

# When Interaction Choices Trigger Business Evolutions

Guillaume Godet-Bar, Sophie Dupuy-Chessa, and Dominique Rieu

Laboratoire LIG, University of Grenoble  
681, rue de la Passerelle – BP 72  
38402 St Martin d’Hères, France  
firstname.surname@imag.fr

**Abstract.** In the context of development methods, early collaborations between specialists (SE, HCI, business, usability experts . . . ) allows having a broader view of the development possibilities, notably in terms of user-system interaction. Consequently, in-depth transformations of the processes and concepts can be considered with minimum financial or temporal impact. We discuss in this paper the opportunities for business process evolutions emerging from the application of our collaborative method, based on the choices made for designing the interaction.

**Keywords:** Information Systems, Human-Computer Interaction, Collaborative Design, Evolution

## 1 Introduction

The evolution of computer technologies, in terms of communication (wireless networking) and interaction device (visualization headsets, tactile gloves) deeply alter the classical, implicit perception of Human-Computer Interaction (HCI). The user can now evolve in environments blending real and virtual entities. We shall use the concept of “Augmented Reality system” (AR system) to designate any interactive system that superimposes virtual data onto the real world. A classical example of AR system can be found in [1].

These new opportunities sometimes lead HCI specialists to design the future application from a totally different, usability-oriented, point of view, and therefore explore aspects of the future system that are sometimes unforeseen from a purely “functional core side”. In order to address these issues, we have proposed an adaptation of the Symphony development method [2]. While this method is originally Software Engineering oriented and focuses on the design of the functional core and business aspects of the system, we have integrated HCI practices and models for designing classical as well as complex interfaces. This method is also a medium for showing the impact of HCI choices on business evolution, from the bottom-most levels of refinement of the specifications, up to the business definition level.

We summarize in Section 2 the essential notions on which the Symphony method is based. Then, Section 3 describes the principles for envisaging business evolution. We conclude this paper by giving some perspectives on future works.

## 2 The Symphony method

We use the Symphony method as a medium for merging SE and HCI activities. It is a user-oriented, business component-based development process originally proposed by the UMANIS company. It has already been extended by [2] and [3], mainly in order to improve reusability of components, and lately to integrate the design of complex interfaces such as Augmented Reality systems.

Symphony is organized into three design branches, similarly to 2TUP, into a Y-lifecycle: functional aspects are treated in the left branch, technical concerns in the right branch, and a central branch merges both developments. As this development process has already been addressed previously, we shall focus on the aspects pertinent for business evolution, which occurs in the left branch of the development cycle:

- The functional branch features two essential phases: Specification of requirements and Analysis of requirements,
- All phases aim at refining models and scenarios outlined in the previous phase,
- SE and HCI-oriented activities are realized in parallel, by design actors specialized either in usability, business expertise, Software Engineering or Human-Computer Interaction, setting out from common scenarios elaborated during the Inception phase. Therefore it is likely that two different approaches to the final system will be developed, both valuable in terms of the aspects they are expected to concentrate on (for instance functionalities for SE specialists, usability for HCI experts),
- Collaboration points are envisaged at specific stages of development, in order to ensure consistency of adopted design options, synchronize points of view, establish conceptual links between models or take collegial decisions on design choices.

The functional branch ultimately produces a set of dynamic and static models –mostly UML models– for describing the application. We have described in [3] a specific model type for describing business concepts (Business Objects) and interactional concepts (Interactional Objects).

The following section details the different stages of business evolution, starting from the end of the Analysis phase and using Business Objects and Interactional Objects, up to the Specification of requirements.

## 3 Different stages of business evolution

We explore in this section how, starting from a coordination activity for modifying the Business Objects model and the Interactional Objects model, an evolution of the business space may be triggered at different levels of refinement of the Information System.

– **Business Objects model evolution**

Even though Interactional Objects are generally more complex than most Business Objects, due to the intrinsic complexity of interactions and interfaces, we have set up a systematic analysis of the structural imbalance between Interactional Objects and Business Objects.

In particular, we evaluate whether the Interactional Objects model aspects of the system which in fact correspond to business concerns. Our analysis aims at deciding whether the Information System would benefit from transferring and adapting this data to the business space.

For instance, Augmented Reality systems often feature topological data, usually used for positioning the user or artifacts in tridimensional virtual environments superimposed on the physical world. A possible transfer of competences would consist in augmenting the Business Objects managing location concepts with architectural plans. Subsequently, Interactional Objects would remain responsible for managing the visual representation of the architectural plans.

– **Activity evolution**

In the context of the Symphony method, an essential aspect of the specification of the business space at the Activity level is the identification of computerized tasks realized by internal actors, using Use Cases. The latter are organized into logical packages corresponding to Business Process Components (i.e., units representing uninterrupted exchanges between external actors and the Business Process).

While focus at the Business Objects model level was on transfers of competences, we concentrate at the Activity level on the way the resulting new data (in our example, topological data) may be collected, organized and used.

From a development process point of view, Use Cases for carrying these new activities consequently appear.

– **Business Process evolution**

At the Business Process level, the Symphony method focuses on describing the interaction between external actors and the system.

One of the essential activities at this level of description is the identification of Business Process Components, that is, uninterrupted exchanges between actors and the Business Process. In order to achieve this, these interactions are represented using UML sequence diagrams.

Driving business evolution to this level means capitalizing on the new Use Cases introduced at the Activity level. In particular, use of the new data by different (or new) Business Processes Components and the intervention of new external actors, within the Information System, may be envisaged. This may enable automatization of manual tasks, facilitate certain processes. . .

Concerning our example, topological data could be used for estimates, providing access to location-related data *in situ* (using head-mounted display and positioning technologies), providing location-dependent services etc.

– **Business definition evolution**

Beyond reorganizing Business Processes, business evolution may be taken to the point that new services are handled by the Information System, thus

changing the definition of business. New Business Processes may be added, that will need their own entire iteration of the development process. Consequently, new interaction choices will be made for these new processes and new actors that may also eventually affect the business space. In our example, integrating topological data into the business space could be used for organizing virtual tours for potential clients.

## 4 Conclusion and perspectives

We have detailed in this paper how describing the interactional aspects of a system may affect the design of the business space, when an effort for rationalizing the balance of competences between the two conceptual domains is made. The design of complex interactive systems puts even more strains on this problematic, because of their intrinsic requirements for intuitiveness, continuity of interaction, the numerous inputs they need to integrate. . .

In the context of the Symphony development method, we have proposed a process for capitalizing on the evolutions induced by interaction choices, from concepts reorganization to the redefinition of the business.

In the current instance of the Symphony method, we tackle the specifications evolution, which is closely linked to that of data reorganization, by relying on the concept of Iteration Plan, from the Rational Unified Process [4]. Thus, we do not need to identify cluttering "iteration pathways" for each activity. Instead, programmed evolutions are progressively added to the coming Iteration Plan. Final decision on the application of features of the iteration can thus be regularly discussed with stakeholders.

As is often the case with new constructions in methodology, we need to confirm our propositions with further experiments. This may be delicate, given that this method applies to complex interactive systems integrated into larger Information Systems. Consequently, we shall focus future works on the adequate description of Iteration Plans centered on business evolution, both in terms of models and process.

## References

1. Ishii, H., Ullmer, B.: Tangible bits: towards seamless interfaces between people, bits and atoms. In: CHI '97: Proceedings of the SIGCHI conference on Human factors in computing systems, New York, NY, USA, ACM (1997) 234–241
2. Hassine, I., Rieu, D., Bounaas, F., Seghrouchni, O.: Symphony: a conceptual model based on business components. In: SMC'02, IEEE International Conference on Systems, Man, and Cybernetics. Volume 2. (2002)
3. Godet-Bar, G., Rieu, D., Dupuy-Chessa, S., Juras, D.: Interactional objects: Hci concerns in the analysis phase of the symphony method. In: 9th International Conference on Enterprise Information Systems ICEIS'2007, Funchal, Madeira (2007) 37–44
4. Jacobson, I., Booch, G., Rumbaugh, J.: The Unified Software Development Process. Addison-Wesley (1999)