



The Ballad of a Successful Upgrade

The Ballad of a Successful Upgrade

Kylie Kiser, Financial Services Firm

In 2016, we went through a major upgrade from CRM 2011 to 2016. Our highly-customized system has many workflows, plugins, integrations, and over 500 users involved who all needed to be trained!

The groundwork for the project was discussed and planned for several years but it wasn't until May 2016 when the brunt of the project work was done leading up to our upgrade weekend late October 2016. The work hours during upgrade weekend were less than expected and the production issues that needed to be resolved quickly were minimal – the upgrade was a huge success!

I want to share the lessons we learned so hopefully they can help you with your upgrade as well!

What Went Well

Super Users

The biggest contributing factor to the upgrade's success was the Super User team. We worked with Supervisors to select key users across multiple departments who assisted with testing, attended extra trainings, and were our first line of defense for questions during training and post-upgrade. Some of the Super Users even proactively solicited feedback from their peers which the project team then used to build enhancement requests and send newsletters with tips based on that feedback. Additionally, if any specific users were having trouble we could deploy a Super User to assist them face-to-face.

A few of these Super Users were also involved in the firm-wide training. We designed the training to be a general overview (not role specific) to account for scale (500 users!). We focused the training on navigation and stressed changes were only cosmetic. Supervisors were relied on to provide any group-specific information to their staff. We also had a few end-users lead the group sessions instead of trainers. This allowed users see their peers who use the system every day in the same way they did, and who could relate to their needs. This built confidence in the system and demonstrated how easy it was to learn.

We also worked with several users individually that we expected to be potential detractors (lovingly referred to as “Unhappies”). Before the upgrade, we gave those users special training sessions and private space to listen to their concerns and feedback. We also used those sessions to stress that their feedback was important but that they needed to be positive with other users. This ensured that any complaints during training and post-upgrade came to the project team instead of co-workers so we could address any issues and create learning opportunities. This also gave those “Unhappies” a sense of empowerment, knowing that we relied on their input and we were available to help them learn the system with confidence.

Consolidated Feedback

A major focus for the project team was ensuring that no one team member became overwhelmed with questions or feedback. To resolve this, we created an FAQ page on our intranet for announcements, training materials, and our list of Super Users (to call first). We also used a SharePoint list to capture testing feedback for quick and efficient triage. This allowed us to have one project team member go through and dismiss or reassign feedback as necessary so the team could focus on the most pressing issues.

Finally, we created a shared email inbox for handling direct questions from users. We directed users to email any questions they felt were not answered by the FAQ so they could get personal responses by the project team or our help desk. This prevented individual project team members from being overwhelmed with questions and gave us a way to track feedback.

Multiple Dry Runs

The upgrade timeline included multiple check points where we stopped work and rebuilt the development environment. This allowed us to iron out timeline details as well as make sure our solutions were capturing all the necessary requirements.

During our post-mortems after the upgrade, we realized several of the go-live day issues could have been resolved or prevented if there had been time for an additional dry run and testing period.

Open Development Environment

In addition to having a testing team of 30+ people, that included the project team and our Super Users, we invited all users to access the development environment. This allowed Supervisors the firsthand experience and time to test their procedures, revise documentation, and prepare staff as they saw fit. It also gave concerned users an opportunity to run through their daily tasks prior to go-live. This hands-on approach greatly helped relieve stress for the users who took advantage of it. During training classes, all users were asked to log in and spend half of the hour-long class walking through on their own.

Documentation

The previous system upgrade occurred in 2012 and went from version 4.0 to 2011. Substantial resources were spent documenting the current state of the system since that time. The effort included flow charts for the 250+ workflows and data mapping for the 20+ integrations. This allowed the project team to quickly gather lists of impacted systems and determine our test footprint.

What Could be Improved

Testing

We involved our Super Users to assist with testing since our system is so large and customized and has numerous integrations. This allowed our Super Users a behind-the-scenes look into the different areas of the business they may not be familiar with as part of their daily duties. This was great exposure for the Super Users to learn more about our system but was also a challenge as they did not always have the necessary level of comfort to be fully effective. This could have been improved by ensuring that our test instructions were easy to follow even for those with limited experience in the workflows, forms, or integrations they were testing. The test plan build out phase for the upgrade was a huge undertaking, and a valuable lesson we learned was how important it will be to keep those test plans and other documentation current going forward to make future upgrades easier.

Training

There were a few important things we learned from our upgrade training sessions:

1. It is important to explain to users the “Why” behind the upgrade. This helps center the audience and keep a positive tone. Training is as much about the content as it is about the perception of the content. It was very important to the project team to keep release communication positive and show users the value in the upgrade.
2. Users listen to their peers. Training sessions were primarily handled by our training team but a few sessions were hosted by Super Users. This helped engage users and build confidence in the upgrade.
3. Printed materials are important. We did not use any printed materials in the training sessions. We had several resources posted online that we asked users to print out if they felt they needed them, and many users did. Distributing hardcopies of those quick reference guides in advance would have helped with many of the initial questions and concerns we received from users.

Internal Momentum and Priorities

Of course, the upgrade was top priority for the CRM Team but that does not mean it was top priority for all teams. The lesson we learned was the importance of internal kick off meetings between different groups to ensure expectations and timelines are set and agreed upon. Additionally, it is very important to enforce those decisions and timelines throughout the upgrade.

What to Adopt for the Future

Documentation and Ownership

In the future, we will ensure we have full documentation for customizations, integrations, and system owners. We were well prepared for this upgrade because we already mapped out our data flow and integrations, but there were still some integrations with very little information

available that required extra resources to figure out who should be working on it, where to start with testing, and who should sign off on the changes.

Always be Prepared to Upgrade

Another time-consuming part of our upgrade was handling our 20+ integrations. Even though we had been using CRM 2011 for several years, the integrations were still using the 4.0 end points. Upgrading these end points could have been done over several years, but instead needed to be done in several months concurrent with the upgrade. This greatly increased our testing burden on release weekend.

Going forward the team will work on reviewing existing code and to identify what can be replaced with out of the box features and to reduce technical debt.

Final Thoughts

Thanks to all the team members involved, the upgrade was a huge success! We were able to execute the upgrade over a weekend with minimal disruption to staff. Users could begin working Monday morning because they were already trained and familiar with the “new system” and knew the changes were only cosmetic. I hope these lessons will help your upgrade succeed too!



 **D365UG
CRMUG**

www.crmug.com/join