Results of the second evaluation of matching tools

Christian Meilicke, José Luis Aguirre, Jérôme Euzenat, Ondrej Sváb-Zamazal, Ian Horrocks, Cassia Trojahn dos Santos

SEALS
Semantic Evaluation at Large Scale

FP7 – 238975

D12.6 Results of the second evaluation of matching tools

Coordinator: Christian Meilicke
With contributions from: Jose-Luis Aguirre-Cervantes, Jérôme Euzenat, Ondřej Šváb-Zamazal, Ernesto Jiménez-Ruiz, Ian Horrocks, Cássia Trojahn
Quality Controller: Ning Li
Quality Assurance Coordinator: Raúl García Castro

Document Identifier:	SEALS/2012/D12.6/V1.0
Class Deliverable:	SEALS EU-IST-2009-238975
Version:	version 1.0
Date:	June 14, 2012
State:	final
Distribution:	public

Executive Summary

In this deliverable, we report on the results of the second SEALS evaluation campaign for Ontology Matching Tools. Our campaign has been carried out in coordination with the OAEI (Ontology Alignment Evaluation Initiative) 2011.5 campaign. Contrary to the official title of the deliverable, it is already the third SEALS evaluation campaign for Ontology Matching. This is explained in §1, where we briefly review the history of integrated OAEI/SEALS campaigns.

In §2, we describe the design of the campaign with respect to data sets, evaluation criteria, methodology and tools that have been evaluated. As in the previous campaigns we have been using again the *Anatomy*, *Benchmark* and *Conference* test data set. In addition, we have also included two new data sets called *MultiFarm* and the *Large BioMed* data set. The *MultiFarm* data set is a new challenging data set for evaluating multilingual ontology matching. The *Large BioMed* data set comprises the largest ontologies ever used in an automated OAEI setting. Besides discussing the quality of the generated alignments in terms of precision and recall, we have also been interested in the runtimes of the evaluated systems and their scalability with respect to ontology size and available resources. Within the 2011.5 campaign, we followed the same methodology as in OAEI 2011. However, for the current campaign we have deployed and executed all systems with the help of the SEALS virtualization software. Moreover, we have stored the results of the campaign in the SEALS results repository. Thus, it will be possible to visualize the results later on with a generic results visualization component.

In §3 we describe the results of the campaign in detail. In particular, our evaluation results show that there are matching systems that

- can match large and even very large ontologies;
- scale well with respect to the number of available cores;
- are well suited to match ontologies from the biomedical domain;
- generate logically coherent results;
- are well suited for matching different versions of the same ontology;
- can match ontologies described in different languages;
- favor precision over recall or vice versa.

The detailed results presentation informs tool developers on strengths and weaknesses of their tool. Moreover, they help users in choosing a well-suited tool for a specific integration task. Finally, we highlight some of the lessons learned in §4 and conclude the deliverable with final remarks in §5.

Document Information

IST Project Number	FP7 – 238975	**Acronym**	SEALS
Full Title	Semantic Evaluation at Large Scale		
Project URL	`http://www.seals-project.eu/`		
Document URL			
EU Project Officer	Athina Zampara		

Deliverable	**Number**	12.6	**Title**	Results of the second evaluation of matching tools
Work Package	**Number**	12	**Title**	Matching Tools

Date of Delivery	**Contractual**	M37	**Actual**	20-06-12
Status	version 1.0		final ☒	
Nature	prototype ☐ report ☒ dissemination ☐			
Dissemination level	public ☒ consortium ☐			

Authors (Partner)	Jose-Luis Aguirre-Cervantes (INRIA), Jérôme Euzenat (INRIA), Ondřej Šváb-Zamazal (University of Economics, Prague, Czech Republic), Ernesto Jiménez-Ruiz (University of Oxford), Ian Horrocks (University of Oxford), Cássia Trojahn (INRIA)			
Resp. Author	**Name**	Christian Meilicke	**E-mail**	christian@informatik.uni-mannheim.de
	Partner	University of Mannheim	**Phone**	+49 621 181 2484

Abstract (for dissemination)	This deliverable reports on the results of the second SEALS evaluation campaign (for WP12 it is the third evaluation campaign), which has been carried out in coordination with the OAEI 2011.5 campaign. Opposed to OAEI 2010 and 2011 the full set of OAEI tracks has been executed with the help of SEALS technology. 19 systems have participated and five data sets have been used. Two of these data sets are new and have not been used in previous OAEI campaigns. In this deliverable we report on the data sets used in the campaign, the execution of the campaign, and we present and discuss the evaluation results.
Keywords	ontology matching, ontology alignment, evaluation, benchmarks

Version Log			
Issue Date	**Rev No.**	**Author**	**Change**
13/04/2012	1	Christian Meilicke	Set up overall structure
15/05/2012	2	Christian Meilicke	Filled Section 1 and 2
16/05/2012	3	Christian Meilicke	Added content to results sections
22/05/2012	4	Christian Meilicke	Lessons leraned and conclusions inserted
29/05/2012	5	Jose Luis Cervantes	Benchmark evaluation inserted
30/05/2012	6	Christian Meilicke	Final Revision
31/05/2012	7	Jérome Euzenat	Minor changes
11/06/2012	8	Christian Meilicke	Addressed comments of Quality Controller

Project Consortium Information

Participant's name	Partner	Contact
Universidad Politécnica de Madrid	UNIVERSIDAD POLITECNICA MADRID	Asunción Gómez-Pérez Email: asun@fi.upm.es
University of Sheffield	The University Of Sheffield.	Fabio Ciravegna Email: fabio@dcs.shef.ac.uk
Forschungszentrum Informatik an der Universität Karlsruhe	FZI	Rudi Studer Email: studer@fzi.de
University of Innsbruck	STI · INNSBRUCK	Daniel Winkler Email: daniel.winkler@sti2.at
Institut National de Recherche en Informatique et en Automatique	informatiques mathématiques inria	Jérôme Euzenat Email: Jerome.Euzenat@inrialpes.fr
University of Mannheim	UNIVERSITÄT MANNHEIM	Heiner Stuckenschmidt Email: heiner@informatik.uni-mannheim.de
University of Zurich	University of Zurich Department of Informatics DIS Dynamic and Distributed Information Systems	Abraham Bernstein Email: bernstein@ifi.uzh.ch
Open University	The Open University	Liliana Cabral Email: L.S.Cabral@open.ac.uk
Semantic Technology Institute International	STI · INTERNATIONAL	Alexander Wahler Email: alexander.wahler@sti2.org
University of Oxford	UNIVERSITY OF OXFORD	Ian Horrocks Email: ian.horrocks@comlab.oxford.ac.uk

Table of Contents

List of Figures

List of Tables

1. Introduction

The Ontology Alignment Evaluation Initiative[1] (OAEI) is an international initiative that organizes the evaluation of ontology matching systems [6]. The OAEI annual campaign provides the evaluation of matching systems on consensus test cases, which are organized by different groups of researchers. OAEI evaluations have been carried out since 2004. First OAEI campaigns have been executed with a very low level of automation. From 2010 OAEI is supported by SEALS and meanwhile three OAEI campaigns have been executed with SEALS technology with an increasing degree of automation.

OAEI 2010 A web based evaluation approach has been chosen. The evaluation workflow itself has been executed on the SEALS infrastructure, while the matching tools were still running on the machines of the developers. OAEI 2010 was executed under the umbrella of the first SEALS evaluation campaign. We have reported about this event in deliverable D12.3 [15].

OAEI 2011 Evaluation workflow and matching tools have been executed under full control of the campaign organizers. Thus, all results were completely reproducible for the first time in the history of OAEI. However, the SEALS virtualization infrastructure was not yet ready and could not be used for this purpose. OAEI 2011 did not fit into the official schedule for the two SEALS campaigns that have originally been planned. Thus, we reported only briefly on our second campaign in deliverable D12.5 v2.0-beta [11].

OAEI 2011.5 Nearly the complete campaign has been executed on top of the SEALS virtualization infrastructure. Moreover, matching tools have been retrieved from the SEALS tools repository; test data has been accessed from the SEALS test repository; for a subset of evaluations the SEALS results repository has been used for storing the results.

One has to notice that the OAEI campaigns have been collocated with the ISWC Ontology Matching workshop for several years. Thus, the execution and evaluation phase of OAEI happens by default each year in autumn. Since OAEI campaigns are widely accepted in the matcher community, we tried to combine the idea of a more continuous OAEI evaluation with the second official SEALS evaluation campaign, which is actually the third evaluation campaign conducted by WP12.

In this deliverable, we report on the results of the third OAEI/SEALS integrated campaign, that has been introduced to the matcher community as OAEI 2011.5. The remainder of the deliverable is organized as follows. We briefly review the evaluation design of the 2011.5 evaluation campaign (§2), presenting the evaluation data sets, criteria and metrics, and the list of participants. Then we present the results (§3) for each data set. Finally, we comment on the main lessons learned (§4) and conclude the paper (§5).

[1] http://oaei.ontologymatching.org

2. Evaluation Campaign

In the following we describe the test data used within the third SEALS campaign (referred to as OAEI 2011.5) and the motivation of its choice. We continue with a short description of the criteria and metrics used in the context of this campaign. Then we describe the overall methodology we followed to conduct the campaign. Finally, we give an overview on the participants of the campaign.

2.1 Test data

We have again been using the *Anatomy*, *Benchmark* and *Conference* test data set. On the one hand these data sets are well known to the organizers and have been used in many evaluations. On the other hand these data sets come with a high quality reference alignment that allows for computing compliance based measures, such as precision and recall.

In addition, we have also included two new data sets called *MultiFarm* and *Large BioMed* data set. While the evaluations related to *MultiFarm* have been executed by SEALS members, the generation of and the evaluation related to the *Large BioMed* data set has partially been conducted by non SEALS members using SEALS technology. In a similar way, we have been working together with the organizers of the Conference track. The fact that researchers external to the SEALS project are using SEALS technology is an important step towards the adoption of SEALS technology in the matching community.[1]

The OAEI terminology differs slightly from the SEALS terminology. An evaluation scenario in the context of a SEALS evaluation campaign is called a track in the context of OAEI. We use both notions in the same meaning in the following sections.

2.1.1 Benchmark test data

For this track, the focus of this campaign was on scalability, i.e., the ability of matchers to deal with data sets of increasing number of elements. To that extent, we considered four seed ontologies from different domains and with different sizes; as for previous campaigns, Benchmark test suites (or data sets) were generated from these seed ontologies. Table 2.1 summarizes the information about ontologies' sizes. From the ontologies shown in the table, `jerm` and `provenance` are completely new in OAEI campaigns; `biblio` is the traditional ontology used to generate the very first Benchmark data set, and `finance` was already considered in OAEI 2011.

2.1.2 Anatomy test data

The data set of this track has been used since 2007. For a detailed description we refer the reader to the OAEI 2007 [5] results paper. The ontologies of the anatomy track are the NCI Thesaurus describing the human anatomy, published by the National

[1] In particular, we would like to thank Ernesto Jimenez Ruiz (Large BioMed track) and Ondrej Svab-Zamazal (Conference track) for their effort and collaboration in the context of OAEI 2011.5.

Attribute	biblio	jerm	provenance	finance
classes+properties	97	250	431	633
instances	112	26	46	1113
entities	309	276	477	1746
triples	1332	1311	2366	21979

Table 2.1: Benchmark track ontologies' sizing attributes

Cancer Institute (NCI), and the Adult Mouse Anatomical Dictionary, which has been developed as part of the Mouse Gene Expression Database project. Both resources are part of the Open Biomedical Ontologies (OBO). The alignment between these ontologies has been created by experts of the domain [3] and, with support from other researchers, we have improved the alignment in the last years [1].

2.1.3 Conference test data

The Conference test data is also known as the OntoFarm data set [14]. Currently, there are sixteen ontologies within the OntoFarm data set. The ontologies differ in numbers of classes, properties, and in their DL expressivity. Overall, the ontologies have a high variance with respect to structure and size, which makes the matching process harder. During the last four years reference alignments have been created, extended and refined. These high quality reference alignments are the basis for the evaluation we conducted in OAEI 2011.5.

2.1.4 MultiFarm test data

A detailed description of the MultiFarm data set can be found in [10]. The data set has been created by twelve different experts from the field of ontology matching and multilingual ontology representation. The lead in the creation of the data set has been taken by WP 12. The data set is based on translating the OntoFarm data set, described above. The resulting data set contains multilingual test cases for the languages English, Chinese, Czech, Dutch, French, German, Portuguese, Russian and Spanish. Overall, it consist of $36 \times 49 = 1764$ testcases.

As argued in [10], the data set avoids typical problems of data sets for multilingual ontology matching. These data sets are often based on the usage of one (or several) ontologies that are translated into corresponding ontologies. The resulting matching task(s) is (are) about matching the original ontology against the translated ontology. MultiFarm uses a different principle. Two different ontologies, for which a reference alignment from OntoFarm is known, are translated. Via exploiting both the translations and the reference alignment, a new non-trivial matching task is generated.

2.1.5 Large Biomedical test data

This data set is based on the Foundational Model of Anatomy (FMA), SNOMED CT and the National Cancer Institute Thesaurus (NCI). These ontologies are very important and very large ontologies from the domain of biomedical research. For that reason the track offers three testcases that use fractions of increasing size from the ontologies NCI.

small module This data set consists of two fragments/modules of FMA and NCI, which represent their respective overlappings. The FMA module contains 3,696 concepts (5% of FMA), while the NCI module contains 6,488 concepts (10% of NCI).

extended module This data set consists of two fragments/modules of FMA and NCI, which represent their respective extended overlappings. The FMA (extended) module contains 28,861 concepts (37% of FMA), while the NCI (extended) module contains 25,591 concepts (38% of NCI).

whole This data set consists of the whole FMA and NCI ontologies, which consist of 78,989 and 66,724 concepts, respectively.

The silver standard that is used as reference alignment is based on an original UML mapping set, extracted from the MRCONSO.RRF file[2] of the UMLS [2] 2009AA distribution files (see [8] for details). In [8] and [7] two refinements of the UMLS mappings were presented that do not lead to inconsistencies. For OAEI 2011.5 the refined subset proposed in [7] was used.

2.2 Evaluation criteria and metrics

All of the five test data sets feature reference alignments. Thus, we can evaluate for all test sets the compliance of matcher alignments with respect to the reference alignments. In the case of *Conference*, where the reference alignment is available only for a subset of test cases, compliance is measured over this subset. The most relevant measures are precision (true positive/retrieved), recall (true positive/expected) and f–measure (aggregation of precision and recall). In case of the Large Biomedical test data set the silver standard, which has been created as described in [7], was used for computing precision and recall.

In OAEI 2011.5, we focus additionally on runtimes and, moreover, on scalability issues. By using the SEALS virtualization infrastructure, we could for the first time run the systems in different settings. In particular, we decided to check whether the number of available cores has an impact on the runtimes. Thus, we executed systems in a 1-core, 2-core and 4-core environment, where all other parameters (e.g. available RAM) have been fixed. These scalability test have conducted for the anatomy data set and in a similar way for the Large BioMed data set.

In the context of the Benchmark data set it has been analyzed whether systems scale with respect to an increasing size of the ontologies to be matched. Scalability has been addressed from two aspects: `compliance` where as usual we compute precision, recall and F-measure; and `runtime` where we report on measured runtimes for tools execution. For that purpose, test suites based on different reference ontologies have

[2] A description of this format is available at http://www.ncbi.nlm.nih.gov/books/NBK9685/

been created with the test data generator; for each seed ontology, three data sets of 94 tests each one were used for measuring compliance, and just one data set of 15 tests were used for measuring runtime. The choices of subsets of different size in the Large BioMed track was motivated by similar considerations.

For OAEI 2011.5, we omitted to measure the degree of incoherence for the Conference data set. This was based on the decision to create and run the MultiFarm evaluation, which took more time than expected. However, an evaluation concerned with alignment coherence has, instead of that, been conducted for the Large BioMed track.

2.3 General methodology

The process of the complete evaluation campaign can be divided into the following three phases. Note that in D12.3 [15] we divided the evaluation campaign into four phases. However, due to the higher degree of automation, the preparatory phase involves now also the preliminary testing of tools.

Preparatory phase (starting in January 2012) Ontologies and alignments are provided to participants, who have the opportunity to send observations, bug corrections, remarks and other test cases. Participants ensure that their systems can load the ontologies to be aligned and generate the alignment in the correct format (the Alignment API format [4]). In addition, participants have to ensure that their system correctly implements the SEALS interface and that their system can execute the complete evaluation runs with the use of a delivered client software.

Execution phase (between March 18th 2012 and up to now) The OAEI organizers execute the matching systems for each of the tracks. In doing so, we make use of the SEALS virtualization infrastructure and store the raw results that have been generated. In case that systems break while executing the first runs, we inform the respective tool developers. In case that quick fixes can be delivered, we execute these updated tool version. This approach has in particular been applied for the Benchmark track.

Evaluation phase (starting in April 2012) Raw results are analyzed and interpretations are derived from these results. Reports are written, published and the matcher community is informed on the results. We also made the results of our evaluation campaign available in the web via http://oaei.ontologymatching.org/2011.5/results/index.html.

An important ingredient for the success of OAEI 2011.5 is the use of the SEALS client for evaluating ontology matching systems. It is a java based command line tool that is given to developers of matching systems and, in particular, to the potential OAEI participants. Once a matching tool is wrapped against this client, the tool can be locally evaluated against all of the data sets described above (with the exception of some blind test from MultiFarm and Benchmark). Moreover, the local client executes

System	2011	between	2011.5	State, University
AgrMaker	√			US, University of Illinois at Chicago
Aroma	√			France, INRIA Grenoble Rhône-Alpes
AUTOMSv2			√	Finland, VTT Technical Research Centre
CIDER	√			Spain, Universidad Politécnica de Madrid
CODI	√	√	√	Germany, Universität Mannheim
CSA	√			Vietnam, University of Ho Chi Minh City
GOMMA			√	Germany, Universität Leipzig
Hertuda			√	Germany, Technische Universität Darmstadt
LDOA	√		*	Tunisia, Tunis-El Manar University
Lily	√			China, Southeast University
LogMap	√	√	√	UK, University of Oxford
LogMapLt		√		UK, University of Oxford
MaasMtch	√		√	Netherlands, Maastricht University
MapEVO	√		√	Germany, FZI Forschungszentrum Informatik
MapPSO	√		√	Germany, FZI Forschungszentrum Informatik
MapSSS	√	√		US, Wright State University
Optima	√			US, University of Georgia
WeSeEMtch			√	Germany, Technische Universität Darmstadt
YAM++	√		√	France, LIRMM

Table 2.2: Field of participants

exactly the same evaluation workflow that we execute in the final evaluation on top of the SEALS platform. Thus, the tool developer has the means to test whether his tool both correctly wraps the interface and can cope with the data in an appropriate way.

2.4 Participants

For OAEI 2011.5, we decided to evaluate all tools that have been uploaded to the SEALS tools repository via the portal for OAEI 2011 or at a later time. Thus, our evaluation also includes OAEI 2011. We just replaced the old versions of the evaluated systems by the current version (in case that an updated version was available). The matching systems we evaluated can be divided in three categories:

OAEI 2011 systems These systems have been participating in OAEI 2011. No updates are known to us or have been uploaded to the SEALS portal after the deadline of OAEI 2011.

updated systems These systems particpated in OAEI 2011 and they have been updated in the time between OAEI 2011 and OAEI 2011.5. However, these updates have been made independently from the fact that OAEI 2011.5 will be conducted in spring 2012.

OAEI 2011.5 systems These systems have been updated for OAEI 2011.5 or have been participating for the first time in an OAEI campaign.

Table 2.2 gives an overview on all of our participants. The developer of LogMap, for example, have uploaded a first version for OAEI 2011, between OAEI 2011 and

2011.5 they have uploaded an new version, and finally a version for OAEI 2011.5 has been uploaded. For GOMMA only a 2011.5 version has been uploaded. The system did not participate in OAEI 2011. Finally, we evaluated 19 systems. We have not included the OAEI 2011 systems Serimi and Zhishi.links in our evaluation because they are only designed for the instance matching task. Moreover, we have excluded OACAS and OMR because technical problems prevented us from executing them (we already had similar problems in OAEI 2011). Several days after the final submission deadline a new version of LDOA appeared (marked by *) that could not be included in the final evaluation.

3. Evaluation Results

In the following we present the most important results of the campaign. This chapter is divided in five sections, in which we describe the results for each of our five data sets. A more detailed description of these results can be found in the webpages linked from http://oaei.ontologymatching.org/2011.5/results/index.html. This is also the webpage that we used to inform participants on the results of the evaluation.

3.1 Benchmark results

From the 19 systems listed in Table 2.2, 14 systems participated in this track. The reason is that several systems participating for the first time required Jdk 1.7 to be run, and the systems that were left did not present a new version for this campaign and could not be ran in these conditions. The excluded systems are AgrMaker, CIDER, CSA, LDOA and Optima.

As we stated in §2.2, the focus of this campaign was on scalability, addressed from both compliance and runtime measurement. For both aspects, Benchmark evaluations have not been conducted on the SEALS hardware but on the machines of the track organizers. However, all evaluations of this track have used important parts of the SEALS technology, in particular the SEALS tools, test data and results repositories. For each aspect all systems have been executed in the same conditions whose specifications and results are given below.

3.1.1 Benchmark compliance results

Benchmark compliance tests have been executed on three two cores and 8GB RAM Debian virtual machines (VM) running continuously in parallel, except for finance test suite which required 10GB RAM for some systems. An exception of all this was CODI, which needs specific requirements/tools that track organizers were not able to install in their machines due to academic license problems. CODI was executed on a two core and 8GB RAM Ubuntu VM on SEALS hardware infrastructure. For tools other than CODI, the fact that the systems were running in parallel had no impact in compliance results.

Table 3.1 [1] presents the average precision, F-measure and recall for the 3 runs and for each test suite. Very insignificant variations have been found for a few systems between average measures for different runs of the same test suite. A few tools presented problems to process some test suites, maybe due to the fact that new ontologies were used to generate those test suites.

Systems on the table are first ordered according to the number of test suites for which the tools were able to finish at least one run, then by the best F-measure for biblio test suite which was the only one finished by all tools. The table shows that the rating of success decays as the size of the ontology increase. All systems were able to pass biblio test suite, 13 systems passed jerm test suite, eleven systems passed

[1] n/a: not able to run this test suite – u/r: uncompleted results, crashed or got stuck with some test

System	biblio			jerm			provenance			finance		
	Prec.	F-meas.	Rec.	Prec.	F-meas.	Rec.	Prec.	F-meas.	Rec.	Prec.	F-meas.	Rec.
MapSSS	0.99	0.86	0.75	0.98	0.76	0.63	0.98	0.75	0.61	0.99	0.83	0.71
Aroma	0.97	0.76	0.63	0.99	0.96	0.93	0.78	0.60	0.49	0.90	0.70	0.57
WeSeE	0.89	0.67	0.53	0.99	0.68	0.51	0.97	0.64	0.48	0.96	0.69	0.54
Hertuda	1.00	0.67	0.50	0.96	0.66	0.50	0.59	0.54	0.50	0.75	0.60	0.50
GOMMA	0.79	0.67	0.58	0.97	0.67	0.51	0.14	0.22	0.55	0.84	0.66	0.55
LogMapLt	0.7	0.58	0.50	0.98	0.67	0.51	0.99	0.66	0.50	0.90	0.66	0.52
MaasMtch	0.49	0.50	0.52	0.52	0.52	0.52	0.50	0.50	0.50	0.52	0.52	0.52
LogMap	0.69	0.48	0.37	1.00	0.66	0.50	1.00	0.66	0.49	0.96	0.60	0.43
MapEVO	0.43	0.37	0.33	0.06	0.04	0.03	0.02	0.01	0.01	0.04	0.02	0.01
MapPSO	0.58	0.20	0.12	0.06	0.05	0.05	0.08	0.07	0.05	0.28	0.16	0.11
Lily	0.95	0.75	0.62	0.93	0.71	0.58	0.92	0.68	0.54	u/r	u/r	u/r
YAM++	0.99	0.83	0.72	0.99	0.72	0.56	u/r	u/r	u/r	n/a	n/a	n/a
CODI	0.93	0.75	0.63	1.00	0.96	0.93	n/a	n/a	n/a	n/a	n/a	n/a
AUTOMSv2	0.97	0.69	0.54	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a

Table 3.1: Precision, F-measure and recall for Benchmark track

Matching System	biblio	jerm	provenance	finance	Average F-measure	Standard deviation
MapSSS	0.86	0.76	0.75	0.83	0.80	0.05
Aroma	0.76	0.96	0.6	0.7	0.76	0.15
WeSeE	0.67	0.68	0.64	0.69	0.67	0.02
LogMapLt	0.58	0.67	0.66	0.66	0.64	0.04
Hertuda	0.67	0.66	0.54	0.6	0.62	0.06
LogMap	0.48	0.66	0.66	0.6	0.60	0.08
GOMMA	0.67	0.67	0.22	0.66	0.56	0.22
MaasMtch	0.5	0.52	0.5	0.52	0.51	0.01
MapPSO	0.2	0.05	0.07	0.16	0.12	0.07
MapEVO	0.37	0.04	0.01	0.02	0.11	0.17

Table 3.2: Average F-measure and standard deviation for the Benchmark track.

provenance test suite, and only ten systems passed finance test suite. MapPSO had the singularity that it finished just one run for jerm, and two runs for finance.

Table 3.2 presents the F-measure average for the four test suites and the associated standard deviation. Only those systems that were able to finish at least one run for all test suites are included in the table; systems are ordered according to the best F-measure average.

Several remarks can be done for the results of the table. First, there is no system that behaves better than the other ones for all test suites. Second, based on the standard deviations, seven systems have what we could consider a stable behavior ($\sigma \leq 0.1$). Third, the results show that MapSSS seems to perform better than the other systems, with Aroma, WeSeE and LogMapLt as followers. For MapSSS this confirms what has already been observed in OAEI 2011. Finally, MapPSO and MapEVO, which belong to the same familiy, have very low results for the biblio benchmark, and were not able to generate meaningful alignments for the other benchmarks. For MapPSO, one reason could be that their algorithm uses an evaluation function suited only for some test ontologies which tells how good an alignment is.

Matching System	biblio	jerm	provenance	finance
LogMapLt	7	7	8	32
Hertuda	9	25	75	94
Aroma	9	10	27	63
GOMMA	10	10	24	61
LogMap	16	16	20	53
MapSSS	26	52	98	66494
MaasMtch	36	303	1284	2341
AUTOMSv2	63	n/a	n/a	n/a
YAM++	76	3428	u/r	n/a
MapPSO	140	968	u/r	u/r
MapEVO	158	194	751	64913
Lily	433	2645	10970	u/r
WeSeE	1025	2087	5315	7446

Table 3.3: Runtime measurement (in seconds) for Benchmark track.

3.1.2 Benchmark runtime results

For runtime, a 3GHz Xeon 5472 (4 cores) machine running Linux Fedora 8 with 8GB RAM was used. CODI was excluded from these tests as it needs specific requirements that we were not able to meet due to academic license problems. AUTOMSv2 was tested only with biblio test suite as it throws an exception with other test suites. YAM++ and Lily were not able to finish some test suites as they got stuck at one test for more than 12 hours.

Table 3.3 [2] presents the runtime measurement (in seconds) for data sets composed of 15 tests. Systems on the table are ordered according to the runtime measurement for the biblio test suite.

Figure 3.1 shows a semi-log graph for runtime measurement against test suite size in terms of classes and properties with the y-axis representing the runtime in a logarithmic scale.

LogMapLt is the fastest tool, followed in the same range by GOMMA, Aroma and LogMap; LogMapLt being a lightweight version of LogMap, the shape of their graphs is almost the same. GOMMA and Aroma exhibit very close behaviors. Hertuda had a good result for biblio test suite, but its response clearly degrades for the other test suites. The rest of the tools have bigger results for all test suites, with Lily being the slowest tool.

It can be concluded that there are two categories of tools. The first one comprehends tools for which the results obtained stay inside (or almost inside) the same vertical slice with respect to the logarithmic graph. To this category belong LogMapLt, LogMap, Aroma, GOMMA, Hertuda and WeSeE. The second category includes the rest of the tools, which exhibit big jumps in their measurements; their results start at one vertical slice and finish at a different slice.

The experiments also show that the tools are more sensitive to classes and properties contained in the ontologies than to the number of triples. A graph relating

[2] n/a: not able to run this test suite – u/r: uncompleted results, crashed or got stuck with a single test case.

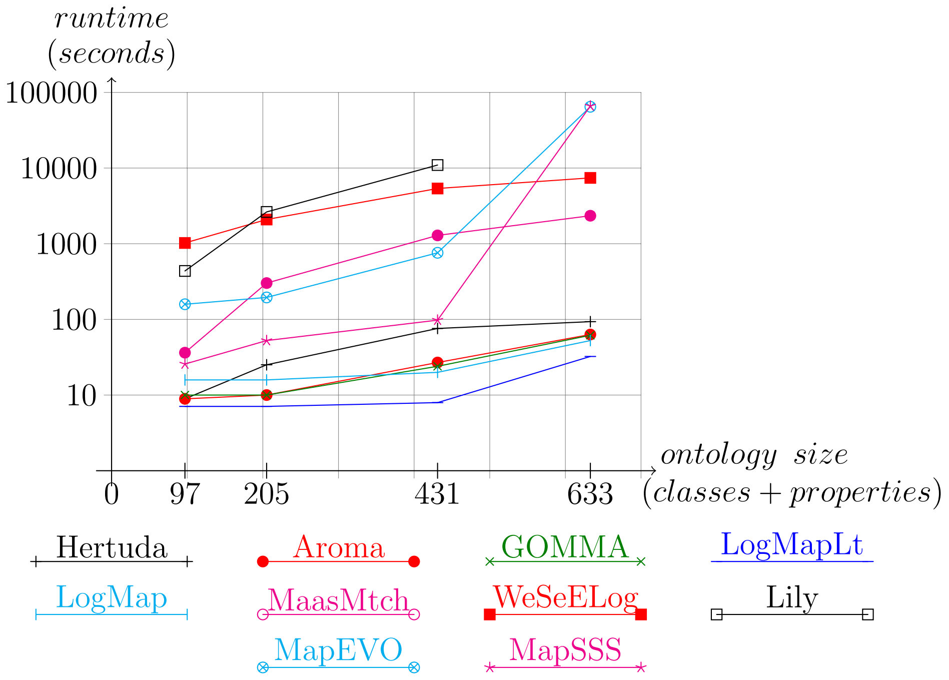

Figure 3.1: Runtime measurement compared to ontology size (classes+properties) for the Benchmark track.

runtime with triples contained in the test suites could also be drawn to support this affirmation, but it is enough to observe the fact that the biblio and jerm original ontologies have almost the same number of triplets, but the results obtained for these test suites are very different for almost all of the tools.

We can not conclude on a general correlation between runtime and quality of alignments. Lily is the slowest tool but it provides good quality alignments, and MapSSS seems to be the tool which provides the highest quality alignments, but its runtime for the finance test suite was the biggest. On the other side, the fastest tools provide in most cases better compliance results than the slowest tools, as it is the case for LogMapLt, LogMap, Aroma and GOMMA when compared with MapEVO and MapPSO.

3.2 Anatomy results

Within the following results presentation, we use a measure introduced as recall+ in 2007 [5]. This measure is based on the usage of a very simple matching system that compares labels for each matchable pair and generates a correspondences if these labels are identical (after a very simple normalization step). We refer to the resulting alignment as $\mathcal{S}$. Given an alignment $\mathcal{A}$ and a reference $\mathcal{R}$, recall+ is then defined as $|\mathcal{A} \cap (\mathcal{R} \setminus \mathcal{S})|/|(\mathcal{R} \setminus \mathcal{S})|$. This measure allows to understand whether a system can find a significant amount of non-trivial correspondences.

Matching System	Size	Precision	Recall	Recall+	F-measure
AgrMaker	1436	0.942	0.892	0.728	0.917
GOMMA-bk	1468	0.927	0.898	0.736	0.912
CODI	1305	0.96	0.827	0.562	0.888
LogMap	1391	0.918	0.842	0.588	0.879
GOMMA-nobk	1270	0.952	0.797	0.471	0.868
MapSSS	1213	0.934	0.747	0.337	0.83
LogMapLt	1155	0.956	0.728	0.29	0.827
Lily	1370	0.811	0.733	0.51	0.77
StringEquiv ($\mathcal{S}$)	934	0.997	0.622	0.000	0.766
Aroma	1279	0.751	0.633	0.344	0.687
CSA	2472	0.464	0.757	0.595	0.576
MaasMtch	2738	0.43	0.777	0.435	0.554

Table 3.4: Precision, recall, recall+ and F-measure for the Anatomy data set.

The results for the Anatomy track are presented in Table 3.4. Top results in terms of F-measure are generated by Agreementmaker[3] and GOMMA. These systems are closely followed by CODI and LogMap. We have executed the GOMMA system in two settings. In one setting, we activated the usage of UMLS as background knowledge (bk) and in another setting we deactivated this feature (no-bk). The setting, in which background knowledge is used, generates a result that is five percentage points better than the setting with deactivated background knowledge. To our knowledge AgreementMaker uses also UMLS as background knowledge.

Some systems could not top the quality of the trivial alignment $\mathcal{S}$. While those systems find many non-trivial correspondences, low F-measures are caused by low precision. There are also a few systems (AUTOMsv2, Hertuda, WeSeE, YAM++ not shown in the table) that could not generate a meaningful alignment. These systems failed with an exception, did not finish within the given time frame, or generated an empty/useless alignment (less than 1% F-measure). Note that we stopped the execution of each system after 10 hours.

For measuring runtimes, we have executed all systems on virtual machines with one, two, and four cores each with 8GB RAM. Runtime results shown in Figure 3.2 are based on the execution of the machines with one core. We executed each system three times. In the presented results we report on average runtimes. The fastest systems are LogMap (and LogMapLite), GOMMA (with and without the use of background knowledge) and AROMA. Again, we observe that the top systems can generate a high quality alignment in a short time span. In general, there is no positive correlation between the quality of the alignment and a long runtime.

As reported, we have also measured runtimes for running the matching systems in a 2-core and 4-core environment. There are some systems that scale well and some systems that can exploit a multicore environment only to a very limited degree. AROMA,

[3] AgreementMaker was executed in its 2011 version. In this year AgreementMaker used machine learning techniques to choose automatically between one of three settings optimized for the Benchmarks, Anatomy and Conference data set. The results shown are thus based on a setting optimized with respect to the Anatomy track.

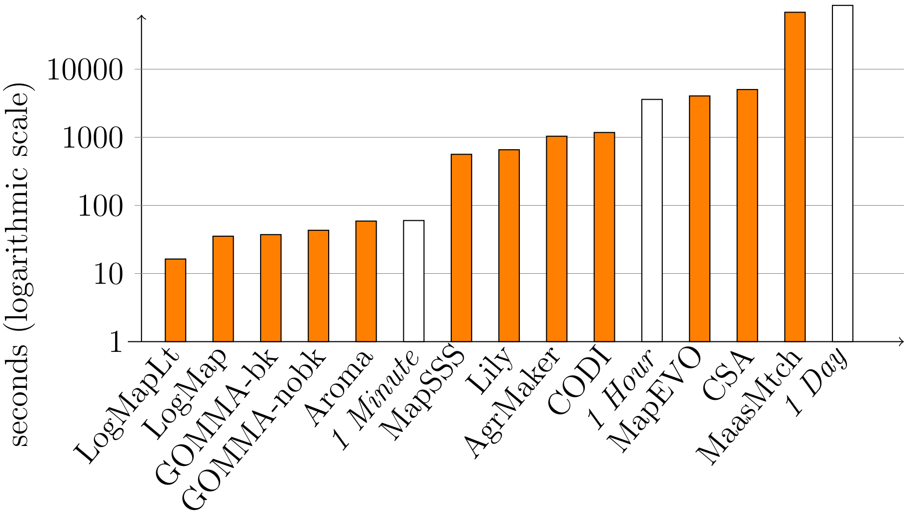

Figure 3.2: Runtimes in seconds for the Anatomy track.

LogMap, GOMMA reduce their runtime on a 4-core environment up to 50%-65% compared to executing the system with 1-core. The top system in terms of scalability is MaasMatch. Here we measured a reduction to 40%. However, these results are ambiguous. In particular, we observed a relatively high variance in measured runtimes. As already told, we executed each system three times. It is not always clear whether the high variance is related to a non-deterministic component of the matching system, or whether this might be related to uncontrolled interference in the infrastructure. Moreover, we observed that running a system with 1-core vs. running it with 4-cores has absolutely no effect on the order of systems. The differences in runtimes are too strong and the availability of additional cores cannot change this. For that reason, we have supressed a detailed presentation of these results.

In the future we have to execute more runs (>10) for each system to reduce random influences on our measurements. Moreover, we have to put our attention also to scalability issues related to the availability of memory.

3.3 Conference results

For the evaluation conducted in the context of OAEI 2011.5 the available reference alignments have been slightly refined and harmonized. The new reference alignment has been generated as a transitive closure computed on the original reference alignment. In order to obtain a coherent reference alignment, conflicting correspondences have been inspected and the conflicts have been resolved by removing one of the involved correspondences. As a result the degree of correctness and completeness of the new reference alignment is probably slightly better than for the old one. However, the differences are relatively restricted.

Table 3.5 shows the results of all participants with regard to the new reference alignment. There are precision, recall, $F_{0.5}$-measure, F_1-measure and F_2-measure computed for the threshold that provides the highest average F_1-measure. The F_1-measure is the

Matching system	**Threshold**	**Precision**	**F$_{0.5}$-measure**	**F$_1$-measure**	**F$_2$-measure**	**Recall**
YAM++	-	0.78	0.75	0.71	0.67	0.65
CODI	-	0.74	0.69	0.63	0.58	0.55
LogMap	-	0.78	0.70	0.61	0.55	0.50
WeSeE	0.33	0.67	0.61	0.55	0.49	0.46
Hertuda	-	0.74	0.65	0.55	0.48	0.44
Baseline-2	-	0.74	0.65	0.54	0.47	0.43
LogMapLt	-	0.68	0.62	0.54	0.48	0.45
GOMMA	-	0.80	0.67	0.53	0.44	0.40
AUTOMSv2	-	0.75	0.64	0.52	0.44	0.40
Baseline-1	-	0.76	0.64	0.52	0.43	0.39
MaasMatch	0.83	0.56	0.53	0.49	0.45	0.43
MapSSS	-	0.47	0.47	0.46	0.46	0.46
MapPSO	0.96	0.35	0.11	0.06	0.04	0.03
MapEVO	0.86	0.04	0.03	0.02	0.01	0.01

Table 3.5: Precision, recall, and different F-measures for the Conference track.

harmonic mean of precision and recall. F$_2$-measure (for beta=2) weights recall higher than precision and F$_{0.5}$-measure (for beta=0.5) weights precision higher than recall.

The matchers shown in the table are ordered according to their highest average F$_1$-measure. Additionally, there are two simple string matchers as baselines. Baseline-1 is a string matcher based on string equality applied on local names of entities that were lowercased before. Baseline-2 is an enhanced variant of Baseline-1 with three string operations: removing of dashes, underscore and 'has' prefix from all local names. These two baselines divide the matchers into four groups:

- Group 1 consists of the best matchers (YAM++, CODI, LogMap, WeSeE and Hertuda). These systems have better results than Baseline-2 in terms of average F$_1$-measure.
- Group 2 consists of those matchers that perform worse than Baseline-2 in terms of average F$_1$-measure but still better than Baseline-1 (LogMapLt, GOMMA, AUTOMSv2).
- Group 3 (MaasMtch and MapSSS) contains matchers that are worse than Baseline-1 but are better in terms of average F$_2$-measure. These matchers seem to favor precision over recall.
- Finally, group 4 consists of matchers (MapPSO and MapEVO) performing worse than Baseline-1 in all respects.

For better comparison with previous years we also computed the same scores, as presented in Table 3.5, based on the old reference alignment. The results computed based on the old reference alignments are almost in all cases better by 0.03 to 0.04 points. The means also that YAM++ can top the best results achieved in OAEI 2011 by 0.09 percentage points. This is a surprisingly good result. The order of matchers according to F$_1$-measure is preserved except for CODI and LogMap. These systems change their order.

3.4 MultiFarm results

For this first evaluation[4] of the data set, we use a subset of the whole data set. In this subset (a) we omitted the ontologies edas and ekaw; (b) we suppressed the test cases where Russian and Chinese are involved. The reason for this is that most participating systems are still based on using no specific multilingual technique, that might still work (to some limited degree) on matching German on English, but will fail when matching, for instance, French to Russian or Chinese.

Within the MultiFarm data set it can be distinguished between two types of test cases. (1) Those test cases where two different ontologies have been translated in different languages, and (2) those test cases where the same ontology has been translated in different languages. Test cases of the second type are those test cases where different versions of the same ontology have to be matched. Good results for these test cases might not depend on multilingual methods, but on the ability to exploit the fact that both ontologies have an identical structure and that the reference alignment covers all entities described in the ontologies. It can be supposed that these test cases are dominated by specific techniques designed for matching different versions of the same ontology.

To our knowledge three participating systems use specific multilingual methods. These systems are WeSeE, AUTOMSv2 and YAM++. The other systems are not specifically designed to match ontologies in different languages, nor do they make use of a component that can be utilized for that purpose. WeSeE-Match and and YAM++ use Microsoft Bing to translate labels contained in the input ontologies to English. The translated English ontologies are then matched using standard matching procedures of WeSeE and YAM++. AUTOMSv2 re-uses a free Java API named WebTranslator to translate the ontologies to English. This process is performed before AUTOMSv2 applies its standard matching methods.

First of all we aggregated the results for all test cases of types (i) and (ii). The results are presented in Table 3.6. The systems not listed in this table have generated empty alignments for the test cases of type (i), or have thrown some exceptions.

First of all, significant differences between results measured for test cases (i) and (ii) can be observed. While the three systems that implement specific multilingual techniques clearly generate the best results for test cases of type (i), only one of these systems is among the top systems for type (ii) test cases. This subset of the data set is dominated by the systems YAM++, CODI, and MapSSS. These systems generate also good results for the Benchmark data sets. At the same time there is no (or only a very weak) correlation between results for test cases of type (i) and type (ii). For that reason, we analyze in the following only the results for test cases of type (i). In particular, we also do not include them in the representation of aggregated results. Results for test cases of type (i) can, instead of that, be interpreted as results for matching different versions of the same ontology.

So far we can conclude that specific methods work much better than state-of-the-art techniques applied to MultiFarm test cases. This is a result that we expected.

[4] A preliminary report on these results has been published at the IWEST workshop [12]. The results presented here include the results of the IWEST paper extended by the results obtained for new OAEI 2011.5 participants.

	Different ontologies (type i)				Same ontologies (type ii)			
Matching system	**Size**	**Precision**	**Recall**	**F-measure**	**Size**	**Precision**	**Recall**	**F-measure**
YAM++	1838	0.54	0.39	0.45	5838	0.93	0.48	0.63
AUTOMSv2	746	0.63	0.25	0.36	1379	0.92	0.16	0.27
WeSeE	4211	0.24	0.39	0.29	5407	0.76	0.36	0.49
CIDER	737	0.42	0.12	0.19	1090	0.66	0.06	0.12
MapSSS	1273	0.16	0.08	0.10	6008	0.97	0.51	0.67
LogMap	335	0.36	0.05	0.09	400	0.61	0.02	0.04
CODI	345	0.34	0.04	0.08	7041	0.83	0.51	0.63
MaasMtch	15939	0.04	0.28	0.08	11529	0.23	0.23	0.23
LogMapLt	417	0.26	0.04	0.07	387	0.56	0.02	0.04
MapPSO	7991	0.02	0.06	0.03	6325	0.07	0.04	0.05
CSA	8482	0.02	0.07	0.03	8348	0.49	0.36	0.42
MapEVO	4731	0.01	0.01	0.01	3560	0.05	0.01	0.02

Table 3.6: Precision, recall, recall+ and F-measure.

However, the absolute results are still not very good, if compared to the top results of the Conference data set (approx. 0.7 F-measure). From all specific multilingual methods, the techniques implemented by YAM++ generate the best alignments in terms of F-measure. Remember that YAM++ has also generated the best results for the Conference track. YAM++ is followed by AUTOMSv2 and WeSeE. It is also an interesting outcome to see that CIDER can generate clearly the best results compared to all other systems with non-specific multilingual systems.

As expected, the systems that apply specific strategies to deal with multilingual ontology labels outperform the other systems: YAM++, followed by AUTOMS and WeSeE, respectively, outperforms all other systems. For these three systems, looking for each pair of languages, the best five F-measures are obtained for en-fr (0.61), cz-en (0.58), cz-fr (0.57), en-pt (0.56), and en-nl/cz-pt/fr-pt (0.55). Apart the ontology structure differences, most of these language pairs do not have overlapping vocabularies (cz-pt or cz-fr, for instance). Hence, the translation step has an impact on the conciliation of the differences between languages. However, as expected, it is not the only impact factor, considering that YAM++ and WeSeE are based on the same translator, nevertheless, YAM++ outperforms WeSeE for most of the pairs. Looking for the average of these three systems, we have the following pairs ranking: en-fr (0.47), en-pt (0.46), en-nl (0.44), de-en (0.43) and fr-pt (0.40), with English as a common language (due to the matchers strategy of translation).

For the other group of systems, CIDER is ahead the others, providing the best scores: de-en (0.33), es-pt (0.30), es-fr (0.29), de-es (0.28) and en-es (0.25). MapSSS, LogMap, and CODI are the followers. For all of these four systems, the pairs es-pt and de-en are ahead in their sets of best F-measures. These two pairs contain languages whose vocabularies share similar terms. Once most of the systems take advantage of label similarities it is likely that it may be harder to find correspondences between cz-pt than es-pt. However, for some systems their five best score includes these kind of pairs (cz-pt, for CODI and LogMapLt or de-es for LogMap).

Matching System	Size	Precision	Recall	F-measure	Runtime$_S$	Runtime$_L$	Unsat.	Degree
GOMMA-bk	2878	0.925	0.918	0.921	34	67	6292	61.78%
LogMap	2739	0.935	0.884	0.909	20	41	2	0.02%
GOMMA	2628	0.945	0.857	0.899	27	50	2130	20.92%
LogMapLt	2483	0.942	0.807	0.869	10	12	2104	20.66%
Aroma	2575	0.802	0.713	0.755	68	140	7558	74.21%
MaasMatch	3696	0.580	0.744	0.652	9437	-	9718	95.42%
CSA	3607	0.514	0.640	0.570	14414	26580	9590	94.17%
MapSSS	1483	0.840	0.430	0.569	571	937	565	5.55%
MapPSO	3654	0.021	0.025	0.023	-	41686	10145	99.62%
MapEVO	633	0.003	0.001	0.002	2985	5252	9164	89.98%

Table 3.7: Results for small module of the Large Biomedical track. Runtime-S refer to the runtime in seconds when executing the system on the server, Runtime-L refers to the runtime on the laptop.

3.5 Large Biomedical results

As explained above, the data set of the Large Biomedical track consists of three subsets of increasing size. In the following we focus on the results obtained for the small subset. We refer the reader to the webpage of the track for a complete presentation of the results.[5] Note that the evaluations of the Large Biomedical track have been conducted using parts of the SEALS technology, namely the SEALS client with its implicit usage of the SEALS test data repository. However, the evaluations have not been conducted on the SEALS hardware but on the machines of the track organizers. This has been (1) a standard laptop with 2 cores and 4Gb RAM, and (2) a high performance server with 16 CPUs and 10 Gb. This illustrates the flexibility of the tools developed within the SEALS project.

In total, nine systems have been able to cope with the smallest of the matching problems defined by the track. The results for these systems are shown in Table 3.7. GOMMA-bk obtained the best results in terms of both recall and F-measure while GOMMA-nobk provided the most precise alignments. GOMMA (with its two configurations) and LogMap are bit ahead with respect to Aroma, MaasMatch, CSA and MapSSS in terms of F-measure. MapSSS provided a good precision, however the F-measure was damaged due to the low recall of its mappings. Nevertheless, these tools can deal with large ontologies such as FMA and NCI and they will be very helpful for the creation of the future silver standard reference alignment for the track.

MapPSO and MapEVO are two special cases for which we did not obtain meaningful alignments. Both systems generate comprehensive alignments, however, they only found a few correct correspondences. Furthermore, when running in the server, MapPSO threw an exception related to the parallelization of its algorithm. The reason of such low quality results is mostly due to MapEVO and MapPSO configurations for this track. MapEVO and MapPSO algorithms work iteratively converging towards an optimum alignment and are designed to be executed on a large parallel infrastructure.

[5] http://www.cs.ox.ac.uk/isg/projects/SEALS/oaei/results2011.5.html

MapEVO and MapPSO authors reduced significantly the number of iterations and parallel threads due to infrastructure availability.

The runtimes were quite good in general. Only MaasMatch and CSA needed more than 2.5 and 4 hours, respectively, to complete the task. Furthermore, MaasMatch did not finished after more than 12 hours of execution in the laptop setting. We can also appreciate that, in some cases, times in the server are reduced in more than 50%. This conincides with the observations we made for MaasMatch and its performance on the Anatomy track.

Regarding mapping coherence, only LogMap generates an almost clean output. The table shows both (1) the number of unsatisfiabilities when reasoning (using HermiT) with the input ontologies together with the computed mappings, and (2) the ratio/degree of unsatisfiable classes with respect to the size of the merged ontology. From the results presented in the table, it can be concluded that even the most precise mappings (GOMMA-nobk) lead to a huge amount of unsatisfiable classes. This proves the importance of using techniques to assess the coherence of the generated alignments. Unfortunately, LogMap and CODI are the unique systems (participating in the OAEI 2011.5) that use such techniques.

4. Lessons Learned

In the previous sections we have discussed the results for each track on its own in detail. In the following we highlight the most interesting lessons learned.

(a) Some matching systems generate unexpected (positive or negative) results. An example for this is the performance of YAM++ [13] on the Conference data set. Without a description of the matching system (or its extensions compared to previous campaigns), it is hard to understand why a systems performs as observed.[1]

(b) The MultiFarm data set has been accepted by the community as a new challenge in OAEI. Three systems have implemented specific methods for matching ontologies in different languages. This is an unexpected development given the short time that MultiFarm was available to the community.

(c) Finally, eight systems could generate an alignment for the small module of the Large Biomedical data set. These are more systems than we expected. Moreover, the track attracted the participation of GOMMA [9], a system that is specifically designed to match ontologies from life science domain.

(d) There is no correlation between a long runtime and the quality of the generated alignment. This has already been a result of previous campaigns that becomes visible again in different OAEI 2011.5 tracks. In particular, the benchmark track shows that some of the fastest tools provide in most cases good compliance results. That was the case for LogMapLt, LogMap, Aroma and GOMMA. However the tools having the better results for compliance tests (MapSSS and Lily) are situated in the group of the slowest tools.

(e) The results of the Benchmark scalability experiments show that some tools were not able to process all the Benchmark test suites due to exceptions thrown at runtime. One possible reason is because new ontologies unknown for tool developers were used in these experiments. Short data sets were published at the end of the evaluation campaign, and the developers of those tools reported that the bugs are already fixed.

(f) An interesting point for the benchmark runtime scalability experiments is that the tools are more sensible to the number of classes and properties contained in the ontologies than to the number of triples. The biblio and jerm test suites used in this track contain almost the same number of triples with jerm having almost three times the number of classes and properties than biblio. In all cases, the time spent for processing the jerm data set was greater than or equal to the time spent for processing the biblio data sets.

(g) The scalability experiments of the Anatomy and Large BioMed track have ambiguous results. On the one hand the results of BioMed show that a system as

[1] Previous OAEI campaigns have been collocated with the Ontology Matching workshop and each participant was asked to write a paper that describes the system on a few pages published in the workshop proceedings. This information is missing in OAEI 2011.5.

MaasMatch can generate an alignment on a server with several cores and fails on a laptop. On the other hand the differences between runtimes are so strong that additional cores (1 core vs. 4 cores) could not change the order of measured runtimes.

(h) Systematic scalability experiments related to the impact of available RAM are missing. This is an important aspect that has to be taken into account in OAEI 2012.

(i) The results for the MultiFarm data set have shed light on the distinction between matching different versions of the same ontology and matching different ontologies that describe the same (or overlapping) domain(s). It seems that some systems use very specific methods to generate good results for matching different versions of the same ontologies.

5. Final Remarks

This deliverable presented the results of the 2011.5 OAEI/SEALS integrated campaign. All OAEI tracks have been conducted in the SEALS modality, i.e., all matching systems have been executed with the use of the SEALS client and for most of the tracks the SEALS virtualization infrastructure has been used. The new technology introduced in OAEI affected both tool developers and organizers to a large degree and has been accepted positively on both sides.

While the SEALS client has already been given to participants in OAEI 2011, which happened between the first and the second official SEALS campaign, the technical advances of the platform had their main impact on side of the evaluation campaign organizers. We could profit from the computational power offered by the platform. Thus, it was possible to conduct the scalability experiments and to generate results for all five OAEI tracks.

In the deliverable that reported on the first campaign (OAEI 2010) we concluded with several plans as future work. In the following we list these points and comment on each of them.

- **Develop a test generator that allows a controlled automatic test generation.** This test data generator has already been used for OAEI 2011 and we have been using an enhanced version to generate the test data sets of the Benchmark track.

- **Find or generate more well suited data sets to be used in the campaign.** Two additional tracks based on completely new data sets have been offered. Moreover, in the benchmark track four automatically generated data sets have been used.

- **Measure runtime and memory consumption in a controlled execution environment.** We have used the SEALS virtualization infrastructure to measure the runtime of the matching systems in a controlled environment. We have not yet measured memory consumption. Instead of that we have focused on the impact of available cores on the runtime of a system.

- **Guarantee the reproducibility of the results** The technology used in OAEI 2011 and OAEI 2011.5 allows us to reproduce all results.

- **Integrate additional visualization components** A generic visualization framework is currently under development. This framework allows to visualize results of different research areas in a unique way.

Overall, we conclude that we managed to reach most of our goals. This is confirmed indirectly by a high acceptance of the SEALS Ontology Matching campaigns, by the uptake of SEALS technology on side of the tool developer, and by the positive feedback from the matching community.

References

[1] Elena Beisswanger and Udo Hahn. Towards valid and reusable reference alignments - ten basic quality checks for ontology alignments and their application to three different reference data sets. *Journal of Biomedical Semantics*, 2012.

[2] Oliver Bodenreider. The unified medical language systemumls): integrating biomedical terminology. *Nucleic acids research*, 32, 2004.

[3] Oliver Bodenreider, Terry F. Hayamizu, Martin Ringwald, Sherri De Coronado, and Songmao Zhang. Of mice and men: Aligning mouse and human anatomies. In *AIMA 2005 Symposium Proceedings*, pages 61–65, 2005.

[4] Jérôme Euzenat. An API for ontology alignment. In *Proceedings of the 3rd International Semantic Web Conference (ISWC)*, pages 698–712, Hiroshima (JP), 2004.

[5] Jérôme Euzenat, Antoine Isaac, Christian Meilicke, Pavel Shvaiko, Heiner Stuckenschmidt, Ondrej Svab, Vojtech Svatek, Willem Robert van Hage, and Mikalai Yatskevich. Results of the ontology alignment evaluation initiative 2007. In Pavel Shvaiko, Jrme Euzenat, Fausto Giunchiglia, and Bin He, editors, *Proceedings of the 2nd ISWC international workshop on Ontology Matching, Busan (KR)*, pages 96–132, 2007.

[6] Jérôme Euzenat, Christian Meilicke, Pavel Shvaiko, Heiner Stuckenschmidt, and Cássia Trojahn dos Santos. Ontology alignment evaluation initiative: six years of experience. *Journal on Data Semantics*, XV:158–192, 2011.

[7] Ernesto Jimenez-Ruiz and Bernardo Cuenca Grau. Logmap: Logic-based and scalable ontology matching. In *10th International Semantic Web Conference*, pages 273–288, 2011.

[8] Ernesto Jimenez-Ruiz, Bernardo Cuenca Grau, Ian Horrocks, and Raffael Berlanga. Logic-based assessment of the compatibility of umls ontology sources. *Journal of Biomedical Semantics*, 2011.

[9] Toralf Kirsten, Anika Gross, Michael Hartung, , and Erhard Rahm. Gomma: a component-based infrastructure for managing and analyzing life science ontologies and their evolution. *Journal of Biomedical Semantics*, 2011.

[10] Christian Meilicke, Rául Garícia Castro, Fred Freitas, Willem Robert van Hage, Elena Montiel-Ponsoda, Ryan Ribeiro de Azevedo, Heiner Stuckenschmidt, Ondřej Šváb-Zamazal, Vojtech Svátek, Andrei Tamilin, Cássia Trojahn, and Shenghui Wang. Multifarm: A benchmark for multilingual ontology matching. *Journal of Web Semantics*, 2012.

[11] Christian Meilicke, Cássia Trojahn dos Santos, Jérôme Euzenat, and Heiner Stuckenschmidt. D12.5 v2.0-beta: Iterative implementation of services for the automatic evaluation of matching tools. Technical report, SEALS Project <http://sealsproject.eu>, December 2011.

[12] Christian Meilicke, Cassia Trojahn, Ondrej Sváb-Zamazal, and Dominique Ritze. Multilingual ontology matching evaluation - a first report on using multifarm. In *Proceedings of the Second International Workshop on Evaluation of Semantic Technologies (IWEST 2012)*, pages 1–12, 2012.

[13] DuyHoa Ngo, Zohra Bellahsene, and Remi Coletta. YAM++ results for OAEI 2011. In *Proceedings of the 6th International Workshop on Ontology Matching*, 2011.

[14] Ondrej Svab, Vojtech Svatek, Petr Berka, Dusan Rak, and Petr Tomasek. Ontofarm: Towards an experimental collection of parallel ontologies. In *Poster Track of ISWC*, Galway, Ireland, 2005.

[15] Cássia Trojahn, Christian Meilicke, Jérôme Euzenat, and Ondřej Šváb Zamazal. Results of the first evaluation of matching tools. Technical Report D12.3, SEALS Project, November 2010.