datadir are automatically appended to the innodb_directories value regardless of whether the innodb_directories variable is defined explicitly. Those directories and their subdirectories are scanned at startup for undo tablespaces files. An undo tablespace file moved to any of those directories is discovered at startup and assumed to be the undo tablespace that was moved.

The default undo tablespaces (innodb_undo_001 and innodb_undo_002) created when the MySQL instance is initialized must reside in the directory defined by the innodb_undo_directory variable. If the innodb_undo_directory variable is undefined, default undo tablespaces reside in the data directory. If default undo tablespaces are moved while the server is offline, the server must be started with the innodb_undo_directory variable configured to the new directory.

The I/O patterns for undo logs make undo tablespaces good candidates for SSD storage.

Configuring the Number of Rollback Segments

The innodb_rollback_segments variable defines the number of rollback segments allocated to each undo tablespace and to the global temporary tablespace. The innodb_rollback_segments variable can be configured at startup or while the server is running.

The default setting for innodb_rollback_segments is 128, which is also the maximum value. For information about the number of transactions that a rollback segment supports, see Section 15.6.6, “Undo Logs”.

Truncating Undo Tablespaces

There are two methods of truncating undo tablespaces, which can be used individually or in combination to manage undo tablespace size. One method is automated, enabled using configuration variables. The other method is manual, performed using SQL statements.

The automated method does not require monitoring undo tablespace size and, once enabled, it performs deactivation, truncation, and reactivation of undo tablespaces without manual intervention. The manual truncation method may be preferable if you want to control when undo tablespaces are taken offline for truncation. For example, you may want to avoid truncating undo tablespaces during peak workload times.

Automated Truncation

Automated truncation of undo tablespaces requires a minimum of two active undo tablespaces, which ensures that one undo tablespace remains active while the other is taken offline to be truncated. By default, two undo tablespaces are created when the MySQL instance is initialized.

To have undo tablespaces automatically truncated, enable the innodb_undo_log_truncate variable. For example:

```
mysql> SET GLOBAL innodb_undo_log_truncate=ON;
```

When the innodb_undo_log_truncate variable is enabled, undo tablespaces that exceed the size limit defined by the innodb_max_undo_log_size variable are subject to truncation. The innodb_max_undo_log_size variable is dynamic and has a default value of 1073741824 bytes (1024 MiB).

```
mysql> SELECT @@innodb_max_undo_log_size;
+----------------------------+
| @@innodb_max_undo_log_size |
+----------------------------+
|                 1073741824 |
+----------------------------+
```

When the innodb_undo_log_truncate variable is enabled: