UNIVERSITY
OF TRENTO

DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY

38050 Povo – Trento (Italy), Via Sommarive 14
http://www.dit.unitn.it

OPENKNOWLEDGE DELIVERABLE 3.3.: A METHODOLOGY FOR ONTOLOGY MATCHING QUALITY EVALUATION

Mikalai Yatskevich, Fausto Giunchiglia, Fiona McNeill
and Pavel Shvaiko

August 2007

Technical Report DIT-07-062

OpenKnowledge* Deliverable 3.3.: A methodology for ontology matching quality evaluation**

Coordinator: Mikalai Yatskevich[1]
with contributions from
Fausto Giunchiglia[1], Fiona McNeill[2], and Pavel Shvaiko[1]

[1] Department of Information and Communication Technology (DIT),
University of Trento, Povo, Trento, Italy
{yatskevi|fausto|pavel}@dit.unitn.it
[2] The University of Edinburgh, Edinburgh, UK
f.j.mcneill@ed.ac.uk

Abstract This document presents an evaluation methodology for the assessment of quality results produced by ontology matchers. In particular, it discusses: (*i*) several standard quality measures used in the ontology matching evaluation, (*ii*) a methodology of how to build semi-automatically an incomplete reference alignment allowing for the assessment of quality results produced by ontology matchers and (*iii*) a preliminary empirical evaluation of the OpenKnowledge ontology matching component.

1 Introduction

The OpenKnowledge (OK) system is a peer-to-peer network of knowledge or service providers. Each computer in the network is a peer which can offer services to other peers. OK is viewed as an infrastructure, where we only provide some core services which are shared by all the peers, while all kinds of application services are to be plugged on top of it. These plug-in applications are called the OK Components (OKCs). Notice that the OKCs link services to the OK infrastructure and may not actually contain the services themselves.

Interaction between OKCs is a very important part of the architecture. By using the Lightweight Coordination Calculus (LCC) [6], developers are able to define the Interaction Models (IMs) that specify the protocol that must be followed in order to offer or use a service. OKCs are the ones in charge of playing the IM roles. Since there is no *a priori* semantic agreement (other than the

* OpenKnowledge is a 3 year long STREP project financed under the European Commission's 6th Framework Programme. See, http://www.openk.org/ for details.

** The originally planned title of this deliverable as from the project proposal was "Benchmarking methodology". However, this new title better reflects current contents of the deliverable, and therefore, is used here.

IM), the ontology matching component is used to automatically make semantic commitments between the interacting parts.

The OK matching component is designed to solve the semantic heterogeneity problem on the various stages of the OK interaction lifecycle. It is composed of the matchers of three kinds, namely *label*, *node* and *structure preserving* matchers. In total these three categories result in around two dozens of concrete matchers, see the OK Deliverable D4.1 [4] for details. This, in turn, raised the issues of their empirical validation and comparative evaluation. One of the challenges of the ontology matching evaluation is how to build large scale evaluation datasets; specifically, a large *set of reference correspondences* or *reference alignment* against which the results produced by ontology matchers are to be compared. Notice that the number of possible correspondences grows quadratically with the number of entities to be compared. This often makes the manual construction of the reference correspondences demanding to the point of being infeasible for large scale matching tasks.

In this deliverable we review the methodological foundations of the ontology matching evaluation and provide a preliminary evaluation of the OK ontology matching component on a dataset build out of the real world ontologies.

The rest of the deliverable is organized as follows. Section 2 gives a brief introduction to the key notions of the ontology matching evaluation. Section 3 discusses the features of an incomplete reference alignment and illustrates how a dataset allowing for the quality evaluation of ontology matchers can be constructed semi-automatically by suitably extending an incomplete reference alignment. Section 4 discusses a preliminary dataset that we have created as well as an evaluation of the OK matching component on that dataset, while Section 5 concludes the deliverable.

2 Evaluation measures

The ontology matching evaluation theme has been given a chapter account in [2]. Its more recent advances have been described in [7] and a short summary was also presented in the OK Deliverable D3.1 [8]. Thus, in this section we only briefly overview the most relevant basic concepts at work along the lines of ontology matching quality measures (§2.1) and performance measures (§2.2).

2.1 Quality measures

The commonly accepted measures for qualitative matching evaluation are based on the well known in information retrieval measures of relevance, such as *precision* and *recall* [10]. Let us consider Figure 1. The calculation of these measures is based on the comparison between the correspondences produced by a matching system (denoted S) and a complete set of reference correspondences (denoted H) considered to be correct. H is represented by the area inside the dotted circle. It is usually produced by humans. Finally, we denote as M the set of all possible correspondences, namely the cross product of the entities of two input ontologies.

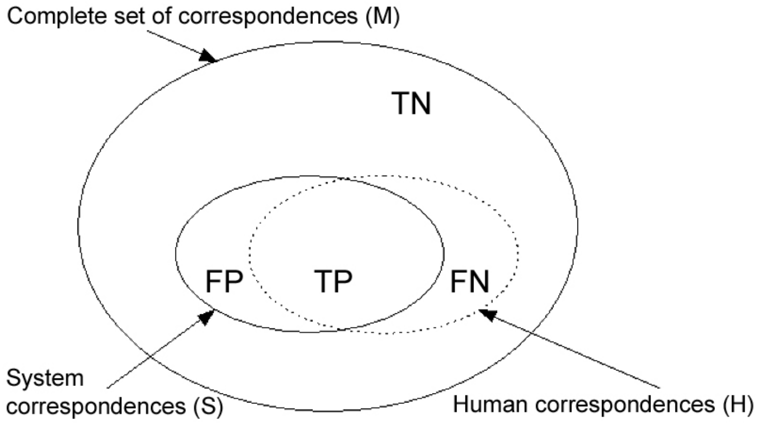
Complete set of correspondences (M)
TN
FP
TP
FN
System correspondences (S)
Human correspondences (H)

Figure 1. Basic sets of correspondences.

The correct correspondences found by a matching system are called the *true positives* (TP) and computed as follows:

$$TP = S \cap H \tag{1}$$

The incorrect correspondences found by a matching system are called the *false positives* (FP) and computed as follows:

$$FP = S - S \cap H \tag{2}$$

The correct correspondences missed by a matching system are called the *false negatives* (FN) and computed as follows:

$$FN = H - S \cap H \tag{3}$$

The incorrect correspondences not returned by a matching system are called the *true negatives* (TN) and computed as follows:

$$TN = M - S \cap H \tag{4}$$

We call the correspondences in H the *positive correspondences*, and the correspondences in N as defined in Eq. 5, the *negative correspondences.*

$$N = M - H = TN + FP \tag{5}$$

Precision is a correctness measure. It varies in the [0, 1] range, the higher the value, the smaller the set of wrong correspondences (false positives) which have been computed. It is calculated as follows:

$$Precision = \frac{|TP|}{|TP + FP|} = \frac{H \cap S}{S} \tag{6}$$

Recall is a completeness measure. It varies in the [0, 1] range, the higher the value, the smaller the set of correct correspondences (true positives) which have not been found. It is calculated as follows:

$$Recall = \frac{|TP|}{|TP + FN|} = \frac{H \cap S}{H} \tag{7}$$

Ontology matching systems are often not comparable based only on precision or recall. In fact, recall can be maximized at the expense of precision by returning all possible correspondences, i.e., the cross product of the entities from two input ontologies. At the same time, higher precision can be achieved at the expense of lower recall by returning only few (correct) correspondences. Therefore, it is useful to consider both measures simultaneously or a combined measure, such as F-measure.

In particular, F-measure is a global measure of the matching quality. It varies in the [0, 1] range. It allows the comparing of systems by their precision and recall at the point where their F-measure is maximal. Here, we use F-measure, which is a harmonic mean of precision and recall; that is, each of these measures is given equal importance. It is calculated as follows:

$$F\text{-}Measure = \frac{2 * Recall * Precision}{Recall + Precision} \tag{8}$$

In order to calculate precision, recall and F-measure, the complete reference alignment H must be known in advance. This opens up a problem of the reference alignment acqusition. The problem is that the construction of H is usually a manual process which, in the case of matching, is quadratic with respect to the size of the ontologies to be matched. This manual process often turns out to be unfeasible for large datasets. For instance, in the dataset of [1], built out of Google, Yahoo and Looksmart web directories, each model has the order of 10^5 entities. This means that construction of H would require the manual evaluation of 10^{10} correspondences.

2.2 Performance measures

Performance measures assess the resource consumption for matching ontologies. These measures are of high importance in the OK settings that require some form of real time performance in order to avoid having a user waiting too long for the OK system respond. Unlike the quality measures, performance measures depend on the processing environment and the underlying (OK) system. Thus it is often difficult to obtain objective evaluations, because they are based on the usual measures, namely execution *time* in seconds and main *memory* in bytes. The important point here is that algorithms that are being compared should be run under the same conditions.

3 An incomplete reference alignment

In this section we discuss a methodology of how to build semi-automatically an incomplete reference alignment allowing for the evaluation of both recall (§3.1) and precision (§3.2) of the results produced by the ontology matchers.

3.1 Evaluation of recall

Figure 2 illustrates a situation where an incomplete reference alignment, called *TaxME* [1], is used for the ontology matching evaluation. As from Figure 2, it contains only part of the correspondences in H. The key difference between Figure 2 and Figure 1 is the fact that a complete reference alignment, namely the area inside the dotted circle in Figure 1 (and in Figure 2), is simulated by exploiting an incomplete one, namely by the area inside the dashed circle in Figure 2.

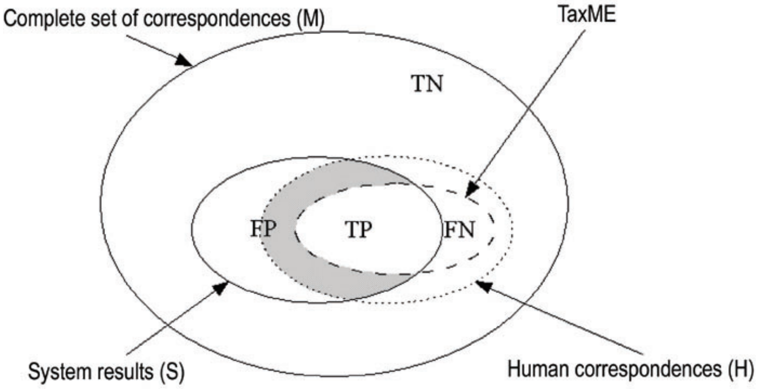
Complete set of correspondences (M)
TaxME
TN
FP
TP
FN
System results (S)
Human correspondences (H)

Figure 2. Correspondence comparison using an incomplete reference alignment.

Thus, if we assume that *TaxME* is a good representative of H we can use Eq. 7 for an estimation of the recall measure. In order to ensure that this assumption holds, a set of requirements have to be satisfied:

Correctness: namely the fact that *TaxME* $\subset H$ (modulo annotation errors).

Complexity: namely the fact that state of the art ontology matching systems experience difficulties when run on *TaxME*.

Discrimination capability: namely the fact that different sets of correspondences taken from *TaxME* are hard for the different systems.

Incrementality: namely the fact that *TaxME* allows for the incremental discovery of the weaknesses of the tested systems.

3.2 Evaluation of precision

In order to calculate the measure of precision (see Eq. 6) we need to know a complete set of reference correspondences H considered to be correct. As has already been mentioned, the computation of H in case of large applications often results in an infeasible manual effort. Also, notice that we can not use an incomplete reference alignment composed of positive correspondences, namely *TaxME*, either. In particular, FP can not be computed, since there is a part of it, which is unknown (see Eq. 9). In Figure 2 this unknown part (of false positives) is represented by the gray area.

$$FP_{unknown} = S \cap (H - \mathit{TaxME}) \tag{9}$$

A reference alignment for the evaluation of both recall and precision can be constructed following the *TaxME2* approach [11]. Specifically, Eq. 10 defines *TaxME2* by extending *TaxME* with an incomplete reference alignment containing *only* the negative correspondences. These are denoted as N_{T2} and $N_{T2} \subset M - H$, see Figure 2.

$$\mathit{TaxME2} = \mathit{TaxME} \cup N_{T2} \tag{10}$$

TaxME2 has to be a good representative of M and satisfy the requirements described in Section 3.1. Let us make two observations. The first one is that the correctness requirement significantly limits the size of N_{T2}, since each correspondence has to be evaluated by a human annotator, i.e., $|N_{T2}| \ll |M - H|$. The second observation is that N_{T2} must be big enough in order to be the source of meaningful results. Therefore, we require N_{T2} to be at least of the same size as *TaxME*, namely $|N_{T2}| \geq |\mathit{TaxME}|$.

The set of negative correspondences N_{T2} is computed out of the complete set of correspondences M (see Figure 1) in the two macro steps. The first step is called the *candidate correspondences selection*. During this step we acquire the set M', such that $M' \subseteq M$, see Figure 3. M' should contain a big number of "hard" negative correspondences, namely the correspondences with high value of similarity, which is incorrect according to a manual annotation.

The second step is called the *negative correspondences selection*. During this step we semi-automatically prune all the positive correspondences from M' in order to retain only the negative correspondences. Specifically, N_{T2} is computed from FP as shown in Eq. 11, where FP_i stands for FP produced by running the *i-th* matching system on M' (see Figure 3, where the grey area stands for FP_i).

$$N_{T2} = \bigcup_i FP_i \tag{11}$$

A set of matching systems whose results are exploited for constructing N_{T2} should be heterogeneous, that is that the selected systems make mistakes on the different sets of correspondences. This construction schema ensures that N_{T2} is

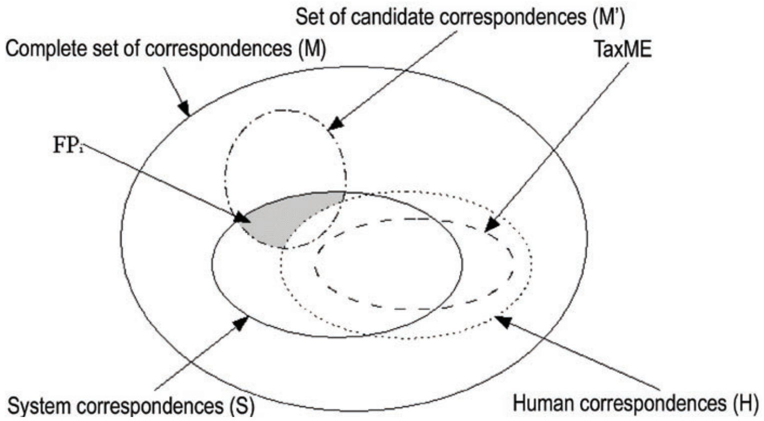
Set of candidate correspondences (M')
Complete set of correspondences (M)
TaxME
FP$_i$
System correspondences (S)
Human correspondences (H)

Figure 3. Sets of correspondences in *TaxME2*.

hard for all existing systems and discriminative given that the set of matching systems evaluated on M' is representative and heterogeneous.

We have discussed here only the key properties of an incomplete reference alignment dataset allowing for the evaluation of both recall and precision as well as outlined how this dataset can be built. Further technical details concerning the construction of such a dataset, for example, out of Google, Yahoo and Looksmart web directories can be found in [1,11]. It is worth noting that this methodology has already proved to be practically useful and the *TaxME* and *TaxME2* resulting web directories datasets have been exploited in the various ontology alignment evaluation campaigns - OAEI[1], see for details [7].

We plan to follow the above mentioned methodology when designing an evaluation dataset within the settings of the OK testbeds, such as e-response [9].

4 Preliminary evaluation

At present, a formalization, e.g., of the OK e-response scenario, is still under way, and hence the material necessary to apply the methodology of Section 3 is not available in order to build an evaluation dataset based on it. However, we have already made an implementation of the exact and approximate structure preserving semantic matching algorithms as described in the OK Deliverables D4.1 [4], D3.4 [12] and summarized in [3]. We have manually built a preliminary evaluation dataset in order to empirically validate the structure preserving matching by means of measures presented in Section 2 as well as to show practical use of these measures and a related analysis of the obtained results based on them.

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

In particular, we have evaluated the matching quality of the algorithms on 132 pairs of first order logic terms. Half of the pairs were composed of the equivalent terms (e.g., *journal(periodical-publication)* and *magazine (periodical-publication)*) while the other half were composed from similar but not equivalent terms (e.g., *web-reference(publication-reference)* and *thesis-reference (publication-reference)*). The terms were extracted from different versions of the Standard Upper Merged Ontology (SUMO)[2] and the Advance Knowledge Transfer (AKT)[3] ontologies. We extracted all the differences between versions 1.50 and 1.51, and between versions 1.51 and 1.52 of the SUMO ontology and between versions 1, 2.1 and 2.2 of the AKT-portal and AKT-support ontologies[4]. These are both first order ontologies, so many of these differences mapped well to the potential differences between terms that we are investigating. However, some of them were more complex, such as differences in inference rules, or consisted of ontological objects being added or removed rather than altered, and had no parallel in our work. These pairs of terms were discarded and our tests were run on all remaining differences between these ontologies. We have therefore simulated the situation when the web service descriptions are defined in one version of the ontology and the constraints in an IM are expressed exploiting the other version of the same ontology.

In our evaluation we have exploited three measures of matching quality, namely precision, recall, and F-measure as describe in Section 2. While computing these measures we have considered the correspondences holding among first order terms rather than the nodes of the term trees. Thus, for instance, *journal(periodical-publication*$_1$*)=magazine(periodical-publication*$_2$*)* was considered as a single correspondence rather than two correspondences, namely *journal= magazine* and *periodical-publication*$_1$*=periodical-publication*$_2$. The evaluation was performed on a Pentium 4 computer (1.5GHz) with 512 Mb of RAM.

Interestingly enough, our exact structure matching algorithm was able to find 36 correct correspondences what stands for 54% of recall with 100% precision. All mismatches (or correct correspondences not found by the algorithm) corresponded to structural differences among first order terms which exact structure matching algorithm is unable to capture. Several examples of correctly found correspondences are given below:

```
meeting-attendees(has-other-agents-involved)
meeting-attendee(has-other-agents-involved)

r&d-institute(Learning-centred-organization)
r-and-d-institute(Learning-centred-organization)

piece(Pure2,Mixture)
part(Pure2,Mixture)
```

[2] http://ontology.teknowledge.com/

[3] http://www.aktors.org

[4] See http://dream.inf.ed.ac.uk/projects/dor/ for full versions of these ontologies and analysis of their differences.

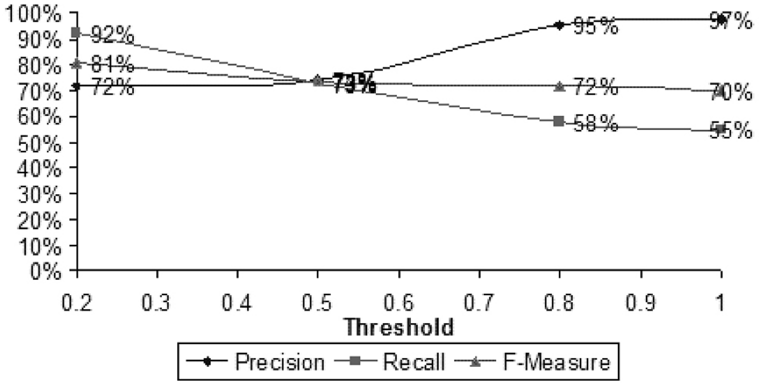

Figure 4. The matching quality measures depending on the cut-off threshold value for approximate structure matching algorithm.

```
has-affiliatied-people(Affiliated-person)
has-affililated-person(affiliated-person)
```

The first and second examples illustrate the minor syntactic differences among the terms, while the third and fourth examples illustrate the semantic heterogeneity in the various versions of the ontologies.

Figure 4 presents the matching quality measures depending on the cut-off threshold value (that ranges in [0, 1] and controls whether a correspondence should be retained or discarded) for approximate structure preserving matching algorithm [3]. As illustrated in Figure 4, this algorithm demonstrates high matching quality on the wide range of threshold values. In particular, F-measure values exceed 70% for the given range.

Table 1 summarizes the time performance of this matching algorithm. It presents the average time taken by the various steps of this algorithm on 132 term matching tasks. As illustrated in Table 1, step 1 and step 2 of the node matching algorithm significantly slow down the whole process. However, these steps correspond to the linguistic preprocessing that can be performed once offline [5]. Given that the terms can be automatically annotated with the linguistic preprocessing results based on the work in [5] once when changed, the overall runtime is reduced to 4.2 ms, which corresponds roughly to 240 term matching tasks per second.

Table 1. Time performance of the approximate structure matching algorithm (average on 132 term matching tasks).

	Node matching [5]: steps 1 and 2	Node matching [5]: steps 3 and 4	Structure matching [3]
Time, ms	134.1	3.3	0.9

5 Conclusions

In this deliverable we have briefly overviewed standard measures used for the ontology matching evaluation. We outlined a methodology for semi-automatic acquisition of reference alignments allowing for the assessment of quality results produced by ontology matchers. Finally, we provided preliminary empirical evaluation of the OpenKnowledge ontology matching component based on the manually created dataset of first order terms from the SUMO and AKT ontologies.

References

1. Paolo Avesani, Fausto Giunchiglia, and Mikalai Yatskevich. A large scale taxonomy mapping evaluation. In *Proc. 4th International Semantic Web Conference (ISWC)*, pages 67–81, Galway (IE), 2005.
2. Jérôme Euzenat and Pavel Shvaiko. *Ontology matching*. Springer, Heidelberg (DE), 2007.
3. Fausto Giunchiglia, Fiona McNeill, and Mikalai Yatskevich. Web service composition via semantic matching of interaction specifications. `http://eprints.biblio.unitn.it/archive/00001131/01/080.pdf`. The University of Trento, Technical report DIT-06-080, 2006.
4. Fausto Giunchiglia, Fiona McNeill, Mikalai Yatskevich, Zharko Alekovski, Alan Bundy, Frank van Harmelen, Spyros Kotoulas, Vanessa Lopez, Marta Sabou, Ronny Siebes, and Annette ten Tejie. *OpenKnowledge Deliverable 4.1: Approximate Semantic Tree Matching in OpenKnowledge*. `http://www.cisa.informatics.ed.ac.uk/OK/Deliverables/D4.1.pdf`, 2006.
5. Fausto Giunchiglia, Mikalai Yatskevich, and Pavel Shvaiko. Semantic matching: Algorithms and implementation. *Journal on Data Semantics*, IX:1–38, 2007.
6. Sindhu Joseph, Adrian Perreau de Pinninck, Dave Robertson, Carles Sierra, and Chris Walton. *OpenKnowledge Deliverable 1.1: Interaction Model Language Definition*. `http://www.cisa.informatics.ed.ac.uk/OK/Deliverables/D1.1.pdf`, 2006.
7. Pavel Shvaiko, Jérôme Euzenat, Heiner Stuckenschmidt, Malgorzata Mochol, Fausto Giunchiglia, Mikalai Yatskevich, Paolo Avesani, Willem Robert van Hage, Ondřej Šváb, and Vojtěch Svátek. *KnowledgeWeb Deliverable 2.2.9: Description of alignment evaluation and benchmarking results*. `http://exmo.inrialpes.fr/cooperation/kweb/heterogeneity/deli/kweb-229.pdf`, 2007.
8. Pavel Shvaiko, Fausto Giunchiglia, Marco Schorlemmer, Fiona McNeill, Alan Bundy, Maurizio Marchese, Mikalai Yatskevich, Ilya Zaihrayeu, Bo Ho, Vanessa Lopez, Marta Sabou, Joaqín Abian, Ronny Siebes, and Spyros Kotoulas. *OpenKnowledge Deliverable 3.1: Dynamic Ontology Matching: a Survey*. `http://www.cisa.informatics.ed.ac.uk/OK/Deliverables/D3.1.pdf`, 2006.
9. Lorenzo Vaccari, Maurizio Marchese, and Pavel Shvaiko. *OpenKnowledge Deliverable 6.6: Emergency Response GIS Service Cluster*. `http://www.cisa.informatics.ed.ac.uk/OK/Deliverables/D6.6.pdf`, 2007.
10. Cornelis Joost (Keith) van Rijsbergen. *Information retrieval*. Butterworths, London (UK), 1975.
11. Mikalai Yatskevich, Fausto Giunchiglia, and Paolo Avesani. *A Large Scale Dataset for the Evaluation of Matching Systems*. `http://eprints.biblio.unitn.it/archive/00001015/01/035.pdf`. The University of Trento, Technical report DIT-06-035, 2006.

12. Mikalai Yatskevich, Fausto Giunchiglia, Fiona McNeill, and Pavel Shvaiko. *OpenKnowledge Deliverable 3.4: Specification of ontology matching component.* `http://www.cisa.informatics.ed.ac.uk/OK/Deliverables/D3.4.pdf`, 2007.