IRML – INFORMATION RETRIEVAL MODELING LANGUAGE

João Ferreira[1], Alberto Silva[2], and José Delgado[3]
[1] *ISEL,* [2] *INESC-ID,* [2,3]*Instituto Superior Técnico*
[1]jferreira@deetc.isel.ipl.pt
[2]alberto.silva@acm.org
[3]Jose.Delgado@tagus.ist.utl.pt

Keywords: Modeling Language, Information Retrieval.

Abstract: We propose a specific language (created for the IR area) that provides a common notation and concepts for the design of IR systems. The language is based on UML extension mechanisms with specific stereotypes for IR. From this language (UML Profile) we define standard libraries of models and code templates that can be used in the development of IR systems. The main goal is to provide a novel approach that can guide the design of IR systems, using a common notation and concepts in a modular environment.

1 INTRODUCTION

Information Retrieval (IR) has mainly developed during the last four decades based on algorithms and methods. Personalization of Web retrieval applications is mandatory due to the follow requirements: (1) user specific information needs; (2) communication (e.g. mainly the rise of wireless devices); (3) geographic location. Personalization means new IR applications build based on previous ones or even in the optimization. There is also a need for a common language, that can provide a common notation, uniform concepts, a baseline to formulate problems in the IR community and also for the reuse of IR software. To fulfill this gap, we propose in this paper: (1) the IRML, a language based on UML extension mechanisms focused for the IR area; and a set of (2) IR-Models, derived from an IR-Language, which provides standard libraries for these kind of systems. These two points are part of an ambitious project concerned with automatic generation of IR systems through models and appropriate templates (figure 1).
The benefits of this process are significant, namely: (1) it facilitates the IR-System building process (e.g. increasing customization possibilities); (2) this standard, if well accepted, it promotes a collaborative environment within the IR Community, originating and supporting faster development. This approach also provides tools to facilitate changes in IR systems and the simultaneous development of IR systems by different teams. Each team starts to create models and templates for code generation based on a common IRML language and users or researchers define IR systems based on models, later transformed to specific programming languages and or platforms.

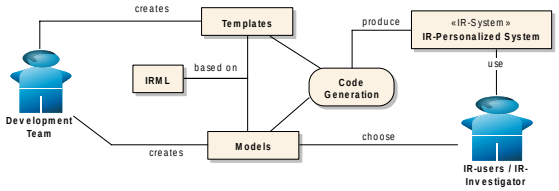

Figure 1: New approach to building IR systems.

2 IRML OVERVIEW

OMG proposes a new software development paradigm knows as Model Driven Architecture (MDA) [1]. MDA is strongly based on UML2 which is based on the following main principles: modularity, layer division and extensibility. This new version goes in line with the main objective of an IR-Language. This subject can be explored at [2, 3, 4]. Based on UML,

we will propose a meta-model for IR, based on a stereotype (Figures 2 and 3). To simplify the conception of IR-Systems we propose three views, (Figure 4). These views act like the different views in an architectural project of a house (or other kind of engineering projects) and simplifies the process by dividing the problem into smaller pieces. ViewPoints act as organizational "tools" for grouping together closely related ViewPoints [5,6]. The views are chosen as a compromise between simplicity (more views) and complexity (fewer views). From the construction of several systems (more than 100) we checked that these three views simplify a lot the process of building IR systems and, at the same, time are enough for problem description. External relations of IR systems are described in IR-UseCaseView, the information of an IR system is described in IR-InformationView and the manipulation of the information is described in the IR-ProcessView.

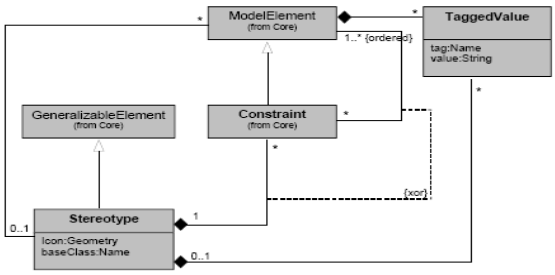

Figure 2: Extension mechanism of UML.

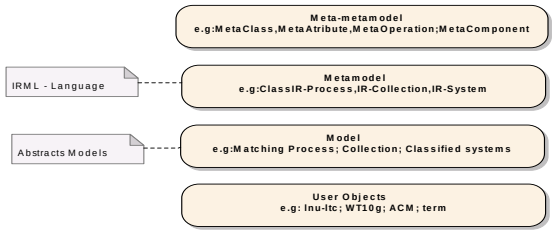

Figure 3: UML architecture in four layers.

IR-UseCaseView defines IR-Actors and their actions on the system. In this view, the external relations of the system and the main objectives are also defined.

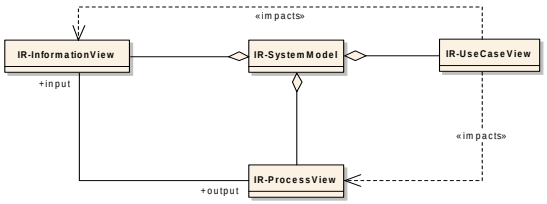

Figure 4: Main views proposed for an IR-System.

IR-InformationView defines the system input and output data. In this view the data structure and respective relationships are shown by using UML class diagrams.

IR-ProcessView defines attributes and the sequence of actions to transform input into output, according to the proposed objectives.

In Table 1, summarizes all the stereotypes included in the IRML language. Table 2 identifies the base class from which they derive. The metamodel of IRML is depicted in Figure 6 and the description of each stereotype is carried out in section 3.

Relations C - Creates; V - Validates; U - Use; E- Evaluation; O-Optimization	IR-Actor	IR-Autor	IR-User	IR-Authority	IR-Investigator	IR-Document	IR-Collection	IR-Process	IR-IndexProcess	IR-OptimizationProcess	IR-EstimationProcess	IR-MatchingProcess	IR-Index	IR-InformationNeeds	IR-Query	IR-UserProfile	IR-KnowladgeSpace	IR-Dictionary	IR-ClassifiedSystem	IR-Community	IR-System	IR-Results
IR-Actor																						
IR-Autor						C																
IR-User															C	C						U
IR-Authority							C								C			C	C	V		
IR-Investigator						U	C	C	C	C	C	C									U	E
IR-Document		C						U					U								U	
IR-Collection				C	U																U	
IR-Process					C																U	
IR-IndexProcess					C	U							C					U	U		U	
IR-OptimizationProcess					C								O		O	O					U	O
IR-EstimationProcess					C							O									U	
IR-MatchingProcess					C						O		U		U	U					U	C
IR-Index					C	U		C	O			U						O	O		U	
IR-InformationNeeds																						
IR-Query			C	C					O			U						O	O		U	
IR-UserProfile			C						O			U						O	O	O	U	
IR-KnowladgeSpace																						
IR-Dictionary				C				U					O		O	O					U	
IR-ClassifiedSystem				C				U					O		O	O					U	
IR-Community				V												O					U	
IR-System					U	U	U	U	U	U	U	U	U		U	U		U	U	U		C
IR-Results			E		E				O		C										C	

Table 1: IR stereotypes and their relationships.

IR stereotype relations: (C) Create identifies creator; (U) Use, is part of process (O) Optimization, process to improve results; (V) Validation, action of checking consisty of an automatic process; (E) Evaluation, checks results.

IR Profile	Base Class
IR-InformationNeeds	Class
IR-Query	Class
IR-UserProfile	Class
IR-KnowladgeSpace	Class
IR-Dictionary	Class
IR-ClassifiedSystem	Class
IR-Community	Class
IR-System	Package
IR-Service	Package
IR-Use Case View	Package
IR-Data View	Package
IR-Process View	Package
IR-Results	Use case, Class

Table 2: Base Classes of the IR stereotypes.

3 IR-SYSTEMS

IR systems are used to recover relevant information for the users. Users express their needs by a set of terms that system use to compare with document representative through appropriate method.

3.1 IR-UseCaseView

The main actors are represented in Figure 5 and described below:

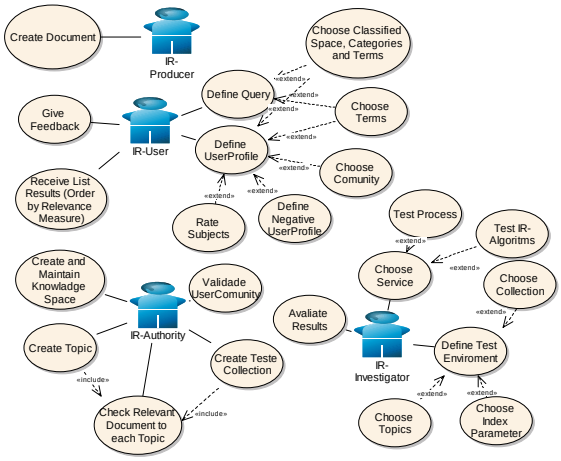

Figure 5: Use Case View of an IR-System.

IR-User, responsible for defining the information needs (query or user profile), receives system results (list of documents ordered by relevance) and gives feedback to the results received;

IR-Producer, creates documents to be stored;

IR-Authority, creates and maintains the knowledge space, creates test collections and defines topics with relevant documents from an associated test collection. User communities that are automatically identified by the IR-System are validated through this entity.

IR-Investigator, uses the testing platform (IR-System) to validate IR-Algorithms or IR-Process created. It chooses the test environment, the test collection and the topics and evaluates the results.

3.2 IR-Data View

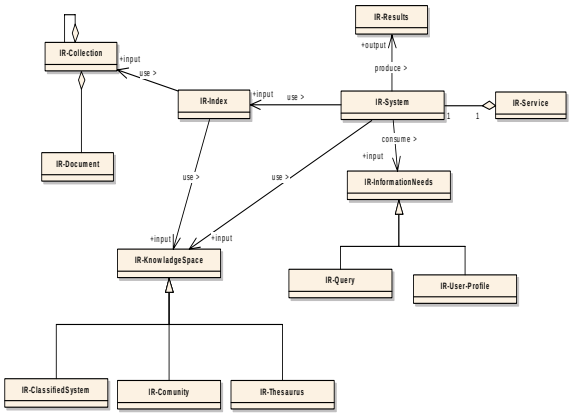

Figure 6: Data View of an IR profile.

IR-Document, information produced by the author, is supposed to be recovered by the user. The main associated problems are: available in different formats, Human language is subjective and depending on the context and in some cases authors give wrong information on purpose in order to be present on top positions of retrieval results from a search engine.

IR-Collection, group of documents stored. The largest collection is the Web.

IR-Index, is the output of IR-IndexProcess and the input of IR-MatchingProcess.

IR-InformationNeeds is an input of IR-MatchingProcess, constituted by a list of terms (with assigned weights) and divided in two main classes: (1)**IR-UserProfile**, stable user information needs; (2)**IR-Query**, that represents momentaneous information needs.

IR-KnowledgeSpace represents a specifically organized (by an authority) information space, divided into: (1) Classification Systems (generic or area specific); (2) User Community; (3) Thesaurus.

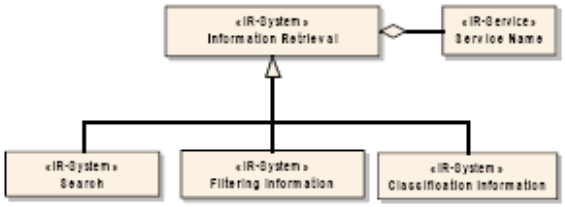

Figure 7: Main IR-Systems.

IR-System, integrated process resources that transforms input data into a list of documents organized by relevance. The system can be divided in three main systems: Search, Filtering Information systems which send new information periodically or inform the users about changes in documents previously received; Classification of information system which allocates documents to pre-defined categories in a classified system. The main retrieval systems are identified in Figure 7.

3.3 IR-Process View

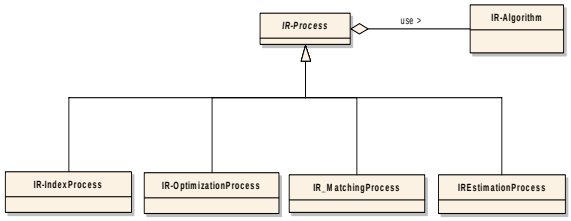

Figure 8: Process view of an IR profile.

IR-Process is a concept which involves data transformation by a pre-defined sequence of activities and is divided into four main processes. Additional processes such as translation are identified as IR-Processes.

IR-IndexProcess transforms collection documents into a small representative, following the process described in Figure 9.

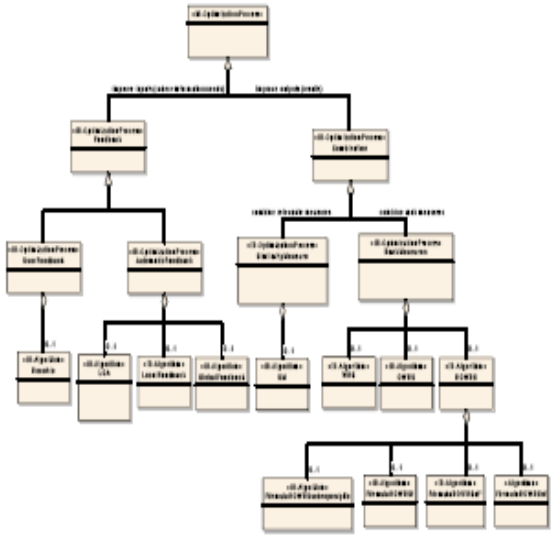

Figure 9: Index process of a documents' collection.

IR-MatchingProcess compares document representatives with users's information needs according to an IR-Algorithm. All methods listed in Figure 10 can use the same index (generic index) for all methods. For a detailed analysis see [7].

IR-OptimizationProcess, the main objective is to improve the output of the system and is divided in two main areas: (1) work on input data, mainly feedback process (automatic, manual or a mixture of both); (2) work on output data and combination (also called fusion) of results. The main approaches in Figure 11 are: Similarity Merge (combination of relevant measures) or Weighted Rank Sum (WRS), that uses rank-based scores (e.g. 1/rank) in place of document scores. Several forms of WRS can be explored. A detailed discussion can be found in [8].

IR-EstimationProcess, based on a test collection estimate parameters for: (1) classification algorithms; (2) language models; (3) other IR-algorithms. For a detailed analysis see [9].

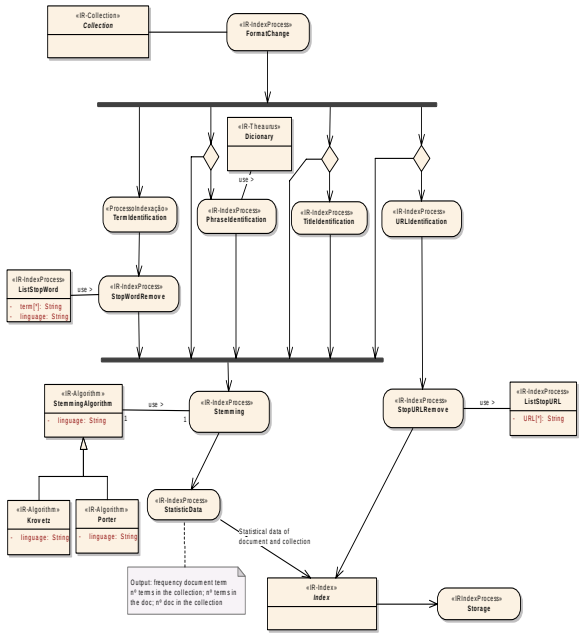

Figure 10: Package view of IR-Matching Process.

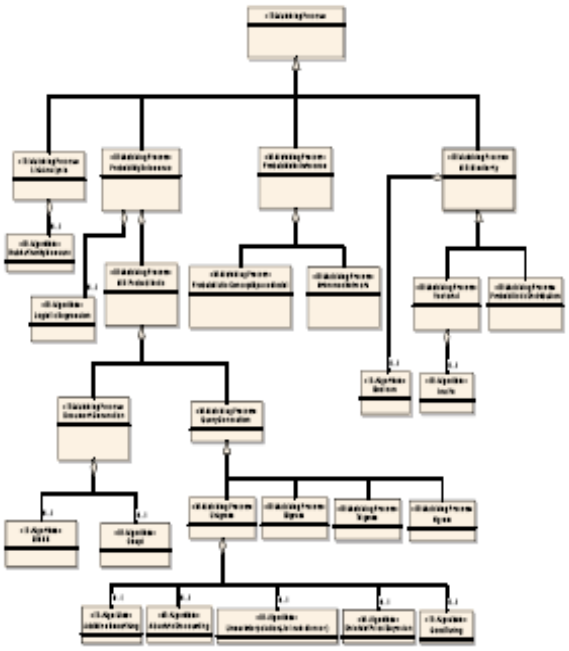

Figure 11: Description of an IR-OptimizationProcess.

4 CONCLUSION

This paper introduces the main issues of IRML language that guides the design of IR systems by providing a common notation and uniformization of concepts. According the proposed approach, the complexity of the design and even the build of this kind of systems is improved through the use of the different views proposed. Abstract models provide standard libraries that can be used in the design of the system as one of the starting points on the systems' construction. This idea, once accepted by the IR community, provides a common, collaborative and shared environment for the IR community. This is a starting point for the development of personalized IR-Systems (Figure 12), based on models and templates oriented to the automatic code generation. Naturally, this approach facilitates the development of IR applications.

Through the proposed approach, reuse of IR programming modules will be much easier, and personalization of an IR system will be faster because with this framework developers need to change or to create specific modules that will work on top of other common IR modules. Automatic code generation allows the creation of small and or customizable programming modules which can allow the construction of specific IR systems.

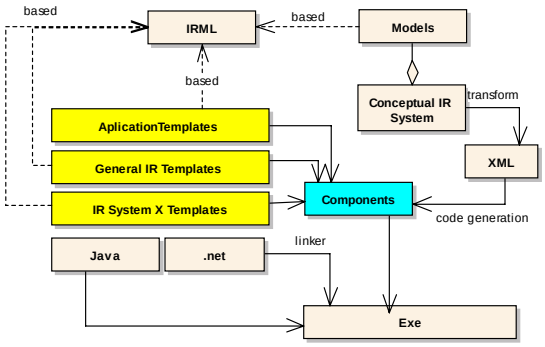

Figure 12: Top-vision of the model-based approach for the development of IR systems.

IRML language promotes a model-driven approach to the development of IR applications, which is a key factor for defining a novel generation of IR CASE tools for the construction of specific (personalized) IR

applications, supporting advanced features such as connection devices, multimedia and geographic location.

Obviously, these are very ambitious ideas, that can be detailed and focused in a medium turn research agenda and they are the application of XIS project [10] to information retrieval.

REFERENCES

[1]OMG. Model Driven Architecture. http://www.omg.org/mda/

[2] Booch, G.; Jacobson, I.; & Rumbaugh, J. (1999) The Unified Modeling Language. Reference Manual. Reading, MA: Addison-Wesley, 1999.

[3] Silva, A; Videira, C. (2005). UML - Metodologias e Ferramentas CASE (2nd edition), ed. Centro Atlântico (in Portuguese).

[4] OMG. "White Paper on the Profile mechanism", Version 1.0, OMG Document ad/99-04-07. OMG UML Working Group.

[5] Easterbrook, S.; Finkelstein A.; Kramer J. and Nuseibeh B. (1994); "Coordinating Distributed ViewPoints: The Anatomy of a Consistency Check"; Concurrent Engineering: Research and Applications, August 1994; CERA Institute, West Bloomfield, USA.

[6] Nuseibeh, B.; Kramer, J.; Finkelstein, A. (1994) A Framework for Expressing the Relationships Between. Multiple Views in Requirements Specification. IEEE Transactions on Software Engineering Volume 20, Issue 10, October 1994.

[7] Ferreira, João; Silva, Alberto; Delgado, José (2005). A modular platform applicable to all statistical retrieval models, Proceedings of the ITA05, 7-9 de September 2005, Wrexham, Wales.

[8] Ferreira, João; Silva, Alberto; Delgado, José (2004). How to Improve Retrieval effectiveness on the Web, Proceedings of the IDAS e-Society 2004, Avila (Spain) 16-19 July 2004.

[9] Ferreira J. A Methodology to Design Information Retrieval Systems. Ph.D. Thesis, Instituto Superior Tecnico, Lisbon, 2006 (in Portuguese).

[10] http://berlin.inesc-id.pt/projects/xis/