

# Promoting a Registry Slave

In a [replicated IceGrid deployment](#), you may need to promote a slave to be the new master if the current master becomes unavailable. For example, this situation can occur when the original master cannot be restarted immediately due to a hardware problem, or when your application requires a feature that is only accessible via the master, such as the [resource allocation](#) facility or the ability to modify the [deployment](#) data.

To promote a slave to become the new master, you must shut down the slave and change its `IceGrid.Registry.ReplicaName` property to `Master` (or remove the property altogether). On restart, the new master notifies the nodes and registries that were active before it was shut down. An inactive registry or node will eventually connect to the new master if its default locator proxy contains the endpoint of the new master registry or the endpoint of a slave that is connected to the new master. If you cannot afford any down-time of the registry and want to minimize the down-time of the master, you should run at least two slaves. That way, if the master becomes unavailable, there will always be one registry available while you promote one of the slaves.

With Ice versions 3.5.0 and prior, a slave synchronizes its database upon connecting to the new master, therefore it is imperative that you promote a slave whose database is valid and up-to-date. To verify that the promoted master database is up-to-date, you can start the new master with the `--readonly` command-line option. While this option is in force, the new master does not update its database, and slaves do not synchronize their databases. You can review the master database with the IceGrid [administrative tools](#) and, if the deployment looks correct, you can restart the master without the `--readonly` option to permit updates and slave synchronization.

Starting with Ice 3.5.1, a check has been added to ensure that an out-of-date master database cannot overwrite a newer slave database. In such a scenario, an error message is printed out on the consoles of the IceGrid master and slave, and the slave connection to the master is aborted. You can still connect to the slave and master with the IceGrid [administrative tools](#) to verify their databases. Once you determine which database to keep, you can restart either the slave or the master with the `--initdb-from-replica=<name>` option to initialize the out-of-date database with the correct version. This check only works when both the master and slave use Ice 3.5.1 or later.

Note that there is nothing to prevent you from running two masters. If you start two masters and they contain different versions of the deployment information, some slaves and nodes might get updated with out-of-date deployment information (causing some of your servers to be deactivated). You can correct the problem by shutting down the faulty master, but it is important to keep this issue in mind when you restart a master since it might disrupt your applications.

## See Also

- [Registry Replication](#)
- [Resource Allocation using IceGrid Sessions](#)
- [Using IceGrid Deployment](#)
- [icegridadmin Command Line Tool](#)