ARTICLE IN PRESS

Web Semantics: Science, Services and Agents on the World Wide Web xxx (xxxx) xxx

Contents lists available at ScienceDirect

Web Semantics: Science, Services and Agents on the World Wide Web

journal homepage: www.elsevier.com/locate/websem

Grounding knowledge acquisition with ontology explanation: A case study

Ana Cristina B. Garcia [a,*], Adriana S. Vivacqua [b]

[a] Universidade Federal do Estado do Rio de Janeiro, Brazil
[b] Universidade Federal do Rio de Janeiro, Brazil

ARTICLE INFO

Article history:
Received 18 December 2017
Received in revised form 8 December 2018
Accepted 10 December 2018
Available online xxxx

Keywords:
Explanation
Knowledge acquisition
Ontology validity
Ontology engineering

ABSTRACT

Knowledge validation is still a challenge when constructing knowledge-based systems. It is one of the major reasons for user rejection and disagreement between project participants. Systematic and periodic reviews of the domain ontology, with a formal agreement of the whole development team (including the experts) are a recommended good practice. Nevertheless, these reviews do not guarantee system success. This paper presents a case study of the construction process of a knowledge-based system. The process involved a group of experts with varied work experience. A great deal of negotiation happened during knowledge acquisition meetings, which took place during a 6-month period. After each meeting, changes in the ontology were verified through a web-based questionnaire, from which either consensual agreement was reached (and changes implemented) or the need for a new meeting was ascertained. An explanatory review at the beginning of each meeting further solidified the understanding of all participants. This cyclic process led to a final version of the ontology, ratified by all participants. This model supports diagnosis and prediction of failures in mechanical drilling rigs in oil exploration sites. Unexpectedly, during system trials, experts disagreed with results, which raised questions about the validity of the domain ontology. The system's explanation module provided a cornerstone for a reflective process that helped identify inconsistencies and corrections needed. These reflections led to adjustments to the ontology, and a reflection about previous decisions and element definitions. Explanations, derived from the ontology and instantiated using real scenarios, shed light on knowledge gaps and semantic inconsistencies of the domain model. In this paper we have three main goals: (1) to present our ontology construction process; (2) to highlight a particular situation where results were inadequate; and (3) to show how the explanation system helped experts and knowledge engineers identify gaps. We also present lessons learned from the whole process, that may apply in other situations.

1. Introduction

Artificial Intelligence (AI) is gaining popularity as a means to address multiple problems, and expectations are high. Easy Internet access, cheap sensors and an abundance of open information have created a rich information environment that businesses can take advantage of. However, it is still a challenge for humans to manipulate such large amounts of data. Machine learning techniques have become popular in classification domains in which labeled cases are available [1]. Knowledge-based systems (KBS) are based on human expertise, and offer a succinct way to cover a domain. In KBS, the expert knowledge is usually available for review [2], so that outcomes can be inspected.

At the core of many KBS is an ontology describing expert knowledge of a domain, which can be manipulated by an inference engine. The ontology is a specification of the relevant concepts and relationships between them to support predefined tasks [3]. The knowledge acquisition (KA) process is a fundamental step in this scenario. Ontology construction may be a top-down or bottom-up process: the first involves acquiring tacit knowledge from experts' explanations [4,5], and the second involves acquiring knowledge by investigating specific cases [6]. The choice of process depends on the availability of the domain knowledge and experts. The top down approach fits well when there are available experts with deep knowledge of the domain. This approach has been used in many engineering domains, with successful results, particularly for diagnostic tasks [7–10]. The bottom up approach works better when there are multiple concrete cases available, from which knowledge can be extracted and generalized.

Although building an ontology is sometimes equated to an art, there are well-accepted methods to guide the process, such as Methontology [11], NeOn [12] and Xtreme Collaboration [13]. Methontology defines the ontology construction process as a procedure with seven steps: specification, knowledge acquisition,

* Corresponding author.
E-mail addresses: cristina.bicharra@uniriotec.br (A.C.B. Garcia), avivacqua@dcc.ufrj.br (A.S. Vivacqua).

https://doi.org/10.1016/j.websem.2018.12.005
1570-8268/© 2019 Elsevier B.V. All rights reserved.

conceptualization, integration, implementation, evaluation and documentation [11]. NeOn focuses on ontology reuse, so finding and adjusting an existing ontology are the main challenges [12]. Xtreme Collaboration (XC) does not prescribe steps, but goals that must be achieved by a group of experts during face-to-face meetings [13], where experts are not allowed to leave until the goals have been reached. XC tends to speed up the process, as participants actively try to reach consensus within the given time frame. In XC, it is important that participants be committed to the task, and accountable for the results [13,14], so that they engage in the process in a constructive way (as opposed to accepting any solution). We combined the Methontology process with XC techniques to increase expert engagement and commitment to the KBS, without losing the robustness provided by Methontology. These techniques have been successfully applied to multiple domain areas, but not combined before.

This paper describes the ontology construction process used to build a domain ontology for a KBS to diagnose failures in oil drilling rigs. We had successfully used this process before, but this time, when the system was deployed, feedback about system outcomes indicated that the ontology needed further revision. This led not only to a revision of the ontology, but also to a reflection on the process. An ontology-based explanation feature embedded in the KBS helped us trace the problem back to KA sessions, from which we extracted three take-away lessons: (1) using toy problems as examples masks conflicts among experts, (2) avoiding small conflicts during KA leads to larger problems at a later stage and (3) ontology explanation can be a valuable tool for semantic validation. In this paper, we present ontology construction process, highlighting the issues found with the final version and how the explanation system helped identify knowledge gaps. The next section describes the domain, the ontology construction method and the final system. Section 3 presents our case study in detail, explaining the construction of an ontology to be used for diagnosis of failures in oil drilling rigs. A discussion of the case and lessons learned follows in Section 4, with our reflection on the inadequate results and their causes. In Section 5 present related work, and the conclusion and future work are presented in Section 6.

2. Domain and ontology construction method

In this section we briefly present the domain involved, the method adopted for the construction of this ontology and the KBS itself.

2.1. Problem domain: diagnosing failures in oil drilling rigs

Onshore oil production involves drilling deep holes in oil reservoir areas located inland. This process requires an artificial lifting method to extract oil from the reservoir and bring it to the surface. When compared to offshore drilling, this is a simple process that involves a set of about 100 pieces of equipment (schematically illustrated in Fig. 1). These are organized into systems, such as: surface system, sub-surfacing system, tubing components, sucker rod system and casting. For example, the subsurface system is composed by a plunger, pump barrel, gas anchor, traveling valve and standing valve.

The oil drilling rig is a mechanical system composed of a set of equipment organized to achieve the overall function of lifting oil from the underground reservoir to the surface. An ontology representing this system is shown in Fig. 2. The ontology was encoded in OWL[1] using WebProtege,[2] as shown in Fig. 3. In this model, each component is a piece of equipment with a form, performing a behavior to achieve a function. The sucker rod is a cable (**form**), which is connected to the pumping unit. As the beam pumping system rocks back and forth (**behavior**), it triggers (**function**) the rod string, the sucker rod and the sucker rod pump.

Each equipment is subject to environmental and operational conditions, such as: local weather, reservoir corrosion rate and operators' skills. The performance of each equipment affects the overall performance of the oil drilling rig system. In case of anomalous operating conditions or inadequate oil well production performance, the reservoir manager will be contacted to decide whether production should be shutdown. Whenever a failure occurs in an underground component (sub-surface equipment, tubing, sucker rod), the entire system must be taken apart, so that all components can be sent to the mechanical workshop for verification. This means oil production ceases until the equipment is replaced. Thus, maintenance means ceasing or reducing oil production, which causes economic losses and may become a business problem.

Corrective maintenance happens only when a piece of equipment breaks down unexpectedly. This is the worst possible situation, because maintenance teams may not be available at short notice, or may not have the components required for replacement. This usually leads to long periods of production stoppage. To avoid this situation, preventive maintenance is periodically scheduled. *Preventive* maintenance involves stopping the whole production site based on a predefined schedule, even when there is no indication of failure. This means that equipment may be stopped unnecessarily (nothing to fix/tune), wasting time and money. In addition, maintenance plans are generally formulated based on manufacturer schedules for inspection and substitution of new equipment. However, refurbished equipment is frequently used as a replacement for new ones, and may require different maintenance plans. *Predictive* maintenance tries to balance the advantages and disadvantages of both preventive and corrective maintenance. It involves trying to predict when the equipment will fail and scheduling preventive maintenance before it does. This is a probabilistic approach, and usually more cost effective, but it requires enough understanding of the process for accurate predictions to be made.

Brazilian onshore oil production is small: up to 5K Barrels per day (BPD) per drilling rig (as a point of comparison, offshore production may reach up to 100K BPD per well). For this reason, maintenance of onshore oil wells is not a high priority for Brazilian oil companies. In addition, there are about 5000 onshore drilling rigs, spread over a very large geographical area (thousands of kilometers), and these are serviced by a single onshore maintenance division. Thus, maintenance logistics is a complex problem.

The maintenance crew is small and must travel around the country carrying specialized equipment (to service the rig) and replacement equipment (to substitute faulty parts). Once local repairs are done, the crew must bring back the broken parts to the mechanical shop. All of this back and forth is a costly process for the organization. Drilling equipment is highly affected by environmental conditions, such as reservoir toxicity, soil type, weather conditions and location. When a piece of equipment fails it is always substituted for another one while the faulty equipment is taken away for maintenance. This is done to reduce cost and production downtime.

It is important to notice that whenever an intervention occurs, all of the parts removed from the well must be inspected, tested and diagnosed for repair (if necessary), or documented regarding usage and storage. Interventions involve taking the entire well apart and putting it back together, and exchanging faulty components in the process. As mentioned before, replacement components may not be new ones, they may be refurbished items. This means that the replacement parts' condition should be able

[1] Available at https://github.com/avivacqua/BM-Ontology or at https://webprotege.stanford.edu/#projects/6d44e83e-029e-4bda-8a0c-44d31dc62f4c/edit/Classes?selection=Class(owl:Thing) (WebProtege login required).

[2] https://webprotege.stanford.edu/.

Please cite this article as: A.C.B. Garcia and A.S. Vivacqua, Grounding knowledge acquisition with ontology explanation: A case study, Web Semantics: Science, Services and Agents on the World Wide Web (2019), https://doi.org/10.1016/j.websem.2018.12.005.

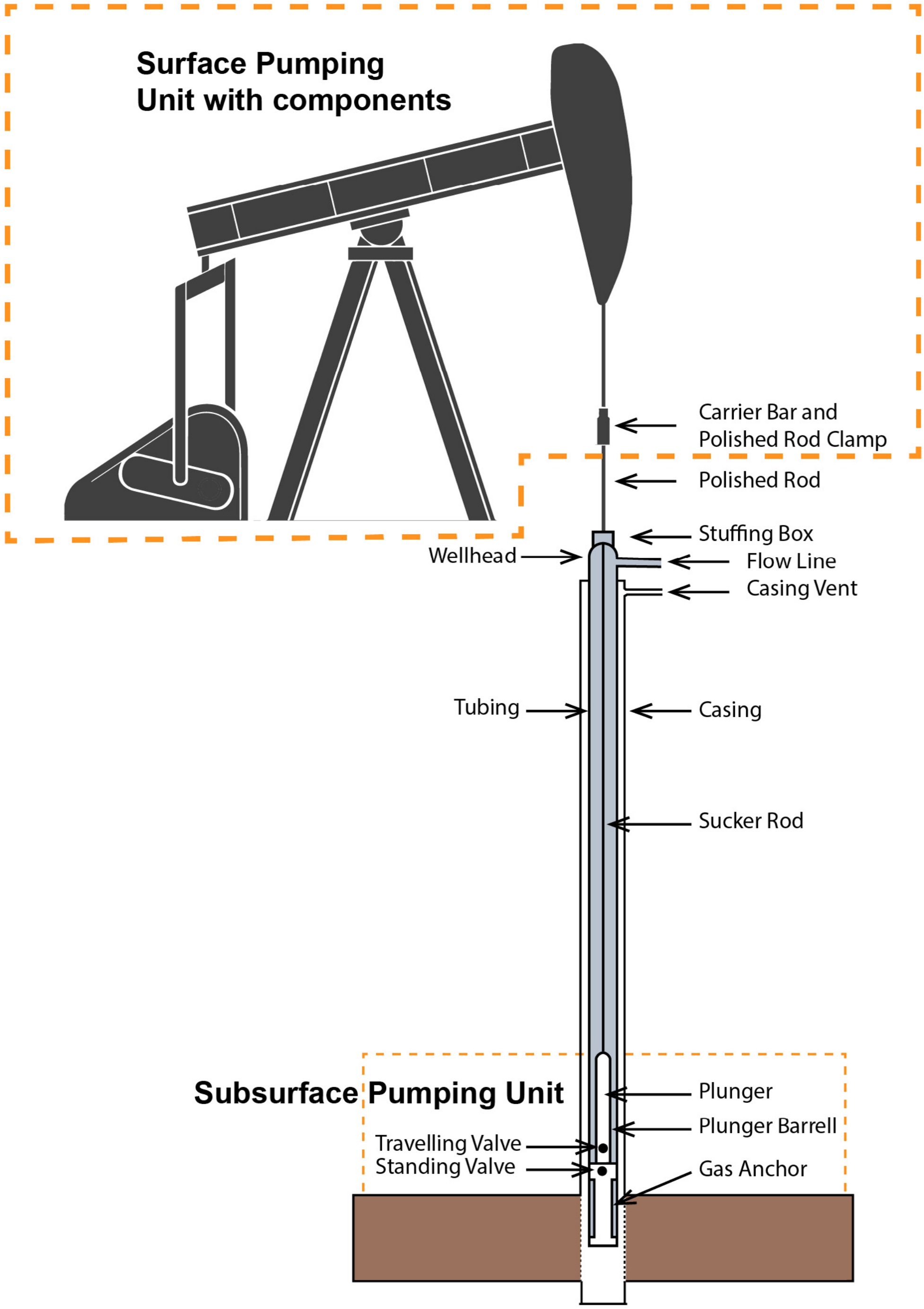

Fig. 1. Onshore oil production equipment main components.

to withstand the site's corrosion rates. These conditions are inferred by observing, over time, equipment degradation between maintenance cycles. During each cycle, inspection procedures and tests are performed on the equipment component to characterize the failure. Based on international standard ISO 14224 and expert experience with the diagnostic process, the following elements characterize each failure:

- data: data describing the event (failure);
- descriptor: evidence of equipment failure;
- mechanism: physical, chemical or other process leading to a failure;
- mode: observable consequences of the failure;
- root cause: circumstances associated with design, manufacture, installation, operation and maintenance, probably leading to a failure;
- agent: agent responsible for the failure.

Failure descriptor, mechanism and mode are used as evidence to determine an equipment's usage status and to determine root causes and the underlying cause agent for a failure. Depending on the component, there will be a set of possible causal agents and expected causes. Experts use a combination of widely accepted diagnostic method such as Failure Modes and Effects Analysis (FMEA) [15] and Fault Tree Analysis (FTA) [16] to build a root cause tree for each oil well rig failure.

Please cite this article as: A.C.B. Garcia and A.S. Vivacqua, Grounding knowledge acquisition with ontology explanation: A case study, Web Semantics: Science, Services and Agents on the World Wide Web (2019), https://doi.org/10.1016/j.websem.2018.12.005.

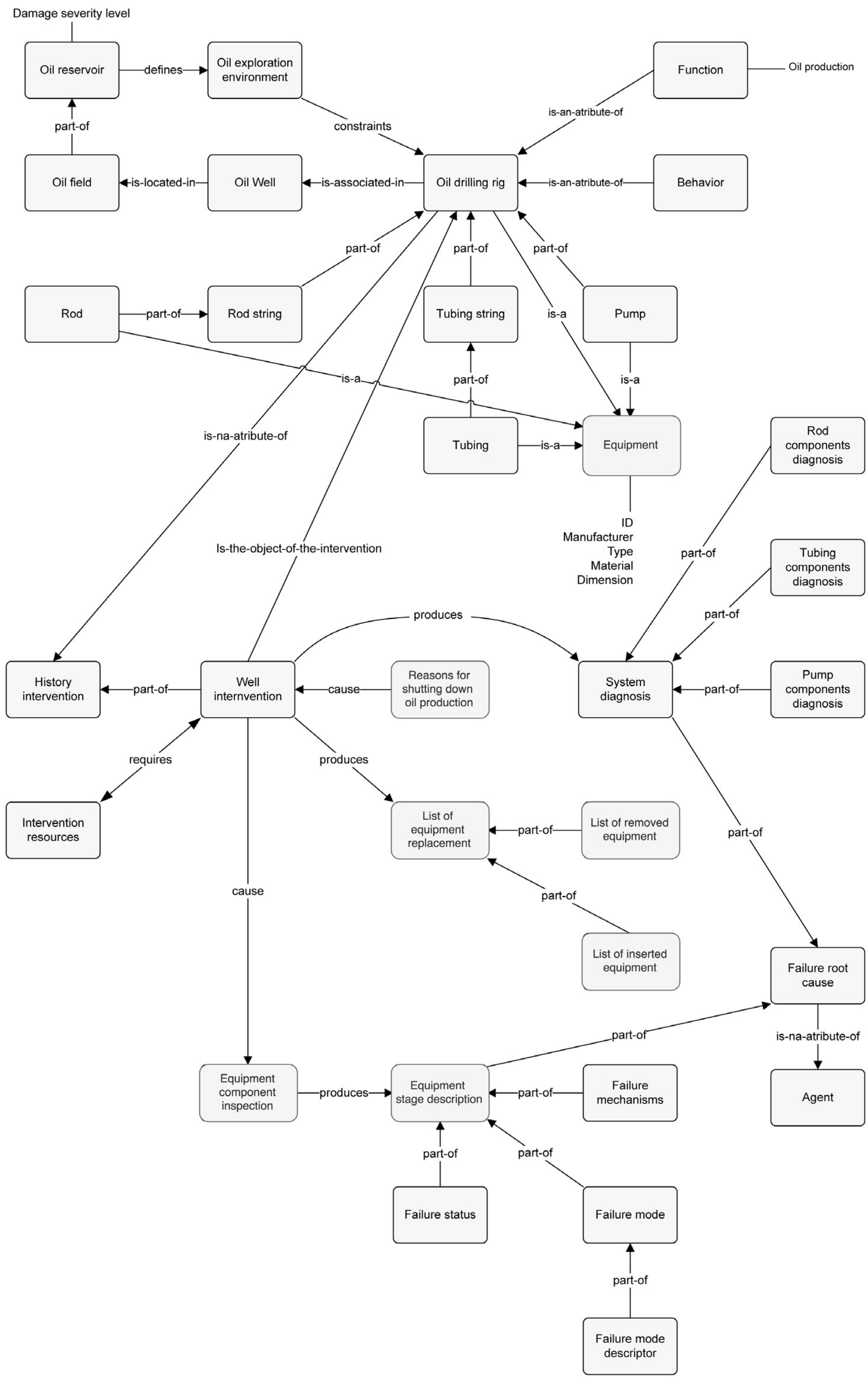

Fig. 2. A partial view of the equipment ontology with attributes, redrawn for clarity.

For example, a production loss in an oil well occurs whenever a leak on the Column String or on the Subsurface Pump is detected. A leak on the column string means there is a hole in the Tubing, which is an expected consequence environmental corrosion of the reservoir. On the other hand, a leak on the pump reflects a problem in the valve ball, which is generally a manufacturing problem, caused by the equipment producer (manufacturer). Fig. 4 illustrates this example.

Diagnosing and predicting a failure requires expertise and a broad knowledge of the domain. In addition, selecting the equipment to be swapped in during maintenance is a complex issue, which can be modeled as a multi-objective optimization problem [17–19]. Multiple factors need to be taken into account, such as reservoir corrosion levels, operation complexity and maintenance schedule. The optimal solution will be to select refurbished equipment that can resist corrosion for the longest time, and that is easiest to replace, considering the available maintenance crew

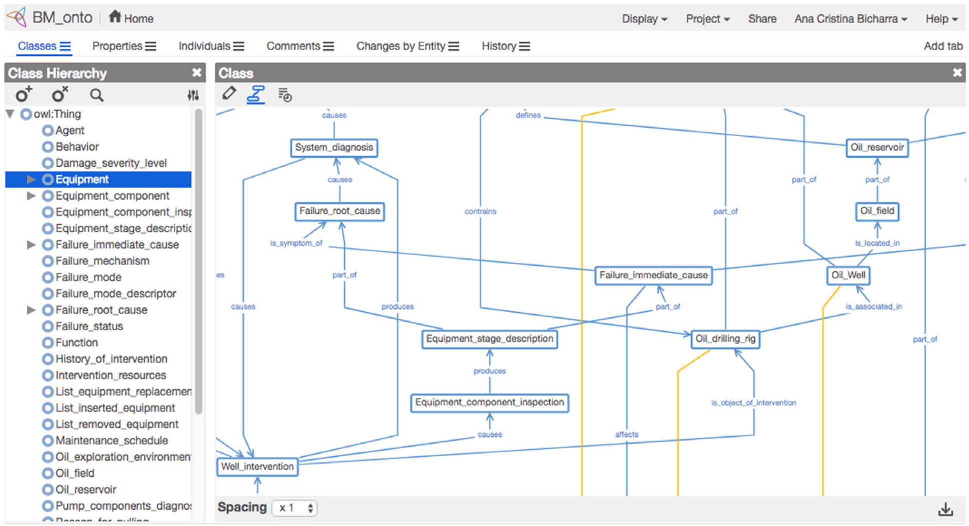

Fig. 3. The ontology in OWL can be found at.

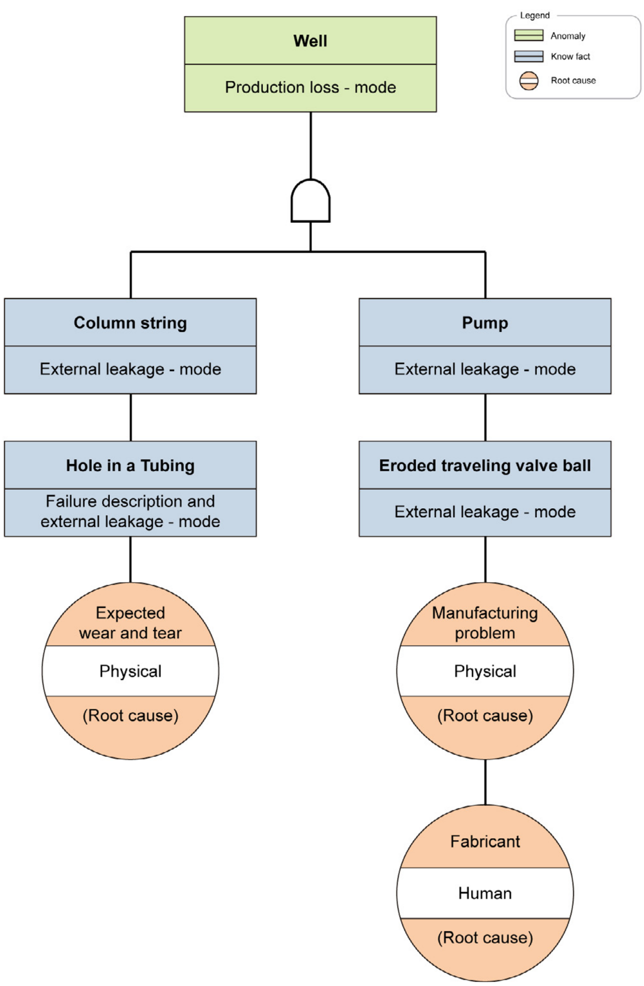

Fig. 4. A partial view of a resulting root cause analysis.

travel schedule and distance. To handle this problem, a KBS was introduced, which had an ontology at its core.

Domain: Onshore Oil Drilling
Date: 03/15/2012
Conceptualized-by: Ana Cristina B Garcia
Implemented-by: Adriana Vivacqua

Purpose: Ontology about the equipment involved in the onshore oil drilling process, their multiple parts and possible failures, to be used in a KBS to diagnose pump failures

Level of Formality: semi-formal

Scope: List of roughy 400 concepts involved in the process and their instances and an average of 10 values for each.

Sources of Knowledge: Expert knowledge,Organizational records

Fig. 5. Requirements Specification.

2.2. Construction process: Methontology and Xtreme collaboration

The ontology construction process adopted was a combination of the Methontology [11] method with Xtreme Collaboration dynamics [20]. We had successfully used this combination for ontology construction in previous projects, so it made sense to use it again.

Xtreme collaboration [20] is a method for knowledge acquisition that brings together a group of people with a common goal, well defined and strict time limits to achieve it. Being physically collocated forces people to concentrate on the task at hand and simplifies knowledge sharing and group awareness. In meetings, experts become aware of the multiple points of view, which leads them to think about the different issues and try to reach an agreement while constructing the knowledge model. The environment is set up so that experts have access to domain-related material they can refer to if needed.

Methontology [11] is an established methodology for ontology construction, composed of the following steps (these do not have to be executed in this particular order, but should be part of the process):

- Specification, where requirements indicating the purpose and scope of the ontology are explicitly documented;

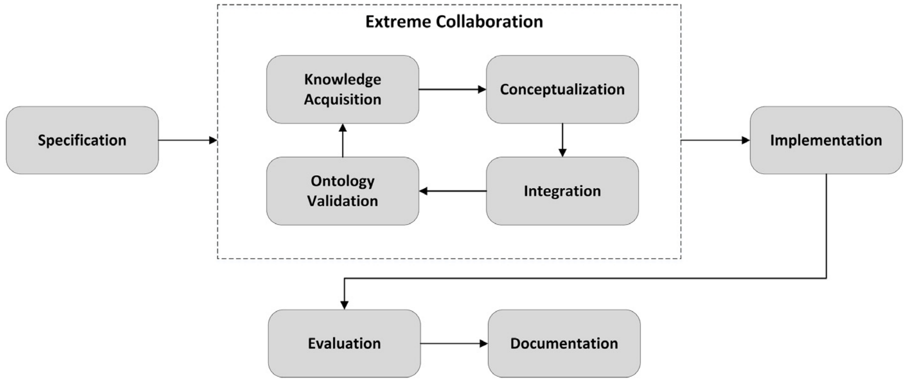

Fig. 6. Ontology Construction process: KA, Conceptualization and Integration steps were conducted using Xtreme Collaboration dynamics. An additional step of ontology Validation was conducted after Integration (still in Xtreme Collaboration conditions), so that the conceptual model was verified by the experts before implementation. The Xtreme Collaboration loop continued until ontology creation was finalized.

- Knowledge Acquisition, where expert knowledge is captured and organized, usually through meetings and questionnaires;
- Conceptualization, where domain elements are structured into a conceptual model;
- Integration, where information from other ontologies is tied to the representation under construction;
- Implementation, where the ontology is encoded using a standard, formal language, such as Prolog or OWL;
- Evaluation, where the ontology and its use are evaluated; and
- Documentation, which is elaborated as the ontology is built.

In our case, we introduce a *Validation* step after Integration, prior to Implementation, Evaluation and Documentation. This is to ensure that the conceptual model is correct and accepted by experts before investing time and effort with implementation. Specification was the first step in our project, conducted during an initial meeting between the main client, ontology engineers and participating experts to agree on project goals and define what should be built. An example of the requirements document can be seen in Fig. 5. This document would drive the remainder of the process.

Knowledge Acquisition, Conceptualization and Integration, along with the added Validation phase, were conducted under Xtreme Collaboration conditions: the group worked together in facilitated face-to-face meetings that eventually led to agreement regarding the concepts and relationships in the ontology. These activities were repeated until the ontology construction came to an end. In between meetings, KEs would search for applicable ontologies and bring them to the meeting for discussion and possible integration (Integration step). KEs also revised and consolidated the model, generated formal questionnaires for validation and prepared material for the next meeting. Sometimes, KEs would ask experts to bring more information or example cases to the next meeting (these tasks were referred to as *homework*). The beginning of each meeting consisted of a group review of the current ontology (Validation step), to ensure the model was correct before moving further.

Implementation and Evaluation were conducted by KEs and an implementation team, after the meeting cycle had finished. The full system was then evaluated by the experts (who were also the intended users), and final documentation was generated (Documentation step). It should be noted that part of the documentation was generated in between meetings, as KEs revised and consolidated the ontology for review at the next meeting. This sped up the Documentation step later on, as much of the ontology had already been previously summarized. Fig. 6 shows the ontology construction process.

2.3. A KBS to diagnose failures in onshore oil rigs

MechDrill (MD) is a project to build a KBS to diagnose oil drilling rig failures. MD-KBS, illustrated in Fig. 8, was designed to help petroleum engineers (the users) diagnose problems based on sensor data and symptoms observed during operation. Sensor readings are continuously recorded, and the onshore oil drilling rig site description is input at the beginning of the operation or when changes happen. On site operators report anomalous equipment behavior, and site supervisors assess the status of all onshore oil drilling rigs as part of their weekly routine.

We implemented the MechDrill knowledge-based system (BM system in Portuguese [21]) in JAVA, and knowledge was represented using Drools (Java Rule Engine). The knowledge base is composed of 157 inference rules. Each rule is represented as Condition–Action pairs, encoded in XML, such as the one shown in Fig. 7. The system was implemented, validated and deployed. It has been successfully used in the Brazilian oil company in real situations.

Diagnosis is an on-demand activity: when requested, the system combines data from multiple sources, applies the heuristics, searches for similar cases and infers the status of every equipment and system at all sites. In case of failure, the system also engages in root cause analysis to infer the causes. In addition, the system designs a maintenance plan taking into account the maintenance crew's routes, availability of replacement components and number of acceptable refurbished parts.

The domain ontology is a core element of the MD-KBS: it guides computational reasoning and provides the basis for creating explanations for the answers. The ontology covers artifact description, including the oil drilling rig system and its subsystems (e.g., the hydraulic system). It defines not only the different dimensions of the problem and their correlations, but also the range of possible values for each variable. For instance, each *Intervention* has a number of associated variables (or attributes), which are the dimensions for analysis. These attributes include, among others: equipment, cause of failure, type of failure, failure mechanism, failure mode and resources needed for intervention. Each of these attributes has a fixed set of possible categorical values specified in the ontology. The oil rig failure diagnosis ontology was created with three goals in mind:

- Creating common ground among experts: although the group of experts was from the same company, they came from different locations, followed distinct schools of reasoning and had different experiences. Frequently, they adopted different strategies to diagnose problems. The process of building the ontology helped build a shared knowledge base containing the aggregated know-how of the organization;

Table 1
Participants profile.

Id	Role	Place	Background	Experience
1	Expert Coord.	RJ	Petrol Eng.	Average
2	Expert	ES	Petrol Eng.	High
3	Expert	BA	Petrol Eng.	High
4	Expert	AL	Petrol Eng.	High
5	Expert	SE	Petrol Eng.	High
6	Expert	SE	Petrol Eng.	High
7	Expert	RN	Petrol Eng.	High
8	KA Modeler	RJ	KE	None
9	KA Coord.	RJ	KE	None

```
< pump unit >
  < rule name = "Operational Root Cause" >
    < pump_unit_status:condition >
      < pump_unit:inlet_power failure_mechanism = "corrosion" >
      < pump_unit:status == "internal leakage" >
    < pump_unit_status >
    < pump_unit:inlet_power >
      < pump_unit_status:condition >
      < pump_unit_status:actions >
          < pump_unit_status:failure_status = "critical failure" >
      < pump_unit_status:actions >
  < rule >
```

Fig. 7. Sample diagnosis rules.

- Representing the diagnosis rules to be used by the KBS system; and
- Creating a script for automatic explanation generation (to explain KBS results).

In the next section, we describe the ontology creation process in detail.

3. Building an ontology for oil rig failure diagnosis

In this section we present the process of constructing the MechDrill ontology. We begin with a description of the participants, followed by a description of the process and the ontology.

3.1. Participants

Our study involved a group of seven experts from the same company, presented in Table 1. They have similar backgrounds, but work at different sites, located in different Brazilian states. Their local environments are very different regarding weather, number of employees and size of the drilling site. However, they share the same maintenance problems and maintenance crew services. They are all very experienced petroleum engineers and, except for an Expert Coordinator from the central office, all played the same role of Expert in the knowledge acquisition process. The meetings were all face-to-face meetings. Participants' workplaces were very distant from each other, so they had to travel to attend meetings. They were officially assigned by the organization to participate in the knowledge acquisition sessions.

3.2. The knowledge acquisition process

Prior to the start of the process, the expert coordinator provided extensive material and bibliography to get the knowledge engineers (KEs) acquainted with the domain. The group of experts did not receive any training or previous information concerning ontology or its construction. The knowledge acquisition sessions, conducted over a period of 6 months, were necessary to understand the domain and build the domain ontology. Each session took about 4 h, although the first and the last ones took the full day. At the beginning of each session, the goal for that meeting was clearly defined, in accordance with XC dynamics.

The knowledge acquisition process, illustrated in Fig. 9, involves two roles: the KE modelers and the experts. At the beginning of the project, during Specification phase, KE modelers gathered from the experts:

- domain related questions the final ontology should be able to address, such as: *What is the difference between an accident and a state of non-conformity?* and
- a set of relevant cases that should be covered by the ontology, which were retrieved from the company databases.

This material was used to guide the following KA meetings and adjust the ontology at each iteration. After the specification had been defined, the process entered an acquisition–conceptualization–integration–validation cycle, which reached an end only when the ontology was completely defined and ratified by experts. The cycle involved XC meetings, plus pre- and post-meeting steps. These steps are necessary to consolidate the information acquired during the meeting and to prepare for the next meeting. At the very beginning of the cycle, KEs checked whether there were any ontologies to be integrated to the project. None was found, so the integration phase was skipped (had there been one, its concepts would have been brought for in-meeting discussion).

3.2.1. Pre-meeting

Prior to each meeting, KEs prepared a presentation of the current stage of the ontology, so experts could review the work as it progressed. KEs also formulated an agenda and defined the semi-structured guidance necessary to run the KA sessions (questions and cases). They prepared questions to tease out information from experts, and searched for cases to help illustrate and ground discussion.

In addition, each participant was usually asked to bring one or more case studies to help explain the concepts to be covered in the meeting. They stated they selected the cases that would best clarify their explanations at later interviews.

3.2.2. In-meeting

The first session followed an open brainstorm style, so that the initial concepts could be defined, at a high level of abstraction. These concepts then formed the basis for the organization of the following meetings, subdividing the ontology into topics and working on detailing these different sections, one at a time. The remaining sessions were semi-structured, and started with a review of the current ontology and validation of the work done in previous sessions (which usually took about 30 min). This was followed by a guided discussion on the topic of the day.

The face-to-face group meetings started with the KE presenting the existing ontology using a standard script to explain its elements, and then define the goals of the meeting (this was usually to work on the definitions of a section of the ontology). During the explanation, experts pointed out errors or misunderstandings in the ontology that needed to be fixed, and further discussed any outstanding points.

Questions prepared by KEs started the discussion among the experts, who were steered towards agreement by KEs. Periodically, KEs summarized the decisions and revised the goals for the meeting, so that the group stayed on track. Fig. 10 shows one of the sessions with its participants in action. Participants used slides, drawings, writings and exercises to explain their points of view and discuss concepts. Experts discussed the concepts using examples and counterexamples within the meeting time frame. On occasion, experts would take some time to search for material to better explain their points of view during the meeting.

Please cite this article as: A.C.B. Garcia and A.S. Vivacqua, Grounding knowledge acquisition with ontology explanation: A case study, Web Semantics: Science, Services and Agents on the World Wide Web (2019), https://doi.org/10.1016/j.websem.2018.12.005.

During discussion, if no conflict was detected, the concept was considered validated. When conflicts happened, experts were given some time to discuss and resolve them during the meeting. In the case of conflicting deadlocks, KEs would stop the discussion and ask participants to bring extra material for discussion at the next session. This material should provide support for experts' arguments and/or example cases to illustrate the concepts under discussion. All sessions were videotaped to help KEs encode the concepts and update the domain ontology.

The final part of the meeting (about 10 min) involved a quick summary of the work done during the meeting; verifying if the intended goals had been met; scheduling and setting the discussion points for the next meeting; and defining activities that experts had to do in between meetings (*homework*). These usually involved searching for information and/or real cases to illustrate a particular conflicting or questionable element.

3.2.3. Post-meeting

After each meeting, KEs consolidated and organized the material not only to prepare for the next meeting, but also to prepare a validation questionnaire. This questionnaire was sent to experts so they could answer between meetings. There were questions about the representation (the ontology itself), the perceived degree of participant conflict and the depth of meeting discussion. The partial ontology (outcome of the meeting) was sent to the team to get official approval from each of the participants. Each participant had to state his/her level of agreement with the elements presented, and this was considered an official document for which participants could be held accountable. If they were not in agreement, they could request that further discussion be conducted about some of the elements. Answers to the questionnaire helped organize the agenda for the next meeting.

It should be noted that this post-meeting ontology consolidation greatly helped the documentation phase: parts of the ontology

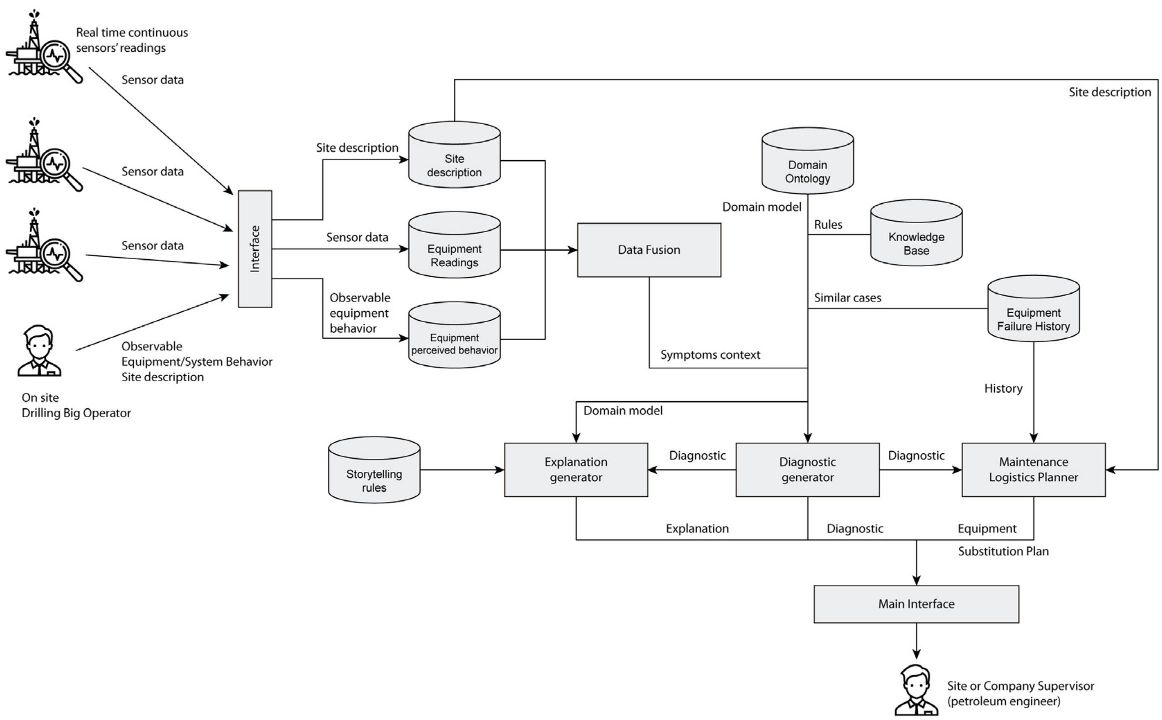

Fig. 8. The MD-KBS architecture.

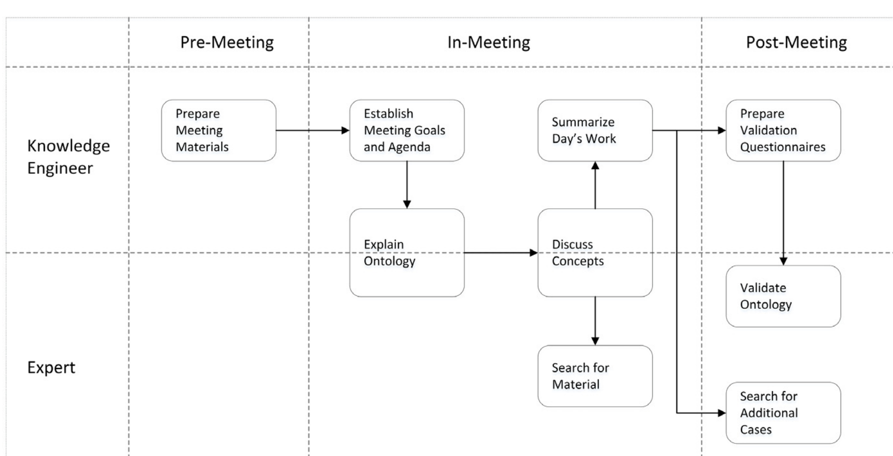

Fig. 9. The Xtreme Collaboration meeting process dynamics, highlighting the major activities for the Expert and KE roles at each moment of the process: Pre-meeting, In-meeting and Post-meeting. The process loops back with the results of experts' validation feeding into the pre-meeting stage for the next meeting.

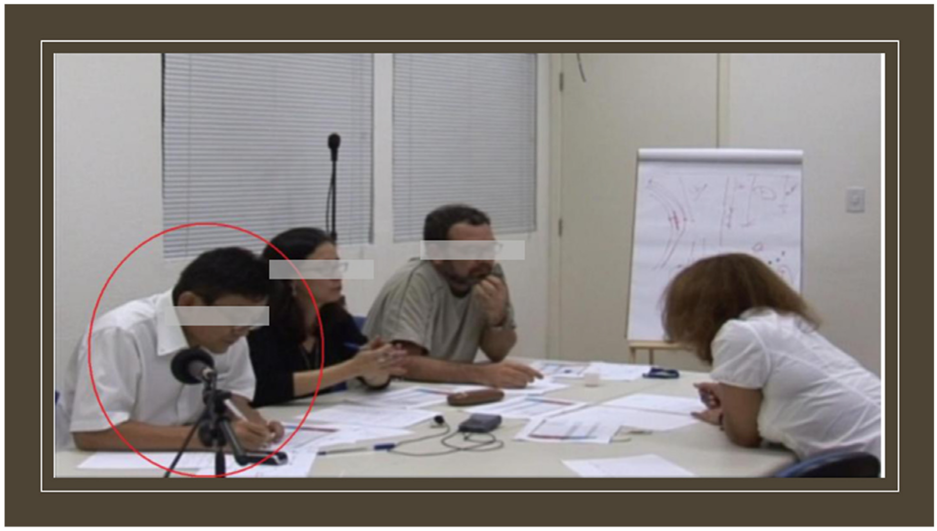

Fig. 10. Knowledge acquisition session.

were documented so that they could be reviewed at meetings, which meant that the final documentation stage involved mostly putting these parts together. The resulting ontology has about 400 concepts and each concept has and average of 10 unique values (possible instances). For example, when instantiated, the concept *Root Cause* may have one of the following concrete values: *Improper Manufacturing*; *Faulty Operation*; *Hazardous Environment*; *Lack of Maintenance*; *Poor Design Specification* (among others).

3.3. The drilling rig diagnosis ontology

The final ontology describes the multiple artifacts found in an oil drilling rig system and their relationships. The main goal of the system is detecting and explaining equipment failure, so each artifact's behavior is described as an expected behavior and an observed behavior. These allow the system to detect anomalies (when a potential problem is happening). Possible causes (immediate and root causes) are mapped so that the system can reach a diagnosis.

The ontology describes an oil drilling rig system as an engineering system composed of equipment that in turn are made up of components that may also be composed of smaller components. Components, equipment and the system itself are artifacts that accomplish a function, perform a behavior and have a form. An artifact is designed according to a specification that conforms to the expected form, function and behavior. It is constructed under given conditions, to operate under certain operational and environmental conditions. Sensors are embedded in artifacts and their measurements reflect the artifact's perceived behavior, which may indicate an anomaly when compared with its expected behavior (defined in the design specification). An *Anomaly* is caused by an *Immediate Cause* that sometimes hides the true reason, called a *Root Cause*. *Electrical System Shutdown*, *Sensor Failure* and *Rotor Blade Failure* are examples of *Immediate Causes* while *Lack of Maintenance* and *Hazardous Environment* are examples of root causes.

The ontology was documented using a semantic graph, a formal representation (rules in the KB) and a textual description for human understanding, as shown in Tables 2 and 3. This material was used to guide knowledge acquisition and also for the implementation of the ontology.

Table 2
Ontology documentation template-CONCEPT.

Concept Attribute	Value
Name	String
Description	Textual explanation
Presentation Order	Display order during the explanation
Values	Anything
Calculation Method	Pointer to relationship
Evaluation Method	Pointer to relationship
Dependency	Pointers to affecting concepts
Effect	Pointers to affected concepts
Data source	Source for data entry
References	Literature, case examples and KA Id

Table 3
Ontology documentation template-RELATIONSHIP.

Relationship	Value
Name	String
Description	Textual explanation
Method	Heuristic, Mathematical formulas, Process
Input Concept	Pointer to concepts
Goal Concept	Pointers to concepts
References	Literature, case examples and KA Id

3.4. Ontology explanation and validation

The quality of any ontology-based system will be, at best, as good as its ontology. There are methods, such as oQual [22] and OntoClean [23], and tools, such as, OOPS! [24] and MoKi [25], to assist ontology validation. Nonetheless, the validation focus has been on syntactic issues, such as identifying cycles or language non-compliance. However, semantics also impacts system outcomes. By semantic validation, we mean human understanding of and agreement with the ontological representation. This should also include evaluation through analysis of system outcomes. We could draw a parallel with a computer program, in which syntactic errors are identified through inspection by the compiler and semantic problems are only identified at run time. Problems with ontology syntax can be identified a priori, while semantic problems require running cases. Given that the ontology was part of a computational system, it had to be properly validated, or the system would be fated to fail. In this section we describe our ontology evaluation process.

There are many approaches to ontology evaluation [26]: the most common are through human assessment [27] and task result comparison [28]. The first approach is subject to human interpretation and checks overall acceptance of the given domain conceptualization by the group that built it or that is going to use it. The second checks precision by verifying whether the task

Please cite this article as: A.C.B. Garcia and A.S. Vivacqua, Grounding knowledge acquisition with ontology explanation: A case study, Web Semantics: Science, Services and Agents on the World Wide Web (2019), https://doi.org/10.1016/j.websem.2018.12.005.

the ontology was designed for was accomplished. No matter the approach, an ontology should be checked considering many facets. Gangemi [22] lists the following dimensions to take into account when evaluating an ontology:

- the *structural* dimension which concentrates on syntax and the formal semantics of the ontology;
- the *functional* dimension reflects the intended usage of the ontology and its components;
- the *usability-profiling* dimension focuses on the communication context of an ontology.

There are many supporting tools to assist structural evaluation, especially if the ontology is represented as a semantic graph. Knowledge coverage, graph depth and term consistency are some of the metrics that can be used. Guarino [23] suggests checking structural conformity by looking at concepts such as essentiality, rigidity and unity. Functional evaluation involves checking the mapping between the concepts and the world itself. It depends on human acceptance of the conceptualization of the domain. Precision and accuracy are the most frequently used metrics. Usability-profiling is also dependent on human interpretation, but deals with the synthesis of the content and not only with subsets of the domain. People sometimes agree on ad hoc conceptualizations but strongly reject the outcomes reached by using the ontology.

We evaluated our ontology considering the structural, functional and usability dimensions. At each ontology revision, we visually inspected the semantic graph looking for cycles and inconsistencies. Afterwards, we used a formal language to represent the concepts and ran examples to identify systematic problems.

As mentioned earlier, the domain ontology was used as the basis for the knowledge base and inference rules, and was also used to generate explanations about the system's rationale when a solution was reached. This is accomplished by following the concepts in the ontology and their relations. In addition to presenting experts with results to assess correctness, the systems provides a computationally constructed explanation to make sure the decision process was accurate.

The explanation followed a systematic script. Instead of presenting the entire semantic network at once, it connects parts of the ontology using a story-like thread. Each concept's explanation naturally flows from previous ones, as the explanations delve into more detail on each element involved in reaching a solution. This approach helped the experts understand not only the rationale underlying the system's results, but also KEs' understanding of the knowledge contained in the ontology.

This explanation script was used at each meeting, during the initial validation phase and was then implemented for use in the ontology-based explanation for the KBS. During meetings, the explanation followed a systematic, but not automatic, generation process (automatic explanation generation was only implemented in the final system). The explanation follows a presentation order that is indicated in each concept's attributes, and was adjusted to maximize user understanding. Fig. 11 illustrates a simple example of the ontology explanation script.

The questionnaires used in between meetings asked experts for different ways to define concepts, the possibility of excluding or merging concepts, the understandability of the ontological representation and the mapping of the concepts to the real world domain. The goal of the questionnaire was to check structural, functional and usability aspects of the ontology with experts. In parallel, we computationally encoded the ontology to look for systematic inconsistencies and flaws. We used a rule-inference engine coupled with the ontology representation and checked the partial results looking for flaws in the knowledge base. There was no systematic pattern for refining the ontology, but the number of required changes to existing concepts diminished as the ontology grew, which can be seen as a sign that the model was converging.

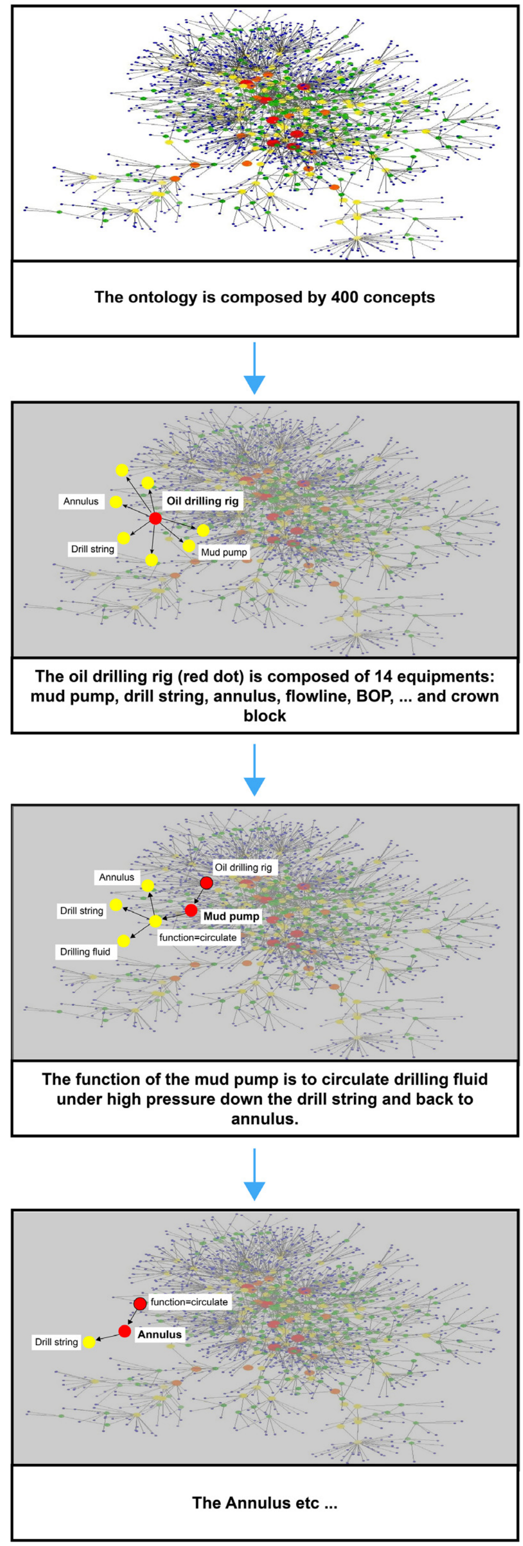

Fig. 11. Ontology explanation script.

3.5. Expected vs actual system outcomes

Even though we followed a straightforward procedure for building the ontology, system outcomes were inadequate when running real life cases, and experts sometimes disagreed with system's results. There were a few scenarios in which our experts agreed with the system's diagnosis, but disagreed with the root causes (these situations were labeled *partial agreement*). In other cases, some experts, but not all of them, disagreed with the entire results (these situations were labeled *conflicting agreement*). There were also scenarios in which the group agreed with the system's diagnosis, regarding which part was affected, but not exactly on which specific component failed (these situations were labeled *superficial agreement*). This led to a review of the ontology, and a reflection on the process, to try to understand what caused this mismatch between the model and its application. It should be noted that the experts were willing to go back to the ontology and work with us on alterations because they felt as responsible for the ontology as the KEs. The process of collaboratively building the model generated accountability so they also wanted to make sure the project had good results. We feel this is a consequence of their involvement in the creation process during XC meetings.

After fixing the model, we went back to interview experts, in order to better understand what had caused these error situations. We conducted semi-structured interviews, using the ontology-based explanations to show the rationale for answers provided by the KBS. This investigation led to the observation that two of the experts systematically disagreed with each other and with the results. The results provided by the system for the concrete cases were in accordance with the organization's reported diagnosis. These had come from these experts' divisions, so we had expected they would agree with them, but that was not the case.

We identified three conditions for a mismatch between the expected answer and the one provided by the KBS:

- *Expert disagreement with details, based on their beliefs and training*. In this case, nothing can be done to accommodate the expert. The organizational perspective prevailed, but an observation was made that this was not consensual knowledge.
- *Missing concepts that would better distinguish the specific diagnosis*. This condition was noticed due to actual cases that pushed the limits of the ontology scope. This called for additional meetings to refine the ontology and add the missing concepts.
- *Concept misrepresentation in the ontology*. This condition happened when the experts agreed among themselves but disagreed with the explanation provided by the KBS. This is an actual error that should be taken into account when measuring the system's precision. In this case, the error needed to be investigated and a revision to the ontology was necessary.

The following sections further discuss the takeaway lessons we learned from these observations.

4. Discussion and lessons learned

Using video recordings of knowledge acquisition sessions as a basis, we investigated possible reasons for the wrong outcomes and reflected on the process as a whole, to extract lessons that could be observed in our next projects. In this section, we discuss the results of our reflection in action.

4.1. The semantic validation challenge: "Everything looks fine from a distance..."

In some of the cases, the general diagnosis was accepted by experts, but more specific elements were questioned. For instance, in one situation, all experts agreed that the problem was caused by a hydraulic system failure, but disagreed on the specific faulty component.

Looking back at the knowledge acquisition video records, we could identify this agreement/disagreement pattern. Experts tended to converge at an abstract level of diagnosis, but disagreed at more detailed levels. Checking their backgrounds, we verified that they came from different schools of reasoning and worked in very different operational settings, which led to different experience and working knowledge. This should be a reason to enrich the ontology and add more detail to accommodate the differences, not impoverish it. This observation led us to our second observation.

4.2. The dangers of conflict avoidance

The seven experts that participated knew and respected each other, and had worked before in related projects. They collaborated for almost six months to construct this ontology, coming together at least once a month, answering ontology evaluation questionnaires, reviewing materials and arguing among themselves until consensus on the domain conceptualization was reached.

Reviewing the video recordings of the ontology construction sessions, we observed that participants deviated from concrete cases at meetings, using instead toy examples to explain how each concept fit in specific situations. Knowledge acquisition sessions lasted more than 4 h, sometimes the entire day. Sometimes, experts would try to avoid conflicts with each other as time passed. They would speak in abstract terms or more generically about rules and concepts, which later did not fit the real life scenarios. This might have been a reason for the poor result seen later on: the ontology was not appropriately anchored on the real world. While more abstract concepts and theoretic rules may apply to toy problems, they do not fit well to real world scenarios. Still, this situation was unexpected, since the entire group was officially accountable for the quality of the system and the intermediate verification questionnaires were signed by all participants.

4.3. The importance of diverse and real cases

This follows from the previous two lessons Actual real-life cases were much more fuzzy and complex when compared with the examples selected by the experts. When questioning the experts, all of them stood by their choices advocating that the selected examples had educational value. They did not realize the need to discuss exceptions and more complex cases. In fact, they did not realize what they called exceptional situations also happen frequently and can have serious consequences. In future cases, we will enforce the usage of real life examples for illustration, avoiding abstract generalizations.

4.4. The role of explanation in ontology validation

The explanation played a fundamental role in the identification of problems with our ontology. The explanation followed a script that navigated through the ontology to explain a result, given a set of symptoms and the context in which the failure happened. Experts could follow and understand the ontology and pinpoint the incompleteness, flaws and interpersonal disagreements.

These observations led us to include in our KA process a random selection of case examples, prototyping ontology-based explanations (even without the KBS ready), as recommended by agile software development methods [29]. We are now applying this new method in other projects.

4.5. The value of expert involvement

Experts were highly involved in ontology construction in the MechDrill project. While we have involved domain experts in ontology construction in previous projects, the XC dynamics applied to the discussion and continuous revision process generated higher participant commitment and a feeling of ownership towards the

Please cite this article as: A.C.B. Garcia and A.S. Vivacqua, Grounding knowledge acquisition with ontology explanation: A case study, Web Semantics: Science, Services and Agents on the World Wide Web (2019), https://doi.org/10.1016/j.websem.2018.12.005.

Table 4
Comparison of projects involving ontology construction.

Project	Domain	Project Goal	Ontology Usage	Number Experts	Construction Process	Results
ADDVac	Ventilation and Air Conditioning System (HVAC)	Project Design and Documentation	KB construction and Inference	1	Methontology	Design assistant tool successfully developed and deployed
ADDProc	Design and Processing Plant Processing Plant	Project Documentation Documentation	KB construction and Inference and Inference	17	Methontology	Design assistant tool successfully developed but not deployed
ADDDut	Oil Pipeline	Failure Diagnosis	KB construction and Inference	4	Methontology	Failure diagnosis tool successfully developed and deployed
ADDSub	Oil Pipeline	Design and Documentation	KB construction and Inference	4	Methontology	Design assistant tool successfully developed and deployed
DMRisk	Failure Records Oil Production	Managerial process analysis	Data Mining	8	Methontology + XC	Root cause analysis tool successfully developed and deployed
MechDrill	Onshore Oil Drilling	Failure Diagnosis	KB construction and Inference	7	Methontology + XC	Failure diagnosis tool successfully developed and deployed

ontology. Experts felt accountable for the final product and its results, as they had spent considerable time arguing and defending their perspectives. Experts had an active role building the ontology, not the knowledge engineers (who served as guides and facilitators at meetings). This is an important distinction, as some of our previous projects have shown. In previous projects with the same company, we experienced a number of different configurations and refined the ontology construction technique. These projects are summarized in Table 4 and explained below.

ADDVac involved only one expert, who was also intended the user, in the creation of an ontology for a system to assist and document design choices, specifically for the domain of ventilation and air conditioning (VAC) systems for oil production plants. The project applied case-based reasoning techniques and the ontology was used as the basis for reasoning and explanation. The system had satisfactory results and was adopted by the company, being used in real life scenarios. Ontology construction was conducted in a traditional way, with interviews and document reviews for knowledge acquisition, and followed the Methontology process closely.

ADDProc involved multiple experts to help design the ontology that would serve as the basis for a KB and inference engine in a system to assist and document design choices for the domain of offshore oil production plants. This domain is more complicated than VAC, and involves multiple interconnected systems and experts responsible for each one. Ontology construction also involved interviews and document reviews with each expert individually, and followed the Methontology process fairly closely. This project also applied case-based reasoning techniques. The system had satisfactory results but was not adopted by the company, it was never used in real life. We believe that the fact that experts did not actively engage in the process and discuss among themselves led to the system not being adopted.

ADDut also involved multiple experts, who designed an ontology that served as the basis for a failure diagnosis decision support system. The company wanted to detect problems as soon as they occurred, to avoid environmental disasters caused by oil spills. The company uses sensors and mathematical models to detect leaks, but with low accuracy. They wanted to test other techniques to improve results. Domain experts were engineers and operators with more than 10 years of experience. KA took more than the planned 10 sessions, but the system was developed and deployed with good results.

Multiple experts were also involved in ADDSub. Although the domain was very complex, involving a multi-objective optimization function in which decisions were highly dependent on each other, experts were used to working together, so there was not much conflict among them. They designed an ontology that served as the basis for a layout algorithm: pipeline layout accounts for about 30% of the total platform cost, which reaches billions of dollars. The ontology was constructed in a classic fashion. The system yielded good results and was deployed at the company.

DMRisk also brought together multiple experts, to design an ontology that served as the basis for a data mining algorithm. This was part of a system to assist upper management in the aggregate analysis of oil production plants (based on incident reports). This was a strategic project for the company, and pressure was high for it to work. Experts worked at an upper management level, and had a strong understanding of the processes and unexpected events that might happen at an oil production plant. We applied XC techniques in this project, and experts came together weekly to discuss and construct the ontology. The system yielded good results and was adopted by the company at a strategic level.

MechDrill is discussed in this paper. It involved multiple experts, who designed an ontology that served as the basis for a KB and inference engine, in a system to diagnose failures in pumping equipment. Given the previous results, we again applied Methontology coupled with XC techniques, refining the process and the steps. The final ontology required added revision, but the system yielded good results and was adopted by the company, being used frequently.

Looking back at these previous projects and talking to experts involved, one of the most striking distinctions seems to be the increased acceptance and adoption of projects created using XC dynamics, particularly when multiple stakeholders are involved. ADDVac involved only one expert, so there was no disagreement during project construction and adopting XC did not make sense. ADDUT was a simple domain with an existing mathematical model to guide the domain model. ADDSub involved multiple experts in a complex domain, but with little disagreement among the experts.

In projects dealing with complex domains and involving multiple stakeholders with conflicting points of view (such as ADDProc, DMRisk and MechDrill), the application of XC led to a semi-formal agreement on how to address the task, more commitment on the team, and a feeling of ownership of the ontology and the whole project. Being involved in the project also made them accountable for the results. Since they had validated the partial ontology at

Please cite this article as: A.C.B. Garcia and A.S. Vivacqua, Grounding knowledge acquisition with ontology explanation: A case study, Web Semantics: Science, Services and Agents on the World Wide Web (2019), https://doi.org/10.1016/j.websem.2018.12.005.

intermediate stages and the full model at the end, they were also responsible for system outcomes, good or bad. This generated stakeholder buy-in, an interest in seeing the project succeed and also drove adoption at a later stage.

DMRisk and MechDrill both adopted the same methodology for ontology construction, but MechDrill was the only one where the system's results where inadequate and a revision of the ontology was necessary after implementation. We investigated what factors could have caused these poor results (these were reported in previous subsections). It should also be noted that the added accountability created by the XC process meant that experts understood the need to participate in the revision and were willing to spend time on it. Thus, we believe expert involvement in the whole process is important to generate more commitment to the process and drive system adoption, and XC is one way to achieve that.

4.6. User impressions

After deploying the MechDrill system, we conducted two focus group interviews: one with KEs and another with domain experts. We were interested in their perception of the KA process and the quality of the outcome (the KBS system). The interviews were planned to take about 1 h each. They were annotated for further analysis. We asked five type of questions:

1. KA process evaluation: All meetings had a set of pre-defined goals to be reached. There were no pre-defined duration for the meetings, but, instead, an expected outcome. What is your perception of that? Did it work? Could you adjust to meeting dynamics? Cite your perception on pros and cons about that.
2. KA participants' role: What is your perception of the KE role in the KA meetings? What was your perception of the domain experts' role in the KA meetings?
3. Conflict mitigation: How were conflicts mitigated during the KA meetings? What would you have done differently had there been no time constraint?
4. Ontology quality: There were some problems with the first complete ontology. From your point of view, why did these happen?
5. General perception: From your perspective, what are the positive and negative aspects of the KA process?

Only 5 out of the 7 domain experts participated in the focal group interview due to the logistics of bringing in two that lived in distant states. The interview with domain experts lasted longer than the expected 1 h. The two KE participated in a separate focus group. Table 5 shows a summary of observations gathered in these interviews. There are pros and cons to imposing a strict time frame on discussions and strict goals for meetings. Ontology creation is faster, experts develop a feeling of ownership and conflicts focus on technical issues, but concepts might be superficially explored. Including an ontology-based explanation was fundamental to identify flaws and inconsistencies that the toy examples used during the meetings did not reveal.

5. Related work

Research on ontology engineering has been plentiful in recent years, with a number of case studies being presented (e.g.: [29–35], among others). Research focuses have varied depending on the KA phase, from initial knowledge acquisition [30] to knowledge evolution and maintenance (e.g.: [32,36]), and on specifics of application domain areas (e.g.: health tourism [33], philosophy [37] and biomedical resources [35]).

Many researchers focus on the use of ontology to provide a standardized vocabulary to describe elements in a domain (which in turn increases interoperability) [31–35,38]. This has been a recent emphasis of ontology research, and, while an important one, is very different from ours. In these cases, ontology is used to annotate resources and help retrieve them at a later stage, using methods such as Keyword Expansion. Our goal, in contrast, was to build a system that uses the ontology not only for annotation, but also as a basis for reasoning and explanation using inference mechanisms and rules defined in the ontology. This difference in purpose leads to different requirements regarding checking and validation from those designed for annotation.

According to [39], ontology evaluation can take one of three forms: Evolution, Rules or Metrics. In fact, many papers mention evaluation based on structural measures (e.g.: [40]) or logical statements. These work well with automation and lead to the development of tools to expedite the process. However, they tend to leave the human out of the loop, which can be a problem. Experts will be the ones who can best analyze the semantic aspect of the ontology and determine whether the concepts and rules encoded actually make sense in the real world. In [41], an additional category is added, for methods that provide human analysis. However, there is still little mention of methods that tie back to the concrete, real-world elements the ontology is supposed to represent. In [36], this issue is addressed, and metrics are proposed to assess how realistic the ontology is, using responses to objective questionnaires.

Many researchers seem to look at ontology construction as a process that does not require the involvement of domain experts, focusing instead on logical formalisms, knowledge extraction from existing resources and/or ontology reuse. We prefer to involve experts in the process, as their knowledge is fundamental for a system to work properly. In addition, they will be the ones using and ratifying the system at a later stage, so it is important to encode their expertise and have their buy-in.

In [24], authors present a table comparing different tools and the coverage of ontology evaluation dimensions on each one. It is striking that the *semantic application* dimension is not covered by any of the tools mentioned. Semantic application refers to the fit between the ontology and its intended use. This is exactly where we came across problems in this process: the ontology had been validated according to all other aspects, but was still not ready for use.

Most of the reviewed approaches use questionnaires or reviews to validate the model with experts, as we did. However, that does not seem to be enough to generate good review of the model, so that it can effectively be considered validated. As a result of our experience, we are focusing our research on the development of explanations, experimenting with different forms of content presentation and how these can be used for ontology validation.

6. Conclusion

This paper presented a case study of domain ontology construction involving multiple experts, that, although following a robust KA method, when implemented as part of a system, yielded inadequate results, indicating that the ontology was not ready for use. Reflecting on the case, we realized the importance of an ontology-based explanation to identify the problems within the process. In our experience, consistency checks and ontology inspection alone do not guarantee semantic quality.

The acquisition of tacit knowledge from multiple experts offers additional challenges and affects the quality of the resulting domain ontology. Experience and background differences between experts may emerge when they are faced with actual cases. Accuracy metrics may indicate the problem, but will not be sufficient to identify the reasons for the problems. In addition, most evaluation

Table 5
Participants' KA process evaluation.

	Knowledge Engineer	Domain Expert
KA process evaluation	"... lots of extra work in between meetings" "... I recall in the first KA meeting, two experts started arguing on how a specific symptom should be applied to a diagnosis they could not stop arguing. They had different backgrounds, training, experience..." "... It was a long meeting. So we decided " on a new rule: maximum duration for any discussion. Otherwise, bring more evidence..." "... stressful to manage 7 experts older than me..." "... They started asking: what is ontology? In the end, they were feeling that they were ontology experts. But they are not.." "... interesting experience and good results.."	"... sometimes meetings took too long..." "... at least we could see the outcomes..." "... I liked the ontology tool ..." it helped organize my thoughts..." "... I do not like to be interrupted..." ".. I was arguing, but I could see that I was going in circles..." "... it helped us reach an agreement..."
KA participants' role	"... domain experts were very reluctant to accept the long meetings..." "... Sometimes experts gave up for a while during a meeting and came back, in a next meeting, with strong evidences."	"... it seemed they knew what they were doing..." "... I do not know how to judge..."
Conflict mitigation	"... I am glad, we negotiated a maximum duration for the conflicts..." "... The explanation was powerful to help them agree and also encourage new discussion.." "... I felt the explanation was a type of help system. It is powerful..." "... After this project, I became an advocate for explanation"	"... sometimes they interrupted too much... " "... sometimes they triggered discussions..." "I liked when they started drawing and telling stories" "... a lot of homework..." "... a lot of work..."
Ontology quality	"... we almost failed.. one of the diagnosis was completely wrong" "... the topic was discussed among the experts and they brought simple examples..." "... they brought simple problems to illustrate their point of views. That was a big mistake..." "... the system explanation helped them identify flaws during the system's test phase, in time for fixing.."	"... I don't know about the ontology, but the system was successfully deployed and is in use within the company..."
General perception	"... I will set up smaller goals..." "... I have mixed feelings about the maximum time limit for discussion and strict meeting goal.." "... overall, it worked well" "Explanation, explanation, explanation!"	"... long meetings..." "... a lot of work..." "... successful..." "... job with focus..." "... I recommend..."

measures assess the quality of the results, but cannot identify the causes for errors. For this reason, problems may slip by undetected until system deployment. Using an ontology-based explanation played an essential role in detecting the flaws in the model and the process. Including the experts in the process through XC was fundamental to increase stakeholder buy-in for the project and commitment to the results.

Once shown the explanation generated using concrete test cases, experts questioned the validity of the ontology, both in terms of concepts used and of the relations between them. The application of the instantiated ontology to real cases created a better understanding of the domain representation for the experts, and brought to light the shortcomings of the ontology. Faced with concrete examples, experts could detect inconsistencies in the model and tease out conflicting issues among their perspectives, reconcile them and demand revisions to the ontology. This situation intrigued us, as the ontology had been extensively revised and checked for inconsistencies through the use of questionnaires and individual interviews, and we had expected this would ensure correct results, accepted by the experts that participated in the development. Instead, the explanation revealed the shortcomings of the domain model, supporting the ontology review process. We now see great value in including ontology-based explanations in the ontology construction process.

To summarize, our case study sheds light on important issues related to the construction of any knowledge-based computer system. We have learned that:

- The XC strategy helps (1) speed up knowledge acquisition in domain where multiple experts are involved and (2) increase stakeholder buy-in. Participants are committed to listen and understand the others' perspectives in order to reach an acceptable model (preferable consensual), and have ownership of the model, as they were directly involved in its creation.
- Using real domain cases during ontology development is a good way to elicit conflicts at the design stage. Simple cases are good to illustrate particular issues, but do not lead to a robust domain model.
- Providing a symbolic explanation is fundamental to increase acceptance of a knowledge-based system. Explanations help participants understand and audit the ontology, strengthening its verification.

Besides these, we noticed that exploring the explanation and understanding the rationale for a system's answer, increases participant trust in the model. The ontology plays an important role in the generation of human understandable explanations, which is still a challenge for machine learning systems.

Using XC + Methontology requires added work to setup meetings and build explanations at each revision. Additionally, it requires the construction of an automatic explanation module. While this method would be applicable to any domain, the extra cost incurred is only justified in domains with multiple interdependent decisions, involving different areas of expertise and in which multiple solutions are possible. Domains such as emergency management and hospital work, where experts make decisions based not

Please cite this article as: A.C.B. Garcia and A.S. Vivacqua, Grounding knowledge acquisition with ontology explanation: A case study, Web Semantics: Science, Services and Agents on the World Wide Web (2019), https://doi.org/10.1016/j.websem.2018.12.005.

only on norms and regulations, but also on their personal previous experiences that might differ from each other. In these cases, there is a large amount of tacit knowledge to be acquired, which also often includes conflicting rationale for decisions. This justifies the extra effort needed to setup and conduct XC meetings and explanation sessions; and to tease out and resolve conflicts using concrete examples. This also means the ontology and the knowledge base might require periodic revisions. Simpler domains, where most decisions are based on well-documented rules and experts do not deviate from them or do not disagree with each other might not need such a complex setup, as the ontology can be acquired through more conventional means.

6.1. Future work

The quest continues to design tools that can help maintain people on the same track, and make them aware of discrepancies in points of view. We believe people will make sense of what they are designing when they see it in the form of concrete examples or cases [42]. That could save time, increase acceptance and diminish rework during ontology construction. Our study highlights a need for better semantic evaluation methods for ontologies and we believe that including system-based explanations during the whole ontology construction process will result in improved system quality overall.

In the explanation system, facts were tied together with a script, which we thought looked like a simple story. With this in mind, we are now working on ontology validation through storytelling. We are working on formalizing a method to walk an expert user through the ontology by telling the story of a real life case. For this to work, a few elements must be in place:

- Starting point: an entry point for the story should be defined in the ontology, providing an initial concept. This will usually be the most central concept of the ontology.
- Storytelling flow: every story has a temporal flow, or an event flow. For each ontology, an ideal flow of concepts must be defined, which tells the storyteller how to move between concepts and relations. This will usually be done by the knowledge engineer assisting the experts, and may vary depending on the objective of the ontology.
- Domain corpus: more than one story should be selected from a corpus of real life situations. This set of stories should be representative of the full set of examples, covering a few stories of each type.
- Reader: ideally, more than one expert should go through the process, validating the ontology against each case. Preferably, this should include experts who participated in ontology construction as well as others who did not. For cross validation, multiple experts should be told the same story, in order to appropriately verify that the ontology is valid.

When told, the story should reveal features of the events that took place and the rationale for the result, following a predefined process. The story should be told according to the concepts and values contained in the ontology. It should be noted that the story does not necessarily need to be long or contrived: it may be one paragraph, describing how concepts are linked to each other, or it may be a long causal chain. We have been experimenting with this technique in a new project, where it has been fairly well received. This seems to be an appropriate solution, since this new project deals with temporal and causal event chains. The greatest advantage is that the stories are concrete cases, which ensures that the ontology covers real life situations.

Building an ontology from tacit expert knowledge will always incur the risks reported in this paper. People may perceive and express their knowledge in different ways. Incompatibilities are normal since participant experiences vary with training and work exposure. A top-down KA process may benefit from including more test cases and from exposing the ontology through constant systematic case explanations. On the other hand, flooding the KA process with cases as in a bottom-up approach may lead to the problem of overfitting and lack of generalization. Finding the right balance between concrete cases and tacit knowledge is an open challenge.

Acknowledgments

This work was supported by CNPq, Brazil grants 401835/2013-8 and 309171/2015-6; and FAPERJ, Brazil grant E-26/202.753/2015.

References

[1] Z. Yin, J. Hou, Recent advances on SVM based fault diagnosis and process monitoring in complicated industrial processes, Neurocomputing 174 (2016) 643–650.
[2] L. Branting, Data-centric and logic-based models for automated legal problem solving, Artif. Intell. Law 25 (1) (2017) 5–27.
[3] T. Gruber, Ontology of folksonomy: A mash-up of apples and oranges, Int. J. Semant. Web Inf. Syst. 3 (1) (2007) 1–11.
[4] G. Castrogiovanni, D. Ribeiro-Soriano, A. Mas-Tur, N. Roig-Tierno, Where to acquire knowledge: Adapting knowledge management to financial institutions, J. Bus. Res. 69 (5) (2016) 1812–1816.
[5] S. Ryan, R.V. OConnor, Acquiring and sharing tacit knowledge in software development teams: An empirical study, Inf. Softw. Technol. 55 (9) (2013) 1614–1624.
[6] W. Fan, A. Bifet, Mining big data: current status, and forecast to the future, ACM sIGKDD Explorat. Newsl. 14 (2) (2013) 1–5.
[7] W.P. Wagner, Trends in expert system development: A longitudinal content analysis of over thirty years of expert system case studies, Expert Syst. Appl. 76 (2017) 85–96.
[8] M. Titah, S. Aitouche, M.D. Mouss, A. Soussa, Externalising and reusing of tacit knowledge in manufacturing task, Int. J. Knowl. Manag. Stud. 8 (3–4) (2017) 351–374.
[9] W. Kim, S. Katipamula, A review of fault detection and diagnostics methods for building systems, Sci. Technol. Built Environ. (2017) 1–18.
[10] D. Dou, S. Zhou, Comparison of four direct classification methods for intelligent fault diagnosis of rotating machinery, Appl. Soft Comput. 46 (2016) 459–468.
[11] M. Fernández-López, A. Gómez-Pérez, N. Juristo, METHONTOLOGY: From Ontological Art Towards Ontological Engineering, in: Proceedings of the Ontological Engineering AAAI-97 Spring Symposium Series, American Association for Artificial Intelligence, 1997, pp. 33–40, Ontology Engineering Group? OEG, http://oa.upm.es/5484/.
[12] M.C. Suárez-Figueroa, A. Gómez-Pérez, M. Fernandez-Lopez, The NeOn methodology framework: a scenario-based methodology for ontology development, Appl. Ontol. 10 (2) (2015) 107–145.
[13] A.C.B. Garcia, J. Kunz, M. Ekstrom, A. Kiviniemi, Building a project ontology with extreme collaboration and virtual design and construction, Adv. Eng. Inform. 18 (2) (2004) 71–83.
[14] A.S. Vivacqua, A.C.B. Garcia, A. Gomes, BOO: Behavior-oriented ontology to describe participant dynamics in collocated design meetings, Expert Syst. Appl. 38 (2) (2011) 1139–1147.
[15] D.H. Stamatis, Failure Mode and Effect Analysis: FMEA from Theory to Execution, ASQ Quality Press, 2003.
[16] W. Lee, D.L. Grosh, F.A. Tillman, C.H. Lie, Fault Tree Analysis, Methods, and Applications A Review, IEEE Trans. Reliab. 34 (3) (1985) 194–203.
[17] E. Moradi, S.F. Ghomi, M. Zandieh, Bi-objective optimization research on integrated fixed time interval preventive maintenance and production for scheduling flexible job-shop problem, Expert Syst. Appl. 38 (6) (2011) 7169–7178.
[18] A. Van Horenbeek, L. Pintelon, P. Muchiri, Maintenance optimization models and criteria, Int. J. Syst. Assur. Eng. Manag. 1 (3) (2010) 189–200.
[19] S. Martorell, J.F. Villanueva, S. Carlos, Y. Nebot, A. Sánchez, J.L. Pitarch, V. Serradell, RAMS+ C informed decision-making with application to multi-objective optimization of technical specifications and maintenance using genetic algorithms, Reliab. Eng. Syst. Saf. 87 (1) (2005) 65–75.
[20] G. Mark, Extreme collaboration, Commun. ACM 45 (6) (2002) 89–93, http://dx.doi.org/10.1145/508448.508453, URL http://doi.acm.org/10.1145/508448.508453.
[21] G.V.L. Moises, S.F.A. Andrade, A.C.B. Garcia, et al., Sucker-rod pumping failures diagnostic system, in: SPE Annual Technical Conference and Exhibition, Society of Petroleum Engineers, 2010, pp. 1–12.

[22] A. Gangemi, C. Catenacci, M. Ciaramita, J. Lehmann, Modelling ontology evaluation and validation, in: ESWC, vol. 4011, Springer, 2006, pp. 140–154.

[23] N. Guarino, C. Welty, Evaluating ontological decisions with OntoClean, Commun. ACM 45 (2) (2002) 61–65.

[24] M. Poveda-Villalón, M. Suárez-Figueroa, A. Gómez-Pérez, Validating ontologies with OOPS!, Knowl. Eng. Knowl. Manag. (2012) 267–281.

[25] M. Dragoni, A. Bosca, M. Casu, A. Rexha, Modeling, Managing, Exposing, and Linking Ontologies with a Wiki-based Tool, in: N. Calzolari, K. Choukri, T. Declerck, H. Loftsson, B. Maegaard, J. Mariani, A. Moreno, J. Odijk, S. Piperidis (Eds.), Proceedings of the Ninth International Conference on Language Resources and Evaluation, LREC 2014, Reykjavik, Iceland, May 26-31, 2014., European Language Resources Association (ELRA), 2014, pp. 1668–1675, URL http://www.lrec-conf.org/proceedings/lrec2014/summaries/769.html.

[26] J. Raad, C. Cruz, A survey on ontology evaluation methods, in: Proceedings of the International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, IC3K 2015, SCITEPRESS - Science and Technology Publications, Lda, Portugal, 2015, pp. 179–186, http://dx.doi.org/10.5220/0005591001790186, URL https://doi.org/10.5220/0005591001790186.

[27] A. Lozano-Tello, A. Gómez-Pérez, Ontometric: A method to choose the appropriate ontology, J. Database Manag. 2 (15) (2004) 1–18.

[28] R. Porzel, R. Malaka, A task-based approach for ontology evaluation, in: ECAI Workshop on Ontology Learning and Population, Valencia, Spain, 2004, pp. 1–6.

[29] M. Copeland, A. Brown, H.E. Parkinson, R. Stevens, J. Malone, The SWO project: A case study for applying agile ontology engineering methods for community driven ontologies, ICBO 7 (2012) 2012.

[30] M. Fernández-López, A. Gómez-Pérez, Overview of methodologies for building ontologies, Knowl. Eng. Rev. 17 (2) (2002) 129–156.

[31] M. Silalahi, D.I. Sensuse, A case study in a socio-technical approach in ontology engineering, in: Computer Science and Engineering Conference (ICSEC), 2016 International, IEEE, 2016, pp. 1–6.

[32] C. Tempich, H. Pinto, S. Staab, Ontology engineering revisited: An iterative case study, in: European Semantic Web Conference, Springer, 2006, pp. 110–124.

[33] C. Chantrapornchai, C. Choksuchat, Ontology construction and application in practice case study of health tourism in Thailand, SpringerPlus 5 (1) (2016) 2106–2106.

[34] A. Thunkijjanukij, A. Kawtrakul, S. Panichsakpatana, U. Veesommai, et al., Lesson learned for ontology construction with Thai rice case study, in: press. World Conference on agricultural information and IT, 2008, pp. 495–502.

[35] C. Torniai, M.H. Brush, N. Vasilevsky, E. Segerdell, M. Wilson, T. Johnson, K. Corday, C. Shaffer, M. Haendel, Developing an application ontology for biomedical resource annotation and retrieval: challenges and lessons learned, ICBO 2011 (2011) 101–108.

[36] W. Ceusters, B. Smith, A realism-based approach to the evolution of biomedical ontologies, in: AMIA Annual Symposium Proceedings, vol. 2006, American Medical Informatics Association, 2006, p. 121.

[37] P. Garbacz, Challenges for ontological engineering in the humanities: a case study of philosophy, Int. J. Metadata Semant. Ontol. 11 (2) (2016) 124–135.

[38] P. Green, M. Rosemann, Applying ontologies to business and systems modelling techniques and perspectives: Lessons learned, J. Database Manag. (JDM) 15 (2) (2004) 105–117.

[39] S. Tartir, I. Arpinar, A.P. Sheth, Ontological evaluation and validation, in: Theory and Applications of Ontology: Computer Applications, Springer, 2010, pp. 115–130.

[40] P.W. Lord, R.D. Stevens, A. Brass, C.A. Goble, Investigating semantic similarity measures across the Gene Ontology: the relationship between sequence and annotation, Bioinformatics 19 (10) (2003) 1275–1283.

[41] L. Obrst, W. Ceusters, I. Mani, S. Ray, B. Smith, The evaluation of ontologies, Semant. web (2007) 139–158.

[42] D. Kirsh, Interaction, external representation and sense making, in: Proceedings of the 31rst Annual Conference of the Cognitive Science Society, 2009, pp. 1103–1108.

Please cite this article as: A.C.B. Garcia and A.S. Vivacqua, Grounding knowledge acquisition with ontology explanation: A case study, Web Semantics: Science, Services and Agents on the World Wide Web (2019), https://doi.org/10.1016/j.websem.2018.12.005.