

# **Migrating to Microsoft Project Server 2007: Lessons from the Field**

Victor A. Richardson, PMP, MCSD, MCITP  
Managing Consultant, Catapult Systems



**Catapult Systems**

So, you've been using Project Server 2003 for ages and The Powers That Be have finally decided that it's time to upgrade to Microsoft Office Project Server 2007 (MOPS 2007). Great! But while your executives and other stakeholders are getting excited about using all the new bells and whistles, you're starting to consider the magnitude of the project – and it's a somewhat daunting proposition. Not to worry! With a bit of prior proper planning, the migration to MOPS 2007 needn't be difficult at all – and this article can help.

Before we get started, I should clarify a couple of things. First, I refer to this upgrade as a "migration." Why? Well, you're updating Project Server 2003 to Project Server 2007, of course; however, unlike many simple software upgrades, data also migrates due to the data structure changes (which will be important to keep in mind later when dealing with reports and customizations). Second, this article isn't about the ins and outs of performing a Project Server migration, the technical aspects of installing MOPS 2007, scheduling database backups, or any of the other technical knowledge that's essential for performing a migration. This document basically assumes you already understand the technology, but need help with migration planning from someone who has "been there – done that." To that end, this article discusses the following key planning areas:

- Communication: Disclosure and discovery.
- Test migration: Yes, they're that important.
- Production migration: Let's get ready to roll (out).
- Post rollout: Prepare for Murphy's Law. Seriously.

I also list some references at the end of the article if you wish to find more technical information (including the recently released MOPS 2007 migration best practices whitepaper). Some of them are official Microsoft documentation while others are from other Project Server specialists trying to assist people contemplating a migration to MOPS 2007.

## ***Communication: Information is Key***

Solid and complete communication is essential to the success of any project – and a MOPS migration is no different. People need to know what to expect from the migration, as well as when to expect it. Not only does this increase end-user excitement and subsequent adoption of MOPS 2007, but it also reduces the potential post-migration tech support load. For these reasons, it's incredibly important to have a communication plan in place. Relevant topics include:

- General migration overview: What exactly is happening?
- Time of the migration: When is it happening?
- Duration of outages: How will it affect my work?
- Training plan and schedule: How will I know what to do?
- Contact information for tech support: What if I have a problem?
- New MOPS 2007 features: What's in it for me?

In addition to the communication plan, try arranging a brief demo during lunch to show off the new features of MOPS 2007 (in person or via LiveMeeting). You'll cover those bells and whistles in depth during user training, of course, but again, it helps to give out little doses of information here and there.

When exactly should you start communicating the upcoming changes to your users and stakeholders? The short answer is "as soon as humanly possible." Keep in mind that communicating your message across an enterprise can be more complicated than it appears – you may have to involve certain executives, departments, committees, etc., and get their input and approval in advance. If you propose your communication plan at the beginning of your migration, you increase your chances of getting the relevant information out to users in a timely fashion and avoiding the "fun" of last-minute scrambling.

## **Questions are Critical**

One of the secrets of preparing for a successful migration is to ask questions – and lots of them! Why? Because until you understand the full scope of your current solution as well as what resources you need for the upgrade, you can't accurately continue with the next steps in your preparation. The following are some important questions to ask yourself and your team/stakeholders at the beginning of a migration project. The list isn't all-inclusive, but it will definitely get you going in the right direction.

### **1. Do you have the necessary technical expertise?**

It goes without saying that there are multiple technical items that need to be addressed during a migration. It's critical that you have savvy technical resources on board at the start of your project (either in-house or from a consulting firm) to ensure you don't miss anything.

The right technical resource knows to ask questions such as, "Is it necessary to have SP2a installed in the Project Server 2003 environment?" or "Does form authentication need to be set up – should we even set it up?" These are just a couple of the many questions that will come up during a migration. Be smart and don't skimp on solid technical expertise.

## **2. What's our migration window?**

A full migration won't take place in one day, so you have to plan on downtime for the system. Here is where windows of opportunity become tricky. For example, if a company can't afford any downtime during the first and last weeks of a month due to inventory and payroll processing, then it becomes pretty important to get the system working during the two weeks in the middle of the month. If you miss that window, the migration just slipped about a month.

## **3. How many projects and workspaces are there?**

It's possible that there are more workspaces than projects. This sometimes occurs because we're dealing with Windows SharePoint Services (WSS), and team sites (minus a project plan) might have been created for people to collaborate on something without it being linked to a project plan.

Why is this important? Well, if you're dealing with, say, only 10 projects and workspaces (assuming a one to one ratio) that need to be migrated, then it won't take an enormous amount of time to perform a migration – you can probably finish it in under four hours at the most. However, if you're dealing with 1,500 projects and 2,000 workspaces, you're looking at 10 to 15 hours or even longer, which will dramatically impact your test migrations. You can't do multiple migrations in a normal workday; you might be lucky enough to do one in a day. By getting this information ahead of time, you cut off one potential schedule slippage avenue.

## **4. What customizations (if any) have been done to the workspaces?**

Some customizations that worked fine in WSS 2.0 may cause problems in WSS 3.0. Therefore, it's crucial to document all customizations in your current solution so that if problems arise during the upgrade, you have a place to start troubleshooting. You should note every new document library, column, or web part. Even if all that was done was a modification to a value in a standard column, you should identify it.

## **5. How many users are there in each role?**

While there are some similarities between Project Server 2003 and MOPS 2007, there are also significant differences. Therefore you must ensure all users – from team members to executives – get the proper training with the new technology. Do an analysis ahead of time of how many project managers, team members, etc. are in the organization so you can determine the necessary licensing and training requirements. (And remember to include this information in your communication plan.)

## **6. Are there any SRS reports, InfoPath forms or Microsoft Project macros?**

Unfortunately, there's no "easy" button to push to make SQL Server Reporting Services (SRS) reports that worked just fine with Project Server 2003 work with MOPS 2007 – the SQL code for the reports has to be completely rewritten. This means you must have a solid understanding of the databases and tables within. More than likely you'll upgrade from SQL Server 2000 to SQL Server 2005, which means you'll use SRS 2005. This will dictate that you might even need to make more changes to the reports than just the SQL.

It's also possible that InfoPath forms and Microsoft Project macros will require rewriting as well. While I'm highlighting just these two technologies, you should consider any customization as a potential issue. Maybe something was done with Excel. Perhaps there's a custom application in place for surveys. Whatever it may be, you need to document it and decide how to handle it, and (referring to that first question above) if you have the necessary technical resources to handle it.

That said; keep in mind that not everything needs to be migrated. This can be an excellent time to evaluate the custom items currently in place and decide whether (a) they're being used at all, (b) they need to be upgraded or improved along with the migration, or (c) they can be eliminated outright. Think of it as a Spring Cleaning of sorts that ensures your solution stays as streamlined and elegant as possible – your stakeholders will thank you.

## **7. How many users do you have that need Project Professional 2007?**

This is important because if the number is less than 10, you might want to install the software on each user's desk manually. If it's more than that, you probably want to roll out Project 2007 automatically. However, keep in mind that Project 2003 doesn't connect to MOPS 2007, nor does Project 2007 connect to Project Server 2003. If you're only doing a gradual migration, then you must ensure users have both Project 2003 and Project 2007, as appropriate. Also, be sure to ascertain whether or not the user really needs to have Project Professional 2007 installed – it may be that they just need the access provided by the Microsoft Client Access License (CAL), which could save your organization a bundle.

## ***Test Migration***

By now you may be asking, "Jeez, how hard can a migration be?" The answer varies depending on multiple factors identified during preparation. If you're dealing with a few projects and basically no customizations, then it probably won't be very difficult. However, the more projects you have, the more likely it is that problems will arise. With hundreds of projects and workspaces, it's quite possible there will be a corrupted list item somewhere, a customization won't work correctly in WSS 3.0, or a project just won't migrate for some reason. Often the issues aren't difficult to correct, but you never know until you practice the migration.

That's why test migrations are another key to a successful MOPS 2007 upgrade. One benefit of doing a test migration is the confidence it gives you. If you test migrate and everything migrates over cleanly, when you do it for real, you go into it thinking, "This has worked before in an environment just like this." It's empowering to know you've been successful before, and you can therefore be successful again. Also, you're dealing with production data, so it's important to have successful test migrations to back up your approach to doing the production migration. More practice allows you to completely test out how you'll be doing the production migration, and lets you identify potential hiccups. The more practice you get, the better your chances of having a successful production migration. I recommend completing at least two successful test migrations – even if the first one went perfectly. "Practice makes perfect" is a cliché for a reason – because it's true.

## **Test Environment**

Before you can perform your test migrations, you need the proper environment. If you're purchasing new hardware for MOPS 2007, then you can perform test migrations on that hardware. That will be an ideal setup, as you'll gain accurate information on how long the migration will take, and you can smooth out any potential issues with the hardware long before the production migration takes place.

Setting up a virtual environment is another attractive option. If you decide to go this route, you should set up virtual test and production environments. It's best if they're configured the same way and running on the same or similar hardware. You might use Microsoft Hyper-V, Microsoft Virtual Server 2005 or VMware ESX Server – all are viable options. There are also PC versions available. The main ones are Microsoft VPC 2007 and VMware Workstation; both work well. Here are some advantages and disadvantages to using a virtual environment over using a server.

### **Advantages**

- Easy to restart if a mistake is made to the environment.
- Easy to be mobile (if using a virtual environment that's housed on a laptop).
- Easy to share the image between several people so multiple tests can take place simultaneously.
- You'll have an environment people can use to test out the development/customization of the custom items you've identified.
- A test/development environment is already ready after the production migration takes place.

### **Disadvantages**

- Does not mimic the production migration exactly.
- New issues can arise during the production migration since you haven't been practicing in the actual production environment.
- May not be as "powerful" as the actual production migration.
- If the database is really large, then your VPC image is going to be large too. This will more than likely cause performance issues since you're dealing with an image on a laptop.

In the end, what one company chooses may be different from what another chooses, but that doesn't mean one is right and the other is wrong. The important thing is for the solution to fit your company's needs and provide an environment where test migrations can be performed and users can be trained before MOPS 2007 goes live.

## **Custom Items**

During this phase of the project you need to convert any items that don't get migrated automatically and ensure these items work against a MOPS 2007 database. These items should be easy to identify if you've already answered questions about customization and reports, which I referenced earlier. The list includes:

- Project Professional 2007 filters
- Project Professional 2007 groupings
- Custom Project workspace web parts
- InfoPath forms
- SRS Reports

This is another area where in which a virtual environment really helps. You can continue to practice the test migration in one virtual environment while developers work simultaneously in another to ensure the SRS reports and other custom items function correctly. With yet another virtual environment you can allow key people to begin testing the custom items you have to modify.

Keep in mind that once you're finished doing test migrations, no modifications (such as new custom fields) should be added to the Project Server 2003 environment. While adding new fields technically shouldn't require a new migration, it would be smart to perform one since Murphy's Law is always out there – especially where software is concerned!

## **Key Steps, Roles and Responsibilities**

A migration involves multiple steps. Everyone plays a key part. Outline who does what when clearly and succinctly in a "Roles and Responsibilities" document, and then allow all relevant stakeholders to review and approve the document. It doesn't need to specify exactly how someone makes backups of the databases, but it does need to state that backups will be done, as well as designate the person responsible for them. This document really begins taking shape during the test migrations. During this time you'll also start quantifying how long a particular task will take – and it's a great idea to document this information. For example, it's important to know if upgrading the WSS content database is a three-hour or 13-hour process, since that can help you more accurately plan if you need to perform a future migration or plan for disaster recovery.

## ***Production Migration***

Well, you've practiced, and then practiced some more – and now you feel ready to move the migration into production. Here are some key areas to consider.

### **Testing**

The first day after the migration is a critical time. If the unimaginable happens, you have to decide quickly if you're going to roll back or fix the issue. The test migrations you've performed should mitigate this risk; however, for risk management purposes you need to have a contingency plan in place. In the "Key Steps and Roles and Responsibilities" document specify who tests what, so there is good solid coverage of the system during the first day of testing.

During the first day of testing you don't need all of the SRS reports or other custom items working – they won't anyway until you point them to the new production MOPS 2007 databases. After you've gotten through the initial testing of the system, you can begin to point all of your custom items to the new production databases. This shouldn't be too difficult since you noted what needed to be changed and where while you were rewriting all of the custom items during the test migrations. Your testing also doesn't need to be as extensive if you tested the items a lot during the practice migrations.

### **Training**

Plan for user training to occur before MOPS 2007 goes live. Waiting until the whole migration is complete can cause major lags in continuity and productivity due to lack of user proficiency – and that can lead to some unhappy stakeholders. Remember also that the most effective training is "hands on," so you will need access to an environment (test probably) that has been properly configured.

At some point you'll have to push out Project 2007, which you should have planned for earlier. Remember that all users need to have the new software before you can deliver training to them. Keep in mind also that with a migration project, training can be a little more challenging than if it were just a brand-new, clean install of MOPS 2007. First, you have to worry about users having both versions of software installed on their computers (as previously mentioned). Second, you may have to contend with existing process baggage: "But this is how it was done it Project 2003..."

## ***Post Rollout: Hoping for the Best, Planning for the Worst***

You've communicated your migration information to your stakeholders and users, asked all the right questions, run successful test migrations, tested the system, and then trained all your users. It's all smooth sailing from here, right? Well....not necessarily. Granted, with the proper preparation you'll have already captured the large majority of potential issues and problems. However, the fact remains that since it's impossible to do 100 percent complete testing, items can and will be missed.

For example, SRS reports that worked well in previous testing might not for one or two projects post migration. This can be especially true for an SRS report that's pulling data back from a project workspace. There may be some sort of corruption on that workspace or even in just one custom list that didn't show up on the other projects. Additionally, it's more than likely some projects won't get migrated because they were modified externally or open at the time of migration, or some such. Those projects will need to be migrated at some later point and linked up to their respective workspace.

Allotting time for some extra post-migration work should always be included in the project plan to allow you to deal with any potential problems, and mitigate application outages.

In the end you'll realize that, despite its initially challenging appearance, a Project Server 2007 migration is a lot like any other technical project. Simply remember:

- Consider the people – their fears and apprehension – and address them. Manage their expectations.
- Consider the environment. All of the preparation you did up front is critical to ensuring a successful production migration.
- Be a project manager. Plan the project, deal with issues, and prepare for risks – all of the normal tasks a good project manager should perform.

By looking ahead and planning out strategies for dealing with potential roadblocks, you'll not only have an easier time maintaining your patience, you'll probably have happier users and faster, broader user adoption. If you've done your due diligence to understand the potential problem areas in the project and given yourself plenty of time for practicing the work, your actual migration should offer relatively smooth sailing.

### **About the Author**

Victor A. Richardson, PMP, MCSD, MCITP, is a managing consultant at Catapult Systems in the EPM Solutions team and is the EPM local solution lead in Houston. He has over 14 years in the IT industry with solid project management skills developed from both a consulting environment as well as from various industry companies. Victor has a BBA in MIS from The University of Texas. Contact him at [victor.richardson@catapultsystems.com](mailto:victor.richardson@catapultsystems.com).



## ***Additional Resources***

### **Software Technology**

Here's a brief synopsis of the various technologies involved in a Project Server migration:

#### Project Server 2003

- Windows SharePoint Services (WSS) 2.0 (optional)
- Project Professional 2003
- Project Web Access (PWA)
- SQL Server 2000 or SQL Server 2005

#### Microsoft Office Project Server 2007 (MOPS)

- WSS 3.0 (not optional)
- Project Professional 2007
- Project Web Access (PWA)
- SQL Server 2000 or SQL Server 2005

Please note that there's no "mixing and matching" of Project Server 2003 and Project Professional 2007 or Project Server 2007 and Project Professional 2003. It's basically an all-or-nothing proposition.

### **Useful Links**

[Best Practices for Migrating to Project Server 2007](#)

[Migration Guide for Office Project Server 2007](#)

[Performing Migrations](#)

[Pre-migration Verification and Cleanup](#)

[Post-migration Tasks for Project Server 2007](#)

[Troubleshoot Migration for Project Server 2007](#)

[Upgrading Project Server 2003 to Project Server 2007](#)

[Sample Project Server 2003 to Project Server 2007 Migration Plan](#)

[Project Server Data Migration – Cheat Sheet](#)

[MS Project Reporter Blog](#)