June, 2010

GNU Development Tools for QorIQ and PowerQUICC Processors

FTF-NET-F0676

Phil Brownfield

Software Product Manager, NMG

freescale™ semiconductor

Objectives

- The GNU toolchain is a vital component in many of Freescale customers' and partners' development environments.
- In this session we will review recent changes affecting these free software tools for processors based on Power Architecture® technology
  - Enhancements supporting the new QorIQ processor family
  - Enhancements supporting multicore debug with GDB
  - Changes in related non-GNU software components
- We will also review Freescale's process for GNU toolchain development




Agenda

- GNU overview
- Freescale, CodeSourcery and the Open Source community
- GNU component status for Power Architecture technology
  - GCC
  - Binutils
  - GLIBC, EGLIBC
  - Newlib
  - LIBSTDC++
  - GDB

freescale semiconductor

Open Source and Freescale Networking Processors

- Freescale participates in free and open source software communities supporting processors based on Power Architecture® technology
  - GNU and related development tools
  - Linux® OS kernel and related U-Boot firmware projects
- Participation is driven by requirements from multiple sources
  - Customer needs
  - Power Architecture ecosystem partner needs
  - Freescale's internal needs

freescale semiconductor ™

The GNU Compiler Collection

- The GNU Compiler Collection (GCC) is a project of the Free Software Foundation (FSF)
- GCC supports compilation for C, C++, Fortran, ADA, JAVA and other languages
- GCC supports software development on various hosts for various native and cross-development target systems
  - Supported by associated run-time library projects: GLIBC, Libstdc++
  - Some frequently-used run-time libraries are not FSF projects
    - EGLIBC, Newlib, uClibc, BSD libc, Bionic open source C libraries
- GCC is “free software” released under the GNU Public License
  - GCC version 4.2.1 and before are released under GPLv2
  - GCC version 4.2.2 and later are released under GPLv3

GNU Community Participation

- Freescale assures high-quality GNU tools support for all our processor cores based on Power Architecture® technology
  - GCC, Binutils, EGLIBC, Newlib, LIBSTDC++, GDB
  - Run-time execution on Linux® OS and bare metal target systems
- Development done in collaboration with CodeSourcery
  - CodeSourcery is a strategic Freescale partner with strong GNU expertise
  - Freescale shares all its GNU toolchain work with the community
  - Overall effort spans new feature development, ports, maintenance, bug fixes
- Freescale includes GNU toolchains with its Linux OS releases

About CodeSourcery

- Sourcery G++
  - Personal & Professional Edition
    - Eclipse IDE
    - GNU C/C++ compilers
    - JTAG debug
    - Flash programming
    - Support
  - Lite Edition
    - Command-line tools
    - Available at zero cost to end users

- Company
  - Employs official maintainers of GCC, GDB, Binutils, GLIBC, and other key components
  - Has contributed over ten thousand enhancements to the GNU
  - Founded in 1997

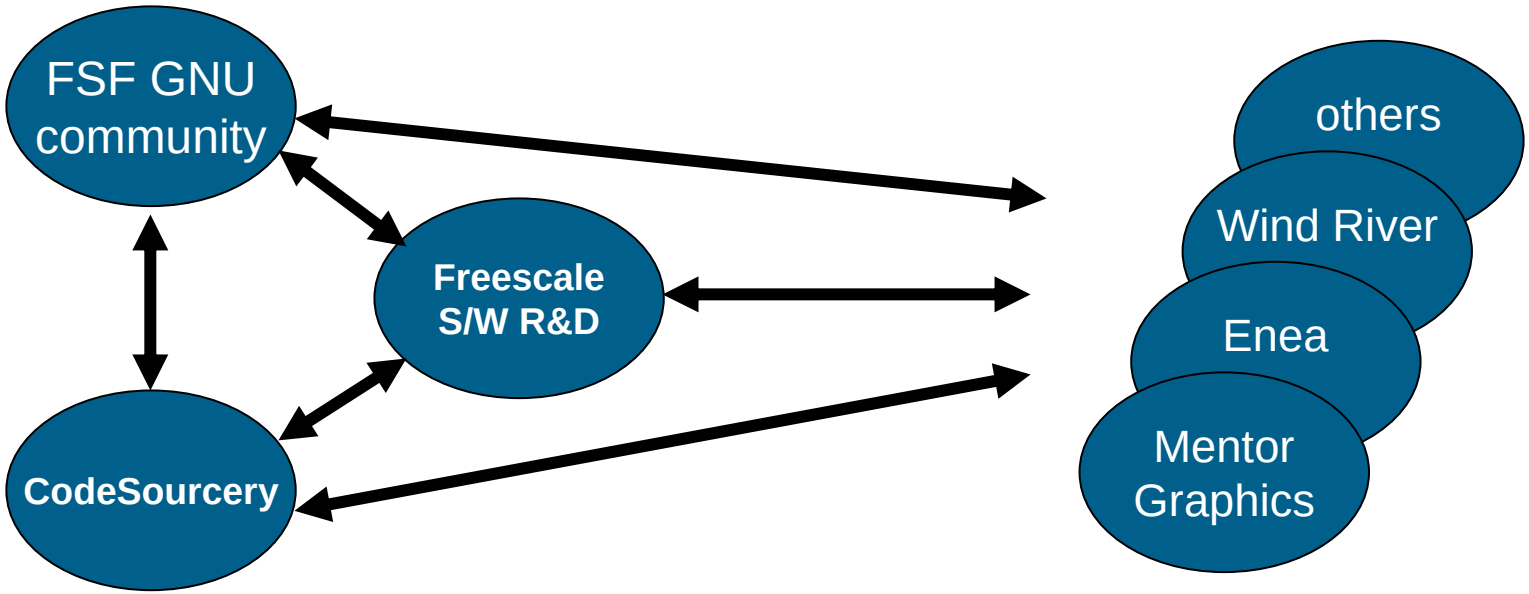
- Services
  - Porting toolchain to new architectures and operating systems
  - Optimization of compiler, tools, and libraries
  - Feature enhancements
  - Technical support
  - Contribution to FSF
- Selected Partners

freescale semiconductor
Power.ORG MEMBER
WIND RIVER
ENEA
montavista
Corporate Patron Free Software Foundation

Slide courtesy of CodeSourcery

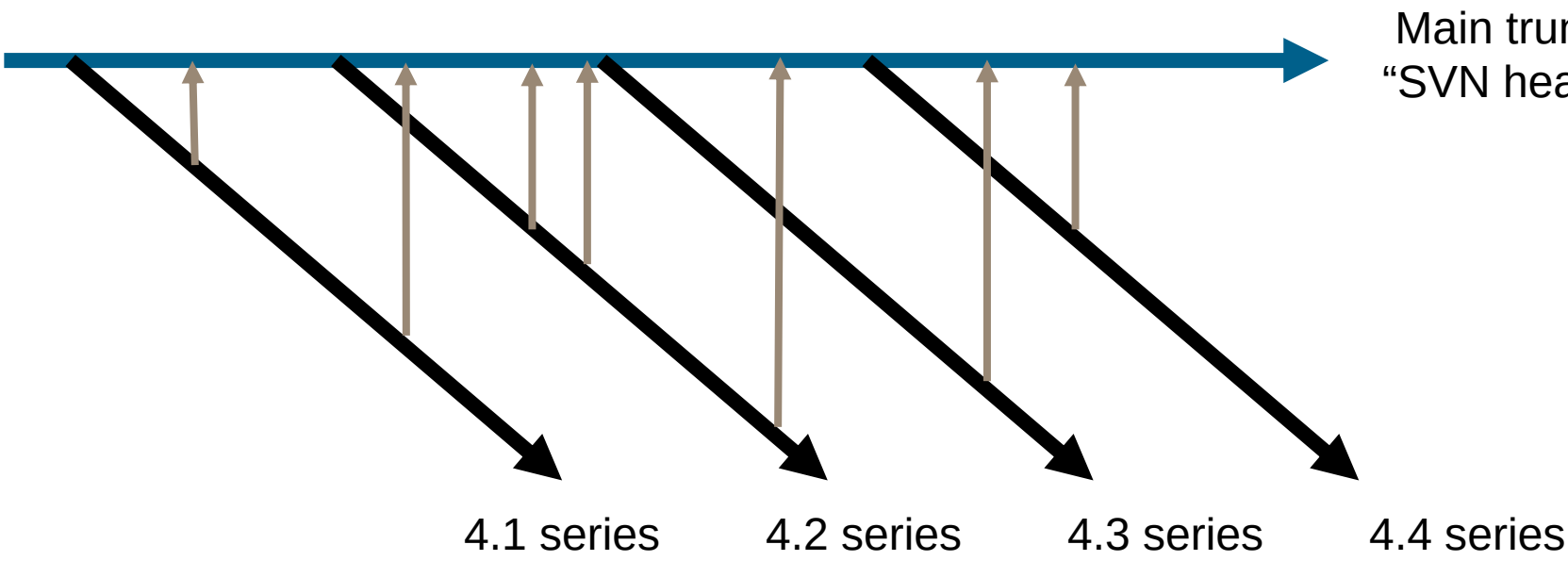
freescale semiconductor ™

Freescale Ecosystem GNU Relationships



- Freescale's Networking and Multimedia Group (NMG) software R&D team is the focal point for GNU technology exchange with the GNU community, including ecosystem members
- Exchanges are in both directions
- Vendors may rely on Freescale for early support for new processors, but tend to periodically synchronize to the FSF main trunk

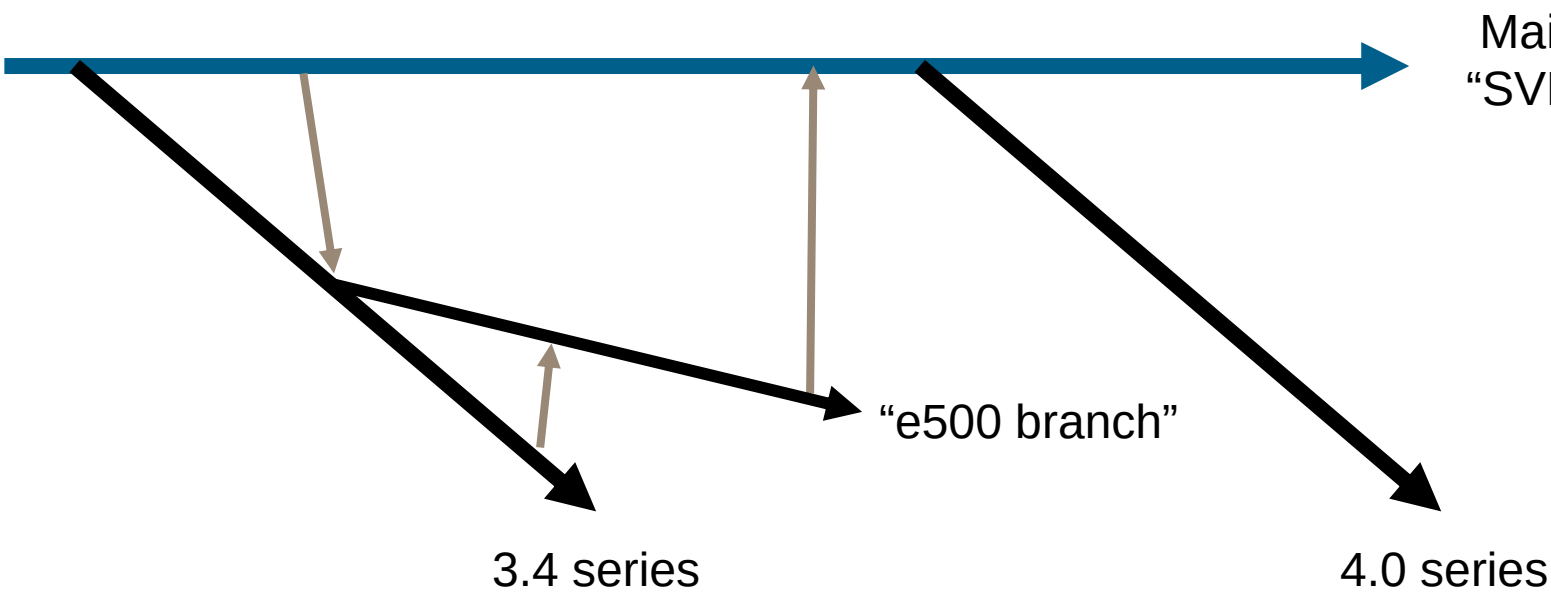
GCC Community Release Cycles



- Major GCC development is focused on the "main trunk"
  - Development choices driven to make gcc into a better gcc
- Numbered release series branch from the main trunk
  - Bug fixes, occasional minor development occur in release series
  - Release series fixes may be fed back into the main trunk
  - Main trunk enhancements in general not backported to release series branches
- Other projects typically follow the same pattern described here for GCC

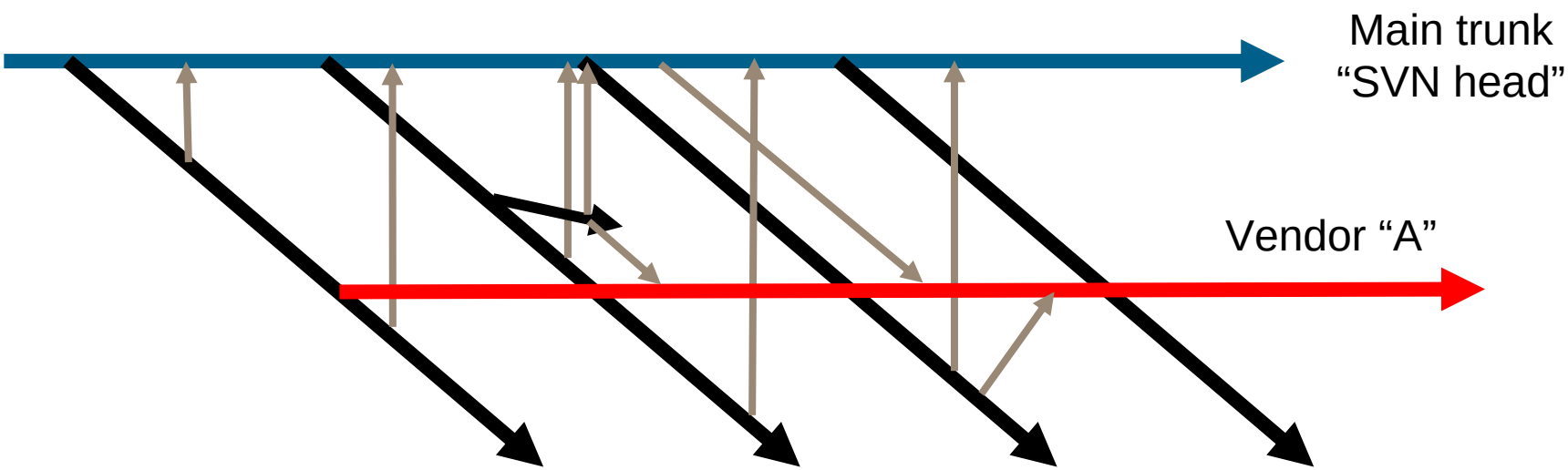
freescale semiconductor

GCC Branches



- Experimental branches allow development that does not meet the community’s objectives for the main trunk or a release series
- For example, e500v2 DPFP support was stabilized in a 3.4 branch
  - DPFP support was implemented first in main trunk
  - On completion, fixes were accepted into the main trunk

GCC Commercial Releases



- Most OS providers that support processors based on Power Architecture technology offer a GNU toolchain
  - Mentor Graphics, Wind River, Enea, QNX, Semihalf, many others
- Typically based on a numbered release series
  - Features may be forward/back ported from other release series, experimental branches, main trunk
  - Forward/back ports often motivated by increased processor support
  - Often add additional features to support own OS or integrate with other tools
- Porting and support services offered by GNU toolchain specialists
  - CodeSourcery and others

freescale semiconductor ™

GCC Status Snapshot

- A full-featured, highly-portable compiler with support for C, C++, ADA, JAVA, Fortran and other languages
- Most recent FSF releases are GCC 4.5.0, GCC 4.4.3 and GCC 4.3.4
- Recent Freescale Linux® releases for QorIQ processors include GCC 4.4.1
- Release of GCC 4.5.1 by FSF anticipated in July 2010
- Licensed under GPLv3; GCC 4.2.1 was the last GPLv2 release
- http://gcc.gnu.org/

Supported Power Architecture GCC Triplets

- When GCC is built, the target system type is named via what's called a triplet.
- Current cores, based on Power Architecture® technology, are supported by triplets in toolchains delivered by Freescale identified in the table below
  - Other Power Architecture triplets exist but are not delivered by Freescale today
- Each triplet implements support for either the Linux® ABI or the PowerPC Embedded ABI (EABI)

GCC triplet	ABI	Cores
powerpc-unknown-linux-gnu powerpc-unknown-linux-gnualtivec powerpc-unknown-linux-gnuspe	Linux	All
powerpc-*-elf	Linux	e500mc, e500mc64
powerpc-*-eabi powerpc-*-eabialtivec powerpc-*-eabispe	EABI	8xx, 603/e300, e500v1, e500v2, e600

Recent GCC Developments

- Recent GCC features for Power Architecture® technology
  - Support for new e500mc core (FSF GCC4.4, CodeSourcery Fall 2008)
  - Support for new e500mc64 core (FSF GCC4.5, CodeSourcery FTF 2010 Release)
  - Support for e300c2 and e300c3 core variants
  - New powerpc-*-elf triplet combines Linux® ABI with baremetal C library
  - Robust GOMP (GNU OpenMP) support
    - Syntax for parallel code sequences via #pragma hints
    - Implemented in C, C++, Fortran frontends, with backing libgomp library
- Other recent features of interest
  - Link Time Optimizations across separately compiled modules
  - Improving support for emerging C++0x ISO standard C++
  - Increasing use of new SSA framework for optimizations

GNU Binutils Status Snapshot

- A suite of tools to create and manipulate binary programs
  - Includes the GNU assembler (as) and GNU linker (ld)
- Most recent FSF release is GNU Binutils 2.20
- Latest Freescale QorIQ Linux® releases include Binutils 2.19
- Licensed under GPLv3
- http://sourceware.org/binutils/

Free/Open Source C Library Options

- There are a variety of open source C library choices, under a variety of different licenses
- Target system
  - Linux® OS: GLIBC, EGLIBC, uClibc
  - BSD: BSD libc
  - Bare metal: Newlib
- Completeness of API and resultant object code size a choice factor
- Choice of library is “baked in” GCC at build time
  - A given GCC binary build generates code for just one C library
  - Freescale tests GCC builds with EGLIBC and Newlib

GNU C Library (GLIBC) Status Snapshot

- A very complete standard C library suite for Linux® OS
  - Development closely coordinated with Linux kernel community
- Most recent FSF release is GLIBC 2.11
- Licensed under LGPLv2
- http://www.gnu.org/software/libc/libc.html

Embedded GLIBC (EGLIBC) Status Snapshot

- A very complete standard C library suite for embedded Linux® OS
  - API and ABI shared with GLIBC for source-level application compatibility
  - Development focused on a broad range of processor architectures
  - Recently adopted by Debian GNU/Linux and Ubuntu as their standard C library
- Not a GNU project
  - Administration by CodeSourcery; Freescale is a sponsor
  - Essentially implemented as patch sets to corresponding GLIBC branches
- Most recent branch is EGLIBC 2.11
- Latest Freescale QorIQ Linux releases are based on EGLIBC 2.10
- Licensed under LGPLv2
- http://www.eglibc.org/home

EGLIBC and NPTL

- Native POSIX Thread Library (NPTL) is a new implementation of process threads for Linux® OS
  - Supersedes earlier LinuxThreads implementation
  - Implementation affects the Linux kernel as well as C library
- Recent work to EGLIBC has matured NPTL on Power Architecture® cores

Newlib Status Snapshot

- Very functional C library but more memory-conscious than GLIBC
- Contains required C standard functions, but not all features expected by Linux® or BSD
- Not a GNU project; sponsored by Red Hat
- Most recent release is Newlib 1.18.0
- Licensed under multiple open source licenses, but not GPL or LGPL
- http://sources.redhat.com/newlib/

Libstdc++-V3 Status Snapshot

- Freescale project to implement an ISO Standard C++ Library
  - Source bundled with FSF GCC releases since GCC 3.0
  - In theory, may be used with other compilers
  - Depends on a separate underlying C library package
- Licensed under GPLv3 with a special runtime library exception
- http://gcc.gnu.org/libcstdc++

GCC Triplets Revisited

- Since choice of library is “baked in” when GCC is built, each triplet implements support for one C library
- The table below identifies library support integrated into triplets in toolchains delivered by Freescale identified in the table below

GCC triplet	ABI	C Library	C++ Library	Cores
powerpc-unknown-linux-gnu powerpc-unknown-linux-gnualtivec powerpc-unknown-linux-gnuspe	Linux	EGLIBC	Libstdc++	All
powerpc-*-elf	Linux	Newlib	Libstdc++	e500mc, e500mc64
powerpc-*-eabi powerpc-*-eabialtivec powerpc-*-eabispe	EABI	Newlib	Libstdc++	8xx, 603/e300, e500v1, e500v2, e600

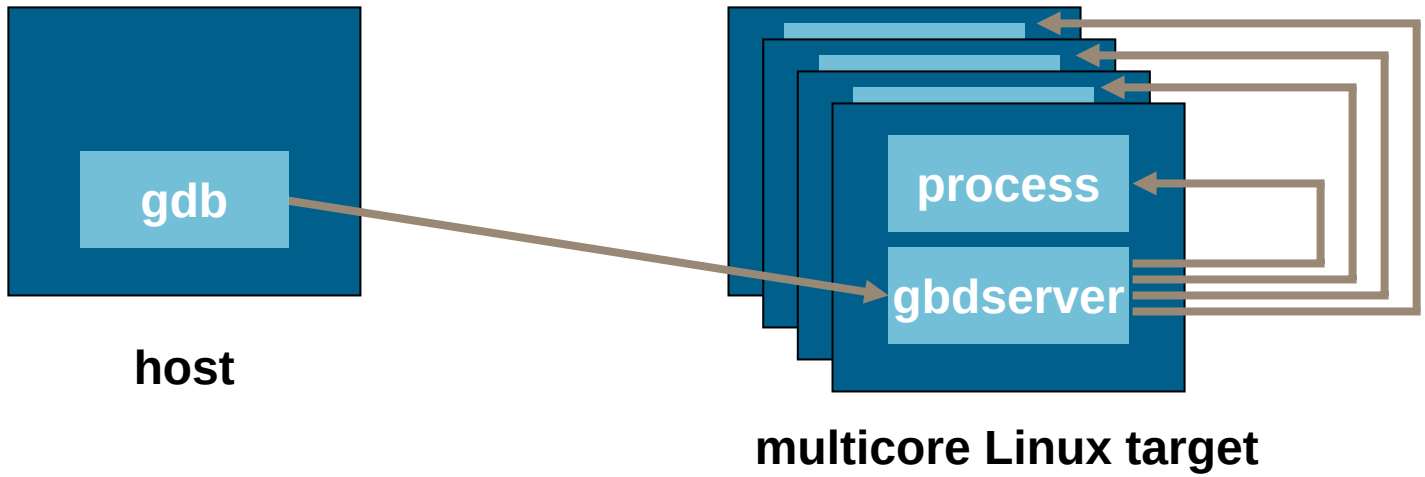
GDB Status Snapshot

- The GNU Project Debugger allows visibility to another executing program
  - Supports resident or cross debugging
- Most recent Freescale release is GDB 7.1
- Recent Freescale QorIQ Linux® releases include GDB 6.8
- Development towards GDB 7.2 is underway
- Licensed under GPLv3
- http://www.gnu.org/software/gdb/gdb.html

GDB Enhancements for Multicore Debug

- Multiprocess debug is a requisite for useful multicore debug
  - GDB was long aware only of one process in one address space
- Recent enhancements added some multiprocess debug capability
  - GDB aware of multiple process state information
  - Ability to debug multiple processes sharing same address space
- Freescale sponsored additional enhancements made by CodeSourcery
  - Ability to debug multiple processes in multiple address spaces with multiple program images
  - Extensions to GDB remote debug protocol adding process identifier
  - Remote GDBSERVER / GDB stub must map processes to cores
- Requirement to live within prior GDB implementations
  - Backward-compatible remote debug protocol a must
  - Self-hosted (non-remote) debug would have broken compatibility

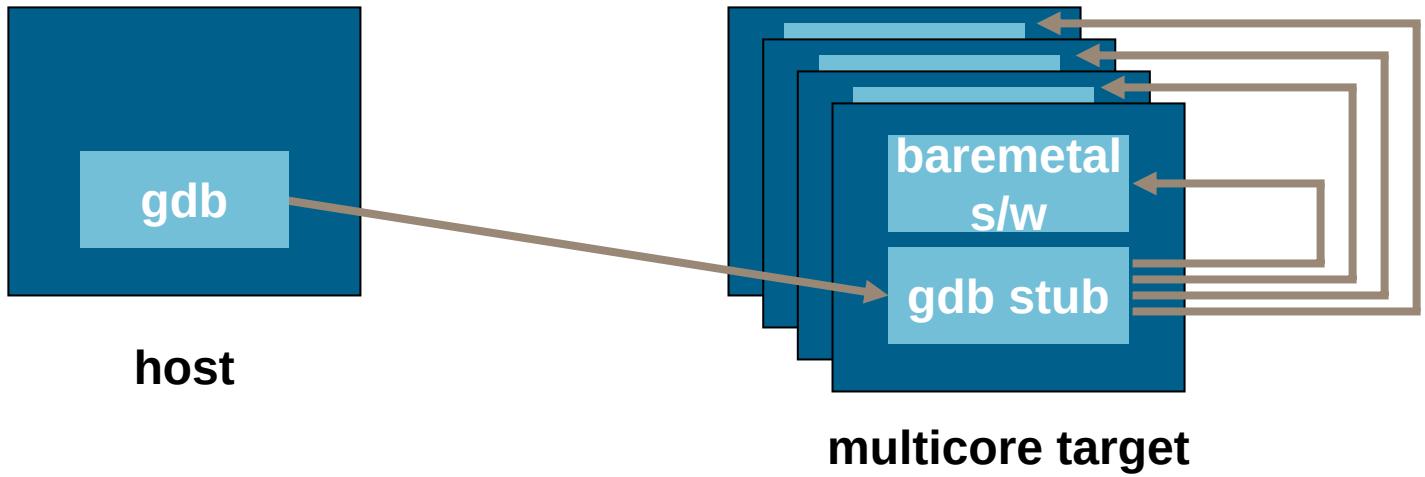
Remote Linux® OS Debug With GDBSERVER



- Supports multicore application-level debug, layered above Linux OS
- GDB on a host computer utilizes the GDB remote protocol to communicate with GDBSERVER running on a Linux target
- GDBSERVER now enhanced to manage debug of multiple processes running under SMP Linux OS, regardless of core

freescale semiconductor ™

Remote Baremetal Debug With GDB Stub

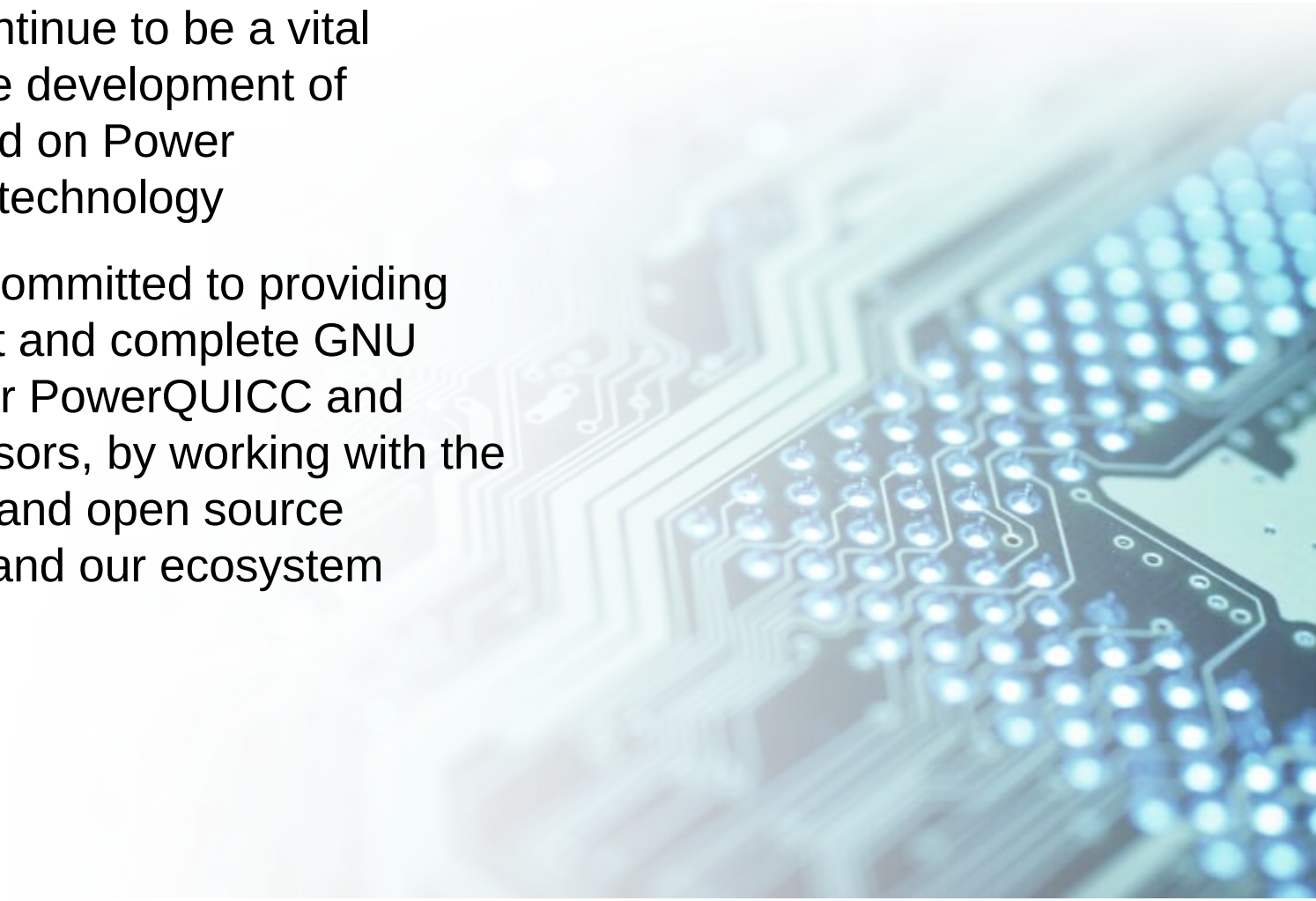


- Supports multicore bare metal debug
- GDB remote protocol communicates with a GDB stub on one core
- A GDB stub can manage debug of baremetal software on all cores, possibly through communication with agents on other cores
  - Must maintain awareness of which Process ID assigned to which core

freescale semiconductor

In Closing

- GNU tools continue to be a vital enabler for the development of systems based on Power Architecture® technology
- Freescale is committed to providing timely, current and complete GNU support for our PowerQUICC and QorIQ processors, by working with the free software and open source communities and our ecosystem partners



freescale semiconductor ™


freescale™
semiconductor