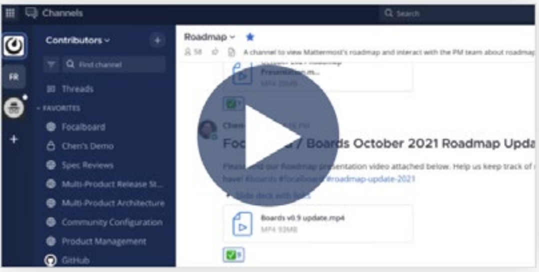
How to upgrade to Mattermost v6.0

Thanks to a lot of hard work and excellent feedback from the open source community, Mattermost v6.0 is now generally available! This is a major platform upgrade with new capabilities and wider use case coverage, which many server administrators will be eager to take advantage of. Due to the scope of the changes introduced in v6.0, admins will want to plan ahead for a bit of downtime, especially for larger database migrations. Here's a rundown of what's new and what to expect.

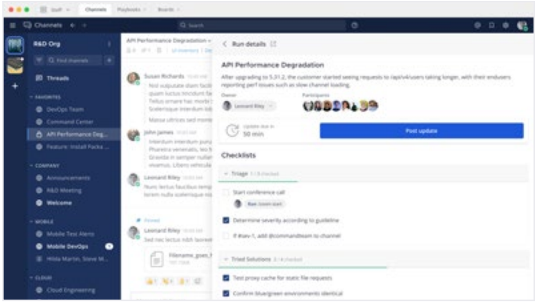
New Features: Integrated Playbooks and Boards

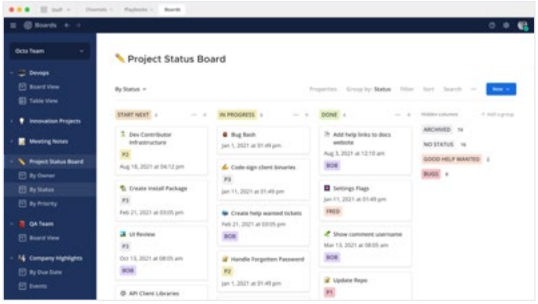
This release introduces a new era for collaboration across the software development lifecycle. In addition to core Channels for messaging, you now have Playbooks for structured processes, and Boards for collaborative project planning and management. We've also expanded the App Framework so you can build and host even more custom integrations to suit your particular use case. These tools are all available in one integrated UI so your team can avoid switching context to get status updates, task assignments, and other workflow touchpoints.

Watch our demo to see how Mattermost provides visibility and control for common developer workflows.


Channels
Contributors
Find channel
Threads
FAVORITES
Focalboard
Chen's Demo
Spec Reviews
Multi-Product Release St...
Multi-Product Architecture
Community Configuration
Product Management
GitHub
Roadmap
A channel to view Mattermost's roadmap and interact with the PM team about roadmap
Boards October 2021 Roadmap Upda
Boards v6 9 update.mp4

Other great reasons to upgrade include an improved user experience, bug fixes, security fixes, and mobile app compatibility. Read the full release notes to learn more.





Tips for a Smooth Upgrade

To help plan ahead, please read through this entire blog post before starting, and please see the Upgrade Guide in our documentation.

Mattermost v6.0 is a major upgrade and introduces database schema changes in order to improve database and application performance. This affects both MySQL and PostgreSQL databases. Depending on the size and configuration of your database, the migration may take a significant amount of time and may lock the tables for posts.

To minimize the impact of the migration, we suggest that you first backup your database and test the backup, then run several of the required queries before the actual upgrade. Read the detailed analysis and data on query run times for a representative database migration of 10M+ posts or for 70M+ posts.

Also important to note: because of a change to the clustering code, you will be unable to have a Mattermost v5.x database running at the same time as a v6.0 database. Within a cluster, nodes from different versions cannot run, so scheduled downtime is especially important for clustered environments. Read more in the important upgrade notes forum post.

Upgrade Procedure Overview

1. Review the Mattermost Changelog and Important Upgrade Notes in the docs
2. Backup your database and configuration files*
3. Download the new version of the Mattermost Server
4. Extract the files for the new Mattermost Server version
5. Stop your current server from running
6. Copy the new files to your install directory and remove the temporary files
7. Start your Mattermost server

*If you are running in a high availability environment, you will need to follow steps 2-7 for each server or node in the cluster.

General Upgrade Best Practices

- Ensure you've read the Mattermost Changelog and Important Upgrade Notes
- Test an upgrade in a test or staging environment
- If you are planning on downtime, ensure you plan enough time to finish and test the upgrade
- Communicate the downtime to your users ahead of the scheduled time
- Have a rollback plan just in case there is an issue (please see our detailed downgrade documentation)

Upgrade Options

A Mattermost v6.0 server will run significant database schema changes that can cause a large startup time depending on the dataset size. Zero downtime will not be possible, but can be minimized depending on your environment and availability requirements. You can review all the specific database queries and their expected run times here.

1. Small Mattermost instances (Low effort, long downtime)

Use this option when you have fewer than 1 million posts in the database. At most 5 minutes of downtime is expected for MySQL, and less than 1 minute for PostgreSQL.

This is the process of starting a v6.0 server normally. During this migration process, various tables will be locked which will render those tables read-only during that period. Once the server finishes migration you can start the application.

2. Medium to Large instances (Medium effort, medium downtime)

Use this option when you have more than 1 million posts in the database. Long running queries are scheduled for off-hours to reduce downtime. With 10 million posts in the database, at most 30 minutes of downtime is expected for MySQL, and less than 1 minute for PostgreSQL. Downtime increases roughly linearly with the size of the Posts database table.

This process will require SQL queries to be executed manually on the server (to avoid causing a table lock, you can choose to use the pt-online-schema-change tool for MySQL). For Postgres, the table locking is very minimal. The advantage is that since there are a lot of queries, you can take the time to run individual queries during off-hours. All queries except #11 are safe to be executed this way. Then the normal method in (1), above, can be followed.

3. Highly available environments (High effort, low downtime)

Use this option when you use a highly available environment. At most 1 minute of downtime is expected both for MySQL and PostgreSQL.

This process requires everything of (2), plus it tries to minimize the impact of query #11. You can do this by following (2), then start v6.0 along with a running v5.x server, while monitoring the application logs. As soon as the v6.0 application starts up, you'll need to bring down a v5.x node. This is a delicate and hands-on process, but minimizes the downtime to only a few seconds.

The major thing to note with all these options is that if you are running in a highly available environment, a v6.0 server is incompatible with a v5.x server in a cluster. Therefore, while upgrading, it is required that no other v5.x server runs when a v6.0 server is brought up.

Resources to Use in Preparation for Your Upgrade

Here are all the resources we've referenced in this document:

- Upgrade docs
- Downgrade docs
- Changelog
- Important upgrade notes (docs)
- Important upgrade notes (forums)
- Mattermost v6.0 DB Schema Migrations Analysis
- Release notes blog

Upgrade to Mattermost v6.0 Today

To preview the new Mattermost experience with Playbooks and Boards, log on to the Mattermost Community Server. It's where all the open source work on the platform happens, where you can always see some of the latest and greatest updates, and where you can ask questions and make contributions of your own.

When you're ready to upgrade, please follow the comprehensive upgrade guide.

Staff and community members are available to answer any questions you have in the Peer-to-Peer help channel on the Community Server.

For our existing customers, please reach out to your Customer Success Manager if you need any assistance.