$\mathcal{K}$-$\mathcal{MORPH}$: A Semantic Web Based Knowledge Representation and Context-driven Morphing Framework

Sajjad Hussain[1] and **Syed Sibte Raza Abidi**[1]

Abstract. A knowledge-intensive problem is often not solved by an individual knowledge artifact; rather the solution needs to draw upon multiple, and even heterogeneous, knowledge artifacts. Each knowledge artifact may differ in terms of its modality, origin, and format; and may have different functional/operational roles in different problem-contexts. The synthesis of multiple knowledge artifacts to derive a 'comprehensive' knowledge artifact is a non-trivial problem. In this paper, we propose a semantic web based knowledge representation and morphing framework $\mathcal{K}$-$\mathcal{MORPH}$ that (a) semantically models the knowledge of various knowledge artifacts found in different modalities as ontologies; (b) semantically annotates the heterogeneous knowledge artifacts based on their respective ontologies; (c) represents the domain-specific constraints and specifications for the morphed knowledge, and treats them as a problem-context; (d) defines morphing constructs, to identify problem-specific knowledge components from the entire knowledge artifacts; (e) reconciles related knowledge components; and (f) generates a verified 'morphed' knowledge artifact that contains reconciled problem-specific knowledge from multiple artifacts. We discuss the architecture of a prototype medical knowledge morpher to show the need of knowledge morphing in medical domain.

1 INTRODUCTION

Knowledge originates in an assortment of knowledge artifacts–each artifact captures specific conceptual, contextual, functional and operational aspects of an underlying domain. For our purposes, we aim to apply knowledge for decision support and planning purposes. We argue that, central to knowledge-centric activities is the need to 'reason' over all available knowledge artifacts in order to (a) infer new knowledge, (b) test hypotheses, (c) suggest recommendations and actions, and (d) query rules to prove problem-specific assertions or theorems. The challenge, therefore, is to allow the reasoning process to simultaneously operate over multiple heterogeneous knowledge sources in order to derive a comprehensive reasoning outcome that builds on the different problem-specific perspectives dispersed across multiple knowledge artifacts that may differ in terms of modality and functional intensions. This challenge leads to the concept of 'knowledge morphing' that is defined as "the intelligent and autonomous fusion/integration of contextually, conceptually and functionally related knowledge objects that may exist in different representation modalities and formalisms, in order to establish a comprehensive, multi-faceted and networked view of all knowledge pertaining to a domain-specific problem"–Abidi 2005 [1].

From our perspective, which deals mainly with healthcare decision support [10], a knowledge artifact is basically a knowledge object, having a defined representation formalism, that encapsulates a specific kind of knowledge. The key knowledge modalities that we deal with are (a) explicit knowledge that is represented in terms of the following knowledge artifacts–clinical practice guidelines, clinical pathways and medical literature [2, 9]; (b) experiential knowledge represented as past cases and medical records; (c) observational knowledge that is derived from operational data and represented as data models and induced rules. Our knowledge morphing solution aims to synthesize these different knowledge artifacts, as per the problem description–i.e. the problem's *context*.

In this paper, we present our approach to pursue knowledge morphing. We propose a Semantic Web based Knowledge Morphing framework $\mathcal{K}$-$\mathcal{MORPH}$ that focuses on two aspects:

1. *Knowledge Representation*: Domain-specific knowledge representation is achieved through the use of *ontologies*. For each type of knowledge artifact we have developed a specific ontology that firstly models the generic structure (i.e. the form) of the knowledge artifact and then encodes the knowledge inherent within the artifact (i.e. its function) as an instance of the ontology [2, 9]. Representation of the problem that is mitigating knowledge morphing is pursued through the definition of a *problem-context* that encapsulates the problem specification–i.e. input and output elements, intension of the solution based on the morphed knowledge and domain-specific constraints.
2. *Context-driven Knowledge Morphing*: The knowledge morphing process comprises three main tasks: (i) specification of the *morphing construct* that explicitly defines the morphing intension, potential problem-specific knowledge constructs within the candidate knowledge artifacts, problem-context and morphing functions; (ii) knowledge morphing through *ontology reconciliation* based on a *proof-level ontology alignment* mechanism that synthesizes multiple artifact-specific ontology sub-constructs to yield a comprehensive multi-facted morphed knowledge object; and (iii) *validation and verification* of the morphed knowledge.

2 Knowledge Morphing

In principle, knowledge morphing aims to generate a comprehensive knowledge artifact with respect to a specific problem context. In practice, knowledge morphing aims to reconcile multiple knowledge resources–in our case knowledge artifacts represented as dis-

---

[1] NICHE Research Group, Faculty of Computer Science, Dahousie University, Canada, email: {hussain, sraza}@cs.dal.ca

tinct ontologies–to generate a morphed knowledge artifact. We argue that typically knowledge artifacts entail knowledge that is broader then a specific problem's scope. For instance, in healthcare a clinical guideline may contain knowledge about the diagnosis, treatment, prognosis and follow-up care for a particular disease. Therefore, we posit that the integration of entire knowledge artifacts unnecessarily exacerbates the complexity of establishing interoperability between multiple artifacts for no meaningful purpose. Rather, our approach for knowledge morphing follows three steps: (i) identify the knowledge components (or sub-artifacts) within a knowledge artifact that are pertinent towards the given problem description; (ii) extract the identified sub-artifacts as candidate constructs for knowledge morphing. Given that the original knowledge artifacts are represented as ontologies, the sub-artifacts will be represented as sub-ontologies that are validated for conceptual consistency and completeness; and (iii) reconcile or align the sub-ontologies to generate a new sub-ontology that represents the 'morphed' knowledge artifact as shown in Figure 1. In this way, our knowledge morphing approach pursues highly-specific ontology alignment guided by the problem's context–i.e. a single knowledge morphing context (akin to a query) forms the basis of the process. This also means that as the problem context changes a new morphed knowledge artifact will be developed. The re-usability of morphed knowledge is another interesting problem that we will be subsequently investigating.

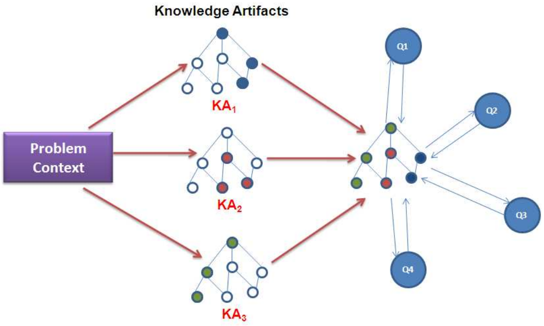

Figure 1. Knowledge Morphing

It may be noted that the literature suggests other approaches to knowledge morphing problem from different perspectives. ECOIN is one notable framework that performs semantic reconciliation of independent data sources, under a defined context [7]. Semantic reconciliation is performed at the context level by defining *conversion functions* between contexts as a network. ECOIN approach believes on the *single ontology, multiple views* notion [7], and introduces the notion of *modifiers* to explicitly describe the multiple specializations/views of the concepts used in different data sources. It exploits the modifiers and conversion functions, to enable context mediation between data sources, and reconcile and integrate source schemas with respect to their conceptual specializations. Another recent initiative towards knowledge morphing is the OpenKnowledge project [5]. The OpenKnowledge framework supports the knowledge sharing among different knowledge artifacts, not by sharing their asserted statements, instead by sharing their *interaction models*. An interaction model provides a context in which knowledge can be transmitted between two (or more) knowledge sources (peers). This approach has a closer relevance with semantic service composition [8], where each interaction model (stands for a knowledge source) can be seen as a service that interacts with other services based on their service descriptions and business logics.

3 $\mathcal{K}$-$\mathcal{MORPH}$ ARCHITECTURE

We adopt a Semantic Web (SW) architecture [3] to address the problem of knowledge morphing. Given that at the core of knowledge morphing is the need to semantically model the different knowledge artifacts, we believe that the SW offers a logic-based framework to (a) semantically model the knowledge of various knowledge artifacts found in different modalities as ontologies; (b) semantically annotate the heterogeneous knowledge artifacts based on their respective ontologies; (c) capture and represent the underlying domain concepts, and the semantic relationships that are inherent within a problem-context, in terms of a domain ontology; (d) ensure interoperability between multiple ontologically defined knowledge artifacts; and (e) maintaining changes, evolution and management of ontologies.

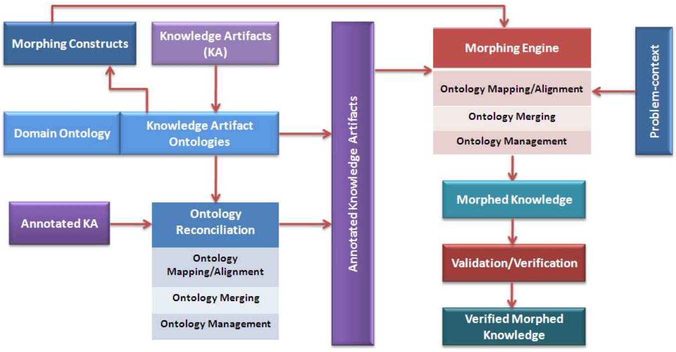

Figure 2. High-level schematic of $\mathcal{K}$-$\mathcal{MORPH}$

$\mathcal{K}$-$\mathcal{MORPH}$ comprises the following elements (see Figure 2).

1. *Domain Ontology* is used to capture and represent explicit domain knowledge in terms of generic and standardized concepts.
2. *Knowledge Artifact Ontologies* represent the structure and content of different knowledge artifacts–each knowledge artifact type is represented by its unique knowledge artifact ontology. These ontologies are both guided by and reflect the domain ontology.
3. *Knowledge Artifact Annotation* is the process to annotate the content of a knowledge artifact with respect to its corresponding knowledge artifact ontology. An annotated knowledge artifact is called an *Ontology-encoded Knowledge Artifact* (OKA).
4. *Morphing Constructs* specify the problem of knowledge morphing for a given context in terms of declarative knowledge dictating how to operate with the available knowledge artifacts to derive the problem-specific knowledge components (or sub-ontologies).
5. *Ontology Reconciliation* process involves the alignment of two (or more) candidate OKAs to yield morphed knowledge.
6. *Reconciliation of Other Annotated Ontologies* deals with knowledge that is annotated using other ontologies and attempts to find correspondences between ontologies, and then mapping the annotated ontologies into knowledge artifact ontologies [6].
7. *Morphing Engine* is the main component that handles the knowledge morphing process through proof-level ontology alignment. It takes as input a problem-context, OKAs and morphing constructs and then performs the following:

    (a) Identifies the knowledge components in a knowledge artifact ontology that have relevance with the problem-context.
    (b) Maps/aligns all identified knowledge components.
    (c) Finds inconsistencies in aligned knowledge components.
    (d) Merges knowledge components via merging rules.

8. *Validation and Verification*: The morphed knowledge can be validated by employing proof engines, and verified against the expert knowledge.

The above-mentioned $\mathcal{K}$-$\mathcal{MORPH}$ elements are described below.

3.1 Knowledge Representation and Annotation via Ontologies

In $\mathcal{K}$-$\mathcal{MORPH}$, a necessary step for knowledge morphing is to pursue knowledge formalization in order to support domain-specific inferencing based on declarative and procedural knowledge. Declarative knowledge describes the domain concepts, potential problems and probable solutions. Such declarative knowledge can be causal, qualitative, descriptive or quantitative. Procedural knowledge describes how to apply the knowledge to actually solve domain-specific problems, whilst taking into account, and satisfying the unique operational constraints of a domain-specific institution.

We use ontologies to model a knowledge artifact as it allows (i) formalization of domain-specific knowledge; (ii) conceptualization of the knowledge along declarative and procedural dimensions; (iii) annotation of the knowledge based on an ontological model; (iv) reuse and evolution of the knowledge; (v) use of standard terms and concepts; and (vi) identification of similar knowledge components that can potentially be aligned to achieve knowledge morphing. For our purposes, an Ontology is formally defined as follows:

Definition 1 (Ontology) *Let $\mathcal{V}$ be the set of structured vocabulary, and $\mathcal{A}_x$ be the set of axioms about $\mathcal{V}$, which are formulated in formal language $\mathcal{L}$. An ontology O is defined by the following tuple:*

$$O := \langle \mathcal{L}, \mathcal{V}, C, H_C, R, H_R, I, \mathcal{A}_x \rangle$$

where, concepts $C \subseteq \mathcal{V}$ of the schema are arranged in a subsumption hierarchy H_C. Binary relations $R \subseteq \mathcal{V}$ exist between pairs of concepts. Relations can also be arranged in a subsumption hierarchy H_R. (Meta-)Data is constituted by instances $I \subseteq \mathcal{V}$ of specific concepts. Additionally, one can define axioms $\mathcal{A}_x = \mathcal{L}(\mathcal{V})$ which can be used to infer knowledge from already asserted knowledge.

An Ontology $O' := \langle \mathcal{L}, \mathcal{V}, C', H'_C, R', H'_R, I', \mathcal{A}'_x \rangle$ is a sub-ontology of O, where $C' \subseteq C, H'_C \subseteq H_C, R' \subseteq R, H'_R \subseteq H'_R, I' \subseteq I, \mathcal{A}'_x \subseteq \mathcal{A}_x$; and written as $O' \prec O$.

3.1.1 Domain Ontology and Knowledge Artifact Ontology

In $\mathcal{K}$-$\mathcal{MORPH}$, knowledge artifacts are represented using two different (but inter-related) ontologies, namely: (i) *Domain Ontology*; and (ii) *Knowledge Artifact Ontology*. A domain ontology serves as a high-level ontology that describes the fundamental concepts of the domain–i.e. declarative knowledge. It serves two purposes: (i) Standardization of the domain-specific concepts and relations defined in the knowledge artifact ontologies; and (ii) Specification of abstract knowledge links between contextually and functionally congruent knowledge components in different knowledge artifact ontologies. The execution of these knowledge links, through proof engines, eventually leads to knowledge morphing.

A knowledge artifact ontology serves as a lower-level ontology that captures both the structure and content of a particular knowledge artifact–such as a practice guidelines [2], past cases and so on. Each knowledge artifact is represented by an individual knowledge artifact ontology that models the semantic relations inherent in the knowledge artifact, and characterizes the procedural knowledge as a sequence of control structures. Each control structure may deal with the identification, rationalization, ordering, execution and quantification of a domain-specific action and its effects.

3.1.2 Contextualizing Ontologies

Ontologies and contexts are used to model a domain with different views. Ontologies define a shared model that provides a global perspective, whereas contexts are used to realize a local aspect of a domain. Contextualizing an ontology deals with an adaptation of its ontology model to support a local view [11, 12]. In $\mathcal{K}$-$\mathcal{MORPH}$, each knowledge artifact ontology models the procedural knowledge of a knowledge artifact. However, the intended semantics and implementation details of each procedure may vary in different contexts. Contextualizing a knowledge artifact ontology can provide its local view that models (i) a specific interpretation of its ontology concepts, and (ii) an implementation of its procedural knowledge that can be applied in a particular context.

3.2 Morphing Constructs

In order to capture the behaviour of context, under which two or more knowledge artifacts can morphed to solve a specific problem, we defined a *Morphing Construct*. The morphing construct supervises the knowledge morphing process (see section 3.4), and provides a context for determining when, where and how two or more knowledge artifacts need to be reconciled. A Morphing Construct is a tuple that contains context-specific knowledge components and is formally defined as follows:

Definition 2 (Context Declaration) *A Context Declaration $\mathcal{C}_x = \langle l, \mathcal{A}'_x \rangle$ is a tuple comprised of a context label l, and a set of axioms $\mathcal{A}'_x \subseteq \mathcal{A}_x$ that specifies the problem-context and domain-specific constraints, under which ontology-encoded knowledge artifacts are allowed to morph.*

Definition 3 (Morphing Construct) *Let $O_\mathcal{K}$ be a knowledge artifact ontology. Morphing construct $\mathcal{M}_c = \langle O'_\mathcal{K}, \mathcal{C}_x, c_\mathcal{D} \rangle$ is a tuple of a contextualized knowledge artifact sub-ontology $O'_\mathcal{K} \prec O_\mathcal{K}$, a context declaration $\mathcal{C}_x$, and a domain concept $c_\mathcal{D}$ from a domain ontology $O_\mathcal{D}$.*

Example # 1:

1. Let $O'_\mathcal{K} = CPG'$ is a contextualized sub-ontology of CPG (see section 4) [2].
   ```
   CPG′ = [hasRecommendation(X,R),
   hasDecisionCriteria(R,C), hasFollowup(R,F),
   intendedPatient(X,P), hasTimeInterval (R, T), ...]
   ```
2. Let $\mathcal{C}_x = Cx_1 =$
   ```
   <cdss,[(forLocation(cdss, halifax) ∨ forLocation(cdss,
   toronto)), hasResources (cdss, patient_care),
   applicableTo(cdss, resident_patient),
   hasPractioners(cdss, family_physician),
   hasInclusionCriteria (cdss, evidence_based_Recom),
   hasExclusionCriteria (cdss, follow-up_Recom), ...]>
   ```
3. Let $c_\mathcal{D} = cpgBasedRecom$ a concept from the Domain Ontology

An example morphing construct can be written as

$$Mc_1 =< CPG', Cx_1, cpgBasedRecom >$$

A contextualized sub-ontology $O'_{\mathcal{K}}$ represents how certain knowledge components of a knowledge artifact ontology $O_{\mathcal{K}}$ can be utilized under a problem-context. The above example shows an example morphing construct Mc_1 for a knowledge artifact ontology for Clinical Practice Guidelines (CPG) (see section 4) [2]. CPG' is defined as a contextualized sub-ontology that can be utilized under the problem-context Cx_1, augmenting the domain concept $c_{\mathcal{D}} = cpgBasedRecom$. By the declarative knowledge of morphing constructs, sub-ontologies are served as contextualized ontologies of given knowledge artifact ontologies that provide all the contextually-relevant knowledge components that need to be reconciled, to produce a morphed knowledge artifact.

3.3 Ontology Reconciliation

Our knowledge morphing approach is based on the reconciliation of sub-ontologies to yield a unified 'morphed sub-ontology'. Ontology reconciliation among ontologies is normally performed by (i) identifying conceptual similarities among two source ontologies; (ii) aligning and mapping sources ontologies based on identified similarities; (iii) merging, integrating, mediating source ontologies based on found mappings/alignments; and (iv) finding and resolving semantic inconsistencies in reconciled ontologies [6].

Mapping and alignment between ontologies have been carried out based on their lexical, conceptual, and structural similarities [6]. We believe such mappings can became more 'trustworthy' by finding similarities among entities that are driven from the underlying ontology axioms; and so their proofs. Alignments established between entities that takes account in the underling ontology axioms and their proofs, are called *proof-based alignments*. The underling ontology axioms and proofs are served as a declarative semantic model for describing a domain that ontology relates to. By identifying *proof-based alignment* candidates, mappings and alignments will then be consistent with the semantic model, and befitted with the declarative knowledge provided by the ontology axioms.

A *proof-level ontology* is an ontology where each of its triples ($\langle subject, predicate, object\rangle$) are entailed by triples that are not necessarily from the same ontology. An ontology (that represents a relational schema) can be seen as a proof-level ontology, where each of its triples are asserted facts, and are entailed by null (denoted as $\perp \models T$). An ontology at the proof-level can provide the justifications behind inferred instances based on ontology-based and user-defined axiomatic systems (that are modeled in $L(V) = Ax$).

We argue that proof-level ontologies can serve as better candidates for ontology alignment process. If *proof-based alignment* is established among two (inferred) entities in triples T_1 and T_2 from two proof-level ontologies along their proofs ($\mathbb{T}' \models T_1$ and $\mathbb{T}'' \models T_2$, where $\mathbb{T}', \mathbb{T}'' \subseteq \mathbb{T}$), then entities appear in their justifications (modeled as set of ontology triples $\mathbb{T}', \mathbb{T}'' \subseteq \mathbb{T}$) can be treated as the next alignment candidates. *Proof-based alignment* approach ensures that such alignment candidates are aligned in a target ontology.

Proof-based alignment not only finds a similarity between entities, but also maintains the relationship between aligned entities with their original proof structures. After an entity e in one ontology is *proof-based aligned* with an entity f from another ontology, additional proofs can be generated for the new aligned entity f. Such proofs will be analogous to the proof of e. Analogous proofs represent similar reasoning strategies used in a particular domain but expressed in different terminologies.

3.4 Morphing Engine

Our *Morphing Engine* inputs the problem-context, ontology-encoded knowledge artifacts (OKAs), domain ontology, and morphing constructs. It employs the ontology reconciliation process, supervised by the morphing constructs and domain axioms; and generates a morphed knowledge artifact. Morphing constructs lead to identify the contextualized OKAs to be reconciled; whereas a domain ontology provides domain axioms that specify domain-specific constrains to be fulfilled during the morphing process. An abstract process of morphing engine is defined as follows:

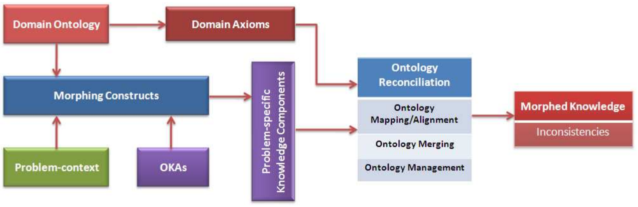

Figure 3. $\mathcal{K}$-$\mathcal{MORPH}$: Morphing Engine

Definition 4 (Knowledge Morphing Process) *Let $\mathbb{O}_{\mathcal{K}}$ be a set of ontology-encoded knowledge artifacts, $\mathbb{O}_{\mathcal{D}}$ be the set of domain ontologies, $\mathbb{C}_x$ be the set of problem-contexts, Π be the set of morphing constructs, and $\mathcal{I} \subseteq Ax$ be the set of logical inconsistencies in the morphed OKA. Knowledge Morphing Process is then the function*

$$\mathcal{MORPH} : \mathbb{O}_{\mathcal{K}} \times \mathbb{O}_{\mathcal{D}} \times \mathbb{C}_x \times 2^{\Pi} \longrightarrow \mathbb{O}_{\mathcal{K}} \times 2^{\mathcal{I}}$$

An abstract architecture of our morphing engine is shown in Figure 3. It first employs the problem-context to determine the problem-specific knowledge components from different knowledge artifact ontologies using morphing constructs. Morphing constructs also delivers the correspondence between identified knowledge components and domain concepts (see section 3.2). Domain ontology provides *Domain Axioms* that describe the semantic relationships among domain concepts. Once the correspondence between the knowledge components and domain concepts is achieved, the morphing engine employs the ontology reconciliation process that (i) computes the semantic correspondence between knowledge components based on the semantic relationships between their corresponding domain concepts; (ii) aligns and then merges knowledge components based on their correspondence; (iii) identifies and resolves semantic inconsistencies, if present; and (iv) generates a morphed (ontology-encoded) knowledge artifact, and unresolved inconsistencies in it.

3.5 Evaluation: Validation and Verification

Once the morphed knowledge artifact is generated, $\mathcal{K}$-$\mathcal{MORPH}$ employs an evaluation process to validate the morphed knowledge. Some of the approaches [4] we plan to involve for evaluating the morphed OKA, are as follows: (i) evaluating, whether results generated from the morphed OKA in a particular application under a specific context are 'satisfactory'; (ii) evaluating logical consistencies, by checking whether the morphed OKA model is consistent with pre-defined domain-specific theories provided by domain experts; (iii) evaluating the morphed OKA against a "golden standard", if available; and (iv) evaluating, whether a pre-defined structure and design principles are maintained in the morphed OKA.

4 USING $\mathcal{K}$-$\mathcal{MORPH}$ FOR CLINICAL DECISION-MAKING

Clinical decision making involves an active interplay between various medical knowledge artifacts to derive pragmatic solutions for a clinical problem [10]. We are currently developing a prototype *Medical Knowledge Morpher* (as shown in Figure 4) that deals with the three different medical knowledge artifacts, namely, (i) Electronic Patient Records (EPR), (ii) Clinical Practice Guidelines (CPG), and (iii) Clinical Pathways (CP). Each knowledge artifact, despite targeting the same domain knowledge, has a different purpose. For instance, CPGs are systematically developed disease-specific recommendations to assist clinical decision-making in accordance with the best evidence [2]. CP serve as institution-specific workflows that guide the care process in line with the evidence-based medical knowledge found in CPG [9]. EPR are containers of patient's longitudinal medical information.

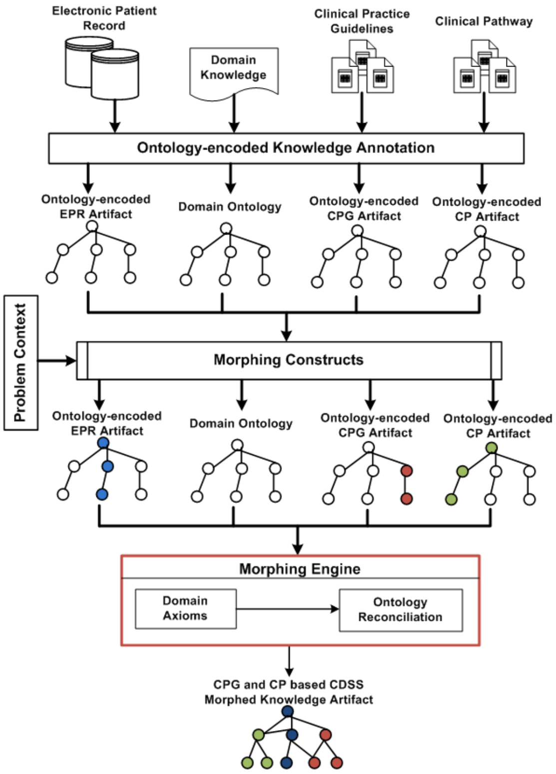

Figure 4. Medical Knowledge Morphing

For clinical decision making we need an active interplay between these three distinct artifacts as follows: The EPR determines the clinical context that in turn determines which CPG need to be referred to make the 'right' clinical decisions. Based on the context, the morphing construct will determine the clinical intention, the knowledge needs for the given intention and the knowledge resources to be utilized. The two knowledge sources in this case–i.e. the CPG and CP –both now need to be integrated to optimally apply the knowledge for clinical decision making. The CPG will provide the declarative knowledge and it needs to be aligned with the procedural knowledge contained by CP. Knowledge morphing is therefore needed at two levels: (a) morphing the different knowledge components from multiple knowledge artifacts of the same type–i.e. recommendations from multiple CPGs; and (b) morphing different knowledge artifact types–i.e. synthesizing CPG and CP. The morphed knowledge artifact will consist of operational relations between EPR, CPG, and CP knowledge artifacts and serve as a holistic knowledge artifact to support clinical decision making in terms of (a) evidence-based recommendations based CPG-based knowledge, based on the patient scenario recorded in EPR, and also (b) institution-specific workflow knowledge to pragmatically execute the recommendations.

5 CONCLUDING REMARKS

Optimal and complete decision support needs a comprehensive knowledge-base. Developing such a self-contained knowledge-base as an independent entity is a challenging undertaking. One possible approach is to systematically leverage multiple knowledge sources to develop a comprehensive knowledge-base–such a composite knowledge-base not only manifests the specializations of its constituent sources but also broadens the knowledge coverage whilst maintaining the uniqueness and independence of the original knowledge sources. In this paper, we presented our knowledge morphing approach, and the $\mathcal{K}$-$\mathcal{MORPH}$ framework, to pursue the development of a comprehensive, multi-facted knowledge-base. We are currently developing a prototype medical knowledge morpher to support clinical decision-making process.

ACKNOWLEDGEMENTS

This research is funded by a grant from Agfa Healthcare (Canada).

REFERENCES

[1] SSR Abidi, 'Medical knowledge morphing: Towards the integration of medical knowledge resources', *18th IEEE International Symposium on Computer-Based Medical Systems, Dublin*, (June 23-24 2005).

[2] SSR Abidi and Shapoor Shayegani, 'Modeling the form and function of clinical practice guidelines: An ontological model to computerize clinical practice guidelines', in *Knowledge Management for Healthcare Processes Workshop at ECAI 2008*, Patras, Greece, (July 21-25 2008).

[3] T Berners-Lee, J Hendler, and O Lassila, 'The semantic web', *Scientific American*, **284**(5), 34–43, (2001).

[4] Janez Brank, Marko Grobelnik, and Dunja Mladenic, 'A survey of ontology evaluation techniques', in *conference on Data Mining and Data Warehouses (SiKDD)*, (2005).

[5] Dave Robertson et. al., 'Open knowledge: Semantic webs through peer-to peer interaction', Technical Report DIT-06-034, University of Trento, Povo, Italy, (May 2006).

[6] Jérôme Euzenat and Pavel Shvaiko, *Ontology matching*, Springer-Verlag, Heidelberg (DE), 2007.

[7] Aykut Firat, Stuart Madnick, and Benjamin Grosof, 'Contextual alignment of ontologies in the ecoin semantic interoperability framework', *Inf. Technol. and Management*, **8**(1), 47–63, (2007).

[8] Fausto Giunchiglia, Fiona McNeill, and Fiona McNeill, 'Web service composition via semantic matching of interaction specifications', in *WWW2007*, Banff, Canada, (May 8-12). ACM Press.

[9] K. Hurley and SSR Abidi, 'Ontology engineering to model clinical pathways: Towards the computerization and execution of clinical pathways', in *20th IEEE Symposium on Computer-Based Medical Systems*, Maribor, Slovenia, (June 20-22 2008). IEEE Press.

[10] VL Patel, JF Arocha, and J Zhang, *Cambridge Handbook of Thinking and Reasoning*, chapter Thinking and reasoning in medicine, Cambridge University Press, 2004.

[11] Aviv Segev and Avigdor Gal, 'Putting things in context: A topological approach to mapping contexts to ontologies', *Journal on Data Semantics IX, LNCS 4601*, 113–140, (2007).

[12] Tony Veale and Yanfen Hao, 'Corpus-driven contextualized categorization', in *Contexts and Ontologies workshop at ECAI 2006*, Trento, Italy, (August 2006).