

# Archiving



## Tip

The following webinar recordings available on YouTube also serve to familiarize yourself with the basics of configuring archiving:

- [Historian - putting values into a historical database](#)

We recommend the following blogs::

- [Archiving in SCADA and MES systems](#)
- [Archiving in SCADA and MES systems, part 2](#)
- [Enterprise features of archiving in SCADA and MES systems](#)
- [Enterprise features of archiving in SCADA and MES systems, part 2](#)
- [Enterprise features of archiving in SCADA and MES systems, part 3](#)
- [Archive and PostgreSQL depository databases](#)
- [Migrating depository databases to PostgreSQL in practice](#)
- [Arcsynchro and PostgreSQL depository databases](#)
- [Continuous integration in an archive or how to do a sawtooth diagram](#)
- [Archive - migration of historical databases from Oracle to PostgreSQL](#)
- [Unexpected surprises in archiving](#)

## Architecture

Archiving in D2000 is provided by the [D2000 Archiv](#) process. It has historical values as its children.

Most applications have a single archive process (SELF.ARC) under which all historical values are located. In simple systems, this process has an autostat enabled and is started by the [D2000 Server](#) when the D2000 system starts.

The D2000 Archiv process stores data in the SQL database. PostgreSQL, Oracle, Sybase SQL Anywhere, and Microsoft SQL are supported (SQLite and MySQL databases are also experimentally supported). The PostgreSQL database is recommended. For more information, see the chapter [Data Archiving in D2000 System](#).

Since historical data is one of the most valuable items in SCADA and MES systems, the archive process is one of the first to be configured as redundant. On two (or more) computers, a pair of watchdog + D2000 Archiv process is configured using the [SCS Config](#) utility with the parameter */W<name>*, *<instance>* indicating the name and number of the instance (e.g. */WSELF,1* or */WSELF,2*) and with the parameter */S*, */RD*, or */RF* specifying the server or redundancy group to which the archive connects (see [Starting user processes](#)).

Each of the instance archive processes archives the same historical values (children of SELF.ARC). The D2000 Server process controls which instance is active (to which read requests are directed). Other instances are passive.

Instance archives are also useful when performing archive server maintenance (patching Windows, SQL database, firmware, D2000, etc.) without losing data.

The [Arcsynchro](#) utility, which D2000 Archive can run automatically at startup (see the [ArcsynchroAuto](#) archive parameter), is used to synchronize data between archives (patching a hole after a failure of one of the instance archives or migrating archive data between different SQL databases).



## Advanced tips

Querying and setting the active instance is possible with the [INSTANCE](#) command (e.g. *INSTANCE GET ACTIVE SELF.ARC* or *INSTANCE SET ACTIVE SELF.ARC 2*). Information about whether the archive instance is active is also in the *Active* column in the system structured variables [SV.\\_System\\_ArchivPerformance](#) and [SV.\\_System\\_Proces](#).

Archive processes can also operate in load-balancing mode when multiple instances are active and the [D2000 Server](#) process distributes data reading requests between them. The load-balancing mode can be turned on with the command [SET\\_LOAD\\_BAL\\_MODE](#).

## Types of historical values

### Primary historical values

[Primary historical values](#) archive the values of specific D2000 objects (whether they are object values from communication, calculated values, manually entered, or other). Primary archiving can be on-change or periodic.

### Statistical historical values

[Statistical historical values](#) calculate a statistical function over the values of other historical values (primary, but also other statistical or calculated). Statistical historical values are periodic.

## Calculated historical values

[Calculated historical values](#) recalculate the values of other archives (primary, but also other statistical or calculated) with an arbitrary expression. They can be on-change or periodic.

## Historical values filled from a script

[Historical value filled from a script](#) (value storage) serves as a store of values that are not created by direct archiving of values of another object in the system, statistical calculation or calculation in a calculated historical value. Filling in values is only possible from the ESL script or manually through the [D2000 HI](#) process. The value storage can be variable or periodic. The periodic value storage can have value playback enabled - in real time it publishes the values that were previously inserted by the script (e.g. production planning data).

The [UPDATEARCHVAL](#) and [INSERTARCHAR](#) actions are used to fill in data from the script.

Note: These script actions can also write to primary, statistical, and calculated archives. Attention - repeated recalculation of statistical and calculated archives will overwrite manually entered values with new calculated values!

**Important:** source historical values must be parented by the same archive process (SELF.ARC by default) as the statistical and calculated historical values that use them. This limitation is the main reason why most systems only have a single SELF.ARC archive process.

## History depth

Each historical value can have its own history depth configured. The D2000 Archive process periodically deletes values older than the history depth every few hours in such a way that at least one value older than the history depth always remains.

In addition, it is possible to configure in the D2000 Archiv process the so-called [depository databases](#) that provide time-unlimited archiving for those historical values that have the *Depository* option enabled.

## Simple and structured archives

Historical values can be simple or structured.

Primary historical values can archive a column of a structured variable or the entire structured variable (this is not recommended).

Statistics historical values can calculate statistics over a columnar or structured archive (and optionally publish the value to a column of the structure or to the entire structure).

Calculated historical values can calculate an expression over columnar historical values (and optionally publish the value to the structure column).

More information and an example can be found in the document [Working with Structures](#).

## Periodical archives

The archiving period for all types of periodic historical values must be such that the 24-hour interval can be divided into an integer number of periods (i.e. 1 minute or 15 minutes or 2 hours but not 7 hours).

Note: Using the actions [CALCSTATFUNC](#) and [CALCSTATFUNCARR](#), it is possible to calculate on-demand a statistical function over any historical value and for an arbitrary period (e.g. per calendar month) from the ESL script.

## Recalculations and calculations on demand

- With the [CALCONDEMANDSTAT](#) action, any statistical or calculated historical value can be recalculated on demand from the ESL script.
- With the [CALCARCHEXPR](#) action, an arbitrary expression using existing historical values can be calculated on demand from an ESL script.
- With the [CALCSTATFUNC](#) and [CALCSTATFUNCARR](#) actions, it is possible to calculate on-demand a statistical function over any historical value and for an arbitrary period (e.g. per calendar month) from the ESL script.
- With the [RECALC](#) command, it is possible to request the recalculation of several archived values according to the name mask (and all historical values calculated from them).

## Reading from the archive

Reading from the archive can be triggered in several ways:

- By opening a [graph](#) - the D2000 HI process automatically generates requests to read all the values displayed in the graph.
- By opening a picture in [history mode](#).
- By actions for reading from the archive: [GETARCHVAL](#), [GETARCHARR](#), [GETARCHARR\\_TO\\_CNT](#), [GETARCHROW](#), [GETARCHCOL](#) and [GETARCHSTRUCT](#).
- By requesting historical values from outside the D2000 system:
  - Reading via the [D2000 Gateway](#) from another D2000 system as a response to the [GETOLDVAL](#) command.
  - Reading via [D2000 Kom](#) process (server protocols, e.g. [IEC 870-5-104 Server](#)).
  - Reading via [D2000 OBJApi](#) function [GetArchivData](#).
  - Reading via [D2000 VBApi](#) (from Microsoft Visual Basic or Microsoft Excel programs) with functions [VBApiGetArchData](#) (simple historical value) or [VBApiGetArchDataItem](#) (item of a structured historical value).
  - Reading via [D2000 Java API](#).
  - Reading via [D2000 ODBC driver](#),
  - Reading via [D2000 REST API](#).

