

Cray XT Compilers

Geir Johansen

Cray Inc.

Overview

- Cray XT programming Environment
- PGI Compilers
- GCC Compilers
- PathScale Compilers
- Performance
- Guidelines for Choosing a Compiler

Cray XT Compilers

- Portland Group Compilers (PGI)
- GNU Compiler Collection (GCC)
- PathScale Compiler Suite
- All provide:
 - C/C++ compiler
 - Fortran 90/95 compiler
 - AMD64 code generation with SSE2 support
 - OpenMP support

Cray XT Programming Environment

- A cross-compiler environment
 - Compiler runs on Linux login node
 - Executable runs on Catamount compute node or Compute-Node-Linux (CNL) compute node
- *modules* utility
 - Initializes environment for specific compiler
 - Allows easy swapping of compilers and compiler versions
- Compiler Scripts
 - Catamount & CNL compiler options
 - Catamount & CNL system libraries and header files
 - Compiler specific programming environment libraries

modules utility

- Main programming environment module for each compiler
 - *PrgEnv-pgi*
 - *PrgEnv-gnu*
 - *PrgEnv-pathscale*
- *PrgEnv* sub-modules
 - *pgi* PGI compilers
 - *gcc* GCC compilers
 - *pathscale* PathScale compilers
 - *xt-pe* Compiler driver scripts
 - *xt-mpt* MPICH2 and SHMEM libraries
 - *acml* AMD Core Math library
 - *xt-libsci* Cray scientific library

Compiler Driver Scripts

- Compiler commands
 - *cc*
 - *CC*
 - *ftn*
- Compiler scripts locate:
 - Catamount & CNL compiler options
 - Catamount & CNL system libraries and header files
 - Compiler specific programming environment libraries
- Don't call compiler directly
- Use vendor man pages
 - *man pgcc, man pgCC, man pgf90*
 - *man gcc, man g++, man gfortran*
 - *man pathcc, man pathCC, man pathf90*

modules Example

```
$ module load PrgEnv-pathscale
```

```
$ ftn -version
```

```
/opt/xt-pe/2.0.03/bin/snos64/ftn: INFO: catamount target is being used
```

```
PathScale EKOPath(TM) Compiler Suite: Version 2.5
```

```
Built on: 2006-08-22 21:02:46 -0700
```

```
Thread model: posix
```

```
GNU gcc version 3.3.1 (PathScale 2.5 driver)
```

```
Copyright 2000, 2001 Silicon Graphics, Inc. All Rights Reserved.
```

```
Copyright 2002, 2003, 2004, 2005, 2006 PathScale, Inc. All Rights Reserved
```

```
See complete copyright, patent and legal notices in the
```

```
/opt/pathscale/share/doc/pathscale-compilers-2.5/LEGAL.pdf file.
```

```
$ module swap PrgEnv-pathscale PrgEnv-pgi
```

```
$ ftn -V
```

```
/opt/xt-pe/2.0.03/bin/snos64/ftn: INFO: catamount target is being used
```

```
pgf90 7.0-2 64-bit target on x86-64 Linux
```

```
Copyright 1989-2000, The Portland Group, Inc. All Rights Reserved.
```

```
Copyright 2000-2007, STMicroelectronics, Inc. All Rights Reserved.
```

```
$ module swap pgi/7.0.2 pgi/7.0.3
```

```
$ ftn -V
```

```
/opt/xt-pe/2.0.03/bin/snos64/ftn: INFO: catamount target is being used
```

```
pgf90 7.0-3 64-bit target on x86-64 Linux
```

```
Copyright 1989-2000, The Portland Group, Inc. All Rights Reserved.
```

```
Copyright 2000-2007, STMicroelectronics, Inc. All Rights Reserved.
```

```
$
```

Cray XT OS Considerations

- Using Multiple Compilers for an Application
 - Cannot mix Fortran – Fortran modules
 - Cannot mix C++ -- Name Mangling
 - C code is fine
- Modified Programming Libraries for Catamount
 - PGI
 - libpgc.a
 - libpgf90.a
 - libpgftnrtl.a
 - libC.a
 - PathScale
 - libpathfstart.a
 - libpathfortran.a
 - libpathfortran_p.a
- Compiling for Service Nodes

PGI Compilers



THE PORTLAND GROUP

- Initial Compiler Supported for Cray XT3
- Most Used Compiler for Cray XT3
 - CrayPat
- *ftn -default64* option
- IOBUF
- Current supported version is 7.0-3

PGI Compiler Options

- Chapter 2 of PGI User Guide has overview of optimization option
- www.spec.org/cpu2006
- Essential compiler options:
 - -fast (same as -fastsse)
 - O2 -Munroll=c:1 -lre -Mvect=sse -Mscalarsse
 - Mcache_align -Mflushz
 - -Mipa=fast,inline
 - -Msafeptr=all
 - -O3, O4
 - -Minfo

GNU Compiler Collection



- Cray XT OS generation compiler
- GCC 3.2 version
 - Good for C and C++ code
 - Fortran 77 only
- GCC 4.1
 - Fortran 90/95 functionality
- Current version is 4.1.2

GCC Options

- gcc.gnu.org/onlinedocs/gcc-4.1.2/gcc/
- -ffast-math
- -O3
- -funroll-loops
- -fprefetch-loop-arrays

PathScale Compile Suite



- Cray Fortran front-end (from SGI)
- First available in Cray XT OS 1.5
- Supports the *assign* command for Fortran I/O
- *pathopt2* tool
- Current version is PathScale 3.0
 - Ordered through PathScale

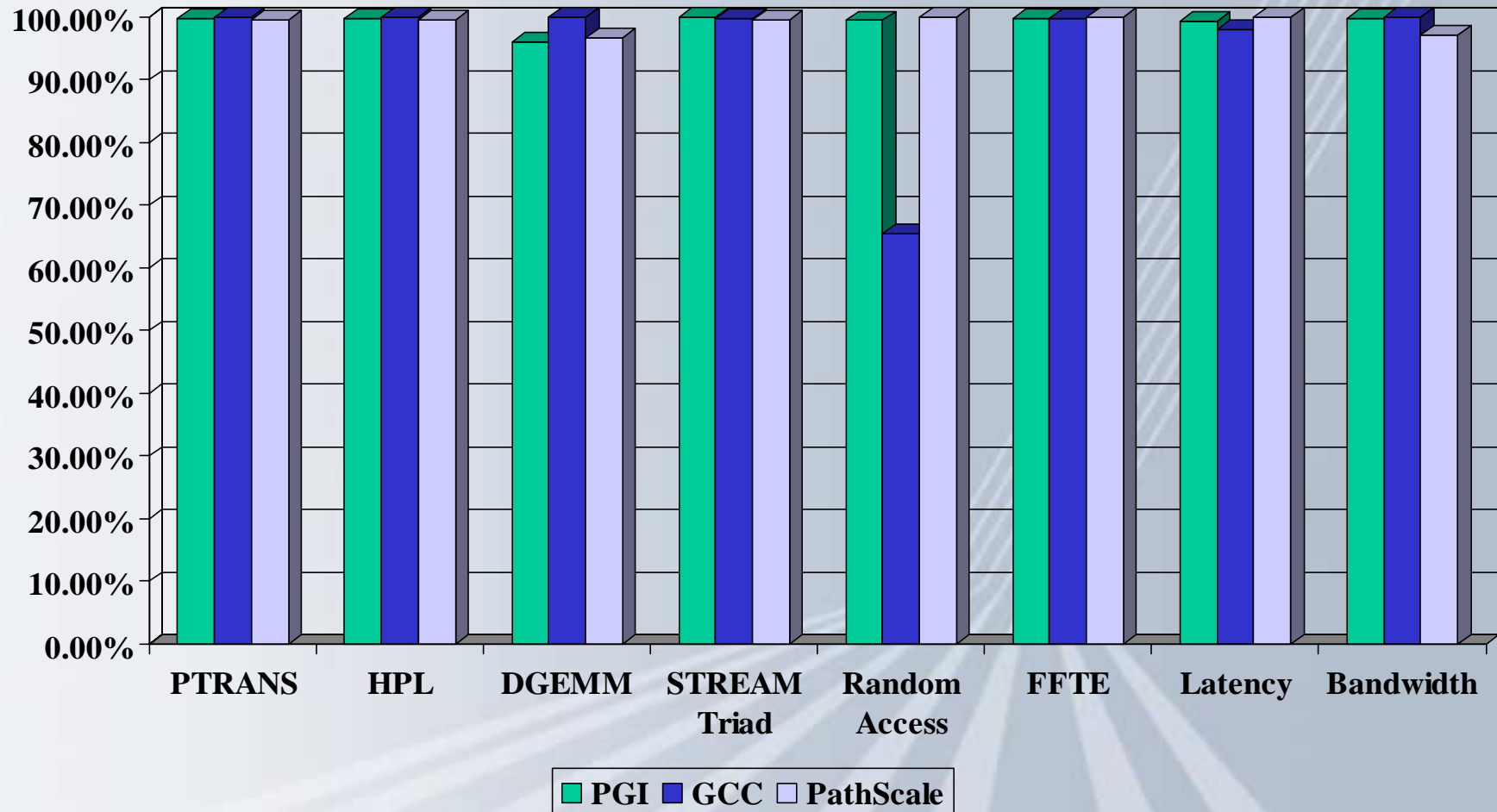
PathScale Options

- Use *eko* (Every Known Optimization) man page
- Chapter 6 of PathScale User Guide gives a quick introduction to optimization parameters
- www.spec.org/cpu2006
- Essential options:
 - -Ofast
 - -O3
 - -ipa
 - -OPT:Ofast
 - -ffast-math
 - -fno-math-errno
 - -OPT:alias=restrict

Performance Testing

- Big difference between compiler release versions
 - Note which release versions is used on published results
 - Data set affects which compiler performed better
- HPCC Challenge Benchmark
- Polyhedron 2005 Fortran Benchmark
- Stepanov C++ Benchmark
- Application and Benchmark Groups

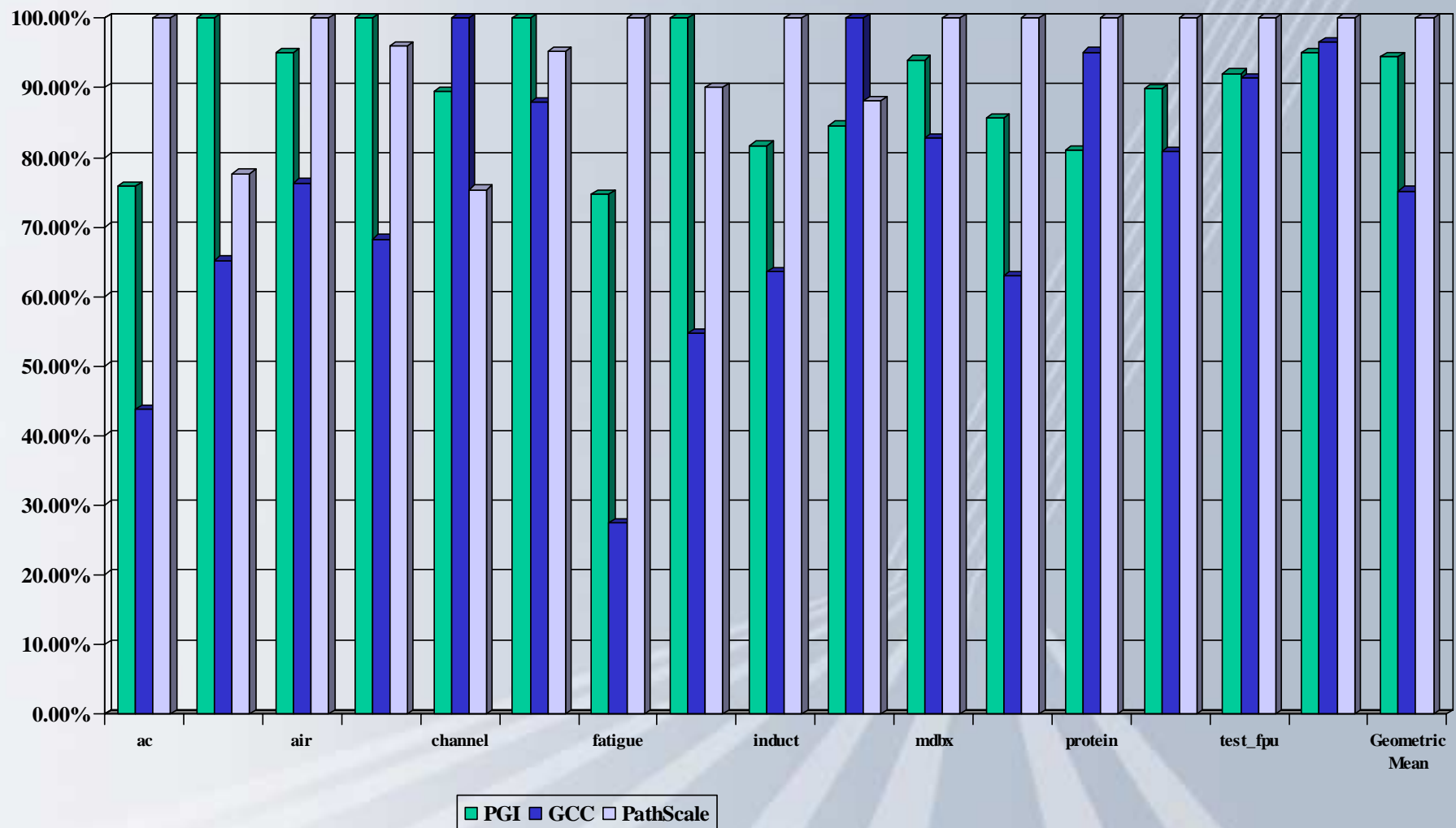
HPC Challenge Benchmark



HPC Challenge Benchmark

- The HPCC benchmark show the C compilers have similar performance
- The GCC compiler lacks a compiler option to specify the *prefetchnta* instruction.
- RandomAccess worked better with PathScale -O2 than -O3 optimization.

Polyhedron 2005 Fortran Benchmark



Polyhedron 2005 Fortran Benchmark

- GCC *gfortran* clearly does not perform as well as PGI and PathScale
- PGI and PathScale usually within 10% of each other. PathScale performing slightly better
 - Geometric mean has ~5% difference
- www.polyhedron.com
- Compilers used:
 - GCC 4.1.2
 - PathScale 3.0
 - PGI 7.0.3

Stepanov Benchmark

- Measures added level of abstraction of C++ code versus a simple C code
- GCC C++ was best
- PGI C++ did not perform well

Cray Inc. Applications & Benchmark Groups

- GCC Fortran is not used
- PGI & PathScale Fortran
 - Some codes perform better with PGI
 - Some codes perform better with PathScale
 - Almost always with 10% of each other
 - Both Fortran compilers are used
- GCC used for C and C++ code

Cray Inc. Applications & Benchmark Groups

- Anecdotal performance information
 - Vectorization
 - Large (code wise) loops
 - Complex Numbers
- Benchmarkers would like more effective compiler directives for PGI and PathScale!

Guidelines in Choosing Compiler

- Fortran Performance
 - PGI and PathScale much better than GCC
- C and C+ Performance
 - All C compilers are used
 - GCC C++

Guidelines in Choosing Compiler

- Cray XT Exposure Time of PGI
 - Libraries
 - *-default64*
 - IOBUF
 - CrayPat
 - TotalView
 - Support for other compilers

Guidelines in Choosing Compiler

- Application ISV Recommendations
 - Compiler Options
- IOBUF vs. assign
 - IOBUF may not help PathScale Fortran file I/O
 - PathScale does support *assign* command
- Performance Support
 - PGI and PathScale compete on performance

Guidelines in Choosing Compiler

- Portability Issues
 - GCC de facto standard for C & C++
 - *gnu attributes* directives
- Fortran 2003
 - GCC *gfortran* implements more Fortran 2003 features

CUG 2005

- PGI Compilers
 - Only compiler supported for Cray XT3
 - Lacking C and C++ features
- GCC Compilers
 - GCC 3.2
 - Fortran 90/95 not supported
- PathScale Compilers
 - Not available on the Cray XT3

CUG 2007

- PGI Compilers
 - Fully functional C and C++
 - Improved performance
- GCC Compilers
 - GCC 4.1
 - Fully functional F90/F95 compiler
- PathScale Compilers
 - Available on the Cray XT
 - Improved performance

Conclusions

- Cray XT Programming Environment
 - Supports multiple compilers
 - Ease of switching compilers and libraries
- Fortran code
 - Use PGI and/or PathScale
- C and C++ code
 - Similar C performance
 - GNU C++
- Cray XT3 Experience
 - PGI most used
 - Start with PGI, try others

Conclusions

- Compilers have improved significantly, and continue to improve.
 - Competitive environment
- Room for improvement
 - Compiler directives

Friendly Rivals



Questions

- geir@cray.com