# SATISFYING USER EXPECTATIONS IN ONTOLOGY-DRIVEN COMPOSITIONAL SYSTEMS

## *A Case Study in Fish Population Modeling*

Mitchell G. Gillespie, Deborah A. Stacey
*School of Computer Science, University of Guelph, 50 Stone Road East, Guelph, ON, Canada*

Stephen S. Crawford
*Department of Integrative Biology, University of Guelph, 50 Stone Road East, Guleph, ON, Canada*
*Chippewas of Nawash Unceded First Nation, RR #5, Wiarton, ON, Canada*

Keywords: Ontology-Driven Compositional System (ODCS), Ontology, System composition, User expectations, Fish population, Modeling.

Abstract: Ontology-Driven Compositional Systems (ODCSs) are designed to assist a user with semi- or fully automatic composition of a desired system. Current research with ODCSs has been conducted around the discovery and composition of web services and alternatively a bottom-up resource management approach to automatic system composition. This paper argues that current ODCSs do not truly satisfy user expectations as the semantic knowledge required to make proper discovery, decision-making and composition has not been fully represented. The authors introduce the beginning of their work of utilizing the inheritance of multiple ontologies to fully represent the functional, data, quality & trust, and execution of compositional units within an ODCS. Furthermore, a case study of fish population modeling is presented.

## 1 INTRODUCTION

For many years, stakeholders have utilized previously implemented algorithms and packages to minimize the work for software developers when designing a software system. One example is the Open Source community that supports software developers with the ability to manually integrate previously composed modules and systems (Feller and Fitzgerald, 2002). Similarly, an increasing number of Web Services allow developers to manually connect to remote services via their own distributed system (Meng et al., 2006). Recent research in this area has shifted focus to understand the requirements and processes of Compositional Systems (Cardoso and Sheth, 2005). Systems that could provide assistance with automatic or semi-automatic system composition using a collection of previously developed algorithms, modules, services and packages.

To comprehend the previously implemented software available in a Compositional System, a semantic knowledge of the various components must be provided. Ontologies are explicit specifications of a conceptualized body of knowledge, and commonly utilized to understand the sharing of knowledge among people and/or software agents (Gruber, 1993). Thus, ontologies are recognized as a appropriate tool to represent the semantic knowledge that drives a Compositional System (Arpinar et al., 2005; Hlomani and Stacey, 2009). Current research from Apinar (2005) focused on a Ontology-Driven Web Service Composition System, and Hlomani and Stacey (2009) shifted their focus to incorporate both distributed and non-distributed components. Overall, the research of Ontology-Driven Compositional Systems (ODCSs) is in its youth and many gaps still need to be approached. Thus far, ODCS research has focused primarily on the discovery and composition of previously developed software, and rarely considers ranking and selection previously developed software to decide which to use. A ranking and selection process requires an understanding of the end-user's expectations to evaluate whether or not a resulting composed system form the ODCS is satisfactory.

Through collaboration with the Integrative Biology Department at the University of Guelph and the

G. Gillespie M., A. Stacey D. and S. Crawford S..
SATISFYING USER EXPECTATIONS IN ONTOLOGY-DRIVEN COMPOSITIONAL SYSTEMS - A Case Study in Fish Population Modeling.
DOI: 10.5220/0003103801330143
In *Proceedings of the International Conference on Knowledge Engineering and Ontology Development* (KEOD-2010), pages 133-143
ISBN: 978-989-8425-29-4

Chippewas of Nawash Unceded First Nation fisheries management program (Crawford et al., 2008), the authors isolated a case study where they can measure user expectations. Why fisheries population modeling? In most cases, fisheries biologists lack the mathematical and computational knowledge to model fish populations properly (Megrey and Moksness, 2009) and ODCSs could provide the appropriate knowledge to allow the successful construction and execution of accurate population models. Furthermore, as a fisheries biologist's knowledge is specialized in population dynamics, their expectations of a composed population modeling system will be more complex than whether or not the computer population models compile and execute. Domain-specific metrics of quality and trust could affect a fisheries biologist's acceptance of certain model components over another.

Utilizing the case study in fisheries population modeling, the authors will illustrate how satisfying user expectations in an ODCS accommodates various research initiatives. Explicitly, compositional systems researchers (Cardoso and Sheth, 2005; Arpinar et al., 2005; Hlomani and Stacey, 2009) will further understand the dynamics involved in the creation of practical applications for ODCS. Also, fisheries management experts will be introduced to an innovative tool to assist with the composition of population models. Implicitly, these population modeling experts will also be embracing a framework for representing instances of population models.

In this paper, current implementations of Ontology-Driven Compositional Systems are assessed, followed by a presentation of work being conducted by the authors. Section 2 introduces the definitions of *System Composition* and *Compositional Systems* and includes a framework of the semantic knowledge and processes required within a *Compositional Systems*. Section 3 assesses current implementations using the definitions and framework from Section 2. Section 4 presents aspects of User Expectations desired in the design of an Ontology-Driven Compositional System to enhance the quality and trust of the resulting system's outputs. Using the desired user expectations, Section 5 introduces the current work by the authors, including the fisheries population modeling case study. Finally, discussion of an evaluation technique for Ontology-Driven Compositional Systems and future work is presented.

# 2 SYSTEM COMPOSITION & COMPOSITIONAL SYSTEMS

*System Composition* is the process by which two or more previously implemented compositional units (*e.g.* algorithms, packages, services) are constructed together to create a holistic functional system. This definition can refer to both manual construction by a software developer (*i.e.* without the aid of any computational tools) or any type of computer assisted technique. *Compositional Units* are the *functional* "black-box" algorithms, packages or services that receive a given input, that provide a function or service and send a calculated output.

*Computer-Assisted System Composition* (CASC) is a process by which a human user utilizes the aid of a *Compositional System* (CS) to assist in the construction of a *Resulting System*. Hlomani and Stacey (2009) define a *Compositional System* as "a system that allows its components to be put together in a systematic manner to achieve a common goal or to derive yet another functional application". A *Resulting System* is the final output of interconnected *Compositional Units* to be executed by the user. Hlomani and Stacey (2009) refer to the *Resulting System* as "yet another functional application". With CASC, the degree of user interaction when conducting CASC could range from requiring user input at every stage of the *System Composition* to no interaction in a completely automatic system composition.

## 2.1 Semantic Knowledge Requirements

All compositional systems hold some form of semantical knowledge about the compositional units they wish to utilize (Srivastava and Koehler, 2003; Cardoso and Sheth, 2005). For web services specifically, Cardoso (2005) described four different types of semantical knowledge to be considered by web service compositional systems. This paper adapts the work of Cardoso (2005) to provide five different types of semantics to be considered in all forms of compositional systems, ontology-driven or otherwise:

1. **Functional Semantics:** knowledge about the function, features and compositional purpose of the compositional units and input/output data.
2. **Data Semantics:** knowledge about the order, format and structure of the input/output for a given compositional unit.
3. **Quality & Trust Semantics:** knowledge that characterizes how well a given compositional unit performs and other metrics which distinguish if certain compositional units are suitable for a given

system composition (adapted from Cardoso and Seth's "QoS Semantics" ).

4. **Execution Semantics:** knowledge about locations, requirements and environment dependancies that need to be utilized or satisfied for a compositional unit to be executed in a resulting system.
5. **Timeline Semantics:** knowledge about the compositional and chronological requirements to create sections of or full implementations of a resulting system.

The Functional, Data, Quality & Trust, and Execution semantics represent the knowledge about each given instance of a compositional unit, whereas the Timeline semantics are utilized for describing appropriate combinations of compositional units. The collection of the five semantics allow for unique specification of each compositional unit (functional, data, quality & trust, and execution), and general combinations

Let's investigate an example. A computer population model could be an instance of a compositional unit (CU). This CU could be described to have a "catch-at-age" functional purpose (Function Semantics), and receive a temporal harvest vector as input data (Data Semantics). However, the harvest vector requires data aggregation from month to year, therefore another CU is required to execute that function successfully. The Timeline Semantics would provide the general combination of a population model CU and data aggregation CU, while the specific semantics would provide the ability to match the correct instance of a data aggregation CU (i.e. a vector aggregation tool from month to year) to the population model CU.

## 2.2 Discovery, Decision-making and Composition Processes

Within computational systems three general processes occur: discovery, decision-making and actual composition (adapted from Cardoso (2005)). Discovery refers to the process of using the semantic knowledge to **match** compositional units that meet the requirements for a desired resulting system. Decision-making refers to the process of **ranking and selecting** the discovered compositional units for the system, and finally Composition is the process of automatically constructing the compositional units together to form the actual resulting system to be executed. Overall, Figure 1 illustrates how a compositional system utilizes the semantic knowledge of compositional units to conduct discovery, decision-making and composition of a resulting system.

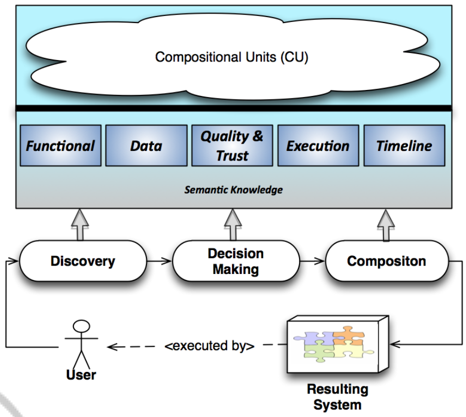


Figure 1: Discovery, Decision-Making and Composition processes in a Compositional System utilizing the semantic knowledge of compositional units to create a resulting system for the user to execute. Compositional units may or may not be locally stored and/or managed (*i.e.* a downloadable package would be locally stored by the compositional system, whereas a web service would not).

All compositional systems require their semantic knowledge represented in one relational form or another. As more functions and features of compositional systems are considered, the relationship between all the different facets of knowledge becomes more difficult to maintain. Enter Ontologies!

# 3 ONTOLOGY-DRIVEN COMPOSITIONAL SYSTEMS

Hlomani and Stacey (2009) define an Ontology-Driven Compositional System as "a prototype system [...] that demonstrates the power and suitability of using ontologies as the main driver for a compositional system". Using the definitions from Section 2, this paper states that *an Ontology-Driven Compositional System (ODCS) is a compositional systems which utilizes ontologies to perform computer-assisted system composition for a user, where the ontologies represent the functional, data, quality & trust, execution and timeline semantic knowledge of the included compositional units*. The ODCS conducts semantic reasoning on the ontologies for the discovery, decision-making and composition of a resulting system. Figure 2 (an adaptation from Figure 1) illustrates the ontological representation of semantic knowledge.

Certain computational characteristics of the compositional units (*e.g.* a web service or downloadable package) affect the specific focus of a compositional system. Two variations of ODCSs are presented: Ontology-Driven Web Service Composition Systems

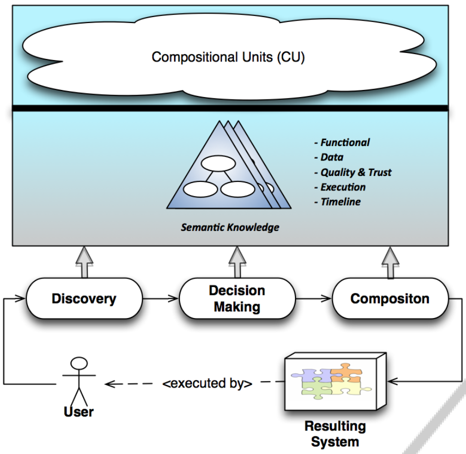


Figure 2: Adaptation of Figure 1 to illustrate how ontologies are utilized by ODCSs during the process of system composition.

(Meng et al., 2006; Arpinar et al., 2005) and a "resource management"-based Ontology-Driven Composition System (Hlomani and Stacey, 2009).

## 3.1 Ontology Driven Web Service Composition

With the growing popularity of distributed web applications, there is a strong focus on delivering "Semantic Web Services" (Cardoso and Sheth, 2005). Semantic Web Services are defined as instances of web services that fit within a semantic representation for users and/or developers to discover and utilize in their distributed software systems. Cardoso and Sheth (2005) emphasized the importance of ontologies to represent the semantic knowledge to allow an artificial intelligence to reason and discover available web services. Many research foci have expanded from this initiative by proposing the use of ontologies to assist with semi- or fully-automatic web service composition(Meng et al., 2006; Arpinar et al., 2005).

Arpinar et. al. (2005) considered all processes presented in Section 2.2 with specific application to a winery, web service case study. The system architecture of their ODCS utilized three ontologies to drive the composition: a *Domain ontology*, a *Web Services ontology*, and a *Process ontology* (Figure 3). The *Domain ontology* was only utilized to represent the data input and output, while the actual web services were represented in a general *Web Services ontology*.

Arpinar et. al. (2005) referenced work by Zhang (2004) where the defined *Web Services ontology* held entity titles like 'Wine-Searcher' (sub-class of 'Searching Service') representing instances of web services utilized to find different types of wines. The

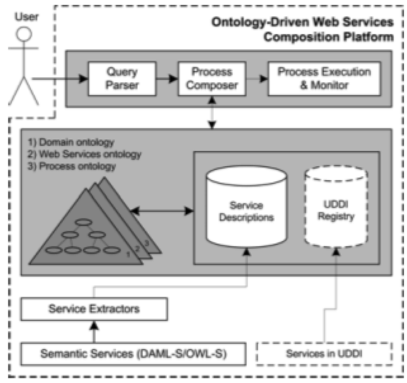


Figure 3: System Architecture Diagram of the Ontology-Driven Web Service Composition Platform (Arpinar et al., 2005).

Domain ontology listed properties such as wine name, vintage, winery, etc. which allowed the user to define ranges and filters to the 'Searching Service'.

Finally, the *Process Ontology* represented sequences of services that were connected to one another if their interfaces were semantically matched. The work by Arpinar et. al. (2005) focused most of their work on the interface input/output matching between web services (*i.e.* compositional units) and the execution of the final resulting system. The quality of web services was a consideration through a set of quality criteria adopted from Zeng et. al. (2003), however Arpinar et. al. (2005) recognized quality as a feature that had to be further developed and researched for their system.

Ontology-Driven Web Service Composition has gained large amounts of popularity as many stakeholders hold an interest in the growing web services economy and are willing to commit funds and effort (Cardoso and Sheth, 2005). However, most of the work on Ontology-Driven Web Service Composition is focused heavily on the discovery, matching and composition of compositional units, and less on their ranking and selection (decision-making).

## 3.2 Resource Management Focused Ontology-driven Compositional System

Hlomani and Stacey (2009) approached the design of an Ontology-Driven Compositional System in a different manner. Their compositional system was designed (in theory) to accept compositional units implemented with distributed or non-distributed functionality. Furthermore, the ODCS contained a physical copy of all non-distributed compositional units giving the ODCS a 'resource management' focus.

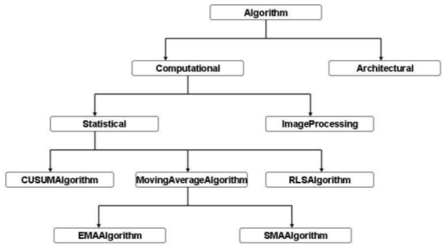


Figure 4: An Illustration of the Functional Semantics Represented in the Algorithm Ontology created by Hlomani and Stacey (2009).

The overall goal of Hlomani and Stacey (2009) was to create a *plug and play* hub of compositional units to allow users to explore the discovery, decision-making and composition of a variation of resulting systems.

The ODCS for Hlomani and Stacey (2009) utilized two ontologies to drive their compositional system: an *Algorithm ontology* and a *Execution Timeline ontology*. Rather than web services, the compositional units in Hlomani and Stacey's ODCS were algorithms, thus the Algorithm ontology (Figure 4) provided an extensive standard description of algorithms (*i.e.* functional properties and the classification of input and output). Similar to the Process ontology in Arpinar et. al. (2005), the Execution Timeline ontology provided semantic descriptions to assist with the composition of events.

To date, Hlomani and Stacey (2009) is the only ODCS research outside the web services focus. Similar to Arpinar (2005) the ODCS focused on the functional, data and execution semantic knowledge, however they did note the importance of ranking the quality and trust of compositional units. One unique strength noted from their work, was the explicit distinction between compositional units that were actual computational components (*i.e.* Algorithm Components) and "helper" compositional units which assisted to *glue* together inputs and outputs of computational units that could otherwise not be connected (*i.e.* Architectural Components).

## 3.3 Overall Assessment of Current Implementations

After a brief investigation of recent implementations of an Ontology-Driven Compositional Systems, the authors identified important strengths and weaknesses:

**Strength: Methods of Discovery and Composition**

Both Arpinar et. al. (2005) and Hlomani and Stacey (2009) focused most of their efforts on discovery and composition. As ODCSs are in their youth, a prototype with minimal complexity is optimal. Both ODCSs utilize a unique ontology to assist with defining the combination of compositional units required for a certain resulting system requested by the user. The Input/Output matching algorithms have been implemented successfully, however a larger knowledge base of compositional units, input/output and resulting systems would provide opportunities for more robust matching. Hlomani and Stacey (2009) accepted this recognition by adding functional semantic knowledge that described "architectural" compositional units to act as *glue* to match more input/output.

**Weakness: Representation of Domain-Specific Functional Semantic Knowledge**

The functional semantic knowledge of compositional units and their input/output has been limited to the use of a small *domain ontology*, or a minor inheritance relationship in one ontology. Creating a more robust functional semantic knowledge representation of compositional units has been proposed, but not investigated (Arpinar et al., 2005). Furthermore, as the number, variability and specification of different types of compositional units increase, more attributes and descriptors will be necessary to distinguish the difference between which compositional units are desired by the user.

**Weakness: Quality & Trust of Compositional Units**

Currently, only minor efforts have been attempted to rank and select compositional units in the "Decision-Making" process in an ODCS (Hlomani and Stacey, 2009; Arpinar et al., 2005). The ranking of quality was focused mostly on computational performance-based metrics and did not consider domain-specific and qualitative-like performance metrics. Research within the "Semantic Web Services" (Cardoso and Sheth, 2005) domain has focused on improving ranking of quality (Zeng et al., 2003; Tran, 2008; Wang et al., 2006), however that research has not yet been implemented into an ODCS. The semantic representation of "Trust" was considered only by Arpinar et. al. (2005) with a brief mention of reputation, however many more elements should be considered.

### 3.4 Designing an ODCS from User Expectations

The weaknesses identified are focused mostly around the satisfaction of a user's expectations of a resulting composed system. A user expects to utilize and match as many compositional units as possible and be provided with a robust and holistic view of the compositional units. Furthermore, a user should expect to rank compositional units based on aspects of performance and/or non-performance quality metrics to assess which compositional units he/she or other users/experts trust.

## 4 DEFINING USER EXPECTATIONS FOR AN ODCS

User satisfaction is driven by a user's initial expectations. Satisfying user expectations of an ODCS is therefore defined as "creating a composed system that satisfies the initial expectations of a user to the point where he/she is able to accept the resulting composed system". One aspect of the use of ODCSs is that users need to specify input/output requirements before the resulting system is composed. Therefore, a user's expectations can be retrieved at this time as well. This paper introduces three user expectations to drive the operation of ODCSs: leverage and acquisition of knowledge, acknowledgment of trust and satisfaction of performance. Obviously, other user expectations could be considered, but for the context of this paper these three are examined.

### 4.1 Leverage & Acquisition of Knowledge

Users are more satisfied with a system when the amount of their cognitive processing and knowledge requirements are decreased. (Zhang and von Dran, 2002). ODCSs could be utilized by users who have varying sets and levels of predetermined knowledge. For example, one user with comprehensive knowledge of programming and software engineering techniques will understand the dynamics of matching input and output. However, a second user without that knowledge will need to further leverage the assistance of the ODCS. Users may also not comprehend domain-specific aspects of a compositional unit relevant to the resulting system they wish to create. (*e.g.* a fisheries biologist may not understand the dynamics of statistical modeling in the running of population models). As stated in Section 2.1, compositional systems must hold different types of semantic knowledge and the user expects to utilize that network of expert knowledge in the ODCS.

Using the five different types of semantic knowledge in compositional systems (Section 2.1), three facets of expert knowledge are described: computational system, domain-specific and task-oriented. Computational system expert knowledge is the system-specific function, data, quality and execution knowledge in a compositional system. This would be very useful for a software developer, but not for individuals without some sort of programming knowledge. The domain-specific expert knowledge is the "real world" function, data, quality, and execution knowledge of the compositional units being used to make the resulting system. This information is usually embedded within a program and not explicitly differentiated. Finally, the task-oriented expert knowledge the user expects to leverage is the task and goal semantic knowledge of the compositional system.

Deconstructing user expectations further, as a user is leveraging different facets of expert knowledge, s/he will also be consciously or unconsciously acquiring that knowledge. Acquisition of computational system and/or domain-specific expert knowledge could be obtained by exploring and experimenting with different selections of compositional units (*e.g.* a fisheries biologist could research more computer population models by including different combinations in a resulting composed system from a ODCS to gain more understanding why a given population model is trusted more than another).

### 4.2 Deconstructing Quality & Trust

As emphasized previously, the end-user of a ODCS does not only leveraging and aquire knowledge. S/he also considers the quality of the compositional units to be a important factor of the resulting system. The following two user expectations relate directly to the "Quality & Trust" semantic knowledge presented in Section 2.1.

#### 4.2.1 Acknowledgment of Trust

An increasing number of software developers and human-computer interaction specialist acknowledge that if a user does not trust aspects of software systems or interfaces, than s/he automatically expects to be satisfied less. Duez et. al. (2006) states, "If [users] do not trust the new automated tools, they will not use them  no matter how useful or efficient they might be". In an ODCS, a user expects to be provided with compositional units that are tested and accepted by various software and domain-specific experts. Also, each user is unique and s/he may hold higher regard for certain software developers or domain experts, and respect that those experts have the knowledge to understand certain compositional units more clearly. Thus, a higher ranking should be assigned to the compositional units that the certain experts have either programmed or trusted as the user will be more satisfied.

As explained, the knowledge of different types of experts is leverages, therefore attributes of trust that a software developer would consider is program creator, number of times utilized, reputation from other programming experts (especially security reputation), minimal number of errors during testing, minimal number of bugs etc. Whereas, a domain-specific expert would trust similar and inherited features, yet each unique domain would have separate features that also must be considered (*e.g.* a population modeling expert would consider the number of simulation studies to be an appropriate measure).

#### 4.2.2 Satisfaction of Performance

Ultimately, the resulting system generated from a compositional system is still a piece of computational software. Therefore, most users will still consider performance metrics to be extremely important. Programming specific features like estimated time of execution, amount of memory required or, distance of distributed machines will always be important. Simple metrics and protocols like QoS can be utilized. Domain-specific performance proves slightly more difficult, as the quality of performance could be measured by different numerical representations. For example, many fisheries management population models may only produce a successful modeling simulation some of the time due to factors such as data availability, data quality, or model complexity (Megrey and Moksness, 2009).

## 5 FIRST ATTEMPT AT AN ODCS MOTIVATED BY USER EXPECTATIONS

After the investigation of current ODCSs in Section 3, and a definition of user expectations in Section 4, an ODCS motivated by user expectations can be investigated. This paper utilized work conducted by Hlomani and Stacey (2009) with a focus on the discovery and decision-making processes. Figure 5 illustrates how the authors have begun to adapt the ontologies and processes to satisfy the user expectations defined in the section above. The figure generalizes how a domain ontology represents a unique set of semantic knowledge, yet through ontology merging and inheritance all required semantic knowledge can be represented as a whole.

Currently, the authors are focused on the holistic capture and development of a Compositional Unit ontology (CU ontology), a Statistical modeling CU ontology, and a Fish Population modeling CU ontology to improve the discovery (*e.g.* matching) and decision-making (*e.g.* ranking and selection) processes for the end-user. Section 5.1 describes the adapted compositional unit ontology previously utilized by Hlomani and Stacey (2009). Section 6 follows by providing an example of how domain-specific ontologies (Statistical modeling CU ontology, and Fish Population modeling CU) will inherit the CU ontology (and each other) to fully represent the case-study domain in a holistic fashion.

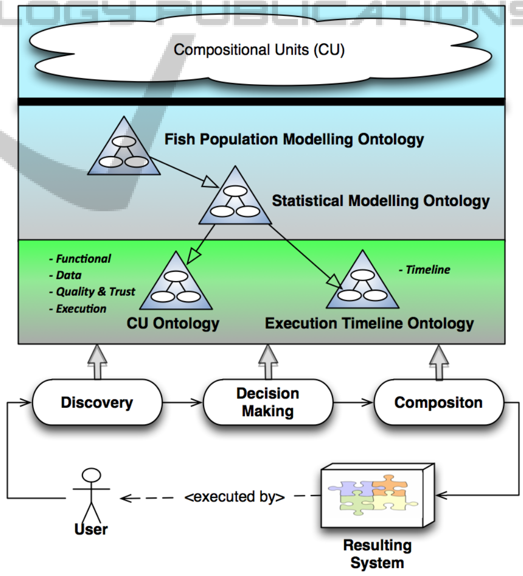


Figure 5: Adaptation of Figure 2 to illustrate how this paper's ODCS utilizes inheritance and merging of multiple ontologies to satisfy the user expectations of leveraging knowledge and incorporation of trust.

### 5.1 CU Ontology

The CU ontology was adapted from Hlomani and Stacey (2009) by evolving their Algorithm ontology into what this paper defines as the Compositional Unit ontology (CU ontology). The change of name to "Compositional Unit" was necessary since the term "Algorithm" does not semantically define all possible units of composition (see Section 2 for

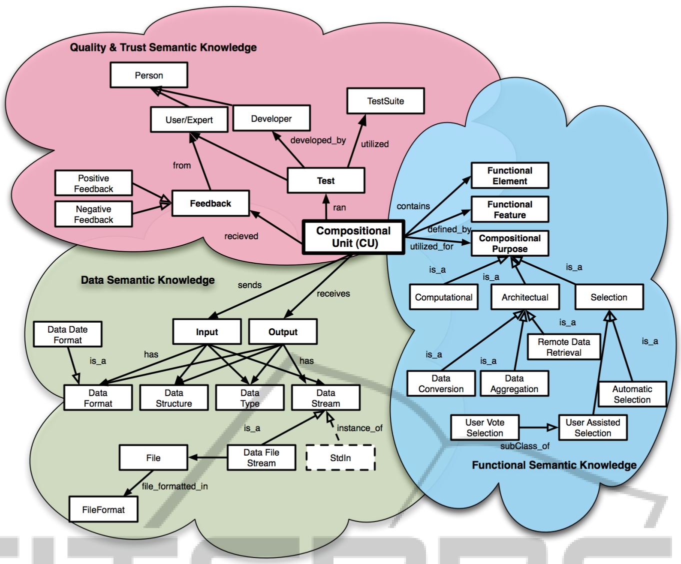


Figure 6: A few of the essential entities in the prototype Compositional Unit Ontology (CU ontology). The CU Ontology is proposed to be the base system ontology that is utilized along with the Timeline ontology to compose systems. As Figure 5 illustrates, domain-specific ontologies inherit and merge with the CU ontology to allow the classification of specific domain knowledge. Section 6 presents a case study of this paradigm.



the definition of a compositional unit). Similar to Hlomani and Stacey's Algorithm ontology, the CU ontology represents the functional, data, quality & trust, and execution semantic knowledge of single compositional units. The *Timeline ontology* has not yet been adapted by the authors, but will be approached in the future. Figure 6 provides an illustration of some of the entities within the CU ontology. As the focus of this paper is on satisfying user expectations in an ODCS, the execution and timeline semantic knowledge will not be presented because it is not a high priority concern for the end-user.

**Functional Semantic Knowledge.** The functional semantic knowledge represented in the CU ontology attempts to provide an explanation of the features, elements, and compositional purpose of the compositional units. The term "features" refers to high-level characteristics of a given instance of a compositional unit and the "elements" refers to functional characteristics *within* an instance of a compositional unit. Finally, "compositional purpose" refers to the reason why compositional units are presented. Similar to Hlomani and Stacey (2009), most compositional units are "computational" components that provide services described by the "features" and "elements", however some act like *glue* and are semantically represented as "architectural".

**Data Semantic Knowledge.** Most of the data semantic knowledge is utilized during the matching of input/output (*i.e.* discovery) of the compositional units for a resulting system. Both the input/output could have defined format, structure, and/or type; format refers to string-like protocols that are followed (*i.e.* regular expressions, or string date formats), and structure refers to the method by which the data is represented (*e.g.* matrix, vector, etc.). Furthermore, input/output of the compositional units will be received/sent via a given data stream which could be local or remote.
aa

**Quality & Trust Semantic Knowledge.** The quality & trust semantic knowledge in the CU ontology focuses on characteristics that provide the ability to rank and select the compositional unit. Information like positive and negative feedback could dictate whether an end-user would select the given compositional unit. Alternatively, an end-user may not trust a certain expert or developer, therefore any feedback or tests by that expert or developer should be ignored in a ranking process.

## 6 CASE STUDY: FISHERIES POPULATION MODELING

Fisheries managers understand that the anthropogenic effects of harvesting must be studied in a transpar-

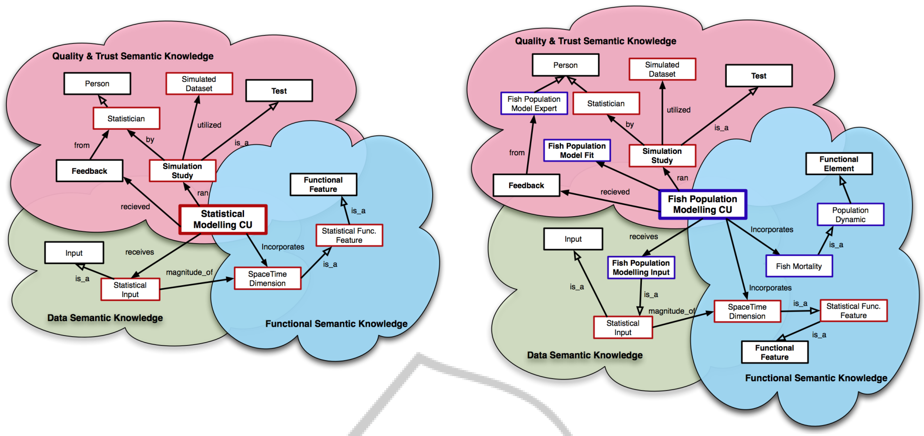


Figure 7: A case study of fisheries population modeling: an example of how multiple ontologies could inherit the CU ontology and further describe the functional, data, and quality & trust of the compositional units within their own domain.

ent, accountable and scientifically-defensible manner (Walters and Martell, 2004). Fisheries biologists are specifically responsible for collecting and analyzing data to estimate fish population abundance, assess human and non-human sources of mortality, and to ensure sustainable harvest levels (Quinn and Deriso, 1999). Unfortunately, many fisheries management decisions are based largely on qualitative indicators or expert opinion of fish population status, rather than estimates derived from more complex mathematical modeling tools developed over the past 30 years (NRC, 2005; Methot, 2009). In some cases, the biologists are constrained by the quantity and/or quality of data required for the mathematical models, while in other cases the biologists are themselves limited in their understanding of how these models function (Stringer et al., 2009).

Through a unique fisheries research collaboration between the Chippewas of Nawash Unceded First Nation and the University of Guelph, the authors have investigated the domain of fisheries population models to understand how fisheries biologists could: [1] leverage and acquire the computational, mathematical, and statistical knowledge to appropriately employ the population models, and [2] enhance their level of trust in the quality and performance of the generated system. The following two sub-sections present how the authors ODCS could be utilized to achieve the stated objectives. The first sub-section introduces an example of how fisheries managers could leverage a network of expert knowledge through the inheritance and merging of the multiple domain ontologies. The second sub-section provides a specific focus on quality and trust semantic knowledge and how it has been considered through the multiple domain ontologies as well.

## 6.1 Leveraging Knowledge for Fish Population Models

Fisheries population models are mathematical/statistical models implemented as computer algorithms specifically to focus our understanding on important factors that determine population abundance and condition (Hilborn and Walters, 1992). Therefore, two levels of domain knowledge need to be considered past the semantic knowledge represented in the Algorithm Ontology. As shown in Figure 5, the first domain of knowledge considered is Statistical Modeling, followed by the domain of Fish Population Modeling itself. Considering functional semantic knowledge in particular, the Statistical Modeling ontology describes high level functional features such as spatial/temporal dimensions or stochastic/deterministic processes and similar functional features of data, such as whether or not given parameters are sampled using normal, log-normal or uniform distributions. The Fish Population Modeling ontology would hold functional features such as mortality and/or recruitment implementations utilized, and functional features of data such as the functional purpose of its input (*i.e.* mortality, recruitment, etc.). Figure Figure 7 provides an example of how the two domain ontologies in this case study inherit knowledge and merge with the CU ontology.

## 6.2 Incorporating Quality & Trust

Within each domain presented in this case study, different types of metrics need to be considered to measure and rank aspects of quality and trust specifically. As shown in Figure 7, statisticians would trust a given population model CU more if simulation studies had been conducted and fisheries managers may wish to trust population models that have gotten feedback from expert population modelers whom they trust. More complex metrics of trusting quality and performance are also included, however, these are not considered in the examples provided here.

## 7 DISCUSSION OF FUTURE WORK

There are many gaps that must be investigated in ontology-driven compositional systems. Currently two facets of future work are being considered: further development of domain-specific ontologies and investigation of evaluation techniques.

### 7.1 Further Domain-specific Ontology Capture

Aside from ongoing adaptations of the ODCS of Hlomani and Stacey (2009) to the *CU ontology*, the domain-specific ontologies (*Statistical modeling CU* and the *Fish Population modeling CU*) still require further development. The authors have organized a series of three focus group sessions with the Integrative Biology Department and the Saugeen-Ojibway/Nawash First Nations fisheries managers to approach this. The group sessions will continue to focus on aspects of functionality, data, and quality & trust.

### 7.2 Investigation of Evaluation Techniques

Because ODCS research is still very new, no attempts at defining a ”golden standard” have been made. Like similar research in ontology capture, engineering and validation it is very difficult to argue whether or not a given ontology is properly representing the given knowledge domain (Uschold and King, 1995; Gangemi et al., 2005). This argument is dynamically dependent on the ”eye of the beholder”, as separate individuals will have different opinions on whether or not a given ontology represents the state of knowledge. In the future, the authors will investigate how a ”golden standard” would be defined for ontologies within an ODCS with a specific focus on classifying how well the ontologies represent the different tenets of semantic knowledge described in Section 2.1. This ”golden standard” would then have to be adapted to facilitate the various domain-specific ontologies, as a certain domain of semantic knowledge may require different standards.

Expanding further, the benefit of this paper's research is the overall goal: to satisfy user expectations of a computer assisted system composition process within an ODCS. Therefore, the authors will run user interviews with domain users of varying levels/sets of knowledge (*i.e.* statisticians, population modellers, software developers, fisheries biologist, etc.) to investigate how well that goal is satisfied.

## ACKNOWLEDGEMENTS

The authors would like to acknowledge all of the collaboration with the Integrative Biology Department at the University of Guelph and the Chippewas of Nawash Unceded First Nation, specifically Chief & Council, Scott Lee, Ryan Lauzan, Dan Gillis, and Jasper Tey. Mitchell Gillespie would like to acknowledge the patience, kindness, assistance and opportunities that Dr. Deborah Stacey has provided during his work as a graduate student. Last, but not least, he acknowledges all of the support he receives from his partner (Kathryn Marsilio), and his family (Katie, Tom, and Mugz). Deborah Stacey wishes all of her graduate students were so dynamic and productive.

## REFERENCES

Arpinar, I. B., Zhang, R., Alemen-Meza, B., and Maduko, A. (2005). Ontology-driven web services composition platform. *Information Systems and e-Buisness Management*, 3:175–199.

Cardoso, J. and Sheth, A. (2005). Introduction to semantic web services and web process composition. In *First International Workshop on Semantic Web Services and Web Process Composition*, Lecture Notes in Computer Science, pages 1–13. Spinger.

Crawford, S., Gillis, D., and Rooney, N. (2008). A review of population level ecological risk assessments for the candu owners group. Technical report, CANDU Owners Group, Toronto, ON, Canada.

Crawford, S., Muir, A., and McCann, K. (2001). Ecological basis for recommendation of 2001 saugeen ojibway commercial harvest tacs for lake whitefish (coregonus clupeaformis) in lake huron, report prepared for the chippewas of nawash first nation. Technical report, University of Guelph, Wiarton, ON, Canada. (revised with references 11 July 2002; revised with response to OMNR comments 02 January 2003).

Duez, P. P., Zuliani, M. J., and Jameison, G. A. (2006). Trust by design: Information requirements for appropriate trust in automation. Technical report, IBM Canada Ltd.

Feller, J. and Fitzgerald, B. (2002). *Understanding Open Source Software Development*. Addison-Wesley, London.

Gangemi, A., Catenacci, C., Ciaramita, M., and Lehmann, J. (2005). A theoretical framework for ontology evaluation and validation.

Gillis, D., Tey, J., Gillespie, M., and Crawford, S. (2009). Do fisheries biologists have appropriate tools for assessing dynamics of harvested fish populations? Not yet submitted to Journal.

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

Hilborn, R. and Walters, C. J. (1992). *Quantitative fisheries stock assessment: choice, dynamics and uncertainty*. Chapman and Hall, New York.

Hlomani, H. (2009). A bottom-up approach to system composition using ontologies. Master's thesis, University of Guelph, Guelph, ON, Canada.

Hlomani, H. and Stacey, D. (2009). An ontology driven approach to software systems composition. In *Proceedings of the 2009 International Conference of Knowledge Engineering and Ontology Development*. INSTICC.

Majithia, S., Ali, A. S., Rana, O. F., and Walker, D. W. (2004). Reputation-based semantic service discovery. *Enabling Technologies, IEEE International Workshops on*, 0:297–302.

Megrey, B. A. and Moksness, E. (2009). *Past, present, and future trends in the use of computers in fisheries research*, pages 1–30. Computers in Fisheries Research, 2nd Edition. Springer Science.

Meng, X., Junliang, C., Yong, P., Xiang, M., and Chuanchang, L. (2006). A dynamic semantic association-based web service composition. In *Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence*. IEEE.

Methot, R. D. J. (2009). *Stock assessment: operational models in support of fisheries management*, pages 137–165. Fish & Fisheries Series 31. Springer Science, Netherlands.

NRC (2005). Improving fish stock assessments.

Quinn, T. J. and Deriso, R. (1999). *Quantitative fish dynamics*. Ofxord University Press, New York.

Srivastava, B. and Koehler, J. (2003). Web service composition – current solutions and open problems. In *ICAPS 2003 Workshop on Planning for Web Services*, Trento, Italy.

Stringer, K., Clemens, M., and Rivard, D. (2009). *The changing nature of fisheries management and implications for science*, pages 97–111. Fish & Fisheries Series 31. Springer Science, Netherlands.

Tran, V. X. (2008). Wsqosonto: A qos ontology for web services. In *2008 IEEE International Symposium on Service Oriented System Engineering*, pages 233–238. IEEE.

Uschold, M. and King, M. (1995). Towards a methodology for building ontologies. In *In Workshop on Basic Ontological Issues in Knowledge Sharing, held in conjunction with IJCAI-95*.

Walters, C. J. and Martell, S. J. D. (2004). *Fisheries Ecology and Management*. Princeton University Press, New Jersey.

Wang, X., Vitvar, T., Kerrigan, M., and Toma, I. (2006). *A QoS-Aware Selection Model for Semantic Web Services*, pages 390–401. Lecture Notes in Computer Science. Springer, Berlin, Germany.

Wang, Y. and Vassileva, J. (2007). Toward trust and reputation based web service selection: A survey. In *Proc. Intl. Transactions on Systems Science and Applications*, volume 3, pages 118–132.

Zeng, L., Benatallah, B., Dumas, M., Kalagnanam, J., and Sheng, Q. Z. (2003). Quality driven web services composition. In *Proceedings of the 12th International conference on the World Wide Web*, pages 411–421. ACM.

Zhang, P. and von Dran, G. M. (2002). User expectations and rankings of quality factors in different web site domains. *International Journal of Electronic Commerce*, 6(2):9–33.