Defining Process Performance Indicators by Using Templates and Patterns*

Adela del–Río–Ortega, Manuel Resinas, Amador Durán, and Antonio Ruiz–Cortés

Universidad de Sevilla, Spain
{adeladelrio,resinas,amador,aruiz}@us.es

Abstract. *Process Performance Indicators* (PPIs) are a key asset for the measurement of the achievement of strategic and operational goals in process–oriented organisations. Ideally, the definition of PPIs should not only be unambiguous, complete, and understandable to non–technical stakeholders, but also traceable to business processes and verifiable by means of automated analysis. In practice, PPIs are defined either informally in natural language, with its well–known problems, or at a very low level, or too formally, becoming thus hardly understandable to managers and users. In order to solve this problem, in this paper, a novel approach to improve the definition of PPIs using templates and ontology–based linguistic patterns is proposed. Its main benefits are that it is easy to learn, promotes reuse, reduces ambiguities and missing information, is understandable to all stakeholders and maintains traceability with the process model. Furthermore, since it relies on a formal ontology based on Description Logics, it is possible to perform automated analysis and infer knowledge regarding the relationships between PPI definitions and other process elements.

Keywords: Business Process Management, Process Performance Management, Key Performance Indicator, Process Performance Indicator, Templates, Patterns.

1 Introduction

Many companies are adopting a process–oriented approach in their business. In order to measure progress towards their business goals, it is important to evaluate the performance of their *business processes* (BPs) by means of the so–called *Process Performance Indicators* (PPIs), a particular case of *Key Performance Indicators* (KPIs) dedicated to BPs. For example, for the process depicted in Fig. 1, some PPIs could be defined based on metrics such as the average time of the Analyse RFC activity, the registered/approved RFC ratio, or the average delay of elevating a RFC to committee.

PPIs are recommended to satisfy the SMART criteria [1], i.e to be *Specific*, *Measurable*, *Achievable*, *Relevant* and *Time–bounded*, but also to be understandable, traced to the related BPs and automatically analysable [2,3,4]. A notation for PPI definition satisfying these requirements is still a challenge, mainly because of the conflict between understandability and automatic analysis. In practice, PPIs are defined either in

* This work has been partially supported by the European Commission (FEDER), Spanish Government under the CICYT project SETI (TIN2009–07366); and projects THEOS (TIC–5906) and ISABEL (P07–TIC–2533) funded by the Andalusian local Government.

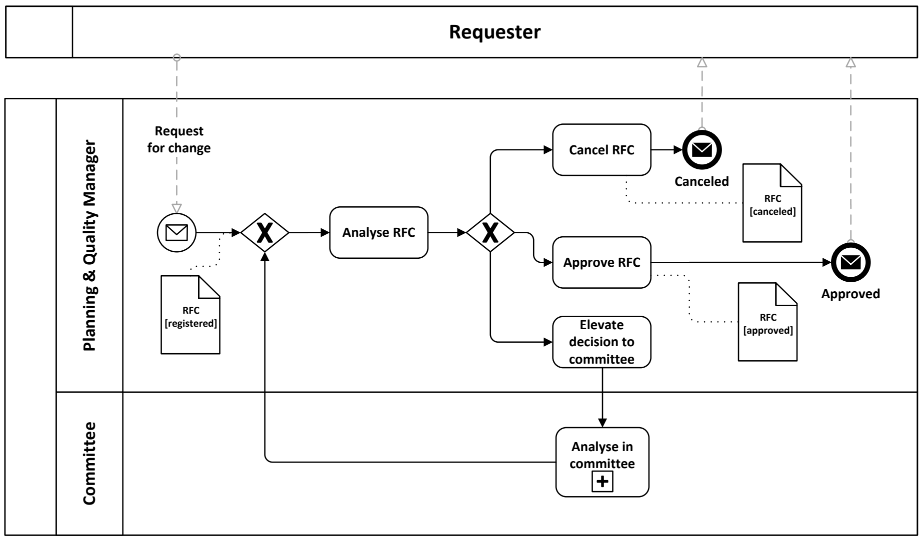

Fig. 1. Sample business process: Request for Change (RFC) management

(1) natural language, with its well–known problems of ambiguity and incompleteness; (2) at implementation level; or (3) too formally, becoming thus hardly understandable for managers and users.

In this paper we address this challenge and propose a novel approach to improve the definition of PPIs using templates and linguistic patterns (L–patterns, i.e. very used sentences in natural language that can be reused by parametrisation), which have been successfully applied in the areas of Requirements Engineering [5,6] and Service Level Agreements [7]. The proposed notation is formally supported by the PPINOT ontology [3], allowing their automated analysis using Description Logics.

2 PPI Template

Our proposal for PPI template, inspired by the requirements templates originally proposed in [5], is shown in Table 1 and an example is shown in Table 2. It has been designed in order to fulfil the SMART criteria [1] and is heavily based on the PPINOT ontology [3]. As commented in [5], using templates helps to organise the information in a structured form, reduces ambiguity, promotes reuse, and also serves as a guide to avoid missing relevant information. The notation used in the template is the following: words between “<” and “>” are placeholders for either literals (lower case) or L–patterns (upper case); words between “{” and “}” and separated by “|” are one–only options; words between “[” and “]” are optionals. The meaning of the template fields is the following:

- **Identifier** and **descriptive name**: unique PPI identifier, needed for traceability, and a self–descriptive name for the PPI.

Table 1. Template for PPI specification

<table>
<tr><td>PPI–<<i>ID</i>></td><td><i><PPI descriptive name></i></td></tr>
<tr><td>Process</td><td><<i>process ID the PPI is related to</i>></td></tr>
<tr><td>Goals</td><td><<i>strategic or operational goals the PPI is related to</i>></td></tr>
<tr><td>Definition</td><td>The PPI is defined as {
<<i>DurationMeasure</i>> | <<i>CountMeasure</i>> | <<i>ConditionMeasure</i>> |
<<i>DataMeasure</i>> | <<i>DerivedMeasure</i>> | <<i>AggregatedMeasure</i>> }
[expressed in <<i>unit of measure</i>>].</td></tr>
<tr><td>Target</td><td>The PPI value must {
be {greater | lower} than [or equal to] <<i>bound</i>> |
be between <<i>lower bound</i>> and <<i>upper bound</i>> [inclusive] |
fulfil the following constraint: <<i>target constraint</i>> }</td></tr>
<tr><td>Scope</td><td>The process instances considered for this PPI are {
the last <<i>n</i>> ones |
those in the analysis period <<i>AP–x</i>> }</td></tr>
<tr><td>Source</td><td><<i>source from which the PPI measure can be obtained</i>></td></tr>
<tr><td>Responsible</td><td>{ <<i>role</i>> | <<i>department</i>> | <<i>organization</i>> | <<i>person</i>>}</td></tr>
<tr><td>Informed</td><td>{ <<i>role</i>> | <<i>department</i>> | <<i>organization</i>> | <<i>person</i>>}</td></tr>
<tr><td>Comments</td><td><<i>additional comments about the PPI</i>></td></tr>
</table>

- **Process** and **goals**: traces to the process for which the PPI is defined and to the strategic or operational goals the PPI is related to (*Relevant* SMART criteria).
- **Definition**: kind of measure and units, if needed, the PPI is based on (*Specific* and *Measurable* SMART criteria). Corresponding measure L–patterns are described in next section.
- **Target**: target value of the PPI for achieving previously referenced goals (*Achievable* SMART criteria).
- **Scope**: number of process instances or analysis period considered for computing the PPI value (*Time–Bounded* SMART criteria). Due to space limitations, analysis period descriptions are not included in this paper (see [3,4] for more information).
- **Source of information**: source from where the required information to compute the PPI is gathered.
- **Responsible** and **Informed**: resources in charge of or interested in the PPI. They can be persons, roles, departments or organisations.
- **Comments**: any other relevant information that cannot be fitted in previous fields.

3 L–Patterns for PPI Specification

Following [5,6], L–patterns are integrated in the proposed PPI template because filling blanks in prewritten sentences is easier, faster and less error–prone than doing it from scratch. The six proposed L–patterns are described in this section.

Table 2. PPI specification example

PPI–001	**Average time of RFC analysis**
Process	Request for change (RFC)
Goals	• BG–002: Improve customer satisfaction • BG–014: Reduce RFC time–to–response
Definition	The PPI is defined as *the average of* Duration of Analyse RFC activity.
Target	The PPI value must be lower than or equal to 1 *working day*.
Scope	The process instances considered for this PPI are the last *100* ones.
Source	Event logs of BPMS.
Responsible	*Planning and quality manager*
Informed	*CIO*
Comments	*Most RFCs are created after 12:00.*

3.1 Duration Measure L–pattern

In the PPI context, a duration can be defined as the difference between two events, considering as events not only BP event triggerings but also BP element transitions. Following the BPMN 2.0 specification [8], we consider *activities*, *pools* and *data objects* as elements; and *ready, active, withdrawn, completing, completed, failing, failed, terminating, terminated, compensating* and *compensated* as states (data object states are user–defined). Having said that, the *DurationMeasure* L–pattern can be defined as:

the duration between the time instants when <*event*$_1$> *and when* <*event*$_2$>

where <*event*> is defined as:

{ <*BP element*> *changes to state* <*BP state*> | <*BP event*> *is triggered* }

For example, in order to measure the duration of the Analyse RFC activity, the L–pattern can be instantiated as:

the duration between the time instants when RFC analysis activity changes to state active *and when* RFC analysis activity changes to state completed

3.2 Count Measure L–pattern

A count measure for PPIs counts the number of times a specific *event*—as considered in previous section—happens. Therefore, its corresponding L–pattern is as simple as *the number of times* <*event*>, for example:

the number of times Analyse RFC activity changes to state completed

3.3 Condition Measure L–pattern

A condition measure takes boolean values depending on either the state of a BP element or a condition specified on a data object. The two corresponding L–patterns are:

<BP element> { *is currently* | *has finished* } *in state <BP state>*

Data object <object> satisfies: *<condition on object properties>*

For example:

Activity Analyse in committee *is currently in state* active

Data object RFC *satisfies*: priority = high

3.4 Data Measure L–pattern

A data measure takes the value of a specific property of a data object. The L–pattern is as simple as: *the value of <property> of <object>*. For example, assuming the RFC data object has a property indicating the affected departments:

the value of affected departments *of* RFC.

3.5 Derived Measure L–pattern

A derived measure is a function defined over other measures expressed using some of the previous L-patterns. For the sake of simplicity, they are referred to by means of a symbolic name. In this case, the L–pattern includes the expression of the function and a mapping from function variables to the measures of other measures:

the function <expression over $x_1 \ldots x_n$*>, where* $\{$ *<*x_i*> is <Measure*$_i$*>* $\}_{i=1..n}$

For example, assuming two Measures such as Number of approved RFCs and Number of registered RFCs, a derived measure for the ratio of RFCs approved from registered could be defined as:

the function $\frac{a}{r} * 100$, *where* a *is* Number of approved RFCs and r *is* Number of registered RFCs

3.6 Aggregated Measure L–pattern

In a similar way to derived measures, aggregated measures are defined over one of the previous measures by applying one aggregation function, i.e. sum, maximun, minimum, average, etc. The corresponding L–pattern is the following:

the { *sum* | *maximum* | *minimum* | *average* | ... } of *<Measure>*

An example of the use of an aggregated measure L–pattern can be seen in the sample PPI definition in Table 2.

4 Conclusions and Future Work

As a major conclusion we can claim that it is possible to define PPIs with a notation that is easy to learn, promotes reuse, reduces ambiguities and avoids missing information, is understandable to all stakeholders, maintains traceability with the process model, and can be automatically analysed. The only price to pay is to restrict the employed sentences to the ones allowed by the underlying PPINOT ontology [3].

Some possible lines for future work can include adapting templates when more feedback from real scenarios is available, discovering more patterns, specially for the definition of resource–aware PPIs, and developing a tool to integrate it into the PPINOT tool, allowing thus the definition of PPIs through either the approach presented here or using our graphical notation, and their subsequent analysis, enabling also the automatic generation of documentation.

References

1. Shahin, A., Mahbod, M.A.: Prioritization of key performance indicators: An integration of analytical hierarchy process and goal setting. International Journal of Productivity and Performance Management **56** (2007) 226 – 240
2. Franceschini, F., Galetto, M., Maisano, D.: Management by Measurement: Designing Key Indicators and Performance Measurement Systems. Springer Verlag (2007)
3. del Río-Ortega, A., Resinas, M., Ruiz-Cortés, A.: Defining process performance indicators: An ontological approach. In: Proceedings of the 18th International Conference on Cooperative Information Systems (CoopIS). OTM 2010, Part I. (October, 2010) 555–572
4. del Río-Ortega, A., Resinas, M., Ruiz-Cortés, A.: PPI definition and automated design-time analysis. Technical report, Applied Software Engineering Research Group (2012)
5. Durán, A., Bernárdez, B., Ruiz-Cortés, A., Toro, M.: A Requirements Elicitation Approach based in Templates and Patterns. In: Proc. Workshop de Engenharia de Requisitos (WER'99), Buenos Aires, Argentina (1999)
6. Durán, A., Ruiz-Cortés, A., Corchuelo, R., Toro, M.: Supporting Requirements Verification using XSLT. In: Proc. IEEE Joint International Conference on Requirements Engineering. (2002) 165–172
7. Ruiz-Cortés, A., Martín-Díaz, O., Toro, A.D., Toro, M.: Improving the automatic procurement of web services using constraint programming. Int. J. Cooperative Inf. Syst. **14**(4) (2005) 439–468
8. OMG: Business Process Model and Notation (BPMN) Version 2.0 (Jan 2011)