

被淘汰！

This location is being phased out. Please, continue with [BTrace@GitHub](#)

BTrace

BTrace is a safe, dynamic tracing tool for the Java platform. BTrace can be used to dynamically trace a running Java program (similar to [DTrace for OpenSolaris](#) applications and OS). BTrace dynamically instruments the classes of the target application to inject tracing code ("bytecode tracing"). Tracing code is expressed in Java programming language. There is also integration with DTrace for the OpenSolaris platform.

You can enhance VisualVM with the [BTrace plugin](#) to be able to write and deploy BTrace scripts directly from VisualVM. Alternatively, if you'd prefer you can run **BTrace** from command line shells as well.

There is also a [NetBeans integration](#) that makes it easier to write and deploy BTrace scripts. For your convenience you can directly install the [BTrace Update Center definition](#) which gives you access to the integration modules right from within the IDE plugin manager.

License

BTrace is available under [GNU Public License v.2 w/Classpath Exception](#).

Download

Actual BTrace binary realease is available for download from Documents & files link.

Maven

BTrace binaries are available as Maven artifacts as well. You can get them [here](#)

User Guide 用户指南

[BTrace User's Guide](#)

Building from the sources

BTrace sources are available from Mercurial repository. After checking out the sources, you can use NetBeans to build BTrace using the [NetBeans project setup instructions](#). There is also a [BTrace Developer's Guide](#).

开发者指南

Known Issues 已知问题

- **JVM crash**: We have seen JVM crashes with BTrace. We have fixed few such crashes as well. JVM could crash due to the following **reasons**:
 1. BTrace **submits invalid .class file to VM** (bug in BTrace instrumentor)
 2. **Hotswap bug** in HotSpot JVM.

To enable debugging such scenarios, please turn on the following options in \$BTRACE/bin/btrace (or btrace.bat) script. By default, the System properties `com.sun.btrace.debug` amd `com.sun.btrace.dumpClasses` are set to **false** in the scripts. Please change these to **true**. Also, you may want to add **-verify:all** option as well - which will force bytecode verification of all loaded classes. When you report a JVM crash with BTrace, please attach/send the following:

1. **BTrace trace class** that caused the crash.
2. **hs_err_pidXXX** log produced by JVM.
3. *.class file(s) saved by BTrace setting the `com.sun.btrace.dumpClasses` option.
4. debug output from BTrace by setting the `com.sun.btrace.debug` option.

How to contribute to BTrace?

If you like BTrace and want to contribute fixes/features to BTrace, your are always welcome! You (or your company) have to sign the [Sun Contributor Agreement](#). Then, please join this project as "developer" using your kenai.com id.

BTrace on the web

- [Who is calling this method?](#) 查看本方法调用栈
- [BTrace Script Concepts](#)
- [Using VisualVM, BTrace & Good Old Web Search](#)
- [Analysing Memory Leaks with BTrace](#) 分析内存泄漏
- [btrace for performance tuning](#) 性能调优
- [BTrace the ultimate JVM probe](#) 终极JVM探测
- [Socket Monitoring: Now Using BTrace](#) 套接字监视
- [BTrace Fix on the Mac](#)
- [Using BTrace in Java 6](#)
- [BTrace == DTrace for Java](#)
- [BTrace showing promises](#)
- [SailFin work and BTrace](#)
- [Runtime profiling and monitoring Java apps with btrace](#) 运行时分析和监视Java应用