Chapter 20
Testing Complex and Dynamic Business Processes

Krishnendu Kunti
Infosys Technologies Limited, India

Bijoy Majumdar
Infosys Technologies Limited, India

Terance Bernard Dias
Infosys Technologies Limited, India

ABSTRACT

In this chapter we deal with testing of business processes implemented using computer systems. We have discussed challenges associated with business process testing, identified aspects of business processes that need to be tested and capabilities that the testing tool(s) or environment should have in order to perform such testing. The chapter also discusses a commonly used software testing methodology in light business process testing for provisioning of structured mechanism for business process testing. One of the aspects of managing complex and dynamic business processes is making sure that the process delivers what is required of it at all times. Dynamics of the business may require frequent changes in the business process and whenever such changes takes place there is a need to test the process thoroughly to ascertain that the process is still working according to the requirements laid down for it. This becomes even more important if the business process is implemented using computer systems since over a period of time the computer software becomes more prone to error as it is updated frequently to accommodate the business changes hence the requirement for testing complex business processes.

INTRODUCTION

Traditionally, business processes (BPs) and roles were defined within organizations. These processes became mature and robust over a period of time. As business grew, computer systems were developed in order to cater to business needs. But since, in erstwhile times the advantages of modular and distributed systems were not very well known, most of the systems developed were monolithic in nature.

These monolithic applications worked in silos; performed high level activities required by individual business units and used custom data formats. This did not matter much since these systems required very little communication with other systems and whenever they did communicate, they used point to point, tightly coupled batch mode file transfers. As a result of the monolithic and closed nature of these systems, business functions could not be reused across systems. Such a scenario led to replication of business function across multiple systems. The resulting system architecture offered predictability but at the same time took away flexibility and hence any scope of easy evolution of the process concerned. Over the years as organizations grew in size, mergers and acquisitions took place. Also, more and more activities of the organization started getting outsourced. This led to steep increase in the number interactions and interacting partners. The competition also increased over time and there was a need to modify BPs quickly to stay ahead of the competition. This led to the need of agile or dynamic business processes by which new processes and products can be delivered to the customer quicker and with minimal effort. Supporting dynamic BPs required loosely coupled architecture which could not possibly be achieved using traditional tight coupling between processes where a whole range of activities were bundled into a single system. The solution lay in functional decomposition of monolithic systems into smaller modules, which could be exposed as services and arranged in the most desired format according to need of the BP. Such architecture enabled wider access of business functions (often defined using standard based interfaces) which in turn facilitated re-usability, increased flexibility and eased communication.

Modern BP principles along with enabling architecture paradigms like Service Oriented Architectures (SOA (Service Oriented Architecture, 2008) and technologies like Process modeling/enactment engine, Rules engine, composite applications, etc. allow creation of dynamic BP. While dynamic BPs are great for business they are difficult to manage on the technical front. One of the many challenges associated with managing complex and dynamic BP implementations is its testing. This chapter will discuss the challenges in testing of BP implementations as well as an approach for BP development and testing using the V-Model for software development. It also discusses the aspects of BP implementation that needs to be tested, scope of BP testing and capabilities that a BP implementation testing system should possess. Finally, it talks about the future trend in this area and the conclusion.

Since this chapter deals with testing of BP implementation using computer systems rather than testing the BP itself, from here onwards we will refer to BP implementations simply as BPs for better readability.

BACKGROUND

BPs differs from traditional application in that they may span across multiple stakeholders, enterprise IT layers, standards, security zones, may be partially automated, may partially require human intervention or may comprise of both legacy and new applications. Also, they may involve some complex rules and business policies. The process should also be flexible enough architecturally so that it can be modified quickly to meet the constantly changing business needs. Although this has become possible today with adoption of SOA, the challenges involved in testing such BPs have multiplied manifolds. Existing approaches of development and testing can be used for BPs, however testing requirements in BPs differ a lot from other applications. This is an important area and there is very little literature available on it. In this chapter we hope to discuss the various aspects of BP testing to some level of detail so that it may be useful to any team trying to develop and test dynamic BPs.

Related Content

Solving Large Distribution Problems in Supply Chain Networks by a Column Generation Approach
Rodolfo G. Dondoand Carlos A. Mendez (2014). *International Journal of Operations Research and Information Systems (pp. 50-80).*
www.irma-international.org/article/solving-large-distribution-problems-in-supply-chain-networks-by-a-column-generation-approach/117779

Negation Handling in Machine Learning-Based Sentiment Classification for Colloquial Arabic
Omar Alharbi (2020). *International Journal of Operations Research and Information Systems (pp. 33-45).*
www.irma-international.org/article/negation-handling-in-machine-learning-based-sentiment-classification-for-colloquial-arabic/264181

Project Portfolio Risk Management: Managing Risk in Case of Investment Portfolio
Nicolae Postvaruand Bogdan Leonte (2018). *Global Business Expansion: Concepts, Methodologies, Tools, and Applications (pp. 1549-1570).*
www.irma-international.org/chapter/project-portfolio-risk-management/202284

A Sensitivity Analysis of Critical Genetic Algorithm Parameters: Highway Alignment Optimization Case Study
Eungcheol Kim, Manoj K. Jhaand Min-Wook Kang (2015). *International Journal of Operations Research and Information Systems (pp. 30-48).*
www.irma-international.org/article/a-sensitivity-analysis-of-critical-genetic-algorithm-parameters/124760

Semi-Automatic Derivation and Application of Personal Privacy Policies
George Yee (2007). *E-Business Innovation and Process Management (pp. 348-365).*
www.irma-international.org/chapter/semi-automatic-derivation-application-personal/8688