



Successful multi-country ERP definition and deployment

Delivering actual harmonization and efficiency gains

Marc Peeters and Onne Schouten

Realizing harmonization- and efficiency gains by deploying a multi-country ERP solution is not obvious. Often, ERP programs fail to achieve time, budget and quality objectives. Our practice learned that harvesting benefits in the timing-, cost- and quality domain is possible when a careful planned out program is followed. Topics like senior management commitment, strict terms of engagement and a smart deployment strategy are vital. The plan should be addressed upfront and adhered to during the deployment.



E.M. Peeters MSc RE
is a director at KPMG International.
peeters.marc@kpmg.nl



O. Schouten MSc
is a manager at KPMG Advisory N.V.
schouten.onne@kpmg.nl

Introduction

Improving quality while reducing costs

Improving the quality of IT solutions (to better support business) whilst reducing the IT budget is a general challenge for most companies these days. An IT strategy in line with both goals is typically executed by multinational enterprises by the global harmonization of business processes supported by a single ERP solution. The ERP solution is centrally designed and developed. After deployment the system is maintained and supported according to standardized processes either on a global or regional level.

Pursued benefits

The benefits of this approach are evident and can be categorized as “cost” and “quality.” Shared development expenditures, economies of scale in deployment (re-use of

knowledge), less dependency on key IT personnel and post go-live governance and support should lower IT expenditures. Qualitative benefits result from the increased maturity in terms of harmonized governance and IT support processes, and the quality of the IT system itself. As the development costs are shared, the technological standard that becomes available is often of a higher level than individual firms or subsidiaries could afford.

Failures

Although not many would doubt the theory underlying this strategy, in practice the results in terms of improved quality and reduced costs are often disappointing. Disagreement on functional scope, deployment budget and timing overrun, lack of leadership commitment, over-engineered IT solutions and limited benefits from shared-service centers due to “not-so-harmonized-consensus-solutions” are common practice.

Local firms should adopt global processes and policies instead of adapting system functionality

Goal

In this article we describe how you could improve your chance of success in reaching the vision of harmonized simple IT solutions and processes. This is based on our experience with the implementation of ERP solutions in client cases as well as our own rollout program. We have identified what challenges you most likely will face along the road, what practical guidelines you could adopt, and what drivers for success our experience shows us are generally applicable.

Scope

This article is limited to the process of solution definition and deployment. It does not address the challenges of establishing a shared-support center. The approach and tips mentioned in this article are most applicable for enterprises that face the challenge of defining and deploying an IT solution in more than ten countries.

Structure

Most organizations that face the challenges of harmonizing IT solutions cross-border adopt one of the approaches as proposed by the large ERP vendors and service providers. These approaches are typically structured in line with the sequential phases stated in Figure 1.

In this article we will not argue this widely accepted approach nor propose a completely different way of managing such a project. Instead we will take this structure as a starting point from which to provide insights and tips – based on practical experience – to increase the chance of success.

These four phases will be addressed. For each phase we describe the typical approach and its limitations. Pitfalls are discussed as well as our tips and best practices to increase the chance of a successful implementation.

1 Commitment

Multi-country ERP programs are lengthy projects that can easily take five to ten years from the initial idea through to solution definition, pilot-project rollout and finally global adoption. Practice teaches us that approximately half of these projects will fail. Budget overruns of more than 100% are unfortunately not unusual.

Considering the impact and size of such a program, a number of steps should be followed that can be considered

pre-requisites. If these steps are not taken, or if no preliminary management agreement is reached in these areas, a challenging and intense project like this is likely to fail.

Senior management commitment and business strategy alignment

It is essential to have the program supported continually and consistently by senior management, even if there are setbacks along the road. The starting point is to have the project embedded or at least aligned with the business strategy already supported by senior management. This will help to communicate transparently how the ERP project is supporting the global strategy. A thorough and realistic business case will support this message. The governance structure of the organization and its culture determine how this step should be detailed exactly and which parties (e.g., management of subsidiaries) should be involved.

It is important to keep in mind that an IT project should never be employed to try to change certain business processes. This is often one of the root causes of the resistance to change. Preferably, these changes should be implemented before the IT project starts.

Guiding principles

When the project eventually reaches the stage of deployment it will become evident how a clear set of guiding principles defined and/or supported by global senior management is very helpful. In other words, defining a set of guiding principles should be done at this early stage and not during the actual design or deployment.

Think of guiding principles as agreed terms that can be leveraged (and reviewed) during the project to make decisions and solve differences of opinions. Refer to the box “Sample guiding principles” for a number of sample principles.



Figure 1.

The core functional scope should focus on the high volume transactional processes. For incidental cases such as fiscal and regulatory reports a solution outside the core system should be considered

Sample guiding principles

- Local firms should adopt global processes and policies instead of adapting system functionality (subject to tax, legal, or statutory requirements)
- Changes should only be part of the global template in case X number of countries in Y regions agree
- Set the percentage of business processes to be covered by the central solution (e.g., only high-volume standardized processes, or 70% of the total number of business processes)
- Historic data (closed transactions) will not be migrated to the new transactional system, only active master data and open transactions
- Language support of the system and support organization is limited to a global defined set of supported languages

The outcome of this phase includes some form of statement by senior management in which the business strategy alignment, business case, benefits, guiding principles and (most importantly) the unconditional commitment of management to this project are recorded and communicated.

2 Central Design & Development

When the decision to develop and deploy a global ERP solution has been reached, the solution is to be defined and built.

Requirements gathering – pitfalls

The typical approach for determining functional requirements consists of a series of central workshops with representatives from the regions and/or individual countries. If not carefully planned and facilitated, these workshops can easily deteriorate into fruitless speculation about perceived “must haves” for the new system. Common pitfalls are:

- The business model is either not fully prepared, or, if it is prepared, it is quite often impractical, as it is too theoretical or too generic.
- There is no consensus on functional scope; meaning there is no agreement on one system with endless different

configurations and process choices that could please every individual. This results, by definition, in increased costs for development, deployment and maintenance.

- The appropriate stakeholders are not involved in defining the central template, which results in the template not being recognized / accepted by local entities. During deployment this will cause frustration among the individual countries, as they feel insufficiently involved in the design phase. This can seriously jeopardize the success of the country deployment and, as a result, negatively impact the whole project.

Requirements gathering – best practice

These pitfalls seem obvious, although avoiding them requires careful preparation and strict attention during the actual workshop.

Ideally, the business process model should contain the Key Performance Indicators for the business. Once the ERP system is rolled out, this should allow for real-time evaluation of the efficiency gains achieved and of other performance indicators such as sales orders in the pipeline, contract profitability, realized purchase management targets, etc. Establishing the optimal team to define the ERP solution depends on the type of organization, the culture and the size (number of countries) of the program. Depending on these variables either local representatives should be directly involved or global representatives should head out to the local countries to identify and incorporate requirements. Involving local representatives gives them the benefit of being able to shape the final solution. For that benefit they should pay the price of complying with some terms of engagement. That is, accepting the guiding principles as defined by global management and being committed to realizing a harmonized solution. This could mean not being able to incorporate all local requirements.

The task of the workshop facilitator is to validate each perceived requirement against the guiding principles before adding it to the project’s functional scope. Personal characteristics and language skills of participants should be taken into account in composing a balanced team and during the actual workshop. It is the task of the workshop facilitator to ensure that all participants, not only the most outspoken, manage to set out their requirements.

In order to critically discuss the requirements as set out by participants, it helps when the facilitator, and preferably also other participants, have a pragmatic “thinking outside the box” attitude. Ask the audience, “what is the business case for the requirements?” “How much money does the company earn with these changes, and how does this compare to the development cost and ongoing maintenance?” Keep in mind that workarounds in the old system should not lead to unnecessary developments in the new system, if the original problem does not exist in the new system. In other words: “ask why, instead of how.”

Validate and classify to determine the functional scope

As shown in Figure 2, the outcome of the requirement-gathering workshops is a set of criteria identified by the workshop participants as being necessary. It is unrealistic to expect full agreement on the functional scope and prioritization. Therefore the requirements should be critically tested against the “guiding principles” (as previously discussed). A proven successful approach is distinguishing three categories of requirements: (1) “core” (2) “options to core” and (3) “out of project scope.”

Firstly, the core scope should primarily focus on the high volume transactional data in order to improve process efficiency on a global scale. For incidental cases such as fiscal and regulatory reports a solution outside the core system should be considered. Although this might not be desirable from a controlling point of view, in all such cases the cost of realization and ongoing maintenance should be taken into account.

Secondly, the “options to core” are functionalities that are not applicable to all countries. They are, however, necessary for a number of countries or regions. On many occasions these requirements are enforced by local regulatory requirements or market practices. In some cases, such as multilingual or multi-currency requirements, a system solution is inevitable. In order to limit the number of “options to core” being developed, a number of principles should be taken into account. (1) The option should only be included in the shared solution if the functionality is required/recognized by a pre-defined number of countries or (sub-)regions. (2) At least a significant part of the development and / or maintenance costs will be paid for by the entities requesting the functionality. This is to encourage entities to carefully consider the need for functionalities and liaise with other countries to come up with shared requirements applicable to multiple countries. The benefit of defining functionalities as “options to core” for individual firms is the fact that development / maintenance costs can be shared with the group of entities. “Options to

core” should be developed as add-on functionality and may never impact core functionality negatively.

Thirdly, requirements that are identified as being “out of project scope” will obviously not be supported by the harmonized system. Local entities should find local solutions to take care of these requirements. Reasons for excluding certain functions from the project scope could be (a combination of): (1) the number of countries is limited, (2) the functions are not used often, (3) the template solution is too complex, or (4) the cost of realization is too high.

The outcome of this validation and classification exercise is an agreed-upon solution which is a recognized and signed off by all participants.

Lock-down configuration

Locked-down configuration enables efficient deployment and cost-effective post-go-live support.

One of the key documents to be used for maintaining the global template is the configuration guide. This document defines the items that will be subject to configuration control, either via parameter settings, value lists or the set-up of the organizational structure. Where possible, default settings for the various parameter settings will be determined and a standardised organizational structure will be defined. The configuration can be applied in an imaginary reference country. This reference country can serve as a model for configuration during local country deployment. It also allows for the option to include the country-specific pack of the ERP solution for the local rollouts.

Locked-down configuration (or at least a locked-down baseline) is potentially a big cost-saver, since it avoids a lot of duplicated effort by local organizations in trying to determine what is the perceived most optimal configuration.

Standardized implementation tooling

The traditional approaches for template definition focus on the functional scope of the ERP solution. Obviously this is the most important and concrete part of the solution definition for future users, however, from a program perspective, the design should not end there.

Country deployment projects consist of a number of similar repetitive activities. As part of a rollout program, it pays off to invest once (during the early stage of the program) in smart re-usable toolkits for such activities. Standardized toolkits, materials and templates should be established in the areas of data migration (standardized data mapping, tooling and verification reports), project management (tools and templates) and change management (communication, training and support material). It

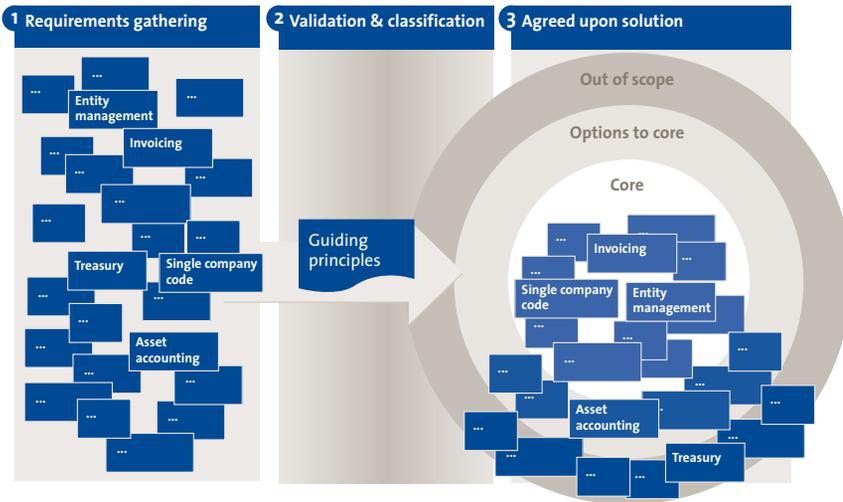


Figure 2.

generally also pays off to invest in creating standardized connectors to link the system to areas that are by definition non-standard (e.g., E-invoicing) but are essential. For integration (connecting legacy applications) it is beneficial to develop a standardized information bus. This bus provides easy access to standardized data buckets, and tools to quickly build views to establish legacy system integration.

Governance and support model

A deployment and operations design strategy should be determined consisting of the infrastructural, organizational and delivery support design. When defining an appropriate support model, legal restrictions with respect to issues such as data privacy regulation, time zones, language, local staff rates and service level requirements need to be taken into consideration. Centralizing the IT operations function is preferable as it reduces cost, improves process maturity and reduces dependency on employees. Central hosting and alignment of service level requirements could also be beneficial. However, a thorough analysis of local legislation and business requirements with respect to data privacy, security and availability should be considered. The strictest requirements of an individual country with respect to these parameters will apply to the whole landscape. The negative impact on the total cost may exceed the benefits of harmonization.

A well planned governance and support model should address the organization design and procedures in the areas of service delivery and service support (e.g., incidents, changes and configuration). Dealing with changes

in complex ERP environments is a delicate task with a potentially big impact. Guidelines for system and configuration changes should not be limited to the general review and approval process for any changes to the configuration of the system. Steps to assess the impact of proposed changes to the configuration setting should also be considered, assessing the cost, timing, and ability to meet the requirements for each proposed change. In addition, the impact on other parts of the configuration should be clear. This means that for proposed changes, the system architecture must be analyzed to determine its impact on interoperability, performance, and ongoing maintenance after go-live. Especially for complex ERP systems, analyzing the impact of proposed changes could be difficult.

Technical development

When the functional scope of the solution is agreed upon (both “core” and “options to core”) and the toolkits for repetitive activities are specified, the solution can technically be build and delivered. A cost-effective way of doing so is appointing a dedicated global development factory, which could be sourced offshore.

The delivered product should be tested thoroughly and signed off by the team that was involved in the solution definition. The acceptance of the solution is a huge milestone in the program, and it forms the starting point for local deployment.

3 Local Pre-deployment

When the solution is designed and built, it is ready for local deployment. Depending on the involvement of the local leadership during the previous phases, it might be necessary to obtain their commitment to the program in general and the guiding principles in particular.

Next, each local entity will go through a process of impact analysis and detailed fit/gap. In the impact assessment the level of organizational and technical change will be identified in order to properly scale the deployment project team: for example, to determine the level of local change management needed.

During the detailed local fit/gap it should be determined to what extent the local entity can start using the system straightaway and where local requirements should be addressed. Local requirements may result from local legislation or local business-critical requirements due to such things as specific local market situations.

In traditional ERP deployment, local requirements are facilitated by local amendments (coding) on top of the global solution. However, we advise avoiding this, as facil-

Ask ‘why’ instead of ‘how’ during requirements gatherings

Every IT service provider will present itself as an ‘implementation partner’. But how do you anchor this partnership in the process?

itating local requirements by coding does impact capital expenditure, maintenance cost, scalability and complexity in a negative way.

Solving local gaps

A more appropriate approach is to identify these gaps and challenge them on their business rationale and the alternative solutions. This will help to classify a significant amount of requirements into the “change management” category.

For the remainder (the must-haves) either an “option to core” should be selected or a solution outside the shared ERP application should be established.

During the fit/gap analysis phase, many of the gaps typically identified are related to local tax, legal, and statutory requirements. In many cases it is advisable to solve these gaps outside of the shared solution. Especially when these cases refer to low volume, low frequency processes (e.g., a once-a-year legal reporting requirement) this is often the most rational approach.

4 Local Deployment

The final step of the program is the actual local deployment of the ERP solution in multiple countries. This should be the logical successor of all previous phases, activities and agreements. Often one country is appointed for a pilot, to finally assess whether the approach, materials and solution itself are fit for the purpose. After a successful pilot, other countries will follow.

Combined local – global team

Ideally the local deployment team consists of both local and global employees. The local team should be selected based on their knowledge of local processes and systems, as well as their organizational knowledge and their (potential) dedication to the “adopt not adapt” strategy. The global (or central) team consists of resources that ideally are involved with the project for a significant amount of time in order to re-use their knowledge and skills in several countries.

Supply / demand

Every IT service provider will present itself as an “implementation partner.” But how do you anchor this partnership in the process? A clear demand/supply focus in the

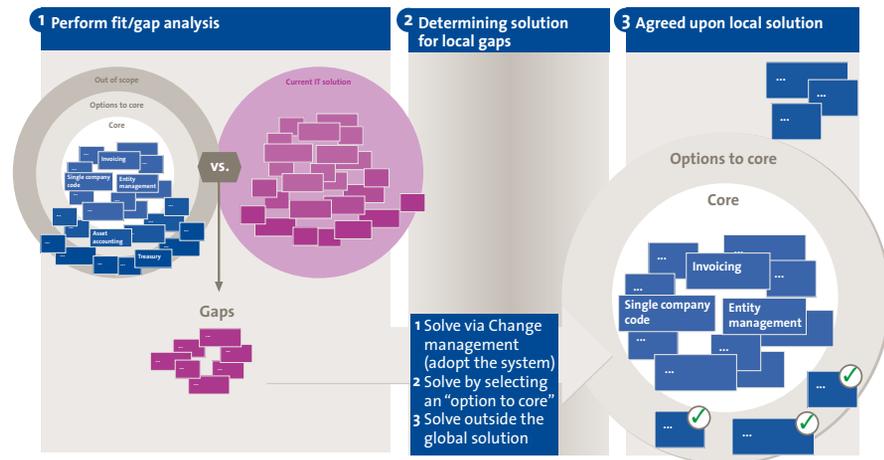


Figure 3.

organization, with transparent activities, deliverables and shared responsibilities, does help. This model is also beneficial in case the internal IT organization is responsible for deployment activities. The model is divided into multiple work streams, each based on a separate functional area (e.g., sales, operations or finance) or the main project task (data migration or integration). Each of these functional work streams is managed by a local business representative, supported by a counterpart of the IT organization. Even more important is the involvement of dedicated business representatives, who are completely integrated within the project team. These representatives should have a clear responsibility for successfully completing the project. Furthermore, their yearly appraisal should mainly depend on the successful completion of the rollout.

Two project leads need to be appointed. Both of these project leads need to keep the other focused on the main target. One of these two project leads is responsible for the “supply” side. He is a representative of the project organization. His main responsibility is making sure the project is completed within the desired time-frame. The other project lead is responsible for the “demand” side. He is a representative of the business. His main responsibility is ensuring the system meets the functional aspects required, as well as keeping the total project cost within reason.

The external supplier needs to make sure there is a good balance between quality and price. However, a better

Lessons learned

- Ensure top management support – Pro-active involvement of a top-level, local business manager should add credibility to the effort required for the implementation
- Once the project is well underway, the project plan should not be put aside. Obviously a thorough plan needs to be drafted at project commencement, describing in detail the various tasks to be performed to implement the ERP solution. Practice teaches us that on many occasions, the project plan is only used to a limited extent once the project is underway and project management is being swayed by the issues of the day. Therefore it is advisable to use a variant whereby the main project plan will contain as much detail as required to maintain an overview of all the underlying work-stream activities and also focus on the project milestones, key deliverables and the main interdependencies. The associated work-stream plans will contain a higher level of granularity and clear identification of the interdependencies on other streams. All plans need to be actively used to monitor the project's progress and will have to be updated on a regular basis to reflect the actual project progress, using, for example, a rolling wave planning approach. On updating the plan, specific attention should be given to the lessons learned for subsequent implementation projects.
- Clearly define the business benefits while being open about the system's constraints – Identify what benefits the organization will get and the areas where the new system might not be an improvement
- Be open and transparent in communicating to the user community – Don't lose sight of the impact to the end user, and engage them throughout the various phases of the implementation project
- Do not underestimate the change management effort required – Especially when choosing an implementation methodology with a template approach
- Perform a pilot implementation – This will validate the core system template and the implementation tooling, all in order to minimize risk in large implementation projects
- Make sure each vendor shares the risk of the project – If possible, base a part of the vendor's license fee on the efficiency gains to be reached after go-live
- Adopt the ERP policies and procedures as defined in the template, rather than adapting the system – Where possible, customization should be avoided, as this can be expensive both as a one-time expense and as potentially endangering the ability to upgrade to newer versions of the software. Customization should therefore only be allowed when a solid business case can be made for it.

Be open and transparent in communicating to the user community – Don't lose sight of the impact to the end user and engage them throughout the various phases of the implementation project

method would be to base the license fee on the efficiency gains achieved by the implementing the new system.

Gather input locally, process centrally

The functional scope is determined and a standard (locked-down) configuration is provided. There are, however, always parameters to be set up locally. Think of the organizational structure, local VAT, etc. An efficient process for gathering this information is using standardized configuration guides and "list of values," e.g., in the form of a questionnaire to be populated by the local entity. The central team should facilitate the required knowledge transfer on-site, to explain the impact of certain local configuration choices. When the data is gathered, the configuration setup for the local entity can be performed centrally to reduce unnecessary travel costs and avoid re-debating choices when the team would be visible on-site.

Deploy as a factory

For each individual local entity, the adoption of a new ERP system is a unique, large project. However, for the central project organization responsible for deployment, this individual entity is just one of many. By approaching the deployment as a factory the activities can be standardized, and employees involved can specialize in specific areas, re-using the knowledge gained.

Because this approach guarantees a continuous flow of work, it enables you to staff the central deployment team with full-time employees. FTEs provide the advantage of complete dedication and efficient execution of activities. This approach will also enable a shorter project timeline, reducing costs.

Stick to the agreed local solution

During the project execution it may be tempting to reconsider decisions made during the previous phase. IT consultants are typically blessed with the skill to satisfy the client. In this case however, this attitude can work counterproductively as reconsidering decisions already made may have a significant impact on related processes and will have a negative impact on budget and timing. It is therefore crucial to remind project team members about the importance of sticking to the agreed local solution, and of the impact of deviating.

If an exceptional event occurs (e.g., a newly discovered gap appears), such that a deviation from the agreed scope is required, the change should be approved by local leadership before investments are made.

Change management is vital

In traditional IT projects the customer requirements are gathered and realized, but in the harmonized template rollout approach not all requirements are met. Although there are good reasons for this approach, it will require significant change-management activities to explain the message to the local entity. The role of local leadership is vital in stressing the vision and benefits of this approach. Local leadership will be challenged by future users, but should stay committed to the vision in order to enjoy success at the end.

Depending on the culture underlying the enterprise and the country, the change-management approach should be tailored to fit the local needs. Involving a local change manager and champions will help to create a better basis to communicate with the local entity appropriately.

Save cost: limit the deployment timeline and size of the project team

The previous stages included several ways to reduce project costs: re-using standardized tooling, sticking to one-way-of-working, etc. The easiest way to reduce the cost during the local deployment stage is to reduce the timeline for the project and the project team. The more ambitious the timeline is, the more team members are focused on making quick decisions and working efficiently. Reducing the number of project team members increases the productivity and commitment to the tasks at hand. Reducing the project size can also be accomplished by dedicating only full-time resources.

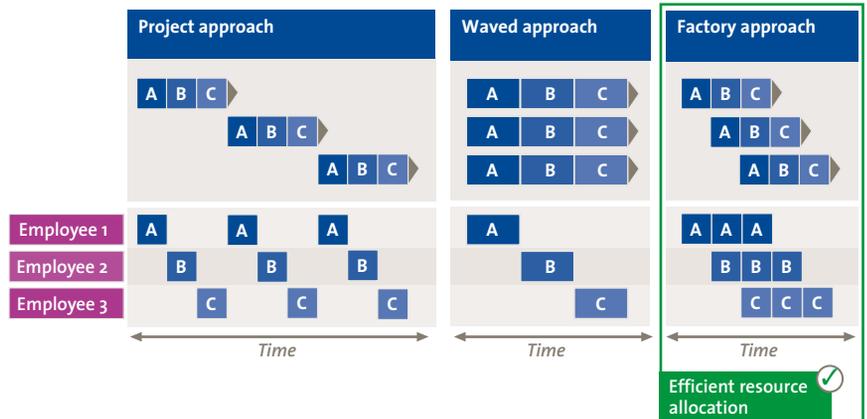


Figure 4.

5 Conclusion

Nowadays many ERP systems used by multi-national companies are outdated and in desperate need of upgrading. In the current economic situation, awareness of the costs involved in upgrading these systems is growing. Multi country ERP programs can be lengthy projects that can easily take five to ten years from the initial idea through to solution definition, pilot project rollout and finally global adoption. Practice teaches us that approximately half of these projects will fail. Budget overruns of more than 100% are unfortunately not unusual.

Why do these projects fail, despite the obvious “lessons learned?” Main pitfalls relate to:

1. The business process model not being completely prepared;
2. The wrong stakeholders being involved during the central template definition;
3. Confusion about the scope of the ERP implementation template options;
4. A pilot implementation on which either too much money was spent or which is too focused on a specific country; or
5. A lack of embedded KPI's within the template solution to improve the efficiency after go-live.

A successful implementation on a global scale is absolutely feasible, when keeping the following aspects in mind:

1. Clearly defined project scope;
2. Local management should have a high commitment to the “adopt not adapt” policy;
3. Flexibility in the template to meet local requirements by configuration instead of development;
4. Change requirements should be tied to a cost model in which countries are prompted to cooperatively contribute to a solution; and
5. Countries should clearly see the benefits of the system compared to their “old” local system, as well as the drawbacks where the new ERP solution will take some getting used to.

About the Authors

E.M. Peeters MSc RE is a director at KPMG International, heading the Regional Application Support group within the EMA region. Over the last 20 years, he has gained experience in national and international ERP projects and implementation of software solutions for the Asset Finance Industry.

O. Schouten MSc is a manager at KPMG Advisory N.V. He gained experience in large international ERP projects in various roles during the initiation, deployment and optimization phases of the ERP life cycle.

The authors wish to thank Duco Peeters for his contribution to the realization of this article.