

The Tao way to anti-fragile software architectures: the case of mobile applications

Vincenzo Grassi
University of Roma Tor Vergata
Roma, Italy
vincenzo.grassi@uniroma2.it

Raffaella Mirandola
Politecnico di Milano
Milano, Italy
raffaella.mirandola@polimi.it

Abstract—In an ever-changing world software architects need to embrace uncertainty and changes to exploit them in building better systems. We suggest that looking at the Taoist philosophy and its incarnation in the martial arts domain can help to promote this mental attitude and to get a better understanding of the potentialities of existing architectural approaches, using as a testbed the field of mobile applications design.

Index Terms—Software Architecture, Resilience, Anti-Fragility, Mobile Applications

I. INTRODUCTION

[...] *Change is certain. Peace is followed by disturbances; departure of evil men by their return. Such recurrences should not constitute occasions for sadness but realities for awareness, so that one may be happy in the interim.* [...] (I Ching - (12th Century BC) - Chinese Book of Changes)

Architecting software applications today means taking into account an ever-changing world where unexpected events, both at design time and runtime, play a key role and cannot be neglected. In this context, the goal of a software architect is to design applications that, despite the continuous occurrence of these changes, are able to carry out their tasks maintaining at a high level the user experience. In recent years, this goal has been increasingly encompassed within the conceptual frameworks of *resilience* and *anti-fragility* that refer, respectively, to systems able to absorb changes and although temporarily changing, recover from them, or able to absorb changes and get even better afterward [14], [16], [18], [19]. Designing resilient or anti-fragile software system has thus become a primary concern for software architects, and their mental attitude should be permeated by the awareness of this reality, which must go along with a clear understanding of a full range of possible options to be adopted in practice to this end.

To achieve this state of mind, and the ability of translating it into effective actions, our position is that it can be fruitful to get inspired by principles and practices developed within fields seemingly very distant from the field of software development. In this perspective, we do not intend to propose new design methodologies or development tools, but rather show how ideas coming from distant fields can help to get a better awareness and a more coherent and unifying view of already

known methodologies and tools, as well as insights about their potential and their integrated use.

In this attempt to build a bridge between distant worlds, on the software side we will focus on the mobile applications domain, where variability and uncertainty represent, even more than in other domains, the rule rather than the exception. Within this domain, we will focus on the definition stage of software architecture: indeed, given the “foundational” spirit of our discussion, this is the stage where fundamental decisions are taken that will drive the next more fine grained development steps.

On the other side of the bridge we are trying to establish, we will look at the world of martial arts rooted in the Eastern culture. Thus, in the remaining part of this paper, we will first briefly highlight, in Section 2, some basic principles underpinning these martial arts. Then, in Section 3, we will draw a parallel with architectural strategies that have emerged in the recent past as well suited to tackle the challenges of mobile applications design, and will show how looking at these strategies through the lenses of Eastern martial arts may help to achieve a unifying view of their scope, potentialities and mutual relationships. Finally, in the last section we will draw a possible conclusion resulting from our discussion.

II. EASTERN MARTIAL ARTS ESSENTIALS

Looking at the principles underlying many Eastern martial arts, we can recognize the presence of a common core, which has one of its most known expressions in the Taoist philosophy, expressed in some fundamental texts like the *Tao Te Ching* and *I Ching - Chinese Book of Changes*. A central aspect of this philosophy is the dualistic view of the two principles (Yin and Yang) that underlie our world and drive its evolution, where each of them temporarily prevails and then fades into the other one, according to a never ending seamless cycle, where the harmony of the world is strictly related with the dynamic equilibrium between these two principles.

This vision has provided guidance in the Eastern world for many different facets of human life, through what we could call its domain-specific instantiations, spanning several fields that include, for example, medical and ethical issues. In particular, we can recognize the presence of this vision in the martial arts domain: indeed, as any Eastern martial arts (e.g. tai-chi-chuan, aikido) practitioner knows, his/her

capability of not being overcome by, and possibly succeed against an opponent, relies on a dual ability: (i) the (defensive) ability of not being controlled by the opponent, and (ii) the (offensive) ability of controlling the opponent. The former ability can be looked at as the manifestation of the Yin principle, and is achieved by avoiding any rigidity (the opponent should not be able to find our “center”, finding instead “emptiness” in front of her/him). The latter can be looked at as the manifestation of the Yang principle, and is achieved by exploiting any “resource” that the opponent could offer, for example by exploiting any leverage s/he could offer us, or, even better, by turning against the opponent her/his own energy, thus in a sense multiplying our own energy.



Fig. 1: Tao principle.

In line with the Tao principles, these two abilities should not be considered as separate, but should instead be seen as a continuum, where from time to time each of them prevails but then transforms itself into the other one, depending on circumstances (as illustrated in the well known Tao symbol (Fig. 1). Looking at this through the

lenses of resilience and anti-fragility, we can consider an Eastern martial art master as a perfect example of resilient and anti-fragile system: s/he is able to absorb attacks and to use those attacks to improve the effectiveness of hers/his own.

III. A YIN-YANG VISION OF ARCHITECTURAL STRATEGIES FOR MOBILE APPLICATIONS

“[...] the degree of [system] resilience depends [...] on how large is the set E of “changes” that the system can successfully cope with, how large is the set G of goals that can be maintained despite of changes, how large is the set of initial states when a change occurs from which the system is still able to achieve its goals G , and how short is the time T it takes to “win” its fight [...]” (from [10], page 129)

The quote reported above, where we have purposely evidenced in boldface the last part of the sentence, suggests the relationship between the goal of designing resilient (or anti-fragile) mobile applications and the world of martial arts. The degree of resilience/anti-fragility of a mobile application can indeed be seen as the result of its “fight” against the environment where it is executed, which acts as the application opponent: changes occurring in the environment may be seen as actions put in place by the environment itself to make the application deviate from its intended behavior, thus preventing it from achieving its own goals.¹ To make the application

¹By environment we refer to the part of the external world, including both physical and software entities, with which the application interacts and in which the effects of the application will be observed and evaluated [8], where the distinction between the environment and the application is made based on the extent of control.

able to exit as a winner from this fight, the software architect should thus adopt architectural strategies that make it instead able to maintain the intended behavior (resilience), or, even better, improve it (anti-fragility).

In this respect, two *architectural strategies* have emerged and consolidated in the recent past as specifically suitable to tackle the challenges a mobile application must face:

- i *self-adaptation*: its goal is to make the application able to autonomously cope with dynamically and frequently changing operating environments [4],
- ii *cyber foraging*: its goal is to make the application able to overcome the limits of the portable/pervasive devices where it (or at least a part of its components) is usually deployed on, by opportunistically exploiting possibly nearby external resources [2].

Strategies (i) and (ii) can be described in terms of a collection of *architectural tactics* [3], where each tactic covers some specific facet of the overall strategy. In general, despite its narrowly focused nature, an architectural tactic is not an “atomic” entity, but can be further refined in sub-tactics, giving rise to a hierarchical structure. For the cyber foraging strategy, a thorough hierarchical classification of its constituent tactics has been provided in [11]. Figure 2 depicts a partial view of this classification. For self-adaptation, some general surveys (e.g. [6], [9], [15]) have been already presented in literature, enlightening several different facets of this strategy. We complement these surveys by outlining in Appendix IV a characterization of the self-adaptation strategy, specifically aimed at describing it in terms of a hierarchical classification of possible tactics. Figure 3 depicts a synthetic view of this classification.

If we confront these two architectural strategies with the martial arts strategies outlined in Section II, we see that a parallel can be drawn between them, in light of the Tao principles. To this end, in the two paragraphs below, we argue that self-adaptation and cyber foraging can be looked at as, respectively, the manifestation of the Yin and Yang principles (and of their incarnation in the martial arts domain) within the domain of mobile software architectures, both concurring to the achievement of the common goal of making a mobile application able to win its fight against the environment where it operates. Indeed, analogously to Yin and Yang, each of them can be thought of separately, but yet includes the other one as part of itself, and, from a martial arts perspective, they can be looked at as mainly defensive and mainly offensive strategies, respectively.

a) *Distinct strategies*: Self-adaptation and cyber foraging represent two clearly distinct ways of facing the challenges of the environment where the mobile application operates, and each of them can be applied without recurring to the other one. Self-adaptation expresses a mainly Yin (defensive) approach upon the occurrence of a change that can make the application deviate from its intended behavior: the idea is that the application should not “resist” to it, but rather modify itself to adapt to it, thus making the attack (i.e., change) ineffective and the application still able to carry out its task.

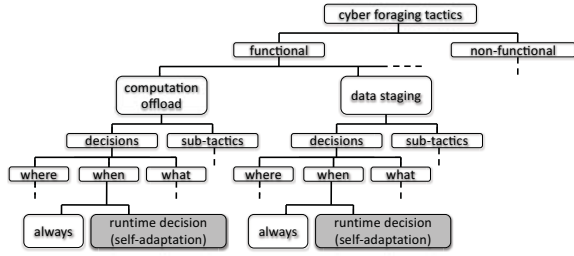


Fig. 2: Cyber foraging tactics (partial view, adapted from [11]).

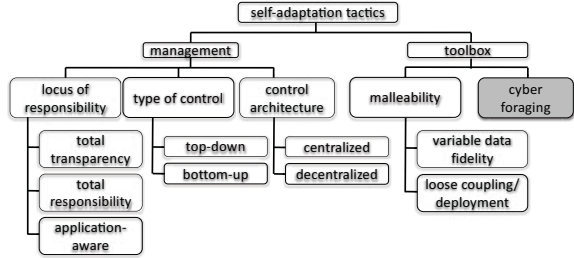


Fig. 3: Self-adaptation tactics (see Appendix IV).

On the other hand, cyber foraging expresses a mainly Yang (offensive) approach: its idea is that the application should exploit the resources that the environment could, maybe transiently, offer to it, to carry out its task more effectively and efficiently (e.g., to a greater extent, more quickly, or with a minor cost, etc.).

b) *Partially overlapping strategies*: Self-adaptation and cyber foraging are not completely separate strategies, as each of them can contain the other one as part of itself. With regard to cyber foraging, as discussed in [11], one of the architectural decisions that concur to the definition of this strategy is *when* it (some of its tactics) should be applied. If the answer is *always*, we have a system that blindly applies cyber foraging, irrespective of specific environment or internal conditions. If the answer is *runtime decision*, this evidently opens the door to considerations about the opportunity of using cyber foraging in a given environment, thus entering the realm of self-adaptation.

On the other hand, as we highlight in Appendix, cyber foraging can be considered as one of the possible tools within the self-adaptation strategy toolbox. We can of course think of a self-adaptive system that does not make use of this tool, but excluding it a priori would clearly be a limit to the self-adaptation possibilities.

Figures 2 and 3 summarize the separation and intertwine-ment of the two strategies, where the grey boxes in each figure highlights how the tree of tactics that describes each of the two strategies contains the other one as one of its elements.

IV. DISCUSSION: LEARNING FROM TAO

Whoever assists to an exhibition of masters of some Eastern martial arts, remains impressed by the harmony, effectiveness and apparent simplicity of their actions against some opponent. Whatever we can think in general of the Taoist philosophy

and of its applications to several human life domains, it is undoubtable that the application of its principles in the martial arts domain represents a success story, culminated in the development of a set of winning fighting practices based on the seamless transition between emptiness and fullness, Yin and Yang actions. In our discussion we have drawn a parallel between this domain and the domain of mobile applications with its self-adaptation and cyber foraging architectural strategies. Both these strategies can concur to the design of resilient/anti-fragile architectures, able to absorb (and, in the best case, thrive on) changes coming from the environment or from the system itself. However, if we look through the lenses of Tao (and its martial arts manifestation) at literature concerning these strategies, or at real world examples of their adoption, we can often perceive a sense of imbalance and lack of completeness, as the focus seems concentrated on only one of them, while the contribution of the other one is only implicitly, if not at all, considered.

We can find evidences of this in the literature presenting surveys or collections of solutions/tactics/strategies focused on cyber foraging (e.g., [2], [11]), or self-adaptation (e.g., [6], [7], [9], [15]). In the real world, popular applications like existing virtual assistants for smartphones rely on cloud resources (cyber foraging) but lack self-adaptation capabilities allowing them to work in different conditions, for instance when distant cloud servers are not reachable. On the other hand, as outlined in [6], self-adaptation in mobile applications is often applied at hardware level or at the platform level without exploiting external resources. This focus on only one of the two strategies risks to obscure the fact that to fully exploit their potential equal attention must be given to both of them: a self-adaptive application that does not include cyber foraging in the set of possible tools to be used limits the range of possible responses to changes; but including cyber foraging within this set requires giving full attention to all the related issues, like for instance external resources discovery and provisioning [2], [11]. Analogously, an application using cyber foraging out of any self-adaptation framework risks to suffer from excessive rigidity, which can lead to inefficient use of resources, or to inability in facing unforeseen situations; but to consider also self-adaptation, we must pay attention to the whole set of related issues, like for instance how the self-adaptation logic itself is architected [17].

To conclude, if we conventionally adopt the Tao as a symbol of perfection and harmony, then the lesson we can draw from our discussion is that if a software architect wants to attain that harmony and perfection in the design of a mobile application, fully attaining the goal of making it resilient/anti-fragile, s/he should treasure the achievements coming in particular from the world of martial arts: focusing only on a purely Yin (self-adaptation) or purely Yang (cyber foraging) strategy will hardly lead to the design of a fully resilient or, even better, anti-fragile mobile application (not to talk of not adopting none of them). To win the fight against its environment, the design of a mobile application should be based on a suitable and seamless combination of both strategies, giving equal attention to both

of them, and to the issues they bring with them: they must proceed hand-in-hand, ready to give way to or prevail on the other one, depending on circumstances, as the never ending Yin-Yang cycle suggests to us.

REFERENCES

- [1] B. R. Badrinath, A. Fox, L. Kleinrock, G. J. Popek, P. L. Reiher, and M. Satyanarayanan. A conceptual framework for network and client adaptation. *Mob. Networks Appl.*, 5(4):221–231, 2000.
- [2] R. Balan, J. Flinn, M. Satyanarayanan, S. Sinnamohideen, and H.-I. Yang. The case for cyber foraging. In *Proceedings of the 10th Workshop on ACM SIGOPS European Workshop*, EW 10, page 8792, New York, NY, USA, 2002. Association for Computing Machinery.
- [3] L. Bass, P. Clements, and R. Kazman. *Software Architecture in Practice*. Addison-Wesley Professional, third edition, 2012.
- [4] B. H. Cheng et al. Software engineering for self-adaptive systems: A research roadmap. In B. H. Cheng, R. de Lemos, H. Giese, P. Inverardi, and J. Magee, editors, *Software Engineering for Self-Adaptive Systems I*, volume 5525 of *Lecture Notes in Computer Science*, pages 1–26. Springer, 2009.
- [5] P. T. Eugster, P. A. Felber, R. Guerraoui, and A.-M. Kermarrec. The many faces of publish/subscribe. *ACM Comput. Surv.*, 35(2):114–131, June 2003.
- [6] E. Grua, I. Malavolta, and P. Lago. Self-adaptation in mobile apps: a systematic literature study. pages 51–62, 05 2019.
- [7] M. C. Huebscher and J. A. McCann. A survey of autonomic computing - degrees, models, and applications. *ACM Comput. Surv.*, 40(3), 2008.
- [8] M. Jackson. The meaning of requirements. *Ann. Softw. Eng.*, 3:521, Jan. 1997.
- [9] C. Krupitzer, F. M. Roth, S. VanSyckel, G. Schiele, and C. Becker. A survey on engineering approaches for self-adaptive systems. *Pervasive Mob. Comput.*, 17(PB):184206, Feb. 2015.
- [10] A. D. Leeuw and H. Volberda. On the concept of flexibility: A dual control perspective. *Omega*, 24(2):121 – 139, 1996.
- [11] G. Lewis and P. Lago. Architectural tactics for cyber-foraging: Results of a systematic literature review. *Journal of Systems and Software*, 107:158–186, 2015.
- [12] B. D. Noble, M. Satyanarayanan, D. Narayanan, J. E. Tilton, J. Flinn, and K. R. Walker. Agile application-aware adaptation for mobility. *SIGOPS Oper. Syst. Rev.*, 31(5):276287, Oct. 1997.
- [13] V. Pejović. Towards approximate mobile computing. *GetMobile: Mobile Comp. and Comm.*, 22(4):912, May 2019.
- [14] J. Ramezani and L. Camarinha-Matos. Approaches for resilience and antifragility in collaborative business ecosystems. *Technological Forecasting and Social Change*, 151, 02 2020.
- [15] M. Salehie and L. Tahvildari. Self-adaptive software: Landscape and research challenges. *ACM Trans. Auton. Adapt. Syst.*, 4(2), May 2009.
- [16] N. N. Taleb. *Antifragile*. Random House, New York, 2012.
- [17] D. Weyns and et al. On patterns for decentralized control in self-adaptive systems. In R. de Lemos, H. Giese, H. A. Müller, and M. Shaw, editors, *Software Engineering for Self-Adaptive Systems II*, volume 7475 of *Lecture Notes in Computer Science*, pages 76–107. Springer, 2010.
- [18] D. D. Woods. Four concepts for resilience and the implications for the future of resilience engineering. *Reliability Engineering & System Safety*, 141:5 – 9, 2015. Special Issue on Resilience Engineering.
- [19] D. D. Woods. Essentials of resilience, revisited. In *Handbook on Resilience of Socio-Technical Systems*, chapter 4, pages 52–65. Edward Elgar Publishing, 2019.

APPENDIX: ARCHITECTURAL TACTICS FOR SELF-ADAPTIVE MOBILE APPLICATIONS

In our classification², we start from the distinction between the *managing* and *managed* subsystems that together constitute a self-adaptive system [17]. From this, we derive a first

²For space reasons, we just outline in this appendix a very coarse grain and concise classification of architectural tactics to implement the self-adaptation strategy for mobile applications. It is mainly oriented towards the goal of this paper; as such, it lacks the thoroughness and details that can be found, for example, in the classification presented in [11] for the cyber foraging strategy.

separation into two categories of possible tactics that concur to the realization of the self-adaptation strategy:

- 1) tactics concerning the *adaptation management* logic, that is the organization of the managing subsystem;
- 2) tactics concerning the *adaptation toolbox*, that is by what means the managed subsystem can be modified to adapt it to an occurred change.

Then, in the next two subsections we synthetically outline possible tactics that can be classified under one of these two categories. We refer to the cited references for more details.

A. Adaptation management tactics

a) Locus of responsibility: Under this category we include different approaches that can be adopted to define the degree of collaboration of the self-adaptation logic with the application layer: according to [12], they include: (i) *total transparency*, (ii) *total responsibility*, (iii) *application-aware*;

b) Type of control: This category concerns different organizations of the self-adaptation logic, according to a [4]: (i) *top-down* tactic, where the adaptation logic is implemented through explicit feedback loop(s), or (ii) *bottom-up* tactic, where adaptation emerges as the result of elementary interactions among a large number of components;

c) Control architecture: This category concerns different approaches in the definition of the managing subsystem architecture, including [17]: (i) *centralized control*; (ii) *decentralized control*.

B. Adaptation toolbox tactics

a) Malleability: Under this category we include the approaches that can be adopted to reduce the “rigidity” of the managed subsystem, thus increasing the freedom degrees that can be exploited by the managing subsystem: (i) *variable data fidelity* tactics concern the possibility of varying the “fidelity” of the computed [13] or transmitted [1], [12] data with respect to a reference instance of the same data; (ii) *loosely coupled architecture* tactics concern the adoption of architectural elements and decisions that reduce the tightness of the bindings among different components (e.g., by using loosely coupled connectors like publish-subscribe) or among components and the underlying platform (to support the possible migration of components) [5].

b) Cyber foraging: We have already introduced cyber foraging in Section III as an architectural strategy by itself, that can be also adopted irrespective of any self-adaptation tactic; here we just remark that it can also be seen as a possible tool for the implementation of self-adaptation, where the application of the different tactics classified in [11] ((i) *computation offload* tactic and (ii) *data staging* tactics) can be driven at runtime by some self-adaptation logic.