

Business System Support Case Study

Author: Keith Harrison-Broninski

BUSINESS SYSTEM SUPPORT CASE STUDY 1

INTRODUCTION 1

Purpose 1

Scope 1

Structure 1

BUSINESS SYSTEM SUPPORT – PROCESS DIAGRAM 2

EXAMPLE SCENARIOS 3

Urgent new requirement during release cycle 3

Two concurrent and interdependent projects 3

Writing of functional specifications 4

AUTHOR BIO 4

Introduction

Purpose

The process described in this document was originally compiled to aid in the implementation of business system support within a global manufacturing company, for use in ERP enhancement or upgrade, reporting suite, other application maintenance, and any other type of business system support, whether or not that work is formally considered to be a “project”.

The person responsible for re-engineering this process spent several months drawing up process documents which no-one used, and struggled even to understand, before turning to the techniques of Human Interaction Management, and finding that everything they had written so far, and more, could be expressed in a single 1-page diagram – further, a diagram that elucidated the processes concerned so clearly that all parties involved were able to agree on a re-engineered version within 2 meetings.

Scope

The document provides a single, simple diagram that shows all Roles involved in the process – these Roles are generic, whatever the type of project. Their interactions are shown as clear boxes. Their activities are shown as shaded boxes.

The notation used is a simplified and enhanced form of the standard process modelling technique known as Role Activity Diagramming – for further details on this technique, see <http://www.human-interaction-management.com>.

The process diagram is readable by anyone without training, and was used for publication on an internal company Web site for use by everyone within the teams concerned. However, the diagram is further illustrated and explained in this document via the use of example scenarios.

Structure

The document has 2 main sections:

Example Scenarios

In this section we present some example scenarios: typical situations from business system support that illustrate the process shown above in action.

Urgent new requirement during release cycle

Let us suppose that a new business requirement has arisen. The Requirement Sponsor enters the requirement into the Bug Tracking System as usual, but the next release cycle has already been prioritized and planned – development is already underway. However, the Requirement Sponsor judges that the requirement is sufficiently urgent to warrant inclusion in the current release anyway.

In this case, the Requirement Sponsor should **not** approach functional developers, IT or the Implementation Owner directly. Their first step should be to approach a member of the Prioritization Team. This person will consider the request and make a judgement as to whether they agree with the Requirement Sponsor about the urgency of the requirement.

If the Prioritization Team Member does agree with the Requirement Sponsor, they will approach the Release Manager to request that the requirement is included in the current release. The Release Manager will consider whether or not this can be done without impacting release dates.

If it can be done without impact on release dates, the Release Manager will arrange with the Implementation Owner for the new work to be carried out.

Otherwise, the Requirement Sponsor still has the option to escalate the matter to the Business Systems Owner, who will make their own judgement on the matter. If the Business Systems Owner decides the work should be included in the release after all, they will notify both the Release Manager and the Requirement Sponsor.

The Release Manager will then have to adjust their priorities. Some work may have to be left out of the release, or the manner in which it is carried out changed, to free up enough development resource to carry out the new work. Once the Release Manager has made this adjustment, they will notify both the Implementation Owner (who will in turn notify their team of the priority adjustment) and the Business Systems Owner.

Note that if it is necessary to alter the work carried out by IT as well as by developers, this can only be done in the beta release. The requirement to do this will be supplied to IT along with the alpha test results. Any functional changes in the new requirement that are dependent on IT will then be done as part of the beta release.

Two concurrent and interdependent projects

It is generally the case that more than one development process is underway within Business System Support at the same time – and there will inevitably be

interdependencies between the various streams of work. This is managed via the following simple means: the person who is acting as Implementation Owner for each project may also be a Requirement Sponsor on other projects.

In other words, if project 1 has dependencies on project 2, the Implementation Owner for project 1 should use the Bug Tracking System to enter these dependencies as requirements on project 2. Then they will be prioritized and planned as per usual. Similarly, the Implementation Owner for project 2 may enter bugs for project 1, if project 2 also has dependencies on project 1.

Note in particular that the same exception and escalation mechanism described above can be used if dependencies on another project only manifest themselves once development work on that project has already started.

Writing of functional specifications

In order that each Implementation Owner can maintain proper control over the systems they are responsible for – keep each system **consistent**, and **maintainable** going forward – the development process assumes that:

- Business requirements will be submitted by any member of Business System Support, via the Bug Tracking System, using a template that provides sufficient information for the work to be actioned.
- Functional change specifications will be created and revised only by members of the functional team concerned with that particular system, who are familiar with all parts of the system concerned and its intended future maintenance path.

Hence we see in the above process diagram that specifications for each change are drawn up by Functional Development rather than the Requirements Sponsor.

This is not to say that each functional specification will result in new or revised *program* specifications. In many cases, depending on the nature of the change required, it will be appropriate to provide details of the change in a small document separate from existing program specifications.

It is the responsibility of the Implementation Owner to make sure that the documentation for their entire system is kept up to date, in whatever form they judge to be most suitable for use by Business System Support and future maintenance. Some aspects of the system may be better documented via an online data dictionary than by text documents, for example.

Author Bio

Keith Harrison-Broninski (keith.harrison-broninski.info) is a consultant and software developer whose book “Human Interactions: The Heart And Soul Of Business Process Management” is described by industry commentators as “[a key component of] the

overarching framework for 21st century business technology”(www.bptrends.com) and
“*a must read for Process Professionals and Systems Analysts alike*” (www.bpmg.org).
Find out more about the process modelling and management techniques of Human
Interaction Management at www.human-interaction-management.info.