Application of Digital Ecosystem Design Methodology Within the Health Domain

Maja Hadzic and Elizabeth Chang

Abstract—We define a digital ecosystem (DES) as the dynamic and synergetic complex of digital communities consisting of interconnected, interrelated, and interdependent digital species situated in a digital environment that interact as a functional unit and are linked together through actions, information, and transaction flows. The design of DESs requires the integration of a number of different and complementary technologies, including agent-based and self-organizing systems, ontologies, swarm intelligence, ambient intelligence, data mining, genetic algorithms, etc. The integration of multiple technologies and the resulting synergetic effects contribute to the creation of highly complex, dynamic, and powerful systems. The application of DESs within different domains has the power to transform these domains by giving them a more intelligent and a more dynamic nature. In this paper, we illustrate how a DES design methodology can be used to systematically create a Digital Health Ecosystem (DHES). We address the key steps associated with the DES design and focus specifically on the use of the electronic health records within the DHES. The design methodology framework illustrated in this paper serves as a navigating tool during the design of DHESs.

Index Terms—Digital ecosystems (DESs), DESs design methodology, Digital Health Ecosystems (DHESs), digital health environment, digital health species (DHS).

I. INTRODUCTION

A. DESs Definition

THE DIGITAL-ECOSYSTEM (DES) concept has recently been adopted by the computer and information society to represent an innovative way of creating complex and dynamic information systems. A DES is analogous to the biological ecosystems in nature. Just as the biological ecosystems are composed of a variety of interrelated biological species that interact with each other and with their biological environment, so is a DES composed of a variety of interrelated digital species (DS) that interact with each other and with their digital environment (DE) [1]. The DES infrastructure is a DE which is populated by DS.

The first constituent of DESs is the *DS*. DSs are analogous to biological species. They transpose mechanisms from living organisms, like autonomy, viability, and self-organization, to arrive at novel knowledge and architectures [2], [3]. DSs usu-

ally form digital communities. The majority of DSs consist of "hardware" together with its associated "software." Together, the hardware and the software enable the DS to perform optimally and fulfill its role within the DES. The hardware is analogous to the body of biological species, whereas the software is analogous to the breath of life of biological species. In nature, a body without the breath of life is dead. Similarly, hardware is useless without any application running on it.

On the other hand, a DS can be "alive" and process information, but its actions may not bring any benefits to the DES or even may be damaging to the DES. DS can have hardware and software, but without a clear definition of the DS' purpose of existence, it can start acting maliciously toward itself and the whole DESs. For this reason, we define "purpose" as the third component that completes the definition of a DS. A DS's purpose corresponds to the human soul which constantly seeks to bring us toward the ultimate purpose of our existence. For example, a drug addict or a pedophile is a living being who has clearly lost the purpose for which the body was originally created. We are focusing on purpose-driven DES design where we need to design DSs so as to align their goals with the goals of the complete DES.

The second constituent of DESs is *DE*. DE is analogous to the biological environment. DE is an environment in which DSs jointly live, function, and relate.

We define a DES as follows: a DES is the dynamic and synergetic complex of digital communities consisting of interconnected, interrelated, and interdependent DS situated in a DE that interact as a functional unit and are linked together through actions, information, and transaction (AIT) flows.

B. DESs *for Various Domains*

Let us consider domain-driven DES design. Each domain (e.g., health, business, government, law, and industry) can have its own DES. In this way, the Digital Health Ecosystem (DHES), Digital Business Ecosystem (DBES) [4], Digital Government Ecosystem (DGES), Digital Law Ecosystem (DLES), Digital Industry Ecosystem (DIES), etc. can be created, which all jointly contribute to the design of the Universal DES (UDES). UDES is a superset of all DES, or

$$UDES = DHES \cup DBES \cup DGES \cup DLES \cup DIES \cup \dots.$$

Analogously, the universal biological ecosystem is a superset of the sea ecosystem, ocean ecosystem, polar ice-cap ecosystem, tropical-island ecosystem, etc. Furthermore, the tropical-island ecosystem is a superset of a rainforest ecosystem, mangrove-swamp ecosystem, and coral-reef

ecosystem. For example, in an analogous way, a DHES can be specifically designed for the health domain as a part of UDES. Furthermore, each health subdomain (e.g., dental, physiotherapy, mental, nutrition, and surgery) can have its own DES. This will result in DHES being a superset of Digital Dental Ecosystem (DDES), Digital Physiotherapy Ecosystem (DPES), Digital Mental Ecosystem (DMES), Digital Nutrition Ecosystem (DNES), Digital Surgery Ecosystem (DSES), etc, or

$$DHES = DDES \cup DPES \cup DMES \cup DNES \cup DSES \dots.$$

Various digital health species (DHS) can be designed and interconnected to form a collaborative network and link different hospitals, health services, general practitioners, pharmacies, health systems, health information resources etc., thereby producing outcomes that are highly beneficial for all parties involved.

C. Information and Transaction Flows

In the same way that nutrients and the energy cycle within the biological ecosystems keep the biological species alive, so do the information and transactions flow within the DES keep the DS alive. The information flow can be any idea that is expressed by a formal or natural language, digitalized and transported within the DES, and processed by computers or humans. In a DHES, information about a patient can be digitally captured in a personalized medical record in the form of a medical card. More complex tasks will not only be based on the information flows but will also involve transactions. These transactions may involve activities, such as money transactions between patient and pharmacist when purchasing prescribed medications. A DHES should be able to autonomously perform most of these transactions requiring minimal involvement of humans.

D. DESs Vision

The implementation of the DES in the different domains and for the different applications will not only improve but will also transform the traditional rigorously defined environments, such as centralized (client–server) or distributed (peer-to-peer) models, into loosely coupled, domain-specific, and demand-driven interactive digital communities [1]. A successfully designed and implemented DES should produce beneficial outcomes. This will attract other DS to join the DES and participate in the collaborative actions carried out within the DES. This will result in a strong collaborative and mutually beneficial environment. For example, a DHES that operates successfully would attract other health institutions to join the DHES rather than compete with it. A pilot DHES can be designed to be implemented within government health institutions and organizations. Private health institutions and organizations may be interested to join this DHES once they notice that the DHES is performing successfully and efficiently.

The aim of this paper is to clarify the concepts of DES, to introduce a five-step methodology for the design of DES, and to illustrate the implementation of this methodology within the health domain with specific focus on electronic health records (EHRs). Our ideas are inspired by the natural processes, and

we draw the analogy with the biological ecosystem throughout this paper.

II. E-HEALTH IN NEED FOR DESs

The implementation of EHR systems has gained momentum, particularly in the U.S., Canada, and U.K. [5].

A. Shortcomings of the Existing EHR Systems

The Australian government's EHR initiative Health *Connect* [6] conducted a number of trials in 2002, 2003, and 2004 in Tasmania, Northern Territory, Queensland, New South Wales, and South Australia. The project has shown success on a smaller scale, but the major problem of sharing crucial patient information has remained unsolved. It has been reported that every year, more than 3000 deaths in Australia are attributed to reasons that could have been prevented with more effective e-health systems [7]. A report by the Australian Broadcasting Corporation [7] from November 2007 states that "*one of the major killers of patients is the medical system's own chronic inability to share vital information.*"

Currently, a large number of medical practitioners in Australia use their own software to store and manage patient information. The format in which patient information is stored varies from one medical practitioner to another. Moreover, different medical institutions do not use systems that will enable them to exchange patient information, even in situations where they are treating the same patient.

We conclude: 1) that the lack of a standardized format in which patient information is being kept and 2) the lack of infrastructure to enable sharing of the information among different medical and health organizations and institutions are two of the main obstacles to the effective and efficient sharing of patient information.

Additional implementation barriers have been identified in [8]. These include financial expenditure, resistance to change, lack of office technology, lack of technological standards, increasing staff workloads, and healthcare culture, including attitude, workflows, [9] and relationships.

B. Standardization Efforts

The need for standardized or interpretable EHR is evident [10]. The two most popular standardization efforts use smart card and radio-frequency identification (RFID) technologies. These technologies support the creation of machine-readable medical records where the digital components are used to screen and read personal medical records.

A large number of companies (e.g., Almex Ltd. SmartCard Solutions, Smart Health Solutions, Electronia, and Giesecke & Devrient), provide smart health cards. The smart cards differ from each other in the way they store, process, and manage information, and they way they deal with the associated privacy and security issues.

RFID for people has attracted much attention in the last couple of years. Even though the Food and Drug Administration (FDA) approved VeriChip in 2003, they have reported a number of potential dangers associated with its use [11]–[13].

The choice has to be made carefully. Different factors must be considered, with a need to strike a balance between the advantages and possible problems that may arise.

C. Toward the Change for the Better

Effective EHR implementation and networking could save more than $81 billion annually through improved health-care efficiency and safety [14]. Moreover, the estimated savings could be doubled through the use of health information technology in the prevention and management of chronic disease.

Hillestad *et al.* [14] urge the implementation of the required changes within the health-care system. All hospitals, health institutions, and organizations and patients need to work together to identify mutually supportive roles that will enable them to move toward a shared vision.

A good solution to the current situation requires focusing on the outcomes such as 1) standardized EHRs and 2) high-capacity and quality infrastructure to support sharing of patient information and also on the transition period, namely, on the way we bring the current medical system into this new era of standardized and shared EHRs.

The least "painful" change for medical practitioners would include keeping the complexity behind the screen. One of the ways to achieve this is by giving medical practitioners the option of keeping the same interface (which can be done by limiting the change to the back-end system), or changing the interface to match the standardized format. For medical practitioners resisting the change, the translation between different health-record formats will be done behind the screen, and their view of the patient record will not change. This will give them the façade of the old system. In contrast, medical practitioners more receptive to change will choose the standard interface.

III. DHES's DESIGN

Understanding the origin of the biological ecosystem is of great value, particularly if we aim to design systems that mimic the biological ecosystem and are analogous to it. A great deal of discussion has arisen on this matter, and this ongoing debate has resulted in two major hypotheses: 1) the Big Bang and Evolution Theory and 2) the creation and intelligent design. Evolutionists claim that the universe is almost 14 billion years old and that evolution made way for interspecies conversion (one species to become another species e.g., apes becoming humans). Creationists believe that the Earth is less than 10 000 years old and that an intelligent mind (i.e., God) is the author of all creation in the same way that a painter is the author of a painting and a builder of a building. Creationists also believe in evolution, but this evolution is limited to intraspecies evolution which explains the different types of the same species, such as different breeds of a dog.

The biological ecosystem itself is a great source of inspiration and knowledge. If we are now to create DESs which are analogous to biological ecosystems, we need to adopt either the evolutionist or the creationist perspective. This understanding forms the basis of, and will affect, the way we view DESs and approach their design.

We have proposed a preliminary five-step methodology for the design of DES that consists of the following steps:
1) define roles of different DS;
2) define DS collaborations;
3) make the DS intelligent;
4) protect the DES by implementing security requirements;
5) enable, improve, and/or construct individual DS.

A. Define Roles of Different DS

When considering the capabilities of different DS, it is important to do the following.
1) Establish **intuitive (AIT) flow**. DES needs to be designed in a way that enables and supports intuitive AIT flow. One way to approach this problem is to follow the functional process(es) in our minds from beginning to end, and progressively identify corresponding roles needed to bring these process(es) to completion.
2) Identify **DS roles** required to establish the intuitive flows. The more complex the functional processes are, the greater is the diversity of roles required to carry these processes to completion.
3) Accordingly, identify different **DS types**. Lastly, we define different DS types associated with the identified roles. Sometimes, a DS type has multiple roles; other times, multiple DS types are required to jointly fulfill a specific role within the DES.

DS Functioning as One: A DES is usually composed of DSs with different but complementary capabilities. Within a DHES, we may have different kinds of DHS, such as medical devices, measuring apparatus, monitors, personal digital assistants (PDAs), computers, etc. As the different DSs possess capabilities that are complementary to each other, they must function as one in order to achieve the synergetic effect. The DSs need to work cooperatively, coordinate their actions, share the overall task, integrate their results, and so on. This synergy makes the DS interrelated, interconnected, and interdependent with regard to the same goal they share.

Example 1: Patient Making an Appointment: The need for increased efficiency and effectiveness within health-care domain has been emphasized by numerous studies [27]. For illustration purposes, we take a common scenario of a patient making an appointment to see a doctor (see Fig. 1). We can make use of a machine-readable card (DHS1) and a scanner (DHS2) to make this information accessible and visible using a computer which is linked into the DHES network (DHS3).

The access to the information available via medical card needs to be limited. The receptionist does not require access to the specific medical information, such as laboratory test results, in order to make an appointment. For example, the receptionist's access can be limited to information about patient demographics, appointments, and billing records. Other DHSs of the DHES network inform the receptionist about the available services and timetables of the doctors so that she can make an appointment relevant to the patient's need.

Why a card and not a microchip?: The use of a medical card is preferred over VeriChip technology (http://www.verichipcorp.com/). A VeriChip implant uses radio

DHS1: electronic medical record of patient A

DHS2: scanner

DHES

DHS3: receptionist's computer linked into DHES network

Personal info:
Patient demographics
Appointments
Billing records

+current need

DHES info:
Services
Personnel
Timetables
Transactions…

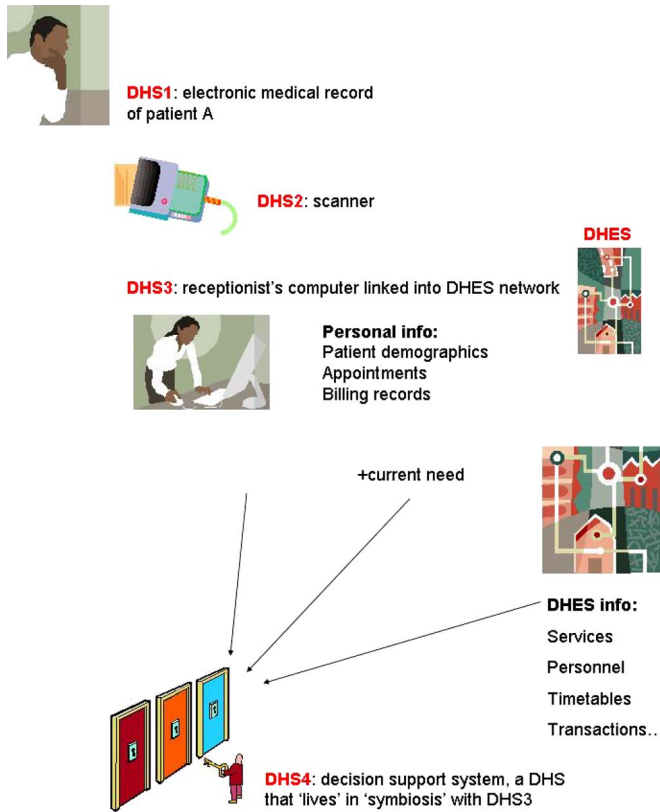**DHS4**: decision support system, a DHS that 'lives' in 'symbiosis' with DHS3

Fig. 1. Use of EMRs in DHES.

waves to transmit medical and financial account information to reader devices. The FDA approved VeriChip technology in 2004. However, there are several major negative effects associated with the VeriChip device [11]. These include adverse tissue reaction, electrical hazards, magnetic resonance imaging incompatibility, electromagnetic interference, failure of implanted transponder, migration of implanted transponder, and so on. Also, it is possible for the electrical currents to be induced in conductive metal implants which can result in potentially severe patient burns.

K. Albrecht, the founder and director of Consumers Against Supermarket Privacy Invasion and Numbering, said in a statement [11]: "*By omitting this information from their press material, the companies marketing the VeriChip have painted an inaccurately rosy picture of their product that could mislead consumers into believing the devices are completely safe.*"

Additional disadvantages of microchips were reported in 2007 [12]. The formation of malignant sarcomas and other cancers around or adjacent to the implanted microchips have been reported. Overall, it was concluded that the microchips had induced the cancers.

In addition to health risks, the FDA has also identified the problem of compromised data security [13]. Criminals can use a reader device to capture information from an implanted chip as well as duplicate this information and create a cloned chip with the same function. Additionally, the reader device can be used to track an individual. K. Albrech concludes that [13] "*That tracking potential, coupled with VeriChip's potential health risks, makes the VeriChip a very poor choice for medical patients seeking safety and security.*"
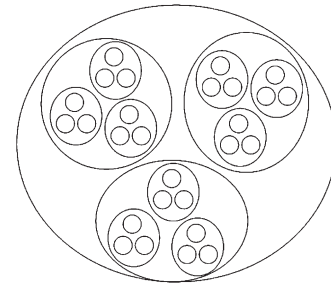

Fig. 2. Holonic structure.

Digital symbiosis: As shown in Fig. 1, a decision support system (DHS4) can be designed to help the receptionist in making decisions. More information on decision support systems can be found in [29]. In our example, DHS3 and DHS4 are dependent on each other and complement each other in the realization of their shared goal. Together, they form a complex relationship which is analogous to a symbiotic union in biological ecosystems.

As an example of biological symbiosis, we will consider vanilla plants which live in symbiosis with the Melipoma bee. Vanilla plants depend on a special kind of Melipoma bee for their pollination. The Melipoma bee lives only in Mexico, and, originally, the Europeans could not grow vanilla plants in Europe until they discovered this symbiosis. After this, they started carrying out artificial pollination of the vanilla plants in Europe. This amazing phenomenon poses some intriguing questions which, for some, may be difficult to answer. Was the vanilla plant or the Melipoma bee created first (or evolved first), and how could it survive without its symbiotic partner?

Coming back to our example from Fig. 2, we can apply this symbiotic phenomenon to our DES by designing DSs that will function together in mutually beneficial actions. We can say that DHS3 and DHS4 "live" in "symbiosis" with each other.

B. Define DS Collaborations

In the first step of our DES design methodology, we described how to identify different types of DSs according to their different roles within the DES. In this phase, we will highlight the importance of structural organization of the DS within a DES. These two steps may sound similar, but the difference is that in the first step, we defined DS roles within a DES, while in this step, we define the DS position within DES; i.e., we are concerned with inter-DS interaction. Furthermore, collaborations between different systems can be aligned as illustrated in [28].

The aim of this step is as follows.

1) To determine the **behavior** of **DS** that will enable the DES to function in the most efficient way. When determining DS behavior, it is important to enable the functional process within the DES to easily flow to completion and the communication between different DS to be established at any point. The DES needs to function effectively and efficiently through the synergetic effort of different DS.

The DS behavior can be classified into three main categories.

a) Self-organization [2], [3], [15]. The self-organization is characterized by autonomous organization of DS in a way they consider to be the most optimal one. A system can be designed to predict interactions between the different DS [16].

b) Predetermined organization. In the predetermined organization, DS do not have the freedom to group with each other according to their own choices and preferences but are communicating and collaborating with each other using fixed paths defined by their designer. This situation does not really correspond to the natural ecosystems, but this approach may be used in circumstances where greater control of DS actions is needed.

c) Hybrid organization. This is a combination of the previous two or the ability of both self- and predetermined organization. The combination of self- and predetermined organization can be used in cases which not only require greater control over DS actions in some places but also permit autonomous actions of DS in other places. This may be the case in the design of DHESs. On one side, it may be needed to closely monitor and control the performance of some important DHSs. The DHES needs to be designed to allow for predetermined organization in the case of these critical DHSs. On the other side, if the DHES is established between a number of loosely coupled health institutions and organizations and if this DHES envisages other members freely joining it, it needs to permit the self-organizing behavior of other less critical DHSs.

2) Define the optimal DES structure. The second step in defining DS collaboration by defining the optimal DES structure includes focusing on the organization of DS within the DES. The DS can be organized in the following way:

a) chaotic;
b) orderly;
c) semiorderly.

In most cases, those DES structured in an orderly way are expected to function more efficiently. These orderly structures may take any shape, e.g., holonic structure is associated with self-organizing swarm intelligent [17] behavior. Here, the autonomous DSs group together to form hierarchies and hierarchies of hierarchies (holons) [15]. In Fig. 2, every holon is represented by a circle and each of the three top-level holons is itself a hierarchy of three holons, each hierarchy being of three holons.

C. Make the DS Intelligent

Depending on the role of each DS within a DES, it will certainly require some amount of intelligence that will enable it to perform its actions optimally. In simpler cases, a knowledge base may be used to equip the DS with this intelligence. In

TABLE I
ONTOLOGY VERSUS KNOWLEDGE BASE

	Ontology	Knowledge base
actions	communication	problem solving
knowledge	domain knowledge	operational knowledge & restricted domain knowledge
objectives	conceptual structures of a domain	particular states of a domain
consistency	facts always true	facts true for a particular state of affairs
applicability	shared by community	specific to system

more complex cases where DSs need to act more intelligently, ontologies [18] may be used to encode this knowledge.

Ontology Versus Knowledge Base: We summarize the main differences between ontology and knowledge bases in Table I [19].

Ontology defines a vocabulary used by agents to perform meaningful communications about a domain of interest. In contrast, a knowledge base contains the knowledge agent needs in order to solve problems or answer queries about a specific domain.

The ontologies describe domain knowledge, while knowledge bases describe operational knowledge or restricted domain knowledge.

While an ontology is a conceptual structure of a knowledge domain, a knowledge base models particular states of a knowledge domain required to enable a particular knowledge-based system to carry out problem solving.

A knowledge base models facts that are true for these chosen states of the knowledge domain. Conversely, ontology models facts that are always true within the chosen knowledge domain.

As the ontologies describe a knowledge domain and hold facts that are always true within this knowledge domain, they can be shared across all communities within this knowledge domain and used by any applications created for this knowledge domain. Conversely, a knowledge base is specific to the system and can be used only by the system for which it was created.

Ontologies for DES: Generally, DES should function more effectively and efficiently when decisions and actions that are carried out within this DES are supported by ontologies. Ontology is used for representing the knowledge domain and may be used at different levels and for different purposes within a DES, including information presentation, information retrieval, reasoning, and communication between different DS.

1) Information presentation and information retrieval. Ontologies usually have a tree-structured form and enable easier identification of the conceptual relationships in data. A DS may request information from another DS before it performs, for example, money transactions. The use of ontologies enables the DS to access and retrieve the requested information in a more controlled and systematic way.

2) Reasoning. An ontology may be used to equip DSs with reasoning capabilities that will enable them to function optimally and support tasks such as carrying out of timely
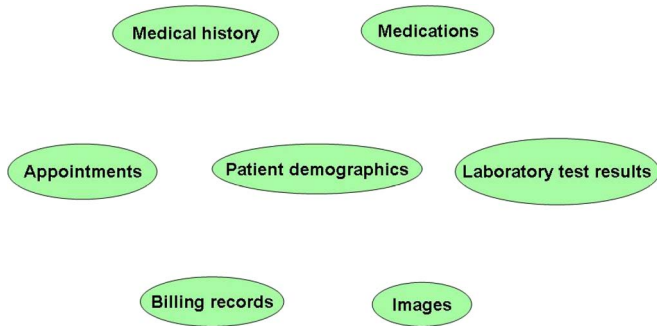
Fig. 3.   Top-level ontology concepts in EMR.

and efficient interventions. For example, an ontology designed to contain knowledge about the structure of the DES and the functions of various DSs within this DES can specifically be used by DSs that have the functions of managing, directing, and assigning tasks to different DSs.

3) Communication. If the different DS are to function efficiently and in unity, they need to share a common understanding of the domain concepts. An ontology can be used to provide the vocabulary needed for communication and enables cooperative DS to communicate with each other, coordinate their actions, and function in unity.

Ontologies for DHES: In a DHES, most DHSs need to understand the electronic medical record (EMR) information. For this reason, the standardization of EMR data needs to take place. Ontologies can be used for this purpose [1]. As the ontologies add semantics to the model, the implementation of the ontologies within DES enables meaningful interpretation of the data.

We propose the design of the generic EMR ontology (GEMRO) to capture and represent all the EMR information. This information may include data about patient demographics, medical history, medications, laboratory test results, images such as radiology images or clinical photographs, billing records, appointments, etc. (Fig. 3). Assigning instances (instantiation) to the GEMRO concepts results in specific EMR ontology (SEMRO). The SEMRO can be used within DHES to act as personal medical records. The DHS1 from Fig. 1 (medical card) includes SEMRO which contains medical information specific to this patient and was created by filling in this specific information in a data template as defined by the GEMRO concepts.

Within a DES, all DSs commit to a common ontology. This means that they agree on the meaning of the concepts and relationships between those concepts as defined in the ontology and agree to use the ontology in a coherent and consistent manner. In our example of DHEs, all DHSs commit to the common GEMRO. Because ontologies are stored as machine-readable files, DHS can read and process instantiated GEMRO files or SEMRO (personalized EMRs). In Fig. 1, DHS2 reads the information from DHS1 and makes it available via DHS3 to the receptionist.

Ontology Design and Evaluation: In order to make the DS intelligent, we need to do the following.

1) **Design** the ontology (or knowledge base). A number of different ontology-design methodologies exist:

a) knowledge engineering methodology (KEM) [20];
b) DOGMA methodology [21];
c) TOVE methodology [22];
d) METHONTOLOGY [23];
e) ontoagent methodology [24].

2) **Evaluate** the ontology (or knowledge base). An ontology may be evaluated from different points of view:

a) The technical point of view, assessing the quality of the designed ontology. For the purpose of evaluation of quality of the designed ontology, we refer to the five criteria suggested by Gruber [18]:
   i) clarity;
   ii) coherence;
   iii) extendibility;
   iv) minimal encoding bias;
   v) minimal ontological commitment.

b) The practical point of view, assessing the usability of the designed ontology. The usability of the designed ontology can be evaluated through the following:
   i) matching of the originally defined ontology purpose and scope against the designed ontology;
   ii) assessment of conceptual coverage;
   iii) assessment of practical usefulness.

c) The mathematical point of view, evaluating the ontology design through use of mathematical models. As ontologies can be viewed as sets of related concepts, set theory is usually used for constructing mathematical models to define and reason about ontological models [25].

D. Protect DES by Implementing Security Requirements

Security Aspects: Security plays an important role in the development of DES. When developing DES, our goal must be to provide as much security as possible. Within a DES, the following security needs [26] should be addressed.

1) **Authentication**: proving the identity of DS. The interacting DSs need to prove identity to each other to ensure that, for example, the confidential information has been exchanged correctly. For example, the DHS2 (scanner) from Fig. 1 needs to prove its identity to DHS1 (medical card) in order to access the personalized medical information available through DHS1.

2) **Availability**: guaranteeing the accessibility and usability of information and resources to authorized DS. The authorized DS must be able to access target information when needed. For example, DHS of the same type as DHS2 and belonging to the same DHES must be able to access information provided by DHS1. The information contained within DHS1 should not be made available to the DS of other DESs.

3) **Confidentiality**: information is accessible only to authorized DS and inaccessible to others. The access to specific information needs to be controlled by allowing only authorized DS to access this information. Further limitation of information access can be introduced through provision of different accessing points. We will see further in the example shown in Fig. 5 that the doctor has access to additional information that is invisible to the receptionist.

4) **Nonrepudiation**: confirming the involvement of a DS in a certain interaction. It is highly advisable to record and confirm every interaction within DES and all DSs involved in this interaction. For example, the event of DHS2 accessing information from DHS1 needs to be recorded and confirmed to ensure that the correct medical record was accessed, and not the medical record of some other patient.

5) **Integrity**: assuring that the information remains unmodified from source entity to destination entity. It is very important not to unwittingly modify information while circulating within a DES. In our example from Fig. 1, DHS2 and DHS3 must assure that the information displayed via DHS3 is identical to the information specified in DHS1.

The aforementioned properties are related to the information flow within the DES. Additional security requirements relate to the action and transaction flows and these include the following.

6) **Compliance**: acting in accordance with the given set of regulations and standards. The designer must give the DSs clear instructions regarding their roles and tasks within the systems that will enable them to be successful in performing their actions for their own benefit and for the benefit of the whole DES. From the example in Fig. 1, DHS1 (medical card) must correctly store medical information and make this information available to authorized DSs when requested.

7) **Service**: DSs need to be designed to serve one another for mutually beneficial purposes. DS needs to fulfil its role within the DES and perform tasks assigned to it, but it also needs to serve other DS when needed. If some DSs are under malicious attack, other members of the DES need to make their best efforts to protect those DSs and the whole DES.

8) **Dedication**: complete commitment of the DS to the DES goal and purpose. Every DS has a unique role within DES. Some DS roles are more important than others, but, generally speaking, all roles are important in jointly achieving the overall DES goal. Different DSs are working on different aspects of the overall goal, and they act like different organs in a body. Heart, lungs, liver, stomach, eyes, etc., all have unique functions and, in unity, enable the body to live and function optimally. Analogously, all DSs have unique functions within the DES and, in unity, enable the DES to live and function optimally. All DSs must commit to the overall goal and purpose of the DES and function in unity and as one in order to jointly achieve this goal.

Awareness of the Environment: Security issues may arise both inside and outside the DES, such as during interaction with the environment. As different DSs play different roles within the DES, with respect to security, some of them will be more important, and may be assigned more security responsibilities than others [26]. For this reason, the designer should consider the following:

1) role that each individual DS plays in the security of the DES;

2) actions that operate within the DES that are most critical with respect to security;

3) actions that operate during the interaction of the system with the environment that are most critical with respect to security;

4) factors outside the DES that are most critical with respect to security;

5) part of the DES that is most susceptible to attack from the outside.

Knowledge to Protect: The designer must equip the DS with the knowledge that addresses the eight aforementioned security requirements. The value of this knowledge is great, as it has the power to protect the individual DS and the whole DES from potential dangers and destruction. For this reason, it needs to be developed by the designer with great care and followed by the DS precisely.

Simply stated, this means that each DS needs to know what is right and what is wrong and must be able to willingly control and direct its actions toward outcomes beneficial for itself and for all members of the DES. We, as creators of DES and DS, have better knowledge than the DS themselves and better insight into the DS' nature, strength, and weaknesses and generally speaking, into different aspects and problems that may arise within the DES. For this reason, we are best suited to identify, format, and give the required knowledge to the DS. This can be done as follows:

1) **identify** critical security issues within the DES;
2) effectively **address** those issues;
3) clearly set the **boundaries;**
4) **encode** this knowledge in a form that can be understood by all DSs of the DES;
5) **equip** the DSs with this knowledge.

After the identification of critical security issues and effectively addressing those issues, it is necessary to implement this knowledge within the DES. The knowledge needs to be encoded in a format understandable by DS. The DSs will use this knowledge in situations when their individual security and/or security of the DES is in danger. They will use this knowledge to direct their actions toward protection and productivity.

E. Enable, Improve, and/or Construct Individual DS

DSs need to be designed to meet the requirements associated with their roles within the DES. In some cases, some existing DSs need to be enabled and/or improved in order to be used within the DES. In other cases, no existing DS can be used to fulfill the role of a specific DS within DES, so this DS needs to be designed.

Enabling and Improving Existing DS: Some DS roles can be fulfilled by existing devices, such as PDAs, computers, monitoring devices etc., and we do not need to redesign them. However, in most cases, redesign is needed to enable them to be used within a specific DES. The addition of some special software programs and features may be required in order to enable DS to be used within a DES. For example, the DHS3 in Fig. 1 is a simple computer, and special software has been added to it to enable this computer to function as DHS3 within the DHES. In this case, we need to design a software program

to access, read, understand, and manage EMRs (SEMRO software), and another software program to schedule the appointments (appointment software). These special features may be unique to DHS3, or they may be added to all DHSs of the DHES in situations where the DHSs need to perform their task collectively.

Constructing New DS: For all the roles that could not be fulfilled by the existing DS, new DSs need to be designed. For example, we need to design a machine-readable medical card and scanner associated with this card if no existing ones are suitable to be used by the proposed system.

During this construction step, it is important to do the following:

1) identify required DS **features**;
2) design software and hardware that will support these features [**Digital Organs** (DOs)];
3) **assemble** the DO into a DS body;
4) **evaluate** the DS.

On the basis of the required roles, we can identify desired DS features. The identified DS features can then be incorporated into the design of DOs. The different DOs will then be assembled together in a body to form DS. The DS is usually evaluated from the practical point of view, namely, how well the DS functions against the proposed objectives.

In our example, the SEMRO software and appointment software are two DOs of the DHS3. In nature, this would correspond with the identification of roles of an animal within a biological ecosystem and designing organs to specifically fulfill these roles. For example, this would involve designing eye, brain, blow hole, flippers, heart, liver, lung etc. for a dolphin and intelligently assembling these organs into a body for an optimum result (e.g., putting eyes in the direction of the movement of a dolphin and not, for example, behind the dorsal fin).

Variety of Species: A variety of species can be found in nature. We have dolphins, sharks, butterflies, roses, willows, lions, zebras, etc. We also notice a great variety within specific species. Many different kinds of butterflies exist with different colors, combinations of colors, shapes, sizes, various patterns, etc. We look at this biological variety and reason how this great variety can be achieved within DES. We consider the three possibilities shown in Fig. 4. The different DOs are represented by different shapes, and the different content of the same DO is represented by a different color. We identify the three main ways of creating the variety of DS within a DES.

1) Variety in content of DO. The first kind of variety is created by using the same DO to construct different DSs but changing the *content* of the DO for different DSs. This approach is used when creating DSs of the same type.

   In our example shown in Fig. 5, both receptionist and doctor have a computer. The DHS3 and DHS5 both have SEMRO software in common, but the SEMRO software of a receptionist differs from the SEMRO software of a doctor, as the doctor needs additional access to information about the patients' medical history, medications, results from laboratory tests, etc. In nature, this kind of variety corresponds with the difference between
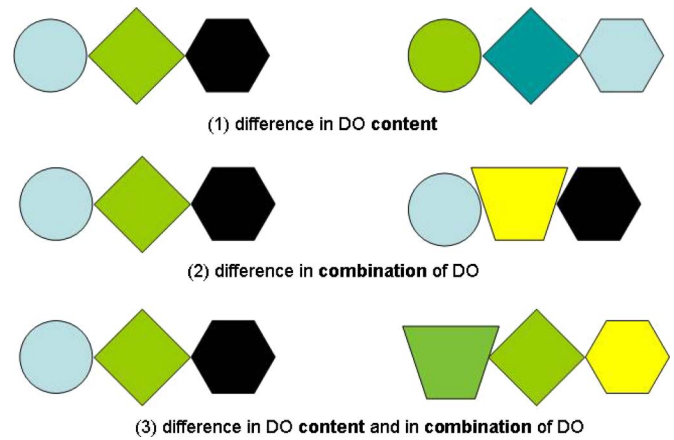
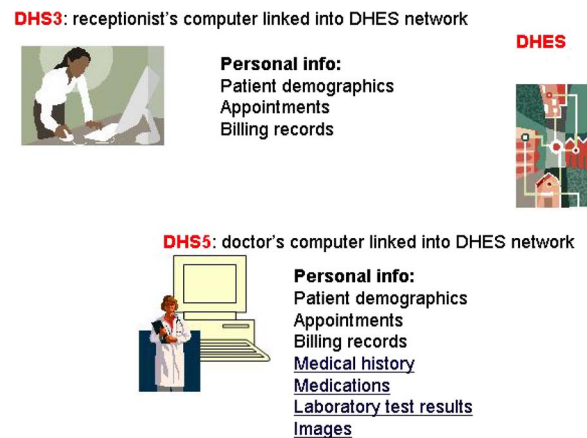

Fig. 4. Creating variety of DSs within DES.

Fig. 5. Two DHSs of the same type.

two organisms of the same species. For example, we humans are so different from each other but all have originated from parents; one male and one female parent. The inbuilt mechanism of genetic crossover (exchange of genetic material between parents) has provided a way for the creation of a completely new individual; an individual different from its parents and with genetic material being a unique combination of parental genetic material. In the computer and information society, we have already started applying these genetic mechanisms that are commonly known as genetic algorithms. The biological mechanisms are much more complex than the current genetic algorithms, and further study of biological processes and associated regulating mechanisms is needed to help develop digital ecosystems with greater variety, complexity, and effectiveness.

2) Variety in combination of DOs. The second kind of variety is created by using the same DOs but changing the *combination* of these DOs for different DSs. This approach can be used in situations such as designing the different DSs to function as one and linking them into interdependence where one cannot exist without the other. For example, DHS4 from Fig. 1 is a decision support system and needs DHS3 to be implemented

on. DHS3 needs DHS4 to function intelligently. We say that DHS3 and DHS4 live in symbiosis with each other. In nature, this phenomenon is commonly known as symbiosis, and we have given an example of vanilla plants and the Melipoma bee. Different kinds of symbiosis exist: mutualism (where both species benefit), commensalisms (where only one species benefits without influencing the other), and parasitism (where only one species benefits at the cost of the other).

3) Variety in both content and combination of DOs. The third and most common option of creating variety between DSs is to create DSs to differ from each other in the *combination* of the DOs used to design them and in the *content* of these DOs. The greatest diversity of DSs is created in this way. Greater variety can be achieved through a decreased number of similarities in the DO content and in their combination resulting in larger differences between DSs. In nature, this kind of variety is created between different species. Variety on a smaller scale is created between the species that belong to the same family of species, such as variety between all species belonging to the Equidae (horse family including horse, zebra, and donkey). Variety on a larger scale is created between species that belong to the different families and even to the different kingdoms, such as differences between the species that belong to the plant and animal kingdoms.

Digital Versus Genetic Code: In the digital world, we create DSs, and each DS is encoded by the sequence of atomic software encodings, namely, unique sequences of zero and one. All the biological species are also encoded. Each organism is uniquely defined by its own genetic sequence found in the deoxyribonucleic acid (DNA) or ribonucleic acid molecule. Most biological species contain DNA as its genetic material and are encoded by unique sequences of only four different nucleotide bases: adenine (A), cytosine (C), guanine (G), and thymine (T). Two chains of nucleotide base sequence are linked together and twisted into the shape of a double helix. This unique DNA sequence directs the assembly of amino acids into proteins that have a specific function in the body. An abnormal change in the genetic material (i.e., mutation) may result in the formation of nonfunctional protein and development of a disease. For example, Huntington's disease is a genetic disorder that develops in people who inherited a larger than normal Huntington gene. This larger gene produces abnormal protein that begins to kill brain cells in middle age.

The whole microscopic world of genetics and associated mechanisms such as crossing over, replication, translation, mutation, regulation, etc., form a very interesting source of inspiration for both information and computer scientists.

IV. OVERVIEW OF THE DESIGN METHODOLOGY

The design of a DES can be followed through the five phases shown in Fig. 6. When the preliminary DES is implemented, some additional requirements arise so that the corresponding design phases need to be repeated until the designer is satisfied with the outcomes.
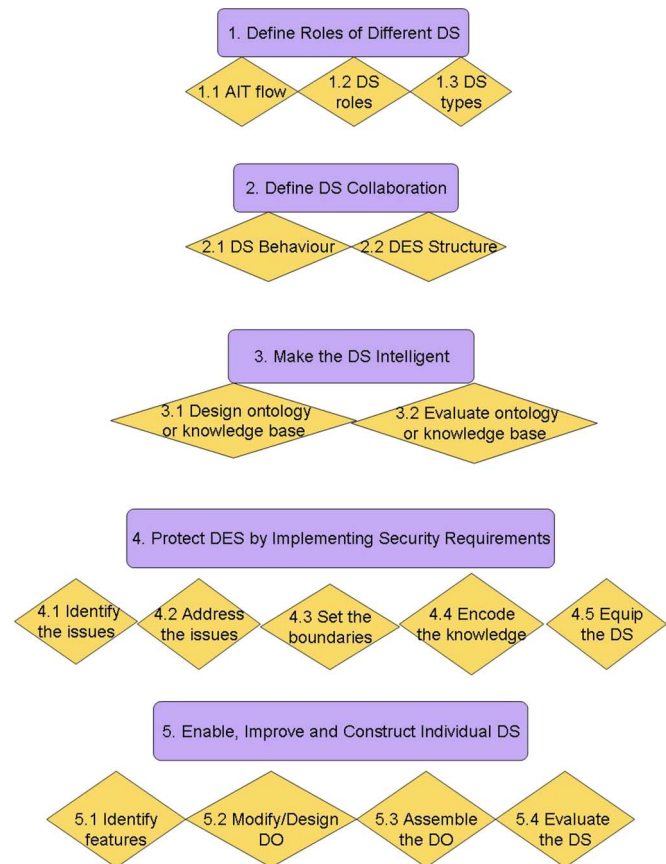

Fig. 6. DES design methodology.

In the first phase *Define Roles of Different DS*, intuitive AIT flow is defined, required DS roles identified, and corresponding DS types proposed.

In the second phase *Define DS Collaboration*, DS behavior is defined (which can be self-organization, predetermined organization, and self- and predetermined organization) as well as their organizational structure (which can be chaotic, semiorderly, or orderly).

The third phase *Make the DS Intelligent*, deals with the issue of DS intelligence. Ontologies and knowledge bases can be used to equip DS with intelligence, and they need to be evaluated before their use and implementation within DES.

In the fourth design phase *Protect DES by Implementing Security Requirements*, the security issues associated with 1) information flows (authentication, availability, confidentiality, nonrepudiation and integrity) and 2) actions and transaction flows (compliance, service, and dedication) are carefully addressed and the boundaries clearly set. This knowledge is encoded in the common language, and the DS are equipped with this knowledge.

In the fifth phase *Enable, Improve*, and *Construct Individual DS*, special features are added to the existing DSs to enable them to function efficiently within the DES, and new DSs are designed. The design of new DS includes identification of the desired features, designing of DO to mirror these features, assembling of the DO into a DS body, and evaluation of the designed DS. The variety of DSs is a result of the variety of the DO sand their contents.

V. CONCLUSION AND FUTURE WORK

We have explained DES through analogy with biological ecosystems and presented a conceptual framework of the DES design methodology. As we are creating DES, a "creativist" approach was taken and a methodology was developed to address the different aspects of DES including DS roles, organization, intelligence, security, and individual design.

An example of DHES, specifically, the handling of EMRs within this DHES system, was used to illustrate the implementation of the DES design methodology framework within the health domain.

The stepwise approach to the DES design process gives a better insight and increases control over the design process. We discussed each design step individually and highlighted its main characteristics and importance in the whole DES design process. Specific issues have not been discussed in great detail, and they fall outside the scope of this paper. We have given only a brief overview and the major steps associated with the DES design. This methodology is to be improved and refined as more experience is gained with the design of such systems.

REFERENCES

[1] M. Hadzic, T. Dillon, and E. Chang, "Use of digital ecosystem and ontology technology for standardization of medical records," in *Proc. Inaugural IEEE Int. Conf. DEST*, 2007, pp. 595–601.

[2] G. Briscoe and P. De Wilde, "Digital ecosystems: Evolving service-orientated architectures," in *Proc. 1st Int. Conf. Bio-Inspired Models Netw., Inf. Comput. Syst.*, 2006, vol. 275.

[3] G. Briscoe, S. Sadedin, and G. Paperin, "Biology of applied digital ecosystems," in *Proc. Inaugural IEEE Int. Conf. DEST*, 2007, pp. 458–463.

[4] E. Muntaner-Perich and J. L. de la Rosa Esteva, "Using dynamic electronic institutions to enable digital business ecosystems," in *Coordination, Organizations, Institutions, and Norms in Agent Systems II*. Berlin, Germany: Springer-Verlag, 2007.

[5] A. Cornwall, "Electronic health records: An international perspective," *Health Issues*, no. 73, pp. 19–23, 2002.

[6] D. Rowlands, "HealthConnect: A health information network for all Australians," in *Person-Centered Health Records: Toward Healthy People*. New York: Springer-Verlag, 2005, pp. 242–258.

[7] N. Grimm, Outdated Medical System Threatening Lives, 2007. [Online]. Available: http://www.abc.net.au/7.30/content/2007/s2050008.htm

[8] Accenture's Health & Life Sciences Group, Electronic Health Records Survey: Achieving High Performance in Health Care, 2005. [Online]. Available: http://www.worldcongress.com/events/nw600/pdf/Electronic%20Health%20Records%20Survey%20FINAL%200705.pdf

[9] S. Kim and R. E. Giachetti, "A stochastic mathematical appointment overbooking model for healthcare providers to improve profits," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 36, no. 6, pp. 1211–1219, Nov. 2006.

[10] A. T. S. Chan, J. Cao, H. Chan, and G. Young, "A web-enabled framework for smart card application in health services," *Commun. ACM*, vol. 44, no. 9, pp. 77–82, Sep. 2001.

[11] WorldNetDaily.com, Implantable Chip's Medical Dangers, 2003. [Online]. Available: http://www.infowars.com/print/bb/implantable_chips_medical.htm

[12] K. Albrecht, Microchip-Induced Tumors in Laboratory Rodents and Dogs: A Review of the Literature 1990–2006, 2007. [Online]. Available: http://www.antichips.com/cancer/albrecht-microchip-cancer-full-paper.pdf

[13] CASPIAN, FDA Letter Raises Questions about VeriChip Safety, Data Security, 2004. [Online]. Available: http://www.spychips.com/devices/verichip-fda-report.html

[14] R. Hillestad, J. Bigelow, A. Bower, F. Girosi, R. Meili, R. Scoville, and R. Taylor, "Can electronic medical record systems transform health care? Potential health benefits, savings and costs," *Health Affairs*, vol. 24, no. 5, pp. 1103–1117, Sep./Oct. 2005.

[15] M. Ulieru, "Internet-enabled soft computing holarchies for e-health applications," in *New Directions in Enhancing the Power of the Internet*. Berlin, Germany: Springer-Verlag, 2003, pp. 131–166.

[16] S. M. Lee and A. R. Pritchett, "Predicting interactions between agents in agent-based modeling and simulation of sociotechnical systems," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 38, no. 6, pp. 1210–1220, Nov. 2008.

[17] P. M. Kanade and L. O. Hall, "Fuzzy ants and clustering," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 37, no. 5, pp. 758–769, Sep. 2007.

[18] T. Gruber, "Towards principles for the design of ontologies used for knowledge sharing," *Int. J. Human Comput. Stud.*, vol. 43, no. 5/6, pp. 907–928, Nov./Dec. 1995.

[19] T. Dillon, E. Chang, M. Hadzic, and P. Wongthongtham, "Differentiating conceptual modelling from data modelling, knowledge modelling and ontology modelling and a notation for ontology modelling," in *Proc. 5th APCCM*, 2008, pp. 7–17.

[20] M. Uschold and M. Gruninger, "Ontologies: Principles, methods, and applications," *Knowl. Eng. Rev.*, vol. 11, no. 2, pp. 93–155, 1996.

[21] J. De Bo, P. Spyns, and R. Meersman, "Creating a "DOGMAtic" multilingual ontology infrastructure to support a semantic portal," in *Proc. OTM Workshops*, vol. 2889, *Lecture Notes in Computer Science*, 2003, pp. 253–266.

[22] M. Gruninger and M. Fox, "Methodology for the design and evaluation of ontologies," in *Proc. Workshop Basic Ontological Issues Knowl. Sharing (IJCAI)*, 1995, pp. 1–10.

[23] M. Fernandez, A. Gomez-Perez, and N. Juristo, "METHONTOLOGY: From ontological Art towards ontological engineering," in *Proc. AAAI Spring Symp. Ser. Ontological Eng.*, 1997, pp. 33–40.

[24] M. Hadzic and E. Chang, "Onto-agent methodology for design of ontology-based multi-agent systems," *Int. J. Comput. Syst. Sci. Eng.*, vol. 6, no. 99, pp. 65–76, 2007.

[25] C. Wouters, T. S. Dillon, J. W. Rahayu, E. Chang, and R. Meersman, "Ontologies on the MOVE," in *Proc. 9th Int. Conf. DASFAA*, 2004, pp. 812–823.

[26] H. Mouratidis, P. Giorgini, and G. A. Manson, "Modelling secure multi-agent systems," in *Proc. 2nd Int. Joint Conf. AAMAS*, 2003, pp. 859–866.

[27] T.-H. Cheng and P. J.-H. Hu, "A data-driven approach to manage the length of stay for appendectomy patients," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 39, no. 6, pp. 1339–1347, Nov. 2009.

[28] X. Zhao, C. Liu, Y. Yang, and W. Sadiq, "Aligning collaborative business processes—An organization-oriented perspective," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 39, no. 6, pp. 1152–1164, Nov. 2009.

[29] J. E. Kottemann, K. M. Boyer-Wright, J. F. Kincaid, and F. D. Davis, "Understanding decision-support effectiveness: A computer simulation approach," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 39, no. 1, pp. 57–65, Jan. 2009.

Maja Hadzic holds PhD in Health Information Systems and Master's Degree in Biochemistry. She is currently a Dr Maja Hadzic is a Postdoctoral Research Fellow at the Digital Ecosystems and Business Intelligence Institute of the Curtin University of Technology. She has a multi-disciplinary background and capacity to work across different domains. Her current research interests include Digital ecosystems, Ontologies, Multi-agent Systems, Data Mining and Mental Health.

Elizabeth Chang is Professor at the Curtin University of Technology. She is the Director of the Digital Ecosystems and Business Intelligence Institute. Professor Chang has an extensive background, knowledge and skills in academia, commerce and industry. She has done significant work in the areas of Ontologies, XML Technologies, Data Mining of Complex Data, P2P Applications, Trust, Reputation and Recommendations Technologies, Software Engineering and Fuzzy Systems Applications in IT and Internet Systems.