models to address concrete issues in the field of digital libraries. Specifically, in the field of archives the formalization of NESTOR allows us to address some known problems and, at the same time, to push the boundaries of the discipline. To this purpose we present three use cases. The first is called *"detaching the archives"*, the goal of which is to allow variable granularity sharing of archival metadata in a distributed environment; the aim is to free and exchange a specific archival description (or a set of descriptions) independently from the whole archive, since in any moment the context of this description can be reconstructed. This use case shows how NESTOR allows for addressing known problems regarding the state-of-the-art of digital archives; we consider the issues regarding interoperability between digital archives and metadata exchange. The 5S model has been used for modeling the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) [103] that is the *de-facto* standard for metadata sharing between digital libraries in distributed environments. The formal definition of NESTOR allows us to exploit the theoretical common ground with the 5S model to extend the OAI-PMH and allow it to manage and exchange complex hierarchical data structure in a flexible way, thus overcoming a well-known problem affecting the current archival description standard. The extension of OAI-PMH will make possible the exchange of data belonging to a hierarchy with a variable granularity without losing the relationships between the other data in the hierarchy.

The second use case is called *"unchaining the archives"*, the goal of which is to open up archival data in the Web by exploiting the potentialities of the Linked Open Data (LOD) [55] paradigm and to enrich the archival descriptions with related digital objects. This use case shows how NESTOR helps to push the boundaries of the discipline by creating new possibilities for archives. Indeed, the reality of modern archival records creation is that documents may exist in "multiple contexts and have multiple and complex relationships that describe their significance and value" [62]. Furthermore, new archival trends encourage the adoption of a "plural, provisional and interpretative perspective" [71] in the description of the archives. This vision leads to the creation of multiple connected hierarchies of entities that must respect the archival rules and NESTOR along with its relationships with the 5S model addresses this aspect in a formal way with tangible outcomes. Furthermore, archival practice is experiencing a transformation process which promotes the definition of complex relationships between the resources of interest and the constitution of compound digital objects [62]. For similar reasons, in the wider context of digital libraries we are experiencing a wide-ranging diffusion of the Open Archives Initiative Object Reuse and Exchange (OAI-ORE)[1].

Archives as a meaningful part of digital libraries can take advantage of using the LOD approach instantiated by means of OAI-ORE [62]; indeed, a methodology for representing archives in OAI-ORE would allow richer methods for modeling archival descriptions and can also provide additional and flexible visualizations of the documents that would not be restricted to

[1] `http://www.openarchives.org/ore/`

the "old linear view inspired by the paper tradition" [62]. At the same time, it is commonly agreed [62, 71, 88] that new approaches, such as the adoption of the OAI-ORE model, should add to, but not undermine, the fundamental archival theory.

The formal basis we define allows us to model an archive as an OAI-ORE instance while retaining its hierarchical structure and the archival bond [39], and to propose a methodology to map archival descriptions into OAI-ORE showing how it enables both the preservation of their original order and the definition of new types of relationships.

The third use case is called *"socializing the archives"*, the goal of which is to assist archivists and general users in enriching, consulting, and understanding archives by means of annotations. This use case shows how NESTOR allows us to transform the archives into a new type of information infrastructure that can be user-centered and is able to support content management tasks together with tasks devoted to communication and cooperation [60]. The main way of reaching this goal is to support the archivists by considering the way in which they work [81, 93] and, as a consequence, by enriching the archives through digital annotations. Indeed, annotations foster collaboration between archivists, researchers and general users by playing a central role both in the phase of *creation* and in the phase of *consultation* of archival metadata. In the creation phase archivists have to select and describe the archival material and annotations allow them to explain and discuss their choices, thus enabling users to properly access and consult the archival metadata. In the consultation phase, annotations are exploited to find relationships between different parts of an archive or between different archives; for instance, users can exploit annotations to move from one archive to another guided by the expertise of the archivists that annotated them.

The archival community has developed "content and data structure standards" [86] to facilitate the description, management and access to the archival resources; however, these standards can be difficult for archivists to use [27] and are often implemented in ways that can negatively affect their description activity [108]. Therefore, there has been a proliferation of digital archival systems based on diversified descriptive methodologies and metadata; also from the annotation point-of-view a lot of research has been done that has led to the design and development of variegated annotation systems [7].

This heterogeneity turns into an interoperability problem when we need to access and consult archival metadata managed by different digital archive systems and annotations created and handled by different systems. Moreover, annotations under certain conditions as well as archives can be opportunely organized in a hierarchical way. The 5S model extended through NESTOR allows for the formal modeling and managing of multiple hierarchies which are exploited to create a common basis between the archives through the NESTOR model and the annotations through the FAST formal model [7, 38].

3. Related Work

3.1. State-of-the-art of the Archives

Archival description is defined in [80] as "*the process of analyzing, organizing, and recording details about the formal elements of a record or collection of records, to facilitate the work's identification, management, and understanding*"; archival descriptions have to reflect the peculiarities of the archive, retain all the informative power of a record, and keep trace of the provenance and original order in which resources have been collected and filed by archival in-

stitutions [47]. This is emphasized by the central concept of *fonds*[2], which should be viewed primarily as an "intellectual construct", the conceptual "whole" that reflects an organic process in which a records creator produces or accumulates series of records [25]. In this context, provenance becomes a fundamental principle of archives often referred to as "*respect des fonds*" which dictates that resources of different origins be kept separate to preserve their context [29, 47].

[29] highlights that maintaining provenance leads archivists to evaluate records on the basis of the importance of the creator's mandate and functions, and fosters the use of a hierarchical method. The hierarchical structure of the archive expresses the relationships and dependency links between the records of the archive by using what is called the archival bond defined as "*the interrelationships between a record and other records resulting from the same activity*" [80]. Archival bonds, and thus relationships, are constitutive parts of an archival record: if a record is taken out from its context and has lost its relationships, its informative power would also be considerably affected. Therefore, archival descriptions need to be able to express and maintain such structure and relationships in order to preserve the context of a record.

Archival description proceeds from the general to the specific as a consequence of the provenance principle and has to show, for every unit of description, its relationships and links with other units and to the general fonds. Therefore, archival descriptions produced according to the International Standard for Archival Description (General) (ISAD(G)) [57] take the form of a tree. In Figure 3 we can see the ISAD(G) hierarchical model: any number of intermediate levels are possible between any shown in the model. Entities are in a vertical relationship of subordination with the entity they belong to; the hierarchical representation is further complicated by the fact that the entities which belong to the same father have a "horizontal-type" relationship – they need to be represented according to a significant sequence which reflects the position that they have in the logical and/or the material order of the archive.

The principles of ISAD(G) are put into action by the Encoded Archival Description (EAD) standard [82, 95] for encoding archival descriptions. EAD is based on eXtensible Markup Language (XML) [105] and it succeeded because "*for the first time archivists have been offered a data structure standard that accommodates a hierarchical structure for the presentation of a variety of descriptions*" [53] and it enables archivists to be software independent.

EAD is composed of three high-level components: `<eadheader>`, `<frontmatter>`, and `<archdesc>`. The `<eadheader>` contains metadata about the archive descriptions and includes information about them such as title, author and date of creation. The `<frontmatter>` supplies publishing information and is an optional element, while the `<archdesc>` contains the archival description itself and constitutes the core of EAD. The `<archdesc>` may include many high-level sub-elements, most of which are repeatable. The most important element is the `<did>` or descriptive identification which describes the collection as a whole. The `<did>` element is composed of numerous sub-elements intended for brief, clearly designated statements of information and they are available at every level of description. Finally, the `<archdesc>` contains an element that facilitates a detailed analysis of the components of a fonds, the `<dsc>` or description subordinate components. The `<dsc>` contains a repeatable recursive element, called `<c>` or component. A component may be an easily recognizable archival entity such as series, subseries or items. Components not only are nested under the `<archdesc>` element, they are also usually nested inside one another. Components usually are indicated with `<cN>` tag, where $N \in \{01, 02, \ldots, 12\}$.

[2]The term *fonds* is not a commonly used English word. It is derived from the French [54] and in the archival context it is used both for the singular and plural form of the noun.

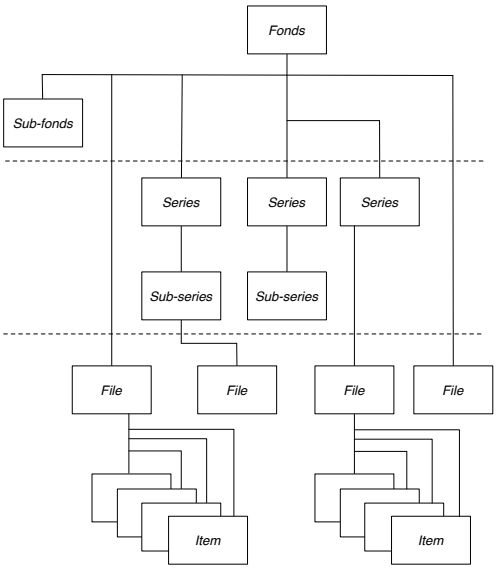


Figure 3: Hierarchical organization of the archives and of the archival descriptions according to ISAD(G) [57].

EAD reflects the archival structure and holds relationships between entities in an archive. In addition, EAD encourages archivists to use collective and multilevel description, and because of its flexible structure and broad applicability, it has been embraced by many repositories [63].

On the other hand, EAD allows for several degrees of freedom in tagging practice, which may turn out to be problematic in the automatic processing of EAD files, since it is difficult to know in advance how an institution will use the hierarchical elements. The EAD permissive data model may undermine the very interoperability it is intended to foster. Indeed, it has been underlined that only EAD files meeting stringent best practice guidelines are shareable and searchable [86]. Moreover, there is also a second relevant problem related to the level of material that is being described. Unfortunately, the EAD schema rarely requires a standardized description of the level of the materials being described, since the `<level>` attribute is required only in the `<archdesc>` tag, while it is optional in `<cN>` components and in very few EAD files this possibility is used, as pointed out by [83]. As a consequence, the level of description of the lower components in the hierarchy needs to be inferred by navigating the upper components, maybe up to the `<archdesc>`, where the presence of the `<level>` attribute is mandatory. Therefore, access to individual items might be difficult without taking into consideration the whole hierarchy.

We highlight this fact in Figure 4 where we present the structure of an EAD file. In this example we can see the top-level components `<eadheader>` and `<archdesc>` and the hierarchical part represented by the `<dsc>` component; the `<level>` attribute is specified only in the `<archdesc>` component. Therefore, the archival levels described by the components of the `<dsc>` can be inferred only by navigating the whole hierarchy. Moreover, sharing and searching archival description might be made difficult by the typical size of EAD files with a very deep hierarchical structure. Indeed, each EAD file is a description of a whole collection of items rather than the description of an individual item. On the other hand, users are often interested in the information described at the item level, which is typically buried very deeply in the hierarchy

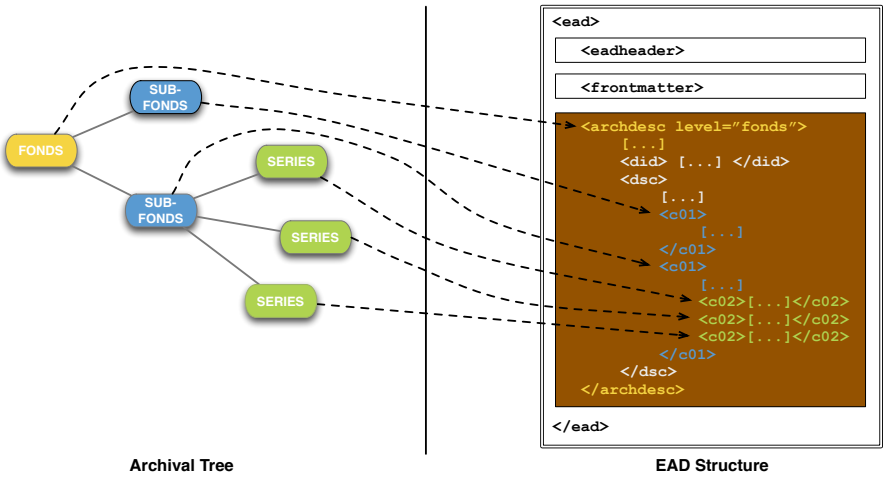


Figure 4: EAD representation of an archive.

and might be difficult to reach [92].

EAD presents some difficulties both for the expert user (i.e. the archivists who find the "complexity of EAD itself to be a deterrent to implementation" [108]) and the general user who has to consult and interpret the archival data without specific knowledge of archival theory and practice. One difficulty is related to the reconstruction of the archival context starting from an element buried in the hierarchy; this difficulty related to the data/system model on which EAD is based may be reflected in a similar difficulty and disorientation for the user in the perception of the context which supply the information needed to satisfy the her/his information requirements. Another concern is that in some cases EAD makes searches more complicated for users [108].

These problems are also enhanced by the lack of a systematic user study about the perception and the usefulness of EAD for the end-user. Note that in the recent past few institutions have developed formal evaluations for monitoring the effectiveness of EAD. Archivists are basing their perceptions regarding end-user utilization of EAD on very little quantitative or systematic qualitative data [89], so it is not easy to measure the end-users level of engagement [79] with the archival data. One of the goals of NESTOR is to provide a flexible model to handle archival data in order to facilitate the interpretation, utilization, sharing and also visualizations of archival resources; the importance of these aspects are assessed by several studies about the functionality and the usability of electronic resources [72] and we take them into account in the use cases presented in Sections 7 and 9.

When we need to relate one or more digital objects to their archival descriptions represented as metadata, EAD introduces some more limitations. Indeed, each `<cN>` tag of the EAD may contain a description of a digital object or a bunch of digital objects. These objects are usually reachable by means of a Uniform Resource Identifier (URI); the link from EAD to a digital object or group of objects can be made at any level, but "*it should be made at the level where the object(s) is described or implied in EAD*" [77]. To this end EAD provides a `<dao>` tag which allows us to specify a URI to an external digital object which is part of the described material (see Figure 5a); furthermore, EAD also provides an `<extptr>` element to point to a digital object that is not part of the described materials [77]. By means of these tags we can link one external

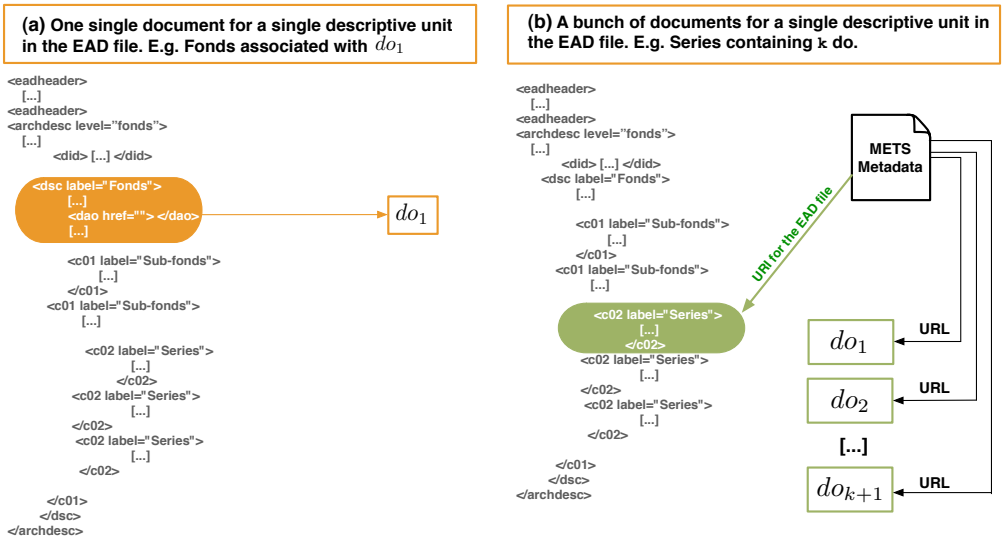


Figure 5: The common solution to link the EAD file with the described digital objects.

digital object to each archival division, but we cannot link more than one digital object to a specific division. The current solution to this problem exploits third-party components – i.e. the so-called "digital wrappers"[3]; a relevant example is the Metadata Encoding and Transmission Standard (METS) metadata that is used as an in-between component for relating a bunch of digital objects to an EAD component [46, 97] – see Figure 5b. NESTOR in conjunction with the LOD paradigm enables the definition of a more flexible solution to the problem. This solution, presented in Section 8 in the "detaching the archives" use-case, exploits the clear distinction between structure and content enabled by NESTOR to instantiate an archive as an OAI-ORE instance which exposes archives as compound digital objects in the Web.

3.2. State-of-the-art of Digital Library Models

In order to settle a theoretical common ground where it is possible to establish relationships between NESTOR and the different models proposed in the field of digital libraries, we describe and discuss the following state-of-the-art models: (i) the 5S formal model, (ii) the DELOS reference model, and (iii) the Europeana Data Model (EDM). These models are different in the scope they pursue and in the way in which they are defined, but they all aim at providing a means to model data, services, or applications in the digital libraries realm.

The Streams, Structures, Spaces, Scenarios, Societies (5S) [44, 48, 49] is a formal model and draws upon the broad digital library literature in order to have a comprehensive base of support. It was developed largely bottom up, starting with key definitions and with elucidation of digital library concepts from a minimalist approach. It is built around five main concepts: (i) *streams* are sequences of elements of an arbitrary type, e.g. bits, character, images, and so on; (ii)

[3]Digital wrappers "*are pieces of software for binding digital content files and their metadata together and for specifying the logical relationships among the content files*" [46].

structures specify the way in which parts of a whole are arranged or organized, e.g. hypertexts, taxonomies, and so on; (iii) *spaces* are sets of objects together with operations on those objects that obey certain constraints, e.g. vector spaces, probabilistic spaces, and so on; (iv) *scenarios* are sequences of related transition events, for instance, a story that describes possible ways to use a system to accomplish some functions that user desires; and, (v) *societies* are sets of entities and relationships between them, e.g. humans, hardware and software components, and so on.

Starting from these five main concepts, it provides a definition for a minimal digital library which is constituted by: (i) a repository of digital objects; (ii) a set of metadata catalogs containing metadata specifications for those digital objects; (iii) a set of services containing at least services for indexing, searching, and browsing; and, (iv) a society.

While these broad concepts can be also in common with archives, when you look at the specific way in which they are formally defined, you realize that the definitions cannot be straightforwardly applied to the archives case without at least some extension. We will discuss this in further detail with the presentation of an extension of 5S via NESTOR in Section 6.

The DELOS Reference Model [20] is a high-level conceptual framework that aims at capturing significant entities and their relationships with the digital library universe with the goal of developing more concrete models of it. The DELOS Reference Model and the 5S model address a similar problem with different approaches; the former does not provide formal definitions, but it provides a way to model and manage the resources of the digital library realm. The 5S on the other hand is a formal model providing mathematical definitions of the digital library entities that can be used to prove properties, theorems and propositions like in [48, 50].

So the DELOS Reference Model is similar to the 5S model in its broader goal but instead of using a mathematical formalism, it relies on concept maps [75, 76] because of their simplicity and immediacy and it highlights six main domains in the digital library universe: (i) *content*: the data and information that digital libraries handle and make available to their users; (ii) *user*: the actors (whether human or not) entitled to interact with digital libraries; (iii) *functionality*: the services that digital libraries offer to their users; (iv) *quality*: the parameters that can be used to characterize and evaluate the content and behaviour of digital libraries; (v) *policy*: a set of rules that govern the interaction between users and digital libraries; and (vi) architecture: a mapping of the functionality and content offered by a digital library onto hardware and software components.

These six main domains represent the high level containers that help organize the DELOS Reference Model. For each of these domains, the fundamental entities and their relationships are clearly defined. Even if the 5S model and the DELOS Reference Model are at two different levels of abstractions and make use of different languages and formalisms to represent the digital library universe, it is possible to make bridges and mappings between the two, as for example has been done for the quality domain [8].

It is possible to express the high-level entities and the relationships grasped by NESTOR throughout the concepts defined in the DELOS Reference Model with no or little extension to the model, but it would be very difficult to express in the DELOS Reference Model the constraints that are present in NESTOR. Moreover, the DELOS Reference Model is not a formal model and thus it would not be possible to formally prove the properties of the modeled reality of interest.

At a different level and without the ambition of modeling the whole digital library universe, we can consider the EDM [28, 30, 31] which aims at structuring the data managed by Europeana[4], a major effort of the European Union to create a digital library containing the cultural

[4] `http://www.europeana.eu/`

heritage of Europe. EDM adheres to the modeling principles that underpin the approach of the Web of Data ("Semantic Web") [16, 55]. A common model like EDM can instead be seen as an anchor to which various finer-grained models can be attached, making them at least partly interoperable at the semantic level, while the data retain their original expressivity and richness. It is thus possible to convert EAD concepts to and represent them in EDM [22, 56]. The same holds true also in the case of NESTOR [39], passing through OAI-ORE [66], with the additional benefit of exploiting the formal model to precisely define these mappings, constraining them, and proving their properties ahead as we show in Section 8.

NESTOR along with all the models presented here can be employed by general-purpose digital library architectures, such as Greenstone [107], and Fedora Commons[5], in order to provide support for modeling data and resources and to enhance their applications and services. These initiatives aim at providing a common architectural and software platform that can be exploited to build a digital library; therefore, they address a different set of problems from NESTOR, the 5S model, the DELOS reference model, and EDM.

3.3. State-of-the-art on Nested Sets

The intuitive idea of nested sets was proposed by Knuth in [64] without any formal definition and it has been mainly exploited in the field of relational databases as an alternative approach for implementing some integer encodings to efficiently solve recursive queries in Structured Query Language (SQL) [23, 59, 74, 100].

In Figure 6 we report the original representation of nested sets proposed by Knuth. Figure 6a represents an instance of the general idea of nested sets: "*A collection of sets in which any pair of sets is either disjoint or one contains the other*" [64]. Figure 6b represents a linear nested sets view. Matching parentheses can be seen as delimiting a set, contained in the sets delimited by more external matching parentheses. The parent-child relationships are retained by the nesting inside the parentheses. The representation of the tree in Figure 6c works in the same way by exploiting the idea of *indentation*.

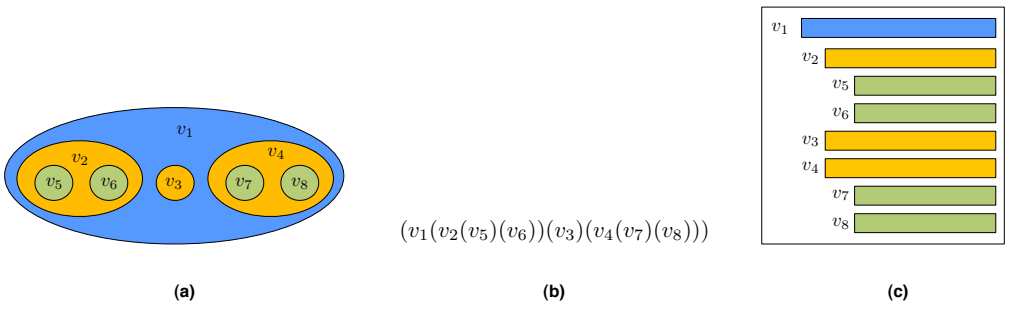


Figure 6: Previous alternative graphical representations of the tree proposed by Knuth [64]: (a) nested sets; (b) nested parentheses; (c) indentation.

We exploited this idea in the field of digital libraries by proposing some applications to the realm of the archives; indeed, in [34] the idea of using a nested organization of subsets has been exploited to allow the exchange of archival metadata between distributed digital libraries.

[5] `http://fedora-commons.org/`

An initial formulation of the first model composing NESTOR which is the Nested Sets Model (NS-M) was presented in [35] and then it was improved in [11]. In [35] the second model called Inverse Nested Sets Model (INS-M) was introduced and applied to represent, manage and exchange archival data between distributed digital libraries. However, in this paper, we have completely reworked the formal definitions of NESTOR and propose a brand new formalism which also allows us to better express the properties of the model. This work has been reviewed and extended in [12] where a mapping between the two newly defined set data models is proposed along with a preliminary definition of an algebra to operate on NESTOR (this aspect is not discussed in this article). In [37] the INS-M was exploited to define an algorithm to find the lowest common ancestor between two objects in a hierarchy. To the best of our knowledge, the INS-M has not been addressed before in the literature and both models, NS-M and INS-M, are defined here from a formal point of view.

4. NESTOR: The Formal Model

NESTOR defines two set-based data models: The Nested Sets Model (NS-M) and the Inverse Nested Sets Model (INS-M). They are both formally defined in the context of set theory [52, 58]. We present the NS-M and then the INS-M. We will maintain this order in the whole presentation of NESTOR. We define both NS-M and INS-M as a collection of subsets where specific conditions must hold. Note that for the sake of readibility all the proofs are gathered and reported in Appendix A.

The first definition regards the NS-M; basically, we define a collection of subsets (i.e. C) of a set (i.e. A) and then we impose some constraints on the subsets of A (i.e. $H, K \subset A$) which belongs to C. NS-M is defined as a Nested Sets Collection (NS-C) which is a collection of subsets where two conditions must hold. In the following definition, the first condition (4.1) states that set A which contains all the subsets of the collection must belong to the NS-C itself. The second condition states the intersection of every couple of sets in the NS-C is not the empty-set only if one set is a proper subset of the other one.

Definition 1. *Let A be a set and let C be a collection of subsets of A. Then C is a* ***Nested Sets Collection*** *(NS-C) if:*

$$A \in C, \tag{4.1}$$

$$\forall H, K \in C \mid H \cap K \neq \emptyset \Rightarrow H \subseteq K \vee K \subseteq H. \tag{4.2}$$

This definition formally defines how an archive can be modeled by means of the NS-M as shown in Figure 1. The collection of subsets C is the considered archive; the first condition says that there is a set – i.e. the "fonds" – which contains all the subsets – i.e. "subfonds", "series", etc. – of the archive. The second condition says that two subsets such as two "series" cannot have common elements, thus their intersection is always empty.

Now, we can introduce the Inverse Nested Sets Collection (INS-C) which defines the INS-M. We define an INS-C as a collection of subsets where two conditions must hold. The first condition (4.3) states that C must contain the *bottom set* (i.e. the common subset of all the sets in C), call it B, which is the common subset of all the sets in C. The second condition (4.4) states that if we consider three sets K, H and L in C such that H is a subset of K and K is not equal to L, then the intersection between L and K is not the same as the intersection between H and L or H is not a subset of L and vice versa.

Definition 2. *Let A be a set and let C be a collection. Then, C is an* ***Inverse Nested Sets Collection*** *(INS-C) if:*

$$\exists! B \in C \mid \forall K \in C, B \subseteq K, \tag{4.3}$$

$$\forall H, K, L \in C \mid H \subseteq K, L \neq K \Rightarrow (L \cap K = H \cap L) \vee (H \subseteq L) \vee (L \subseteq H). \tag{4.4}$$

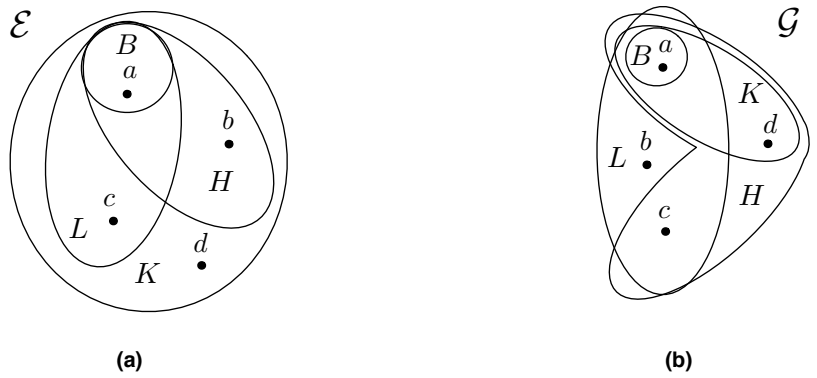


Figure 7: Collections of subsets which are not Inverse Nested Sets Collections (INS-C). In (a) the set K violates condition 4.3 of Definition 2 and in (b) sets K and H violate condition 4.4.

This definition can be further explained by taking into account the example from Figure 7(a); let us consider the collection $\mathcal{E} = \{B, H, K, L\}$ represented on the left hand side of the figure, where $B = \{a\}, H = \{a, b\}$, $K = \{a, b, c, d\}$ and $L = \{a, c\}$. In this case, $H \subseteq K$, $L \neq K$ and $H \nsubseteq L \wedge L \nsubseteq H$ but $L \cap K = \{a, c\} \neq H \cap L = \{a\}$; therefore, the collection represented in Figure 7 is not an INS-C. If we consider the collection of subsets $\mathcal{G} = \{B, H, K, L\}$ represented in Figure 7(b), where $B = \{a\}, H = \{a, c, d\}$, $K = \{a, d\}$ and $L = \{a, b, c\}$, we can see that $K \subseteq H$ and that $\exists L \in \mathcal{G} \mid L \nsubseteq K \wedge K \nsubseteq L$ but $L \cap K = \{a\} \neq H \cap L = \{a, c\}$, thus $\mathcal{G}$ is not an INS-C.

This definition formally defines how an archive can be modeled by means of the INS-M as shown in Figure 2. If the collection C is the archive we intend to model, the first condition says that there must exist an archival division which all other divisions share; this means that the "fonds" must be the archival division common to all the other divisions in the archive. Basically, this is another way to say that all the archival divisions are dependant on the same "fonds". The second condition extends this fact by saying that if two or more archival divisions, say "series", belong to the same archival branch, then they must have in common the same "subfonds" and "fonds".

4.1. NESTOR Separation between Structure and Content

In the context of information access systems it is important to separate between intensional and extensional aspects of information; for instance, in relational database management systems there exists the distinction between metadata (i.e. intensional level) and data (i.e. extensional level).

In NESTOR it is possible to delineate a clear distinction between intension and extension of a collection of sets and thus, between structure and content; indeed, from the structural point-of-view, a collection of subsets is represented by the sets in the collection and their inclusion

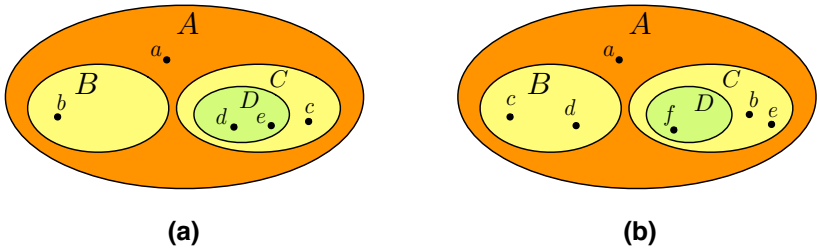


Figure 8: Two valid instances of the NS-C used in Example 1.

dependencies. A collection of subsets at the intensional level is defined by its structure. Let us consider an example based on the NS-M knowing that these considerations are also valid for the INS-M. We can say that $C = \{A, B, C\}$ where $B \subseteq A$, $C \subseteq A$ and $B \nsubseteq C \wedge C \nsubseteq B$ is a NS-C because it respects conditions 4.1 and 4.2 of Definition 1. In this way, we know the structure of the collection and we know which relationships hold between the sets. From the archival point-of-view, this means that we can model an archive just by considering its archival divisions and by defining the relationships between them without taking into account their actual content.

When we consider a collection of subsets C from the the content point-of-view, it means that we refer to its extensional level. In this case a collection of subsets C is represented by the extension of the sets composing it; the properties of the sets are then verified by inspecting the sets and verifying the elements that they contain. In this case, we say that the content of a collection of subsets defines the extension of such a collection. From the archival point-of-view, this means that we can model an archive just by considering the actual content of its archival divisions without explicitly defining the relationships between them. Therefore, we can say that $C = \{A, B, C\}$ where $A = \{a, b, c, d\}$, $B = \{b\}$ and $C = \{c, d\}$ is the extension of a NS-C. In the next example we can see a NS-C defined at the intensional level which is instantiated by two different NS-C specified at the extensional level.

Example 1. *Let us consider the following NS-C defined at the intensional level:* $C = \{A, B, C, D\}$ *where* $B \subseteq A$, $C \subseteq A$, $D \subseteq C$ *and* $B \nsubseteq C \wedge C \nsubseteq B$. *Then,* $A = \{a, b, c, d, e\}$, $B = \{b\}$, $C = \{c, d, e\}$, $D = \{d, e\}$ *– represented in Figure 8(a) – is a valid instance for C, as well as* $A = \{a, b, c, d, e, f\}$, $B = \{c, d\}$, $C = \{b, e, f\}$ *and* $D = \{f\}$ *– represented in Figure 8(b); indeed, they both satisfy the specified structural conditions.*

This very example can be described in the context of the archives by exploiting the very simple archive modeled by means of the NS-M shown in Figure 9; it allows us to see how it is possible to define the intension and the extension of an archive thanks to NESTOR.

Example 2. *Let us consider the archive represented by the NS-M in Figure 9. At the the intensional level, the archive can be modeled as follows:* $C = \{\texttt{fonds}, \texttt{subfondsA}, \texttt{subfondsB}, \texttt{seriesA}\}$ *where* $\texttt{subfondsA} \subseteq \texttt{fonds}$, $\texttt{subfondsB} \subseteq \texttt{fonds}$, $\texttt{serieA} \subseteq \texttt{subfondsB}$, $\texttt{subfondsA} \nsubseteq \texttt{subfondsB}$ *and* $\texttt{subfondsB} \nsubseteq \texttt{subfondsA}$.

Then, $\texttt{fonds} = \{\texttt{summary}, \texttt{letterA}, \texttt{letterB}, \texttt{letterC}\}$, $\texttt{subfondsA} = \{\texttt{letterA}\}$, $\texttt{subfondsB} = \{\texttt{letterB}, \texttt{letterC}\}$, $\texttt{seriesA} = \{\texttt{letterD}\}$ *is a valid instance for C; it describes the extension of the archive.*

Both the structural and the content aspects are important for the treatment of the NESTOR model. We exploit the structure defined at the intensional level to define the properties of

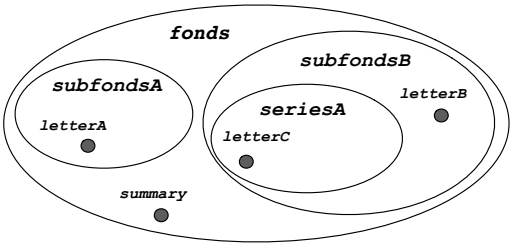


Figure 9: A synthetic archive modeled and represented by means of the NS-M used in Example 2.

NESTOR, whereas we exploit the extensional level to perform set operations which manipulate the content of the subsets composing the collections.

In the following we make extensive use of the concepts of collection of proper subsets and supersets and of direct subsets and supersets. Let C be a collection of sets and $A \in C$ be a set, we define:

- $\mathcal{S}^{-}(A) = \{B \in C \ : \ A \subset B\}$ to be the **collection of proper supersets** of A in C;
- $\mathcal{S}^{+}(A) = \{B \in C \ : \ B \subset A\}$ to be the **collection of proper subsets** of A in C.
- $\mathcal{D}^{-}(A) = \{B \in C \ : \ ((A \subset B) \wedge (\nexists E \in C \mid A \subset E \subset B))\}$ to be the **collection of direct supersets** of A in C.
- $\mathcal{D}^{+}(A) = \{B \in C \ : \ ((B \subset A) \wedge (\nexists E \in C \mid B \subset E \subset A))\}$ to be the **collection of direct subsets** of A in C.

4.2. Properties of the Nested Sets Model

Many properties of the NS-M are derived from the straightforward application of set theory as we show in the following example which takes into account the intensional level of the NS-M.

Example 3. *Let C be a NS-C. For all $H, K \in C \mid H \subseteq K$ we can easily derive that $H \cup K = K$ and $H \cap K = H$. As well we can say that for all $H, K \in C \mid H \nsubseteq K \wedge K \nsubseteq H \Rightarrow H \setminus K = H \wedge K \setminus H = K$.*

In this example we see that the sets in a NS-C behave exactly as one would expect under the operations of union, intersection and set difference. Let us see an example which shows how these operations behave at the extensional level.

Example 4. *Let $C = \{A, B, C\}$ be a NS-C, where $B \subseteq A$ and $C \subseteq B$. Then let us consider the following instance: $A = \{a, b, c, d, e\}$, $B = \{c, d, e\}$ and $C = \{e\}$. Then, $B \cup C = \{c, d, e\} = B$ and $B \cap C = \{e\} = C$.*

Let us consider a NS-C C; the next proposition shows that for all $H \in C$, H has at most one direct superset.

Proposition 1. *Let C be a NS-C. Then, $\forall H \in C, |\mathcal{D}^{-}(H)| \leq 1$.*

The following corollary to this proposition shows that the set with minimum cardinality in the collection of supersets of H is its direct superset.

Corollary 2. *Let C be a NS-C, $H \in C$ be a set, $\mathcal{S}^-(H)$ be the collection of proper supersets of H and $K \in \mathcal{S}^-(H)$ where $\forall L \in \mathcal{S}^-(H), |K| \leq |L|$ be the subset with minimum cardinality in $\mathcal{S}^-(H)$. Then, $\mathcal{D}^-(H) = K$.*

Finally, the next proposition proves that the direct subsets of H are always disjoints.

Proposition 3. *Let C be a NS-C and $H \in C$ be a set, then $\forall K, L \in \mathcal{D}^+(H), K \cap L = \emptyset$.*

4.3. Properties of the Inverse Nested Sets Model

The following proposition shows the behaviour of union and set difference in the INS-M under specific conditions. Property 4.5 shows that, given an INS-C, the union of two disjoint sets is a set which does not belong to the INS-C; whereas, Property 4.6 shows that the difference between two sets in the given INS-C is a set not belonging to the INS-C.

Proposition 4. *Let C be an INS-C and $\{H, K\} \in C$ two sets where $H \neq K$. Then,*

$$((H \not\subseteq K) \wedge (K \not\subseteq H)) \Leftrightarrow H \cup K = L \notin C \tag{4.5}$$

$$H \setminus K = L \notin C. \tag{4.6}$$

Let us consider an INS-C C, then for all $H \in C$, H has at most one direct subset.

Proposition 5. *Let C be an INS-C. Then, $\forall H \in C, \ |\mathcal{D}^+(H)| \leq 1$.*

The following corollary to this proposition proves that for all $H \in C$, the set with maximum cardinality in the collection of subsets of H is its direct subset.

Corollary 6. *Let C be an INS-C, $H \in C$ be a set, $\mathcal{S}^+(H)$ be the collection of proper subsets of H and $K \in \mathcal{S}^+(H)$ where $\forall L \in \mathcal{S}^+(H), |K| \geq |L|$ be the subset with higher cardinality in $\mathcal{S}^+(H)$. Then, $\mathcal{D}^+(H) = K$.*

We know that for all $H, K \in C$ where C is an INS-M, the intersection between them is never empty, otherwise Condition 4.3 of Definition 2 does not hold. The next proposition proves that the intersection between H and K is the set with maximum cardinality among all of their common subsets.

Proposition 7. *Let C be an INS-C and $H, K, L \in C$ be three sets such that $H \cap K = L$, then $\forall\ W \in (\mathcal{D}^+(H) \cap \mathcal{D}^+(K)), W \neq L \Rightarrow |L| > |W|$.*

4.4. Equivalence Between the NS-M and INS-M

In the following we prove the equivalence between the two proposed set data models by presenting two functions ζ and ξ which allow us to go back and forth from a NS-C to an INS-C and vice versa. The possibility of mapping between one model and the other allows us to model an archive by means of both the presented models, thus exploiting the properties that are better suited for the necessities we may have.

Definition 3. *Let A be a set and C and $\mathcal{E}$ be two collections of subsets of A. We define $\zeta : C \to \mathcal{E}$ to be a function such that for all $H \in C$ there exists $K \in \mathcal{E}$ such that:*

$$K = \bigcup_{L \in \{H \cup \mathcal{S}^-(H)\}} \Big(L \setminus \bigcup_{W \in \mathcal{D}^+(L)} W\Big) \tag{4.7}$$

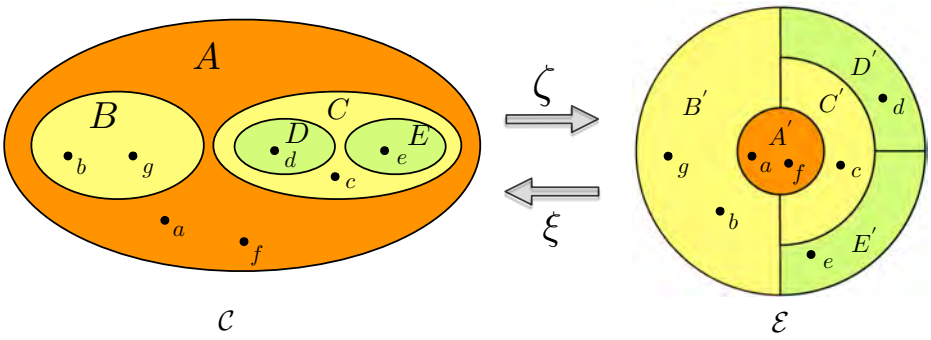


Figure 10: Mapping between NS-C and INS-C through the ζ and ξ functions.

For every set $H \in C$, the ζ function takes into account all its supersets – i.e. $H \cup \mathcal{S}^{-}(H)$; for each one, say L, of these supersets, ζ retains all the elements that exclusively belong to L – i.e. $L \setminus \bigcup \mathcal{D}^{+}(L)$, the elements which are in L and do not belong to any other direct subset of L. Then, the set $K = \zeta(H)$ contains the union of all the elements of all the considered sets.

Definition 4. *Let C and $\mathcal{E}$ be two collections of subsets. We define $\xi : C \to \mathcal{E}$ to be a function such that for all $H \in C$ there exists $K \in \mathcal{E}$ such that:*

$$K = \Big(H \cup \bigcup_{L \in \mathcal{S}^{-}(H)} L\Big) \setminus \bigcup_{L \in \mathcal{D}^{+}(H)} L \tag{4.8}$$

The ξ function maps every set $H \in C$ into another set, call it $K \in \mathcal{E}$. K is defined by the union of all the elements belonging to H and to its supersets minus all the elements belonging to the subsets of H itself.

The next theorem shows that NS-M and INS-M have the same expressive power by proving that if we apply the function ζ to a NS-C we obtain an INS-C as output.

Theorem 8. *Let C be a NS-C then $\zeta(C) = \mathcal{E}$ is an INS-C.*

Now, let us see how the ξ function allows us to map an INS-C into a NS-C.

Theorem 9. *Let C be a INS-C then $\xi(C) = \mathcal{E}$ is a NS-C.*

In Figure 10 we can see the mapping between the two set data models through the ζ and ξ functions.

5. Equivalence between the Archival Tree and NESTOR

Archivists use the tree as the model of an archive because it expresses the multileveled and hierarchical nature of the relationships between the archival divisions [57]. As discussed in the previous sections, NESTOR adopts instead an approach based on set inclusion relationships and we have intuitively shown that it is suitable for modeling an archive.

The formal definition of the set data models and their properties we gave in Section 4 prove that the nested sets idea is not just an alternative graphical representation of the tree, but a proper

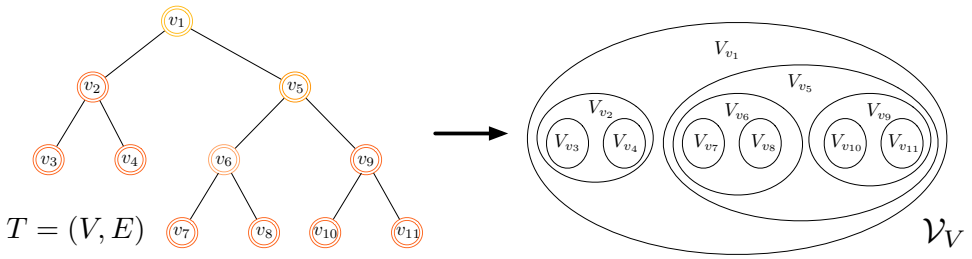


Figure 11: A tree $T = (V, E)$ and a NS-C $\mathcal{V}_V$ mapped from it.

data model that can be exploited to represent and manage hierarchies. In order to exploit this model in the archival context it is necessary to prove that these newly defined set data models are as expressive as the tree and that they can model all the facets of archival reality. We prove that the expressive power of the set data models and the tree are formally comparable and that the set data models allow us to explicitly represent aspects of the reality that are problematic to capture with the tree. The major concern of the tree is on the hierarchical structure defined between the entities represented by means of it; the set data models allow us to do the same by means of collections of sets and at the same time to add a further expressive dimension represented by the elements belonging to the sets.

In this section, we formally prove that modeling an archive by means of a tree is equivalent to modeling it with the set-based approach adopted in NESTOR. To this end, we present two formal mappings from the tree to the NS-M and INS-M models and vice versa, thus verifying their equivalence.

5.1. Equivalence between Tree and NS-M

First of all, we present the formal mapping between the tree and the NS-M. The mapping procedure creates a set for each node of the tree and defines the inclusion order between the newly created sets using the information brought by the edges connecting the nodes of the tree. For instance, let $T = (V, E)$ be a tree; if we consider an edge $e_{j,k} \in E$, then we have to create two sets J and K corresponding to the nodes $v_j, v_k \in V$ such that $K \subseteq J$; indeed, from $e_{j,k}$ we know that v_j is the parent of v_k and so set J will be the superset of K. In order to properly understand the mappings, it is worthwhile introducing two concepts we will widely use in the following. We define with $\Gamma^+(v_i)$ the set of **all the descendants** of v_i in V (including v_i itself); vice versa $\Gamma^-(v_i)$ is the set of **all the ancestors** of v_i in V (including v_i itself).

Theorem 10. *Let $T = (V, E)$ be a tree and let C be a collection of subsets where $\forall v_i \in V$, $\exists!\ H \in C = \Gamma^+(v_i)$. Then C is a Nested Sets Collection.*

This theorem shows us that if we map a tree into a collection of subsets following the described rules, we obtain a NS-C. In Figure 11 we show how a tree can be mapped in a family of subsets $\mathcal{V}_V$ as proved by Theorem 10.

The following theorem shows that a NS-C can be mapped into a tree by creating a node from every set in the NS-C. Two sets J and K in the NS-C correspond to two nodes v_j and v_k in the tree and the edge $e_{j,k}$ between them is created if and only if J is the direct superset of K.

Theorem 11. *Let C be a NS-C, V be a set of nodes and E be a set of edges where $\forall v_j \in V, \exists!\ J \in C \wedge \forall e_{j,k} \in E, \exists!\ J, K \in C \mid K \subseteq J$. Then $T = (V, E)$ is a tree.*

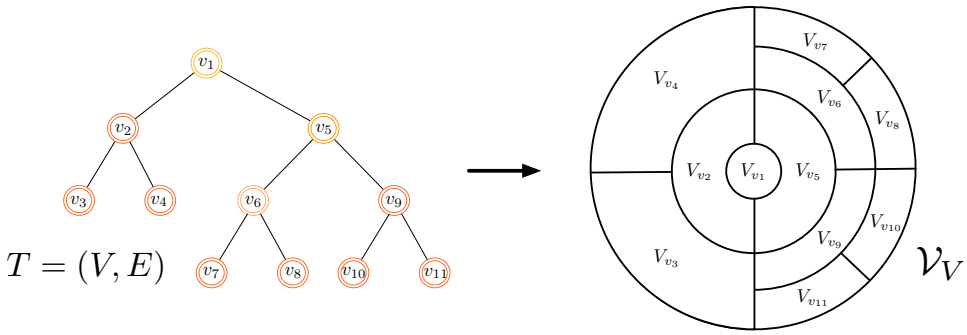


Figure 12: A tree $T = (V, E)$ and an INS-C $\mathcal{V}_V$ mapped from it.

We have formally defined the relationships between a tree with the NS-M; we know that a tree can be mapped into a NS-C where every node of the tree is mapped into a set of the collection and vice versa.

5.2. Equivalence between Tree and INS-M

Now we can present the corresponding theorems for the INS-M which show how a tree can be mapped into an INS-C and vice versa. Basically, every couple of nodes v_j and v_k is mapped into a couple of sets J and K. If there is an edge between v_j and v_k, say $e_{j,k}$, then the set J created from v_j is defined as a subset of the set K created from v_k. The mapping between a tree and an INS-C reverses the idea described for the mapping of a tree into a NS-C; if a node is a parent of another node in a tree, this is mapped into a set which is a subset of the set created from its child node.

Theorem 12. *Let $T = (V, E)$ be a tree and let C be a collection of subsets where $\forall v_i \in V, \exists! I = \Gamma^{-}(v_i)$. Then C is an INS-C.*

This theorem shows us that if we map a tree into a collection of subsets following the described rules, we obtain an INS-C. In Figure 12 we show how a tree can be mapped in a family of subsets $\mathcal{V}_V$ as shown by Theorem 12.

Now we can see how an INS-M C is mapped into a tree $T = (V, E)$; the following theorem shows that if we map every couple of sets $\{A_j, A_k\} \in C$ into a couple of nodes $\{v_j, v_k\} \in V$ such that there is an edge $e_{j,k} \in E$ if and only if A_j is a direct subset of A_k, then the graph defined by the nodes in V connected by the edges in E is a tree.

Theorem 13. *Let C be an INS-C, V be a set of nodes and E be a set of edges where $\forall v_j \in V, \exists! J \in C \wedge \forall e_{j,k} \in E, \exists! J, K \in C \mid J \subseteq K$. Then $T = (V, E)$ is a tree.*

6. Extending the 5S Model via NESTOR

As discussed in Sections 1 and 3, the 5S model needs some kind of extension to be tailored to the specific case of archives.

The notion of *descriptive metadata specification*[6] (definition 12 [49, p. 292]) is suitable either for representing, for each archival division, a descriptive metadata – e.g. a metadata describing a

[6] In this section, we use italics to highlight definitions taken from the 5S model.

series, a sub-fonds, or an archival unit – or for representing the archive as a whole, as it happens in the case of EAD.

When it comes to the definition of *metadata catalog* (definition 18 [49, p. 295]), there is no means to impose a structure over the descriptive metadata in the catalog. Therefore, if you use separate *descriptive metadata specifications* for each archival division, as in the former case, this would prevent the possibility of expressing the relationships between these archival divisions, i.e. you would lose the possibility of retaining the archival bond. This means that an archive cannot be properly modeled throughout the 5S model without losing one of its main properties.

Moreover, in a *metadata catalog*, there is no means to associate (sub–)parts of the *descriptive metadata specifications* to the *digital objects* (definition 16 [49, p. 294]) that they describe, but you can only associate a whole descriptive metadata to a whole digital object.

Therefore, if you represent an archive as a whole with a single *descriptive metadata specification*, as in the latter case, it would not be possible to associate (sub-)parts of that descriptive metadata to the different digital objects corresponding to the various archival divisions; this does not allow the definition of compound digital objects and it is a barrier towards the adoption of the LOD paradigm in the archival context as discussed in Section 8. Furthermore, this strongly limits the interoperability between digital archives and the possibility of sharing archival metadata with variable granularity.

Our extension to the 5S model is thus organized as follows:

- using the notion of *structure* (definition 2 [49, p. 288]), we introduce the notion of **NESTOR structure**, as a structure that complies with the constraints of NS-M or INS-M;
- using the notion of *metadata catalog*, we introduce the notion of **NESTOR metadata catalog**, as a metadata catalog that exploits a NESTOR structure to retain the archival bonds;
- using the notion of *digital library* (definition 24 [49, p. 299]), we introduce the notion of **digital archive**, as a digital library where at least one of the *metadata catalogs* is a NESTOR metadata catalog.

Definition 5. *Let C be a Nested Set Collection (NS-C) on a set A. A **NS-M structure**(A) is a structure $(NS\text{-}G, L, \mathcal{F})$, where L is a set of label values, $\mathcal{F}$ is a labeling function, and $NS\text{-}G = (V, E)$ is a directed graph where $\forall v_j \in V, \exists!\ J \in C \wedge \forall e_{j,k} \in E, \exists!\ J, K \in C \mid K \subseteq J$.*

Definition 6. *Let C be an Inverse Nested Set Collection (INS-C) on a set A. A **INS-M structure**(A) is a structure $(INS\text{-}G, L, \mathcal{F})$, where L is a set of label values, $\mathcal{F}$ is a labeling function, and $INS\text{-}G = (V, E)$ is a directed graph where $\forall v_j \in V, \exists! J \in C \wedge \forall e_{j,k} \in E, \exists! J, K \in C \mid J \subseteq K$.*

Definition 5 applies Definition 1 on page 14 and Theorem 11 on page 20 to the definition of *structure* in the 5S model, ensuring that the resulting structure complies with the NS-M. Note that the set of label values L and the labeling function $\mathcal{F}$ are not strictly needed for the NS-M, but they can be useful in the context of the 5S and this feature, in turn, may extend the NS-M with semantic possibilities. Similarly, definition 6 applies definition 2 on page 15 and theorem 13 on page 21.

Definition 7. *Given a set A, a **NESTOR structure**(A) is either a NS-M structure(A) or a INS-M structure(A).*

The definition of *metadata catalog* in the 5S model can be expressed as follows. Let H be a set of handles to *digital objects* and M a set of *descriptive metadata specifications*, then a *metadata catalog* is a function $DM : H \times 2^M$.

Definition 8. *Let H be a set of handles to digital objects and M a set of descriptive metadata specifications, a metadata catalog DM is a* ***NESTOR metadata catalog*** *if:*

$$\forall h_i \in H \mid \exists M_i \in 2^M \ \wedge \ DM(h_i) = M_i \Rightarrow |M_i| = 1 \tag{6.1}$$

$$\exists \text{ NESTOR structure}(M) \tag{6.2}$$

Condition 6.1 imposes that, if exists, there is only one *descriptive metadata specification* for a given *digital object* because, in archival practice, every single metadata describes a unique archival division, being it a level in the archive or a digital object [57]. Condition 6.2 ensures that the relationships between the different archival divisions are compliant with the *descriptive metadata specifications* in M.

Definition 9. *A* ***digital archive*** $(\mathcal{R}, DM, Serv, Soc)$ *is a digital library where*

- $\mathcal{R}$ *is a repository;*
- *at least one of the metadata catalogs in the set of metadata catalogs DM is a NESTOR metadata catalog;*
- $Serv$ *is a set of services containing at least services for indexing, searching, and browsing;*
- Soc *is a society.*

Definition 9 extends the definition of *digital library* in the 5S model requiring that at least one of the *metadata catalog*s is a NESTOR one, i.e. there exists at least one *metadata catalog* capable of retaining the archival bonds. This definition has several consequences. Firstly, more than one NESTOR metadata catalogs can be present in the same digital archive, thus making it possible to express different archival descriptions over the same set of *digital objects*. This extends the current practice in which a system for managing an archive is usually capable of managing only one description of the archive, thus giving only one point-of-view on the material held [27, 61, 71]. Secondly, you can mix NESTOR and not-NESTOR metadata catalogs which allows for the seamless integration of different visions of the managed *digital objects* within the same digital archive. This opens up the possibility of exploiting the whole breadth of methodologies and tools available in the digital library field with the archives.

6.1. A Sample Instantiation of the Extended 5S Model

Let us consider the sample archive shown in Figure 13 where there are four archival divisions (i.e. a fonds, two series, and a unit) each one containing one or more digital objects representing the content of that archival division. In particular, the archival unit contains two digital objects (for instance, they could be the digitalization of two pages of a letter); as observed in Section 3.1, EAD cannot natively handle this case, whereas with NESTOR this is straightforward as we show in the following.

According to Definition 9 to model this digital archive throughout the extended 5S we need to define a repository $\mathcal{R}$, a NESTOR metadata catalog DM_C, a set of services $Serv$, and a society Soc. The 5S defines a repository (definition 19 [49, p. 295]) as a tuple $(\mathcal{R}, \text{get}, \text{store}, \text{del})$

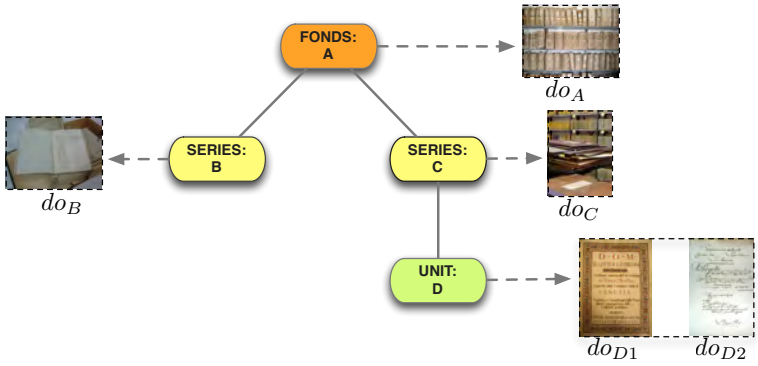


Figure 13: A sample archive composed by four divisions with one or more associated digital objects.

where $\mathcal{R} \subset 2^{C_{do}}$ and C_{do} is the considered collection of digital objects; in this example $C_{do} = \{do_A, do_B, do_C, do_{D1}, do_{D2}\}$.

A NESTOR metadata catalog $DM_{C_{do}}$ is a set of pairs associating a handle (i.e. $h_i \in H$) to a descriptive metadata specification (i.e. md_i) and for which exists a NESTOR structure. So, $DM_{C_{do}} = \{(h_A, \{md_A\}), (h_B, \{md_B\}), (h_C, \{md_C\}), (h_{D1}, \{md_D\}), (h_{D2}, \{md_D\})\}$. Now, we need to build a NESTOR structure over this metadata catalog; as defined in Definition 7, a NESTOR structure can be either a NS-M structure or a INS-M structure. In this example we present only a NS-M structure because the INS-M structure can be derived following the same procedure. For the archive in Figure 13 a NS-M structure is defined from the intensional point-of-view as $C = \{A, B, C, D\}$ where $B \subseteq A$, $C \subseteq A$, $D \subseteq C$, $C \nsubseteq B$, and $B \nsubseteq C$; from the extensional point-of-view it is defined as $A = \{md_A, md_B, md_C, md_D\}, B = \{md_B\}, C = \{md_C, md_D\}, D = \{md_D\}$. In this case we defined the NS-M structure in a set-based fashion, but by employing Theorem 11 is possible to define it in a graph-based fashion (i.e. thus obtaining the graph NS-G) mapping the NS-C into a tree. A NS-M Structure requires also a set L of labels and a function $\mathcal{F}$ mapping from the NS-G to L; so, $L = \{\texttt{fonds}, \texttt{series}, \texttt{unit}\}$, $\mathcal{F}(A) = \texttt{fonds}$, $\mathcal{F}(B) = \texttt{series}$, $\mathcal{F}(C) = \texttt{series}$, and $\mathcal{F}(D) = \texttt{unit}$. Note that in this case, each of the presented descriptive metadata specification (i.e. $md_A, \ldots, md_D$) describes one or more digital objects. Referring back to the archival state-of-the-art, each metadata could be seen a part of the EAD metadata standard; for instance, md_A describes a fonds thus it can be represented by the `c1` component in EAD – i.e. $\langle$`c1 level = fonds`$\rangle$.

To complete the definition of digital archive we need to specify the set of services as they are defined by the 5S (Definition 7 [49, p. 290]); for an archive a possible set of services can be *Serv* = {`browse`, `search`, `describe`, `update`, `store`} which allows the users to browse the archive, search for a specific piece of information, describe a new archival resource and update the existing resources. The users of the digital archives are defined by the 5S concept of society (Definition 10 [49, p. 292]), which for an archive can be instantiated as *Soc* = {`archivist`, `student`, `general public`, `historian`}. In Figure 14 we give a graphical representation of the sample archive of Figure 13 represented via the extended 5S Model.

The extension of the 5S Model via NESTOR represents an actual bridge between these two formal models which allows for a realization of an integrated and inteoroperable environment for LAM. In particular, this explicit connection allows the archives to live and cooperate with other methodologies initially not built for archives paving the road for an actual sharing of functional-

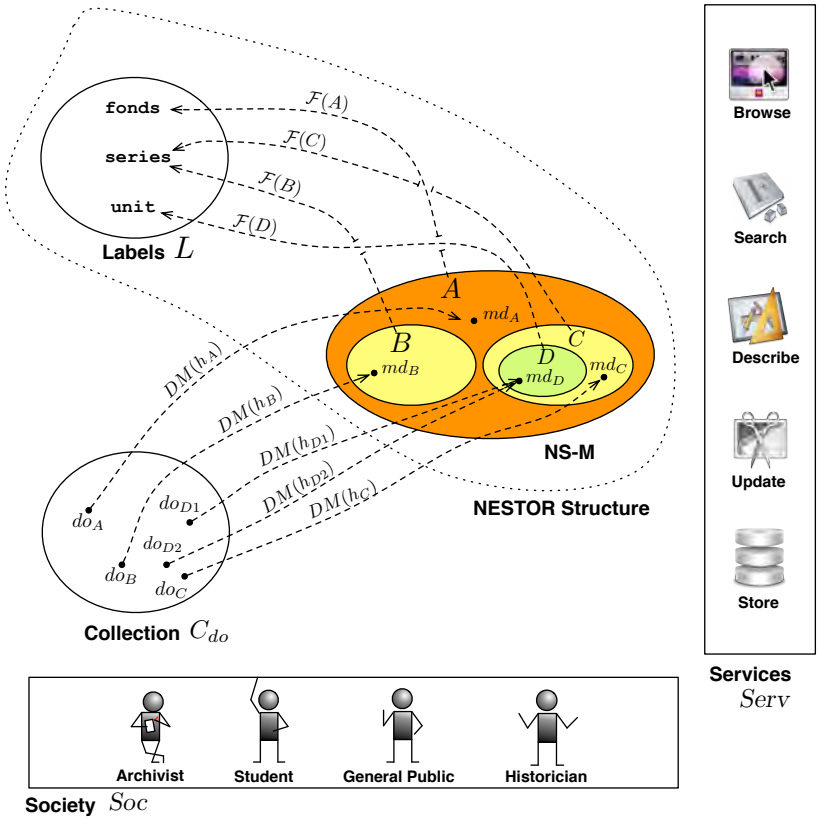


Figure 14: A graphical representation of the instantiation of the extended 5S Model for the sample archive of Figure 13.

ities in the LAM context.

A meaningful example is OAI-PMH which is formally defined in the context of the 5S and can now be employed by the archives without changing its internal functioning and broadening its functionalities (see Section 7). This theoretical framework is also employed for exposing archives through the LOD paradigm instantiated by OAI-ORE. Lastly, the very methodology adopted for extending the 5S model can be adapted for connecting NESTOR to the FAST formal model in order to enrich the archives by means of collaboration tools such as digital annotations as we show in Section 9.

7. Use Case: Detaching the Archives

Modeling digital archives throughout the extended 5S model opens-up new ways of representing and handling archival resources. A relevant advancement resides in the possibility of adopting widely-used digital library technologies within the archives without changing their inner functioning or modifying them and, at the same time, retaining all the archival fundamental characteristics – e.g. the context of resources and the archival bond.

The Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH)[7] is the *de-facto* standard for metadata exchange in digital libraries [15, 42, 51, 67, 92] and it has also been modeled by means of the 5S model allowing for "the specification and automatic generation of digital library applications" [49].

OAI-PMH is open from an architectural point-of-view and a low-barrier mechanism for repository interoperability. It is based on Web standards such as HyperText Transfer Protocol (HTTP) [41] and XML, and on two main components, *Data Providers* and *Service Providers*, where the former are repositories that export records in response to requests from a software service called harvester and the latter are those services that harvest records from Data Providers and provide added-value services built on top of the aggregated harvested metadata.

The protocol defines a harvesting procedure called *selective harvesting* which is of interest for our purposes. Selective harvesting is based on the concept of *OAI-set*, which enables logical data partitioning by defining groups of records (i.e. OAI-records), and permits the harvesting only of records owned by a specified OAI-set. An OAI-set is identified by a `setSpec` which is a mandatory and unique handle for a set within the repository. OAI-set organization may be flat or hierarchical, where hierarchy is expressed in `setSpec` field by the use of a colon [:] separated list indicating the path from the root of the set hierarchy to the respective node. For example, if we define an OAI-set the `setSpec` of which is "A", its sub-set "B" would have "A:B" as `setSpec`. In this case B is seen by the protocol as a proper sub-set of A: $B \subset A$. Harvesting from a set which has sub-sets will cause the repository to return metadata in the specified set and recursively to return metadata from all the sub-sets. In our example, if we harvest set A, we also obtain the items in sub-set B [102].

OAI-PMH is formally described by means of the 5S model. Data and Service providers are represented as (electronic) *Societies*; the communications between the Data and Service providers are *Streams*; and, the sets, metadata, and schemas are *Structures* [49, p. 283].

When it comes to archives, as discussed in Section 3, EAD is the reference standard to be considered. It represents an archive as a monolith and every description is embedded in the archival structure (see Figure 4 at page 10). This means that *content and structure are interlinked in the same XML file* and they cannot be handled separately.

Several state-of-the-art mapping initiatives clashed with this problem; for instance, [21] described the possibility of mapping Machine Readable Cataloging (MARC) metadata into EAD but not vice versa because MARC does not allow for retaining the archival structure like EAD does. The problem of retaining the archival structure and at the same time being able to exchange metadata with variable granularity also affects many other mapping initiatives. Indeed, a common solution for exchanging archival metadata in distributed environments is to map EAD into a collection of lightweight metadata – i.e. Dublin Core (DC) metadata – that can be exchanged and accessed with a variable granularity [19, 40, 84, 85]. The main problem with these solutions is that the DC metadata cannot retain the archival structure by themselves. Instead they have to be related by means of several links to the EAD structure, thus they are not independent from the original EAD file. Several proposals solve the mapping problem from the content point-of-view, but they do not provide a way to retain the archival structure without referring to the original EAD file or to a relational database which is, by its nature, not easibly shareable with variable granularity while retaining the archival context [24].

There are services providing support for sharing archival metadata, such as the OCLCs

[7] `http://www.openarchives.org/pmh/`

ArchiveGrid[8], where "EAD-encoded findings aids are harvested by agreement between institutions and the ArchiveGrid service, and aggregated together with HTML-encoded inventories and collection-level descriptions" [87]. The main criticism of these approaches is that they do not support the wide distribution of data "that is essential for archives to participate fully in a constantly changing information environment" [87] and that they do not provide for variable granularity access and exchange of them.

Other general-purpose digital library systems do not take into account the possibility of managing and sharing archival metadata like EAD. For instance, Greenstone[9] has an extensible mechanism for handling a wide variety of file formats through document plugins, but the best way to get EAD into Greenstone would be through an EAD-plugin written specifically for this because no mapping service is provided. Fedora Commons[10] provides support for ingesting and visualizing EAD files but does not provide any solution for mapping or exchanging these files in a distributed environment.

Modeling an archive throughout a unique EAD file also limits the information access possibilities. Indeed, the unique entry point to access the information is the root of the file and then we have to navigate the hierarchy to access the information of interest. In order to overcome this issue we can define some superstructures to the EAD; for instance, we can settle some predefined entry points by the use of XPointers[11] pointing to specific elements of the XML or by using predefined paths driving the user through the hierarchical structure.

A direct consequence is that in a distributed environment where it is necessary to exchange data between repositories we are forced to exchange the archive as a whole. Indeed, we cannot share a specific piece of information – e.g. the descriptions of the documents belonging to a specific series – without extracting it from the EAD file and losing in this way the structural information retained thanks to the nested tags in the EAD itself [33, 85, 108]. This leads to difficulties in fully exploiting the OAI-PMH within the archives. Indeed, OAI-PMH can be used only to exchange the whole archive as a monolithic unit, thus many of the useful functionalities of the protocol cannot be exploited.

To this end, we exploit NESTOR along with the extended 5S to propose a general solution for modeling the archives, thus overcoming the presented limitations and enabling a full exploitation of standard digital library technologies within digital archives.

Let us consider a family of subsets $\mathcal{F}_I$ on a NS-C indexed by a set I composed of `<setspec>` values. Elements of I must ensure that each `<setspec>` complies with the NS-M constraints, that is $i \in I = \{s_0 : s_1 : \ldots : s_j\}$ means that it exists an $F_j \in \mathcal{F}_I$ such that $F_j \subset \ldots \subset F_1 \subset F_0$. Every $F_j \in \mathcal{F}_I$ is an OAI-set identified by a `setspec` value in I.

The `setspec` values for each $F_k \in \mathcal{F}_I$ are built in such a way to maintain the inclusion order between the sets. If an F_k has no superset its `setspec` value is composed only of a single value (`<setspec>`s_k`</setspec>`). Instead if a set F_h has supersets, e.g. F_a and F_b where $F_b \subset F_a$, its `setspec` value must be the combination of the name of its supersets and itself separated by the colon [:] (e.g. `<setspec>`$s_a : s_b : s_h$`</setspec>`). Furthermore, let $OAI = \{oai_0, \ldots, oai_n\}$ be a set of OAI-records, then each $oai_i \in F_j$ must contain the `setspec` of F_j in its header.

Let us consider the archive represented by the NS-C in Figure 1 at page 3. As we can see in Figure 15, each set composing this nested set structure is mapped into an OAI-Set with a proper

[8] `http://http://archivegrid.org/`
[9] `http://www.greenstone.org/`
[10] `http://www.fedora-commons.org/`
[11] `http://www.w3.org/TR/xptr-framework/`

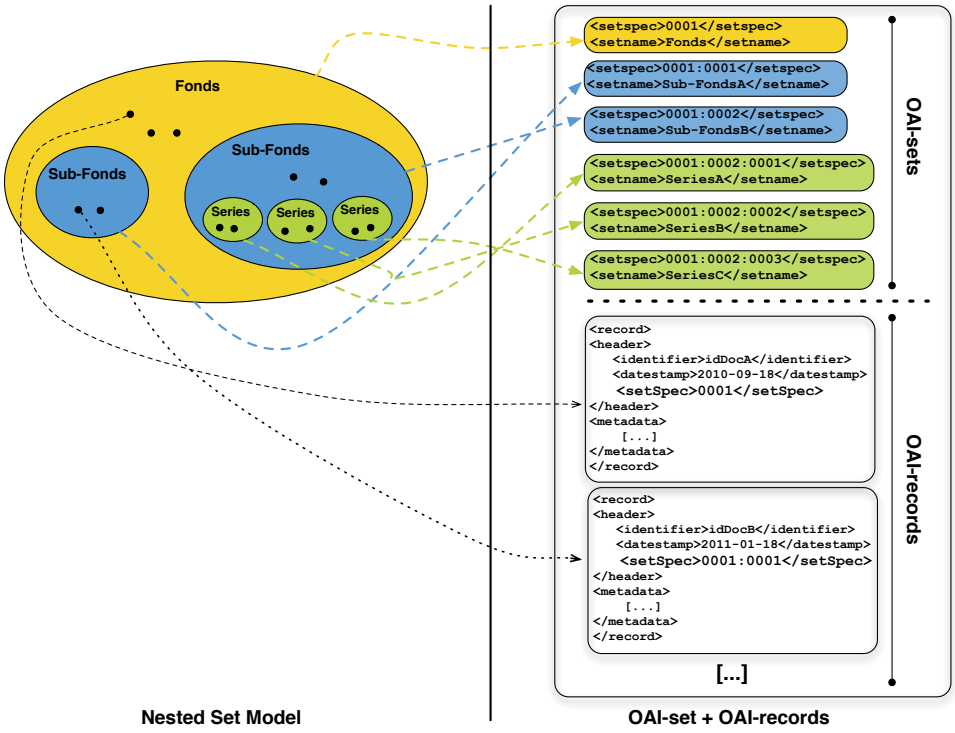


Figure 15: An archive represented throughout the NS-M and mapped into OAI-sets and OAI-records.

`setSpec`; the set called "fonds" is mapped into an OAI-set with < `setspec` > `0001` < /`setspec` >. This set has two subsets that are mapped into two OAI-sets: < `setspec` > `0001` : `0002` < /`setspec` > and < `setspec` > `0001` : `0003` < /`setspec` > and so on for the other sets.

We can see that the hierarchical relationships and thus the inclusion order between the sets is maintained by the identifiers of the OAI-sets which are defined as materialized paths from the root to the identified set. Each single archival description is mapped into a metadata belonging to an OAI-set; the membership information is added to the header of these metadata that are seen as OAI-records. In this way each archival description can be encoded by a single metadata without any constraints on its format; indeed, an OAI-set can contain different kinds of metadata formats. With this model we do not impose any conditions on the archival descriptions, thus allowing the possibility of changing the metadata, updating the information or adding a new metadata format without affecting the structure of the archive and without changing the data model.

An important aspect that has to be highlighted is that this implementation also maintains the horizontal dimension of the archival hierarchy – i.e. the order between the subsets of a set. In Figure 15 we can see that we can talk of the first sub-fonds of the fonds (we named it Sub-fondsA) or of the second series of sub-fonds. This is possible because the OAI `setspecs` define the inclusion order between the OAI-sets but also a partial order between the OAI-sets which are common subsets of another OAI-set.

In the same way, we can use the INS-M with OAI-PMH. Let $\mathcal{G}$ be an INS-C indexed by a set J (i.e. a family of subsets $\mathcal{G}_J$) composed by `<setspec>` values such that $j \in J = \{s_0 : s_1 : \ldots : s_k\}$

means that $\exists\, G_k \in \mathcal{G}_J = G_k \subset \ldots \subset G_1 \subset G_0$.

In $\mathcal{G}_J$, unlike in $\mathcal{F}_I$, the following case may happen:
Let $\{G_i, G_k, G_w\} \in \mathcal{G}_J$, then it is possible that $G_w \subset G_i$ and $G_w \subset G_k$ but either $G_i \nsubseteq G_k$ and $G_k \nsubseteq G_i$. If we consider $\mathcal{G}_J$ composed only of G_i, G_k and G_w, the identifier of G_i is `<setspec>`s_i`</setspec>` and the identifier of G_k is `<setspec>`s_k`</setspec>`. Instead, the identifier of G_w must be `<setspec>`$s_i : s_w$`</setspec>` and `<setspec>`$s_k : s_w$`</setspec>` at the same time; this means that in $\mathcal{G}_J$ there are two distinct OAI-sets, one identified by `<setspec>`$s_i : s_w$`</setspec>` and the other identified by `<setspec>`$s_k : s_w$`</setspec>`. This is due to the fact that the intersection between OAI-sets in OAI-PMH is not set-theoretically defined [103]; indeed, the only way to get an intersection of two OAI-sets is by enumerating the records. This means that we can know if an OAI-record belongs to two or more sets just by seeing whether there are two or more `<setspec>` entries in the header of the record. In this case the records belonging to G_w will contain two `<setspec>` entries in their header: `<setspec>`$s_i : s_w$`</setspec>` and `<setspec>`$s_k : s_w$`</setspec>`; note that only the `<setspec>` value is duplicated and not the records themselves.

Let us consider the sample archive represented by the INS-C in Figure 2 on page 4. In Figure 16 we can see how the INS-C is mapped into a collection of OAI-sets and OAI-records. We obtain four sets from the common subset – i.e. the fonds of the sample archive – with four different identifiers: "0004:0001", "0001:0001:0001", "0002:0001:0001" and "0003:0001:0001". The sets mapped from the children of the root are defined in the same way. The sets related to the series are identified by "0001", "0002" and "0003". We can see that the OAI-records belonging to the "fonds" have four `setspecs` in the header because the fonds in the INS-M representation is the common subset of four other sets, thus it has four different associated OAI-sets.

These instantiations of the set data models have three main relevant features which are also important aspects defining the flexibility and adaptability of NESTOR: (i) they clearly divide the structural elements (i.e. the sets) from the content elements (i.e. the archival descriptions); (ii) they do not bind the archival descriptions to a unique, fixed and predefined metadata format; (iii) they exploit digital library technologies, like OAI-PMH, without any change in their internal functioning and without any extension.

We make available a variable granularity access to the structure and to the content of an archive. Indeed, each OAI-set is individually accessible as well as each single metadata. From an OAI-set we can easily reconstruct the relationships with the other OAI-sets by exploiting the `setspec` organization; from a metadata we can reconstruct the relationships with the other metadata thanks to the membership information contained in their header.

By means of OAI-PMH it is possible to exchange a specific part of the archive while at the same time maintaining the relationships with the other parts of it. The NS-M fosters the reconstruction of the lower levels of a hierarchy; thus, with the pair formed by NS-M and OAI-PMH applied to an archive, if a harvester asks for an OAI-Set representing for instance a sub-fonds it recursively obtains all the OAI-subsets and items in the subtree rooted in the selected sub-fonds.

The INS-M fosters the reconstruction of the upper levels of a hierarchy which in the archival case often contain contextual information which permit the relationships of archival documents to be inferred with the other documents in the archive and with the production and preservation environment.

The choice between a NS-M or INS-M should be made on the basis of the application context. For instance, often the information required by a user is stored in the external nodes of the archival tree [92]. If we model the archival tree by means of the INS-M, when a harvester requires an external node of the tree it will receive all the archival information contained in the

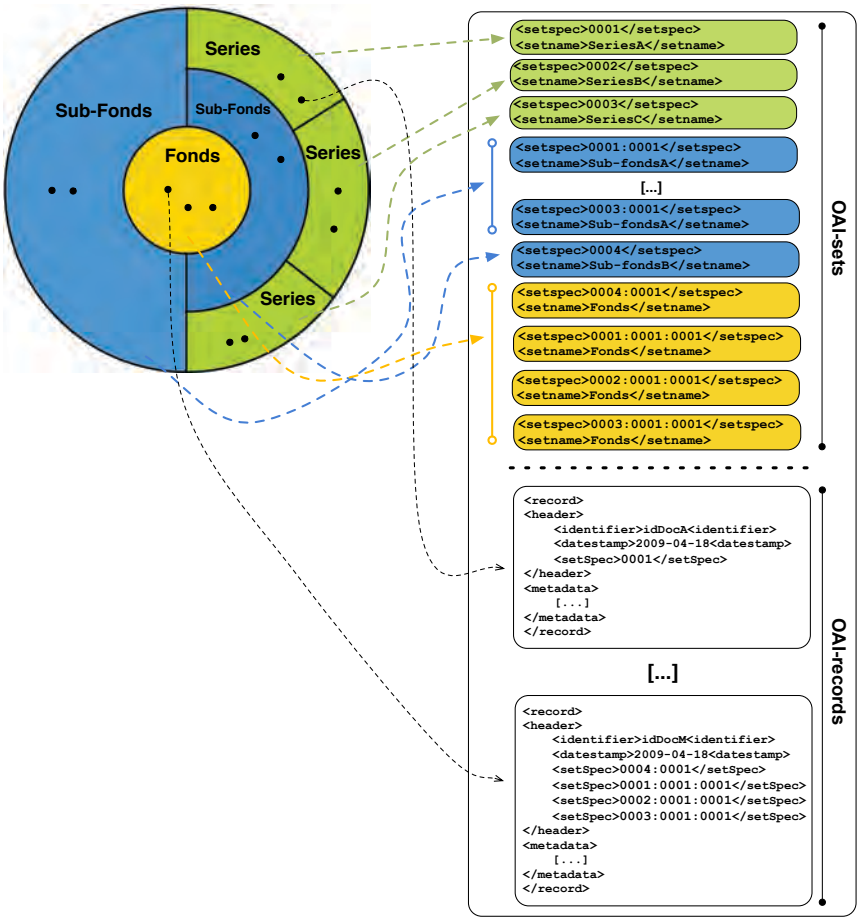


Figure 16: An archive represented throughout the INS-M and mapped into OAI-sets and OAI-records.

nodes comprised in the path going from the required node up to the root of the archive. This means that a Service Provider can offer a potential user the required information stored in the external node and also all the information stored in its ancestor nodes and thus its context.

Furthermore, the possibility of going from one set data model to the other by means of the defined mapping functions is very useful in the archival context because we can address the user requirements in the most effective way without being bound the properties of a single model of choice.

8. Use Case: Unchaining the Archives

Currently, archival practice is moving towards the definition of complex relationships between the resources of interest as well as the constitution of compound digital objects. To this end archives can take advantage of using the LOD paradigm which eases the access to the resources, enhances the interoperability by moving the focus from the systems managing the data to

the data themselves, and provides additional and flexible representations of archival resources. In the context of digital libraries, the LOD paradigm can be instantiated by means of Open Archives Initiative Object Reuse and Exchange (OAI-ORE) which has a precise focus in the representation and management of compound digital objects.

In this section we define a formal basis that provides a means for: (i) defining OAI-ORE instances which are consistent with the fundamental archival principles; and, (ii) overcoming the issues affecting state-of-the-art solutions in the field of digital libraries we preseted in Section 3.1. In order to exploit OAI-ORE within the archives there is the need to model the archival structure – which is the mean to retain all the archival characteristics such as the archival bond – into OAI-ORE. The 5S models OAI-ORE as a *Structure* [65]; this very model extended via NESTOR allows us to impose conditions on the 5S *Structure* – i.e. by defining it as a *NESTOR Structure* (Definition 7) – thus creating OAI-ORE instances accordingly to the archival practice.

The OAI-ORE defines a machine-readable and standard mechanism for defining aggregations of resources on the Web. By means of OAI-ORE we can identify a bunch of resources related to each other as a single entity enabling the access and exchange of them at an aggregation level of granularity. The OAI refers to these aggregations as "*compound objects*". Compound units are aggregations of distinct information units that, when combined, form a logical whole. Some examples [101] of these are a digitized book that is an aggregation of chapters, where each chapter is an aggregation of scanned pages, and a scholarly publication that is an aggregation of text and supporting materials such as datasets, software tools, and video recordings of an experiment; also the archives can be seen as aggregations of archival metadata describing archival objects which in turn can have a digital form.

The OAI-ORE data model is based on three main kinds of resources: *Aggregation*, *Aggregated Resources* and *Resource Map*. An Aggregation is defined as a resource representing a logical collection of other resources. An Aggregation is a logical construct and thus it has no representation; it is described by a Resource Map which can be seen as a materialization of the Aggregation. A Resource Map must describe a single Aggregation and must enumerate the constituent Aggregated Resources; a resource is an "Aggregated Resource" in an Aggregation only if it is asserted in a Resource Map. Each resource in the OAI-ORE data model is identified by a URI. The OAI-ORE data model is expressed by the Resource Description Framework (RDF)[12], so its instances are expressed as RDF graphs. An RDF graph is defined by a set of triples (s, p, o) expressing the relationship defined by a predicate p between a subject s and an object o; s and o may be a URI with an optional fragment identifier, a literal or a blank (having no separate form of identification). Properties p are URI references[13].

In order to explain how an archive can be properly modeled as an instance of the OAI-ORE data model and thus be exposed as LOD, we have to consider that the issues of EAD determined by the lack of distinction between structure and content is emphasized when we take into account the *digital objects*. A single EAD metadata can directly point-to at most one digital object at the time and to overcome this problem an *ad-hoc* solution which exploits METS as a meta-structure over EAD has been proposed [97] with the limitations we discussed in Section 3 and depicted in Figure 5 on page 11.

In the previous section a methodology is described for mapping an EAD file into NESTOR which preserves the full informative power of the metadata. In this way, NESTOR can be used as a model to describe an archive from scratch as well as a mapping component that allows us to

[12] `http://www.w3.org/RDF/`

[13] `http://www.w3.org/TR/rdf-concepts/`

manipulate and transform the EAD files while respecting archival principles [35]. We exploit this very methodology to establish a direct and formal connection between OAI-ORE and NESTOR. In this use-case we present the mapping towards the NS-M; the mapping towards the INS-M can be derived symmetrically. This methodology allows for exposing EAD files as LOD in the Web; in the literature, to our knowledge, there are only two alternatives to this, but no one considers a mapping towards OAI-ORE. The first is an ontology-based metadata integration which maps EAD into the CIDOC Conceptual Reference Model (CRM) ontology [96] in order to address heterogeneity between different metadata formats. This methodology does not provide a solution of creating compound digital objects and exposing them as LOD. The second alternative [22] maps EAD to EDM in order to provide access to archival data from many access points; in this case the problem of relating archival descriptions in EAD with several digital objects and expose them on the Web is not specifically addressed.

In order to exploit OAI-ORE in the context of archives we provide a compact formal definition of the framework that will be exploited for bridging with NESTOR. We indicate with $UA \subset U = \{ua_1, \ldots, ua_k, \ldots, ua_n\}$ the set of URI identifying the Aggregations and with $\eta_A : UA \rightarrow R$ the restriction of η ($\eta|_A$) to UA; the image of η_A is the set of Aggregations $A \subset R = \{a_1, \ldots, a_k, \ldots, a_n\}$. In the same way, we indicate with $URM \subset U$ the set of URI identifying the Resource Maps and we define $\eta_{RM} : URM \rightarrow R$ to be the restriction $\eta|_{RM}$ where $RM \subset R$ is the set of Resource Maps. Finally, we indicate with $UAR \subset U$ the set of URI identifying the Aggregated Resources[14]. We define $\eta_{AR} : UAR \rightarrow R$ to be the restriction $\eta|_{AR}$ where $AR \subset R$ is the set of Aggregated Resources. Every $rm_i \in RM$ must describe one and only one $a_j \in A$, but a_j may be described by more than one Resource Map; thus, we indicate with $\varphi_{RMA} : RM \rightarrow A$ a function which maps a Resource Map to the Aggregation it materializes. Every $ar_i \in AR$ may be aggregated by more than one $a_j \in A$.

OAI-ORE comes with other two important features: *Proxy* and *Nested Aggregations*. A Proxy is a resource that indicates an Aggregated Resource in the context of a specific Aggregation and it is associated with an Aggregated Resource via an assertion in a Resource Map describing the Aggregation that is the context of the Proxy [68]. We indicate with $UP \subset U = \{up_1, \ldots, up_k, \ldots, up_z\}$ the set of URI identifying the Proxies. We define $\eta_P : UP \rightarrow R$ to be the restriction $\eta|_P$ where $P \subset R$ is the set of Proxies. Proxies allow us to define relationships between Aggregated Resources. We indicate with $\varphi_{PAR} : P \rightarrow AR$ a function which maps a Proxy to the Aggregated Resource for which it is a Proxy and with $\varphi_{PA} : P \rightarrow A$ a function which maps a Proxy to the Aggregation in which it is a Proxy.

The *Nested Aggregations* feature enables the definition of Aggregations of Aggregations; this is consistent in the OAI-ORE data model because an Aggregation is a Resource which can also be seen as an Aggregated Resource of another Aggregation. Thanks to this feature, an order exists between Aggregations, call it $\prec_a$; more formally: for all $a_i, a_j \in A$ we say that $a_i \prec_a a_j$ if and only if the Aggregation a_i is aggregated by a_j. It is important to notice that $\prec_a$ cannot define any orders between any OAI-ORE entities other than Aggregations; in fact, to define an order between Aggregated Resources we must use Proxies. Now, we can summarize the concept of *OAI-ORE Data Model* thanks to the next definition.

Definition 10. *Let $\mathcal{E} = \{A, R, AR, P, UA, UR, UAR, UP\}$ be the collection of OAI-ORE entity sets and $\Phi = \{\eta_A, \eta_{RM}, \eta_{AR}, \eta_P, \varphi_{RMA}, \varphi_{PAR}, \varphi_{PA}\}$ be the set of OAI-ORE functions. We define $O = \langle \mathcal{E}, \Phi \rangle$ to be an OAI-ORE Data Model.*

[14]Please note that the definition of the sets UA, URM, UAR is a mere convention to indicate URIs pointing to different kinds of resources in OAI-ORE and they do not stand for different kinds of URIs [101].

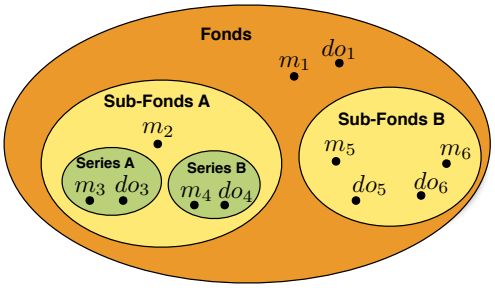


Figure 17: A sample archive containing metadata and digital objects modeled and represented by means of the NS-M.

In order to model an archive by means of OAI-ORE we need a methodology to identify the archival resources and to express the relationships between them. We have seen that we can represent a tree by means of the NS-M and that an archive can be modeled by means of a tree as well as by a NS-C. Therefore, we can model an archive throughout OAI-ORE by starting from its representation in the NS-M. We need to define a mapping between a NS-C C and an OAI-ORE model $O = \langle \mathcal{E}, \Phi \rangle$; in order to do this we have to take into account the two main entities of NESTOR which are: the sets and the elements (i.e. resources) belonging to them.

The intuitive idea is that every set $H \in C$ becomes an Aggregation $a_h \in A$ and consequently, every resource $r_t \in R$ belonging to H becomes an aggregated resource $ar_t \in AR$ aggregated by a_h. Furthermore, for every pair of sets $\{H, K\} \in C$ such that $H \subseteq K$, it is possible to create a pair of aggregations $\{a_h, a_k\} \in A$ such that $a_h \prec_a a_k$ where $\prec_a$ is the order relation defined above.

Every set in a collection of subsets can be mapped into an Aggregation in the OAI-ORE model; the inclusion order between the sets is maintained by the relation defined between the Nested Aggregations of OAI-ORE. Then, by means of the function φ_{RMA} a Resource Map is associated with each Aggregation. Every resource belonging to a set H in the NS-C is mapped into Aggregated Resources belonging to the Aggregation mapped from H. Therefore, we can map a NS-C into a correspondent OAI-ORE model and be sure that the hierarchical dependencies are properly retained. This means that if we model an archive through a NS-C then we define an OAI-ORE instance of the archive which retains the original hierarchical structure of the archive and the archival bond of archival resources.

The presented formal basis guarantees that an archive modeled by means of the NS-M can be mapped into an instance of the OAI-ORE Data Model, thus retaining the fundamental archival hierarchy. In this section we show how we can define different kinds of relationships between the resources; furthermore, we show how a proper use of Proxies can preserve the order between the resources within the same archival division. It is worthwhile to provide a concrete example of how this formal basis can be applied to a sample archive modeled by the NS-M by describing the mapping methodology step-by-step with the help of some mapping tables.

Let us take into account the sample archive represented in Figure 17; this archive is composed of five archival divisions – i.e. one fonds, two sub-fonds and two series – each containing metadata and digital objects. In NS-M these divisions are represented by means of five sets and the hierarchical relationships are retained by means of the inclusion dependencies between the sets. In Table A we can see the mapping of the sets into the OAI-ORE Aggregations and in Table B we can see how the inclusion dependencies are mapped into Nested Aggregations. These two mappings show us how to represent the structure of a sample archive in an instance of the

Sets	Aggregations
fonds	a_1
subFondsA	a_2
subFondsB	a_3
seriesA	a_4
seriesB	a_5

Table A
Mapping of sets into aggregations

Nested Sets	Nested Aggregations
subFondsA $\subset$ fonds	$a_2 \prec_a a_1$
subFondsB $\subset$ fonds	$a_3 \prec_a a_1$
seriesA $\subset$ subFondsA	$a_4 \prec_a a_2$
seriesB $\subset$ subFondsA	$a_5 \prec_a a_2$

Table B
Mapping of nested sets into nested aggregations

Elements	Aggregated Resources
m_1	ar_a
do_1	ar_b
m_2	ar_c
m_3	ar_d
[...]	[...]
do_6	ar_m

Table C
Mapping of elements into aggregated resources

Elements and Sets	Aggregations and Aggregated Resources
$m_1 \in$ fonds	a_1 aggregates ar_a
$do_1 \in$ fonds	a_1 aggregates ar_b
$m_2 \in$ subFondsA	a_2 aggregates ar_c
$m_3 \in$ seriesA	a_4 aggregates ar_d
[...]	[...]
$do_6 \in$ subFondsB	a_3 aggregates ar_m

Table D
Mapping of the elements belonging to sets into aggregated resources belonging to aggregations

Aggregated Resources	Proxies
ar_a	p_a
ar_b	p_b
ar_d	p_d
ar_e	p_e
[...]	[...]
ar_m	p_m

Table E
Proxies for the aggregated resources

p_a isMetadataOf p_b
p_d isMetadataOf p_e
[...]
p_l isMetadataOf p_m

Table F
The use of property "isMetadataOf"

OAI-ORE data model.

Each set in the NS-C contains several elements which are metadata or digital objects. For instance, the set "fonds" contains two elements: a metadata (i.e m_1) and an associated digital object (i.e. do_1). The set "sub-fondsA" contains only a metadata (i.e. m_2), the set "seriesA" contains a metadata (i.e m_3) and an associated digital object (i.e. do_3), and so on and so forth. In Table C we can see how the elements are mapped into Aggregated Resources and in Table D how the Aggregated Resources are associated with the correct Aggregations. We can see that an element belonging to a set – e.g. $m_2 \in$ `subfondsA` – is mapped into an Aggregated Resource – e.g. ar_c – aggregated by the Aggregation a_2 which corresponds to the set `subfondsA`. Table E and Table F show how we can use Proxies to associate the metadata with the digital objects they describe. OAI-ORE allows us to define different kinds of relationships between the Aggregated Resources using the Proxies. For instance, in Table F we can see that two Proxies p_a and p_b associated to ar_a and ar_b respectively are related by the relationship "`isMetadataOf`"; thus, throughout p_a and p_b we can say that the Aggregated Resource ar_a is a metadata describing the digital object ar_b. The relationships between the Aggregated Resources can reflect the order between the archival descriptions within a common archival division; in this way, we are sure that the OAI-ORE representation of the archive respects the original order principle. We can see that within this methodology it is quite simple to extend the range of the relationships connecting the Aggregated Resources and to define in this way new semantic associations between the archival resources.

In Figure 18 we can see the RDF graph representing the OAI-ORE instance of the sample archive in Figure 17. In this figure we represent the Aggregations, the Aggregated Resources and the Proxies associated to a_1; for readability we have omitted showing the other Proxies and the Resource Maps. This methodology makes it possible to model and describe the archives from scratch by means of OAI-ORE while allowing archivists to easily express relationships between archival metadata and digital objects. Archival principles are preserved and still have primary importance for understanding archival resources; at the same time, OAI-ORE offers the possibility of defining new relationships between the resources enabling the definition of new

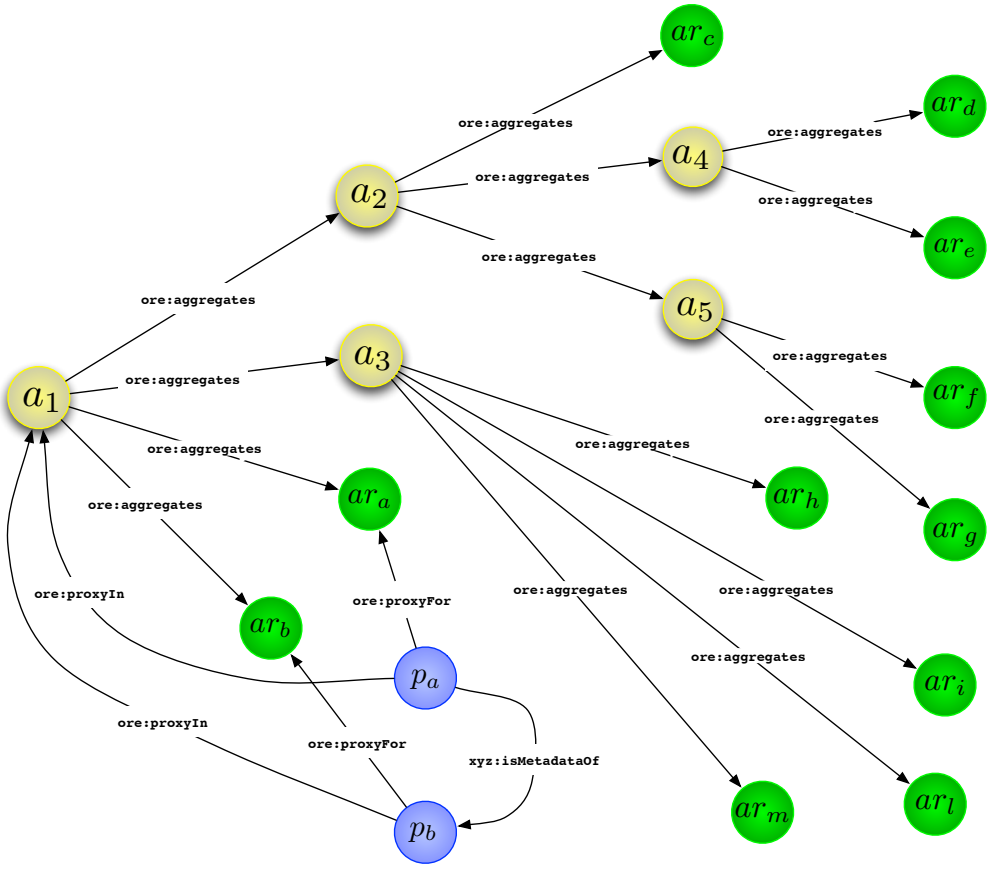


Figure 18: An instance of OAI-ORE which models a sample archive composed of descriptive metadata and related digital objects.

services over the archives. This methodology provides a means to define archival compound objects that can be shared with the systems which already employ OAI-ORE and that can be exposed as LOD on the Web. Lastly, in Figure 18 we can see an alternative to the state-of-the-art solutions depicted in Figure 5 on page 11 which overcome the presented issues and at the same time add more flexibility and expressive power to the archives.

Lastly, this methodology and the described formal basis guarantee the backward compatibility with other archival descriptive standards; for instance, a methodology to map the archival descriptions modeled by OAI-ORE into EAD can be easily defined. Indeed, we know how to map EAD into a NS-C and a NS-C into an instance of the OAI-ORE data model. In the same way, we can map the archival descriptions modeled by OAI-ORE into an EAD file by reversing the presented methodology[15]. In this context, the formal basis of NESTOR acts as an interoperability layer between EAD and OAI-ORE and guarantees the possibility of going from one

[15]Note that the backward compatibility can be limited by the fact that the EAD expressive power is inferior to that of OAI-ORE.

model to the other.

9. Use Case: Socializing the Archives

Archives need to take into account the end-user who is going to consult them and does not have the competencies and the experience to properly interpret archival data and to use finding aids. Annotations are a valuable and well-known means for collaboration which can help in socializing the archives by opening them to the general public and by helping the interpretation of information [1, 4, 9]. This is also in line with the current tendencies in Web 2.0 where available resources can be augmented with user-generated contents which then provide alternative access points for searching and browsing resources.

The goal of this use case is to show how NESTOR can be employed as a bridge between archival theory and practice and the model used for annotations. NESTOR is exploited both for the theoretical basis it defines and for the alternative representations of archives it provides. Indeed, from the representation of the INS-M it is possible to generate original visualizations of archival data enriched with annotations and to exploit this to enrich user experience, e.g. when getting rid of the results of a search involving annotations and archival data.

To this purpose we rely on NESTOR and on the Flexible Annotation Semantic Tool (FAST) annotation model [7] for handling archival and annotation aspects and for showing how annotations can be enclosed in the "NESTOR view" of the archives. The formal integration between NESTOR and FAST is described in [36] and it is not presented here, because on the one hand, it requires a deep understanding of the FAST model which is out of the scope of this article, and on the other hand, it follows a formal methodology close to the one presented in the previous section for relating NESTOR and OAI-ORE.

Beyond formally modeling what an annotation is, FAST introduces a full range of operators that allow users to either search and retrieve annotations on the basis of their content or to search and retrieve annotated resources on the basis of the annotations over them [5, 6, 32]. To this end, FAST makes use of the extended boolean model [90] to allow for mixing exact and best match queries and explicitly takes into consideration the hypertext existing between annotations and annotated resources in order to modify the scores of and rank annotations and/or annotated resources according to the paths connecting them. All the search operators and modifiers are exposed via a simple query language based on Contextual Query Language (CQL) [78], developed and maintained by the Library of Congress in the context of the Z39.50 Next Generation (ZING) project and suitable to be embedded in HTTP requests and Web services.

For example, it is possible to express queries like the following one

```
fast.annotation.text =/thread=halfThread "illuminated manuscript"
  and/match=looseMatch
fast.annotation.author.identifier =/thread=halfThread ferro
```

which searches for annotations about `illuminated manuscripts` and authored by the user `ferro`, where the former is a best match (search engine-like) search clause while the latter one is an exact match (database-like) search clause. The two clauses are mixed with a relaxed boolean operator (`looseMatch` modifier), meaning that annotations matching only one of them will be retrieved even if ranked lower than annotations matching both of them. Moreover, not only the content of the annotation is taken into account but also the hypertext of annotations (`halfThread` modifier) contributes to the final result list, meaning that if, for example, an annotation a_j talking about `illuminated manuscripts` annotates another annotation a_i, also a_i will be part of the

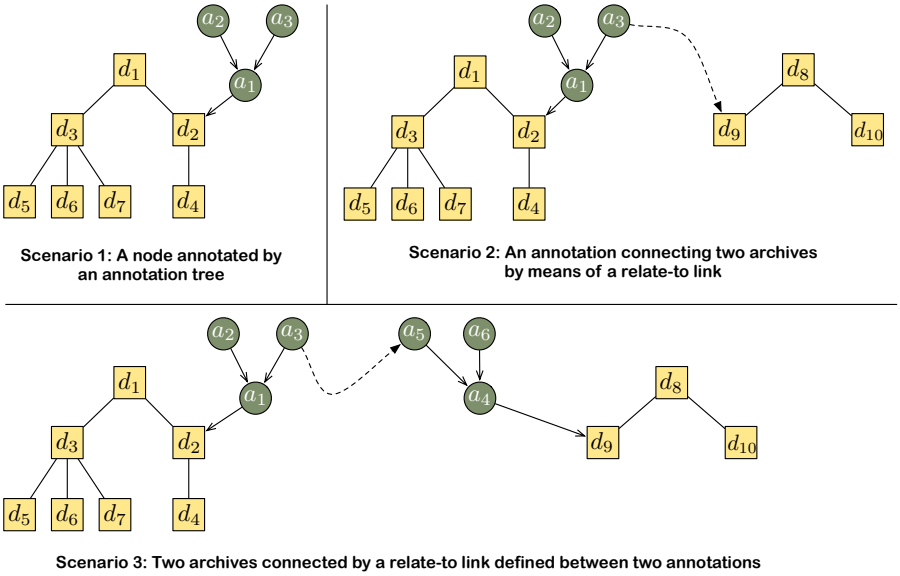


Figure 19: Annotations: Three possible scenarios in the archival context.

final result list, even if ranked lower. Similar mechanisms apply to search and retrieve annotated resources on the basis of their annotations.

Therefore, here we focus on how to exploit FAST and NESTOR to enhance the user experience when searching for annotations and annotated archival data and to set the ground to model and study the alternatives we can exploit to better access and visualize annotated archival resources.

Figure 19 presents three possible scenarios; in this figure an archive is represented as a document tree where the nodes are named as "$d_1, d_2, \ldots$" for convenience; for the same reasons annotations are indicated as "$a_1, a_2, \ldots$". In the first scenario we consider an archival tree where the node d_2, annotated by a_1, is the root of an annotation tree composed of three annotations. The second scenario shows that a_3 which is part of an annotation tree annotating d_2 is connected to a second archive by means of a "relate-to" link[16]. In the third scenario, we can see two archives connected by a relate-to link defined between two annotations – i.e. a relate-to link between a_3 and a_5.

Suppose now the user has issued a query which retrieves the following resources: a_2, d_9, and a_5. How can we better serve these results to the user in the three above scenarios in order to make him easily grasp their overall context? We present three possible scenarios showing how annotation trees can be attached to an archive and then we show how they can be modeled through the INS-M and represented by means of the DocBall, as shown in Figure 20.

In the first scenario we need to join an "archival DocBall" representing the archive and an "annotation DocBall" representing the annotation tree originally attached to node d_2 of the archive – see Figure 19a. The resulting DocBall is shown in Figure 20a, where a_1 is a superset of d_2. The

[16] A "relate-to" link is different from the other links because it relates two different archival or annotation trees [7].

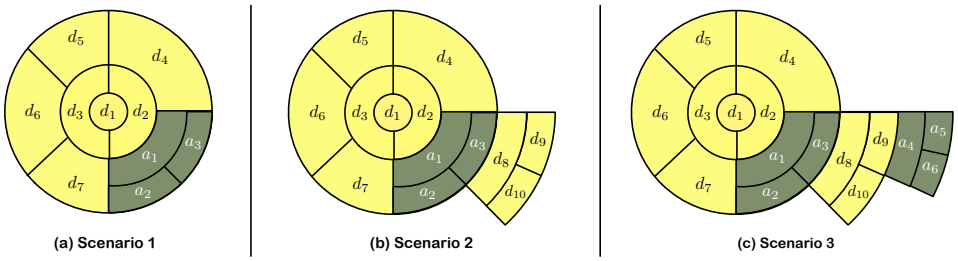


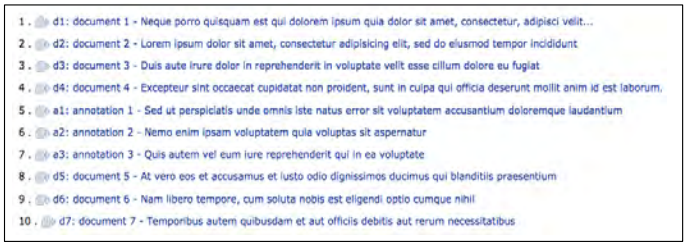
Figure 20: DocBall representations of archive annotation scenarios.

second scenario presents the same annotated archive we have seen in the first scenario enriched by the relationship of annotation a_3 with the node d_9 of a second archive. In this case, we use a DocBall representing the first archive within its annotations – call it "DocBall A" (see Figure 20a) – and a DocBall representing the second archive – call it "DocBall B". In order to join these two DocBalls connected by annotation a_3, we add the inner sector of DocBall B – i.e. d_8 – to DocBall A as a superset of a_3. The resulting DocBall (see Figure 20b) provides us with an integrated view of the two archives connected by the annotation tree rooted in a_1. The third scenario enhances this idea; indeed, in this case both "DocBall A" and "DocBall B" represent annotated archives that have to be joined together. We follow the methodology presented for scenario 2 by taking the inner sector of DocBall B – i.e. d_8 which represents the root of the second archive – and adding it to DocBall A as a superset of the annotation – in this case a_3 – which relates the two archives to each other. The general methodology of joining two DocBall can be summarized as follows; let D_A and D_B be two DocBall, where section s_A of D_A is *related to* section s_B of D_B. To join D_A with D_B, the inner section of D_B must be added to D_A as a superset of s_A.

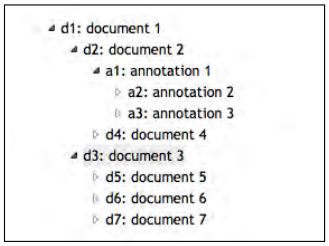
We can use the alternatives representations of Figure 19 to devise different strategies to represent search results involving annotations and archival data in order to understand what is the most suitable according to the user needs, various user categories, and the performed tasks. In this context, the two models, NESTOR and FAST, provide a sound basis which ensure to present alternatives and equivalent representation to end user, still keeping the overall coherence and possibility of passing from one to the other.

For example, Figure 21 shows a possible prototype that can be exploited to compare alternative presentation strategies: (a) provides a typical ranked list, in a Google-like fashion; (b) presents a traditional tree-like view of the archival data; (c) exploits the DocBall visualization introduced above to give an overall view of the search results.

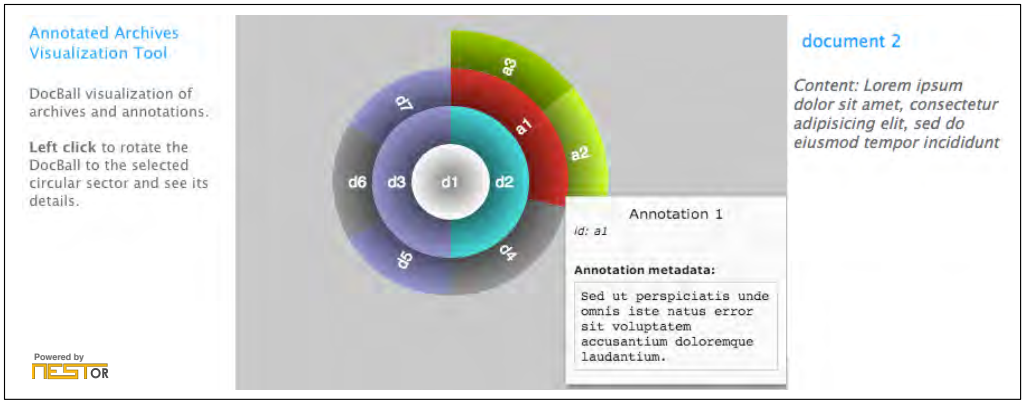
The DocBall is in the center of the canvas and when we move the pointer over a circular section a tooltip appears showing the content of this section; if we click on a section, the DocBall rotates and the selected section is highlighted. In this figure we selected section d_2 the content of which is shown in the right column and the tooltip shows the content of a_1. In this way the user can select an archival section, see its content in the right column and view the content of annotations or other archival divisions by means of the tooltip. We can see that archival documents and annotations are represented as circular sectors with different colors in the DocBall. The use of colors may be an effective way of distinguishing between the sectors which are documents and those which are annotations. Furthermore, the DocBall could become ineffective if there are many sectors that have to be represented. In this case an expand/compress strategy can be adopted as well as it is used to shows the branches of very large trees.



(a) Ranked List



(b) Tree View



(c) DocBall View

Figure 21: Three alternative search results presentation strategies: (a) Ranked List, (b) Tree View, (c) DocBal View.

This use case enables a comprehensive view of archival structure and content together with its annotations; furthermore, it highlights the relationships between different archives and how it is possible to enhance the role of annotations in the archival context and the expertise of archivists in the description as well as in the search phase within the archives. Finally, this use case represents a first-step in the direction of providing user with alternative interactive means to access archives and opens the way to further studies to understand what is the user preferred interaction style, which visualizations are most suitable for which tasks, and how these visualizations help the user in keeping the overall context of archival data and annotations.

10. Conclusion and Future Work

Building foundations and a formal theory for digital libraries is a longstanding issue in the field, dating back to mid-60s of the last century [70], and this challenge has been accepted only very recently, for example, by the 5S model [49] and the DELOS Reference model [20]. Archives are a fundamental constituent of our cultural heritage and digital libraries are the natural choice for managing and providing access to their assets.

Nevertheless, the foundational models of digital libraries have been built around the most general concepts but without specifically dealing with the peculiar features of the archives. This hampers the possibility of fully exploiting and applying them for defining a theory for digital archives, intended as digital libraries with specific characteristics that fit in the archival domain.

We think that the archival domain deserves a formal theory as well and that this theory has to be reconciled with the more general theories for digital libraries in order to disclose to archives the full breadth of methodologies and technologies which have been developed over the last two decades in the digital library field.

To this end, we have introduced an original formal model, called NESTOR, which exploits the inclusion relationships among sets as a means of representing the notions of context and hierarchy which are central to archives. Then, we extended the 5S model to introduce the notion of digital archive as a specific case of digital library complying with archival constraints. Finally, we applied this extension to three concrete use cases: (i) "detaching the archives" which is the case of interoperability between digital archives giving a concrete account of how digital library technologies can be disclosed to archives; (ii) "unchaining the archives" which shows how the archives modeled with NESTOR can form compound digital object exposed as LOD in the Web; and, (iii) "socializing the archives" which describe how NESTOR can enhance the role of annotations in the archives by helping both archivists and end-user in the description and interpretation of archival resources.

Future work will concern the formal definition of creation, deletion, update, and search operations on digital archives via NESTOR and the study of their properties.

This, in turn, will open up the possibility to further extend the 5S model. Indeed, according to this model, a minimal digital library has to offer, at least, indexing, searching and browsing services [49, p. 299]. The formal definition of the query and update operations in NESTOR will thus allow us to precisely describe what these services are in the case of digital archives.

Moreover, the formal definition of the above mentioned operations will allow us to study their computational complexity, thus characterizing their definition with upper bounds for time and space costs. This will also represent a further addition to the 5S model since, not only we will be able to express what minimal digital library services are in the case of digital archives, but we will also be able to characterize them from a performance point of view.

Therefore, we also plan to carry out extensive experimentation to assess the scalability and actual execution times of the proposed operations on real and synthetic datasets, as we have just started to explore [94]. This will then complete the formal modeling and the extension of the 5S model with experimental data.

Finally, merging this modeling effort with other existing formal models, as we did in the case of annotation, will move digital archives to the next generation, making them not only browsing and consulting tools, but active means where researchers, students, and practitioners can interact with and augment archival content with user-generated contents, tags, and annotations. This will require not only to design and develop services which exploit these joint formal models but also to conduct detailed user studies to understand which solutions are best suited for supporting information access and use tasks of different user categories. Moreover, by relying exposing archives as LOD on the Web, as in the "unchaining the archives" use case, and their connections with annotations, as in the "socializing the archives" use case, it will be possible to readily integrate in digital archives recent activities such as the Open Annotation Model [91], currently discussed by the World Wide Web Consortium (W3C), which is easily representable by means of the FAST annotation model as well.

Acknowledgements

The authors would really like to express their gratitude to Maristella Agosti for her continued support and contributions to this work which benefited from her precious suggestions and valu-

able feedback. Sincere thanks are due to Floriana Esposito and Carlo Meghini for their comments on the main ideas of the model. We would like to thank Fausta Bressani and Andreina Rigon of the Cultural Heritage Directorate (*Direzione Beni Culturali*) of the Italian Veneto Region (*Regione del Veneto*) and Erilde Terenzoni and Cristina Tommasi of the Archival Supervising Office for the Italian Veneto Region (*Soprintendenza Archivistica per il Veneto*) of the Ministry of Cultural Heritage.

Least but not last, thanks and appreciation are due to the anonymous reviewers who really helped us in increasing the quality of the paper to make its contributions more exploitable in a multi-disciplinary context.

The work reported has been carried out in the context of an agreement between the Italian Veneto Region[17] and the University of Padua. The CULTURA[18] (contract no. 269973) and the PROMISE network of excellence[19] (contract n. 258191) projects, as part of the 7th Framework Program of the European Commission, have partially supported the reported work.

Authors' Vitae

Nicola Ferro is assistant professor in Computer Science at the Department of Information Engineering of the University of Padua, Italy. His main research interests are digital libraries and archives, their architectures, interoperability, and evaluation, as well as multilingual information access and its evaluation. He is chair of the Steering Committee of CLEF initiative and he coordinates the PROMISE FP7 network of excellence on experimental evaluation of multilingual and multimodal information access system. He has published more than 90 papers on digital library architectures, interoperability and services; multilingual information access and its experimental evaluation; the management of the scientific data.

Gianmaria Silvello is a post-doc researcher at the Department of Information Engineering of the University of Padua, Italy since 2011 and he hold a Ph.D. in Information Engineering of the Doctorate School in Information Engineering of University of Padua. Since 2006 he has been working on the design and development of a digital archive system called SIAR (Regional Archival Information System) in cooperation with the Italian Veneto Region and the Archival Supervising Office for the Italian Veneto Region of the Italian Ministry of Cultural Heritage. Since 2010 he has been working on the field of Information Retrieval Evaluation within the PROMISE European network of excellence.

References

[1] Agosti, M., Bonfiglio-Dosio, G., Ferro, N., November 2007. A Historical and Contemporary Study on Annotations to Derive Key Features for Systems Design. International Journal on Digital Libraries (IJDL) 8 (1), 1–19.

[2] Agosti, M., Esposito, F., Ferilli, S., Ferro, N. (Eds.), 2013. Digital Libraries and Archives - Proc. 8th Italian Research Conference (IRCDL 2012). Communications in Computer and Information Science (CCIS) 354, Springer, Heidelberg, Germany.

[3] Agosti, M., Esposito, F., Thanos, C. (Eds.), 2010. Digital Libraries - 6th Italian Research Conference, IRCDL 2010. Revised Selected Papers. Vol. 91 of Communications in Computer and Information Science. Springer, Heidelberg, Germany.

[17] `http://www.regione.veneto.it/`

[18] `http://www.cultura-strep.eu/`

[19] `http://www.promise-noe.eu/`

[4] Agosti, M., Ferro, N., 2003. Annotations: Enriching a Digital Library. In: Koch, T., Sølvberg, I. T. (Eds.), Proc. 7th European Conference on Research and Advanced Technology for Digital Libraries (ECDL 2003). Lecture Notes in Computer Science (LNCS) 2769, Springer, Heidelberg, Germany, pp. 88–100.

[5] Agosti, M., Ferro, N., 2005. Annotations as Context for Searching Documents. In: Crestani, F., Ruthven, I. (Eds.), Proc. 5th International Conference on Conceptions of Library and Information Science – Context: nature, impact and role (CoLIS 5). Lecture Notes in Computer Science (LNCS) 3507, Springer, Heidelberg, Germany, pp. 155–170.

[6] Agosti, M., Ferro, N., 2006. Search Strategies for Finding Annotations and Annotated Documents: the FAST Service. In: Legind Larsen, H., Pasi, G., Ortiz-Arroyo, D., Andreasen, T., Christiansen, H. (Eds.), Proc. 7th International Conference on Flexible Query Answering Systems (FQAS 2006). Lecture Notes in Artificial Intelligence (LNAI) 4027, Springer, Heidelberg, Germany, pp. 270–281.

[7] Agosti, M., Ferro, N., 2008. A Formal Model of Annotations of Digital Content. ACM Transactions on Information Systems (TOIS) 26 (1), 3:1–3:57.

[8] Agosti, M., Ferro, N., Fox, E. A., Gonçalves, M. A., December 2007. Modelling DL Quality: a Comparison between Approaches: the DELOS Reference Model and the 5S Model. In: Thanos, C., Borri, F., Launaro, A. (Eds.), Second DELOS Conference - Working Notes. `http://146.48.87.21:80/OLP/UI/1.0/Disseminate/12166605710ZKGwQOHOi/a221216660571SYkbMk3C`.

[9] Agosti, M., Ferro, N., Frommholz, I., Thiel, U., 2004. Annotations in Digital Libraries and Collaboratories – Facets, Models and Usage. In: Heery, R., Lyon, L. (Eds.), Proc. 8th European Conference on Research and Advanced Technology for Digital Libraries (ECDL 2004). Vol. 3232 of Lecture Notes in Computer Science. Springer, Heidelberg, Germany, pp. 244–255.

[10] Agosti, M., Ferro, N., Silvello, G., 2009. Access and Exchange of Hierarchically Structured Resources on the Web with the NESTOR Framework. In: Baeza-Yates, R., Berendt, B., Bertino, E., Lim, E.-P., Pasi, G. (Eds.), Proc. 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology. IEEE Computer Society, Los Alamitos, CA, USA, pp. 659–662.

[11] Agosti, M., Ferro, N., Silvello, G., 2010. The NESTOR Framework: Manage, Access and Exchange Hierarchical Data Structures. In: Martoglia, R., Bergamaschi, S., Lodi, S., Sartori, C. (Eds.), Proc. 18th Italian Symposium on Advanced Database Systems (SEBD 2010). Società Editrice Esculapio, Bologna, Italy, pp. 242–253.

[12] Agosti, M., Ferro, N., Silvello, G., 2011. How to Handle Hierarchically Structured Resources Addressing Interoperability Issues in Digital Libraries. In: Biba, M., Xhafa, F. (Eds.), Learning Structure and Schemas from Documents. Springer-Verlag, Heidelberg, Germany, pp. 17–49.

[13] Agosti, M., Jose Borbinha, J., Kapidakis, S., Papatheodorou, C., Tsakonas, G. (Eds.), 2009. Proc. 13th European Conference on Research and Advanced Technology for Digital Libraries (ECDL 2009). Springer, Heidelberg, Germany.

[14] Aho, A., Ullman, J. D., 1992. Foundations of Computer Science. Computer Science Press, New York, New York, USA.

[15] Bell, J., Lewis, S., 2006. Using OAI-PMH and METS for exporting metadata and digital objects between repositories. Program: electronic library and information systems 40 (3), 268 – 276.

[16] Bizer, C., Heath, T., Berners-Lee, T., 2009. Linked Data - The Story So Far. International Journal on Semantic Web and Information Systems (IJSWIS) 5 (3), 1–22.

[17] Borgman, C. L., 1999. What are Digital Libraries? Competing Visions. Information Processind & Management 35 (3), 227–243.

[18] Borgman, C. L., 2003. From Gutenberg to the Global Information Infrastructure: Access to Information in the Networked World (Digital Libraries and Electronic Publishing). The MIT Press, Cambridge (MA), USA.

[19] Bountouri, L., Manolis, G., 2009. Interoperability Between Archival and Bibliographic Metadata: An EAD to MODS Crosswalk. Journal of Library Metadata 9 (1/2), 98–133.

[20] Candela, L., Castelli, D., Ferro, N., Ioannidis, Y., Koutrika, G., Meghini, C., Pagano, P., Ross, S., Soergel, D., Agosti, M., Dobreva, M., Katifori, V., Schuldt, H., December 2007. The DELOS Digital Library Reference Model. Foundations for Digital Libraries. ISTI-CNR at Gruppo ALI, Pisa, Italy, `http://www.delos.info/files/pdf/ReferenceModel/DELOS_DLReferenceModel_0.98.pdf`.

[21] Carini, P., Shepherd, K., 2004. The MARC Standard and Encoded Archival Description. Library Hi Tech 22 (1), 18–27.
URL `http://www.emeraldinsight.com/10.1108/07378830410524468`

[22] Casarosa, V., Meghini, C., Gardasevic, S., 2013. Improving On-Line Access to Archival Data. In: [2], pp. 153–162.

[23] Celko, J., 2000. Joe Celko's SQL for Smarties: Advanced SQL Programming. Morgan Kaufmann, San Francisco, California, USA.

[24] Combs, M., Matienzo, M. A., Proffitt, M., Spiro, L., 2010. Over, Under, Around, and Through: Getting Around Barriers to EAD Implementation. A publication of OCLC Research in support of the RLG Partnership.

[25] Cook, T., 1993. The Concept of Archival Fonds and the Post-Custodial Era: Theory, Problems and Solutions. Archiviaria 35, 24–37.
[26] Crestani, F., Vegas, J., de la Fuente, P., 2004. A Graphical User Interface for the Retrieval of Hierarchically Structured Documents. Inf. Process. Management 40 (2), 269–289.
[27] Discovery, E., Shaw, S., Reynolds, P., May/June 2007. Creating the Next Generation of Archival Finding Aids. D-Lib Magazine 13 (5/6).
[28] Doerr, M., Gradmann, S., Hennicke, S., Isaac, A., Meghini, C., Van de Sompel, H., 2010. The Europeana Data Model (EDM). In: IFLA 2011: World Library and Information Congress: 76th IFLA General Conference and Assembly. Gothenburg, Sweden.
[29] Duranti, L., 1998. Diplomatics: New Uses for an Old Science. Society of American Archivists and Association of Canadian Archivists in association with Scarecrow Press, Lanham, Maryland, USA.
[30] Europeana, October 2011. Europeana Data Model Primer. http://pro.europeana.eu/documents/900548/770bdb58-c60e-4beb-a687-874639312ba5.
[31] Europeana, February 2012. Definition of the Europeana Data Model elements – Version 5.2.3, 24/02/2012. http://pro.europeana.eu/documents/900548/bb6b51df-ad11-4a78-8d8a-44cc41810f22.
[32] Ferro, N., 2009. Annotation Search: The FAST Way. In: [13], pp. 15–26.
[33] Ferro, N., Silvello, G., July 2008. A Distributed Digital Library System Architecture for Archive Metadata. In: Agosti, M., Esposito, F., Thanos, C. (Eds.), Post-proceedings of the Forth Italian Research Conference on Digital Library Systems (IRCDL 2008). ISTI-CNR at Gruppo ALI, Pisa, Italy, pp. 99–104.
[34] Ferro, N., Silvello, G., 2008. A Methodology for Sharing Archival Descriptive Metadata in a Distributed Environment. In: Christensen-Dalsgaard et al., B. (Ed.), Proc. 12th European Conference on Research and Advanced Technology for Digital Libraries (ECDL 2008). Lecture Notes in Computer Science (LNCS) 5173, Springer, Heidelberg, Germany, pp. 268–279.
[35] Ferro, N., Silvello, G., 2009. The NESTOR Framework: How to Handle Hierarchical Data Structures. In: [13], pp. 215–226.
[36] Ferro, N., Silvello, G., 2010. FAST and NESTOR: How to Exploit Annotation Hierarchies. In: [3], pp. 55–66.
[37] Ferro, N., Silvello, G., 2011. The NESTOR Model: Properties and Applications in the Context of Digital Archives. In: Mecca, G., Greco, S. (Eds.), Proc. 19th Italian Symposium on Advanced Database Systems (SEBD 2011). Università della Basilicata, Italy, pp. 274–285.
[38] Ferro, N., Silvello, G., 2013. Empowering Archives Through Annotations. In: [2], pp. 57–68.
[39] Ferro, N., Silvello, G., 2013. Modeling Archives by means of OAI-ORE. In: [2], pp. 216–227.
[40] Ferros, L., Ramalho, J. C., Ferreira, M., 2008. Creating a National Federation of Archives Using OAI-PMH. In: Ramalho, J. C., Correia, J., Abreu, S. (Eds.), Proc. of the National Conference on XML Applications and Associated Technologies (XATA2008). pp. 3–12.
[41] Fielding, R., Gettys, Y., Mogul, J., Frystyk, H., Masinter, L., Leach, P., Berners-Lee, T., June 1999. Hypertext Transfer Protocol – HTTP/1.1. RFC 2616.
[42] Foulonneau, M., Cole, T. W., Habing, T. G., Shreeves, S. L., 2005. Using collection descriptions to enhance an aggregation of harvested item-level metadata. In: Marlino, M., Sumner, T., Shipman, F. (Eds.), Proc. 5th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL 2005). ACM Press, New York, USA, pp. 32–41.
[43] Fox, E. A., Akscyn, R. M., Furuta, R. K., Leggett, J. J., April 1995. Digital Libraries. Communications of the ACM (CACM) 38 (4), 22–28.
[44] Fox, E. A., Gonçalves, M. A., Shen, R., 2012. Theoretical Foundations for Digital Libraries: The 5S (Societies, Scenarios, Spaces, Structures, Streams) Approach. Morgan & Claypool Publishers, USA.
[45] Fox, E. A., Hix, D., Nowell, L. T., Brueni, D. J., Wake, W. C., Heath, L. S., Rao, D., September 1993. Users, User Interfaces, and Objects: Envision, a Digital Library. Journal of the American Society for Information Science (JASIS) 44 (8), 480–491.
[46] GDO, C., January 2011. CDL Guidelines for Digital Objects. Version 2.0. Tech. rep., University of California. California Digital Library.
[47] Gilliland-Swetland, A. J., 2000. Enduring Paradigm, New Opportunities: The Value of the Archival Perspective in the Digital Environment. Council on Library and Information Resources, Washington, DC, USA.
[48] Gonçalves, M. A., Fox, E. A., Watson, L. T., April 2008. Towards a digital library theory: a formal digital library ontology. International Journal on Digital Libraries 8 (2), 91–114.
[49] Gonçalves, M. A., Fox, E. A., Watson, L. T., Kipp, N. A., April 2004. Streams, Structures, Spaces, Scenarios, Societies (5S): A Formal Model for Digital Libraries. ACM Transactions on Information Systems (TOIS) 22 (2), 270–312.
[50] Gonçalves, M. A., Watson, L. T., Fox, E. A., 2004. Towards a Digital Library Theory: A Formal Digital Library Ontology. In: Dominich, S., van Rijsbergen, C. J. (Eds.), ACM SIGIR Mathematical/Formal Methods in Information Retrieval Workshop (MF/IR 2004). http://www.dcs.vein.hu/CIR/mfir_2004.html [last visited 2007, March 23].

[51] Hagedorn, K., 2003. OAIster: a "no dead ends" OAI service provider. Library Hi Tech 21 (2), 170–181.
[52] Halmos, P. R., 1960. Naive Set Theory. D. Van Nostrand Company, Inc., New York, NY, USA.
[53] Haworth, K. M., 2001. Archival Description: Content and Context in Search of Structure. In: Pitti, D., Duff, W. M. (Eds.), Encoded Archival Description on the Internet. The Haworth Press, Inc., pp. 7–26.
[54] Hayworth, K. M., 1993. The Voyage of RAD: From the Old World to the New. Archiviaria 35, 55–63.
[55] Heath, T., Bizer, C., 2011. Linked Data: Evolving the Web into a Global Data Space. Synthesis Lectures on the Semantic Web: Theory and Technology. Morgan & Claypool Publishers, USA.
[56] Hennicke, S., Olensky, M., de Boer, V., Isaac, A., Wielemaker, J., 2011. Conversion of EAD into EDM Linked Data. In: Prediu, L., Hennicke, S., Nürnberger, A., Mitschick, A., Ross, S. (Eds.), Proc. 1st International Workshop on Semantic Digital Archives (SDA 2011) http://ceur-ws.org/Vol-801/. pp. 82–88.
[57] International Council on Archives, March 1999. ISAD(G): General International Standard Archival Description, 2nd edition. Ottawa: International Council on Archives.
[58] Jech, T., 2003. Set Theory - The Third Millenium Edition. Springer-Verlag, Berlin, Heidelberg, Germany.
[59] Kamfonas, M., October/November 1992. Recursive Hierarchies: The Relational Taboo! The Relational Journal.
[60] Kani-Zabihi, E., Ghinea, G., Chen, S. Y., 2010. Experiences with Developing a User-Centered Digital Library. IJDLS 1 (1), 1–23.
[61] Kaplan, D., Sauer, A., Wilczek, E., 2011. Archival Description in OAI-ORE. Journal of Digital Information 12 (2).
[62] Kaplan, D., Sauer, A. and Wilczek, E., 2010. Archival Description in OAI-ORE. In: OR2010: The 5th Int. Conf. on Open Repositories.
[63] Kiesling, K., 2001. Metadata, Metadata, Everywhere - But Where Is the Hook? OCLC Systems & Services 17 (2), 84–88.
[64] Knuth, D. E., 1997. The Art of Computer Programming, third edition. Vol. 1. Addison Wesley, Reading, MA, USA.
[65] Kozievitch, N. P., Torres, S. R., 2010. Describing OAI-ORE from the 5S Framework Perspective. In: Chowdhury, G., Koo, C., Hunter, J. (Eds.), The Role of Digital Libraries in a Time of Global Change. Vol. 6102 of Lecture Notes in Computer Science. Springer, Berlin Heidelberg, pp. 260–261.
[66] Lagoze, C., Van De Sompel, H., Johnston, P., Nelson, M., Sanderson, R., Warner, S., October 2008. ORE Specification – Abstract Data Model – Version 1.00. http://www.openarchives.org/ore/1.0/datamodel.
[67] Lagoze, C., Van De Sompel, H., Nelson, M., Warner, S., December 2008. The Open Archives Initiative Protocol for Metadata Harvesting – Version 2.0. http://www.openarchives.org/OAI/openarchivesprotocol.html.
[68] Lagoze, C., Van de Sompel, H., P., J., Nelson, M., Sanderson, R., Warner, S., 2008. ORE Specification - Abstract Data Model. Tech. rep., OAI.
[69] Lesk, M., 1997. Practical Digital Libraries. Books, Bytes & Bucks. Morgan Kaufmann Publishers, San Francisco, California, USA.
[70] Licklider, J. C. R., 1965. Libraries of the Future. The MIT Press, Cambridge, Massachussetts, USA.
[71] Light, M., Hyry, T., 2002. Colophons and Annotations: New Directions for the Finding Aids. The American Archivists 65, 216–230.
[72] Makri, S., Blandford, A., Cox, A. L., Attfield, S., Warwick, C., 2011. Evaluating the Information Behaviour Methods: Formative Evaluations of two Methods for Assessing the Functionality and Usability of Electronic Information Resources. Int. J. Hum.-Comput. Stud. 69 (7-8), 455–482.
[73] Marchionini, G., Maurer, H., April 1995. The Roles of Digital Libraries in Teaching and Learning. Communications of the ACM (CACM) 38 (4), 67–75.
[74] Na, G., Lee, S., 2006. A Relational Nested Interval Encoding Scheme for XML Data. In: Bressan, S., Küng, J., Wagner, R. (Eds.), Database and Expert Systems Applications. Vol. 4080 of Lecture Notes in Computer Science. Springer Berlin Heidelberg, Germany, pp. 83–92.
[75] Novak, J. D., 1990. Concept maps and Vee diagrams: two metacognitive tools to facilitate meaningful learning. Instructional Science 19 (1), 29–52.
[76] Novak, J. D., Cañas, A. J., 2008. The Theory Underlying Concept Maps and How to Construct and Use Them. Technical Report IHMC CmapTools 2006-01 Rev 2008-01, Florida Institute for Human and Machine Cognition (FI), USA.
[77] OAC Working Group, M. S. S., April 2005. OAC Best Practice Guidelines for EAD. Version 2.0. Tech. rep., 61 pages.
[78] OASIS Search Web Services Technical Committee, April 2012. searchRetrieve: Part 5. CQL: The Contextual Query Language Version 1.0. http://docs.oasis-open.org/search-ws/searchRetrieve/v1.0/searchRetrieve-v1.0-part5-cql.pdf.
[79] O'Brien, H. L., Toms, E. G., 2010. The Development and Evaluation of a Survey to Measure User Engagement. Journal of the American Society for Information Science and Technology 61 (1), 50–69.
URL http://dx.doi.org/10.1002/asi.21229

[80] Pearce-Moses, R., 2005. Glossary of Archival And Records Terminology. Society of American Archivists.
[81] Pearson, J., Buchanan, G., Thimbleby, H., 2009. Improving Annotations in Digital Documents. In: [13], pp. 429–432.
URL http://dx.doi.org/10.1007/978-3-642-04346-8_51
[82] Pitti, D. V., 1999. Encoded Archival Description. An Introduction and Overview. D-Lib Magazine 5 (11).
[83] Prom, C. J., 2002. Does EAD Play Well with Other Metadata Standards? Searching and Retrieving EAD Using the OAI Protocols. Journal of Archival Organization 1 (3), 51–72.
[84] Prom, C. J., 2003. Reengineering Archival Access Through the OAI Protocols. Library Hi Tech 21 (2), 199–209.
[85] Prom, C. J., Habing, T. G., 2002. Using the Open Archives Initiative Protocols with EAD. In: Hersh, W., Marchionini, G. (Eds.), Proc. 2nd ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL 2002). ACM Press, New York, USA, pp. 171–180.
[86] Prom, C. J., Rishel, C. A., Schwartz, S. W., Fox, K. J., 2007. A Unified Platform for Archival Description and Access. In: Rasmussen, E., Larson, R. R., Toms, E., Sugimoto, S. (Eds.), Proc. 7th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL 2007). ACM Press, New York, USA, pp. 157–166.
[87] Riley, J., Shepherd, K., 2009. A Brave New World: Archivists and Shareable Descriptive Metadata. American Archivist 72 (1), 91–112.
[88] Ross, S., 2007. Digital Preservation, Archival Science and Methodological Foundations for Digital Libraries. In: Keynote Address at the 11th European Conf. on Digital Libraries (ECDL), Budapest.
[89] Roth, J. M., 2001. Serving up EAD: An Exploratory Study on the Deployment and Utilization of Encoded Archival Description Finding Aids. American Archivist 64 (2), 214–237.
[90] Salton, G., Fox, E. A., Wu, H., November 1983. Extended Boolean Information Retrieval. Communications of the ACM (CACM) 26 (11), 1022–1036.
[91] Sanderson, R., Ciccarese, P., Van De Sompel, H., February 2013. Open Annotation Data Model – Community Draft, 08 February 2013. http://www.openannotation.org/spec/core/.
[92] Shreeves, S. L., Kaczmarek, J. S., Cole, T. W., 2003. Harvesting Cultural Heritage Metadata Using the OAI Protocol. Library Hi Tech 21 (2), 159–169.
[93] Siemens, L., Cunningham, R., Duff, W., Warwick, C., 2011. A Tale of two Cities: Implications of the Similarities and Differences in Collaborative Approaches Within the Digital Libraries and Digital Humanities Communities. LLC 26 (3), 335–348.
[94] Silvello, G., 2012. Structural and Content Queries on the Nested Sets Model. In: Ferro, N., Tanca, L. (Eds.), Proc. 20th Italian Symposium on Advanced Database Systems (SEBD 2012). Edizioni Libreria Progetto, Padova, Italy, pp. 283—288.
[95] Society of American Archivists, 2003. Encoded Archival Description: Tag Library, ver. 2002. Society of American Archivists, http://www.loc.gov/ead/tglib/.
[96] Stasinopoulou, T., Bountouri, L., Kakali, C., Lourdi, I., Papatheodorou, C., Doerr, M., Gergatsoulis, M., 2007. Ontology-Based Metadata Integration in the Cultural Heritage Domain. In: Goh, D., Cao, T., Slvberg, I. T., Rasmussen, E. (Eds.), Asian Digital Libraries. Looking Back 10 Years and Forging New Frontiers. Vol. 4822 of Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 165–175.
[97] Sugimoto, G., van Dongen, W., August 2009. Archival Digital Object Ingestion into Europeana (ESE-EAD Harmonization). Tech. rep., Europeana v1.0.
[98] Timms, K., Fall 2009. New Partnerships for Old Sibling Rivals: The Development of Integrated Access Systems for the Holdings of Archives, Libraries, and Museums. Archiviaria 68, 67–96.
[99] Trant, J., 2009. Emerging Convergence? Thoughts on Museums, Archives, Libraries, and Professional Training. Museum Management and Curatorship 24 (4), 369–387.
URL http://dx.doi.org/10.1080/09647770903314738
[100] Tropashko, V., 2005. Nested Intervals Tree Encoding in SQL. SIGMOD Record 34 (2), 47–52.
[101] Van de Sompel, H., Lagoze, C., August 2007. Interoperability for the Discovery, Use, and Re-Use of Units of Scholarly Communication. CTWatch Quarterly 3 (3).
[102] Van de Sompel, H., Lagoze, C., Nelson, M., Warner, S., 2002. Implementation Guidelines for the Open Archive Initiative Protocol for Metadata Harvesting - Guidelines for Harvester Implementers. Tech. rep., Open Archive Initiative, p. 6.
[103] Van de Sompel, H., Lagoze, C., Nelson, M., Warner, S., 2003. The Open Archives Initiative Protocol for Metadata Harvesting (2nd ed.). Tech. rep., Open Archive Initiative, p. 24.
[104] Vitali, S., 2010. Archival Information Systems in Italy and the National Archival Portal. In: [3], pp. 5–11.
[105] W3C, September 2006. Extensible Markup Language (XML) 1.1 (Second Edition) – W3C Recommendation 16 August 2006, edited in place 29 September 2006. http://www.w3.org/TR/xml11/.
[106] Witten, I. H., Bainbridge, D., 2003. How to Build a Digital Library. Morgan Kaufmann Publishers, San Francisco (CA), USA.
[107] Witten, I. H., Bainbridge, D., Nichols, D. M., 2009. How to Build a Digital Library, 2nd Edition. Morgan Kauf-

mann Publishers, San Francisco (CA), USA.
[108] Yako, S., 2008. Its Complicated: Barriers to EAD Implementation. American Archivist 71 (2), 456–475.
[109] Zorich, D. M., Waibel, G., Erway, R., Zorich, 2008. Beyond the Silos of the LAMs: Collaboration Among Libraries, Archives and Museums. Tech. rep., OCLC Programs and Research, Dublin, Ohio, USA. Published online at: http://www.oclc.org/content/dam/research/publications/library/2008/2008-05.pdf.

Appendix A. NESTOR Properties: Proofs

This appendix reports the proof of theorems and propositions presented in Section 4.

Appendix A.1. Properties of the Nested Sets Model

The following is the proof of Proposition 1 on page 17 which proves that every set in a NS-C has at most one direct superset.

Proof. Ab absurdo suppose that $\exists H \in C$ such that $|\mathcal{D}^-(H)| > 1 \Rightarrow \exists K, L \in \mathcal{D}^-(H) \mid H \subseteq K \wedge H \subseteq L \wedge L \nsubseteq K \wedge K \nsubseteq L \Rightarrow K \cap L = H \Rightarrow C$ is not a NS-C (condition 4.2 of Definition 1). □

The following is the proof of Corollary 2 on page 18. This corollary proves that if we consider the collection of supersets of a set in a NS-C (say H), the set with minimum cardinality is the direct superset of H.

Proof. We know from Proposition 1 that $|\mathcal{D}^-(H)| \leq 1$. Then, ab absurdo suppose that $\forall L \in \mathcal{S}^-(H), |K| \leq |L|$ and that $\exists W \in \mathcal{S}^-(H) \mid ((|W| > |K|) \wedge (\mathcal{D}^-(H) = W))$. This means that $H \subseteq W \wedge H \subseteq K$ and by definition of NS-M $W \subseteq K \vee K \subseteq W$. If $W \subseteq K \Rightarrow |W| < |K|$; if $K \subseteq W \Rightarrow |K| < |W|$. So if $\mathcal{D}^+(H) = W \Rightarrow |W| < |K|$. □

The following is the proof of Proposition 3 on page 18 which says that the direct subsets of a set in a NS-M are always disjoints.

Proof. Ab absurdo suppose that $K \cap L \neq \emptyset \Rightarrow K \cap L = W$ such that $|W| \geq 1 \wedge W \nsubseteq K \wedge W \nsubseteq L \Rightarrow C$ is not a NS-C. □

Appendix A.2. Properties of the Inverse Nested Sets Model

The following is the proof of Proposition 4 on page 18 showing the behaviour of union and set difference in the INS-M under specific conditions.

Proof. Let us prove property 4.5. ($\Rightarrow$). Ab absurdo suppose that $((H \nsubseteq K) \wedge (K \nsubseteq H)) \Rightarrow H \cup K = L \in C$. This means that $H \subseteq L \wedge K \subseteq L$. Without any loss of generality, let us take into account $H \subseteq L$; in this case $\exists K \in C \mid ((H \nsubseteq K) \wedge (K \nsubseteq H))$ but $L \cap K = K \neq H \cap K \Rightarrow C$ is not an INS-C.

($\Leftarrow$). Ab absurdo suppose that $H \cup K = L \notin C \Rightarrow H \subseteq K \vee K \subseteq H$. $H \subseteq K \Rightarrow H \cup K = K \in C \wedge K \subseteq H \Rightarrow K \cup H = H \in C$.

Let us prove property 4.6. Ab absurdo suppose that $H \setminus K = L \in C$.

If $(H \setminus K = L \in C) \wedge K \subseteq H \Rightarrow L \cap K = \emptyset \wedge L \cap H = L \Rightarrow ((L \nsubseteq K) \wedge (K \nsubseteq L)) \Rightarrow \exists H, K, L \in C \mid K \subseteq H \wedge ((L \nsubseteq K) \wedge (K \nsubseteq L)) \wedge (K \cap L \neq L \cap H) \Rightarrow C$ is not an INS-C.

If $(H \setminus K = L \in C) \wedge ((H \nsubseteq K) \wedge (K \nsubseteq H)) \Rightarrow L \subseteq H \wedge ((L \nsubseteq K) \wedge (K \nsubseteq L)) \Rightarrow H \cap K \neq \emptyset \neq L \cap K = \emptyset \Rightarrow C$ is not an INS-C. □

The following proof shows that the sets in an INS-C verify the properties defined by Prposition 5 on page 18.

Proof. Ab absurdo suppose that $\exists H \in C$ such that $|\mathcal{D}^+(H)| > 1 \Rightarrow \exists K, L \in \mathcal{D}^+(H), K \neq L \mid L \subseteq H \vee K \subseteq H$. This means that $L \nsubseteq K \wedge K \nsubseteq L$ because $L, K \in \mathcal{D}^+(H)$ and $L \cap K \neq H \cap L$ thus C in not an INS-C because it violates condition 4.4 of Definition 2. □

The following proves corollary 6 to the precedent proposition and shows that for all $H \in C$, the set with maximum cardinality in the collection of subsets of H is its direct subset.

Proof. We know from Proposition 5 that $|\mathcal{D}^+(H)| \leq 1$. Then, ab absurdo suppose that $\forall L \in \mathcal{S}^+(H), |K| \geq |L|$ and that $\exists W \in \mathcal{S}^+(H)$ such that $|W| < |K| \wedge \mathcal{D}^+(H) = W$. This means that $W \subset H \wedge K \subset H$. If $|W| < |K| \Rightarrow W \subset K \Rightarrow W \subset K \subset H \Rightarrow \mathcal{D}^+(H) \neq W$. □

The next proof (Proposition 7 on page 18)shows that the intersection between H and K is the set with maximum cardinality among all of their common subsets.

Proof. Ab absurdo suppose that $H \cap K = L$ and that $\exists W \in \mathcal{D}^+(H) \cap \mathcal{D}^+(K), W \neq L \Rightarrow |W| > |L| \Rightarrow (W \subseteq (H \cap K)) \wedge (L \subseteq (H \cap K)) \Rightarrow L \subseteq W \Rightarrow H \cap K = W$. □

Appendix A.3. Equivalence Between the NS-M and INS-M

The following proof of Theorem 8 on page 19 shows that NS-M and INS-M have the same expressive power by proving that if we apply the function ζ to a NS-C we obtain an INS-C as output.

Proof. To prove that $\zeta(C) = \mathcal{E}$ is an INS-C we have to verify if it satisfies the two conditions of Definition 2.

Condition (4.3). By the definition of NS-C we know that $\exists! A \in C \mid \forall H \in C, H \subseteq A$. We know that $\mathcal{S}^-(A) = \emptyset$, that $\forall H \in C, H \neq A, \mathcal{S}^-(H) \neq \emptyset$; we call $B = \zeta(A) = \bigcup(A \setminus \mathcal{D}^+(A))$. $\forall H \in C, H \neq A, \mathcal{D}^+(H) \subset \mathcal{D}^+(A) \Rightarrow \mathcal{S}^-(H) \neq \emptyset \Rightarrow \zeta(A) \subset \zeta(H) \Rightarrow \forall K \in \mathcal{E}, B \subseteq K$.

Condition (4.4). Let us consider three sets $H, K, L \in C$ such that $\zeta(H) = H' \in \mathcal{E}, \zeta(K) = K' \in \mathcal{E}, \zeta(L) = L' \in \mathcal{E}$.

Ab absurdo suppose that $\forall H', K', L' \in \mathcal{E} \mid H' \subseteq K', L' \neq K' \Rightarrow (L' \cap K' \neq H' \cap L') \wedge (H' \nsubseteq L') \wedge (L' \nsubseteq H')$. This means that, $(H' \nsubseteq L') \wedge (L' \nsubseteq H') \Rightarrow (L \| H)$. $(L' \cap K' \neq H' \cap L') \Rightarrow \nexists V' \in \mathcal{E} \mid L' \cap K' = V' = H' \cap L' \Rightarrow \nexists V' \in \mathcal{E} \mid V' \subseteq H' \wedge V' \subseteq K' \wedge V' \subseteq L' \Rightarrow \nexists V \in C \mid L \subseteq V \wedge K \subseteq V \wedge H \subseteq V \Rightarrow C$ is not a NS-C. □

Let us see an example showing how the ζ function can be applied to the sample NS-C shown on the left-hand side of Figure 10 on page 19; in Figure A.22 we can see each step of this mapping procedure.

Example 5. *Let C be a NS-C and let $C = \{A, B, C, D, E\}$ where $A = \{a, b, c, d, e, f, g\}$, $B = \{b, g\}$, $C = \{c, d, e\}$, $D = \{d\}$ and $E = \{e\}$. Then $\zeta(C) = \mathcal{E} = \{A', B', C', D', E'\}$, where:*

$\zeta(A) = A' = \bigcup_{H \in \{A \cup \mathcal{S}^-(A)\}}(H \setminus \bigcup \mathcal{D}^+(H)) = A \setminus \bigcup\{B, C\} = \{a, b, c, d, e, f, g\} \setminus \{b, c, d, e, g\} = \{a, f\}$ *(step 1 of Figure A.22).*

$\zeta(B) = B' = \bigcup_{H \in \{B \cup \mathcal{S}^-(B)\}}(H \setminus \bigcup \mathcal{D}^+(H)) = (B \setminus \{\emptyset\}) \cup (A \setminus \bigcup\{B, C\}) = \{b, g\} \cup \{a, f\} = \{a, f, b, g\}$ *(step 2 of Figure A.22).*

$\zeta(C) = C' = \bigcup_{H \in \{C \cup \mathcal{S}^-(C)\}}(H \setminus \bigcup \mathcal{D}^+(H)) = (C \setminus \{D, E\}) \cup (A \setminus \bigcup\{B, C\}) = \{c\} \cup \{a, f\} = \{c, a, f\}$ *(step 3 of Figure A.22).*

$\zeta(D) = D' = \bigcup_{H \in \{D \cup \mathcal{S}^-(D)\}}(H \setminus \bigcup \mathcal{D}^+(H)) = (D \setminus \{\emptyset\}) \cup (C \setminus \{D, E\}) \cup (A \setminus \bigcup\{B, C\}) = \{d\} \cup \{c\} \cup \{a, f\} = \{d, c, a, f\}$ *(step 4 of Figure A.22).*

$\zeta(E) = E' = \bigcup_{H \in \{E \cup \mathcal{S}^-(E)\}}(H \setminus \bigcup \mathcal{D}^+(H)) = (E \setminus \{\emptyset\}) \cup (C \setminus \{D, E\}) \cup (A \setminus \bigcup\{B, C\}) = \{e\} \cup \{c\} \cup \{a, f\} = \{e, c, a, f\}$ *(step 5 of Figure A.22).*

Now, let us see the proof of Theorem 9 on page 19 by showing how the ξ function allows us to map an INS-C into a NS-C.

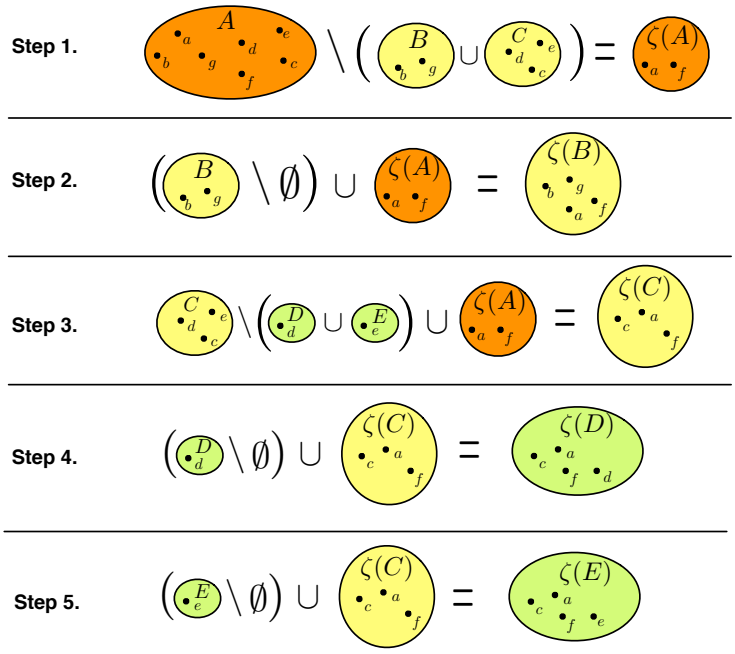


Figure A.22: From the NS-C to the INS-C through the ζ function: step-by-step.

Proof. Let us prove that $\mathcal{E}$ respects condition 4.1 of Definition 1. $\exists B \in C \mid \forall H \in C, B \subseteq H \Rightarrow \mathcal{D}^+(B) = \emptyset \wedge \mathcal{D}^+(H) \neq \emptyset \wedge (H \cup \mathcal{S}^-(H)) \subseteq (B \cup \mathcal{S}^-(B) \Rightarrow \forall H \in C, \bigcup(H \cup \mathcal{S}^-(H)) \setminus \bigcup \mathcal{D}^+(H) \subseteq \bigcup(B \cup \mathcal{S}^-(B)) \setminus \bigcup \mathcal{D}^+(B) \Rightarrow \xi(H) \subseteq \xi(B)$.

Let us prove that $\mathcal{E}$ respects condition 4.2. $\forall H, K \in C \mid H \subseteq K \Rightarrow (\bigcup \mathcal{S}^-(K) \subseteq \bigcup \mathcal{S}^-(H)) \wedge (K \in \bigcup \mathcal{S}^-(H)) \wedge (\bigcup \mathcal{D}^+(H) \subseteq \bigcup \mathcal{D}^+(K)) \Rightarrow (\bigcup(K \cup \mathcal{S}^-(K)) \setminus \bigcup \mathcal{D}^+(K)) \subseteq (\bigcup(H \cup \mathcal{S}^-(H)) \setminus \bigcup \mathcal{D}^+(H))) \Rightarrow \xi(K) \subseteq \xi(H)$.

Ab absurdo suppose that $\exists \xi(H), \xi(K) \in \mathcal{E} \mid (\xi(H) \cap \xi(K) \neq \emptyset) \wedge (\xi(H) \nsubseteq \xi(K)) \wedge (\xi(K) \nsubseteq \xi(H)) \Rightarrow \exists \xi(L) \in \mathcal{E} \mid ((\xi(L) \subseteq \xi(H)) \wedge (\xi(L) \subseteq \xi(K))) \Rightarrow \exists L \in C \mid (H \subseteq L) \wedge (K \subseteq L) \wedge (H \| K) \Rightarrow (L \cap K \neq K \cap H) \wedge (H \cap K \neq H \cap K) \Rightarrow C$ is not a INS-C. □

Example 6. *Let C be an INS-C and let $C = \{A, B, C, D, E\}$ where $A = \{a, f\}$, $B = \{a, b, f, g\}$, $C = \{c, a, f\}$, $D = \{d, c, a, f\}$ and $E = \{e, c, a, f\}$. We can see a graphical representation of this INS-C on the right side of Figure 10 and each step of the mapping procedure in Figure A.23.*

If we apply the ξ function we obtain the following result:

$\xi(A) = A' = \bigcup(A \cup \mathcal{S}^-(A)) \setminus \bigcup \mathcal{D}^+(A) = \bigcup\{A, B, C, D, E\} \setminus \emptyset = \{a, b, c, d, e, f, g\}$ (step 1 of Figure A.23).

$\xi(B) = B' = \bigcup(B \cup \mathcal{S}^-(B)) \setminus \bigcup \mathcal{D}^+(B) = \bigcup\{B\} \setminus A = \{a, f, b, g\} \setminus \{a, f\} = \{b, g\}$ (step 2 of Figure A.23.

$\xi(C) = C' = \bigcup(C \cup \mathcal{S}^-(C)) \setminus \bigcup \mathcal{D}^+(C) = \bigcup\{C, D, E\} \setminus A = \{c, a, f, d, e\} \setminus \{a, f\} = \{c, d, e\}$ (step 3 of Figure A.23.

$\xi(D) = D' = \bigcup(D \cup \mathcal{S}^-(D)) \setminus \bigcup \mathcal{D}^+(D) = D \setminus \bigcup\{C\} = \{d, c, a, f\} \setminus \{c, a, f\} = \{d\}$ (step 4 of Figure A.23).

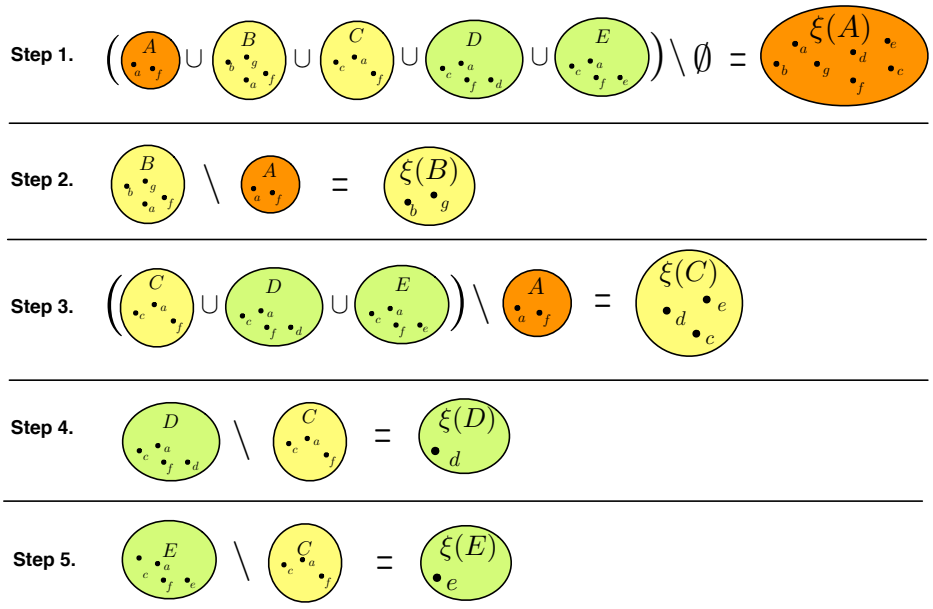


Figure A.23: From the INS-C to the NS-C through the ζ function: step-by-step.

$\xi(E) = E' = \bigcup(E \cup \mathcal{S}^-(E)) \setminus \bigcup \mathcal{D}^+(E) = E \setminus \bigcup\{C\} = \{e, c, a, f\} \setminus \{c, a, f\} = \{e\}$ *(step 5 of Figure A.23.*

Appendix A.4. Proofs of the Equivalence between the Archival Tree and NESTOR

Theorem 10 on page 20 which shows the equivalence between the tree and the NS-M is proved by the following proof.

Proof. Let us consider a bijective family of subsets[20] $\mathcal{V}_V : V \to C$ where the set of nodes V is its index set of the family and $\forall v_i \in V$, $V_{v_i} = \Gamma^+(v_i)$. Let $v_r \in V$ be the root of the tree then $V_{v_r} = \Gamma^+(v_r) = V$ and thus $V \in \{V_{v_i}\}_{v_i \in V}$ (condition 4.1, Definition 1).

Now, we prove condition 4.2 of Definition 1. Let $v_h, v_k \in V$, $h \neq k$ such that $V_{v_h} \cap V_{v_k} = \Gamma^+(v_h) \cap \Gamma^+(v_k) \neq \emptyset$, ab absurdo suppose that $\Gamma^+(v_h) \nsubseteq \Gamma^+(v_k) \ \wedge \ \Gamma^+(v_k) \nsubseteq \Gamma^+(v_k)$. This means that the descendants of v_h share at least one node with the descendants of v_k but they do not belong to the same subtree. This means that $\exists\, v_z \in V \mid d_V^-(v_z) = 2$, but then $T = (V, E)$ is not a tree. □

Example 7. *Let $T = (V, E)$ be a tree where $V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}, v_{11}\}$ and $E = \{e_{1,2}, e_{1,5}, e_{2,3}, e_{2,4}, e_{5,6}, e_{5,9}, e_{6,7}, e_{6,8}, e_{9,10}, e_{9,11}\}$, thus $\Gamma^+(v_1) = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}, v_{11}\}$, $\Gamma^+(v_2) = \{v_2, v_3, v_4\}$, $\Gamma^+(v_3) = \{v_3\}$, $\Gamma^+(v_4) = \{v_4\}$, $\Gamma^+(v_5) = \{v_5, v_6, v_7, v_8, v_9, v_{10}, v_{11}\}$,*

[20] Let A be a set, I a non-empty set and C a collection of sets of A. Then a function $\mathcal{A} : I \to C$ is defined to be a **family** of subsets of A. We call I the **index** set and we say that the collection C is **indexed** by I.

$\Gamma^+(v_6) = \{v_6, v_7, v_8\}$, $\Gamma^+(v_7) = \{v_7\}$, $\Gamma^+(v_8) = \{v_8\}$, $\Gamma^+(v_9) = \{v_9, v_{10}, v_{11}\}$, $\Gamma^+(v_{10}) = \{v_{10}\}$, *and* $\Gamma^+(v_{11}) = \{v_{11}\}$.

Let $\mathcal{V}_V$ *be a collection, where* $V = \{V_{v_1}, V_{v_2}, V_{v_3}, V_{v_4}, V_{v_5}, V_{v_6}, V_{v_7}, V_{v_8}, V_{v_9}, V_{v_{10}}, V_{v_{11}}\}$, $V_{v_1} = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}, v_{11}\}$, $V_{v_2} = \{v_2, v_3, v_4\}$, $V_{v_3} = \{v_3\}$, *and* $V_{v_4} = \{v_4\}$, $V_{v_5} = \{v_5, v_6, v_7, v_8, v_9, v_{10}, v_{11}\}$, $V_{v_6} = \{v_6, v_7, v_8\}$, $V_{v_7} = \{v_7\}$, $V_{v_8} = \{v_8\}$, $V_{v_9} = \{v_9, v_{10}, v_{11}\}$, $V_{v_{10}} = \{v_{10}\}$, *and* $V_{v_{11}} = \{v_{11}\}$. *Then, from Theorem 10 it follows that* $\mathcal{V}_V$ *is a NS-C.*

In this example the instance of the family $\mathcal{V}_V$ *specifies the nodes of the tree as elements of the sets; it is possible to see that if the nodes of the tree contain elements other than the node itself, these elements would become the content of the sets in* $\mathcal{V}_V$.

The tree $T = (V, E)$ *and the collection* $\mathcal{V}_V$ *mapped from it are represented in Figure 11.*

The following proof verifies Theorem 11 on page 20 by showing that a NS-C can be mapped into a tree by creating a node from every set in the NS-C.

Proof. We have to prove that $\exists! v_r \in V$ such that
$|E^-(v_r)| = 0 \wedge \forall v_j \in V, j \neq r, |E^-(v_j)| = 1$. Ab absurdo suppose that $\exists v_r, v_k \in V$ such that $(|E^-(v_r)| = 0 \wedge |E^-(v_k)| = 0) \vee \exists v_j \in V$ such that
$|E^-(v_j)| > 1$. If $\exists v_r, v_k \in V$ such that $|E^-(v_r)| = 0 \wedge |E^-(v_k)| = 0$ it means that both v_r and v_k have no ancestors; this means that $\exists R, K \in C \mid \mathcal{S}^-(R) = 0 \wedge \mathcal{S}^-(K) = 0$ but by the definition of NS-C we know that there is a set $T \in C$ that is the common superset of all the sets in C, then $\nexists J \in C \mid J \neq T \wedge \mathcal{S}^-(J) = 0$.

If $\exists v_j \in V$ such that $|E^-(v_j)| > 1$ this means that $\exists v_k, v_l \in V$ such that they are both parents of $v_j \Rightarrow \exists K, L \in C \mid J \subseteq K \wedge J \subseteq L \Rightarrow (L \cap K = J) \wedge (L \nsubseteq K \vee K \nsubseteq L) \Rightarrow C$ is not a NS-C. □

Now we can prove the corresponding theorems for the INS-M which show how a tree can be mapped into an INS-C and vice versa.

The mapping between a tree and an INS-C reverses the idea described for the mapping of a tree into a NS-C; if a node is a parent of another node in a tree, this is mapped into a set which is a subset of the set created from its child node. The following proof proves Theorem 12 on page 21.

Proof. Let us consider a family of subsets $\mathcal{V}_V : V \to C$ where the set of nodes V is its index set of the family and $\forall v_i \in V$, $V_{v_i} = \Gamma^-(v_i)$.

Let us prove condition 4.3 of Definition 2. Let $v_r \in V$ be the root of T. $\mathcal{V}_V(v_r) = V_{v_r} = \Gamma^-(v_r) = \{v_r\} \Rightarrow \forall v_j \in V, \Gamma^-(v_r) \subseteq \Gamma^-(v_j) \Rightarrow V_{v_r} \subseteq V_{v_j}$.

Let us prove condition 4.4 of Definition 2. Ab absurdo suppose that $\exists V_{v_k}, V_{v_h}, V_{v_l} \in \mathcal{V}_V \mid ((V_{v_h} \subseteq V_{v_k}) \wedge (V_{v_l} \nsubseteq V_{v_h}) \wedge (V_{v_h} \nsubseteq V_{v_l})) \Rightarrow V_{v_l} \cap V_{v_k} \neq V_{v_l} \cap V_{v_h}$.

This means that $\exists v_h, v_k, v_l \in V \mid ((\Gamma^-(v_h) \subseteq \Gamma^-(v_k)) \wedge (\Gamma^-(v_l) \nsubseteq \Gamma^-(v_h)) \wedge (\Gamma^-(v_h) \nsubseteq \Gamma^-(v_l))) \Rightarrow \Gamma^-(v_l) \cap \Gamma^-(v_k) \neq \Gamma^-(v_l) \cap \Gamma^-(v_h)$. $\exists v_j \in V \mid v_j \in (\Gamma^-(v_l) \cap \Gamma^-(v_k)) \wedge v_j \notin (\Gamma^-(v_l) \cap \Gamma^-(v_h)) \Rightarrow v_h \in \Gamma^-(v_k) \wedge v_j \in \Gamma^-(v_k) \wedge v_j \in \Gamma^-(v_l) \wedge v_j \notin \Gamma^-(v_h)$. This means that v_k and v_h must belong to the same branch of T; we know that $v_j \in \Gamma^-(v_l) \wedge v_j \in \Gamma^-(v_k)$, thus v_k and v_l must have v_j as a common ancestor and $v_j \notin \Gamma^-(v_h)$. This means that $\{v_j, v_k, v_l\} \in \Gamma^+(v_h)$, but $((\Gamma^-(v_l) \nsubseteq \Gamma^-(v_h)) \wedge (\Gamma^-(v_h) \nsubseteq \Gamma^-(v_l))) \Rightarrow d_V^-(v_l) > 1 \Rightarrow T$ is not a tree. □

Example 8. *Let* $T = (V, E)$ *be a tree where* $V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}, v_{11}\}$ *and* $E = \{e_{1,2}, e_{1,5}, e_{2,3}, e_{2,4}, e_{5,6}, e_{5,9}, e_{6,7}, e_{6,8}, e_{9,10}, e_{9,11}\}$, *thus* $\Gamma^+(v_1) = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8, v_9, v_{10}, v_{11}\}$, $\Gamma^+(v_2) = \{v_2, v_3, v_4\}$, $\Gamma^+(v_3) = \{v_3\}$, $\Gamma^+(v_4) = \{v_4\}$, $\Gamma^+(v_5) = \{v_5, v_6, v_7, v_8, v_9, v_{10}, v_{11}\}$,

$\Gamma^+(v_6) = \{v_6, v_7, v_8\}$, $\Gamma^+(v_7) = \{v_7\}$, $\Gamma^+(v_8) = \{v_8\}$, $\Gamma^+(v_9) = \{v_9, v_{10}, v_{11}\}$, $\Gamma^+(v_{10}) = \{v_{10}\}$, *and* $\Gamma^+(v_{11}) = \{v_{11}\}$.

Let $\mathcal{V}_V$ *be a family, where* $V = \{V_{v_1}, V_{v_2}, V_{v_3}, V_{v_4}, V_{v_5}, V_{v_6}, V_{v_7}, V_{v_8}, V_{v_9}, V_{v_{10}}, V_{v_{11}}\}$, $V_{v_1} = \{v_1\}$, $V_{v_2} = \{v_1, v_2\}$, $V_{v_3} = \{v_1, v_2, v_3\}$, *and* $V_{v_4} = \{v_1, v_2, v_4\}$, $V_{v_5} = \{v_1, v_5\}$, $V_{v_6} = \{v_1, v_5, v_6\}$, $V_{v_7} = \{v_1, v_5, v_6, v_7\}$, $V_{v_8} = \{v_1, v_5, v_6, v_8\}$, $V_{v_9} = \{v_1, v_5, v_9\}$, $V_{v_{10}} = \{v_1, v_5, v_9, v_{10}\}$, *and* $V_{v_{11}} = \{v_1, v_5, v_9, v_{11}\}$. *Then, from Theorem 12 it follows that* $\mathcal{V}_V$ *is an INS-C. The tree* $T = V, E$ *and the family* $\mathcal{V}_V$ *mapped from it are represented in Figure 12.*

Now we can prove Theorem 13 on page 21 by showing how an INS-M C is mapped into a tree $T = (V, E)$.

Proof. We have to prove that $(\exists!\ v_r \in V$ such that $|E^-(v_r)| = 0) \wedge (\forall v_j \in V, j \neq r, |E^-(v_j)| = 1)$. Ab absurdo suppose that $\exists v_r, v_k \in V$ such that
$(|E^-(v_r)| = 0 \wedge |E^-(v_k)| = 0) \vee \exists v_j \in V$ such that $|E^-(v_j)| > 1$.

If $\exists v_r, v_k \in V$ such that $((|E^-(v_r)| = 0) \wedge (|E^-(v_k)| = 0)) \Rightarrow \exists J, K \in C \mid (\mathcal{S}^-(J) \cap \mathcal{S}^-(K) = \emptyset) \Rightarrow \nexists B \in C \mid B \subseteq J \wedge B \subseteq K \Rightarrow C$ is not an INS-C.

If $\exists v_j \in V$ such that $|E^-(v_j)| > 1 \Rightarrow \exists J, K, L \in C \mid (K \subseteq J \wedge L \subseteq J \wedge K \cap L = \emptyset) \Rightarrow L \cap K = \emptyset \neq L \cap J = L \Rightarrow C$ is not an INS-C. □