MUSEUMFINLAND —Finnish Museums on the Semantic Web

Eero Hyvönen, Eetu Mäkelä, Mirva Salminen, Arttu Valo,
Kim Viljanen, Samppa Saarela, Miikka Junnila, and Suvi Kettula

Helsinki Institute for Information Technology (HIIT),
University of Helsinki, and Helsinki University of Technology
P.O. Box 5500, 02015 TKK, FINLAND
FirstName.LastName@cs.Helsinki.FI
http://www.cs.helsinki.fi/group/seco/

Abstract

This article presents the semantic portal MUSEUMFINLAND for publishing heterogeneous museum collections on the Semantic Web. It is shown how museums with their semantically rich and interrelated collection content can create a large, consolidated semantic collection portal together on the web. By sharing a set of ontologies, it is possible to make collections semantically interoperable, and provide the museum visitors with intelligent content-based search and browsing services to the global collection base. The architecture underlying MUSEUMFINLAND separates generic search and browsing services from the underlying application dependent schemas and metadata by a layer of logical rules. As a result, the portal creation framework and software developed has been applied successfully to other domains as well. MUSEUMFINLAND got the Semantic Web Challence Award (second prize) in 2004.

Key words: semantic web, information retrieval, multi-facet search, view-based search, ontology, recommendation system

1 Why Museums on the Semantic Web?

A special characteristic of cultural collection contents is semantic richness. Collection items have a history and are related in many ways to our environment, to the society, and to other collection items. For example, a chair may be made of oak and leather, may be of a certain style, was designed by a famous designer, was manufactured by a certain company during a time period, was used in a certain building together with other pieces of furniture, and so on. Other collection items, locations, time periods, designers, companies etc. can be related to the chair through

Preprint submitted to Elsevier Science *9 May 2005*

their properties and implicitly constitute a complicated semantic network of associations. This semantic network is not limited to a single collection but spans over other related collections in other museums. The network of semantic associations can be extended to contents of other types in other organization, as well.

Much of the semantic web content will be published using semantic portals [1] [24]. Such portals typically provide the end-user with two basic services: 1) a search engine based on the semantics of the content [2] and 2) dynamic linking between pages based on the semantic relations in the underlying knowledge base [6]. Semantic web technology [2] enables new possibilities when publishing museum collections on the web [15]:

Collection interoperability in content Web languages, standards, and ontologies make it possible to make heterogeneous museum collections of different kind mutually interoperable. This enables, e.g., the creation of large inter-museum exhibitions.

Intelligent applications More versatile, user-friendly, and useful applications based on the semantics of the collections can be created.

To realize these ideas in practice, we have developed a semantic web portal called "MUSEUMFINLAND—Finnish Museums on the Semantic Web" [3]. This system contains an inter-museum exhibition of over 4,000 cultural artifacts, such as textiles, pieces of furniture, tools etc. Also metadata concerning some 260 historical sites in Finland were incorporated in the system. The goals for developing the system were the following:

Global view to distributed collections It is possible to use the heterogeneous distributed collections of the museums participating in the system as if the collections were in a single uniform repository.

Content-based information retrieval The system supports intelligent information retrieval based on ontological concepts, not on simple keyword string matching as is customary with current search engines.

Semantically linked contents A most interesting aspect of the collection items to the end-user are the implicit semantic relations that relate collection data with their context and to each other. In MUSEUMFINLAND, such associations are exposed dynamically to the end-user by defining them in terms of logical predicate rules that make use of the underlying ontologies and collection metadata.

Easy local content publication The portal should provide the museums with a cost-effective publication channel.

Museum databases are usually situated at different locations and use different database systems and schemas. This creates a severe obstacle to information retrieval.

[1] See, e.g., http://www.ontoweb.org/.

[2] http://www.w3.org/2001/SW/

[3] http://museosuomi.cs.helsinki.fi/

To address the problem, the web can be used for creating a single interface and access point through which a search query can be sent to distributed local databases and the results combined into a global hit list. This "multi-search" approach is widely applied and there are many cultural collection systems on the web based on it, such as the portals Australian Museums Online [4] and Artefacts Canada [5] .

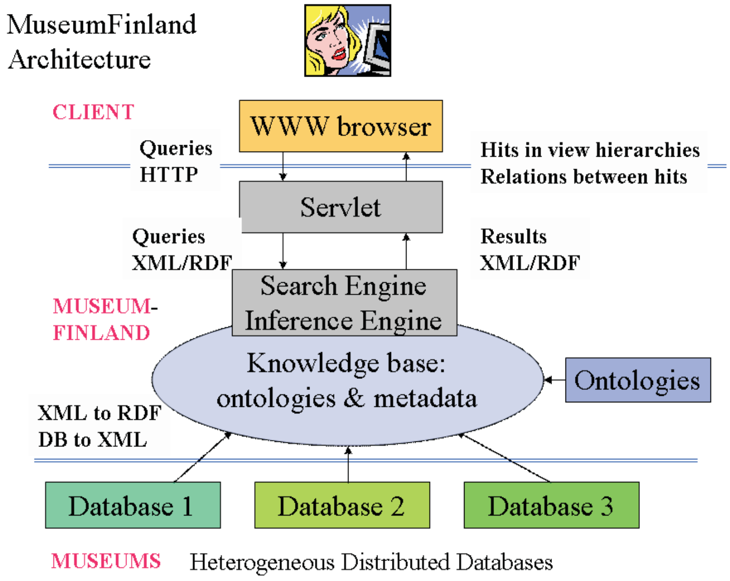


Fig. 1. Information retrieval in MUSEUMFINLAND. Local database contents are first merged and the query is evaluated with respect to the global interrelated data.

A problem of multi-search is that by processing the query independently at each *local database*, the *global* dependencies, associations between objects in different collections are difficult to found. Since exposing semantic associations between collections items is one of our main goals, MUSEUMFINLAND cannot be based on the multi-search paradigm. Instead, the local collections are first consolidated into a global repository, and the search queries are answered based on it (cf. figure 1). Mutually shared conceptual models, ontologies, are used for enriching the content and for making the collections interoperable. To show the associations to the end-user, the collection items are represented as web pages interlinked with each other through the semantic associations. The MUSEUMFINLAND home page is the single entry point through which the end-user enters the global semantic WWW space. A challenge in this approach is that a separate content creation process is needed for consolidating the global repository based on local databases.

This paper presents MUSEUMFINLAND from different viewpoints [15, 13, 19, 18, 25]. The creation and structure of the ontologies underlying the system is first discussed. After this we explain how content from the museum databases can be imported into the global RDF(S) [6] [21, 1] repository conforming to the shared on-

[4] http://www.amonline.net.au/
[5] http://www.chin.gc.ca/
[6] http://www.w3.org/RDF/

tologies. Next the semantic search and browsing services of MUSEUMFINLAND are explained from the end-user's viewpoint, and adaptation of the system to new data is briefly discussed. Then we get down to the implementation and describe the general architecture underlying the system, and its components. The paper concludes by discussing the lessons learned as well as related and future work.

2 Ontologies

Ontology	Content	Classes	Instances
Artifacts	Classes for tangible collection objects	3227	0
Materials	Substances that the artifacts are made of	364	0
Situations	Situations, events, and processes in the society	992	0
Actors	Persons, organizations, and other active agents	26	1715
Locations	Continents, countries, cities, villages, farms etc.	33	864
Times	Eras, centuries, etc. as labeled time intervals	57	0
Collections	Museum collections included in the system	22	24

Table 1
The ontologies used in the MUSEUMFINLAND portal. The numbers indicate classes and individuals in actual use in the first version of the portal. The total number of all classes and individuals in the underlying ontologies is about 10,000.

MUSEUMFINLAND uses the seven domain ontologies that are listed in table 1.

(1) The Artifacts ontology is a hyponymy taxonomy of tangible collection objects, such as pottery, cloths, weapons, etc. All artifact exhibits in the system belong to some class in this ontology. The taxonomy was extended with properties available from an underlying thesaurus MASA [23] (to be discussed later in more detail). In some parts of the ontology, more properties have been defined but are not used in the current version of MUSEUMFINLAND.
(2) The Materials ontology is a hyponymy taxonomy of the artifact materials, such as steel, silk, tree, etc. The classes are based on MASA.
(3) The Actors ontology defines classes of agents, such as persons, companies etc., and individuals as instances of these classes.
(4) The Situations ontology is a taxonomy that includes intangible happenings, situations, events, and processes that take place in the society, such as farming, feasts, sports, war, etc. The classes are based on MASA.
(5) The Locations ontology represents areas and places on the Earth. It contains classes such as Continent, Country, County, City, Farm etc. The main content in the ontology is its individual location instances (e.g., Helsinki or Finland) and their mutual meronymy relations (e.g., Helsinki is a part of Finland).

(6) The Times ontology is a meronymy of various predefined historical periods. First, there are categories representing special eras of interest such as the Middle Ages and the time of the World War II. Second, there is a linear breakdown hierarchy of centuries and decennia. The properties of time concepts are a human readable label of period and the beginning and end year of the time interval.

(7) The Collections ontology is a taxonomy that classifies the collections included in the portal under the museums hosting them. The properties of the taxonomy indicate the name and the hosting museum of the collection.

All taxonomy classes in MUSEUMFINLAND are instances of metaclasses for which properties such as the creator, description, date of creation, etc. can be specified.

The seven domain ontologies were created by three main methods: *manual editing*, *thesaurus transformation*, and *ontology population*. In the following, these methods and the schemas of the created ontologies are discussed in more detail.

2.1 Manual editing

Ontologies are typically created or enhanced by hand using an ontology editor. This is feasible, e.g., with small ontologies, semantically complex ontologies, or if there are no thesauri or other data repositories available for computer-based ontology creation. In our case, the Collections and Times ontologies were created in this way. All ontologies have been enhanced manually to some extent even if much of the creation work could be automated. In this work the Protégé-2000 [7] editor with its RDF plug-in was mostly used.

2.2 Thesaurus transformation

Controlled vocabularies and thesauri are usually used when indexing collection items in a database. A thesaurus employs a small number of relationships to organize the terms, such as information about broader (BT), narrower (NT) and related terms (RT), as well as properties instructing the human thesaurus user, such as ”see” reference (USE), its reciprocal relation ”use for” (UF), and scope note (SN) [5]. Sometimes references to synonyms, antonyms, and homonyms may be explicitly presented, too.

In Finland, the most notable and widely used thesaurus for cultural content in Finnish is MASA [23] maintained by the National Board of Antiquities [8] . MASA

[7] http://protege.stanford.edu/

[8] http://www.nba.fi/

consists of some 6000 terms and employs the usual thesaurus relations NT, BT, RT, USE, UF, SN discussed above. This repository was available as a database and its terms could be used as a basis for creating a new, larger cultural ontology called MAO (6768 classes).

When transforming a thesaurus into an ontology [36], the NT/BT relations can be used as a first approximation for the subsumption taxonomy. However, lots of manual corrections are needed for several reason. First, the semantics of the NT/BT relation typically includes different forms of both hyponymy and meronymy, which may not be desirable. Second, the relations are often defined locally without considering a larger global context, such as transitivity of the NT/BT relation. For example, the entry Make-up mirror can be a narrower term (NT) of Mirror and the entry Mirror can be a narrower term of Furniture. However, one should not infer from this transitively that a make-up mirror is a piece of furniture like one could with a proper subsumption (subClassOf) hierarchy. Third, the NT/BT relations are not systematically developed in thesauri. In the case of MASA thesaurus, for example, it turned out that there were about 2600 roots that had no broader term among the 6000 terms. Hundreds of new intermediate classes had to be defined in order to create a complete developed class hierarchy needed for the MUSEUMFINLAND system. Fourth, the thesauri may also contain some errors that have not been detected by the term bank system used for editing the thesaurus. In MASA, for example, some missing reciprocal links and even circularity in the NT/BT relation was detected.

MASA thesaurus was transformed into MAO in three steps:

(1) A meta-level for MAO-ontology was created using Protégé-2000. This meta-level consists of meta-classes that describe the properties of the ontological classes to be created as MAO classes. The meta-properties fall into two categories: 1) Semantic relations of the thesaurus as they are, such as BT, NT, etc. 2) Metadata documenting the meaning and creation history of the classes, such as creator, date-of-creation, etc.

(2) An RDF Schema structure (a Protégé-2000 project) conforming to the RDFS representation conventions of Protégé-2000 was created automatically from the database. This structure represented the entries of the thesaurus as classes organized into an initial rdfs:subClassOf taxonomy corresponding to the NT/BT relation.

(3) A human editor, museum curator, edited the hierarchy further with Protégé-2000 into a proper taxonomy by introducing new concepts and by re-organizing the classes. Some 700 new classes were created during this phase.

In this way, three domain ontologies, Artifacts, Materials, and Situations in table 1 emerged as sub-ontologies of MAO. These ontologies were later on extended based on collection item data from the collections of the National Museum [9], Espoo City Museum [10], and Lahti City Museum [11].

2.3 Ontology population

By ontology population we refer to a process, where a class structure of an ontology already exists and is extended by creating individuals based on some data repository. This can be done either by a computer or by a human editor. In our case, the Locations and Actors ontologies in table 1 were created in this way by a semi-automatic process.

The class structure of the Locations ontology is small and could be created by hand (classes like Continent, Country, City, Farm etc.). An initial set of a couple hundred individual countries and cities was generated automatically from official data sources, such as the list of Finnish cities and counties. However, most of the instance data had to be populated from the collection databases, since the museum databases include specific location information—for example specific estates or historic locations—that were not available in the official data sources. For these locations some meronymy relations could be identified automatically. This is because many collection data entries contained both a general and a more particular location term (c.f., Paris in France vs. Paris in Texas), from which the meronymy relation could be deducted. For ambiguous location names, the rdf:type and part-of properties had to be edited by a human editor.

As in Locations, the class structure of the Actors ontology is small including classes such as Person, Woman, Company, etc., and could be created by hand. Most of the resources in the ontology are instances, such as particular persons. The individuals were populated from the databases. In some cases, the class of the instance could be deduced from the original data. If not, the computer made a guess and let the human editor check the result. For example, it may be known that a certain string, say “John Doe”, is a person’s name but the sex has not been represented explicitly. The computer can then create an instance of class Person and let the editor change the class to either Woman or Man.

2.4 Enriching ontologies with cultural and common sense knowledge

A major goal of MUSEUMFINLAND is to provide the end-user with semantic association links relating collection contents with each other. Such association are based

[9] http://www.nba.fi/en/nmf/
[10] http://www.espoo.fi/museo/
[11] http://www.lahti.fi/museot/

on cultural and common sense knowledge about the society and its functions. It tells, for example, how, in what context, and for what reason different artifacts have been used. Much of this kind of knowledge falls outside of traditional taxonomic ontological knowledge and is not explicit in the metadata descriptions, either.

We therefore decided to enrich the knowledge base of MuseumFinland with addition cultural and common sense knowledge. Such knowledge serves two purposes:

- From the end-user's view-point, it enables semantic link generation and semantic browsing. This feature will be discussed in detail in the coming sections.
- From the cataloger's view-point, it makes the cataloging process simpler because many additional annotations can be automatically created. For example, if we know that the artifact is a doctor's hat, then there is no need to tell that it is related to academic ceremonies, because this inference can be drawn by a simple rule.

Additional knowledge was incorporated into the system in two ways: 1) by explicit associations and 2) by more complex logic rules using them (in addition to ontological knowledge and metadata).

A few simple explicit association types of form *X isRelatedTo Y* were identified. First, we envisioned that the events taking place in the society, i.e., the Situations ontology, are of central importance for creating useful semantic linkage. Therefore additional association triples of form $(artifact, is-related-to-event, situation)$ were created. These relations were defined by a museum curator with the user-friendly N3-notation [12] . For example:

```
masa:spade mapping:is-related-to-event masa:forestry.
masa:Christmas_tree mapping:is-related-to-event masa:Christmas.
```

Second, artifacts are related to each other, which can be represented by the triple $(artifact_1, is-related-to-artifact, artifact_2)$. For example, sailing ships are related to sails, screw drivers to screws, etc. Thirdly, there are association between artifacts and materials. Altogether, 301 different associations between ontology classes were created in this way.

Based on the ontologies, associations, annotation schema, and the metadata from the databases, a set of more complex labeled associations between resources were defined in terms of predicate logic rules. These rules (to be discussed in more detail later) exploit, e.g., the fact that the associations are inherited along the rdfs:subClassOf hierarchy, make use of the relations defined in MASA, and use the various metadata annotation properties of the collection artifacts.

[12] http://www.w3.org/DesignIssues/Notation3.html

The association types in MUSEUMFINLAND are primitive from the semantic view point, but could be defined easily and can be used in practice as a basis for creating many useful links on the user interface. We have found the idea of using the Situations ontology as a means for linkage creation very promising. At the moment we are developing a more elaborate event ontology with properties such as agent, object, time etc. as is customary in knowledge representation research [32]. We envision, that more useful semantic links can be created by making the distinction between the different roles in which individuals participate in event, and that the prize to be paid in terms of creating more complex metadata is worth paying in many situations.

3 Content creation process

The collection item (meta)data MUSEUMFINLAND came from the four databases of table 2. The databases were situated in different locations and used four different database schemas and cataloging systems that were based on three different database systems. The column Items indicates the number of collection objects taken from the databases for the pilot version of MUSEUMFINLAND. Only a small fraction of collections was used. The selection was based on, e.g., the museums' publication prioritization, quality of metadata, whether the data record contained an image of reasonable quality, etc.

Museum	**DB system**	**Cataloging System**	**Items**	**Content**
Espoo City Museum	Ingress	Escoll	1190	artifacts
Lahti City Museum	Ingress	Antikvaria	1587	artifacts
National Museum	MS Server	Musketti	1351	artifacts
National Museum	MS Access	MS Access	256	hist. sites

Table 2
The databases used in the MUSEUMFINLAND pilot version.

These local heterogenous databases were transformed into a global, syntactically and semantically interoperable knowledge base in RDF format, which conforms to the set of global museum ontologies (table 1). The annotation process was designed to meet two requirements: First, new museum collections need to be imported into the MUSEUMFINLAND portal as easily as possible and with as little manual work and technical expertise as possible. Second, the museums should have maximal local freedom in annotations and need to commit to only necessary restrictions and complications imposed by the portal and the other content providers. For example, two museums may use different terms for the same thing. The system should be able to accept the different terms as far as the terms are consistently used and their local meanings with respect to the global reference ontologies are provided.

Figure 2 depicts the annotation process that consists of three major parts. First syntactic homogenization is obtained by transforming the relational database records into a shared XML language, cf. the DB2XML arrow on the left. The result is a set of em XML cards. Second, terminology definitions in RDF, called em term cards, are created based on the XML data, cf. the lower XML2RDF arrow. The transformation is performed by a tool called Terminator. The term cards map XML level literals onto URIs in the museum ontologies. Third, semantic interoperability is obtained by transforming the XML cards—with the help of term cards—into RDF form that conforms to the global museum ontologies, cf. the upper XML2RDG arrow on the right. The result is a set of *RDF cards*. This transformation is performed by a tool called Annomobile. In the following, the three parts of the annotation process are discussed in more detail.

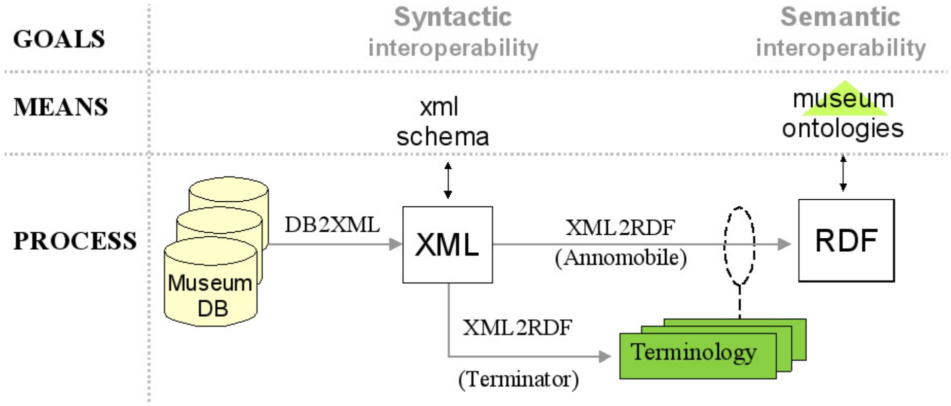


Fig. 2. The content creation process in MUSEUMFINLAND.

3.1 *Syntactic homogenization based on XML*

The first step in combining the heterogeneous relational databases is to gain syntactic interoperability by transforming database contents into a shared XML format. This means, for example, that the data record fields meaning the same thing but under different labels in different databases, such as “name of object” and “object name”, are mapped onto the same XML attribute value. The transformation procedure from database to XML depends on the database schema and system at hand, and is described more in detail in [29].

The reasons for using an intermediate XML level and XML transformation step in the annotation process are: Firstly, XML provides a simple, open language by which the participating museums can agree upon the syntax for representing collection data. Secondly, database system dependent parts of the whole annotation process can now be separated into the database to XML transformation, and the remaining steps that can be shared by all museums.

Based on the schema, each collection item has an XML description of its own called the XML card. For example, the XML card representing a calendar is presented below[13]:

```
<artifactCard created="2003-7-29 10:43:16">
  <artifactId> ECM:22461:1 </artifactId>
  <artifactType> Christmas calendar,
         Finland’s Scouters Assoc. </artifactType>
  <museum> Espoo City Museum </museum>
  <material> cardboard </material>
  <keywords>
    <keyword> Christmas </keyword>
    <keyword> calendar </keyword>
    <keyword> scouts </keyword>
  </keywords>
  <placeOfUsage> Tapiola, Espoo </placeOfUsage>
  <creator> Ulla Vaajakoski </creator>
  ...
  <photo> photos/image3451.jpg </photo>
</artifactCard>
```

An XML card presents the main features of a collection object by sub-elements. The values of the features, such as the string ”Espoo City Museum” in the sub-element `<museum>`, are read from the underlying database tables. However, there are often difficulties in creating such strings. In below some of them are listed and the solution approaches taken in MuseumFinland outlined.

- Imprecise data. The information available is often imprecise in different ways. One has be able to make the distinction between the following cases: 1) The value is missing but existing, i.e., *unknown*. For example, the creator of a painting may be unknown. 2) The *value does not exist*. For example, a telephone machine may not have the artistic style property at all. 3) The value is *uncertain*. For example, the manufacturing time of a chair may be somewhere during 1850-1870. In MuseumFinland unknown values are represented by a special symbol, missing properties by are identified by empty values, and uncertainty is represented by time intervals and by using more general classes as values. For example, class “metal” can be used for uncertain metallic material.
- Complex values. The value of a property is often a combination of facts that may be stored in different database tables. For example, an artifact may have a genus name (e.g., “toy”) with a species name (“Donald Duck toy”), additional colloquial names, and names in different languages. Such detailed information should not be lost. In our system, complex values are simply concatenated into a string by using a semicolon as the separation mark, i.e., the value may be a set of values.
- Dealing with errors. The information available is in many cases syntactically erroneous due to typing errors. This is a problem that should of course be solved already when cataloging the items, but errors occur and have to be dealt with. In MuseumFinland the system creates a log file for erroneous or not matching cards and lets the human editor make needed corrections.

[13] The example is translated and slightly simplified from the original version in Finnish.

Property	**Meaning**
singular	Singular form of the term as a string
plural	Plural form of the term
concept	URI of the concept in an ontology
definition	Definition of the term or info from a data source
usage	Value that tells whether the term is obsolete or in use
comment	Any additional information concerning the term

Table 3
Term card properties.

3.2 Terminology creation

A terminology is represented by a term ontology, where the notion of the term is defined by the class Term. The class Term has the properties of table 3. They are inherited by the term instances called *term cards*. A term card associates a term as a string with an URI in an ontology represented as the value of the property *concept*. Both *singular* and *plural* forms are stored explicitly for two reasons. First, this eliminates the need for Finnish morphological analysis that is complex even when making the singular/plural distinction. Second, singular and plural forms are used with different meaning in Finnish thesauri. For example, the plural term “operas” would typically refer to different compositions and the singular “opera” to the abstract art form. To make the semantic distinction at the term card level, the former term can be represented by a term card with missing singular form and the latter term with missing plural form. Property *definition* is a string representing the definition of the term. Property *usage* is used to indicate obsolete terms in the same way as the USE attribute is used in thesauri. Finally, the *comment* property can be filled to store any other useful information concerning the term, like context information, or the history of the term card.

A term ontology is represented by a Protégé-2000 project that consists of the Term class as an RDF Schema, term instances in RDF, and the referenced ontology represented as an included project.

Initial sets of term cards were created automatically based on the MASA cultural thesaurus and the ontologies of MuseumFinland. The morphological tool MachineSyntax [14] was used for creating plural or singular forms for the term cards when needed.

New term cards are created automatically for unknown terms that are found in artifact record data. The created term cards are automatically filled with contextual

[14] http://www.conexor.fi/m_syntax.html

information concerning the meaning of the term. This information helps the human editor to fill the `concept` property. For example, assume that one has an ontology M of materials and a related terminology T. To enhance the terminology, the material property values of a collection database can be read. If a material term not present in T is encountered, a term card with the new term but without a reference to an ontological concept can be created. A human editor can then define the meaning by making the reference to the ontology.

After this new term cards were extracted by examining the XML cards before transforming them into RDF. Figure 3 depicts the general term extraction process in MUSEUMFINLAND. The process involves a local process at each museum and a global process at MUSEUMFINLAND. The tool Terminator extracts individual term candidates from the museum collection items presented in XML. The entity of one item is called an *term card*. A human editor annotates ambiguous terms or terms not known by the system. The result is a set of new term cards. This set is included in the museum's local terminology and terms of global interest can be included in the global terminology of the whole system for other museums to use.

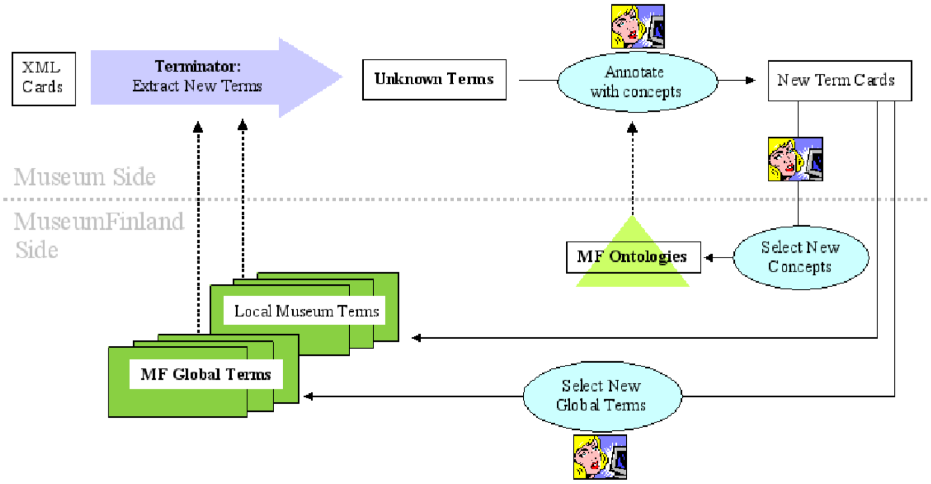


Fig. 3. Creating new term cards in MUSEUMFINLAND.

The global terminology consists of terms that are used in all the museums. It reduces the workload of individual museums, since these terms do not need to be included in local terminologies. The local term base is important because it makes it possible for individual museums to use and maintain their own terminologies.

The global term base can be extended when needed. For example, when creating new terms, it may occur that there is no appropriate concept in the ontologies that a new term can be associated with. In this case, the term is associated with a more general concept and a suggestion is made to MUSEUMFINLAND for extending the ontology later on with a more accurate concept.

A problem of the term creation approach described above is how to deal with complex textual expressions involving several primitive concepts. Some expressions, such as “woman’s dress” have been lexicalized into entries in MASA thesaurus, and have been consequently modeled in the MAO ontology as well as classes. However, there are lots of similar kind of possible expressions whose representation as a class would not be feasible, such as “man’s spectacles”, “nylon wardrobe”, “mixture of cotton and polyamid” etc.

To get insight into this problem, an empirical study was conducted about the textual expressions that were used in describing the artifact type and material fields of the collection objects. Terminator separated terms that could not be identified by using the initial terminology defined in the MASA thesaurus. 413 problematic expressions were found for describing the artifact type in about 4000 descriptions many of which involved two descriptors: the general term, e.g., “trousers”, and its specifier, e.g. “jeans”. The most common problems were: Complex pharases (36%), e.g. “sleeping bag with a zip”; combining user and artifact descriptions (25%), e.g., “child’s hat”; combining material and artifact descriptions (20%), e.g. “leather gloves”; combining usage and artifact information (18%), e.g., “sport shirt”; combining manufacturing technique and artifact type (7%), e.g., “woven shirt”. Common problems in the descriptions of the material field were: Confusing trademarks and materials, e.g., “banlon” vs. “polyamid”; spelling errors; using inflected morphological forms (partitive forms); mixing material and form information, e.g., “cotton fabric”; multiple descriptions, e.g., “nylon and wool”; mixing material with numerical information, e.g., “87% cotton”.

The problem of associating complex unknown descriptions with ontological URIs was solved in two ways. First, if the complex description seemed to be used more than once, a corresponding term card was created to take care of other instances of such descriptions. Second, unique, erroneous and confusing descriptions were annotated by hand when encountered later while transforming the XML cards into RDF cards by Annomobile.

3.3 Semantic annotation based on ontologies

The last step in the content creation process (cf. fig. 2) is creation of the semantically interoperable RDF cards based on the XML cards. Interoperability is obtained by replacing—using term cards—literal data values on the XML level with the ontological concepts and individuals on the RDF level. For example, the XML card presented in page 11 would translate into the RDF card below:

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:card="http://www.fms.fi/RDFCard#">
 <card:RDFCard
   rdf:about="http://www.fms.fi/rdfCard#card11023">
   card:artifactId="16851"
   card:artifactType-www="calendar"
   card:artifactType="http://www.fms.fi/artifacts#calendar"
   card:museum-www="Espoo City Museum"
   card:museum="http://www.fms.fi/agents#EspooCityMuseum"
   card:material-www="cardboard"
   card:material="http://www.fms.fi/materials#cardboard"
   ...
 </card:RDFCard>
...
</rdf:RDF>
```

The elements of the XML cards fall in two categories: *literal features* and *ontological features*. Literal features are to be represented only as literal values on the RDF level, too. They are, for example, used in the user interface. Ontological feature values need to be linked to not only literal values but to ontological resources (URIs), too. For example, in the above RDF card the feature `artifactId` is literal and is not connected with the ontology resources. In contrast, the ontological feature `material` is represented with the literal property `material-www` and the ontological property `material` that has an RDF resource (URI) as its value. This URI connects the card resource with the material ontology and through it with other resources.

The XML to RDF transformation is done by the tool Annomobile. Its transformation algorithm [14] creates for each XML card feature *f*, represented as an XML element, a corresponding RDF triple with the corresponding predicate name *f*-www and literal object value. For ontological features, an additional triple is created. It has the same predicate name *f* and the object value consists of the URIs of the possible resources that the literal feature value may refer to according to the term cards. Each ontological feature is associated with one or more ontologies defining the range of possible feature values, as presented in table 4. The range restrictions are used for disambiguating homonymous terms referring to resources in different ontologies. For example, the meaning of the Finnish term “villa” as a kind of residence can be excluded, if the term is used as a value of the material feature, where it means wool.

When mapping ontological feature values to URIs in domain ontologies, two major problem situations occur related to 1) unknown values and 2) homonyms. In case of unknown values, there are no applicable term card candidates in the terminology. The solution to this is to map the feature value either to a more general concept or to a resource considered unknown. For example, if one knows that an artifact was created in some city in Lapland, one can create an unknown instance of the class City, tell that it is a part of Lapland, and annotate the place-of-manufacture feature with this instance.

The problem of homonymous terms occurs when there are homonyms within the range of ontologies used for annotating the ontological feature at hand. For example, the Finnish literal term “kilvet” as a value of the artifact-type feature, can mean either a signboard or a coat of arms, and these interpretations cannot be disambiguated by using the range information of table 4. The solution employed in Annomobile is to fill the RDF card with all potential choices, inform the human

Ontological feature	**Range ontology**
artifact-type	Artifacts
material	Materials
keyword	Any ontology
creator	Actors
place-of-creation	Locations
time-of-creation	Times
user	Actors
place-of-usage	Locations
situation	Situations
collection	Collections

Table 4
Major ontological feature ranges of artifacts in MUSEUMFINLAND.

	Museum 1	Museum 2	Museum 3
Total of annotated items	1354	1682	3010
Items with homonyms	543 (72,43%)	470 (27,94%)	529 (17,57%)
of which disambiguated	426 (70,61%)	407 (24,20%)	389 (12,92%)
of which not disambiguated	116 (8,57%)	63 (3,75%)	140 (4,65%)

Table 5
Results of disambiguating ontological feature values in MUSEUMFINLAND.

editor of the problem, and ask him to remove the false interpretations on the RDF card manually.

Our first experiments indicate, that at least in Finnish not much manual disambiguation work is needed, since homonymy typically occurs between terms referring to different domain ontologies. However, the problem still remains in some cases and is likely to be more severe in languages like English having more homonymy.

Table 5 shows some statistical results that were obtained in an annotation process experiment. The number of museum collection items totaled 6046, and every item had nine features on the average that needed to be linked to ontological concepts by Annomobile. All fields could contain multiple literal values, all of which should be linked to different ontological concepts. For example, the place-of-usage feature could contain several location names. The table indicates that homonyms occur quite often in the data, but in most cases they belong to different domains, and the simple disambiguation scheme based on feature value ranges worked fairly well in practice.

4 End-User's Perspective

MUSEUMFINLAND provides the end-user with two services:

- *A semantic view-based search engine* that is based on the underlying knowledge base consisting of ontologies and instance data.
- *A semantic linking system* by which the user can find out semantic associations within the portal content, and use the associations for browsing.

In this section we describe these knowledge-based services from the end-user's viewpoint. The services are provided to the end-user via two different user interfaces: one for desktop computers and one for mobile devices. In below the desktop computer web interface is first presented.

4.1 A semantic view-based search engine

The search engine of MUSEUMFINLAND is based on the multi-facet search paradigm [28, 9]. Here the concepts used for indexing are called *categories* and are organized systematically into a set of hierarchical, orthogonal taxonomies. The taxonomies are called subject *facets* or *views*. In multi-facet search the views are exposed to the end-user in order to provide her/him with the right query vocabulary and for presenting the repository contents and search results along different views.

Each category is related to a set of search objects that we will call its *projection*. The *extension* E of a category is the union of its projection P and the extensions of its subcategories S_i: $E = P \cup S_1 \cup S_2 \cup \ldots \cup S_n$. A search query in multi-facet view-based search is formulated by selecting categories of interest from the different facets, typically one selection from a facet. The answer to the query is simply the intersection of the extensions E_i of the selected categories: $A = \cap\{E_i\}$. For example, by selecting the category "Chairs" from the Artifact facet, and "Helsinki" from the Place of Manufacturing facet, the user can express the query for retrieving all chairs (of any subtype) manufactured in Helsinki (or in any of its suburbs and other locations within Helsinki).

MUSEUMFINLAND classifies the collection items along nine views organized in four groups, as presented in table 6. The Artifact Views describe the physical aspects of the collection item (artifact type and materials). The Creation Views tell who manufactured or created the artifact, as well as the location and time of the creation. The Usage Views indicate the user of the artifact, place of usage, and situations in which the artifact is used. Finally, the Collection View classifies the museums and collections participating in the portal.

The novelty of MUSEUMFINLAND with respect to traditional view-based search systems lies in the use of ontologies. The nine views are projected from the seven ontologies of table 1 by a set of logical rules to be discussed later in more detail.

View type	**Facet view**	**Underlying ontology**
Artifact	Artifact type	Artifacts
	Material	Materials
Creation	Creator	Actors
	Location of creation	Locations
	Time of creation	Times
Usage	User	Actors
	Location of usage	Locations
	Situation of usage	Situations
Museum	Collection	Collections

Table 6
The nine view facets in the MUSEUMFINLAND portal based on the domain ontologies of table 1.

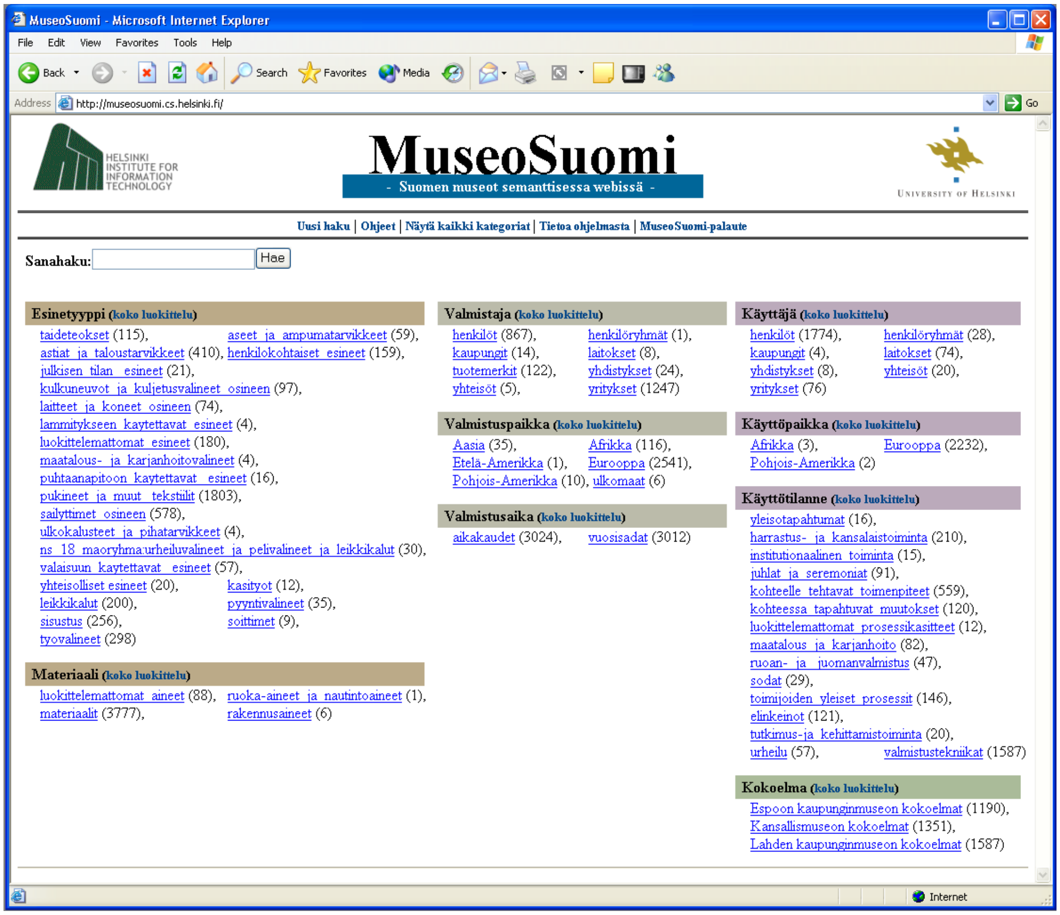
Facets can be used for helping the user in information retrieval in many ways. First, the facet hierarchies give the user an overview of what kind of information there is in the repository. Second, the hierarchies can guide the user in formulating the query in terms of appropriate concepts. Third, the hierarchies do not suffer from the problems of homonymous query terms. Fourth, the facets can be used as a navigational aid when browsing the database content [9]. Fifth, the number of hits in every category that can be selected next can be computed *beforehand* and be shown to the user [28]. In this way, the user can be hindered from making a selection leading to an empty result set—a recurring problem in information retrieval systems—and is guided toward selections that are likely to constrain (or relax) the search appropriately.

4.1.1 Search interface

Figure 4 shows the initial search interface of MUSEUMFINLAND. The nine facets are shown (in Finnish), such as Artifact type ("Esinetyyppi") and Material ("Materiaali"). For each facet, the next level of sub-categories is shown as a set of links. A query is formulated by selecting a category by clicking on its name. When the user selects a category c in a facet f, the system constrains the search by leaving in the result set only such objects that are annotated in facet f with some sub-category of c or c itself. For example, figure 5 depicts the situation after selecting the sub-category Tools ("työvälineet") from the Artifact type facet ("Esinetyyppi"). Now, the interface shows the nine facets on the left, while result set of the made selection is shown on the right.

The result set is grouped by the sub-categories of Tools, such as Textile making tools (“tekstiilityövälineet”) and Tools of folk medicine (“kansanlääkinnän työvälineet”). Hits in different categories are separated by horizontal bars and can be viewed page by page independently in each category. The number of hits shown in each sub-category is determined from the number of sub-categories in the result set in order to maximize useful information on the limited screen space. In this case, all subcategories do not fit on the screen, and only a single line of hits is shown for each subcategory.

By default, the results of the search are grouped by the subcategories of the last selection, but the system also supports grouping along the other views. Items in the result set that do not belong in any of the groups are gathered in an “Other hits” group. For example, in the situation of figure 5, grouping by Museum Collection would provide the user a quick and intuitive view on what tools there are in each collection of the participating museums.


MuseoSuomi - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
Media
Address http://museosuomi.cs.helsinki.fi/
Go
HELSINKI INSTITUTE FOR INFORMATION TECHNOLOGY
UNIVERSITY OF HELSINKI
Hae
Internet

MuseoSuomi

- Suomen museot semanttisessa webissä -

Uusi haku | Ohjeet | Näytä kaikki kategoriat | Tietoa ohjelmasta | MuseoSuomi-palaute

Sanahaku:

Esinetyyppi (koko luokittelu)

taideteokset (115), aseet ja ampumatarvikkeet (59),
astiat ja taloustarvikkeet (410), henkilokohtaiset esineet (159),
julkisen tilan esineet (21),
kulkuneuvot ja kuljetusvalineet osineen (97),
laitteet ja koneet osineen (74),
lammitykseen kaytettavat esineet (4),
luokittelemattomat esineet (180),
maatalous- ja karjanhoitovalineet (4),
puhtaanapitoon kaytettavat esineet (16),
pukineet ja muut tekstiilit (1803),
sailyttimet osineen (578),
ulkokalusteet ja pihatarvikkeet (4),
ns 18 maoryhma:urheiluvalineet ja pelivalineet ja leikkikalut (30),
valaisuun kaytettavat esineet (57),
yhteisolliset esineet (20), kasityot (12),
leikkikalut (200), pyyntivalineet (35),
sisustus (256), soittimet (9),
tyovalineet (298)

Materiaali (koko luokittelu)

luokittelemattomat aineet (88), ruoka-aineet ja nautintoaineet (1),
materiaalit (3777), rakennusaineet (6)

Valmistaja (koko luokittelu)

henkilöt (867), henkilöryhmät (1),
kaupungit (14), laitokset (8),
tuotemerkit (122), yhdistykset (24),
yhteisöt (5), yritykset (1247)

Valmistuspaikka (koko luokittelu)

Aasia (35), Afrikka (116),
Etelä-Amerikka (1), Eurooppa (2541),
Pohjois-Amerikka (10), ulkomaat (6)

Valmistusaika (koko luokittelu)

aikakaudet (3024), vuosisadat (3012)

Käyttäjä (koko luokittelu)

henkilöt (1774), henkilöryhmät (28),
kaupungit (4), laitokset (74),
yhdistykset (8), yhteisöt (20),
yritykset (76)

Käyttöpaikka (koko luokittelu)

Afrikka (3), Eurooppa (2232),
Pohjois-Amerikka (2)

Käyttötilanne (koko luokittelu)

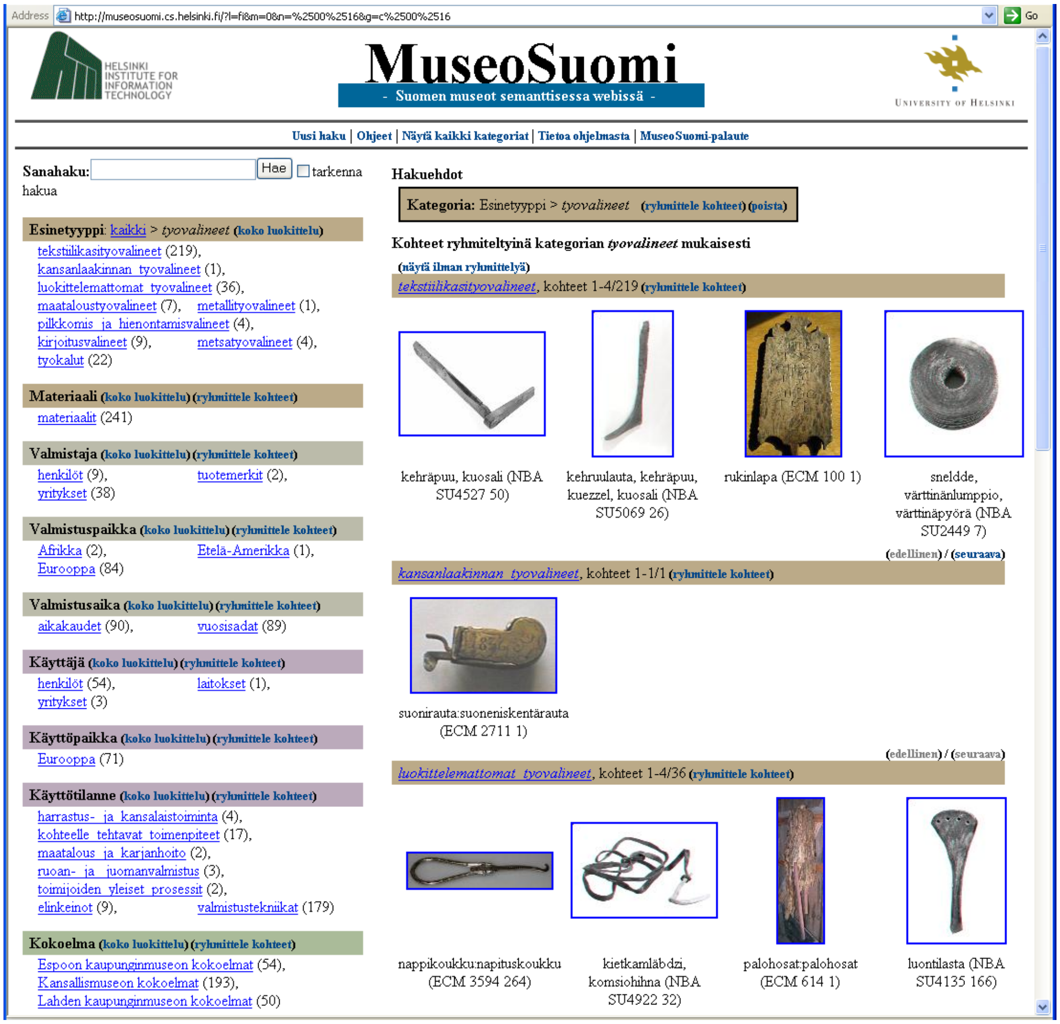
yleisotapahtumat (16),
harrastus- ja kansalaistoiminta (210),
institutionaalinen toiminta (15),
juhlat ja seremoniat (91),
kohteelle tehtavat toimenpiteet (559),
kohteessa tapahtuvat muutokset (120),
luokittelemattomat prosessikasitteet (12),
maatalous ja karjanhoito (82),
ruoan- ja juomanvalmistus (47),
sodat (29),
toimijoiden yleiset prosessit (146),
elinkeinot (121),
tutkimus-ja kehittamistoiminta (20),
urheilu (57), valmistustekniikat (1587)

Kokoelma (koko luokittelu)

Espoon kaupunginmuseon kokoelmat (1190),
Kansallismuseon kokoelmat (1351),
Lahden kaupunginmuseon kokoelmat (1587)

Fig. 4. The initial search interface of MUSEUMFINLAND with its nine facets.

The user can refine the query further by selecting another category on the left. For example, assume that the user selects category Farming and cattle tending (“Maatalous ja karjanhoito”) in the view Situation of usage in figure 5. When answering a query, three things happen. First, the result set on the right is refined to the in-


Address http://museosuomi.cs.helsinki.fi/?l=fi&m=0&n=%2500%2516&g=c%2500%2516
Go
HELSINKI INSTITUTE FOR INFORMATION TECHNOLOGY
MuseoSuomi
- Suomen museot semanttisessa webissä -
UNIVERSITY OF HELSINKI
Uusi haku | Ohjeet | Näytä kaikki kategoriat | Tietoa ohjelmasta | MuseoSuomi-palaute
Hae

Sanahaku: Hae ☐ tarkenna hakua

Esinetyyppi: kaikki > *tyovalineet* (koko luokittelu)
tekstiilikasityovalineet (219),
kansanlaakinnan tyovalineet (1),
luokittelemattomat tyovalineet (36),
maataloustyovalineet (7), metallityovalineet (1),
pilkkomis ja hienontamisvalineet (4),
kirjoitusvalineet (9), metsatyovalineet (4),
tyokalut (22)

Materiaali (koko luokittelu) (ryhmittele kohteet)
materiaalit (241)

Valmistaja (koko luokittelu) (ryhmittele kohteet)
henkilöt (9), tuotemerkit (2),
yritykset (38)

Valmistuspaikka (koko luokittelu) (ryhmittele kohteet)
Afrikka (2), Etelä-Amerikka (1),
Eurooppa (84)

Valmistusaika (koko luokittelu) (ryhmittele kohteet)
aikakaudet (90), vuosisadat (89)

Käyttäjä (koko luokittelu) (ryhmittele kohteet)
henkilöt (54), laitokset (1),
yritykset (3)

Käyttöpaikka (koko luokittelu) (ryhmittele kohteet)
Eurooppa (71)

Käyttötilanne (koko luokittelu) (ryhmittele kohteet)
harrastus- ja kansalaistoiminta (4),
kohteelle tehtavat toimenpiteet (17),
maatalous ja karjanhoito (2),
ruoan- ja juomanvalmistus (3),
toimijoiden yleiset prosessit (2),
elinkeinot (9), valmistustekniikat (179)

Kokoelma (koko luokittelu) (ryhmittele kohteet)
Espoon kaupunginmuseon kokoelmat (54),
Kansallismuseon kokoelmat (193),
Lahden kaupunginmuseon kokoelmat (50)

Hakuehdot

Kategoria: Esinetyyppi > *tyovalineet* (ryhmittele kohteet) (poista)

Kohteet ryhmiteltyinä kategorian *tyovalineet* mukaisesti

(näytä ilman ryhmittelyä)

tekstiilikasityovalineet, kohteet 1-4/219 (ryhmittele kohteet)

kehräpuu, kuosali (NBA SU4527 50)
kehruulauta, kehräpuu, kuezzel, kuosali (NBA SU5069 26)
rukinlapa (ECM 100 1)
sneldde, värttinänlumppio, värttinäpyörä (NBA SU2449 7)

(edellinen) / (seuraava)

kansanlaakinnan tyovalineet, kohteet 1-1/1 (ryhmittele kohteet)

suonirauta:suoneniskentärauta (ECM 2711 1)

(edellinen) / (seuraava)

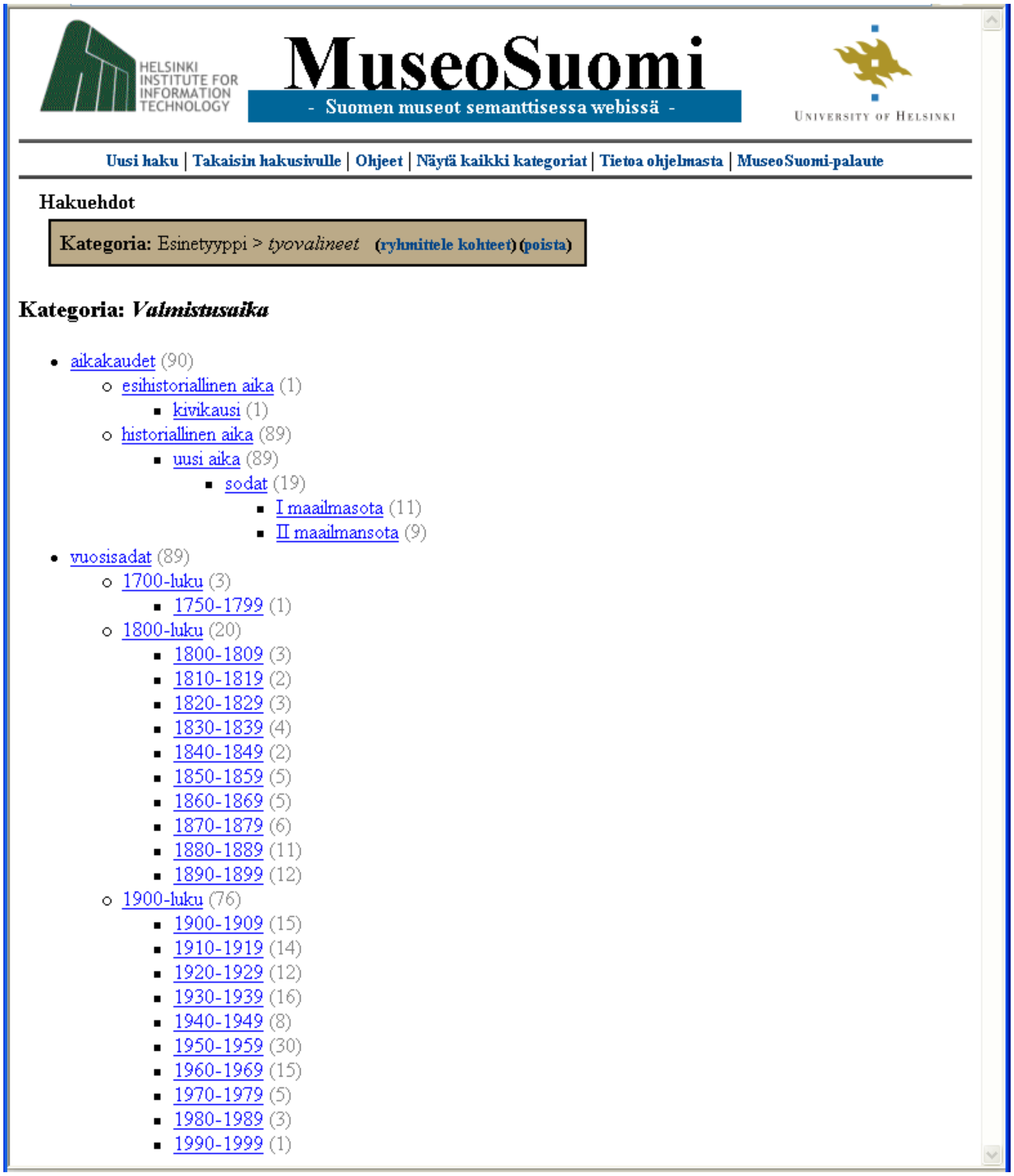
luokittelemattomat tyovalineet, kohteet 1-4/36 (ryhmittele kohteet)

nappikoukku:napituskoukku (ECM 3594 264)
kietkamläbdzi, komsiohihna (NBA SU4922 32)
palohosat:palohosat (ECM 614 1)
luontilasta (NBA SU4135 166)

Fig. 5. MUSEUMFINLAND search interface after selecting the category link Tools ("työvälineet").

tersection of previous selections; here the result is tools used in farming and cattle tending. Second, the selected view is changed to expose the subcategories of the selected category. Third, the size *n* of the result set resulting from the selection of any category link seen on the screen is recomputed proactively, and a number (*n*) is shown to the user after the category name. This number tells that if the category is selected next, then there will be *n* hits in the result set. For example, in figure 5, the number 193 in the Collection facet ("Kokoelma") on the bottom tells that there are 193 tools in the collections of the National Museum ("Kansallismuseon kokoelmat"). A selection leading to an empty result set ($n = 0$) is removed from the facet (or alternatively disabled and shown grayed out, depending on the user's preference). In this way, the user is hindered from making a selection leading to an empty result set, and is guided toward selections that are likely to constrain the search appropriately. The query can be relaxed by making a new selection on a higher level in the facet or by dismissing the facet totally from the query.

Above, the category selection was made among the direct sub-categories listed in the facets. An alternative way is to click on the link Whole facet (“koko luokittelu”) on a facet. The system then shows all possible selections in the whole facet hierarchy with hit counts. For example, if the user selects in the situation of figure 5 the link Whole facet of the facet Time of Creation (“Valmistusaika”), the system shows how the tools in the current result set are classified according to the selected facet (see figure 6). This gives the user a good overview of the distribution of items over a desired dimension. By using the option of only graying out categories with no hits, it is also immediately see in what categories the collections are lacking artifacts. This may be a useful pease of information for, e.g., the collection manager.


HELSINKI INSTITUTE FOR INFORMATION TECHNOLOGY
MuseoSuomi
- Suomen museot semanttisessa webissä -
University of Helsinki
Uusi haku | Takaisin hakusivulle | Ohjeet | Näytä kaikki kategoriat | Tietoa ohjelmasta | MuseoSuomi-palaute

Hakuehdot

Kategoria: Esinetyyppi > *tyovalineet* (ryhmittele kohteet) (poista)

Kategoria: *Valmistusaika*

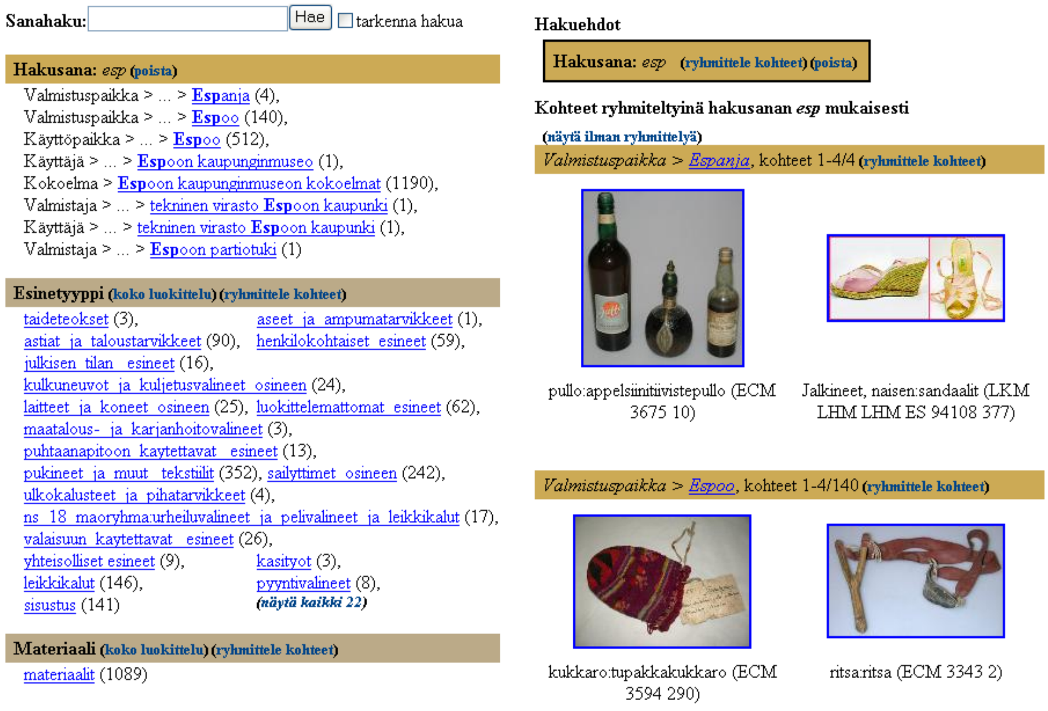
- aikakaudet (90)
  - esihistoriallinen aika (1)
    - kivikausi (1)
  - historiallinen aika (89)
    - uusi aika (89)
      - sodat (19)
        - I maailmasota (11)
        - II maailmansota (9)
- vuosisadat (89)
  - 1700-luku (3)
    - 1750-1799 (1)
  - 1800-luku (20)
    - 1800-1809 (3)
    - 1810-1819 (2)
    - 1820-1829 (3)
    - 1830-1839 (4)
    - 1840-1849 (2)
    - 1850-1859 (5)
    - 1860-1869 (5)
    - 1870-1879 (6)
    - 1880-1889 (11)
    - 1890-1899 (12)
  - 1900-luku (76)
    - 1900-1909 (15)
    - 1910-1919 (14)
    - 1920-1929 (12)
    - 1930-1939 (16)
    - 1940-1949 (8)
    - 1950-1959 (30)
    - 1960-1969 (15)
    - 1970-1979 (5)
    - 1980-1989 (3)
    - 1990-1999 (1)

Fig. 6. The Time facet hierarchy classifying the result set of tools in figure 5.

4.1.2 Combining keyword and view-based search

View-based search does not a panacea for information retrieval. Google-like keyword search interface is usually preferred [4] if the user is capable of expressing her information need terms of accurate keywords. MUSEUMFINLAND seamlessly integrates this functionality with view-based search in the following way: First, the search keywords are matched against category names in the facets in addition to text fields in the metadata. A new dynamic facet is created in the user interface. This facet contains all facet categories whose name (or other property values) matches the keyword. Intuitively these facet categories tell the different interpretations of the keyword, and by selecting one of them next the right choice can be made. This approach also solves the search problem of finding relevant categories in facets that contain thousands of categories. Second, a result set of object hits is shown. This result set contains all objects contained in any of the categories matched in addition to all objects whose metadata directly contains the keyword. The hits are grouped by the categories found.

The result of a sample keyword search is shown in figure 7. Here, a search for "esp" has matched, for example, the categories Spain ("Espanja" in Finnish) and Espoo in the facet Location of Creation and the category Espoo City Museum ("Espoon kaupunginmuseo") in the facet User ("Käyttäjä"). The categories found can be used to constrain the multi-facet search as normal, with the distinction that selections from the dynamic facet replace selections in their corresponding facets and dismiss the dynamic facet.



Sanahaku: Hae □ tarkenna hakua

Hakusana: *esp* (**poista**)

Valmistuspaikka > ... > **Esp**anja (4),
Valmistuspaikka > ... > **Esp**oo (140),
Käyttöpaikka > ... > **Esp**oo (512),
Käyttäjä > ... > **Esp**oon kaupunginmuseo (1),
Kokoelma > **Esp**oon kaupunginmuseon kokoelmat (1190),
Valmistaja > ... > tekninen virasto **Esp**oon kaupunki (1),
Käyttäjä > ... > tekninen virasto **Esp**oon kaupunki (1),
Valmistaja > ... > **Esp**oon partiotuki (1)

Esinetyyppi (**koko luokittelu**) (**ryhmittele kohteet**)

taideteokset (3), aseet ja ampumatarvikkeet (1),
astiat ja taloustarvikkeet (90), henkilokohtaiset esineet (59),
julkisen tilan esineet (16),
kulkuneuvot ja kuljetusvalineet osineen (24),
laitteet ja koneet osineen (25), luokittelemattomat esineet (62),
maatalous- ja karjanhoitovalineet (3),
puhtaanapitoon kaytettavat esineet (13),
pukineet ja muut tekstiilit (352), sailyttimet osineen (242),
ulkokalusteet ja pihatarvikkeet (4),
ns 18 maoryhma:urheiluvalineet ja pelivalineet ja leikkikalut (17),
valaisuun kaytettavat esineet (26),
yhteisolliset esineet (9), kasityot (3),
leikkikalut (146), pyyntivalineet (8),
sisustus (141) (***näytä kaikki 22***)

Materiaali (**koko luokittelu**) (**ryhmittele kohteet**)

materiaalit (1089)

Hakuehdot

Hakusana: *esp* (**ryhmittele kohteet**) (**poista**)

Kohteet ryhmiteltyinä hakusanan *esp* mukaisesti

(**näytä ilman ryhmittelyä**)

Valmistuspaikka > Espanja, kohteet 1-4/4 (**ryhmittele kohteet**)

pullo:appelsiinitiivistepullo (ECM 3675 10)

Jalkineet, naisen:sandaalit (LKM LHM LHM ES 94108 377)

Valmistuspaikka > Espoo, kohteet 1-4/140 (**ryhmittele kohteet**)

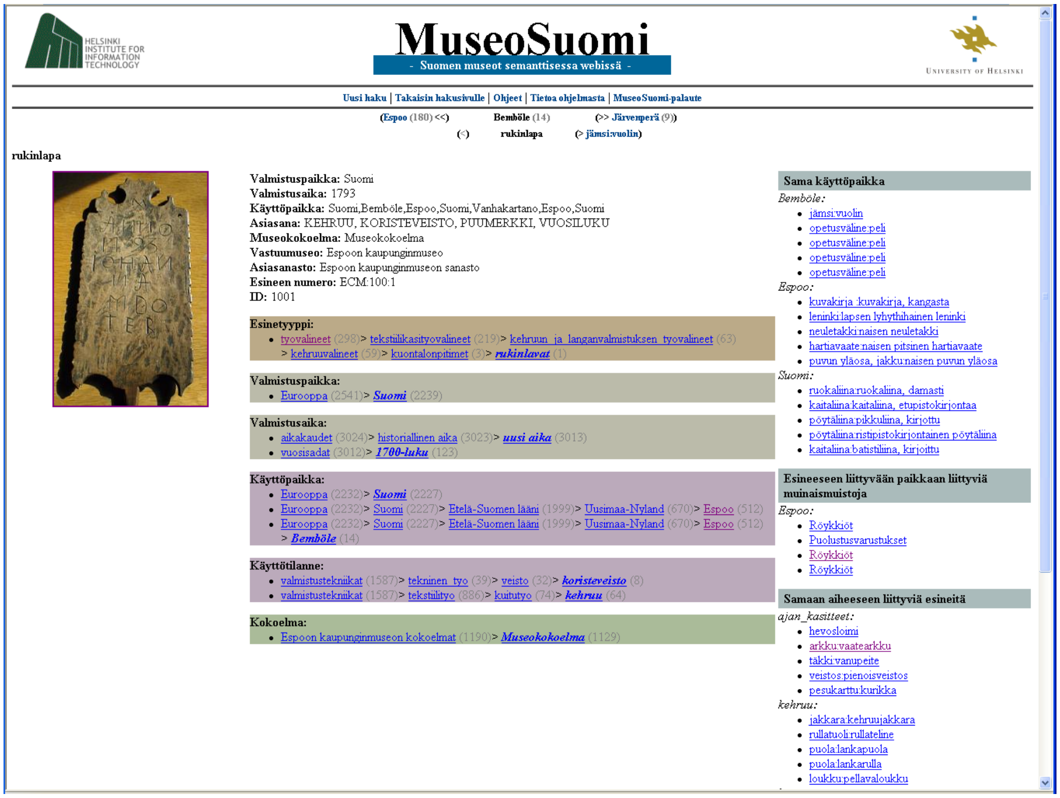
kukkaro:tupakkakukkaro (ECM 3594 290)

ritsa:ritsa (ECM 3343 2)

Fig. 7. Using the keyword search for finding categories.

At any point during multi-facet search the user can select any hit found by clicking on its image. The corresponding data object is then shown as a web page, such as the one in figure 8. The example depicts a special part, distaff (“rukinlapa” in Finnish) used in a spinning wheel. The page contains the following information and links:

(1) On top, there are links to directly navigate in the groups and results of the current query.
(2) The image(s) of the object is (are) depicted on the left.
(3) The metadata of the object is shown in the middle on top.
(4) All facet categories that the object is annotated with are listed in the middle bottom as hierarchical link paths. A new search can be started by selecting any category there.
(5) A set of semantic links to related artifacts is shown on the right.



HELSINKI INSTITUTE FOR INFORMATION TECHNOLOGY

MuseoSuomi

- Suomen museot semanttisessa webissä -

UNIVERSITY OF HELSINKI

Uusi haku | Takaisin hakusivulle | Ohjeet | Tietoa ohjelmasta | MuseoSuomi-palaute

(Espoo (180) <<) Bemböle (14) (>> Järvenperä (9))

(<) rukinlapa (> jämsi:vuolin)

rukinlapa

Valmistuspaikka: Suomi
Valmistusaika: 1793
Käyttöpaikka: Suomi,Bemböle,Espoo,Suomi,Vanhakartano,Espoo,Suomi
Asiasana: KEHRUU, KORISTEVEISTO, PUUMERKKI, VUOSILUKU
Museokokoelma: Museokokoelma
Vastuumuseo: Espoon kaupunginmuseo
Asiasanasto: Espoon kaupunginmuseon sanasto
Esineen numero: ECM:100:1
ID: 1001

Esinetyyppi:
- tyovalineet (298)> tekstiilikasityovalineet (219)> kehruun_ja_langanvalmistuksen_tyovalineet (63) > kehruuvalineet (59)> kuontalonpitimet (3)> ***rukinlavat*** (1)

Valmistuspaikka:
- Eurooppa (2541)> ***Suomi*** (2239)

Valmistusaika:
- aikakaudet (3024)> historiallinen aika (3023)> ***uusi aika*** (3013)
- vuosisadat (3012)> ***1700-luku*** (123)

Käyttöpaikka:
- Eurooppa (2232)> ***Suomi*** (2227)
- Eurooppa (2232)> Suomi (2227)> Etelä-Suomen lääni (1999)> Uusimaa-Nyland (670)> Espoo (512)
- Eurooppa (2232)> Suomi (2227)> Etelä-Suomen lääni (1999)> Uusimaa-Nyland (670)> Espoo (512) > ***Bemböle*** (14)

Käyttötilanne:
- valmistustekniikat (1587)> tekninen_tyo (39)> veisto (32)> ***koristeveisto*** (8)
- valmistustekniikat (1587)> tekstiilityo (886)> kuitutyo (74)> ***kehruu*** (64)

Kokoelma:
- Espoon kaupunginmuseon kokoelmat (1190)> ***Museokokoelma*** (1129)

Sama käyttöpaikka

Bemböle:
- jämsi:vuolin
- opetusväline:peli
- opetusväline:peli
- opetusväline:peli
- opetusväline:peli

Espoo:
- kuvakirja :kuvakirja, kangasta
- leninki:lapsen lyhythihainen leninki
- neuletakki:naisen neuletakki
- hartiavaate:naisen pitsinen hartiavaate
- puvun yläosa, jakku:naisen puvun yläosa

Suomi:
- ruokaliina:ruokaliina, damasti
- kaitaliina:kaitaliina, etupistokirjontaa
- pöytäliina:pikkuliina, kirjottu
- pöytäliina:ristipistokirjontainen pöytäliina
- kaitaliina:batistiliina, kirjoittu

Esineeseen liittyvään paikkaan liittyviä muinaismuistoja

Espoo:
- Röykkiöt
- Puolustusvarustukset
- Röykkiöt
- Röykkiöt

Samaan aiheeseen liittyviä esineitä

ajan_kasitteet:
- hevosloimi
- arkku:vaatearkku
- täkki:vanupeite
- veistos:pienoisveistos
- pesukarttu:kurikka

kehruu:
- jakkara:kehruujakkara
- rullatuoli:rullateline
- puola:lankapuola
- puola:lankarulla
- loukku:pellavaloukku

Fig. 8. Web page depicting a collection object, its metadata, facet categories, and semantic recommendation links to other collection object pages.

The semantic links on the right reveal to the end-user a most interesting aspect of the collection items: the implicit semantic relations that relate collection data with their context and each other. The links provide a *semantic browsing* facility to the end-user. For example, in figure 8 there are links to objects used at the same location (categorized according to the name of the common location), to objects related to similar events (e.g., objects used in spinning, and objects related to concepts of time, because the distaff in question has a date carved onto it), to objects manufactured at the same time, and so on. Since a decoratively carved distaff used to be a typical wedding gift in Finland, it is also possible to recommend links to other objects related to the wedding events, such as wedding rings. These associations are exposed to the end-user as link groups whose titles and link names explain to the user the reason for the link recommendation.

4.3 The mobile user interface

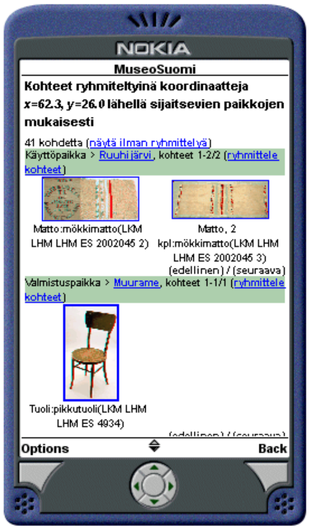
MUSEUMFINLAND has been designed so that the same content and services can easily be rendered to the end-users in different ways. To demonstrate this, we created another user interface for MUSEUMFINLAND to be used by WAP 2.0 (XHTML/MP) compatible devices.

In a mobile environment the adaptability of user interface is important. The MUSEUMFINLAND takes this aspect into concern in following ways. Firstly, empty results can be eliminated, which is a nice feature in an environment where data transfer latencies and costs are still often high. Secondly, the elimination of infeasible choices makes it possible to use the small screen size more efficiently for displaying relevant information. Thirdly, the semantic browsing functionality is a simple and effective navigation method in a mobile environment.

The mobile interface repeats all functionality of the PC interface, but in a layout more suitable to the limited screen space of mobile devices. In addition, to better facilitate finding interesting starting points for browsing, some mobile-specific search shortcuts were created. The search results are shown first up front noting the current search parameters for easy reference and dismissal, as seen in figure 9. Below this, the actual search facets are shown. In the mobile user interface selectable sub-categories are not shown as explicit links as in the PC interface, but as drop-down lists that replace the whole view when selected. This minimizes screen space usage while browsing the facets, but maximizes usability when selecting sub-categories from them. In-page links are provided for quick navigation between search results and the search form.

The item page (corresponding to figure 8) is organized in a similar fashion, showing first the item name, images, metadata, annotations, semantic recommendations, and finally navigation in the search result. There are also in-page links for jumping quickly between the different parts of the page.

The mobile user interface also provides two distinct services aimed specifically for mobile use. First, the interface supports search by the geographical location of the mobile device in the same manner as in the concept-based keyword search. Any



MuseoSuomi

Kohteet ryhmiteltyinä koordinaatteja *x*=62.3, *y*=26.0 lähellä sijaitsevien paikkojen mukaisesti

41 kohdetta (näytä ilman ryhmittelyä)

Käyttöpaikka > Ruuhijärvi, kohteet 1-2/2 (ryhmittele kohteet)

Matto:mökkimatto(LKM LHM LHM ES 2002045 2)

Matto, 2 kpl:mökkimatto(LKM LHM LHM ES 2002045 3)

(edellinen) / (seuraava)

Valmistuspaikka > Muurame, kohteet 1-1/1 (ryhmittele kohteet)

Tuoli:pikkutuoli(LKM LHM LHM ES 4934)

(edellinen) / (seuraava)

Options **Back**

Fig. 9. Results of a mobile geolocation search initiated somewhere near Ruuhijärvi, Finland.

entries in the Location ontology near the current location of the mobile user are shown in a dynamic facet as well as all data objects made *or* used in any of these locations. In addition, any objects directly annotated with geographical coordinates near the mobile user are shown grouped as normal. This feature gives the user a one-click path to items of likely immediate interest. Second, because navigation and search with mobile devices is tedious, any search state can be "bookmarked", sent by email to a desired address, and inspected later in more detail by using the more convenient PC interface.

5 Adapting services for new content

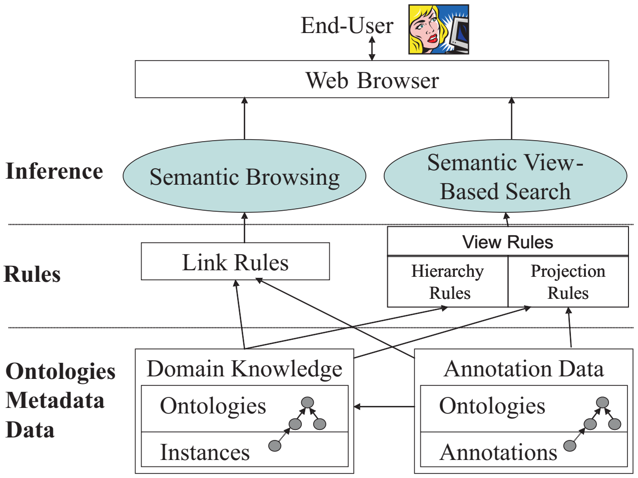


Fig. 10. Architecture of MUSEUMFINLAND on the server side.

Figure 10 depicts the relation between contents and services in MUSEUMFINLAND on the server side. The system is used by a web browser that provides the Semantic view-based search and Semantic browsing services to the end-user. The services are based on two forms of content: 1) Domain Knowledge consists of ontologies (cf. table 1) that define the domain concepts and the individuals. 2) Annotation Data describes the metadata of the data resources (cf. table 2) represented as RDF cards.

A technical innovation of MUSEUMFINLAND is to introduce an intermediate mapping layer of logical rules between the content and semantic services: Link Rules for the browsing service and View Rules for the search engine. By using the rules the generic service engines can be separated from domain and annotation specific details and be adapted to contents of different kind by changing the rules only. The rules are defined declaratively in terms of Prolog predicates operating on RDF triples as in [12].

In the following, the idea of View Rules and Link Rules is described in more detail by using examples. We use SWI-Prolog [15] as the inference engine and SWI-Prolog syntax in the examples [16] .

5.1 *Creating views from ontologies by view rules*

A view is a hierarchical index-like decomposition of category resources where each category is associated with a set of subcategories and a set of directly related data items. A view is defined in terms of ontologies by specifying a view rule predicate called `ontodella_view`. It contains the following information: 1) the root resource URI, 2) a *hierarchy rule* defined by a binary subcategory relation predicate, 3) a binary *projection rule* predicate that maps search objects onto the view categories, and 4) a label for the view. An example of a view rule predicate is given below:

```
ontodella_view(
  ’http://www.cs.helsinki.fi/seco/ns/2004/03/places#earth’,
  place_sub_category, place_of_use_leaf_item,
  [fi:’Käyttöpaikka’, en:’Place of Usage’] % the labels
).
```

Here the URI on the second line is the root resource, `place_sub_category` is the name of the hierarchy subcategory predicate and `place_of_use_leaf_item` is the projection rule predicate. The label list contains the labels for each supported language, here in Finnish (fi) and in English (en).

The root URI defines the resource in a domain ontology that will become the root of the view hierarchy tree, while the hierarchy rule specifies how to construct the

[15] http://www.swi-prolog.org

[16] The syntax used in the examples is translated from Finnish and is slightly simplified for better readability.

facet hierarchies from the domain ontologies. Hierarchy rules are needed in order to make the classifications shown to the user independent from the design choices of the underlying domain ontologies. The view-based search engine itself does not know about the ontologies, it deals with tree-like category hierarchies.

We have used two hierarchy rules to extract a facet from the RDF(S)-based domain knowledge. Firstly, the rdfs:subclassOf hyponymy relation can be used in facets such as Artifact type, and the projection rules map RDF cards of corresponding artifacts to these categories. Second, places constitute a part-of meronymy. Creating views along this dimension is a natural choice for the location facets in the user interface. For example, in the above view rule, the binary subcategory predicate `place_sub_category` can be defined by the containment property `isContainedBy` in the following way:

```
place_sub_category( ParentCategory, SubCategory ) :-
  SubCategoryProperty =
    ’http://www.cs.helsinki.fi/seco/ns/2004/03/places#isContainedBy’,
  rdf( SubCategory, SubCategoryProperty, ParentCategory ).
```

A projection rules tells when an RDF card instance is a member of a category. For example, the rule `place_of_use_leaf_item` in our example above could be defined as follows:

```
place_of_use_leaf_item( ResourceURI, CategoryURI ) :-
  Relation = ’http://www.cs.helsinki.fi/seco/ns/2004/03/artifacts#usedIn’,
  rdf(ResourceURI, Relation, CategoryURI ).
```

Based on hierarchy and projection rules, the view categories can be generated by iterating through the predicate `ontodella_view`, and by recursively creating the category hierarchies using the subcategory rules starting from the given root category. At every category, all relevant resources are attached to the category based on the projection rules.

Hierarchy rules tell how the views are projected logically. A separate question is how these hierarchies should be shown to the user. Firstly, the ordering of the subresources may be relevant. For example, the sub-happenings of an event should be presented in the order in which they take place and persons be listed in alphabetical order. The ordering of the sub-nodes can be specified by a configurable property; the sub-categories are sorted based on the values of this property. Second, one may need a way to filter unnecessary resources away from the user interface. For example, the ontology is typically created partly before the actual annotation work and may have more classes and details than were actually needed. Then empty categories should be pruned out. A hierarchy may also have intermediate classes that are useful for knowledge representation purposes but are not very natural categories to the user. Such categories should be present internally in the search hierarchies but should not be shown to the user. Third, the names for categories need to be specified. For example, the label for a person category should be constructed from the last and first names represented by distinct property values.

5.2 Semantic link rules

Links can be created in various ways [30]. In our work, we have been considering the following alternatives:

- User *profile-based recommendations* are based on information collected by observing the user, or in some cases by asking the user to explicitly define the interest profile. Based on the user's profile, recommendations are then made to the user either by comparing the user's profile to other users' profiles (collaborative filtering/recommending) or by comparing the user's profile to the underlying document collection (content-based recommending). The strength of user profile-based recommendations is that they are personalized and hence serving better the user's individual goals. In MUSEUMFINLAND we decided not to use profiles due to following reasons: First, a precondition for personalization is that the users can be identified which was considered not feasible in MUSEUMFINLAND. Second, profiling is difficult because many users use the system perhaps only once in their lifetime. Finally, it is difficult to identify whether the user likes or dislikes the current data without asking the user to rate every image explicitly. A weakness of collaborative filtering is that explaining the recommendations to the user can be difficult, because they are mostly based on heuristic measures of the similarity between user profiles and database contents, and on the user's actions.
- With *similarity-based recommendations* we refer to the possibility to compare the semantical distance between the metadata of resources. The nearest resources are likely to be of more interest and could be recommended to the user. A difficulty of this recommendation method is how to measure the semantical distance between metadata. The most similar RDF card may not be the most interesting one but rather just another similar artifact. One method is to use the count of common or intersecting annotation resources as a distance measure [34].
- The idea of *rule-based recommendations* is that the domain specialist explicitly describes the notion of "interesting related resource" with generic logic rules. The system then applies the rules to the underlying knowledge base in order to find interesting resources related to the selected one. This method has several strengths. Firstly, the rule can be associated with a label, such as "Other artifacts used in event *x*", that can be used as the explanation for the recommendations found. It is possible to deduce the explanation label as a side effect of applying the rule. Semantic linking rules are described by the domain specialist. The rules and explanations are explicitly defined and are not based on heuristic measures, which could be difficult to understand and motivate. Secondly, the specialist knows the domain and may promote the most important relations between the resources. However, this could also be a weakness if the user's goals

and the specialists thoughts about what is important do not match, and the user is not interested in the recommendations. Thirdly, the rule-based recommendations do not exclude the possibility of using other recommendation methods but provides an infrastructure for applying any rules. For example, the recommendation rules could perhaps be learned by observing the users actions and then used in recommending images for the current or future users.

In a precursor system [20] of MUSEUMFINLAND, we implemented and tested a profile-based and similarity-based recommendation system that recommended semantically similar resources. The recommendations were not static but were modified dynamically by maintaining a user profile and a history log of image selections. Then a rule-based semantic linking system was implemented due to the benefits discussed above and is in use in MUSEUMFINLAND. This link system is described in more detail in the next section.

6 Architecture and implementation

MUSEUMFINLAND has been implemented by using a tool called ONTOVIEWS [17] [25]. This tool was developed during the project but has later been applied to creation of other semantic portals as well [22, 25].

ONTOVIEWS consists of the three major components shown in figure 11:

(1) The logic server ONTODELLA provides the system with reasoning services, such as category view projection and dynamic semantic link recommendations.
(2) The search engine ONTOGATOR is a generic view-based RDF search engine, responsible for the multi-facet search functionality of the system.
(3) The third component ONTOVIEWS-C binds the services of ONTOGATOR and ONTODELLA together, and provides the user interfaces.

More thorough overviews of the three components are given in the following subsections.

6.1 ONTODELLA *rule framework*

A prototype rule framework called ONTODELLA has been developed to provide a logic engine for defining and executing the View and Linking rules of figure 10.

[17] The software is available at http://www.cs.helsinki.fi/group/seco/museums/dist/ in open source.

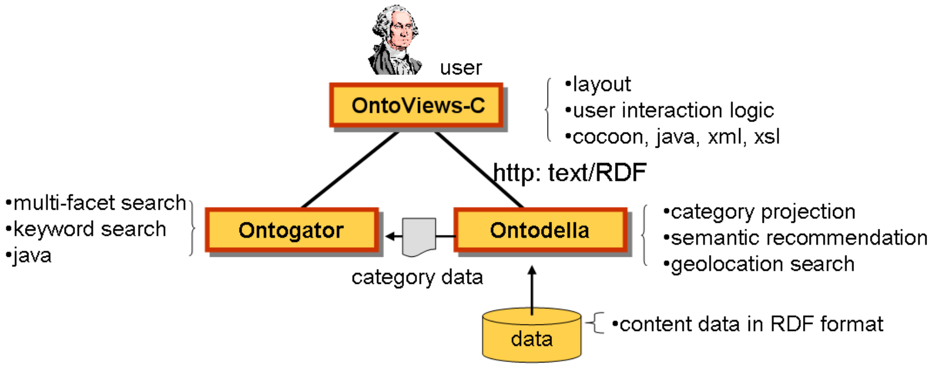


Fig. 11. The components of ONTOVIEWS.

ONTODELLA is a multi-threaded web server which provides remote access to execute the rules in the framework. The web server and the rule execution framework are written using SWI Prolog[18] and its readily available HTTP libraries. For the mobile user interface, ONTODELLA has been extended to provide simple point-of-interest search based on geo-coordinates available from the mobile phone.

ONTODELLA provides services for view creation, semantic link generation, and geolocation search. View creation is done by a separate process before starting MUSEUMFINLAND due to the long time required to execute the hierarchy and projection rules, and due to the size of the view trees. Linking services and geolocation search are run dynamically on request. In below, these services are explained in more detail.

6.1.1 View creation service

View creation service provides necessary hooks for executing the hierarchy and projection predicates. The view creation algorithm traverses the ontologies by using the given predicates dynamically in a depth-first search. The resulting view structure is serialized in RDF/XML according to a model derived from the Annotea Bookmark Schema[19]. This structure is used by ONTOGATOR as the basis for the view-based search. An example of a view category is given below:

```
<ogt:Category>
  <rdfs:label xml:lang="fi">Tapahtuma</rdfs:label>
  <rdfs:label xml:lang="en">Action</rdfs:label>
  <ogt:projectionOf rdf:resource="http://www.seco.org/MAO#prosessit"/>
  <fms:rootcatid>0</fms:rootcatid>
  <rdf:type>
    <rdf:Description rdf:about="http://www.seco.org/MAO#MAOconcept">
      <rdfs:label xml:lang="fi">MAOconcept</rdfs:label>
    </rdf:Description>
  </rdf:type>
  <ogt:subCategories rdf:parseType="Collection">
```

[18] http://www.swi-prolog.org
[19] http://www.w3.org/2003/07/Annotea/BookmarkSchema-20030707

```
    <!-- ... subcategories' ogt:Category-elements within ... -->
  </ogt:subCategories>
  <ogt:topicOf rdf:parseType="Collection">
    <rdf:Description rdf:about="http://www.seco.org/annotaatiot#taulu_Inst_0"/>
    <rdf:Description rdf:about="http://www.seco.org/prosessi#Prosessi_Inst_18888"/>
    <rdf:Description rdf:about="http://www.seco.org/prosessi#Prosessi_Inst_58888"/>
    <!-- ... more items classified into the category ... -->
  </ogt:topicOf>
</ogt:Category>
```

The structure of the serialized categories uses anonymous RDF nodes to represent the view category tree and its projected leaf resources. Each category has as its properties an RDF collection containing the subcategory information (`ogt:subCategories`) and an RDF collection listing out the URIs of the actual resources that have been projected to this specific category (`ogt:topicOf`). All resource properties may have alternative labels (`rdfs:label`) with respective language information added to them, and the same category resource can be visualized using different labels depending on the application. The property `ogt:projectionOf` refers to the original ontology resource corresponding to the category. The property `ogt:topicOf` lists the projection of the category, i.e., resources that belong directly to the category. In addition, all root categories have the property `fms:rootcatid` whose value is an integer identifier. It is used by the user interface for ordering the roots.

The projected resources of a category are represented as Bookmark instances. Each bookmark has its properties and labels listed out like with categories discussed above. A bookmark of the category example above is listed below for illustration:

```
<bm:Bookmark
  rdf:about="http://www.seco.org/prosessi#Prosessi_Inst_18888">
  <ogt:projectionOf
    rdf:resource="http://www.seco.org/prosessi#Prosessi_Inst_18888"/>
  <rdfs:label xml:lang="fi">Kullervon tarina</rdfs:label>
  <fms:MAOprosessi>
    <rdf:Description rdf:about="http://www.seco.org/MAO#kertomukset">
      <rdfs:label xml:lang="fi">kertomukset</rdfs:label>
    </rdf:Description>
  </fms:MAOprosessi>
  <fms:kuvaus>Kullervon tarina on kertomus huono-onnisesta Kullervosta.
  </fms:kuvaus>
  <rdf:type>
    <rdf:Description rdf:about="http://www.seco.org/prosessi#YhdistelmaProsessi">
      <rdfs:label xml:lang="fi">prosessi:YhdistelmaProsessi</rdfs:label>
    </rdf:Description>
  </rdf:type>
</bm:Bookmark>
```

6.1.2 Semantic link service

ONTODELLA also provides a dynamic semantic link service based on linking rules. In response to a semantic linking service request with a given URI, the framework calls for all defined semantic link rules. Each link rule can be arbitrarily complex and is defined by a domain specialist. A linking rule is described by a predicate of the form

$$predicate(SubjectURI,TargetURI,Explanation)$$

that succeeds when the two resources $SubjectURI$ and $TargetURI$ are to be linked. The variable $Explanation$ is then bound to an explanatory label (string) for the link.

In the following, one of the more complex rules — linking items related to a common event— is presented as an example:

```
related_by_event( Subject, Target, Explanation ) :-

ItemTypeProperty =
  ’http://www.cs.helsinki.fi/seco/ns/2004/03/artifacts#item_type’,
ItemTypeToEventRelatingProperty =
  ’http://www.cs.helsinki.fi/seco/ns/2004/03/mapping#related_to_event’,

% check that both URIs correspond in fact to artifacts
isArtifact(Subject), isArtifact(Target),
% and are not the same
Subject \= Target,

% find all the item types the subject item belongs to
rdf(Subject, ItemTypeProperty, SubjectItemType),
rdfs_transitive_subClassOf(SubjectItemType,SubClassOfSubjectItemType),

% find all the events any of those item types are related to
rdf(SubClassOfSubjectItemType, ItemTypeToEventRelatingProperty,
Event),
% and events they include or are part of
(
  rdfs_transitive_subClassOf(Even, SubOrSuperClassOfEvent),
  DescResource=TransitiveSubOrSuperClassOfEvent;
  % or
  rdfs_transitive_subClassOf(SubOrSuperClassOfEvent, Event),
  DescResource=Event;
),

% find all item types related to those events
rdf(TargetItemType, ItemTypeToEventRelatingProperty,
SubOrSuperClassOfEvent),
% and all their superclasses
rdfs_transitive_subClassOf(SuperClassOfTargetItemType,
TargetItemType),

% don’t make uninteresting links between items of the same type
SuperClassOfTargetItemType \= SubjectItemType,
not(rdfs_transitive_subClassOf(SuperClassOfTargetItemType,
SubjectItemType)), not(rdfs_transitive_subClassOf(SubjectItemType,
SuperClassOfTargetItemType)),

% finally, find all items related to the linked item types
rdf(Target, ItemTypeProperty, SuperClassOfTargetItemType),

list_labels([DescResource], RelLabel),
Explanation=[commonResources(DescResource), label(fi:RelLabel)].
```

The rule goes over several ontologies, first discovering the object types of the objects, then traversing the object type ontology, relating the object types to events, and finally traversing the event ontology looking for common resources. Additional checks are made to ensure that the found target is an artifact and that the subject and target are not the same resources. Finally, information about the relation is collected, such as the URI and the label of the common resource, and the result is returned as the link label.

Each rule returns as a result a (possibly empty) set of associated URIs with explanatory labels. The results are grouped according to the rule which generated them and according to the resource that caused the linking. For example, in a rule providing links to collection items manufactured at the same place, the URI of the shared place can be returned as the link causing resource.

ONTODELLA returns the results in XML form that is transformed into HTML by the component ONTOVIEWS-C. In the user interface, the result groups form classified collections of links that can be presented under classification titles subtitled by link causing resources. For example, in the lower right corner of figure 8 there is the title Objects related to the same theme ("Samaan aiheeseen liittyviä esineitä") and under it two subtitles corresponding to two link causing resources: Concepts of time ("ajan käsitteet") and Spinning ("kehruu"). Under the latter subtitle, the first link "jakkara:kehruujakkara" (Spinning chair) points to the web page of a chair used in spinning.

Besides using the subject, also the relation and object parts of the query can be provided as parameters to ONTODELLA linking service. For example, if only a relation URI (i.e., a rule identifier) is given to the semantic links service, then the result will be the set of all semantic links provided by that rule. This might be useful, for example, when debugging the results of new rules.

For every request, all semantic linking rules are evaluated anew so that a maximal number of different links can be generated and are freshly deduced. ONTODELLA provides a mechanism to limit the maximum number of link results, too.

6.1.3 Geolocation search

The Geolocation search gets as input a set of coordinates. In response, the service returns a fixed length ordered list of the location resources nearest to the coordinates, and a corresponding list of bookmarks annotated with the coordinates.

Our current implementation only allows spot coordinate annotation of the searched resources. This scheme is sufficient for resources annotated with precise coordinates, such as ancient burial sites and fortresses out in the field. Many museum objects are, however, often annotated only with a more generic location of variable size and unspecified form, such as the Lapland area. Mapping coordinates onto such ontological categories is not supported in MUSEUMFINLAND yet. For a production quality system, the coordinate to resource mapping service should be implemented in a more generic setting, e.g. within a GIS system.

ONTOGATOR defines and implements an RDF-based query interface that is used to separate view-based search logic from the user interface. The interface is defined as an OWL [20] ontology [21] , and is based on selectors that can be used to query for both view category hierarchies and the projection resources of their categories based on various criteria, such as category, keyword, and geolocation-based constraints. The query is represented in XML/RDF form.

The search result of ONTOGATOR is expressed as an RDF-tree that conforms to a fixed order XML-structure. This allows us to use also XML tools such as XSLT to process the results more easily. Since the search results are used in building user interfaces, every resource is tagged with an rdfs:label.

Figure 12 illustrates what happens in an ONTOGATOR search. The query on the left calls for bookmarks that 1) belong to a subcategory *S* of a view category hierarchy 2 and 2) contain a given *keyword*. The results on the right are grouped according to an independent additiona view hierarchy with the root category *G*. Grouping is based on the next sublevel of *G* as in figure 5. Those bookmarks found that do not belong in the grouping hierarchy are returned in the ungrouped category *U*. In the user interface, the results can be shown in groups 1.1, 1.2, and *U*.

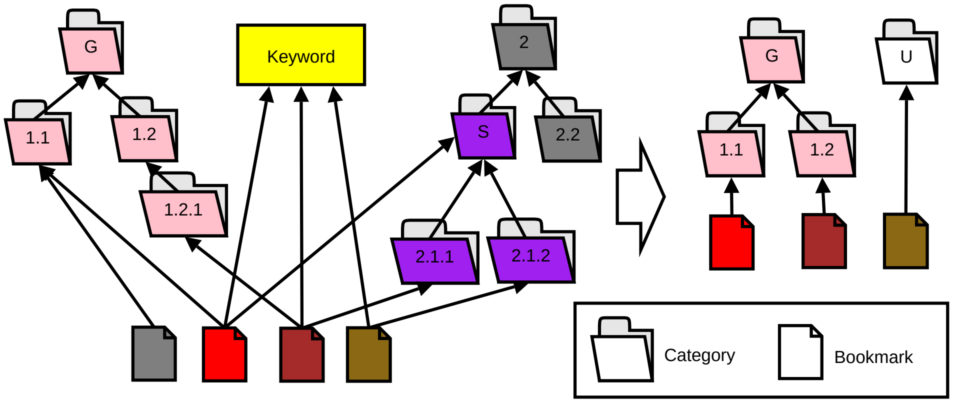


Fig. 12. A keyword plus category selector search with results grouped into an independant, partially cut hierarchy

The RDF query interface allows many options to filter, group, cut, annotate, and otherwise modify the results. An example of a simple ONTOGATOR query in RDF/XML-format is given below. There are three facet selectors each of which specify a category selection, such as `%07%04`. The result is an intersection of the extensions of these categories. Additional facet selector properties are set to formulate the result, such as `ogt:maxBookmarks` that limits the number of bookmarks returned (cf. the explanatory comments in the code).

[20] http://www.w3.org/OWL/

[21] http://www.cs.helsinki.fi/group/seco/ns/2004/03/ontogator#

```
<ogt:FacetSelector
 rdf:about="http://museosuomi.cs.helsinki.fi/internal/ogtQuery#bookmarksByCategory">
  <ogt:incSubCategories>DIRECT</ogt:incSubCategories>
  <!-- ^cut grouping hierarchy to one sublevel^ -->
  <ogt:category>%07%04</ogt:category> <!-- selected category -->
  <ogt:maxBookmarks ogt:limit="8"/> <!-- return only 8 bookmarks -->
  <ogt:maxSubCategories ogt:limit="8"/>
  <!-- ^and a maximum of 8 subcategories^ -->
  <ogt:incBookmarkProperty
    rdf:resource="http://.../ns/2004/03/18-itemcard#www_pictureurl"/>
  <!-- ^include this bookmark property in the result^ -->
  ...
  other options
  ...
  <ogt:constraint rdf:parseType="Collection">
    <rdf:Description
     rdf:about="http://museosuomi.cs.helsinki.fi/internal/ogtQuery#s%00%0A"/>
    <rdf:Description
     rdf:about="http://museosuomi.cs.helsinki.fi/internal/ogtQuery#s%07%04"/>
  </ogt:constraint>
</ogt:FacetSelector>
<ogt:FacetSelector ogt:category="%00%0A"
 rdf:about="http://museosuomi.cs.helsinki.fi/internal/ogtQuery#s%00%0A"/>
<ogt:FacetSelector ogt:category="%0704"
 rdf:about="http://museosuomi.cs.helsinki.fi/internal/ogtQuery#s%07%04"/>
```

An example of a result to a query in RDF/XML format is presented below. It is a hierarchic facet view tree with categories that are associated with bookmarks and additional useful information, such as the number of hits in a category. The tree contains the search result information that is needed for constructing the search result page in HTML on the user interface.

```
<ogt:Facet>
  <ogt:facetSelector
   rdf:resource="http://museosuomi.cs.helsinki.fi/internal/ogtQuery#bookmarksByCategory"/>
  <!-- ^the facet selector responsible for generating this result^ -->
  <ogt:bookmarkHits>5</ogt:bookmarkHits> <!-- a total of 5 bookmarks returned -->
  <!-- results are returned grouped in a tree hierarchy, in this case
       under the facet "Situation of use", under the category "Acts affecting the item"
  -->
  <ogt:subCategories rdf:parseType="Collection">
    <ogt:Category ogt:catid="%07%04" rdfs:label="Acts affecting the item">
      <ogt:bookmarkHits>5</ogt:bookmarkHits>
      <ogt:subCategoryOf ogt:catid="%07" rdfs:label="Situation of use"/>
      <ogt:hasRoot>
        <ogt:Category ogt:catid="%07" rdfs:label="Situation of use"/>
      </ogt:hasRoot>
      <ogt:subCategories rdf:parseType="Collection">
        <ogt:Category ogt:catid="%07%04%07" rdfs:label="Modification">
          <ogt:bookmarkHits>1</ogt:bookmarkHits>
          <ogt:subCategoryHits>1</ogt:subCategoryHits>
          <ogt:topicOf rdf:parseType="Collection">
            <!-- the only bookmark associated with this category -->
            <bm:Bookmark rdfs:label="Ceramic receptible"
              rdf:about="...ns/2004/03/18-esinekortti#LahtiLKM_LHM_LHM_ES_2000057_1">
              <fms:www_pictureurl>
                .../Museo/Lahti/kuvat/LKM_LHM_LHM_ES_2000057_1.jpg
              </fms:www_pictureurl>
            </bm:Bookmark>
          </ogt:topicOf>
        </ogt:Category>
        <ogt:Category ogt:catid="%07%04%08" rdfs:label="Cleaning">
          ...
        </ogt:Category>
      </ogt:subCategories>
      <ogt:subCategoryHits>2</ogt:subCategoryHits>
      <ogt:topicOf rdf:parseType="Collection"/>
      <ogt:directBookmarkHits>0</ogt:directBookmarkHits>
      <!-- there were no direct bookmarks in "Acts affecting the item" -->
    </ogt:Category>
  </ogt:subCategories>
  <ogt:subCategoryHits>1</ogt:subCategoryHits>
  <ogt:topicOf rdf:parseType="Collection"/>
  <ogt:remainder>0</ogt:remainder>
  <!-- there were no bookmarks that could not be grouped under "Acts affecting the item" -->
</ogt:Facet>
```

6.3 OntoViews-*C*

The user interface, interaction and control component of OntoViews, OntoViews-C is built on top of the Apache Cocoon framework [22] . Cocoon is a framework based wholly on XML and the concept of pipelines constructed from different types of components, as illustrated in figure 13. A pipeline always begins with a generator, that generates an XML-document. Then follow zero or more transformers that take an XML-document as input and output a document of their own. The pipeline always ends in a serializer that serializes its input into the final result, such as an HTML-page, a PDF-file, or an image. It is also possible for the output of partial pipelines to be combined via aggregation into a single XML-document for further processing. Execution of these pipelines can be tied to different criteria, e.g, to a combination of the request URI and requesting user-agent.

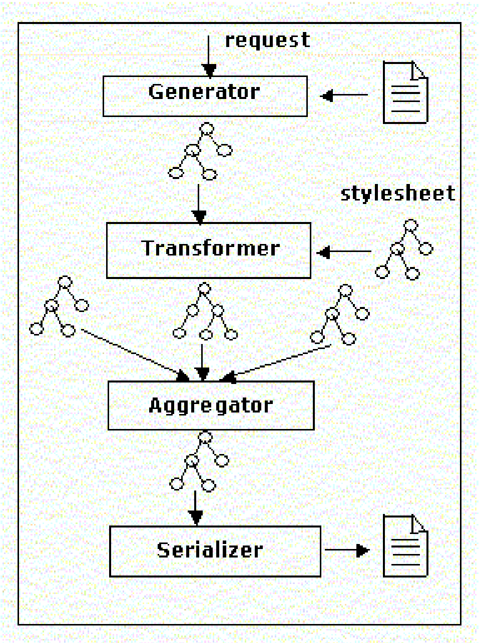


Fig. 13. The components of a Cocoon pipeline.

In OntoViews-C, all of the intermediate components produce not only XML, but

[22] http://cocoon.apache.org/

valid RDF/XML. Figure 14 depicts two pipelines of the ONTOVIEWS-C system. The pipe lines look alike, but result in quite different pages, namely in the search result page seen in figure 5 (and another similar page used for depicting results of the keyword search), and in the item page seen in figure 8. This is due to the modular nature of the pipelines, which makes it possible to split a problem into small units and reuse components.

Every pipeline that is tied to user interaction web requests begins with a user state generator that generates an RDF/XML representation of the user's current state. While browsing, the state is encoded wholly in the request URL, which allows for easy bookmarking and also furthers the possibilities of using multiple servers. This user state is then combined with system state information in the form of facet identifiers and query hit counts, and possible user geolocation based information. This information is then transformed into appropriate queries for the Ontogator and Ontodella servers depending on the pipeline.

In the Search Page pipeline on the left, an Ontogator query returning grouped hits and categories is created. In the Item Page pipeline on the right, Ontogator is queried for the properties and annotations of a specific item and its place in the result set, while Ontodella is queried for the semantic links relating to that item. The Ontogator search engine is encapsulated in a Cocoon transformer, while the Ontodella transformer is actually a generic Web Services transformer that creates a HTTP-query from its input, executes it, and creates SAX events from the HTTP-response. The RDF/XML responses from the search engines are then given to user interface transformers depending on the pipeline and the device that originated the request. These transform the results into appropriate XHTML or to any other format, which is then run through an internationalization transformer for language support and serialized. Most of the transformations into queries and XHTML are implemented with simple XSLT-stylesheets. In this way, changes to layout are very simple to implement, as is the creation of new interfaces for different media. The mobile interface to MUSEUMFINLAND discussed earlier was created in this way quite quickly.

All of the transformer components can also be made available for use in other web applications as Web Services, by creating a pipeline that generates XML from an HTTP-query and returns its output as XML. In this way, other web applications could make use of the actual RDF-data contained in the system, querying the Ontogator and Ontodella servers directly for content data in RDF/XML-format. It also provides a way of distributing the processing in the system to multiple servers. For example, ONTOVIEWS-C instances running Ontogator could be installed on multiple servers, and a main ONTOVIEWS-C handling user interaction could distribute queries among these servers in a round-robin fashion to balance load.

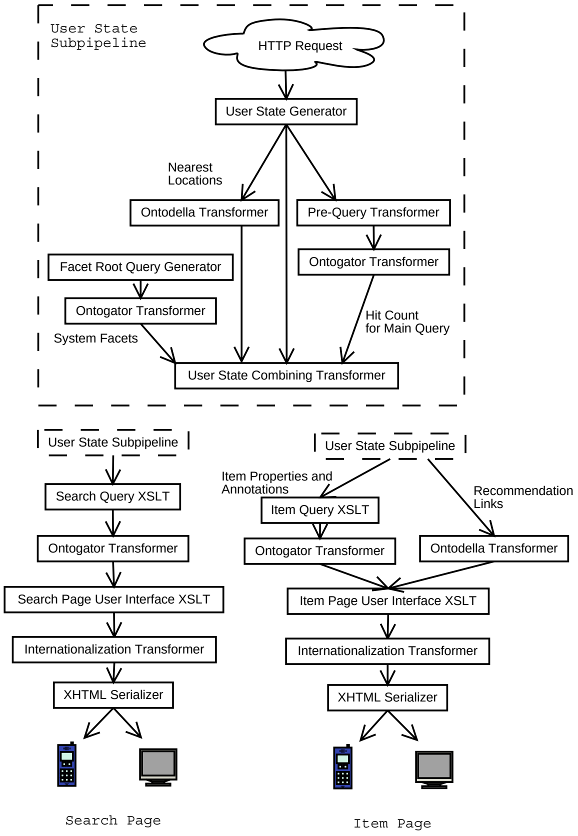


Fig. 14. Two Cocoon pipelines used in ONTOVIEWS-C.

7 Discussion

7.1 *Contributions*

MUSEUMFINLAND demonstrates the power semantic web technologies to solving interoperability problems of heterogeneous museum collections when publishing them on the web. The power of the application comes from the use of ontologies and logic:

Exact definitions By using ontologies, the museums can define the concepts used in cataloging in a precise, machine understandable way.

Terminological interoperability The terms used in different institutions can be made mutually interoperable by mapping them onto common shared ontologies. The ontologies are not used as a norm for telling the museums what terms to use, but rather to make it possible to tolerate terminological variance as far as the terminology mapping from the local term conventions to the global ontology is provided.

Ontology sharing Ontologies provide means for making exact references to the external world. For example, in MUSEUMFINLAND, the location ontology (villages, cities, countries, etc.) and the actor ontology (persons, companies, etc.) is shared by the museums in order to make the right and interoperable references. For example, two persons who happen to have the same name should be disambiguated by different URIs, and a person whose name can be written in many ways, should be identified by a single URI to which the alternative terms refer.

Automatic content enrichment Ontological class and individual definitions, cultural and common sense rules, view projection rules, semantic linking rules, and consolidated metadata enrich collection data semantically.

Intelligent services Ontologies can be used as a basis for intelligent services to the end-user. In MUSEUMFINLAND, the view-based multi-facet search engine is based on the underlying ontological structures and the semantic link recommendation systems reveals to the end-user the underlying semantical context of the collection items and their mutual relations.

A semi-automatic content creation process [19, 18] was developed for the museums for transforming their databases into RDF conforming to the shared ontologies. A problem encountered here was that the original museum collection metadata was not systematically annotated, which resulted in manual work when populating the term ontology. The homonymy problem encountered when mapping literal data values to ontology resources was another major problem, but resulted in less manual work than terminology creation. The semi-automatic annotation tools Terminator and Annomobile proved out to be decent programs for the purposes of the project. The annotation process could be fully automated if the collection cataloging systems were enhanced with datafields for storing URIs in addition to literal descriptions.

A technical innovation of MUSEUMFINLAND is to combine benefits of the multi-facet view-based search paradigm [28, 9] with semantic web ontology techniques and reasoning. Logic rules were used for separating the semantic search and link generation services from the underlying domain specific ontologies and (meta)data. In this way, we could separate the generic parts of the system into the tool ONTO-VIEWS [25] that has been applied to other application domains as well. The prize of the adaptability is that somebody has to create the view and link rules in Prolog, which can be a difficult task if the input data is not directly suitable for generating the needed projections and links.

When using ONTODELLA, the rules for creating category trees and projections were fairly easy to formulate and verify. The idea of semantic link rules appeared to be a good concept if you know exactly what kind of link rules you want and the data enables the reasoning of those links. We set out to create “intriguing” semantic links for the end-user. However, subjectivity of intrigueness made it difficult 1) to choose what semantic link rules to create, 2) to evaluate the “intrigueness” of the rule, and 3) to order the resulting links based on their relevance.

The use of the Cocoon-based implementation of the ONTOVIEWS appeared to be a good solution compared to our previous test implementations [20, 14, 17], since it is eminently portable, extendable, modifiable, and modular. This flexibility is a direct result of designing the application around the Cocoon concepts of transformers and pipelines, in contrast to servlets and layout XSLT. We have used ONTOVIEWS in the creation of a semantic yellow page portal [22], and (using a later version of the tool) a test portal based on the material of the Open Directory Project (ODP) [23] . These demonstrations are based on ontologies and content different from MUSEUMFINLAND. With the ODP material, the ONTOGATOR and ONTOVIEWS-C subparts of the system were tested to scale up to 2.3 million data items and 275,000 view categories with search times of less then 5 seconds on an ordinary PC server.

The use of XSLT in most of the user interface and query transformations makes it easy to modify the interface appearance and to add new functionality. However, it has also led to some quite complicated XSLT templates in the more involved areas of user interaction logic, e.g., when (sub-)paging and navigating in the search result pages. In using XSLT with RDF/XML there is also the problem that the same RDF triple can be represented in XML in different ways but an XSLT template can be only tied to a specific representation. In our current system, this problem can be avoided because the RDF/XML serialization formats used by each of the subcomponents of the system are known, but in a general web service environment, this could cause complications. However, the core search engine components of ONTOVIEWS would be unaffected even in this case because they handle their input with true RDF semantics.

7.2 Related work

Lots of research has been done in annotating web pages or documents using manual or semiautomatic techniques and natural language processing, for example CREAM and Ont-O-Mat by[7] and the SHOE Knowledge Annotator [10]. Stojanovic et al. [33] present an approach that resembles ours in trying to create a mapping between a database and an ontology, but they haven't tackled the questions of integrating many databases or using global and local terminology to make the mapping inside a domain. Also [8] addresses the problems of mapping databases to ontologies, but their way of doing the mapping is very different from ours, trying to get the data dynamically out of the database and involving the database owner. In [31] a related concepts-terms-data model has been used to define different elements used for creating an ontology out of a thesaurus.

The idea of linking collection items with semantic associations was inspired by Topic Maps [26]. However, in our case the links are not given by a map but are

[23] http://www.dmoz.org/

determined by logical inference using the underlying RDF(S) ontology and RDF metadata. Another application of this idea to generating semantically linked static HTMl sites from RDF(S) repositories is presented in [12]. Logic and dynamic link creation on the semantic web has been discussed, e.g. in the work on Open Hypermedia [6, 3], and in the Promooototori system [17]. In the HyperMuseum [35], collection items are also semantically linked with each other. Here linking is based on shared words in the metadata and their linguistic relations, such as synonymy and antonymy. In contrast, our system is not based on words but on ontological references in the underlying RDF(S) knowledge base and the links can be defined freely in terms of logical ruels. The idea of annotating cultural artifacts with ontologies has been explored, e.g., in [11]. Other ontology-related approaches used for indexing cultural content include Iconclass [24] [37] and the Art and Architecture Thesaurus [25] [27].

Much of the web user interface and user interaction logic of MUSEUMFINLAND is based on Flamenco's multi-facet search [9]. In ONTOVIEWS, however, several extensions to this baseline have been added, such as the tree view of categories (figure 6), the seamless integration of concept-based keyword and geolocation search, extended navigation in the result set, and semantic browsing. The easy addition of these capabilities was made possible by basing the system on RDF.

7.3 Towards a more versatile cultural semantic portal

We are investigating how new kinds of cultural RDF material, conforming to different ontologies, can be imported into MUSEUMFINLAND. In the next version of the system called "CultureSampo", more versatile annotation schemas will be used based on events and processes that take place in the society. CultureSampo will contain, e.g., photographs, paintings, folk lore, videos, external web pages, and documents in addition to the artifacts and historical sites present in the current version of MUSEUMFINLAND.

References

[1] D. Brickley and R. V. Guha. *Resource Description Framework (RDF) Schema Specification 1.0, W3C Candidate Recommendation 2000-03-27*, February 2000. http://www.w3.org/TR/2000/CR-rdf-schema-20000327/.

[2] Stefan Decker, Michael Erdmann, Dieter Fensel, and Rudi Studer. Ontobroker: Ontology based access to distributed and semi-

[24] http://www.inconclass.nl/

[25] http://www.getty.edu/research/conducting_research/vocabularies/aat/

structured unformation. In *DS-8*, pages 351–369, 1999. citeseer.nj.nec.com/article/decker98ontobroker.html.

[3] P. Dolong, N. Henze, and W. Neijdl. Logic-based open hypermedia for the semantic web. In *Proceedings of the Int. Workshop on Hypermedia and the Semantic Web, Hypertext 2003 Conference, Nottinghan, UK*, 2003.

[4] J. English, M. Hearst, R. Sinha, K. Swearingen, and K.-P. Lee. Flexible search and navigation using faceted metadata. Technical report, University of Berkeley, School of Information Management and Systems, 2003. Submitted for publication.

[5] D. J. Foskett. Thesaurus. In *Encyclopaedia of Library and Information Science, Volume 30*, pages 416–462. Marcel Dekker, New York, 1980.

[6] C. Goble, S. Bechhofer, L. Carr, D. De Roure, and W. Hall. Conceptual open hypermedia = the semantic web? In *Proceedings of the WWW2001, Semantic Web Workshop, Hongkong*, 2001.

[7] S. Handschuh, S. Staab, and F. Ciravegna. S-cream - semi-automatic creation of metadata. In *Proceedings of EKAW 2002, LNCS*, pages 358–372, 2002.

[8] S. Handschuh, S. Staab, and R. Volz. On deep annotation. In *Proceedings of International World Wide Web Conference*, pages 431–438, 2003.

[9] M. Hearst, A. Elliott, J. English, R. Sinha, K. Swearingen, and K.-P. Lee. Finding the flow in web site search. *CACM*, 45(9):42–49, 2002.

[10] J. Hefflin, J. Hendler, and S. Luke. Shoe: A knowledge representation language for internet applications. Technical report, Dept. of Computer Science, University of Maryland at College Park, 1999.

[11] L. Hollink, A. Th. Schereiber, J. Wielemaker, and B.J. Wielinga. Semantic annotations of image collections. In *Proceedings KCAP'03, Florida*, 2003.

[12] E. Hyvönen, M. Holi, and K. Viljanen. Designing and creating a web site based on RDF content. In *Proceedings of WWW2004 Workshop on Application Design, Development, and Implementation Issues in the Semantic Web, New York, USA*. CEUR Workshop Proceedings, Vol-105, 2004. http://ceur-ws.org.

[13] E. Hyvönen, M. Junnila, S. Kettula, E. Mäkelä, S. Saarela, M. Salminen, A. Syreeni, A. Valo, and K. Viljanen. Finnish Museums on the Semantic Web. User's perspective on MuseumFinland. In *Proceedings of Museums and the Web 2004 (MW2004), Seleted Papers, Arlington, Virginia, USA*, 2004. http://www.archimuse.com/mw2004/papers/hyvonen/hyvonen.html.

[14] E. Hyvönen, M. Junnila, S. Kettula, S. Saarela, M. Salminen, A. Syreeni, A. Valo, and K. Viljanen. Publishing collections in the Finnish Museums on the Semantic Web portal – first results. In *Proceedings of the XML Finland 2003 conference. Kuopio, Finland*, 2003. http://www.cs.helsinki.fi/u/eahyvone/publications/ xmlfinland2003/FMSOverview.pdf.

[15] E. Hyvönen, S. Kettula, V. Raatikka, S. Saarela, and Kim Viljanen. Semantic interoperability on the web. Case Finnish Museums Online. In Hyvönen and Klemettinen [16], pages 41–53. http://www.hiit.fi/publications/.

[16] E. Hyvönen and M. Klemettinen, editors. *Towards the semantic web and web*

services. Proceedings of the XML Finland 2002 conference. Helsinki, Finland, number 2002-03 in HIIT Publications. Helsinki Institute for Information Technology (HIIT), Helsinki, Finland, 2002. http://www.hiit.fi/publications/.

[17] E. Hyvönen, S. Saarela, and K. Viljanen. Application of ontology-based techniques to view-based semantic search and browsing. In *The semantic web: research and applications. First European Semantic Web Symposium, ESWS 2004, Heraklion, Greece*. Springer–Verlag, Berlin, May 2004. 92–106.

[18] E. Hyvönen, M. Salminen, and M. Junnila. Annotation of heterogeneous database content for the semantic web. In *Proceedings of the 4th International Workshop on Knowledge Markup and Semantic Annotation (SemAnnot 2004), Hiroshima, Japan*, November 2004.

[19] E. Hyvönen, M. Salminen, S. Kettula, and M. Junnila. A content creation process for the Semantic Web. In *Proceeding of OntoLex 2004: Ontologies and Lexical Resources in Distributed Environments, May 29, Lisbon, Portugal*, 2004.

[20] E. Hyvönen, A. Styrman, and S. Saarela. Ontology-based image retrieval. In Hyvönen and Klemettinen [16], pages 15–27. http://www.hiit.fi/publications/.

[21] O. Lassila and R. R. Swick (editors). Resource description framework (RDF): Model and syntax specification. Technical report, W3C, February 1999. W3C Recommendation 1999-02-22, http://www.w3.org/TR/REC-rdf-syntax/.

[22] M. Laukkanen, K. Viljanen, M. Apiola, P. Lindgren, and E. Hyvönen. Towards ontology-based yellow page services. In *Proceedings of WWW2004 Workshop on Application Design, Development, and Implementation Issues in the Semantic Web, New York, USA*. CEUR Workshop Proceedings, Vol-105, 2004. http://ceur-ws.org.

[23] R. L. Leskinen, editor. *Museoalan asiasanasto*. Museovirasto, Helsinki, Finland, 1997.

[24] A. Maedche, S. Staab, N. Stojanovic, R. Struder, and Y. Sure. Semantic portal — the SEAL approach. Technical report, Institute AIFB, University of Karlsruhe, Germany, 2001.

[25] E. Mäkelä, E. Hyvönen, S. Saarela, and K. Viljanen. Ontoviews—a tool for creating semantic web portals. In *Proceedings of the 3rd International Semantic Web Conference (ISWC 2004), Hiroshima, Japan*, pages 797–811. Springer–Verlag, Berlin, November 2004.

[26] Steve Pepper. The TAO of Topic Maps. In *Proceedings of XML Europe 2000, Paris, France*, 2000. http://www.ontopia.net/topicmaps/materials/rdf.html.

[27] T. Peterson. Introduction to the Art and Architechure thesaurus, 1994. http://shiva.pub.getty.edu.

[28] A. S. Pollitt. The key role of classification and indexing in view-based searching. Technical report, University of Huddersfield, UK, 1998. http://www.ifla.org/IV/ifla63/63polst.pdf.

[29] V. Raatikka and E. Hyvönen. Ontology-based semantic metadata validation. In Hyvönen and Klemettinen [16], pages 28–40. http://www.hiit.fi/publications/.

[30] J. Ben Schafer, Joseph A. Konstan, and John Riedl. E-commerce recommen-

dation applications. *Data Mining and Knowledge Discovery*, 5(1/2):115–153, 2001.

[31] D. Soergel, B. Lauser, A. Liang, F. Fisseha, J. Keizer, and S. Katz. Reengineering thesauri for new applications: the agrovoc example. *Journal of Digital Information*, (4), 2004.

[32] J. Sowa. *Knowledge Representation. Logical, Philosophical, and Computational Foundations*. Brooks/Cole, 2000.

[33] L. Stojanovic, N. Stojanovic, and R. Volz. Migrating data-intensive web sites into the semantic web. In *Proceedings of the ACM Symposium on Applied Computing SAC-02, Madrid, 2002*, pages 1100–1107, 2002.

[34] Nenad Stojanovic, Rudi Studer, and Ljiljana Stojanovic. An approach for the ranking of query results in the semantic web. In Dieter Fensel, Katia Sycara, and John Mylopoulos, editors, *Proceeedings of the second international semantic web conference ISWC2003, Sanibel Island, Florida*, number 2870 in LNCS, pages 500–516. Springer–Verlag, Berlin, 2003.

[35] Peter Stuer, Robert Meersman, and Steven De Bruyne. The HyperMuseum theme generator system: Ontology-based internet support for active use of digital museum data for teaching and presentations. In D. Bearman and J. Trant, editors, *Museums and the Web 2001: Selected Papers*. Archieves & Museum Informatics, 2001. http://www.archimuse.com/mw2001/papers/stuer/ stuer.html.

[36] Mark van Assem, Maarten R. Menken, Guus Schreiber, Jan Wielemaker, and Bob Wielinga. A method for converting thesauri to RDF/OWL. In *Proceeedings of the third international semantic web conference ISWC2004, Hiroshima, Japan*. Springer–Verlag, Berlin, October 2004.

[37] J. van den Berg. Subject retrieval in pictorial information systems. In *Proceedings of the 18th international congress of historical sciences, Montreal, Canada*, pages 21–29, 1995. http://www.iconclass.nl/texts/history05.html.