# Multilingual Online Translation

Non multa, sed multum

| | |
|---|---|
| Contract No.: | FP7-ICT-247914 |
| Project full title: | MOLTO - Multilingual Online Translation |
| Deliverable: | D8.2 Multilingual grammar for museum object descriptions |
| Security (distribution level): | Public |
| Contractual date of delivery: | 1 March 2012 (M24) |
| Actual date of delivery: | 16 March 2012 |
| Type: | Prototype |
| Status version: | V1.2 |
| Author(s): | Dana Dannélls, Aarne Ranta, Ramona Enache |
| Task responsible: | UGOT |
| Other contributors: | Milen Chechev and Mariana Damova |

# Contents

**1 Introduction** **3**
1.1 The purpose of this document . . . . . 3
1.2 Use cases . . . . . 4

**2 The grammar development workflow** **4**

**3 The grammar** **4**
3.1 Ontology verbalization . . . . . 4
3.2 Discourse pattern generation . . . . . 6
3.3 General design issues . . . . . 7
3.4 Generation Results . . . . . 9

**4 Conclusion and future work** **10**

# 1 Introduction

During the last decade, there has been a shift from developing natural language systems to developing domain independent applications that are capable of producing natural language descriptions directly from Web ontologies (Schwitter and Tilbrook (2004); Fuchs et al. (2008); Williams et al. (2011)). Many of the existing systems employ verbalisation methods to present the content of the ontology structure to particular subset of users, almost exclusively in English language. The problem with the existing verbalization methods is that they assume each ontology statement is mappable to natural language, which is not always the case, in particular for languages other than English. Moreover, the task of producing adequate natural language descriptions by employing verbalization methods is very difficult if not impossible.

We have developed a grammar application in GF that applies natural language generation techniques to generate multilingual descriptions about museum artefacts, starting from the CIDOC-CRM ontology. We opted for a layered representation of the natural language generation system, where the ontology represents the first layer, which provides the semantic structure and the instances that we will verbalize. A second layer is the natural language generation grammar, which defines the way in which we combine ontology data in order to create text. Due to the multilingual context, the generation grammar aims to be general enough to allow the same assertions to be expressed in all the languages.

At our knowledge, this is the first attempt to develop a prototype that exploits natural language generation techniques such as applying discourse strategies to generate multilingual descriptions in at least five languages from semantic web content, in particular from OWL ontology standards such as CIDOC-CRM.

## 1.1 The purpose of this document

Work package 8 foresees several tasks:

- integrate data from the Gothenburg City Museum (GCM) with the Conceptual Reference Model CIDOC-CRM ontology standard
- build a prototype of a cross-language retrieval and representation system to be tested with objects in the museum
- implement a multilingual domain application grammar that is capable of generating well-formed object descriptions from CIDOC-CRM

The purpose of this document is to describe the developed domain specific application grammar we have been implementing. The grammar presented here allows to generate well-formed multilingual natural language descriptions about museum artefacts with the aim of empowering users who wish to access cultural heritage information through different computing devices.

### 1.2 Use cases

Some of the the key benefits of the grammar application that is being developed in this project are:

1. acceptability and usability by other grammarians
2. acceptability and usability by the industrial environment
3. possibilities of reuse by other applications

## 2 The grammar development workflow

To accomplish the goals of this workpackge we found it necessary to develop a specific ontology model to store and present detailed information about specific artefacts that are available in the Gothenburg City Museum database. The focus was on painting objects, as described in Dannélls (2011). The main model for the painting ontology development was the CIDOC–CRM, which provided a detailed conceptual reference as a starting point for the ontology design. Using the painting ontology as a point of departure we were able to develop two different generations modules: one that applies direct verbalization (section 3.1) and another that exploits discourse pattern generation (section 3.2).

The painting ontology containing data from the museum has been integrated into the Reason-able View of Linked Data for Cultural Heritage, Damova and Dannélls (2011) which is also part of workpackage 4, see Damova (2011). Basically, we integrated the paintings and the other objects in one single repository, providing an interoperable framework that is to used retrieve information about museum objects, Dannélls et al. (2011).[1] From this Web repository it is possible to retrieve a set of RDF triples that provides a formal description about the museum object, including the name of the object, the painter who created it, the year of painting, the material that was used to execute the object, the current location of the object, how it was acquired, its value and other semantic information that is given both in the form of Id's and canned text, such as about the content or historical knowledge about an object.

The retrieved information forms the input to the domain grammar application from which we are able to generate multilingual descriptions as described below (section 3.2).

## 3 The grammar

### 3.1 Ontology verbalization

An important component in the natural language generation system is the layer that connects the ontology with the generation grammar. On one step we get it by importing the Painting ontology in GF, as it contains the instances of most fields that describe the paintings. However, the painting names and painters need to be imported directly from the database.

[1] `http://museum.ontotext.com`

Unlike previous experiments with representing SUMO (Suggested Upper-Merged Ontology) in GF (Enache (2009); Enache and Angelov (2010)), the representation of the painting ontology doesn't use dependent types, since simple types are enough to describe the concepts and relations and model the hierarchy in a simplified way that suffices the needs of the generation grammar.

For example, the classes `OilPainting` and `Painting`, along with the inheritance relation between them are represented as GF categories:

```
cat Painting ;
cat OilPainting ;

fun OilPainting_Painting : OilPainting -> Painting ;
```

where the inheritance relation is modeled as a coercion function between the two types. In the concrete syntaxes, all coercion functions will be linearized as the identity, since they shouldn't be visible in verbalization.

The instances are represented as GF instances of the GF mappings of their ontology classes. For example:

```
fun AerosolPaint : Material ;
```

The advantage is that when porting the Painting grammar to a new language, one could linearize certain categories to different parts of speech, depending on how the concepts are expressed in the language.

Developing a concrete syntax for the ontology grammar is quite straightforward for English, but could be a challenge for other languages. The examples from the current grammar were translated manually, because the number of paintings that we described was very small.

However, in the future, we plan to build the lexicon multilingual lexicon automatically in 2 steps:

- port the painting ontology (at least the classes and instances) in another language by using the lexical translation tools from WP3, which would ensure a semantics-preserving mapping of the lexical units by using multilingual resources, such as DBPedia.

- as we assume that the painter name and painting name are the only attributes that might not be found in the Painting Ontology, we add them on the fly, for the painting that we want to verbalize and not for the whole database. The reason is that they will be represented as proper names, whereas the other features could be mapped to different parts of speech depending on the language. Moreover, the names of the painting and its creator can be translated using the same resources as in the previous step, before added to the concrete syntax.

The automation of lexicon acquisition will be possible as soon as there will be an integration of the WP3 tools to the grammar development ones.

## 3.2 Discourse pattern generation

In the first deliverable of this workpackage, e.g. Dannélls (2011) we presented a number of features and discourse patterns that we learned by analyzing a large set of well-formed object descriptions. Below we summarize some of the discourse patterns the analysis reveled.

- painting paintingtype painter
- painting painter year
- painting museum painter size
- painting painter represented museum
- painting material year painter
- painting painter year museum colour size

The initial idea was to follow these features and patterns when generating multilingual descriptions.

We isolated a number of attributes of paintings that we decided to focus on in the prototype development. We agreed that each description should convey at least 3 main features of a painting. This assumption enable us to define a default representation in GF and thereby always produce a description about an object. These three features are:

1. the name of the painting — `Painting`
2. the name of the painter — `Painter`
3. the type of painting (for example, *oil painting*) — `PaintingType`

and 5 optional ones that allow us to generate more detailed descriptions:

1. the colours used in the painting — `Colour`
2. the size of the painting — `Size`
3. the material of the painting — `Material`
4. the year when the painting was created — `Year`
5. the museum where the painting is currently displayed — `Museum`

The difference between the 2 categories of features, is that we don't expect to find all the optional categories in all the painting descriptions from the database, but we want to have only one representation for the instances of paintings from the ontology which we will verbalize and only one verbalization function in the grammar, which would be easier to port to new languages.

The solution is to wrap the categories representing optional features in a number of categories inspired by the `Maybe` type from Haskell, which retains the presence/absence of the feature and in case the feature is present, its value.

Hence, we can represent the text generation as one function taking all the features as arguments:

```
MkGenText :
   Painting -> Painter -> PaintingType ->
   OptColour -> OptSize -> OptMaterial ->
   OptYear -> OptMuseum -> GenText ;
```

where `OptColour`, `OptSize`, `OptMaterial`, `OptYear` and `OptMuseum` are the wrapper categories. The concrete representations of `MkGenText` opt for different text patterns, depending on the presence of the optional parameters.

For the cases, where the name of the painter or the painting is missing, we created a number of instances that indicate the absence of the features:

- `NoPainting`
- `NoPainter`

In this way, we get the most detailed description that one can form with the available features. This differentiates the current approach from the previous one, described in Dannélls et al. (2012), where each text-generation pattern is represented separately, so that the user can choose the sort of descriptions that she wants.

The reason why the current grammar only retains the most informative description is that the implementation of the patterns contains redundancies and entails more effort from the grammar writer. On the other hand, the usage of the patterns provides more options for natural language generation, because we can control the amount of information that we describe.

The grammar structures is ported to 5 languages: English, French, Italian, Finnish and Swedish.

## 3.3 General design issues

The current approach reduces the use of dependent types to a minimum, in order to keep the grammar simpler and make it easier for users to port it to new languages. The only use of dependent types in the current grammar is for representing the painting structure. This is necessary for enforcing a predefined structure on the generated text. For example the definition (def) below enforces the generated text (MkGenText) to bear the eight features.

```
fun
  vtext2gtext : VerifiedText -> GenText ;
def
  vtext2gtext (MkVerifiedText pg pr pt cr se ml yr mm _) =
MkGenText pg pr pt cr se ml yr mm ;
```

This makes it possible to control the natural language generation; keeping the descriptions consistent with the ontology. For example, the painting `GSM940042Obj` is represented as following:

```
GSM940042ObjPainting : CompletePainting
 GSM940042Obj MiniaturePortrait  JKFViertel (MkYear (YInt 1814))
 (MkMuseum GoteborgsCityMuseum)  (MkColour Grey)  (MkSize (SIntInt 349 776))
 (MkMaterial Wood) ;
```

Thus, when a description is generated, we get all the information that is associated to it in the Painting ontology.

This is however just an optional feature, because one can preserve the semantic consistency by adding another layer, exterior to the grammar, that calls the text-generation function with the proper arguments. But the possibility of having it inside the grammar is more attractive, since it shows the power of GF.

In any case, the dependent types don't bring about any change when porting the grammar to new languages, as they are just a way to group together features that used for generation.

The previous version of the grammar featured a more extensive use of dependent types including the type used to represent a painting and all its attributes that is preserved in the current implementation. Moreover, the previous grammar used semantic definitions for functional programming-inspired pattern-matching on the relevant features that each pattern needs to use. This entails that the case analysis is done just once – in the abstract syntax and doesn't need to be repeated for each language. The current implementation implements it in the concrete syntax, which could lead to code repetition across the languages.

In both cases, the dependent types don't need to be implemented in the concrete syntax, and the less-experienced user won't need to manipulate them in order to port the grammar to a new language. The current grammar is even more user-friendly, as the dependent types are almost seamless, and the users don't need to use them, if the grammar is used within a runtime system that doesn't provide support for them, such as Java or C.

When adapting the grammar to a new language, the only thing the grammarian needs to create about is one function, `MkGenText` and the lexicon.

```
MkGenText painting painter paintingtype colour size material year museum =
   let
```

```
 s1 : Text = mkText (mkS pastTense
  (mkCl painting (mkVP (mkVP (mkVP (passiveVP paint_V2) material.s)
  (SyntaxEng.mkAdv by8agent_Prep (mkNP painter))) year.s))) ;

  sizeS : S = mkS (mkCl it_NP size.s) ;
  colourS : S = mkS (mkCl it_NP (mkVP (passiveVP paint_V2) colour.s)) ;

  s2 : Text = case <size.isGiven, colour.isGiven> of {
    <True,True>  => mkText (mkS and_Conj sizeS colourS) ;
    <True,False> => mkText sizeS ;
    <False,True> => mkText colourS ;
    _            => emptyText
   } ;

  s3 : Text = case museum.isGiven of {
    True => mkText (mkS
              (mkCl (mkNP this_Det paintingtype)
                 (mkVP (passiveVP display_V2) museum.s))) ;
    _ => emptyText
   } ;
in
   mkText s1 (mkText s2 s3) ;
```

## 3.4 Generation Results

The application outputs consist of short, well-formed natural language descriptions in five languages. On the syntactic level, sentence structures contain passive constructions, aggregations and generation of referring expressions. Some examples are given below.

Painting: MkGenText GSM940051Obj BrynolfWennerberg PortraitPainting NoColour NoSize (MkMaterial Wood) (MkYear (YInt 1889)) (MkMuseum GoteborgsCityMuseum)

- PaintingEng: Hisingen was painted on wood by Brynolf Wennerberg in 1889. This portrait is displayed at the City Museum of Gothenburg.
- PaintingFin: Maalauksen Hisingen on maalannut Brynolf Wennerberg puulle vuonna 1889. Tämä muotokuva on esillä Göteborgin kaupunginmuseossa.
- PaintingFre: Le tableau Hisingen a été peint sur bois par Brynolf Wennerberg en 1889. Ce portrait est exposé dans le musée municipal de Göteborg.
- PaintingIta: Il quadro Hisingen è stato dipinto su legno da Brynolf Wennerberg nel 1889. Questo ritratto è esposto nel museo municipale di Goteburgo.
- PaintingSwe: Hisingen målades på trä av Brynolf Wennerberg år 1889. Den här porträttmålningen är utställd på Göteborgs stadsmuseum.

Painting: MkGenText GSM980019Obj AnnaLindskog OilPainting (MkColour Black) (MkSize (SIntInt 435 365)) (MkMaterial Canvas) (MkYear (YInt 1885)) (MkMuseum GoteborgsCityMuseum)

- PaintingEng: The girl was painted on canvas by Anna Lindskog in 1885. It is of size 435 by 365 and it is painted in black. This oil painting is displayed at the City Museum of Gothenburg.
- PaintingFin: Maalauksen Flickan on maalannut Anna Lindskog kankaalle vuonna 1885. Se on kokoa 435 kertaa 365 ja se on maalattu mustalla. Tämä öljymaalaus on esillä Göteborgin kaupunginmuseossa.
- PaintingFre: Le tableau Flickan a été peint sur toile par Anna Lindskog en 1885. Il est de taille 435 sur 365 et il est peint en noir. Cette peinture à l' huile est exposée dans le musée municipal de Göteborg.
- PaintingIta: Il quadro Flickan è stato dipinto su tela da Anna Lindskog nel 1885. Misura 435 per 365 ed è dipinto in nero. Questo dipinto ad olio è esposto nel museo municipale di Goteburgo.
- PaintingSwe: Flickan målades på duk av Anna Lindskog år 1885. Den är av storlek 435 gånger 365 och den är målad i svart. Den här oljemålningen är utställd på Göteborgs stadsmuseum.

# 4 Conclusion and future work

The presented grammar consists of one discourse patten that contains a small amount of features a painting object description should convey. From this pattern we are able to generate different descriptions depending on the information that is available about this object. The main advantage of the grammar is that it can be ported the other languages very easily, by only modifying one pattern and changing the lexical entities.

One of the functionalities the current grammar does not cover is the ability to combine different features across different sentences. However, the simplicity of the grammar makes the working effort of adding new patterns for distributing features differently across sentences minor. Moreover, with only small modifications, such as selecting other types of referring expressions, we are able to increase the fluency of the output results depending on the language.

In the nearest future we intend evaluate the generation results and port the grammar to 10 additional languages. We plan to increase the coverage of grammar and the lexicon for at least 5 languages.

It will be interesting to test how the grammar performs with different objects and on other domains. Another possible future direction is to generate texts in different formats that can be adaptable to different user needs, for example by modifying the style of the generated texts in terms of syntactic variations.

# References


Mariana Damova. *Data Models and Alignment*, May 2011. Deliverable 4.2. MOLTO FP7-ICT-247914.

Mariana Damova and Dana Dannélls. Reason-able view of linked data for cultural heritage. In *Proceedings of the third International Conference on Software, Services and Semantic Technologies (S3T)*, 2011.

Dana Dannélls. *D.8.1 Ontology and corpus study of the cultural heritage domain*, 2011. URL `http://www.molto-project.eu/`. Deliverable of EU Project MOLTO Multilingual Online Translation.

D. Dannélls, Damova M., Enache R., and Chechev M. Multilingual online generation from semantic web ontologies. In *Proceedings of the World Wide Web Conference (WWW2012)*, Lyon, France, 2012.

Dana Dannélls. The painting ontology. *Journal of applied ontologies*, 2011. Submitted.

Dana Dannélls, Mariana Damova, Ramona Enache, and Milen Chechev. A Framework for Improved Access to Museum Databases in the Semantic Web. In *Language Technologies for Digital Humanities and Cultural Heritage*, 2011.

Ramona Enache. Reasoning and language generation in the sumo ontology. Master's thesis, Chalmers University of Technology, 2009.

Ramona Enache and Krasimir Angelov. Typeful ontologies with direct multilingual verbalization. *Workshop on Controlled Natural Languages (CNL) 2010*, 2010.

Norbert E. Fuchs, Kaarel Kaljurand, and Tobias Kuhn. Attempto Controlled English for Knowledge Representation. In Cristina Baroglio, Piero A. Bonatti, Jan Małuszyński, Massimo Marchiori, Axel Polleres, and Sebastian Schaffert, editors, *Reasoning Web, Fourth International Summer School 2008*, number 5224 in Lecture Notes in Computer Science, pages 104–124. Springer, 2008.

R. Schwitter and M. Tilbrook. Controlled Natural Language meets the Semantic Web. In *Proceedings of the Australasian Language Technology Workshop*, pages 55–62, Macquarie University, 2004.

Sandra Williams, Allan Third, and Richard Power. Levels of organisation in ontology verbalisation. In *Proceedings of the 13th European Workshop on Natural Language Generation (ENLG)*, page 158–163, Nancy, France, September 2011.