SeLeNe — Self E-Learning Networks
IST-2001-39045

WP4 Deliverable 4.2
Personalisation and Trails in Self e-Learning Networks

Kevin Keenoy, Mark Levene and Don Peterson

Birkbeck College and Institute of Education, University of London

Abstract

This deliverable begins with a discussion of one level at which the process of searching for learning objects in a SeLeNe can be personalised. This involves the specification of the semantics of a SeLeNe User Profile, which takes some elements from existing learner profile schemes and adds extra custom elements where existing schemes fail to provide the expressiveness necessary for effective personalisation. We suggest structured vocabularies for the specification of learning goals and learning styles, which together enable learners to express their educational needs within their profile.

We then go on to consider how the SeLeNe User Profile could be applied to inform the search process in two possible search scenarios, and finally we discuss SeLeNe's support for *trails* of learning objects, and how in combination with the user profile these can support learners and other users of the system.

London, UK
January 31, 2004

The SeLeNe Project

Life-long learning and the knowledge economy have brought about the need to support a broad and diverse community of learners throughout their lifetimes. These learners are geographically distributed and highly heterogeneous in their educational backgrounds and learning needs. The number of learning resources available on the Web is continuously increasing, thus indicating the Web's enormous potential as a significant resource of educational material both for learners and instructors.

The SeLeNe Project aims to elaborate new educational metaphors and tools in order to facilitate the formation of learning communities that require world-wide discovery and assimilation of knowledge. To realise this vision, SeLeNe is relying on semantic metadata describing educational material. SeLeNe offers advanced services for the discovery, sharing, and collaborative creation of learning resources, facilitating a syndicated and personalised access to such resources. These resources may be seen as the modern equivalent of textbooks, comprising rich composition structures, 'how to read' prerequisite paths, subject indices, and detailed learning objectives.

The SeLeNe Project (IST-2001-39045) is a one-year Accompanying Measure funded by EU FP5, running from 1st November 2002 to 31st October 2003, extended until 31st January 2004. The project falls into action line V.1.9 CPA9 of the IST 2002 Work Programme, and is contributing to the objectives of Information and Knowledge Grids by allowing access to widespread information and knowledge, with e-Learning as the test-bed application. The project is conducting a feasibility study of using Semantic Web technology for syndicating knowledge-intensive resources (such as learning objects) and for creating personalised views over such a Knowledge Grid.

Executive summary

This deliverable (4.2) is part of the SeLeNe Workpackage 4 on Syndication and Personalisation of Educational Resources. Workpackage 4 has two main objectives:

- To investigate techniques for syndication and personalisation of distributed, autonomous RDF description bases.
- To design language primitives for defining user views over distributed RDF description bases and for deriving composite learning objects' descriptions from those of their constituent learning objects.

Accessing RDF description bases in SeLeNe raises two basic technical challenges: (1) flexible mediation of the different RDF schemas employed by the RDF description bases, and (2) personalisation of learning objects' descriptions and schemas according to the educational needs and interests of learning objects' providers (i.e. instructors) and consumers (i.e. learners).

Concerning the first problem, the IEEE LOM has effectively achieved the integration of the various educational metadata standards, as is reported in Deliverable 2.1 of Workpackage 2. Thus, in the context of a SeLeNe, we are assuming that metadata about learning

objects are represented using an RDF/S binding of the IEEE LOM. However, fine-grained descriptions expressed in domain or topic-specific taxonomies may also be made available by instructors. Hence, this workpackage is investigating a flexible articulation of different domain/topic-specific taxonomies which can be used for e-learning, as well as the automatic generation of semantic descriptions for composite learning objects using the descriptions of their constituent learning objects. The taxonomies used for this purpose and the resulting descriptions can easily be represented using RDF/S. This work is reported in Deliverable 4.1.

Concerning the second problem, two major issues are involved: (a) specification by learners of their educational needs, and (b) adaptation of learning objects to these needs. The first issue requires the representation of educational needs in a "learner profile" using the e-learning schemas as well as the domain or topic-specific taxonomies available in SeLeNe. It also requires unstructured, keyword-based querying facilities, which can be translated automatically into the structured RDF/S queries supported by the SeLeNe system. The result of these unstructured queries may be returned in a special form of composite learning object called "trails". To address the second issue we need methods for dynamically adapting learning material to the preferences of a learner. This requires ranking of query results by matching the descriptions of the returned learning objects against the a learner's profile. These issues are discussed in this Deliverable, 4.2.

Specifying educational needs or describing educational material according to personalised e-learning RDF/S schemas (for both learners and instructors) requires formalisms for defining declarative views over learning object descriptions and schemas, and this work is reported in Deliverable 4.3. This deliverable also discusses the structured RDF/S querying facilities supported by SeLeNe.

Also needed are techniques for detecting changes in learning objects' descriptions or users' personal profiles, and for notifying users who have subscribed to be notified of such changes. Techniques for the provision of this kind of reactive functionality over RDF descriptions of learning objects and users are reported in Deliverable 4.4.

The main focus of this Deliverable (4.2) is to discuss the semantics of the SeLeNe user profile, its representation using RDF/S, and how the profile can be employed to support personalisation in SeLeNe. Existing learner profile specifications are considered and some of their weaknesses highlighted — the suggested SeLeNe user profile specification augments these existing specifications to help overcome some of their weaknesses. Structured vocabularies for the specification of learning objectives and learning styles are suggested, enabling learners to express their educational needs.

Following the specification of the user profile, its use within the process of searching for learning objects using a SeLeNe is discussed. This involves the translation of unstructured, keyword-based queries into a structured query language and the filtering and ranking of query results according to the user profile. Finally, support for trails of learning objects is discussed.

Revision Information

Revision Date	Version	Changes
July 22, 2003	0.1	Detailed Outline
July 25, 2003	0.2	First Draft Proposal
September 18, 2003	0.3	Second Draft Proposal (Major rewrite of 0.2 following discussions at meeting in Crete)
September 26, 2003	1.0	First Full Version (Minor revisions of 0.3)
October 3, 2003	1.1	Additional appendices containing RDF Schema
October 9, 2003	1.2	Minor updates to 1.1
November 5, 2003	1.3	Minor updates to 1.2
December 2, 2003	1.4	Minor alteration to RDF Schema
January 7, 2004	2.0	Revisions (following discussion at the final project meeting in London)
January 31, 2004	2.1	Final version

Contents

1 Personalisation in SeLeNe **7**

2 User Profiles **8**
2.1 Educational Modelling Languages (EMLs) 9
2.2 IMS-LIP and PAPI 9

3 The SeLeNe User Profile **10**
3.1 User-Supplied Information 11
3.2 Implicitly Gathered Information 14
3.3 Profile Storage and Access 15

4 Specification of User's Knowledge and Experience **16**

5 Specification of Learning Objectives **19**

6 Specification of Learning Styles **23**

7 Support for SeLeNe's Active Functionality **24**

8 Search and Retrieval **26**
8.1 Translation of Unstructured Queries into RQL 27
8.2 Indexed Search 29
8.3 Dynamics of the Search Process 30
8.4 Filtering and Ranking Search Results 30

9 Trails **33**
9.1 Advanced Learning Material Modelling 33
9.2 Trails in SeLeNe 33
9.3 Deriving Trails using Semantic Relationships 35
9.4 Support for Browsing as a Trail 36

10 Open Challenges **36**

References **39**

Appendices **41**

A Example RDF Learner Profile **41**

B RDF Schema for PAPI Elements used in SeLeNe **44**

C RDF Schema for IMS-LIP Elements used in SeLeNe **46**

D RDF Schema for IMS-RCD Elements used in SeLeNe **50**

E RDF Schema for SeLeNe-specific Elements of the User Profile **51**

F RDF Schema for Trails **53**

1 Personalisation in SeLeNe

One user requirement for SeLeNe is that personalisation should be one of the services provided to users. It is a service that can be switched on and off at will, but when active the personalisation should be as transparent to the user as possible.

One of the objectives of this workpackage is the specification by learners of their educational needs, with a view to then providing them with Learning Objects (LOs) that will address these needs. For this reason the main area of personalisation discussed in this document is as part of the search process — when users search for LOs the results returned to them will depend on who they are as well as their query, since different LOs may be more appropriate for different learners.

Personalisation will have an effect on search results returned from a keyword-based query at three different levels:

- **Filtering** of the returned LOs — excluding those LOs deemed unsuitable for the learner, even though they satisfied the original query
- **Ranking** of the returned LOs — the 'best' LO for one user may be different from the 'best' LO for another, but personalised ranking means that they can both have the most suitable LO for them returned at the top of their search results
- **Presentation** of results — users will have different preferences for the display of their search results (e.g. display results as *trails* or as a simple list, display 10 results per page or 50 results per page)

Filtering and ranking are discussed further in Section 8.4. A full discussion of presentational issues is beyond the scope of this document, although there are some comments regarding requirements of the SeLeNe user interface throughout.

Some aspects of personalisation can also take place even before a query is submitted for evaluation: personalised queries can be constructed using information stored in the profile, by re-formulating or annotating the user's original query to reflect elements of their profile. For example, if the profile contains the information that the user only speaks English, all keyword searches they enter could be annotated with *lang:en* automatically. This is the first stage of *Filtering*.

An overview of the interactions and data flow necessary to provide personalisation of search results (as described above) is shown in Figure 1. If the system shown is to be implemented using a service-based architecture (as described in *SeLeNe Deliverable 3: A Grid Service Framework for Self e-Learning Networks* [22]), then all interactions between the components shown could be via a core *Communication* service.

Keyword-based query is not the only way that users can locate LOs in a SeLeNe — the schema of the LO descriptions can also be browsed to find relevant LOs, providing facilities such as 'browse by author' and 'browse by subject'. Personalisation of the browsing process can occur at two levels. Firstly, the user's view of the LO information space can be personalised by the definition of an RVL view over the RDF metadata (via a simple user

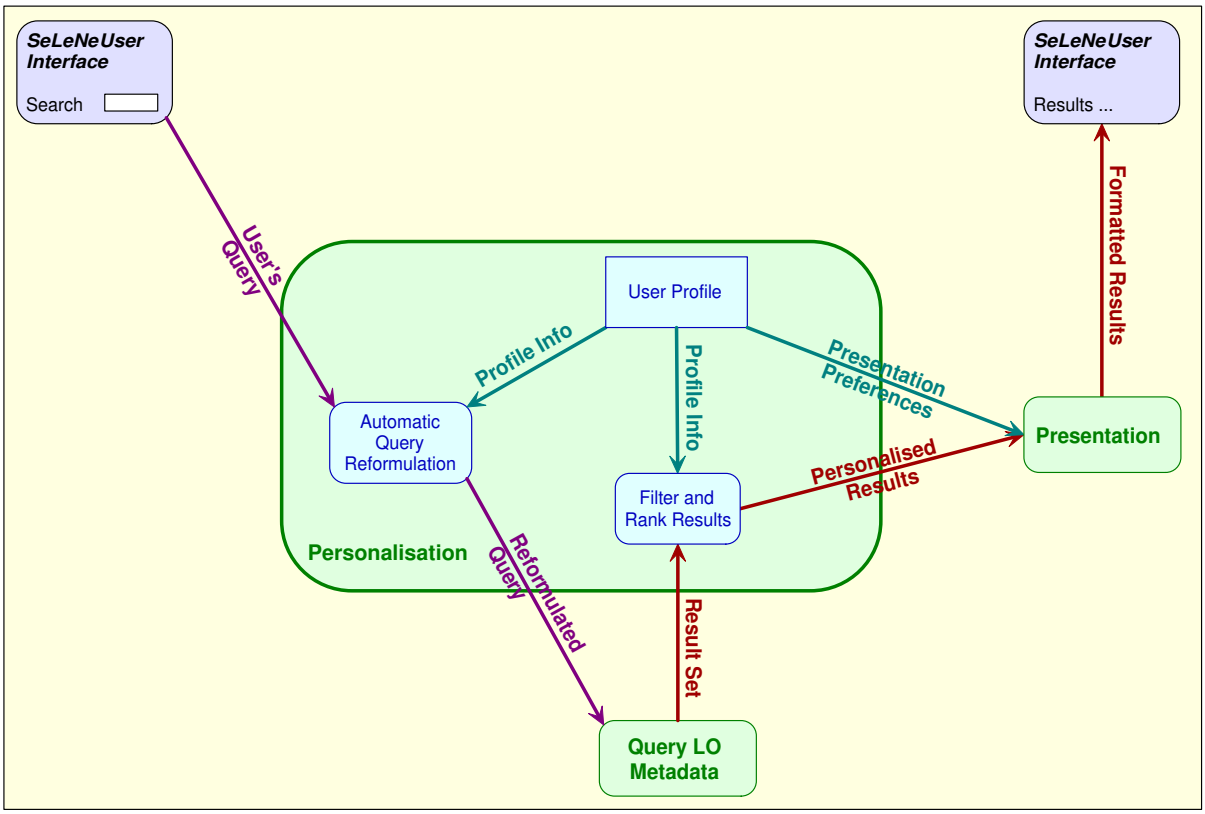

Figure 1: Overview of the provision of personalised search results in SeLeNe

interface), allowing users to restrict the information they see to only those attributes of interest to them, organised in their preferred manner. These personal views can also be queried directly when searching for LOs. Personalisation via RVL views is covered in *SeLeNe Deliverable 4.3: Views and Structured Querying in Self e-Learning Networks* [12]. Secondly, the system can use knowledge of a user's preferences (either those explicitly supplied by the user or those learned by the system itself) to recommend individual LOs or categories of LO to the user as they are browsing. Details of how this could work are provided in Section 9.4 of this document.

2 User Profiles

The provision of personalised access to LOs requires a profile to be stored for each user and for each group of users (where 'group of users' here means a set of individual SeLeNe users who have agreed to collaborate in some activity, for example a class of learners all studying the same topic, and who use a single 'group profile' for the collaborative activity). The profiles for individual learners, for LO providers and for groups of SeLeNe users will differ slightly in their semantic models. The focus here will be on learner profiles, although most of what is said will apply to all SeLeNe profiles.

Numerous schemes for learner profiles have been proposed. Here we briefly consider some of the most sophisticated, and whether they can provide the information necessary for the kind of personalisation envisaged in SeLeNe.

2.1 Educational Modelling Languages (EMLs)

The CEN's Information Society Standardization System (CEN/ISSS) defines an EML as "*a semantic information model and binding, describing the content and process within a 'unit of learning' from a pedagogical perspective in order to support reuse and interoperability.*" [18]. Their survey of EMLs [18] gives an overview of the six main existing languages. All six of these have XML bindings, but some (3 out of the 6) also have SGML bindings. EMLs are supposed to describe both the content of learning units and the interactions and activities of students and teachers. However, the survey found that all of the proposals have so far concentrated on the description of learning material (a 'unit of learning' — e.g. a course), rather than on the description of the actors involved in the learning process.

Only two of the current proposals (OUNL-EML [16] and PALO [21]) proceed to model people as well as course content. In these proposals people are defined in terms of the roles they play (what activities they participate in) for workflow modelling. This kind of user modelling is of limited use in the context of SeLeNe, as the most important information for SeLeNe's personalisation services is the learner's current goals, experience and preferences, which are not covered by workflow modelling.

OUNL-EML can "describe personalisation aspects within units of learning, so that the content and activities within units of learning can be adapted based on the preferences, prior knowledge, educational needs and situational circumstances of users"[18], but SeLeNe is not concerned only with individual units of learning, and OUNL-EML doesn't give any indication of how to specify the user attributes (i.e. preferences, prior knowledge, etc.) themselves. It also includes a "personal dossier" for individual students, which records assessment, grading and "time-spent on assignment" information. Again, this is not the kind of information needed by SeLeNe to provide personalised access to LOs.

2.2 IMS-LIP and PAPI

There are other learner profile proposals that do not fall into the category of EMLs. Two of the most important and well-developed of these are the IEEE LTSC's Personal and Private Information (PAPI) Standard [15] and the IMS Learner Information Package (LIP) [23].

There is considerable overlap in the information that each scheme stores about a learner, much of which is also needed in the SeLeNe user profile. A common data model for learner profiles is desirable [6], so rather than propose yet another schema for demographic and other information that is already adequately catered for by existing profile schemes we propose (similarly to Dolog and Nejdl [6]) to use some elements from both PAPI and IMS-LIP in the SeLeNe profile, extending the data model with our own elements only where existing specifications fail to be expressive enough.

The use of RDF to encode the profile data allows us to pick elements from multiple schemas and remain interoperable with other RDF-enabled systems (for example, if a system uses only PAPI elements then when presented with a SeLeNe profile it can easily read just those elements, ignoring the properties defined in other schemas). With the plethora of specifications available and the lack of even a *de facto* standard, we see the use of RDF to select from multiple existing schemas in order to create customised, application-specific data models as being the dominant trend in user modelling in the future.

The shortcomings of existing specifications are generally in the recording of competencies, learning goals and learning styles. This is acknowledged even by their authors — Brendon Towke (who is a member of the IEEE LTSC PAPI Working Group) said of the existing standard:

"*If I have an LMS that stores IQ as an important characteristic of the learner, which of the six information types is the appropriate place to store it? It seems that none of them are, thus pointing out the need for either an additional information type or an expansion of the meaning of one of the existing types.* [...] *Similar arguments can be made for learning styles, cognitive styles, personality styles, affective styles, learning goals, and career goals. All of these data elements should have a place in the data model, and currently, they don't*" [8].

IMS-LIP improves on PAPI slightly by providing a string field for learning goals. While a string field is fine if the profile is to be read by a human, in the case of machine processing of profiles for automatic personalisation we believe that a more structured data element is necessary. Neither scheme allows space in the profile for the description of user preferences (for example, learning styles and preferences for particular authors or publishers).

Section 3 gives details of the elements of PAPI and IMS-LIP selected for use as part of the SeLeNe profile and discusses general profile management issues. Later sections concentrate on the SeLeNe-specific extensions to existing profiling schemes.

3 The SeLeNe User Profile

We see the profiles used in SeLeNe being composed of two main parts: information explicitly supplied by the user and information collected transparently by observing the user's behaviour. We here give details of the semantics of each of these parts, which include some elements from PAPI and IMS-LIP, along with some SeLeNe-specific elements. Although there is clearly further information on users that could be stored in the profile (for example, the rest of the PAPI and IMS-LIP elements), we believe that the information as described here is sufficient to provide the SeLeNe personalisation functionality.

A simplified overview of the profile schema is shown in Figure 2 (the arrow mimics the RDF *Property*). Appendix A contains an example SeLeNe user profile in RDF-XML; the RDF schemas are in the appendices that follow. Fragments from the example profile of Appendix A are used hereafter to illustrate the profile elements being discussed.

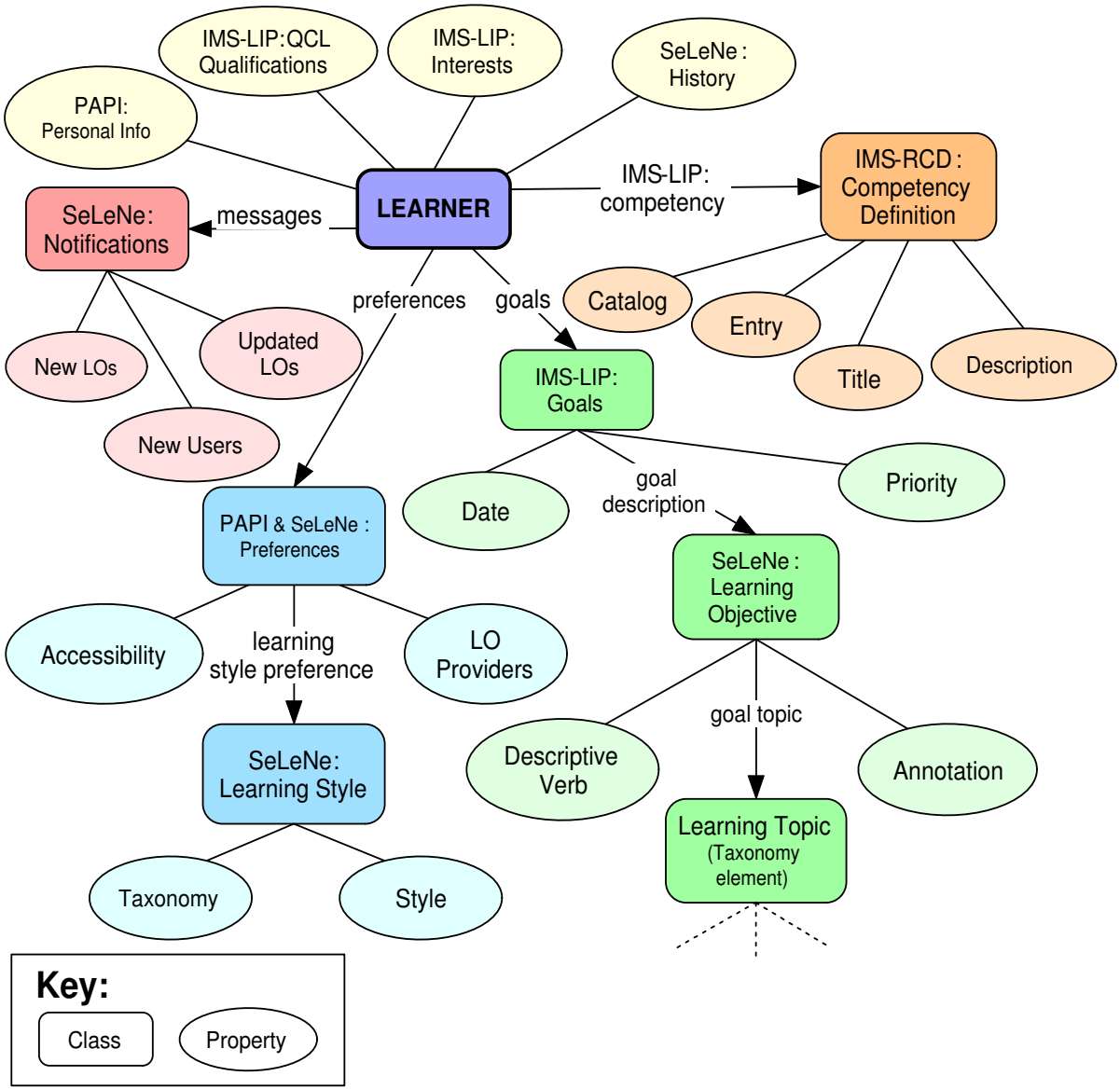

Figure 2: Simplified overview of the SeLeNe user profile schema

3.1 User-Supplied Information

When initially registering with a SeLeNe the user will have the option to supply some information for their profile, and this can be expanded upon or modified at a later date. Of course, the user could choose not to supply any personal information, but the more data the system has the better it will be able to serve the user.

The user-supplied information will include general demographic data and other personal data that can help in identifying useful LOs, and this will be stored using the following elements from PAPI and IMS-LIP:

- **PAPI** elements
  - Learner Personal Information
    - My Personal Identifier List (*SeLeNe and any other usernames/ids*)
    - Name List (*the name of the learner*)
    - Postal Address List
    - Personal Bucket (*a "bucket" for adding name-value pairs; Age, Languages, etc.*)
  - Learner Preference Information
    - Post-Include-Preference-Hid-List (*a collection of preferences to be taken into consideration — e.g. Accessibility requirements*)
- **IMS-LIP** elements
  - QCL (*received Qualifications, Certifications and Licences*)
    - Organisation
    - Level
    - Title
    - Date
    - Description
  - Interest
    - Typename (*the type of interest* — [*Recreational* | *Vocational* | *Domestic*])
    - Description
  - Competency
    - Exrefrecord (*description of competency using an appropriate externally defined structure — see* Section 4 *for details of the structure chosen for use in SeLeNe*)
  - Goal
    - Typename (*the type of goal* — [*Work* | *Education* | *Personal*])
    - Description (*description of the goal — see* Section 5 *for the specification of a structured vocabulary for goal description*)
    - Date
    - Priority

It should be noted that while we believe the elements listed here are a sufficient subset of PAPI and IMS-LIP for use in SeLeNe, the use of RDF means that elements can easily be added or removed without affecting the comprehensibility of the profile — for example, an IMS-LIP 'accessibility' data element could be used instead of (or as well as) the PAPI Post-Include-Preference-Hid-List element, or the 'activity' learner information element could be added if necessary for a specific application or personalisation algorithm.

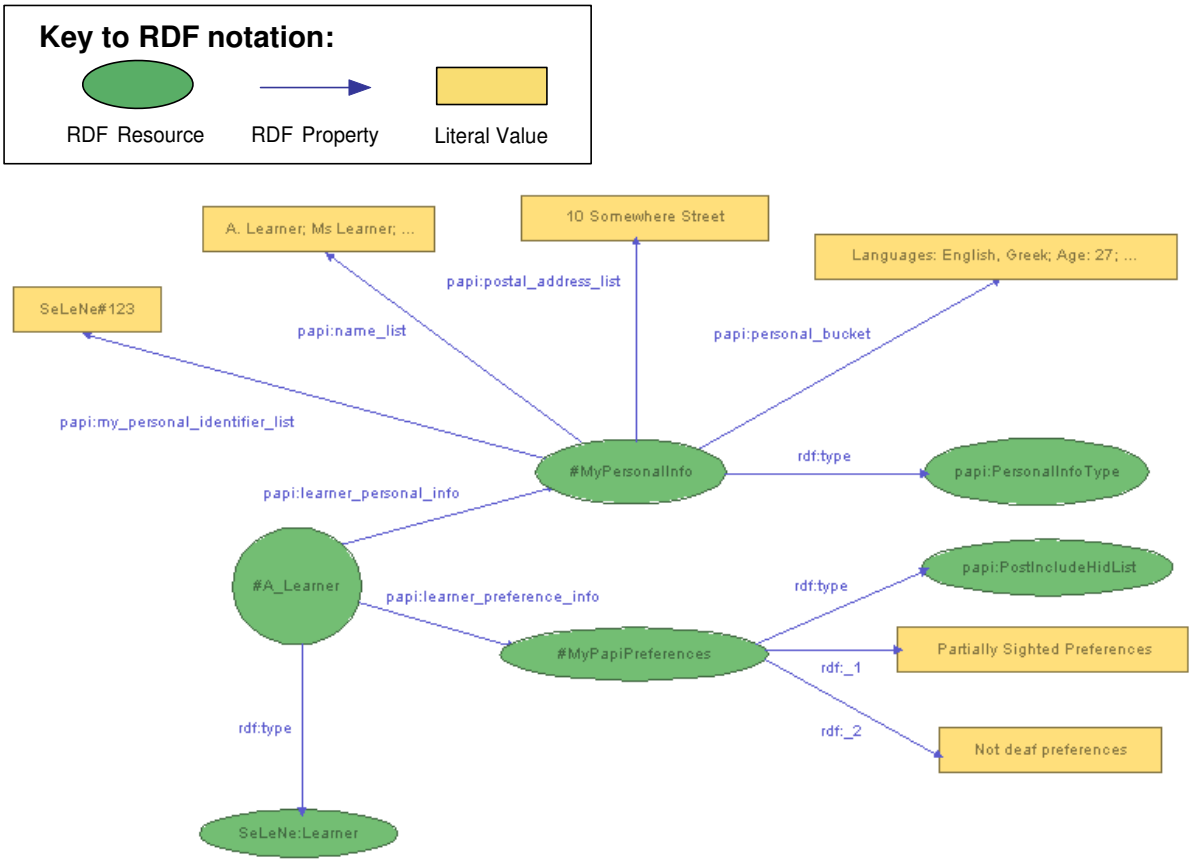

Figure 3: RDF fragment showing the use of PAPI elements in an example user profile

The user registration process should provide a series of forms to allow the entry of this data, with either text fields or a list of optional choices where appropriate.

The final two IMS-LIP elements listed here, i.e. Competency and Goal, are probably the most important information needed by the system in order to provide effective personalisation. Unfortunately they are also the most difficult to specify in a clear, concise and unambiguous way. Section 4 discusses the specification of prior knowledge and competencies. Section 5 discusses the structured specification of learning goals and objectives.

An example RDF fragment for the PAPI data elements used is shown in Figure 3, and for the IMS-LIP elements in Figure 4. These (and all subsequent examples) are fragments of the example RDF learner profile that can be found in Appendix A. Appendix B contains the schema for the PAPI elements, and Appendix C that for the IMS-LIP elements used.

Of the elements chosen, only *ims-lip:goal_typename* and *ims-lip:interest_typename* have a restricted set of possible values prescribed in their specification (as can be seen in the list of **IMS-LIP** elements given previously). In future it may be desirable to replace string-valued elements such as *ims-lip:qcl_organisation* and *ims-lip:qcl_level* by RDF properties referencing RDF resources in taxonomies of educational institutions (and other awarding

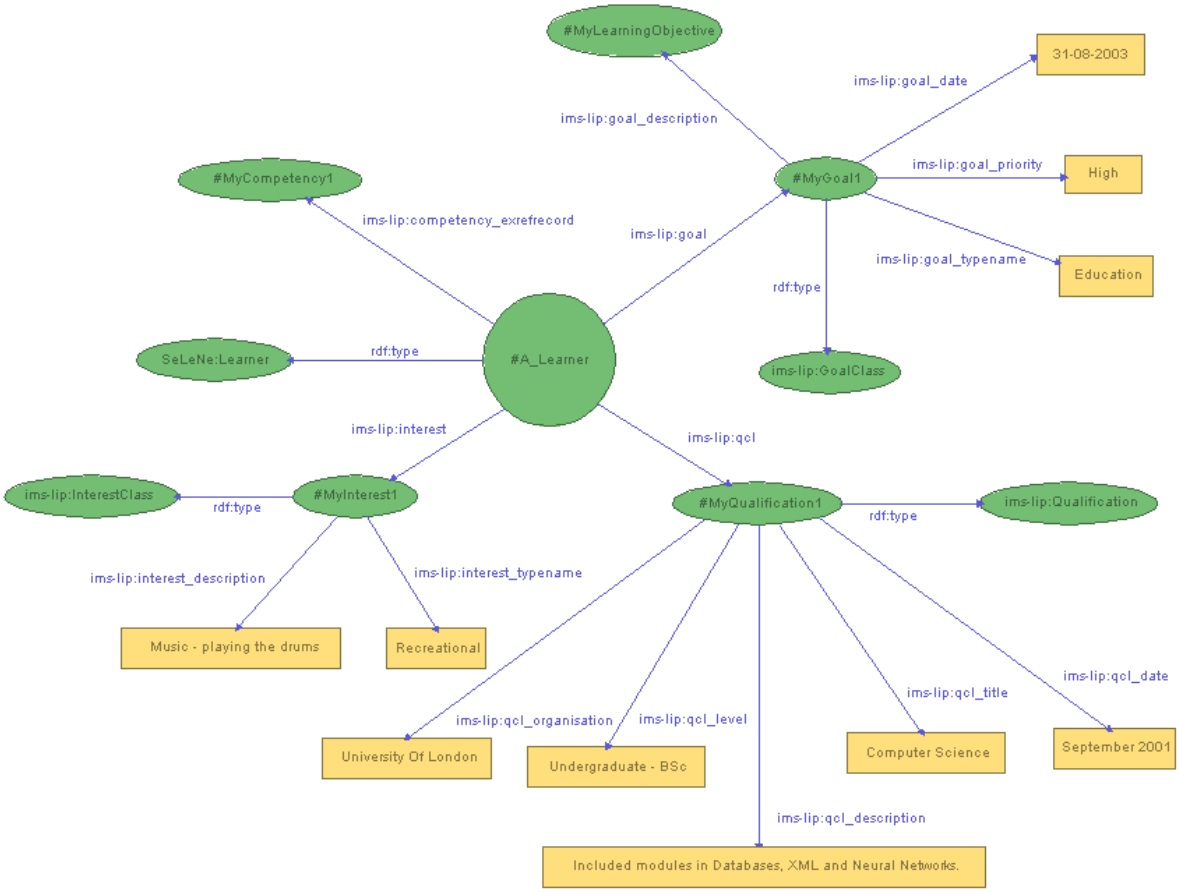

Figure 4: RDF fragment showing the use of IMS-LIP elements in an example user profile

bodies) and attainment levels. The utility that would be provided by this is discussed in detail in Section 4.

In the case of a profile for a group of users, rather than for an individual, the initial information will be supplied by the creator of the group. The main difference between learner and group profiles is that group profiles contain a list of users who are members of the group instead of the PAPI Learner and Preference information. The interests, competencies and goals specified by the group's creator for inclusion in the group profile are those relevant to the whole group. One of the members is recorded as being the "owner" of the group (usually, but not necessarily, the creator of the group), and has editing rights on the group profile. Members of the group can remove themselves from the group, but not edit other group properties.

3.2 Implicitly Gathered Information

The continuous collection of information implicit in user interactions with the system provides the *adaptive* part of the profile. As the user interacts with the SeLeNe system

aspects of their behaviour can give clues as to their interests and preferences. If the system records certain user behaviour these implicit indicators of interest can be used to aid the personalisation functionality. A *History* of searches and LO access is recorded, including dates and details of the LOs browsed. The history information gathered can be stored in the profile as a set of *Trails*, capturing the history and other information in a standard format (see Section 9.2 for details). This dynamic, growing history of LO interaction contains a wealth of information that can be mined, with the right techniques, to reveal information useful for personalisation, for example:

- preferred authors and publishers;
- the user's learning styles;
- areas of interest within topics.

Thus the history part of the profile enables the provision of adaptive personalisation functionality that takes account of the user's recent behaviour. The initially user-supplied information mentioned above can be automatically adjusted to better and more fully reflect the user (for example, if the user accesses many LOs in German then 'German' can be added to the list of languages in their "PAPI Personal Bucket", even if they did not initially specify that they speak German). Automatic updates of the user profile based on their evolving history could be implemented by specifying a standard set of Event-Condition-Action rules (ECA rules) that are automatically registered with each SeLeNe profile (ECA rules are discussed in *SeLeNe Deliverable 4.4: ECA Rule Languages for Active Self e-Learning Networks* [17]). This standard set would include, for example, rules that codify behaviour such as "If the history contains X LOs by Y then add Y as a preferred author" and "If the history contains $X\%$ of LOs catering to learning style Y then add Y to the record of their learning styles".

A further use of the user history becomes available when profiles are shared with other users (either network-wide or within a defined group). In this case collaborative filtering techniques [14] can be employed to provide personalised suggestions to users based on the behaviour of other, similar users.

3.3 Profile Storage and Access

Most data in a SeLeNe is foreseen to be distributed across the entire network. However, as mentioned in *SeLeNe Deliverable 3: A Grid Service Framework for Self e-Learning Networks* [22], a user's profile should **not** be distributed. There are two main reasons for this:

- The entire profile is needed to do effective personalisation. If parts of the profile are inaccessible because the nodes where those parts are stored are currently unreachable then the quality of personalisation will be reduced. The quality of personalisation should not depend on the state of the network.

- Distributing the profile across several machines increases security and privacy risks.

Instead, the profile should be stored locally — either on the user's own machine, or at the local SeLeNe node only. The latter option is probably better as it means that from whatever machine the user logs in, the profile will always be available. Performance issues may arise if there are many users registered at a single node and they each have a large profile — storage, retrieval and transfer of them could cause problems. This can be simply avoided by placing a limit on the size of the profiles stored at nodes with limited capabilities, with the oldest or least used information in the history being deleted when the size is over the limit.

User profiles must be security protected to preserve the privacy of SeLeNe users. It may be that users allow the system access to the history part of their profile to enable collaborative filtering and the discovery of group emergent trails (see Section 9.4 for details) — this is the 'price' they pay for the advantages of personalised access to LOs. As well as sharing their profile information with the system, a user can choose to share their profile with other users too. Other users should only be allowed access to a profile if they have explicit permission from the person whose profile it is.

4 Specification of User's Knowledge and Experience

Existing specifications for learner profiles generally include space for a record of the learner's experience and current knowledge. The most sophisticated of these schemes appears to be that proposed in the IEEE PAPI standard. The PAPI profile contains a section of 'Performance Information' specifically "to provide improved or optimized learning experiences" [15]. From this description this would appear to be ideal for use in the SeLeNe profile but, as we shall see, unfortunately it turns out that in its current state PAPI provides inadequate data to provide effective personalisation.

The two parts of the PAPI Performance Information most relevant to SeLeNe are:

1. **Learner Performance Information**, which is a list of identifiers for "learning experiences" that the learner has taken part in, and their associated grades and awarding body. All granularity levels are covered, so it could include the percentage score on a recently taken test as well as a degree classification, for example.

2. **Learner Portfolio Information**, which includes

   - a list of pieces of work produced by the learner that demonstrate various accomplishments and work
   - a *media_competency_definition_list*, which is a list of references to LTSC Competency Definitions that are related to the learner's accomplishments or works[1]

[1]Unfortunately, these Competency Definitions do not yet exist. The LTSC WG investigating their definition has found that there are many proprietary self-declared standards, where the only common elements are a proprietary ID and some verbose textual description. No proposals towards a genuine

While obviously useful for a human reader of the profile and for administrative purposes, knowledge of the name of a course and the grade achieved is not really useful in automatically deciding if a LO under consideration for presentation to the user covers material that the user already knows. Nor does it tell us in enough detail what prerequisite knowledge the user has. A learner's educational history is obviously an important part of any learner profile, but we have chosen to use the IMS-LIP's "QCL" data element to record this, as it has a simpler structure and also contains a 'description' element, which PAPI's Performance Information does not. The Portfolio Information might be of more use to SeLeNe if the LTSC definition of competencies that it references was complete, but unfortunately it isn't.

The IMS Reusable Competency Definitions (RCD)[2] [9] is in a more complete state of development, and for each competency includes the elements

- **Identifier** (mandatory) — globally unique
- **Title** (mandatory) — short, human-readable name for the competency
- **Description** (optional) — human-readable description of the competency
- **Definition** (optional) — structured description of the competency

This gives a syntactic standard for describing competencies, but does not provide the competencies themselves. The actual competency taxonomies for the Definition part are supposed to be provided by third parties, but so far do not seem to have been produced. We have chosen IMS-RCD elements to form the basis of competency definitions in SeLeNe. An example of their use is shown in Figure 5, and the RDF schema for IMS-RCD elements of the user profile can be found in Appendix D. Genuine competency taxonomies, rather than a subject taxonomy as shown, would be used in SeLeNe when and if production of such taxonomies is taken up in the future.

To ensure getting the maximum amount of information from the users it should be as easy as possible for them to submit information about their current levels of knowledge when they register. This means they will need to be shielded from the intricacies of the taxonomy structure by the user interface. The huge range of knowledge and experience that even a small group of users would bring to the system presents a significant challenge in interface design — encoding all the possible taxonomies of competencies into the interface for the user to choose from seems an insurmountable task. One possible solution may be to allow users to complete a series of free-text boxes (qualifications, professional experience, etc.) and try to infer some of the correct RCD competencies through text analysis or natural language processing techniques. Another possibility, that greatly simplifies matters, arises when resources from RDF taxonomies, rather than string values, are used to represent elements such as *ims-lip:qcl_organisation* and *ims-lip:qcl_level* (as mentioned

standard have yet been published, although the IMS-RCD (Reusable Competency Definitions) may form the basis of such a standard in future.

[2]Also known as "Reusable Definition of Competency or Educational Objective" (RDCEO).

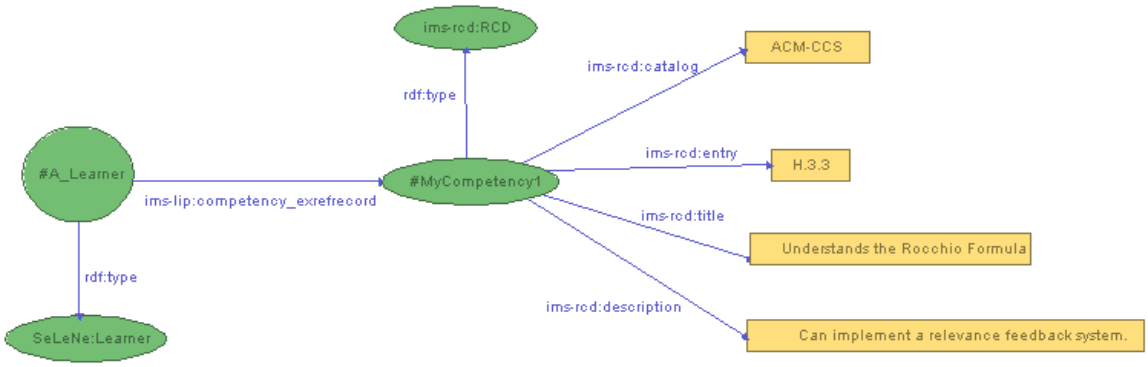

Figure 5: RDF fragment showing the use of IMS-RCD elements in an example user profile

in Section 3.1). If taxonomies of awarding bodies, educational institutions and the courses that they run are available, and the course descriptions have *articulations* to standard curricula descriptions (such as the ACM Curricula Recommendations [2]), then when a learner provides the information that they have studied at institution X and passed course Y there, the learning outcomes of the course can be looked up and automatically added to the 'Competencies' section of the profile. For example, if a user tells the system that they have completed course CS252 at the Computer Science Department of the University of Crete then the learning outcomes from that course (relating to the main principles of Object Oriented Programming) can be automatically added to the user's competencies in their profile. This process is shown in Figure 6.

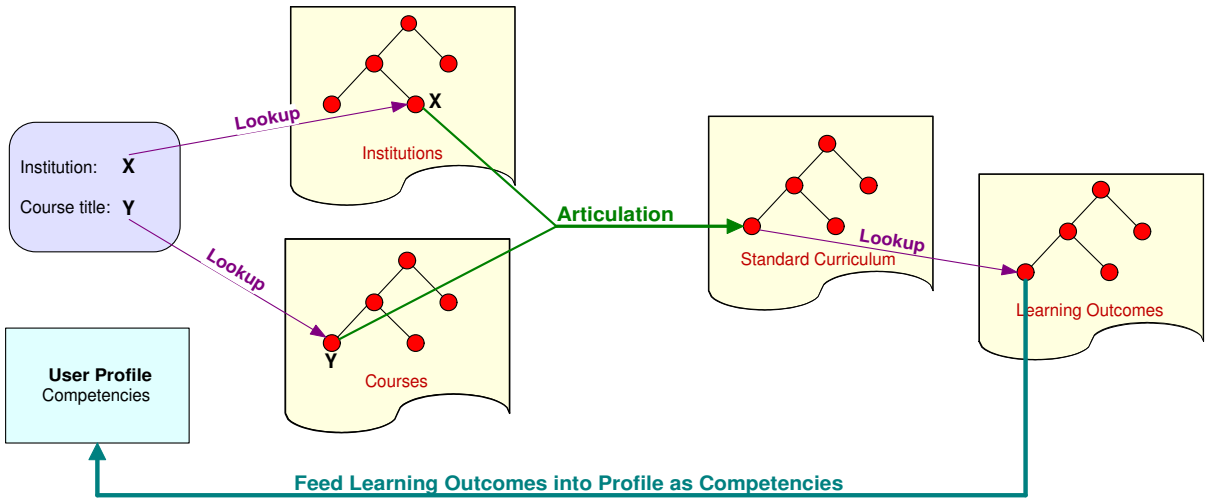

Figure 6: Possible mechanism for automatic extraction of competencies from qualification details

The specification of current knowledge and experience in a form that can be automatically understood provides much potential for future research work; meaningful taxonomies

of competency need to be developed, as do methods for people to specify their competencies without needing to become familiar with a myriad of taxonomies.

5 Specification of Learning Objectives

We said in Section 3.1 that the profile used in SeLeNe will record the current learning goals of the user using IMS-LIP "Goal" elements. These include a textual "Description" as part of them. Rather than simply having a text section of the profile for the user to describe their current aims, it would be useful to have a more formal, structured way for users to express their learning goals and to specify their educational needs. The use of a single restricted vocabulary with proper structure for both user goal specification (in the User Profile) and LO learning objective specification (in the LO descriptions within the SeLeNe repository) would go some way towards providing formal criteria for the "equality of objectives". The availability of such criteria would make matching of LO metadata against a profile easier, thus making more effective personalised selection of LOs possible, and so help the user find LOs that will enable them to achieve their goals.

There seems to be very little work specifically on how learners can express their learning needs; however, there is some advice on how teachers or authors of learning materials can specify the learning outcomes of their lessons, courses and LOs[3]. If learners express their needs in the same terms as LO providers express the desired learning outcomes for their LOs, then accurate matching of LOs to users' profiles should be possible. It seems widely accepted that learning outcomes should be expressed as competencies — what a learner will be able to do as a result of following the learning materials. Most guidelines available are quite general and give a list of "the kinds of word you might want to use". A more formal taxonomy of educational objectives was specified by a committee of college and university examiners in 1956 (known as "Bloom's Taxonomy"), which identified six different types of educational objective, along with outcome-illustrating verbs which characterise each type [4]:

1. **Knowledge:** of terminology, specific facts, universals and abstractions in a field. Knowledge here means the remembering (recalling) of appropriate, previously learned information.

    - *defines, describes, enumerates, identifies, labels, lists, matches, names, reads, records, reproduces, selects, states, views*

2. **Comprehension:** Understanding the meaning of informational materials.

    - *classifies, cites, converts, describes, discusses, estimates, explains, generalises, gives examples, makes sense out of, paraphrases, restates (in own words), summarises, traces, understands*

[3]OUNL-EML [16] includes 'learning objectives' as part of its information model for 'units of learning', but unfortunately gives no hint as to how these should be specified.

3. **Application:** The use of previously learned information in new and concrete situations to solve problems that have single or best answers.

    - *acts, administers, articulates, assesses, charts, collects, computes, constructs, contributes, controls, determines, develops, discovers, establishes, extends, implements, includes, informs, instructs, operationalises, participates, predicts, prepares, preserves, produces, projects, provides, relates, reports, shows, solves, teaches, transfers, uses, utilises*

4. **Analysis:** The breaking down of informational materials into their component parts, examining (and trying to understand the organisational structure of) such information to develop divergent conclusions by identifying motives or causes, making inferences, and/or finding evidence to support generalisations.

    - *breaks down, correlates, diagrams, differentiates, discriminates, distinguishes, focuses, illustrates, infers, limits, outlines, points out, prioritises, recognises, separates, subdivides*

5. **Synthesis:** Creatively or divergently applying prior knowledge and skills to produce a new or original whole.

    - *adapts, anticipates, categorises, collaborates, combines, communicates, compares, compiles, composes, contrasts, creates, designs, devises, expresses, facilitates, formulates, generates, incorporates, individualises, initiates, integrates, intervenes, models, modifies, negotiates, plans, progresses, rearranges, reconstructs, reinforces, reorganises, revises, structures, substitutes, validates*

6. **Evaluation:** Judging the value of material based on personal values/opinions, resulting in an end product, with a given purpose, without real right or wrong answers.

    - *appraises, compares & contrasts, concludes, criticises, critiques, decides, defends, interprets, judges, justifies, reframes, supports*

We believe that this taxonomy can form the basis of a "recommended vocabulary" for the specification of learning objectives in SeLeNe. Obviously, users of the system do not want (or need) to learn the intricacies of Bloom's taxonomy; the user interfaces for registration of users and of LOs should encourage the use of verbs from a representative subset (i.e. a few terms from each category) of the outcome-illustrating verbs given, by presenting them as options when describing learning goals/outcomes. The verbs chosen for this recommended vocabulary are given in Table 1, and are those that would make most sense as part of the specification of both a learner's goals and of a LO's learning outcomes. For example, the verb "understand" has been included and the verb "reframe" has been omitted from the list of representative verbs because we believe a learner is unlikely to specify an objective as "*I want to be able to reframe X*" but may well say "*I want to understand X*". As the subset of verbs chosen cover each of Bloom's six categories they should cover most objectives that learners and authors wish to specify.

Category	Outcome-Illustrating verbs
Knowledge	define, describe, label, list, reproduce
Comprehension	describe, discuss, explain, summarise, understand
Application	assess, determine, implement, predict, report, show, solve, teach, use
Analysis	correlate, differentiate, distinguish, illustrate, recognise
Synthesis	communicate, compare & contrast, design, formulate, model, plan
Evaluation	appraise, compare & contrast, criticise, decide, interpret

Table 1: Subset of representative verbs for each category in Bloom's taxonomy of educational objectives

A user's learning objective (goal) can then be described by a combination of one of these verbs with a term from a learning topics taxonomy. The combination of verbs with a topics taxonomy thus forms a structured vocabulary for describing user goals and LO outcomes. To avoid this being overly restrictive for users the structured vocabulary can be augmented with free text entries when necessary.

For example, using the ACM Computing Classification System (CCS) ontology [1], a user's learning objective might consist of the structured part "*describe* an element of *H.3.3:relevance feedback*" and the unstructured "Rocchio formula" (where "H.3.3" is the category "Information Storage and retrieval, Information Search and Retrieval", which has 'relevance feedback' specified as sub-topic '3' within the ontology). When a LO is found that has the structured "be able to *describe* an element of *H.3.3:relevance feedback*" as part of the 'learning outcomes' section of its metadata, a direct match is found, and then the additional free text parts can be compared to further determine the exact relevance of the LO to the user.

We define a *LearningObjective* class to form the range of the *ims-lip:goal_description* property. *LearningObjective* has three properties, which together provide a framework for the structured description of learning objectives:

1. *learning_objective_verb* — indicates one of the verbs from Table 1 (the possible values are not enforced by the RDF schema at present, but a "learning objectives" data-type could be defined for the range of this property to ensure that only outcome-illustrating verbs are allowed);

2. *learning_objective_topic* — indicates the topic that the learning objective is about, referenced as an entry in the RDF binding of a subject taxonomy (such as that for the ACM-CCS developed at ICS-FORTH as part of this project, which is available at `http://139.91.183.30:9090/RDF/VRP/Examples/ACM-CCS.rdfs`);

3. *learning_objective_annotation* — indicates additional textual description of the learning objective; for example, to specify areas within the topic at a greater level of detail than is catered for by the subject taxonomy.

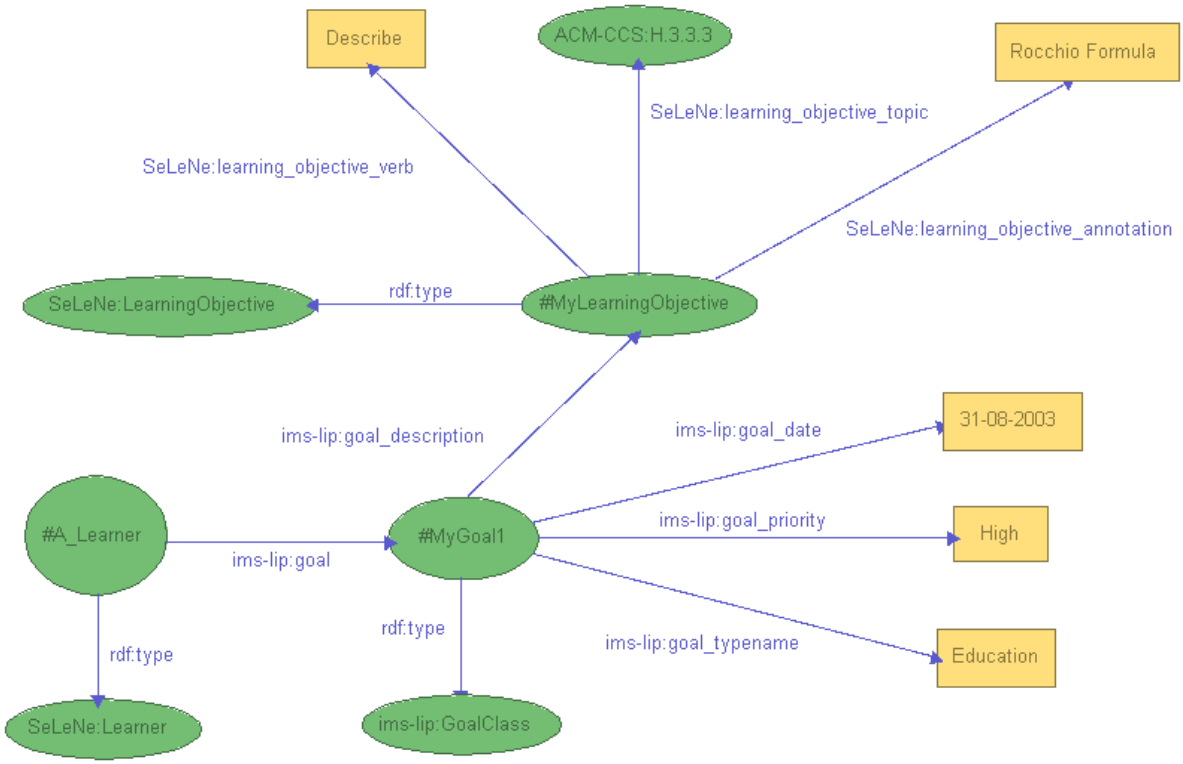

Figure 7: RDF fragment showing the structured learning objective description elements used in a SeLeNe user profile

Figure 7 shows how these properties can be used to codify the information in the example above, and so provide a structured *ims-lip:goal_description* using the *LearningObjective* type. The RDF schema for learning objectives is contained Appendix E.

A condition for "approximate equality" of learning objectives can be defined as follows: two *LearningObjective*s are approximately equal if both their *learning_objective_verb* and *learning_objective_topic* are the same. The equality is only approximate because the *learning_objective_annotation* parts of the two learning objectives could differ. We believe that the annotation part is very important, and so should not be omitted from the profile, even though this omission would give a simple condition for "full" equality of learning objectives. If the learning objective were to be specified by only a *verb* plus a *topic*, users would be far too restricted in what they could express. The most detailed specification a learner could give of their goals would be something like "I want to *assess* Object Oriented Languages" (OOP is D.3.2.11 in the ACM-CCS taxonomy), which is far too general. Any learner who had this as a broad learning objective would want to break it down into several smaller, short-term, achievable objectives when searching for LOs to interact with — "I want to *understand* C++", "I want to *criticise* Java", and so forth. The additional annotation is crucial to specifying the learning objectives to a reasonable level of detail and so only approximate equality of learning objectives can be defined. This is still much better than the

situation where matching can only be done by comparing completely unstructured blocks of text.

The user interface is the key to ensuring that terminology used in LO metadata matches that in learner profiles as closely as possible. By presenting the same structured vocabulary to users registering a LO as to users describing their learning goals we avoid as much ambiguity as possible, while still allowing full expressiveness in free-text parts of the description.

6 Specification of Learning Styles

Another useful piece of information that is not catered for by existing learner profile schemes is the learning and cognitive styles of the learner. Different people learn in different ways, so what might be a useful LO for one person might not suit someone with a different learning style. For example, an interactive applet that allows the user to "virtually" experiment with different electric circuit layouts may help an "activist" learner to understand the relationship between current, voltage and resistance in a circuit. The same applet may just confuse a "theorist" learner, who would be better off with a LO that simply gives Ohm's Law as $V = IR$.

We think it is important for a learner profile to include the learning and cognitive styles of the learner. If the learning/cognitive styles catered for by a LO are recorded in its description, then these can be matched against the learner's styles to suggest more useful LOs to them (possibly at the Filtering and Ranking stage of the personalisation process - see Section 8.4). As with the specification of learning objectives (Section 5) we believe that provision of a structured vocabulary for the description of learning/cognitive styles is preferable to simple free-text description. Unlike the case of learning objectives, several different taxonomies have already been produced for learning and cognitive styles, many of which have standard tests that can be taken to reveal which styles a person is biased towards. We suggest that a learning/cognitive style can be described by the specification of a combination of a learning styles taxonomy with a particular style from that taxonomy. Thus the learning style class has a pair of properties, $\{Taxonomy, Descriptor\}$ that specify a learning style. This is similar to the way that the pair of properties $\{Catalog, Entry\}$ specify a competency in the IMS-RCD schema.

Table 2 gives a list of learning and cognitive style taxonomies that could be used and Figure 8 gives an example showing the use of one of these learning style taxonomies to specify one of the user's learning styles in the SeLeNe profile. The RDF schema for learning and cognitive styles can be found in Appendix E. This schema could be simplified if, in future, RDF representations of the various learning/cognitive style taxonomies are made available. The *LearningStyle* type with its *learning_style_taxonomy* and *learning_style_descriptor* properties could be done away with and the *learning_style_preference* property could instead take as its range direct references to the learning style taxonomies, in much the same way that *learning_objective_topic* references subject taxonomies at present (see Section 5).

This is a preliminary proposal, and its application needs to be explored more fully

Learning/Cognitive Style Taxonomy	Learning/Cognitive Style Descriptors
Gardner	Linguistic, Logical-Mathematical, Bodily-Kinesthetic, Musical, Spatial, Interpersonal, Intrapersonal
Gregorc	Concrete Sequential, Abstract Sequential, Concrete Random, Abstract Random
Honey and Mumford	Reflector, Theorist, Activist, Pragmatist
Witkin's GEFT	FD-EF, FD-PF, FM-EF, FM-PF, FI-EF, FI-PF
Kolb LSI	Converger, Diverger, Assimilator, Accommodator
Myers-Briggs	ESTJ, INFP, ESFP, INTJ, ESFJ, INTP, ENFP, ISTJ, ESTP, INFJ, ENFJ, ISTP, ENTJ, ISFP, ENTP, ISFJ

Table 2: Examples of learning and cognitive-style taxonomies, and their respective style descriptors

before a decision can be made as to which cognitive and learning style taxonomies will be the most useful in this context. The list of taxonomies given in Table 2 may need to be modified (e.g. Dunn and Dunn's taxonomy [7] or Sternberg's "thinking styles" [24] may also be useful, or some of the taxonomies may turn out not to be useful). Some of the taxonomies have been criticised for being too general to apply in specific situations, and it may be that new, more concrete taxonomies need to be developed for use in specific applications such as SeLeNe.

It is possible that mappings between the different taxonomies exist; for example, it may be that Honey and Mumford's "Activist" type corresponds closely to Gregorc's "Concrete-Random" type. In this case one area for future research would be to articulate the mappings between different cognitive and learning style taxonomies so that SeLeNe can syndicate learning styles described using different taxonomies (i.e. form a set of *articulation*s [20] from one learning style taxonomy to another).

7 Support for SeLeNe's Active Functionality

As described in *SeLeNe Deliverable 2.2: Self e-Learning Networks — Functionality, User Requirements and Exploitation Scenarios* [11] and elaborated upon in *SeLeNe Deliverable 4.4: ECA Rule Languages for Active Self e-Learning Networks* [17], users can choose to be automatically notified of various types of change in the metadata repository. A *Notifications* section of the user profile is provided to store these personal notifications until the user has chosen to read their pending notifications. *Notifications* has three properties, each with an *rdf:Bag* as its range. The three properties reflect the three types of event that users can choose to be notified of:

1. *new_LOs*, that represents the addition of new LOs to the repository that may be of interest to the user;

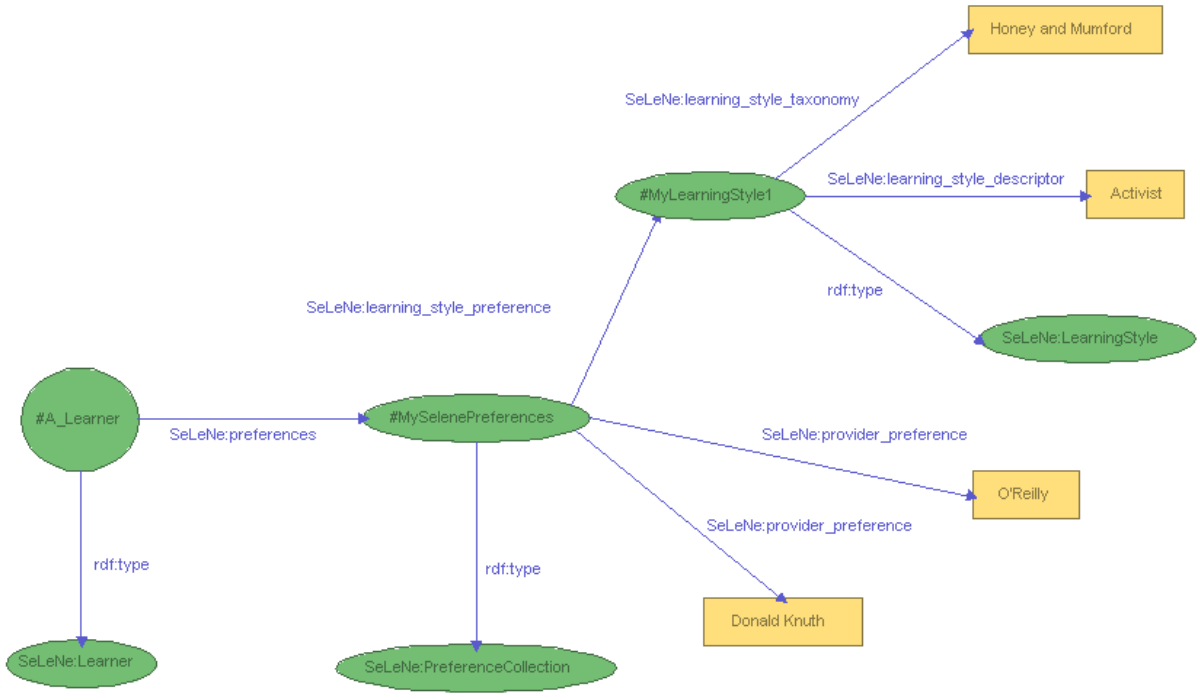

Figure 8: RDF fragment showing a structured learning style description in a SeLeNe user profile

2. *updated_LOs*, that represents updates to LOs that the user already knows about and has 'subscribed' to be told of changes to;

3. *new_users*, that represents the registration of new users who may be of interest to the user (i.e. they have similar interests, are part of the same class, etc.).

When a user submits a request for notification of some event to SeLeNe an ECA rule is generated and stored in the rule base. When the event the user asked to be notified of occurs, the appropriate ECA rule is triggered to perform the action specified in its "action" part. The "action" part in the case of user-notification rules will specify the insertion of a new property between one of the *Notification* bags of the user's profile and the resource affected by the event that originally triggered the rule. For example, when a LO that the user has subscribed to is updated, a *rdf_x* property is added between the *updated_LOs* bag in their profile and the resource that was updated.

The *Presentation* service should check the user profile for entries in the *Notification* section every time the user logs on to the system, and also periodically while they are logged on. When there are notifications in the profile a message should be displayed to the user. Once the notification message has been received by the user they can choose whether or not to delete it from their profile.

Figure 9 shows an example user profile with two pending notifications — one of a new LO of interest and one of a new user of interest. The RDF schema for the elements

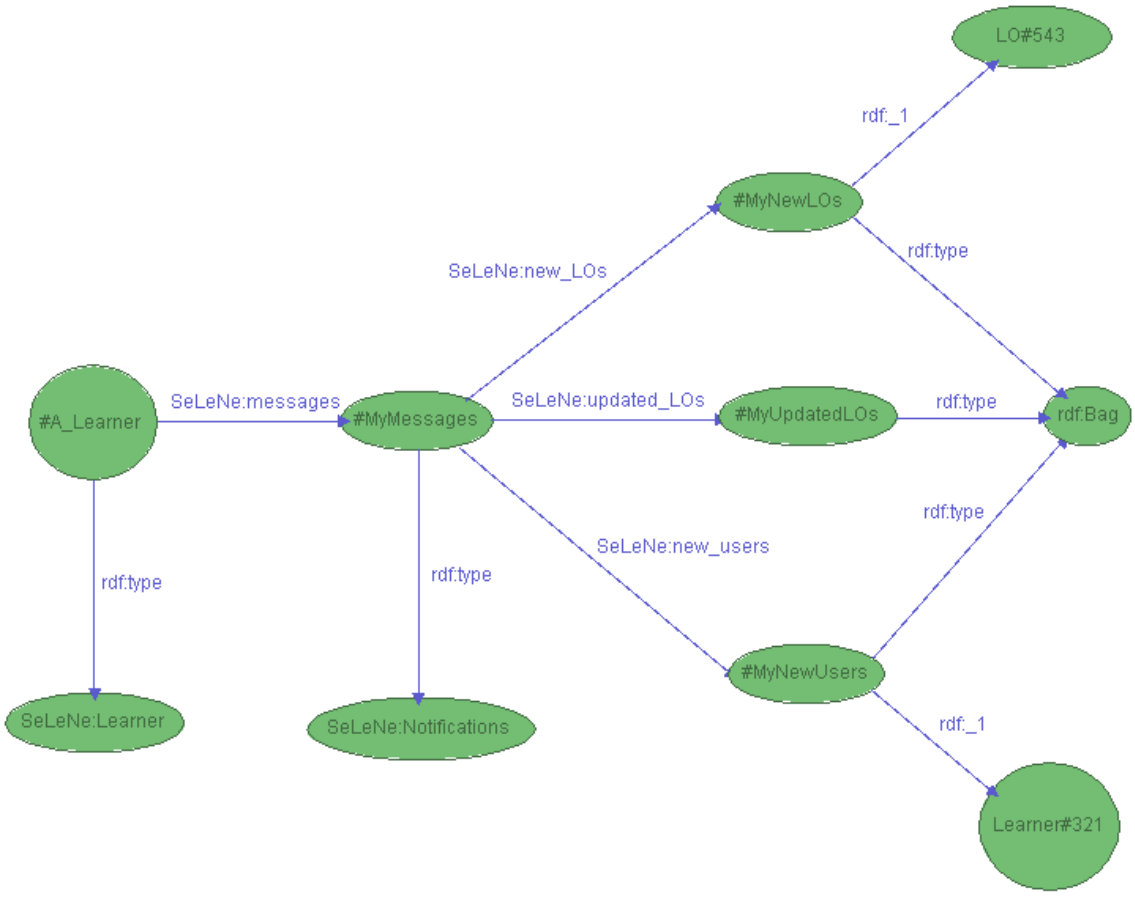

Figure 9: RDF fragment showing the profile elements used by automatic notification services in SeLeNe

described in this section can be found in Appendix E.

8 Search and Retrieval

The SeLeNe metadata repository exists to allow users to search for the accessible LOs which are of interest to them. The user interface will allow the user to find LOs by simply submitting keywords, possibly with annotations which augment the search terms with known metadata terms, as specified in *SeLeNe Deliverable 2.2: Self e-Learning Networks — Functionality, User Requirements and Exploitation Scenarios* [11]. Such a search will present the results of a query as a ranked list of LOs, which the user can then browse. There are various possibilities for the underlying search mechanism acting over the RDF data, two of which are discussed in the following two sections.

8.1 Translation of Unstructured Queries into RQL

The first possibility is to query the RDF data directly, via a structured query language (*RDF Query Language* — RQL) over the RDF metadata distributed across the SeLeNe, or possibly over a user-defined view of this metadata (defined using the *RDF View Language* — RVL).

The result returned by one of these queries is an unordered set of LO metadata that satisfies the (RQL) query. However, the result eventually returned to the user is a ranked list of LOs. For this to be possible some translation and mediation is needed in two places:

1. Translation is needed between the user's search box, which collects keyword queries, and the SeLeNe *Query* service, which executes RQL queries.

2. Ranking and filtering of the results must be performed before they are presented to the user.

This section discusses issues relating to (1), and Section 8.4 covers (2).

In cases where the keyword query is annotated to specify which parts of the LO description should be searched, translation into an RQL query is relatively straightforward. It is simply a matter of fitting the keywords and attributes specified by the user into an RQL "template" query, with one entry in the `FROM` clause for each attribute specified, and one in the `WHERE` clause for each keyword. For example, translation of the query

author:Newton *course*:calculus

into RQL is achieved by forming the query

```
SELECT X
FROM {X}author{Y},{X}course{Z}
WHERE Y LIKE '*Newton*'
AND Z LIKE '*calculus*'
```

Clearly, in order to construct annotated keyword queries of this sort the user must have knowledge of the RDF schema in use. The user should not be expected to learn the schema, and instead support for this mode of querying should be provided by showing a list of the attributes that exist in the schema along with the search interface.

Cases where no annotation is provided are likely to be the more commonly used mode of search. Research into Web search behaviour has shown that typically users submit only 2-3 keywords as a query, and rarely use advanced search options [10] — there is no reason to think that typical user behaviour will differ when searching a SeLeNe for LOs. This "free text" only search is more problematic to deal with. The most obvious and simplest solution is to assume that the user wants to find LOs that contain the query terms anywhere in the metadata describing them (i.e. in any attribute). We can then take the list of keywords to be either:

- a disjunctive query, and retrieve metadata about each LO that contains *any* of the query terms anywhere within it

or

- a conjunctive query, and retrieve metadata about each LO that contains *all* of the query terms anywhere within it.

So given a query "<term$_1$ >...<term$_i$ >...<term$_n$ >" we could construct and execute an RQL query of the form:

```
SELECT X
FROM {X}@P{Y;string}
WHERE Y LIKE '*term 1*'
OR Y LIKE '*term 2*'
⋮
OR Y LIKE '*term i*'
⋮
OR Y LIKE '*term n*'
```

For the conjunctive version of the query we would simply need to replace the `OR`s with `AND`s. In the previous annotated case (above) the `AND` could similarly be replaced with `OR` if we wanted to take the user's query as being disjunctive. However, we believe that the default behaviour should be as in the queries shown — using conjunction in the annotated case (as the returned LOs should exactly match the user's specification) and disjunction in the non-annotated case, ranking those LOs matching more of the query terms higher than those with fewer term matches.

The problem is that an exhaustive search of this sort will have very poor performance when there are more than a few attributes to search. If RQL is to be used then the space of attributes to search must be narrowed down in order to give acceptable performance.

There are two possible approaches to reducing the search space:

1. The simplest way is to notice that even though no annotation is provided by the user, in reality there are only a few fields of the LOM descriptions that keyword search will be appropriate for. Users are most likely to be searching for LOs on a particular subject or by a particular author, so rather than searching every attribute of the description satisfactory results could be achieved even where the search is limited to only a small number of attributes, say *General.Title*, *General.Description*, *General.Keyword* and *Lifecycle.Contribute.Entity:VCard.FN* (i.e. the full name of the contributor).

2. A more sophisticated approach would be to "guess" which fields should be searched for a particular search term, and then search only a limited number of fields for each. If global statistics on keyword occurrence within the distributed RDF repository are available then these could be used to aid the decision of which fields to

execute the search on, i.e. the most likely fields only. For example, if it is known that globally 98% of occurrences of the term "John" appear in values of the *Lifecycle.Contribute.Entity:VCard.FN* field then when "John" is submitted as part of a query the system could decide to only search this field.

Either of these approaches should make the use of RQL queries over the RDF descriptions feasible in terms of performance.

8.2 Indexed Search

A second possible strategy for executing keyword-based queries over the RDF descriptions would be to use techniques developed in the areas of Information Retrieval and Web Search, which have proved highly effective in supporting keyword-based queries over other kinds of document corpora in the past (for example, library catalogues and hypertext collections).

These techniques require an index to be maintained, which maps keywords contained in the resources (or in our case in the resource descriptions) to a list of the resources containing that keyword. In SeLeNe the index would have to contain all the terms that occur in LO descriptions as part of any attribute, for each term listing the descriptions (i.e. their location in the RDF repository) that contain that term within them. When a keyword-based query is submitted the keywords can be looked up in the index to find the list of descriptions containing any or all of the search terms. These descriptions can then be retrieved, ranked and presented to the user.

A problem with applying these techniques in the SeLeNe environment is that there is not necessarily knowledge of the entire corpus (i.e. collection of LO metadata) at any one node, so in such cases no full index can be created. A possible solution to this could be to maintain a partial index at each node, covering only the LO metadata stored at the node itself. This partial index could be updated each time a new LO is registered (or unregistered), ensuring that the index at each node is always 'fresh'. The *Query* service would then need to send any query submitted to it to all of its peers for evaluation and then aggregate the returned results into a ranked list. In fact, this is not all that different from the way that large Web search engines such as Google (*www.google.com*) work already[4] — their indices are far too large to be stored on a single machine so they are distributed across many, and queries must be executed on many partial indices before the overall results are collected together. This shows that the technique is certainly feasible in an environment such as SeLeNe, the main difference here being that a search engine can decide centrally how the indices should be distributed to give optimal performance whereas in SeLeNe the configuration of the partial indices would be dictated by geography alone.

[4]There are also similarities to *Meta*-search engines, which collect results from multiple search engines and aggregate the results from each into a single ordered list.

8.3 Dynamics of the Search Process

With either of the above search strategies (RQL or indexed search) it might be desirable to do some semantic matching to expand the keyword set before submitting a query to the *Query* service. This could be done using a thesaurus such as WORDNET [13] to find synonyms and other words related to the query terms, in order to expand the range of LO descriptions retrieved. In the case of an RQL query, extra "`OR Y LIKE...`" clauses could be added, and for indexed search the related words could simply be tagged on to the end of the user's original query.

Queries could be submitted to the *Query* service directly from the user interface. This would be trivial in the case of indexed search, and where the *Query* service requires structured RQL queries these could be constructed on the client machine, within the user interface program (for all but the option using global statistics this would be possible even if the interface is browser-based, as long as some form of scripting is enabled in the browser). However, if the first stage of personalisation — automatic query reformulation — is to take place, then all queries should instead be routed through the *Trails and Adaptation* service for translation and reformulation (as shown in Figure 1). Here the query is augmented with extra information from the user profile, for example which languages the LOs should be in and which learning styles they should support. This should reduce the amount of post-retrieval *Filtering* needed, and so speed up the response time (as it should be much quicker to add a few search terms before query execution than to filter out potentially hundreds of undesirable LOs later).

The *Query* service should not return the results directly to the user, but should instead route them back via the *Trails and Adaptation* service. This service will then have:

- a set of query results
- the query that was originally issued
- the user's profile

and from these can generate a filtered, ranked list of LOs to return to the user, as described in the next section.

8.4 Filtering and Ranking Search Results

The *Query* service will return a set of LO descriptions — all those LOs that satisfy the user's query. The user wants to be able to find exactly the right LO quickly, without having to browse too many of the results, so rather than present the results exactly as they are returned by the *Query* service some processing is done first.

If a profile of the user is not available (or the user has personalisation turned off) then all that can be done at this stage is some rudimentary ranking of the result set, possibly using standard ranking techniques from Information Retrieval and Web search. One possibility for this would to be to employ the Vector Space Model from traditional IR [3]. Feature vectors representing the textual parts of the LO descriptions would need to be constructed

(as the system has no access to the LOs themselves), as well as one representing the user query. The Web search technique of giving variable weightings to keywords depending on their position in the text could be applied in the construction of the feature vectors (for example, a description where a keyword matches in the "Title" will be weighted higher than one where it only occurs in the "Description"). The similarity of the LO description to the query can then be calculated using the cosine similarity measure (i.e. the cosine of the angle between the vectors representing the query and the LO under consideration), and the results ranked in order of similarity.

However, we anticipate that usually some minimal profile will be available to the system, as users should supply at least some minimum information into their profile when first registering. In this case the ranking of LOs will involve personalisation. This means that the system can attempt to show the user only those results likely to be most relevant to them personally, as well as relevant to the query in general.

The first step in this processing is to *Filter* the results — remove all those LOs that we are certain will be of no use to the user. At this stage, for example, any LOs in languages that the user does not understand can be eliminated, as can those not meeting accessibility requirements, those at a far too high or low level of difficulty and possibly those covering only material that the learner is already completely familiar with.

Next, the remaining set of LO descriptions must be *Ranked* in order of relevance to the user. Whereas filtering can be done with just the user profile, ranking a set of results should take the original query into consideration too (i.e. relevance must be judged against the combination of user profile and query, not just the profile).

The best algorithm to use for this ranking is still an open question, but it will take into consideration:

- Relevance of the LO to the query (possibly judged using Information Retrieval techniques as described above)
- How well the LO caters for the user's accessibility requirements
  - This can be ascertained by matching accessibility information in the LO description (the *LOM:Classification.Purpose.Accessibility* element) with the *papi:Learner_Preference_Info* element of the user profile.
- Whether the user has the prerequisite knowledge and experience
  - Several things will need to be considered to ascertain this. The most simple is to check that the level of the LO matches the level that the learner is studying at (the *LOM:Educational.Context* gives a rudimentary indication of this, and the user's age, contained in the *papi:personal_bucket* of the profile can be checked against the *LOM:Educational.TypicalAgeRange* part of the LO description).
    The *Requires* relation provided in LOM should give more details about what the user needs to know before tackling the current LO, but this relates to specific prerequisite LOs that the learner should have used, rather than to prerequisite

knowledge in general. However, information about prerequisite LOs can be translated into information about prerequisite knowledge using the technique described at the end of Section 4: the learning outcomes from the required LO can be *articulated* into the competencies that the learner requires, and these can then be matched against the IMS-RCD competency definitions in the user profile to find if the user has the prerequisite competencies.

- Matching between the user's goals and the learning objectives of the LO
  - The user's goals can be found in the *ims-lip:goal* element of the user profile, described using the *SeLeNe:LearningObjective* type. The LO descriptions should reference similar *LearningObjective* types showing what the user of the LO can expect to learn from it. These can then be easily matched by *verb*, *topic* and *annotation* using the condition for approximate equality of objectives given in Section 5.
- If the user's learning styles are those catered for by the LO
  - The user's learning styles are recorded using their *PreferenceCollection*'s *learning_style_preference* properties. These can be matched with the part of the LO descriptions that detail which learning styles the LO caters for best.
- If the user is likely to prefer it for other reasons (it is by a preferred author, say)
  - Any *provider_preference*s in the *PreferenceCollection* can be matched against the *LOM:LifeCycle.Contribute.Entity* parts of the LO description. The *ims-lip:interest* parts of the profile can also be considered here.
- The user's most recent activity
  - Updates can be made automatically to other parts of the profile, as described in Section 3.2, and "good next LO to choose" recommendations can be given on the basis of *trails* derived from the LO descriptions, as described in Section 9.3.

The clear individual semantics of each section of the user profile allows focussed matching against relevant sections of the LO descriptions. For a LO to be a "good" LO for the user, the greatest possible number of different elements will match to some degree. Clearly, though, some factors are more important than others to the user and a good algorithm for combining them will reflect this. For example:

- If LO X caters for one of the user's learning styles but is not very relevant to the original query then other, more relevant LOs should be ranked higher even if their descriptions don't list one of the user's learning styles;
- If LO Y has a learning outcome that matches one of the user's goals but is far too difficult for the user to tackle (they have none of the prerequisite knowledge, say) then again other LOs (closer to the user's level) should be ranked higher.

With so many factors to take into consideration, discovery of which algorithms work better or worse for which groups of users requires much further work and testing, and is beyond the scope of this project. It may be that the ranking algorithm itself needs to adapt to the individual, and will differ from user to user (an additional section could be added to the user profile to store information about parameters used by the ranking algorithm).

9 Trails

9.1 Advanced Learning Material Modelling

Current learning technology specifications allow for some simple ordering and sequencing of e-Learning resources to be specified in a way that can be understood by learning management systems (LMSs) (for example SCORM, IMS Content Packaging and IMS Simple Sequencing). OUNL-EML, which is being used as the basis for the IMS Learning Design specification (where it is integrated with IMS Content Packaging and IMS Simple Sequencing), adds the ability to integrate learning designs ('instructional designs') to enable more advanced e-Learning applications [5], by allowing the semantic description of LOs in the context of a particular unit of learning (e.g. a course), and the relationships between LOs in the workflow for learners and instructors. This allows for the expression of the course designer's own pedagogical models.

The expressiveness and detail provided by these schemes is much more than is required in SeLeNe's trails, as we do not envisage SeLeNe playing the role of a LMS. All that is required is a lightweight way to represent sequences of LOs as *trails*.

9.2 Trails in SeLeNe

Trails are simply linear sequences of LOs, which can have additional annotation to explain why the LOs have been sequenced as they have and to provide extra pedagogical information that is not included in the descriptions of the LOs themselves. For example, there may be learning outcomes that the trail supports but that the individual LOs do not, as the pedagogic purpose of a trail may well be different from that of its components.

In the RDF representation of trails we define them as a sub-class of the RDF *Sequence*, the *Sequence* being the sequence of LOs forming the *trail*. Two associated properties, *name* and *annotation*, provide any necessary additional information. Appendix F gives the RDF schema for SeLeNe trails and their properties. Figure 10 shows the use of an *Emergent Trail* to represent the history part of a learner profile.

As discussed in *SeLeNe Deliverable 2.2: Self e-Learning Networks — Functionality, User Requirements and Exploitation Scenarios* [11] we see three distinct classes of trail appearing in a SeLeNe: authored, derived and emergent. Trails can be either user-defined or automatically generated by the system. User-defined trails can be either explicitly authored (*authored trails*) or emerge from user browsing activity (*emergent trails*). Automatically generated trails are derived from semantic links between LOs, inferred from their

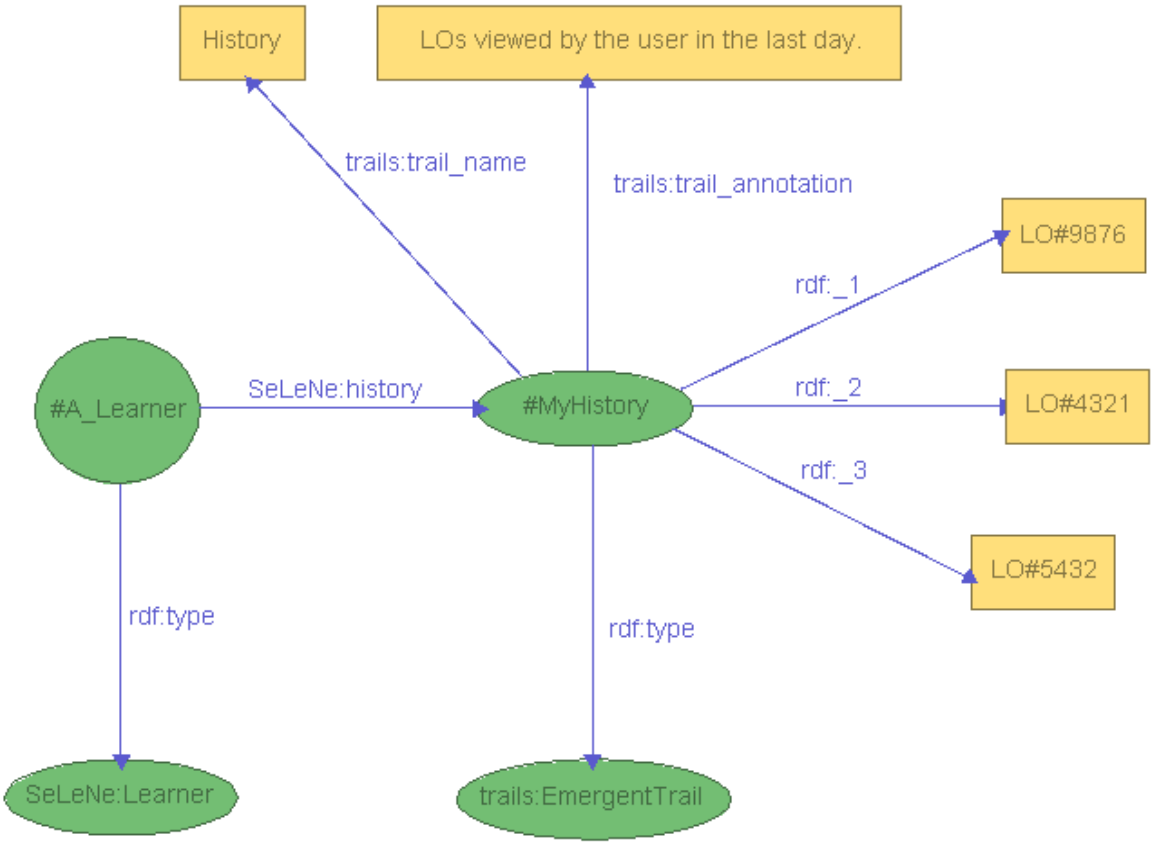

Figure 10: RDF fragment showing the use of a SeLeNe *Trail* to represent the History part of an example user profile

descriptions (*derived trails*).

Recording a trail should be possible with minimal user effort. For authored trails this means that a trail authoring tool is a necessary part of the SeLeNe user interface. For emergent trails data mining techniques can be applied to the 'history' section of the user profile to find emergent trails with little or no effort needed on the user's part. Derived trails are derived by the system and by definition require no effort from the user.

The trails available as "independent" LOs (i.e. those that have a description in the SeLeNe metadata repository and that can be referenced by a URL) will only be authored trails, but the other types of trail can serve as the basis for authored trails. When the system presents an emergent or derived trail to the user (as they inspect their history, or as derived trails are created from the results of a user query) the user will have the option of importing the trail into the trail authoring tool. The user can then annotate and edit the trail as they see fit before saving it and making it available at a URL. In this way SeLeNe provides support to trail authors — the system can suggest trails as a starting point for the authoring process, rather than the authors having to start from scratch each time.

There is also the question of how trails should be presented to the user. Detailed user interface design is beyond the scope of this project, but in any implementation several things will need to be considered:

- The presentation of a trail should convey as much information about the trail as possible, while not confusing the user with too cluttered a display;
- Interaction with a trail should be intuitive;
- There should be navigational support so that the user always knows where in a trail s/he is — some sort of "you are here" marker will probably be useful.

Some work has already been done on visualisation of RDF, including ICS-FORTH's RDF-Suite [19] that provides clickable browsing of RDF schema and descriptions, and the W3C's IsaViz RDF authoring tool [25] that provides rudimentary clickable graphs. Further work is needed before this kind of technology can be smoothly integrated into a usable user interface for a system such as SeLeNe.

9.3 Deriving Trails using Semantic Relationships

SeLeNe gives the user the option of having a list of *trails* of LOs returned from a query rather than a list of individual LOs. The trails most likely to be of use in this context will be short (maybe 2-3 LOs in length) and informative. Algorithms for the automatic generation of trails in hypertext systems already exist, such as the Best Trail Algorithm [26], where trails are formed by creating paths through the hyperlinks between pages. The trails returned by SeLeNe must be automatically generated, but the links between LOs are not explicit hyperlinks — they must be derived by the system from the information contained in the LO metadata.

In order to derive trails of LOs from their RDF descriptions the semantic relationships between LOs need to be queried. Some of these will be explicitly specified in the LOM:Relation element of the LO descriptions, although the kinds of relationship explicitly supported by IEEE-LOM are limited and need to be expanded to include relationships such as "...is a prerequisite for...", "...is a slower-paced version of...", "...elaborates upon...", "...gives background material on..." and "...is in the same curriculum as...". Once the set of possible relationships that can be described is expanded, forming trails of LOs using these explicit relationships becomes relatively straightforward: from a collection of LOs containing LO_1, LO_2 and LO_3 (among others), where LO_1 is a prerequisite for LO_2, which is in turn a prerequisite for LO_3, then the system can easily derive the trail

$$LO_1 \rightarrow LO_2 \rightarrow LO_3$$

with the annotation "prerequisites" by simply inspecting the LOM:Relation field of each description.

As well as supporting the derivation of trails the user may also want to query these semantic relationships directly. For example, the user should be able to ask the system

to "find me a LO that elaborates on material in this one here", or "find me a LO that gives background/preliminary material on this LO". The advanced search interface should therefore allow for this mode of querying.

9.4 Support for Browsing as a Trail

As the user is browsing LOs (and the browsing behaviour is logged to the history trail in the user profile) the *Trails and Adaptation* service can actively recommend the next LO to look at, effectively generating trails of length two (i.e. a trail consisting of the current LO and a suggestion for the next one) at every stage of the user's browsing, based on the user profile.

The recommendations can be derived in several ways:

- from the semantic relationships between the current LO and other LOs in the SeLeNe repository, as described in Section 9.3 above;
- from the user's profile plus LO metadata — perhaps suggesting LOs that cover more advanced material on the same topic, and also suit the user's preferences (learning style, accessibility, etc.);
- through a process of collaborative filtering, suggesting as the next step a LO that other similar users browsed after seeing the current LO (where similar users can be identified by having similar preferences or similar histories of LO access).

10 Open Challenges

Many issues for further research and development have been identified in this deliverable. Some of these are issues specific to SeLeNe, that will need to be resolved before any complete implementation of the system is possible. Specifically:

- This deliverable has concentrated on the semantics of the SeLeNe learner profile. While much of the discussion is general to all SeLeNe profiles, precise semantic models for *Group* and *LO Provider* profiles still need to be defined (some of the idiosyncracies of group profiles are discussed in Section 3.1, but an exact specification is needed).
- The generic set of ECA rules governing the update of a profile based on its 'history' section, providing SeLeNe's adaptive functionality, needs to be formally specified.
- User interface design will be crucial to the usability, and hence the usage of, SeLeNe's advanced functionality. When inputting data about themselves or LOs, users must be shielded from the complex taxonomies of topics, competencies and goals in use by the system. Interaction with trails of LOs needs to be intuitive.

Other issues are more general and highly relevant to the e-learning community as a whole, as well as to any future development of SeLeNe:

- Perhaps the most important issue is the question of how a profile should be matched against LO metadata — in the context of SeLeNe this arises in providing a personalised ranking of query results, but the more general question of how to tell if a LO is likely to be relevant and useful to a learner will be an issue for any e-learning system attempting to do some form of personalisation. As such, this issue merits a great deal of further work, both theoretical and empirical.

- A problem for any system employing profiling is that of keeping the profile up-to-date. Whilst users are more inclined to keep a profile up-to-date manually if they can immediately reap the benefits of doing so, adaptive technology is needed to free the user from this burden altogether. ECA rules monitoring the user's activity history is one approach to providing this adaptation, but there is huge scope for the development of other approaches to automatic update and adaptation of user profiles.

- Many taxonomies of learning and cognitive styles exist, some competing and others complementary to one another. Not all of these will provide categorisations useful for e-learning systems, and some taxonomies may be closely related to, or possibly even subsumed by, others. Discovery of which taxonomies are useful for e-learning applications will require empirical testing, as will the discovery of any mappings that exist between taxonomies. We believe that the creation of a standard ontology of learning styles for use in e-learning would be an important development.

- A further issue regarding learning styles is the question of how knowledge of them should be used in matching LOs to a user — it may be that presenting learners with only LOs that suit their learning styles proves to be detrimental to their learning, as they will only ever consider problems from a limited range of perspectives. If this is the case then an e-learning system should on occasion deliberately present LOs that cater to learning styles *different* from those of the learner, in order to stretch the learner's understanding of a concept.

- The user's current competencies are a crucial part of any learner profile. Development of standard taxonomies of competency definitions would enable the widespread use of the IMS-RCD syntactic framework for the recording of user competencies.

- Once competency definitions exist, a useful further development would be articulations (mappings) from qualifications to competencies, to allow automatic inference of user competencies from qualification data (as described in Section 4).

Finally, it should be noted that there are many more LOs available on the Web than will ever be explicitly registered by users of a SeLeNe. Of course, SeLeNe users can submit a description of any Web-based resource they come across for inclusion in the SeLeNe repository, but an interface to these Web-based LOs without requiring their explicit registration by a user would be a useful facility. One possible way to achieve this would be to take a user's query and, as well as searching the SeLeNe repository, execute a search using a Web search engine. A minimal RDF description of the resources returned by the Web search

could be automatically generated, and the results then personalised by ranking according to the user's profile or organising them into trails, thus augmenting Web search with the benefits of SeLeNe's personalisation services[5].

[5]Ideally this would employ a specialised Web search engine using an index of learning material only. Current search engines return mainly non-learning-oriented material for most queries. The strategy described here could go some way to identifying LOs relevant to the user, but the results from a general-purpose Web search would probably remain far inferior to those from LOs explicitly registered with SeLeNe.

References

[1] Association of Computing Machinery. *ACM Computing Classification System (ACM-CCS)*. http://www.acm.org/class/1998/.

[2] Association of Computing Machinery. *ACM Curricula Recommendations*. http://www.acm.org/education/curricula.html.

[3] R.A. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press and Addison-Wesley, Reading, Ma., 1999.

[4] B. S. Bloom and D. R. Krathwohl, editors. *Taxonomy of Educational Objectives: The Classification of Educational Goals: Handbook I, Cognitive Domain.* Longman, New York, 1956.

[5] M. de Croock et al. Alfanet Deliverable D12 – State-of-the-art, October 25th, 2002.

[6] P. Dolog and W. Nejdl. Challenges and benefits of the Semantic Web for user modelling. In *Proceedings of AH2003: Workshop on Adaptive Hypermedia and Adaptive Web-Based Systems*, Hungary, USA and UK, 2003.

[7] R. Dunn, K. Dunn, and M.E. Freeley. Practical Applications of the Research: Responding to Students' Learning Styles – Step One. *Illinois State Research and Development Journal*, 21(1):1–21, 1984.

[8] IEEE Learner Model Working Group. (p1484.2) Ballot to adopt: PAPI Learner Version 7, Dec 2000.

[9] IMS Global Consortium Inc. IMS. IMS Reusable Definition of Competency or Educational Objective - Information Model, Version 1.0 Final Specification, 2002.

[10] B.J. Jansen and U. Pooch. A review of Web Searching Studies and a Framework for Future Research. *Journal of the American Society for Information Science and Technology*, 52:235–246, 2001.

[11] K. Keenoy, G. Papamarkos, A. Poulovassilis, M. Levene, D. Peterson, P.T. Wood, and G. Loizou. SeLeNe Deliverable 2.2: Self e-Learning Networks – Functionality, User Requirements and Exploitation Scenarios, 2003. Available from `http://www.dcs.bbk.ac.uk/selene/reports/`.

[12] A. Magkanaraki, V. Christophides, and D. Plexousakis. SeLeNe Deliverable 4.3: Views and Structured Querying in Self e-Learning Networks, 2003. Available from `http://www.dcs.bbk.ac.uk/selene/reports/`.

[13] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller. *Five Papers on WordNet.* Princeton Cognitive Science Laboratory (`http://www.cogsci.princeton.edu/~wn/`), New Jersey, 1993.

[14] B. Mobasher, R. Cooley, and J. Srivastava. Automatic Personalization based on Web Usage Mining. *Communications of the ACM*, 43(8):142–151, 2000.

[15] LTSC Learner Model Working Group of the IEEE. IEEE p1484.2/d7, 2000-11-28 Draft Standard for Learning Technology – Public and Private Information (PAPI) for Learners (PAPI Learner). Technical report, 2000.

[16] Open University of The Netherlands. *OUNL-EML.* http://eml.ou.nl/.

[17] G. Papamarkos, A. Poulovassilis, and P.T. Wood. SeLeNe Deliverable 4.4: ECA Rule Languages for Active Self e-Learning Networks, 2003. Available from `http://www.dcs.bbk.ac.uk/selene/reports/`.

[18] A. Rawlings, P. van Rosmalen, R. Koper, M. Rodríguez-Artacho, and P. Lefrere. CEN/ISSS WS/LT Learning Technologies Workshop - "Survey of Educational Modelling Languages (EMLs)", September 19th, 2002.

[19] ICS-FORTH RDFSuite. http://139.91.183.30:9090/rdf/.

[20] P. Rigaux and N. Spyratos. SeLeNe Deliverable 4.1: Generation and Syndication of Learning Object Metadata, 2003. Available from `http://www.dcs.bbk.ac.uk/selene/reports/`.

[21] M. Rodríguez-Artacho. PALO Language Overview (Draft). Technical report, Jan 2002.

[22] G. Samaras, K. Karenos, and E. Christodoulou. SeLeNe Deliverable 3: A Grid Service Framework for Self e-Learning Networks, 2003. Available from `http://www.dcs.bbk.ac.uk/selene/reports/`.

[23] C. Smythe, F. Tansey, and R. Robson. IMS Learner Information Package Information Model Specification. Technical report, 2001.

[24] R. J. Sternberg. *Thinking Styles.* Cambridge University Press, 1999.

[25] W3C. *IsaViz: Visual Authoring Tool for RDF.* http://www.w3.org/2001/11/IsaViz/.

[26] R. Wheeldon and M. Levene. The Best Trail Algorithm for Assisted Navigation of Web Sites. In *Proceedings of the 1st Latin American Web Congress*, Santiago, Chile, November 2003.

Appendices

A Example RDF Learner Profile

```
<?xml version="1.0"?>
<rdf:RDF
    xmlns:ims-lip="http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/ims-lip#"
    xmlns:ims-rcd="http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/ims-rcd#"
    xmlns:papi="http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/papi#"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:SeLeNe="http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/learner#"
    xmlns:trails="http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/trails#">

 <rdf:Description rdf:about="#A_Learner">
   <rdf:type rdf:resource=
     "http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/learner#Learner"/>
   <ims-lip:qcl rdf:resource="#MyQualification1"/>
   <ims-lip:interest rdf:resource="#MyInterest1"/>
   <ims-lip:goal rdf:resource="#MyGoal1"/>
   <ims-lip:competency_exrefrecord rdf:resource="#MyCompetency1"/>
   <papi:learner_personal_info rdf:resource="#MyPersonalInfo"/>
   <papi:learner_preference_info rdf:resource="#MyPapiPreferences"/>
   <SeLeNe:history rdf:resource="#MyHistory"/>
   <SeLeNe:messages rdf:resource="#MyMessages"/>
   <SeLeNe:preferences rdf:resource="#MySelenePreferences"/>
 </rdf:Description>

 <rdf:Description rdf:about="#MyPapiPreferences">
   <rdf:type rdf:resource=
     "http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/papi#PostIncludeHidList"/>
   <rdf:_1>Partially Sighted Preferences</rdf:_1>
   <rdf:_2>Not deaf preferences</rdf:_2>
 </rdf:Description>

 <rdf:Description rdf:about="#MyPersonalInfo">
   <rdf:type rdf:resource=
     "http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/papi#PersonalInfoType"/>
   <papi:postal_address_list>10 Somewhere Street</papi:postal_address_list>
   <papi:name_list>A. Learner; Ms Learner; ...</papi:name_list>
   <papi:my_personal_identifier_list>
     SeLeNe#123
   </papi:my_personal_identifier_list>
```

```
  <papi:personal_bucket>
    Languages:English, Greek; Age:27;
  </papi:personal_bucket>
</rdf:Description>

<rdf:Description rdf:about="#MyGoal1">
  <rdf:type rdf:resource=
    "http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/ims-lip#GoalClass"/>
  <ims-lip:goal_priority>High</ims-lip:goal_priority>
  <ims-lip:goal_date rdf:datatype="http://www.w3.org/2001/XMLSchema#date">
   31-08-2003
  </ims-lip:goal_date>
  <ims-lip:goal_description rdf:resource="#MyLearningObjective"/>
  <ims-lip:goal_typename>Education</ims-lip:goal_typename>
</rdf:Description>

<rdf:Description rdf:about="#MyQualification1">
  <rdf:type rdf:resource=
    "http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/ims-lip#Qualification"/>
  <ims-lip:qcl_level>Undergraduate - BSc</ims-lip:qcl_level>
  <ims-lip:qcl_date>September 2001</ims-lip:qcl_date>
  <ims-lip:qcl_title>BSc Computer Science</ims-lip:qcl_title>
  <ims-lip:qcl_organisation>
    University Of London
  </ims-lip:qcl_organisation>
  <ims-lip:qcl_description>
     Included modules in Databases, XML and Neural Networks.
  </ims-lip:qcl_description>
</rdf:Description>

<rdf:Description rdf:about="#MyInterest1">
  <rdf:type rdf:resource=
    "http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/ims-lip#InterestClass"/>
  <ims-lip:interest_description>
    Music - playing the drums
  </ims-lip:interest_description>
  <ims-lip:interest_typename>Recreational</ims-lip:interest_typename>
</rdf:Description>

<rdf:Description rdf:about="#MyCompetency1">
  <rdf:type rdf:resource=
    "http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/ims-rcd#RCD"/>
  <ims-rcd:catalog>ACM-CCS</ims-rcd:catalog>
```

```
  <ims-rcd:entry>H.3.3</ims-rcd:entry>
  <ims-rcd:title>Understands the Rocchio Formula</ims-rcd:title>
  <ims-rcd:description>
    Can implement a relevance feedback system
  </ims-rcd:description>
</rdf:Description>

<rdf:Description rdf:about="#MyLearningObjective">
  <rdf:type rdf:resource=
    "http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/learner#LearningObjective"/>
  <SeLeNe:learning_objective_verb
   rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
    Describe
  </SeLeNe:learning_objective_verb>
  <SeLeNe:learning_objective_topic rdf:resource=
    "http://139.91.183.30:9090/RDF/VRP/Examples/ACM-CCS.rdfs#H.3.3.3"/>
  <SeLeNe:learning_objective_annotation
   rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
    Rocchio Formula
  </SeLeNe:learning_objective_annotation>
</rdf:Description>

<rdf:Description rdf:about="#MySelenePreferences">
  <rdf:type rdf:resource=
    "http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/learner#PreferenceCollection"/>
  <SeLeNe:learning_style_preference rdf:resource="#MyLearningStyle1"/>
  <SeLeNe:provider_preference>Donald Knuth</SeLeNe:provider_preference>
  <SeLeNe:provider_preference>O'Reilly</SeLeNe:provider_preference>
</rdf:Description>

<rdf:Description rdf:about="#MyLearningStyle1">
  <rdf:type rdf:resource=
    "http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/learner#LearningStyle"/>
  <SeLeNe:learning_style_taxonomy>
   Honey and Mumford
  </SeLeNe:learning_style_taxonomy>
  <SeLeNe:learning_style_descriptor>
    Activist
  </SeLeNe:learning_style_descriptor>
</rdf:Description>

<rdf:Description rdf:about="#MyHistory">
  <rdf:type rdf:resource=
```

```
      "http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/trails#EmergentTrail"/>
    <trails:trail_name>History</trails:trail_name>
    <trails:trail_annotation>
      LOs viewed by the user in the last day
    </trails:trail_annotation>
    <rdf:_1>LO#9876</rdf:_1>
    <rdf:_2>LO#4321</rdf:_2>
    <rdf:_3>LO#5432</rdf:_3>
  </rdf:Description>

  <rdf:Description rdf:about="#MyMessages">
    <rdf:type rdf:resource=
      "http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/learner#Notifications"/>
    <SeLeNe:new_LOs rdf:resource="#MyNewLOs"/>
    <SeLeNe:updated_LOs rdf:resource="#MyUpdatedLOs"/>
    <SeLeNe:new_users rdf:resource="#MyNewUsers"/>
  </rdf:Description>

  <rdf:Description rdf:about="#MyNewLOs">
    <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Bag"/>
    <rdf:_1 rdf:resource="LO#543"/>
  </rdf:Description>

  <rdf:Description rdf:about="#MyUpdatedLOs">
    <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Bag"/>
  </rdf:Description>

  <rdf:Description rdf:about="#MyNewUsers">
    <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Bag"/>
    <rdf:_1 rdf:resource="Learner#321"/>
  </rdf:Description>
</rdf:RDF>
```

B RDF Schema for PAPI Elements used in SeLeNe

```
<?xml version="1.0"?>
<rdf:RDF
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
   xml:base="http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/papi#" >

  <rdf:Description rdf:about="#PersonalInfoType">
```

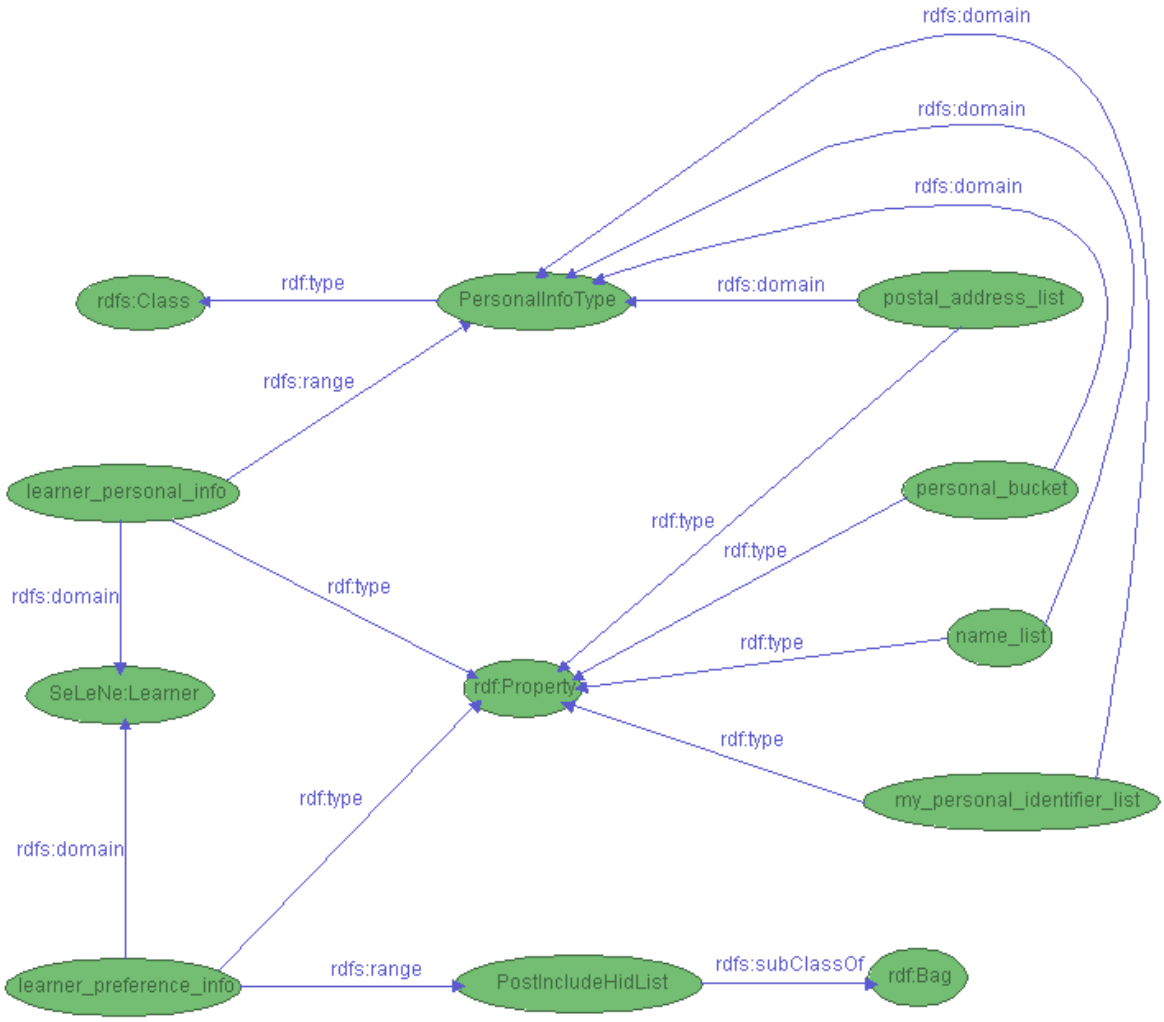

Figure 11: RDF schema for PAPI elements used in SeLeNe

```
  <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
</rdf:Description>

<rdf:Description rdf:about="#learner_personal_info">
  <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
  <rdfs:domain rdf:resource=
    "http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/learner#Learner"/>
  <rdfs:range rdf:resource="#PersonalInfoType"/>
</rdf:Description>

<rdf:Description rdf:about="#my_personal_identifier_list">
  <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
```

```
    <rdfs:domain rdf:resource="#PersonalInfoType"/>
  </rdf:Description>

  <rdf:Description rdf:about="#name_list">
    <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
    <rdfs:domain rdf:resource="#PersonalInfoType"/>
  </rdf:Description>

  <rdf:Description rdf:about="#personal_bucket">
    <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
    <rdfs:domain rdf:resource="#PersonalInfoType"/>
  </rdf:Description>

  <rdf:Description rdf:about="#postal_address_list">
    <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
    <rdfs:domain rdf:resource="#PersonalInfoType"/>
  </rdf:Description>

  <rdf:Description rdf:about="#PostIncludeHidList">
    <rdfs:subClassOf rdf:resource=
      "http://www.w3.org/1999/02/22-rdf-syntax-ns#Bag"/>
  </rdf:Description>

  <rdf:Description rdf:about="#learner_preference_info">
    <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
    <rdfs:domain rdf:resource=
      "http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/learner#Learner"/>
    <rdfs:range rdf:resource="#PostIncludeHidList"/>
  </rdf:Description>
</rdf:RDF>
```

C RDF Schema for IMS-LIP Elements used in SeLeNe

```
<?xml version="1.0"?>
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#" >

  <rdf:Description rdf:about="#competency_exrefrecord">
    <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
```

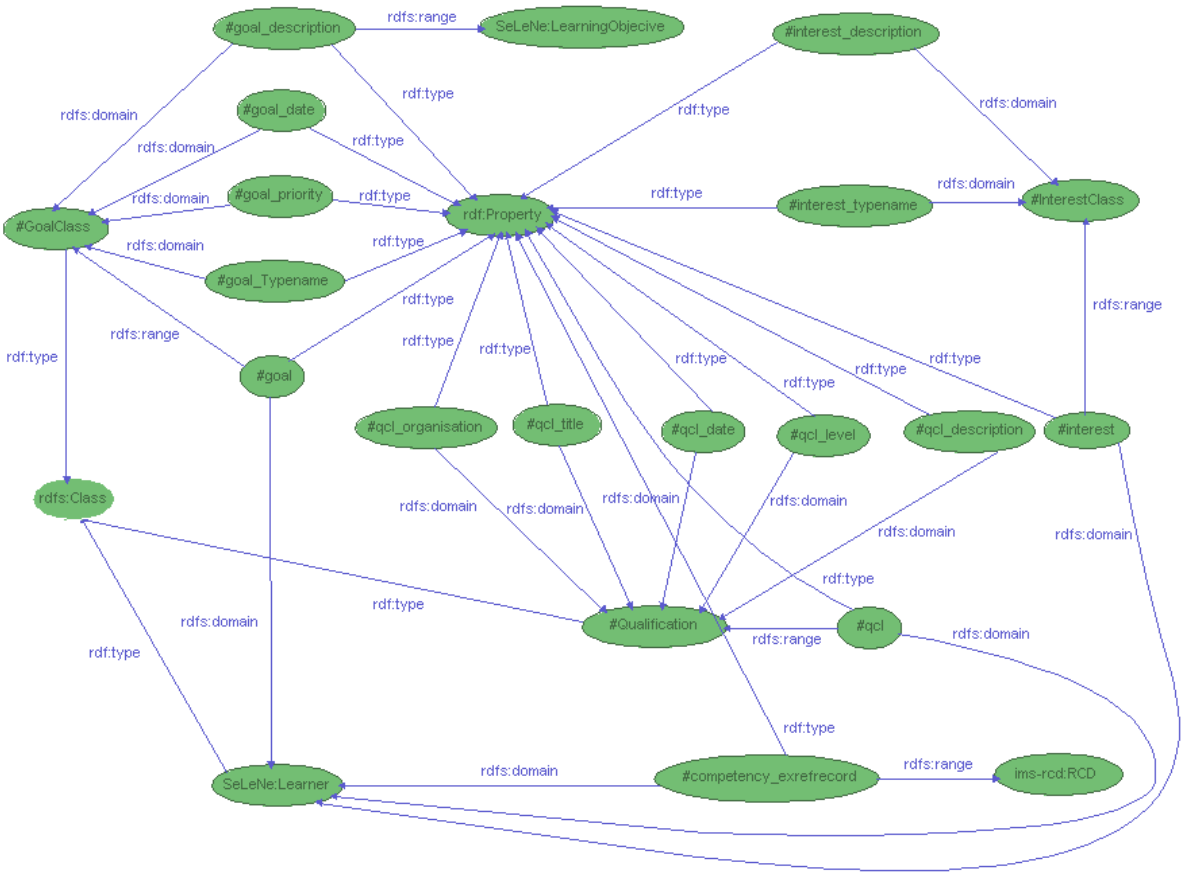

Figure 12: RDF schema for IMS-LIP elements used in SeLeNe

```
  <rdfs:domain rdf:resource=
    "http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/learner#Learner"/>
  <rdfs:range rdf:resource=
    "http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/ims-rcd#RCD"/>
</rdf:Description>

<rdf:Description rdf:about="#GoalClass">
  <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
</rdf:Description>

<rdf:Description rdf:about="#goal">
  <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
  <rdfs:domain rdf:resource=
    "http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/learner#Learner"/>
  <rdfs:range rdf:resource="#GoalClass"/>
</rdf:Description>
```

```
<rdf:Description rdf:about="#goal_date">
  <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
  <rdfs:domain rdf:resource="#GoalClass"/>
</rdf:Description>

<rdf:Description rdf:about="#goal_description">
  <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
  <rdfs:domain rdf:resource="#GoalClass"/>
  <rdfs:range rdf:resource=
    "http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/learner#LearningObjective"/>
</rdf:Description>

 <rdf:Description rdf:about="#goal_priority">
  <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
  <rdfs:domain rdf:resource="#GoalClass"/>
</rdf:Description>

<rdf:Description rdf:about="#goal_typename">
  <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
  <rdfs:domain rdf:resource="#GoalClass"/>
</rdf:Description>

<rdf:Description rdf:about="#InterestClass">
  <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
</rdf:Description>

<rdf:Description rdf:about="#interest">
  <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
  <rdfs:domain rdf:resource=
    "http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/learner#Learner"/>
  <rdfs:range rdf:resource="#InterestClass"/>
</rdf:Description>

<rdf:Description rdf:about="#interest_typename">
  <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
  <rdfs:domain rdf:resource="#InterestClass"/>
</rdf:Description>

<rdf:Description rdf:about="#interest_description">
  <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
  <rdfs:domain rdf:resource="#InterestClass"/>
</rdf:Description>
```

```
 <rdf:Description rdf:about="#Qualification">
   <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
 </rdf:Description>

 <rdf:Description rdf:about="#qcl">
   <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
   <rdfs:domain rdf:resource=
     "http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/learner#Learner"/>
   <rdfs:range rdf:resource="#Qualification"/>
 </rdf:Description>

 <rdf:Description rdf:about="#qcl_date">
   <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
   <rdfs:domain rdf:resource="#Qualification"/>
 </rdf:Description>

 <rdf:Description rdf:about="#qcl_description">
   <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
   <rdfs:domain rdf:resource="#Qualification"/>
 </rdf:Description>

 <rdf:Description rdf:about="#qcl_level">
   <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
   <rdfs:domain rdf:resource="#Qualification"/>
 </rdf:Description>

 <rdf:Description rdf:about="#qcl_organisation">
   <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
   <rdfs:domain rdf:resource="#Qualification"/>
 </rdf:Description>

 <rdf:Description rdf:about="#qcl_title">
   <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
   <rdfs:domain rdf:resource="#Qualification"/>
 </rdf:Description>
</rdf:RDF>
```

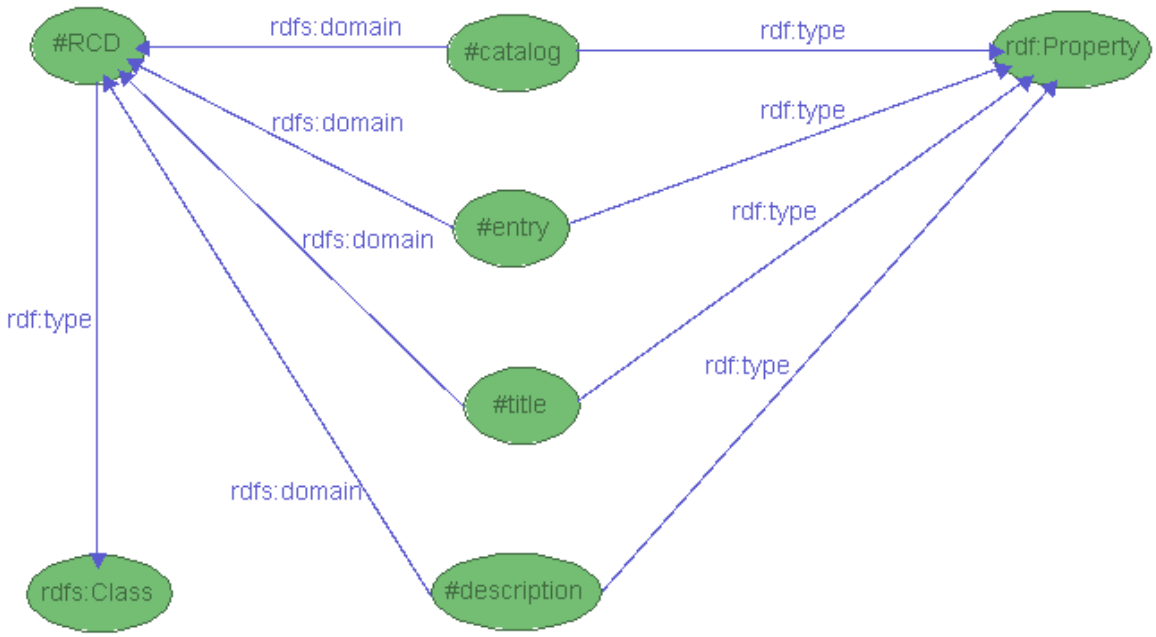

Figure 13: RDF schema for IMS-RCD elements used in SeLeNe

D RDF Schema for IMS-RCD Elements used in SeLeNe

```
<?xml version="1.0"?>
<rdf:RDF
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#" >

 <rdf:Description rdf:about="#RCD">
   <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
 </rdf:Description>

 <rdf:Description rdf:about="#entry">
   <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
   <rdfs:domain rdf:resource="#RCD"/>
 </rdf:Description>

 <rdf:Description rdf:about="#catalog">
   <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
   <rdfs:domain rdf:resource="#RCD"/>
 </rdf:Description>
```

```
 <rdf:Description rdf:about="#title">
   <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
   <rdfs:domain rdf:resource="#RCD"/>
 </rdf:Description>

 <rdf:Description rdf:about="#description">
   <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
   <rdfs:domain rdf:resource="#RCD"/>
 </rdf:Description>
</rdf:RDF>
```

E RDF Schema for SeLeNe-specific Elements of the User Profile

```
<?xml version="1.0"?>
<rdf:RDF
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
   xml:base="http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/learner#" >

 <rdf:Description rdf:about="#Learner">
   <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
 </rdf:Description>

 <rdf:Description rdf:about="#preferences">
   <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
   <rdfs:domain rdf:resource="#Learner"/>
   <rdfs:range rdf:resource="#PreferenceCollection"/>
 </rdf:Description>

 <rdf:Description rdf:about="#history">
   <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
   <rdfs:domain rdf:resource="#Learner"/>
   <rdfs:range rdf:resource=
     "http://www.dcs.bbk.ac.uk/~kevin/selene/rdf/trails#EmergentTrail"/>
 </rdf:Description>

 <rdf:Description rdf:about="#PreferenceCollection">
   <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
 </rdf:Description>
```

```
<rdf:Description rdf:about="#provider_preference">
  <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
  <rdfs:domain rdf:resource="#PreferenceCollection"/>
</rdf:Description>

<rdf:Description rdf:about="#LearningObjective">
  <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
</rdf:Description>

<rdf:Description rdf:about="#learning_objective_verb">
  <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
  <rdfs:domain rdf:resource="#LearningObjective"/>
</rdf:Description>

<rdf:Description rdf:about="#learning_objective_topic">
  <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
  <rdfs:domain rdf:resource="#LearningObjective"/>
</rdf:Description>

<rdf:Description rdf:about="#learning_objective_annotation">
  <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
  <rdfs:domain rdf:resource="#LearningObjective"/>
</rdf:Description>

<rdf:Description rdf:about="#LearningStyle">
  <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
</rdf:Description>

<rdf:Description rdf:about="#learning_style_preference">
  <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
  <rdfs:domain rdf:resource="#PreferenceCollection"/>
  <rdfs:range rdf:resource="#LearningStyle"/>
</rdf:Description>

<rdf:Description rdf:about="#learning_style_taxonomy">
  <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
  <rdfs:domain rdf:resource="#LearningStyle"/>
</rdf:Description>

<rdf:Description rdf:about="#learning_style_descriptor">
  <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
  <rdfs:domain rdf:resource="#LearningStyle"/>
</rdf:Description>
```

```
 <rdf:Description rdf:about="#Notifications">
   <rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-schema#Class"/>
 </rdf:Description>

 <rdf:Description rdf:about="#messages">
   <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
   <rdfs:domain rdf:resource="#Learner"/>
   <rdfs:range rdf:resource="#Notifications"/>
 </rdf:Description>

 <rdf:Description rdf:about="#new_LOs">
   <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
   <rdfs:domain rdf:resource="#Notifications"/>
   <rdfs:range rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Bag"/>
 </rdf:Description>

 <rdf:Description rdf:about="#updated_LOs">
   <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
   <rdfs:domain rdf:resource="#Notifications"/>
   <rdfs:range rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Bag"/>
 </rdf:Description>

 <rdf:Description rdf:about="#new_users">
   <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
   <rdfs:domain rdf:resource="#Notifications"/>
   <rdfs:range rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Bag"/>
 </rdf:Description>
</rdf:RDF>
```

F RDF Schema for Trails

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#" >

 <rdf:Description rdf:about="#Trail">
   <rdfs:subClassOf rdf:resource=
     "http://www.w3.org/1999/02/22-rdf-syntax-ns#Seq"/>
 </rdf:Description>
```

```
 <rdf:Description rdf:about="#AuthoredTrail">
   <rdfs:subClassOf rdf:resource="#Trail"/>
 </rdf:Description>

 <rdf:Description rdf:about="#DerivedTrail">
   <rdfs:subClassOf rdf:resource="#Trail"/>
 </rdf:Description>

 <rdf:Description rdf:about="#EmergentTrail">
   <rdfs:subClassOf rdf:resource="#Trail"/>
 </rdf:Description>

 <rdf:Description rdf:about="#trail_name">
   <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
   <rdfs:domain rdf:resource="#Trail"/>
 </rdf:Description>

 <rdf:Description rdf:about="#trail_annotation">
   <rdf:type rdf:resource="http://www.w3.org/1999/02/22-rdf-syntax-ns#Property"/>
   <rdfs:domain rdf:resource="#Trail"/>
 </rdf:Description>
</rdf:RDF>
```

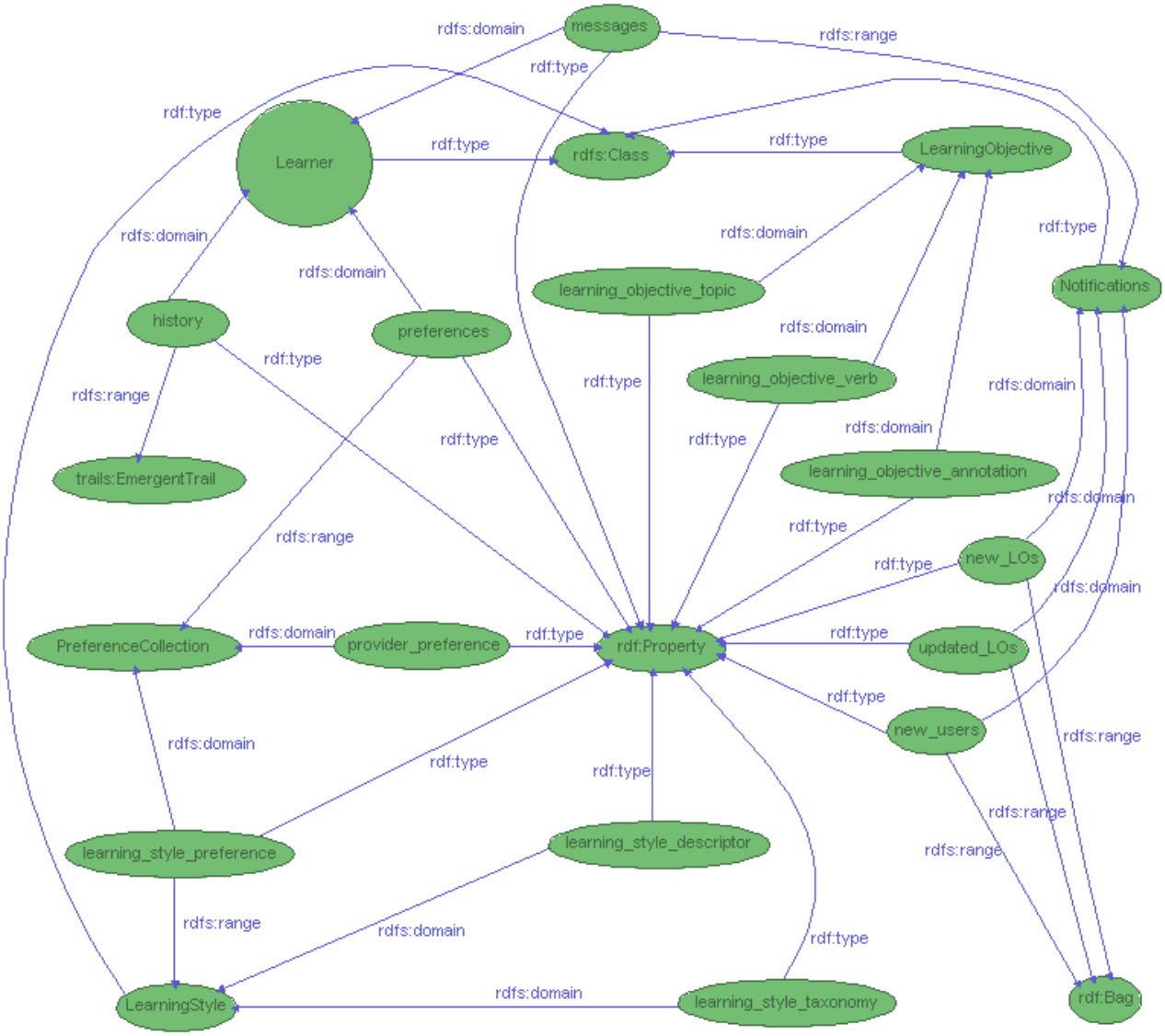

Figure 14: RDF schema for SeLeNe-specific elements of the user profile

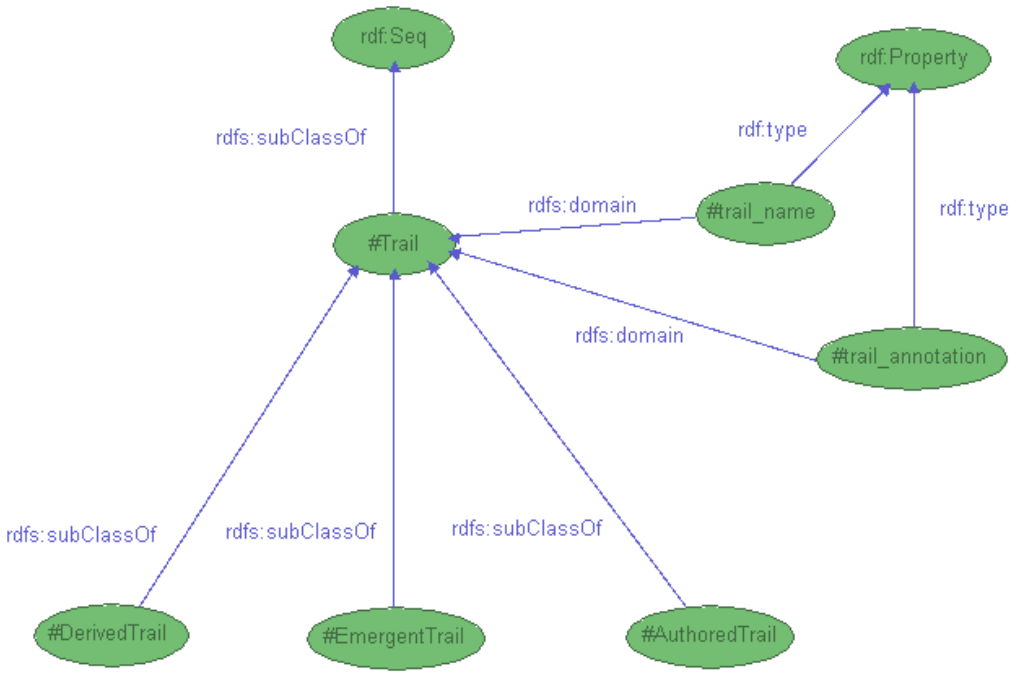

Figure 15: RDF schema for trails