

# Machbase Features

## Support for Various Table Structures

---

Machbase provides four table types for users according to one's usage. (Tag, Log, Volatile, Lookup)

This is because client requirements for storing sensor data are very diverse and any one business does not have just one specific data pattern.

Therefore, it's important to understand these business requirements and select the appropriate tables for them.

The table below shows the characteristics of each table.

Index

---

- Support for Various Table Structures
- Hardware Support for Various Sizes
- Tag Analyzer: Data Visualization Solution Support
- Write Once, Read Many
- Lock-free Architecture Support
- High Speed Data Storage
- STREAM Function Support
- Configuring Real-Time Index
- Real-Time Data Compression
- Outstanding Query Performance
- Time Series Data Characteristic s SQL Syntax Support
- Supports Text Search Function
- Optional Deletion Support

- Automated

Data

Collection

Table Type	Tag Table	Log Table	Volatile Table	Lookup Table
PURPOSE	Optimized for processing sensor time series data in the form of <i>&lt;sensor name, time, sensor value&gt;</i>	Optimized for processing PLC log time series data  (text included)	Real-time processing of volatile memory data	Manages master data that can be stored permanently
DESCRIPTION	Used when storing sensor data at high speed, extracting corresponding data at high speed, or creating statistical tables in real-time  Mainly stores real-time sensor data	Used when storing log data including text and analyzing it in the form of general DBMS  Mainly stores historical user data	Used when Insert, Delete, Update, Select is required for memory-based performance (tens of thousands per second)  <b>All data is lost when the system is shut down.</b>  Mainly used for key-value based monitoring.	Used to permanently store user-editable master data.  SELECT has high-speed performance,  but INSERT, UPDATE, and DELETE provide disk-based performance.
TABLE STRUCTURE	<i>&lt;Sensor name, time, sensor value&gt;</i> is the basic type, with the ability for assigning additional columns.	Any schema possible	Any schema possible (Primary Key can be assigned)	
INSERT (INPUT) PERFORMANCE	Millions per second	Millions per second	Tens of Thousands per second	Hundreds per second
SELECT	Sensor Name + Limited Time Range	All inquiries possible		
DELETE	Real time deletion of data <u>before</u> an arbitrary point	Real time deletion of arbitrary point / interval data	Primary Key Record Delete Support ( Primary Key Designation Required)	
UPDATE	Not supported ( Only Metadata column is editable)	Not supported	Primary Key Update Support ( Primary Key Designation Required)	
STORAGE SIZE LIMITS	Disk limit		Memory limit	
INDEX STRUCTURE	Three-step partitioning real-time index ( default creation)	LSM index	Red / Black memory index	
STREAM SUPPORT	arget only (save target)	Both source / target (read and save target)	Not possible	
CONSIDERATIONS	Consider enough storage to erase historical data	Consider as temporary storage for Tag input	Consider memory limit	

## Hardware Support for Various Sizes

Machbase provides various product editions according to user environments as listed below.

### Edge Edition

This product runs on small Edge devices running on ARM or Intel ATOM CPUS.

However, even in such small devices, Machbase can be useful for storing and filtering tens of thousands of sensor data per second. It is mainly required for storing various sensor data at the terminal stage of robots and factory production lines, buildings, etc. at high speed and high capacity.

### Fog Edition

This product is used to achieve high-speed data processing on a single server.

It runs on Windows or Linux operating systems based on Intel x86 CPU and provides very fast sensor data storage and analysis that other DBMSs can not provide.

In most cases, it is used to store real-time data input from hundreds or more of edge devices and to perform secondary analysis.

### Cluster Edition

This product was developed for the purpose of storing large-scale sensor data for large manufacturing plants.

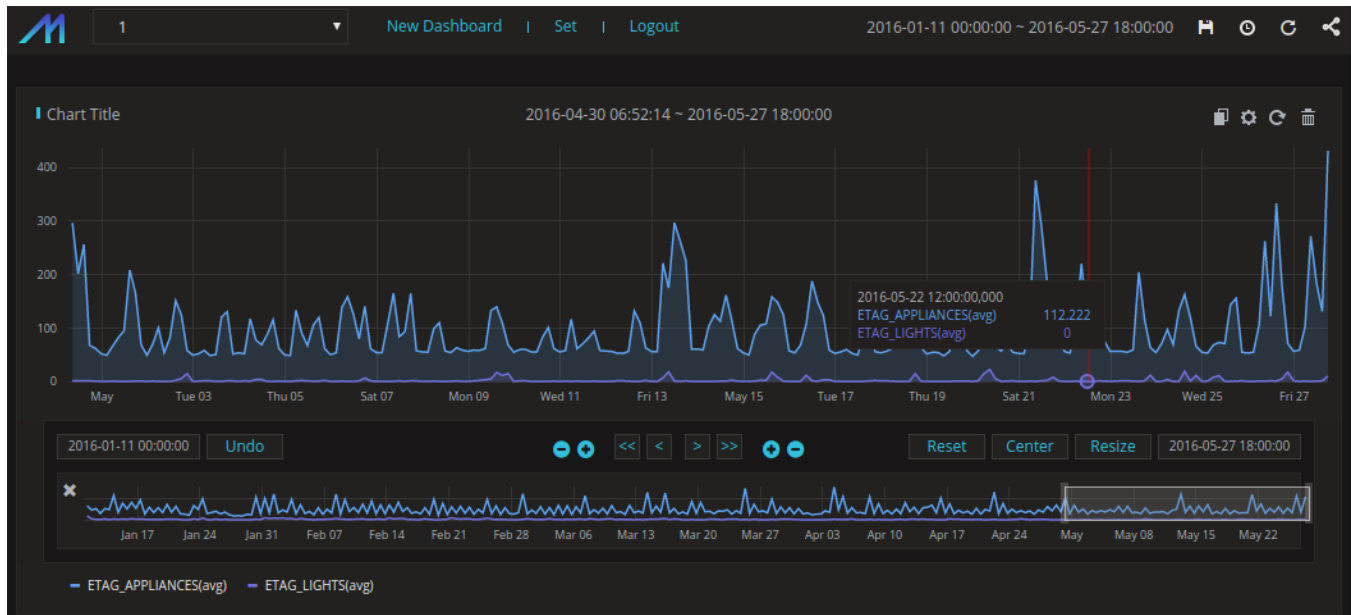
A number of physical servers operate in clusters to store more than 10 million data per second in semiconductor or display, power generation, and steel production processes. It is used in an increasingly data-rich environment where data capacity needs to be continuously maintained.

## Tag Analyzer: Data Visualization Solution Support

Machbase provides real-time visualization of hundreds of millions of sensor data stored in Machbase (since Version 5).

In other words, an arbitrary tag ID is designated, and the trend chart for the period in which the ID is input can be instantaneously checked on the web-based basis.

In addition, it provides not only simple tag data but also a statistical chart during that period, so statistical analysis is possible beyond simple visualization.



## Write Once, Read Many

Sensor data is rarely edited or deleted once it is entered into the database.

Therefore, Machbase is designed so that once the key time series data is inputted to maximize the characteristics of the machine data, an UPDATE can not occur.

Once the log data has been entered, it cannot be altered or deleted by malicious users, so there should be no concerns.

## Lock-free Architecture Support

The most important aspect in sensor data processing is that data input, update, delete operation and read operation should be processed as independent as possible without conflicts.

Because of this, Machbase is designed not to [allocate](#) any locks for the SELECT operation, and it is designed with a high performance structure that never conflicts with the operation of input or deletion changes.

Therefore, even when hundreds of thousands of data are entered and some of them are deleted in real time, the SELECT operation can speed up statistical operations on millions of records.

## High Speed Data Storage

Machbase provides data storage performance that is exponentially faster than conventional databases. Even if there are many indexes in a specific table, data can be received from at least 300,000 to at most 10 million per second.

This is possible because Machbase is designed to optimize time series data.

## STREAM Function Support

---

Starting from Machbase Version 5, Edge and Fog Editions provide STREAM functionality to support real-time data filtering.

This STREAM performs a condition evaluation on real-time data input in DBMS at high speed and transmits the result to an arbitrary table. This function is very useful for generating a warning when the value of a certain sensor exceeds a specific range or real time evaluation of internally input data is needed.

## Configuring Real-Time Index

---

Machbase innovatively improves on conventional database structure (where the more indexes you have the slower your data insert performance is) and can build indexes in near real-time, even with hundreds of thousands of data inserts per second.

This feature is a key technology for analyzing time series data, such as machine data, because it provides a powerful functional foundation for instant retrieval of actual data as it occurs.

## Real-Time Data Compression

---

The characteristic of time series data such as machine data is that data is generated constantly. This inevitably means that not only will the storage space of the database become eventually inadequate, but it will not have enough data to process.

In particular, although conventional databases input data at a high speed, as the number of indexes increases, the occupied data space also greatly increases. Therefore, conventional databases are quite unsuitable for storing and analyzing machine data. Machbase uses two innovative real-time compression techniques to compress and store up to a hundred times more data without any setbacks in performance.

### Logical Real-Time Data Compression Technology Support

First, Machbase supports logical real-time data compression technology.

This is based on the data redundancy of the machine data derived from a column-type database. It is an innovative technique to reduce the data storage space by coding redundant data as the number of data having the same value increases, which allows high redundancy data to be compressed hundreds of times the original amount.

### Physical Data Compression Technology (Patented Technology)

The second is Machbase's patented physical data compression technology.

This is a technology that reduces the amount of physical data to be stored by dividing a physical data block to be stored in a disk into a predetermined size partition, compressing it into a disk separately, and further reducing the I/O cost caused by the system. This helps to increase the efficiency of the storage space by compressing the actual logically compressed data once more.

## Outstanding Query Performance

---

The innovative and technological superiority of Machbase is that the search and statistical analysis of millions or tens of millions of previously stored historical data is very fast, even with the simultaneous input of hundreds of thousands of data per second.

This is possible because of Machbase's own indexing technology that provides superior performance for both insertion and analysis, and will play a key role in real-time business decision making. Unlike conventional databases, Machbase can process two or more indexes in a single query, which can be expected to perform several times faster when processing data in parallel. The following is an example of using two or more indexes in a single query.

```
SELECT * FROM table1 WHERE c1 = 1 and c2 = 2;
```

## Time Series Data Characteristics SQL Syntax Support

---

In the case of sensor data, the newest data is several times more valuable than the older data, and also the "access frequency" of the latest data is characterized as being several times more compared to old data.

For this reason, Machbase supports time series data features through two types of tables: Tag and Log.

### Log Table

The log table supported by Machbase has the following features.

#### First, it automatically saves input time

Whenever a record is stored in the database, a timestamp in nanoseconds is stored as a field called `_arrival_time`. This means that all records stored by Machbase can be searched for or given condition on a time basis.

#### Second, it prioritizes lookup of recent data

When retrieving data, the latest time is output before the old time. That is, when SELECT is performed, the latest data is output first. The result is the descending sort based on the `_arrival_time` column mentioned earlier.

#### Third, the DURATION keyword

The DURATION keyword is provided to enable quick lookup of specific time range data based on input time. In the case of machine data analysis, these characteristics are provided at the SQL level because they often specify a specific time range. This makes it easy to analyze data without stating "where" clause to complex time operators.

```
-- Example 1) View data statistics from 10 minutes ago
SELECT SUM(traffic) FROM t1 DURATION 10 MINUTE;

-- Example 2) View data statistics for 30 minutes from 1 hour ago
SELECT SUM(traffic) FROM t1 DURATION 30 MINUTE BEFORE 1 HOUR;
```

### Tag Table

The tag table that is supported from Machbase 5.0 has the following features.

#### First, high-speed TAGID / time condition search performance

The tag table is excellent at any time and any ID based search performance. It boasts ultra-fast data extraction performance that can not be achieved with existing RDBMSs, ensuring the same speed even when billions of sensor data are stored.

#### Second, the high-speed tag data input

The tag table supports high-speed data input. As in the previous log table, data can be input without difficulty even with the input of hundreds of thousands of sensor data per second.

#### Third, real-time statistics function

The tag table supports real-time statistics function. Machbase automatically generates five types of statistics in real time for the data stored in this tag table and provides a function to access them in real time.

## Supports Text Search Function

---

One of the most important practical uses for users to store and use logarithmic time series data is to determine if a specific event occurred at a particular point in time.

Time-series data processing is possible at a specific point in time, but in most cases the occurrence of a specific event requires searching for a specific "word" in a text field stored in a particular column. However, in a traditional database, in order to search for a word in a specific field, the exact match or LIKE clause is used to check the condition of some initial character through B + Tree. In most cases, this results in a very slow response. That's why searching for a particular word in a conventional database is very weak and frustrating. On the other hand, with Machbase, the SEARCH keyword based on the log table is provided to enable real-time word search. This makes it possible to quickly search for any error text generated from the equipment.

```
-- Example 1) Output record containing Error or 102 in msg field
SELECT id, ipv4 FROM devices WHERE msg SEARCH 'Error' or msg SEARCH '102';

-- Example 2) Output record containing Error and 102 in msg field
SELECT id, ipv4 FROM devices WHERE msg SEARCH 'Error 102';
```

## Optional Deletion Support

---

In the case of sensor data, it is true that deletion operations are rarely generated after insertion.

However, with embedded devices, there is a limited storage space that is not carefully managed by users. In this case, if a 'disk full' occurs or a failure occurs due to machine data, the company could suffer a lot of damage. Machbase provides the ability to delete records for a given condition in this environment. Therefore, embedded developers can use CRON or periodic programs to easily manage Machbase to not keep data over a certain size.

## For Log Tables

The following commands are supported:

```
-- Example 1) Delete oldest last 100.
DELETE FROM devices OLDEST 100 ROWS;

-- Example 2) Delete all but 1000 most recent.
DELETE FROM devices EXCEPT 1000 ROWS;

-- Example 3) Delete all of them from now on except one day.
DELETE FROM devices EXCEPT 1 DAY;

-- Example 4) Delete all data from before June 1, 2014.
DELETE FROM devices BEFORE TO_DATE('2014-06-01', 'YYYY-MM-DD');
```

## For Tag Tables

The following command is supported:

```
-- Delete all data from before June 15, 2016.
DELETE FROM tag BEFORE TO_DATE('2016-06-15', 'YYYY-MM-DD');
```

## Automated Data Collection

---

Machbase provides a "Collector" function that reads data from scattered machine data log files and automatically transfers them.

It not only collects pre-formatted data such as syslog and web server logs, but also provides a function that can be easily converted and automatically collected even if the log format is arbitrarily defined by the user.