Enhanced Publication Management Systems

A systemic approach towards modern scientific communication

Alessia Bardi
Institute of Information Science and Technologies, Italian National Research Council (ISTI-CNR), Pisa, Italy
alessia.bardi@isti.cnr.it

Paolo Manghi
Institute of Information Science and Technologies, Italian National Research Council (ISTI-CNR), Pisa, Italy
paolo.manghi@isti.cnr.it

ABSTRACT
Enhanced Publication Information Systems (EPISs) are information systems devised for the management of enhanced publications (EP), i.e., digital publications enriched with (links to) other research outcomes such as data, processing workflows, software.

Today, EPISs are typically realised with a “from scratch” approach that entails non-negligible implementation and maintenance costs.

This work argues for a more systemic approach to narrow those costs and presents the notion of Enhanced Publication Management Systems, software frameworks that support the realisation of EPISs by providing developers with EP-oriented tools and functionalities.

Categories and Subject Descriptors
D.2.11 [Software Architectures]: Data abstraction—Domain-specific architectures; H.2.1 [Database Management]: Logical Design—Data Models; H.3.7 [Information Storage and Retrieval]: Digital Libraries

Keywords
Enhanced publication, Information systems, Scholarly communication

1. INTRODUCTION
Enhanced publications (EPs) are digital publications enriched with, and/or linked to, related research results, such as research data, processing workflows, software, and connections among them [3] [2].

Fig. 1 shows an example of EP composed of semantically linked parts that represent different kinds of outcome of a scientific research (scientific publication in PDF, datasets, processing workflow).

The types of entities, in terms of structure and semantics, that can form an EP are defined by an EP data model. An EP data model can be adopted by an Enhanced Publication Information System (EPIS), that is an information system devised for the management of EP's compliant to a specific EP data model. [2]

The realization and maintenance of EPISs typically entail non-negligible costs, as designers and developers have little or no technological support oriented to EPs. In fact, they will realize EP-oriented software addressing the peculiarities of the community to serve, by integrating technologies that are general purpose (e.g., databases, file stores) and Digital Library-oriented (e.g., repository software, cataloging systems). The resulting products are often not flexible enough to be adapted to the evolving requirements of the community they target and hardly re-usuble and configurable for different application domains with similar requirements.

A more systemic approach would decrease the cost of realization and maintenance of EPISs and support developers with EP management tools that hide the complexity of the underlying technologies.

Like Data Base Management Systems had been introduced to collect common functionalities used by different applications dealing with data, EPMSs could be introduced into the scholarly communication scenario, offering to developers an EP-oriented framework for the modern representation of scientific research outcomes.

Copyright is held by the author/owner(s).
WWW 2015 Companion, May 18–22, 2015, Florence, Italy.
ACM 978-1-4503-3473-0/15/05.
http://dx.doi.org/10.1145/2740908.2742026.

Figure 1: An example of enhanced publication

Figure 2: EPMS functional areas
2. ENHANCED PUBLICATION MANAGEMENT SYSTEMS

An EPMS is a framework that should provide tools to support the development of EPIS for any application domain. As such, an EPMS should not make any assumption about the structure and semantics of the EPs it will manage. Instead, an EPMS should provide tools for the definition of EP data models, including the selection of ontologies to use, and its functionalities should be configurable based on a given EP data model. By analysing the requirements of existing EPISs, we have identified a set of minimal requirements that an EPMS should satisfy:

1. Supporting the integration of heterogeneous content from dynamic data sources
2. Supporting the adoption of different storage back-ends
3. Enabling EP sharing via standard protocols (e.g., Linked Data, OAI-PMH/ORE, Search/Retrieve via URL (SRU), OpenSearch)
4. Supporting portability
5. Supporting the enrichment and curation of EPs
6. Enabling the definition of customized EP data models
7. Offering languages for EP definition, manipulation and access
8. Supporting the addition of new domain-specific functionalities

The high-level design of an EPMS has been designed so that each functional area addresses a specific class of requirements, as shown in Fig. 2.

In particular, in order to allow the customization of the EP data model, the EP Modelling area offers the EP meta-model (Fig. 3), which is a data model oriented to the design of EP data models [1]. EP is a Resource that Aggregates at least one narration (via the has text part relationship) and possibly other Parts. A narration represents a digital publication to enhance and can be of type File (i.e. a file locally hosted), Ref (i.e. a link to a remote file), or Structured Text (e.g. JATS XML). Parts are Resources that reflect the data model features described above: embedded parts (Embedded), structured-text parts (Structured Text), reference parts (Ref), executable parts (Executable), and generated parts (Generated). Resources can have a unique identifier and descriptive metadata. Resources can be linked with each other via relationships (Rel) (for example to represent versioning relationships between EPs). The attribute label, a free term or a term from an Ontology, expresses the semantics of a relationship.

Acknowledgements

This work is partially supported by the EC-funded project OpenAIREplus (FP7-INFRA-2011-2, Grant Agreement 2835 95).

This poster was also presented at the Force2015 Conference, Oxford, UK.

3. REFERENCES

