



Learning from the Cell Life-cycle: a Self-Adaptive Paradigm

Antinisca Di Marco, Gallo Francesco, Inverardi Paola, Rodolfo Ippoliti

► **To cite this version:**

Antinisca Di Marco, Gallo Francesco, Inverardi Paola, Rodolfo Ippoliti. Learning from the Cell Life-cycle: a Self-Adaptive Paradigm. Proc. of the 4th European Conference on Software Architecture (ECSA2010), 2010, Copenhagen, Denmark. 2010. <inria-00512415>

HAL Id: inria-00512415

<https://hal.inria.fr/inria-00512415>

Submitted on 18 Nov 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Learning from the Cell Life-Cycle: A Self-adaptive Paradigm

Antinisca Di Marco¹, Francesco Gallo¹, Paola Inverardi¹, and Rodolfo Ippoliti²

¹ Dipartimento di Informatica - University of L'Aquila

² Dipartimento di Biologia di Base ed Applicata - University of L'Aquila
{antinisca.dimarco,francesco.gallo,
paola.inverardi,rodolfo.ippoliti}@univaq.it

Abstract. In the software domain, self-adaptive systems are able to modify their behavior at run-time to respond to changes in the environment they run, to changes of the users' requirements or to changes occurring in the system it-self. In life science, biological cells are power entities able to adapt to the (unpredictable) situations they incur in, in a complete decentralized fashion. Learning adaptation mechanism from the cell life-cycle, we propose in this paper a new architectural paradigm for self-adaptive software systems.

1 Motivations and Background

Biological systems inspire systems design in many directions [4,6,5]. Among the biological systems, cells represent a model for their ability to undergo adaptation changes as a response to environmental stimuli. Each cell is able to check its status, to duplicate (mitosis) and to kill itself (apoptosis) in case of need[1].

Stem cells are primitive cells with the singular ability to generate all other types of cells and to give rise to progenitor cells in adult tissues able to substitute ageing cells [1]. They, regardless of their origin, have three general properties: (*i*) they are able to divide and renew themselves even after a long period of inactivity, (*ii*) they are not specialized cells, that is they do not make work of a specific organ or tissue and (*iii*) they give rise to several specialized cell types.

Once a stem cell is activated, depending on the type of internal or external received signal, it may choose to mutually execute:

- **Symmetric Stem Cell Division (SSCD)** process, in which the stem cell clones itself producing a new stem cell identical to it.
- **Asymmetric Stem Cell Division (ASCD)** process, a first step of specialization, that originates as outcome of the stem cell and a new cell called Transit Amplifying Cell (TAC) or progenitor cell. TAC is the real origin of the (tissue) cell proliferation and it represents an intermediate cell that has lost the status of stem and that can evolve into a tissue cell.
- **Apoptosis (A)** process, it happens when the stem cell is induced to differentiate into cells unable to enter in the cellular process and it is then sent to death.

TAC, in turn, can mutually start the **Terminal Differentiation (TD)** process (that represents the final specification step), in which the cell passes to a specific status (Differentiated Cell (DC)), depending on the type of tissue or organ in which it is located; or the **Apoptosis (A)** process. TAC undergoes the apoptosis if it occurs in an errors, or the generation from the stem cell fails in meeting the required behavior specification. Finally the DC may face either to mitosis or apoptosis. **Mitosis (M)** is a process that substitutes the DC with two new daughter cells of the same type (DC').

The basic properties of the cell life-cycle that have inspired our paradigm are (i) the reflective properties of stem cells, i.e. the ability to self-preservation in spite of changes induced by the environment and (ii) the ability to capture stimuli from the environment and generate the associated behaviors. This two properties can be summarized as: *a universal machine that can interpret the signals from the environment and generate the code that implements the desired functionality*. The paradigm also embeds the cell apoptosis process [3] to improve the efficiency of the whole system by means of the programmed death of all cells that are no more useful as they are, and that cannot evolve any more.

2 Bio-inspired Paradigm for Self-adaptive Systems

The definition of the paradigm we propose is given by means of a mapping between the paradigm entities and the cellular process concepts inspiring them. Figure 1 shows such a mapping reporting at the left-side the cellular process concepts whereas at the right-side the corresponding paradigm entities.

The key concept of the paradigm is the stem module that is inspired to the stem cell. In our paradigm, we refer to multipotent stem cell as stem cell that is able to generate different types of differentiated cells as required by the organ making up its environment. The organ establishes the boundary of the stem cell specialization capabilities, limiting its generation power to the differentiated cells of the tissues composing it. The organ concept is mapped to the functionalities provided by the system. The system can evolve by adaptation in different ways always guaranteeing the core services (organ functionalities) as required by the users. In other words, the organ functionalities and the user requirements represent the invariant for the biologic and software systems, respectively. In general, the *invariant* (Inv) represents a collection of properties that meets system and which must also be satisfied after successive and repeated adjustments induced in the system. The stem cell represents a huge container of information, which is capable if properly stimulated, to interpret signals from the environment and generate the appropriate behaviour. In particular, the stem cell itself contains all the means to understand the problem shaped as changes, and generate the "solution". It implements the basic functionalities that allow the cell to survive and evolve over time, preserving them in a sort of kernel. In our metaphor, the stem cell can be compared to the universal machine able to execute all possible behaviors throughout interpretation of the corresponding code. To improve the efficiency of the paradigm, the stem module is devised as an engine able

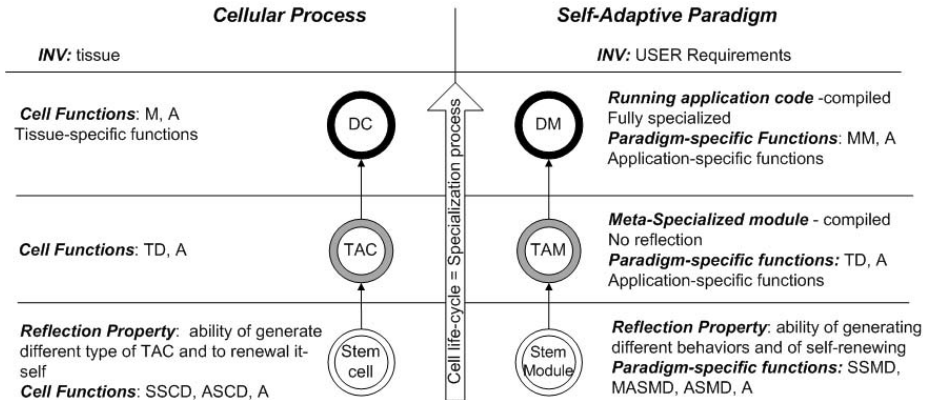


Fig. 1. Mapping from the Cell Life-cycle to the Self-adaptive Paradigm

to treat behaviors as data, interprets them to generate compiled code (i.e., the Differentiated Module) implementing the new behavior.

Both the stem cell and the stem module are characterized by the *reflection property*, i.e. the ability to self-renew, to capture changes in the system and in its environment, and to generate code to implement different, possible new, behaviors in case of need.

Generate code suitable for the implementation of specific functionality is equivalent to passing from a condition of generalization to one of specialization. This transition occurs through a series of steps or specialized functions that we describe below. The first step is one of the following *division*:

- **ASMD** (Asymmetric Stem Module Division), this process results in a first step of specialization: our universal machine in response to a given external event, generates a specialized meta module software, called TAM (Transit Amplifying Module), which represents a software module compiled with the ability to further specialize into the final code implementation of the functionalities associated to the event or to self eliminate, mimicking the cellular concept of **Apoptosis** (A). Moreover, by exploiting the reflection ability of the universal machine, it duplicates by creating an identical copy of itself.
- **MASMD** (Meta Asymmetric Stem Module Division), this process is very similar to the one seen above, but there is no duplication of the universal machine. This is because in most cases this is unnecessary and expensive computationally. This process is not present in the cellular life-cycle. We introduce it to improve the efficiency of the paradigm.
- **SSMD** (Symmetric Stem Module Division), this process results in a duplication of the universal machine only. This process is very rare also in biology, as computationally expensive.

The next step of the specialization process, Terminal Differentiation (TD), is the final generation of specialized code, which in our paradigm is denoted by **DM**

(Differentiated Module). This module represents the compiled code that will run on the underlying hardware and exhibits the highest degree of efficiency. Again, to improve the efficiency of the paradigm, TD step generates a DM maintaining a copy of the TAM.

The concept of mitosis in our framework is very close to the one in the cellular process. The difference is that in our paradigm, the production of multiple copies does not imply the destruction of the original one. It will be denoted by MM (Meta Mitosis).

In our paradigm, the running software system is composed by a stem module, a set of TAM (one for each type) and a set of differentiated modules (possible more instances coming from the same TAM type) implementing the application specific functions. The stem module has the ability to face the unforeseeable changes since it contains all the logic to generate new behaviors in case some unforeseeable changes occur the system must adapt. Whereas the more specialized modules (i.e., the set of DM) provide the services required by users and implement the adaptation logic to foreseen and foreseeable changes. In our mind, the canonical feedback control loop (such as, the MAPE-K loop) [2] is implemented in the DMs either in a centralized or in distributed way. Finally, TAMs implement an intermediate step of specialization between stem module and DM, introduced to reduce the complexity of the specialization step, as it happens in the cellular process. To improve the paradigm efficiency, we maintain a copy of each type of generated TAM from which restarts the TD step in case of necessity without to involve the stem module again.

Acknowledgement. This work is partially supported by the EU-funded CONNECT project (FP7-231167) and by the Italian PRIN d-ASAP project.

References

1. Alberts, Johnson, Lewis, Raff, Roberts, Walter (eds.): *Molecular Biology of the Cell*, 4th edn. Garland Publishing, Inc., New York (2004)
2. Brun, Y., Serugendo, G.D.M., Gacek, C., Giese, H., Kienle, H.M., Litoiu, M., Müller, H.A., Pezzè, M., Shaw, M.: Engineering self-adaptive systems through feedback loops. In: *Software Engineering for Self-Adaptive Systems*, pp. 48–70 (2009)
3. Degtarev, Y.J.A.: Expansion and evolution of cell death programmes. *Nat. Rev. Mol. Cell Biol.* 9(5), 378–390 (2008)
4. Horn, P.: Autonomic computing: Ibm perspective on the state of information technology. In: Presented at AGENDA 2001, Scottsdale, AR, IBM T.J. Watson Labs, NY (2001)
5. Shen, C.-C., Li, K., Jaikao, C., Sridhara, V.: Ant-based distributed constrained steiner tree algorithm for jointly conserving energy and bounding delay in ad hoc multicast routing. *ACM Trans. Auton. Adapt. Syst.* 3(1), 1–27 (2008)
6. Snyder, P.L., Greenstadt, R., Valetto, G.: Myconet: A fungi-inspired model for superpeer-based peer-to-peer overlay topologies. In: SASO, pp. 40–50 (2009)