Open64: the State of the Community and the Road Ahead

Fred Chow

Open64 Workshop

Mar 22, 2009

PathScale™

What brought us together

- Over 100 man-years' worth of compiler development effort
- Funded by SGI in the 90's
- Made available to the public in 2000
- State-of-the-art infrastructure
- Optimization focus
- Production quality
- Catering well to small-team development environment

PathScale™

The State of the Community

Open64 Is Widely Used in Teaching

- Alternative to GCC
- Established knowledge base
- Case study for different optimizations
- Vehicle for implementing course projects

PathScale™

Open64 Has Enabled Many Areas of Compiler Research

- Code Analysis and Optimization Algorithms
- Target-specific Code Generation and Optimization
- Support for Embedded Systems
- Parallelization and Vectorization
- Support of New Languages or Language Extensions
- Program Analysis/Advisory Tools

PathScale™

Academic Research Institutions Involved with Open64

- U. of Alberta
- UC Berkeley
- Chinese Academy of Science
- U. of Delaware
- U. of Edinburgh
- Fudan University
- U. of Ghent
- Georgia Institute of Technology
- U. of Houston
- INRIA
- Michigan Technological U.
- U. of Minnesota
- Rice U.
- Seoul National U.
- U. of Southern California
- U. of Texas, Austin
- Tsinghua U.

PathScale™

Many Companies in Industry have Embraced Open64

- Way to leverage work already done
  - work done before open-source
  - work done by the community
- People with Open64 expertise produce greater yields
  - no need for learning curve
  - expertise can be gained in academia and applied in industry
  - expertise can be re-used in different jobs
  - the pool is increasing over time
- Avoid leaving performance on the table

PathScale™

Industrial Companies Involved with Open64

- Absoft
- Cognigine (acquired by Huawei)
- Coherent Logix
- Convey Computer
- Equator (acquired by Pixelworks)
- Google
- HP
- Intel
- NVidia
- PathScale
- Qualcomm
- SiCortex
- SimpLight
- STMicro
- Tensilica

Open64 has Big Impact on the Compiler Industry

- Commonly used in performance studies
- Proprietary compilers have used Open64 as reference
- We raised the performance standard
- Other compilers have adopted our approaches
- gcc forced to catch up in optimization areas
- The end users benefit

PathScale™

Supported Processor Targets

- CEVA
- Coherent Logix Multi-core DSP
- Convey HC-1 co-processor
- Cyclops (IBM)
- Itanium
- IXP (Intel)
- MIPS
- NVidia GPU
- PowerPC (experimental)
- Qualcomm DSP
- SimpLight DSP
- ST200 (STMicro)
- x86/x86-64
- Xscale (experimental)
- Xtensa (Tensilica)

PathScale™

Supported Languages

- C/C++
- Fortran
- OpenMP
- UPC
- Coarray Fortran
- CUDA
- Java

PathScale™

Credits need to go to

- Promotion by people involved with Open64
- Support by academic institutions
- Funding secured by academic institutions
- Funding by industrial companies
- Contributions by individual developers
- Adoption by end users

PathScale™

Roles of Academic Institutions

- Play neutral roles among industrial companies
- Host repositories and forums
- Serve gate-keeping functions
- Secure research grants
- Perform quality research

PathScale™

Limits of Academic Institutions

- Work has to be unique
  - Work must contain new ideas
- Work usually not disclosed until ready
- Work mostly experimental nature
  - Limited to prototypes
- Less teamwork
- Could be educational exercise
  - Tolerate multiple efforts
  - Can be incomplete

PathScale™

Roles of Industrial Companies

- Develop open64-based products for the market
- Play support role for academic institutions
  - Bring research to practice
- Update compiler to evolving standards
- Enhance the user experience
- Promote Open64 products
  - Enlarge the user base
- Make money to fund more development work

PathScale™

Limits of Industrial Companies

- Work must be driven by business needs
- Two sources of funds:
  - Company subsidies
  - Revenue from sales and support
- Tight schedules for deliverables
- Mainly low-risk projects

PathScale™

Work Exclusively Performed by Industrial Companies

- Update front-ends
- QA and productization
- Reconcile differences among the branches
- Benchmark tuning
- Support the end users

PathScale™

PathScale is a Key Industrial Partner of Open64

- Contributions from PathScale to date:
  - Retargeted Open64 to x86/x86-64 and MIPS
  - Enhance command-line compatibility with gcc
  - Fortran front-end improvements
  - GNU front-end updates - 3.3.1, 4.0.2, 4.2.0
  - Separated GNU front-end from Open64 source tree
  - Added C++ exception handling support
  - Updated ipa_link to binutils-2.16.1
- Ongoing project:
  - Boot linux kernel on x86
- Future plan:
  - Work with Open64 repository

PathScale™

The Open64 Ecosystem

- Our community is diverse
  - Necessary to maintain creative ingredients
- Organizations have different interests and goals
- People busy with their own schedules
- Each will do just enough to meet its goals

So far, too much reliance on natural forces . . .

PathScale™

Lack of Partnership

- Waiting game, hoping someone will do the work
- Work eventually done by whoever has the most urgent need
- Little collaboration among development projects

PathScale™

Inadequate Communication

- No expectation of when a feature will be available
- May end up with duplicate efforts

PathScale™

Lack of Co-ordination

- Incompatible approaches towards solving a problem
- Parts that do not work together well
- Conflicting changes could be hard to reconcile

PathScale™

Loose Structure

- Organizations catering to their own goals
  - Cherry-pick what's good for them
  - Protection from changes irrelevant to them
- Only loose coupling among the bodies of work
- Proliferation of branches in the repository
- Delay in merging changes until the problem gets out of hand

PathScale™

The Road Ahead

Critical Front-end Work

- Update C/C++ to GNU 4.3 or 4.4
  - Needed for the newest Linux distros
  - Pre-requisite: resolution of GPLv3 issue
  - Continue to track GNU releases
- Support more GNU extensions
- C++ robustness
- Fortran 2003/2008 Standards

PathScale™

Other Desirable Front-end Work

- UPC
- Coarray Fortran
- GFortran front-end
- OpenMP 3.0
  - needs GNU 4.4
- CUDA
  - Native support in front-ends
  - Both C/C++ and Fortran
- OpenCL
- Java

PathScale™

General Optimization Improvement

- Alias analysis
- IPA info to backend
- Code versioning infrastructure
- Prefetch generation
- Vectorization capabilities
- Coarse-grain parallelization
- Code size optimization (-Os)
- Better whirl2c and whirl2f
- Dynamic Compilation

PathScale™

Increase Adoption by End Users

- Boot linux kernel
- Make part of linux distribution
- Improve support of debugging and other GNU tools
- IDE integration (e.g. Eclipse)
- Native compilers for Windows and MAC
- Documentation improvement

PathScale™

What we need

1. Increased Partnerships

- Joint efforts to attack problems
- Less reliance on individual organizations
- Spread responsibilities around
- Improve delivery schedules

PathScale™

2. More Communication

- Mailing lists
  - Rotating moderators?
- Wiki pages
- Workshops and other events

PathScale™

3. More Co-ordination

- Steering committee
- Contact Person for each organization
- Discussion forums

PathScale™

4. Tighter Structure

- Need a unified voice
- Consortium with paid membership?
  - Duties/responsibilities for members
  - Fund work of general interests
- Special interest groups?
  - Organize repository branches accordingly

PathScale™

5. Greater Generosities

Contributions can be:

- Your open64-related work
- Documentation
- Services
  - *testing*
  - *bugs*
  - *merging branches*
  - *support of users*

PathScale™

"As you give, so shall you receive"

- Matthew 7:12 & Luke 6:31

Comments and Suggestions?