

# Flexible B2B Processes: the Answer is in the Nodes

Giacomo Piccinelli, Anthony Finkelstein  
University College London  
London (UK)

E-mail: {g.piccinelli,a.finkelstein}@cs.ucl.ac.uk

Tommaso Costa  
Hewlett-Packard Labs  
Bristol (UK)

E-mail: tomcos@hplb.hpl.hp.com

The time and costs involved in connecting the IT systems of two companies impact the actual formation of business relationships. A flexible infrastructure for process management is instrumental for rapid and cost-effective B2B integration. One dimension of flexibility that system integrators identify as critical is node-level interaction. In this paper, we discuss the findings of the Nile project on B2B process integration. In particular, we present the methodology defined in Nile for the semi-automatic reconciliation of node-level incompatibilities.

**Keywords:** B2B integration, B2B processes, RosettaNet.

## Introduction

Business processes capture the operational knowledge of a company [2, 8, 9]. As information artefacts, process specifications embed the logic that drives a wide and differentiated set of business resources. The design of a process specification reflects the tension between conflicting requirements [1, 3, 11]. On the one side, simplicity determines execution efficiency. On the other side, flexibility determines business agility. Such tension is particularly noticeable for business-to-business (B2B) processes [4, 6, 10].

The main challenge for the specification of a B2B process is that not all the resources are known at design time. As an example, let us consider the procurement processes for a PC manufacturer P&C Computers. The production processes of P&C require stability in order to achieve economies of scale. Hence, the inward-looking part of the procurement processes cannot be changed depending on the specific supplier for specific parts. At the same time, P&C may depend on different suppliers to cope with fluctuating volumes of demand and price oscillations. The outward-looking part of the P&C's procurement processes needs to adapt to the processes of the various suppliers. Flexibility is an intuitive solution to the problem, but a more precise answer is required in order to devise effective business solutions.

In this paper, we present and discuss the findings of the Nile project on B2B integration. In essence, the answer seems to lay in node-level interaction. In section 2, we present an overview of Nile. In section 3, we discuss the issues of node-level interaction. In section 4, we describe the methodology devised in Nile to address node-level conflicts. In section 5, we outline a technical framework substantiating the Nile methodology. Closing remarks and directions for development conclude the paper.

## The Nile project

The Nile project was a joint research activity between HP, RosettaNet [5], and UCL (University College London). The area of research was the automation of B2B processes, with focus on standardisation issues [7]. RosettaNet provided the working context, hence the choice of the name. The two main areas of activity for the project were the investigation of user requirements and the definition of a solution framework.

The user community was mainly composed of system integrators with expertise on RosettaNet-based solutions. The level of engagement ranged from one-off completion of questionnaires or telephone interviews, to constant short-cycle feedback. Considering the different levels engagement and interaction, we estimate the number of users involved at around 15 representing in excess of 10 RosettaNet integration projects. The mix of integration projects included one-off one-to-one RosettaNet channels, as well as development frameworks for one-to-many channels. The main objective of the requirement gathering process was to the identification of the most critical factors in the implementation of a B2B process. The outcome of the investigation was that the single most important issue in B2B processes was node-level interaction. In essence, flexibility is required in terms of adapting the information exchange in individual process nodes. At least in terms of system integration, adapting the overall process flow seems less of an issue. Using the P&C example, the issue is the definition of Send-Order or Request-Notification nodes that adapt to different suppliers. Different suppliers may require the nodes used in different order, but the problem seems easier to solve.

The solution framework proposed by Nile focuses on a methodology to capture formally different node-level interaction requirements. Interaction requirements from different companies can then be reconciled using a semi-automatic procedure. A set of technology components substantiating the solution framework is currently under refinement.

## **Node-level interaction**

The basic elements of a workflow process [2] are roles, activities, data, and execution logic. Roles model business entities responsible for specific activities. Business entities can be software systems or human beings. Production Supervisor and Account Manager are examples of roles. Progress Update and Prepare Invoice are examples of activities. Data represent the raw material as well as the by-product of an activity. A node is an atomic unit within a process. In the simplest case a node includes one role, one activity, and a set of data. For example, a node can specify that Account Manager has to Prepare Invoice based on details for the order Order Number. The execution logic of a process determines the order in which nodes are executed.

In the case of a B2B process, some of the nodes involve entities and data outside the administrative domain of one company. We refer to such nodes as interaction nodes. Nile's findings were that the context-specific usage of data in interaction nodes is among the biggest difficulties to overcome in B2B process integration. In essence, the problem is that the same data can be used differently with different business partners. Moreover, the same data can be used differently with the same business partner in different contexts. The granularity of such context can vary from process groups down to individual nodes in individual processes. As an example, a payment process can involve different stages. The actual usage of a Credit Note form in the Deposit Payment node can be different than in the Final Payment node. Differences can be as subtle as having different order identification numbers for open and for completed orders. What if one of the parties expects a single number to be used?

## **The Nile methodology**

Standardisation initiatives such as RosettaNet are instrumental to the solution of node-level interaction issues. Still, standards face a reality made of diversified business infrastructures. For example, RosettaNet itself allows optional fields. Different choices on optional fields can make two RosettaNet-compliant solutions unable to interact. Indications from the system integrators also hint at the fact that actual and intended content of a field may be different in

practice. One approach is to define detailed global standards, and enforce strict compliance. Leaving standardisation bodies to pursue such route, the pragmatic approach taken in Nile is to provide formal and automated support to current practices.

The methodology proposed in Nile targets the reuse of business and technical knowledge. The idea is to describe in a formal way business contexts and the related usage of data. Relations between contexts are also formally expressed. The end result is that when a new context arises (e.g. because of a new supplier), all the previous knowledge on node-level interaction can be immediately and automatically reused. The assumption in Nile is that data are mainly represented as forms composed of a structured set of fields. The knowledge base proposed by Nile is based on association in the format:

```
(1) ({Processes}, {Partners}, {Forms})
    ::
    Field (cardinality, content, back-end mapping)

(2) ({Processes}, {Partners}, {Forms})
    ::
    {Field (cardinality, content),
     ... ..
     Field (cardinality, content)}
    →
    Field (cardinality, content, back-end mapping)
```

In both cases, the first part of the association defines the context and the second part the usage information for the field. The context includes a set of processes and/or process categories, a set of partners and/or partner categories, and a set of forms and/or form categories. For associations of type 1, the second part includes a field identifier and related usage definition. The usage definition includes information on cardinality, content, and the actual back-end source of data. Association of type 2 allow what can be interpreted as a further refinement of the context in terms of the usage of other files in the forms. An example of associations for P&C could be:

```
({any}, {CAT:Big_Customer, Company_A}, {Invoice})
::
{E-mail (1..1, "e-mail address"),
 Telephone (0, "telephone number")}
→
Reply (1..1, "e-mail address", SAP.xyz(...))
```

The association captures information on any process of P&C in which an invoice is exchanged with either the specific company *Company\_A* or a customer of category *Big\_Customer*. If e-mail is used and telephone is not used, then the *Reply* field is used and contains the email address retrieved with the request *xyz* to the SAP system of P&C. Whenever a new process is set up (e.g. between P&C and a big customer *Customer\_B*) and an *Invoice* is used, the requirement about the *Reply* field is automatically highlighted. If different or more specific requirements emerge for *Customer\_B*, the information directly augments the knowledge based.

## Outline of the reconciliation framework

The possibility to retrieve automatically the information related to the data for an interaction node is a basic capability provided by Nile. Such capability can be used within one company

without dependencies on its business partners. Still, interaction requirements of different companies have to be compared and reconciled. If two companies have modelled their interaction requirements with Nile, the discovery of differences can be done automatically. Specific components of the Nile framework can be used for generating external views on knowledge base of a company, and for comparing such views from different companies. The resolution of the differences requires human intervention, but the outcome can be fed into the knowledge base for future reuse.

## Conclusions

The investigation carried out within Nile points to interaction nodes as the main source of rigidity in B2B process integration. Standardisation helps simplifying the problem, but a complete working solution almost inevitably requires a degree of customisation. The approach of Nile is based on formal representation of business and technical knowledge in order to foster reuse. In particular, formal representation enables automation, which is a crucial issue in the development of complex business solutions.

## Acknowledgements

Special recognition goes to the various members of RosettaNet and Hewlett-Packard for their contributions in terms of both project information and feedback.

## References

- [1] Farbey, B. and Finkelstein, A. "*Software Acquisition: a business strategy analysis*" Proc. Requirements Engineering (RE01), 2001.
- [2] Fisher L. (Ed.) "*Workflow Handbook*" Workflow Management Coalition and Future Strategy Inc., 2002.
- [3] Haug H., Olsen E.W., Vallet G., and Becart O. "*Software Management Approaches: Project Management, Estimation, and Life Cycle Support: Software Best Practice*" Springer-Verlag, 2001.
- [4] Kaplan S. and Sawhney M. "*B2B E-Commerce hubs: towards a taxonomy of business models*" Net Market Makers, 1999.
- [5] Kak R. and Sotero D. "*Implementing RosettaNet E-Business Standards for Greater Supply Chain Collaboration and Efficiency*" RosettaNet online, 2002.
- [6] Kalakota R. and Whinston A. "*Frontiers of electronic commerce*" Addison Wesley, 1996.
- [7] Piccinelli G., Finkelstein A., and Stammers E. "*Automated Engineering of e-Business Processes: the RosettaNet Case Study*" Proc. Int. Conference on Systemic, Cybernetics, and Informatics (SCI02), Orlando, Florida, 2002.
- [8] Reilly B. and Block J. "*Next-generation E-Commerce processes and systems*" Electronic Commerce Strategies Report, GartnerGroup, 1997.
- [9] Schwartz D.G. "*Cooperating Heterogeneous Systems*" Kluwer Academic Publisher, 1995.
- [10] Timmers P. "*Electronic commerce - Strategies and models for business-to-business trading*" Addison Wesley, 1999.
- [11] Valacich J., Hoffer J., and George J. "*Essentials of System Analysis and Design*" Prentice Hall, 2000.

**Giacomo Piccinelli** is a Research Fellow in the Computer Science Department of University College London. His main areas of research include software architectures, process management, and service-oriented computing. His current interests are in the area of architectures for grid- and service-oriented software systems.

**Anthony Finkelstein** is Professor of Software Systems Engineering and Head of Department at University College London, Department of Computer Science. His research is in the area of software systems engineering and in particular in requirements engineering. His current interests are in the area of managing distributed information in software development.

**Tommaso Costa** is an MSc student at Hewlett-Packard Labs. His main areas of activity include business-to-business information systems and electronic marketplaces.