Final results of the Ontology Alignment Evaluation Initiative 2011*

Jérôme Euzenat[1], Alfio Ferrara[2], Willem Robert van Hage[3], Laura Hollink[4], Christian Meilicke[5], Andriy Nikolov[6], François Scharffe[7], Pavel Shvaiko[8], Heiner Stuckenschmidt[5], Ondřej Šváb-Zamazal[9], and Cássia Trojahn[1]

[1] INRIA & LIG, Montbonnot, France
{jerome.euzenat,cassia.trojahn}@inria.fr
[2] Universita degli studi di Milano, Italy
ferrara@dico.unimi.it
[3] Vrije Universiteit Amsterdam, The Netherlands
W.R.van.Hage@vu.nl
[4] Delft University of Technology, The Netherlands
l.hollink@tudelft.nl
[5] University of Mannheim, Mannheim, Germany
{christian,heiner}@informatik.uni-mannheim.de
[6] The Open University, Milton Keynes, UK
A.Nikolov@open.ac.uk
[7] LIRMM, Montpellier, FR
francois.scharffe@lirmm.fr
[8] TasLab, Informatica Trentina, Trento, Italy
pavel.shvaiko@infotn.it
[9] University of Economics, Prague, Czech Republic
ondrej.zamazal@vse.cz

Abstract. Ontology matching consists of finding correspondences between semantically related entities of two ontologies. OAEI campaigns aim at comparing ontology matching systems on precisely defined test cases. These test cases can use ontologies of different nature (from simple directories to expressive OWL ontologies) and use different modalities, e.g., blind evaluation, open evaluation, consensus. OAEI-2011 builds over previous campaigns by having 4 tracks with 6 test cases followed by 18 participants. Since 2010, the campaign has been using a new evaluation modality which provides more automation to the evaluation. In particular, this year it allowed to compare run time across systems. This paper is an overall presentation of the OAEI 2011 campaign.

1 Introduction

The Ontology Alignment Evaluation Initiative[1] (OAEI) is a coordinated international initiative, which organizes the evaluation of the increasing number of ontology matching systems [10; 8; 15]. The main goal of OAEI is to compare systems and algorithms

* This paper improves on the "First results" initially published in the on-site proceedings of the ISWC workshop on Ontology Matching (OM-2011). The only official results of the campaign, however, are on the OAEI web site.

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

on the same basis and to allow anyone for drawing conclusions about the best matching strategies. Our ambition is that, from such evaluations, tool developers can improve their systems.

Two first events were organized in 2004: (i) the Information Interpretation and Integration Conference (I3CON) held at the NIST Performance Metrics for Intelligent Systems (PerMIS) workshop and (ii) the Ontology Alignment Contest held at the Evaluation of Ontology-based Tools (EON) workshop of the annual International Semantic Web Conference (ISWC) [17]. Then, a unique OAEI campaign occurred in 2005 at the workshop on Integrating Ontologies held in conjunction with the International Conference on Knowledge Capture (K-Cap) [1]. Starting from 2006 through 2010 the OAEI campaigns were held at the Ontology Matching workshops collocated with ISWC [9; 7; 2; 5; 6]. Finally in 2011, the OAEI results were presented again at the Ontology Matching workshop collocated with ISWC, in Bonn, Germany[2].

Since last year, we have been promoting an environment for automatically processing evaluations (§2.2), which were developed within the SEALS (Semantic Evaluation At Large Scale) project[3]. This project aims at providing a software infrastructure for automatically executing evaluations, and evaluation campaigns for typical semantic web tools, including ontology matching. Several OAEI data sets were evaluated under the SEALS modality. This provides a more uniform evaluation setting.

This paper serves as a synthesis to the 2011 evaluation campaign and as an introduction to the results provided in the papers of the participants. The remainder of the paper is organized as follows. In Section 2, we present the overall evaluation methodology that has been used. Sections 3-6 discuss the settings and the results of each of the test cases. Section 7 overviews lessons learned from the campaign. Finally, Section 8 outlines future plans and Section 9 concludes the paper.

2 General methodology

We first present the test cases proposed this year to the OAEI participants (§2.1). Then, we discuss the resources used by participants to test their systems and the execution environment used for running the tools (§2.2). Next, we describe the steps of the OAEI campaign (§2.3-2.5) and report on the general execution of the campaign (§2.6).

2.1 Tracks and test cases

This year's campaign consisted of 4 tracks gathering 6 data sets and different evaluation modalities:

The benchmark track (§3): Like in previous campaigns, a systematic benchmark series have been proposed. The goal of this benchmark series is to identify the areas in which each matching algorithm is strong or weak. The test is based on one particular ontology dedicated to the very narrow domain of bibliography and a number of alternative ontologies of the same domain for which reference alignments are

[2] http://om2011.ontologymatching.org
[3] http://www.seals-project.eu

provided. This year, we used new systematically generated benchmarks, based on other ontologies than the bibliographic one.

The expressive ontologies track offers real world ontologies using OWL modeling capabilities:

Anatomy (§4): The anatomy real world case is about matching the Adult Mouse Anatomy (2744 classes) and a part of the NCI Thesaurus (3304 classes) describing the human anatomy.

Conference (§5): The goal of the conference task is to find all correct correspondences within a collection of ontologies describing the domain of organizing conferences (the domain being well understandable for every researcher). Additionally, 'interesting correspondences' are also welcome. Results were evaluated automatically against reference alignments and by using logical reasoning techniques.

Oriented alignments: This track focused on the evaluation of alignments that contain other relations than equivalences. It provides two data sets of real ontologies taken from a) Academia (alterations of ontologies from the OAEI benchmark series), b) Course catalogs (alterations of ontologies concerning courses in the universities of Cornell and Washington). The alterations aim to introduce additional subsumption correspondences between classes that cannot be inferred via reasoning.

Model matching: This data set compares model matching tools from the Model-Driven Engineering (MDE) community on ontologies. The test cases are available in two formats: OWL and Ecore. The models to be matched have been automatically derived from a model-based repository.

Instance matching (§6): The goal of the instance matching track is to evaluate the performance of different tools on the task of matching RDF individuals which originate from different sources but describe the same real-world entity. Instance matching is organized in two sub-tasks:

Data interlinking (DI) This year the Data interlinking track focused on retrieving New York Times (NYT) interlinks with DBPedia, Freebase and Geonames. The NYT data set includes 4 data subsets: persons, locations, organizations and descriptors that should be matched to themselves to detect duplicates, and to DBPedia, Freebase and Geonames. Only Geonames has links to the Locations data set of NYT.

OWL data track (IIMB): The synthetic OWL data track is focused on (i) providing an evaluation data set for various kinds of data transformations, including value transformations, structural transformations, and logical transformations; (ii) covering a wide spectrum of possible techniques and tools. To this end, the IIMB benchmark is generated by starting from an initial OWL knowledge base that is transformed into a set of modified knowledge bases by applying several automatic transformations of data. Participants are requested to find the correct correspondences among individuals of the first knowledge base and individuals of the others.

Table 1 summarizes the variation in the results expected from the tests under consideration.

test	formalism	relations	confidence	modalities	language	SEALS
benchmarks	OWL	=	[0 1]	open	EN	√
anatomy	OWL	=	[0 1]	open	EN	√
conference	OWL-DL	=, <=	[0 1]	blind+open	EN	√
di	RDF	=	[0 1]	open	EN	
iimb	RDF	=	[0 1]	open	EN	

Table 1. Characteristics of the test cases (open evaluation is made with already published reference alignments and blind evaluation is made by organizers from reference alignments unknown to the participants).

This year we had to cancel the Oriented alignments and Model matching tracks which have not had enough participation. We preserved the IIMB track with only one participant, especially because the participant was not tied to the organizers and participated in the other tracks as well.

2.2 The SEALS platform

In 2010, participants of the Benchmark, Anatomy and Conference tracks were asked for the first time to use the SEALS evaluation services: they had to wrap their tools as web services and the tools were executed on the machines of the tool developers [18].

In 2011, tool developers had to implement a simple interface and to wrap their tools in a predefined way including all required libraries and resources. A tutorial for tool wrapping was provided to the participants. This tutorial described how to wrap a tool and how to use a simple client to run a full evaluation locally. After local tests had been conducted successfully, the wrapped tool was uploaded for a test on the SEALS portal[4]. Consequently it was executed on the SEALS platform by the organisers in a semi-automated way. This approach allowed to measure runtime and ensured the reproducibility of the results for the first time in the history of OAEI. As a side effect, this approach ensures also that a tool is executed with the same settings for all of the three tracks. This was already requested by the organizers in the past years. However, this rule was sometimes ignored by participants.

2.3 Preparatory phase

Ontologies to be matched and (where applicable) reference alignments have been provided in advance during the period between May 30^{th} and June 27^{th}, 2011. This gave potential participants the occasion to send observations, bug corrections, remarks and other test cases to the organizers. The goal of this preparatory period is to ensure that the delivered tests make sense to the participants. The final test base was released on July 6^{th}, 2011. The data sets did not evolve after that, except for the reference alignment of the Anatomy track to which minor changes have been applied to increase its quality.

[4] `http://www.seals-project.eu/join-the-community/`

2.4 Execution phase

During the execution phase, participants used their systems to automatically match the ontologies from the test cases. In most cases, ontologies are described in OWL-DL and serialized in the RDF/XML format [3]. Participants were asked to use one algorithm and the same set of parameters for all tests in all tracks. It is fair to select the set of parameters that provides the best results (for the tests where results are known). Beside parameters, the input of the algorithms must be the two ontologies to be matched and any general purpose resource available to everyone, i.e., no resource especially designed for the test. In particular, participants should not use the data (ontologies and reference alignments) from other test cases to help their algorithms.

Participants can self-evaluate their results either by comparing their output with reference alignments or by using the SEALS client to compute precision and recall.

2.5 Evaluation phase

Participants have been encouraged to provide (preliminary) results or to upload their wrapped tools on the SEALS portal by September 1^{st}, 2011. Organizers evaluated the results and gave feedback to the participants. For the SEALS modality, a full-fledged test on the platform has been conducted by the organizers and problems were reported to the tool developers, until finally a properly executable version of the tool has been uploaded on the SEALS portal. Participants were asked to send their final results or upload the final version of their tools by September 23^{th}, 2011. Participants also provided the papers that are published hereafter.

As soon as first results were available, these results were published on the respective web pages by the track organizers. The standard evaluation measures are precision and recall computed against the reference alignments. For the matter of aggregation of the measures, we used weighted harmonic means (weights being the size of the true positives). This clearly helps in the case of empty alignments. Another technique that was used is the computation of precision/recall graphs so it was advised that participants provide their results with a weight to each correspondence they found. New measures addressing some limitations of precision and recall have also been used for testing purposes as well as measures for compensating the lack of complete reference alignments. Additionally, we measured runtimes for all tracks conducted under the SEALS modality.

2.6 Comments on the execution

For a few years, the number of participating systems has remained roughly stable: 4 participants in 2004, 7 in 2005, 10 in 2006, 17 in 2007, 13 in 2008, 16 in 2009, 15 in 2010 and 18 in 2011. However, participating systems are now constantly changing.

The number of covered runs has increased more than observed last year: 48 in 2007, 50 in 2008, 53 in 2009, 37 in 2010, and 53 in 2011. This is, of course, due to the ability to run all systems participating in the SEALS modality in all tracks. However, not all tools participating in the SEALS modality could generate results for the anatomy track (see Section 4). This does not really contradict the conjecture we made last year that

systems are more specialized. In fact, only two systems (AgreementMaker and CODI) participated also in the instance matching tasks, and CODI only participated in a task (IIMB) in which no other instance matching system entered.

This year we were able to run most of the matchers in a controlled evaluation environment, in order to test their portability and deployability. This allowed us comparing systems on the same execution basis. This is also a guarantee that the tested system can be executed out of their particular development environment.

The list of participants is summarized in Table 2.

System	AgrMaker	Aroma	CSA	CIDER	CODI	LDOA	Lily	LogMap	MaasMtch	MapEVO	MapPSO	MapSSS	OACAS	OMR	Optima	Serimi	YAM++	Zhishi	Total=18
Confidence	√	√	√	√		√	√	√	√	√	√		√	√		√	√	√	
benchmarks	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√		√		16
anatomy	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√		√		16
conference	√	√	√	√	√	√	√	√	√	√	√	√			√		√		14
di	√															√		√	3
iimb					√														1
Total	4	3	3	3	4	3	3	3	3	3	3	3	2	2	3	1	3	1	53

Table 2. Participants and the state of their submissions. Confidence stands for the type of result returned by a system: it is ticked when the confidence has been measured as non boolean value.

Two systems require a special remark. YAM++ used a setting that was learned from the reference alignments of the benchmark data set from OAEI 2009, which is highly similar to the corresponding benchmark in 2011. This affects the results of the traditional OAEI benchmark and no other tests. Moreover, we have run the benchmark in newly generated tests where YAM++ is indeed having weaker performances. Considering that indeed benchmarks was one of the few tests on which to train algorithms, we decided to keep YAM++ results with this warning.

AgreementMaker used machine learning techniques to choose automatically between one of three settings optimized for the benchmark, anatomy and conference data set. It used a subset of the available reference alignments as input to the training phase and clearly a specific tailored setting for passing these tests. This is typically prohibited by OAEI rules. However, at the same time, AgreementMaker has improved its results over last year so we found interesting to report them.

The summary of the results track by track is provided in the following sections.

3 Benchmark

The goal of the benchmark data set is to provide a stable and detailed picture of each algorithm. For that purpose, algorithms are run on systematically generated test cases.

3.1 Test data

The systematic benchmark test set is built around a seed ontology and many variations of it. The ontologies are described in OWL-DL and serialized in the RDF/XML format. The reference ontology is that of test #101. Participants have to match this reference ontology with the variations. Variations are focused on the characterization of the behavior of the tools rather than having them compete on real-life problems. They are organized in three groups:

Simple tests (1xx) such as comparing the reference ontology with itself, with another irrelevant ontology (the wine ontology used in the OWL primer) or the same ontology in its restriction to OWL-Lite;

Systematic tests (2xx) obtained by discarding features from a reference ontology. It aims at evaluating how an algorithm behaves when a particular type of information is lacking. The considered features were:

- *Name of entities* that can be replaced by random strings, synonyms, name with different conventions, strings in another language than English;
- *Comments* that can be suppressed or translated in another language;
- *Specialization hierarchy* that can be suppressed, expanded or flattened;
- *Instances* that can be suppressed;
- *Properties* that can be suppressed or having the restrictions on classes discarded;
- *Classes* that can be expanded, i.e., replaced by several classes or flattened.

Four real-life ontologies of bibliographic references (3xx) found on the web and left mostly untouched (there were added xmlns and xml:base attributes). This is only used for the initial benchmark.

This year, we departed from the usual bibliographic benchmark that have been used since 2004. We used a new test generator [14] in order to reproduce the structure of benchmark from different seed ontologies. We have generated three different benchmarks against which matchers have been evaluated:

benchmark (biblio) is the benchmark data set that has been used since 2004. It is used for participants to check that they can run the tests. It also allows for comparison with other systems since 2004. The seed ontology concerns bibliographic references and is inspired freely from BibTeX. It contains 33 named classes, 24 object properties, 40 data properties, 56 named individuals and 20 anonymous individuals. We have considered the original version of benchmark (referred as **original** in the subsections above) and a new automatically generated one (**biblio**).

benchmark2 (ekaw) The Ekaw ontology, one of the ontologies from the conference track (§5), was used as seed ontology for generating the Benchmark2 data set. It contains 74 classes and 33 object properties. The results with this new data set were provided to participants after the preliminary evaluation.

benchmark3 (finance) This data set is based on the Finance ontology[5], which contains 322 classes, 247 object properties, 64 data properties and 1113 named individuals. This ontology was not disclosed to the participants.

[5] http://www.fadyart.com/ontologies/data/Finance.owl

Having these three data sets allows us to better evaluate the dependency between the results and the seed ontology. The SEALS platform allows for evaluating matchers against these many data sets automatically.

For all data sets, the reference alignments are still limited: they only match named classes and properties and use the "=" relation with confidence of 1. Full description of these tests can be found on the OAEI web site.

3.2 Results

16 systems have participated in the benchmark track of this year's campaign (see Table 2). From the eleven participants last year, only four participated this year (AgreementMaker, Aroma, CODI and MapPSO). On the other hand, there are ten new participants, while two participants (CIDER and Lily) have been participating in the previous campaigns as well. In the following, we present the evaluation results, both in terms of runtime and compliance with relation to reference alignments.

Portability. 18 systems have been registered on the SEALS portal. One has abandoned due to requirements posed by the platform and another one abandoned silently. Thus, 16 systems bundled their tools into the SEALS format. From these 16 systems, we have not been able to run the final versions of OMR and OACAS (packaging error). CODI was not working on the operating system version under which we measured runtime. CODI runs under Windows and some versions of Linux, but has specific requirements not met on the Linux version that has been used for runnning the SEALS platform (Fedora 8). Some other systems still have (fixable) problems with the output they generate[6].

Runtime. This year we were able to measure the performance of matchers in terms of runtime. We used a 3GHz Xeon 5472 (4 cores) machine running Linux Fedora 8 with 8GB RAM. This is a very preliminary setting for mainly testing the deployability of tools into the SEALS platform.

Table 3 presents the time required by systems to complete the 94 tests in each data set[7]. These results are based on 3 runs of each matcher on each data sets. We also include the result of a simple edit distance algorithm on labels (edna). Unfortunately, we were not able to compare CODI's runtime with other systems'.

Considering all tasks but finance, there are systems which can run them within less than 15mn (Aroma, edna, LogMap, CSA, YAM++, MapEVO, AgreementMaker, MapSSS), there are systems performing the tasks within one hour (Cider, MaasMatch, Lily) and systems which need more time (MapPSO, LDOA, Optima). Figure 1 better illustrates the correlation between the number of elements in each seed ontology and the time taken by matchers for generating the 94 alignments. The faster matcher, independently from the seed ontology, is Aroma (even for finance), followed by LogMap,

[6] All evaluations have been performed with the Alignment API 4.2 [3] with the exception of LDOA for which we had to adapt the relaxed evaluators to obtain results.

[7] From the 111 tests in the original benchmark data set, 17 of them have not been automatically generated: 102–104, 203–210, 230–231, 301–304. For comparative purposes, they were discarded.

	original		biblio		ekaw		finance	
System	Runtime	Top-5	Runtime	Top-5	Runtime	Top-5	Runtime	Top-5
edna	1.07		1.06		1.00		33.70	
AgrMaker	12.42	√	—x		2.81	√	3.81h	√
Aroma	1.05		1.10	√	0.77		10.83	√
CSA	2.47	√	2.61	√	3.69	√	3.10h	√
CIDER	32.50		30.30		28.08		**46.15h**	√
CODI	—Error	√	—Error	√	—Error	√	—Error	
LDOA	28.94h		29.31h		17h		—T	
Lily	48.60		48.18	√	8.76		—T	
LogMap	2.45		2.47		2.16		—Error	
MaasMtch	28.32		36.06		35.87		29.23h	√
MapEVO	6.77		7.44		9.96		1.25h	
MapPSO	3.05h		3.09h		3.72h		**85.98h**	
MapSSS	8.84	√	—x		4.42	√	—x	
OACAS	—Error		—Error		—Error		—Error	
OMR	—Error		—Error		—Error		—Error	
Optima	3.15h		2.48h		**88.80h**		—T	
YAM++	6.51	√	6.68	√	8.02	√	—T	

Table 3. Runtime (in minutes) based on 3 runs, and the five best systems in terms of F-measure in each data set (top-5). 'Error' indicates that the tool could not run in the current setting; or their final version has some packaging error. 'T' indicates that tool could not process the single 101 test in less than 2 hours. 'x' indicates that the tool breaks when parsing some ontologies. Results in bold face are based on only 1 run.

CSA, YAM++, AgreementMaker (AgrMaker) and MapSSS. Furthermore, as detailed in the following, AgreementMaker, CSA, CODI and YAM++ are also the best systems for most of the different data sets.

For finance, we observed that many participants were not able to deal with large ontologies. This applies to the slowest systems of the other tasks, but other problems occur with AgreementMaker and MapSSS. Fast systems like LogMap could not process some of the test cases due to the inconsistency of the finance ontology (as CODI). Finally, other relatively fast systems such as YAM++ and Lily had to time out. We plan to work on these two issues in the next campaigns.

Compliance. Concerning compliance, we focus on the benchmark2 (ekaw) data set. Table 4 shows the results of participants as well as those given by edna (simple edit distance algorithm on labels). The full results are on the OAEI web site.

As shown in Table 4, two systems achieve top performance in terms of F-measure: MapSSS and YAM++, with CODI, CSA and AgreementMaker as close followers, respectively. Lily and CIDER had presented intermediary values of precision and recall. All systems achieve a high level of precision and relatively low values of recall. Only MapEVO had a significantly lower recall than edna (with LogMap and MaasMatch (MaasMtch) with slightly lower values), while no system had lower precision.

system	**refalign**			**edna**			**AgrMaker**			**Aroma**			**CSA**			**CIDER**			**CODI**			**LDOA**		
test	Prec.	FMeas.	Rec.	Prec.	FMeas.	Rec.	Prec.	FMeas.	Rec.	Prec.	FMeas.	Rec.	Prec.	FMeas.	Rec.	Prec.	FMeas.	Rec.	Prec.	FMeas.	Rec.	Prec.	FMeas.	Rec.
1xx	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.96	1.00	.96	.25	.19	.15	.77	.86	.97
2xx	1.00	1.00	1.00	.50	.51	.51	.98	.71	.56	.93	.68	.53	.82	.72	.64	.89	.70	.58	.94	.74	.61	.51	.51	.51
H-mean	1.00	1.00	1.00	.50	.51	.52	.98	.71	.56	.93	.68	.53	.82	.73	.65	.89	.70	.58	.93	.73	.60	.51	.51	.51
Symmetric	1.00	1.00	1.00	.53	.53	.54	.98	.71	.56	.94	.68	.54	.83	.73	.66	.91	.71	.59	.94	.73	.61	.52	.52	.52
Effort	1.00	1.00	1.00	.53	.53	.55	.98	.71	.56	.94	.68	.54	.84	.73	.66	.90	.71	.59	.93	.72	.60	.52	.52	.52
P-oriented	1.00	1.00	1.00	.56	.56	.57	.98	.71	.56	.95	.68	.54	.84	.74	.67	.92	.72	.60	.95	.74	.61	.52	.52	.52
R-oriented	1.00	1.00	1.00	.56	.56	.57	.98	.71	.56	.95	.68	.54	.84	.74	.67	.92	.72	.60	.95	.74	.61	.52	.52	.52
Weighted	1.00	1.00	1.00	.71	.60	.52	.98	.71	.56	.95	.64	.49	.87	.58	.44	.91	.68	.54	.93	.73	.60	.57	.52	.48

	Lily			**LogMap**			**MaasMtch**			**MapEVO**			**MapPSO**			**MapSSS**			**YAM++**			**Optima**		
test	Prec.	FMeas.	Rec.	Prec.	FMeas.	Rec.	Prec.	FMeas.	Rec.	Prec.	FMeas.	Rec.	Prec.	FMeas.	Rec.	Prec.	FMeas.	Rec.	Prec.	FMeas.	Rec.	Prec.	FMeas.	Rec.
1xx	1.00	1.00	1.00	1.00	1.00	1.00	1.00	.99	.98	.99	1.00	1.00	.99	.96	.92	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
2xx	.93	.70	.57	.99	.66	.49	.99	.60	.43	.54	.31	.21	.63	.63	.62	.96	.77	.64	.97	.74	.60	.59	.55	.52
H-mean	.93	.70	.57	.99	.67	.50	.99	.61	.44	.55	.32	.22	.64	.63	.62	.96	.77	.64	.97	.74	.60	.60	.56	.53
Symmetric	.93	.71	.58	.99	.67	.50	.99	.61	.44	.63	.33	.22	.66	.64	.63	.97	.77	.64	.97	.74	.60	.60	.56	.53
Effort	.93	.71	.58	.99	.67	.50	.99	.61	.44	.63	.33	.23	.66	.64	.63	.97	.77	.64	.98	.74	.60	.61	.56	.53
P-oriented	.94	.71	.58	.99	.67	.50	.99	.61	.44	.64	.33	.23	.68	.66	.65	.97	.78	.65	.98	.74	.60	.61	.56	.54
R-oriented	.94	.71	.58	.99	.67	.50	.99	.61	.44	.64	.33	.23	.68	.66	.65	.97	.78	.65	.98	.74	.60	.61	.56	.54
Weighted	.95	.56	.39	.99	.55	.38	.99	.61	.44	.75	.34	.22	.76	.58	.47	.96	.77	.64	.99	.14	.07	.60	.56	.53

Table 4. Results obtained by participants on the Benchmark2 (ekaw) test case (harmonic means). Relaxed precision and recall correspond to the three measures of [4]: symmetric proximity, correction effort and oriented (precision and recall). Weighted precision and recall takes into account the confidence associated to correspondence by the matchers.

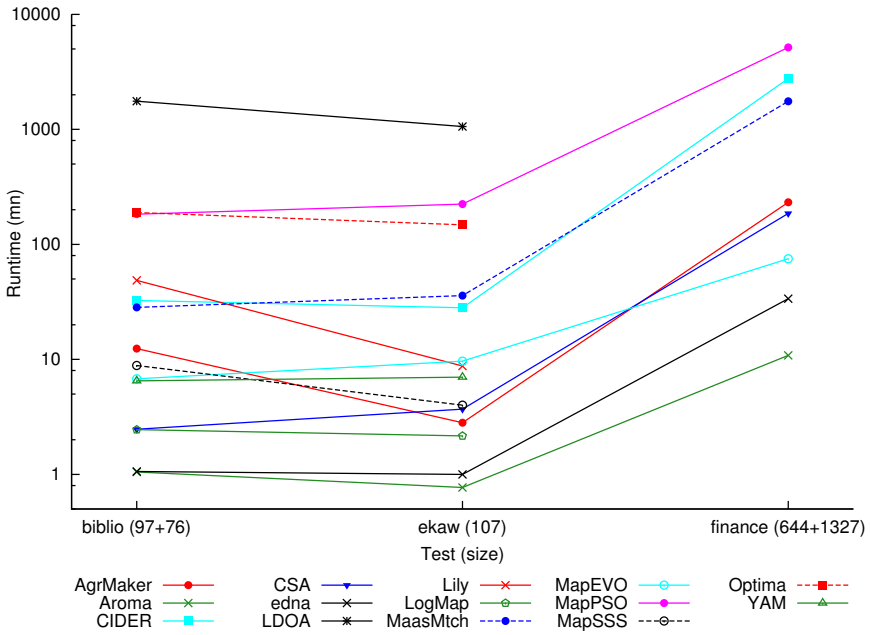

Fig. 1. Logarithmic plot of the time taken by matchers (averaged on 3 runs) to deal with different data sets: biblio, ekaw and finance

Looking at each group of tests, in simple tests (1xx) all systems have similar performance, excluding CODI. As noted in previous campaigns, the algorithms have their best score with the 1xx test series. This is because there are no modifications in the labels of classes and properties in these tests and basically all matchers are able to deal with the heterogeneity in labels. Considering that Benchmark2 has one single test in 1xx, the discriminant category is 2xx, with 101 tests. For this category, the top five systems in terms of F-measure (as stated above) are: MapSSS, YAM++, CODI, CSA and AgreementMaker, respectively (CIDER and Lily as followers).

Many algorithms have provided their results with confidence measures. It is thus possible to draw precision/recall graphs in order to compare them. Figure 2 shows the precision and recall graphs. These results are only relevant for the results of participants who provide confidence measures different from 1 or 0 (see Table 2). As last year, they show the real precision at n% recall and they stop when no more correspondences are available (then the end point corresponds to the precision and recall reported in Table 4).

The results have also been compared with the relaxed measures proposed in [4], namely symmetric proximity, correction effort and oriented measures ('Symmetric', 'Effort', 'P/R-oriented' in Table 4). Table 4 shows that these measures provide a uniform and limited improvement to most systems. As last year, the exception is MapEVO, which has a considerable improvement in precision. This could be explained by the fact this system misses the target, by not that far (the false negative correspondences found by the matcher are close to the correspondences in the reference alignment) so the gain provided by the relaxed measures has a considerable impact for this system. This may also be explained by the global optimization of the system which tends to be glob-

ally roughly correct as opposed to locally strictly correct as measured by precision and recall.

The same confidence-weighted precision and recall as last year have been computed. They reward systems able to provide accurate confidence measures (or penalizes less mistakes on correspondences with low confidence) [6]. These measures provide precision increasing for most of the systems, specially edna, MapEVO and MapPSO (which had possibly many incorrect correspondences with low confidence). This shows that the simple edit distance computed by edna is valuable as a confidence measure (the weighted precision and recall for edna could be taken as a decent baseline). It also provides recall decrease specially for CSA, Lily, LogMap, MapPSO and YAM++ (which had apparently many correct correspondences with low confidence). The variation for YAM++ is quite impressive: this is because YAM++ provides especially low confidence to correct correspondences. Some systems, such as AgreementMaker, CODI, MaasMatch and MapSSS, generate all correspondences with confidence = 1, so they have no change.

Comparison across data sets. Table 5 presents the average F-measure for 3 runs, for each data set (as Table 3, some of these results are based on only one run). These three runs are not necessary: even if matchers exhibit non deterministic behavior on a test case basis, their average F-measure on the whole data set remains the same [14]. This year, although most of the systems participating in 2010 have improved their algorithms, none of them could outperform ASMOV, the best system in the 2010 campaign.

With respect to the original benchmark data set and the new generated one (original and biblio in Table 5), we could observe a 1-2% constant and negative variation in F-measure, for most of the systems (except CODI and MapEVO). Furthermore, most of the systems perform better with the bibliographic ontology than with ekaw (a variation of 5-15%). The exceptions are LDOA, LogMap and MapPSO, followed by MaasMatch and CIDER with relatively stable F-measures. Although we have not enough results for a fair comparison with finance, we could observe that CSA and MaasMatch are the most stable matchers (with less variation than the others), followed by Aroma, CIDER and AgreementMaker, respectively.

Finally, the group of best systems in each data set remains relatively the same across the different seed ontologies. Disregarding finance, CSA, CODI and YAM++ are ahead as the best systems for all three data sets, with MapSSS (2 out of 3) and AgreementMaker, Aroma and Lily (1 out of 3) as followers.

3.3 Conclusions

For the first time, we could observe a high variation in the time matchers require to complete the alignment tasks (from some minutes to some days). We can also conclude that compliance is not proportional to runtime: the top systems in terms of F-measure were able to finish the alignment tasks in less than 15mn (with Aroma and LogMap as faster matchers, with intermediary levels of compliance). Regarding the capability of dealing with large ontologies, many of the participants were not able to process them, leaving room for further improvement on this issue.

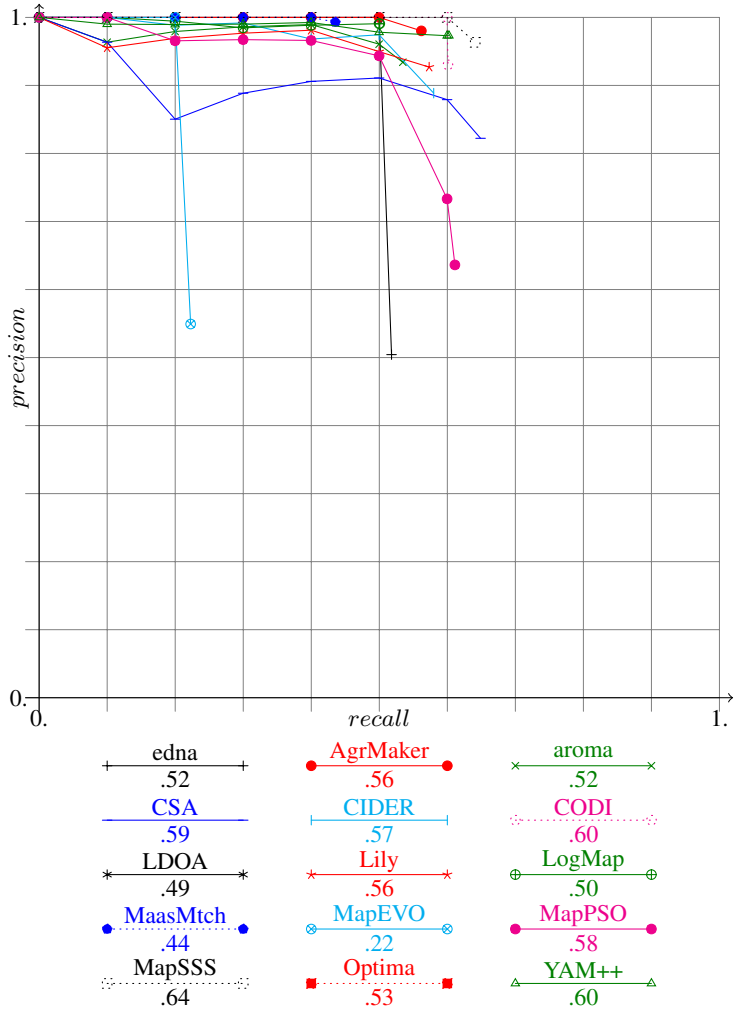

Fig. 2. Precision/recall graphs for benchmarks. The alignments generated by matchers are cut under a threshold necessary for achieving $n\%$ recall and the corresponding precision is computed. The numbers in the legend are the Mean Average Precision (MAP): the average precision for each correct retrieved correspondence. Systems for which these graphs are not meaningful (because they did not provide graded confidence values) are drawn in dashed lines.

	2010		2011							
	original		**original**		**biblio**		**ekaw**		**finance**	
System	Fmeas.	Top-5	Fmeas.	Top-5	Fmeas.	Top-5	Fmeas.	Top-5	Fmeas.	Top-5
ASMOV	.93	√								
AgrMaker	.89	√	.88	√	x		.71		.78	
Aroma	.59		.78		.76	√	.68		.70	√
CSA			.84	√	.83	√	.73	√	.79	√
CIDER			.76		.74		.70		.67	√
CODI	.55		.80	√	.75	√	.73	√	x	
edna	.51		.52		.51		.51		.50	
LDOA			.47		.46		.52		T	
Lily			.76		.77	√	.70		T	
LogMap			.60		.57		.66		x	
MaasMtch			.59		.58		.61		.61	√
MapEVO			.41		.37		.33		.20	
MapPSO	.61		.50		.48		.63		.14	
MapSSS			.84	√	x		.78	√	T	
Optima			.64		.65		.56		T	
YAM++			.87	√	.86	√	.75	√	T	

Table 5. Results obtained by participants on each data set (based on 94 tests), including the results from the participants in 2010, and the top-five F-measure (five better systems in each data set).

With respect to compliance, newcomers (CSA, YAM++ and MapSSS) have mostly outperformed other participants, for the new generated benchmarks. On the other hand, for the very known original benchmark data set, none of the systems was able to outperform the top-performer of the last year (ASMOV).

4 Anatomy

As in the previous years, the anatomy track confronts the existing matching technology with a specific type of ontologies from the biomedical domain. In this domain, many ontologies have been built covering different aspects of medical research. We focus on fragments of two biomedical ontologies which describe the human anatomy and the anatomy of the mouse. The data set of this track has been used since 2007. For a detailed description we refer the reader to the OAEI 2007 results paper [7].

4.1 Experimental setting

Contrary to the previous years, we distinguish only between two evaluation experiments. Subtask #1 is about applying a matcher with its standard setting to the matching task. In the previous years we have also asked for additional alignments that favor precision over recall and vice versa (subtask #2 and #3). These subtasks are not part of the anatomy track in 2011 due to the fact that the SEALS platform does not allow for running tools with different configurations. Furthermore, we have proposed a fourth subtask, in which a partial reference alignment has to be used as an additional input.

In our experiments we compare precision, recall, F-measure and recall+. We have introduced recall+ to measure the amount of detected non-trivial correspondences. From 2007 to 2009, we reported on runtimes measured by the participants themselves. This survey revealed large differences in runtimes. This year we can compare the runtimes of participants by executing them on our own on the same machine. We used a Windows 2007 machine with 2.4 GHz (2 cores) and 7GB RAM allocated to the matching systems.

For the 2011 evaluation, we improved again the reference alignment of the data set. We removed doubtful correspondences and included several correct correspondences that had not been included in the past. As a result, we measured for the alignments generated in 2010 a slightly better F-measure ($\approx$+1%) compared to the computation based on the old reference alignment. For that reason we have also included the top-3 systems of 2010 with recomputed precision/recall scores.

4.2 Results

In the following we analyze the robustness of the submitted systems and their runtimes. Further, we report on the quality of the generated alignment, mainly in terms of precision and recall.

Robustness and scalability. In 2011 there were 16 participants in the SEALS modality, while in 2010 we had only 9 participants for the anatomy track. However, this comparison is misleading. Some of these 16 systems are not really intended to match large biomedical ontologies. For that reason our first interest is related to the question, which systems generate a meaningful result in an acceptable time span. Results are shown in Table 6. First, we focused on the question whether systems finish the matching task in less than 24h. This is the case for a surprisingly low number of systems. The systems

that do not finish in time can be separated in those systems that throw an exception related to insufficient memory after some time (marked with 'X'). The other group of systems were still running when we stopped the experiments after 24 hours (marked with 'T').[8]

Obviously, matching relatively large ontologies is a problem for five out of fourteen executable systems. The two systems MapPSO and MapEVO can cope with ontologies that contain more than 1000 concepts, but have problems with finding correct correspondences. Both systems generate comprehensive alignments, however, MapPSO finds only one correct corespondence and MapEVO finds none. This can be related to the way labels are encoded in the ontologies. The ontologies from the anatomy track differ from the ontologies of the benchmark and conference tracks in this respect.

Matcher	Runtime	Size	Precision	F-measure	Recall	Recall+
AgrMaker	634	1436	.943	.917	.892	.728
LogMap	24	1355	.948	.894	.846	.599
AgrMaker_{2010}	-	1436	.914	.890	.866	.658
CODI	1890	1298	.965	.889	.825	.564
NBJLM_{2010}	-	1327	.931	.870	.815	.592
Ef2Match_{2010}	-	1243	.965	.870	.792	.455
Lily	563	1368	.814	.772	.734	.511
StringEquiv	-	*934*	*.997*	*.766*	*.622*	*.000*
Aroma	39	1279	.742	.679	.625	.323
CSA	4685	2472	.465	.576	.757	.595
MaasMtch	66389	438	.995	.445	.287	.003
MapPSO	9041	2730	.000	.000	.001	.000
MapEVO	270	1079	.000	.000	.000	.000
Cider	T	0	-	-	-	-
LDOA	T	0	-	-	-	-
MapSSS	X	0	-	-	-	-
Optima	X	0	-	-	-	-
YAM++	X	0	-	-	-	-

Table 6. Comparison against the reference alignment, runtime is measured in seconds, the size column refers to the number of correspondences in the generated alignment.

For those systems that generate an acceptable result, we observe a high variance in measured runtimes. Clearly ahead is the system LogMap (24s), followed by Aroma (39s). Next are Lily and AgreementMaker (approx. 10mn), CODI (30mn), CSA (1h15), and finally MaasMatch (18h).

Results for subtask #1. The results of our experiments are also presented in Table 6. Since we have improved the reference alignment, we have also included recomputed precision/recall scores for the top-3 alignments submitted in 2010 (marked by subscript 2010). Keep in mind that in 2010 AgreementMaker (AgrMaker) submitted an alignment that was the best submission to the OAEI anatomy track compared to all previous

[8] We could not execute the two systems OACAS and OMR, not listed in the table, because the required interfaces have not been properly implemented.

submissions in terms of F-measure. Note that we also added the base-line *StringEquiv*, which refers to a matcher that compares the normalized labels of two concepts. If these labels are identical, a correspondence is generated. Recall+ is defined as recall, with the difference that the reference alignment is replaced by the set difference of $R \setminus A_{SE}$, where A_{SE} is defined as the alignment generated by *StringEquiv*.

This year we have three systems that generate very good results, namely AgreementMaker, LogMap and CODI. The results of LogMap and CODI are very similar. Both systems manage to generate an alignment with F-measure close to the 2010 submission of AgreementMaker. LogMap is slightly ahead. However, in 2011 the alignment generated by AgreementMaker is even better than in the previous year. In particular, AgreementMaker finds more correct correspondences, which can be seen in recall as well as in recall+ scores. At the same time, AgreementMaker can increase its precision. Also remarkable are the good results of LogMap, given the fact that the system finishes the matching task in less than half a minute. It is thus 25 times faster than AgreementMaker and more than 75 times faster than CODI.

Lily, Aroma, CSA, and MaasMatch (MaasMatch) have less good results than the three top matching systems, however, they have proved to be applicable to larger matching tasks and can generate acceptable results for a pair of ontologies from the biomedical domain. While these systems cannot (or barely) top the String-Equivalence baseline in terms of F-measure, they manage, nevertheless, to generate many correct non-trivial correspondences. A detailed analysis of the results revealed that they miss at the same time many trivial correspondences. This is an uncommon result, which might, for example, be related to some pruning operations performed during the comparison of matchable entities. An exception is the MaasMatch system. It generates results that are highly similar to a subset of the alignment generated by the *StringEquiv* baseline.

Using an input alignment. This specific task was known as subtask #4 in the previous OAEI campaigns. Originally, we planned to study the impact of different input alignments of varying size. The idea is that a partial input alignment, which might have been generated by a human expert, can help the matching system to find missing correspondences. However, taking into account only those systems that could generate a meaningful alignment in time, only AgreementMaker, implemented the required interface. Thus, a comparative evaluation is not possible. We may have to put more effort in advertising this specific subtask for the next OAEI.

Alignment coherence. This year we also evaluated alignment coherence. The anatomy data set contains only a small amount of disjointness statements, the ontologies under discussion are in $\mathcal{EL}$++. Thus, even simple techniques might have an impact on the coherence of the generated alignments. For the anatomy data set the systems LogMap, CODI, and MaasMatch generate coherent alignments. The first two systems put a focus on alignment coherence and apply special methods to ensure coherence. MaasMatch has generated a small, highly precise, and coherent alignment. The alignments generated by the other systems are incoherent. A more detailed analysis related to alignment coherence is conducted for the alignments of the conference data set in Section 5.

4.3 Conclusions

Less than half of the systems generate good or at least acceptable results for the matching task of the anatomy track. With respect to those systems that failed on anatomy, we can assume that this track was not in the focus of their developers. This means at the same time that many systems are particularly designed or configured for matching tasks that we find in the benchmark and conference tracks. Only few of them are robust "all-round" matching systems that are capable of solving different tasks without changing their settings or algorithms.

The positive results of 2011 are the top results of AgreementMaker and the runtime performance of LogMap. AgreementMaker generated a very good result by increasing precision and recall compared to its last years submissions, which was the best submission in 2010 already. LogMap clearly outperforms all other systems in terms of runtimes and still generates good results. We refer the reader to the OAEI papers of these two systems for details on the algorithms.

5 Conference

The conference test case introduces matching several moderately expressive ontologies. Within this track, participant results were evaluated using diverse evaluation methods. As last year, the evaluation has been supported by the SEALS platform.

5.1 Test data

The collection consists of sixteen ontologies in the domain of organizing conferences. Ontologies have been developed within the OntoFarm project[9].

The main features of this test case are:

- *Generally understandable domain.* Most ontology engineers are familiar with organizing conferences. Therefore, they can create their own ontologies as well as evaluate the alignments among their concepts with enough erudition.
- *Independence of ontologies.* Ontologies were developed independently and based on different resources, they thus capture the issues in organizing conferences from different points of view and with different terminologies.
- *Relative richness in axioms.* Most ontologies were equipped with OWL DL axioms of various kinds; this opens a way to use semantic matchers.

Ontologies differ in numbers of classes, of properties, in expressivity, but also in underlying resources. Ten ontologies are based *on tools* supporting the task of organizing conferences, two are based on experience of people with *personal participation* in conference organization, and three are based on *web pages* of concrete conferences.

Participants were asked to provide all correct correspondences (equivalence and/or subsumption correspondences) and/or 'interesting correspondences' within the conference data set.

[9] http://nb.vse.cz/~svatek/ontofarm.html

5.2 Results

This year, we provided results in terms of F_2-measure and $F_{0.5}$-measure, comparison with two baseline matchers and precision/recall triangular graph.

Evaluation based on the reference alignments. We evaluated the results of participants against reference alignments. They include all pairwise combinations between 7 different ontologies, i.e. 21 alignments.

Matcher	Prec.	$F_{0.5}$Meas.	Rec.	Prec.	F_1Meas.	Rec.	Prec.	F_2Meas.	Rec.
YAM++	.8	.73	.53	.78	**.65**	.56	.78	.59	.56
CODI	.74	.7	.57	.74	.64	.57	.74	.6	.57
LogMap	.85	**.75**	.5	.84	.63	.5	.84	.54	.5
AgrMaker	.8	.69	.44	.65	.62	.59	.58	**.61**	.62
$BaseLine_2$	**.79**	**.7**	**.47**	**.79**	**.59**	**.47**	**.79**	**.51**	**.47**
MaasMtch	.83	.69	.42	.83	.56	.42	.83	.47	.42
$BaseLine_1$	**.8**	**.68**	**.43**	**.8**	**.56**	**.43**	**.8**	**.47**	**.43**
CSA	.61	.58	.47	.5	.55	.6	.5	.58	.6
CIDER	.67	.61	.44	.64	.53	.45	.38	.48	.51
MapSSS	.55	.53	.47	.55	.51	.47	.55	.48	.47
Lily	.48	.42	.27	.36	.41	.47	.37	.45	.47
AROMA	.35	.37	.46	.35	.4	.46	.35	.43	.46
Optima	.25	.28	.57	.25	.35	.57	.25	.45	.57
MapPSO	.28	.25	.17	.21	.23	.25	.12	.26	.36
LDOA	.1	.12	.56	.1	.17	.56	.1	.29	.56
MapEVO	.27	.08	.02	.15	.04	.02	.02	.02	.02

Table 7. The highest average $F_{[0.5|1|2]}$-measure and their corresponding precision and recall for some threshold for each matcher.

For better comparison, we evaluated alignments with regard to three different average[10] F-measures independently. We used $F_{0.5}$-measure (where $\beta = 0.5$) which weights precision higher than recall, F_1-measure (the usual F-measure, where $\beta = 1$), which is the harmonic mean of precision and recall, and F_2-measure (for $\beta = 2$) which weights recall higher than precision. For each of these F-measures, we selected a global confidence threshold that provides the highest average $F_{[0.5|1|2]}$-measure. Results of these three independent evaluations[11] are provided in Table 7.

Matchers 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 which were lowercased before. $Baseline_2$ enhances $baseline_1$ with three string operations: removing of dashes, underscores and "has" words from all local names. These two baselines divide matchers into four groups. Group 1 consists of best matchers (YAM++, CODI, LogMap and AgreementMaker) having better results than $baseline_2$ in terms of F_1-measure. Matchers which perform worse than $baseline_2$ in terms of F_1-measure but still better than

[10] Computed using the absolute scores, i.e. number of true positive examples.

[11] Precision and recall can be different in all three cases.

$baseline_1$ are in Group 2 (MaasMatch). Group 3 (CSA, CIDER and MapSSS) contains matchers which are better than $baseline_1$ at least in terms of F_2-measure. Other matchers (Lily, Aroma, Optima, MapPSO, LDOA and MapEVO) perform worse than $baseline_1$ (Group 4). Optima, MapSSS and CODI did not provide graded confidence values. Performance of matchers regarding F_1-measure is visualized in Figure 3.

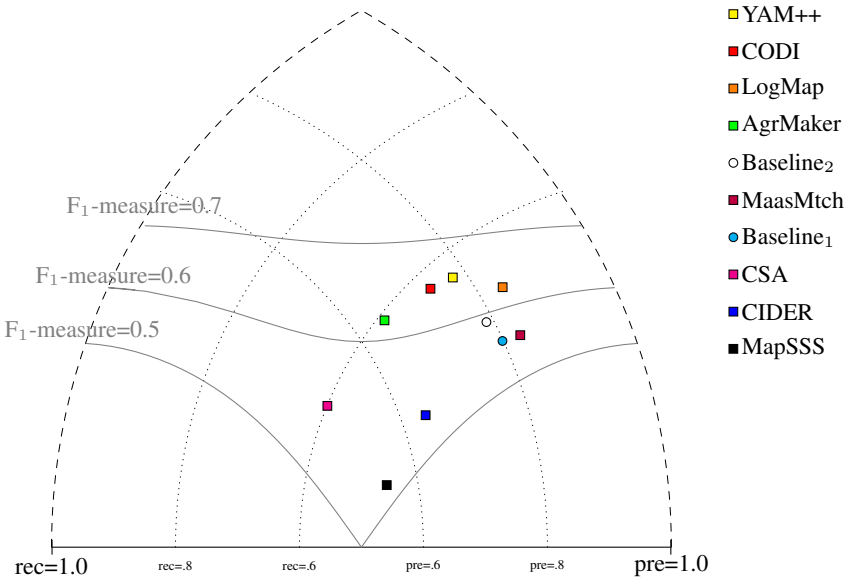

Fig. 3. Precision/recall triangular graph for conference. Matchers of participants from the first three groups are represented as squares. Baselines are represented as circles. Dotted lines depict level of precision/recall while values of F_1-measure are depicted by areas bordered by corresponding lines F_1-measure=0.[5|6|7].

In conclusion, all best matchers (group one) are very close to each other. However, the matcher with the highest average F_1-measure (.65) is YAM++, the highest average F_2-measure (.61) is AgreementMaker and the highest average $F_{0.5}$-measure (.75) is LogMap. In any case, we should take into account that this evaluation has been made over a subset of all possible alignments (one fifth).

Comparison with previous years. Three matchers also participated in the previous year. AgreementMaker improved its average F_1-measure from .58 to .62 by higher precision (from .53 to .65) and lower recall (from .62 to .59), CODI increased its average F_1-measure from .62 to .64 by higher recall (from .48 to .57) and lower precision (from .86 to .74). AROMA (with its AROMA- variant) slightly decreased its average F_1-measure from .42 to .40 by lower precision (from .36 to .35) and recall (from .49 to .46).

Evaluation based on alignment coherence. As in the previous years, we apply the Maximum Cardinality measure proposed in [13] to measure the degree of alignment

incoherence. Details on the algorithms can be found in [12]. The reasoner underlying our implementation is Pellet [16].

The results of our experiments are depicted in Table 8. It shows the average for all test cases of the conference track, which covers more than the ontologies that are connected via reference alignments. We had to omit the test cases in which the ontologies `Confious` and `Linklings` are involved as source or target ontologies. These ontologies resulted in many cases in reasoning problems. Thus, we had 91 test cases for each matching system. However, we faced reasoning problems for some combinations of test cases and alignments. In this case we computed the average score by ignoring these test cases. These problems occurred mainly for highly incoherent alignments. The last row in Table 8 informs about the number of test cases that were excluded. Note that we did not analyze the results of those systems that generated alignments with precision less than .25.

Matcher	AgrMaker	AROMA	CIDER	CODI	CSA	Lily	LogMap	MaasMtch	MapSSS	Optima	YAM
Size	13.9	14.1	17.9	9.5	50.8	17	8	7.5	10	31.3	10.1
Inc. Alignments	49/90	58/88	69/88	0/91	69/69	70/90	8/91	21/91	51/90	73/84	41/91
Degree of Inc.	12%	16%	13%	0%	>29%	14%	2%	4%	9%	>31%	7%
Reasoning problems	1	3	3	0	22	1	0	0	1	7	0

Table 8. Average size of alignments, number of incoherent alignments, and average degree of incoherence. The prefix $>$ is added if the search algorithm stopped in one of the testcases due to a timeout of 10min prior to computing the degree of incoherence.

CODI is the only system that guarantees the coherence of the generated alignments. While last year some of the alignments were incoherent, all of the alignments generated in 2011 are coherent. LogMap, a system with special focus on alignment coherence and efficiency [11], generates in most cases coherent alignments. A closer look at the outliers reveals that all incoherent alignments occured for ontology pairs where the ontology `Cocus` was involved. This ontology suffers from a very specific modeling error based on the inappropriate use of universal quantification. At the third position we find MaasMatch. MaasMatch generates less incoherent alignments than the remaining systems. This might be related to the high precision of the system. Contrary to LogMap, incoherent alignments are generated for different combinations of ontologies and there is no specific pattern emerging.

It is not easy to interpret the results of the remaining matching systems due to the different sizes of the alignments that they have generated. The more correspondences are contained in an alignment, the higher is the probability that this results in a concepts unsatisfiability. It is not always clear whether a relatively low/high degree of incoherence is mainly caused by the small/large size of the alignments, or related to the use of a specific technique. Overall, we conclude that alignment coherence is not taken into account by these systems. However, in 2011 we have at least some systems that apply specific methods to ensure coherence for all or at least for a large subset of generated alignments. Compared to the previous years, this is a positive result of our analysis.

6 Instance matching

The goal of the instance matching track is to evaluate the performance of different matching tools on the task of matching RDF individuals which originate from different sources but describe the same real-world entity. Data interlinking is known under many names according to various research communities: equivalence mining, record linkage, object consolidation and coreference resolution to mention the most used ones. In each case, these terms are used for the task of finding equivalent entities in or across data sets. As the quantity of data sets published on the Web of data dramatically increases, the need for tools helping to interlink resources becomes more critical. It is particularly important to maximize the automation of the interlinking process in order to be able to follow this expansion.

Unlike the other tracks, the instance matching tests specifically focus on an ontology ABox. However, the problems which have to be resolved in order to correctly match instances can originate at the schema level (use of different properties and classification schemas) as well as at the data level, e.g., different formats of values. This year, the track included two tasks. The first task, data interlinking (DI), aims at testing the performance of tools on large-scale real-world data sets published according to the linked data principles. The second one (IIMB) uses a set of artificially generated and real test cases respectively. These are designed to illustrate all common cases of discrepancies between individual descriptions (different value formats, modified properties, different classification schemas). The list of participants to the instance matching track is shown in Table 9.

Dataset	**AgrMaker**	**SERIMI**	**Zhishi**	**CODI**
DI-nyt-dbpedia-locations	√	√	√	
DI-nyt-dbpedia-organizations	√	√	√	
DI-nyt-dbpedia-people	√	√	√	
DI-nyt-freebase-locations	√	√	√	
DI-nyt-freebase-organizations	√	√	√	
DI-nyt-freebase-people	√	√	√	
DI-nyt-geonames	√	√	√	
IIMB				√

Table 9. Participants in the instance matching track.

6.1 Data interlinking task (DI) – New York Times

This year the data interlinking task consists of matching the New York Times subject headings to DBpedia, Freebase and Geonames. The New York Times has developed over the past 150 years an authoritative vocabulary for annotating news items. The vocabulary contains about 30,000 subject headings, or tags. They are progressively published as linked open data and, by July 2010, over 10,000 of these subject headings, in the categories People, Organizations, Locations and Descriptors, have been published[12].

[12] http://data.nytimes.com/

The New York Times data set was used in OAEI 2010 track on very large crosslingual resources.

The reference alignments are extracted from the links provided and curated by The New-York Times. However, the set of reference links has been updated to reflect the changes made to the external data sets during the year. In particular, several missing links were added, links pointing to non-existing DBPedia instances were removed, and links to instances redirecting to others were updated. Moreover, the Descriptors facet has been removed from the evaluation, since there was not a clear identity criterion for its instances.

Facet	**# Concepts**	**Links to Freebase**	**Links to DBPedia**	**Links to Geonames**
People	4,979	4,979	4,977	0
Organizations	3,044	3,044	1,965	0
Locations	1,920	1,920	1,920	1,920

Table 10. Number of links between the New-York Times corpus and other data sources.

Subject heading facets are represented in SKOS. Each subject heading facet contains the label of the skos:Concept (skos:label), the facet it belongs to (skos:inScheme), and some specific properties: nyt:associated_article_count for the number of NYT articles the concept is associated with and nyt:topicPage pointing to the topic page (in HTML) gathering different information published on the subject. The Location facet also contains geo-coordinates. The concepts have links to DBpedia, Freebase and/or GeoNames.

	AgreementMaker			SERIMI			Zhishi.links		
Dataset	Prec.	FMeas.	Rec.	Prec.	FMeas.	Rec.	Prec.	FMeas.	Rec.
DI-nyt-dbpedia-loc.	.79	.69	.61	.69	.68	.67	.92	.92	.91
DI-nyt-dbpedia-org.	.84	.74	.67	.89	.88	.87	.90	.91	.93
DI-nyt-dbpedia-peo.	.98	.88	.80	.94	.94	.94	.97	.97	.97
DI-nyt-freebase-loc.	.88	.85	.81	.92	.91	.90	.90	.88	.86
DI-nyt-freebase-org.	.87	.80	.74	.92	.91	.89	.89	.87	.85
DI-nyt-freebase-peo.	.97	.96	.95	.93	.92	.91	.93	.93	.92
DI-nyt-geonames.	.90	.85	.80	.79	.80	.81	.94	.91	.88
H-mean.	.92	.85	.80	.89	.89	.88	.93	.92	.92

Table 11. Results of the DI subtrack.

DI results. An overview of the precision, recall and F_1-measure results per data set of the DI subtrack is shown in Table 11. A precision-recall graph visualization is shown in Figure 4. The results show a variation in both systems and data sets. Zhishi.links produces consistently high quality matches over all data sets, and obtains the highest overall scores. Matches to DBpedia locations (DI-nyt-dbpedia-loc.) appear to be difficult as AgreementMaker and SERIMI perform poorly on both precision and recall. This is not the case for Freebase locations (DI-nyt-freebase-loc.) and to a much lesser extent for Geonames (DI-nyt-geonames). We hypthesize that this is due to many locations not being present in DBPedia. Agreementmaker's scores considerably higher on People

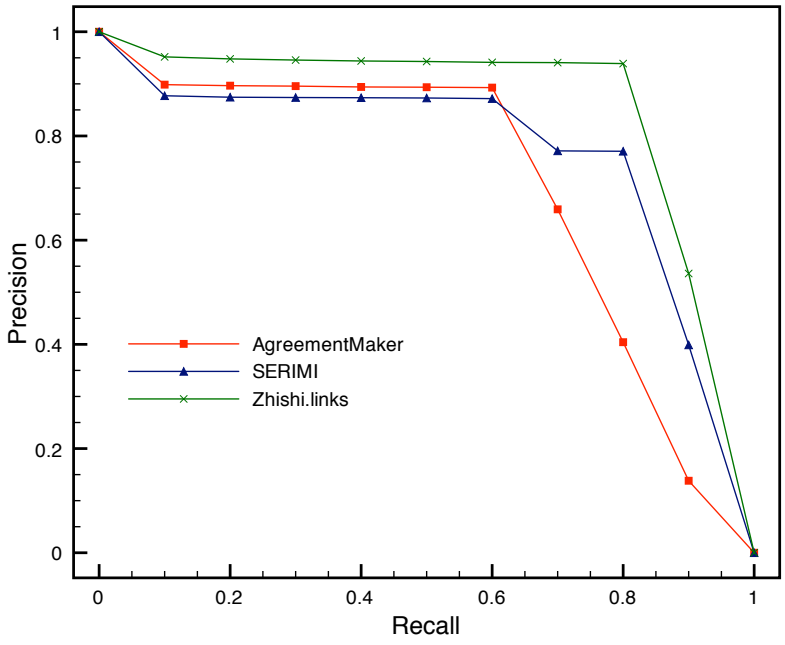

Fig. 4. Precision/recall of tools participating in the DI subtrack.

than on Locations and Organizations, which can be observed in both the DBpedia and the Freebase data set.

6.2 OWL data task (IIMB)

The OWL data task is focused on two main goals:

1. to provide an evaluation data set for various kinds of data transformations, including value transformations, structural transformations and logical transformations;
2. to cover a wide spectrum of possible techniques and tools.

To this end, we provided the ISLab Instance Matching Benchmark (IIMB). Participants were requested to find the correct correspondences among individuals of the first knowledge base and individuals of the other one. An important task here is that some of the transformations require automatic reasoning for finding the expected alignments.

IIMB is composed of a set of test cases, each one represented by a set of instances, i.e., an OWL ABox, built from an initial data set of real linked data extracted from the web. Then, the ABox is automatically modified in several ways by generating a set of new ABoxes, called *test cases*. Each test case is produced by transforming the individual descriptions in the reference ABox in new individual descriptions that are inserted in the test case at hand. The goal of transforming the original individuals is twofold: on one side, we provide a simulated situation where data referring to the same objects are provided in different data sources; on the other side, we generate different data sets

with a variable level of data quality and complexity. IIMB provides transformation techniques supporting modifications of data property values, modifications of number and type of properties used for the individual description, and modifications of the individuals classification. The first kind of transformations is called *data value transformation* and it aims at simulating the fact that data expressing the same real object in different data sources may be different because of data errors or because of the usage of different conventional patterns for data representation. The second kind of transformations is called *data structure transformation* and it aims at simulating the fact that the same real object may be described using different properties/attributes in different data sources. Finally, the third kind of transformations, called *data semantic transformation*, simulates the fact that the same real object may be classified in different ways in different data sources.

The 2011 edition of IIMB is created by extracting data from Freebase, an open knowledge base that contains information about 11 million real objects including movies, books, TV shows, celebrities, locations, companies and more. Data extraction has been performed using the query language JSON together with the Freebase JAVA API[13]. IIMB2011 is a collection of OWL ontologies consisting of 29 concepts, 20 object properties, 12 data properties and more than 4000 individuals divided into 80 test cases.

Test cases from 0 to 20 contain changes in data format (misspelling, errors in text, etcetera); test cases 21 to 40 contain changes in structure (properties missing, RDF triples changed); 41 to 60 contain logical changes (class membership changed, logical errors); finally, test cases 61 to 80 contain a mix of the previous. One system, CODI, participated in this task. Its results (Table 5) show how precision drops moderately and recall drops dramatically as more errors are introduced.

IIMB results An overview of the precision, recall and F_1-measure results per set of tests of the IIMB subtrack is shown in Table 5. A precision-recall graph visualization is shown in Figure 6.

	codi		
test	Prec.	FMeas.	Rec.
001–010	.94	.84	.76
011–020	.94	.87	.81
021–030	.89	.79	.70
031–040	.83	.66	.55
041–050	.86	.72	.62
051–060	.83	.72	.64
061–070	.89	.59	.44
071–080	.73	.33	.21

Fig. 5. Results of the IIMB subtrack.

[13] `http://code.google.com/p/freebase-java/`

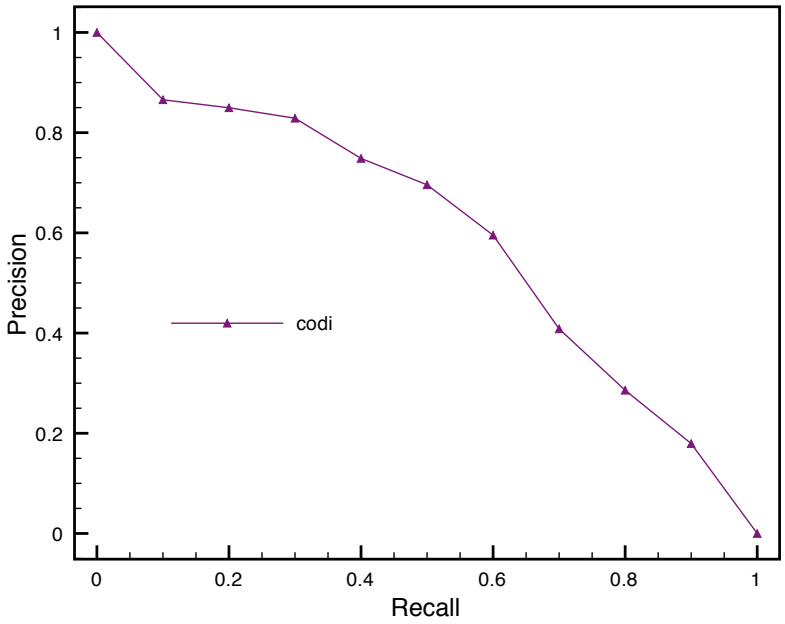

Fig. 6. Precision/recall of the CODI tool participating in the IIMB subtrack.

7 Lesson learned and suggestions

This year we implemented most of our 2010 future plans by providing a common platform on which evaluation could be performed. There still remains one lesson not really taken into account that we identify with an asterisk (*) and that we will tackle in the coming months. The main lessons from this year are:

A) This year again indicated that requiring participants to implement a minimal interface was not a strong obstacle to participation. The interface allows for comparing matchers on the same or similar hardware. It also allows for running more tests or reproducing results without running a new campaign.
B) By using the SEALS platform, we have eliminated the network issue that we had last year with web services and we can better testify the portability of tools.
C) The client available for testing and evaluating wrapped tools was intensively used by participants to test and improve their systems. So, interoperability and the ability to get immediate feedback was appreciated by the tool developers. Moreover, participants could use the client to generate preliminary results to be included in their papers.
D) There is a high variance in runtimes and there seems to be no correlation between runtime and quality of the generated results.
*E) The low number of systems that could generate results for the Anatomy track is an uncommon result. It seems that not many matching systems of this year are capable of matching large ontologies (>1000 entities). Even if we had introduced new benchmark generation facilities, we have not used it towards scalability benchmarks. We plan to address this in the next few months.

F) Last years we reported that there are not many new systems entering the competition. This year we had many new participants. Only a minority of systems participated in one of the previous years.
G) Two systems have not fully respected the OAEI rules. YAM++ used a setting learned from the reference alignments of the 2009 benchmark data set. Due to the fact that we run benchmarks also with newly generated tests, we decided to keep the YAM++ results with this warning. AgreementMaker used a specific setting to distinguish between Benchmarks, Anatomy and Conference. As AgreementMaker has improved its results over the last year, we decide to report on them as well. For the next campaigns we plan to be more attentive on these aspects.
H) In spite of claims that such evaluations were needed, we had to declare the model matching and oriented alignments tracks unfruitful. It is a pity. This confirms that setting up a data set is not sufficient for attracting participants.
I) More surprising, there are only a few matchers participating in the instance matching track. This is especially surprising given the high number of papers submitted and published on this topic nowadays. It seems that people involved in instance matching should cooperate to propose standard formats and evaluation modalities that everyone would use.

8 Future plans

In 2012, for logistic reasons, we plan to have an intermediate evaluation before OAEI-2012. This evaluation will concentrate on exploiting fully the SEALS platform and, in particular on:

- performing benchmark scalability tests by reducing randomly a large seed ontology;
- generating discriminating benchmarks by suppressing easy tests;
- adding new tasks, such as multilingual resources, on the SEALS platform.

We plan to run these tests within the next six months with the already registered tools that would like to be evaluated as well as with new tools willing to enter. These partial results will be integrated within the results of OAEI-2012.

9 Conclusions

The trend of the previous years, the number of systems and tracks they participate in, seem to stabilize. The average number of tracks entered by participants in 2011 (2.9) is above that of 2010 (2.6). This number is dominated by the use of the SEALS platform: each tool entering there can be evaluated on three tasks. It does not invalidate last year's remark that tools may be more specialized.

This year, systems did not deliver huge improvements in performance with respect to last year's performers which did not participate. However, AgreementMaker improved its results of last year to become one of the top performer. In addition, we have been able to test runtime and consistency of the tested matchers and noticed ample differences between systems. This may become a differentiating feature among matchers.

All participants have provided a description of their systems and their experience in the evaluation. These OAEI papers, like the present one, have not been peer reviewed. However, they are full contributions to this evaluation exercise and reflect the hard work and clever insight people put in the development of participating systems. Reading the papers of the participants should help people involved in ontology matching to find what makes these algorithms work and what could be improved. Sometimes participants offer alternate evaluation results.

The Ontology Alignment Evaluation Initiative will continue these tests by improving both test cases and testing methodology for being more accurate. Further information can be found at:

`http://oaei.ontologymatching.org`.

Acknowledgments

We warmly thank the participants of this campaign. We know that they have worked hard for having their matching tools executable in time and they provided insightful papers presenting their experience. The best way to learn about the results remains to read the following papers.

We are grateful to Dominique Ritze for participating in the extension of the reference alignments for the conference track.

We thank Jan Noessner for providing data in the process of constructing the IIMB data set. We are grateful to Martin Ringwald and Terry Hayamizu for providing the reference alignment for the anatomy ontologies and thank Elena Beisswanger for her thorough support on improving the quality of the data set.

We also thank the other members of the Ontology Alignment Evaluation Initiative steering committee: Yannis Kalfoglou (Ricoh laboratories, UK), Miklos Nagy (The Open University (UK), Natasha Noy (Stanford University, USA), Yuzhong Qu (Southeast University, CN), York Sure (Leibniz Gemeinschaft, DE), Jie Tang (Tsinghua University, CN), George Vouros (University of the Aegean, GR).

Jérôme Euzenat, Christian Meilicke, Heiner Stuckenschmidt and Cássia Trojahn dos Santos have been partially supported by the SEALS (IST-2009-238975) European project.

Ondřej Šváb-Zamazal has been supported by the CSF grant P202/10/0761.

References

1. Benhamin Ashpole, Marc Ehrig, Jérôme Euzenat, and Heiner Stuckenschmidt, editors. *Proc. of the K-Cap Workshop on Integrating Ontologies*, Banff (Canada), 2005.
2. Caterina Caracciolo, Jérôme Euzenat, Laura Hollink, Ryutaro Ichise, Antoine Isaac, Véronique Malaisé, Christian Meilicke, Juan Pane, Pavel Shvaiko, Heiner Stuckenschmidt, Ondrej Sváb-Zamazal, and Vojtech Svátek. Results of the ontology alignment evaluation initiative 2008. In *Proc. 3rd International Workshop on Ontology Matching (OM) collocated with ISWC*, pages 73–120, Karlsruhe (Germany), 2008.
3. Jérôme David, Jérôme Euzenat, François Scharffe, and Cássia Trojahn dos Santos. The alignment api 4.0. *Semantic web journal*, 2(1):3–10, 2011.
4. Marc Ehrig and Jérôme Euzenat. Relaxed precision and recall for ontology matching. In *Proc. of the K-Cap Workshop on Integrating Ontologies*, pages 25–32, Banff (Canada), 2005.

5. Jérôme Euzenat, Alfio Ferrara, Laura Hollink, Antoine Isaac, Cliff Joslyn, Véronique Malaisé, Christian Meilicke, Andriy Nikolov, Juan Pane, Marta Sabou, François Scharffe, Pavel Shvaiko, Vassilis Spiliopoulos, Heiner Stuckenschmidt, Ondrej Sváb-Zamazal, Vojtech Svátek, Cássia Trojahn dos Santos, George Vouros, and Shenghui Wang. Results of the ontology alignment evaluation initiative 2009. In *Proc. 4th Workshop on Ontology Matching (OM) collocated with ISWC*, pages 73–126, Chantilly (USA), 2009.
6. Jérôme Euzenat, Alfio Ferrara, Christian Meilicke, Andriy Nikolov, Juan Pane, François Scharffe, Pavel Shvaiko, Heiner Stuckenschmidt, Ondrej Sváb-Zamazal, Vojtech Svátek, and Cássia Trojahn dos Santos. Results of the ontology alignment evaluation initiative 2010. In Pavel Shvaiko, Jérôme Euzenat, Fausto Giunchiglia, Heiner Stuckenschmidt, Ming Mao, and Isabel Cruz, editors, *Proc. 5th ISWC workshop on ontology matching (OM) collocated with ISWC, Shanghai (China)*, pages 85–117, 2010.
7. 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 *Proc. 2nd International Workshop on Ontology Matching (OM) collocated with ISWC*, pages 96–132, Busan (Korea), 2007.
8. 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.
9. Jérôme Euzenat, Malgorzata Mochol, Pavel Shvaiko, Heiner Stuckenschmidt, Ondrej Svab, Vojtech Svatek, Willem Robert van Hage, and Mikalai Yatskevich. Results of the ontology alignment evaluation initiative 2006. In *Proc. 1st International Workshop on Ontology Matching (OM) collocated with ISWC*, pages 73–95, Athens, Georgia (USA), 2006.
10. Jérôme Euzenat and Pavel Shvaiko. *Ontology Matching*. Springer, Heidelberg (DE), 2007.
11. Ernesto Jiménez-Ruiz and Bernardo Cuenca Grau. Logmap: Logic-based and scalable ontology matching. In *Proc. 10th International Semantic Web Conference (ISWC)*, pages 273–288, 2011.
12. Christian Meilicke. *Alignment Incoherence in Ontology Matching*. PhD thesis, University Mannheim, 2011.
13. Christian Meilicke and Heiner Stuckenschmidt. Incoherence as a basis for measuring the quality of ontology mappings. In *Proc. 3rd International Workshop on Ontology Matching (OM) collocated with ISWC*, pages 1–12, Karlsruhe (Germany), 2008.
14. Maria Roşoiu, Cássia Trojahn dos Santos, and Jérôme Euzenat. Ontology matching benchmarks: generation and evaluation. In Pavel Shvaiko, Isabel Cruz, Jérôme Euzenat, Tom Heath, Ming Mao, and Christoph Quix, editors, *Proc. 6th International Workshop on Ontology Matching (OM) collocated with ISWC, Bonn (Germany)*, 2011.
15. Pavel Shvaiko and Jérôme Euzenat. Ontology matching: state of the art and future challenges. *IEEE Transactions on Knowledge and Data Engineering*, 2012, to appear.
16. Evren Sirin, Bijan Parsia, Bernardo Cuenca Grau, Aditya Kalyanpur, and Yarden Katz. Pellet: a practical OWL-DL reasoner. *Journal of Web Semantics: Science, Services and Agents on the World Wide Web*, 5(2):51–53, 2007.
17. York Sure, Oscar Corcho, Jérôme Euzenat, and Todd Hughes, editors. *Proc. of the Workshop on Evaluation of Ontology-based Tools (EON) collocated with ISWC*, Hiroshima (Japan), 2004.
18. Cássia Trojahn dos Santos, Christian Meilicke, Jérôme Euzenat, and Heiner Stuckenschmidt. Automating OAEI campaigns (first report). In *Proc. International Workshop on Evaluation of Semantic Technologies (iWEST) collocated with ISWC*, Shanghai (China), 2010.

Grenoble, Milano, Amsterdam, Delft, Mannheim, Milton-Keynes, Montpellier, Trento, Prague, November 2011