Planning for the ongoing support and maintenance that accompany implementation of new enterprise resource planning systems may be more essential to realizing benefit from a technology investment than choosing the product with the most features.

Implementation Never Ends! The Postimplementation Organizational and Operational Implications of ERP

Philip J. Goldstein

You can finally exhale. The implementation is over. The consultants have all gone home. The implementation team room has been converted back to its original use (such as a conference room or a supply trailer). The end users have been trained, all the interfaces are working, and you watch proudly as the first set of students is admitted using the new system. The hard work is over. The staff can go back to their old jobs. Right? Not so fast. In many ways, the important work has just begun.

Operating, maintaining, and upgrading a modern student information system (SIS) is not easy. Institutions are finding it challenging to absorb this work into existing organizations and staffing models. Gonzaga University reported in an article in EDUCAUSE Quarterly that more than half the costs of owning an enterprise resource planning (ERP) system stem from indirect costs such as the staff time required to operate and maintain the system (Powel and Barry, 2005). These costs far outweigh the initial outlay of funds to purchase hardware and software. However, properly executing the ongoing work of running a new system is critical to realizing the benefits that led to its adoption in the first place.

Should we be surprised by the ongoing workload to maintain a modern SIS? Probably not. After all, many institutions have moved to implement new systems because these systems offered the promise of capabilities that would grow and evolve as the institution's needs change. What has come as a surprise to most institutions is the level of effort it takes not only to
perform major upgrades but to keep pace with minor system changes as well after the implementation is over. In fact, many believe it takes two to three years after the implementation to understand a new system, fine-tune it, and realize its benefits.

A significant portion of the system’s maintenance workload falls to the information technology (IT) department. In fact, IT departments typically experience an increase in staffing costs after the implementation of new finance, human resource, or student information systems. They typically must employ either more backfill staff to handle the ongoing maintenance of new technology or more technically skilled staff with higher compensation costs.

However, technology staff cannot maintain new systems alone. Just as the initial implementation team required a user-driven, collaborative effort with IT, so too does ongoing systems maintenance. This need has also caught most institutions by surprise. In fact, user departments typically see net increases in their staffing costs. Certainly efficiencies in processes may enable the organization to work with fewer people. However, this is often offset by the need for new kinds of positions to help manage the new system and to take advantage of its more extensive capabilities. As with technology, these new positions typically require greater skill levels and command higher compensation. This effect is not unique to SISs. The same postimplementation impacts are seen in other major systems, including finance and human resources.

To understand the phenomenon of continual implementation, it is useful to look back at where we have come from. Most institutions’ legacy SISs were static. Changes were limited to annual financial aid regulatory updates or the addition of new reports. Occasionally the IT group would have the resources to add some new data fields or to create a new feature. Users participated in the governance of the new system and the occasional testing of a new feature but did not have to play a significant role in its day-to-day operation.

Today’s SISs are much different. Vendors release updates, upgrades, and patches at a rapid pace. New features seem to become available every twelve to eighteen months (or sooner). Like a high-performance car that requires constant tuning, modern systems require frequent fine-tuning to maintain their alignment with evolving business processes and user needs. New systems also capture more information and make available better tools for users to do their own analyses. No longer do they need to wait for IT to develop a new report, but this change creates a new and different workload for some organizations.

The phenomenon of continuous implementation has had four broad impacts on institutions. It has

- Created a need for more active and integrated systems governance
- Necessitated a continual commitment to operations improvement
- Altered the traditional boundaries between IT and user areas in systems management
- Given rise to the need for a new user skill set
This chapter describes these impacts and identifies successful strategies that institutions can employ to manage them.

**Governance**

Most institutions have always had some form of SIS governance committee. During implementation of the new ERP system, most institutions form governance groups to set priorities, coordinate cross-functional issues, and make project decisions. Postimplementation operations require that these governance groups remain intact, active, and empowered.

**The Need for Governance.** The need for continuous, active governance is driven by three factors. Decisions must be made on a routine basis to allocate scarce resources, operations must be coordinated across disparate areas, and upgrades must be managed as if they were mini-implementation projects. Each of these factors is explored in detail in the paragraphs that follow.

The implementation of an SIS rarely delivers all the functionality that users ultimately want or need. The need to limit the project scope to contain risk and accommodate budget constraints is one reason. Another reason is the need to work with the system for a period to really understand its capabilities. This process enables an institution to make better decisions about which pieces of functionality it really needs. It is not unusual to see institutions with a backlog of work to be done after the initial implementation. Typically this work includes development of additional reports, activation of additional aspects of the system, and development or implementation of complementary technologies that work with the SIS, such as a scheduling module.

In all of these examples, the human resources required to create the additional capabilities are the same functional and technical staff who are occupied with operating the system. Most institutions have precious little discretionary capacity to do project work. A governance mechanism is needed to allocate this scarce resource and to establish priorities for enhancements and extensions of capability.

Coordinating operations across and among functions is another challenge that requires good postimplementation governance. In setting out to replace legacy SISs, higher education institutions have sought solutions that were more integrated than the old systems. Integration among the components of the SIS and, to a lesser extent, between the SIS and the human resource or finance system is viewed as a major driver of new implementations. Modern SISs have delivered integration. However, the byproduct of more integrated systems is an increased need to coordinate operations. Departments must share data fields and agree to common definitions of data elements. A simple example of this is the need to use address fields in a consistent way among offices. Integrated systems have accompanied a greater integration of business processes.

Modern SISs have enabled such process integration as one-stop student services
and Web self-services that tend to blur the boundaries that historically mapped business processes to individual functional departments. This too requires a greater level of coordination and governance than the previous systems required. What were once isolated changes that could be made independently by one function must now be carefully considered, tested, and vetted to make sure they work for all users.

The need for coordination does not stop at the doors of the student services offices. If the institution has also implemented a new finance, human resource, or advancement system, then there is a far greater degree of integration between those systems and the SIS as well. It is likely that all systems share basic biodemographic information about students and faculty. For example, an individual’s primary and secondary address is shared across all components of the system. The student accounts system is likely to be tightly coupled to the financial system, and both systems are likely to hold many data fields and business processes in common. Likewise, there may be a greater coupling between the records module and the advancement system to track students’ interests and activities as they transition to alumni status. The human resource system may share information and business processes with the SIS to manage staff who are also students and students who are also employees. This integration creates the need for more joint decision making. Just as the records office cannot change how it uses the system without considering the potential impact on other student services offices, the SIS governance group cannot make changes to the SIS without considering the impact of those changes on other enterprise systems. For example, the naming conventions of academic departments must be kept consistent between the records module and the human resource system, and the use of an address field by the admissions office may need to be coordinated with the users of the advancement system. Just as during the implementation, these ongoing adjustments need a place to be vetted, analyzed, and decided. Integrated systems require integrated governance.

Finally, the ongoing cycle of upgrades required to maintain a modern SIS also has driven the need for governance. Upgrades are in fact mini-implementation projects. As in a full implementation, priorities need to be set, policies and business practice changes need to be approved, and strong executive sponsorship is required to sustain institutional commitment. Fortunately, the issues are typically not as complex or numerous during an upgrade as during a full implementation. However, they are significant enough that they cannot be adequately addressed by individual managers acting only in their own unit’s interest. Instead, a broader governance mechanism is required.

**Effective Governance.** To meet the challenges just described, successful institutions often elect to retain permanently the governance groups established for implementation. In SISs this typically requires at least a steering committee that represents the major stakeholders in student services. Representation is drawn from the major student service functions, including
records, student accounts, financial aid, admissions, and advising. In larger institutions, the group may also include academic administrators from various schools or academic departments. In addition, there should be representation from the IT organization.

The governance of the SIS must also link to broader IT governance and priority setting at the institution. As discussed previously, the integrated nature of modern systems and the scarcity of resources require governance to span major systems as well as individual functions. In this regard, the SIS governance group may actually be a subcommittee of a larger administrative technology group or an overall IT committee.

The SIS governance body fills the role between the high-level strategy setting performed by an administrative or IT steering committee and the day-to-day decision making performed by functional managers in student services and IT managers responsible for the support of the SIS. The top-level IT governance function (IT steering committee) must address such questions as the following:

- What criteria should be used to select discretionary technology projects?
- What contractual or regulatory-driven maintenance must be performed on the administrative systems?
- What level of resources can be devoted to discretionary projects overall?
- What level of resources (people and dollars) can be allocated to the SIS this year?

Similarly, functional managers provide a layer of governance that ensures that day-to-day management of the SIS is performed effectively. Functional managers are responsible for decision making regarding the following:

- Day-to-day operations of the system
- Execution of discretionary technology projects
- Altering system configurations that affect single functional areas
- Approving the application of patches of limited scope

Between these two layers, a cross-functional SIS governance group is required to set the future strategy for the SIS and settle tactical questions that require coordination among multiple functions. At a strategic level, the SIS governance body should be empowered to set priorities for the implementation of new modules, to approve the development of additional modifications to the system (or the development of small programs to work with the SIS), and to request the purchase and integration of additional third-party products. The members of the governance body should serve as advocates to higher-level IT governance groups to request and obtain resources for the SIS. Finally, they should champion the effective use of the
SIS (and other support technologies) and lead efforts to extract maximum benefits from the investment in technology.

Tactically, the SIS governance group must coordinate decision making that crosses functional boundaries. The group should be empowered to approve changes to the configuration of the software that affect multiple functional areas. Maintaining the integrity of the data in the SIS is another important role for the SIS governance group. The group as a whole or through a subcommittee should be charged with approving any changes to the use of any system data fields that are shared across functional areas. Similarly, they should represent the system's users in broader discussions regarding the use of data fields that are shared by the SIS and other administrative applications.

Resource and schedule coordination is the final aspect of SIS governance. Maintaining a modern SIS requires the evaluation and implementation of frequent product upgrades. Some are limited in scope and affect mainly the technical support staff. Others can introduce new product functionality or significant changes to existing functionality. It is the role of the governance group to coordinate the upgrade process. Critical decisions include the following:

- When within the vendor-specified timeframe should the upgrade be performed?
- How does the upgrade need to be coordinated with other major activities that occupy the time of the student service professionals?
- If there is optional new functionality available in the upgrade, should it be deployed?
- Which staff will work on the upgrade and how will their time be back-filled in their home department?

The upgrade process is also a time to evaluate whether modifications that have been made to the system by the institution are still necessary. As institutions grow more familiar with the workings of a system, they can discover that modifications they previously thought were necessary can be supported by the base functionality in the system or accommodated through business process changes. The governance group must be accountable for making these decisions.

**Multicampus Governance.** Several institutions share SISs across multiple campuses. For example, the campuses within large, multicampus systems such as the State University of New York (SUNY) system, the California State (Cal State) system, and the University of North Carolina system have collaborated on the implementation or operation of their SISs. Similarly, some small, private colleges share an SIS with one another or with a larger university. For example, Drexel University operates an SIS for itself and Cabrini College.
The models of collaboration vary. All of the collaborators within a group use the same software, but not necessarily in identical ways. Some groups use the same software across multiple institutions or across campuses of the same institution. Some maintain separate configurations of software for each campus but operate them out of a shared IT organization. In general, the more tightly coupled the solution is, the greater the need for cross-institutional decision making.

These multicampus arrangements require the same type of governance group as that required by a single institution. In addition, institutions engaged in such collaboration may require discrete governance groups for each major student service function that spans campuses. For example, the overall student services group could have reporting to it separate subcommittees for financial aid, records, student accounts, admissions, and advising. These subcommittees would coordinate use of the SIS for their domain (that is, the financial aid module would be governed by the financial aid subcommittee) across campuses. Overall coordination would be provided by a single SIS steering committee with representation from each campus.

Alternatively, where each campus in the collaboration retains local autonomy and control of the SIS, the governance would more closely resemble that of a single campus implementation. This would be the case for a scenario such as the Cal State SIS. The Cal State system contracts with a single entity to host all of the campuses of the system. However, individual campuses control their own instance or copy of the software, which enables them to configure the setup uniquely to fit some of the specific needs of their campus. However, even in this less-integrated arrangement, there is a need for overall coordination of the SISs to make decisions about the timing of upgrades or the implementation of additional modifications.

Continual Operations Improvement

During implementation, project teams become well-versed in the traditional work phases of design, build, test, and implement. Significant effort is expended to design business processes that work well with the new technology, to build or configure the system to reflect the business practices and policies of the institution, and to test that configuration prior to implementation. This cycle does not end with the implementation. It is repeated even after the major implementation activities are completed. The need to fine-tune processes and the configuration of technology continually is another reason that implementation never ends.

Small Change, Big Impacts. Embedding technology in student service processes has enabled significant innovation and gains in efficiency. The last ten years have seen most institutions significantly increase the amount of technology they use in delivering student services. A consequence of this increased reliance on technology is that there is more to maintain. Each time
an institution needs to change a business practice or policy, it must research how best to reflect that change in the configuration of its SIS. For example, the decision to add a new course or major will require the configuration of additional rules in the degree audit and records module of the new system. These configuration changes need to be researched, modeled, and tested before they can be put into effect.

The change just described is fairly contained and can likely be addressed by one or two individuals. More significant changes require more extensive effort. For example, an institution could decide to offer students the opportunity to preregister for classes a full semester early. If this practice was not anticipated during the implementation, the system must be reconfigured to adapt to a new set of procedures. The impact of the change could ripple through many parts of the system. For these more significant changes, an institution is in effect performing a mini-implementation project and must go back into the same mode of working they followed during their initial implementation effort. A team must be formed to research the following questions:

- Which aspects of the system are affected by the change?
- What tables or reports need to be modified?
- How does the configuration need to change to reflect the change in process?

Just as during the initial implementation, the changes that are being made need to be thoroughly tested before the institution can begin to use them.

**Upgrades.** Upgrades are another event that triggers the need to fine-tune processes and system configuration. An upgrade can bring with it new functionality that needs to be researched, configured, and tested. Sometimes an upgrade can also alter existing functionality. This requires the institution to revise its processes or alter some of its configuration choices. A major upgrade that alters many aspects of the system is also an opportunity to review previous configuration choices to make sure they are as effective as possible. In this regard, an upgrade can become an opportunity to take stock of how well the institution’s business processes are functioning with the new system and to make changes that improve upon the set of choices made during the initial implementation. Vendors provide significant upgrades as frequently as every eighteen months. Smaller upgrades come out even more frequently. Therefore, the workload associated with upgrades is a significant component of continuous implementation.

The scope of change involved in an upgrade is often broad enough that the institution feels as though it is again implementing the system. Upgrades require the institution to take the time to research and understand how the new version of the software differs from the prior version. Vendors typically make available documentation that helps institutions perform this analysis.
Teams of functional and technical staff must spend time evaluating how the changes in the new version will affect the institution. Are there new ways to use the system to meet the needs of the institution? Are there completely new components available in the new version of the system that the institution may want to use? How will procedures need to change? Will users require retraining? These are the dimensions of the upgrade that must be evaluated and planned. The answer to each question will drive a set of tasks that must be completed as part of the upgrade process.

The changes involved during an upgrade are significant and must be tested prior to use. Even if all the software delivered by the vendor is free of defects, testing may reveal issues that have been created by the way the institution has configured the system. Thorough testing is certainly required to make sure that any new features available in the upgraded version are working properly. Finally, thorough testing will ensure that the new version of the system can still interface effectively with other software on campus.

Upgrades are also reminiscent of initial implementation work because they come with firm deadlines. These deadlines are driven by two factors. First, software vendors provide defined timeframes in which an upgrade must be accomplished. Failure to comply with these deadlines could void the software maintenance agreement with the vendor. Second, because upgrades consume so much staff time, they must be planned around day-to-day operations. SISs are especially challenging because there are few true slow periods throughout the year and often they are not shared by all the student services offices.

Finally, vendors release patches to their software on a monthly or weekly basis. Patches have a far narrower scope than an upgrade. Typically they affect just a portion of the system and are provided to address a deficiency found in the software code or to upgrade the technology. While much of the work of applying patches falls to the IT staff, some support from functional staff may be required. Functional staff may need to spend time evaluating the impact of the patch and advise IT whether it needs to be applied, or they may be called on to test the patch prior to its implementation.

**Continuous Improvement.** The workload of maintaining processes and system configuration is continual and significant. The effort to keep pace with vendor upgrades alone can stretch institutional resources thin. Though this work is difficult, its benefits are significant in that it provides institutions with the opportunity to expand and fine-tune how they are using technology. Few institutions can get all aspects of their implementation right the first time. A 2002 study of ERP implementations in higher education by the EDUCAUSE Center for Applied Research (ECAR) found that institutions experience a one- to two-year decline in productivity after their initial implementation (Kvavik and Katz, 2002). As staff members learn the new technology, the institution is positioned to see gains in productivity and satisfaction that surpass the preimplementation levels.
So, there is a payback to continuous implementation. Institutions that can retain their focus on continually fine-tuning their software and processes can achieve higher levels of effectiveness. The student services area appears to be realizing some of the benefits of this aspect of continual implementation. In 2005, ECAR did a follow-up of its 2002 ERP study to look at institutional satisfaction with the state of performance of business processes. The study, *Good Enough! IT Investment and Business Process Performance*, found that the institutions surveyed were most satisfied with their student service processes (Kvavik and Goldstein, 2005). However, achieving higher levels of process performance requires not only sustained effort but also a different kind of skill set among functional staff. This implication for staffing is explored in greater detail in the last section of this chapter.

**Whose System Is It Anyway?**

Another postimplementation impact that institutions experience is change in the division of responsibilities for system support between the IT department and other functional departments. As the preceding sections have described, the new world of continual implementation places significant responsibility on functional departments to participate in system management and support. This change is not unique to SISs; it is a byproduct of modern ERP systems. With such systems, much control is placed in the hands of nontechnical staff to access information and control the way the system operates.

**Historical Division of Responsibility.** Compared to modern ERP systems, the older legacy SISs changed little once they were implemented. Annual regulatory upgrades would be made to the system to reflect changes in financial aid policy, or an institution might make one or two modifications to the system. In the past, legacy systems were rarely controlled by tables or configuration values that could be easily changed by an end user; the knowledge and skills of a programmer were required to make even simple changes. Similarly, unless the institution had deployed a data warehouse or user-friendly reporting tools, extracting data from legacy SISs also required the skills of an IT professional.

As a result, users depended on the IT organization to maintain and enhance the SIS. It was not unusual in many legacy environments for IT staff to be responsible for programming reports, developing modifications, testing, and regulatory upgrades. If the institution introduced significant changes to its policies and procedures, an IT staff member would research how those changes rippled through the system. Users engaged in prioritizing needs for reports and modifications and helped to specify the changes they needed. A user with some technical skill might also develop some reports. However, at most institutions the responsibility for maintaining and operating the SIS was assigned to the IT organization.
**Change Drivers.** With modern ERP systems, the historical division of responsibilities is changing. Responsibility for maintaining the SIS no longer rests solely with the IT organization. Several factors are driving this change. First, the technology itself has changed. For example, tables largely control the operating systems that drive today’s SISs. Once they are trained in the meaning of these setup tables and codes, non-IT professionals can understand how these tables and codes are used to control how the system functions. The implication is that an IT professional in a central IT department is no longer needed to modify a program whenever the institution needs to add a major, change an approval authority, or add an academic department. In fact, most changes to the system to support policy or process change can be made by a non-IT professional. Another technological change that has altered roles and responsibilities is related to reporting. Many institutions invested in new SISs to improve access to information. While the capability of SISs to improve access to information has proved to be somewhat limited, other complementary technologies have become available that make the extraction and manipulation of data much less complex than it used to be. As institutions have embraced data marts and warehouses, they have found that users of these data configurations can access information without assistance from an IT professional. At many institutions, the central IT department is now involved only in developing the most complex reports.

The second change driver has been the workload in central IT departments. Today, the resources of most IT departments are consumed by the effort to maintain existing technologies. Just as functional areas have been surprised by the ongoing workload of maintaining new systems, IT departments have been consumed by the effort required to maintain those systems technically. At the same time, the proliferation of technology use (driven in part by developments such as self-service for students) has given IT a new set of responsibilities in end-user support and IT security. A recent ECAR study of the state of IT funding found that 70 to 80 percent of the resources of most IT organizations are committed to maintaining existing technology (Goldstein, 2004). Very little time and money are available for enhancements or extensions of service. As a result, IT organizations have needed end-user departments to take on a larger role in system enhancement and maintenance.

The third change driver has been growth in the breadth of capability of SISs. The systems have become so robust that it is nearly impossible for an IT professional to learn the ins and outs of a single module, let alone of the whole system. Developing an understanding of the capabilities of a modern SIS requires the specialized knowledge and focus of a functional staff member in a student services office.

**New Division of Roles and Responsibilities.** As a result of these drivers, the responsibility for managing the SIS (or any ERP application) has shifted from being an IT-centered responsibility to a shared responsibility. Functional offices now play a significant role in the day-to-day management
Table 4.1. Division of Roles and Responsibilities for SIS Management

<table>
<thead>
<tr>
<th>Central IT</th>
<th>Functional Offices</th>
<th>Shared Responsibility</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maintain hardware, operating systems, and databases</td>
<td>Maintain configuration of the software</td>
<td>Research new releases of the software</td>
</tr>
<tr>
<td>Develop modifications, custom applications, and interfaces to and from the student system</td>
<td>Run batch processes</td>
<td>Design new functionality</td>
</tr>
<tr>
<td>Migrate patches, upgrades, and changes into production (live) environment</td>
<td>Develop reports and queries</td>
<td>Troubleshoot complex issues</td>
</tr>
<tr>
<td>Troubleshoot problems that stem from hardware, database, or interface issues</td>
<td>Troubleshoot issues and errors encountered by users</td>
<td>Perform upgrades</td>
</tr>
<tr>
<td>Test new system features and functions as needed</td>
<td>Train users</td>
<td></td>
</tr>
</tbody>
</table>

of the system and are vital partners in cyclical maintenance activities such as upgrades. The IT organization's primary focus is on the infrastructure that surrounds the SIS. It maintains the hardware, databases, and operating systems on which the SIS runs. It develops and maintains interfaces, and creates specialized applications and some complex reports. On the other end, the functional areas are responsible for operating the application. They maintain the tables used to configure the system, develop commonly used reports and queries of the system, and execute major batch processes such as mailings. Table 4.1 presents a typical division of responsibility between central IT and the student services functional offices for the operation of an SIS.

A New Working Relationship

Modern information systems, including SISs, create a shared responsibility between the user department and the IT department. Just as during the initial implementation, troubleshooting complex problems, developing new functionality, and performing major upgrades requires both IT and functional staff. Close coordination and frequent communication must be maintained between user areas and the functional units to make sure these ongoing adjustments occur effectively.

Institutions find executing these shared responsibilities challenging, in part because their method of working with IT is changing. With legacy systems it was not unusual for each functional office in student services to have one or more dedicated programmers in the central IT department supporting their module. While these individuals did not report to the registrar or financial aid director per se, they were 100 percent focused on the needs and priorities of that user area. Over time, deep relationships developed and the IT professional came to understand the needs of that functional area quite
well. Today, IT organizations are moving away from providing a dedicated programmer for each module of the SIS and toward a pooled model in which application developers are cross-trained in multiple aspects of the system.

The method of requesting that IT undertake a project is also changing. In the past, functional units had direct access to and control of the work priorities of their assigned programmer. At most institutions, this created an informal but effective mechanism for identifying and prioritizing projects. To manage their workload better, IT organizations are instituting more formal mechanisms for requesting assistance from IT. Requests must be made through the formal governance mechanisms described at the beginning of this chapter. Each request must be weighed and evaluated against competing proposals from other student service areas or other administrative offices on campus. Although this increased level of structure is often beneficial, it can have an unintended consequence of fracturing the relationship between IT and functional units. Leadership from both areas must work with their staffs to explain how structures are changing and why. The student services area must also keep in mind that these changes do not mean they have less ability to improve the system continuously. As discussed previously, users now have much more ability to effect change to the SIS directly, without the involvement of IT.

Absorbing these changes in responsibilities is also difficult if the functional departments have not anticipated them. Most student service organizations do not have the capacity to take on these responsibilities within their existing staffs. They lack either the required skill sets or sufficient numbers of staff, or both. The next section discusses the new kind of position emerging in functional areas to facilitate the system management responsibilities.

The Role of the Functional Analyst. The system management, continuous improvement, and cyclical maintenance responsibilities (such as upgrades) that are borne by functional departments are significant. Many departments have found it difficult to absorb these responsibilities within existing staff positions. For one thing, the skill sets are different. The system management responsibilities require a hybrid skill set that includes knowledge of the functional area, knowledge of technology, and consultative skills such as project management and process analysis. The workload is also too substantial to be performed by staffs that are also charged with delivering front-line services to students. There just is not enough time to do both jobs well.

Institutions are increasingly recognizing this problem and are creating new positions to focus on systems operations and management. The names of these positions vary. For the sake of simplicity, let’s call this position functional analyst. The role of a functional analyst is much like that of a functional team leader (or external consultant) during the implementation phase. Functional analysts serve as translators among functional users, technical staff, and the software. They have sufficient technical knowledge to perform the system management responsibilities assigned to functional units, and sufficient functional knowledge to maintain the software in a way
that meets the needs of users. Often they serve as the liaisons between the functional departments that employ them and the central IT organization. Typically their responsibilities include the following:

- Maintain system configuration tables.
- Run batch processes.
- Develop standard reports and queries.
- Troubleshoot software issues.
- Train end users.
- Develop specifications for new reports, modifications, and functionality.
- Test new functionalities.
- Maintain system user documentation.
- Identify process improvement opportunities.
- Maintain ancillary technology applications such as Web sites and imaging.

Creating functional analyst positions also magnifies the effectiveness of all end users of the system. The functional analysts provide expert understanding of the software system. Because they are located close to end users, they can provide immediate training, troubleshooting support, and assistance to staff in finding ways to use the SIS (and related technologies) as effectively as possible. They can instigate and facilitate continuous improvement projects and research potential new technology applications on behalf of all users. The presence of full-time analysts also gives the institution some highly productive staff it can draw on to serve as project resources during major initiatives such as an upgrade.

**Organizing Functional Analysts.** Determining the number of functional analysts an institution requires is not an exact science. Staffing needs depend in part on the complexity of the SIS software, the level of technology knowledge present in the student services staff, and the size of the institution. The backlog of process and reporting issues that are carried over from the initial implementation to the day-to-day operations is also a factor. Finally, the expected frequency and scale of vendor upgrades is also a determining factor. Most often, the number of analysts an institution has is a function of how many it can afford. Within reason, most institutions find that they can productively use as many analysts as they can afford to hire.

For large institutions, there is benefit to assigning one functional analyst to each of the four main student service functional offices. Some data-intensive areas such as records or admissions may benefit from having multiple analysts. For small organizations, it may be beneficial to share one or two analyst positions across multiple student service functions. As with any shared position, it is important to agree to clear lines of authority, accountability, and oversight for the position.

On a day-to-day basis, the priorities of the functional analyst should be set by the area to which they report. However, there should also be a
mechanism, either through the SIS governance bodies or via the organizational structure of student services (if there is an overall leader for student services, for example), to assign the analyst to work on divisionwide priorities. Such a mechanism enables the staff to be pooled to work on major projects such as an upgrade or to help out if there is a growing backlog of work in another functional area.

**Recruiting Functional Analysts.** There is no single career path from which good functional analysts emerge. Some are former applications developers who have acquired deep understanding of one or more functional areas. Others are former frontline student services staff who, through work on an implementation project, have developed a working knowledge of technology in general and of the SIS in particular. Many are former SIS implementation consultants who are interested in a more stable work-life balance. Regardless of background, the essential skills of the functional analyst include the following:

- Effective communication
- Project management
- Process analysis
- Experience implementing SISs
- Ability to work independently
- Strong problem-solving skills

During the recruiting process for functional analysts, functional units should solicit the input of the administrative systems staff in central IT. They should provide input into the job descriptions of functional analysts and be included in the interview process. Once functional analysts are hired, they should be encouraged to work closely with the IT organization. They should be invited to attend administrative computing staff meetings (or meetings of the SIS support team). They should seek opportunities to join the IT staff in attending professional conferences and user-group meetings. Finally, functional areas should solicit the input of central IT when conducting performance evaluations on functional analysts.

Forging a close working relationship between IT staff and the functional analysts has two benefits. The primary benefit is, of course, a more productive and effective support team for the SIS. This relationship may also help the institution retain the functional analyst. Since there is no natural institutional career path for functional analysts, it is difficult to motivate and retain them through the promise of promotion. Rather, they may be motivated by the chance to learn new skills (and new technologies) and to participate in challenging projects. A close partnership with IT can help the student services organization provide its functional analysts with these opportunities.
Summary

Modern SISs offer many benefits in terms of enhanced functions that drive the improvement of services and increase access to information to support decision making. SISs are essential to many innovations in student services, such as self-service. In addition, the information captured by SISs provides vital data that can help institutions improve retention, spot trends in enrollment, and forecast demand for courses. However, these systems require a commitment that extends far beyond the initial implementation. During the planning and selection phases, institutions need to weigh carefully the total cost of owning and operating an SIS. Funding and staffing plans that acknowledge the continuous implementation effort these systems require need to be established for both IT and functional departments. A realistic plan for meeting this ongoing support challenge may in fact be more essential to realizing benefit from a technology investment than choosing the product with the most features.

Ongoing success also requires a new partnership between IT and student service functional offices. Functional areas must recognize and accept that they are co-owners of the system and must play an active role in its maintenance. IT organizations must accept this shared responsibility and cede to functional areas the authority they need to govern and maintain the system. Effective governance structures are needed to coordinate the many operational decisions that will arise as a byproduct of the integrated nature of modern SISs. These same governance groups must be empowered to make hard choices among numerous priorities for enhancement and among the scarce resources available to act on these priorities.

Finally, student service organizations need to change how they think about their staffing needs. They need to create positions like the functional analyst role that are dedicated both to managing the SIS and to extracting additional benefit from it. In this way, the institution will be in the best position to maximize the return on its investment in technology.

References


PHILIP J. GOLDSTEIN is an independent higher education consultant and research fellow with the EDUCAUSE Center for Applied Research. He has more than sixteen years of experience consulting with colleges and universities on the implementation and use of technology to improve administrative services.