



Processes in Practice

Rob Davis
Senior BPM Consultant
IDS Scheer

Rob.Davis@ids-scheer.com

Process Reuse – do we understand what it really means?

Reuse is a key enabler for improving business efficiency. It's a global market, We can't afford to do things differently in every country and for every product, but at the same time we want flexibility and innovation...and we don't want to keep paying to have new processes and IT systems developed. So we have to reuse everything we can, but, in practice, reuse is harder than it sounds, both conceptually and managerially.

The concept of reuse has been around a long time, and it is widely employed for IT systems and software development. The rise of BPM techniques and the desire to better manage processes has led to increased awareness and desire to reuse processes. However, reuse is no "silver bullet"; it works well for small granularity "black box" physical components (e.g., Windows widgets), but it is harder to make work for larger, more complex and abstract components. Interest in process reuse has also been fueled by the development of Service Oriented Architecture (SOA) approaches to IT development. SOA has raised awareness in the IT community of the need for effective processes to reuse the SOA components, and it has also become apparent that BPM is a key enabler for the effective definition of SOA services (components). In the long term, the real goal is the development of "reusable business services" that comprise people, process, and system (IT).

However, before thinking about the long-term goals, we first need to understand what reuse is and how we can apply it to processes.

Wikipedia defines reuse:

"To use an item more than once. This includes conventional reuse where the item is used again for the same function, and new-life reuse where it is used for a new function."

This is quite a good definition, but, of course, we might achieve this in an accidental or serendipitous way so need we need to amend it to read, "*The planned use of an item more than once.*"

We can employ reuse in many ways. Employing best practice is one form of reuse (we are reusing concepts and ideas); process standardization is another (we are reusing the whole process); reusing process components or business services is yet another. A good process is one that does not reinvent everything from scratch, but reuses many process and IT components.

The Difficulties of Reuse

It is important when designing or modeling processes to build in reuse right from the start. However, simply copying a process block from one model to another does not guarantee actual reuse when the process is implemented. Designers can fool themselves into thinking they are

achieving reuse when in practice they are not. For instance, consider a company that sells products to retail customers and also delivers large consulting projects to corporate customers by making use of those products. The question is: Can we reuse the same sales process for retail customers and corporate customers? At first sight, the answer would seem to be yes. A typical sales process might be comprised of the steps shown in Figure 1.

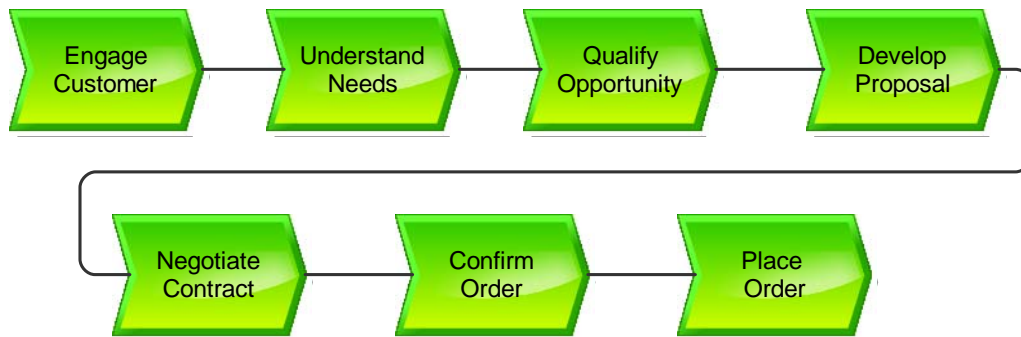


Figure 1. A Typical Sales Process

These steps would seem to apply to both types of customers, but for a retail customer (who may be purchasing on-line) many of these steps would be very lightweight. In fact, most of the steps would be done by the customers themselves clicking on a web page while the process itself would not be involved in understanding needs or in negotiation. However, for a corporate customer these initial process steps could be extensive and detailed. So, although the basic steps are the same for both customers, the detail is very different, and we would be fooling ourselves if we said we were reusing the same actual process. What we are in fact doing is reusing the same process “pattern” or template that is based on industry best practice for selling. This is a valuable form of reuse, but it is different from actual process reuse where the same people using the same IT systems go through exactly the same steps for both categories of customer.

So when we design processes, and especially when creating a corporate process architecture, we need to be very clear about exactly what is being reused. Unfortunately, there is currently very little understanding about the different types of reuse, most people being focused (and targeted) on achieving process standardization. However, standardization, while yielding the most benefit, is also the hardest to achieve, and many people get lost in the complexity of trying to promote standardization while managing many product and global variations. What is actually needed is a more informed and flexible approach to process reuse that recognizes that there are different types of reuse, with different benefits, and that some are more appropriate than others in different situations.

The example above also highlights one of the main issues with reuse – the difficulty of describing reusable components. Simply looking at the name of a process is not enough. We need sufficient information to understand the exact function of the process, what context it is intended to be operated in, and what constraints and conditions it imposes. The IT world often talks of “black box” components where the only information provided is a functional description and details of the component interface. The “black box” concept assumes that what’s inside the box (i.e., how it is implemented) is not important. However, in the process world, experience tells us we need to take a more “glass box” approach where we can look inside the process component and see how it operates and interacts with the rest of the business. The interfaces between a process and the business are often more complex than a low level IT component.

We also have to consider the non-technical, but nevertheless fundamental, issue of who pays for reuse and what they get for it. This topic was much researched about 10 years ago under the banner of Component Based Development (CBD) [1], which was striving to develop techniques to

reuse large scale IT components. This was not dissimilar to the current aims of process and business service reuse. The wisdom developed at the time was that a component had to be used in at least three different applications before it could be said to be “reusable.” In addition, it was necessary to provide corporate funding to develop the component in a way that would make it truly reusable. Each individual customer of the component was only likely to pay for the development of the functionality that met their needs. A voluntary and altruistic approach to designing components for the common good was unlikely to succeed without some central funding.

Types of Reuse

In order to progress with effective process reuse we need to clearly understand the different types of reuse. Ideally, the BPM community should develop an agreement on reuse taxonomy so we can start to talk about reuse in an informed way, using a common language. Below is an initial attempt to define the key categories of process reuse.

Structural Reuse

Structural reuse is the reuse of key concepts and best practice that removes the need to start from scratch when creating new process designs for a business or organization. These are rarely reused in their entirety, but adapted and extended to meet specific business needs. An important element of structural reuse is that adherence to these frameworks provides a common language for cross-organizational working, procurement, and industry collaboration.

- Architectural Frameworks (e.g., Zachman, TOGAF)
- Generic Process Frameworks (e.g., APQC)
- Industry Process Frameworks (e.g., SCOR, eTOM)
- Industry Best Practice (e.g., ITIL, MIT)

Design Reuse

Design reuse is the systematic and planned reuse of common approaches and resources to achieve business objectives. It would typically be implemented in a hierarchical way. At the highest level, generic process patterns would be established and reused (e.g., the sell process example above), while at the lowest level business services and process components would be reused. In the middle ground we strive to design standard processes (e.g., a single process implemented by the same people and systems) for all products and locations, etc., but recognize that it is often necessary to reuse the same process pattern while having different variants of the actual process for different scenarios.

- Generic process reuse – high level patterns; variance at detailed level
- Process variants – modifications of previous process designs
- Patterns – designs for common business activities
- Business services – common business capabilities
- Standard processes – a single process implementation for all scenarios

Implementation Reuse

Implementation reuse is a slightly artificial category, but it refers to items that can be reused by a process, but that are not specifically defined or designed-in by the process community. For instance, if the IT department has implemented a single order handling system, then this will be reused by processes, but will not have specifically resulted from process design reuse. Similarly, SOA components may be reused or the processes may implemented by shared services organizations. However, to get the real benefit from IT and shared service reuse, there should be joint reuse-focused design by the IT and process communities. There are now more developments in this area, particularly as the IT community is starting to see the value of process in delivering SOA.

- Shared service processes – business functions delivered by a common process
- IT Applications

- IT Services – SOA components
- Workflow patterns

Achieving Reuse

To achieve effective reuse we need

- A better understanding of reuse principles
- To understand the right granularity for reuse
- To think about reuse hierarchies and how they link with IT SOA
- To understand how to describe process components (and their context) so they can be reused
- Support from tools for designing-in and managing reuse
- To link reuse with managing variation and rules
- To move SOA techniques up the value chain
- To promote the concept of process patterns

The concept of design patterns has been around for some time and is widely used in software design, following the seminal work by Gamma, et al [2]. A process pattern can be described as

“An idea that has been useful in one practical context and will probably be useful in others” [3].

Process design sounds like an ideal application for such a concept, and all process designers must have noticed how often the same fragments of process (e.g., seeking authorization from a customer and sending them a reminder if it's not received) occur time and time again. However, patterns have been slow to be picked up by the process community (the last article in BPTrends was probably that by Dan Atwood in 2006 [4]).

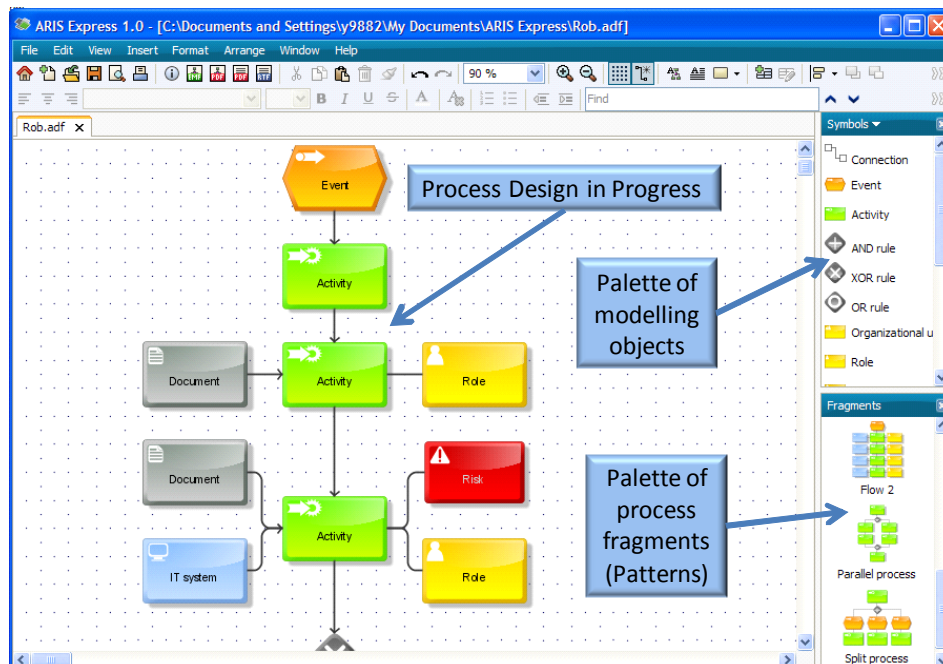


Figure 2. Process Fragments in ARIS Express

Initial work was pioneered by W.M.P. van der Aalst, et al., [5] who concentrated on low-level workflow patterns, and more recent work has focused on BPMN patterns [6]. However, there is no reason to think that the process community would not benefit from similar work to identify best practice business-level process patterns. In the longer term, making effective use of process patterns requires tool support. This is now starting to emerge, and Figure 2 shows how process fragments (patterns) are now supported by ARIS Express.

In summary, process reuse is a vital topic with significant potential benefit, and needs more discussion and research by the process community.

References

- [1] Davis et al. "Component-based Network Systems Engineering". Artech House, 2000.
- [2] Gamma et al. "Design Patterns: Elements of Reusable Object-Oriented Software". Addison-Wesley, 1994.
- [3] Stephenson, Christine, Bandara, Wasana. "Enhancing Best Practice in Public Health: Using Process Patterns for Business Process Management". In: ECIS 2007 - The 15th European Conference on Information Systems, 7-9 June 2007, St. Gallen, Switzerland.
- [4] Dan Atwood, "BPM Process Patterns: Repeatable Design for BPM Process Models". BP Trends May 2006.
- [5] van der Aalst, W.M.P., ter Hofstede, A.H.M., Kiepuszewski, B., and Barrios, A.P., "Workflow Patterns", Technical Report, Eindhoven University of Technology, Eindhoven, Distributed and Parallel Databases, 14(3), pages 5-51, July 2003.
- [6] Petia Wohed, Wil M.P. van der Aalst, Marlon Dumas, Arthur H.M. ter Hofstede, Nick Russell. "Pattern-based Analysis of BPMN - an extensive evaluation of the Control-flow, the Data and the Resource Perspectives". In: Information and Software Technology Archive Volume 50, Issue 12 (November 2008).