pHealth 2022
B. Blobel et al. (Eds.)

doi:10.3233/SHTI220962

CTS2 OWL: Mapping OWL Ontologies to CTS2 Terminology Resources

Sara MORA [a], Bernd BLOBEL[b], Roberta GAZZARATA [c] and Mauro GIACOMINI [a,1]
[a] *Department of Informatics, Bioengineering, Robotics and System Engineering (DIBRIS), University of Genoa, Italy*
[b] *Medical Faculty, University of Regensburg, Regensburg, Germany*
[c] *Healthropy s.r.l., Savona, Italy*

Abstract. The advancement of healthcare towards P5 medicine requires communication and cooperation between all actors and institutions involved. Interoperability must go beyond integrating data from different sources and include the understanding of the meaning of the data in the context of concepts and contexts they represent for a specific use case. In other words, we have to advance from data sharing through sharing semantics up to sharing clinical and medical knowledge. According to the Good Modeling Best Practices, we have to start with describing the real-world business system by domain experts using Domain Ontologies before transforming it into an information and communication technology (ICT) system, thereafter specifying the informational components and then transforming the system into an implementable solution. Any representation style – in the system development process acc. to ISO 10746 called system view – is defined by a related ontology, to be distinguished from real-world domain ontologies representing the knowledge spaces of involved disciplines. The system enabling such representational transformation shall also support versioning as well as the management of historical evolutions. One of such systems is the Common Terminology Service Release 2 (CTS2), which is a standard that allows the complete management of terminological contents. The main objective of this work is to present the choices we made to transform an ontology, written in the standard *Ontology Web Language* (OWL), into the CTS2 objects. We tested our transformation approach with the Alzheimer's Disease Ontology. We managed to map all the elements of the considered ontology to CTS2 terminological resources, except for a subset of elements such as the equivalentClass derived from restrictions on other classes.

Keywords. Ontology, CTS2, semantic interoperability, terminology resources, biomedical field

1. Introduction

Healthcare systems currently undergo a transformation towards integrated, interoperable, knowledge-based, policy-driven, highly dynamic, and fully distributed ecosystems according to the personalized, preventive, predictive, participative precision (P5) medicine paradigm [1]. This requires communication and cooperation of actors from multiple disciplines with specific perspectives, contexts, objectives, using their special methodologies, languages, knowledge, and skills. The challenge of P5 medicine

[1] Corresponding Author: Mauro Giacomini, University of Genoa, Via All'Opera Pia 13, 16145 Genoa, Italy; E-mail:mauro.giacomini@unige.it

ecosystems is the proper representation, mapping and matching of their domain-specific knowledge at any representation level from the real-world business system through the related IT systems, the information and finally the data used, deploying more and more constrained languages from natural languages up to computational ones. Thereby, mapping between elements from different domains or different viewpoint can only be performed at horizontal level, i.e., at the same level of granularity. To get there, components must be specialized or generalized, respectively. The corresponding representation of system-oriented, multi-domain, ontology-based, policy-driven P5 ecosystems using ISO 23903 [2] is shown in Figure 1.

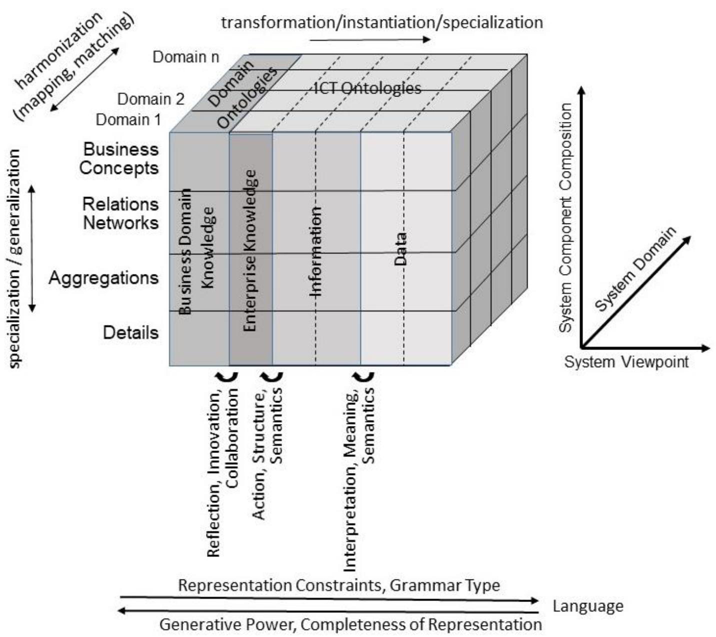

Figure 1. Model and framework for representing multi-domain, knowledge-based, ontology-based, policy-driven ecosystems.

While, from an ICT system development process perspective, that transformation into the viewpoint-specific representation style is clearly defined, e.g. in ISO/IEC 10746 Open distributed processing [3], the transformation of health systems based on existing standards is an open issue to be addressed in this paper. Thereby, we have to solve the mapping between domains represented by domain ontologies and the representational transformation between different viewpoints. Examples for the data view representation are database management system schemas or coding systems. Regarding the semantical representation, terminologies, thesauri, taxonomies, glossaries, data dictionaries, or vocabularies are used. The aforementioned semantical resources are collection of terms (entities) that are linked to a specific domain. They aim at creating a complete documentation that support the correct usage of such terms.

An ontology is *an explicit specification of a conceptualization* [4], going beyond terminologies by describing all the specific knowledge about its entities including their contexts, constraints and relationships by using attributes and relations [5]. Its main objective is to facilitate knowledge sharing unambiguously. Therefore, an ontology constitutes an efficient strategy to aggregate, standardize, represent and communicate biomedical concepts and related information as well as, together with machine learning (ML) methods, it can be used for automatic reasoning [6] and predictions [7]. In the biomedical field, ontologies became particularly important for semantic interoperability and data integration [8,9] as they could be considered, for each medical concept, as the justification of the specification [10].

However, we should consider that the evolution of medical knowledge might cause changes both in the concepts and in the terms that are used to define them. So, it is necessary to use systems that allow versioning and management of the historical evolution of information and data [11]. As there exist many different types of ontologies, it is necessary to manage the challenge of creating, maintaining and mapping ontological concepts. Here, the ISO 21838 Top level ontologies [12] shall be used.

Many different tools were developed, in various languages and countries, to deal with the important problem of medical terminology management [13,14,15], but their usage was highly localized. In the following years, a more general and standard solution was widely adopted by the scientific community, the *Common Terminology Service Release 2* (CTS2) [16,17]. The CTS2 standard provides a universally implementable set of operations that can be used for the complete management of terminological contents. In particular, its functional profiles allow the creation, updating, deletion, reading, searching and versioning of terminology resources. CTS2 was created with the aim of providing a structure suitable for organizing information by using consistent and computable mechanisms. Therefore, it supports the improvement of information exchange quality.

The main objective of this work is to present the choices we made to transform the elements of one ontology presented in the standard *Ontology Web Language* (OWL) into CTS2 terminology resources. An ontology has a great descriptive ability, and the CTS2 helps in maintaining its integrity over the versions as it keeps track of all changes made on a specific version. To the best of our knowledge, this is the first work that reports this kind of transformation.

2. Methods

The purpose of our work is to document in detail all the choices we made in the process of transformation of an ontology into CTS2 resources.

2.1. Ontology Elements & Relations

A medical ontology is mainly characterized by the definition of a domain that comprises all the knowledge attributable to a given topic. Some examples of relevant medical ontologies, ranging over different areas, are: *Human Phenotype Ontology* (HPO) [18], *Infectious Diseases Ontology (IDO)* [19], *Epilepsy and Seizures Ontology* (EPSO) [20]. Within an ontology, the various objects are defined and interact with each other according to logical properties, reciprocal restrictions between and on objects, groups and semantic sets. The three main types of objects of an ontology are:

- **Class**: every single object belonging to the domain.
- **Annotation Properties**: further information purely attributable to the Class itself and independent of the others. Examples from the Alzheimer's Disease Ontology (ADO) are synonym, comment, label.

– **Object Properties**: A restriction, which places a limit on the values of a certain Class that respects a certain Property. In most cases, this Property is defined within the ontology, so it has logical value only within it.

The *rdfs: subClassOf* construct is implemented in OWL with the same logical value and meaning it has in the RDF Schema2, that is: if a Class C1 is defined as a subClassOf of another Class C2, then the set of elements that make up C1 they must at least be a subset of those that make up C2. A Class is therefore by definition a subClass of itself, as the subset can also be the whole set.

The *equivalentClass* construct, on the other hand, indicates that the set of elements of Class C1 are equivalent to those of Class C2, therefore the two sets must have exactly the same number and the same elements.

2.2. CTS2 Terminology Resources

CTS2 standard is distributed through the HL7 *Service Functional Model* (SFM) and the *Object Management Group* (OMG) *Service Technical Model* (STM), that respectively provide service interface specification at a functional level and technical requirements of the service3. Within this work, we will refer to the OMG model.

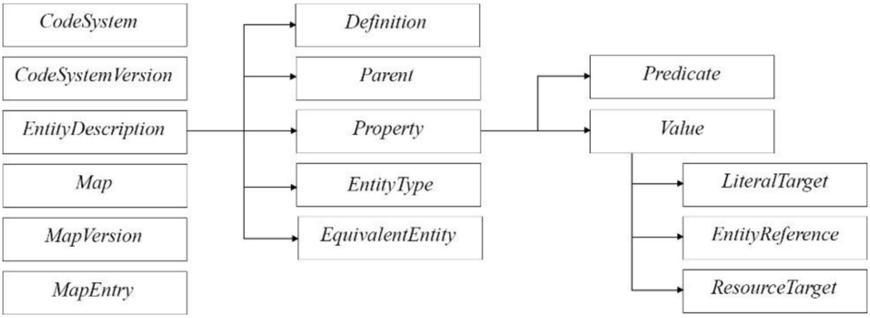

Figure 2. Hierarchical structure of the CTS2 resources that we used to map the ontology elements.

The CTS2 terminology resources summarized in Figure 2 are:

– **CodeSystem**: a classification system or a code system or an ontology or a thesaurus, etc. It includes information about publisher, release cycles, purpose, etc.
– **CodeSystemVersion**: a version of a CodeSystem. It includes information about release date, release format, contact information, etc.
– **EntityDescription**: a description about a class or a role or an individual within a specific CodeSystemVersion.
– **Map**: a collection of rules for transforming entities of a CodeSystem into entities represented in a second one. It includes information about creators, intended use, CodeSystem involved, etc.
– **MapVersion**: a specific version of a Map that maintains both the original and the destination CodeSystemVersion.

---

2 https://www.w3.org/TR/rdf-schema/

3 http://www.omg.org/spec/CTS2/1.2/

- **MapEntry**: a definition of a set of rules that identifies how a single Entity that belongs to the original CodeSystemVersion maps to null, one or more target Entities that belong to the destination CodeSystemVersion.

The core CTS2 resource is the EntityDescription. It is the most complex but interesting item to investigate. Some of its metadata that we considered during the translation process are:

- **Definition**: An explanation of the intended meaning of a concept. An EntityDescription may have multiple definitions, each derived from a different source, represented in a different language or having a different purpose.
- **Parent**: The set of direct “parents” asserted by DescribingCodeSystemVersion. It is the responsibility of the service to determine what predicate(s) represent “parent/child” relationships.
- **Property**: Additional “non-semantic” (annotation) assertions about the entity being described that do not fit into the other categories.
- **EntityType**: The set of type(s) a resource can take and it should include owl:Class, owl:Individual, rdf:Property, or skos:Concept, although it may carry many other types as well.
- **EquivalentEntity**: An entity that has been determined to be equivalent to the about entity in the context.

The element we mainly focused on is **Property** and its two main components are:

- **Predicate**: The name or URI of the property predicate. It can be literal or an EntityDescription itself, so an Annotation Property or an Object Property.
- **Value**: The target(s) of the property. Note that this can only represent the literal format of the property. The details about the original property will be found in the *CorrespondingStatement* if the CTS2 implementation supports the statement profile. So, the attribute value of a property is of Class *StatementTarget* and it can be of three types:
  1. Literal Target: when the statement type is LITERAL. It can be used for properties like the entity “label” or “comment”.
  2. Entity Reference Target: the URI and optional namespace/name when the target type is ENTITY. It can be used when a property refers to another entity.
  3. Resource Target: when the statement type is RESOURCE.

An entity may have more than one value for the same predicate, so it is necessary to create a list of *StatementTargets* containing all the items and then assign the list to *Property.Value*, while *Property.Predicate* remains unchanged.

3. Results

The ontology we tested our mapping system on is the *Alzheimer's Disease Ontology* (ADO) [21]. The list of CTS2 resources that we used in our translation process is composed by the following three main elements:

- **CodeSystem**. The functional profile involved is the *CodeSystem Catalog MaintenanceService*. Two of the input parameters that we provided were: *Uniform Resource Identifier* (URI), external link of the resource; *Name*, the local identifier of the catalog that we want to create.
- **CodeSystem Version**. The functional profile involved is the *CodeSystem Version Catalog Entry*. As for the CodeSystem one the input parameters is *Name*, a local

descriptor that uniquely identifies this version in the specific CodeSystem. Another input parameter is the *versionOf* which the name or URI of the code system that the version belongs to.

– **EntityDescription**. The functional profile involved is the *Entity Description Maintenance Service*. For each entity two important input parameters are: *EntityID*, the entity code and/or namespace identifier of the entity to be created; *Describing-CodeSystemVersion*, the URI or local identifier of the CodeSystemVersion that this entity belongs to. Once created, each entity can be updated to insert the information linked to each single ontology class or property. The components of an ontology element can be divided in two main categories: components that can be directly mapped to an element of the CTS2 EntityDescription and elements that need to be mapped into properties because they do not fit into any predefined item.

The first category includes the elements in Table 1. Among them, Example and Definition are lists that could have zero or more items, while at least one Parent is always present.

Table 1. Correspondence between elements of an ontology class that can be directly mapped to CTS2 EntityDescription components.

Ontology concept metadata	**CTS2 resource metadata**
Example	.Example
IsDefinedBy	.Definition
SubClassOf	.Parent
EquivalentClass	.EquivalentEntity

The elements that belong to the second category can be divided into the four main groups displayed in Table 2.

Table 2. Four cases to which all the elements, not contained in Table 1, of an ontology class/property can belong to: (a) predicate is a text and value is a text, (b) predicate is a text and value is an entity, (c) predicate is an entity and value is a text, and (d) predicate is an entity and value is an entity.

	Value is a text	**Value is an Entity**
Predicate is a text	The **predicate** contains the name and namespace of the statement predicate *(type: EntityNameOrURI)*. The **value** element is a statement target of type LITERAL *(type: Opaque- Data)*.	The **predicate** contains the name and namespace of the statement predicate *(type: EntityName-OrURI)*. The **value** element is a statement target of type ENTITY *(type: EntityNameOrURI)*.
Predicate is an Entity	The **predicate** contains the URI of the entity of type Annotation Property *(type: EntityNameOrURI)*. The **value** element is a statement target of type LITERAL *(type: Opaque-Data)*.	The **predicate** contains the URI of the entity of type Object Property *(type: EntityNameOrURI)*. The **value** element is a statement target of type ENTITY *(type: Entity-NameOrURI)*.

Figure 3 shows some examples of the translation applied to the selected ADO ontology.

```
<owl: Class rdf:about="&AlzheimerOntology;Behavioral_therapies">
 <rdfs:label>Behavioral_therapies</rdfs:label>
 <rdfs:SubClassOf rdf:resource="&span;Process"/>
 <rdfs:SubClassOf>
  <owl:Restriction>
   <owl:onProperty rdf:resource="&NDDUO;is_entity_used_in"/>
   <owl:someValuesFrom rdf:resource="&NDDUO;non_pharmacotherapy_treatment_context"/>
 </rdfs:subClassOf>
 <rdfs:isDefinedBy>
  Behaviour therapy of behaviour therapy is an approach to psycotherapy in the
  behaviourism tradition that focuses on a set of methods designed for reinforcing
  desired and eliminating undesired behaviours without concerning itself with the
  psychoanalytic state of the subject.
 </rdfs:isDefinedBy>
 ...
</owl:Class>

<owl: Class rdf:about="&snap;Continuant">
 <rdfs:label rdf:datatype="&xsd;string">continuant</rdfs:label>
 <owl:disjointWith rdf:resource="&span;Occurrent"/>
 ...
</owl:Class>
```

Annotation Property and Literal Target
EntityDescription.Parent
Object Property and Entity Reference
EntityDescription.Definition
Textual predicate and Entity Reference

Figure 3. Examples of ADO elements translated into CTS2 terminology resources.

Label, highlighted in orange, belongs to case (c) as the predicate is an Annotation Property and value is text. *SubclassOf* with a restriction, highlighted in green, belongs to case (d) as the predicate is an Object Property *("is entity used in")* and the value is the entity *"non pharmacoterapy treatment context"*. *disjointWith*, highlighted in purple, belongs to case (b) as the predicate is textual and the value is the entity *"continuant"*. Figure 3 also shows two examples of elements that belong to Table 1. In particular, *SubclassOf alone* is the element that defines a hierarchical relation, so the element contained between the two tags is one of the entities of the ontology that constitutes the parent. On the contrary, *IsDefinedBy* presents a specific information about the entity and it is composed of free text.

4. Discussion

The development and usage of standard terminologies became extremely important in the scenario of data integration and improved the quality of the resulting outcome. However, terminologies evolve, new terms are inserted while others are withdrawn, so a system able to track all changes is necessary to maintain the integrity of the terminology resource. A standard solution that could be adopted is using the corresponding *Fast Healthcare Interoperability Resources* (FHIR) of HL7 Italy as shown in [22]. The major limitation of this approach is that only hierarchical relations are allowed, while extensions like important elements of the ontology could not be mapped, e.g. annotations and object properties. A non-standard solution is Protégé[4], which is an open-source tool that supports the creation and management of ontologies. In particular, it keeps track of changes and makes the revision history available. The main advantage in using Protégé is that it is specifically built for ontologies, so the system can deal with all ontology components [5]. A limitation is that as far as we know there is no *Application Programming Interface* (API) available allowing a rich set of operation, e.g. the creation and update of an ontology. Users can interact with the database only through the web interface (WebProtégé), thus requiring a huge human intervention. In our work, we

[4] https://protege.stanford.edu/

selected the Common Terminology Service Release 2 first because it is a standard solution and it is services-based, and second because we have already used it for the management of vocabularies in several scenarios [16,23,24]. The main advantage is that it provides a specification for the implementation of a set of operations that allow the complete management of all aspects of CTS2 resources. A limitation of this approach is however that there is a reduced subset of ontology relations, which we could not map into CTS2 resources, e.g., when *equivalentEntity* elements are obtained as restrictions on other entities.

In personalized medicine, it is essential to accurately describe the situation of a patient at all levels (from macroscopic [25] to molecular [26]) [27,28]. This kind of systems combines on the one hand the analytical nature of ontologies and on the other hand the systematic approach and rigor provided by standardized tools, such as the CTS2, to preserve the evolutionary history of the terminological systems. Therefore, we believe that it can be considered one of the enabling tools for the real implementation of the pHealth paradigms.

Acknowledgements

The Virtual Brain Cloud Project that has received funding from the European Union's Horizon 2020 research and innovation program under the grant agreement No 826421 has financed the presented work.

References

[1] Blobel B, Oemig F, Ruotsalainen P and Lopez DM. Transformation of Health and Social Care Systems—An Interdisciplinary Approach Toward a Foundational Architecture. Front. Med. 2022;9:802487. doi: 10.3389/fmed.2022.802487
[2] International Organization for Standardization. ISO 23903:2021 Health informatics – Interoperability and integration reference architecture – Model and framework. ISO: Geneva; 2021.
[3] International Organization for Standardization. ISO/IEC 10746:2009 Information technology – Open Distributed Processing (all parts). ISO: Geneva.
[4] Gruber TR. Toward principles for the design of ontologies used for knowledge sharing? International journal of human-computer studies. 1995;43(5-6):907-28.
[5] Rubin DL, Noy NF, Musen MA. Protege: a tool for managing and using terminology in radiology applications. Journal of digital imaging. 2007;20(1):34-46.
[6] Chen H, Luo X. An automatic literature knowledge graph and reasoning network modeling frame- work based on ontology and natural language processing. Advanced Engineering Informatics. 2019;42:100959.
[7] El-Sappagh S, Alonso JM, Ali F, Ali A, Jang JH, Kwak KS. An ontology-based interpretable fuzzy decision support system for diabetes diagnosis. IEEE Access. 2018;6:37371-94.
[8] Min H, Turner S, De Coronado S, Davis B, Whetzel T, Freimuth RR, et al. Towards a standard ontology metadata model. Faculty Research. 2016.
[9] Bandrowski A, Brinkman R, Brochhausen M, Brush MH, Bug B, Chibucos MC, et al. The ontology for biomedical investigations. PloS one. 2016;11(4):e0154556.
[10] Kiourtis A, Mavrogiorgou A, Kyriazis D. FHIR Ontology Mapper (FOM): aggregating structural and semantic similarities of ontologies towards their alignment to HL7 FHIR. In: 2018 IEEE 20th International Conference on e-Health Networking, Applications and Services (Healthcom). IEEE; 2018. p. 1-7.
[11] Mantas J, et al. Formal specification of an ontology-based service for EHR interoperability. Quality of Life Through Quality of Information: Proceedings of MIE2012. 2012;180:78.
[12] International Organization for Standardization. ISO/IEC 21838:2021 Information technology – Top-level ontologies (TLO). ISO: Geneva; 2021.

[13] Cimino JJ, Clayton PD, Hripcsak G, Johnson SB. Knowledge-based approaches to the maintenanceof a large controlled medical terminology. Journal of the American Medical Informatics Association. 1994;1(1):35-50.
[14] Lee S, Song SJ, Koh S, Lee SK, Kim Hg. National Medical Terminology Server in Korea. In: International Conference on Security-Enriched Urban Computing and Smart Grid. Springer; 2010. p. 541-4.
[15] Gazzarata R, Giannini B, Giacomini M. A SOA-based platform to support clinical data sharing. Journal of healthcare engineering. 2017;2017.
[16] Gazzarata R, Monteverde ME, Vio E, Gubian L, Borgo I, Giacomini M, et al. A terminology service compliant to CTS2 to manage semantics within the regional HIE. European Journal of Biomedical Informatics. 2017;13(1).
[17] Seerainer C, Sabutsch SW. eHealth Terminology Management in Austria. In: MIE; 2016. p. 426-30.
[18] Robinson PN, Mundlos S. The human phenotype ontology. Clinical genetics. 2010;77(6):525-34.
[19] Cowell LG, Smith B. Infectious disease ontology. In: Infectious disease informatics. Springer; 2010. p. 373-95.
[20] Sahoo SS, Lhatoo SD, Gupta DK, Cui L, Zhao M, Jayapandian C, et al. Epilepsy and seizure ontology: towards an epilepsy informatics infrastructure for clinical research and patient care. Journal of the American Medical Informatics Association. 2014;21(1):82-9.
[21] Malhotra A, Younesi E, Gundel M, Muller B, Heneka MT, Hofmann-Apitius M. ADO: A disease ontology representing the domain knowledge specific to Alzheimer's disease. Alzheimer's & dementia. 2014;10(2):238-46.
[22] Metke-Jimenez A, Lawley M, Hansen D. FHIR OWL: Transforming OWL ontologies into FHIR terminology resources. In: AMIA Annual Symposium Proceedings. vol. 2019. American Medical Informatics Association; 2019. p. 664.
[23] Canepa S, Roggerone S, Pupella V, Gazzarata R, Giacomini M. A semantically enriched architecture for an Italian laboratory terminology system. In: XIII Mediterranean Conference on Medical and Biological Engineering and Computing 2013. Springer; 2014. p. 1314-7.
[24] Mora S, Madan S, Gebel S, Giacomini M. Proposal of an Architecture for Terminology Management in a Research Project. In: Digital Personalized Health and Medicine. IOS Press; 2020. p. 1371-2.
[25] Lazarova E, Mora S, Maggi N, Ruggiero C, Vitale AC, Rubartelli P, et al. An Interoperable Electronic Health Record System for Clinical Cardiology. In: Informatics. vol. 9. Multidisciplinary Digital Publishing Institute; 2022. p. 47.
[26] Izzo M, Mortola F, Arnulfo G, Fato MM, Varesio L. A digital repository with an extensible data model for biobanking and genomic analysis management. BMC genomics. 2014;15(3):1-15.
[27] Degeling K, Koffijberg H, IJzerman MJ. A systematic review and checklist presenting the main challenges for health economic modeling in personalized medicine: towards implementing patient-level models. Expert review of pharmacoeconomics & outcomes research. 2017;17(1):17-25.
[28] Maggi N, Gazzarata R, Ruggiero C, Lombardo C, Giacomini M. Cancer precision medicine today: Towards omic information in healthcare systems. Tumori Journal. 2019;105(1):38-46.