Increase productivity with a proven methodology for change request management

by Anton Karnaukhov

Anton Karnaukhov
Business Process Technologies Manager,
Pacific Coast Companies, Inc.

Anton Karnaukhov is manager of the Business Process Technologies group at Pacific Coast Companies, Inc., in Sacramento, California. He has an MBA from Heriot-Watt University and a BS/BA from Western Carolina University specializing in computer information systems. He has managed SAP ERP and SAP BI functions since 2006 focusing on corporate information management, IT process optimization, and service delivery. Anton has more than six years of SAP implementation and development experience. You may contact him via email at anton.karnaukhov@paccoast.com.

When the dust settles after an initial SAP ERP implementation, SAP shops typically find themselves facing a continuous stream of service requests — everything from a new pricing condition type for a customer discount in the SAP Sales and Distribution (SAP SD) module to a new query in the SAP NetWeaver Business Intelligence (SAP NetWeaver BI) system. These requests aren't usually part of a formal project; they tend to be standalone units of work that require less than a week apiece to implement. However, don't think that handling change requests is a simple task. Business process experts (BPXs) can expect to spend anywhere between 15% and 30% of their time managing and implementing change requests. (This estimate is based on my personal observations and experience managing SAP teams.) That's why it's so important to have a reliable, scalable methodology for managing change requests, one that each member of the BPX team can easily understand and follow.

All change requests should be entered into a single database. Only a specific group of key authorized business users can initiate a change request entry, ensuring that a business unit (BU) authorizes each change request. During its lifetime, a change request goes through a number of key phases, including approval, realization, and business testing. In each phase, the system tracks certain key performance indicators (KPIs) to ensure that the management of the change request process is transparent to both IT management and the individual BPXs. The system then groups these KPIs logically and presents them on a number of dashboards, allowing real-time visibility into the overall effectiveness of the process, workload distribution, and the existence of any bottlenecks along the way. The KPIs also provide a solid foundation for managing and increasing the productivity of the BPX team.

This article discusses a methodology for managing the change requests. It covers each of the key phases of change request management: the approval of IT to proceed with the proposed change, the realization of a

What is ITSM?

In the 1980s, the British government developed a collection of best practices observed in the IT service industry and published them under the title Information Technology Infrastructure Library (ITIL). ITIL provides a detailed description of a number of key IT functions along with a comprehensive list of processes, tasks, procedures, responsibilities, and KPIs that you can tailor to any IT organization. These best practices — more specifically, service delivery and service support — have evolved into a body of knowledge now typically referred to as IT Service Management (ITSM).

The main manifesto of ITSM is that providers of IT service can no longer focus just on technology and output; rather, they have to concentrate on the quality of the services they provide and focus on their relationship with their customers. This internal transformation from being product-focused to being process- and service-focused is certainly nothing new in the world of business management, but IT management has been falling behind for years. Thus, it's no surprise that many of the principles of ITSM take their roots from the work of such Quality Management gurus as Dr. W. Edwards Deming (www.deming.org) and Dr. Joseph M. Juran (www.juran.com).

developed solution, and testing by the business side to ensure that the solution achieves the change requested. The article also provides a series of KPIs to help you manage this process. Then, the article discusses the technical information you may want to include in a listing of change requests. It also contains a recommended workflow for change requests.

Everyone involved in the change request process will find interest in this article: business users, BU managers, IT managers, project managers, and BPXs. Why would project managers care? Because a complex change request may become a project if it exceeds the scope of a change request.

Now, let's examine what change request management actually means and how it differs from change management.

Defining change request management

First, I'll describe the difference between *change* management and change *request* management:

- **Change management** is the term with which you're probably more familiar. It focuses primarily on implementing standardized methods and procedures to minimize the risk of a negative impact when you install system changes in the live environment. This discipline focuses on managing *change* and, thus, spans a number of key topics including identifying, communicating a need for, and recording changes; assessing the impact, cost, benefit, and risk of proposed changes; developing business justification and obtaining approval; managing and coordinating a proposed change implementation; and monitoring and reporting on the implementation itself.
- **Change request management** is a subset of the change management process. It focuses on managing the *requests* for system changes, as opposed to the changes themselves. This process is the focus of this article.

The IT Service Management (ITSM) methodology has recently popularized the term *change request* (also called *Request for Change* in the ITSM world). For more information on ITSM, see the sidebar on this page.

A change request is a specific document that makes a formal request for a change within a system. According to ITSM guidelines,[1] a change request is one possible outcome of a service request. Other potential outcomes include providing production support over the phone or email, helping with mass changes to master data, or even requesting formal training. In fact, you can consider nearly any interaction between an IT service provider employee and a user to be a service request, whether it is production support, bug resolution, or an inquiry about possible functionality. Service requests are not limited to interaction with developers; they also include interactions with help desk technicians, systems analysts, and even IT management.

You can measure the *maturity* of an ITSM implementation by the breadth (number of processes) and depth (level of detail per managed process) of the IT processes that you have brought under the ITSM umbrella. ITSM deals with the management of a number of major IT processes:

- Incident management
- Problem management
- Configuration management
- Change management
- Release management
- Service level management
- Financial management
- Capacity management
- Service continuity management
- Availability management
- Security management

You don't implement all of these processes right from the start. It's too time-consuming, so, in a way, the maturity of an ITSM implementation is also tied to the age of that implementation.

Depending on the maturity of your ITSM methodology implementation, your enterprise might not want to record as formal change requests certain routine changes with clearly defined change models; your business might prefer to leave these routine changes as pure service requests instead. For example, you probably shouldn't treat the creation of a new user or the maintenance of output records for printer assignment in SAP ERP as change requests because you can control these routine changes with clearly defined procedures. Thus, there is no need to go through a more formal change request management process for these kinds of requests.

Configuration and development changes within SAP systems (whether SAP ERP, SAP NetWeaver BI, or SAP Supply Chain Management) usually move across systems (development to test to production) via transport requests. Not all changes within a system require a transport request, however. You can make a number of changes directly in the production environment: for example, changing pricing and output records. I suggest recording every SAP system change that requires a transport request as a formal change request to avoid any confusion about when you do and don't need one. A simple rule stating that every SAP system change requiring a transport request needs to submit a corresponding formal change request helps to avoid that confusion.

Although in mature ITSM implementations every non-standard change is recorded as a change request, typical SAP-centric IT shops usually limit change requests to any service requests that are not part of a formal project. As a rule, overall project management tracks any project-related changes, but non-project-related changes require a completely different vehicle for change management and implementation: hence, the change request.

Each SAP shop typically has a rule that determines which business requests to treat as projects and which ones to handle with less-formal change requests. Most often, this rule is based on the number of hours estimated to complete the request. For example, you might want to establish a rule to treat

[1] The best source for IT Service Management guidelines and publications is the IT Service Management Forum (ITSMF). You can find its USA chapter at www.itsmfusa.com. A good overview book for IT Service Management is *Foundation of IT Service Management Based on ITIL V3* from the ITSMF.

as a formal project any change request that requires more than 40 actual hours of IT involvement (elapsed time will be higher). Such a rule ensures that projects receive the attention and review needed for a larger IT effort and more substantial costs. Then, you handle all other requests through change request management instead of project management.

An example of a change request might be to create a new item category within SAP SD to process customer returns or to implement new material groups for a recently introduced product line for material master data management. (A material group in SAP ERP is typically a unique number along with the corresponding description assigned in material master data to enable consolidated reporting and price maintenance.) SAP shops can have a substantial number of change requests — typically from 20 to 100 requests per month — given the number of SAP systems in place, the breadth of functionality in use in those systems, and the maturity of each implementation.

Now that the definitions for change request and change request management are in place, let's outline the change request management process.

Change request management phases

The tasks you need to perform to manage change requests can be divided into several different phases: initiation (change request), approval, realization, business testing, and move to production. **Figure 1** represents the big-picture view of change request management.

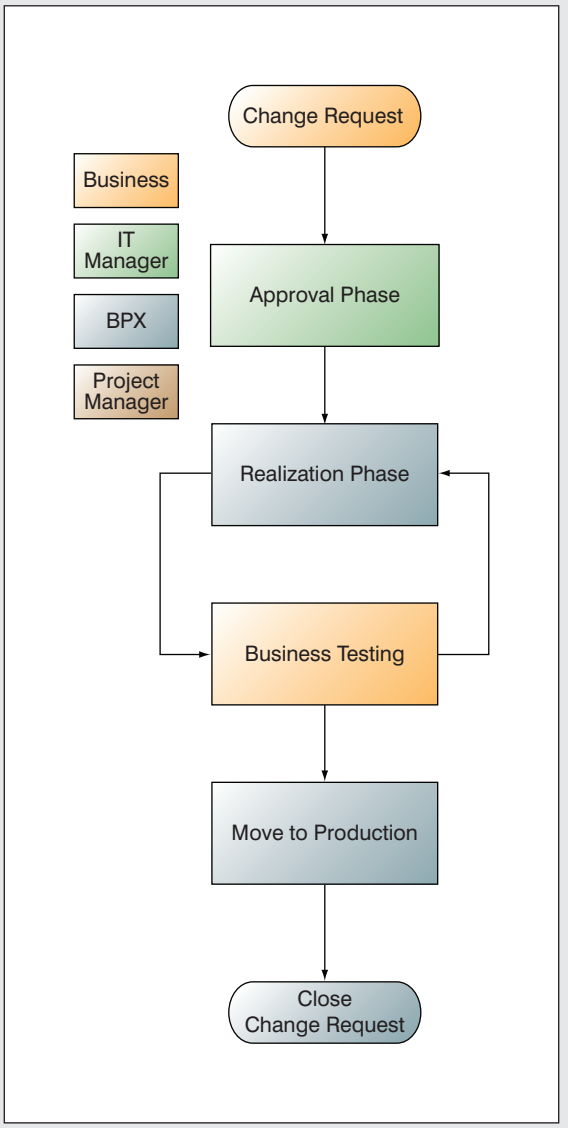


Figure 1 Managing the flow of change requests

Depending on the size of your organization, the number of people from the BUs and IT involved in initiating, managing, and governing change requests can vary dramatically:

- On the IT side, at least one IT manager should manage the change request process. This role is typically referred to as the *change manager* in an ITSM context. This person handles the approval phase and determines whether the change request achieves project status or can remain as a change request. This person is responsible for reviewing, accepting, and classifying all change requests.

  In larger organizations, ITSM also identifies the role of *change coordinator*. The person in this role assists the change manager and represents the change manager in various parts of the organization. A change coordinator also assists the change manager in gathering preliminary requirements,

assessing the complexity and priority of changes requested by the BU for which the change coordinator is responsible.

- On the BU side, each BU should appoint at least one dedicated *business liaison*, a person who is responsible for receiving, evaluating, filtering, and entering change requests into the change request database. The business liaison is also responsible for coordinating the testing of the developed change, as part of the business testing phase, and communicating back to the BPX whether the solution meets the business requirements or needs more development work.
- The BPX is responsible for the realization phase of the change, works with the Basis team to move the change to production, and closes the change request. This phase includes determining the necessary changes to configuration, assigning the change request to the appropriate development manager if custom development is needed, and sending the solution to the change request to business testing. If business testing doesn't accept the current iteration of the solution, it goes back to the BPX for a further round of development. If business testing accepts the solution, the BPX moves it to production and closes the change request.
- The maturity of IT as an internal business function is measured by its level of assimilation with the actual BUs and the amount of effort required to govern system changes. The less mature and more separated the IT function is from the rest of the organization, the more coordination and communication you need to determine on which changes IT should focus.

  Depending on how mature your IT is, it might be necessary to install a governing body, which should include both IT and business managers. This body would review and prioritize change requests on a regular basis, especially right after the initial SAP ERP implementation. According to ITSM methodology, a Change Advisory Board (CAB) usually serves this purpose. As the SAP implementation matures, only significant and complex changes should involve the CAB. In time, business liaisons and the change manager can manage and govern all change requests; the CAB only needs to deal with larger project-related items.

After the business liaison has initiated the change request, IT must approve it.

Approval phase

The change manager reviews each incoming change request, evaluates it for business value, effort, and risk, and assigns it to a BPX, as shown in **Figure 2** on the next page. Each request must follow two basic rules:

- It must be achievable within the project's time threshold (in the diagram, that's 40 actual hours).
- It must gain enterprise-wide approval, if the change affects multiple BUs.

When multiple BUs within an organization use a single SAP system, any change requests that apply to this system and affect multiple BUs need enterprise-wide approval. For example, one BU might request a change that affects the information contained in a key figure within an SAP NetWeaver BI query that many of the BUs within the organization use. In such a case, it's critical for the change manager to communicate the proposed change request to the business liaisons of all of the BUs that the change affects. An enterprise-wide change request should move into the realization phase only after it has received formal approval from all relevant BUs. As a rule, if one affected BU didn't approve the change, then the original requestor has to pursue approval of the change from the BU that opposed it.

Each change request should include an estimate of the total amount of effort required from IT to implement the change in the production environment. This approximation can be given in estimated hours or "points" (the use of points is borrowed from Agile Development; for more information on Agile Development and how it pertains to SAP environments, see Burke LaShell, "Agile development for SAP: Get into the Scrum!" *SAP Professional Journal*, May/June 2007).

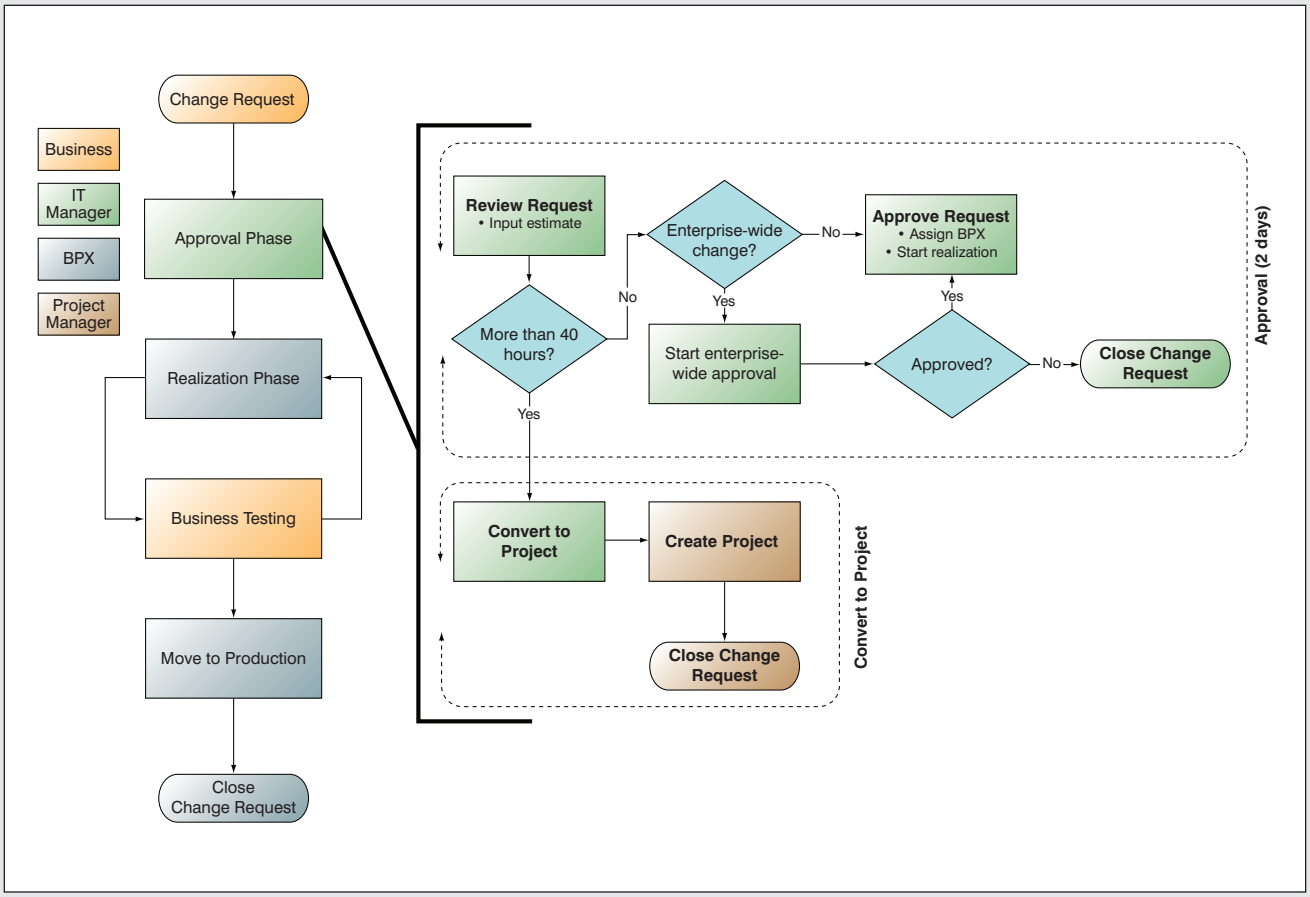


Figure 2 Approving the change request

Using points is typically the better route to take because it simplifies the estimating process. For example, 1 point might represent 8 hours of estimated IT effort. Using this example and the 40-hour project rule discussed earlier, the estimate for a change request might range from 1 to 5 points. Taking into account complete IT involvement — from BPXs to Basis developers — to move the changes across the applicable SAP instances, it's usually a good idea to estimate even the smallest change at 1 point (equated to 8 hours of effort).

Due to a potential conflict of interest, the BPX responsible for realizing the change request should not be the one who makes the estimate. In a perfect world, the change manager would be technically proficient in all areas of SAP and provide point estimates for all requests to ensure the integrity of the estimating process. Unfortunately, however, this is rarely possible, and the change manager sometimes has to ask BPXs to provide the estimate instead. In this case, you can still ensure the integrity of your estimates by comparing them to the actual number of days spent for each estimated point (represented by the average-realization-per-day KPI).

Now, let's define the initial KPIs for your change request management process. The first KPI you need to establish is the average number of days required for approval. As in all of the average duration-based KPIs that you define, it's a good idea to smooth the curve by averaging the duration of the pertinent phase for

individual change requests across a number of months. Although the approval phase is less susceptible to swings in duration, averaging it across a number of months helps deal with outliers and gives a better understanding of the phase's overall performance.

An *outlier* is a change request that takes much longer to process than a typical change request. For example, if the original requestor unexpectedly left the company, that could delay approval, requirements gathering, and even realization. The main point is that it is rarely a good idea to expect to complete every request within a particular timeframe since the IT environment is highly susceptible to change. A much better practice is to evaluate the average performance across all of the change requests in a particular time period: for example, three months.

Let's define the approval KPI as the average number of days over the previous three months that were required to complete the approval phase within the change request management process. The approval phase starts when the designated business liaison creates the change request and ends when the BPX begins the realization phase. (The rest of the KPIs are established in the section "Using KPIs" on page 35.)

Practically speaking, there are three major scenarios that can lead to a rejection from the approval phase and a premature closing of the change request:

- When the estimate is complete, if the change request requires more than 40 actual hours of work. In this scenario, the change request is rejected, closed, and forwarded to the project management office like any other project request for proper review, prioritization, and scheduling.
- In case of an enterprise-wide change, if one of the affected BUs rejects the change request.
- If the change request doesn't require an actual system change. For example, perhaps the requestor doesn't realize that the requested change is already available in production.

In all other cases, the change manager should approve the change request and then move it to the next phase of the change request management process — realization.

Realization phase

Once the change manager has approved a change request and assigned it to a particular BPX, the BPX begins to gather detailed requirements and to make the actual system changes in the development environment. At times, the BPX needs to use the development team (note the "development needed?" box in **Figure 3** on the next page). When the BPX completes the change and tests it within the development environment, he or she moves it to the quality assurance (QA) environment for the business liaison to test.

The realization phase is the most crucial phase of the overall change request management process. Not only does it typically account for the majority of the life span of any given change request, but it is also where you find most of the bottlenecks. Actual realization requires the ability to clearly understand business requirements (a task with which many IT shops struggle), as well as expert system knowledge, preferably in-house. (Note that the 2-week time estimate shown in the figure for the realization phase is elapsed time, not actual work time.)

Since change requests are typically small units of work, it's extremely inefficient to outsource this type of work to consultants. It would take significantly more time to get them up to speed than it would to make the change itself. At the same time, with proper visibility of change request KPIs, this phase typically provides the greatest opportunity for optimization and performance gain, providing higher BPX efficiency in all types of work, not just change requests. The change request management process enforces proper prioritization of work, encourages transparency of BPX performance, and rewards good work coordination and communication skills, all of which have a positive impact on the BPX's overall ability to produce results.

The first step in the realization phase is to gather detailed business requirements. It is important to

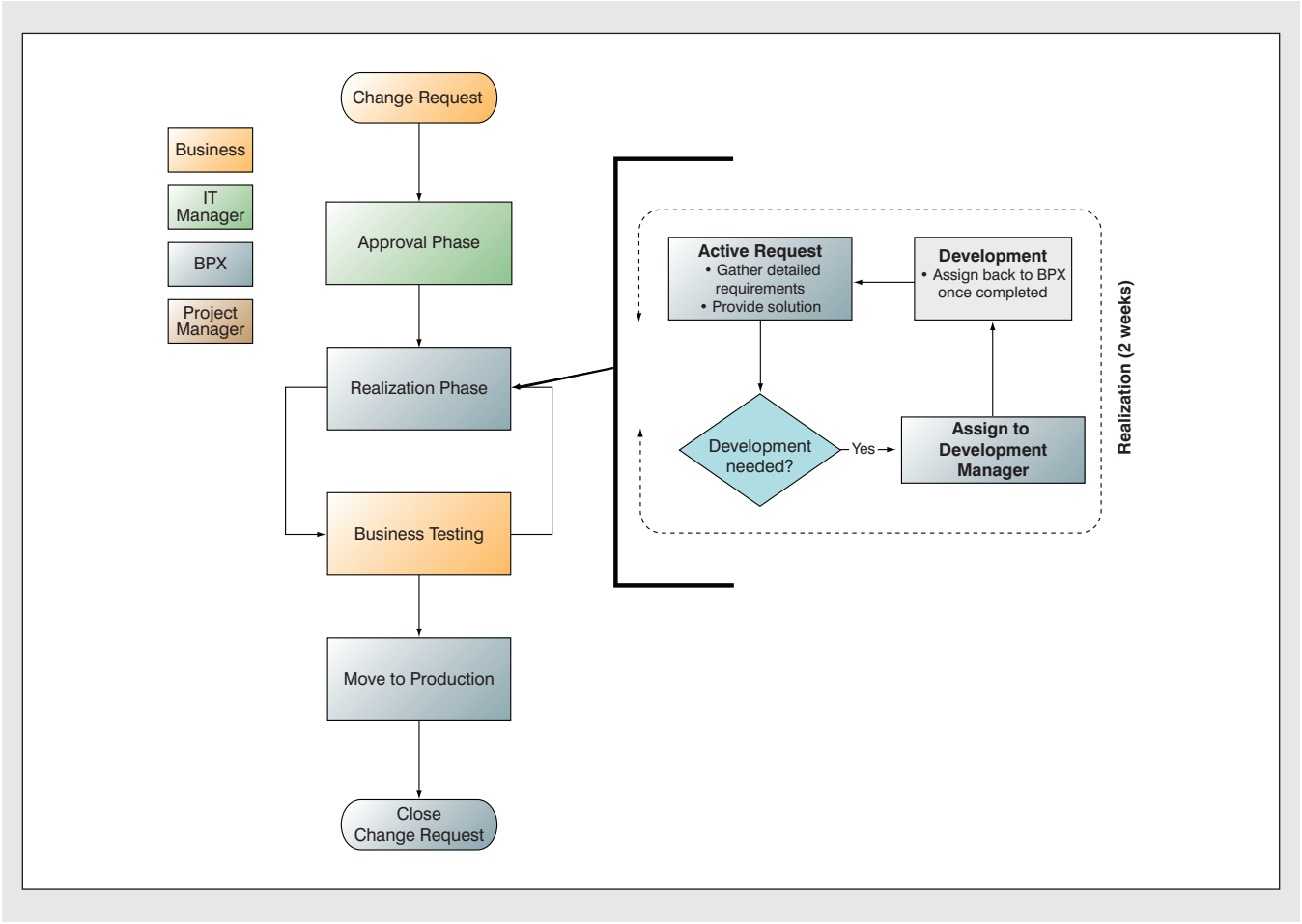


Figure 3 Realizing the change

understand not only *what* the business wants, but also *why* it needs this change and the motivation behind it. Clarity and proper granularity are key components for successfully gathering requirements. Since you're dealing with small, well-defined units of work, you should seldom need to divide a change request into smaller tasks. Therefore, you need to understand each change request well before making any actual system changes within the development environment.

You don't need to be overly formal when you record the requirements for a change request. Too much formality and red tape interferes with the overall process. Short, simple paragraphs that summarize the essence of the business requirements within a single field of the change request database should suffice. Also, it might be helpful to include attachments to the change request business requirements. For example, you might attach a mock-up of a report in a spreadsheet, or perhaps a description of the overall process in a diagram. The key is to develop a vehicle for capturing requirements that provides enough detail to meet your needs and requires the least amount of effort.

Once the actual system changes are made in the development environment, it's a good idea to track the transport request numbers associated with these changes as part of the change request. A transport request records SAP system changes and facilitates the transfer of these changes from one system to another. You specify the transferred components

in the transport request's object list. It's also important to specify the system solution within the change request, including a summary of the configuration and development changes that have been made to the development system.

When you need to develop customized functionality to fulfill a change request, you use the optional development step (**Figure 3**). For example, you might identify a need for customized ABAP development within a user exit. Given the volume of ABAP development that companies have outsourced in recent years, it's a good idea to have your BPX develop a formal technical specification document each time you need custom development. This specification document should include table and field names, as well as any detailed syntax logic that you need to implement. Then, you just attach this document to the change request within the change request database. If the development manager decides to outsource the change, all of the detailed technical requirements are readily available. The custom development step is often circular in nature, involving numerous iterations between the developer and the BPX before they reach a solution.

To measure overall BPX performance within the realization phase, you use an average-realization-per-point KPI. It represents the average number of days between when the BPX receives the change request assignment for realization and when the business liaison receives it for testing. Then, you normalize the average realization KPI for the change request estimate to 1 point, accounting for the differences in complexity among various change requests, so you can meaningfully compare the efficiency of individual BPXs.

For example, let's say BPX A typically takes 20 days to complete the realization phase and he or she is assigned change requests estimated at an average of 4 points. BPX B, on the other hand, spends 14 days to complete the realization phase, even though his or her average change request is estimated at only 2 points. When you normalize the average-realization-per-point KPI, you look at it in terms of how long it takes to complete 1 point of estimated change request effort. In this case, BPX A spends 5 days per point (20 days divided by 4 points) and BPX B spends 7 days per point (14 days divided by 2 points). So, BPX A is actually more efficient than BPX B.

As with any duration-based KPI, it's a good idea to average this indicator over at least three months to determine an individual BPX's performance or that of the overall BPX team. It's also important to mention that average realization KPIs include both the time the business takes to clarify the requirements and the time the developers spend on custom development. Therefore, the individual BPX's KPIs also reflect these figures. Although this practice may seem unfair, it encourages the BPX to focus on any inquiries he or she may have with the business liaisons, and to ensure that the BPX responds to the developers quickly and properly. In the end, the BPX is the ultimate owner of a change request and is responsible for its overall progress and completion.

Each BPX should know his or her individual average realization KPI and how it compares to the overall average realization KPI of the BPX group. This visibility not only fosters a more uniform processing of change requests, but also gives each BPX a clear indication of his or her change request performance.

The realization phase is considered to be complete when the system changes migrate to the QA environment and are assigned to a particular business liaison for the business testing phase.

Business testing phase

Every BU within the organization should have at least one dedicated business liaison who is responsible for entering the change requests and coordinating the testing for that BU. Having a dedicated business liaison enables some initial, internal business filtering before formally passing a request to the IT shop. In fact, a substantial number of requests never make it past the business liaison because a large number of requests relate to existing production functionality, an unknown area for many business users. In a perfect world, the business liaison not only passes down the requests to IT,

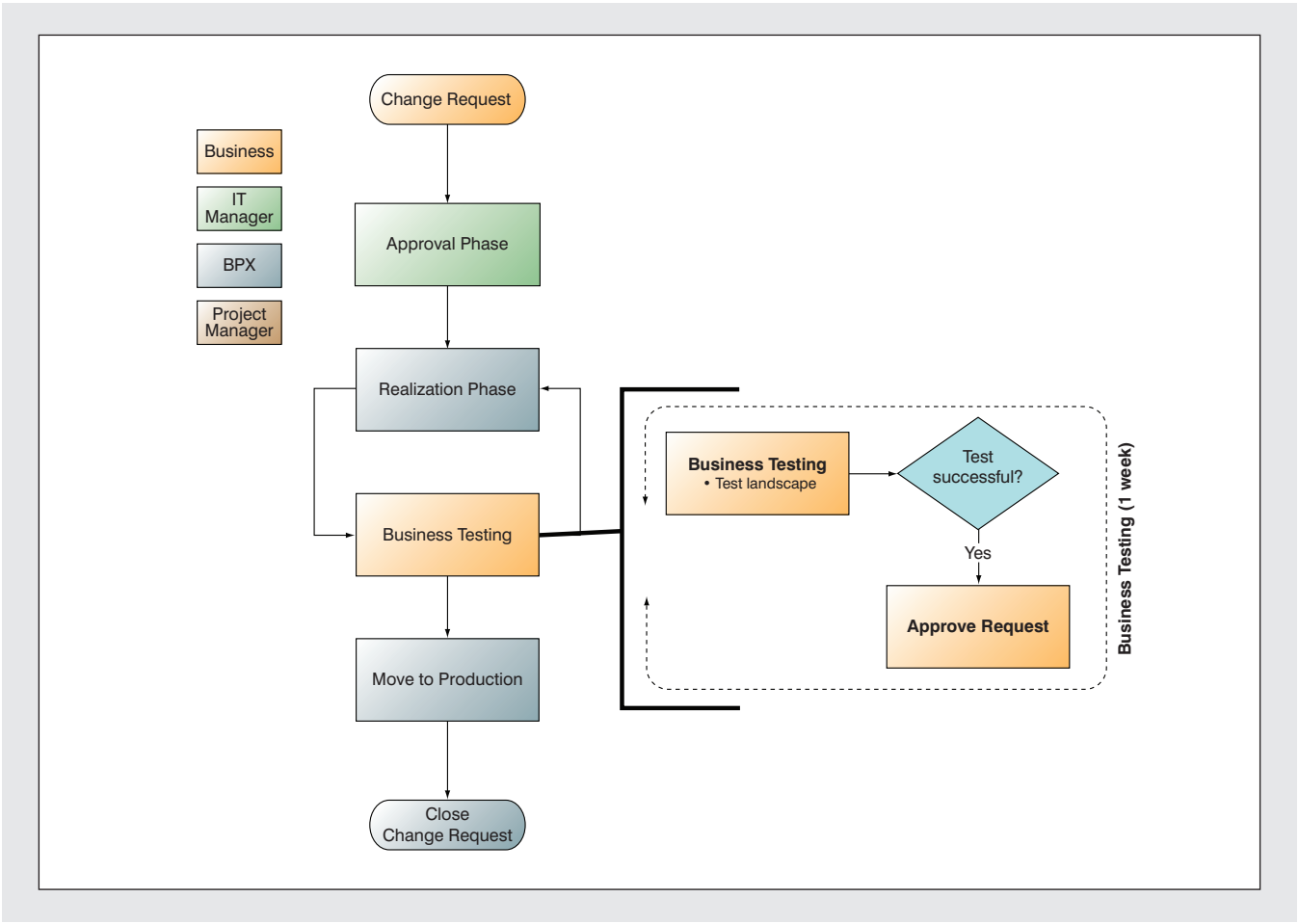


Figure 4 Testing the change

but also becomes an active player in deciding which change requests make sense and which ones to avoid.

Most of the business requirements and business testing communications should pass through the business liaison. In the ideal scenario, the business liaison is an extremely system-savvy individual, who possesses expert knowledge of the end-user interaction with SAP systems across many different modules. This expertise positions the business liaison not only to make knowledgeable decisions about the worthiness of a change request, but also to personally review and test the majority of the system changes in the QA environment, as shown in **Figure 4**, greatly decreasing the overall turnaround time needed for business testing.

Although a single iteration of business testing is sufficient for the majority of change requests, sometimes the business liaison finds that additional system changes are necessary once he or she has had a chance to look at the new functionality. This happens and it doesn't present an issue as long as it's the exception, not the rule. In this case, it's advisable to keep the change request in the business testing phase and not move it back to the realization phase because of technical limitations in the change request management system, whereas moving back and forth between phases can be problematic when calculating duration-based KPIs.

The average-business-testing KPI represents the average number of days spent between the start of the

business testing phase and the point where the change request is approved for production. The value of the average-business-testing KPI should be visible to the individual business liaisons and periodically reviewed with the CAB to ensure acceptable turn-around times. Any given business liaison should be able to easily determine the average number of days he or she has spent on business testing per change request in the previous three months, and then to compare that number to the overall business testing numbers of the entire group of business liaisons. Note that the 1-week time estimate (**Figure 4**) for the testing phase is elapsed time for the change request, not actual work time.

Once the solution provided in the QA environment fully meets the business requirements and is tested successfully, the business liaison formally approves the change request for migration to the production environment. Then, the BPX coordinates the migration of the system changes to the production system with the Basis team through a standard, predefined change control process. Once the migration date is determined, the BPX should communicate it to the business liaison so that, if necessary, the business liaison can distribute the proper documentation to key business users ahead of time.

Using KPIs

So far, I've discussed three main KPIs: the average numbers of days for approval, realization (normalized to 1 point), and business testing. There are a number of other important KPIs that you should continuously measure and review against their target values. (It's important to establish target KPI values and implement these indicators as part of internal IT operations.) The visibility of the KPIs and their overall acceptance by the BPX team ultimately drive the proactive management of BPX productivity.

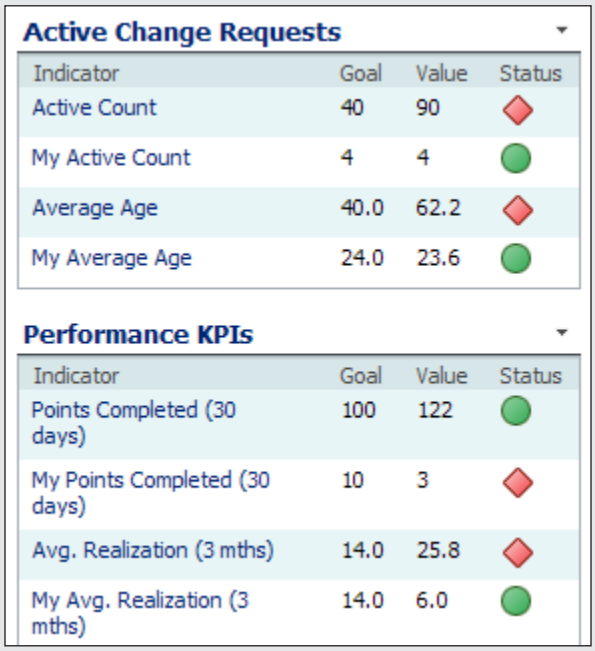
Before I go into the details of other indicators that might be interesting to the overall change request management process, let's take a look at a high-level plan for putting KPIs into practice in your IT shop.

Implementing KPIs

Using the examples provided in this article, the BPX team determines some KPIs that are crucial to the change request management process. It is vital to focus on those indicators that bring the most value, and to keep the number of KPIs to a minimum. Having too many KPIs can contribute not only to confusion, but also to a lack of focus and buy-in from the BPXs. You should aim for no more than six or seven KPIs during the initial implementation of the change request management process. The change managers and BPXs together should compile the list of indicators to gain clarity and commitment from the employees whose performance will ultimately be measured against these KPIs.

For some indicators, it's fairly straightforward to pick a target value. For example, setting the target value for average days for approval at 2 or 3 days is fairly intuitive. The approval phase should have very little variance in average duration, considering the simplicity of the steps involved. You can review and approve most change requests within 24 hours with a few that require more time for enterprise-wide approval (which does typically take a while to complete). All in all, a duration of 2 days average is usually easy to achieve.

Other indicators, such as the average realization per point, require a slightly more scientific approach to setting the target value. Let's take a hypothetical example and try to derive a reasonable estimate. If roughly 25% of BPX time should be spent addressing change requests and a typical BPX has an 8-hour workday, this comes to approximately 2 hours per day, 10 hours per week, and 40 hours per month. Let's also assume that you anticipate roughly three change requests per month to be entered into the change request database for each BPX. Furthermore, an average request in this hypothetical company is estimated to be 2 points, or 16 hours, worth of realization effort. In worst-case scenarios, all of these requests are input on the same day and a BPX begins to work



Active Change Requests

Indicator	Goal	Value	Status
Active Count	40	90	
My Active Count	4	4	
Average Age	40.0	62.2	
My Average Age	24.0	23.6	

Performance KPIs

Indicator	Goal	Value	Status
Points Completed (30 days)	100	122	
My Points Completed (30 days)	10	3	
Avg. Realization (3 mths)	14.0	25.8	
My Avg. Realization (3 mths)	14.0	6.0	

Figure 5 Example of change request KPIs on a company Web page

on them immediately, spending roughly 2 hours (25%) per day on change request-related work.

If a BPX works on all three change requests sequentially, one at a time, he or she will complete the first change request in 8 days elapsed time (16 hours divided by 2 hours per day), the second one in 16 days elapsed time, and the third one in 24 days elapsed time. Taking those three numbers (8, 16, and 24 days) together produces an average realization of 16 days elapsed time per change request. This represents an average of 8 days per point, since you estimated that a typical change request would equate to 2 points of effort. Interestingly enough, actual typical values for average change request realization usually fall between 6 and 10 days elapsed time per estimated point.

I cannot stress enough the importance of making KPIs easily accessible and visible at both individual BPX and team levels. Higher visibility fosters responsiveness and self-management in change request productivity. Consider building an internal company Web page that each BPX can access to compare his or her performance to the overall performance of the team, as well as to see how well the actual KPI values compare to their target values. **Figure 5** provides an example of a number of change request KPIs made visible on a company Web page. Another option would be to broadcast these values via email to both IT management and the BPXs. Finally, make sure that business liaisons also have clear visibility of *their* indicators, including the average number of days for business testing, which is calculated by looking at the average duration required to complete the business testing phase in the last three months.

Now that the KPIs have been determined and made visible to the entire team, make sure you take responsibility for how well you can meet the predetermined objectives. Consider reviewing the KPIs during your ongoing team and 1-on-1 meetings. You can also make the change request KPIs part of your employee appraisal process, tying performance within those indicators to a formal evaluation of the productivity of the BPX.

Understanding workflow

Consider using the following KPIs to help evaluate the makeup of any currently outstanding change requests. These indicators don't specifically represent the performance of the BPX team; instead, they help you to understand how much work is in the pipeline, as well as to see how well the IT team is keeping up with the incoming flow of change requests:

- **Number of active change requests:** This is a straightforward count of the change requests that have not yet been formally closed. An alternative — and arguably more sophisticated — measure could be the number of active points, which would account for any fluctuations in the complexity of outstanding requests.
- **Average age of an active request:** This indicator represents the average number of days that an active change request has been in existence. The purpose of this measurement is to identify how well the BPX team keeps up with demand. Assuming a good average realization is in place, poor performance on this indicator can signal a shortage of BPX resources or possibly the need to dedicate more of the BPX team's time to change request-related work.

The final two KPIs give you an idea of the most recent inflow and outflow of change requests. This indicator helps to pinpoint whether an increase in the number of active change requests is due to higher demand or inferior performance:

- **Number of points requested in the last 30 days:** This indicator measures the total number of estimated points or hours of effort spent on change requests created in the last 30 days. This measurement helps assess the recent volume of change requests that the business has requested.
- **Number of points completed in the last 30 days:** Similar to the previous indicator, this measurement identifies the total number of estimated points included in the change requests closed in the last 30 days. The main purpose of this KPI is to pinpoint whether the group is maintaining an acceptable level of change request closure.

Technical layout and implementation

There are a number of ways to physically implement the complete change request management process in an IT shop, as shown in **Figure 6** on the next page. Typically, you want to maximize the use of your existing software and hardware assets, and choose the route that yields the lowest total cost of ownership (TCO) in the long run.

Before I dive into the specifics of which fields you might want to include in your change request database, let's look at various high-level alternatives to implementing a change request management process from a software perspective:

- Employ service desk software tailored toward the ITSM methodology. Although ITSM is still fairly young, a number of software vendors offer complete service and change management packages.
- Use enterprise portal software (e.g., SAP NetWeaver Portal, Microsoft SharePoint) tailored toward easy creation of lists and KPIs.
- Design a custom Web-based application. Although this can turn out to be the most costly alternative, it provides the most flexibility.
- For the least complicated implementations, use a simple database with a corresponding form (or even a centrally maintained spreadsheet). Although this alternative requires the least amount of effort for implementation, it is extremely limited in terms of functionality and the ability to scale to medium- and large-sized IT shops.

Logically, you should divide the change request database into at least two main areas: the change request listing and the workflow. The change request listing provides the detailed information associated with any particular change request, as shown alphabetically in **Figure 7** on page 39. You might want to track any or all of these items.

The workflow table should track a minimum of information and its items should represent the stages

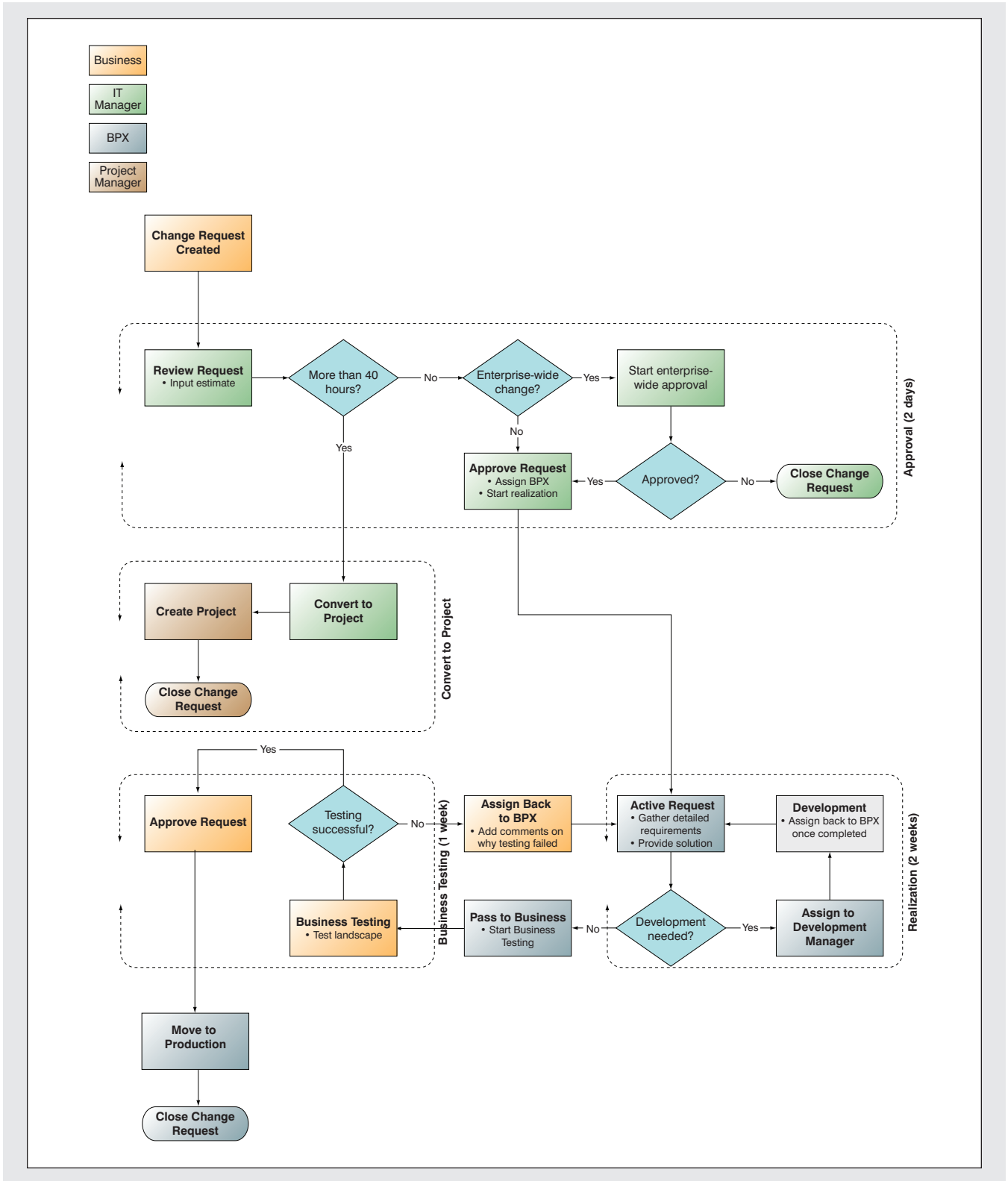


Figure 6 Managing the entire change process

Change request field	Definition
Active age	Number of days for which the change request has been active
Assigned to	Person to whom the change request is currently assigned
BPX	Ultimate owner of the change request, who is responsible for the overall progress and completion of the request
Business liaison	Business representative responsible for clarifying the business requirements
Business process	Designated business process associated with the change request; for example, SAP SD, SAP Human Capital Management (SAP HCM), SAP Financial Accounting
Business unit	Business entity responsible for submitting the change request
Change request number	Number that uniquely identifies the change request
Change request status	Status of the change request indicating where in the overall process the request currently stands, as well as control of the completion of various workflows associated with the change request (e.g., approval, realization, business testing)
Changed by	Name of the person responsible for the last update to the change request
Changed on	Date on which the change request was last updated
Comments	Ongoing comments area that allows the various people involved in the change request to communicate with one another and record important information
Created by	Name of the person who created the change request
Created on	Date on which the change request was created
Detailed description	Business liaison's detailed description of the business requirements
Detailed requirements	BPX's detailed business requirements
Developer	Optional role, representing the developer assigned to the change request
Due date	Optional due date to help prioritize outstanding change requests
Point estimate	Estimated number of points associated with the effort that is required to complete the change request
Priority	Overall importance of the change request — high, medium, low — as the business liaison requests (if too many high-priority change requests are active, the IT/change manager works with the various business liaisons to reevaluate the priorities)
Solution	Summary of the solution used to fulfill the business requirement
System	System associated with the change request (e.g., SAP ERP, SAP NetWeaver BI, EDI)
Title	Short description of the change request
Transport	SAP transport request numbers associated with the change request

Figure 7 Possible database fields to track in change request listing

associated with each change request. The three main duration-based KPIs that track average number of days for completion of approval, realization, and business testing phases are all based on information that is found in the workflow table (see **Figure 8** on the next page).

Workflow item	Definition
Assigned to	Person to whom workflow item is assigned
Change request number	Change request number associated with the workflow item
Closed on	Completion date for workflow item
Created on	Creation date for workflow item
Description	Description of workflow item
Status	Status of workflow item — typically limited to active and closed
Title	Title including type of workflow associated with item and possibly title of corresponding change request
Workflow name	Name identifying type of workflow associated with item

Figure 8 Potential items for workflow table

Conclusion

Eventually, all SAP ERP shops incorporate some sort of a listing, either via spreadsheet or Web page, where they maintain requests for SAP system changes. Although this method provides a formal mechanism for entering and resolving such requests, it often lacks the granularity necessary to actually manage the process. When left with an inability to measure performance and determine the sources of bottlenecks, IT shops see their change request backlog endlessly increase while SAP analysts struggle to keep up.

The change request management methodology described in this article focuses on:

- Defining a robust, yet simple, process for managing SAP change requests
- Establishing the critical areas for measuring performance within that process
- Deriving a number of KPIs with which to measure group and individual BPX productivity

This methodology provides a clear set of standards for managing SAP change requests, one that results in more fluent and robust internal IT processes. Most importantly, a concrete set of KPIs allows change managers to evaluate the performance of their BPX group, as well as boost productivity among the individual BPXs. Ultimately, such an approach results in superior BPX performance, while the ability to measure key areas within the change request process provides the ability to truly manage it.

