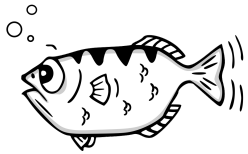
The project goals



- valgrind can act as a gdbserver
  - expose definedness and accessibility to the user in gdb
  - but setup is complicated - always requires 2 terminals
  - article `https://developers.redhat.com/articles/2021/11/01/debug-memory-errors-valgrind-and-gdb`
  - written by Alexandra Hájková and Mark Wielaard
- our main goal - make things "just work", gdb is in control
- secondary goal - provide technical information about the gdb remote protocol
  - `http://sasshkas.blogspot.com/`
- technical cooperation between perftools/gdb, Mark and Alexandra, to share knowledge and code together

Technical tasks

- valgrind side
  - support the extended-remote protocol - enables gdb to start process
  - support the multi-process protocol - enables gdb to follow children
- we need to use an intermediary - vgdb due to technical challenges
  - add –multi mode to vgdb - start valgrind and pass through the gdb remote protocol
  - add multi-process mode
- gdb side: new target - valgrind
  - sets up the environment
  - starts vgdb in –multi (and extended-remote) target mode
  - makes gdb and the target to share the tty for i/o - the most challenging
- tracking upstream
  - valgrind: `https://bugs.kde.org/show_bug.cgi?id=434057`
  - gdb: `https://sourceware.org/bugzilla/show_bug.cgi?id=28916`