A BPTrends Column September 2010

Performance Improvement

Alan Ramias & Cherie Wilkins
Consultants
Performance Design Labs (PDL)

ARamias@ThePDLab.com
CWilkins@ThePDLab.com

Building Metrics for a Process

Our last column kicked off a series on process metrics. We started off by citing some of the problems and pitfalls we have encountered in our work with clients, such as creating metrics (and process management roles) that were unlinked to management of the business; creating disorganized piles of metrics instead of a logical set; measuring too much, too little, or the wrong things.

In this column we provide some principles and a tool to remedy the most significant problems. We will describe the guidelines we follow in creating process metrics for clients and will apply those guidelines using a tool for identifying the right process metrics. This tool is part of a larger toolkit that we employ when helping an organization build a comprehensive process-focused management system; this particular tool is central to the task of producing good, useful process metrics.

To apply the following principles and tools, some assumptions are necessary: You have singled out a business process, you have mapped it in enough detail to identify its major subprocesses or phases, and you understand the existing management system into which these process metrics will be inserted.

Principles for Process Metrics Design

- Every process is designed to reliably produce one or more outputs, so, in deciding what metrics to develop, we always focus at first on process outputs, not activities,. The metrics should measure whether the process not only produces the outputs but also that all appropriate expectations are met every time the process is executed.
- Metrics should be applied to all the significant outputs of the process. If the process is order fulfillment, for example, the output is not just the product but also the invoice, the order documentation, and customer information that will be used again for future orders.
- We always start outside the process itself and try to understand the expectations of the receivers of the outputs. The receiver may be an actual customer, or it may be an internal party who is also a "customer" for a given output. Regardless whether the process has an external or internal customer, the starting point is to understand what is important to that customer – what are the expectations that we can then translate into what we call the "critical dimensions of performance." Once we know the expectations for the process, we then create and distribute metrics along the process that measure all of the relevant critical dimensions of performance, such as timeliness, quality, economics, volume, compliance, and so on.
- In first developing metrics, we focus on *what* to measure, not how measurement is going to happen. There are several reasons for this: First, the decision about whether to create a given metric is more important at first than determining exactly how the data

might be tracked, reported, archived, and so on. While measurement can be costly and sometimes not worth the effort, we have watched some teams talk themselves out of a potentially valuable metric just because they weren't exactly sure at first how to collect the data. Usually, you have choices as to what to collect and report – for example, maybe you focus only on exceptions, or only reporting quarterly – that can reduce the cost. And metrics don't have to be perfect. Collecting data on secondary *indicators* of performance may be quite adequate for triggering a closer look, rather than collecting volumes of data that sometimes obscure what is actually going on. There is also a tendency these days to fixate only on data from systems, but visual data (go and look) and interview data (go and ask) are also viable ways to collect data.

- The most useful performance data helps one see trends in performance. So, metrics that can be constructed to yield a trend are the most useful, and most metrics can be formulated this way. For example, "number of defects" for a given output is not that useful except for fixing a single product, but number of defects by product type by day, week, and month could provide a lot of insight into where performance problems are originating. So in the table below, once we have identified all of the metrics we want, we turn them into trend data by adding an element that enables trend tracking (such as per lot, per day/week/month, per location per week, and the like).
- We seek to identify metrics that will be both leading and lagging indicators of performance. Lagging indicators are the common ones: they provide data on events in the past. But leading indicators provide insight into the future; they center on data that act as an early warning on emerging problems or declining performance. When chosen well, a leading indicator can signal the need for a course correction before the problem gets out of control. The tool we describe below is a great way to identify possible leading indicators.

The Measures Chain

The Measures Chain is used to develop metrics for a given process. The concept is shown in Figure 1. The essence of this tool is to identify and link external requirements to an internal process. The technique is to start with customer requirements and work backwards and downwards into more detail that would be used for each dimension of performance.

The starting point is outside the process, where the process output is received. In Figure 1, the process is order fulfillment, the output is a product, an invoice, and order documentation, and the receivers are customers in a given market. (However, this tool and its principles can be applied to processes that deliver outputs to internal "customers" too.)

Once we know what the customer of the process wants, we can identify where to place appropriate metrics inside the process to see if we are meeting the customer's expectations. We end up creating a "chain" of metrics (hence the name) related to some performance requirement, such as timeliness across the process.

To build a Measures Chain for a given process, we usually create a table like the one in Figure 2. We already know the process inputs and outputs as well as the outputs of each subprocess. We start by asking what the customer requirements or expectations are. These become the highest set of metrics – what we call M-1-External. In Figure 2, the M-1-E metrics for the product are in three dimensions (economics, timeliness, and quality) because the customer has requirements in those three categories. Specifically, the customer cares about percent of deliveries made on time and about price, so those are metrics we place in the table. We also decide to track customer complaints and returns because those are direct forms of feedback from the customer about product quality.

We then ask if there are additional business requirements for this output. Business requirements are those things a customer may not necessarily know or care about (our costs, our internal standards, our compliance requirements, etc.), but we want to measure this output against any of

those requirements that might exist. So in Figure 2 you can see business metrics (called M-1-Internal metrics) in the same dimensions of quality, economics, and timeliness, but price is now reclassified as profit, because the business wants to measure how profitable this product is, and timeliness is measured in process cycle time.

Then we go inside the process and place metrics on the outputs of the subprocesses (M-2 level) in the same dimensions of quality, timeliness, and economics. We can go even lower in the process, down to the activity level, and place M-3 metrics if we think this will provide insight about performance. For example, the output of the process for writing a book is a manuscript; one of the activities for producing the output is editing. One M-2 metric is the cycle time to edit the manuscript, but we may want an M-3 metric on the hours spent editing is that is a particularly problematic part of the process.

Possible Pitfalls

The Measures Chain is a great tool for identifying and classifying possible metrics for a given process, but there are some misconceptions to avoid:

It might seem as though the goal is to identify as many metrics in the Measures Chain as possible, but that should not be your aim. You don't need a metric for every dimension on every subprocess output. You will notice in the table in Figure 2 that there are blank cells. The goal is to identify metrics that provide insight into performance – especially ones that are leading – that give us early warning about possible performance issues. Nevertheless, it is tempting to fill in all the blanks on a Measures Chain Table. We have found that the best way to go about this is to fill out the table twice. The first time, just fill in all the cells with all the metrics that are logically implied. The second time, go through and choose only those metrics that will provide the greatest insight.

As we said earlier, in creating the Measures Chain, you have not necessarily formulated the metrics in a way that you will want to collect and track data. For example, how are we going to measure whether every order meets throughput standards? Will we be able to count every one going through? Can we date-stamp every order? Does a computer log show throughput time? All these are questions to be figured out once there is agreement that we need to track and understand certain kind of process performance, but we haven't figured out when, where, and how the tracking and reporting will happen, and we don't know yet who is responsible for keeping watch on the metrics.

In our next article, we will take on some of these questions, and provide some guidance and a tool for determining who should watch what performance data and what they should be looking for. Till next time.

Figure 1
Measures Chain Model

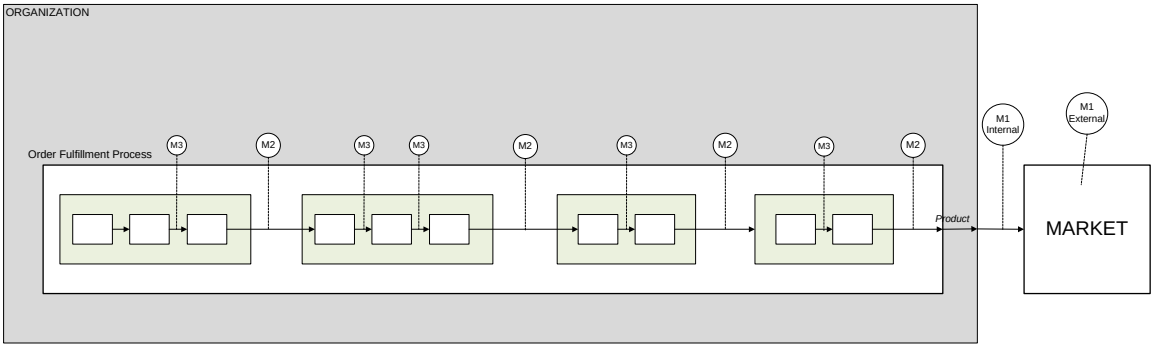

Figure 2
Measures Chain Table

Process: Order Fulfillment						
	Process Phase 1	**Process Phase 2**	**Process Phase 3**	**Process Phase 4**	**Internal End of Process**	**External End of Process**
	Order Entered	Order Assembled	Final Inspection Conducted	Order Shipped		
Inputs	**Outputs**	**Outputs**	**Outputs**	**Outputs**	**Final Outputs**	**Inputs**
▪ Customer order ▪ Product specs ▪ Schedule	▪ Order	▪ Product ▪ Shipping instructions	▪ Inspected product ▪ Inspected invoice	▪ Sealed Product ▪ Invoice	▪ Sealed Product ▪ Invoice	▪ Sealed Product ▪ Invoice
Economics Metrics						
% orders priced within standard costs	Processing cost per order	Manufacturing cost per order	Cost of inspection per order	Shipping costs per order	Total cost per order	Price per product
				Packaging costs per order	Cost of reworked orders	
					Product profitability (revenue minus cost)	
Timeliness Metrics						
	Cycle time of sub-process	Cycle time of sub-process	Cycle time of sub-process	Cycle time of sub-process	Total cycle time of process	% received on time

% orders received within standard delivery time	% entered per standard time	% assembled per standard throughput time	% inspected per standard throughput time	% picked up from dock on time	% shipped to standard schedule	
Quality Metrics						
% complete & accurate customer orders	# of order entry errors	# of production defects	# errors caught in inspection vs. by customer	# of shipping errors	% orders with errors identified in inspection	# of complaints
% complete & accurate product specs		% with shipping instructions included	% inspected according to sampling rules	% packaged per customer order	% orders reworked	# of returns

BPTrends Linkedin Discussion Group

We recently created a BPTrends Discussion Group on Linkedin to allow our members, readers and friends to freely exchange ideas on a wide variety of BPM related topics. We encourage you to initiate a new discussion on this publication or on other BPM related topics of interest to you, or to contribute to existing discussions. Go to Linkedin and join the **BPTrends Discussion Group.**