Applied Ontology 16 (2021) 297–337
DOI 10.3233/AO-210251
IOS Press

A validation & verification driven ontology: An iterative process

Angelina Espinoza [a,*], Ernesto Del-Moral [b], Alfonso Martínez-Martínez [c] and Nour Alí [d]
[a] *Business Information Systems Department, University College Cork, 13 South Mall, T12 KW86, Cork, Ireland*
E-mail: angelina.espinozalimon@ucc.ie
[b] *College of Science and Technology, Universidad Autónoma de la Ciudad de México, Prolongación San Isidro 151, San Lorenzo Tezonco, C.P. 09790, México City, México*
E-mail: del.moral.arcos.ernesto@gmail.com
[c] *Electrical Engineering Department, Universidad Autónoma Metropolitana, Av. San Rafael Atlixco 186, Leyes de Reforma C.P. 09340, México City, México*
E-mail: almm@xanum.uam.mx
[d] *Computer Science Department, Brunel University London, Kingston Ln, London, Uxbridge UB8 3PH, UK*
E-mail: nour.ali@brunel.ac.uk

Abstract. Designing an ontology that meets the needs of end-users, e.g., a medical team, is critical to support the reasoning with data. Therefore, an ontology design should be driven by the constant and efficient validation of end-users needs. However, there is not an existing standard process in knowledge engineering that guides the ontology design with the required quality. There are several ontology design processes, which range from iterative to sequential, but they fail to ensure the practical application of an ontology and to quantitatively validate end-user requirements through the evolution of an ontology. In this paper, an ontology design process is proposed, which is driven by end-user requirements, defined as Competency Questions (CQs). The process is called CQ-Driven Ontology DEsign Process (CODEP) and it includes activities that validate and verify the incremental design of an ontology through metrics based on defined CQs. CODEP has also been applied in the design and development of an ontology in the context of a Mexican Hospital for supporting Neurologist specialists. The specialists were involved, during the application of CODEP, in collecting quality measurements and validating the ontology increments. This application can demonstrate the feasibility of CODEP to deliver ontologies with similar requirements in other contexts.

Keywords: Ontology iterative design process, competency questions, verification & validation metrics, quality indicators, ontology evolution

Accepted by: Megan Katsumi

1. Introduction

Ontologies have been used to support the representation and management of data in several domains. For example, FIRO (Espinoza, Abi-Lahoud, & Butler, 2014) is an ontology used for reasoning over data in the financial domain to support anti-money laundering. Also, ontologies exist in the medical domain such as OpenGalen (OpenGalen Foundation, 2012) or SNOMED-CT (SNOMED International,

*Corresponding author. E-mail: angelina.espinoza.limon@gmail.com.

2015); OpenGalen is expressive enough and SNOMED-CT represents taxonomies, which standardizes medical concepts. One of the most powerful applications for ontologies is to build a knowledge base that can be populated and queried. For instance, in medicine, a knowledge base can be used to support the medical diagnostic record identification and medical dissections, and surgical procedures (Napel, Rogers, & Zanstra, 1999). Also, an ontology-based knowledge base can be queried by users as well as information systems, which can be used for inferencing or most commonly known as reasoning over data. This feature makes ontologies a powerful means to support intelligence and automation in information systems, which is often called a marriage (Pisanelli, Gangemi, & Steve, 2003). Ontologies that are application-oriented need to answer queries representing functional requirements and they also need to satisfy several quality attributes (e.g., accuracy, efficiency, availability, etc.). This means that these ontologies should have a balance between expressiveness (the type of axioms it implements, such as inheritance, symmetry, or functional relationships among others) and the ability to answer the queries according to end-user requirements. To build application-oriented ontologies is the main motivation for our work and we suggest that this kind of ontologies will need a unique process that includes activities, which constantly evaluate the satisfaction of end-user requirements, as well as the verification of the ontology quality.

Creating an ontology is usually time-consuming, error-prone, and requires extensive training and experience (Pazienza & Armando, 2012). In addition, creating a very expressive ontology cannot guarantee its implementation in a knowledge base, to be effectively queried. End users and/or other software systems need to perform queries to support practical tasks. In this support stems the importance of continuously validating and verifying the ontology design, for evaluating whether the ontology is truly delivering the required support. This checking should not be postponed after the completion of the ontology.

In the process of designing an ontology, end-user requirements are captured through Competency Questions (CQs; Grüninger & Fox, 1995, p. 3). CQs are natural language questions that need to be answered by querying an ontology, for solving practical tasks of end-users. Several methodologies and processes do not use CQs to drive the ontology design and therefore, knowledge bases do not support CQs. Those that do use CQs barely use the results of the CQs' responses (by querying), for driving improvements in the ontology design. Some examples are NeOn by Suárez-Figueroa M. C. (2010), which uses the Ontology Requirements Specification Document (including a list of CQs) to guide the development, and METHODOLOGY that defines CQs in the Specification Phase (Fernández, Gómez-Pérez, & Juristo, 1997). If CQs are not used to drive the design process, then it is difficult to validate whether an ontology truly supports the end-user needs. Therefore, in this paper, we propose an approach that considers CQs as drivers for an ontology design, through translating them into queries to be executed in an implemented Knowledge Base (KB), and by including the CQs in the Validation & Verification activities (through the metrics definition and application).

Also, we have found that there are many knowledge engineering processes and methodologies for creating ontologies (as reviewed in the Section Related Work), but the Validation and Verification (V&V) activities that ensure the satisfaction of users and their requirements, are barely included as a backbone for driving the improvements in ontology iterations. In addition, none of the existing approaches propose well-structured processes to be followed, with clear activities and roles. In this paper, we tackle these gaps by proposing an approach called CQ-Driven Ontology DEsign Process (CODEP), an iterative and incremental process for designing ontologies and developing knowledge bases that truly satisfy end-user needs, defined as CQs. The contributions of CODEP are as follows: 1) It defines a well-structured process to be followed, with clear activities and roles; 2) it includes Validation & Verification (V&V) activities

based on quantitative metrics. These metrics are helpful because they (a) support KEs in improving the ontology (b) allow users to quantitatively indicate their satisfaction towards an increment of an ontology and its KB from different perspectives; 3) it supports an incremental and iterative life cycle where ontology versions are produced until users are satisfied and an applied KB is produced. The process produces an ontology that is validated against the expected end-users quality metrics, which ensure effective support for practical purposes, such as the ontology implementation in a KB which can also be mined by information systems.

Additionally, this paper presents how CODEP has been applied in every activity. The application has been performed in collaboration with a medical team at a Mexican Hospital, to create a medical ontology that supports the identification of patients with diagnostic features after suffering traumatic head injuries, to enroll them in a rehabilitation program. Through the application of CODEP, it is shown, how the ontology and its respective implemented KB responds to the CQs, and how several metrics are used to validate the satisfaction of the medical team's requirements. This validation is used to improve the ontology iteratively and incrementally by obtaining feedback from the medical team. The medical team evaluates the accuracy and comprehensibility of queries' responses and knowledge rules, the coverage and completeness of the CQs, and the responses' comprehensibility.

The paper is organized as follows: Section 2 presents an overview of CODEP. Section 3 illustrates in detail how CODEP has been applied to create a medical ontology and develop a knowledge base. Section 4 presents the protocol conducted to validate that the medical ontology satisfies the CQs. Section 5 analyses related work, and finally Section 6 presents the conclusions.

2. The CQ-driven ontology design process

This section presents the CQ-Driven Ontology DEsign Process (CODEP), which is driven by end-users CQs. CODEP has been inspired by taking proved practices from ontology designing experiences in three previous projects (Espinoza, Abi-Lahoud, & Butler, 2014; Nieves, Espinoza, Penya, Ortega De Mues, & Peña, 2013; Espinoza et al., 2013). This process is defined for creating ontologies that will be implemented in Knowledge Bases (KBs) that can be mined. Therefore, this is an application-oriented ontology design process.

CODEP is divided into three main phases: *Phase I CQ Domain Acquisition, Phase II Ontology Building, and Phase III Ontology Verification and Validation*. Fig. 1 summarizes the life cycle. Table 1 shows the activities for each phase with their respective milestones, which produce practical outcomes: *Milestone I: Ontology Vision and Scope, Milestone II: Ontology Beta*, and *Milestone III: Ontology and Knowledge Base Release*.

CODEP is a process that stems from V&V activities, for obtaining feedback from *Subject Matter Experts* (SME) to drive the ontology design process to produce an ontology that is aligned with end-users needs. That is, to validate whether the CQs (stated by the end-users and that are answered with the ontology) are satisfactorily responding to the user's expectations. From the early phase of CQ elicitation, the end-users define their validation criteria. Thus, in the Ontology Building phase, the ontology design is driven by the end-users requirements. The actors involved in CODEP are: 1) the *SME*, which has the expertise in the domain knowledge, e.g., a physician specialist in medicine; an SME can also be one of the end-users of an ontology (e.g., a general physician); and 2) The *Knowledge Engineer* (*KE*) who is responsible for modeling and designing the ontology.

CODEP defines a life cycle model based on the Incremental-Build Model (IEEE-SWEBOK, 2014), which includes from modeling to V&V. It can be observed from Fig. 1 that CODEP defines an **Iteration**

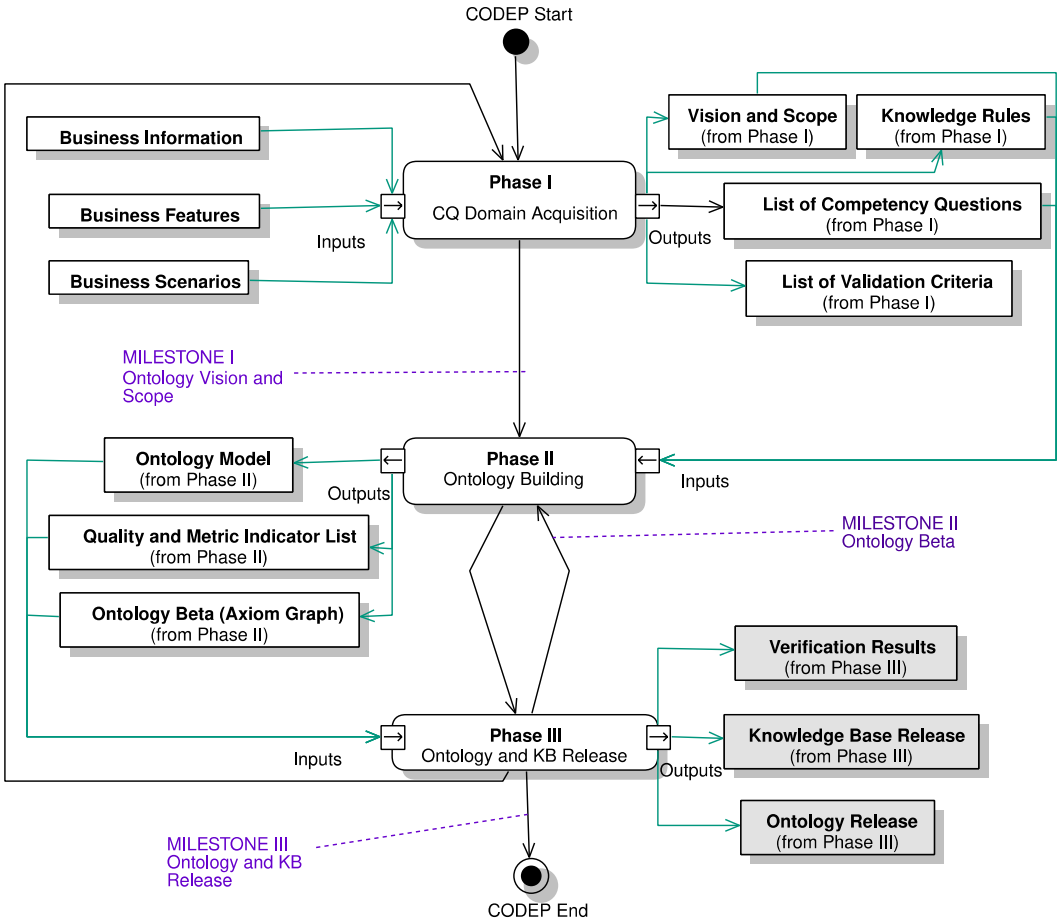


Fig. 1. Life cycle of CODEP.

from Phase I to Phase III (Activities 1–16 in Table 1), and each iteration produces an **increment** (version) of the ontology and KB until a version is released when the validation indicates that the ontology truly meets the end-user criteria. In this sense, an increment might require several small cycles between Phase II and Phase III, to verify the ontology design and validate the ontology (through the KB), according to the quality criteria and indicators. It is in the last iteration that the ontology is released when the desired quality is completely reached out. Also, from Fig. 1 it can be observed that a subset of the CQs needs to be stable and complete to perform Phase I till Phase III. However, if the CQ set needs to be modified, CODEP requires to finish the ongoing iteration (from Activities 1–16 in Table 1), before starting over again a new iteration (start of Phase I) and then editing the CQs for obtaining a new release of the ontology and KB at the end of CODEP (end of Phase III).

3. Specifying and applying CODEP

In the following CODEP specification, the activities description is done along with the case study application, to facilitate the CODEP understanding. The activities' description is stated for a gen-

Table 1
CODEP activities

Phase	Activity
Phase I	1. Problem Identification and Scope (PIS)
	2. Application Domain Analysis (ADA)
CQ Domain Acquisition	3. Competency Question Elicitation and Definition (CQD)
	4. Knowledge Rules Definition (KRD)
	5. Validation Criteria Definition (VCD)
Milestone I: Ontology Vision and Scope	
Phase II	6. Initial Concept Identification (ICI)
	7. Reusable Ontology Investigation (ROI)
Ontology Building	8. Conceptual Synthesis Realization (CSR)
	9. Ontology Design Quality Indicators Definition (OQID)
	10. Ontology Modelling and Design (OMD)
	11. Ontology Axiom Definition (OAD)
	12. Ontology Model Validation (OMV)
	13. Knowledge Rule Implementation (KRI)
Milestone II: Ontology Beta	
Phase III	14. Ontology Verification (OV)
Ontology Verification and Validation	15. Knowledge Base Implementation (KBI)
	16. Ontology Validation (OVA)
Milestone III: Ontology and Knowledge Base (KB) Release.	

eral application, while the case study describes how the process can be performed in a practical setting.

3.1. Case study description

The ontology supports a medical team composed of neurologists, who are specialized in rehabilitating patients with head injuries, in the context of the National Rehabilitation Institute (Instituto Nacional de Rehabilitación – INR, http://www.inr.gob.mx/r08.html), a Mexican hospital and leader in attending patients in rehabilitation services. INR also promotes research projects in several medical areas about rehabilitation. The medical team conduct one of the INR's research protocols called the Cerebrolysin Research Protocol (CRP; INR, 2013). CRP focuses on investigating the functional, cognitive, psychological, and physical effects of the Cerebrolysin® drug, as adjuvant treatment for several sequelae of Traumatic Brain Injury (TBI). The CRP team is constantly seeking patients with specific diagnosis conditions, who meet the CRP's requirements and analyses whether they are candidates for neurological rehabilitation in an in-hospital program that lasts for a year. The medical team, who is specialized in neurology at INR and manages the CRP, needs to perform a specialized evaluation of the patients, who are sent from other hospitals with a preliminary diagnosis of TBI. However, these hospitals are the first-contact medical place after the patient has suffered an accident, and commonly the general physicians in charge are not specialized in making such specific diagnoses for the criteria evaluation. Currently, two situations can happen: 1) INR's medical staff need to travel to the first-contact hospitals to perform the diagnostic analysis and considering the high number of hospitals with A&E services in Mexico City and the time it takes to get there, this becomes an unpractical activity; 2) The first-contact hospitals' general physicians perform a preliminary diagnosis to send potential patients to the INR for the Cerebrolysin treatment. However, as they are not specialists in medical rehabilitation, such preliminary diagnostic,

Table 2
The Inclusion/Exclusion Criteria, as Business Policies (BP) and Business Constraints (BC)

ID	Inclusion Criterion
IC-1 (BP)	Any patient gender (male, female)
IC-2 (BP)	Patients older than 18 years old (inclusive)
IC-3 (BC)	Patients with a sequela diagnosis caused by the TBI
IC-4 (BC)	TBI is severe when the lesion time is between 1 to 6 months (inclusive).
IC-5 (BC)	Patients whose physical/cognitive condition are allocated from the moderate to severe disability, according to the Glasgow Outcome Score (GOS; Jennett & Bond, 1975).
IC-6 (BC)	Patients whose physical/cognitive condition are allocated between the categories III-VI, according to the (Rancho Los Amigos Level of Cognitive Functioning Scale) LCFS scale (Hagen, Malkmus, & Durham, 1972).
IC-7 (BP)	Patients who voluntarily accept to participate in the CRP, or either have a responsible relative who signs the agreement to participate.
	Exclusion Criterion
EC-1 (BP)	Patients younger than 18 years old.
EC-2 (BC)	Patients whose disability cause is not clearly related to the TBI.
EC-3 (BC)	Patients who have another pathological state different or previous to the TBI, which might affect the rehabilitation process, or the functional performance evaluation (e.g. mental disability, amputation, and a previous TBI).
EC-4 (BC)	Patients who have a severe TBI (<1 week) or chronic (>6 months).
EC-5 (BC)	Patients who have a convulsive crisis or with epileptic activity in the initial electroencephalogram (EEG).
EC-6 (BP)	Patients who do not fulfill the basic INR's requirements to be in an intra-hospital rehabilitation program.
EC-7 (BC)	Patients with such agitation that cannot withstand the intravenous treatment for more than one hour.
EC-8 (BC)	Patients who previously have been treated with Cerebrolysin® or who are under another treatment to stimulate their rehabilitation.

could be incomplete or wrong. Both situations cause the loss of candidate patients to be enrolled in the CRP, which affects the medical research and the rehabilitation of current and future patients.

Thus, in the Case Study, it is important to analyze whether the patient fulfills a set of criteria, which is followed by the INR's medical team, to determine if the patient can be treated under the CRP (INR, 2013). Table 2 shows these criteria as a list of Inclusion Criteria and a set of Exclusion Criteria. The application of the inclusion/exclusion criteria follows an order which is determined by the INR's medical team. In this Case Study, some of these criteria might be *business policies* (e.g., that the patient must be at least 18 years old), others might be *business constraints* (e.g., that the TBI is considered "severe" if the lesion time is between 1–6 months. Thus, the CRP (including the inclusion/exclusion criteria) along with the INR organization, will be taken as *Business Information*, which is used to identify the Business Policies (BP) and Business Constraints (BC; Fig. 2, Activity 2).

In this scenario, the ontology will be the means to organize and model the medical rationale required to process patients to be treated under the CRP. The KB (based on the ontology) will support the first-contact GPs at the hospital to perform the specialized diagnostic analysis, through the implementation of the medical criteria as knowledge rules. It is worth mentioning that before starting the ontology design, research was performed to identify processes that are oriented to developing application-oriented ontologies. As commented in Sections Introduction and Related Work, even there are several well-known processes, we were not able to find one which focused on developing ontologies to be used in an application setting, with frequent activities to perform quantitative validation (for measuring if the ontology

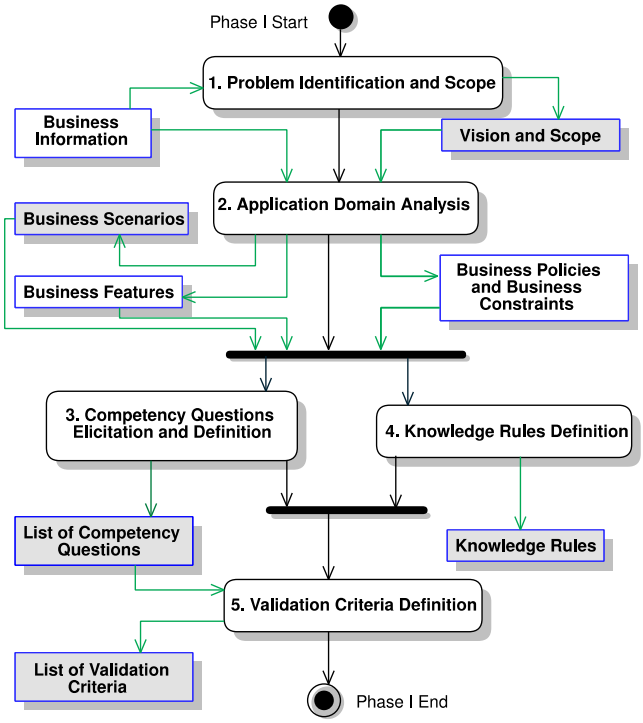


Fig. 2. Phase I flow.

and KB are truly answering the CQs, from the SMEs' perspective), with a guide to applying the process, and that the resulting ontology can be used to set up a knowledge base, able to execute queries from a software application. Therefore, CODEP is inspired by many of these processes but fills the gap by proposing a new process that is structured, includes V&V, and can be driven by the CQs to create an application-oriented KB.

In the following Section 3.2, the description and application of CODEP in every phase are explained.

3.2. CODEP phases and activities

Phase I. ***CQ Domain Acquisition***

The objective of Phase I is to understand the domain to obtain the list of Competency Questions and Knowledge Rules that will be used to drive the ontology (see Fig. 2). **The activities for this phase are as follows**:

Activity 1. ***Problem Identification and Scope (PIS)***

The problem identification of an ontology is key for avoiding the modeling of irrelevant aspects of the domain application, and the scope states the expectations that the ontology must be compliant. The problem identification and scope are defined by conducting several meetings between the knowledge engineer and the SME to identify what are the needs for the ontology. The SMEs are the experts of the domain and they can be end-users of the ontology. From this step, the KE identifies the actors, end-users, and stakeholders of the ontology. The Input is the *Business Information* related to the domain and context

Table 3
Scenarios for ontology usage

Ontology Usage Scenarios	Involved Actors
S1 – The GP and the SP use the ontology for conducting a CRP inclusion/exclusion criteria evaluation of a patient.	GP – General Physician SP – Specialist Physician in Neurology
S2 – The SP uses the ontology to check which of the inclusion/exclusion criteria are not fulfilled, to refuse a patient as a candidate to the CRP.	SP – Specialist Physician in Neurology
S3 – The SP uses the ontology to conclude which patients are candidates for the CRP.	SP – Specialist Physician in Neurology
S4 – The SP uses the ontology to consult current candidates enrolled in the CRP.	SP – Specialist Physician in Neurology

that describes problem elements. The Output is a *Vision and Scope* that clearly describes the subject, the problem boundaries, and the roles involved.

Case Study: Vision and Scope contains the ontology scope, which is "to support the INR's medical staff in the analysis of patients who have just suffered an accident causing a TBI and who are being attended at first-contact hospitals". The aim is to: 1) assist the first-contact hospital's general physician during the patient evaluation to obtain a close diagnostic and to determine the CRP candidates; and 2) support the INR's neurology researcher to identify candidate patients of the CRP protocol without physically attending the first-contact hospitals. The involved actor is the neurology specialist (SME). After stating the ontology scope, the target ontology was named CErebrolysin Research PRotocol Ontology (CERPRO).

Activity 2. ***Application Domain Analysis* (*ADA*)**

The objective is that the KE gets deep knowledge about the concepts that need to be included in the ontology and the level of expressiveness. Specifically, 1) the application domain context (the institution features, the company's business policies and constraints from relevant standards, and a glossary of terms) is studied; the doubts should be solved with the involved people in the domain (staff, customers, or technicians); 2) the set of scenarios (including the tasks that are performed by the scenario's actors) where the ontology is applied are identified; here each actor can have a set of scenarios or many actors can share them. The Input is the *Vision and Scope*, from Activity 1. The Output is three elements: 1) the *Business Features* that describes the institution description and a list of significant domain terms; 2) the *Business Scenarios*, which includes the scenario set per role; and 3) the *Business Policies and Constraints*.

Case Study. In this activity, the KE obtains knowledge from the Cerebrolysin Research Protocol (CRP; INR, 2013) and from the Business Information to get the *Business Features*: medical terms, specific diseases and pathological conditions that a patient could present, the supplied drugs, specific tests, accident types, and cognitive/physical disabilities and the scales to measure them. The KE obtained the *Business Policies and Constraints*, from the INR documentation and the inclusion/exclusion criteria from the CRP. The *Business Scenarios* were obtained through several meetings with the Chief Medical Doctor of the CRP team to identify the ontology end-users (actors) and usages (scenarios), based on the *Ontology Vision and Scope*, from Activity 1. The meetings were structured in a question-answer format, where the questions were sent to the medical team before the meetings. After the meetings, specific inquiries were further addressed via email. As a result, the set of scenarios and actors are summarized in Table 3. These scenarios will be helpful to state the end-user requirements (CQs) in further activities.

Table 4
(Excerpt) Competency Questions (CQs)

ID-CQ	Competency Questions	Dependency
CQ-1	Has the TBI (Traumatic Brain Injury) occurred as a result of an accident (car accident, downfall, ballistic accident, physical aggression)?	No dependency
CQ-5	Has the patient been evaluated under the GOS? If yes, does she/he have a disability level from moderate to severe?	CQ-1
CQ-11	Does the patient have any pathology previous to the current TBI that affects the rehabilitation process, or the performance obtained during the functional evaluations (specifically, previous TBI, mental retardation, convulsive disorder, cognitive impairment, peripheral neuropathy, addictions, or amputations)?	CQ-1, CQ-10

Activity 3. ***Competency Question Elicitation and Definition* (*CQD*)**

This activity focuses on capturing the set of CQs. The CQs are the questions that SMEs need to answer to support their daily work. They are usually the queries that need to be supported with the ontology, and they are defined as focusing on obtaining knowledge (explicitly or implicitly) from the KB. Usually, CQs are in the mind (tacit) of SMEs, and the KE works with SMEs to make them explicit. To be able to explicitly define the CQs, the KE analyses as Inputs: The *Business Features*, Business *Scenarios*, and the *Business Policies and Constraints* from Activity 2. The Output of this Activity is the *List of Competency Questions* (this includes the refined CQs in iterations).

Case Study. Applying this activity results in the specification of CQs as the medical team needs to identify patients with specific diagnosis conditions, after analyzing the *Business Features* and *Business Scenarios* from Activity 2. The CQs have been defined to answer whether a patient satisfies the *Business Policies and Constraints* (the inclusion/exclusion criteria from the CRP). Thus, the SP will use the CQs responses to support the evaluation of the medical condition of the patients, who were initially diagnosed with TBI in the first-contact hospital by the GP. Table 4 shows an excerpt of the 14 CQs (*List of Competency Questions*) dictated by the SME, indicating dependencies among them. The Dependency column indicates which CQ/CQs must be modelled in the ontology, before the CQ indicated in the ID-CQ column; this is simply the sequential order for modeling the CQs. For example, CQ-1 must be implemented before CQ-11, because this latter question asks for previous pathologies to the "current" TBI. This implies firstly implement the CQ to find out if a TBI has occurred as an accident result, in another way, CQ-11 does not make any sense outside this context.

Activity 4. ***Knowledge Rule Definition* (*KRD*)**

This activity aims to identify constraints or restrictions in the knowledge domain, such as regulations, the vision, and scope document (from Activity 1), and relevant domain documentation provided by the SMEs. Constraints are defined as *Knowledge Rules* to implement the business logic in the ontology domain, and they are defined as axioms in the ontology, which can be verified by a reasoner (an intelligent software application). In CODEP, the *Knowledge Rules* are extracted from the Input: *Business Scenarios*, *Business Features*, and *Business Policies and Constraints* (from Activity 2); but it is recommended that the KE refines them along with the SME. The Output is the *Knowledge Rules*, which are implemented in the KB in Activity 12, to apply the ontology quality metrics (e.g., for testing accuracy of the CQs responses). The knowledge rules identification can be done by defining use cases, identifying the flows in such use cases, and complementing them with activity or business process modeling diagrams (e.g., with UML or Business Process Modeling Notation).

Case Study. The *Knowledge Rules* for the case study are extracted from the *Business Policies and Constraints*, for the identification of the patient diagnostic to be enrolled in the CRP. There are several strict conditions in the CRP (Inclusion/Exclusion Criteria, IC/EC), which prevent a patient from being considered as a candidate; such conditions are the Business Policies (BPs) and constraints (BCs) from Table 2, which drives the application of the Cerebrolysin treatment to patients. Thereafter, the neurologist applies the IC/EC criteria following a specific order, as a checklist to the patients for identifying candidates for the CRP. Thus, the *Knowledge Rules* are based on the IC/EC, e.g., the knowledge rule **KR-3** describes that the patient diagnosis must present sequelae originating from the TBI (IC-3), but with no previous pathologies to the current TBI (EC-3). The medical reason is that the rehabilitation process could be affected by the Cerebrolysin drug or the functional performance evaluation. Thereafter, **KR-3** states to verify these two criteria simultaneously, as the GP does in the first-contact hospital; and if one of these two criteria fails, then the patient is not included in the CRP; otherwise, the knowledge rule application carries on to the next knowledge rule, which is **KR-4**.

Activity 5. ***Validation Criteria Definition* (*VCD*)**

Before the creation of the ontology, we recommend defining the *ontology validation criteria*. The validation criteria aim is to check whether the CQs respond to the SMEs' expectations, once an ontology is set up in a knowledge base. For this purpose, the KE will get the CQs' responses through querying the KB, and the SME will indicate if they believe whether the validation criteria are met or not after evaluating such CQs' responses. Thus, the KE and SMEs work together to set these criteria, and they should reflect the functional achievement (e.g., completeness and accuracy of the CQs) but also the non-functional one (e.g., comprehensibility and efficiency). During the validation criteria definition, the communication between the KE and the SME must be very active. Depending on the ontology objective and the SME expectations, different validation criteria can be defined as a set of metrics. In the literature, there are already metrics that can be reused for quality assessment, such as the external quality and quality in use from the SQuaRE model in the series ISO/IEC 2502n (ISO-25023, 2016; ISO-25022, 2016). However, the KE must adapt them to reflect that the requirements specification for CODEP is represented as CQs. The input for this activity is the *List of Competency Questions* (from Activity 3); and the Output is the *List of Validation Criteria*, which will be used for the ontology validation (Activity 15).

Case Study: The INR's medical team is the target audience and the SME is the neurologist specialist, who will perform the validation. The *List of Competency Questions* (from Activity 3) is used to create the validation criteria for CERPRO, as follows: 1) the KEs define a set of possible validation metrics which are presented to the INR's medical team, 2) the medical team selected the relevant metrics based on their needs to obtain automatic support for the simultaneous identification of patient's diagnosis from several medical datasets, and 3) the SMEs were asked to set expected values for each metric, based on their priority of satisfaction (Expected Medical Team's value). To make this communication fluent, emails, recordings, and minutes from one-hour meetings were used, which were held for each version revision.

The *List of Validation Criteria* for CERPRO included: *CQ completeness* (since the medical team is interested in implementing the whole CQs set), *CQ response accuracy* (since the medical team expects accuracy in the responses when identifying diagnostics), and *CQ response comprehensibility* (since the medical team needs to comprehend the responses from the knowledge base with no complication in the technical argot). Each metric has a definition, a metric to collect values to prove whether the ontology

expectations are fulfilled, the range of meaningful values (e.g., 0 the worst value and 1 the best value), the number of CQs needed to calculate the metric and the medical teams' expected value.

- **CQs Response Accuracy**. This refers to the accuracy level of the answers for the CQs. It is determined according to the medical team's expectations.

$$\boldsymbol{aCQ} = \left(\sum_{i=1}^{n} ACQ_i\right)/n; \quad 0 \leqslant aCQs \leqslant 1, 0 \leqslant ACQ_i \leqslant 1, 0 < n \leqslant TCQs \tag{1}$$

aCQ: Zero the worst accuracy, one the best. If *aCQs* is near zero then the medical team's satisfaction is under reasonable expectations, on the contrary, if it is near 1, the satisfaction is high.
i: Number of the CQ being measured.
n: Number of the evaluated CQs (14 for this case study) in a given iteration.
ACQ_i = Accuracy measurement for each CQ_i, from 0 to 1, being 1 the better.
TCQs: The total number of CQs, which were provided by the medical team.

Expected Medical Team's value: 100% that is normalized to 1 for this case study, since the medical team expects a high accuracy.

- **CQs Completeness**. This refers to the number of CQs provided by the doctor, which according to the medical team's perspective, are answered with CERPRO.

$$\boldsymbol{cCQ} = \left(\sum_{i=1}^{n} ICQ_i\right)/TCQs, \quad 0 \leqslant cCQ \leqslant 1; TCQs > 0; ICQ_i \text{ is 0 or 1} \tag{2}$$

cCQ: If *cCQ* is near zero then the CQ implementation is poor, on the contrary, if it is near 1, it is the best.
i: Number of the CQ being measured.
n: Number of the evaluated CQs (14 for this case study in the last iteration) in a given iteration.
ICQ_i: The score for the CQ answer (provided by the ontology/KB). This is given by the end-user (0 or 1).
TCQs: The total number of CQs, which were provided by the medical team.

Expected Medical Team's value: 100% that is normalized to 1 for this case study, since the medical team expects getting responses from the KB for the whole set of CQs that they defined in conjunction with the KE.

- **CQs Response Comprehensibility**. This refers to the comprehensibility level according to the doctor's perspective of the concepts and relationships founded in the CQs answers.

$$\boldsymbol{rCQ} = \left(\sum_{i=1}^{n} RCQ_i\right)/n, \quad 0 \leqslant rCQ \leqslant 1; n > 0, RCQ_i \geqslant 0 \tag{3}$$

rCQ: If *rCQ* is near zero then the CQs' responses' comprehensibility is poor, if it is near to 1 it is high.
i: Number of the CQ being measured.

n: Number of the evaluated CQs (14 for this case study in the last iteration) in a given iteration.
RCQ_i: Measures how comprehensible is the CQ's response through the query to the doctors.

Expected Medical Team's value: 90% that is normalized to 1 for this case study, since the medical team can accept a moderate technical argot in the provided responses from the KB.

Thus, in Activity 15, these metrics are applied to obtain feedback about CERPRO's quality from the SME's perspective, to identify deviations and to correct them, till ***aCQs***, ***cCQ*** and ***rCQ*** become nearly to the end-users expected values.

Milestone I: The Ontology *Vision and Scope* are obtained.

Phase II. ***Ontology Building***

The objective of this phase is to build an ontology model and its axioms (see Fig. 3). The activities for this phase are as follows:

Activity 6. ***Initial Concept Identification* (*ICI*)**

This activity aims to extract the relevant domain nouns and business actions from the outputs of previous activities. The Input to this activity is the *Business Scenarios, the Vision and Scope* document, the list of *Competency Questions*, and the *Knowledge Rules*. Nouns in these inputs will be the initial list of the ontology concepts and the verbs will be relationships among the concepts. This is a highly creative task and can involve interviewing further the SME and additional information could be provided to the KE related to the identified concepts. The Output of this activity is the *Initial Concept List*.

Case Study: The following are used: the CQs, the inclusion/exclusion criteria (which are the *Knowledge Rules*), the scenarios' list, and *Vision and Scope* to identify the relevant concepts. A table was created for the initial domain concepts. After obtaining initial concepts, new concepts were introduced which are not present in the inclusion/exclusion criteria. For example, consider the criterion IC-3 (from Table 2):

"*Patients with a sequela diagnostic caused by the TBI*."

From this statement, the following nouns can be identified as concepts: Patient, Diagnostic, Sequela, and TBI. Then, more information can be discovered about the features which describe *Patient*, *Diagnostic*, and *TBI* through interviewing physicians. For the *Sequela* concept, the SP has further required to detail the classification of a series of illnesses produced as a consequence of the TBI. Relevant concepts were also identified for the application domain analysis of the target ontology. For example, the *Glasgow Outcome Score* (*SNOMED International, 2015*) which is a scale for measuring the damage caused by brain injuries (e.g., cerebral traumas), divides patients into 5 groups: Death, Persistent vegetative state, Severe, Moderate and Low disability. Thus, the *Initial Concept List* is produced to be included in the ontology.

Activity 7. ***Reusable Ontology Investigation* (*ROI*)**

As a result, new relationships and concepts not identified in Activity 6 can be discovered and added to the *Initial Conceptual List*. In this activity, the KE can identify several potential third-party ontologies. In this case, the KE needs to evaluate which one is more appropriate based on the initial concepts identified and the scope. The KE maps the third-party ontologies' concepts with the initial concepts. The SME needs to work with the KE to ensure that the semantics of the concepts. Therefore, the Input is the *Initial Concept List* and the Output is the reusable *Third-party Ontology/ies*, which are selected.

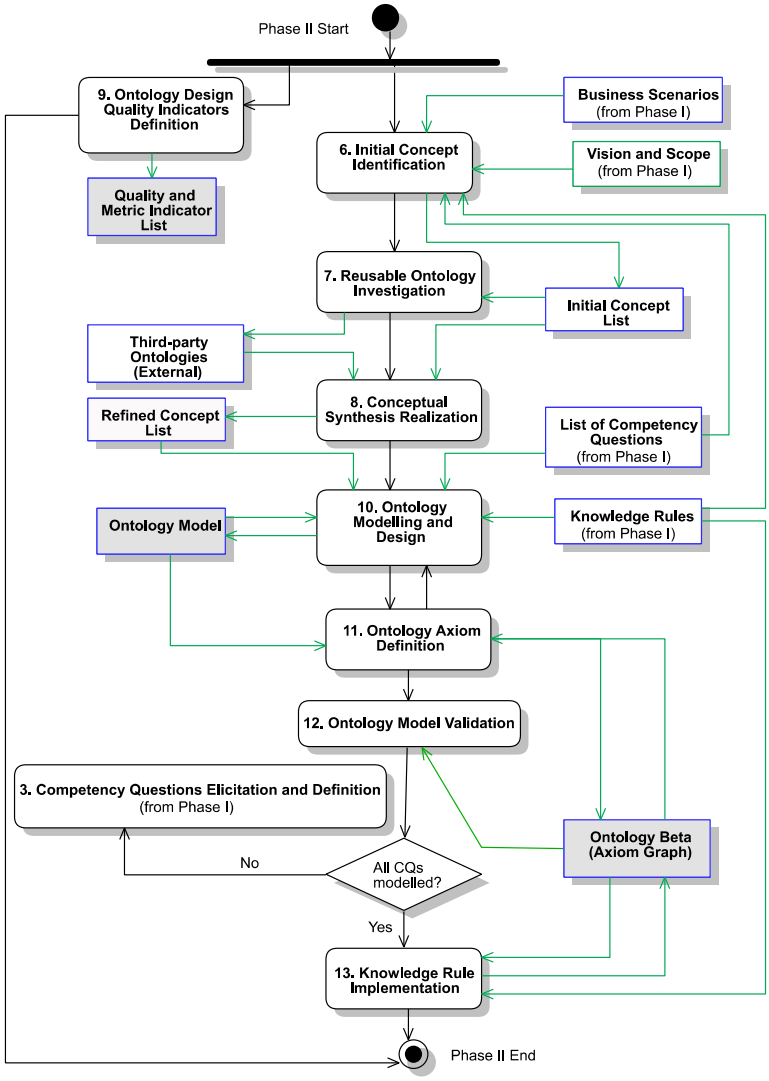


Fig. 3. Phase II flow.

Case Study: An exhaustive search in medical relevant ontologies was performed, by using searching the initial concepts from Activity 6 about the CRP and the required medical concepts for the CQs. As a result, OpenGalen (OpenGalen Foundation, 2012) was identified which provides both: 1) an acceptable taxonomy of medical concepts and procedures and 2) a proper expressivity to support the CQs. Another relevant ontology identified is SNOMED-CT (SNOMED International, 2015). However, the 14 CQs' goals are not met with SNOMED-CT and its expressivity is poor in terms of the OWL-DL potential. Therefore, several OpenGalen's modules were used as a basis to incorporate medical terminology already agreed by an extensive medical research team. In the aim of finding the reusable OpenGalen modules, the semantics of the concepts of the Initial Concept List were matched to the terminology available in OpenGalen. The neurologist team at INR was involved in ensuring the semantics of the medical terminology are valid.

Table 5
(Excerpt) refined concept list

ID	Concept	Concept in CERPRO	Concept in OpenGalen
IC-3	Patient	Class: Patient	Subclass of: Human
	Diagnostic	Class: Diagnostic	Not included
	TBI		Equivalent Class: HeadTrauma
	Sequela	Class: Sequela	Superclass of: • Class: HemiplegicParalysisProcess • Class: AttentionDeficitDesorder • Class: HypotoniaOfMuscle • Class: Memory • Class: Spasm
IC-5	Condition	Class: Condition Subconcept: • Class: Physical • Class: Cognitive	Not included
	Scale	Class: Scale Subconcept: • GlasgowComaScale	Not included
	Disability	Class: Disability Datatype: String (Moderate, Severe)	Not included

Activity 8. ***Conceptual Synthesis Realization (CSR)***

This activity aims to synthesize the concepts obtained from the *Third-Party Ontologies* (Activity 7), with the concepts that have been identified in Activity 6 to eliminate duplicate concepts from the re-used modules. Therefore, the Input of this activity is the *Initial Concept List* and the reusable *Third-Party Ontologies*. This activity must be carefully done since third-party ontologies could be exhaustive, and it is necessary to double-check if each of the concepts from the CQs is already included in the Third-Party ontologies selected. We, therefore, recommend mapping the concepts of the reusable ontology to the ones in *Initial Concept List*. The output is a *Refined Concept List* which will be used to model the ontology in Activity 10.

Case Study: Table 5 shows an excerpt of the resulting conceptual synthesis, including some potential modeling actions for creating the concepts in the target ontology. For example, for IC-3, *Patient, Diagnostic and Sequela* concepts can be modeled as classes in CERPRO, where *Patient* inherits from the class *Human*, which is already defined in OpenGalen. This axiom connects CERPRO with OpenGalen since *Patient* must be defined in CERPRO and it is related to *Human* (by inheritance) from OpenGalen. Similarly, *TBI* is already included in OpenGalen through the concept HeadTrauma. This analysis strategy was followed for each concept as in Table 5 (*Refined Concept List* for CERPRO).

Activity 9. ***Ontology Design Quality Indicators Definition (OQID)***

This activity aims to allow the KE to set quality indicators about the ontology design to be verified in later activities. The KE sets quality indicators related to the design of the ontology. In this way, the design of the ontology is driven by these quality indicators and checked after the modeling stage. Quality indicators are set to assure that the ontology follows quality standards. It allows the KE to verify that the ontology has been designed and implemented according to the quality indicators and if not, the KE can refine the ontology. Quality indicators are different from validation metrics defined in Activity 5.

Validation metrics are end-user-oriented and are checked with users to achieve end-user expectations. Quality Indicator Metrics can be adopted from already defined ones available in the literature such as by Gangemi, Catenacci, Ciaramita, & Lehmann (2005) and by Tartir, Budak Arpinar, Moore, Sheth, & Aleman-Meza (2005). However, these could need to be adapted to be defined in terms of CQs. The Output of this activity is the *Quality and Metric Indicator List*, including the metrics per indicator, to be used for measuring the quality of the ontology.

Case Study: Several quality indicators were defined such as *Modelling Completeness*, *Semantic Consistency between Models*, and *Model Expressiveness*:

- **Model Completeness**: The aim is to assure that all CQs' concepts are considered in the ontology. This indicator measures how many concepts are identified from the CQs and how many of them are modeled in the ontology. A CQ concept could have two or more concepts modeled in CERPRO. For this reason, this metric might be greater than 1.

$$\boldsymbol{comCQ} = \left(\sum_{i=1}^{n} \frac{MC_i}{CQC_i}\right)/n, \quad 0 < CQC_i \leqslant MC_i \tag{4}$$

comCQ: The average of total concepts in the ontology from the n evaluated CQs, <1 the worst, $\geqslant 1$ the best.
i: Number of the CQ being measured.
n: Total number of CQs.
MC_i = Number of modelled concepts in the ontology from the CQ_i.
CQC_i: Number of concepts that are identified from the CQ_i.

- **Semantic Consistency between Models**. The aim is to verify whether the same axiom (concept, data/object property) does not have multiple meanings in the ontologies which were merged from different sources. Specifically, in CERPRO, this verifies if a concept or data/object property identified in the CQs, does not also exist in OpenGalen (the third-party ontology). If not, the concept is modelled in CERPRO.

$$\boldsymbol{SemCM} = \left(\sum_{i=1}^{n} \frac{MCOG_i + MCC_i}{CQC_i}\right)/n, \quad 0 \leqslant MCC_i, MCOG_i, \text{ and } CQC_i > 0 \tag{5}$$

SemCM: 1 the best (none of the concepts are modelled in more than one ontology),
2 the worst (concepts are modelled in both ontologies).
i: Number of the CQ being measured.
n: Total Number of CQs.
MCC_i: Number of concepts of the CQC_i that are modelled in the Ontology (CERPRO).
$MCOG_i$: Number of concepts from the CQC_i that already exists in the third-party ontology (OpenGalen).
CQC_i: Number of concepts that are identified from the CQ_i.

- **Model Expressiveness**. The aim is to ensure that all concepts are related to other concepts in the ontology, to evaluate the expressiveness for describing the domain. Thus, it is the average of the relationships among the concepts in terms of the data and object properties, and the relationships

among parent-child classes (inheritance). Some concepts might be required for modeling purposes, then the ***pR*** number is expected to be bigger than the concept number from the Final Concept List (Activity 8), or at least equal; if it is less than such number, then some concepts might be missed when the relationships were created or were badly modeled.

$$\boldsymbol{pR} = \frac{\sum_{i=1}^{n} \left(\frac{D_i + O_i + H_i}{MC_i}\right)}{n}, \quad O_i, D_i, H_i, n > 0, O_i \text{ or } D_i \text{ or } H_i \geqslant 1 \tag{6}$$

pR: 0 the worst, $\geqslant$1 the best, indicating that the concepts have at least a relationship (a data or an object property, or both). The condition: O_i or D_i or $H_i \geqslant 1$, restricts that each CQi has at least one relationship of any kind, avoiding leaving unconnected classes (unless the domain requirements specifically ask this).

i: Number of the CQ being measured.

n: Total Number of CQs.

O_i: Number of modelled object properties originated from the CQ_i.

D_i: Number of modelled data properties originated from the CQ_i.

H_i: Number of modelled inheritance relationships originated from the CQ_i.

MC_i: Number of the modelled concepts from the CQ_i description.

The metrics will be used in Activity 13 to measure the CERPRO's quality, to improve the quality design through the iterations, till the 3 metrics are near 1. For example, if ***pR*** is near 1 then it implies that a variety of possible elements (such as object, data properties, inheritance, etc.) have been strongly used to express the identified elements from the CQs (since this metric is based on the identified CQs' elements). If ***pR*** is near 0, then this would mean that the resulting ontology model is poor in terms of modeling expressiveness for the CQs' elements.

Activity 10. ***Ontology Modelling and Design*** **(*OMD*)**

As this activity is part of an iterative process, the KE selects a set of CQs that will create an increment of the ontology model. Therefore, the input of this activity is the *List of Competency Questions*, the *Knowledge Rules* and the *Refined Concept List*, and the *Ontology Model* from previous iterations. Each time an increment of the model is generated that satisfies a set of CQs, another set of CQs are selected from the *Refined Concept List* in the next iteration and the previous increment is extended to support a new set of CQs. In the first iteration, this activity includes selecting the language for the ontology modeling and design, and a tool for an ontology graphical representation. A graphical representation is advisable as it allows the KE to discuss the model with SMEs. An example of an ontology design language is OWL-DL (W3C-OWL 2, 2012), and examples of ontology modeling tools are yED (yWorks Software, 2020), which has a plug-in to model OWL-DL ontologies, and Enterprise Architect (SparxSystems, 2013), which uses UML to depict the OWL-DL axioms. Each iteration generates an increment, which is called the *Ontology Model* (Output), and the last iteration will generate the increment that satisfies all the CQs.

During the activity, concepts and their relationships are represented in a graphical model, before designing the ontology, to create a consensus among the participants about the ontology axioms, knowledge rules, and expressiveness level. Also, the graphical ontology model is a means to visually verify if all the concepts and their relationships from the *Refined Concept List* (Activity 8) are being considered in the model.

Due to the iterative nature of CODEP, Activity 10 can be iteratively be updated with Activity 11–12, for modeling, and designing the ontology, and defining the knowledge rules.

Case Study: For this Activity, we selected yED (yWorks Software, 2020) as the tool for the ontology graphical representation, since this tool has a plug-in for modeling OWL-DL ontologies. Fig. 4 and Fig. 5 show the CERPRO v5 model (as the Output of Activity 10), generated based on the *List of Competency Questions*, the *Knowledge Rules*, and the *Refined Concept List*. It can be observed that there is an asterisk in several of the classes (e.g., in Human); this notation is used to indicate that such a class belongs to OpenGalen. From iteration 2, this model was used to improve the CERPRO modeling in each iteration.

Activity 11. ***Ontology Axiom Definition* (*OAD*)**

In this activity, the axioms already specified in the model from Activity 10 are implemented. The Input for this activity is the *Ontology Model* and the *Ontology Beta* (from iteration 2, which includes the axioms in a formal ontology language, implemented through an ontology editor). The objective is to constantly verify that the designed ontology contains all the model expressiveness (inheritance, equivalence, disjoining, etc.), stated in the model. Several graphical modeling tools used in Activity 10 can automatically generate the ontology axioms; however, the KE must manually check that the generated axioms contain all the required expressiveness. This manual check must be done each time the model or ontology is edited. This expressiveness could consist of iteratively defining a set of axioms (the TBox in the ontology OWL language), the domain concepts and the properties (relationships) between them, cardinalities, equivalences, functions, and inheritance. It is possible to use some already known Ontology Design Patterns (ODP; Gangemi & Presutti, 2009).

In this step of this Activity, the ontology consistency checking must be done as many times as the reasoner reports inconsistencies in the ontology until there are no more reported errors. The Output of this activity will be an *Ontology Beta* version.

Due to the iterative nature of CODEP, Activity 11 can be iteratively be updated with Activity 10 and 12, for modeling, and designing the ontology, and defining the knowledge rules.

Case Study: We have used Protégé 5 (Protégé, 2020) which includes an editor for OWL-DL (W3C-OWL 2, 2012), the selected ontology language, and yED for the graphical modeling (since this allows creating graphical ontology models for OWL-DL ontologies). Since there are 2 different tools, each time the model or axioms are edited, special attention had to be taken to check the consistency between the graphical and axiom models. Fig. 6 shows an excerpt of the CERPRO v5 ontology (Output), in Protégé, describing the OWL axioms for the class *Patient*. The classes for *Diagnostic*, *Sequela*, the *Glasgow*, and *RLAS* scales can also be observed (the classes from the *Refined Concept List* from Activity 8); the axioms for these classes are displayed when the class is selected in Protégé. All the CERPRO ontology axioms are defined in the underlying ontology file.

Activity 12. ***Ontology Model Validation* (*OMV*)**

This activity aims to receive feedback on the ontology design as it is easier and less costly to make any changes/updates at this stage than in later activities in the process. In this activity, the KE shows the ontology model and axioms to the SME. For this purpose, it is recommended to use a graphical ontology representation to facilitate the comprehension of the SME. Thus, this is a qualitative validation – as it captures the SME's comprehension about missing or misrepresented concepts, attributes, and relationships.

After performing this activity, a decision can be made either, to perform more iterations to other activities in this phase, specifically activities 6, 10, and 11; or not. The input of this activity is the *Ontology Beta* and the output is the decision about performing more iterations (when some concepts are still missing in the ontology) or going forward to Activity 13 to implement the knowledge rules.

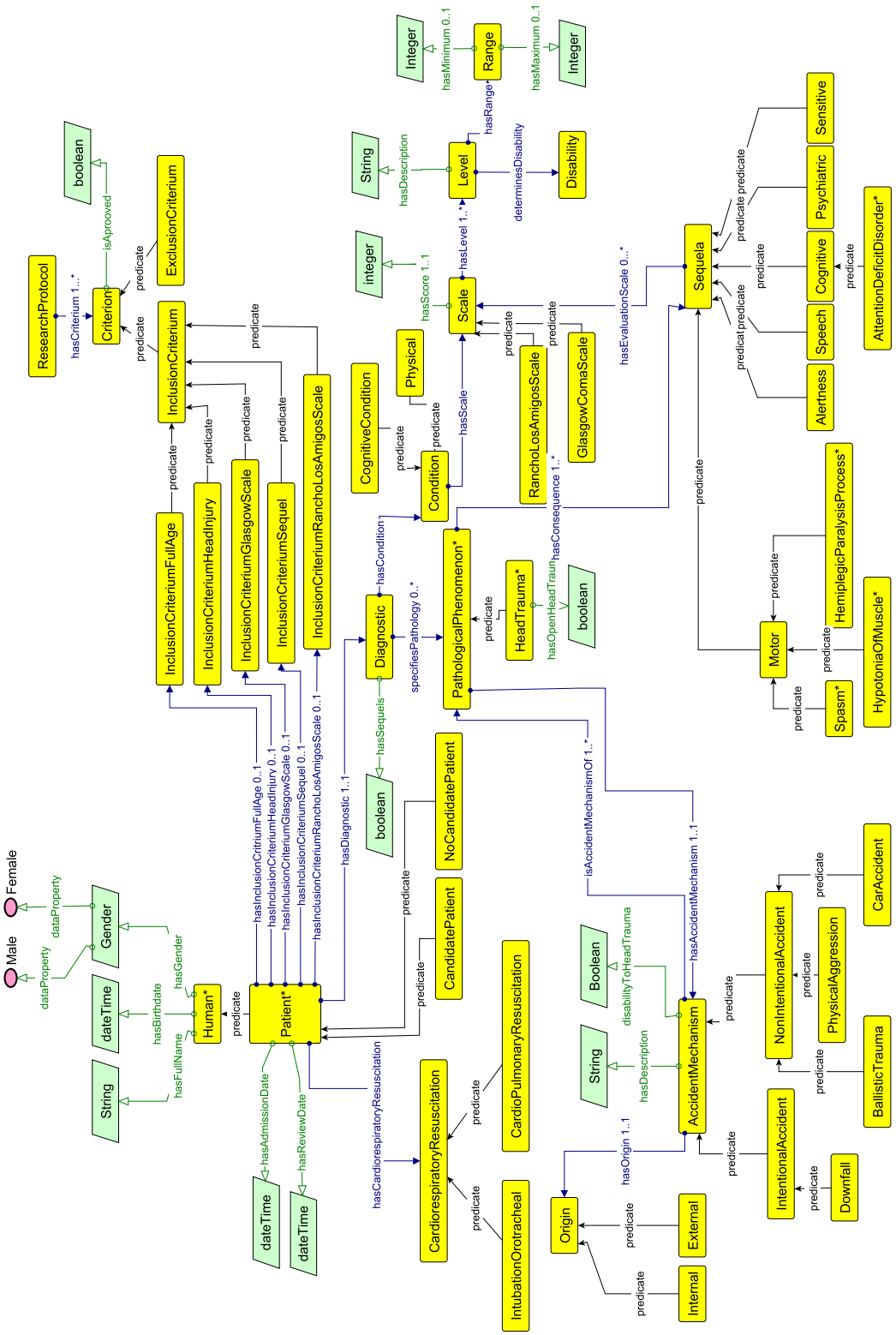


Fig. 4. CERPRO model in yED, showing the INR Inclusion Criteria.

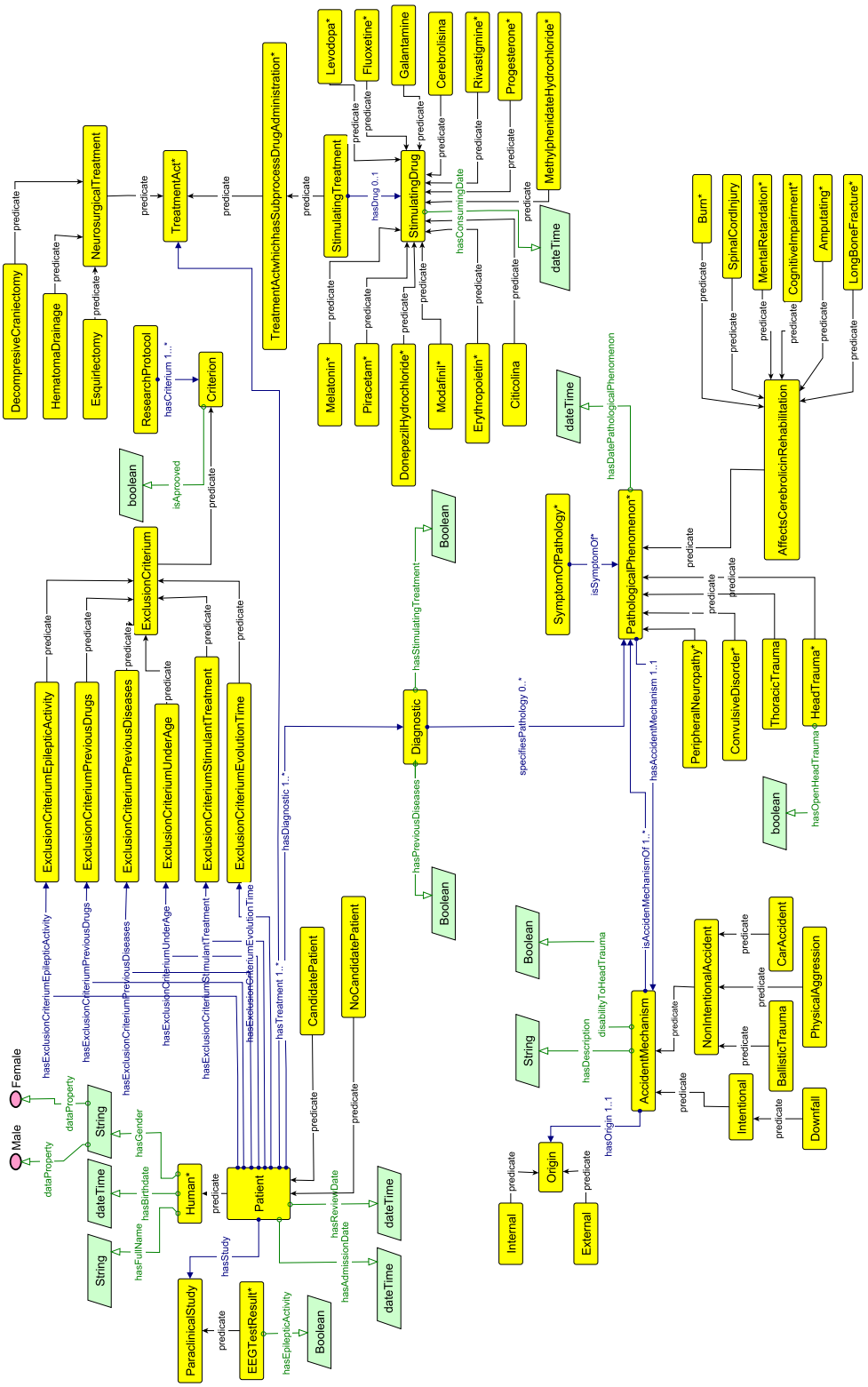


Fig. 5. CERPRO model in yED, showing the INR Exclusion Criteria.

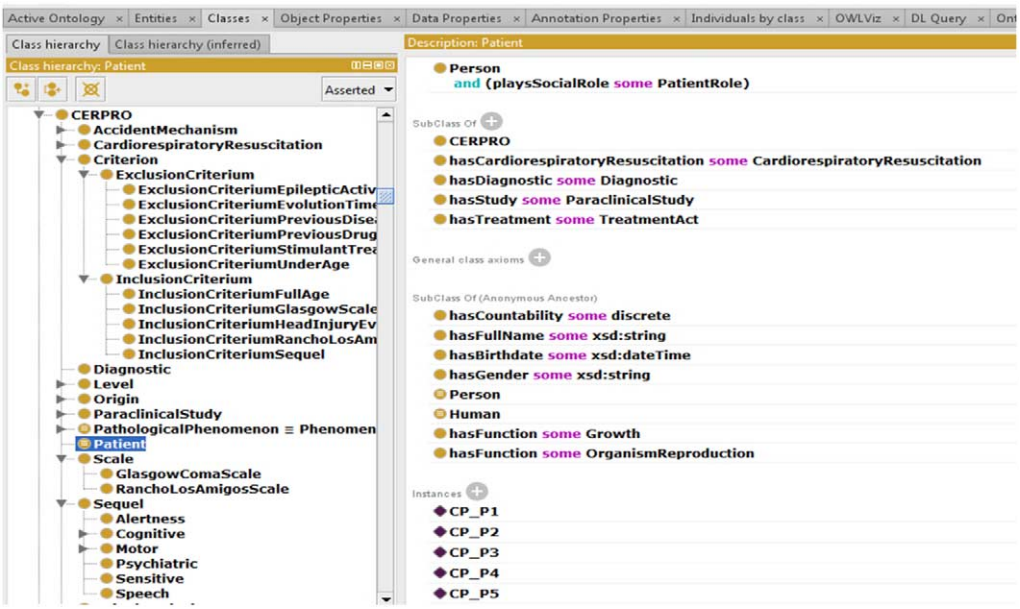


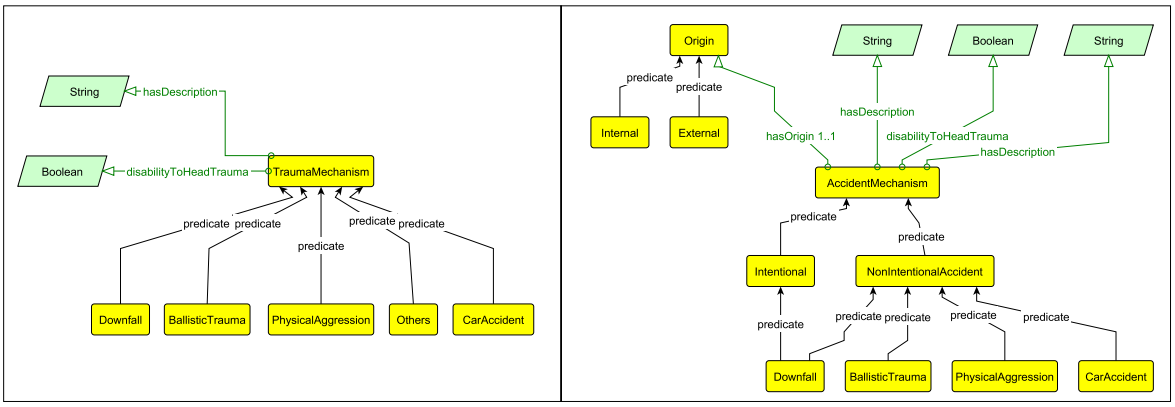
Fig. 6. CERPRO in Protégé, displaying the OWL axioms for the patient class, and patients as instances (CP_P1, CP_P2, etc.).

Case Study: The KE walked through the ontology model with the neurologists. During this exercise, many misconceptions were corrected in the model. For example, in iteration 5 during this validation activity, the neurologist detected that the class *TraumaMechanism* (which indicates how trauma can occur e.g., downfall, ballistic, physical aggression, etc.) had missed concepts. Specifically, more classes were needed to model whether the trauma occurred through an accident (which might be either *intentional* – e.g., in a downfall, or *not intentional* – e.g., physical aggression and whether it has an internal *origin* due to a pre-existent sickness – e.g., epilepsy, heart attack, etc., or an external one – e.g., by a car accident.

Fig. 7 shows how the ontology was changed after this validation. The trauma accident concept was not detected by the KE from the CQs and domain analysis. This situation can happen because SMEs are not aware that they need this in the model early on, until the KE walks them through the model.

Activity 13. ***Knowledge Rule Implementation* (*KRI*)**

In this Activity, the KE implements the *Knowledge Rules* (from Activity 4) in a rule language (e.g., for OWL ontologies), by selecting the best strategy according to the objective of the ontology (e.g., to be used for inferencing knowledge, or querying). Three strategy options are proposed to implement KRs: **Option 1)** for evaluation with rule engines, such as Jess (Sandia National Laboratories, 2020) or RuleML (RuleML Inc., 2020); **Option 2)** as injected axioms in the target ontology, which will be dynamically and periodically evaluated with a reasoner, e.g., Pellet (Clark&Parsia, 2011) with Semantic Web Rule Language (SWRL; W3C-SWRL, 2004); and **Option 3)** as queries to the KB, e.g., in Jena Fuseki (The Apache Software Foundation, 2020) by using a query language, such as SPARQL Query Language for RDF (SPARQL; W3C-SPARQL, 2008). Options 1 and 2 can be set up for an automatic evaluation, whilst option 3 is only executed by direct requests to the KB, through a SPARQL end point, or with an information system. For any selected strategy, the *Knowledge Rules* will be implemented in the Ontology Beta, per each CODEP iteration. Thus, the Input of this activity is the *Knowledge Rules* and the *Ontology*



(a) (b)

Fig. 7. The (a) shows the previous version of the *TraumaMechanism* class; (b) shows the same class but it was renamed as *AccidentMechanism*, with additional classes for modeling whether the trauma was an accident that might be intentional (e.g. a third-person pushed the patient) or NonIntentionalAccident (e.g., a person falls due to a sickness).

Table 6

(Excerpt) Inclusion and Exclusion Criteria (IC and EC respectively) in combination, comprises the Knowledge Rules (KR). In each KR, the IC/EC must be simultaneously executed according to the rows below in this table. For example, KR1 requires to simultaneously execute IC-1 and EC-1. Inclusion and Exclusion Criteria must be simultaneously executed in each KR

Basic Neurological Exploration (At first-contact hospital)			Specialized Neurological Exploration (At INR)		
ID	Inclusion Criterion	Exclusion Criterion	ID	Inclusion Criterion	Exclusion Criterion
KR-1	IC-1	EC-1	KR-7	IC-7	EC-5
KR-2	IC-2	EC-2	KR-8		EC-6
KR-3	IC-3	EC-3	KR-9		EC-7

Beta (this only from iteration 2); and the Output is the *Ontology Beta* (with the implemented knowledge rules).

Due to the iterative nature of CODEP, Activity 13 can be iteratively be updated with Activity 10–12, for modeling, and designing the ontology, and defining the knowledge rules.

Case Study: The knowledge rules from Table 6 were implemented as SPARQL queries to be run in the KB. This decision was made because the INR's medical team required to observe the CQs' responses when performing the validation (later on, in Activity 15); otherwise if the rules were run in the background through a rule engine or a reasoner, it would not be possible to observe the results in real-time by the SME (however, the rule implementation as in Options 2 is part of the on-going work, further the validation task). To illustrate this, **R-3** (from Table 6) is included in Table 7, where both **IC-3** and **EC-3** implemented in SPARQL, need to be simultaneously executed and evaluated in true since **R3** dictates this (Table 6). At the end of this Activity, *CERPRO Beta* contained all the Table 6 knowledge rules, implemented as SPARQL queries. It is pointed out that these queries were reviewed and edited in each of the 11 iterations for CERPRO, which implied reviewing/edit CERPRO as well.

Milestone II: It is obtained the *Ontology Beta*.

Phase III. **Ontology Verification and Validation**

The objective of Phase III is to implement a Knowledge Base from the ontology that is verified and validated (see Fig. 8). The activities for this phase are as follows:

Table 7
Rule R-3 implementation in SPARQL

RULE R-3 (Simultaneous execution of IC-3 and EC-3)		
Queries SPARQL	**IC-3: Sequelae Identification**	**EC-3: Identification of previous pathologies (to the current TBI)**

	PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
	PREFIX cer: <http://www.semanticweb.org/edma/liim/2017/CERPRO-OpenGALEN8#>
	PREFIX opg: <http://www.opengalen.org/owl/opengalen.owl#>
	SELECT ?name ?sq
	WHERE {
	?p rdf:type opg:Patient.
	?p cer:hasFullName ?name.
	?p cer:hasDiagnostic ?d.
	?d cer:specifiesPathology ?phat.
	?phat opg:hasConsequence ?sequel
	BIND(REPLACE(str(?sequel), '^.*(#|/)',"") AS ?sq)
	}
		SELECT ?name ?previousDiseases
		WHERE {
		?patient rdf:type opg:Patient.
		?patient cer:hasFullName ?name.
		?patient cer:hasDiagnostic ?d.
		?d cer:specifiesPathology ?previous.
		?previous rdf:type
		?pathologicalPhenomenon.
		?pathologicalPhenomenon rdfs:subClassOf*
		cer:AffectsCerebrolicinRehabilitation.
		BIND(REPLACE(str(?previous),'^.*(#|/)',"")
		AS ?previousDiseases)
		}

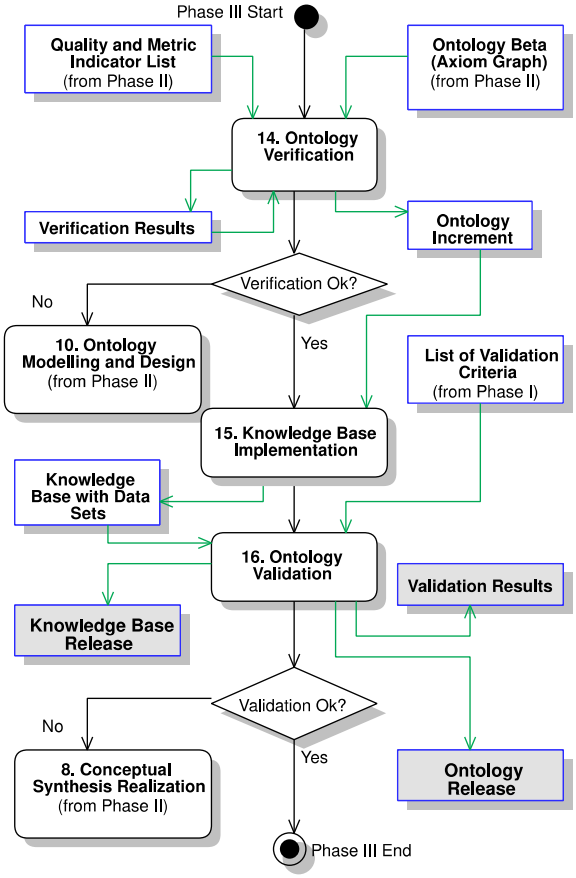


Fig. 8. Phase III flowchart.

Activity 14. ***Ontology Verification*** (OV)

This activity aims at verifying the ontology design by measuring its quality, according to the defined *Quality and Metric Indicator List* (Input from Activity 9). The *Verification* definition in CODEP is based on the one provided in SWEBOK (IEEE-SWEBOK, 2014), as follows: "*Verification is an attempt to ensure that the product is built correctly, in the sense that the output products of an activity meet the specifications imposed on them in previous activities*". In this context, this activity verifies the ontology design in terms of the quality indicators that were defined in Activity 9.

The verification is run by applying the metrics included in the indicators to obtain measurements, which will give insights about the ontology design, and then to correct deviations if the results are under the expectations. This will guide the KE to review and improve the design of the ontology and conduct further iterations if needed. Examples of improvements that can be performed are the addition of missing concepts, axioms, and properties and changes in the hierarchies. Metrics that can be measured are model expressiveness, completeness, and semantic consistency between the target ontology and reused ontologies. The Output will be either: The *Verification Results* and the verified *Ontology Increment*; or the decision to go back to Activity 10 for a new iteration. Otherwise, if the ontology design quality is reached, then the next Activity 15 is performed.

Short iterations from Activity 10–14 might be done, to adjust the ontology modeling and design, until reaching the expected quality as stated in Activity 9.

Case study: The *Quality and Metric Indicator List* (from Activity 9) is applied to measure the CERPRO design quality: *Model Completeness*, *Semantic Consistency between Models*, and *Model Expressiveness*. Obviously, as KEs, our ideal expectation is to get these indicators as near as possible to 100%, however, without applying metrics the perception about reaching such expectations would be only subjective. Thus, these selected metrics gave us quantitative data to measure how close are we getting to 100%, and insight into what specific points should the ontology of CERPRO be improved. In the following, we explain how we made use of the Metrics:

Model Completeness measures if all CQ concepts are implemented as concepts in the ontology. For example, for CQ_1 (see Table 8), the number of concepts that are identified is $CQC_1 = 5$ and the number of modelled concepts is $MC_1 = 8$. Therefore, Completeness for CQ_1 is the ratio 1.6 indicating that 5 concepts come from CQ_1 but 8 concepts were modeled. This means that 3 concepts are not traced back directly from the CQ_1 description, however, these added concepts are needed to complete the design (as the sub-types, *AccidentMechanism*, *IntentionalAccident*, and *NonIntentionalAccident*). For every CQ (CQ_i), the completeness measurements are similarly calculated. The last row in Table 8 shows the total of the metric: ***comCQ*** $= 1.1$; this indicates that all identified concepts from the CQs were modelled in the ontology, including some additional concepts (required for the CQs modeling as commented). According to the indicator range, it can be concluded that CERPRO is in good shape in terms of completeness (however the result shown was obtained after Iteration 11). Otherwise, when ***comCQ*** was near 0 (meaning that CERPRO was poor in terms of the CQs concepts modeling) more CODEP iterations were required from Activities 10–13 until ***comCQ*** became near to 1.

The advantages of using the *Model Completeness* metric in an iteration for driving improvements are as follows:

1) Check the traceability of the concepts from the CQs to the ontology model. As it can be noticed from Table 8, the calculation of the Model Completeness indicator has allowed us to document and keep track of the traceability of the Ontology Model concepts: the ones that are identified directly from the CQ and the ones that are not.

Table 8
(Excerpt) metrics application for measuring the model completeness indicator

ID-CQ	CQ	Identified concepts	Modeled Concepts	Metrics
CQ1	Has the *TBI* (Traumatic Brain Injury) occurred as an accident result (*car accident*, *downfall*, *ballistic accident*, *physical aggression*)?	1. TBI 2. car accident 3. downfall 4. physical aggression 5. ballistic situation Total: 5	**Identified Concepts in the CQ**: 1. HeadTrauma: 2. CarAccident 3. Downfall 4. BallisticTrauma 5. PhysicalAggression Total: 5 **Other Required Concepts to Complete the Modeling**: 6. AccidentMechanism: 6.1. IntentionalAccident 6.2. NonIntentionalAccident Total: 3	$CQC_1 = 5$ $MC_1 = 5 + 3 = 8$ $\frac{MC_1}{CQC_1} = 1.6$

$$\textbf{\textit{comCQ}} = \frac{\sum_{i=1}^{n} \frac{MC_i}{CQC_i}}{n} = \frac{1.6+0.92+1+1.5+1.25+1+1+1+1.14+1+1+1+1+1}{14} = \frac{15.41}{14} = 1.1 \; n = 14$$

2) Add missing concepts and axioms in the Ontology. For example, in iteration 1 the *Model Completeness* metric for CQ_{13} was $MC_{13}/CQC_{13} = 0.875$ (see Table 9). This is a high result; however, the ratio is not 1. This made us check that the CQ_{13} has 8 concepts but only 7 of them were modelled in the Ontology Model in activities 10 and 11. In these activities, the design rationale was to only include the concept of *Disability* in the model, and to have the disability types: severe and moderate as instances. After obtaining 0.875, CQ_{13} was reviewed and checked for missing concepts. It was decided to introduce a new concept: *Range*, to determine the disability type (severe and moderate). This was modeled in the ontology as a class axiom to allow changing the range thresholds (minimum and maximum) according to the medical needs, which would allow flexibility in the disability determination. After updating the ontology axioms in the new 2nd iteration, the $MC_{13}/CQC_{13} = 1$.
3) Update hierarchies. For example, in iteration 1, CQ_2 had 14 nouns (sequela concept, 4 specific sequelae, sequela type concept, 6 sequelae types, patient, and the TBI) were identified from CQ_2 (see Table 9) and the design decision taken was to model the sequela types with the class *SequelType*, and to have the types as instances. This resulted in 8 modelled concepts and the ratio as $MC_2/CQC_2 = 8/14 = 0.57$, which can be improved. This motivated the review of the CQ concepts being modelled. It was then observed that the KB will have a sequel-type instance for each sequela and for each patient's diagnosis, which will fill the KB with repetitive instances. On the contrary, if each sequela type has only an instance and all the patient's sequelae are related to only these few instances, then it would be better to have them as classes. This led us to take the decision of modeling the sequela types as classes, and to classify all the sequelae under such types, creating a sequela hierarchy (see Fig. 4), where each branch is a sequela type. This would give the advantage of using the inherited properties, across the hierarchy tree. Then, in the next 2nd iteration, after the ontology was modified, the ratio became $MC_2/CQC_2 = 13/14 = 0.92$.

Table 10 presents the results for S*emantic Consistency between Models* indicator measurements. In this table we have defined columns for: the *CQ description*, the *Identified Concepts in the CQ* (to docu-

Table 9
(Excerpt) previous model completeness metric calculation, where CQ_2 and CQ_{13} metrics had lower figures

ID-CQ	CQ	Identified concepts	Modeled Concepts	Metrics
CQ2	Does the patient present sequelae that are associated with TBI? What is the sequela type: psychiatric, cognitive, motor, sensitive, alertness, and language)? Is that sequela one of the following: hemiplegic paralysis process, attention deficit disorder, hypotonia of muscle, and spasm?	1. Patient 2. Sequela type: 2.1. alertness 2.2. motor 2.3. sensitive 2.4. language 2.5. psychiatric 2.6. cognitive 3. Sequela: 3.1. hemiplegic paralysis process 3.2. attention deficit disorder 3.3. hypotonia of muscle 3.4. spasm Total: 13	1. Patient 2. Sequela type 3. Sequela: 3.1. HemiplegicParalysisProcess 3.2. HypotoniaOfMuscle 3.3. Spasm 3.4. AttentionDeficitDisorder Total: 7	$CQC_2 = 14$ $MC_2 = 8$
		Repeated concepts in previous CQs 4. TBI Total: 1	**Repeated modeled concepts** 1. HeadTrauma Total: 1	
CQ13	Does the *patient* present a physical/cognitive *condition* in the *Glasgow* scale that locates him/her in a *severe* or *moderate* disability?	1. Condition 1.1. cognitive 1.2. physical 2. Severe disability 3. Moderate disability Total: 5	1. Condition 1.1. cognitive 1.2. physical 2. Disability Total: 4	$CQC_{13} = 8$ $MC_{13} = 7$ $\frac{MC_{13}}{CQC_{13}} = 0.875$
		Repeated concepts in previous CQs: 4. Patient 5. GCS 6. GCS grade Total: 3	**Repeated modeled concepts** 3. Patient 4. GlasgowComaScale 5. GlasgowComaScale grade Total: 3	

$$\textbf{\textit{comCQ}} = \frac{\sum_{i=1}^{n} \frac{MC_i}{CQC_i}}{n} = \frac{1.6+0.57+1+1.5+1.25+1+1+1+1.14+1+1+1+0.875+1}{14} = \frac{14.56}{14} = 1.06 \quad n = 14$$

ment the relevant concepts from the CQ text, *Modelled Axioms in Open Galen*, (to document the axioms that already exist in OpenGalen, which model the concepts from the CQ), *Modelled* Axioms in CERPRO (to document the axioms that model the concepts from the CQ in CERPRO), and the *Metrics* (for each of the 14 CQs to calculate ***SemCM*** according to Equation 5 from Activity 9). The metric calculation is based on the ratio $\frac{(MCOG_i+MCC_i)}{CQC_i}$, which measures the number of modelled concepts in the reused ontology (OpenGalen) and the target ontology (CERPRO) and takes an average by dividing the identified concepts from the CQ_i. Thus, if the metric is near 1, then this means that the concepts are not duplicated. If the metric is near 2, then this means that potential axioms are duplicated in the ontology. For example,

Table 10
(Excerpt) metrics application for measuring the indicator: semantic consistency between models

ID-CQ	CQ	Identified concepts in the CQ	Modeled Axioms in OpenGalen	Modeled Axioms in CERPRO	Metrics
CQ9	Does the patient use any stimulant treatment? Does the stimulating treatment is one of the following: Melatonin, Piracetam, Modafinil, Methylphenidate, Progesterone, Donepezil, Erythropoietin, Citicoline, Rivastigmine, Cerebrolysin, Galantamine, Flupxetine, Levodopa?	1. Stimulating treatment: 1.1 Melatonin 1.2 Piracetam 1.3 Modafinil 1.4 Methylphenidate 1.5 Progesterone 1.6 Donepezil 1.7 Erythropoietin 1.8 Citicoline 1.9 Rivastigmine 1.10 Cerebrolysin 1.11 Galantamine 1.12 Flupxetine 1.13 Levodopa Total: 13 **Repeated concepts in the CQs**: 1. Patient Total: 1	1. Melatonin 2. Piracetam 3. Modafinil 4. MethylphenidateHydrochloride 5. Progesterone 6. DonepezilHydrochloride 7. Erythropoietin 8. Citicoline 9. Rivastigmine 10. Flupxetine 11. Levodopa Total: 11	1. Stimulating Treatment 2. StimulatingDrug: a) Citicoline b) Cerebroly-sin c) Galantamine Total: 5 **Concepts in previous CQs**: 3. Patient Total: 1	$MCOG_9 = 11$ $MCC_9 = 5$ $CQC_9 = 13$ $\frac{11+5}{13} = 1.23$
SemCM $= (\sum_{i=1}^{n} \frac{MCC_i+MCOG_i}{CQC_i})/n$ $n = 14$		$\frac{1.8+2+0.5+0.87+1+1+1+1+1.23+1+1.5+1+1+1}{14} = \frac{15.9}{14} = 1.13$			

for *CQ9* the *Semantic Consistency* is 1.23 $\frac{(MCOG_9+MCC_9)}{CQC_9}$, where $MCOG_9 = 11$, i.e., 11 concepts of the CQ are modeled in OpenGalen (column *Modelled Axioms in OpenGalen*), $MCC_9 = 5$, i.e., 5 concepts of this CQ are modeled in CERPRO (column *Modelled Axioms in CERPRO*), and $CQC_9 = 13$, i.e., there are 13 concepts identified from the CQ (column *Identified Concepts in the CQ*). This considers eliminating those concepts that were already counted in previous CQs (e.g. *Patient* was already counted in the *CQ1* calculation). This metric considers that several concepts are introduced into the ontology for modeling purposes such as *StimulatingTreatment* which is not found in CQ_9, but is needed for creating a hierarchy for *StimulatingDrug* which also requires *Citicoline* and *Cerebrolysin*. According to the metric definition in Equation 5, since the value is very near to 0, the resulting ratio 1.23 is good after performing 11 iterations. The rest of the CQ_i measurements for the 14 CQs are similarly calculated (omitted for space reasons) and the resulting average S*emantic Consistency between Models* is indicated in the last row in Table 10, which was the resulting number in iteration 11, meaning that this metric indicates that there are not duplicate concepts in the ontology.

1) *To check and remove repeated concepts between OpenGalen and CERPRO*. For example, 5 concepts are identified from CQ_{12} (see Table 11) where 3 came from OpenGalen and 3 were just created in CEPRO, resulting in a ratio as, $\frac{(MCOG_{12}+MCC_{12})}{CQC_{12}} = 1.2$ (and $SemCM = 1.17$). This value is over 1, which might indicate that an axiom is repeated. We then reviewed the *CQ*12 and its modeled axioms, and we realized that the data property that was created for CERPRO: *hasDateTrauma* was

Table 11
(Excerpt) previous semantic consistency between models metrics calculation, where CQ12 has repeated axioms

ID-CQ	CQ	Identified concepts in the CQ	Modeled Axioms in OpenGalen	Modeled Axioms in CERPRO	Metrics
CQ12	Does the patient have a TBI between severe (<1 week) or chronic (>6 months)? What is the TBI evolution time?	1. TBI 1.1. Chronic 1.2. Severe 2. TBI evolution time Total: 4 Repeated concepts in the CQs: 1. Patient Total: 1	Classes 1. None Object Properties 1. None DataProperties – hasDatePathological-Phenomenon exactly 1 xsd:date-time Total: 1 Repeated classes in previous CQs: 1. Patient 2. HeadTrauma Total: 2	Classes: 1. SevereHeadInjury 2. ChronicHeadInjury Object Properties 1. None DataProperties 1. hasDateTrauma exactly 1 xsd:datetime Total: 3	$MCC_{12} = 3$ $MCOG_{12} = 3$ $CQC_{12} = 5$ $\frac{3+3}{5} = 1.2$
$\boldsymbol{SemCM} = (\sum_{i=1}^{n} \frac{MCC_i+MCOG_i}{CQC_i})/n$ $n = 14$		$\frac{1.8+2+0.5+0.5+1+1+1+1+1+1.23+1+1.5+1.2+1.2}{14} = \frac{15.93}{14} = 1.17$			

the same meaning as the inherited *hasDatePathologicalPhenomenon* from OpenGalen (from the *HeadTrauma*'s parent class: *PathologicalPhenomenon*). *Thus, we* eliminated the recently created, *hasDateTrauma* property and kept the inherited one. Thus, the new ratio was $\frac{(MCOG_{12}+MCC_{12})}{CQC_{12}} = 1$.

The advantages of using the S*emantic Consistency between Models* metric in an iteration for driving the improvements are as follows:

1) *To trace the concepts between an ontology and a third-party one*. The *SemCM* metric was used in the creation of all the CERPRO classes when reusing the ones from OpenGalen since the objective was to always keep the metric slightly over 1. In each iteration, this metric is calculated to record which concepts were created in CERPRO and which were already modeled in OpenGalen. The method is as follows, for each CQ_i, firstly, we identified for each of the $CQ_i's$ concepts which ones are compatible in OpenGalen by reading the annotations and by analyzing all the concept's axioms. Several concepts in OpenGalen which were identified by the KE, when reviewed by the SME, were not appropriate in terms of the medical description. Therefore, we created new ones in CERPRO to satisfy the doctors' requirements. These decisions can be observed for each CQ_i in Table 10, columns The Modelled Axioms in OpenGalen and CEPRO. Secondly, we counted the *CQi* concepts and the ones from OpenGalen, and we recorded the corresponding CQ_i metric calculation. For each CQi whose ratio over 1, we checked the modeled concepts of these CQs and if similar modeled concepts are found in OpenGalen's axioms for this CQ. If some axioms were already in OpenGalen, then we eliminated any similar axioms that we created in CERPRO.

Table 12
(Excerpt) metrics application for measuring the model expressiveness indicator

ID-CQ	CQ	Identified concepts	Modeled Properties and Classes		Metrics
CQ1	Has the TBI (Traumatic Brain Injury) occurred as an accident result (car accident, downfall, ballistic accident, physical aggression, or others)?	1. TBI 2. car accident 3. downfall 4. physical aggression 5. ballistic situation Total: 5	1. HeadTrauma objectProperty: 1 dataProperty: 1 2. IntentionalAccident objectProperty: 0 dataProperty: 0 3. Downfall objectProperty: 0 dataProperty: 0 4. NonIntentionalAccident objectProperty: 0 dataProperty: 0	5. CarAccident objectProperty: 0 dataProperty: 0 6. BallisticTrauma objectProperty: 0 dataProperty: 0 7. PhysicalAggression objectProperty: 0 dataProperty: 0 8. AccidentMechanism objectProperty: 2 dataProperty: 1 Total: 8	$MC_1 = 8$ $O_1 = 3$ $D_1 = 2$ $H_1 = 6$ $\frac{(O_1+D_1+H_1)}{MC_1} = \frac{11}{8} = 1.37$
$\boldsymbol{pR} = \frac{\sum_{i=1}^{n}(\frac{D_i+O_i+H_i}{MC_i})}{n}$ $n = 14$		$\frac{1.38+1.5+0.5+1+1+0.75+1.75+1.75+0.88+1+1+1+0.88+1.42}{14} = \frac{15.80}{14} = 1.13$			

Table 12 presents the results for the *Model Expressiveness* indicator measurement. The $\boldsymbol{pR}$ measures if the model is presenting *at least* the same number of relationships against the modelled concepts from the CQs (but it is expected to find this number much larger). For each CQ_i, the number of object/data properties and the inheritance relationships are counted, then this number is divided by the modeled classes given by the specific CQ_i, to obtain a ratio that indicates if on average each class has at least a relationship. The result is a balance between the ratio counting for each CQ (one could have no data/object properties just inheritance relationships, but other CQs could compensate this number and have no inheritance relationships), but each CQ's ratio will give a deeper detail about the balance between the modeled CQ's concepts and the relationships in the model. Thus, if $\boldsymbol{pR}$ is less than 1, it indicates that the model requires more expressiveness, and therefore more CODEP iterations are required from Activities 10–12 until $\boldsymbol{pR}$ becomes equal to 1 (or greater). For example, for CQ_1, there are 3 object properties (O_1), 2 data properties (D_1), 6 inheritance relationships (H_7), and 8 modeled classes (MC_1), giving a total of $\frac{(O_1+D_1+H_1)}{MC_1} = 1.37$. The resulting total $\boldsymbol{pR}$ is 1.13 in the last row in Table 12, meaning that every concept has on average an axiom relationship.

The advantage of using the *Model Expressiveness* metric in an iteration for driving the improvements is to identify vertical ontologies (mostly hierarchies). This metric was very helpful in identifying whether a CQ_i is being modeled with only hierarchies with barely data/properties (very vertical shape), in which case, the ratio for the given CQ_i will be near 0. This happens when a CQ_i requires a hierarchy and the class definition lacks axioms as data/object properties, this might be correct if the CQ_i description requires this, but the metric will put quantitatively the vertically-modelled CQ_i and make it possible to review that part of the ontology and to correct the model. For example, a previous calculation of the CQ'_1s ratio (Table 13), $\frac{(O_1+D_1+H_1)}{MC_1}$ gave 0.83, which indicates that there are classes without relationships in that section of the ontology (see Fig. 4, left-bottom side). Initially, in iteration 1 we had the *AccidentMechanism* and its hierarchy connected to *PathologicalPhenomenon*, through an object property. However, the result of this metric made us review the CQ description and the modeling. Thus, we decided to create the accident mechanism type hierarchy, connected as well with the inverse object property, *AccidentMechanismOf*, and to differentiate the *NonIntentionalAccident* from the *IntentionalAccident*, as the Downfall

Table 13
(Excerpt) previous model expressiveness calculation, where CQ1 has a low ratio

ID-CQ	CQ	Identified concepts	Modeled Properties and Classes		Metrics
CQ1	Has the TBI (Traumatic Brain Injury) occurred as an accident result (car accident, downfall, ballistic accident, physical aggression, or others)?	6. TBI 7. car accident 8. downfall 9. physical aggression 10. ballistic situation Total: 5	1. HeadTrauma objectProperty: 1 dataProperty: 0 2. AccidentMechanism objectProperty: 0 dataProperty: 0 3. CarAccident objectProperty: 0 dataProperty: 0	4. PhysicalAggression objectProperty: 0 dataProperty: 0 5. BallisticTrauma objectProperty: 0 dataProperty: 0 6. Downfall objectProperty: 0 dataProperty: 0 Total: 6	$MC_1 = 6$ $O_1 = 1$ $D_1 = 0$ $H_1 = 4$ $\frac{(O_1+D_1+H_1)}{MC_1}$ $= \frac{5}{6} = 0.83$
$pR = \frac{\sum_{i=1}^{n}(\frac{D_i+O_i+H_i}{MC_i})}{n}$ $n = 14$		$\frac{0.83+1.1+0+0.87+1.5+0.5+1.5+0.75+0.88+0.5+0+0+0.7+0}{14} = \frac{9.13}{14} = 0.65$			

concept (since the SME was interested in an intentional downfall, and not caused due to a previous pathology). The resulting CQ'_1s ratio, with these corrections in the next iteration, was 1.37, which means a more balanced ontology model (as observed in Fig. 4). Thus, this metric calculation made us review our knowledge in the domain, to figure out if some concepts or relationships were badly understood, but not exactly missing, which would lead to a model correction in terms of expressiveness.

Therefore, after iteration 11, CERPRO achieved the verification expectations in terms of these 3 metrics.

Activity 15. ***Knowledge Base Implementation (KBI)***

The aim is to create a KB to test if the ontology is properly answering the CQs. This is entirely a software installation task. Thus, the *Ontology Increment* is used (Input from Activity 13) to set up the KB platform in servers, for creating a dataset to be populated with selected trial data for creating triplets (ABox for OWL ontologies). This allows the execution of the knowledge rules (from Activity 12) and the queries (created from the CQs) for validation. The Output of this Activity is the *Knowledge Base* with a dataset.

Case Study: Apache Jena (The Apache Software Foundation, 2020) was chosen as the platform to create the KB, and Fuseki was chosen as the SPARQL End Point to remotely query the KB (from a web browser and a software application). The KB called "CERPRO-KB" was created, which can be found at http://liim.izt.uam.mx:8080/fuseki/ (Fig. 9). For this, the Ontology Increment of CERPRO from Activity 13 was uploaded, producing the dataset, CERPROv-5 (by the time this paper is written). With the support of INR's medical team, the CERPRO-KB was created with anonymized data from patients (see Table 14). This data describes different relevant diagnostics for the medical team, intending to test the KB's responses with already known diagnostics. In an upcoming scenario, INR will keep the knowledge base on its premises to preserve personal data confidentiality.

Activity 16. ***Ontology Validation (OVA)***

This activity aims to validate whether the *Ontology Increment* is answering properly the CQs from the SME's perspective in a quantitative perspective. The *Validation* definition in CODEP is based on the one provided SWEBOK by (IEEE-SWEBOK, 2014), as follows: "*Validation is an attempt to ensure that*

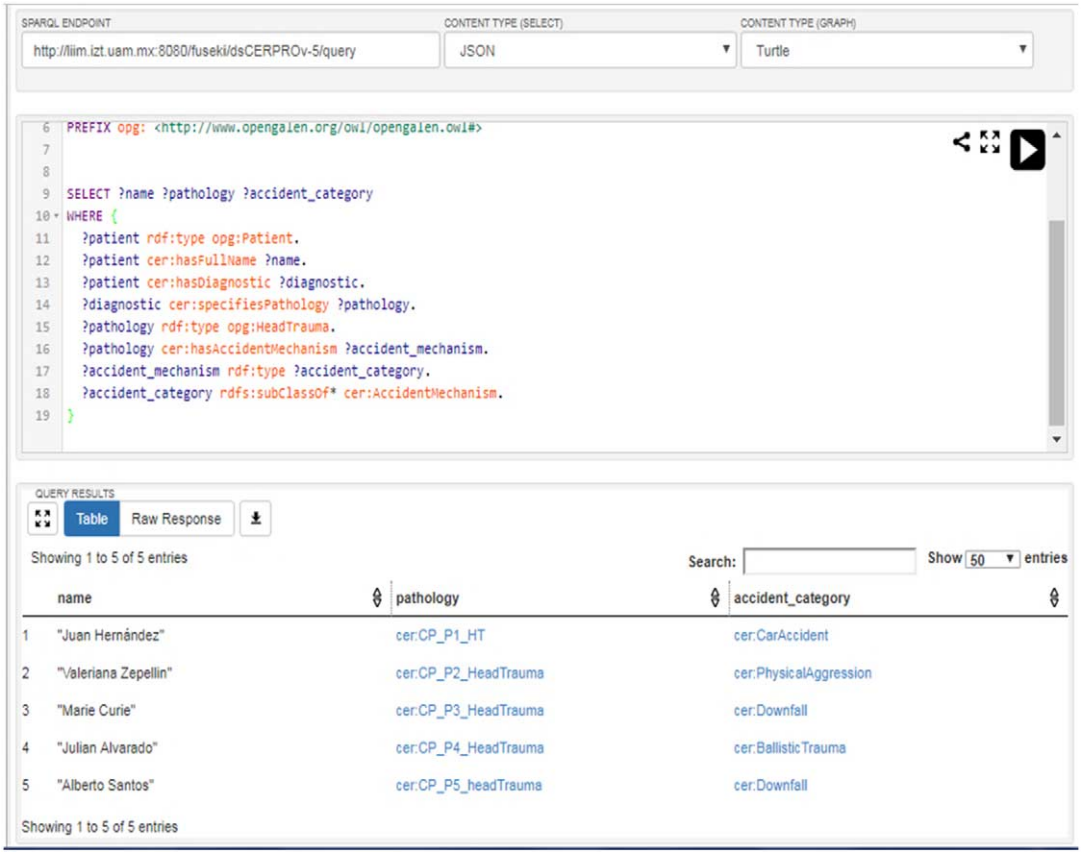


Fig. 9. The SPARQL implementation of CQ-1 and the CERPRO response.

the right product is built – that is, the product fulfills its specific intended purpose". In this context, this activity validates whether the ontology fulfills the CQs that were defined in Phase 1, in terms of querying the KB and evaluating the results according to the *ontology validation criteria*, through the metrics that were defined in Activity 5.

The Input is the KB with a dataset and the *List of Validation Criteria* (from Activity 5), which are applied to measure whether the SME expectations have been satisfied or not. For each metric, the SME is asked to indicate their Expected value which indicates when they are satisfied with the result of the metric.

The Output of this activity is either: 1) the *Validation Results*, the *Ontology Release* version and the *Knowledge Base Release*, then, ending the CODEP process; or 2) the decision to perform another short iteration from Activity 10–15, to update the ontology according to the validation results if the ontology does not fulfill the criteria. Due to the iterative nature of the process, it must be ensured that any changes performed to the ontology during an iteration do not affect the obtained satisfaction when answering the CQs in the previous iteration. Therefore, the final iteration must include all the CQs, already validated.

Short iterations from Activity 10–15 might be done, to adjust the ontology modeling and design, to respond to the CQs as the SME expects.

Case Study: We defined a protocol for performing the validation activity (see Section *Validation Protocol* for details of the metrics application and results) to make sure that the SME's needs about the

Table 14
(Excerpt) trial data with patients data

Patient	Sequels	Previous Pathologies	Consumed Drugs	Epileptic Activity	GOSLCFS		Patient Enrolment into the CRP
CP_P1	CP_P1_AttentionDeficitDisorder	CP_P1_opht-LongBoneFracture	No	False	6	4	Refused (presents previous pathologies)
CP_P2	CP_P2_Hemiplegic-ParalysisProcess	No	CP_P2_Modafinil	False	10	3	Accepted
CP_P3	CP_P3_Hypotonia-OfMuscle	CP_P3_opht-LongBoneFracture	No	False	14	6	Refused (presents previous pathologies and GOS as chronic)
CP_P4	CP_P4_Spasm	No	CP_P4_Methyl-phenidateHydrochloride	False	7	6	Accepted
CP_P5	CP_P5_AttentionDeficitDisorder	CP_P5_MentalRetardation	No	False	11	4	Refused (presents previous pathologies)

CERPRO functionality and responses in the medical diagnostic identification are satisfied, according to their expectations stated in the *Validation Criteria*. This protocol applies the *List of Validation Criteria* that was defined for the case study in Activity 5, as follows: 1) *CQ completeness* (since the medical team is interested in implementing the whole CQs set), 2) *CQ response accuracy* (since the medical team expects accuracy in the responses when identifying diagnosis) and 3) *CQ response comprehensibility* (since the medical team needs to comprehend the responses from the knowledge base with no complication in the technical argot). The validation results were used for further improvements of CERPRO by undergoing iterations of CODEP (from Activity 10–15), which generated new increments of the ontology, till the validation criteria are satisfied near 100% of the expectations. We finally produced the CERPRO-v5, with the validated quality in iteration 11.

Milestone III: The *Ontology* and *Knowledge Base Release* is obtained.

4. Validation protocol

This Validation Protocol was followed in Activity 16 of CODEP to validate whether the CERPRO-KB truly responds to the medical team's needs (stated in the CQs in Activity 3).

The validation research question pursued is as follows: *Does the CERPRO ontology meet the expectations of the medical team, which are stated as CQs? To be able to answer this question, the used criteria are*:

1) *CQ accuracy*
2) *CQ completeness*
3) *CQ comprehension*

These 3-validation criteria follow the definition from Activity 5 (defined according to the medical team's needs), thereafter they are measured through Equations 1–3, respectively.

Data Collection: Before the meeting, 3 different templates were created to capture the measurements for the 3 validation criteria (from Activity 5), where each template has 14 rows (each for each *CQi*), each row has columns for the *CQ-ID*, the CQ description, the metrics' measurements, and the observations/ justification for giving such a measurement. The SME (the medical team's chief) completes the templates.

Table 15
Validation of CERPRO: response accuracy calculation, completeness, and comprehensibility results

CQs Response Accuracy	CQs Completeness	CQs Response Comprehensibility
$\boldsymbol{aCQ} = \frac{\sum_{i=0}^{n} \boldsymbol{ACQ_i}}{\mathbf{14}}; n = 14$	$\boldsymbol{cCQ} = \frac{\sum_{i=0}^{n} \boldsymbol{ICQ_i}}{\mathbf{14}}; n = 14$	$\boldsymbol{rCQ} = \frac{\sum_{i=0}^{n} \boldsymbol{RCQ_i}}{\mathbf{14}}; n = 14$
$1+1+1+1+1$	$1+1+1+1+1$	$1+1+0.8+1+1$
$+0.5+1+1+1$	$+1+1+1+1$	$+0.6+1+1+1$
$= \frac{+1+1+1+1+1}{14} = \frac{13.5}{14} = \mathbf{0.96}$	$= \frac{+1+1+1+1+1}{14} = \frac{14}{14} = \mathbf{1}$	$= \frac{+0.9+1+1+0.8+0.8}{14} = \frac{12.9}{14} = \mathbf{0.92}$

Time-framed Meetings: One-hour sessions were prepared with the medical team, where the knowledge engineer uses the following strategy: 1) executes the queries since the medical team is not experts in doing querying the KB, and 2) the physicians evaluated the observed KB responses (as in Fig. 9). The strategy was done for each of the 14 queries and the associated 3-validation metrics. In each session, the CERPRO-KB, which implements the Ontology Beta, was used.

Results: 11 iterations were conducted and 11 time-framed meetings were organized with the medical team. In iteration 11 the CEPRO-KB is called CERPROv-5 (see Fig. 9 SPARQL Endpoint box, which indicates CERPROv-5). In this last iteration good results were obtained, to finally get CERPRO capable to satisfy the medical team's expectations, according to the metrics' expected results. As it can be noticed from Table 15, the calculated $\boldsymbol{aCQ}$ resulted in **0.96**. Since it is near 1 as defined in Activity 5, it can be concluded that the medical team perceives highly accurate the 14 CQ responses average. Similarly, the $\boldsymbol{cCQ} = 1$ (completeness of all CQs) and $\boldsymbol{rCQ} = 0.92$ (response comprehensibility for each CQ) were calculated in Table 15. As all of them were nearly 1, it can be concluded that CERPRO meets the medical team's requirements in terms of *accuracy*, *completeness*, and *comprehensibility* (based on the CQs' responses and expected results stated in the metrics definition in Activity 5) and no new iterations have to be made.

Discussion: For space reasons, the above only presented the results for the validation protocol of the 11th iteration. However, in previous iterations the feedback obtained from the validation results was included. For example, regarding the *CQs Response Accuracy*, in increment CERPRO 2.0 it was observed that the ontology (through the KB) was not properly responding to $CQ - 11$: "Does the patient have any pathology previous to the current TBI that affects. . . " and as a result, the $\boldsymbol{ACQ_{11}}$ was 0.3, according to the SME. Since this result is much less than the end-users expectations (which is 100% for any *CQi* accuracy, according to the Expected Medical Team's value in Equation 1), thus, this implied that the ontology had to be updated for at least improving $\boldsymbol{ACQ_{11}}$; then, the Iteration 12 was started. It was also identified that in CERPRO 2.0, the TBI concept was modeled as *HeadTrauma* (from OpenGalen) with no specification about whether this is a pathology. Thus, to correct this issue, in the new increment CEPRO 3.0, *HeadTrauma* was modeled as a subclass of OpenGalen's *PathologicalPhenomenon*. It was until CERPRO 2.0 was validated that this issue was detected, and in CERPRO 3.0 this inheritance relationship has been properly defined. This example demonstrates the importance of the ontology validation (Activity 16), as this problem was only detected when the queries to the KB were executed, and the quality validation metrics were calculated.

Another example for new iterations based on validation metrics is regarding the *CQs Completeness*. This metric was used to measure whether each CQ queried is returning complete results. For example, in iteration 1 (CERPRO 1.0), when the result of the query for $CQ - 6$ ("Did the patient require orotracheal intubation or another measure of cardiorespiratory resuscitation. . . ") was viewed by the user, they scored 0.4 for Completeness. Therefore, the KE performed a new iteration to analyze the issue for this CQ. In iteration 2, the KE added a *CardiorespiratoryResuscitation* hierarchy in the ontology. In this way, the

end-user had a practical way to quantify the ontology deliveries, each time that a new CERPRO version was released until the completeness was 100% for all the CQs.

5. Related work

There have been approaches in the literature to support the design and development of ontologies. In this section, we review them and compare them to CODEP.

On-To-Knowledge (Staab, Studer, Schnurr, & Sure-Vetter, 2001) proposes to design an ontology with four main sub-processes: *Feasibility*, *Kick-off*, *Refinement* (which includes *Knowledge elicitation process with domain experts* and ontology *Formalization*), and *Evaluation*. This is the nearest approach to CODEP, since it is the only approach that is application-oriented ontology methodology, considers validation of the CQs/requirements, and tests the KB by obtaining answers for the CQs. However, it does not include metrics (verification or validation) are applied to formally guide the ontology increments, this means, it does not consider Evaluation's feedback loops to systematically improve the ontology design; and neither provides a formal definition of the phases, i.e., no roles and work products are specified to reproduce the process successfully in ontology design projects.

DILIGENT (Pinto, Staab, & Tempich, 2004) is one of the former ontology development methods, and it focuses on managing the ontology design with geographical-distributed teams; its activities are *Build*, *Local Adaptation*, *Analysis*, *Revision*, and *Local update*. It is mainly oriented to elicit, capture and manage the ontology concepts directly from the end-users. This approach has two main limitations: Axioms are only hierarchical which reduces the expressivity of ontologies and metrics are not formally used to perform V&V on the ontology. The process itself stems from whole-process cycles to release a new ontology version. NeOn (Suárez-Figueroa et al., 2007) provides a collection of scenario-based life cycle models for ontology development, in a similar way to how models are used in the software development lifecycle. This approach can be mainly used to guide ontology creation when other resources need to be integrated, such as reused ontologies, ontology patterns, and non-ontological resources (e.g., thesaurus, taxonomies, etc.). The life-cycle includes activities for the ontology formalization and implementation, and activities for V&V; however, it does not include guidance about how to manage the ontology improvements in further increments, based on the results of the V&V. Both DILIGENT and NeOn do not support a CQ-driven ontology development very systematically (for example, through metrics based on CQs), which means they are not processes to produce end-user oriented KBs.

Methontology (Fernández, Gómez-Pérez, & Juristo, 1997) considers CQs in the initial Specification phase, but there are never used in the next phases of the life cycle to validate the ontology release. Methontology supports verification in the Implementation phase, but verification results are not used to formally improve the ontology (for example through metrics). It is neither an iterative nor an incremental approach.

In the approach by Katsumi & Grüninger (2010), a life cycle is proposed for ontology development, which is strongly centered on integrating the use of theorem provers to ensure that ontology models are semantically correct. CQs are used, but end-users or SMEs are not involved since there is not a specific validation activity, which would be necessary to determine that the ontology truly meets end-user needs. The cycle is not iterative or incremental. In the approach by Garcia Castro et al. (2006) an ontology development process is proposed, which is iterative between the modeling and evaluation activities, and with the involvement of end-users in the early stages. The approach stems from defining and eliciting CQs by using Conceptual Maps (CMs), which are considered as an informal ontology model. The CMs

are used to get users' feedback and to improve the CMs. It proposes an iterative evaluation phase to formalize the ontology; however, the approach does not apply metrics and/or measurements to do this evaluation, which means that the ontology improvements are not quantitatively guided and no formal increments can be produced.

In the approach by Lasierra, Alesanco, Guillén, & García, (2013), a three-stage solution is presented: ontology development, ontology deployment in the application domain, and software implementation with the ontology incorporation for the KB. A waterfall lifecycle is proposed, and users are not involved in the process. It is not formally structured (does not describe the involved roles, the inputs/outputs, metrics usage, etc.) and does not include V&V activities.

There are additional approaches that stem from using CQs as the backbone for building the ontology design. In the approach by Malheiros & Freitas (2013), the authors present a system based on an algorithm to iteratively build the ontology based on CQs. It is a promising approach as it attempts to automatically generate the ontology, however, the system does not consider a deep analysis of the domain and does not consider using metrics to quantitatively validate whether the iterative ontology fulfills the CQs. In the approach by Ren et al. (2014), the authors present an approach for testing the ontology with the CQs formulated by stakeholders to automatically find out whether the ontology can answer such CQs, in terms of analyzing the Description Logic (DL) axioms. In the approach by Sousa, Soares, Pereira, & Moniz (2014), the authors suggest a template for writing CQs for building lightweight ontologies (defined as: with poor computational processing and no inferable constructs) that are mainly used by domain experts with no knowledge of OWL-DL. They propose a structure to specify the CQ's approach which can later represent in conceptual maps for ontology building.

In the approach by Grüninger & Fox (1995) and Uschold & Grüninger (1996), a methodology for the ontology design is proposed, which focuses on defining informal CQs as the means to capture the ontology requirements, which will be used to test whether the ontology is answering to the CQs. The formal CQs are used as the driver to implement the ontology expressiveness (axioms in a description logic), and as the means to formally evaluate whether the CQs have been answered by the ontology and its instances. This is an approach in line with ours in the sense that it is also grounding the CQs as a driver for ontology design, and the formal CQs concept is an approach to explore a more automatic evaluation. However, the methodology does not include an activity to consider metrics to evaluate quantitatively the ontology, nor there is guidance about how to proceed with the CQs when iterating the ontology. Since the process is not well-defined as it does not indicate inputs/outputs, roles for its activities it would be difficult to follow.

In the approach by Blomqvist & Öhgren (2006), the authors propose a methodology for manually constructing an ontology, with phases: requirements analysis, building, implementation, evaluation, and maintenance. The evaluation phase is used to compare the consistency between a manually created ontology against a semi-automatic created one. However, the methodology does not evaluate whether an ontology satisfies the goals and its applicability in answering queries and CQs, and the evaluation feedback (ontology verification) is not used for re-designing the ontology in further increments. In the approach by Noy & Mcguinness (2001) authors present a guide to developing ontologies based on the waterfall model (no-iterations). The guide considers a deep domain knowledge analysis. However, it does not consider V&V, nor the configuration of the KB platform; and the ontology development is not driven by CQs.

Some approaches provide verification and validation techniques, such as OntoQA (Tartir, Budak Arpinar, Moore, Sheth, & Aleman-Meza, 2005) and OOPS! (Poveda-Villalón, Suárez-Figueroa, García-Delgado, & Gómez-Pérez, 2009). OntoQA proposes a method to measure the ontology's quality in 3 dif-

Table 16
Features used for the comparison

Abbreviation	Description
UserInv	Indicates activities where end-users are involved. For the analysis purpose, we generalize the ontology development as follows: Domain Acquisition (DA), Ontology Building (OB), and Ontology Validation (OV).
QuantVal	Indicates whether the approach includes quantitative Validation with the user, by the use of metrics explicitly.
QuantVer	Indicates whether the approach includes defining quantitative ontology Verification, by the use of metrics explicitly
ActV&V	Indicates the inclusion of explicit Activities for Ontology Validation & Verification in the process, to incrementally improve the ontology,
IterLC	Indicates whether approaches support an Iterative Life Cycle for developing the ontology.
IncOntDev	Indicates whether approaches support Incremental Ontology Development; where releases are provided to the user.
StruMeth	Well-Structured Methodology, in terms of having a well-defined process: roles, work products, inputs, outputs, and a process guide.
CQ-Driven	Indicates whether the CQ guides the ontology development from the early stages till later stages of the life cycle.
App- KB	Indicates whether the process can create an application-oriented knowledge base (KB).

ferent perspectives: 1) the schema ontology quality, 2) the populated ontology, and 3) the KB compliance to the ontology schema. Particularly, the third perspective could be incorporated into CODEP in Activities 14 in further work, to enhance the CODEP evaluation for the ontology schema (Activity 14) and the populated ontology (Activity 16); and, the first and second approaches could be additional sources for defining metrics, as required in Activities 14 and 16. OOPS! (Poveda-Villalón, Suárez-Figueroa, García-Delgado, & Gómez-Pérez, 2009) provides a method (and online tool) for evaluating the ontology quality from the verification perspective, by providing a catalog of 40 good practices for ontology development and metrics to evaluate the ontology. However, OntoQA does not provide guidance or a process for applying the method in the ontology quality evaluation, but it can be complemented by CODEP to formally update the ontology, based on the suggested validation metrics, as part of CODEP's Activity 16.

This shows that CODEP provides a framework for building ontologies, which can be enriched and tailored with third-party approaches/techniques in specific activities. In fact, as future work, the verification method with (Ren et al., 2014) and the CQs structure proposed by Sousa, Soares, Pereira, & Moniz (2014) could be used to support the automatization of the CODEP process.

6. CODEP evaluation and further discussion

6.1. Comparing CODEP's features to other ontology development processes

This section discusses the CODEP's features in comparison to other available approaches in the literature for ontology design. To compare these approaches, we have used the features described in Table 16.

From Table 17, it can be observed that many approaches do not mention involving the user or domain experts in the process. Those that do, involve users in activities acquiring knowledge about the domain. Two approaches in addition to CODEP do involve the user invalidating the ontology or knowledge base to ensure that the user is satisfied and collect/update the ontology and define new CQs. On-To-

Table 17
Ontology process models benchmarking

Process	UserInv	Quant-Val	QuantVer	Act-V&V	IterLC	IncOnt-Dev	StruMeth	CQ-Driven	App-KB
CODEP	In DA, OV	Yes	Yes	Validation & Verification	Yes	Yes	Yes	Yes	Yes, in Activity 13,15
On-To-Knowledge (Staab, Studer, Schnurr, & Sure-Vetter, 2001)	In DA (Kick-off) and OB (Refinement)	No	No	Only Validation	Yes	Yes	No	Yes	Yes, a KB is set-up in Evaluation phase for testing the ontology, through the CQs
DILIGENT (Pinto, Staab, & Tempich, 2004)	In OB, OV (Local Adaptation)	No	No	Only Validation	Yes	Yes	Partially, for managing changes by the end-user	No	No
NeOn (Suárez-Figueroa et al., 2010)	In DA	No	Yes	Yes	No	No	No	No	No
Methontology (Fernández, Gómez-Pérez, & Juristo, 1997)	In States: DA (Specification, Knowledge acquisition)	No	No.	Only Verification	No	No	No	No, just CQ mentioned in Specification	No
Approach by Katsumi & Grüninger (2010)	No	No	Yes, comprehensive tests results for improving axioms	Only Verification	Yes, but only short iterations: 1) Between Requirements-Design-Verification, and 2) Between Verification	No, only ontology versions generated, but no formal releases	No	No	No
Approach by Garcia Castro et al. (2006)	In DA, OV	No	No	Validation & Verification	Yes	No	No	Partially, concerned on CQ elicitation, then CQs are not used in later stages	No
Approach by Lasierra, Alesanco, Guillén, & García (2013)	In DA	No	No	No	No	No	No	Yes	No
Approach by Grüninger & Fox (1995) and by Uschold & Grüninger (1996)	In DA (Motivating scenario and Purpose and Scope (Uschold & Grüninger, 1996))	No	Yes, Completeness Theorems, but no metrics provided	Only Validation	No, none formal guidance is provided (even it is mentioned that the ontology built is iterative)	No	No	Yes	Yes

Knowledge and Diligent allow the user to propose changes to the ontology, therefore they can be viewed as the user to participate in the Ontology Building/validation.

In terms of approaches supporting validation and verification activities, only 3 approaches include both, the rest include one or the other. CODEP is the only process that includes validating the ontology quantitatively using metrics. The validation in CODEP is performed in two phases: one where the ontology model is validated and then where the knowledge base is validated with the users/domain experts.

It can be noticed that there are few approaches where the ontology is driven by CQs (3 approaches and CODEP). From the 3, only 2 support the development of a knowledge base. But it can also be observed that only approaches that are CQ-driven can develop knowledge bases. This demonstrates the importance of these criteria when suggesting processes for creating ontologies that can be applied.

From the evaluation in Table 17, it can also be observed that none of the approaches propose well-structured processes to be followed, with clear activities and roles. This is one of the factors that motivated us to define the CODEP process, as we could not identify any approach from the literature which can be followed.

6.2. CODEP further evaluation plans

CODEP includes 3 main phases, each with several activities. The phases cover the life cycle of ontology development. In addition, the end-user is highly involved in most of the activities. This has been appropriate for developing the CEPRO ontology where the usage of the results of the ontology are critical to the health of humans and ensure that medical doctors (users) trust the quality of the diagnosis produced by the ontology. The effort of following the CODEP activities can be high from the perspective of the knowledge engineer and the users (domain experts). In certain kinds of projects, a trade-off should be considered between applying all CODEP activities and the effort required without influencing the quality of the produced artifacts (ontology, knowledge base, etc.). In some cases, not all activities have to be applied. In our further work, to provide better guidance on the application and adaptation of CODEP, we can design experiments to evaluate which activities can be skipped and which activities must be kept for certain domains or characteristics of end-users.

We also plan on conducting further studies to evaluate other aspects of the performance of the CODEP process and the quality of the ontologies it produces. One of the aspects that would be of interest to study is bottlenecks. Identifying possible bottlenecks, analyze them, and suggest improvements to CODEP activities to handle such issues. One approach we can follow is the process improvement concept (CMMI Institute, 2020), where we can design measurements and adjust CODEP's activities to solve identified process issues. The process improvement can be performed based on quality attributes of the process such as process comprehension (if the process is well-defined from the knowledge engineer's perspective), visibility (if the activities produce clear results), acceptance (if the process is understood by the end-users, knowledge engineers, etc.), support (which activities are performed with tool support), reliability (if the process allows identifying failures which can affect the resulting ontology), maintainability (if the process can easily evolve) and velocity (in the speed of performing activities to complete the knowledge base).

Further studies comparing CODEP to existing ontology development approaches would be of interest. For this objective, we would have to define metrics where we can measure aspects of CODEP and compare them with the results of the other approaches. There are challenges we can encounter in performing this kind of study to make these comparisons meaningful. We have to apply the same metrics used and published by other approaches and develop ontologies of similar domains and complexities.

7. Conclusions

CODEP focuses on driving the ontology design through competency questions, aiming in producing an accurate ontology in terms of compliance with end-user requirements. Metrics are used to validate quantitatively the compliance of the CQs in the ontology. An innovative aspect of CODEP is that it implements incremental and iterative cycles for designing the ontology, in which the V&V tasks are the backbone to indicate if more increments and iterations are needed or not. Specifically, the significance of designing an ontology by adhering to CODEP is as follows:

- **CQs as the main design driver**. CODEP stems from a deep analysis of the domain before starting the ontology modeling, which ends up with the specification of scenarios and the Competency Questions that are targeted to the needs of end-users. Five Activities are considered to widely embrace the ontology domain, PIS, ADA, CQD, KRD, and VCD (Table 1) producing "The Ontology Vision and Scope" as a milestone. The reason is that in CODEP, the CQs become the main driver for further Activities (modeling, designing, and V&V activities). At the end of the CODEP application, it is possible to produce an application-oriented ontology to support the user's needs in practical applications. The proof-of-concept description in Activity 16, shows how the medical team has proved and evaluated the accuracy of the CERPRO ontology, given a high percentage of the validation criteria that they have considered (done after 11 iterations).
- **V&V as a trustable ontology mechanism**. CODEP makes emphasis on V&V: 1) verification for quality assurance of the ontology releases from the engineering perspective, and 2) validation of the ontology model before the end-users, and 3) validation for quality assurance and requirements (as CQs) fulfillment from the end-user perspective. All evaluations are fundamental to ground ontology applications in domains such as medicine, where criteria such as accuracy and comprehensibility of the ontology, knowledge rules, and trustable KB's responses are key to effectively support the case study with humans, and for integration to information systems. Then, CODEP considers Activities VCD, OQID, OMV, OV, OVA entirely dedicated to defining metrics and collecting the measurements, but most importantly, CODEP manages to use such feedback for incorporating changes in the ontology design, in further iterations until fulfilling the expected quality and the end-users requirements (CQs). This is an important contribution since V&V is certainly included in other ontology development processes as commented in the Related Work. However, none of them use the V&V's results to drive changes in ontology modeling and design, quantitatively with the quality and verifiable results, until the end-user requirements are satisfied; nor they indicate specific guidance to perform this activity. The case study described in Section 3, exhaustively uses V&V activities to show how to obtain a trustable ontology release from the medical team's expectations (validation criteria).
- **An iterative life cycle process**. As commented in Section 2, CODEP adapts the incremental construction model from software engineering (IEEE-SWEBOK, 2014), by defining iterations and increments (Fig. 1). In CODEP, each iteration performs Activities 1–16 to produce an ontology increment, ensuring that it is verified and validated by the end-user. Also, CODEP considers small quality cycles, between Activities 10–16, to perform the V&V until the desired quality (according to the quality criteria in Activities 5 and 9) is obtained. These iterations are repeated until end-users requirements (as CQs) are completely and quantitatively satisfied. In this matter, CODEP stems from evolving the ontology before its delivery to the end-users for practical applications. CODEP does not indicate the number of ideal iterations, since it is up to the KE to determine when the V&V criteria are fulfilled.

- **Milestones as guidance for timely deliveries**. CODEP has been specified as guidance for designing ontologies that will be integrated into information systems and which will need to respond to practical use cases. Thus, as presented in Section 2 CODEP clearly and systematically specified phases, activities, milestones, task inputs/outputs (artifacts), intermediate work products, and the involved roles in each activity. In this guidance, a remarkable aspect of CODEP is the definition of milestones, as specific breakpoints, they have been defined to opportunely deliver advances in an ontology and KB with functionalities ready to be used for practical purposes. For example, Milestone II produces the Ontology Beta version. In the proof-of-concept, precisely the Version Beta of CERPRO was used to create the CERPRO-KB which was completely used to test all the CQs-based queries. This was particularly well accepted by the medical team since they could observe real and practical responses to their CQs. From this point, the medical team could identify further use cases for the applicability of the ontology, such as in the automatic identification of patterns in the TBI images. Other processes, which were analyzed in Section Related Work, did not consider milestones or an equivalent concept of releases, so managing deliveries and tests are not systematic.

Some limitations are:

- The CODEP application in the case study shows an example of exhaustive metrics definition and collection since elements are counted from the ontology (for verification measurements), and the KB responses (for validation measurements). However, it distinguishes our work from others as it guides the KE in verifying and validating whether the ontology fulfills the quality standards stated at the beginning of the process. This also supports the delivery of a high-quality ontology from the end-user's perspective, since their feedback determines whether to stop the modeling and design or not based on meeting their expectations. Thus, it can be argued that this consumed time is valuable against the benefits gained.
- If CODEP is used for designing ontologies with a non-practical purpose, some activities do not have to be followed. For example, Activities 13–16, are oriented to validate the ontology for practical usage. Also, Activities 1–5 must be also revised, since the CQs must be adjusted to a more general scope instead of supporting specific purposes of specific end-users. An example of this situation is related to ontologies which are developed for standardizing concepts in an application domain such as FIBO (EDM Council, 2020) for finance, SNOMED (SNOMED International, 2015) for medicine, and FRO-Solvency Ontology (Jayzed Data Models Inc., 2019). These ontologies commonly define the body of knowledge in their domain intending to avoid ambiguities of concepts and they provide a baseline for further operational ontologies (specific to case studies in organizations). However, even in this case, CQs are still the driver for defining the ontology's objectives and goals, and for the ontology development and verification.

Acknowledgements

The authors thank Sc. Dr. Paul Carrillo-Mora and his team from Instituto Nacional de Rehabilitación, México, for the support in the case study and advising about the medical background for designing CERPRO. This work was partially funded by Enterprise Ireland (project MF20180009) through the Career FIT program, which has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No. 713654; The Royal Academy of Engineering (NRCP1516/1/39) and The Royal Society (NI150203) through the Newton Fund scheme.

References

Blomqvist, E. & Öhgren, A. (2006). Constructing an enterprise ontology for an automotive supplier. In *INCOM'2006: 12th IFAC/IFIP/IFORS/IEEE/IMS Symposium* (Vol. 21, pp. 386–397). Saint-Etienne, France: Elsevier.

Clark&Parsia (2011). *Pellet: OWL 2 Reasoner*. (W3 Consortium) Retrieved on November 10, 2020, from https://www.w3.org/2001/sw/wiki/Pellet.

CMMI Institute (2020). *Capability Maturity Model® 2.0.* USA: Sofware Engineering Institute. https://cmmiinstitute.com/cmmi.

EDM Council (2020). *The Open Semantics Standard for the Financial Industry*. Standard, USA. https://spec.edmcouncil.org/fibo/.

Espinoza, A., Abi-Lahoud, E. & Butler, T. (2014). Ontology-driven financial regulatory change management: An iterative development process. In *ENC 2014-SW-LOD Workshop*, Oaxaca, Mexico.

Espinoza, A., Penya, Y., Nieves, J., Ortega, M., Rodríguez, D. & Peña, A. (2013). Supporting business workflows in smart grids: An intelligent nodes-based approach. *IEEE Transactions on Industrial Informatics*, *9*(3), 1384–1397. doi:10.1109/TII.2013.2256792.

Fernández, M., Gómez-Pérez, A. & Juristo, N. (1997). *METHONTOLOGY: From Ontological Art Towards Ontological Engineering*. Madrid, Spain: AAAI Technical Report SS-97-06. Retrieved 2020.

Gangemi, A., Catenacci, C., Ciaramita, M. & Lehmann, J. (2005). A theoretical framework for ontology evaluation and validation. In *SWAP 2005 – Semantic Web Applications and Perspectives, Proceedings of the 2nd Italian Semantic Web Workshop*, Trento, Italy.

Gangemi, A. & Presutti, V. (2009). Ontology design patterns. *Handbook on Ontologies*, 221–243. doi:10.1007/978-3-540-92673-3_10.

Garcia Castro, A., Rocca-Serra, P., Stevens, R., Taylor, C., Nashar, K., Ragan, M. & Sansone, S.-A. (2006). The use of concept maps during knowledge elicitation in ontology development processes – the nutrigenomics use case. *BMC Bioinformatics*, *7*(267).

Grüninger, M. & Fox, M. (1995). Methodology for the Design and Evaluation of Ontologies. *IJCAI'95 – Workshop on Basic Ontological Issues in Knowledge Sharing*, (p. 10).

Hagen, C., Malkmus, D. & Durham, P. (1972). Original Scale. *Communication Disorders Service, Rancho Los Amigos Hospital*. Downey, California.

IEEE-SWEBOK (2014). *SWEBOK 3.0 – Software Engineering Body of Knowledge*. IEEE Computer Society.

INR (2013). (*Spanish*) *Estudio clínico comparativo sobre los efectos del tratamiento adyuvante con Cerebrolisina en una muestra de pacientes con secuelas de traumatismo craneoencefálico sometidos a un programa intrahospitalario de rehabilitación*. Internal, National Rehabilitation Institute (INR), Mexico City.

ISO-25022 (2016). *ISO/IEC 25022 Systems and Software Engineering – Systems and Software Quality Requirements and Evaluation (SQuaRE) – Measurement of quality in use*. Retrieved on November 10, 2020, from https://www.iso.org/standard/35746.html.

ISO-25023 (2016). *ISO/IEC 25023 Software product Quality Requirements and Evaluation (SQuaRE) Quality model – Measurement of system and software product quality*. ISO.org. Retrieved on November 10, 2020, from https://www.iso.org/obp/ui/#iso:std:iso-iec:25023:ed-1:v1:en.

Jayzed Data Models Inc. (2019). *Solvency Ontology*. USA. https://insuranceontology.com/.

Jennett, B. & Bond, M. (1975). Assessment of outcome after severe brain damage. *The Lancet*, *1*(7905), 480–484. doi:10.1016/S0140-6736(75)92830-5.

Katsumi, M. & Grüninger, M. (2010). Automated reasoning support for ontology development. In *International Joint Conference on Knowledge Discovery, Knowledge Engineering, and Knowledge Management.* (Vol. 272, pp. 208–225). Berlin, Heidelberg: Springer. doi:10.1007/978-3-642-29764-9_15.

Lasierra, N., Alesanco, A., Guillén, S. & García, J. (2013). A three stage ontology-driven solution to provide personalized care to chronic patients at home. *Journal of Biomedical Informatics*, *46*(3), 516–529. doi:10.1016/j.jbi.2013.03.006.

Malheiros, Y. & Freitas, F. (2013). A method to develop description logic ontologies iteratively based on competency questions: An implementation. *ONTOBRAS 2013*, *1041*, 142–153.

Napel, H.T., Rogers, J. & Zanstra, P. (1999). D06.1 Demonstration of the Telematic Infrastructure for a common resource for medical terminology and language for Europe including the designated segments of national classifications. Retrieved on November 10, 2019, from https://www.opengalen.org/download/dissections/D6_1_6.pdf.

Nieves, J., Espinoza, A., Penya, Y., Ortega De Mues, M. & Peña, A. (2013). Intelligence distribution for data processing in smart grids: A semantic approach. *Engineering Applications of Artificial Intelligence*, *26*(8), 1841–1853. doi:10.1016/j.engappai.2013.03.016.

Noy, N. & Mcguinness, D. (2001). *Ontology Development 101: A Guide to Creating Your First Ontology*. California, USA: Stanford. Stanford University.

OpenGalen Foundation (2012). *OpenGalen Ontology*. (OpenGALEN.org) Retrieved on July 07, 2019, from https://www.opengalen.org/.

Pazienza, M.T. & Armando, S. (2012). *Semi-Automatic Ontology Development: Processes and Resources*. USA: Information Science Reference, IGI Global.

Pinto, H., Staab, S. & Tempich, C. (2004). DILIGENT: Towards a fine-grained methodology for DIstributed, loosely-controlled and evolvInG engineering of oNTologies. In *Proceedings of the 16th European Conference on Artificial Intelligence*, Valencia, Spain.

Pisanelli, D., Gangemi, A. & Steve, G. (2003). Ontologies and Information Systems: The Marriage of the Century? *Journal of Information Systems and Technology Management.*

Poveda-Villalón, M., Suárez-Figueroa, M.C., García-Delgado, M.Á. & Gómez-Pérez, A. (2009). *OOPS! (OntOlogy Pitfall Scanner!): Supporting Ontology Evaluation on-Line*. Madrid, Spain: IOS Press.

Protégé (2020). *Protégé Stanford University*. (Stanford University School of Medicine) Retrieved on November 10, 2020, from Protégé: http://protege.stanford.edu/.

Ren, Y., Parvizi, A., Mellish, C., Pan, J., Deemter, K. & Stevens, R. (2014). Towards competency question-driven ontology authoring. In *European Semantic Web Conference (ESWC 2014) – the Semantic Web: Trends and Challenges*, Crete, Greece: Springer Berlin, Heidelberg (pp. 752–767).

RuleML Inc. (2020). *RuleML*. (RuleML Inc.) Retrieved on November 10, 2020, from http://wiki.ruleml.org/.

Sandia National Laboratories (2020). *Jess – The Rule Engine for the Java Platform*. (Sandia National Laboratories) Retrieved on November 10, 2020, from https://jess.sandia.gov/.

SNOMED International (2015). *Systematized Nomenclature of Medicine – Clinical Terms (SNOMED CT)*. Retrieved on July 07, 2019, from http://snomed.org.

Sousa, C., Soares, A., Pereira, C. & Moniz, S. (2014). Establishing conceptual commitments in the development of ontologies through competency questions and conceptual graphs. In *OTM 2014: On the Move to Meaningful Internet Systems: OTM 2014 Workshops*. LNCS (Vol. 8842, pp. 626–635). Berlin, Heidelberg: Springer.

SparxSystems (2013). *Effective Ontology Development using the UML and Enterprise Architect*. Retrieved on November 10, 2019, from https://sparxsystems.com/enterprise_architect_user_guide/15.2/model_domains/mdg_technology_for_odm.html.

Staab, S., Studer, R., Schnurr, H.-P. & Sure-Vetter, Y. (2001). Knowledge processes and ontologies. *IEEE Intelligent Systems*, *16*(1), 26–34. doi:10.1109/5254.912382.

Suárez-Figueroa, M.C. (2010). *NeOn Methodology for building ontology networks: specification, scheduling and reuse* (*Doctoral dissertation*). Informatics Department, Universidad Politécnica de Madrid.

Suárez-Figueroa, M.C., Aguado de Cea, G., Buil, C., Caracciolo, C., Dzbor, M., Gómez-Pérez, A. & Presutti, V. (2007). *D5.3.1 NeOn Development Process and Ontology Life Cycle*. NeOn-project.org, Retrieved 2020.

Tartir, S., Budak Arpinar, I., Moore, M., Sheth, A. & Aleman-Meza, B. (2005). OntoQA: Metric-based ontology quality analysis. In *IEEE ICDM Workshop on Knowledge Acquisition from Distributed, Autonomous, Semantically Heterogeneous Data and Knowledge Sources*, Houston, Texas.

The Apache Software Foundation (2020). *Apache Jena*. (Apache Project) Retrieved on November 10, 2020, from https://jena.apache.org/.

Uschold, M. & Grüninger, M. (1996). Ontologies: Principles, methods and applications. *The Knowledge Engineering Review*, *11*(2), 93–136. doi:10.1017/S0269888900007797.

W3C-OWL 2 (2012). *OWL 2 Web Ontology Language*. (W3C Consortium) Retrieved on November 10, 2020, from https://www.w3.org/TR/owl2-overview/.

W3C-SPARQL (2008). *SPARQL Query Language for RDF*. (W3C Consortium) Retrieved on November 10, 2020, from https://www.w3.org/TR/rdf-sparql-query/.

W3C-SWRL (2004). *Semantic Web Rule Language* (*SWRL*). (W3C Consortium) Retrieved on November 10, 2020, from https://www.w3.org/Submission/2004/03/.

yWorks Software (2020). *yWorks*. (yWorks Company) Retrieved on November 10, 2019, from https://www.yworks.com/products/yed.