



Chapter VI

The Logic of Deferring the Design Process

Nandish V. Patel
Brunel University, UK

The aim of this chapter is to formulate theoretical foundations for deferring the IS design process to inform adaptable usage of deferred systems. A research agenda for Deferred System's Design is set and a new system component, the System-System Environment Interface, is defined as the public interface through which a deferred system can maintain close coupling with the system environment. A Problem-solving and planning technique is suggested, the Deferred-Specified IT/IS Matrix, to aid software developers to position a proposed system development in the system environment. A systems analysis technique is suggested to identify deferment points in a given system design. The chapter identifies seven properties of deferred systems that software developers can use to inform adaptable system designs.

INTRODUCTION

The logic of deferring the information system (IS) design process to so-called “users” is embedded in the systems environment (SE). The thesis of this chapter is

that to build true adaptable IS it is necessary to defer the IS design process to people who make use of the IS in context or the SE. The SE of IS is thus posed as the critical problem that challenges developers of adaptable software in business applications. Adaptable software is needed by business organizations to enable IS to respond to changes in the SE, such as changes to company mission, objectives or business processes.

An understanding of the SE is required to inform what kinds of IS design would be suitable for adaptation. Mechanisms for IS adaptation can be understood better if the source for the need for adaptation itself, the SE, can be modeled to identify issues that need to be reflected in adaptable IS. The SE needs to be explored to understand how it impacts adaptable IS. There are variables and factors in the SE that have not yet been formally identified; these variables and factors have a significant impact on conceptions of IS and adaptable IS design.

Understanding of the SE can also inform the creation of suitable IS development approaches. Current approaches and methodologies for IS development are predominantly system-centric. System-centric means that conceptions of IS and approaches to its development focus on the computer system. They are concerned with determining system requirements and functional needs that can lead to algorithmic design and implementation. This system-centric view imposes limits on thinking of adaptable software. Lessons for building adaptable software need to apply to how adaptable software is developed too, which can be learnt from investigating the SE. In particular, a study of the SE can determine what dimensions of an adaptable IS and the SE can be deferred to people who use IS.

Two issues are critical in deferring the IS design process to people who make use of the IS. The first concerns what kind of conceptions of IS are possible. In the 50 years of computer applications in business, little advance has been made in seeking alternative conceptions of IS. The underlying conception is still rooted in the human-machine interaction paradigm. This paradigm regards the basis of IS as human-machine interaction.

The practical implementation of conceptions of IS rooted in the human-machine interaction paradigm is limited by existing technology and cost considerations. The available technology poses limits on what is possible to achieve in terms of adaptation, though object-oriented systems seem the most promising. A focus on the SE can inform conceptions of adaptable IS based on factors that affect the IS in its SE.

The second issue concerns what approaches are suitable for developing adaptable IS. IS developers have largely adopted the systems development life cycle (SDLC), originally formulated for the human-machine interaction paradigm, for developing IS development methodologies. IS design and implementation are thus based on the SDLC. The human-machine interaction paradigm originally

25 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/logic-deferring-design-process/4217

Related Content

Fault Tolerant Control of an AUV using Periodic Output Feedback with Multi Model Approach

Sneha Joshi and D. B. Talange (2016). *International Journal of System Dynamics Applications* (pp. 41-62).

www.irma-international.org/article/fault-tolerant-control-of-an-auv-using-periodic-output-feedback-with-multi-model-approach/150479

Generating Indicators for Diagnosis of Fault Levels by Integrating Information from Two or More Sensors

Xiaomin Zhao, Ming J. Zuo and Ramin Moghaddass (2013). *Diagnostics and Prognostics of Engineering Systems: Methods and Techniques* (pp. 74-97).

www.irma-international.org/chapter/generating-indicators-diagnosis-fault-levels/69673

Optimizing User Quality of Experience through Overlay Routing, Bandwidth Management and Dynamic Trans-Coding

Maarten Wijnants, Wim Lamotte, Bart De Vleeschauwer, Filip De Turck, Bart Dhoedt, Piet Demeester, Peter Lambert, Dieter Van de Walle, Jan De Cock, Stijn Notebaert and Rik Van de Walle (2012). *Technological Innovations in Adaptive and Dependable Systems: Advancing Models and Concepts* (pp. 160-180).

www.irma-international.org/chapter/optimizing-user-quality-experience-through/63580

BDD-Based Combinatorial Keyword Query Processing under a Taxonomy Model

Shin-ichi Minato and Nicolas Spyrtas (2011). *Interdisciplinary Advances in Adaptive and Intelligent Assistant Systems: Concepts, Techniques, Applications, and Use* (pp. 26-39).

www.irma-international.org/chapter/bdd-based-combinatorial-keyword-query/46283

Methods for Developing Flexible Strategic Information Systems: Is the Answer Already Out There?

Alan Eardley, Hanifa Shah and June Lazander-Reed (2003). *Adaptive Evolutionary Information Systems* (pp. 306-328).

www.irma-international.org/chapter/methods-developing-flexible-strategic-information/4225