Projecting LMF Lexica Towards OWL-DL through LMF-JAPE Patterns to Obtain Interoperable Formats

Lhioui Malek[1], Kais Haddar[2], Laurent Romary[3]

[1] Laboratoire MIRACL, Multimedia, InfoRmation Systems and Advanced Computing Laboratory, Université de Sfax, Tunisie
[2] Laboratoire MIRACL, Multimedia, InfoRmation Systems and Advanced Computing Laboratory, Université de Sfax, Tunisie
[3] Inria & Centre Marc Bloch, Berlin, Germany

ma.lhioui@gmail.com, kais.haddar@yahoo.fr, laurent.romary@inria.fr

Abstract. The development of editors, analyzers, translators and other NLP system types can involve several representation languages. The heterogeneity of representation languages induces the interoperability issue at different levels and in different contexts. In language technology, interoperability proved very crucial nowadays since its lack costs the translation industry a fortune where it is paid primarily for adjusting data formats. With this regard, we consider that representing LMF (Lexical Markup Framework: ISO-24613) lexica in OWL-DL (Web Ontology Language Description Logic) can be a serious attempt to achieve these goals. In this paper, we study the requirements of this proposal. We formulate an OWL ontology variant by explaining LMF mapping process to OWL version. The evaluation of the OWL variant construction of the LMF process is measured using the instantiation of the OWL-DL ontology.

Keywords: Interoperability, LMF, OWL-DL, LMF-JAPE patterns.

1 Introduction

The reusability notion in the past has evolved nowadays into interoperability. "This notion means the ability of information and communication systems to exchange data and to enable the sharing of information and knowledge" (Francopoulo, 2013).

Projects today require such a strategy, and to play the role of the keystone in many domains must rely on interoperability, otherwise they are out of business. A new article by TAUS (TAU, 2011) declares that: "The lack of interoperability costs the translation industry a fortune". In fact, this fortune is compensated mostly in order to adjust data formats. Interoperability concept can solve sharing problems based on involved elements semantics. For example, in the web field, the links between pages have no direct bearing managed by machines. "The Semantic Web is an extension of

the current Web in which information is given well-defined meaning, better enabling computers and people to work in cooperation" (Berners-Lee et al., 2001).

Unlike other areas, the NLP has not recognized an expansion of interoperability concept. The first serious attempts to this notion were after the development of LMF. In fact, standards facilitate greatly the construction of a powerful interoperability strategy. In fact, it provides well-structured lexical resources that are able to be open and shared after relevant mapping. Ontologies are one of the very recognized mapping activities. Indeed, these activities have proven their interest through many academic researches: WordNet or FrameNet are developed to disambiguate the semantic side of the terms or incorporate them into other ontology (Assem et al., 2006).

In this paper, we propose a rigorous method strictly founded to manage the projection from LMF to OWL format as a serious attempt to succeed interoperability between existing data formats particularly LTAG, HPSG, TEI and LMF[1]. This method can be used further in order to operate the transitivity between different existing data formats. In fact, there are several works proposing the mapping LTAG-HPSG[2], HPSG-LMF. So, by this projection LMF-OWL and using the transitivity notion between all the mentioned formats, the mapping between LTAG-OWL and HPSG-OWL will be operational.

We present in this paper an ontological OWL variant detailing the process of transforming the XML to OWL and we can prove its usefulness in NLP field by illustrating a reflection of morphosyntactic annotation.

2 Related Works and General Context

The ontologies construction requires the presence of well-structured methods. Generally, these construction methods are divided on statistical and linguistic approaches (Buitelaar et al., 2005). However, the two approach kinds require two extraction types: concepts and relations.

Statistically, concepts extraction requires analysis of co-occurring terms by studying their distribution or with probability determination. In a second step, relations extraction is the second key step in the construction process that can be determined using similarities between concepts (Grefenstette, 1994). Another method for predicting relations can also be identified using Bayesian networks (Weissenbacher and Nazarenko, 2007) or Text Mining techniques (Grcar et al., 2007).

In order to enrich ontologies, linguistic approaches were designed to collect association rules which are able to identify concepts and relations. Parsers are used in this case (Bourigault, 2002). Other methods may also be used to extract linguistic fingerprints of semantic relations to determine lexical and syntactic patterns. Other recent studies are leaded to enrich classic methods in order to separate content from structure. Therefore, recent attempts lead first to build the ontology core, then to develop them taking advantage from external resources through a preliminary analysis.

[1] LTAG : Lexicalized Tree Adjoint Grammar, HPSG : Head-driven Phrase Structure Grammar, TEI: Text Encoding Initiative

[2] From LTAG to HPSG

In the other side, nowadays, many works consisting on mapping from one format to another have been done. We can mention the mapping process already done by (Wilcock, 2007). This work presents an OWL ontology for HPSG. Another mapping process consists on projecting HPSG syntactic lexica towards LMF (Haddar et al., 2012). In the same context, a rule-based system has been created in order to translate LMF syntactic lexicon into TDL within the LKB[3] platform (Loukil et al., 2010). Furthermore, an owl-DL ontology has been conceived from an LMF input lexicon (Lhioui et al., 2014).

Until now, there are very few standards dedicated to the construction of normalized lexicons if we compare them with standards available for the manipulation of linguistic resources in general. However, the necessity of the construction of normalized lexicons is until now a hard task to be achieved. In fact, the normalization requires particular time resources with no left of human ones which are able to certify compatibility with chosen standards. LMF, which is conceived as a NLP standard, aims at covering large range of many languages. Consequently, having conformity to this standard makes our work in comparison with similar works on a global scale.

Because of its importance, we propose an initiative able to transform lexicon compliant LMF into an ontological OWL-DL variant. This allows supporting the development of reusable lexical data bases and then searching in the field of interoperability in future works.

Since the normalization identify necessary an information common coverage for all lexicons, the built coverage will be fundamental for many tools aiming the exchange and sharing of lexical resources and therefore provide the basis for developing an interoperable framework for with this type of ontological variant, the concept of interoperability: with this kind of ontological variant, interoperability notion will be able to be applied to exchange data and to enable knowledge sharing (Francopoulo, 2013). It will be a mixture of standards and guidelines such as the TEI (Sperberg-McQeen and Burnard, 2014). Thus, the standards will be systematically correlated and guidelines will explain the specification of these standards. Nowadays, having transformation prototype from LMF to OWL is very advisable. This prototype must have a big number of features, which will be explained thereafter.

3 Formalization of the Transformation Process

The transformation process requires a formalization step before its development. In order, to build a rigorous method allowing the transformation from LMF to OWL, a set of steps have to be formalized before any development process.

We start by giving an overview concerning LMF. Then, we present the conceptualization phase. Finally, we motivate this section by representing the construction phase describing the suitable formalisms used in this step.

[3] TDL: Type Description Language, LKB: Language Knowledge Builder

3.1 LMF Overview

After serious activities developed on building lexicons in teamworks, a group of 60 researchers was behind LMF standard creation (Francopoulo, 2013). LMF is an ISO standard which includes monolingual and multilingual lexica. LMF specification follows UML modeling Object Management Group (OMG). LMF is arranged on two big parts: core model and extensions packages. The LMF modelling principles take up the ISO committee TC 37 principles and let a lexical database designer to mix any component of the LMF meta-model with data-categories (Ide and Romary, 2004) in order to create an appropriate model. Data categories behave as UML attribute-value pairs in the diagrams. The core model contains the backbone of a lexical entry. It identifies critical concepts of vocabulary, word, form and sense. LMF core model is characterized by a hierarchical structure involving on several components.

3.2 Conceptualization Phase

In order to move from the source schema for a diagram source (whatever its nature: text, XML, etc.) to the ontology, the majority of construction methods design a conceptualization step in order to ensure the passage from the first plan to the second. The source in this article includes XML files. Conceptualization requires an analysis of the source tags. This preliminary analysis will show its interest in building the bases of ontology concepts and relations. For this reason, all classes of LMF package must be provided in advance an XML form.

Conceptualization requires prior analysis of properties and relations between LMF classes. This explains the creation of an XML file containing properties in the LMF classes that must be offered first.

3.3 Construction Phase

In order to develop the construction phase and after studying the feasibility of the constructed ontology and analyzing knowledge sources, we formally define the ontology as following.

Formal Definition of constructed ontology. The ontology O will be defined in this form: O=(C, R, Hc, Rel, A). C and R are defined as disjoint sets dedicated for concepts and relations respectively. Hc is the hierarchy of concepts which is represented as Hc $\subset$ C × C. The Rel set define the semantic and non-taxonomic relations with two associated functions. It is represented as Rel: R $\rightarrow$ C × C with the domain is defined as dom: R$\rightarrow$C, dom(R) = $\prod$1 (Rel (R)) and the range is represented as range: R$\rightarrow$C, range(R) = $\prod$2 (rel (R)) co-domain.

The instantiation of the formal definition of ontology constructed can be represented as shown in Fig. 1:

This light fragment represented in figure 1 contains the set of concepts ***C= {C1 : lexicalEntry, C2 : form, C3 : form Representation, C4 : representation}***, a set of

relations ***R={r1 : hasForm, r2 : hasFormRepresentation}*** and a set of concept hierarchy ***Hc={H1c}.***

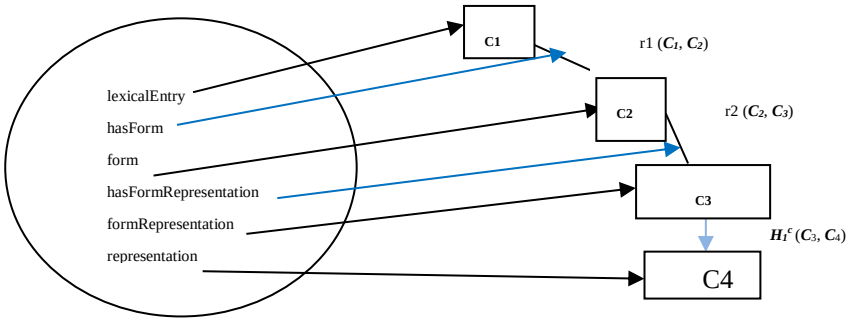

Fig. 1. Representation of LMF core

In order to instantiate the ontology, we have to define the KB[4] set as: ***KB= (O, I, Inst, Instr):*** With ***O = (C, R, HC, Rel, A)*** is the LMF ontology defines formally above, *I* is the instances set, *inst* : $C \rightarrow 2I$ translated the instantiation function of concepts and *instr* : $R \rightarrow 2\ I \times I$ is the instantiation function of relation.

4 Ontology construction

The ontology construction is an important step in the whole process of mapping LMF-OWL. The target for the development of ontological version is the establishment of an interoperable environment enabling the management of lexical resources. For this reason, construction such framework requires the presence of applications that can make possible the exchange of un-formal and unstructured data across the web.

Thus, the prototype is described in five steps: LMF Entities, Namespaces, LMF classes, LMF subclasses and properties as follows.

4.1 LMF Entities

LMF entities are considered as assertion in the new ontology. To automate the construction of this task, we need to define a new pattern following the JAPE[5] syntax:

```
Pattern1: ({Entity.name=="lmf"}) : NewOWLEntiy
- - > : NewOWLEntiy = {value= "http://www.lexicalmarkupframework.org#", rule = R1p}
```

When defining a new entity named "lmf", a new one has to be defined in OWL. This entity will have as value = "http: //www.lexicalmarkupframework.org#." The set is described under the name NewOWLEntity, and therefore, a specific process must be associated as well.

[4] KB : Knowledge Base
[5] JAPE : Java Annotation Pattern Engine

4.2 Namespaces

Automating the construction of namespaces in the OWL-DL variant requires the definition of a new pattern using the same syntax JAPE:

```
Pattern 2: ({Input.var=="URI"}) : NewOWLUsedNamespace
- - > : NewOWLUsedNamespace = {xmlns = "URI#", rule = R2p}
```

The definition of a new namespace is designed by the pattern presented. The new namespace will have as value = "# URI." The set is described under the name NewOWLUsedNamespace.

4.3 Entities and LMF classes

A set of assertions have to be made after the namespaces accuracy. This set enriches the output with comments, labels, version etc. Automating the construction of OWL-DL headers requires the following pattern:

```
Pattern 3: ({Input.var1=="URI"}{Input.var2=="label"}) : NewOWLHeader
- - > : NewOWLHeader = {element = "owl:Ontology", rule = R3p}
```

The pattern defined has as an entry, an URI and a label and provides as an output a new OWL element noted owl: Ontology.

OWL classes are considered as basic components in the resulted ontology. All these classes will be members of the class Thing.

```
Pattern 4: ({Input.var1=="URI"}{Input.var2=="class"} {"class" ∈ LMFClasses}) : NewOWLClass
- - > : NewOWLClass = {element = "owl:Class", rule = R4p}
```

The pattern requires an URI and the class name to produce a new element noted owl: Class. However, one condition must be fulfilled before developing any process. It is mandatory to verify the belonging of the new class to the set of LMFClasses defined in LMF and conceived in the previous section.

4.4 LMF SubClasses

The restrictions list may also contain sub-classes. The subclasses construction mechanism conceived in OWL-DL is defined using the following new pattern:

```
Pattern 5: ({Input.var1=="Entity"}
{Input.var2=="Class"}{var2 ∈ LMFClasses}
{Input.var3=="SubClass"}{var3 ∈ LMFSubC}) : NewOWLSubClass
- - > : NewOWLSubClass = {element = "rdfs:subClassOf", rule = R5p}
```

LMF subClass definition is represented by the pattern defined. The Subclass requires three variables: the entity, the subclass and the class to which it belongs. Two conditions must be fulfilled: the class value of variable 2 and the subclass value of the

variable 3 must belong respectively to class and subclass set predefined in LMF. This pattern will be described under the name of *NewOWLSubClass*.

4.5 LMF Properties

Many informations are interpreted as attributes in either the core or extensions of LMF packages.

```
Pattern 6: ({Input.var1=="Entity"}
{Input.var2=="Property"}{var2 ∈ LMFProperties}
{Input.var3=="range"}
{Input.var4=="domain"}
{Input.var5=="rangeEntity"}) : NewOWLProperty
- - > : NewOWLProperty = {element = "owl:datatypeProperty", rule = R6p}
```

The accuracy of a LMF property is represented by the previous pattern. The property requires three key variables which are "property", "range" and "domain", other ones are optional.

5 General architecture of the method

After conceiving the method for the automatic construction of ontology, we have now to describe the implementation of the complete method to validate our prototype. In this section, we will describe the general architecture of our prototype.

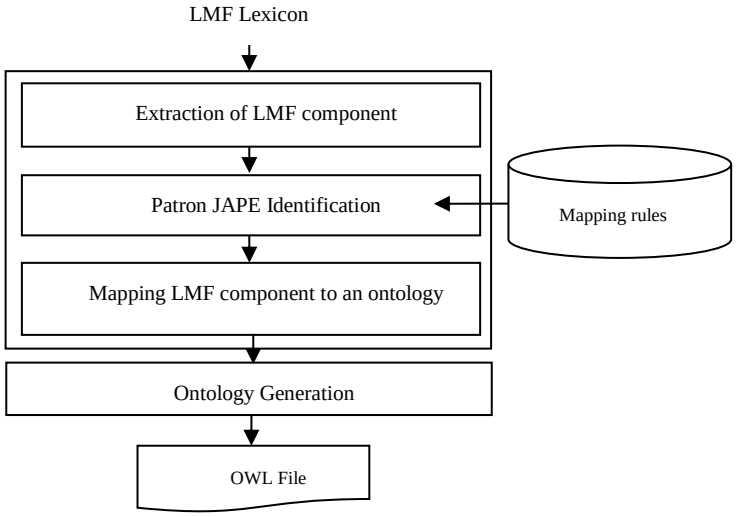

Fig. 2. General Architecture of the methods

Mapping process input is a serialized LMF lexicon into an XML file which may contains all LMF components or simply restraint set of these constituents. In further section we will explain different methods used to extract the XML component, identification of suitable JAPE pattern and the mapping process using the appropriate rules.

5.1 Extraction of LMF component

The first step of the mapping process is to extract the LMF constituent represented by an XML tag in the LMF lexicon. This is an important step in the mapping process. Figure 3 illustrates this module.

```
<?xml version="1.0" encoding="UTF-8"?>
<lexicalRessource dtdVersion="16">
  <globalInformation>
    <feat att="languageCoding" val="ISO 639-3"/>
    <feat att="scriptCoding" val="ISO 15 924"/>
  </globalInformation>
  <lexicon>
    <feat att="language" val="arab"/>
    <lexicalEntry morphologicalPatterns="intransitifVerb">
      <feat att="partOfSpeech" val="verb"/>
      <feat att="root" val="س_ل_ج"/>
      <feat att="scheme" val="فَعَلَ"/>
      <lemma>
        <feat att="writtenForm" val="جَلَسَ"/>
        <feat att="writtenForm" val="-"/>
        <feat att="type" val="صحيح"/>
      </lemma>
    </lexicalEntry>
  </lexicon>
</lexicalRessource>
```

Lexical Resource

Fig. 3. Extraction of LMF component

The example in figure 3 concerns the verb jalasa جلس (to sit). The different characteristics related to the verb are compliant to the LMF standard (ISO-24613).

5.2 Identification of the JAPE pattern

The development of the JAPE pattern identification requires a set of tests in order to verify the coincidence of LMF extracted component with the selected JAPE pattern. Figure 4 illustrates all these tests.

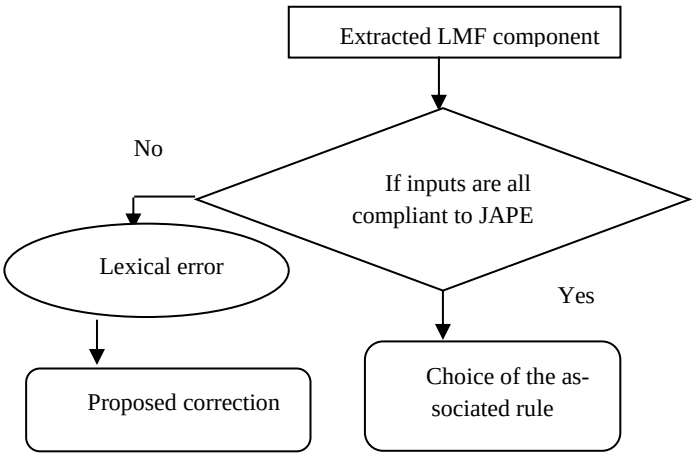

Fig. 4. JAPE pattern identification Schema

Figure 4 indicates that the process of identifying the JAPE pattern consists first on verifying compliance in number and value of the component extracted with inputs from each pattern. This verification ends either by the choice of the associated rule or by a proposed correction of the lexical error.

5.3 Mapping from LMF component to the new ontology

The mapping module of LMF component to an ontology using the JAPE pattern is a key phase in the general architecture. This module consists on creating the suitable concept or relation by applying the associated rule. Figure 5 illustrates this mapping.

Figure 5 indicates that the mapping process of the previous fragment detected the presence of the entities (rdf, rdfs, xsd and owl), a label, a commentary and the two concepts: lexicalRessource and globalInformation.

6 Discussion

The evaluation of the OWL variant construction of the LMF process can be measured using the instantiation of the OWL-DL ontology already constructed. Thus, in this section we choose the morphological package extension. This choice is explained by the fact that this extension is considered crucial in most NLP and Semantic Web applications. This extension is described by two different ways in LMF. The first exposed explicitly inflected forms. The second uses flexion paradigms in order to gener-

ate various forms derived from the lexical entry. The following built prototype instantiation represents inflectional description of the verb "أكل" (to eat).

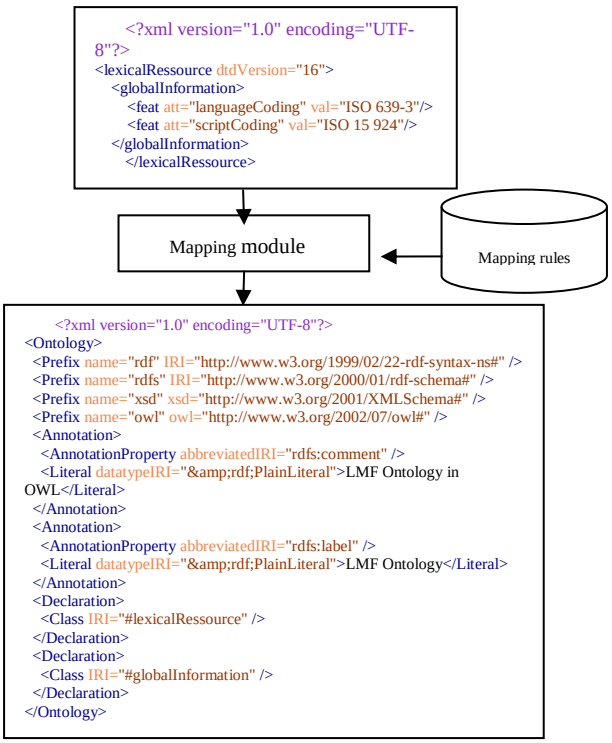

Fig. 5. LMF fragment Mapping

The above example reveals one possible inflected form from a set of 56 possible inflected forms that might have a verb.

Table 1. LMF components Evaluation

JAPE patterns / Number of lexica	LMF Entities	Namespaces	LMF Classes	LMF subclasses	Properties
170	170	680	10710	510	3400

We have applied the constructed prototype for a set of 170 Arabic lexica. The choice of the Arabic language is justified by the availability of these XML files, the necessity of our team work and finally to improve researches in this language.

7 Conclusion

We examined the structure of the LMF standard in order to conceive the OWL-DL ontology core. This ontology may be used for example in a morphosyntactic annotation application. This annotation will play the role of a GATE plugin allowing the disposition of LMF serialization. The underlying idea is to create an interoperable environment evoking dynamism between standards and guidelines. The condition must fulfill these environments is the internal consistency without neglecting the update of modeling involved standards and their serialization. We have proposed a rigorous method based on LMF-JAPE pattern to manage the projection from LMF to OWL format as a serious attempt to succeed interoperability between existing data formats. This method can be used in advance for operating the transitivity between different existing data formats. Consequently, the mapping between LTAG-OWL and HPSG-OWL will be easily operational.

References

1. Assem, V.M., Gangemi, A., and Schreiber, G., 2006, *Conversion of WordNet to a standard RDF/OWL representation*, Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06), Genoa, Italy.
2. Berners-Lee, T., Hendler, J., Lassila, O., 2001, *The semantic web*, Scientific american 284.5, 28-37.
3. Bourigault, D., 2002, UPERY: un outil d'analyse distributionnelle étendue pour la construction d'ontologies à partir de corpus, TALN, Nancy.
4. Buitelaar P., Cimiano P., Magnini B., 2005, Ontology Learning From Text: Methods, Evaluation and Applications, IOS Press.
5. Francopoulo, G., 2013, *Lexical Markup Framework*, US, Great Britain and the United States: ISTE Ltd and John Wiley & Sons, Inc.
6. Grcar M., Klein E., Novak B., 2007, *Using Term-Matching Algorithms for the Annotation of Geo-services*, Postproceedings of the ECML-PKDD Workshops, Springer, Berlin − Heidelberg − New York.Boston, MA: Kluwer Academic Plubisher.
7. Grefenstette, G., 1994, *Explorations in Automatic Thesaurus Discovery*, MA: Kluwer Academic Plubisher, Boston.
8. Haddar, K., Fehri, H., Romary, L., 2012, A prototype for projecting HPSG syntactic lexica towards LMF, JLCL.
9. Ide, N., Romary, L.: A Registry of Standard Data Categories for Linguistic Annotation. In Proc. 4th International Conference on Language Resources and Evaluation - LREC'04 135–138 - http://hal.inria.fr/inria-00099858 (2004)
10. Lhioui, M., Haddar, K., Romary, L., 2014, *Towards modelling LMF lexicons compliant LMF in OWL-DL*, TKE, Berlin.
11. Loukil, N., Ktari, R., Haddar, K., Benhamadou, A., 2010, A normalized syntactic lexicon for arabic verbs and its evaluation within the LKB platform, ACSE, Egypt.
12. NEON Project, D2.5.1 :, 2008, A
13. Library of Ontology design Patterns : reusable solutions for collaborative design of networked ontologies.

14. Sperberg-McQeen, C.M., Burnard, L., 2014, *TEI P5: Guidelines for Electronic Text Encoding and Interchange*, Text Encoding Initiative Consortium Charlottesville, Virginia: the TEI Consortium.
15. TAUS, Report on a TAUS research about translation interoperability, 25 February, 2011.
16. Weissenbacher, D., Nazarenko, A., 2007, Identifier les pronoms anaphoriques et trouver leurs antécédents : l'intérêt de la classification bayésienne, TALN, Toulouse.
17. Wilcock, G., 2007, *An OWL ontology for HPSG*, ACL, Finland.