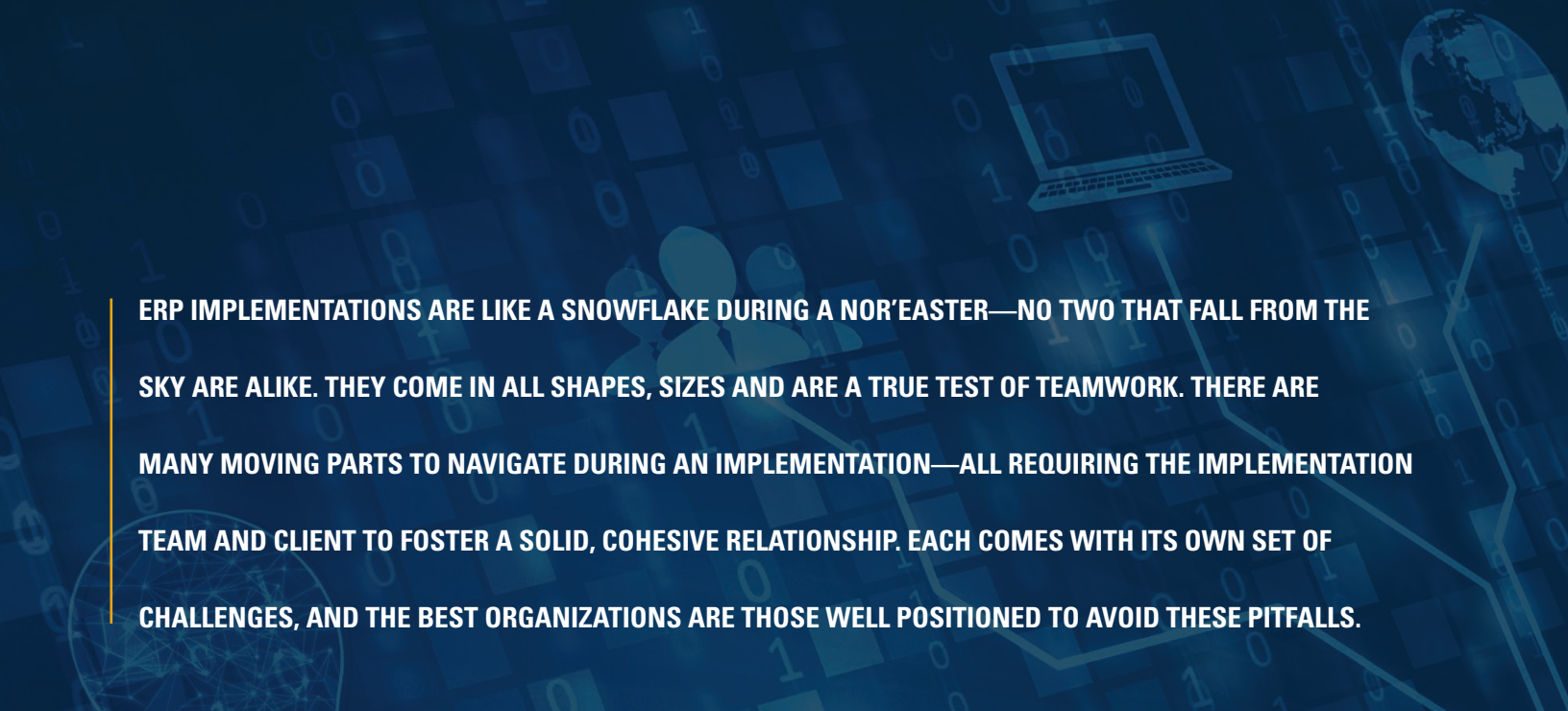


A large, glowing lightbulb is the central focus, containing a bright blue sky with white clouds. The background is a dark, textured surface with a network of white lines connecting various icons of people and stars, suggesting a global or digital network.

THE PATH TO A SUCCESSFUL CLOUD ERP IMPLEMENTATION

**HOW TO PROACTIVELY STAY AHEAD OF
— AND AVOID THE PITFALLS**



ERP IMPLEMENTATIONS ARE LIKE A SNOWFLAKE DURING A NOR'EASTER—NO TWO THAT FALL FROM THE SKY ARE ALIKE. THEY COME IN ALL SHAPES, SIZES AND ARE A TRUE TEST OF TEAMWORK. THERE ARE MANY MOVING PARTS TO NAVIGATE DURING AN IMPLEMENTATION—ALL REQUIRING THE IMPLEMENTATION TEAM AND CLIENT TO FOSTER A SOLID, COHESIVE RELATIONSHIP. EACH COMES WITH ITS OWN SET OF CHALLENGES, AND THE BEST ORGANIZATIONS ARE THOSE WELL POSITIONED TO AVOID THESE PITFALLS.

PERSONNEL BANDWIDTH RESTRICTIONS

To the frustration of many, time is a finite resource and clients still have to perform their current jobs on top of participating in an ERP implementation. Implementation partners understand that learning a modern and sophisticated ERP accounting system can create new stress and is not easy.

One of the keys to avoiding personnel bandwidth restrictions is by proactively addressing it at the start of an implementation:

- Review the project plan and understand the effort required from all stakeholders during different phases of the project.
- Define all roles and responsibilities so the proper people are working on the proper tasks—this allows for effective delegation and efficient communication.

Once the project has already kicked off, there are several things that can be done throughout the project to minimize this risk:

- Hold regular status meetings with all key stakeholders in order to understand upcoming tasks and coordinate the completion of each task with everyone's schedule.
- Set deadlines and assign ownership of tasks so someone is taking responsibility and thinking through how the task fits into their schedule.
- Discuss PTO scheduled during the implementation timeline, so tasks can be scheduled accordingly.



Sometimes it's impossible to predict bandwidth restrictions, but it can be possible to offset this concern by utilizing the following techniques:

KNOW YOUR SLACK TIME AND HOW TO RE-PRIORITIZE TASKS

There is a desired order of completing activities for the most efficient path to go-live, but obstacles always arise.

For example, if all reports are slated to be built and signed-off on before go-live, but an unforeseeable circumstance arises which requires the shifting of tasks—are there certain reports that can be post-launch priorities? By shifting a few tasks to later dates, you are able to keep the overall project timeline—the ultimate goal.

MOVE DEADLINES

While no one wants to alter deadlines, sometimes it's better to delay an action item than to create unnecessary friction, which could have a trickledown effect into other areas of the implementation.

For example, if the primary person responsible for completing the majority of the work is buried in an audit with immovable deadlines, it's better to find additional resources to help or move the deadlines of the implementation than to have unrealistic expectations of when the work can be complete. This scenario requires timely communication to understand when the individual will have more bandwidth and how to reschedule the deadlines.

Running into personnel bandwidth restrictions will halt a project dead in its tracks. It's important to know the project plan, understand everyone's role, and be aware of their time restrictions in order to avoid running into issues.

SCOPE CREEP

At the beginning of an ERP implementation, a number of decisions are critical to define the scope of the project:

- What parts of the ERP system will be set up?
- Who will build and test these parts?
- How long will the project take?

THE AVAILABILITY OF ADDITIONAL CAPABILITIES WITHIN AN ERP SYSTEM CAN LEAD TO A VERY GRADUAL EXPANSION OF THE IMPLEMENTATION, WHICH IS KNOWN AS "SCOPE CREEP". SCOPE CREEP IS WHEN FUNCTIONALITY REQUESTS ARE MADE AFTER THE PROJECT SCOPE IS DEFINED. ADDITIONAL REQUESTS OUTSIDE OF THE ORIGINAL PROJECT SCOPE HAVE THE ABILITY TO NEGATIVELY INFLUENCE THE PROJECT TIMELINE AND BUDGET.

Scope creep can happen at any stage of the project and can have significant effects on the phases that succeed it.

A request for new functionality will require new design, additional configuration time, extra training, and more testing. A new request later in the project can result in the implementation team delaying various phases of the project in order to re-do work that was already completed. This can result in unnecessary strain on the project budget and hours.

The solution to containing the scope of a project requires having a deep understanding of the agreed-upon project parameters (scope) before the implementation starts. It means that project management will have to identify requests that are out of scope and communicate how those requests will be handled. It is important to review the project scope with all stakeholders during the project kickoff or in a separate meeting at the START of the project.



Every project member should be aware of what requests fall out of scope and which ones are in scope, but this responsibility ultimately falls on the project manager of the implementation team.

OUT OF SCOPE REQUESTS CAN BE HANDLED IN TWO WAYS:

1 New requests might be more suitable for a future phase of the project, which means that they will be addressed once the core functions of the system are in place.

This will allow for the entire project team to focus on the more immediate tasks: designing, building, and testing the elements of the system that are most important to the business, with the knowledge that these are not the only changes, but just the ones that need to be put into place first. By doing so, a more holistic view of the project can emerge.

2 Sometimes the new request cannot wait and needs to be part of the current phase of the implementation. If this is the case, a change order should be written and signed by both the client management and the implementation team. A change order is an addendum to the original scope of the project that details the additional hours and budget needed to complete the additional tasks. This ensures everyone is on the same page and avoids uncomfortable conversations later in an implementation.

Sometimes expansion of scope is necessary to put together the proper solution. Other times it is not needed for the current phase of the project and may put unnecessary risk on the project timeline. The key is to have a full grasp of the project scope and the business needs. Doing so will allow the project to stay on-track and help craft the ideal ERP system.



INSUFFICIENT TRAINING

Picture the following scenario: you're a senior staff accountant at an organization implementing Sage Intacct to replace an outdated, on-premises accounting system. Countless hours have been spent gathering system requirements, re-designing the chart of accounts, designing the dimension structure, implementing new workflows, and it's the eve of the long awaited system go-live. Given your seniority on the team, you're expected to hit the ground running and lead the rest of the finance staff. Suddenly, you realize you were never properly trained on how to use this new and sophisticated system—it's like buying a Tesla without a driver's license. There are a number of steps you can take to ensure your people are well trained and ready to hit the ground running.



ONSITE TRAINING IS OPTIMAL

When training on a new ERP system, it is imperative that the trainer is onsite with his/her clients. While conference calls are well suited for status calls and other meetings, face-to-face training for end users is irreplaceable. This format leads to better engagement, more questions, and a smoother overall go-live process. Being in the same room allows all participants to provide their undivided attention. Be sure to include this in the scope of the engagement letter during the sales process!



SANDBOX ENVIRONMENT FOR TESTING:

During the implementation period, the implementation team will complete the initial build of the new system. It is strongly recommended to obtain access to a test copy of the production system. Depending on the software, it may be an additional cost for an extra environment, but that cost will be justified in the long run. Having a sandbox environment for all staff members to have a “safe space” for training and testing the new system will lower the risk of anything going wrong in production. It is recommended to enter actual transactions in the sandbox environment prior to going live in order to simulate real-life activity (please see the next section, “Lack of User Acceptance Testing”, for more information on testing in the sandbox environment). Additionally, it’s advisable to have a Sandbox environment even after go-live, so new development and issues testing can occur in a risk-free area. The last issue anyone wants to encounter is having an unproven configuration change in the production environment cause issues already working functionality.



INTERNAL GROUP TRAINING SESSIONS

Lean on your co-workers! During the training and testing phase, implementers suggest training an hour per day prior to going live in the new system. An example of this would be to schedule ‘lunch and learns’ at the office dedicated to learning system capabilities and new features. Collaborative approaches often yield the best results and the best questions. Finally, make sure to write down all of your questions and review them with the implementation team in order to gain a greater understanding of the new system.



CONDUCT WORKSHOPS WITH YOUR IMPLEMENTER

Even after participating in system training, some individuals might not yet feel comfortable enough navigating the new system to enter a real-life transaction. To fix this, ask your implementer to meet over a screen-sharing conference call. This will allow the implementer to coach you through the process while you’re making the points and clicks throughout the system. This method will serve two purposes: to learn the most effective way of properly completing the task at hand, as well as gain experience with live practice. The more experience with the new system, the faster you’ll learn the ins-and-outs.

All of these techniques can be used together or in different combinations in order to reduce the risk of having an untrained staff.

LACK OF USER ACCEPTANCE TESTING

USER ACCEPTANCE TESTING (UAT) REPRESENTS A PIVOTAL MOMENT IN ERP IMPLEMENTATIONS. IT PROVIDES AN OPPORTUNITY FOR CLIENT MANAGEMENT TO NOT ONLY SEE THEIR SYSTEM DESIGN IN ACTION, BUT TO RECEIVE FEEDBACK FROM EMPLOYEES AND GET A CLEAR IDEA OF HOW THE NEW SYSTEM WILL HELP THEIR BUSINESS.

During a conventionally structured project, UAT will begin only after all system requirements have been gathered by implementers, a complete system design has been agreed to by client leadership, the implementation team has configured the system, and the end-users have been trained to use the system. Well-executed requirements and design sessions will need significant collaboration between implementers and clients, but do not usually include any work in the system by either leadership or end-users. Only when UAT starts do the end-users get an opportunity to see what has been built in the new system and begin to experience the impact the system will have on their current role. The importance of active client participation in this project phase cannot be overstated – implementers will have performed system testing of their own, but not with the comprehensive understanding of specific day-to-day operations that end-users bring. The inclusion of end-users complements the design completed by implementers and client leadership and plays a critical role in the success of any ERP implementation.

Due to its position in the project timeline, as well as the involvement of more resources, there are some inherent risks to UAT. **These are the most common risks that can compromise a successful UAT phase.**

EXTENDED DESIGN PHASE

The Design phase of projects may be extended for many reasons, including insufficient requirements gathering, inability of client leadership to agree on new business processes, or lack of understanding and communication between implementers and clients. Because the Design phase proceeds UAT, its extension can often come at the expense of the UAT timeline.

LACK OF AVAILABLE RESOURCES

UAT will necessarily mean additional responsibilities for end-users, as they will be tasked with system testing as well as the regular requirements of their position. UAT can be seen as a secondary responsibility, and therefore will receive lesser attention than more immediate matters.

LACK OF STRUCTURE AND ACCOUNTABILITY

UAT requires participation and preparation from both implementers and clients. Implementers must establish what is required of end users and provide support and documentation to allow end-users to effectively participate. Client Management must plan ahead to ensure end-users will have enough availability to participate in UAT and effectively carry out their normal tasks. Finally, end-users must actively participate and provide feedback to implementers on what is (or isn't) working. Failure in any of these areas can severely compromise the success of the UAT phase and the ultimate success of the project.

It is also important to note that the risks described do not occur in silos, and in fact are frequently compounded by each other—an extended design phase can delay UAT until a time when resources are not as available, or a lack of available end-users to test can contribute to a compressed UAT phase. Since there is not usually a distinct cause of an ineffective UAT phase, it is crucial that the pieces for a successful one are in place beforehand. This will require the project management team to meet well before UAT begins, to avoid the risks described above.

In addition to making resources available and providing enough time to complete, there are a few essential items that must be complete for UAT to be successful:

1 Implementers must provide quality test scripts for end users. Generally, test scripts will provide the scope of what processes must be tested in the system. Quality test scripts will detail: the task, the steps users must follow to complete the task, the expected results of the task, and how to verify that the task was performed successfully. Because end-users are frequently involved in design and discover sessions only in a limited capacity, the implementers must be sure to give the proper context for the testing, not just where to click. This will help engage end users as members of the project, allow them to see the bigger picture of where the system will help the company, and provide useful practice and experience performing tasks in the new system.

2 Implementers must also be sure that there is a mechanism in place for end users to report issues. This mechanism can be as formal as providing an external system to log and report issues (such as JIRA or Zendesk) or as informal as simply emailing issues and information to implementers. The correct solution will vary widely depending on the size and scope of the project, but without feedback on testing, implementers will be unable to troubleshoot errors and provide additional guidance before the system goes live. This mechanism will also allow client management to know the progress of testing, as well as keeping them aware of potential large project issues that must be discussed.

3 Client management must work with the implementation team to finalize the scope of UAT before the phase begins. Implementers will always have input on what processes should be tested, but it is crucial that the client informs these decisions based on their experience and the needs of the end-users. These will often involve processes that have been changed or reimaged during the implementation, so the client's reasons for undertaking the project, and the benefits they hope to derive from the system, must be stressed here.

Ultimately, a well-executed UAT phase will provide enormous benefits. By having other users test their processes as if it was a live environment, implementers can make design tweaks and troubleshoot system issues well before the system goes live. Client management will be able to get feedback from their own employees on how they feel the system works, as well as observing new business processes in action. Finally, client end-users can get an advance look at the system, as well as the opportunity to gain experience using it in a consequence-free environment.



DATA RECONCILIATION

“Garbage in, Garbage out”*, or as it’s abbreviated in the computer science field, “GIGO”, means the quality of information output from the system is dependent on the quality of information input into the system. If the initial data imported in the system is incorrect, then the data in your reports will also be incorrect, which ultimately defeats the purpose of an ERP implementation. In this scenario, decisions are made based on incorrect data—this is why data reconciliation is critical to a successful project.

DATA RECONCILIATION IS THE PROCESS OF MATCHING THE DATA IMPORTED INTO THE NEW SYSTEM TO THE SOURCE IN WHICH THE DATA ORIGINATED FROM (E.G. PREVIOUS ERP SYSTEM, EXCEL, PAYROLL SYSTEM, BANKING SYSTEM).



Here are some strategies on the data reconciliation process:

- 1.** Make time for it! This activity is largely completed by the client and not the implementer because the client knows their data and has to sign-off on it being correct. Data validation is going to require dedicated time, so make sure you’re aware of the project plan ahead of time and schedule time accordingly.
- 2.** Since the data being reconciled typically spans multiple years and business areas, it is important to appoint an owner of data reconciliation to track the progress of the effort. It is important to ensure the owner has the proper knowledge, training, and resources necessary to complete the task.
- 3.** Validate the data in newly built reports. As part of an ERP implementation, the implementers will likely build new financial statements as well as other reports. Be sure to validate the data in those newly built reports—this accomplishes two things:
 1. it ties out the data
 2. it ensures the reports are built to the business’s needs.

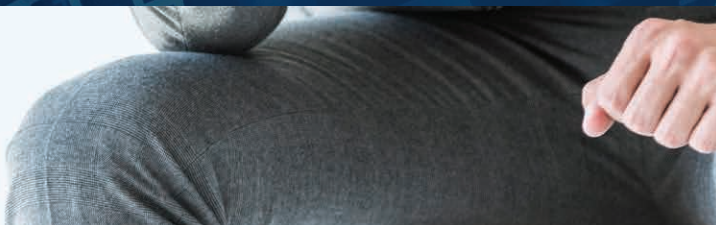
It’s incredibly important to make sure the data is correct before relying on a new ERP system for reports, so make sure to dedicate an adequate amount of attention to reconciling the data.



CONCLUSION

Implementations are a marathon that involve managing a lot of different variables (e.g. timelines, people, functionality, testing, data...). While every implementation is unique, these are the common pitfalls we've come across.

BY KNOWING THESE PITFALLS AHEAD OF TIME, HAVING STRATEGIES TO PREVENT THEM, AND UNDERSTANDING HOW TO DEAL WITH THEM IF THEY OCCUR WILL GIVE THE IMPLEMENTATION THE BEST CHANCE AT BEING SUCCESSFUL – WHICH IS WHAT WE ALL WANT!





ABOUT SC&H

SC&H Group is a nationally recognized management consulting, audit, and tax firm serving clients from rapidly growing mid-market businesses to Fortune 500 companies with global brands. Their Accounting Solutions practice provides outsourced accounting services and Sage Intacct Cloud ERP implementations for organizations in myriad industries. We embrace the future and help clients prepare, innovate, and evolve their businesses in this complex and highly competitive world.

For more than 25 years, SC&H Group has demonstrated its commitment to delivering powerful minds, passionate teams, and proven results on each and every engagement. Learn more at schgroup.com.