

HIGHER EDUCATION ERP: LESSONS LEARNED

USING THIS FRAMEWORK FOR ERP COULD SAVE
YOUR UNIVERSITY MILLIONS OF DOLLARS

By **Dave Swartz** and **Ken Orgill**

Universities often spend in excess of \$20 million each to implement modern enterprise resource planning (ERP) projects that can take two, three, or more years to implement. The early report cards coming in from across the country on ERP projects in higher education show mixed results. In this article we share our experiences and lessons learned on ERP as chief information officers of large universities. Specifically, we provide a framework for approaching an ERP that could save your university millions of dollars.

What Is ERP?

The term enterprise resource planning was coined in the early 1990s. The ERP project yields a software solution integrating information and business processes to enable sharing throughout an organization of information entered once in a database. While ERP had its origins in manufacturing and production planning systems, the scope of ERP offerings expanded in the mid-1990s to include other back-office functions such as order management, financial management, asset management, and human resources management.

The range of functionality of ERP systems has further expanded in recent years to include more applications, such as grants management, marketing automation, electronic commerce, student systems, and supply chain systems. Examples of ERP systems include those from Oracle, SCT (Banner), PeopleSoft, and SAP.

Figure 1 shows the sequence of events in an ERP project. It usually starts with a needs assessment and requirements analysis, and ends in the first cycle with training and a phased implementation. The continuous circle of development shown in Figure 1 suggests that soon after completion of the first phase or cycle of an ERP project, we're back to planning the next phase. Each successive round of development arises from the need to add functionality and the rapidity of upgrades to ERP software.

Benefits of ERP Systems

Many reasons prompt people to start an ERP project. First let's consider the main benefits of an ERP system:

- Improves access to accurate and timely information
 - Enhances workflow, increases efficiency, and reduces reliance on paper
 - Tightens controls and automates e-mail alerts
 - Provides user-friendly Web-based interfaces
 - Streamlines processes and eases adoption of best business practices
 - Establishes a foundation for new systems and integrates existing systems
- A main advantage of ERP systems is

Figure 1



Table 1**Some Documented Benefits of ERP Implementation**

Description	Before	After
Reduction in paper forms	N.A.	15,200 fewer forms processed
Duration of monthly closing	10 days	4 days
Duration of semiannual closing	4.5 months	2 months
Availability of budget reports	Hardcopy monthly	Available online daily
Online access	315 users	1,645 users
Creation of account codes	Manual	Automatic
Alerts	0	16
Approval process	Manual	Electronic
Online requisitions	775	11,400
Paper requisitions	12,973	4,323
Autocreated purchase orders	0	11,565
Online receipts	0	6,054
Supply chain forms	N.A.	Eliminated 21 paper forms
Performance metrics	N.A.	Weekly
Policies and procedures	Inconsistent and undocumented	Desktop manuals and online
Days between letter-of-credit draw downs	30	7

improved access to accurate and timely information. As presidents, chief financial officers, or boards attempt to understand a university's overall performance with existing legacy systems, they may find many different versions of the truth. An ERP system creates a single version of the truth because everyone uses the same system. Furthermore, some legacy systems make developing reports or tapping into transaction data stored on the computer quite challenging. Modern ERP systems often improve upon this process by offering a strong foundation for moving to a data warehouse that can provide even more capability to extract data from administrative information systems.

Another reason to consider an ERP project is to improve workflow and efficiency. For example, following completion of online requisitions, workflow processes can forward the form along the approval path more rapidly than with traditional paper methods. This can shorten the time to complete the process, reduce the likelihood of lost or missing documents, and return quick feedback on the status of a request.

ERP systems can also improve controls and program alerts. Alerts, for example, can use automated e-mail to warn budget managers about budgets in danger of running out of funds. Similarly, controls can be implemented to prevent individuals from overspending budgets.

One exciting development in modern ERP systems is the availability of easy-to-use Web interfaces. These interfaces now enhance some integrated portals with one-stop shopping for a wide range of administrative functionality and information.

The ERP project often prompts significant process reengineering and can breathe new life into ineffective and inefficient departments or processes. During an ERP project you have an opportunity to correct broken processes and replace them with modern, system-enabled, state-of-the-art business practices — you don't just want to pave cowpaths!

The ERP project also creates a foun-

dation for new business processes, such as e-procurement, that can yield significant returns on investment. Other forthcoming applications that will be integrated into ERP systems include e-commerce, customer relations management, and pre- and post-award grants management. You can view the ERP system not only as the foundation, but also as your starter house, integrating a wide range of administrative system functionality that you'll add to over time. ERP vendors continue to develop new functionality you can take advantage of — usually at additional cost, of course.

Finally, one additional benefit we've observed from the ERP process is that the individuals involved often bring away from it a new work ethic that spreads to the surrounding workplace. Table 1 provides some documented benefits derived from a new ERP system.

First Steps to ERP: Well Begun Is Half Done

The planning and preparation process consumes a significant portion of time in an ERP project. Just as a woodcutter will often spend one third of his time sharpening the axe before he starts chopping wood, preparation for

the ERP project makes the job go more smoothly and quickly. One of the first steps involves evaluating the needs and requirements that will drive the implementation of an ERP system.

A needs assessment with a definition of requirements is essential not only to guide the start of the project, but also to gauge the success of the project after completion. You should ask yourself, "What do I want my business to become?" At this stage of the game the needs assessment should stay at a strategic level and not get so detailed that you attempt to impose existing modes of operation on a new ERP system. The basic description of needs should be refined to a set of specific institutional acceptance criteria at an early phase of the project. This statement will be used at a later date to help evaluate the success of the project in meeting these goals.

The next step is to review the different solutions available and see which system can best fit your requirements. As part of the fit analysis, you might develop a detailed accounting of gaps. You'll need to conduct an evaluation that compares the trade-offs among the various solutions. For example, some solutions are more flexible and can

accommodate a wider range of best-practice models. Other systems have less flexibility and will require custom modifications to make changes. You should factor the added costs of modifications into the decision process and establish a formal process for evaluating and prioritizing modifications early in the project.

Another issue to consider is best-of-breed versus integrated solutions. ERP solutions today often have a spearhead application — some are better at finance, others at human resources, and still others at student applications. One option is to integrate best-of-breed elements from different vendors; another is to pursue an integrated solution from a single vendor. While an integrated system might not provide the best available solution in all cases, the advantages may outweigh the benefits of best of breed.

Integrated solutions often leverage the advantages of having an integrated store of data. The challenge in this rapidly changing environment with its frequent new releases is to ensure that the different systems work well together. Over time, the differences between ERP systems have become less pronounced, and multiple vendors now offer integrated solutions. Also, don't underestimate the time saved in dealing with only one vendor versus the burden of maintaining relationships and communications with multiple companies.

Finally, your ERP implementation plan must either follow a phased implementation or attempt the big bang approach. A phased implementation — usually the wisest and most cautious course — takes longer and can incur higher consulting costs. West Virginia University (WVU) chose the big bang approach primarily because of Y2K deadlines and the cost that would result from incorporating temporary stubs and drivers to bring modules up individually. George Washington University (GW) undertook a more conservative, phased implementation approach.

Don't Underestimate the Costs

Many of the direct costs of the ERP project are obvious. As a result, project planners often budget and account for them directly. However, these costs don't represent the total cost of ownership of an ERP system. It's important not to underestimate the total cost lest you nickel and dime your leadership over the course of the project. The direct costs include the costs of the software applications and tools. These applications are often licensed according to the number of users, so planners need to project anticipated growth based on new Web-based applications, not the installed base of legacy systems — which perhaps restricted user access.

Next you need to consider the underlying database management system. As it happens, little competition affects this product space at present, though some ERP vendors are making significant efforts to integrate with more than one database vendor.

Our experience with the hardware environment is that you'll probably need much more hardware than you anticipated. On one of our ERP projects we doubled the amount of disk an expert consultant told us we would need and still ended up using every bit and then some. The hardware components budgeted should include the central servers (CPU, disk, and network equipment). Don't forget the need to upgrade PCs to a designated minimal configuration.

Probably the largest area of costs will

accrue from personnel — project staff, back-filled staff, consultants, recruiters, project managers, and raises for personnel. Don't forget the need and costs for training and mentoring. Contracts with consultants often leave out the important area of knowledge transfer — be sure to include this in your contract. While you may remember to budget for the main consultants to assist you with the project, don't forget consultants you'll bring in to conduct a risk assessment and audit of the project at midstream and prior to cutover.

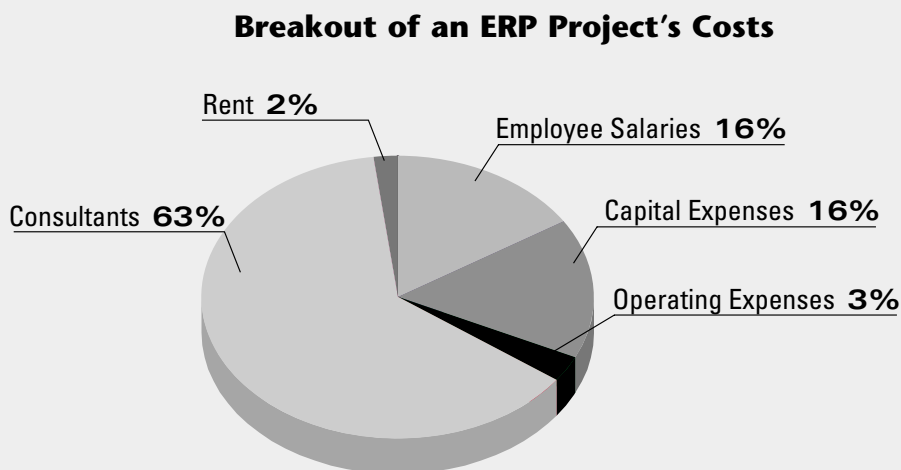
Another cost issue to consider before initiating the project is the cost of ongoing maintenance and future upgrades. ERP upgrades aren't cheap and often involve considerable effort. Make sure your executive leaders know what they're committing to long term, not just for the initial implementation.

Figure 2 provides the breakout of costs for an ERP project. Note that the highest percentage goes to labor (63 percent consultants and 16 percent employees), followed by costs attributed to capital including hardware and software (16 percent), then operating expenses (3 percent) and rent (2 percent).

Contracting for an ERP System

Many institutions will embark on an ERP project with one main vendor because of the advantages of having one conduit for communication. This prime contractor often subcontracts out other work and services as needed.

Figure 2



Any noncontractors who have attempted to build a house and act as their own contractor can attest to the risks of doing it yourself — it's difficult to obtain expert assistance if you don't have a depth of resources. With millions of dollars on the line, it's best to leave this area to the experts.

If you decide to retain a prime contractor for the ERP project, the contract is critically important to your success. Take it very seriously. At WVU we inherited a contract developed without the needed expertise and review. We suffered for it. Develop a precise contract with both legal and ERP expert review. Should you attempt to negotiate a fixed-cost or a time-and-materials contract? We have done both. The fixed-cost approach has some advantages, but the language of the contract will need very careful wording, since vendors will look for loopholes to reduce their costs. On the other hand, a time-and-materials contract should have clear milestones and performance benchmarks to ensure best use of your resources.

We suggest that your contract allow for changing technology during the course of your project. ERP projects can often last for an extended period of time, and new functionality or ERP modules may become available that weren't included in the contract. Finally, be skeptical of vendor promises. Get it in writing!

Even if you have a prime contractor for services, you'll need to build a relationship with the software vendor to deal with difficult issues such as the need to fix software bugs, referred to as technical assistance requests (TARs). TARs often have different levels of severity, from level 1 (most severe) to level 2 or 3. A level 1 TAR in a module may well prevent your system from functioning.

We've found that new releases of software often exhibit a number of these problems, which might prevent you from going live on an intended date without a workaround solution in place. We recommend you have a clear understanding, in writing, from vendors on their commitment to repair

these TARs in a timely manner. A delay of one month in going live could cost your institution hundreds of thousands of dollars. Your implementation partner can help, having lived through this process many times and often having a long-term relationship with the software vendor.

Limit Customizations and Scope

One of the biggest problems in ERP project implementations arises when the institution attempts to customize the new system to fit every existing business practice. Most state-of-the-art ERPs are based on best and current business practices. Existing institutional business practices often have evolved over many years, becoming outdated and arcane. When faced with a choice, many institutions choose to adapt the new system to the old business practices because "we've always done it that way." The resulting modifications add a huge cost to the project and perpetuate an outdated way of doing business. Take the opportunity to reevaluate business practices and workflow processes, possibly incorporating those suggested by the new ERP system. This isn't an easy or quick task, and it's fraught with political peril — but what better opportunity will the institution have to accomplish this important review?

In addition to avoiding the danger of mushrooming customizations, clearly delineate and effectively limit the scope of your project. "Scope creep" can become a major problem in any poorly managed project, but especially in an ERP project. Your contract will help you manage this problem if you make it precise and sufficiently detailed, but especially in a time-and-materials contract the vendor will be more than happy to add functionality for a price.

As the project progresses, members of the campus community will see "critical" functionality that needs to be added and will lobby hard for additions to the original scope. While new functionality shouldn't be rejected outright, you'll have to make hard decisions to keep the project from careening out of proportion. Even

worse is the scenario where customizations quietly happen behind the scenes and without the project manager's knowledge. Take care at all stages of the project to contain scope creep.

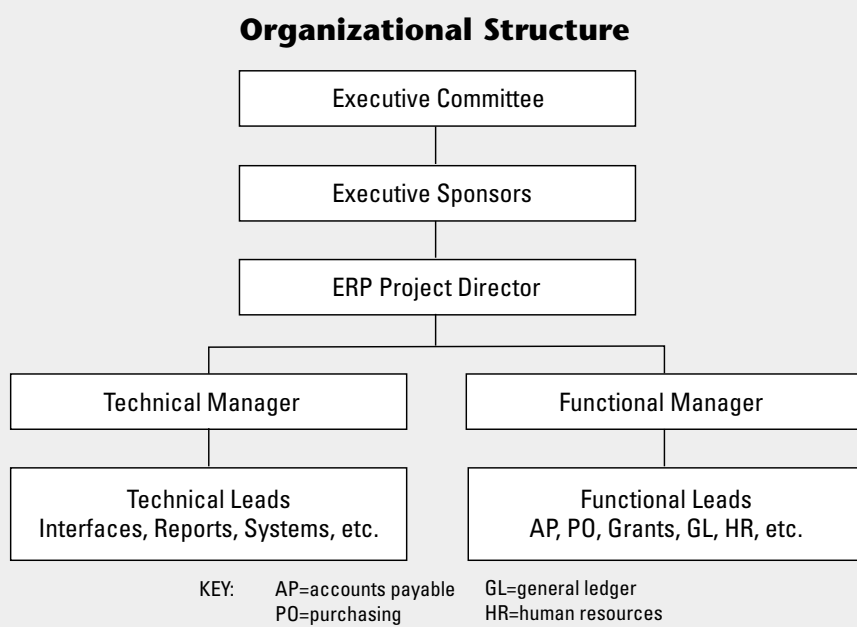
Use of Consultants

Most large-scale ERP projects employ consultants, who can play many different roles. Consultants can help staff the project team, help back-fill positions, take responsibility for project management, audit the project, function as the prime contractor, and serve as the one source for everything from software to hardware and personnel for the ERP system.

On an ERP project you need to select the proper balance between university and outside people. Too many outside people and not enough university people may make it difficult to transfer knowledge of the new ERP system. You may also be missing much-needed information on legacy operations and procedures. It might help to open channels to more than one group of consultants, since it reduces the leverage each has and also provides a greater range of resources.

We found it cost effective at one of our institutions to employ "Big Five" equivalent consulting (at relatively high hourly rates) for tasks requiring intimate product knowledge while using cheaper local talent for more run-of-the-mill programming and other tasks. Check with other universities prior to hiring consultants to make sure they performed well on similar ERP projects elsewhere. Also, plan an exit strategy for consultants, since they're expensive — the monthly run rate (the expense of the project) makes it critical that projects not run over.

Finally, make sure your contract includes a clause that gives you right of refusal over any individual who doesn't perform to your expectations. Generally speaking, you should have at least a week, preferably two, to send back any consultant who isn't right for the task assigned. Also, watch carefully for consultants who've been oversold by their company and are actually relatively inexperienced.

Figure 3

While consultants need to gain experience somewhere, you shouldn't have to pay full price when you're effectively training them.

Project Management

Without question, one of the most important decisions on an ERP project concerns the selection of a project director. This person needs leadership skills and the respect of project members and university administration. Although it's better to find an insider with loyalty to the university, you need an experienced project manager — a professional. If your logical choice for project director has superior knowledge of the functional area, but no formal project training or experience, be very careful. We found the ability to efficiently and effectively run a large project to be the single most important attribute of this key individual, far outweighing any other factor. If experience isn't available, at a minimum make sure you send your designee to formal project management training.

We next suggest that you adopt a standard project methodology that provides some guidance and structure to the project. You want a single integrated project plan, not a collection of independent plans that can't be rolled into a summary report to management.

You'll need to select a manager for

the functional side and another for the technical side of the project. Individual teams, such as the grants management team, need to have a lead identified. The project manager needs to report on a weekly basis to a group of executive sponsors, generally composed of the functional vice presidents (for example, the CFO) and the CIO. A larger group — an executive committee — will provide periodic strategic guidance and support to the project. Some schools have additional advisory committees, such as process owners, who provide focused input from users. Beware — broad-based, consensus decisionmaking doesn't generally work well on ERP projects.

Figure 3 provides an overview of an ERP organizational chart showing the relationships among an executive committee, executive sponsors, and the ERP project manager and leads.

Creating the Single-Team Atmosphere

A typical ERP project involves personnel from a number of departments within the institution as well as a sometimes major injection of consultants. A primary reason for less than successful ERP implementations is the inability of this disparate group to come together in a focused, team-oriented manner. All too often the

team membership polarizes into us-versus-them factions (functional versus technical, everybody versus the contractor, and so on), and the project degenerates into finger pointing.

A successful ERP project requires the functional and technical leadership and teams to develop a strong partnership and a shared commitment to its success. The partnership at the top provides the necessary foundation. Without this joint commitment to work together, don't even attempt an ERP project. Furthermore, if there's a major consultant presence on the project, you'll need key partnerships at every level to maintain the cohesiveness of the team. When possible, consultants should be incorporated directly into the team. This requires major trust on the part of the institution, but is important.

To help the project team become effective, the team should share a location that enables groups to interact and work together away from the day-to-day concerns of the functional units.

Key Functional Issues and Challenges

The ERP functional team will encounter significant challenges not touched on previously, which you should anticipate early in your project:

- Process engineering — Changes in current business processes are often needed and recommended as a result of an ERP project.
- Back-filling of staff — Don't assume it's possible for someone to do everything they did before the ERP started and also serve as an important agent on the ERP project. Be realistic about needs, both on the ERP project and back in the functional unit.
- Training — Seriously consider the type and timing of training. The traditional form of training, where a person goes for days at a time and doesn't interact with the system for months, doesn't work.
- Recruiting and retaining staff — Once you train people on an ERP system, they become more mar-

ketable. You'll need to take into account the difficulty of recruiting key personnel and get started early.

- Conference room pilot (CRP) — Give serious attention to the CRP, which provides a demonstration of the ERP system that users can test drive before the system configuration is locked down.
- Reporting — Will routine reports be printed to the Web? Will users give up their paper easily? The ERP system often enables easier development of ad hoc reports. You'll need to set priorities, since it will be difficult to meet the broad demand for reports on the first day of production.
- Addressing gaps — Gaps will occur between what you perceive is needed and what different ERP solutions can provide. You'll need to make some compromises or get your wallet out to support modifications to the system.
- Setting expectations — Don't make the mistake of raising expectations too high in order to obtain the funds needed. Be more modest in the expectations you set, then exceed them.
- Obtaining user buy in — It's nice to obtain user buy in, but don't think you'll get a consensus at a university.
- Chart of accounts — One feature of an ERP system is the flexibility of supporting a new chart of accounts to accommodate the wider range of fields and account codes needed to replace shadow systems and meet a university's broad range of needs. This is a major undertaking and not as simple as adopting a model available from another, similar university. (At least, so we've been told.) If you adopt a standard model, you could save yourself much time, money, and hassle.
- Validation of data and systems — You'll need to plan a mid-course risk assessment and a preproduction audit before you go live to verify your testing of the system and final production setups.
- Communications — If we had to select one thing we should've done better, it's recognizing the impor-

tance of communications to a successful ERP project. You'll need to work on communications among people on the project and between project staff and users, university management and project staff, and university management and users. Get the word out often and in different forms.

- Returning to functional units — The ERP project has been a "home away from home" for many functional personnel. After the ERP system implementation these people will return to their functional units. Their transition back and the transfer of their ERP knowledge to other functional staff are critical for success and should be planned carefully.

Key Technical Issues and Challenges

The technical issues to consider in your ERP project follow:

- Holding the line on modifications — The technical team needs to work closely with the functional team to hold the line on modifications, since they will inherit the system and have to maintain the modifications. When computing the cost-benefit ratio, don't forget the discounted cost of maintaining the mod through the inevitable rapid upgrades in the future.
- Recruiting the talent needed and retaining them — Personnel issues will hit the technical team as well, especially the difficulty in recruiting applications database administrators.
- How much hardware is enough? — Don't underconfigure your system. It can become a nightmare to spend many months designing and building a system, just to have it perform slowly out of the gate. Remember not only to do functional testing, but also performance and load testing of the system. How large a system should you purchase? Some hardware vendors and consultants have extensive questionnaires used to scale an ERP system. However, a simpler approach is to identify a similarly sized university with the same version ERP system and pro-

posed hardware environment and ask them what's needed. However, be careful that you check with universities that have gone through an implementation and also maintained production along with the resources needed to support a major upgrade of an ERP system. You may find it necessary to upgrade to a larger system just to support the initial implementation.

- Conversion of data — Don't underestimate the challenge of converting years of legacy data into a new format needed for the ERP system. A better approach may be to ship the legacy data to an archive or data mart, then start the new system fresh or with little converted legacy data. Few universities have succeeded in converting more than one year of detailed data and several years of summary data. Be realistic or start early.
- Interfaces — Who's going to maintain interfaces in the new ERP environment? Who's responsible for checking the data prior to it being shipped to the ERP system? The new model of operations will most likely not conform to your previous approach. You'll probably want to accept only clean data into your ERP system, and this places an extra burden on departments with interfaces to the ERP system to provide clean data. Most departments don't have the depth of staff to support their interfaces and clean up data, and have historically relied on the IS department to clean up data once in the system — not a good practice. So, as you push responsibility back to departments, be sure they can handle it.
- Change management and problem tracking — Adopt a formalized process and system for logging change requests and any problems encountered with the system during the development and production phases. At some point shortly after completion of the CRP, you need to put a freeze on change. However, get ready for objections from the functional side and users.

- Minimum desktop requirements — Some ERP systems may require upgrading many of your existing legacy PCs. Some ERP clients don't support Apple Macintoshes. Be sure to factor in the desktop issue and make some early decisions and announcements on who's responsible for upgrading systems.
- Distributed versus centralized production — Various models of production are possible, from distributed to highly centralized. ERP systems can support a distributed model that places more control and responsibility into the hands of the functional units. For many years of centralized legacy production, functional units have thought that if they could only control production, they could also control the priorities and bypass the "bottlenecks" caused by the IS shops. But with the responsibility for production they'll need the same types of controls and procedures, akin to the run books of centralized production environments. They will also need to understand that with the ability to run production comes the responsibility. For example, what happens when a job is started by functional units at 6 p.m. and crashes at 8 p.m.? Who will fix the problem and restart the program so that results will be available the next morning? Functional analysts will grow tired of maintaining nonstop vigilance. If the functional unit asks for the ability to run production initially, be prepared to provide evening, night, and weekend operations support for the system.

Personnel Recruitment and Retention

Probably one of the greatest challenges encountered these days on an ERP project has to do with recruitment and retention of key personnel. You'll also need to consider additional issues:

- Additional compensation for "life of project" efforts — A bonus program will help keep key personnel throughout the project and drive key objectives such as CRP.
- Find and nurture people early — Start early in identifying and recruiting key personnel. You may have to grow your own specialists, so look for people you can count on to stick with you throughout the project and after you go live. Invest in these people, but realize you'll lose some of them during and after the project. A partnership with a large consulting group helps here, since they can provide some depth of resources when needed. Even these companies have problems these days, so you may want to line up several sources.
- Back-filling staff — Make sure you have enough depth on the operational side, since it can really take the steam out of a project to lose key functional staff, pulled off the project to support day-to-day functional operations.
- Staffing matrix — Set some goals for staffing over time, then track progress.
- Knowledge transfer and mentoring — Develop a plan to transfer knowledge from consultants to key university personnel.
- Stress management — The pressures of an ERP project can build up enormous levels of stress in everyone. Manage stress levels through various means, and keep an eye out for employees on the edge, since their stress can cause ripple effects throughout the project. Stress often emerges during disagreements and may require skill in conflict resolution to resolve successfully.
- Team building — Early in the project focus some energy on building teams that can work together even when the going gets difficult.
- Morale of the team — ERP implementations are hard on institutional personnel. The extreme perseverance and dedication required over a long period of time take its toll. Provide periodic downtimes and events to boost and maintain moral. During our projects we periodically provided dinners and lunches, held brief cake and punch parties to celebrate the achievement of important milestones, had barbecues in the parking lot, and supported other activities. These events also gave us the opportunity to recognize

outstanding contributions to the project. After completion of the project, we undertook more major events to recognize team efforts and reward contributions.

Small versus Large School Implementations

Given the cost and scope of most ERP implementations, it's natural to ask if they're beyond the range of smaller schools. Clearly smaller schools need to modify their approach to reduce the cost of an ERP implementation and ongoing production. The first and most important factors to control are the number of custom modifications and the scope creep of the project. The smaller school must also adapt existing processes to the vanilla ERP system. Some ERP vendors now address the special requirements of smaller schools with "quick start" programs that not only contain costs, but also get you into an ERP system in record time. Larger schools may want to consider these approaches as well.

Another area in which to save is outsourcing. Technical expertise is expensive. It's almost always necessary to have at least two of everything — two database administrators (DBAs) or two system administrators — or have a good system of cross-training and support in place. After all, if you have only one DBA, what do you do when that person goes on vacation, gets sick, or takes another, better paying job?

Larger schools generally have sufficient depth of staff, while smaller schools can rarely afford two DBAs making near or in excess of six figures. The best solution may be a consortium, where several smaller schools can share a DBA, or you may want to check out an outsourcing contract as a way to contain costs and reduce risks.

An additional area in which to reduce costs is to minimize the amount of data converted.

The Future of ERP

The ERP space in higher education is moving rapidly. Vendors that hadn't spent much time trying to understand the needs of higher education are

doing much better now — though they can still improve! As a result of the growing competition, vendors are rolling out integrated suites of software that support the thin-client Web interface and object-oriented systems. Frequent new versions make it challenging to keep up with the ERP project that never seems to end. As soon as you finish the implementation, you jump into the next major upgrade.

User groups are growing in importance because they can help influence vendors to become more sensitive to the needs of higher education. They can also provide important forums for schools to share information.

ERP systems producers have rapidly begun to embrace the demands for e-commerce applications such as online billing and payment, e-procurement, and so forth. Maturing data warehouse and data retrieval tools will play a more important role in the future as well.

Sooner or later higher education will wake up to the fact that not every university is terribly different. Given that,

ERP vendors should provide some best-practice models to reduce the cost of ERP implementation. Some ERP vendors do this for smaller schools, but what about the more complex research institutions? They may have to wait.

Finally, as universities move toward common best-practice models, the possibility of outsourcing with an application service provider (ASP) becomes more attractive and economical. One possible vision of the future is for one of the existing Internet service providers (ISPs), such as AOL Time Warner, to maintain the front end of your system through a portal (such as iPlanet) that supports a standard API interface to your production system, maintained by one of the reputable outsourcing ASP companies. This would provide 24-hour, 7 days a week, anywhere, anytime computing. The ISP provides support for your portal, secure remote access when needed, and Web and back-office functions (e-mail, chat, directory, calendar, and so on), while the ASP hosts your production ERP system.

You could add other services such as online courseware or e-commerce (billing and collections through companies such as Sallie Mae) through another outsourcing arrangement with secure links back to the portal. Further, when ready to integrate the system of the future, you can call upon expertise from companies, such as KPMG Consulting, having knowledge of system integration on ERP systems and enterprise portals in higher education. The future integration of Internet technologies, ASP hosting services, and expanded ERP systems could help transform the way higher education institutions operate in the 21st century and move closer to realizing the goal of anytime, anywhere computing. *e*

Dave Swartz (dswartz@gwu.edu) is Chief Information Officer at George Washington University in Washington, D.C. Ken Orgill (korgill@wvu.edu) is Chief Information Officer at West Virginia University in Morgantown, West Virginia.