

How should I archive my historical data?

Questions addressed:

Page 1:

- What is historical data?
- What impact does historical data have on performance?

Page 2:

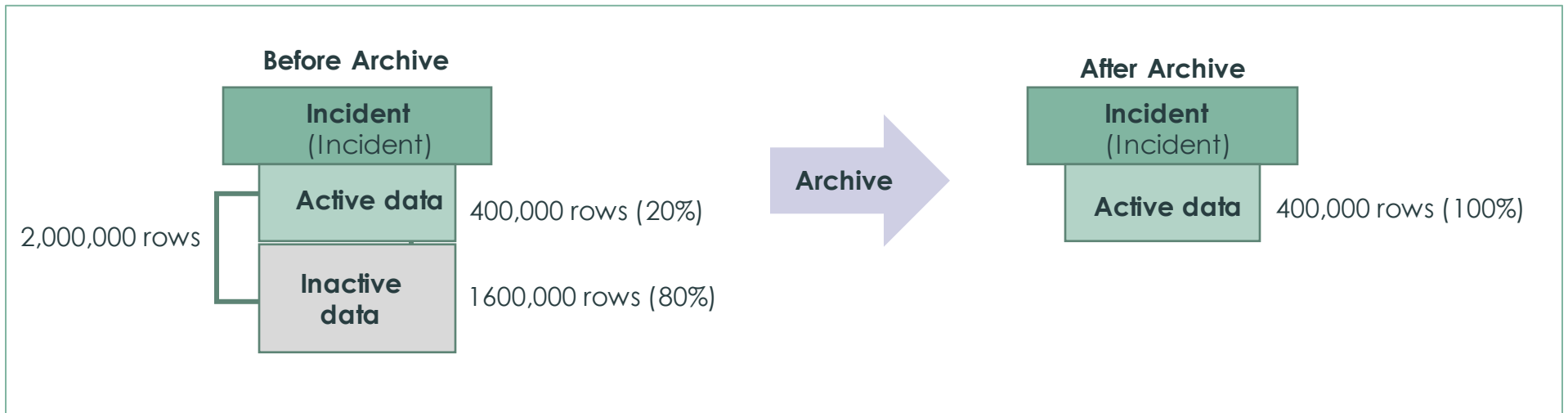
- How should I archive historical data?
- Who can set up archive rules and can the rules be customized?
- What happens if I need to access archived data?

What is historical data?

The longer an instance runs, the more likely it is to accumulate data that's no longer relevant. For example, task records from two years ago are typically less relevant than currently active tasks. You may not be able to delete this data because you need it for auditing or for historical purposes.

What impact does historical data have on performance?

Old data may eventually cause performance issues by consuming system resources and slowing down queries and reports. Archiving historical data will ensure you get the optimal performance and keep core operations also running smoothly.



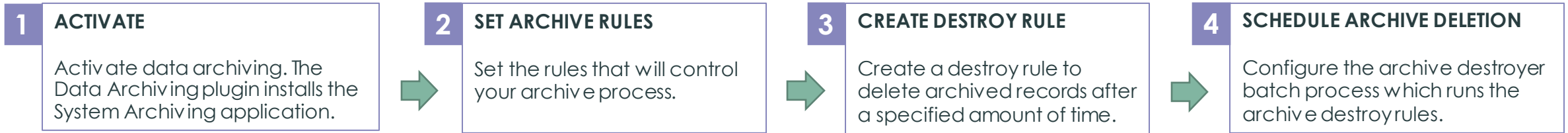
Insider's Tips:

- Archiving is mostly beneficial when the table is large and actively used. The best candidates include task (incident, sc_task, problem, change_request, etc.).
- Emails that are attached to incidents will not be automatically archived with those incidents. There is a separate tool called [Email Retention](#) that will allow you to archive and eventually destroy email messages that you no longer need or when your email table is excessively large.

How should I archive my historical data? (Continued)

How should I archive historical data?

Archive historical data with [Data Archiving](#), which is enabled by default in ServiceNow. Archiving is a scheduled process that runs every hour and executes all archive rules one by one to remove them from immediate access and free system resources. (Note: Archiving is not a solution to reduce your database size.)



Who can set up archive rules and can the rules be customized?

Administrator roles can set up archiving rules. Archive Rules can be set-up from Archive Rules section. Only one archive rule can be set-up per table. Rules can be set-up for both OOB tables and custom tables.

You can set archive rules to control how many records an archive rule processes per batch job, how much sleep time is between each archive rule batch job run (in seconds), and the maximum number of batch jobs to run within the hour.

By default, archive rules are set for 100 records for each batch job, sleeps 1 second between batch jobs, and runs 10 batch jobs in an archive run (every hour). You can customize these settings by [updating system properties](#), including changing the process not to run in peak hours.

With the Paris release, archiving is now a producer/consumer model that can run parallel on multiple nodes. Records to be archived are now recorded in a queue table in batches. Pre-Paris archiver properties don't work with the Paris release.

Insider's tip:

It's better to use more batches than increase the size of the batch. This helps avoid replication lag.

What happens if I need to view data that has been archived?

You can view archived data, change an archive schedule, and restore archived data.

When you restore a record, the instance inserts it back into the primary table and flags the record as having been restored in the log. It's important to note that archive rules will not archive restored records. In order to archive a restored record you must manually archive the record.

With the [Paris release](#), there is an option to re-archive restored records again after a period of time.

Contact your account team who can reach out to SWAT to assist you with any archiving database issues.

[Contact us](#) if you have any questions on this topic or you would like to be a contributor to future ServiceNow best practice content.