

TraceDebate

Pros and cons for adding a Trace level

Numerous users have asked for the addition of a trace level. This page debates the pros and cons of such a change.

Update

The TRACE Level has been added in version 1.2.12 of Log4J, so this debate has to be considered outdated.

PRO

TRACE is a very natural level to use to output chatty information during development. It "traces" the program while DEBUG rather gives more of some "overview"-type debugging information.

There is a user-based need for it:

- It is a frequently requested feature as a quick search of the list archives will show.
- The JBoss group actually did extend log4j to add TRACE, and have distributed their server with the extensions included.
- Adding the TRACE level would involve very little effort.

Lack of development levels:

- TRACE and DEBUG seems like the two most important *development* tools. INFO, WARN, ERROR and FATAL are *production* levels. Thus, if one accepts this point of view, currently log4j offers only one single level dedicated to the development process.

With respect to domains:

- I already use categories for the division of different types of log statements. The categories in my system aren't classnames, I make them in a [LoggingCategories](#) static file with lots of Strings in them, concatenating each type to form a hierarchy of of "domains" if you will. "
-

CON

- It is possible to extend log4j to include the TRACE level. Many people have done so without problem. [The relevant section in the FAQ.](#)
 - Many people use INFO and DEBUG for development, eliminating the need for TRACE. Thus, one uses DEBUG to output chatty information and INFO to output an overview the application's progress.
 - A modification to something as central to log4j as the default list of levels must be done with consideration to not damage existing installations which have customized log4j to include TRACE or other custom levels.
-

Trace being "more verbose" than debug

Where does trace fit into the line of verbosity? Here is a clarification:

"Tracing" an application is usually referring to "step-locking" through code, that is, seeing every single change to every single register in the processor, for every operation. For unixers, "strace" and "ltrace" are the tools by which ~~all~~ system calls / library calls are "traced". The "trace" word hence, in my opinion, have an established meaning as something ~~extremely~~ verbose: it ~~traces~~ the applications every single step.

Debug information, on the other hand, is more for being able to debug the application as a whole (and this is, of course, a very overlapping area!). You'd like to see when the application asks the dbpool for an connection (maybe? - the 40 steps required within the dbpool to fetch this connection are at least definitely trace-level outputs), you'd like to see that the user made some "semantic action", e.g. "loaded customer data". This is more "contextual" information, e.g. "the user such and so made the this and that action", and, implicitly, if you'd turn on trace, you'd see ~~which~~ small, single-steps that have been done to accomplish this action.

Here's some dictionary lookups: [Onlook.com](#) [Dictionary.com](#) [Tiscali's computer dictionary](#) [CSGNetwork.Com](#)

Some quotes:

"A method of checking that a computer program is functioning correctly by causing the changing values of all the variables involved to be displayed while the program is running."

"To follow the course or trail of: trace a wounded deer; tracing missing persons."

"To determine the "flow" of a program, often used as a synonym to debug".

"To locate or discover by searching or researching evidence: trace the cause of a disease."

"A mark left by anything passing; a track; a path; a course; a footprint; a vestige; as, the trace of a carriage or sled; the trace of a deer; a sinuous trace."

"A mark, impression, or visible appearance of anything left when the thing itself no longer exists; remains; token; vestige"

"To follow by some mark that has been left by a person or thing which has preceded; to follow by footsteps, tracks, or tokens."

"a clue that something has been present; "there wasn't a trace of evidence for the claim" "

"A line drawn by a recording instrument, such as a cardiograph." "

Possible interpretations

- The above quotes establish that TRACE is a synonym for DEBUG, or even a better word in the context of logging.
- TRACE is very verbose: it is a "line drawn by a recording instrument", "a mark left by anything passing", "a footprint". It details every single move of the system. -endre