SUPPORTING TERMINOLOGICAL STANDARDIZATION IN CONCEPTUAL MODELS – A PLUGIN FOR A META-MODELLING TOOL

Prototype

Havel, Jean-Marie, University of Muenster - ERCIS, Leonardo-Campus 3, 48149 Muenster, Germany, jeanhavel@gmail.com

Steinhorst, Matthias, University of Muenster - ERCIS, Leonardo-Campus 3, 48149 Muenster, Germany, matthias.steinhorst@ercis.uni-muenster.de

Dietrich, Hanns-Alexander, University of Muenster - ERCIS, Leonardo-Campus 3, 48149 Muenster, Germany, hanns-alexander.dietrich@ercis.uni-muenster.de

Delfmann, Patrick, University of Muenster - ERCIS, Leonardo-Campus 3, 48149 Muenster, Germany, patrick.delfmann@ercis.uni-muenster.de

Abstract

Today's enterprises are accumulating huge repositories of conceptual models, such as data models, organisational charts and most notably business process models. Those models often grow heterogeneously with the company and are thus often terminologically divers and complex. This terminological diversity originates from the fact that natural language allows an issue to be described in a large variety of ways especially when many modellers are involved. This diversity can become a pitfall when conceptual models are subject to model analysis techniques, which require terminologically comparable model elements. Therefore, it is essential to ensure model quality by enforcing naming conventions. This paper introduces a prototype, which intends to resolve all associated issues of terminological standardisation already during the modelling phase or ex-post based on existing models. The modeller is guided through the standardization process by providing an automated list of all correct phrase propositions according to his entered phrase. In this approach, naming conventions can easily be defined and enforced. This leads to terminologically unambiguous conceptual models, which are easier to understand and ready for further analysis purposes.

Keywords: Terminological standardisation, conceptual modelling, linguistics, naming conventions.

1 Motivation

Today, business process modelling projects are increasingly growing with respect to their amount of models and thus their contained modelling elements (Dijkman, La Rosa, & Reijers, 2012). Moreover, many different distributed and collaborating modellers are contributing to those models, which are finally coalescing within those projects (Delfmann, 2010). Often those partitioned models need to be integrated into a larger model. Due to the large expressiveness of the natural language, not only different terms but also many different phrase structures are possible (Delfmann, Herwig, & Lis, 2009) which inevitably results in naming conflicts (Batini, Lenzerini, & Navathe, 1986). Even modellers with a similar educational background might have a different regional or cultural background, which often results in many different ways of labelling a given concept captured in a conceptual model. The terminological ambiguity of model elements leads to many problems. Already during the process of collaborative modelling it might lead to misunderstandings between modellers and thus will eventually end in conflicting changes. A modeller might for example label an activity “Bill control” which may or may not be semantically equivalent to “Check invoice” (Steinhorst, Breuker, Delfmann, & Dietrich, 2012). In order to standardise the sentence, words need to be replaced by dominant synonyms (e.g. the verb “to check” instead of “to control” and the noun “invoice” instead of “bill”). Furthermore, the sentence needs to be transformed into a valid phrase structure, which was previously defined (e.g. <Verb, Imperative> <Noun, Singular > cf. (Kugeler, 2000; Rosemann, 1996)). When analysing models, further issues with terminologically unstandardized models arise. In the domain of business process model analysis (e.g. model merging) it is essential that the modelling elements, i.e. node and edge labels are named consistently in order to allow for comparing them (La Rosa, Dumas, Uba, & Dijkman, 2010, 2013).

The literature reveals a variety of approaches, which intend to resolve naming conflicts in conceptual models (cf. Section 2). However, most approaches are not satisfying enough as they are either only applying guidelines (Kugeler, 2000; Rosemann, 1996), handling single noun ontologies (Koschmider & Oberweis, 2005; Sabetzadeh, Nejati, Easterbrook, & Chechik, 2007), suited for one modelling language (Born, Dörr, & Weber, 2007), not supporting arbitrary phrase structures, and moreover often involve costly *ex-post* standardisation effort. Delfmann, Herwig and Lis (2009) suggest an approach, which allows ensuring terminological standardization already during modelling or ex-post. Thereby, a domain thesaurus is used along with arbitrary complex phrase structures making up the so-called enterprise language. The prototype presented in this paper is based on a variation of the approach and is implemented as a plugin for a meta-modelling tool in order to support any conceptual modelling language. Hereby, additional efforts spent for *ex-post* resolving of naming conflicts and thereby induced costs become dispensable or can be reduced. Moreover, the domain thesaurus can contain nouns, verbs, adverbs, and adjectives, including descriptions of their exact meaning. Term descriptions can become important for the modeller to assure the proper understanding, whenever words are manually or automatically replaced. To automate the term replacement process, synonym relationships between terms are defined as well as one dominant term for each synonym set. The entered phrase is automatically validated against the associated phrase structures and the valid terms, without disrupting the modeller’s flow of knowledge externalization. The modeller is always provided with the necessary information about the preformed changes and the possibility to change any phrase element allowed by the enterprise language. Thus, only unified phrases can be produced while modelling.

The remainder of this paper is structured as follows: Section 2 covers related work and evaluates existing approaches. Moreover, design objectives are formulated against which a large variety of promising tools are evaluated. Section 3 introduces the prototype by first illustrating the adopted procedure model (Section 3.1). Subsequently, Section 3.2 covers the linguistic tagging mechanism. A demonstration of the plugin and the therein-applied enforcement of naming conventions are discussed in Section 3.3. An outlook concludes the paper in Section 4.

2 Related Work

The evaluation of related work is partitioned into two parts. First, related literature and approaches are identified and classified. Second, design objectives are formulated and tools, which are implementing promising approaches, are evaluated against these design objectives.

The identified approaches fall into two dimensions: the amount of terminologically standardised words (*single word* and *phrase structure*) and the time of the standardisation (*ex-ante* and *ex-post*). The first approaches which emerged in the 1980s were of *ex-post single word* related nature and were applied in the area of manual database schema integration and thus in languages such as Entity-Relationship Model (ERM) (Chen, 1976), in which an entity is usually named with one word only (Batini et al., 1986; Batini & Lenzerini, 1984; Bhargava, Kimbrough, & Krishnan, 1991; Lawrence & Barker, 2001; Rahm & Bernstein, 2001). As those approaches are conducted manually and only single nouns are considered, they are not suitable for all conceptual modelling languages. Several other approaches employ ontologies (Gruber, 1993; Guarino, 1998; Preece et al., 2001) for the organisation of terms and also to allow for expressing similarities and relationships between those terms. The approaches can be divided into two classes: The first class contains approaches, which follow the concept of generally accepted ontologies, which are known prior to modelling. These approaches assume that the modellers make use of these ontologies. An example of those *single word related* approaches which applied *ex-ante* term standardisation is the manual term adoption (Greco, Guzzo, Pontieri, & Saccà, 2004). Another similar, but semi-automated approach exists but is restricted to models of the Business Process Model and Notation (BPMN) (Born et al., 2007). The second class contains approaches, which are applying *ex-post* standardisation by deriving an ontology from models that have to be compared. Examples of those *ex-post phrase related* approaches are for example connecting domain ontologies to the terms used in the conceptual models in order to be able to analyse them subsequently (Höfferer, 2007). Other approaches of this class are furthermore making use of a similarity measure being composed of a semantic and a syntactic part, as the authors assume the terms can be considered semantically identical in case they are defined identically within the ontologies (Ehrig, Koschmider, & Oberweis, 2007; Koschmider & Oberweis, 2005; Sabetzadeh et al., 2007). Finally, *ex-ante phrase related* approaches most notably appeared in the 1990s and subsequently. One approach assigns business objects (nouns) and activities (verbs) to modelling elements (Nüttgens & Zimmermann, 1998). Other approaches recommend defining phrase structures as guidelines (e.g. <Verb, Imperative> <Noun, Singular>; such as “check invoice”) for modellers. The adoption of these guidelines, however, is not enforced (Kugeler, 2000; Rosemann, 1996). Leopold, Smirnov and Mendling (2012) extracted seven label structures consisting of nouns and verbs which are often used in process models and created an approach which is optimized for the refactoring into these structures. Delfmann, Herwig and Lis (2009) introduced an approach, which makes use of both terminological as well as arbitrarily complex phrase structures supporting an integrated enforcement during modelling. Summarising, it can be pointed out that *ex-ante* approaches seem most promising as the standardisation can be conducted during modelling which avoids subsequent unification efforts and costs. Moreover, phrase related approaches are usually more broadly applicable, since many conceptual models are process models, whose labels may contain many different word classes rather than only single words. The most promising approaches can be found in the category of the *ex-ante* phrase related class, most notably the approach of Delfmann, Herwig and Lis (2009) which makes use of a combination of terminological and phrase structural standardisation.

In order to efficiently support and realize the aforementioned approach, the following **design objectives** (DOs) can be formulated (cf. Delfmann et al. (2009)): (1) define terms (nouns, verbs, adverbs, adjectives) and synonym relationship between them, (2) define complex phrase structures (the so-called rules), (3) assign the rules to any modelling element type of any modelling language, (4) generate correct phrases out of the modeller’s input by replacing similar words with the (dominant) enterprise version and by applying the rules to the input, (5) generate any possible combination/collocation of words allowed by the defined rules, (6) guide the user in choosing appropriate words by sorting them according to similarity and explaining why a given term was replaced by another, (7) allow the modeller to send a phrase to a review process if s/he thinks that essential words or phrase structures s/he needs are missing in the domain thesaurus, (8) enforce the usage of the rules and the domain terms (making up the enterprise language), (9) perform standardization while modelling, (10) assist the initial creation of the domain thesaurus by extracting terms from existing models.

In order to evaluate the tool support for terminological standardisation in conceptual modelling tools, we composed a list of tools described in a survey paper on conceptual modelling in practice (Davies, Green, Rosemann, Indulska, & Gallo, 2006), in a paper on meta-modelling tools (Dietrich, Breuker, Steinhorst, Delfmann, & Becker, 2013), in a chapter on semantic modelling languages and tools (Becker, Probandt, & Vering, 2012; La Rosa et al., 2011), in a paper on managing process model complexity (La Rosa et al., 2011), in a survey on process management tools (Spath, Weisbecker, Kopperger, & Nägele, 2011) and in a Gartner survey on business process management suites (Sinur & Hill, 2010). The complete list comprises 77 tools, among which are commonly known tools like ARIS, Visio or Rational

Rose. From all evaluated tools only four offer support for terminological standardisation and are evaluated against the major design objectives. Table 1 depicts the results of the tool evaluation in which the tools have been evaluated against the aforementioned design objectives ((✓) is supported, (▫) is partly supported, (✖) is not supported). The last row of the table depicts the results of our meta-modelling tool plugin as a reference. A terminology can only be defined (DO1) by SemTalk (SemTalk, 2013) and icebricks (Becker, Clever, Holler, & Shitkova, 2013) via defining simple terms using a hierarchy between them, but without relations such as synonyms. PICTURE (Becker, Algermissen, Pfeiffer, & Räckers, 2007) and Signavio (Kunze & Weske, 2010) are not able to define any phrase structures, neither simple, nor complex (DO2). Both tools are only able to reuse complete sentences as process elements. SemTalk and icebricks are also not able to define complex phrase structures. The tools distinguish between business objects (nouns) and business activities (verbs) only. SemTalk furthermore distinguishes business activities into present tense for activities and past tense for events. None of the tools is able to define complex phrase structures containing any sort of predefined grammatical structure. The results show that no tool except SemTalk is able to handle arbitrary modelling languages (DO3) since they are not meta-modelling tools. Other meta-modelling tools have not been evaluated, as they do not provide any further functionalities concerning terminological standardisation to the best of our knowledge. Moreover, no tool is able to generate phrases using the enterprise language only (DO4) or allow different combinations of words within the rules (DO5). All tools also fall short of supporting the modeller in selecting the right terms in a given context (e.g. by using similarities) (DO6). Even though some tools provide term repositories, none of them provided the functionality to send a phrase to review in case the available terms are not sufficient (DO7). Furthermore, none of the tools forces the modeller to standardise the sentences (DO8). With respect to their functionalities, all tools allow to perform the respective tasks during modelling (DO9). Finally, no tool was able to extract terms for a domain thesaurus straight from the models (DO10).

The evaluation of tools reveals that none of them is able to resolve most of the aforementioned design objectives. Thus, this leads to the demand for an assisted way of creating consistent conceptual models by applying terminological standardization, which contributes to the existing theory and practice.

	DO1	DO2	DO3	DO4	DO5	DO6	DO7	DO8	DO9	DO10
Signavio	✖	✖	✖	✖	✖	✖	✖	✖	✓	✖
SemTalk	✓	▫	✓	✖	✖	✖	✖	✖	✓	✖
icebricks	▫	▫	✖	✖	✖	✖	✖	✖	✓	✖
PICTURE	✖	✖	✖	✖	✖	✖	✖	✖	✓	✖
Our prototype	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 1: Evaluation of existing tools.

3 Prototype

3.1 Adopted Procedure Model

The procedure model used within this prototype is based on the procedural approach presented by Delfmann, Herwig and Lis (2009). Thereby, an arbitrary amount of phrase structures (the so-called rules) can be assigned to each element type of any conceptual modelling language. The concept of maintaining a domain thesaurus defining terms and their synonyms is also reused within the plugin. However, the procedure of this plugin is considerably changed and adopted to the real life modelling challenges based on earlier experiences with incrementally improved prototypes (Hevner, March, Park, & Ram, 2004). Particularly, an entered sentence is not parsed but rather tagged (cf. Section 3.2).

The procedure of the terminological standardization is depicted in Figure 1. The illustration is simplified, because it assumes that all word replacements can be found via the domain thesaurus and the general

English lexicon. Other cases are discussed in Section 3.3. As soon as the modeller inputs the caption of a modelling element, such as a *BPMN Task*, the plugin loads the assigned rules (cf. step 1 in Figure 1). Then the sentence is parsed into individual words, which are then transformed into their base form (2). Subsequently, auxiliary words are removed and the base form words are getting all possible parts of speech assigned (3). In order to generate propositions of correct phrases to the modeller, a phrase builder tries to obtain correct words for each base form word and each part of speech (4). This is achieved by querying the domain thesaurus and a general English lexicon using WordNet (WordNet, 2013) via a wrapper project called Proxem Antelope (Proxem Antelope, 2013). Finally, (5) phrases are built by combining the replaced words and the previously obtained list of rules. While the standardization procedure depicted in Figure 1 is executed, the modelling process is halted. In our experience, this does not slow down the modeller considerably, because the standardization procedure can be executed in a few milliseconds.

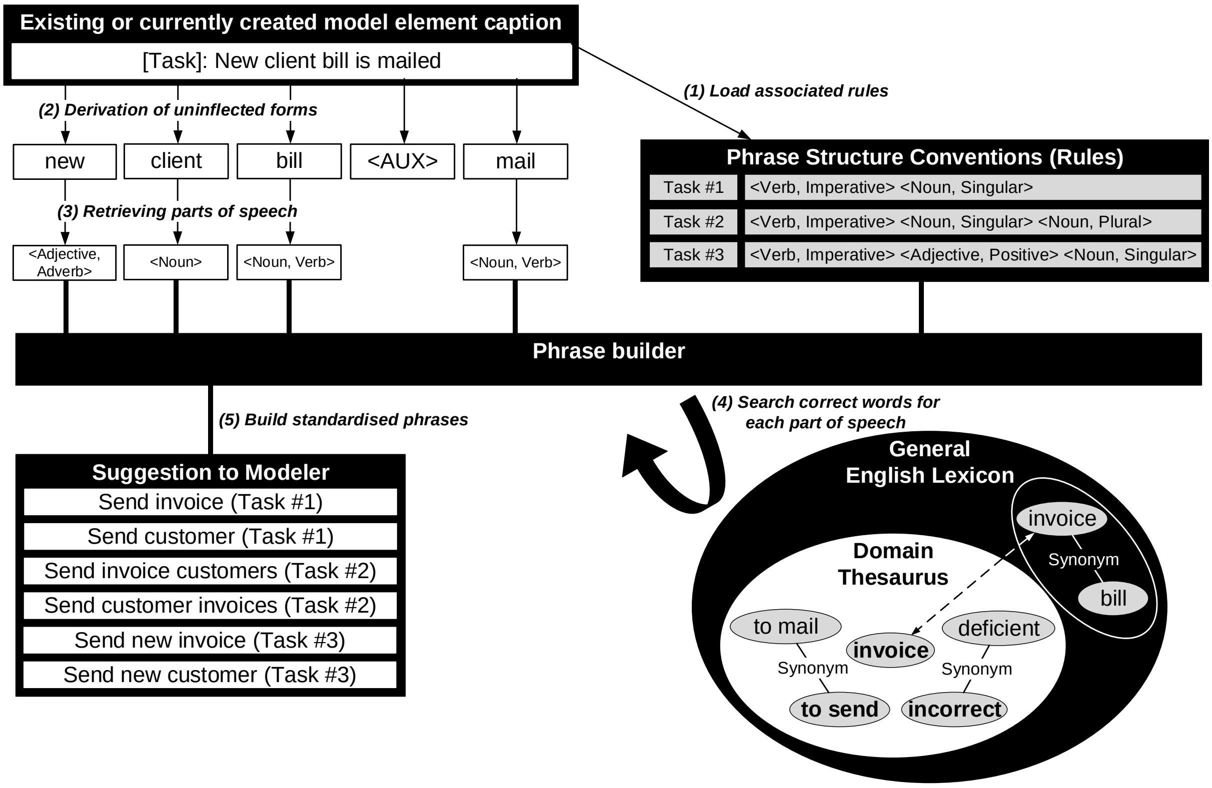

Figure 1. The high-level standardisation process of the plugin.

3.2 Assuring Naming Conventions using Linguistic Tagging

In order to allow automated terminological standardisation during the modelling, a tagger is necessary which is able to tag each single word of a sentence with several parts of speeches. The potential of natural language processing has been recently explored (Becker, Delfmann, Herwig, Lis, & Stein, 2009; Leopold, Smirnov, & Mendling, 2011). These approaches are employing natural language parsers like the Stanford Parser in order to obtain parts of speeches of each word in a sentence. However, many process models and data models only consist of rather short labels (Delfmann et al., 2009) which minimises the context of the labels and thus significantly reduces the correctness of the parsing result. In order to further evaluate these properties of parsers and also taggers three highly used representatives have been tested against a set of models. This list includes the Stanford Parser (Stanford Parser, 2013), LinkGrammar (Link Grammar, 2013) (e.g. used by word processing engines), and the SimpleTagger (Proxem Antelope, 2013). Additionally to the shortness of labels, the capitalisation of the sentence further affects the parsing result. Considering labels starting with capitalised words the evaluation showed that under these circumstances all parsers performed quite badly (*SimpleTagger* 34 %, *LinkGrammar* 31 %, and *Stanford Parser* 93 % misidentification rate). As words often have more than one part of speech, an insufficient context leads to the misidentification. The word "model" for example can either be a verb, a noun, or an adjective. Considering the definition layout for phrase structures (Kugeler, 2000; Ortner, 1997; Rosemann, 1996) the entered phrase "process model" for example could be either <Verb, Noun>, <Noun, Noun>, etc.. The parsers emit a result list ordered by costs using heuristics. Thus, the most probable solution has to be chosen, which is often not what the modeller intended to express by his possibly short sentence. As a consequence and due to the application domain, an own tagger, called *RelaxedTagger*, was developed which suits the needs of the terminological standardization the most. The *RelaxedTagger* does not output a single suggested phrase structure, but retrieves all possible parts of speech for each entered word. This permits greater flexibility in defining standardized sentences, as a word cannot be misinterpreted. It is furthermore assured that the modeller can always derive correctly standardized sentences from its input. In order to obtain all possible parts of speech two different thesauri, first the domain thesaurus and afterwards the natural language lexicon (WordNet), are used. As the meaning of the word contained in the domain specific thesaurus is more important than the manifold meanings within the natural language lexicon, its part of speech is considered first. If a word was found in the domain thesaurus, it is not queried against the natural language lexicon anymore. If the word "model" is for example contained as verb and as noun in the domain thesaurus, the part of speech "Adjective" would not be retrieved from the natural language lexicon anymore. Furthermore, the *RelaxedTagger* is able to handle domain thesaurus terms consisting of multiple words. This means that entering the words "SAP system" somewhere within the sentence, does not return the possible parts of speech {"Verb", "Noun"} for SAP and {"Noun"} for system, but identifies "SAP system" as an own word (noun), when contained in the domain thesaurus.

3.3 Plugin Demonstration

This section provides a demonstration of all the required steps within the standardisation plugin. This includes creating a meta-modelling language, defining domain thesaurus terms and phrase rules, assigning these rules to selected element types and finally adding modelling elements to a model, which invokes the step of standardising their caption. Moreover, each single feature is linked to the design objectives (cf. Section 2). The plugin was developed for a conceptual meta-modelling tool from a former research project (Delfmann, Herwig, Karow, & Lis, 2008). Within the meta-modelling tool a conceptual modelling language (e.g. *BPMN*) can be created. To do so, objects and relationship types need to be defined. Furthermore, shapes and other details such as attributes can be assigned. Further details are omitted as the creation of the meta-modelling language is not the main part of this paper but is required for the later steps. However, it is important to note that the standardisation is not limited to certain modelling languages but can be used with any modelling language (cf. DO3).

Setting up the environment. Subsequently, the domain thesaurus needs to be built up (cf. DO1). This phase is done once for a modelling project or for the whole enterprise. The domain thesaurus can be created from scratch or an existing one can be imported. When adding a word, the plugin automatically tries to determine which part of speech was entered. In case the heuristic was wrong this can always be changed. Figure 3 in the appendix shows the creation of the domain thesaurus. Instead of manually creating the domain thesaurus terms, the term extraction functionality can also be used (cf. DO10). In doing so, terms can quickly be extracted from existing models representing the enterprises vocabulary. Subsequently, the synonyms of the terms have to be defined in order to replace entered terms by dominant enterprise terms (cf. DO1 and Figure 5). Once the terms and their synonyms are captured, the phrase structures (rules) need to be defined (cf. DO2, e.g. cf. Figure 1 and Figure 6). A name is assigned to the rule and furthermore allowed additional words can be defined which can be seen as some additional freedom which can be given to the modeller. Phrase structure elements can be added and moved around and the details of the elements can be specified. Once the rules are defined they are assigned to an element type (e.g. *BPMN Task*) of the conceptual modelling language (cf. DO3). The domain thesaurus as well as the phrase structures should be defined prior to modelling to assure that naming conventions are applied throughout all models.

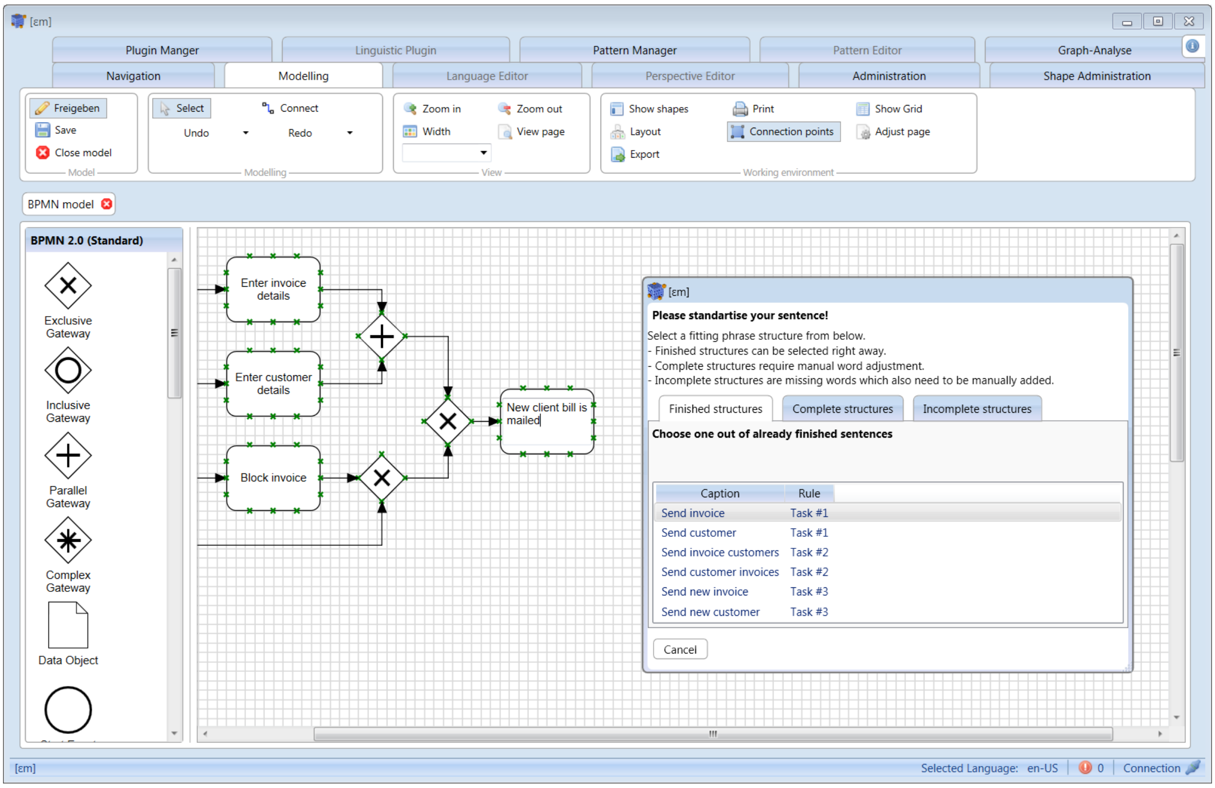

Figure 2. The standardisation dialogue within the plugin interface (part I).

Standardizing phrases. Once all preparations for the terminological standardisation are made a model is created or an existing one is opened. In case the modeller adds a new modelling element, such as *BPMN Task*, together with a new caption, a new standardisation dialogue opens up as depicted in Figure 2 or Figure 8 (cf. DO8 and DO9). Alternatively, elements of existing models can be checked for their standardisation by similarly automated means. Other than generating valid phrases, a main idea is to provide enough flexibility while restricting the space of possible sentences. Flexibility hereby refers to the modeller's flexibility in choosing terms and phrases in order to make sure s/he is able to express what s/he intends. Restriction on the other hand refers to only allowing the rules associated with the respective modelling element. The simplified approach from Section 3.1 definitely assures that only correct phrases can be inserted into a conceptual model, but on the other hand, this simplified approach might possibly be too restrictive and might hinder a one-to-one knowledge externalization from the modeller's expertise into the model. Therefore, in order to provide the modeller with all necessary flexibility, the proposed phrases are tripartite: The first set of phrases only contains phrases which can be built without any further interaction of the modeller. This is done, by replacing words with their synonyms if available and by applying those to the rules assigned to the respective element type (cf. DO4). If the modeller chooses one of these propositions, no further action needs to be taken and thus the sentence is terminologically standardised. The two other sets of propositions on the other hand require some adjustment of the modeller. The second set contains phrases for which enough relevant words with a matching part of speech were provided. This means that the modeller is required to manually replace some words. One reason thereto is that more than one synonym was found from which the modeller has to choose the one which fits his intended expression best. Another reason is that no synonym could be found in the domain thesaurus or in the natural language lexicon. In this case, the modeller is provided with a list of all possible words of the same part of speech from the domain thesaurus. In order to facilitate the selection of the appropriate word, the listed words are sorted by their maximum similarity measure to the entered word (cf. DO6).

Finally, the last set of propositions contains incomplete sentences for which the amount of words was not enough. In this case, the missing words are indicated with placeholders for whom the modeller needs to choose a word from the domain thesaurus. The modeller also has the possibility to compose a new sentence using an empty phrase structure. Furthermore, in case of the last two sets requiring an interaction, the modeller is only able to save the sentence if all words of the rules are resolved which results in a standardized sentence. Moreover, the modeller can obtain information about the changed words and the necessary actions via a tooltip, which is available for each word (cf. DO6). Altogether, in all three possible cases it is not possible to generate a sentence, which is not terminologically standardised. Moreover, the generation of all of the sets considers all possible combinations of words and rules (cf. DO5), which makes it impossible for the modeller to find himself with an undesirable order of words. Providing the three different sets of correct phrase propositions allows the modeller to either easily chose from a selection of already standardised phrases or to change or compose further details of the sentence himself. This makes sure that all saved sentenced are always standardised. On the other hand the plugin provides the modeller with all necessary flexibility, as s/he can basically compose a completely new sentence from the third set of phrases. As a final case, the modeller has the possibility to send a completely manually edited phrase to review in case s/he thinks that there are necessary words or rules missing within the enterprise language. To that end, the plugin also comprises a *Review* section in which the responsible person can review and extract new words from the phrases and add them to the domain thesaurus (cf. DO7, DO10). Furthermore, an existing model can be terminologically standardised. To that end, the standardization procedure can be manually triggered. In doing so, all labels of all model elements are iteratively checked against the domain thesaurus and existing phrase structures. Where required, labels are changed accordingly. If necessary, all possible phrase structures are returned to the modeller who has to select the correct one.

4 Outlook

In this paper, we introduced a prototype which is able to enforce terminological standardisation of labels contained in conceptual models by using a domain thesaurus and a repository of arbitrarily complex phrase structure rules. Thereby, the modeller is supported in producing standardised model labels without restricting flexibility and expressiveness. The prototype is currently at its first level of maturity. In the near future further iterative developments and improvements are planned. An evaluation of the efficiency and the quality of produced models using this prototype is planned. Therefore, one group uses the plugin for the creation of standardised models while another reference group uses only the tool itself and the guidelines provided in the case. Future research will also focus on determining the feasibility and effort of creating a domain thesaurus. We will also use existing process repositories to test the applicability of the *ex-post* standardization feature.

References

Batini, C. and Lenzerini, M. (1984). A Methodology for Data Schema Integration in the Entity Relationship Model. IEEE Transactions on Software Engineering, SE-10(6), 650–664.

Batini, C., Lenzerini, M., and Navathe, S.B. (1986). A comparative analysis of methodologies for database schema integration. ACM Computing Surveys, 18(4), 323–364.

Becker, J., Algermissen, L., Pfeiffer, D., and Räckers, M. (2007). Bausteinbasierte Modellierung von Prozesslandschaften mit der PICTURE-Methode am Beispiel der Universitätsverwaltung Münster. Wirtschaftsinformatik, 49(4), 267–279.

Becker, J., Clever, N., Holler, J., and Shitkova, M. (2013). icebricks - Business Process Modeling on the Basis of Semantic Standardization. In Design Science at the Intersection of Physical and Virtual Design (Vom Brocke, J., Hekkala, R., Ram, S., & Rossi, M. Eds.), pp. 394–399. Helsinki, Finland.

Becker, J., Delfmann, P., Herwig, S., Lis, Ł., and Stein, A. (2009). Towards increased comparability of conceptual models - Enforcing naming conventions through domain thesauri and linguistic grammars. In Proceedings of the 17th European Conference on Information Systems (ECIS) (De Marco, M., Loebbecke, C., & Willcocks, L. Eds.), pp. 2231–2242. Verona, Italy.

Becker, J., Probandt, W., and Vering, O. (2012). Die Grundsätze ordnungsmäßiger Modellierung in semantischen Modellierungssprachen. In Grundsätze ordnungsmäßiger Modellierung, pp. 215–233. Springer, Berlin.

Bhargava, H.K., Kimbrough, S.O., and Krishnan, R. (1991). Unique Names Violations, a Problem for Model Integration or You Say Tomato, I Say Tomahto. ORSA Journal on Computing, 3(2), 107–120.

Born, M., Dörr, F., and Weber, I. (2007). User-friendly Semantic Annotation in Business Process Modeling. In Web Information Systems Engineering – WISE 2007 Workshops (Weske, M., Hacid, M.-S., & Godart, C. Eds.), pp. 260–271. Springer, Berlin.

Chen, P.P.-S. (1976). The Entity-Relationship Model - Toward a Unified View of Data. ACM Transactions on Database Systems, 1(1), 9–36.

Davies, I., Green, P., Rosemann, M., Indulska, M., and Gallo, S. (2006). How do practitioners use conceptual modeling in practice? Data & Knowledge Engineering, 58(3), 358–380.

Delfmann, P. (2010). Verteilte Informationsmodellierung - Methodische technische Konzepte zur Disambiguierung und Analyse arbeitsteilig entwickelter Informationsmodelle. University of Muenster.

Delfmann, P., Herwig, S., Karow, M., and Lis, Ł. (2008). Ein konfiguratives Metamodellierungswerkzeug. In Modellierung betrieblicher Informationssysteme (Loos, P., Nüttgens, M., Turowsk, K., & Werth, D. Eds.), pp. 109–127. GI, Saarbrücken, Germany.

Delfmann, P., Herwig, S., and Lis, Ł. (2009). Unified Enterprise Knowledge Representation with Conceptual Models - Capturing Corporate Language in Naming Conventions. In International Conference on Information Systems (ICIS 2009) (Jay F., N. J. & Currie, W. L. Eds.). Association for Information Systems, Phoenix, Arizona, USA.

Dietrich, H.-A., Breuker, D., Steinhorst, M., Delfmann, P., and Becker, J. (2013). Developing Graphical Model Editors for Meta-Modelling Tools - Requirements, Conceptualisation, and Implementation. Enterprise Modelling and Information Systems Architectures, 8(2), 41–77.

Dijkman, R.M., La Rosa, M., and Reijers, H.A. (2012). Managing Large Collections of Business Process Models - Current Techniques and Challenges. Computers in Industry, 63(2), 91–97.

Ehrig, M., Koschmider, A., and Oberweis, A. (2007). Measuring Similarity Between Semantic Business Process Models. In Asia-Pacific Conference on Conceptual Modelling (APCCM2007) (Roddick, J. F. & Hinze, A. Eds.), pp. 71–80. Australian Computer Society, Ballarat, Victoria, Australia.

Greco, G., Guzzo, A., Pontieri, L., and Saccà, D. (2004). An Ontology-Driven Process Modeling Framework. In Database and Expert Systems Applications (DEXA 2004) (Galindo, F., Takizawa, M., & Traunmüller, R. Eds.), pp. 13–23. Springer Berlin Heidelberg.

Gruber, T.R. (1993). A translation approach to portable ontology specifications. Knowledge Acquisition, 5(2), 199–220.

Guarino, N. (1998). Formal Ontology and Information Systems. In Formal Ontology in Information Systems (Guarino, N. Ed.), pp. 3–15. IOS Press, Amsterdam, The Netherlands.

Hevner, A.R., March, S.T., Park, J., and Ram, S. (2004). Design science in information systems research. MIS Quarterly, 28(1), 75–105.

Höfferer, P. (2007). Achieving business process model interoperability using metamodels and ontologies. In Proceedings of the 15th European Conference on Information Systems (ECIS 2007)

(Österle, H., Schelp, J., & Winter, R. Eds.), pp. 1620–1631. University of St. Gallen, St. Gallen, Switzerland.

Koschmider, A. and Oberweis, A. (2005). Ontology based Business Process Description. In EMOI - INTEROP’05, Enterprise Modelling and Ontologies for Interoperability, Proceedings of the Open Interop Workshop on Enterprise Modelling and Ontologies for Interoperability, pp. 321–333. Porto, Portugal.

Kugeler, M. (2000). Informationsmodellbasierte Organisationsgestaltung. Logos, Berlin.

Kunze, M. and Weske, M. (2010). Signavio-Oryx Academic Initiative. In BPM 2010 Demonstration Track (La Rosa, M. Ed.), pp. 6–10. Hoboken, New Jersey, USA.

La Rosa, M., Dumas, M., Uba, R., and Dijkman, R.M. (2010). Merging Business Process Models. In On the Move to Meaningful Internet Systems: OTM 2010 (Meersman, R., Dillon, T., & Herrero, P. Eds.), pp. 96–113. Springer Berlin Heidelberg, Crete, Greece.

La Rosa, M., Dumas, M., Uba, R., and Dijkman, R.M. (2013). Business Process Model Merging: An Approach to Business Process Consolidation. ACM Transactions on Software Engineering and Methodology, 22(2), 1–42.

La Rosa, M., ter Hofstede, A.H.M., Wohed, P., Reijers, H. a., Mendling, J., and van der Aalst, W.M.P. (2011). Managing Process Model Complexity via Concrete Syntax Modifications. IEEE Transactions on Industrial Informatics, 7(2), 255–265.

Lawrence, R. and Barker, K. (2001). Integrating Relational Database Schemas Using a Standardized Dictionary. In Proceedings of the 2001 ACM Symposium on Applied Computing, pp. 225–230. ACM, New York, NY, USA.

Leopold, H., Smirnov, S., and Mendling, J. (2011). Recognising Activity Labeling Styles in Business Process Models. Enterprise Modelling and Information Systems Architectures, 6(1), 16–29.

Leopold, H., Smirnov, S., and Mendling, J. (2012). On the refactoring of activity labels in business process models. Information Systems, 37(5), 443–459.

Link Grammar. (2013). Link Grammar. Retrieved October 28, 2013, from http://www.link.cs.cmu.edu/link/.

Nüttgens, M. and Zimmermann, V. (1998). Geschäftsprozeßmodellierung mit der objektorientierten Ereignisgesteuerten Prozeßkette (oEPK). In Informationsmodellierung (Maicher, M. & Scheruhn, H.-J. Eds.), pp. 23–35. Deutscher Universitätsverlag.

Ortner, E. (1997). Methodenneutraler Fachentwurf: Zu den Grundlagen einer anwendungsorientierten Informatik. Vieweg+Teubner, Stuttgart.

Preece, A., Flett, A., Sleeman, D., Curry, D., Meany, N., and Perry, P. (2001). Better Knowledge Management Through Knowledge Engineering. IEEE Intelligent Systems, 16(1), 36–43.

Proxem Antelope. (2013). Antelope Framework. Retrieved August 28, 2013, from https://www.proxem.com/en/technology/antelope-framework/.

Rahm, E. and Bernstein, P.A. (2001). A survey of approaches to automatic schema matching. The VLDB Journal, 10(4), 334–350.

Rosemann, M. (1996). Komplexitätsmanagement in Prozeßmodellen: Methodenspezifische Gestaltungsempfehlungen für die Informationsmodellierung. Gabler, Wiesbaden.

Sabetzadeh, M., Nejati, S., Easterbrook, S., and Chechik, M. (2007). A Relationship-Driven Framework for Model Merging. In International Workshop on Modeling in Software Engineering (MISE’07: ICSE Workshop 2007). IEEE, Minneapolis, MN, USA.

SemTalk. (2013). SemTalk. Retrieved August 28, 2013, from http://www.semtalk.com.

Sinur, J. and Hill, J.B. (2010). Magic Quadrant for Business Process Management Suites. Gartner Research.

Spath, D., Weisbecker, A., Kopperger, D., and Nägele, R. (2011). Business Process Management Tools. Fraunhofer-Verlag, Germany.

Stanford Parser. (2013). The Stanford Parser: A statistical parser. Retrieved August 28, 2013, from http://nlp.stanford.edu/software/lex-parser.shtml.

Steinhorst, M., Breuker, D., Delfmann, P., and Dietrich, H.-A. (2012). Supporting Enterprise Transformation Using a Universal Model Analysis Approach. In Proceedings of the 20th European Conference on Information Systems (ECIS) 2012.

WordNet. (2013). WordNet - A lexical database for English. Retrieved August 28, 2013, from http://wordnet.princeton.edu.

Appendix

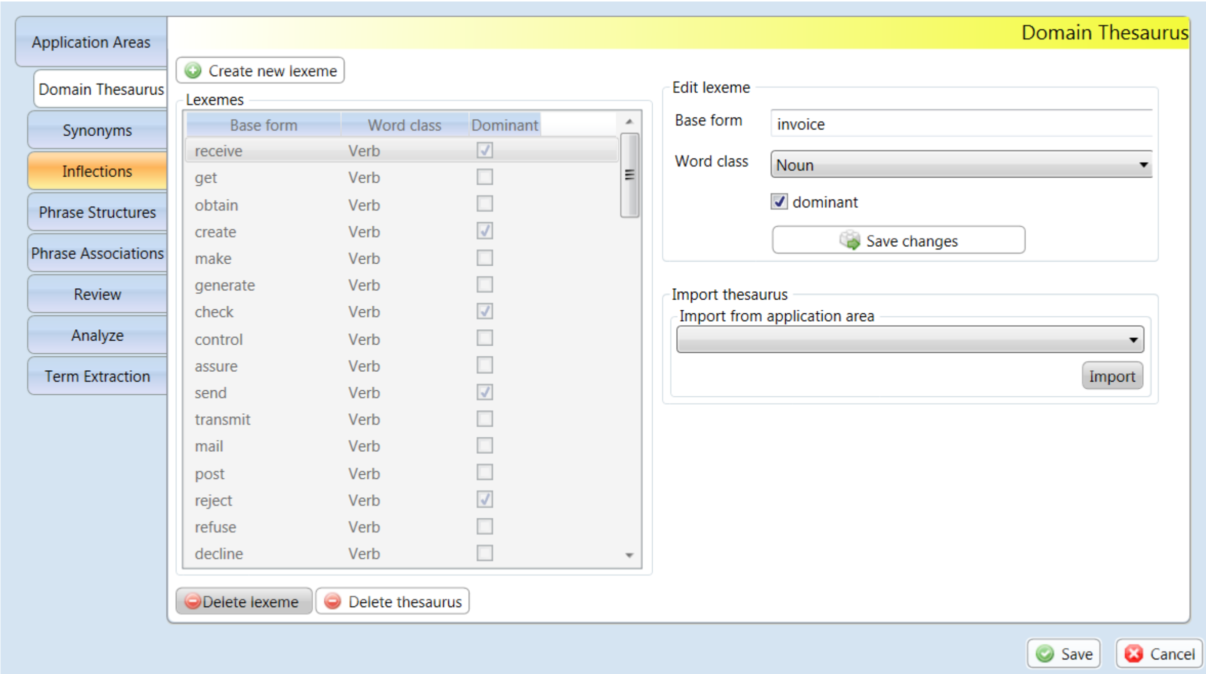

Figure 3. *Building the domain thesaurus.*

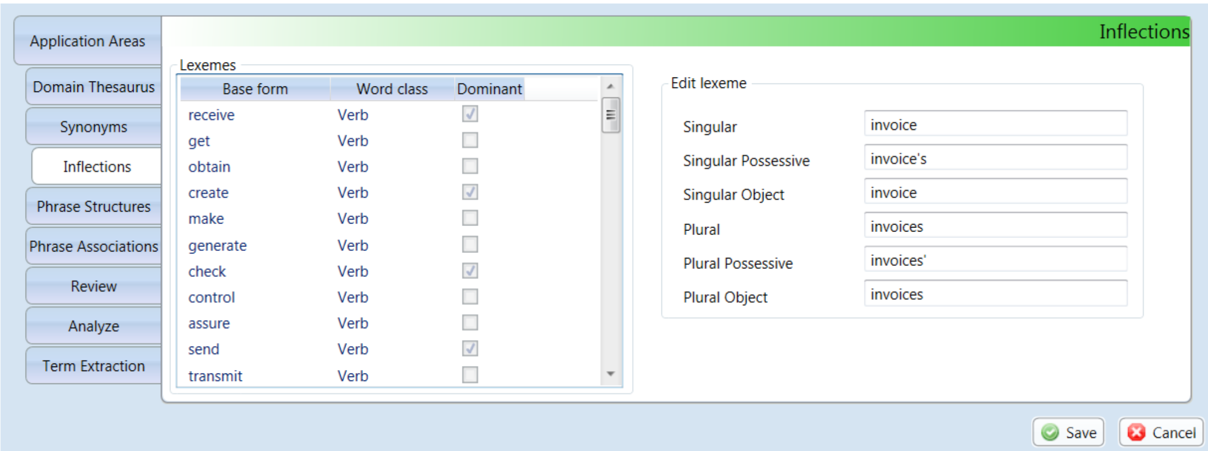

Figure 4. *Checking the automatically created inflections.*

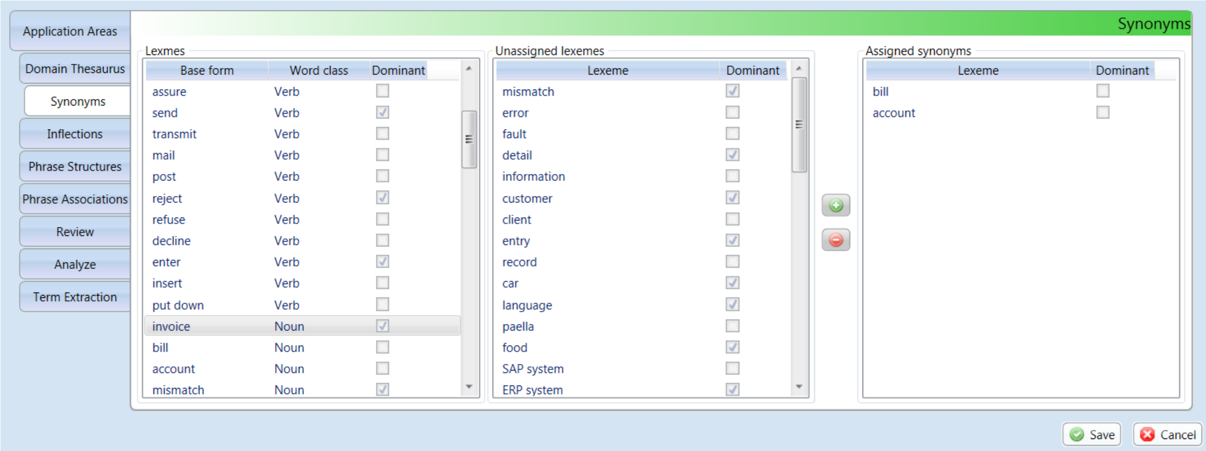

Figure 5. Associating synonyms.

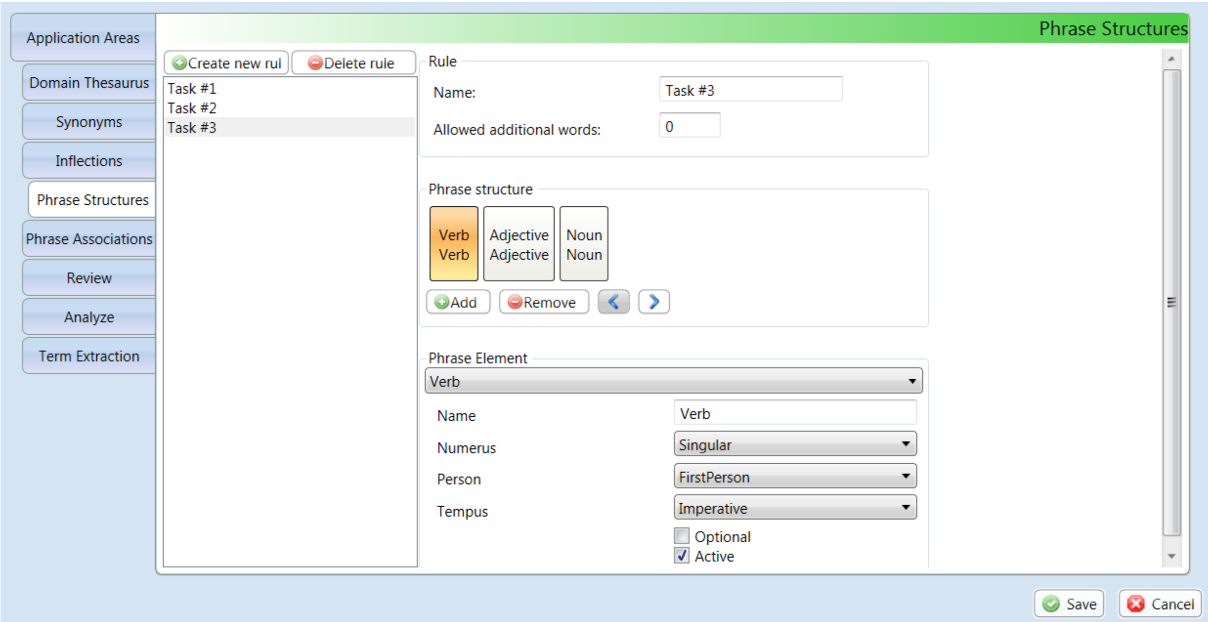

Figure 6. Creating phrase structures (rules).

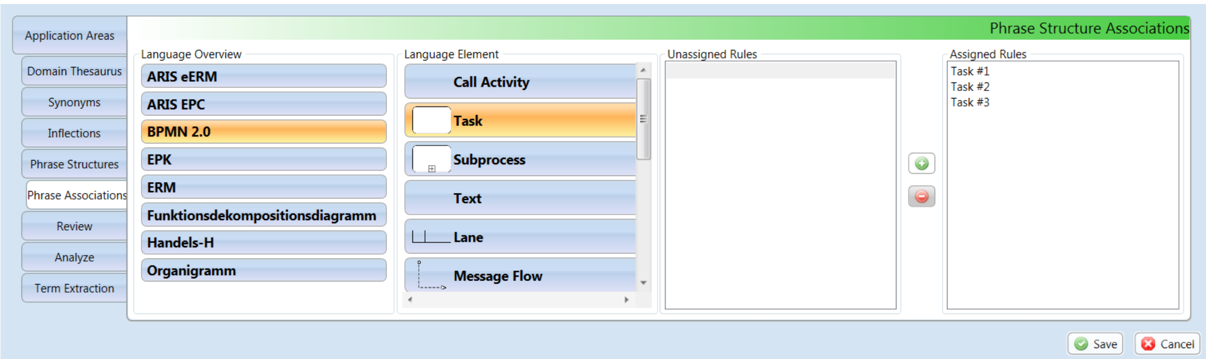

Figure 7. Associating phrase structures (rules) to modelling elements

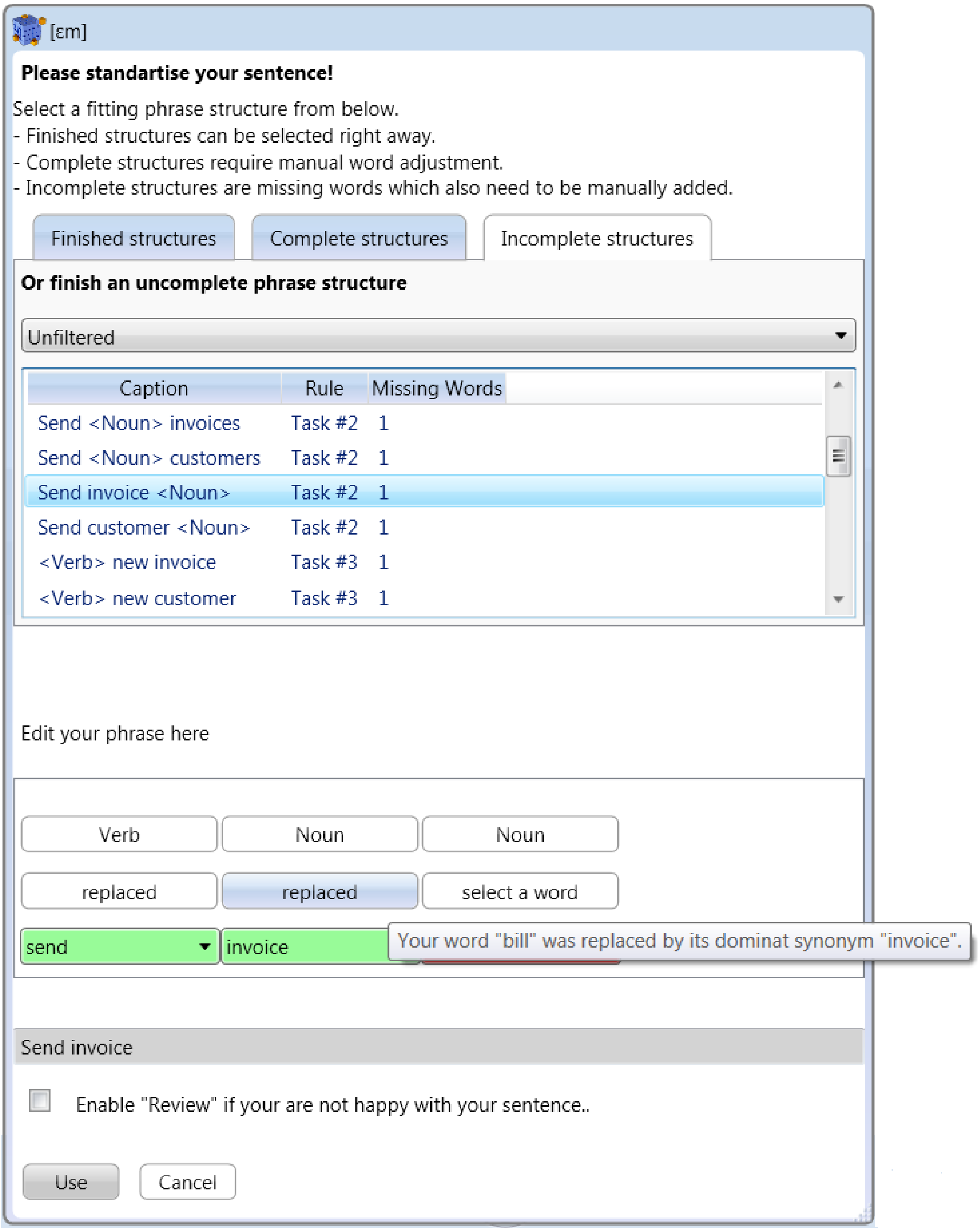

Figure 8. *The standardisation dialogue (part II).*