

# A 5 minute log story

Ryadh Dahimene  
@ryado  
Consulting Architect  
ClickHouse, Inc.







~500 years  
later

/var/log

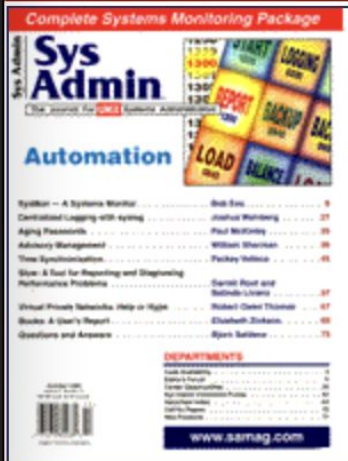
---

**Note –**

Due to limitations in the operating system, the Sun Java System Web Server does not work with log files larger than 2GB on Linux. When the maximum file size is reached, logging stops.

---





Article

[Figure 1](#)

[Figure 2](#)

[Table 1](#)

[oct98.tar](#)

# Centralized Logging with syslog

Joshua Weinberg

"Sys admin" is a title that rarely refers to the administration of a single system. Most organizations have from tens to thousands of computers, often with very high availability requirements, in their enterprise. All of these systems are often managed by only a handful of administrators. In this kind of environment, it is crucial for the administrator to have access to as much information about those systems as possible. Most of this information is found in log files kept on the individual systems. The question is: How can you realistically monitor these log files when they are spread out across so many machines? A solid approach to this problem is through a centralized logging scheme.

This article describes the implementation of such a logging scheme on a network of UNIX and NT systems by which all important system and application log messages are logged on a central log host using **syslog**. I also discuss two included Perl scripts, **flog.pl** and **watch.pl**. **flog.pl** (Listing 1) is used to force programs that use flat files for logging (like UUCP and some Web servers) to use **syslog** instead. **watch.pl** (Listing 2) is used to watch multiple system logs in real-time and highlight important log messages as they appear. (All listings for this article are available from: **ftp.mfi.com** in **/pub/sysadmin**.)

[Sysadmin Magazine](#)

October 1998, Vol. 7 Issue 10

# 1st decade of the 2000s



- The “Golden Era” of the relational model
- Primary Keys, Foreign Keys everywhere
- Rigid when it comes to logging
- Costly \$



21st International Conference on Data Engineering (ICDE'05)  
05-08 April 2005, Tokyo, Japan

**“One Size Fits All”: An Idea Whose Time Has Come and Gone**

Michael Stonebraker  
*Computer Science and Artificial  
Intelligence Laboratory, M.I.T., and  
StreamBase Systems, Inc.  
stonebraker@csail.mit.edu*

Uğur Çetintemel  
*Department of Computer Science  
Brown University, and  
StreamBase Systems, Inc.  
ugur@cs.brown.edu*

**Abstract**

*The last 25 years of commercial DBMS development can be summed up in a single phrase: “One size fits all”. This phrase refers to the fact that the traditional DBMS architecture (originally designed and optimized for business data processing) has been used to support many data-centric applications with widely varying characteristics and requirements.*

*In this paper, we argue that this concept is no longer*

of multiple code lines causes various practical problems, including:

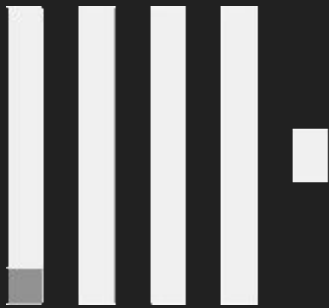
- *a cost problem*, because maintenance costs increase at least linearly with the number of code lines;
- *a compatibility problem*, because all applications have to run against every code line;
- *a sales problem*, because salespeople get confused about which product to try to sell to a customer; and
- *a marketing problem*, because multiple code lines

## 2nd decade of the 2000s



- The rise of NoSQL
- Document stores
- Inverted Indexes
- Search based systems
- JSON
- Less schema
- Often more storage

# And then ... ClickHouse



- Column oriented OLAP
- Great compression (x10 ratios are not uncommon)
- Powerful Analytics
- Horizontally Scalable
- Eg. [Uber Logging platform](#)
  - 3x data compression
  - 10x performance increase
  - ½ the reduction in hardware cost.

