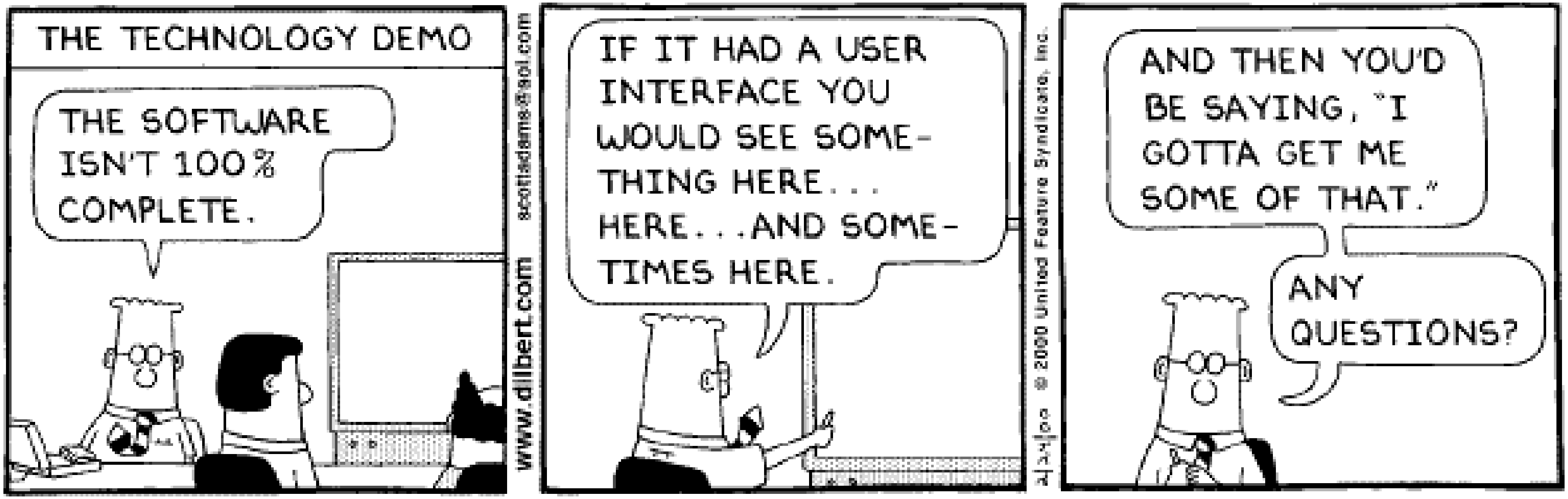
UNIVERSITY OF
WISCONSIN
MADISON

# The Paradyn Port Report

**Barton P. Miller**

bart@cs.wisc.edu

Computer Sciences Department
University of Wisconsin
1210 W. Dayton Street
Madison, WI 53706-1685
USA


THE TECHNOLOGY DEMO
THE SOFTWARE ISN'T 100% COMPLETE.
scottadams@aol.com
www.dilbert.com
IF IT HAD A USER INTERFACE YOU WOULD SEE SOMETHING HERE... HERE...AND SOMETIMES HERE.
3/24/00 © 2000 United Feature Syndicate, Inc.
AND THEN YOU'D BE SAYING, "I GOTTA GET ME SOME OF THAT."
ANY QUESTIONS?

# Supported Systems
## (your mileage may vary)

- Solaris 2.6 & 2.7 (SPARC & x86)
- Windows NT 4.0 (x86)
- Linux 2.2 (x86)
- Irix 6.5 (MIPS)
- *AIX 4.2 & 4.3 (Power2/SP2) - DyninstAPI only*
- *Tru64 Unix 4.0 (Alpha) - DyninstAPI only*

Paradyn v3.0 release target 1st May 2000

βeta version available end March 2000

# Baseline Paradyn UI/Front-end

- Performance Consultant with streamlined callgraph-based /Code hierarchy search
- Unified /Machine/Process/Thread hierarchy
- Tk-based time-histogram and other Visis
- Full set of inclusive and exclusive metrics
- Metric data export

# Baseline Paradynd/runtime-library

- Attach to running processes
- Shared object handling
- Shared-memory metric data sampling
- Multiple dynamic instrumentation heaps
- Instrumentation recursion guards
- Dynamic call-site instrumentation
- Retroactive “catch-up” instrumentation
- 64-bit data path
- Support for GCC and native ***C*** compilers

# Solaris (SPARC)

+ Single-instruction patch jumps
+ Function relocation/expansion
+ System-call interruption/resume
+ Thread instrumentation
  (separate daemon & runtime library)

? SunWorkshop ***f90*** support

# Solaris (x86)

+ Single-instruction patch jump (via trap)

+ System-call interruption/resume

– Instrumentation recursion guards

– Sometimes process dies during attach

– *gethrvtime()* rollbacks

# Windows NT (x86)

+ Single-instruction patch jump (via trap)

+ Support for CodeView executables

+ Visual ***C++*** and Digital Fortran ***f90*** support

– GCC ***C*** support

– Instrumentation recursion guards

– Trap-based instrumentation performance

– “Detach-and-leave-running”

# Linux (x86)

+ Single-instruction patch jump (via trap)

+ Post system-call interruption

? MPICH

– Instrumentation recursion guards

– Trap-based instrumentation performance

– Multi-CPU paradynd sampling rollbacks

- Open issues:
  - Hardware counters for wall-time
  - ? Handling executables without *dlopen()*

# Irix (MIPS)

- Single-instruction patch jumps
- Access to on-stack function parameters
- 32-bit & 64-bit object support
- MIPSpro ***f90*** compiler support
- Origin MPI

# AIX (Power2)

- Paradyn upgrade in progress
  - Dyninst available from Maryland

– Shared object handling

– Dynamic call-site instrumentation

? Retroactive “catch-up” instrumentation

+ SP2/POE MPI

# Tru64 Unix (Alpha)

- Dyninst available from Maryland
  - Paradynd/runtime-library port in development

+ 64-bit support

– Instrumentation recursion guards