Rule Categories for Collection/Item Metadata Relationships

Karen M. Wickett[1], Allen H. Renear[2], Richard J. Urban[3]
Center for Informatics Research in Science and Scholarship
Graduate School of Library and Information Science
University of Illinois at Urbana-Champaign
501 E. Daniel St, MC-493, Champaign, IL 61820-6211, USA
[1]wickett2@illinois.edu, [2]renear@illinois.edu, [3]rjurban@illinois.edu

ABSTRACT
Collections of artifacts, images, texts, and other cultural objects are not arbitrary aggregations, but are designed to support specific research and scholarly activities. Collection-level metadata directly supports this objective, providing critical contextual information. However, exploiting this information, especially in a semantic web environment of linked data, requires a precise formalization of the rules that characterize collection/item metadata relationships. Toward this end we are developing a logic-based framework of relationship rule categories for collection/item metadata. This framework will support metadata specification developers, metadata catalogers, and system designers. In earlier work we described three example rule categories for propagation of information from collections to items. Further reflection, and examination of metadata in an RDF testbed, has revealed eighteen categories, which form an interrelated system with three levels of specificity and formal constraints differentiating categories. This paper summarizes the results of a three year effort, part of the IMLS Digital Collections and Content project.

Keywords
Metadata, Digital Libraries, Collections, Ontologies, Information Organization,

INTRODUCTION
Research collections of artifacts, images, texts, and other cultural objects are not arbitrary aggregations, they are intended to support specific research and scholarly activities and are often very carefully and deliberately designed to serve that purpose (Curral, Moss & Stuart, 2004; Lee, 2000, 2005; Palmer, 2004, 2006). Collection-level metadata directly supports this objective, providing critical contextual information such as the purpose of the collection, its subject, the method of selection, size, nature of contents, coverage, completeness, representativeness, and a wide range of summary characteristics, such as statistical features (Heaney, 2000; Lagoze, et al. 2006; Palmer, 2006). Information of this sort enables collections to fulfill their distinctive role in the research process (Brockman, et al., 2001; Palmer, 2004).

Unfortunately, contemporary retrieval and browsing systems rarely exploit collection-level metadata, reducing the effectiveness of retrieval systems, and depriving researchers of the critical contextual information provided by recognizing membership in collections (Foulonneau, et al., 2005; Wendler, 2004). The problem is particularly acute in systems that aggregate descriptions from multiple sources. These systems focus solely on item-level descriptions (if they use metadata at all) and do not incorporate information available in collection-level metadata. (Christenson & Tennant, 2005; Dempsey, 2005; DLF, 2005; Foulonneau, et al., 2005; Lagoze, et al., 2006; Warner, et al., 2007). As the use of federating and metasearch search engines on the internet continues to grow the significance of this problem grows as well.

To support tools that make the most of both item and collection metadata we need a much better understanding of the kinds of logical relationships that hold between collection-level and item-level metadata attributes. Ultimately, a precise formalization of those relationships (most likely in a semantic web knowledge representation language) will be required so that they may be used by retrieval and browsing systems. While identification of the rules which govern specific metadata attributes will typically be done by persons designing or using those metadata vocabularies, we are supporting that work by developing a formal framework for classifying these relationships and for testing rules empirically against existing metadata assignments. This will allow rule developers to select the rule format that seems to match the attribute semantics they require and, when the metadata vocabulary is already in use, determine whether current practice confirms or refutes their conjectures.

Although collections have long been a consistent feature of library, museum and archival practice, they have only recently been given much theoretical attention. Studies

ASIST 2010, October 22–27, 2010, Pittsburgh, PA, USA.

have addressed their roles in scholarship (Palmer, 2004) and their general features as informational artifacts (Lee, 2000). However, there is still not a consistent understanding of collections in terms of the logical semantics of collection membership or the ontological status of collections themselves. By studying the semantic features of collection description in terms of its relationships to item description, we are opening a window on what it really means to be a collection[1].

THE CIMR FRAMEWORK

In earlier work we developed the basic strategy for a framework of rule categories and discussed some of the technical issues involved (Renear et al., 2008a). Although the potential of our approach has been noted by other metadata researchers (Greenberg, 2009; Lourdi et al., 2009), the preliminary example categories we offered cover at best only a small portion of the semantics of metadata in common use. Further analysis and an empirical examination of actual metadata assignments has led to a larger, more fine-grained framework that better matches actual metadata semantics. This framework consists of a total of 18 rule categories with three levels of specificity. The framework reveals the logical entailments among rule categories and identifies the key formal characteristics of metadata constraint relationships that provide the foundation for inferencing.

Rules and Categories

We express collection/item metadata relationships as rules, where a rule is a logical conditional, an assertion that if something is the case then something else is the case. For example, consider the metadata attribute *marcrel:own*, used to identify the owner of a resource. The relevant concept of ownership might imply the rule *if someone owns a collection then they own each item in the collection*[2]. Such a rule, if made computationally available, could obviously enhance searching, browsing, and analysis, and other metadata processing as well, such as validation and selection.[3]

Identifying specific metadata rules is most appropriately carried out by domain specialists. Rules might be asserted by metadata specification designers as part of defining a metadata schema, or by metadata cataloging supervisors as part of a local policy that specializes a metadata vocabulary. Rules may also be asserted by systems managers to characterize data in the databases they support. Currently the assertion of rules is usually informal and implicit, latent in the natural language prose of specification scope notes and local instructions for assigning metadata, or just existing as general background assumptions with little or no documentation. Rules may also be invoked ad hoc, in particular circumstances and for a particular purpose. What rules apply to the concept expressed by a metadata element can also vary over time, or be different for different communities or projects.

Exploiting information in a semantic web environment of linked data requires precise formalization. However while identifying relevant rules is a task for a domain specialist, both the identification and the precise specification is challenging. The framework we are developing will not only improve our understanding of collection/item metadata relationships, but will help domain specialists identify and express the rules that characterize their data.

Our framework is based on the fact that rules can be grouped together according to their logical form. We say that rules that have the same logical form belong to the same *rule category*. Rules belonging to the same rule category will have the same kinds of logical relationships and the categories themselves will have systematic interrelationships. Our project here is not to legislate specific rules, which is work for domain specialists, but rather to develop a framework of rule categories and, based on that framework, strategies for testing conjectured rules against data. We are carrying this out in first order logic, which is well suited not only to the precise specification of rules and the discovery of interrelationships among rule categories, but will also facilitate the conversion of rules to semantic web languages.

The two top-level categories of the framework refer to the kind of quantification that appears at the item-level in the formulas. Item-level quantification can be either existential (implying that something is true of at least one item in the collection), or universal (implying that something is true of every item in the collection). This gives two general notions of collection/item propagation, universal propagation and existential propagation.

Universal Propagation (UP)

Attributes A and B *propagate universally* =df If a collection *y* has the value *z* for the attribute A, then every

---

[1] We do not offer a formal definition of collection here, but rather begin with the intuitive concept as it is commonly found in library and information science literature, and then explore some logical features (collection/item metadata relationships) that appear to be characteristic of that concept. We see this analysis as contributing to the eventual development of a formal definition.

[2] Of course whether or not such a rule actually holds will depend on the social and legal environment in which the attribute is being used.

[3] Relationships between collection-level attributes and item-level attributes may at first glance appear to be a kind of inheritance, but we argue that they are not. Classical inheritance is based on *is-a-kind-of* or *is-an-instance-of* relationships. The relationship between an item and the collection it is a member of is neither of these; it is unique. This is discussed further in Renear et al. (2008a).

item in the collection has some value *w* for the attribute B such that *w* is related to *z* by the constraint C.

$$\forall y \forall z((A(y,z) \,\&\, Collection(y)) \Rightarrow \forall x(isGatheredInto(x,y) \Rightarrow \exists w(B(x,w) \,\&\, C(w,z)))$$ [4,5]

Existential Propagation (EP)

Attributes A and B *propagate existentially* =df If a collection *y* has the value *z* for the attribute A, then there is some item in the collection that has some value *w* for the attribute B such that *w* is related to *z* by the constraint C.

$$\forall y \forall z((A(y,z) \,\&\, Collection(y)) \Rightarrow \exists x(isGatheredInto(x,y) \,\&\, \exists w(B(x,w) \,\&\, C(w,z))))$$

Specialization Conditions

These general categories of propagation do not place any restrictions on the relationship between the attributes A and B, or on the constraint C. In fact, A and B may be the same attribute. Similarly, the relation between values expressed by the constraint C may in fact be identity.

The possible relations between attributes and conditions on constraints motivate specialization conditions that can be used to characterize the propagation of metadata between collections and items. We call cases where A and B are the same attribute *attribute propagation* (AP), and cases where the constraint C is identity *value propagation* (VP). Cases where A and B are not the same attribute are referred to as *attribute differentiation* (AD), and cases where the constraint C is not identity are referred to as *value constraint* (VC).

$A = B$	attribute propagation (AP)
$\neg (A = B)$	attribute differentiation (AD)
$\forall x \forall y(C(x,y) \equiv (x = y))$	value propagation (VP)
$\neg \forall x \forall y(C(x,y) \equiv (x,y))$	value constraint (VC)

Table 1: Specialization Conditions

Each of the four specialization conditions in Table 1 can be applied to universal propagation (UP) and existential propagation (EP), yielding eight specialized rule categories. These categories appear as the middle column in Figure 1.

[4] The predicate *isGatheredInto*(*x,y*), derived from the Dublin Core Collections Application Profile data model (DCMI, 2007), is used to characterize the membership relationship between items and collections. `

[5] Strictly speaking *Collection(y)* is unnecessary in the antecedent of Universal Propagation although it is needed in Existential Propagation. However we include it so that the two rules are parallel and for a more natural reading.

The specialization conditions on attributes and constraints can also be combined in the following ways (combinations that are logically contradictory are not considered.).

$(A = B) \,\&\, \forall x \forall y(C(x,y) \equiv (x = y))$	AP-VP
$(A = B) \,\&\, \neg \forall x \forall y(C(x,y) \equiv (x,y))$	AP-VC
$\neg (A = B) \,\&\, \forall x \forall y(C(x,y) \equiv (x = y))$	AD-VP
$\neg (A = B) \,\&\, \neg \forall x \forall y(C(x,y) \equiv (x,y))$	AD-VC

Table 2: Combinations of Conditions

Each of the combined specialization conditions in Table 2 can also be applied to UP and EP, yielding the eight fully specialized rule categories shown in Tables 3 and 4. These categories appear as the left-most column in Figure 1.

$\forall y \forall z((A(y,z) \,\&\, Collection(y)) \Rightarrow \forall x(isGatheredInto(x,y) \Rightarrow A(x,z)))$	UP-AP-VP
$\forall y \forall z((A(y,z) \,\&\, Collection(y)) \Rightarrow \forall x(isGatheredInto(x,y) \Rightarrow \exists w(A(x,w) \,\&\, C(w,z)))$	UP-AP-VC
$\forall y \forall z((A(y,z) \,\&\, Collection(y)) \Rightarrow \forall x(isGatheredInto(x,y) \Rightarrow B(x,z)))$	UP-AD-VP
$\forall y \forall z((A(y,z) \,\&\, Collection(y)) \Rightarrow \forall x(isGatheredInto(x,y) \Rightarrow \exists w(B(x,w) \,\&\, C(w,z)))$	UP-AD-VC

Table 3: Universal Propagation Categories

$\forall y \forall z((A(y,z) \,\&\, Collection(y)) \Rightarrow \exists x(isGatheredInto(x,y) \,\&\, A(x,z)))$	EP-AP-VP
$\forall y \forall z((A(y,z) \,\&\, Collection(y)) \Rightarrow \exists x(isGatheredInto(x,y) \,\&\, \exists w(A(x,w) \,\&\, C(w,z))))$	EP-AP-VC
$\forall y \forall z((A(y,z) \,\&\, Collection(y)) \Rightarrow \exists x(isGatheredInto(x,y) \,\&\, B(x,z)))$	EP-AD-VP
$\forall y \forall z((A(y,z) \,\&\, Collection(y)) \Rightarrow \exists x(isGatheredInto(x,y) \,\&\, \exists w(B(x,w) \,\&\, C(w,z))))$	EP-AD-VC

Table 4: Existential Propagation Categories

Example Attributes and Categories

Attributes that express type or genre information are a potential source for rules linking collection-level and item-level metadata. For example, the Dublin Core Collections

Application Profile (DCMI, 2007) includes the attribute *cld:itemType*, defined as "the nature or genre of one or more items in the collection." The direct reference to properties of items in the definition of this collection-level attribute suggests a collection/item metadata relationship.

However, unlike *marcrel:own*, which can be had by items and collections, *cld:itemType* can only be applied to collections. The attribute that is referred to at the item level is "the nature or genre of ... items". So, while we might expect to see the same value at the item level, we will see it reflected by a different attribute (*dc:type* instead of *cld:itemType*). Therefore, this will be a case of attribute differentiation (AD) instead of attribute propagation (AP).

For some pairs of collection- and item-level attributes, we would expect that the values for the attributes will not be equal, only that they will be related by a constraint. We can consider the attribute *dcterms:temporal,* which is used in the Collections Application Profile to indicate the "temporal scope of a collection."

As with the type attributes, we would expect to see an attribute differentiation rule linking *dcterms:temporal* and an item-level date attribute (such as *dc:date*). But we would also expect that instead of seeing the same value appearing at the item level, we would see date values that fall *within* the scope indicated for the collection. For example, given the range "1850-1899" we would expect to see items gathered into the collection with recorded dates that fall within that range. This would be a case of value constraint (VC) instead of value propagation (VP), since the collection-level value is seen as constraining the item-level value.

Spatial attributes follow a similar logic of containment. Collection-level spatial attributes typically constrain the possibilities for item-level spatial attributes, supporting inferences about items.

Logical Relationships Between Categories

The rule categories are related by logical implication as shown in Figure 1. These implications have two sources: the specialization/generalization structure of the framework and the relationship between universal and existential quantification.

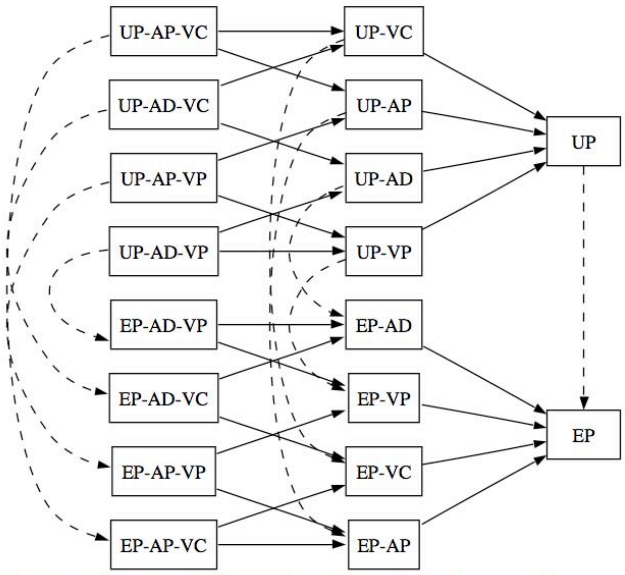

Figure 1: Implications between categories

Because the framework is generated by conjunctively adding specialization conditions to UP and EP it has the logical structure of a specialization/generalization hierarchy. For example, any rule that is in the UP-AP-VC category logically implies the corresponding UP-AP rule, and any UP-AP rule logically implies the corresponding UP rule. This means that attributes that conform to a UP-AP-VC rule will also conform to the corresponding UP-AP rule and attributes that conform to an UP-AP rule will also conform to the corresponding UP rule.

In addition, according to the standard semantics for the universal and existential quantifiers any UP rule logically implies the corresponding EP rule (assuming there are no empty collections). Intuitively: if a collection level attribute implies that every item in a collection has some attribute (UP), then it implies that at least one item in the collection has that attribute (EP).

Relation Properties of Constraints

The nature of the constraint condition, C(*x*,*y*) will determine additional important specializations of the value constraint categories. Many sorts of value constraints are relevant to collection/item metadata rules. These include constraints based on arithmetic relationships, class relationships, generalization relationships (e.g. *is a kind of*), and relationships that are more particular to the value domain (e.g. *is temporally within*). Such relationships can be classified according to their *relation properties* (whether they are transitive, reflexive, symmetric, etc.).

Consider a value constraint like *temporally within*. This relationship is presumably transitive: given temporal intervals *x*, *y*, *z*, if *x* is temporally within *y*, and *y* temporally within *z*, then *x* is temporally within *z*. On the other hand

the constraint *temporally overlaps* is not transitive (two temporal intervals that overlap a third need not themselves overlap); temporal overlap is, however, symmetric: if *x* overlaps *y* then *y* overlaps *x*. Because order properties provide important additional axioms for inferencing further specialization of value constraint categories according to the relation properties of C(*x*,*y*) will obviously be a valuable extension of the framework.[6]

THE CIMR TESTBED

The semantics of metadata attributes is sometimes simply a matter of stipulation; the schema designers agree on what semantics they want their attributes to have and those decisions determine attribute semantics. However, metadata elements are usually defined in natural language sentences that make use of familiar common words that are themselves neither formally defined nor plausibly primitive. In these cases it is difficult to know what was intended by schema designers even assuming they had clear and settled intentions. More importantly the semantic formalization of metadata schemas often happens after the schema is completed and already in use. In that case determining the semantics of metadata attributes as they are actually used is the issue, and not the intentions of the schema designers. For these reasons and others testing conjectured metadata rules against actual metadata is important. To explore how this testing might be accomplished we built an RDF repository containing descriptions from the IMLS Digital Collections and Content (DCC) project (Wickett et al., 2009).

Since CIMR rules are developed in first order logic and refer to facts implied by metadata, we wanted to create a environment for testing that used a knowledge representation language that could support straightforward translation from rules into queries. RDF was therefore a natural choice for data representation as it is explicitly based on a fragment of first order predicate logic, encoding information as subject-predicate-object "triples", which are expressively equivalent to the two place predicates we use in CIMR rules. The associated query language SPARQL provides a mechanism for investigating metadata assignments, and related semantic web ontology and rule languages (OWL and SWRL) can support additional modeling and inferencing. Finally, because we anticipate these rules being used in the emerging semantic web and linked data environment we wanted to do our testing in a similar architecture.

Collections from the IMLS DCC Collection Registry were chosen on the basis of the availability of item descriptions and the appearance of certain properties in those item descriptions. We were interested in examining patterns of values for type, format, temporal, and geographic elements and selected collections with item descriptions that used those attributes. For details on the processing of the OAI-PMH XML records into RDF, see Urban et al. (2010).

Problems for Rule Testing

Our metadata repository was built to support empirical testing of candidate rules, allowing us to search real world metadata for *prima facie* confirmation or refutation of our conjectures. However, testing these rules in an RDF repository is not as straightforward as it might seem.

The rules in the CIMR framework are universally quantified conditionals, evaluated as true if and only if the consequent is true for every case where the antecedent is true. Testing a rule by refutation therefore involves searching for an apparent counterexample, a case where (taking repository statements at face value) the antecedent of the rule is true and the consequent is false. Such a counterexample would then be evidence that the rule is false, although alternative explanations, such as errors made by metadata cataloguers or problems in subsequent processing, must also be considered. If no apparent counterexample is found that might be taken, in the right circumstances, as providing some confirmation of the rule.

Evaluating alternative explanations of apparent counterexamples, and deciding in what circumstances the absence of counterexamples counts as confirmation are familiar challenges in evaluating conditional claims of any kind. But there is a more distinctive and interesting problem in using logic-based queries to search for counterexamples to conditional rules in an RDF repository.

In the formal semantics of first order logic the truth value of a logical formula is relative to an *interpretation*, which is a series of statements assigning predicates to individual things, giving enough information to determine a truth value for the formula. Given an interpretation, a universally quantified conditional is evaluated by searching the list of statements for a counterexample – a statement or group of statements where the antecedent of the conditional is true and the consequent is false. If there is no counterexample, then the conditional is true in the interpretation, if there is a counterexample then the conditional is false in the interpretation.

Like a logical interpretation, our RDF testbed is also a series of statements assigning predicates to individuals, and so might be thought to promise a simple approach to rule evaluation. However the parallel fails in an interesting way.

When interpretations are defined they typically make assignments by listing all true atomic statements; the atomic statements not listed are considered false in that interpretation. This assumption — whatever is not asserted as true is assumed false — is not appropriate for a repository whose statements are directly derived from

[6] To avoid counterintuitive results created by the "trivial satisfaction" of material conditionals these rules must be modalized and a number of other modal restrictions on attributes made. However no ordinary metadata attribute is problematic (Renear et al., 2008b).

metadata records. Metadata records are created in circumstances where there is typically no expectation that statements not asserted as true are assumed false; the policy of "mandatory if applicable" being the exception, not the rule. Rather metadata repositories must be understood as making the "open world assumption," where the absence of a statement does not license inferring the negation of that statement. Metadata is not an exception here; knowledge representation projects, including the semantic web and linked data, also make the open world assumption.

Refutation of our rules in the testbed would still be possible, even under the open world assumption, if the repository contained explicit negations of statements implied by a rule, and in fact specifications of logical interpretations do sometimes indicate which atomic statements are false. However RDF cannot express denials of that sort as it does not have logical negation.

Finally, all of our rules imply only positive atomic statements; there are no negations of atomic statements in rule consequents. This means that it will not be possible for the rule, rather than the repository, to supply the negation needed for a counterexample.

These three things taken together, the open world assumption, the lack of negation in RDF, and rules that imply only positive atomic statements, mean that it is not possible, without additional axioms, to find statements in our RDF repository which are counterexamples to our rules.

Direct refutation is only possible with additional axioms, ones allowing inferences from the presence of some properties to the absence of others. Such axioms are a natural part of formal ontologies and can be expressed in semantic web languages such as OWL and SWRL. However these ontologies have not been developed for DCMI metadata. Our empirical testing is therefore currently focused on exploration and confirmation, rather than refutation, searching for rules that are a best match to patterns observed in the data. Refutation based on additional metadata semantics is planned for future work.

CONFIRMATION OF RULES

In this study, the metadata in our testbed is treated as evidence of the properties held by informational resources. We take the statements at face value and assume that each statement is an assertion of a true fact about a resource. To simplify the process for this relatively small study, we have relied on human reasoning to go from the literal values that are recorded in the repository to the facts that our rules operate against.

Language

In the CIMR testbed, language properties are expressed with *dc:language* for both items and collections. The use of *dc:language* for collections is in accordance with the Dublin Core Collections Application Profile (DCMI, 2007) which gives the usage note "a language of the items in the collection."

Only 16 of the 34 collections in the testbed had sufficient information (both collection and item level attributes recorded) to provide evidence in favor of any collection/item metadata rules. Of those 16, 8 showed a pattern of universal value propagation, and 8 showed a pattern of existential value propagation. The only relation observed between language values in the testbed was identity. That is, in half of the 16 viable collections, the collection-level value for *dc:language* appeared as the value of *dc:language* for every item in the collection, confirming the universal propagation rule.

In the other half of these collections, the collection-level value for *dc:language* appeared as the value of *dc:language* for some, but not all of the items in the collection. However, in these cases the collection-level value was, by far, the most common *dc:language* value of items in the collection. A typical case is a collection with the collection-level value "English" for *dc:language*, that has 974 items with the value "English" for *dc:language*, and 1 item with the value "German." These cases confirm the existential propagation rule.

Every collection that confirms the universal propagation rule also confirms the existential rule. Therefore an existential version of an attribute value propagation rule (EP-AP-VP from Figure 1) was confirmed for *dc:language* in the testbed.

$$\forall y \forall z((language(y,z) \,\&\, Collection(y)) \Rightarrow \exists x(isGatheredInto(x,y) \,\&\, language(x,z)))$$

The pattern of prevalence of the collection-level value at the item level suggests that default reasoning (where a given value propagates by default, but can be overridden where more information is given) could be employed to support the use of collection/item metadata rules referring to language attributes.

Date

There are several attributes that are used to express temporal information in the CIMR testbed. In the following we discuss the relationship between *dcterms:temporal* at the collection level and *dc:date* at the item level. Other elements that express temporal information (coverage and subject attributes) were considered too uncontrolled for this study. The use of *dcterms:temporal* at the collection level is in accordance with the Dublin Core Collections Application Profile, which gives the property the label "Temporal Coverage" and the usage note "an indicator of the temporal scope of the collection".

For the most part, testbed date values at the collection level are from a constrained controlled vocabulary that provides year ranges (e.g. "1850-1899"). Temporal values at the item level appeared in a variety of formatted date strings

and as years or year ranges. The temporal relations of withinness, overlap, covering, and identity were all observed between values at the collection and item levels.

Out of the 34 collections in the testbed, 33 had values for *dcterms:temporal* and associated items with values for *dc:date.* Of these 33 collections, 25 collections showed an existential constraint between the collection and item metadata. In particular, for each value of *dcterms:temporal* at the collection level, some item had a value for *dc:date* that was temporally within the range given at the collection level.

This pattern confirms an existential rule of attribute differentiation between *dcterms:temporal* and *dc:date* with the value constraint of temporal withinness.

$$\forall y \forall z((temporalCoverage(y,z) \,\&\, Collection(y)) \Rightarrow \exists x(isGatheredInto(x,y) \,\&\, \exists w(date(x,w) \,\&\, temporalWithin(w,z))))$$

The prevalence of temporal withinness in the CIMR repository is connected to the use of date ranges in the collection-level controlled vocabulary. However, it is natural to suppose that date information at the collection-level will be more general and comprehensive than information at the item level and that this constraint may be common across many repositories.

Type

Type information is expressed in the CIMR testbed with the collection-level attribute *cld:itemType* and the item-level attribute *dc:type*. The Dublin Core Collections Application Profile defines *cld:itemType* as "the nature or genre of one or more items within the collection" and the Dublin Core Metadata Terms defines *dc:type* as "the nature or genre of the resource."

All 34 collections in the testbed had values for *cld:itemType* and associated items with values for *dc:type*. Only 2 of the collections matched a universal version of a value propagation rule. That is, every item in those collections had the *cld:itemType* value recorded as a value of *dc:type.* However, 12 collections confirmed a universal version of a value constraint rule. Every item in these collections had a value for *dc:type* that was related to the collection's value for *cld:itemType.*

In these cases, the constraint between the collection and item level values was a *generalization* relationship. For example, a collection with the value "photographs/slides/negatives" for *cld:itemType* might have items with the value "image" for *dc:type.* We say that a value *x generalizes* a value *y* when *x* can be applied to everything that *y* can be applied to. In the above, everything that falls within the category "photographs/slides/negatives" also falls within the category of "images", so we say that "image" generalizes "photographs/slides/negatives".[7]

An existential version of a generalization value constraint rule between *cld:itemType* and *dc:type* was confirmed by an additional 5 collections. The collections that confirmed the universal version of this rule also confirm this existential rule, since if the constraint holds for every member of the collection (and the collection has some members), it must hold for some member in the collection. Thus an existential version of value constraint between *cld:itemType* and *dc:type* (EP-AD-VC in Figure 1) was confirmed by 17 of the collections (20 collections if identity is taken as a special case of generalization).

$$\forall y \forall z((itemType(y,z) \,\&\, Collection(y)) \Rightarrow \exists x(isGatheredInto(x,y) \,\&\, \exists w(type(x,w) \,\&\, generalizes(w,z)))$$

The existential version of the rule also matches closely with the usage note for *cld:itemType* given in the Collections Application Profile. The note states that the attribute indicates the nature or genre of "one or more items in the collection," which supports the choice of an existential rule to link this attribute to item-level information.

The collection descriptions in the CIMR testbed are created with the use of a list of type terms from which values for *cld:itemType* are chosen. This has an impact on the constraint that appears between values, since given the choice, many collections might have been described with the value "images". This would have given us a pattern of value propagation rather than value constraint. In our case, the item-level values for *dc:type* were more general (for example "image") than the values used for *cld:itemType* (e.g. "photographs/slides/negatives"), however it is easy to imagine a repository where the item level values are more specific than the collection-level values.

In fact, it was surprising to see that item-level values for type attributes were often more general than the values recorded for collections. Given that collection description is often oriented towards providing information at a summary level, we expected that constraints would move from general to more specific terms (as is the case with dates). This shows that the nature of the constraint for a rule will depend heavily on the attributes under consideration.

As with date ranges and individual dates determining a particular withinness constraint for date attributes, the type vocabularies in use for the collections and the items in the CIMR repository have shaped the generalization constraint for type attributes. Generally, the various vocabularies in

[7] These relations are familiar as "broader term" and "narrower term" relators that appear in thesauri. We take a more general approach since recorded values may be from distinct vocabularies without available term relations.

play in a particular repository or aggregation will influence how patterns between collection and item metadata occur.

Format

Format information is recorded in the testbed with the collection-level attribute *cld:itemFormat* and the item-level attribute *dc:format*. The Dublin Core Metadata Terms defines *dc:format* as "the file format, physical medium, or dimensions of the resource" and the Dublin Core Collections Application Profile defines *cld:itemFormat* as "the media type, physical or digital, of one or more items in the collection." Given the similarity between *cld:itemFormat* and *cld:itemType*, we might expect to see similar patterns of values between the collection and item level descriptions.

Nine of the collections confirmed a universal attribute differentiation rule with value propagation between *cld:itemFormat* and *dc:format*. Two collections confirmed an existential version of the value propagation rule, and since the collections that confirmed the universal rule also confirm this version, we see 11 collections confirming the existential rule (EP-AD-VP).

$$\forall y \forall z((itemFormat(y,z) \,\&\, Collection(y)) \Rightarrow \exists x(isGatheredInto(x,y) \,\&\, format(x,z)))$$

The number of collections confirming this rule is significantly lower than was the case for *cld:itemType* and *dc:type.* This was somewhat surprising, considering the parallel definitions of the attributes.

The lack of confirmation from many of the collections in the testbed seems to stem from the fact that *dc:format* is used in the testbed to convey information about a number of different things. Studies have noted that this attribute is challenging for metadata cataloguers to deploy in practice (Park & Childress, 2009), and the variation in reference in our testbed is most likely connected to the same set of problems.

Item-level format values in the CIMR testbed may reference:

- file types
- material of original
- physical dimensions given in inches or centimeters
- file sizes
- pagination information
- time length of segment (presumably of audio recordings).

Some of the above (e.g. file types and file sizes) seem to apply logically to the digital resource that we consider to be gathered into collections for the CIMR testbed. Generally these were the cases that confirmed our propagation rules. However, many of them (e.g. material of original and physical dimensions) seem to be making reference to a physical resource from which the digital resource was derived.

In cases where the collection level value refers to a digital resource but item level values properly refer to another resource that stands in some special relationship to the digital resource, there is no clear direct relationship between the collection-level and item-level values. These cases therefore don't confirm any of our rules.

Summary

Table 5 summarizes the results of exploring the CIMR testbed in order to confirm rules connecting collection-level and item-level description.

Attribute Pair	Constraints	Rule Category
dcterms:temporal dc:date	temporal within, overlap, covering identity	existential attribute differentiation – value constraint
dc:language dc:language	identity	existential attribute value propagation
cld:itemType dc:type	generalization and specialization identity	existential attribute differentiation – value constraint
cld:itemFormat dc:format	identity	existential attribute differentiation – value propagation

Table 5: Summary of Rule Confirmations

Existential propagation rules were confirmed for each of the attribute pairs examined. These rules are an intuitive match to natural language definitions of collection-level properties like *cld:itemType* and *cld:itemFormat*, which make explicit reference to "one or more" items within the collection having a particular property.

Considering the evidence in the CIMR testbed, the use of *dc:language* in the Dublin Core Collections Application Profile stands out. Despite maintaining the original name, the attribute seems to have an application that is parallel to *cld:itemType* and a natural language definition that refers to a property of items within a collection (suggesting an attribute like *cld:itemLanguage*). The logical characterization of the collection-level attribute in terms of its relation to item-level attributes clarifies the meaning of the language attribute.

The examination of *cld:itemFormat* and *dc:format* also clarifies how these attributes are applied to resources. The definition of *dc:format* ("the file format, physical medium, or dimensions of the resource") seems to range across a variety of ontological categories. This may not be problematic in traditional descriptive environments, but attributes with this kind of variation in reference will

present difficulties for making inferences about resources based on metadata.

By investigating the CIMR testbed, we have confirmed propagation rules for these four pairs of attributes. The regimented examination of metadata attributes in terms of their application to a set of resources is a promising approach for contributing to a systematic understanding of metadata semantics.

CONCLUSION

The framework of relationship rule categories for collection/item metadata has a wide range of potential applications, supporting schema designers, metadata cataloguers, software developers, and systems managers. Most importantly the integration of such a framework into schema design, cataloguing policies, and systems design and configuration can help make available to researchers the collection level context that is vital to the effective use of collections, but that is too often ignored by current systems for retrieval and analysis.

The framework presented here is a small fragment of what is needed. Among other limitations it focuses only on implications between single collection-level attributes and single item-level attributes and it does not yet elaborate categories based on relation properties of value constraints. So more remains to be done. However the fundamental techniques for further developing this framework, as worked out here, seem sound.

The concept of a collection is a foundational one in information organization. An improved understanding of the logical relationships between collection level and item level attributes will not only support practical applications, it will tell us a little more about what collections really are.

ACKNOWLEDGMENTS

This research is supported by a 2007 IMLS NLG Research & Demonstration grant as part of the IMLS Digital Collections and Content project, Principal Investigator, Carole L. Palmer, Center for Informatics Research in Science and Scholarship (CIRSS). Project documentation is available at http://imlsdcc.grainger.uiuc.edu/about.asp. The work reflects contributions from discussions with other DCC/CIMR project members, Wu Zheng, Larry Jackson, Katrina Fenlon, Jacob Jett, and Dave Dubin.

REFERENCES

Brockman, W. et al. (2001). Scholarly Work in the Humanities and the Evolving Information Environment. Washington, DC: Digital Library Federation/Council on Library and Information Resources.

Christenson, H. Tennant, R. (2005). Integrating Information Resources: Principles, Technologies, and Approaches. California Digital Library. http://www.cdlib.org/.

Currall, J., Moss, M., & Stuart, S. (2004). What is a collection? Archivaria, 58, 131-146.

Dempsey, L. (2005). From metasearch to distributed information environments. Lorcan Dempsey's Weblog (October 9, 2005). http://orweblog.oclc.org/archives/000827.html

DLF. (2005). The Distributed Library: OAI for Digital Library Aggregation. OAI Scholars Advisory Panel, June 20-21, Washington, DC. Digital Library Federation.

DCMI. (2007). Dublin Core Collections Application Profile http://dublincore.org/groups/collections/collection-application-profile/. Retrieved May 24, 2010.

DCMI. (2008). Dublin Core Metadata Terms. http://dublincore.org/documents/dcmi-terms/. Retrieved May 31, 2010.

Foulonneau, M., Cole, T. W., Habing, T. G., & Shreeves, S. L. (2005). Using collection descriptions to enhance aggregation of harvested item-level metadata. Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries. ACM Press, 32-41.

Greenberg, J. (2009). Theoretical Considerations of Lifecycle Modeling: An Analysis of the Dryad Repository Demonstrating Automatic Metadata Propagation, Inheritance, and Value System Adoption. Cataloging & Classification Quarterly. 47, 3, p. 380-402.

Heaney, M. (2000). An Analytic Model of Collections and Their Catalogues, UK Office for Library and Information Science.

Lagoze, C. et al. 2006. Metadata aggregation and "automated digital libraries": A retrospective on the NSDL experience. Proceedings of the 6th ACM/IEEE-CS Joint Conference on Digital Libraries. ACM Press, New York.

Lee, H. (2000). What is a collection? JASIS, 51 (12), 1106-1113.

Lee, H. (2005). The concept of collection from the user's perspective. Library Quarterly, 75(1), 67-85.

Lourdi, I. and Papatheodorou, C. and Doerr, M. (2009) Semantic Integration of Collection Description. D-Lib Magazine 15,7/8.

Palmer, C. L. (2004). Thematic research collections. S. Schreibman, R.Siemens, & J. Unsworth (Eds.). Companion to Digital Humanities. Oxford: Blackwell, pp. 348-365.

Palmer, C.L., Knutson, E., Twidale, M, and Zavalina, O. (2006). Collection definition in federated digital resource development. Proceedings of the 69th ASIS&T Annual Meeting (Austin, TX, Nov. 3-8, 2006).

Park, J., & Childress, E. (2009). Dublin Core metadata semantics: An analysis of the perspectives of information professionals. Journal of Information Science, 35 (6), 727-739.

Renear, A.H., Wickett, K.M., Urban, R.J., Dubin, D., Shreeves, S.L. (2008a). Collection/Item Metadata Relationships. In Proceedings of the International

Conference on Dublin Core and Metadata Applications, Berlin, Germany, September 22-26, 2008.

Renear, A. H., Wickett, K. M., Urban, R. J., & Dubin, D. (2008b). The Return of the Trivial: Problems Formalizing Collection-Level/Item-Level Metadata Relationships. Proceedings of the ACM/IEEE Joint Conference on Digital Libraries. Pittsburgh, PA.

Urban, R.J., Wickett, K.M. and Renear, A.H. (2010). A Testbed for Collection-Item Metadata Relationships. GSLIS Technical Report UIUCLIS--2010/4. Retrieved from: http://hdl.handle.net/2142/16604

Warner, S., Bekaert, J., Lagoze, C., Lin, X., Payette, S., & Van de Sompel, H. (2007). Pathways: Augmenting interoperability across scholarly repositories. International Journal on Digital Libraries.

Wendler, R. (2004). The eye of the beholder: Challenges of image description and access at Harvard. In Hillmann, D. I. and Westbrooks, E. L., eds., Metadata in Practice. American Library Association, Chicago, IL, pp. 51-6.

Wickett, K.M., Urban, R.J., Zheng, W., Renear, A.H. (2009) A testbed approach for metadata inference rule development. Workshop On Integrating Digital Library Content with Computational Tools and Services. ACM/IEEE Joint Conference on Digital Libraries (JCDL 2009), June 2009, Austin, TX.