

Methodology and Technology Services



Web www.ies.aust.
com

[Home](#) | [Courses](#) | [Certification](#) | [Projects](#) | [Papers](#) | [EA Blog](#) | [Online Store](#) | [Contact Us](#) |

THE ENTERPRISE NEWSLETTER

Issue No: 31

Business Process Modeling Notation (BPMN)

[Printable PDF Version](#)

By [Clive Finkelstein](#)

CONTENTS

- [Introduction](#)
- [Upcoming Events](#)
- [Feature: Business Process Modeling Notation \(BPMN\)](#)
- [Author](#)

- [Previous Issues of TEN](#)
- [Glossary of Terms](#)
- [New Subscribers to TEN](#)

Editorial

Jan 17, 2006: This issue of TEN continues from TEN27, which covered the concepts of Business Process Management (BPM) languages for SOA, such as Business Process Execution Language (BPEL), Business Process Modeling Language (BPML) and Business Process Specification Schema (BPSS) - and the various BPM products and vendors which I covered in TEN29. These issues were based on extracts from Chapter 14 of my

[Home](#)
[Courses](#)
[Certification](#)
[Projects](#)
[Papers](#)
[TEN Archive](#)
[EA Blog](#)
[Contact Us](#)
[Search](#)
[Links](#)
[Online Store](#)

forthcoming book: *"Enterprise Architecture for Integration: Rapid Delivery Methods and Technologies"*, Artech House, Norwood, MA (March 2006).

This current issue of TEN is a further extract from Chapter 14 of the book. It covers Business Process Modeling Notation (BPMN), which has emerged as a standard approach for business process diagramming to bridge from process models defined by modeling tools for automatic generation of executable XML-based code in BPEL, BPML or BPSS.

Clive Finkelstein
Publisher, The Enterprise Newsletter (TEN)

Upcoming Events

A series of 3-day workshops, titled: *"Quick Wins Enterprise Architecture Workshop"* will be presented in Australia in April - May by Clive Finkelstein. These are scheduled in Sydney (April 19 - 21) and Melbourne (May 2 - 4).

- The Marcus Evans PDF brochure can be downloaded as follows:
- Sydney (April 19 - 21): <http://www.marcusevans.com/events/CFEventinfo.asp?EventID=10517> and
- Melbourne (May 2 - 4): <http://www.marcusevans.com/events/CFEventinfo.asp?EventID=10518>.

Clive Finkelstein will also be presenting his highly successful 2-day seminar first presented in 2005 in Australia and NZ also in Canada on March 27 - 28 in Victoria BC and March 30 - 31 in Edmonton AB, titled: *"Rapid Enterprise Architecture Delivery"*.

- Contact Intervista Institute via email at <Teresa@Intervista-Institute.com>.or at <http://www.intervista-institute.com/educational-programs/planning-2006.html>

He will be presenting a 3-day workshop in London on April 3 - 5, 2006 titled: *"Rapid Delivery of Enterprise Architecture for Business Integration"* (see <http://www.irmuk.co.uk/59/>) and a 2-day seminar on April 6-7 titled : *"Rapid Delivery Technologies for Enterprise Integration"* (see <http://www.irmuk.co.uk/56/>).

[Back to Contents](#)

Business Process Modeling Notation (BPMN)

By [Clive Finkelstein](#)

Business Process Modeling Notation (BPMN)

[In previous issues of TEN](#), we have covered BPEL, WSCI, BPML and BPSS as the four major BPM languages used today for the automatic generation of executable XML code from workflow diagrams. This issue discusses BPMN as an evolving standard for specification of process logic for all of these BPM languages. We will also briefly review some of the software products that support these languages. [Read full Content of TEN#31](#).

Business Process Modeling Notation (BPMN) is emerging as a way to specify business process models and diagrams for any BPM language. As the following extract from the BPMN Specifications^[1] states:

“Business people are very comfortable with visualizing business processes in a flow-chart format. There are thousands of business analysts studying the way companies work and defining business processes with simple flow charts. This creates a technical gap between the format of the initial design of business processes and the format of the languages, such as BPEL4WS, that will execute these business processes. This gap needs to be bridged with a formal mechanism that maps the appropriate visualization of the business processes (a notation) to the appropriate execution format (a BPM execution language) for these business processes.

Inter-operation of business processes at the human level, rather than the software engine level, can be solved with standardization of the Business Process Modeling Notation (BPMN). BPMN provides a Business Process Diagram (BPD), which is a Diagram designed for use by the people who design and manage business processes. BPMN also provides a formal mapping to an execution language of BPM Systems (BPEL4WS). Thus, BPMN would provide a standard visualization mechanism for business processes defined in an execution optimized business process language.”

BPMN uses Private Processes, Abstract Processes and Collaboration Processes.

Private Processes are those that are internal to an organization. These are processes that have been generally called workflow or BPM processes. A single private process will map to a single BPEL XML document. *Abstract Processes* represent the interactions between a private process and another process or participant. Only those activities that are used to communicate outside the private business process are included in the abstract process. All other "internal" activities of the private process are not shown in the abstract process. Thus, the abstract process shows to the outside world the sequence of messages that are required to interact with that business process. A single abstract process may be mapped to a single BPEL abstract process.

Collaboration Processes depict the interactions between two or more business entities. These are defined as a sequence of activities that represent the message exchange patterns between the entities involved. A single collaboration process may be mapped to various collaboration languages, such as ebXML BPSS, RosettaNet, or WSCI.

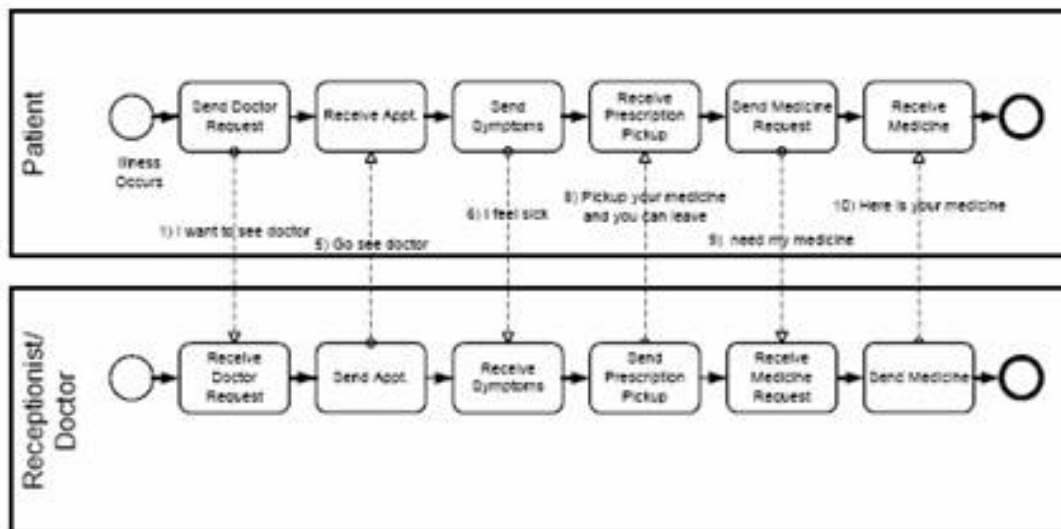


Figure 1: A Process from Different Points of View using BPMN

An example from the BPMN Specifications is shown in Figure 1. this illustrates the message interactions between a patient and a doctor in establishing an appointment to see the doctor, who diagnoses the illness and prescribes medicine, with interactions to collect the medicine.

BPMN specifies standard diagrammatic notations or icons that can be used in process models to define a business process in a BPMN Business Process Diagram (BPD). An example illustrating part of the process diagram is shown in Figure 2 for a Travel Agent example expressed as a

BPMN BPD.

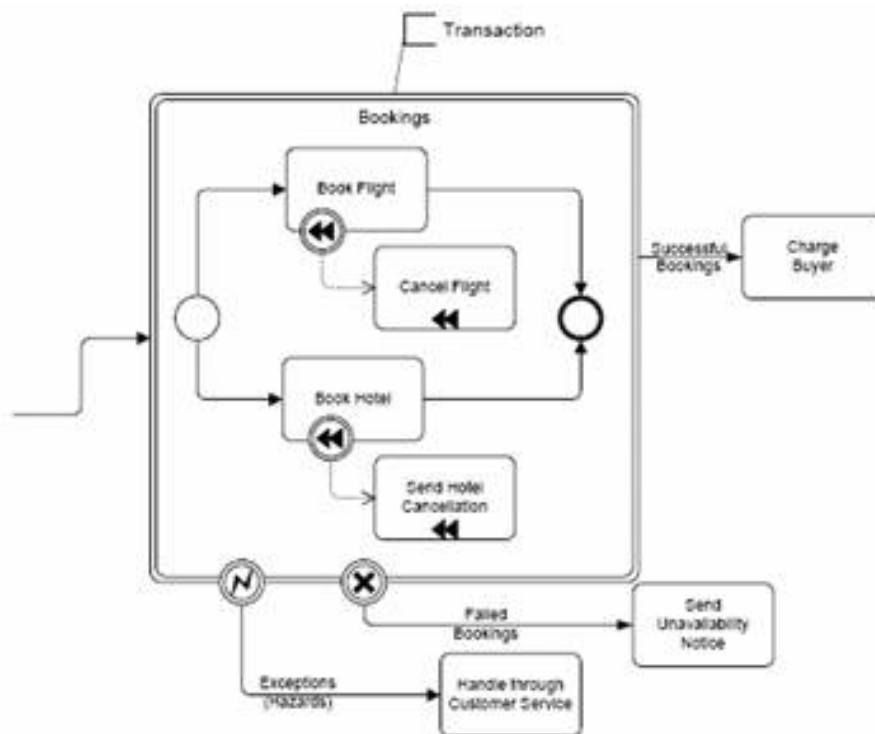


Figure 2: Part of a Travel Agent Example, shown as a BPMN BPD

The BPMN Specifications illustrate, using many examples, generation of XML executable BPEL code from typical process models in BPMN. It describes, in detail, the mapping of BPMN diagrams to generated BPEL XML code.

A complete BPMN example of an e-Mail Voting process model is documented as Figure 92 (on p127) of the BPMN Specifications, with generated BPEL code from this BPMN process included as Appendix A of that document.

The Role of Modeling Tools

Computer Aided Software Engineering (CASE) tools first appeared in the early 1980s. (This term is not used today; instead they are referred to as Modeling Tools.) These tools were based on the premise that computer technologies could assist systems analysts and Data Base Administrators (DBAs) in their analysis and design efforts ... in much the same way that Computer Aided Design (CAD) tools provided design assistance to architects and product designers. Computer Aided Manufacturing (CAM) tools further offer help by translating product designs to robotic commands, so that products can be manufactured using automated factory robots. These CAD tools are widely used today in building architecture and the automated robot-controlled manufacture of automobiles.

It was felt that modeling tools could be used to translate data base designs – developed by DBAs – into the Data Definition Language (DDL) required to implement and install those data base designs using Relational Data Base Management System (RDBMS) products. Many data modeling notations existed in the 1980s; most are still widely used today. Some of these include Entity/Relationship (E/R) modeling,^[ii] IDEF1X,^[iii] Information Engineering^[iv] and modeling notations associated with specific RDBMS products, such as Oracle Designer. These different data modeling approaches were based on the relational theory and normalization principles defined by Edgar Codd^[v] and documented by Chris Date.^[vii]

The rigor of the logical data modeling and physical data base design methods used in the 1980s enabled this automatic DDL generation objective to be realized exceptionally well. Today it is standard practice for DDL code to be generated automatically by modeling tools from logical data models and physical data base designs. In fact, it is now unusual to see anyone manually coding SQL Create Table and Create Index statements.

In the 1980s, it was also felt that the application designs developed by systems analysts could be translated automatically into executable program code. This task was much more difficult. These application designs were expressed diagrammatically as logic using Software Engineering.^[viii] In the 1990s, UML^[ix] diagrams supplanted Software Engineering and became widely used to document application designs and logic. UML diagrams were also perceived as a way to generate executable program code automatically.

There were some successes in automatic generation of executable program code from diagrams: one of the most notable was the Information Engineering Facility (IEF) in the late 1980s and 1990s.^[x] Because of its initial complexity, this approach to automatic code generation was largely considered to be a failure by many. As a result, code generation – hyped by CASE tools in the 1980s and 1990s – eventually lead to “CASE” becoming a derogatory term. It was largely this market perception that saw the more successful of these tools launch themselves as “Modeling Tools”. They wanted to distance themselves from the bad name that the other CASE tools had gained.

The software industry is now moving strongly into the 21st Century. Today, modeling tools are seeing a renaissance in market interest. There is a resurgence in the innovation and application of technologies to automate software design, development and automatic code generation. The next issue of TEN will discuss some of the major modeling tool vendors and the product capabilities that are available today.

Modeling Tool Products and Directions

In the next issue of TEN, we will discuss the product capabilities of some modeling tools that support BPMN, and the strategies used by their vendors. I will not attempt to cover all products in the market. I will only address specific products that, by their innovation, will enable you to see the technology trends that are developing.

Endnotes

[i] The BPMN Specifications are on the XML Cover pages at <http://xml.coverpages.org/BPMNv10Draft.pdf>. A White Paper overview of BPMN is available from the Popkin web site at <http://www.popkin.com/>. Popkin is now part of Telelogic at <http://www.telelogic.com/>.)

[ii] E-R modeling is based largely on the principles established by Peter Chen, in his paper: *"The Entity Relationship Model: Toward a unified view of data"*, ACM Trans. on Database Systems, 9-36 (1976).

[iii] Thomas Bruce, *"Designing Quality Databases with IDEF1X Information Models"*, Dorset House, New York: NY (1991).

[iv] The business-driven Information Engineering (IE) methods were first developed from 1976 – 1980 by Clive Finkelstein and popularized throughout the 1980s by James Martin. These business-driven IE methods have since evolved into the methods that are used today for the rapid delivery of Enterprise Architecture as discussed in the book by Clive Finkelstein: *"Enterprise Architecture for Integration: Rapid Delivery Methods and Technologies"*, Artech House, Norwood, MA (March 2006).

[v] Edgar Codd, *"A Relational Model for Large Shared Data Banks"*, CACM, 13 (6) 377-87 (1970).

[vi] Edgar Codd, *"Extending the Database Relational Model to Capture more Meaning"*, ACM Trans. on Database Systems, 4(4), 397-434 (1979).

[vii] Chris Date, *"Introduction to Data Base"*, Volumes 1 and 2, Addison-Wesley, Reading: MA (1982) and later editions.

[viii] See the books referenced in Chapter 5 by Michael Jackson, Ken Orr, Ed Yourdon and Tom de Marco.

[ix] See the books referenced in Chapter 5 by Grady Booch, Ivar Jacobson and James Rumbaugh.

[x] IEF (Information Engineering Facility) was developed by Texas Instruments and was quite successful. But before code could be generated, a great deal of definition was first needed to capture the detailed logic that was not expressed in diagrams. Once this initial definition had been completed, program code could be automatically generated from diagrams. And when the logic in these diagrams needed to be changed, the new program code could also be automatically regenerated. This product was purchased by Sterling and was renamed Coolgen. It was later purchased by CA and renamed AllFusion.

AUTHOR

Clive Finkelstein is the "Father" of Information Engineering (IE), developed by him from 1976. He is an International Consultant and Instructor, and Managing Director of Information Engineering Services Pty Ltd (IES) in Australia.

Clive Finkelstein's books, online interviews, courses and details are available at <http://www.ies.aust.com/cbfindex.htm>.

For More Information, Contact:

Clive Finkelstein, Managing Director
Information Engineering Services Pty Ltd
PO Box 246, Hillarys, Perth WA 6923 Australia

Details: <http://www.ies.aust.com/cbfindex.htm>

Web Site: <http://www.ies.aust.com/>

Phone: +61-8-9275-3459

Fax: +61-8-6210-1579

Email: cfink@ies.aust.com

| [Home](#) | [Courses](#) | [Certification](#) | [Projects](#) | [Papers](#) | [TEN Archive](#) | [EA Blog](#) | [Online Store](#) |
[Contact Us](#) | [Search](#) |

(c) Copyright 2004-2008 Information Engineering Services Pty Ltd. All Rights Reserved.