PUB - CBS EEB - EA - 39861-1

# Methodology Framework for the Design of Digital Ecosystems

M. Hadzic, E. Chang, T. Dillon

Digital Ecosystems and Business Intelligence Institute, Curtin University of Technology
GPO Box U1987, Perth WA 6845, Australia
e-mail: (m.hadzic, e.chang, t.dillon)@curtin.edu.au

***Abstract*—Digital Ecosystems have recently been introduced into the computer and information societies. Digital Ecosystem (DES) is the dynamic and synergetic complex of digital communities consisting of interconnected, interrelated and interdependent Digital Species (DS) situated in a Digital Environment (DE) that interact as a functional unit and are linked together through actions, information and transaction flows. DES transpose mechanisms from living organisms like autonomy, viability and self-organisation to arrive at novel knowledge and architectures. The proposed DES embraces a number of different technologies such as ontologies, agent-based and self-organizing systems etc. The synergetic effects of these methodologies results in a more efficient, effective, reliable and secure system.**

**DES is still in its early implementation phase. No clear methodology for digital ecosystem design exists yet. In this paper, we propose a methodological framework that consists of five phases and addresses many different aspects of the DES design. In this methodology, we focus on the key factors associated with the DES design such as roles of different digital components within the DES, organisation and collaboration of the digital components, their individual design along with intelligence and security within the DES. More details are introduced at each step and every sequential step takes the DES to a higher level of elaboration. This methodological framework allows better control over the design process and serves as a navigating tool during DES design.**

***Index Terms*—Digital Ecosystems, methodology for design of Digital Ecosystems, Digital Species, Digital Environment**

## I. Introduction

The concept of Digital Ecosystems (DES) has been recently adopted by the computer and information society. A DES is analogous in its essence to the complex ecological community in nature. It is a dynamic, complex and adaptive system that is composed of a variety of interrelated digital parts that interact with each other and with their digital environment [1, 2].

The DES infrastructure is a Digital Environment (DE) which is populated by Digital Species (DS) [3]. DS are analogous to biological species and are usually forming communities. DS consist of hardware together with its associated software. The hardware is analogous to the body of biological species whereas the software is analogous to the life of biological species. In nature, a body without life in it is dead. Similarly, hardware without any application running on it is useless. DE is analogous to the biological environment. A DE is an environment in which DSs are situated and in which they live and function. DSs together with a DE form a dynamic and interrelated complex DES. DES transposes mechanisms from living organisms like autonomy, viability and self-organisation to arrive at novel knowledge and architectures.

We define DES as follows: *A Digital Ecosystem is 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 information flow can be any idea that is expressed by a formal or natural language, digitalised and transported within the DES and processed by computers or humans. More complex tasks will be based on the information flows but will also involve transactions. A DES transcends the traditional, rigorously defined collaborative environment, such as centralised (client-server) or distributed (such as peer-to-peer) models into loosely coupled, domain-specific and demand driven interactive digital communities. A DES is characterised by the beneficial activities that attract other DS to participate and benefit from it. It is not one, but there are a huge number of not related or interrelated processes happening at the same within a DES. Most of these processes are based on knowledge sharing and cooperation.

As a new technology, the DES are relatively unknown or vaguely represented and misunderstood by many. The aim of this paper is to clarify the concepts of the Digital Ecosystem, Digital Species and Digital Environment and to provide an insight in the design of such a system. Our ideas are inspired by the natural processes and we draw the analogy with the biological ecosystem throughout this paper. In the following section, we describe the five phases of the DES design.

## II. DIGITAL ECOSYSTEMS DESIGN

We propose a methodology for the design of DES that consists of the following five steps:

1. define goals of different digital species
2. make the DSs intelligent
3. define DSs' collaborations
4. enable, improve and/or construct individual DSs
5. protect the DES by implementing security requirements

### A. Define goals of Different Digital Species

A DES functions best if it is composed of DSs with different but complementary capabilities. Mobile phones, PDAs, computers, etc. with different applications running on them are all different DSs that are situated within DE and all together create a DES. The different DSs need to function as a whole; they need to work cooperatively, coordinate their actions, share the overall task, integrate their results, and the like. Though the DSs differ from each other, they are interrelated, interconnected and interdependent.

When considering capabilities of different DSs, it is important to:

- establish intuitive Actions, Information and Transaction (AIT) **flow**;
- identify DSs' **goals** required to establish the intuitive flows; and accordingly
- identify different DSs **types**.

As a DES is composed of cooperative DSs, an intuitive action, information and transaction flow needs to be established. We follow these functional process(es) in our minds from beginning to end and progressively identify different aspects of the process(es).

The overall process is composed of a number of sub-processes where each sub-process is associated with its own goals. Corresponding goals that are needed to bring those sub-processes and the overall process to completion need to be clearly defined. A diversity of goals emerges in this way.

All DSs associated with the identified goals need to be clearly specified. It may be necessary to assign exclusive tasks and responsibilities to different DSs. Different DS types are identified according to their goals within the system.

### B. Make the Digital Species Intelligent

There are different ways to provide the DSs with the intelligence. Depending on their role within DES, each DS will require a certain amount of intelligence. In some cases, the required intelligence will be relatively small while in other cases DSs will perform more complex tasks and will require an intelligence exceeding an average level. In simpler cases, knowledge base may be used while in more complex cases DSs need to act more intelligently. Ontologies may be used for this purpose [4]. In the situations where different formats are used within the same DES and the associated DSs need to communicate with each other, the different formats need to be inter-translatable or the two DSs need to be provided with an additional knowledge that will enable them to communicate effectively.

Generally, DES should function more effectively and efficiently when based on ontologies. For this reason, we focus on the ontologies in this section. Ontology is used for representing the knowledge domain and may be used at different levels and for different purposes within a DES, such as:

- task-sharing processes
- information retrieval, analysis and manipulation
- information presentation
- communication between different DSs

Ontology provides the DSs with knowledge of the goal structure and can be used during the process of goal decomposition. In most cases, the overall goal needs to be decomposed into smaller sub-goals. This kind of decomposition is usually hierarchical so that sub-goals are further decomposed into smaller sub-sub-goals and so on. The objective of the goal decomposition is to reach a stage where the sub-problems are of an appropriate granularity so that they may be assigned to individual DSs.

Ontology can be used to locate and retrieve the required information. Machine-readable information content within an information resource can be described using an ontology. This provides DSs with a more controlled and systematic way to look for the requested information. By committing to this ontology, DSs are able to 'understand' the information contained within these resources and to exactly locate and retrieve required information. As ontology represents the domain knowledge, it allows DSs to reason through the available information. For example, a DS may need to recognise and remove redundant and/or inconsistent information, select the relevant information and further work with this information as required.

Ontology can be used to present the retrieved information to other DSs or users in a meaningful way. Semantic relationships in ontologies are machine readable. It may provide a structured overview of some specific information. Moreover, the inherited organisation of ontologies adds taxonomical characteristics to the information and conceptual relationships in data can be easily spotted.

Ontology enables cooperative DSs to communicate with each other. Ontology provides the vocabulary needed for communication. The different DSs can reach a shared understanding by committing to the same ontology. Use of ontology enables the DSs to communicate knowledge, make statements and ask queries about a subject domain. Ontologies are very useful in describing communication between different services at the application level, as a way to relate to the domain of discourse rather than to any single implementation. Use of ontology permits coherent communication and easier information sharing between different DSs, enabling them to cooperate and coordinate their actions and function as a whole unit.

### C. *Define Digital Species' Collaborations*

In the first phase, we described how to identify different types of DSs according to the different functions they need to perform within the DES. In this phase, we emphasise the importance of structural organisation of the DSs within a DES. These two phases may sound similar but the difference is that in the first phase we defined DSs' activities within a DES, while in this phase we define the DSs' position within the DES i.e. we are concerned with inter-DS interaction.

The aims of this step are to:

- determine DSs' **behaviour** that will enable DES to function in the most efficient way; and to
- establish the **structure** of communication paths between the different DSs

The DSs need to be organised in such a way that the functional process within DES can easily flow into its completion and the communication between different DSs can be established at any point. Sometimes, simple structure functions the best while in other cases, complexity of the functions may require complex DES structure. In any case, the DES needs to function precisely and efficiently through the synergistic effort of different DSs.

The first dimension to determine collaboration between different DSs is their organisation. We need to make a decision with regards to the behavioural pattern and organisation of the different DSs. Depending on the purpose and goal of the DES, the DSs may be organised in the following ways:

1. self organisation
2. predetermined organisation
3. self and predetermined organisation

In self organisational structure, the different DSs organise themselves to function most optimally. Technologies such as swarm intelligence [5] have been used predominantly in these cases.

In predetermined organisation, DSs do not have the freedom to group with each other but are communicating and collaborating as determined by their designer. Even though the resulting DES has a rigid structure and does not really correspond to the natural ecosystems, this approach may be used in the situation where greater control of DSs' actions is needed.

Some DES may be designed to allow both types of DSs' organisation. A part of DES may contain DSs that are able to self organise while the organisation of DSs of the other part of DES is predetermined. We call this *partial* organisation. In other situations, organisation of different DSs can periodically switch from self to predetermined organisation and we call this *periodical* organisation. In the third option, it may occur that the organisation of different DSs depends on the situation. We call this *situation-dependent* organisation.

The second dimension to determine collaboration between different DSs is their structure. The different DSs can be structured in (1) chaotic, (2) orderly or (3) semi-orderly ways. It makes more sense that a DES would function more efficiently if it is structured in an orderly way. These structures may take any shapes, such that a holonic structure is associated with swarm intelligent behaviour. Here, the autonomous DSs group together to form hierarchies and hierarchies of hierarchies (holons) [6]. Each DS perceives and evaluates continuous changes that occur in the environment and uses them to make optimal decisions.

### D. *Enable, Improve and/or Construct Individual Digital Species*

DSs need to meet the requirements regarding their functions within the DES. For this reason, existing DSs need to be enabled and/or improved and new DSs designed.

Some DSs are available and we do not need to redesign them such as PDAs, laptops and the like. In most cases, it is needed to enable them to be used within the DES. Some special features may need to be added to include these DSs into the dynamic network of the DES. Other DSs will need to be improved through addition of features that will enable them to function more efficiently within the systems.

During the construction phase it is important to:

- identify required DSs' **features**
- design software and hardware that will support these features (**Digital Organs** (DO))
- **assemble** the DO

For all the functions that could not be covered by the existing DSs, new DSs need to be designed. On the basis of these functions, desired DSs' features are identified and the design of corresponding Digital Organs is started. The practical realisation of this step includes the design of the human interface, DSs interface, Communication module, Cooperative, Role, Domain and Environment Knowledge, History Files, and so on. Human interface is needed in DSs that interact with users. The DSs interface enables DSs of the DES to interact and communicate with each other. The Communication module processes the messages. The messages are written in a language that all DSs of the DES are agreed and committed to.

Ontologies can be used in the communication. The Cooperative Knowledge module contains knowledge a DS needs to cooperate and coordinate its actions with the actions of other DSs of the DES. A DS needs to know how to negotiate and coordinate its actions with other DSs. Role Knowledge is knowledge regarding roles of that particular DSs in the DES. The Domain Knowledge component contains information regarding the domain of interest of that particular multi-agent system. Environment Knowledge contains information regarding the environment in which the DSs are functioning. The History Files contain information about the past experiences of the DS. This feature is of special importance in the DSs that are equipped with the self-learning techniques. The above-mentioned different DOs are assembled together to form DS.

The variety of DSs' systems can be achieved in three different ways:

(1) Different DOs that are used to construct different DSs are the same but the *content* of the DOs is different for different DSs. This approach is used when creating DSs of the same type. For example, different PDAs may be designed to have all DOs in common but those DOs may differ from one PDA to another. These differences are expressed on a more detailed level. In biology, we humans are, in a sense, all the same. We all have two arms, two legs, two eyes and so forth. The primary difference is in size, shape, colour, and the like.

(2) The content of the DOs used to construct different agents are the same; however, different DSs are constructed by a different *combination* of the used DOs. This approach is used when designed DSs function as one and are so closely linked together and interdependent that one can not exist without the other. For example, an Ontology needs a DS to be implemented on, and DS needs Ontology to function intelligently. In nature, this is commonly known as symbiosis (we are talking here only about mutualism and not about parasitism and commensalisms).

(3) The third and most common option is that different DSs differ in the *combination* of the DOs used to construct them *and* in the *content* of these DOs. The greatest diversity of DSs is created in this way. In nature, this corresponds with the variety in different species. In the digital world, this can be compared to the difference between phone and PDA. Greater variety can be achieved through fewer similarities in the DOs combination and their content. This corresponds to the larger differences such as between animals and plants. In digital world, this could be compared to the difference between phone and computer.

No matter how complex or simple one organism is, it always has something in common with others. This is its information contained within the genetic material (DNA or RNA). In the digital world, this is analogous to the sequence of atomic software encodings that is common to all Digital Species.

### *E. Protecting Digital Ecosystem by Implementing Security Requirements*

Security plays an important role in the development of DES. When developing DES, the goal is to provide as much security as possible.

Security can be supported by the existence of properties such as [7]:

1. Authentication: proving the identity of DSs
2. Availability: guaranteeing the accessibility and usability of information and resources to authorised DSs
3. Confidentiality: information is accessible only to authorised DSs and inaccessible to others
4. Non repudiation: confirming the involvement of a DS in a certain communication
5. Integrity: assuring that the information remains unmodified from source entity to destination entity

The above-mentioned properties are critical inside the DES as well as outside such as during the interaction with the environment.

The aim of this design phase is to:

- **identify** critical security issues within the DES
- effectively **address** those issues
- clearly set the **boundaries**
- **encode** this knowledge in a form the can be understood by all DSs of the DES
- **equip** the DSs with this knowledge

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

For the purpose of getting more control over the security of the DES, the designer needs to take more details into consideration. Different DSs will play different roles and, with respect to security, some will be more critical than others. As a consequence, some DSs of the system might have been assigned more security responsibilities than others. When considering security of a DES, it may be necessary to identify features such as:

- the role that each individual DS plays in the security of the DES;
- actions that operate within the DES that are most critical with respect to security;
- actions that operate during the interaction of the system with the environment that are most critical with respect to security;

- factors outside the DES that are most critical with respect to security; and
- the part of the DES that is most susceptible to attack from outside.

A designer needs to create a security network from two perspectives:

- the security of the *individual* DS within the DES
- the security of the *whole* DES

It is very important to provide each DS with the basics knowledge that will protect it from malicious actions. Each DS needs to be equipped with this knowledge as the enemy will always study the system and try to attack the weakest ones. Another important factor here is a sense of belongingness. If some DSs are under attack, other members of the DES need to make their best efforts to protect those DSs and the whole DES.

## III. Overall Discussion

The design of a Digital Ecosystem can be followed through the five phases represented in Figures 1 and 2. When the preliminary DES gets implemented, some additional requirements usually arise. In this case, the corresponding design phases need to be repeated until the designer is satisfied with the outcomes.

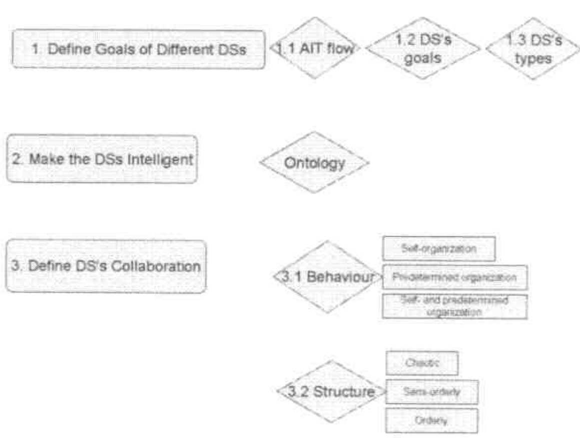


**Fig. 1** Digital ecosystem design methodology (Phase I, II and III)

In the first phase, *Define Functions of Different DSs*, intuitive Action, Information and Transaction (AIT) flow is determined, required DSs functions identified and corresponding DSs' types proposed. The second design phase, *Make the DSs Intelligent*, deals with the issue of DSs' intelligence. Because ontologies are the most expressive knowledge models and enable the system to function the most efficiently, the ontologies were predominantly. This way of thinking is applicable for other approaches. In the third phase, *Define DSs' Collaboration*, DSs' behaviour was defined (which can be self-organisation, predetermined organisation and self- and predetermined organisation) as well as their structure (which can be chaotic, semi-orderly and orderly).

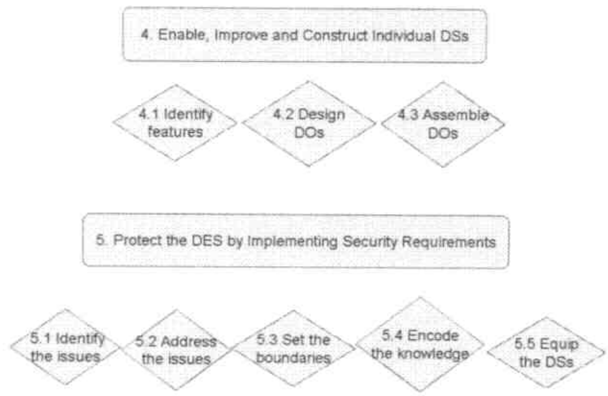


**Fig. 2** Digital ecosystem design methodology (Phase IV and V)

In the fourth phase, *Enable, Improve and Construct Individual DSs*, 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 DSs includes identification of the desired features, designing of DOs to mirror these features and assembling of the DOs into the DSs. A variety of DSs is a result of variety of the DOs and their contents. In the fifth phase, *Protect the DES by Implementing Security Requirements*, the security issues are carefully identified and addressed and the boundaries are clearly set. This knowledge is encoded in the common language and given to the DSs.

## IV. Context of Digital Ecosystems

Digital Ecosystems can be classified under Complex Adaptive Systems (CAS) [8, 9]; DES is a specific form of CAS. Similarity of DES design with Ambient Intelligence needs to be mentioned as well. Ambient Intelligence (AmI) refers to digital environments that are sensitive and responsive to the presence of people [10]. This interaction between human beings and digital information technology can be established through ubiquitous computing devices such as sensors for shape, movement, scent or sound recognition [11]. AmI requires convergence of several computing areas such as ubiquitous computing, intelligent systems, transparent technologies, context awareness and social interaction of objects in the environment [12]. Progress in AmI technologies results in development of better Digital Ecosystems, and vice versa. Those two approaches are sharing the same vision but are taking different points of view. They are highly related and interconnected but show some differences.

To explain the differences, we will take an example. The Intelligent Ambient of a train station can be established through networks of various DSs. Each DS belongs to one of the three DES. One DES controls if the passengers have the correct train tickets (Ticket Control Digital Ecosystem), the second DES may be designed to inform passengers about available train services (Information Digital Ecosystem) and the third DES can be used for security purposes (Security Digital Ecosystem). The whole ambient is intelligent and the three different DESs are making part of it.

The integration process of smaller DESs from different domains may be established to form a Mega Digital Ecosystem that consists of loosely coupled DESs from various domains. Such a Mega Digital Ecosystem would be analogous to the Earth Ecosystem. Lately, technologies are being developed that will enable this integration to take place. Even though the advantages can be numerous, we must not neglect the problems that may arise as a consequence of this integration. This vision of integrated DES raises an important question that requires an answer based on a solid foundation.

## V. Conclusion and Future Work

This paper can provide a more concise understanding of DES and their design, and increase the control over the design process by providing a stepwise insight into the design process. The role of each step in the complete design process was discussed and an analogy with biological ecosystems was provided. We only gave a brief overview of 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] European Commission, 'Technologies for Digital ecosystems – Innovation Ecosystem Initiative'. Retrieved 20th of February 2006 from http://www.digital–ecosystems.org/

[2] European Commission, 'What is an European Digital Ecosystem?'. Retrieved 24th of February 2006 from http://europa.eu.int/comm/enterprise/ict/conferences/doc/p5_de2.pdf.

[3] M. Hadzic, T. Dillon, E. Chang, "Use of Digital Ecosystem and Ontology Technology for Standardization of Medical Records", *Proceedings of the Inaugural IEEE International Digital Ecosystems and Technology Conference*, 2007.

[4] T. Gruber, "Towards principles for the design of ontologies used for knowledge sharing", *International Journal of Human and Computer Studies*, 1995, vol. 43, no. 5-6, pp. 907-928.

[5] M. Ulieru, "Internet-Enabled Soft Computing Holarchies for e-Health Applications", *New Directions in Enhancing the Power of the Internet*, 2003, pp. 131-166.

[6] M. Hadzic, M. Ulieru, E. Chang, "Ontology-Based Holonic Diagnostic System for the Research and Control of New and Unknown Diseases", *Proceedings of the IASTED International Conference on Biomedical Engineering*, 2005.

[7] H. Mouratidis, P. Giorgini, G.A. Manson, "Modelling secure multi-agent systems", *Proceedings of the Second International Joint Conference on Autonomous Agents and Multiagent Systems* (AAMAS 2003), 2003, pp. 859-866.

[8] M. M. Waldrop, Complexity: The Emerging Science at the Edge of Order and Chaos, Simon & Schuster, 1992.

[9] M. Rupert, S. Hassas, A. Rattrout, "The Web and Complex Adaptive Systems", Proceedings of the 20th conference on Advanced Information Networking and Applications (AINA 2006), 2006, vol. 2, pp. 200-204.

[10] A. Gaggioli, "Optimal Experience in Ambient Intelligence", *Ambient Intelligence*, IOS Press, 2005.

[11] M. Alcaniz, B. Rey, "New technologies for Ambient Intelligence", *Ambient Intelligence*, IOS Press, 2005.

[12] N. Shadbolt, "Ambient Intelligence", *IEEE Intelligent Systems*, 2003, vol. 18, no. 4, pp. 2-3.