

Keys to a Successful Software Rollout

This roadmap is designed to help your organization to achieve a successful rollout. While these tips will not guarantee success, they will greatly increase your chances of meeting your goals, and greatly decrease the possibility of failure.

1. Set Goals
2. Have a Plan
3. Build a Committee
4. User Process Documentation
5. Clearly Defined Procedures
6. Clearly Defined Responsibilities
7. Risk Management
8. Communication
9. Support
10. Training
11. Measuring Success
12. Follow Up With Your Committee
13. Vendor Relations

Set Goals

If you don't know where you're going, it's pretty difficult to know how to get there, or even to know when you have arrived. You should define what a successful rollout of @task looks like in your organization. This should be your long term goal. Then, identify the major milestones needed to accomplish this goal. This, then, becomes the basis for your rollout plan and also defines your measurement of success.

Have a Plan

As with any enterprise software, the process of implementation, configuration and user adoption can be time consuming and complex. The most effective way to verify that the rollout runs smoothly is to put a plan in place. The plan will include items such as requirements gathering, installation, administrator training, configuration, user documentation, user training, rollout, and follow-up.

This plan should include dates for each step of the process and be detailed enough to follow on a daily basis.

Rollout is generally accomplished using one of three methods: Pilot, Phased and Shotgun. All three methods have their benefits.

The **Pilot** rollout method involves a team of users that agree to use the software in a limited fashion to evaluate the features of the software and make a determination as to how best to implement the process across the organization. This is most effective when the pilot users are from different groups of the organization and make use of different features of the application. When the Pilot is completed, the rollout generally continues with either a phased or shotgun rollout.

The **Phased** rollout method consists of implementing the software for a single team, usually from the same group, then progressively rolling out to additional groups. The focus for each phase is to implement and evaluate the process. Improvements to the process can be made between phases.

The **Shotgun** method is most effective when used in a small group. The shotgun rollout involves rolling the software and process out to all users at once and making changes as the users provide feedback. The shotgun rollout is not recommended for large organizations.

The manner in which you roll out the software is key to the success of the implementation. In most cases, the phased or pilot approach is recommended.

Build a Committee

Don't try to go it alone. It is likely that @task will affect many, if not all, of the people in your organization. Include them in your requirements gathering, planning and configuration steps. There is nothing worse than investing time and money into getting a system up and running, only to find out that it doesn't meet the needs of 70% of your

users. If possible, have a representative member from each group that will be using @task, from every level of the corporate hierarchy, from individual contributors to executives, as part of your committee, to provide feedback and suggestions.

NOTE: Don't overdo this. While it is helpful to ask for feedback and suggestions, it can easily get out of hand. Make sure that you don't let the committee stall the project. The point is to make sure the primary needs are met, not to try to make @task everything to everybody.

User Process Documentation

@task is designed to be customized to meet the needs of your organization. So, when you are done customizing @task to fit your process, the @task user documentation may be too detailed, or too broad, to fit your needs. One of the most effective things you can do to ensure a successful adoption, is to provide your users with customized documentation on how to use @task to support your organization's processes. This can be as simple as a few screenshots with some text describing where to click and what information to enter, or as detailed as several pages describing each field and how the information they are manipulating is affecting the overall project status picture. Some companies have even created customized Flash tutorials as part of their internal training efforts.

Clearly Defined Procedures

Procedures that are clearly defined and consistent will not necessarily guarantee success, but lack of procedures will almost certainly guarantee failure. Procedures, in this context, refer to both the procedures for use of the software, as well as the plan for implementation. The procedures must be clearly defined and communicated to the team.

Clearly Defined Responsibilities

It is important to know who is responsible for what, such as in the case of internal support or system administration. Who will be responsible for creating custom data fields or creating templates? Who will be the first line of defense for internal product and process questions? Clearly communicating this with your users will make them feel more comfortable in using the product.

Risk Management

Identify the obstacles that you may face in opposition to the rollout and adoption of the software and address those risks up front. These may include users that are devoted to a different process or product, hardware concerns, training concerns, or getting consensus from key stakeholders on processes. Identifying and addressing these risks, before they become obstacles, will both smooth out the potential speed bumps in your rollout, as well as increase your chances of success.

Communication

No matter how detailed your processes are, how well planned your rollout is, or how much support you have from your users, if you don't communicate with your users, the implementation is likely to fail. Good communication is essential to your project rollout.

Support

@task provides online support for using @task, however, since your use of @task will be customized, you will need to decide how support will be handled for your process. Will it be you, or a team of “power users”, or an administrative staff? It will be important to communicate your internal support channel to your users so they know where to go for help with the processes. The easier it is for users to get help, the more likely they will be to use the software.

Training

Training is essential. No matter how simple you make your process, users will be creative and find ways to be confused. A confused user will quickly become a non-user, or worse yet, a chaotic user. Training can, and should be, a combination of internal and external training. It is a good idea to have an AtTask consultant train your administrators, either on-site or at a boot camp in Utah. This allows them to know what the system can do and how to do it. The managers and users can be trained internally by your own staff after the processes have been defined and the system configured. This will allow you to keep the training simple and train them only on what they need to know.

Keep in mind that users often have different learning styles. Some learn better using self-paced CBT packages or manuals, while others need hands-on experience and try to cater your training accordingly.

Measuring Success

How will you know if your rollout was successful? Remember the goals you set above? They are the measurement by which you can measure success. Have you accomplished the goals you set? You may want to define more detailed metrics, such as “reduce time-to-market by two weeks”, or “reduce the cost of projects by \$2000”, or something similar. This may sound like an ROI statement or it may be as simple as providing a rollout report of all projects in the organization to the executive. Whatever you define as a metric, make sure it is discrete and measurable.

Follow Up With Your Committee

“No plan survives first contact with the enemy” - Helmuth von Moltke

Implementation of the software is one thing. Successfully using it is another. Follow-up with your users at each level of the corporate hierarchy to make sure that their needs are

being met. You will likely receive feedback on processes and configuration that you can use to improve. Remember, if the users aren't happy, there is always a chance of mutiny. These follow-ups are key to maintaining good communication with your user base and provide the basis for changes to your plan that ensure you can reach your goals.

Vendor Relations

As with any enterprise software rollout, it is key to maintain a solid relationship with the vendor. AtTask is committed to your success. As such, we want to be a resource to you to assist in resolving issues and providing training and consultation that you need to be successful.

Conclusion

Implementing an enterprise software is no small task and it can feel overwhelming. Just remember to take it one step at a time, follow these tips and you'll find an increased measure of success. And remember, that we are here for you.

If you would like to speak to an @task consultant regarding adoption services that AtTask provides, please call us at (801) 373-3266, option 5.