Repositório institucional do IST

Miguel Fernandes Coxo

Abstract

Institutional repositories allow the storage, management and dissemination of the intellectual product created by an institution and its community members. They provide a complementary method to the traditional system of scholarly communication, 3making it easier to demonstrate the scientific, social and financial value of an institution.

The potential benefits of a institutional repository go beyond the increase of the institution research profile. They increase authors' visibility, provide users with easier access to information and grant funders a wider dissemination of their research outputs.

In spite of the rapid pace at which organizations are establishing institutional repositories, and all the potential benefits they offer, studies have found that one of their biggest existing problems is the lack of contribution by the institution's community.

In this work we propose a solution to ease this problem and facilitate the ongoing maintenance of an institutional repository.

Keywords: Institutional repositories, Automatic Harvesting, Automatic author matching.

1. Introduction

Producing, publishing and managing knowledge has always been the core mission of universities and research institutions. In the traditional scholarly communication, publishing has usually been done through scholarly peer-reviewed journals and the management done by the institution's library and data centers. As more and more information is created in digital formats, institutions have turned their attention to how to better identify and manage important digital assets.

Under the traditional system of scholarly communication, much of the intellectual output and value of an institution's intellectual property is diffused through thousands of scholarly journals (Johnson, 2002). While academics, scholars and researchers publication in these journals reflect positively on the host institution, it becomes hard to demonstrate its scientific, social and financial value when the work is so widely dispersed.

Institutional repositories appear as a practical solution for this problem. An institutional repository(IR) allows the storage, management and dissemination of the intellectual product created by an institution and its community members. By doing so it can complement existing metrics for measuring institutional productivity and prestige. As institutional repositories are natively aligned with institutions' core mission, they provide many benefits to them and their communities. In addition to authors, who gain visibility, and users, who can more easily find information, the potential benefits of IR extend to the institutions, which increase their research profile, and funders, who see wider dissemination of research outputs (Hockx-Yu, 2006).

In spite of the rapid pace at which organizations are establishing institutional repositories, and all the potential benefits they offer, several studies have found that working institutional repositories lack contribution by the institutions' communities. In order to be successful and provide the promised benefits, the IR must be filled with work of enduring value that is searched and cited, otherwise it will just be a set of empty shelves (Foster & Gibbons, 2005).

2. Objective

The objective of this work is to conceive and propose a solution that can reduce the effort required for the deposit of contents and facilitate the ongoing maintenance of an institutional repository. We propose a solution that takes advantage of the already existing services and automatically, and proactively, gathers published work and stores its metadata and full-text content (when allowed) in the IR. We want to test the hypotheses that with this solution the community members will be motivated to continuously deposit their publications, by reducing the required effort for the deposit. This will not only make the researchers life easier, but will also empower the Institution by allowing it to have vision and control over the real work produced.

3. Problem

Whatever the particular focus of an IR is, to be successful it must be filled with work of enduring value that is searched and cited (Foster & Gibbons, 2005). We have found that after building and successfully deploying an IR, one of the main problems most institutions face is getting the community to deposit contents.

In the article by Nancy F. Foster and Susan Gibbons (Foster & Gibbons, 2005) institutional repositories and their adoption problem are analyzed. They highlight the fact that *"Installing the software, however, is just the first step towards a successful IR. Without content, an IR is just a set of empty shelves."*. They also state that *"...in spite of the rapid pace at which organizations are establishing IRs, the quantity of content deposited into them remains quite modest."*, affirmation supported by the April 2004 survey of 45 IRs by Mark Ware (Ware, 2004) that found the average number of documents in a IR to be very low (only 1,250 per repository, with a median of 290).

The 2006 survey of the Association of Research Libraries among its members (Charles W. Bailey, et al., 2006) also revealed that two-tirds (63%) of IR implementers were sufficiently challenged by the task of content recruitment, labeling it "difficult".

In the same sense as the 2004 survey by Mark Ware, the Making Institutional Repositories a Collaborative Learning Environment (MIRACLE) census of institutional repositories in the United States (Rieh, Markey, St. Jean, Yakel, & Kim, 2007b) revealed similar results. About 80% of planning and pilot testers(PPT) and 50% of implementers(IMP) reported that their IR contained fewer than 1000 digital documents. The census results are further discussed in (Rieh, Markey, St. Jean, Yakel, & Kim, 2007a) and report that the surveyed IMP contain an average of 3207 digital documents while PPT contain an average of 2313 digital documents in their IR. These results show an increase of documents, but still show a low IR population.

In 2007 Philip M. Davis and Matthew J. L. Connolly evaluated the Cornell University DSpace Installation (Davis & Connolly, 2007) and came to the conclusion that *"Cornell's DSpace is largely underpopulated and underused by its faculty. Many of its collections are empty, and most collections contain few items. Those collections that experience steady growth are collections in which the university has made an administrative investment; such are requiring deposits of theses and dissertations into DSpace. Cornell faculty have little knowledge of and little motivation to use DSpace."*.

The findings in the Cornell University bring atop the main reason we believe the community doesn't deposit content: lack of motivation. This is especially true for articles, which are made available at publishers' web-sites without requiring the researcher to invest time and effort in cataloging them (required in the process of self-archiving in repositories). Given the ease that researchers can access their work, by simply accessing the respective publisher's web-site, they don't feel motivated to deposit their work in the IR. Sometimes they don't even have a local copy of their final published work. For other types of content there is usually some kind of motivation or a mandate for the deposit. In the case of research tech reports for example, the researcher feels motivated to deposit in the IR, given that he wants to preserve the data and has no other place to do so. Another case are thesis, which a student is required to submit to the university storage site to obtain a master's or a PhD degree (although these are not always made public or are hard to find).

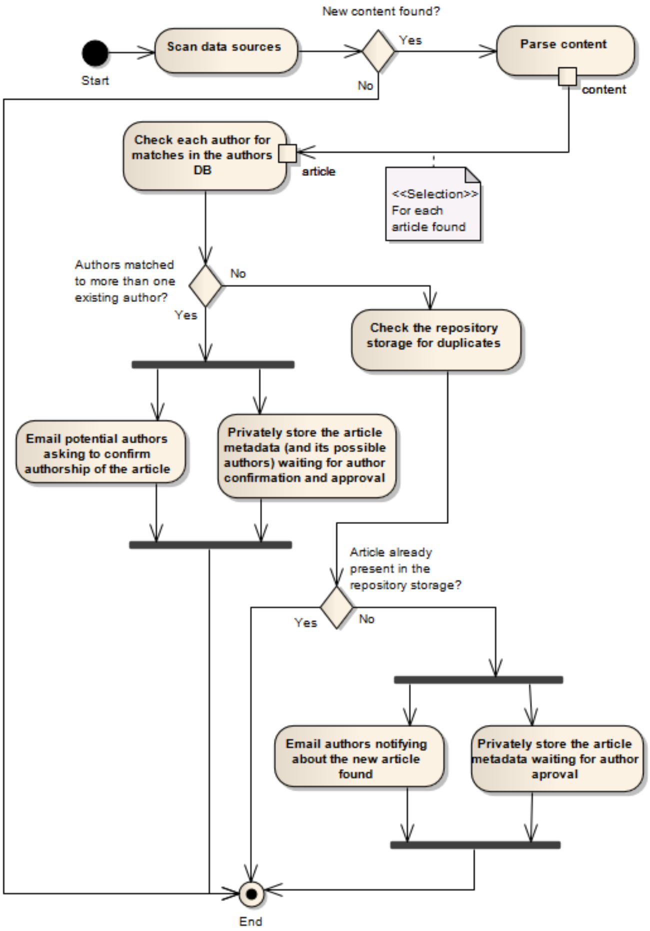
4. Proposed solution

In this work we conceive and propose a solution for an innovative institutional repository. To build an institutional repository solution that instead of being passive and rely on the community initiative for the maintenance of its contents it would be proactive in the gathering of data available on the Internet.

In this section we describe the design and implementation of the proposed solution, named Sotis framework.

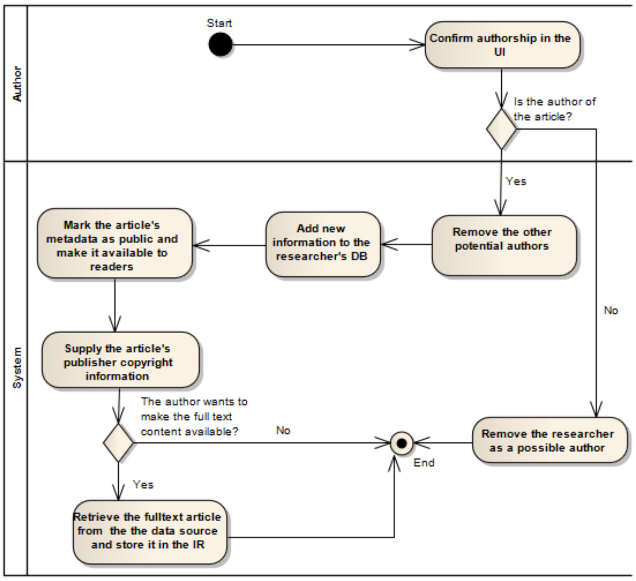
4.1. The automatic harvesting

The main innovation purposed in our solution consists in the automatic retrieval, conversion and ingestion of data (metadata and full text contents) available either in the web or supplied locally. Our solution is also built to allow authors to provide feedback to the system about the authorship of found bibliographic works. In other words, our solution will do its best to match bibliographic works (found in several data sources) to registered authors from the repository. When an absolute match cannot be made it will ask authors about their authorship, in which case the author will provide the system with a positive or negative answer. As consequence the system will learn using the answers of the authors. Figure 1 and Figure 2 represent the two activity diagrams that best describe the two most important workflows that allow our solution to accomplish the previous described objectives.



New content found?
Start
Scan data sources
Yes
No
Parse content
content
Check each author for matches in the authors DB
article
<<Selection>>
For each article found
Authors matched to more than one existing author?
No
Yes
Check the repository storage for duplicates
Email potential authors asking to confirm authorship of the article
Privately store the article metadata (and its possible authors) waiting for author confirmation and approval
Article already present in the repository storage?
Yes
No
Email authors notifying about the new article found
Privately store the article metadata waiting for author aproval
End

Figure 1 - Activity diagram of the ingestion stage



Author
System
Start
Confirm authorship in the UI
Is the author of the article?
Yes
No
Mark the article's metadata as public and make it available to readers
Add new information to the researcher's DB
Remove the other potential authors
Supply the article's publisher copyright information
The author wants to make the full text content available?
No
End
Remove the researcher as a possible author
Yes
Retrieve the fulltext article from the the data source and store it in the IR

Figure 2 - Activity diagram of the confirmation stage

4.2. Domain Model

Another key innovation in our framework, when compared to existing solutions, is that authors, institutions, and publishers are managed as entities rather then just strings within the metadata of a Bibliographic item. This is illustrated in Figure 3 that contains the sotis domain model.

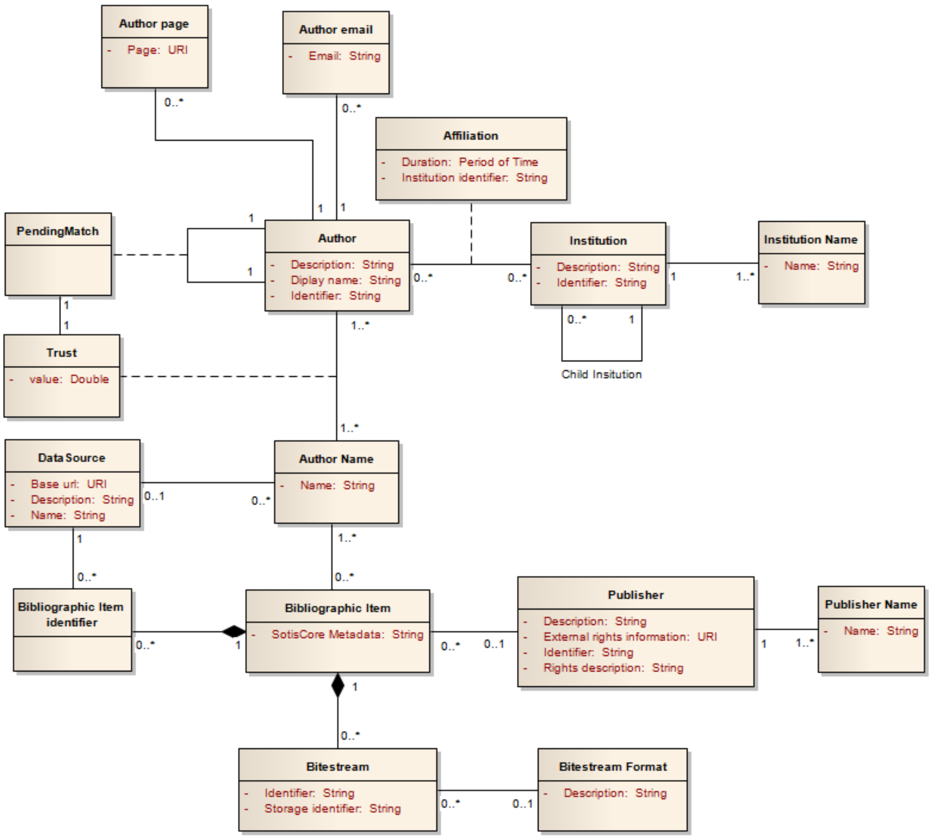


Figure 3 - Sotis Domain Model

Managing entities instead of just storing string allows us, among other things, to identify an author, even when there are several authors with the same name. It also allows us to identify a single author and its works despite the fact that he may have used different variation of his name in different bibliographic works.

4.3. Metadata support

The sotis framework is built so that it can be dynamically extended to ingest any kind of metadata schema that can be converted to the internal Sotis Core(SC) XML schema (The SC schema is a derivation of the qualified Dublin Core schema and based on the Libraries Working Group Application Profile (LAP)[1].

The framework currently uses a XML representation for the sotis core metadata element set. XML was chosen because it is easily legible to humans and because of the powerful tools available for the parsing and transformation of XML (including the XLST language[2]).

This representation follows the guidelines recommendations of the DCMI[3], but uses an attribute for the qualifier instead of defining new xml elements. We have found that this representation is simpler to parse and is more human friendly as it provides information about which element the qualifier is refining (without using a separate definition schema/ description document).

[1] http://dublincore.org/documents/library-application-profile/

[2] http://www.w3.org/TR/xslt

[3] http://dublincore.org/documents/dc-xml-guidelines/

4.4. Framework components

Our framework is composed of several modular components that are outlined in Figure 4. The design of the framework was based on the OAIS functional model (Allinson, 2006).

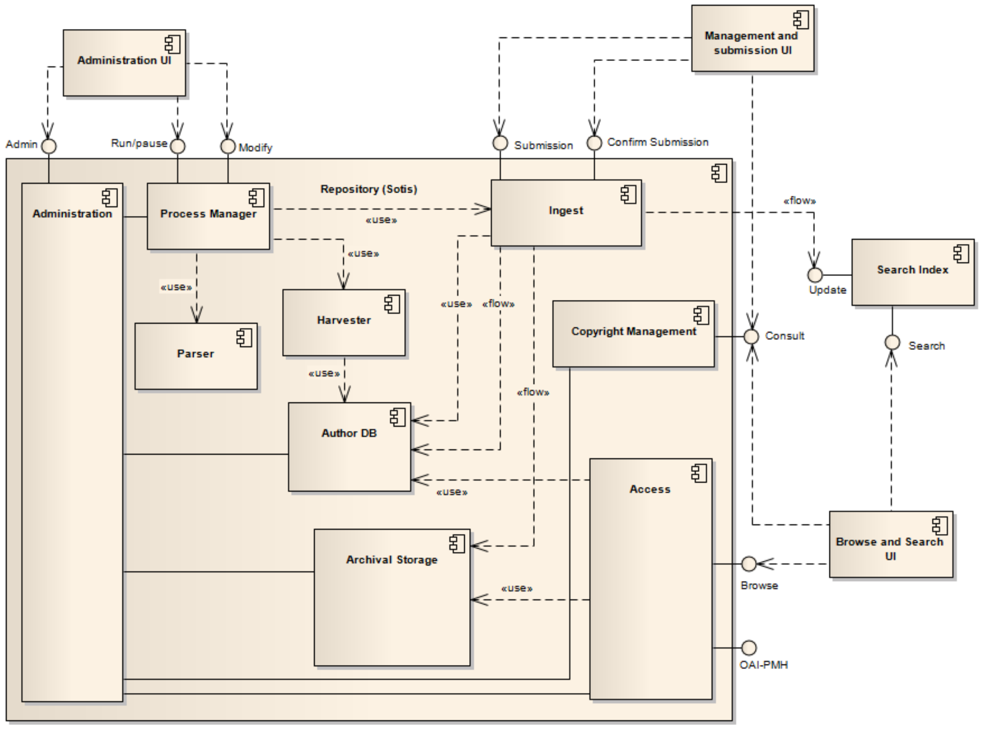


Figure 4 - Framework component diagram

These components are logically separated and each has its clear function. The Parser components is responsible for parsing metadata and convert it to the specified format, usually the xml sotis core metadata schema, but it is also used for example to convert the sotis core schema to the oai-dc[4] schema for our OAI-PMH[5] server. The Harvester component is responsible for harvesting data from several different data sources, from HTML pages to oai-dc xml data from OAI-PMH servers, it retrieves any files that can be referenced by an URI and downloaded using the HTTP protocol. The Ingest component is responsible for ingesting/processing files in the sotis core schema and using the "Process manager" component to run the matching processes against author names found it the files.

A key component in the framework is the "Process manager". This component is responsible for orchestrating processes and tasks that actually invoke/use the previously described components. Our framework was built to take advantage of these processes that are flexible and configurable at runtime.

Currently our Processes are implemented using a custom solution. They are described using a simple xml files and tasks are implemented using java classes loaded at runtime. These processes are also used for the author matching process. To simplify and at the same time empower our matching workflows we decided to take advantage of our own process implementations and used them for the matching process. As consequence the sotis framework supports flexible and reconfigurable author matching processes.

[4] http://www.openarchives.org/OAI/dc.xsd

[5] http://www.openarchives.org/OAI/openarchivesprotocol.html

4.5. Implemented prototype

A framework prototype was built based on the usage and integration of multiple free and open source technologies and programming languages. The core components previously described (Process Manager, Parser, Harvester, Ingest, etc...) were developed using the java programming language and the UI components were developed using the ruby programming language (together with HTML and javascript). The domain objects are stored inside a Mysql database. However the framework uses ORM frameworks that allow it to be DB agnostic. The deployment of the prototype is illustrated in Figure 5. The index component is implemented using the Solr[6] index server configured for out specific needs. Our core components were deployed behind SOAP web-services.

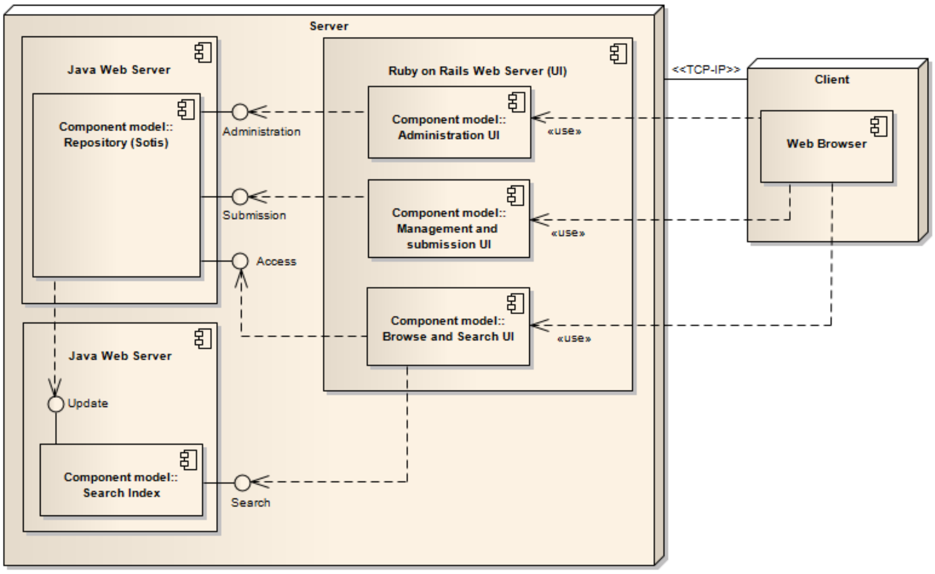
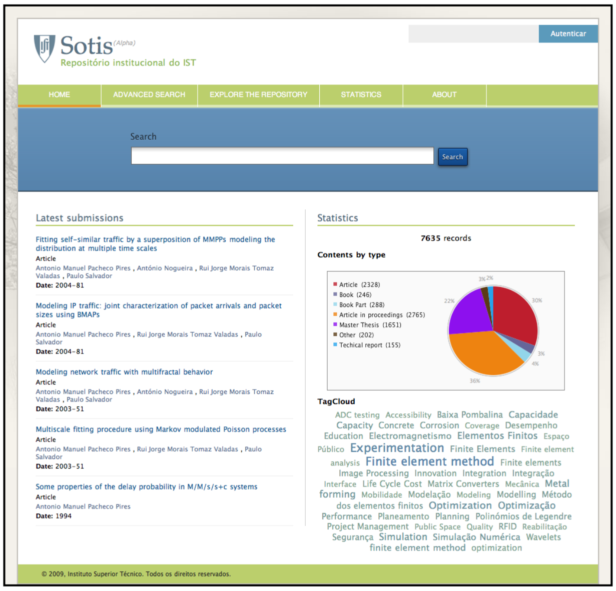


Figure 5 – sotis prototype deployment diagram

We also created a web user interface prototype that takes advantage or our entity management innovation. The forms used for the creation and edition of bibliographic works for example suggest to the user possible author completion values based on true entities and not their names. All the search and browse functionalities are also entity aware. A confirmed author is always displayed in a search/browse as a known entity in the system and not as the name the user used in a particular bibliographic work. The interface, together with the index server, also takes advantage of the author entities and connections to display co-author information. Figure 6 shows an illustrative screenshot of the sotis repository prototype web homepage.

[6] http://lucene.apache.org/solr/



Latest submissions

Fitting self-similar traffic by a superposition of MMPPs modeling the distribution at multiple time scales
Article
Antonio Manuel Pacheco Pires , António Nogueira , Rui Jorge Morais Tomaz Valadas , Paulo Salvador
Date: 2004-81

Modeling IP traffic: joint characterization of packet arrivals and packet sizes using BMAPs
Article
Antonio Manuel Pacheco Pires , Rui Jorge Morais Tomaz Valadas , Paulo Salvador
Date: 2004-81

Modeling network traffic with multifractal behavior
Article
Antonio Manuel Pacheco Pires , António Nogueira , Rui Jorge Morais Tomaz Valadas , Paulo Salvador
Date: 2003-51

Multiscale fitting procedure using Markov modulated Poisson processes
Article
Antonio Manuel Pacheco Pires , Rui Jorge Morais Tomaz Valadas , Paulo Salvador
Date: 2003-51

Some properties of the delay probability in M/M/s/s+c systems
Article
Antonio Manuel Pacheco Pires
Date: 1994

Statistics

7635 records

Contents by type

- Article (2328)
- Book (246)
- Book Part (288)
- Article in proceedings (2765)
- Master Thesis (1651)
- Other (202)
- Techical report (155)

TagCloud

ADC testing Accessibility Baixa Pombalina Capacidade Capacity Concrete Corrosion Coverage Desempenho Education Electromagnetismo Elementos Finitos Espaço Público Experimentation Finite Elements Finite element analysis Finite element method Finite elements Image Processing Innovation Integration Integração Interface Life Cycle Cost Matrix Converters Mecânica Metal forming Mobilidade Modelação Modeling Modelling Método dos elementos finitos Optimization Optimização Performance Planeamento Planning Polinómios de Legendre Project Management Public Space Quality RFID Reabilitação Segurança Simulation Simulação Numérica Wavelets finite element method optimization

© 2009, Instituto Superior Técnico. Todos os direitos reservados.

Figure 6 - Sotis prototype homepage

5. Evaluation

We evaluated our solution by setting up four different data sources and using a predefined default matching process that took advantage of information about author emails, author affiliations and author name similarity. The four data sources used were:

- Our institution's currently working campus activities support system (Fenix) publications. This data source's data was retrieved via a xml export. This data source was used to test the system training and the system flexibility;
- INESC's[7] publication DB. This data source's data was retrieved via a xml export. This data source was used to test the system training and the system flexibility;
- ACM portal digital library[8]. This data source's data was retrieved using our harvester component to harvest HTML pages and using the converter component they were converted to the sotis core schema. This data source was used to test the ability of the system to take advantage of author affiliation information;
- Spriger online digital library[9]. This data source's data was retrieved using our harvester component to harvest HTML pages and using the converter component they were converted to the sotis core schema. This data source was used to test the ability of the system to take advantage of author email information.

Our evaluation environment was previously trained with author information from both Fenix and INESC researchers' databases. From Fenix the system was pre-trained with 2805 authors. These had specified the author affiliations, with IST identifiers and other affiliated institutions, but had no valid emails. From INESC the system was pre-trained with 851 authors. These had their respective affiliations, some with their respective IST's identifiers and almost all had valid emails. The intersection of these two sets of author's results in 189 common authors.

[7] Instituto de Engenharia de Sistemas e Computadores Investigação e Desenvolvimento em Lisboa, http://www.inesc-id.pt/

[8] http://portal.acm.org

[9] http://www.springerlink.com/

From all the data sources the system retrieved and processed 29646 files, from these only 13738 bibliographic works where ingested (many where repeated or not from our institution). From these 13738, 10839 resulted in public works, meaning that at least one of the work's authors was successfully matched, and 2899 private works, meaning that at least one of the work's authors had a possible match to one previously confirmed author matched. The resulting distribution by work type can be seen in Figure 7.

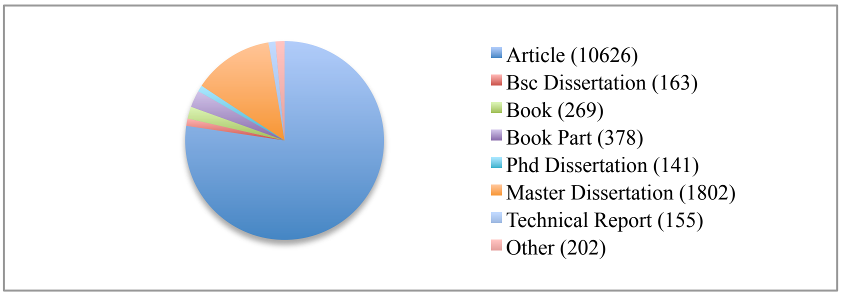


Figure 7 - Total distribution by type

We then chose ten authors from our Computer Science and Engineering Department of Instituto Súperior Técnico and evaluated their results by data source. The results from Fenix and INESC data sources where almost perfect, having only one erroneous authorship suggestion and 410 correct automatic matches. There were no incorrect automatic matches. This was to be expected as the system was previously trained with authors from these sources. Nonetheless, it shows that the sotis framework can be a good solution for information integration from different dispersed publication systems inside an institution or research units related to it (like INESC is related to Instituto Súperior Técnico).

The results from data ingested from ACM and Springer data sources were not as good, but still had a very positive result. In ACM we had a total of 43 correct automatic matches, 41 correct suggestions and only 4 incorrect matches. There were no incorrect automatic matches. From Springer we had a total of 68 correct automatic matches, 7 correct suggestions and 7 incorrect suggestions.

Overall there were no incorrect automatic matches, which is one of the important results of our work. By not burdening the authors with incorrect system matches we can be more successful in conquering their support and convincing them to continuously use the repository. On the other hand, the consequences of possibly missing some bibliographic works is not as drastic, because in the authors point of view the alternative is to have to deposit himself all the information that is already automatically provided to him by our solution.

6. Conclusion

In this work, we have analyzed and described institutional repositories, their objectives, their benefits and implementation issues. We have come to the conclusion that institutional repositories can be implemented without drastically altering the current functioning model, serving as complements to the traditional scholarly communication system.

Trough careful planning, taking in consideration the issues and decisions highlighted in this work, and using one of the many solutions available, we believe that an institution can create and deploy an IR without extensive technical development.

Today however, the development of an IR system is, perhaps, the least challenging task. In the long run, for the IR to be successful and provide the promised benefits its continuous usage by the institution's community must be secured.

With the purpose of minimizing the effort required for the deposit of contents and securing the continuous usage of the institutional repository, we designed a framework and implemented an institutional repository prototype that takes advantage of already existing services and automatically, and proactively, gathers published work and stores its metadata and full-text content (when allowed) in the IR.

For this to be possible we designed the framework to support dynamic and configurable ingestion channels by adding support for the creation of workflow processes. For the intended automatic matching of bibliographic works to their respective authors to be possible we also added support for the management of authors as entities and implemented flexible author matching processes. We also included in the framework an index component crucial for the search and retrieval of ingested contents.

We have evaluated our framework by testing it against four different data sources and demonstrated it to be viable for the harvesting, conversion and ingestion of data from completely different data sources. We also demonstrated that although the core objective of this work wasn't the implementation of an innovative name matching algorithm, the implemented matching process accomplishes good results having a low number of erroneous suggestions and no erroneous automatic matches (for the set of authors tested).

Concluding, in spite of the need of further testing and addition of new data sources, we believe the sotis framework to be a viable solution to lower an institutional repository deposit effort. By doing so, the repository community can be more motivated to continuously use the institutional repository.

7. References

Charles W. Bailey, J., Coombs, K., Emery, J., Mitchell, A., Morris, C., Simons, S., et al. (2006). SPEC Kit 292 Institutional Repositories. *Association of Research Libraries SPEC Kits.*

Davis, P. M., & Connolly, M. J. L. (2007). Evaluating the Reasons for Non-use of Cornell University's Installation of DSpace. *D-Lib Magazine, 13*(14).

Foster, N. F., & Gibbons, S. (2005). Understanding Faculty to Improve Content Recruitment for Institutional Repositories. *D-Lib Magazine.*

Hockx-Yu, H. (2006). Digital preservation in the context of institutional repositories. *Program: electronic library and information systems, 40*(3), 232-243. Preprint available at: http://eprints.rclis.org/7351/.

Johnson, R. K. (2002). Institutional Repositories: Partnering with Faculty to Enhance Scholarly Communication. *D-Lib Magazine, 8*(11).

Rieh, S. Y., Markey, K., St. Jean, B., Yakel, E., & Kim, J. (2007a). Census of Institutional Repositories in the U.S: A Comparison Across Institutions at Different Stages of IR Development. *D-Lib Magazine, 13*(11/12).

Rieh, S. Y., Markey, K., St. Jean, B., Yakel, E., & Kim, J. (2007b). *Census of Institutional Repositories in the United States: MIRACLE Project Research Findings.*: Council on Library and Information Resources.

Ware, M. (2004). Institutional repositories and scholarly publishing. *Learned Publishing, 17*(2).