

---

# Open IMPACT

Wen-mei Hwu

Franklin Woeltge Professor  
University of Illinois at Urbana-Champaign  
<http://www.gelato.uiuc.edu>

# Agenda

---

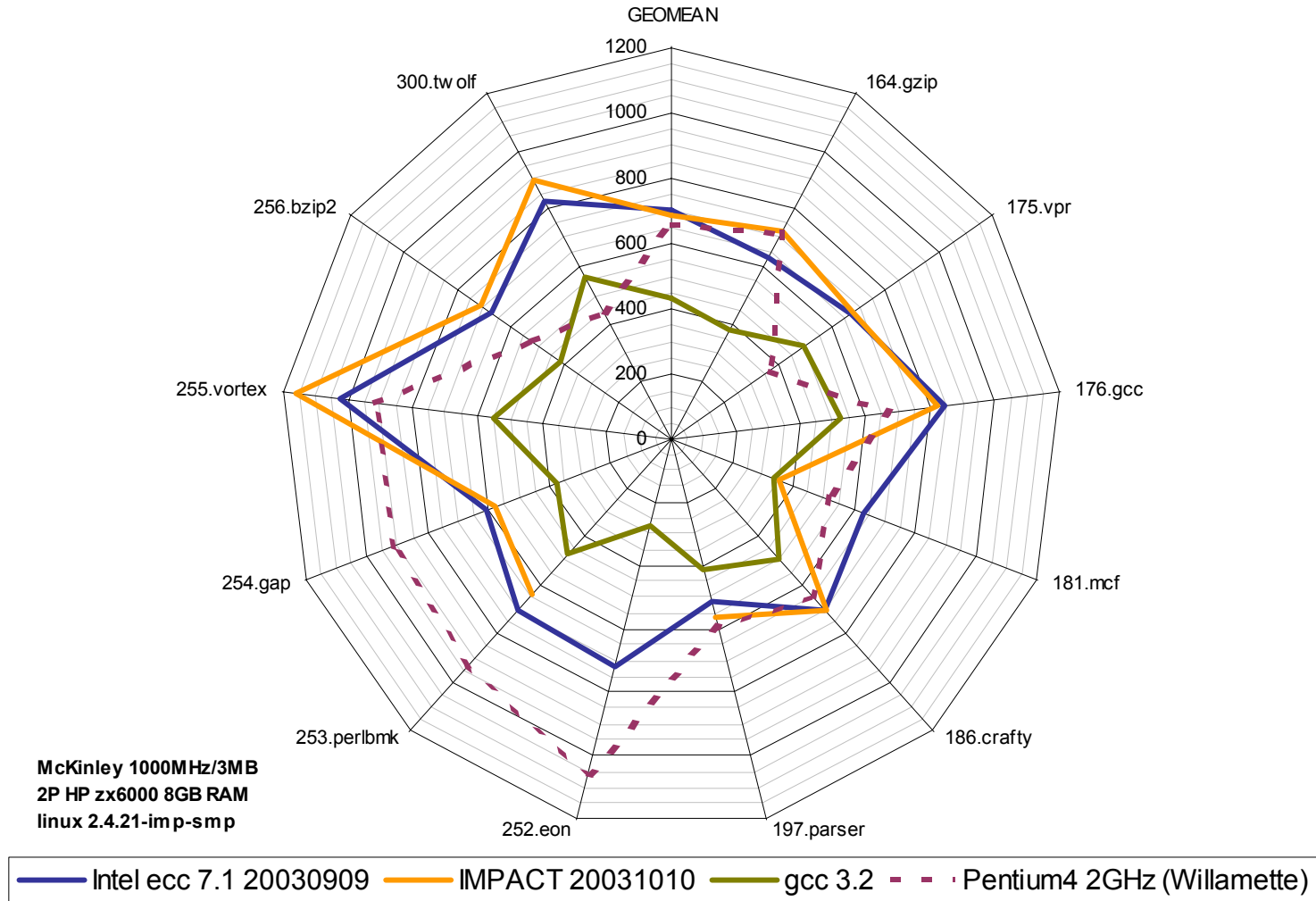
- Rationale, Goals, Plans, Status
- To solicit your questions and comments
  - Applications and systems considerations and problems that we should be aware of
  - Examples of open source code that we should be working on
- To solicit your active participation

# OpenIMPACT

---

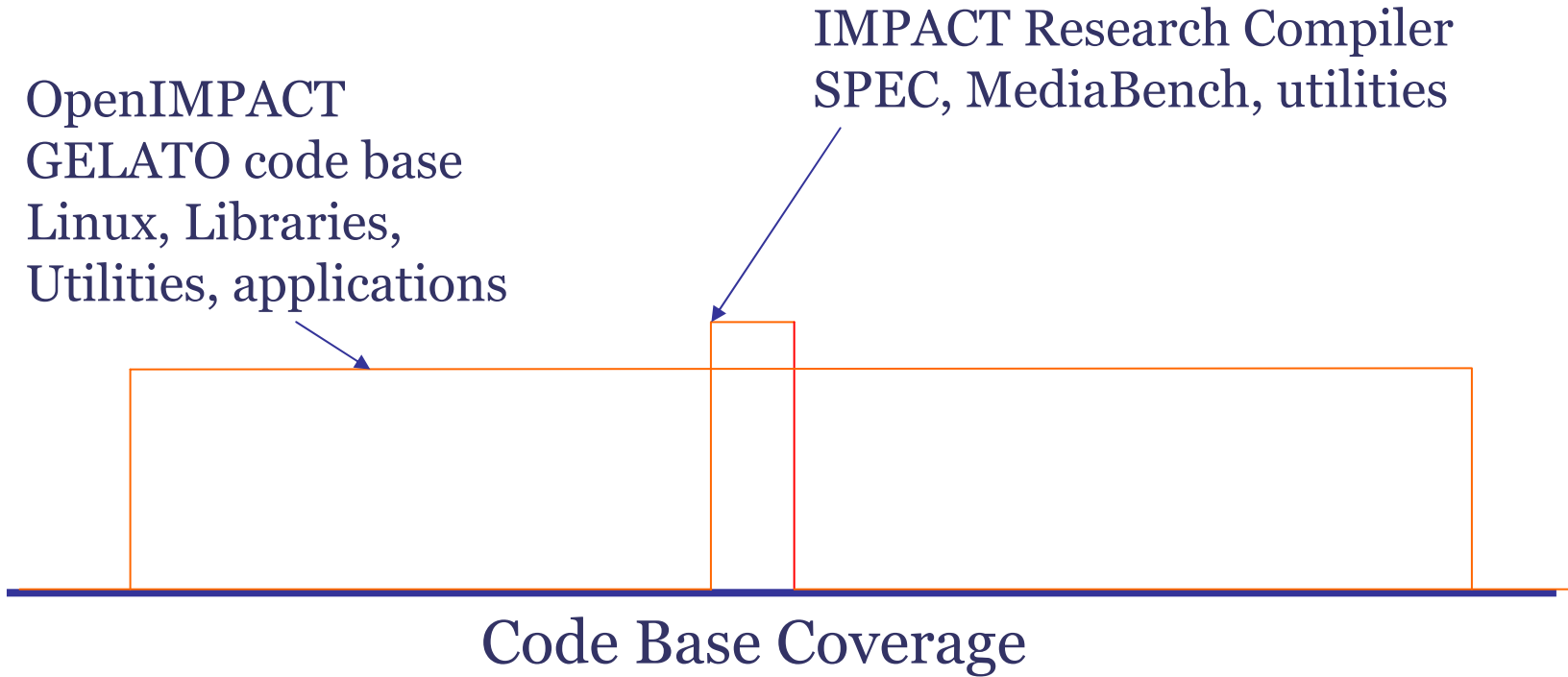
- In 2002, the U. of Illinois released the IMPACT Compiler into the open-source domain as part GELATO ([www.gelato.org](http://www.gelato.org))
- OpenIMPACT encompasses many of the advanced compilation techniques developed by the IMPACT research team, including
  - programmatic logic analysis
  - predicated compilation
  - interprocedural pointer analysis
  - instruction-level parallelism optimizations
  - profile-based optimization
  - speculative hyperblock acyclic and modulo scheduling
- Due to its heritage as a research compiler, OpenIMPACT is designed to achieve maximal output code performance with little concern for compile time and compiler memory usage.

# IMPACT SPECint2000 performance



# Main Challenge – widen the code base

---



We expect to define new research problems!

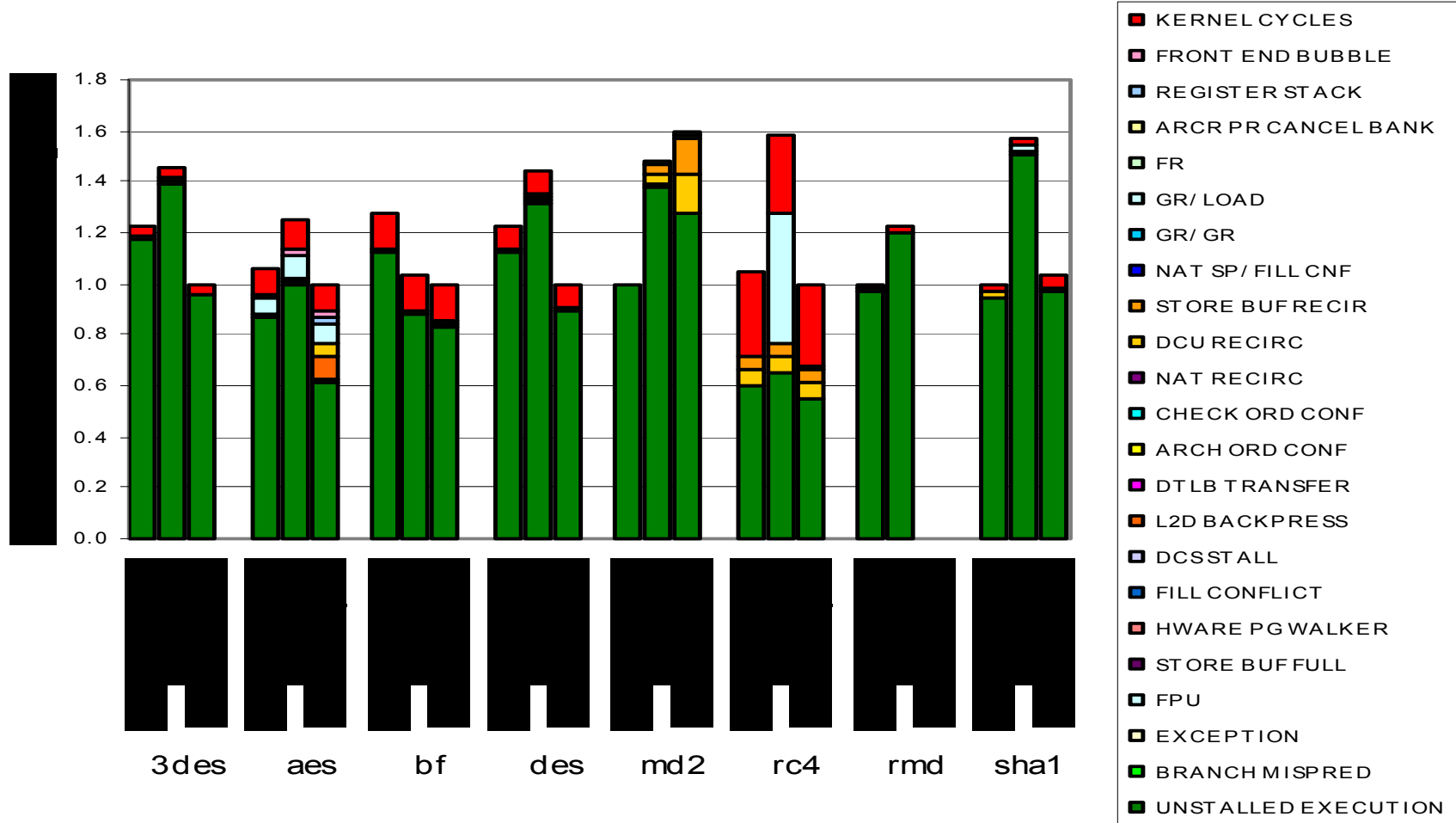
# The Current Model of Operation

---

- The OpenIMPACT engineering team contributes to the GELATO community by producing high-performance software object code using OpenIMPACT and computing resources at the U. of Illinois at Urbana-Champaign
  - All target applications are from the open-source domain.
  - The OpenIMPACT team has produced open-source applications and utilities executable binaries with record-setting performance on Itanium 2 platforms.
  - OpenIMPACT is also released to a community of users who need minimal support from the Illinois team.
  - OpenIMPACT also forms the main code base for future IMPACT research group

# Applications of Event Counters

Cycle accounting comparison

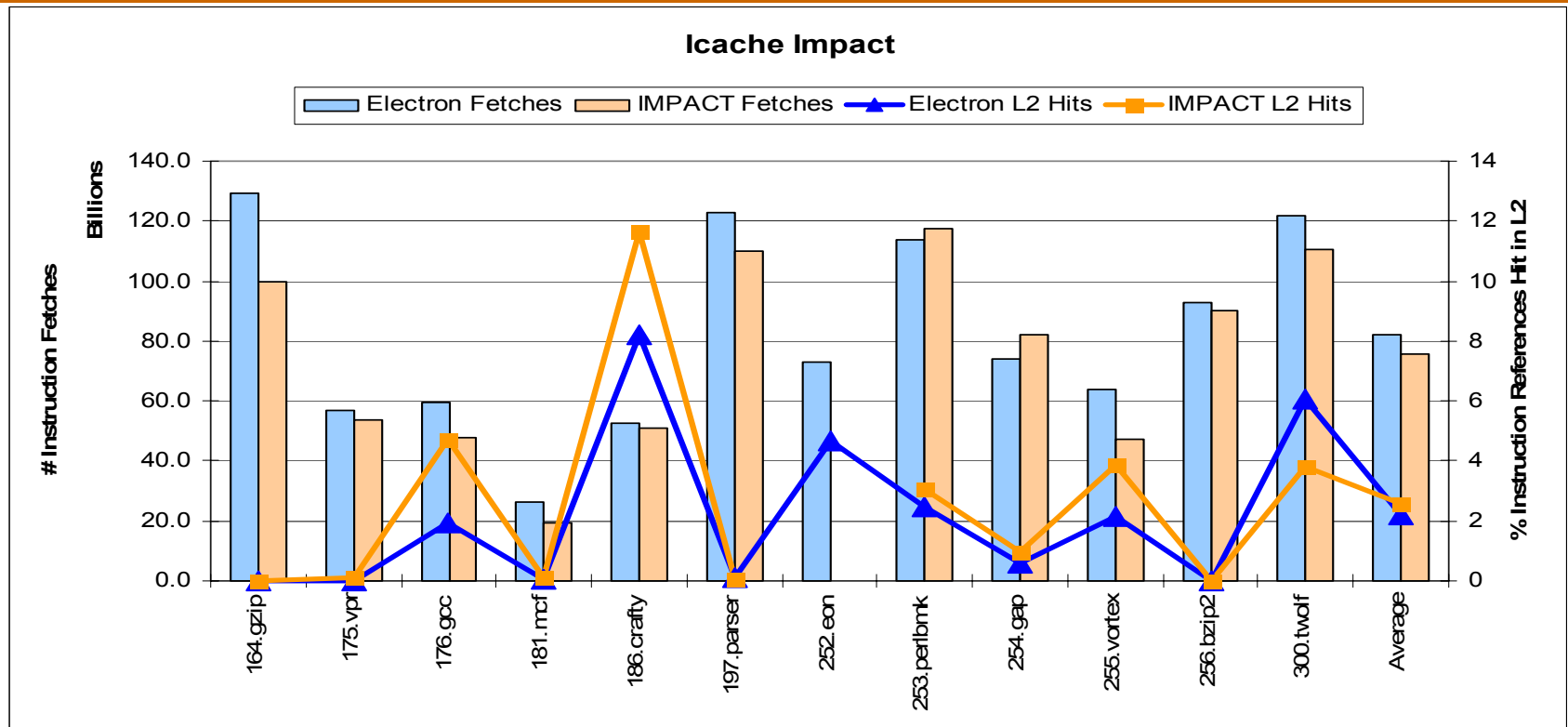


# Current Efforts and Goals

---

- To make OpenIMPACT fully compatible with GCC.
  - This involves scripts and internal enhancements to allow OpenIMPACT to be invoked with GCC makefiles and command lines.
  - The main problem is to make advanced interprocedural compilation and profile-based optimization widely applicable to real code base
- To compile Linux kernel, libraries, utilities, and applications with OpenIMPACT.
  - The ultimate goal is to achieve the full performance potential of IPF processors when executing Linux kernel, library, utility, and application binary code in terms of execution speed and code size.
  - This also requires sophisticated tools and processes for analyzing the quality of our output code.
- To make OpenIMPACT an advanced C++ compiler
  - There is an increasing demand from the GELATO community
- To identify and solve new research problems as we address issues when compiling real software (MARCO/DARPA project)
- To provide support for the executable binaries built and released by the OpenIMPACT team – see [www.gelato.uiuc.edu](http://www.gelato.uiuc.edu)

# Instruction cache impact

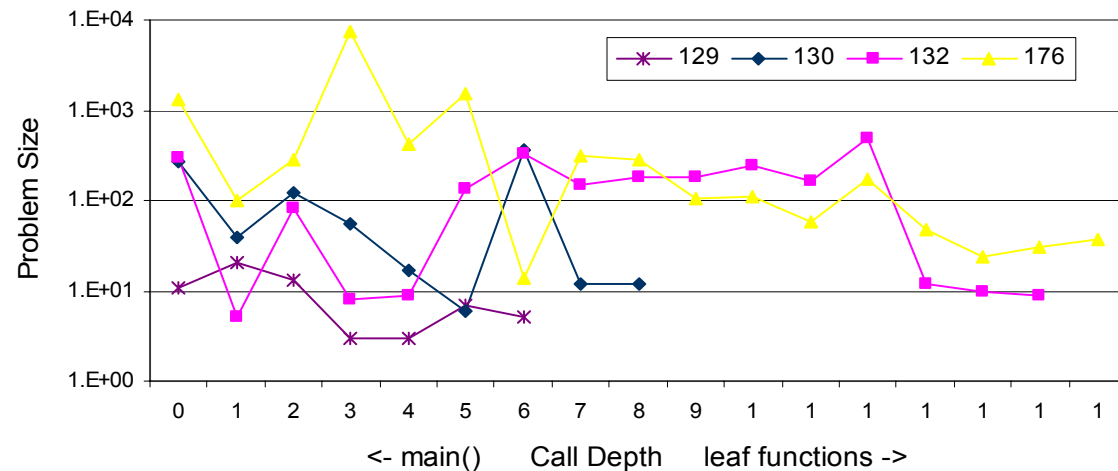
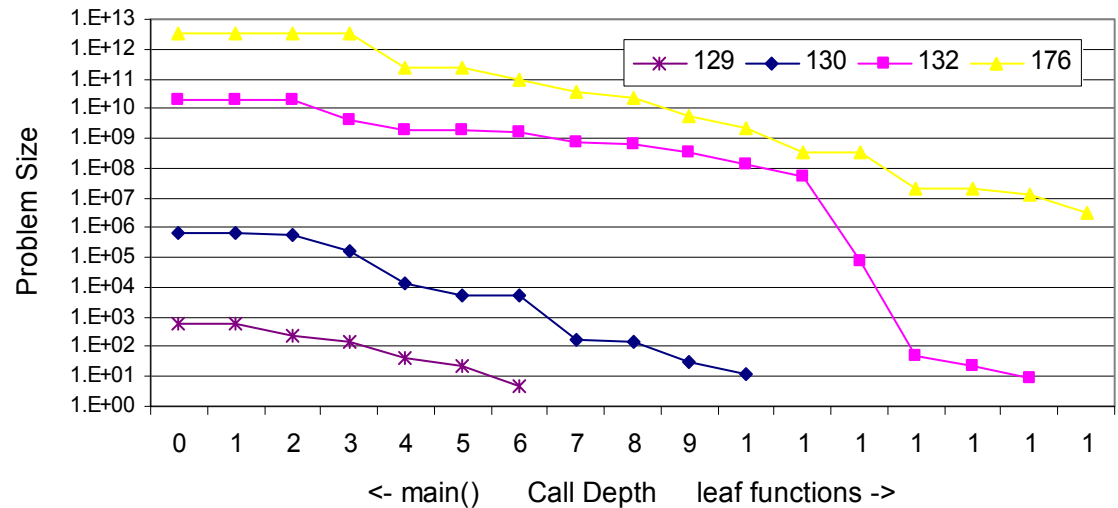


- Fewer cache accesses (higher fetch efficiency)
- Larger cache footprint (more secondary costs)
- Need to deliver ILP in view of total perf. picture

OpenIMPACT Technology and Performance Analysis

# Scaling precise analysis for the real world

- Naïve aggregation of points-to information causes exponential growth
- Our filtering techniques enable precise analysis by minimizing the size of summaries
- Principle: local encapsulation of complexity



# OpenIMPACT Team

---

- **John Sias** is the compiler architect for OpenIMPACT.
  - He is developing new ILP research compilation algorithms and future roadmap of OpenIMPACT.
- **Nacho Navarro** is expert in Linux kernel and libraries
  - He is responsible for addressing OS-specific challenges in compiling, profiling, analyzing, and optimizing kernel code.
- **Kevin Cernekee** is a compiler engineer under GELATO support
  - He is responsible for implementing software and procedures to streamline the development and testing processes, stress testing the compiler, and isolating bugs.
- **Robert Kidd** is a compiler engineer under GELATO support
  - He is responsible for scripting drivers as well as compiler-internal enhancements for increased GCC compatibility and compiler usability in general.
- **Andrew Schuh** is a compiler engineer under DARPA/MARCO support
  - He is responsible for front-end and high-level representation of OpenIMPACT.
- **Erik Nystrom** is interprocedural analysis architect
  - He is developing new scalable deep program analysis algorithms, innovative pointer analysis techniques in particular, to power future OpenIMPACT tools

# Acknowledgements

---

- **Former IMPACT members**
  - David August, Ben-Chung Cheng, Daniel Connors, Kevin Crozier, Brian Deitrich, John Gyllenhaal, Richard Hank, Teresa Johnson, Dan Lavery, Scott Mahlke, Le-Chun Wu
- **Intel Itanium Team**
  - Carole Dulong, John Crawford, Dan Lavery, Steve Skedzielewski, Jim Pierce
- **HP Itanium Team**
  - Richard Holman, Vatsa Santhanam, Carol Thompson, Richard Hank
- **HP Labs CAR Team**
  - Bob Rau, Mike Schlansker, Vinod Kathail, Scott Mahlke
- **HP Labs Linux Team**
  - Brian Lynn, David Mosberger, Hans Boehm, Stephane Eranian
- **HP Philanthropy – Itanium and Itanium 2 grants**
  - Karen Fontana, Tony Napolitan, Ralph Hyver, Chris Hsiung, Rob Bouzon, Gregg Peters
- **Intel - 2 alpha/beta Itaniums**
  - Carole Dulong, Richard Wirt