



PROACTION – Generating Best Practices

Implementation – Conference Room Pilot Completion

By Paul Deis, CEO, PROACTION

818-706-0160

www.proaction.net

5699 Kanan Road, Suite 358

Agoura Hills, CA 91301

paul.deis@proaction.net

Article Summary:

In previous newsletter articles, we discussed a number of critical “get-ready” steps, preparation for Best Practice implementations. While these steps may seem like pencil-sharpening work to some, we and others have long found that using real, Best Practice methods up front saves equally real pain later on. As with other implementation project steps, Best Practice methods are THE way to reduce or eliminate the risk typically associated with implementation projects.

Indeed, it is our view that the high level of risk so often accompanying enterprise software implementation project is specifically and completely due to NOT following known, proven Best Practices. This is classic process management – the degree to which these practices are followed literally determines the quality of the Go Live event and the days that follow it that all of this work is for.

Building on the top management communication and interface work, team selection, and the preparations for the Conference Room Pilot (CRP), in this article we’ll get into the specifics of what the CRP absolutely MUST cover, and cover well.

Topics include:

- **CRP Activities – the Task List**
- **Outputs and results of the CRP**
- **Implementation preparation recap**



CRP Activities - the Task List

Once the team has been assembled, and the CRP workspace and IT work completed, the actual CRP process itself, the real work, can begin. The Best Practice, core CRP process includes, at a minimum, these activities:

- **First –software validation** – once the CRP space has been set up and is operational, the first task is to establish a software configuration baseline, so you ‘know what you got.’ Create a test scenario that, using sample data, allows the team, hands-on, themselves, to step through a basic business cycle and process so the team is assured that it is working as expected, and/or any areas where this is not the case, have been identified and documented. A vendor support consultant may be helpful at this stage – conducting, in effect, a preliminary training session. It is vital that the team actually does the work, not just sit back and watch another demo. Use the “to-be” business process maps and documents as guidelines.

ERP software is notoriously configuration-sensitive. This means that the code itself may be well written and execute properly, but that depending on various parameter and configuration settings, the result you get from a given function may not be what you are expecting, or want.

While in many cases, good vendors have in fact delivered software with actual identifiable flaws, or “bugs” in it, our experience is that configuration settings are just as much of a potential problem. Find both kinds of problems early on, enter them into the Open Issues Tracking log or data base, and keep each of them on the radar screen until it is resolved.

Identifying these kinds of problems as early as possible will give others a chance to remedy them without heroic efforts while allowing the CRP team to move ahead on schedule.

- **Software configuration established and tested** – a Best Practice CRP does not involve setting up new software configurations, then running them, and then repeating the process over and over. The software vendor should be able to furnish a pre-configured version of his software that is close to what your team’s To-Be vision is. (If you aren’t sure what the To-Be vision is, then you aren’t even close to being ready for implementation!) Install and test this one, making minor adjustments if/as needed. We are familiar with too many projects where the entire original implementation budget was consumed and over run by this activity alone – months and months of changing the configuration, then running through the entire business scenario / process again. Exhausting, demoralizing, discouraging, self-defeating, fantastically expensive – otherwise just fine...!
- **Repeat until stable** – if bugs or missing, but expected functions are identified, unless they are minor, repeat the validation process until the software is stable and the team understands how it work, and that it DOES work as intended, and that it DOES support the To-Be process vision.
- **Landmine identification** – as early in the CRP process as possible (or before the CRP starts), the team should identify as clearly as possible any potential issues, which if



unresolved completely, could de-rail the whole project. We call these “landmines” because they have the potential of blowing up, of undoing weeks of hard work, or worse, of seriously damaging the company. These could be major policy issue, a potential acquisition, inappropriate leadership, or an already underway major industry change – anything that could prevent a successful, smooth transition to the new software. Most of the serious implementation disasters we are familiar with have one or more real, serious landmines that just were ignored, swept under the rug, “not my job” or otherwise not resolved. Deadline driven teams can easily be pushed into moving ahead even when things are not ready – almost never with a good result.

- **Open issues tracking** – it is critical that every possible potential difficulty – anything that could cause the live use of the system to cause a problem – be identified, and tracked in a visible way until it is resolved. This is a key part of the CRP team’s activities. The project manager, or a specifically assigned team member, must “ruthlessly” pursue resolution of these as promptly as possible, remembering that the CRP team’s high priority may not be another person’s hot priority. Don’t let any of these languish in someone’s uncompleted Task list, on a Trouble Ticket, or Inbox because they are busy. It is the regular, energetic follow-up of open issues that is more important than the fanciness of the IT tool used to keep track of the information.
- **Initial data conversion** – we prefer doing an initial data conversion, using partial or subsets of live, real data to create a set of hand-match data that everyone already understands. IT professionals will sometimes claim to be able to convert any data but it just isn’t true. Why? The “same” data in two different systems may be handled differently, giving it a different “meaning.” By doing it by hand first, the team will completely understand these often subtle differences. We have never found an exception to this experience, often glossed over by non-users of the data, who in all honesty do not grasp these subtle differences, but that can be critical in some cases. This deeper understanding can allow the team to help design and plan the conversion process – placing the ownership where it is most effective – with the team, not IT “experts.” What YOU mean by “credit approval” may not be the same as what the software “thinks” it is.
- **Detailed to-be process validation and planning** – this is THE critical part of the CRP process. For this the team steps through every part of the to-be business processes, identifying every issue, policy level, procedural, work flow, and anything that affects who owns what parts of the process.

Example: The As-Is business process calls for new customer accounts to have their data gathered by a customer service rep, written up, then forwarded to someone in the credit department who then approves it and enters the new account in the system.

The To-Be process allows the customers to open a new account by themselves via an Internet accessible web site/page. Who is responsible for credit approvals? The CRP process must identify and resolve this question – and all others like it – until every step in each process has been worked through, whoever is needed consulted, approvals obtained, policy conflicts resolved, documentation updated – whatever is required. Do not let schedule push you ahead



regardless of actual preparation progress. If needed, schedule extra work sessions, enlist additional help to offload team members in their regular jobs to catch up. Take the rug, under which problems can be swept, out of the CRP work room.

Outputs and results of the CRP

As the CRP team steps through the Task List outlined above, it should, of course have its eye clearly on the end result – in the form of specific, written documents, plans and detailed understandings of each. At the core is the To-Be process validation, and transition plan that shows clearly how each work group’s members will be able to successfully do their jobs, the work of the company, on Go Live day, and thereafter. The key outputs are:

- **Training materials and plans** – how to adequately train everyone so the Go-Live event and subsequent business operations will be smooth and uneventful, free from lingering unresolved problems. This includes who the instructors will be, what training sessions will be required, when, and who the participants will be. The materials, of course, must be company-specific to be effective, although they can be adapted from vendor-furnished material to some extent.
- **To-Be Adjustments** - to the To-Be process maps and work flow documentation where needed. Some of these will be improvements, and some may not. There may be (new) workarounds or other less-than-desirable actions required. Many new software systems involve some trade-offs – powerful new capabilities but loss of familiar, custom-built functions that did the old system did well. Much is learned during this work, and changes to original visions, plans and understandings are inevitable, even desirable.
- **Validated software configuration** – the team will know what the software does, how it does it, and be able to say with confidence how the business will succeed using the new software’s capabilities. This is a critical aspect to eliminating the risk of the project. It is NOT a technical issue, although often complex settings of parameters and table data that control the software’s function. Large ERP systems can have thousands of settings, hidden away, but controlling functions anyway. The team does not have to understand how these work – how could they? Typically the vendor does not have anyone who understands them all either; no human being could – just make sure the configuration you have works in a way you understand and control.
- **Tested, validated data conversion plan** – data conversion, both software performed and manually entered, requires multiple iterations to overcome a wide variety of often-subtle challenges. The only way to prevent the wrong kind of surprise after Go-Live day is to prepare, run / perform the process, test and validate the results until it works as well as possible. This is another critical risk-elimination step, often the source of serious problems after Go-Live but fortunately completely avoidable. Key aspects of data conversions to bear in mind:
 - **Software performed** – required for all large volumes of data; careful design of the process (there are excellent conversion mapping tools available, but apply thought carefully to their use), then test, re-test, and re-re-test the results by using the data in



simulated live use. The key – can the to-be work flows and business process steps be correctly, reliably performed with the converted data? Data conversion software *does not know the answer to this question, nor can it.*

- **Manually performed** – this is data that is printed out, possibly adjusted in some way and then entered by direct data entry process into the new system. We recommend this method everywhere possible, because problems stemming from these data are rare. Even with sizeable volumes it is possible to hire temporary help or otherwise “crunch” the data into the new system in a short time period.
- **Go-Live Cutoffs** - These are always difficult especially for 24/7, global businesses. However, careful, thoughtful planning will usually enable the implementation team to separate truly high-volume, transaction data from more permanent data so absolutely everything does not have to be converted a few hours before the Go-Live event. Some ways that help:
 - **Convert through a date before Go-Live ahead of time.** This means only the last few days of transaction data has to be converted at the last moment.
 - **Stratify by degree of activity.** Some groups, contracts, products, services, etc., are inevitably less volatile than others. By converting the slower-moving data ahead of time, the last-minute conversion will be shorter and quicker.
- **Volume testing** – the software has been tested under high-volume circumstances, so possible response time problems have been flushed out, fixed and will not cause people in the company to be unable to do their jobs because the system has gotten bogged down. This has been a cause of a number of serious implementation shortfalls and difficulties and must not be neglected, even though it is hard to perform.

Well publicized, major ERP implementation failures have multiple causes. One of the most horrifying, though, is that “everything works” but the system collapsed due to major overload on key servers, network bandwidth, or other killer factors. Just because computers hardware has gotten a LOT faster and MUCH more capable than it used to be does not repeal the Computer Law of Nature – which says that ANY computer, ANY network, CAN be overwhelmed. Make sure you find out BEFORE Go Live day.

Implementation Recap

Of all of the myriad “things that can go wrong” (“Murphy” does live in implementation projects), in our experience, the most important, yet most subtle, thread that runs through all of the aspects is that of making sure that *ownership of the process* is supported, led, and maintained throughout is by far the most critical.

It is the loss of the feeling of ownership by key process owners on the implementation team that is, in actuality, the most common cause of difficulties, shortfalls and outright failures. People find themselves going through the motions, but no longer caring a lot about the outcomes.



If the ownership is strongly felt, the team members will literally be thinking about the project when they are “not working.” This energy and clear focus on getting it right will ensure that serious technical errors are not made, and will cause people to recognize when they don’t know something important, and to get some help with it - before the bomb goes off after Go Live.

The key elements of a low or virtually zero-risk, yet major, enterprise software implementation project, once ownership is established and maintained, are:

- Detailed before and after business process understanding. The team has addressed and resolved all important issues and *knows* how the business will operate under the new system and its processes.
- Landmine clearing / elimination – the team candidly identifies potential hazards or barriers to a successful, smooth transition and has eliminated them.
- Top management fully supports and shows effective leadership where needed for the project.
- An effective interface between the team and top management has been in place from the start.
- Education and training – the team has identified and completes education for everyone in every position that could cause difficulties if left uneducated or poorly trained.

In conclusion to the topic of Best Practice enterprise software implementations, bear in mind that the “devil IS in the details.” These are short articles, hit what we hope are the major steps and show the way for further investigation, learning and preparation. And by preparing and performing an implementation in thorough, complete fashion, you’ll reverse this truism into “the good news is in the details” and have a smooth, successful, glitch-free transition from one enterprise software system into a new one.

Best Practice Q & A

Question: “We have been told by several people that we do not need a dedicated Conference Room Pilot facility, that it can all be done from our regular workstations via Internet meeting technology and conference calls. Why are you so insistent on having a physical place, a meeting room for the team? Isn’t this a bit “old school?”

Answer: “Good question; goes to the heart of how we do, or don’t work together. We are great fans of Internet meetings and conference calls, and have done significant portions of projects using these tools. However, there ARE limitations. As a guideline, the more closely a team needs to work together, to trust each other and communicate not only hard data information, but to perceive more subtle forms of communication, the more physical presence will prove valuable and, in the end, will save major amounts of time.

The lack of trust is the greatest cause of additional work on team-based projects, because it leads



to CYA work, which does not add value to the actual project itself. When people are physically in the same room, learning, growing, arguing, debating, collaborating, disagreeing and resolving issues, there is an opportunity for trust to really grow and strengthen. If you label in-person communication as “100% of the information” that passes between people, as you move further away from this, major portions of this “100%” are lost. Video conferencing would be the closest, followed by Internet meetings, with phone conference calls in last place. Each increment allows the participant to pay less and less attention to what is going on. Everyone is under intense time pressure, it seems, to “multi-task” which is techno-speak for “I’m not really paying attention to you...”

Advanced web meeting technology allows the presentation organizer to discern who is really paying attention to the presentation, as the viewer client software can detect and communicate real-time to the organizer who has moved the window into the background, to work on their email or something else. This doesn’t happen when you are in the same room together. Your CRP is way too important to allow it to be “moved into the background.”