



**HAL**  
open science

# Establishing Semantic Equivalences in Aircraft Ontology to Enable Semantic Interoperability

B. Damayanthi Jesudas, B. Gurumoorthy

► **To cite this version:**

B. Damayanthi Jesudas, B. Gurumoorthy. Establishing Semantic Equivalences in Aircraft Ontology to Enable Semantic Interoperability. 13th IFIP International Conference on Product Lifecycle Management (PLM), Jul 2016, Columbia, SC, United States. pp.319-328, 10.1007/978-3-319-54660-5\_29 . hal-01699724

**HAL Id: hal-01699724**

**<https://hal.inria.fr/hal-01699724>**

Submitted on 2 Feb 2018

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution| 4.0 International License

# Establishing semantic equivalences in aircraft ontology to enable semantic interoperability

B. Damayanthi Jesudas<sup>1</sup>, B. Gurumoorthy<sup>2</sup>

<sup>1,2</sup>Product Lifecycle Management Lab, Centre for Product Design and Manufacturing,  
Indian Institute of Science, Bangalore, India

<sup>1</sup>damayanthi@cpdm.iisc.ernet.in

<sup>2</sup>bgm@cpdm.iisc.ernet.in

**Abstract** In the use of ontologies to address semantic interoperability problems in PLM, either a lexicon or a list of synonyms of the terms in the domain are used to establish semantic mapping. This paper proposes a methodology to establish semantic equivalence of concepts with those in a given ontology based on some textual information about the new concept. This approach involves the classification of the information input into a structure form using natural language processing tools. Aircraft ontology available in literature, is taken as reference domain ontology for this purpose. Text input about the concept is parsed and the parts of speech (POS) tags are analyzed to obtain a structured representation. The structure is then compared to that in the domain ontology to decide if the input concept is part of the reference ontology and if so the concept it is equivalent to is identified. Results from an implementation of this procedure are shown and further work discussed.

**Keywords:** Product Lifecycle Management (PLM), Semantic interoperability, Aircraft ontology, A320 aircraft, POS tagging.

## 1 Introduction

Product Lifecycle Management (PLM) “is a systematic, controlled method for managing and developing industrially manufactured products and related information. The core of PLM (product lifecycle management) is the creation, preservation and storage of data relating to the company’s products and activities” [1]. “A major requirement for efficient PLM is the traceability of the product which is the acquirement of information along the product’s lifecycle about the product” [2]. The information and knowledge resources used during the product development process emerge from different application domains like design, manufacturing, support, production, services, so on and these are developed by experts from the respective domains. Multiple source vendors, contract manufacturers, distribution, and sales partners also add importance to the product by using existing information [3]. “Especially knowledge that is produced from different expert domains is not comprehensible for everybody, though the access to knowledge from these expert domains by others from outside the expert domain is inevitable” [4]. One common

problem in acquiring knowledge is the use of terms that are particular to the group or the expert responsible for the knowledge. This lack of shared vocabulary across users is referred to as the semantic interoperability problem [5].

Semantic interoperability refers to “the ability of different organizations to understand the exchanged data in a similar way” [6]. Semantic interoperability problem arises due to semantic heterogeneity, which occurs whenever two contexts do not share the same interpretation of information [7]. The causes of semantic heterogeneity [8] are,

1. Decentralized content generation. Geographically and functionally distributed teams are involved in the development process [9, 10], and the teams use different ways of representing product data.
2. Multiple perspectives (conceptualizations) of the reality. Each individual has different interpretations as they have different backgrounds and expertise. For example, ‘afterburner’ is also termed as ‘reheater’ in an aircraft and ‘reheater’ is also termed as ‘superheater’ in Energy/Power plants.
3. Ambiguity, inconsistency and vagueness. This happens because of the usage of different terminologies. For example, the part combustion chamber may appear in documents as blast chamber, combustion chamber, firing chamber or combustor [11].

The above mentioned are the main causes of semantic heterogeneity which leads to semantic interoperability problems. Thus, it is evident that different information resources/applications may use different terminologies for the same entity. Even when applications use the same terminology, they often associate different semantics with the terms or represent different parts/systems [11]. Also, “semantics associated with a term in different applications are different” [12]. Thus, it is difficult to achieve semantic interoperability by merely sharing terminologies, as semantic interoperability requires explicit definition of specifying the semantics of the terminologies. This lack of explicit definition of the semantics hinders the information retrieval process.

Ontologies are used to address semantic interoperability as these provide better understanding and processing of information in the form of concepts and relations associated between them. These are used for knowledge representation, as they specify the knowledge in a well-defined and unambiguous manner. The useful features of ontologies are, to share common understanding of the information among different application systems and to reuse the existing domain knowledge. “Ontologies support interoperability by enabling the specification of semantics-preserving mappings between the terminologies of different applications. “If ‘term’ in Application A and ‘term’ in Application B have logically equivalent definitions, then the mapping of these two terms preserves their semantics” [12].

This paper addresses the issue of mapping concepts to a given domain ontology. The result of mapping could be that the concept exists in the given ontology in which case semantic equivalence (important for addressing semantic interoperability) is established. The other two options are that the concept belongs to the ontology but is not present in it. In this case the ontology gets automatically populated. The last case is that the concept does not belong to the domain of the ontology. Traditionally the first case of establishing semantic equivalence has been achieved with keyword

search with the aid of a lexicon or through the use of synonyms. These approaches do not solve the problem as lexicon may not be complete to provide meanings to all terminologies. For example, terminologies like ‘combustion turbine’ an equivalent of ‘gas turbine’, ‘combustor’ equivalent of ‘combustion chamber’ etc., are not found in WordNet. Therefore, it is proposed to give some text as information about the concept and from this unstructured data, structured information is obtained that can then be compared with the domain ontology. This aids to analyse, compare and process the information easily. Here, mapping is between the user’s input and the domain ontology. For this purpose, example aircraft ontology [13] is taken as reference ontology. The text given by the user as input to query the ontology is parsed using Stanford dependency parser [14] for assigning parts of speech tags to each word in the text as noun, verb, adjective, etc. This is analysed to construct a structured representation. Then semantic equivalence is established between user’s input and ontology and semantic information from text is extracted.

## **2 State of the art**

Ontologies are being used for a long time to support semantic interoperability as these define the domain knowledge explicitly. Ciocoiu et al [12] gave an overview of the use of ontologies to facilitate interoperability among manufacturing process applications. It is indicated that interoperability is affected by the context in which it is viewed, different representations of the same information, using different concepts and terminology for the same terms in different contexts to mean different things. In the manufacturing scenario, the problem is that each of the teams will be using different software applications to represent and reason about the products and processes from their particular perspectives.

Stoimenov et al [15] presented architecture for semantic heterogeneity of geographic information based on ontologies. A semantic mediator called GeoNis is provided as a solution, which is a framework for interoperability of GIS applications.

Bittner et al [16] explained the use of ontologies in facilitating interoperability between software systems used in computer aided design, architecture engineering and construction, and geographic information processing. Orgun et al [17] discussed the various approaches to providing interoperability to multiple heterogeneous ontologies, which is still primarily a semi-automated process.

Zhan et al [18] proposed an ontology-based method for discovery and retrieval of geographic information to overcome shortage of approaches based on keywords. A top level ontology is used to find relevant concepts. Besides, implementation for this should be developed. Felic et al [4] presented an approach for process-oriented semantic management that allows management and sharing of knowledge across the product lifecycle. Knowledge links are introduced between knowledge base extensions for knowledge to be understandable between experts, which are a way to achieve semantic interoperability. It is from the perspective of process chain rather than product model. Creating and establishing knowledge links for collaborators of different domains is yet to be worked on.

From the above literature it is evident that a lot of study has been done on the use of ontologies in addressing semantic interoperability. Also, proposals are made in using ontologies as semantic mediators to address semantic interoperability, implementation of which however has not been reported. Ontologies are also employed in document retrieval using semantic similarity measures. In most cases, a top level/local ontology is used for aiding the semantics. Semantic interoperability between ontologies is addressed through ontology mappings. It is proposed that two data types are semantically equivalent if they have the same defining attributes and operations and the same set of constraints on those attributes and operations [17]. Unlike the approaches discussed above, this work classifies a concept with respect to a given ontology without using similarity measures.

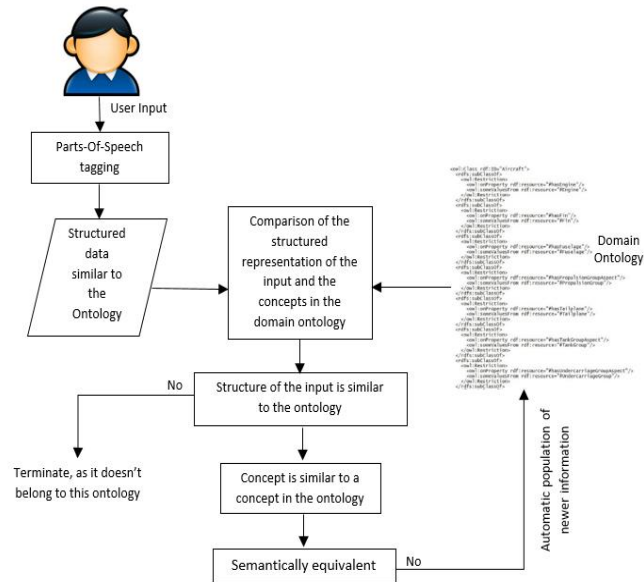
### **3 Methodology**

In the use of ontologies to address semantic interoperability the difficulty is, the use of a lexicon corresponding to the domain or a list of synonyms for each concept in the ontology, that assist in mapping for similar meaning terms. Both approaches, development of lexicon specific to a domain or identification of synonyms for each concept are very cumbersome and completeness of either the lexicon or the list of synonyms can never be guaranteed. An alternate approach is proposed in this paper, which uses NLP techniques for classifying a concept with respect to a given ontology. It is proposed to take some textual information about the concept as input. The user input text is parsed and the parts of speech (POS) tags are analyzed to obtain a structured representation that is then compared with the domain ontology. There are three outcomes of the classification. They are:

1. The concept is similar to a concept in the ontology. In this case semantically equivalence between the two is recorded.
2. The structured form in which the input concept is captured is similar to the structure for concepts in the ontology but the concept is not similar to any concept in the ontology. In this case the concept is added as a new concept to the ontology.
3. The structured form in which the input concept is captured is different from the structure for concepts in the ontology.

The methodology developed is shown in figure 1. The input text is parsed for (POS) Parts-Of-Speech tagging using available natural language processing tools.

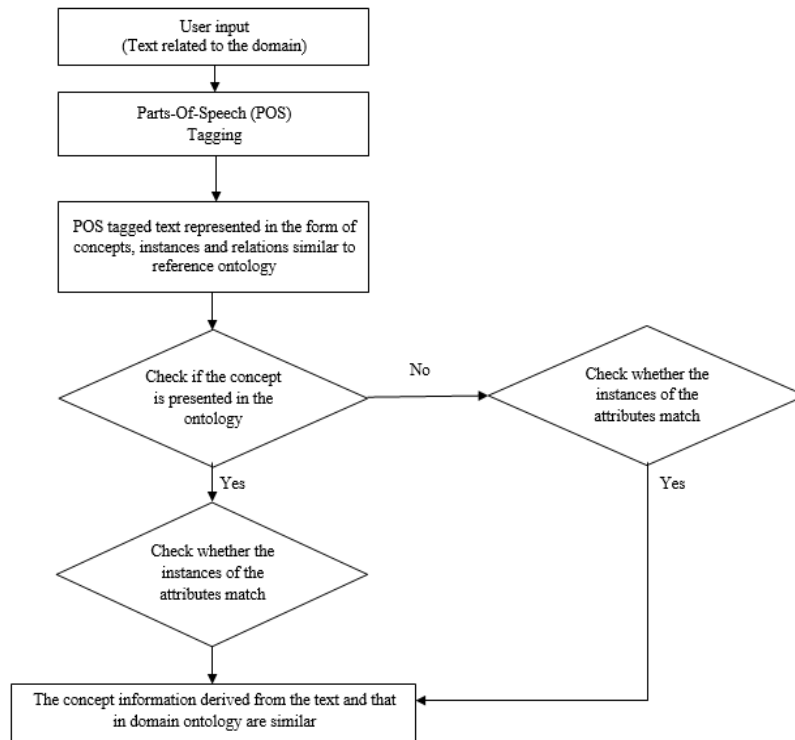
From the POS tagged texts, determiners, prepositions, conjunctions, verb forms etc., are filtered. As the ontology has parts and parameters, verb forms are also filtered. POS tags that falls under different types of nouns, adjectives that describe the nouns and cardinal numbers are considered. The words in the text that are tagged are then analyzed to identify a structured representation for the concept described by the text.



**Fig. 1.** Methodology developed for addressing semantic interoperability for a given domain ontology.

Firstly, the extracted information is searched for similar concepts in the ontology. As some terms appear in more than one concept structure, all terms that appear in a particular concept structure are considered for establishing equivalence. That is, aircraft has engine, fuselage etc., and engine is connected to fin, fuselage and wing. So, fuselage appears both in aircraft attributes as well in engine attributes with a different relation. Primarily in this work all terms appearing in a single concept structure are considered. And the above stated existence of terms in multiple concept structures is another issue which is to be addressed. The first condition is, if the concept is present in the ontology then the attributes corresponding to the concept are checked and if they are same, equivalence is established. The second condition is, if concept is not present in the ontology, then also the attributes are checked and if the attributes are similar for a particular concept, then equivalence is established between them

The flow chart for algorithm for mapping the obtained structure of information and ontology is shown in figure 2.



**Fig. 2.** Flow chart for the algorithm for mapping

## 4 Preliminary results and discussion

The proposed methodology has been implemented. The domain ontology used is, an aircraft ontology available in the public domain [13]. This is built using NeOn Methodology, with four first level basic classes, “Aircraft”, “AircraftSubComponent”, “AircraftAspect”, “AircraftParameter”. Figure 3 shows part of the aircraft ontology. The procedure for obtaining a structure for the input concept and comparison with the domain ontology has been implemented in C# on a Windows platform. Input text that describe concepts is taken from information available on the A320 aircraft online [19, 20, 21]. Some inputs are given generally with A320 data to test the system. First, the input text was chosen such that they contained terminologies related to aircraft domain like ‘aircraft’, ‘engine’, ‘fin’, ‘thrust’, ‘cabin’. These were used as seed words which aid in obtaining the structure. Type of inputs with examples for the three subcases of the first case, semantically equivalent information is provided.

```

- <owl:Class rdf:ID="#Aircraft">
- <rdf:subClassOf>
- <owl:Restriction>
  <owl:onProperty rdf:resource="#hasEngine"/>
  <owl:someValuesFrom rdf:resource="#Engine"/>
</owl:Restriction>
</rdf:subClassOf>
- <rdf:subClassOf>
- <owl:Restriction>
  <owl:onProperty rdf:resource="#hasFin"/>
  <owl:someValuesFrom rdf:resource="#Fin"/>
</owl:Restriction>
</rdf:subClassOf>
- <rdf:subClassOf>
- <owl:Restriction>
  <owl:onProperty rdf:resource="#hasFuselage"/>
  <owl:someValuesFrom rdf:resource="#Fuselage"/>
</owl:Restriction>
</rdf:subClassOf>
- <rdf:subClassOf>
- <owl:Restriction>
  <owl:onProperty rdf:resource="#hasPropulsionGroupAspect"/>
  <owl:someValuesFrom rdf:resource="#PropulsionGroup"/>
</owl:Restriction>
</rdf:subClassOf>

```

**Fig. 3.** Example aircraft ontology [13].

For an example text, ‘A320 aircraft is powered by CFM56 engine with a single-aisle fuselage’, the POS tagged text is ‘A320/NNP aircraft/NN is/VBZ powered/VBN by/IN CFM56/CD engine/NN with/IN a/DT single-aisle/JJ fuselage/NN’. /. From the tagged text, noun (NN), proper noun (NNP), cardinal number (CD), adjective (JJ) are considered and verb forms (VBZ, VBN), preposition (IN), determiner (DT) are filtered, which ends up in useful information. From the above tagged text, the content that is of interest is (A320/NNP aircraft/NN), (CFM56/CD engine/NN) and (single-aisle/JJ fuselage/NN). It is evident that the terminologies ‘aircraft’, ‘engine’ and ‘fuselage’ help in information extraction from input text. The aircraft, noun (NN) is identified by the proper noun (NNP), A320. The ‘engine’ by cardinal number (CD) ‘CFM56’ and ‘fuselage’ by the adjective (JJ) ‘single-aisle’. The information is extracted using C# and the extracted representation for this is shown in figure 4.

A320 aircraft is powered by CFM56 engine with a single-aisle fuselage.		
Aircraft	-	A320
Engine	-	CFM56
Fuselage	-	single-aisle

**Fig. 4.** Extracted representation for a given input

Next, input text that did not contain any keywords from the reference ontology was considered. For example, in the text ‘CFM56 has a bypass ratio 6:1, generating 150 KN thrust’, the term ‘engine’ is not stated for ‘CFM56’. The POS tagged for this text is, ‘CFM56/NNP has/VBZ a/DT bypass/NN ratio/NN 6:1/CD ./, generating/VBG 150/CD kN/NN thrust/NN’./.. The extracted representation for this is shown in figure 5. Here based on comparison with the domain ontology, the CFM56 is labeled as an instance of concept Engine.

CFM56 has a bypass ratio 6:1, generating 150 KN of thrust.		
CFM56		
Bypass ratio	-	6:1
Thrust	-	150KN

**Fig. 5.** Extracted representation for a given input



Figure 6 shows part of the domain ontology and figure 7 shows indication of established equivalence. The text provided by user is explicit, meaning it is clearly stated that A320 is a type of aircraft. It is clearly shown in the ontology, that the top level concept doesn't contain the seed word. In this case, the value of the seed word plays pivotal role and it is used for mapping. Since, all attributes are matching it is indicated as 'match'.

```

- <owl:Class rdf:ID="A320">
- <rdfs:subClassOf>
- <owl:Restriction>
  <owl:onProperty rdf:resource="#hasEngine"/>
  <owl:hasValue rdf:resource="#CFM56"/>
</owl:Restriction>
</rdfs:subClassOf>
- <rdfs:subClassOf>
- <owl:Restriction>
  <owl:onProperty rdf:resource="#hasFuselage"/>
  <owl:hasValue rdf:resource="#single-aisle"/>
</owl:Restriction>
</rdfs:subClassOf>
</owl:Class>

```

**Fig. 6.** Part of domain ontology with A320 aircraft data

Input

```

aircraft: A320
engine: CFM56
fuselage: single-aisle

```

Show Output

Input	Ontology	Indication
aircraft: A320	A320	<b>Match</b>
engine: CFM56	Engine: CFM56	<b>Match</b>
fuselage: single-aisle	Fuselage: single-aisle	<b>Match</b>

**Fig. 7.** Establishing semantic equivalence for user input and domain ontology

Figure 8 shows the other case, wherein the top level concept doesn't match but the entire attributes match. This information is considered similar and is indicated to the user. In this case, it is evident that user has given similar word in different form and the similarity in the values of the attributes help in resolving this problem. As there are numerical values describing aircraft parameters, mapping is based on their equivalence. Also, the extracted information contains terms whereas the information in the ontology contains 'relation' plus the term like 'has' engine. While mapping the relations are replaced with only terminologies.

Input	<pre>tailplane: vertical taper ratio: 0.33 aspect ratio: 1.56</pre>	
<input type="button" value="Show Output"/>		
<b>Input</b>	<b>Ontology</b>	<b>Indication</b>
tailplane: vertical	Fin	<b>Match</b>
taper ratio: 0.33	TaperRatio: 0.33	<b>Match</b>
aspect ratio: 1.56	AspectRatio: 1.56	<b>Match</b>

**Fig. 8.** Establishing semantic equivalence for second case of user input and domain ontology

Also, the text given as input should be grammatically correct, as it is parsed for parts of speech tagging with a standard parser. Grammatical errors may cause inappropriate tagging by the parser because of the misplacement of parts of speech which leads to inappropriate mapping. Therefore, it is advisable to give well-formed text as input, which can be interpreted correctly.

## 5 Conclusions

The use of ontologies in supporting semantic interoperability has problems owing to the problem of different semantics for terminologies. Even, the use of a lexicon or a list of terms doesn't assist in mapping as they don't provide meanings to all terminologies. An alternate approach is proposed in this paper for classifying a concept with respect to a given ontology. It is proposed to take some textual information about the concept as input. The user's input is parsed using Stanford online dependency parser and the parts of speech (POS) tagged text is analysed for extracting information similar to that of the reference ontology. Available aircraft ontology is taken as reference ontology. Text on 'Airbus A320' aircraft is used as input. Three issues that occur while mapping the extracted information from the input text to the reference ontology are identified. They are semantically equivalent information, information that doesn't exist in the ontology and information that doesn't belong to the ontology. In this work, the first case has been addressed. The second and third cases are to be addressed and work is ongoing for the same.

## References

1. Saaksvuori, A., Immonen, A.: Product Lifecycle Management. Springer-Verlag Berlin Heidelberg, New York (2004)
2. Matsokis, A., Kiritsis, D.: An Ontology-Based Approach for Product Lifecycle Management. Journal Computers in Industry, 61(8), pp. 787-797 (2010)

3. Dutta, D., Wolowicz, J.P.: An Introduction to Product Lifecycle Management (PLM). In: 12th ISPE International Conference on Concurrent Engineering: Research and Applications. Fort Worth/Dallas, TX, USA (2005)
4. Felic, A., Konig-Ries, B., Klein, M.: Process-Oriented Semantic Knowledge Management in Product Lifecycle Management. In: 8th International Conference on Digital Enterprise Technology, Procedia CIRP, vol.25, pp. 361-368 (2014)
5. Jesudas, D. B., Gurumoorthy, B.: Natural Language Processing (NLP) Driven Ontology Modeling to Address Semantic Interoperability in Information Retrieval. In: 5th International Conference on Product Lifecycle Modelling, Simulation and Synthesis. Hyderabad (2015)
6. Sharma, S., Goyal, S.B., Shandliya, R., Samadhiya, D.: Towards XML Interoperability. Advances in Computer Science, Engineering & Applications, vol. 166, pp. 1035-1043 (2012)
7. Mao, M.: Ontology Mapping: Towards Semantic Interoperability in Distributed and Heterogeneous Environments. PhD thesis, University of Pittsburgh (2008)
8. Freitas, A.: Coping with data variety in the big data era: The semantic computing approach, Insight Centre for data analytics, Rio big data meetup, <http://www.slideshare.net/andrenfreitas/coping-with-data-variety-in-the-big-data-era-the-semantic-computing-approach> (visited on 24/11/2015)
9. Boeing, <http://787updates.newairplane.com/787-Suppliers/World-Class-Supplier-Quality> (visited on 24/11/2015)
10. Boeing homepage, <http://787updates.newairplane.com/787-Suppliers/World-Class-Supplier-Quality> (visited on 24/11/2015)
11. Gupta, R.K., Gurumoorthy, B.: Shape-Function-Relationship (SFR) Framework for Semantic Interoperability of Product Model. International Conference on Product Lifecycle Management, PLM 2009, Bath (2009)
12. Ciocoiu, M., Gruninger, M., Nau, D.: Ontologies for Integrating Engineering Applications. Journal of Computing and Information Science in Engineering, vol.1, pp. 12-22. American Society of Mechanical Engineers New York (2001)
13. Aircraft-Ontology, <https://github.com/astbhltum/Aircraft-Ontology> (visited on 28/1/2016)
14. Stanford dependency parser online, <http://nlp.stanford.edu:8080/parser/> (visited on 2/2/2016)
15. Stoimenov, L., Stanimirovic, A., Djordjevic-Kajan, S.: Semantic Interoperability Using Multiple Ontologies. In proceedings: 8th AGILE Conference on GIScience, pp.261-270. Estoril, Portugal (2005)
16. Bittner, T., Donnelly, M., Smith, B.: Ontology and Semantic Interoperability. Journal Large-scale 3D Data Integration: Challenges and Opportunities, pp. 139-160 (2005)
17. Orgun, B., Dras, M., Nayak, A., James, G.: Approaches for Semantic Interoperability between Domain Ontologies. Journal Expert Systems, vol.25, pp.179-196 (2008)
18. Zhan, Q., Zhang, X., Li, D.: Ontology-Based Semantic Description Model for Discovery and Retrieval of Geo-Spatial Information. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 32 (2008)
19. A320, <http://www.airbus.com/aircraftfamilies/passengeraircraft/a320family/a320/> (Visited on 02/02/16)
20. A320, <http://www.airbus.com/aircraftfamilies/passengeraircraft/a320family/a320/performance/> (Visited on 02/02/2016)
21. CFM56, [https://en.wikipedia.org/wiki/CFM\\_International\\_CFM56](https://en.wikipedia.org/wiki/CFM_International_CFM56) (Visited on 02/02/2016)