

Pipeline

Knowledge Is Power

www.pipelinepub.com Volume 4, Issue 12

Application Integration Using the SID: A Review

by Wedge Greene

This month, analyst and frequent Pipeline contributor Wedge Greene of LTC, International took a look at the new book, *Application Integration Using the SID* by John Reilly and John Wilmes, which was published by the TM Forum, April 2008. Here is his review:

A Tangible Step Forward in Making NGOSS Work

For years, service providers and vendors have been working with the TeleManagement Forum to find ways to slash the integration tax, transition away from expensive legacy environments, and move toward fast, flexible, and low-cost infrastructures. Thanks to much diligent work, the resulting New Generation OSS (NGOSS) concept was born and over the years has evolved to be more comprehensive - and more comprehensible, too. Corollary tools like the Shared Information DataModel (SID) and the enhanced Telecommunications Operations Map (eTOM), and most recently the Telecommunications Applications Map (TAM), have all been developed to set out a complete vision of that streamlined and efficient world.



© 2006, All information contained herein is the sole property of Pipeline Publishing, LLC. Pipeline Publishing LLC reserves all rights and privileges regarding the use of this information. Any unauthorized use, such as distributing, copying, modifying, or reprinting, is not permitted. This document is not intended for reproduction or distribution outside of www.pipelinepub.com. To obtain permission to reproduce or distribute this document contact sales@pipelinepub.com for information about Reprint Services.

Not for distribution or reproduction.

More and more service providers are undertaking NGOSS projects or considering its use as a unifying architectural framework. However, the scope of the NGOSS framework serves up a sizable barrier to figuring out just where to start and in which direction to proceed. Most of the architects and developers who actually understand NGOSS have worked on the TMF teams and constitute just a small fraction of the resources service providers need to marshal to succeed at NGOSS-driven projects. Membership in the NGOSS club therefore remains too exclusive for it to enable realization of its potential.

This month, we are pleased to welcome a new book providing additional practical insight on how to use the TeleManagement Forum's New Generation Operations and Systems Software (NGOSS) framework to achieve real results.

Some readers already will be familiar with one of the authors, John Reilly. Reilly contributed to NGOSS's eTOM framework and significantly to the SID framework. Perhaps more importantly, John has taught as many as half of all the students who have ever taken a TMF sponsored NGOSS training seminar. For his contributions, John was made a TMF Distinguished Fellow. Along with Martin Creaner, John coauthored the first NGOSS book: *NGOSS distilled*. He then further collaborated with Creaner in *Getting Started with the SID*. This book, just as *Getting Started with the SID*, is sponsored by Progress Software - who this time have one of their own as co-author in John Wilmes. Wilmes is continuing Progress Software's contributions to NGOSS by taking up co-chair of the SID team. Sponsorship by Progress Software provides the additional bonus of an included CD-ROM containing the current release of their free model browser for the SID so you can see the larger context of the examples of integration.



By now, everyone agrees that the current OSS/BSS environment of large numbers of independent applications (those developed internally as well as those purchased from vendors), dedicated databases, redundant security, housekeeping, and management tools add up to an environment whose cost to operate and maintain can no longer be sustained. Telecom IT must be retasked from specific, local jobs into a federation that serves the corporation as a whole. We, unfortunately, also have come to understand that making a move is difficult, expensive, and fraught

with risk. The integration tax is all too real.

One of the early successes of NGOSS was to provide a framework of resources and techniques that enabled some successful integration projects. However, as practitioners solved the platform communication problems initially with message busses, then ESB and now SOA, they came to confront a more intransigent problem: reconciling the many data representations, data structures, databases, and commercial APIs. At its worst, this became a black hole, sucking in all available resources for integration and transformation.

The key players in the TMF NGOSS groups began to sharpen their focus on developing more pragmatic implementation support. *Application Integration Using the SID* provides the template for the “data-driven” approach to application integration.

After a short and succinct explanation of NGOSS as used for integration projects, the book provides guidelines for approaching integration from several viewpoints. This book is not intended to explain NGOSS or the SID; its job is to teach a way of using the frameworks. A reasonable knowledge of NGOSS, UML, and the SID in particular is a pre-requisite for using this book. Structurally, the book uses a drill down approach of examining real examples in successive iterations of detail. It explores from different roles attached to people on the project (business architect, data architect, data modeler, and the developer doing data mapping). It looks at projects that start greenfield from the SID and those that must adjust and link existing interfaces and databases.

This book looks at the integration problem from the perspective of a SOA interface designer: how to define the interfaces so that they can be understood and correctly used by other services. It begins by explaining how to use the eTOM process framework as a source for designing business-oriented use cases which in turn will drive the specification of NGOSS Contracts. It provides guidelines for tough problems like “what is the right size for an NGOSS Contract interface?” It explores creating task-centric (one atomic job), utility-centric (commonly re-used core services), and entity-centric (manage this) service types. It also provides a roadmap for the NGOSS canned interfaces: OSS/J and MTOP. Then it explores the use of tools to automate the generation of machine-implementable interface code, specifically androMDA, Tigerstripe, and the “NGOSS Contract Factory.”

With this foundation laid, the book then proceeds to examine how a data modeler should approach integration, specifically, how to leverage the SID as a core facilitator of integration projects. We see the how an architect can use the common information model, in conjunction with eTOM and TAM frameworks, to:

- Establish scope
- Identify areas in the SID which cover the business problem at hand
- Use the SID definition to analyze and describe ideal interfaces
- Map these into an interface specification technology

Those familiar with the OmniPoint approach to TMN CMIP, will recognize this as a “profiling” exercise. A developer will harmonize with the successive decomposition from the abstract model to the code. The reader is walked through the foundation ABEs that will appear in most every integration project. This is critically important because these represent the common, “power” abstractions that are present in the

SID but are unlikely to have direct representation in the semantics of any physical or logical interface deployed before the SID. Next, the authors stress analyzing the resulting interfaces to extract opportunities for common repeatable use, a goal which will resonate with SOA practitioners. What follows then are the process steps and methods used in mapping interfaces to and from the common model. We found this section immensely valuable.



To actually do data-centric integration, the architect and developer will use modern toolkits and target specific implementations. Both the approach and tools must be selected. To aid in this, the authors lay out clear requirements needed by data integration tools in order to perform the methods described above successfully. Managers who must evaluate proposals and decide among competitive modeling schools will resonate with this chapter. The authors look at ways of importing, modeling, and extending the SID as common model. Then they compare XSLT-based tools to UML-based data mapping tools. Although the SID supports both expressions, the authors clearly conclude in favor of the UML described models and automatic data mapping tools. Progress Software's DataXtend product is used as the example of a data-mapping tool. One can excuse the prominent pitch for the sponsor, since the authors succeed in making their case. Even though the book recommends a UML-based approach, a w3C web services designer who plans to use XSLT will still find the techniques taught for SID-driven integration valuable.

Finally, the book explores four case studies that use some of the different techniques described in the book. The only sales pitch here is for NGOSS. These case studies teach the profiling methods and mapping techniques through concrete example. However, this is the weakest part of the book as only the most general approaches described above show up in these studies. Best of the lot is, not surprisingly, the description of the TMF Catalyst project that demonstrated the feasibility of data-mapping as an integration tool.

The Bottom Line

I classify this as "chapbook" deliberately and with honor. Throughout the history of publishing, chapbooks have made knowledge, usually drawn from other sources,

accessible to the public through reproduction and commentary with the spice of a relevant twist added by the authors. Thomas Paine's chapbook "Common Sense" made the concept of a democratic revolution accessible to the colonial occupants of what would become the USA. "These too are times which try" the intellect and will of those of us trying to establish logical and cost effective OSS/BSS infrastructures. Today, a chapbook still refers to an accessible, inexpensive booklet. At 160 pages, this book is relatively brief (especially when you consider that the NGOSS specifications take up thousands of pages in dozens of documents). *Application Integration Using the SID* can be a quick read, and returned to time and again for reference. Following the recommendations in this book significantly lowers the cost of entry to NGOSS projects by providing a clear, data-driven approach to application and service integration. It will save you lots of time and currency.

If you have news you'd like to share with Pipeline, contact us at editor@pipelinepub.com.