Resource Monitoring for Poly/ML Processes

David Matthews    Magnus Stenqvist    Tjark Weber

pro lingua

UPPSALA UNIVERSITET
Software applications are subject to performance requirements.

Application monitoring helps developers to manage application performance, understand resource requirements, and identify bottlenecks.

Generic system monitor tools only provide very basic information about a process; they are unaware of the internals of its run-time system.
Our Contribution

A graphical monitoring tool for the Poly/ML run-time system.

Poly/ML is a popular open-source implementation of Standard ML. It features multi-threading, parallel garbage collection, 64-bit support, a debugger, and more.

Poly/ML makes detailed information about the state of the run-time system available.
What Is Being Monitored?

17 values from four areas of interest:

- CPU time (4 values)
- Memory footprint (5 values)
- Garbage collection (2 values)
- Threads (6 values)

Additionally, up to eight user counters are available and may be used freely by applications.
Software Architecture

Java frontend
- Model-view-controller pattern
- JFreeChart library to display time series

XML encoded data (via standard pipes)

Poly/ML backend
- calls PolyML.Statistics.getRemoteStats
- forward compatible due to extensible records
Conclusions

You can use our monitoring tool to **visualize the resource requirements of** Standard ML applications at run-time,

- on any system where Poly/ML and Java are available,
- with no changes to the applications being monitored!

Download from

https://bitbucket.org/tjark/poly-ml-monitor