Workshop on Essential Abstractions in GCC

Introduction and Opening Remarks

GCC Resource Center
(www.cse.iitb.ac.in/grc)

Department of Computer Science and Engineering,
Indian Institute of Technology, Bombay

July 2009

Part 1

Genesis and Objectives of GRC

Outline

- Genesis and Objectives of GCC Resource Center
- Motivation behind the Workshop
- Philosophy and Pedogogy of the Workshop

How Did It All Begin?

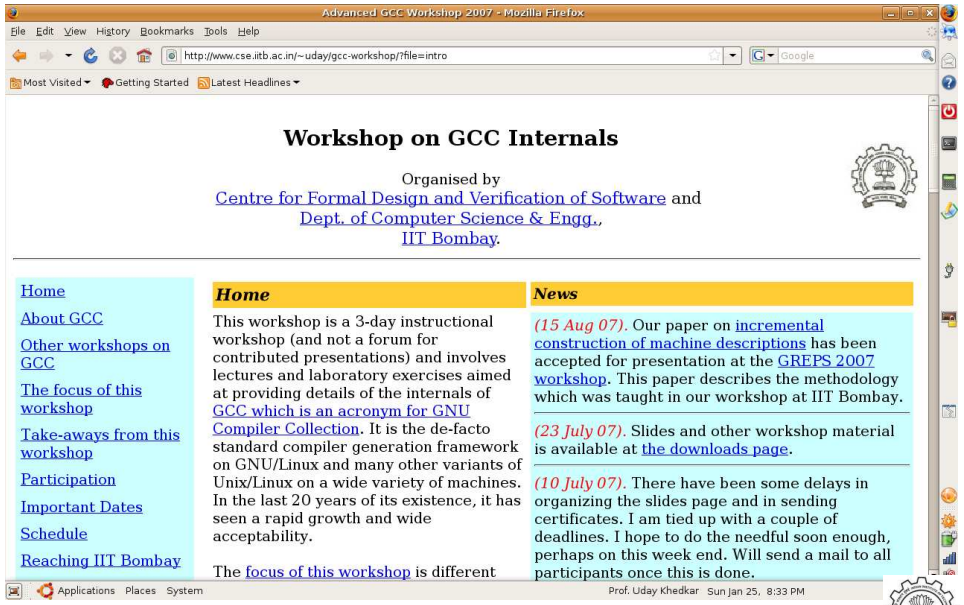
- An Informal Group
  - CSE faculty members at IITB: Uday Khedker
    Amitabha Sanyal
    Supratim Biswas
  - Reasonably long and deep experience of research in compilers
- A Desire

  Performing research grounded in theory and corroborated by empirical evidence
  - Exploring research issues in real compilers
  - Demonstrating the relevance and effectiveness of our research in real compilers

A Modest Start in 2003...

- Our Tool of Experiment The Gnu Compiler Collection
  - Compiler generation framework
    Stable compiler generated for several dozen targets
  - Millions of users
- Our Guinea Pigs
  Several unsuspecting M.Tech. students, external B.E. students, and project engineers

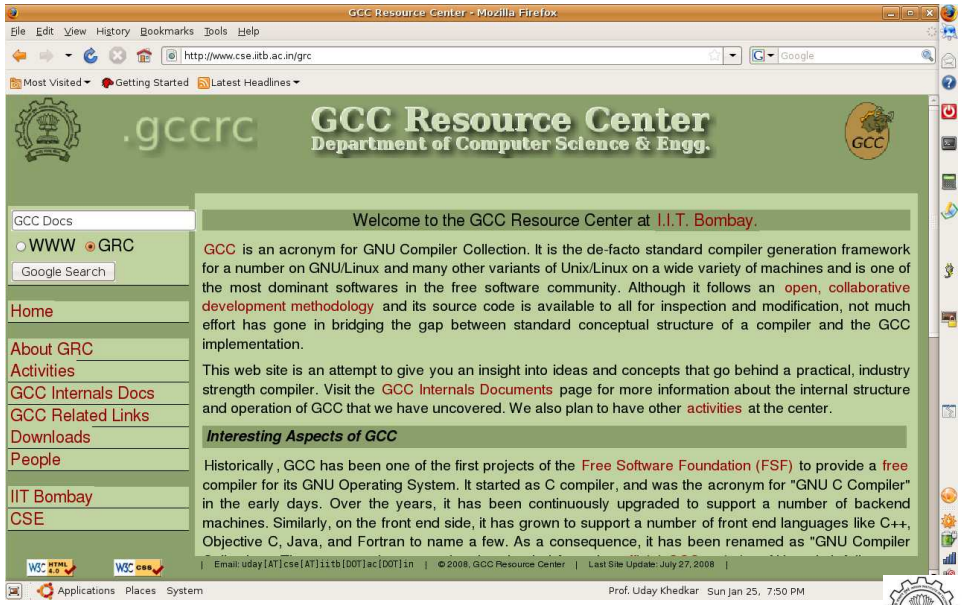
And Then in 2007...



And then in 2008...

Thanks to small seed grants from IITB and IBM Faculty Award...

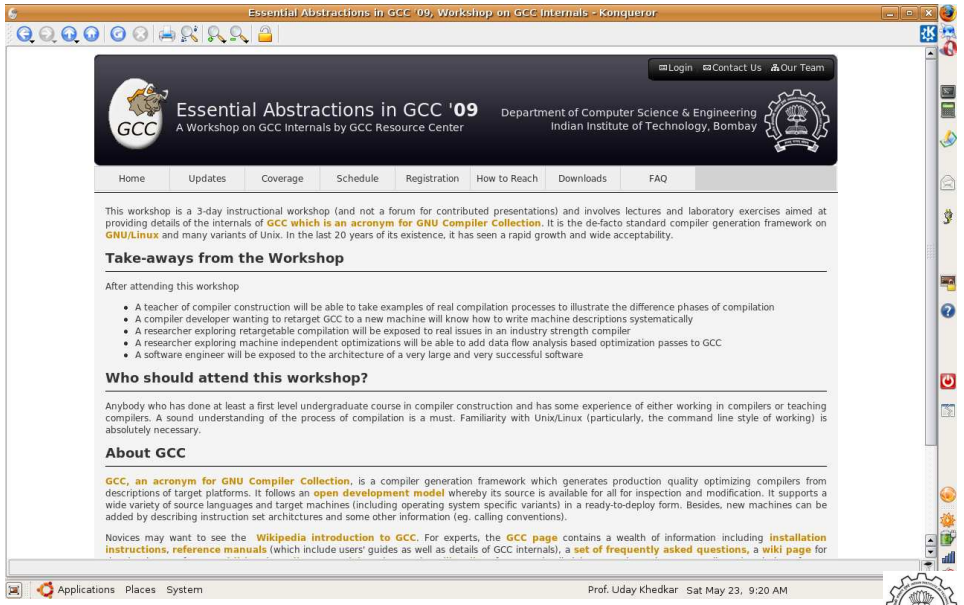
And then in 2008...



Finally in 2009...

- A generous grant from the Department of Information Technology, Ministry of Communication and Information Technology, Government of India.

Plan for July 2009...



Objectives of GCC Resource Center

1. To support the open source movement
   Providing training and technical know-how of the GCC framework to academia and industry.
2. To include better technologies in GCC
   Whole program optimization, Optimizer generation, Tree tiling based instruction selection.
3. To facilitate easier and better quality deployments/enhancements of GCC
   Restructuring GCC and devising methodologies for systematic construction of machine descriptions in GCC.
4. To bridge the gap between academic research and practical implementation
   Designing suitable abstractions of GCC architecture

Broad Research Goals of GCC Resource Center

- Using GCC as a means
  - Adding new optimizations to GCC
  - Adding flow and context sensitive analyses to GCC (In particular, pointer analysis)
- Using GCC as an end in itself
  - Changing the retargetability mechanism of GCC
  - Cleaning up the machine descriptions of GCC
  - Systematic construction of machine descriptions
  - Facilitating optimizer generation in GCC

GRC Training Programs

Title	Target	Objectives	Mode	Duration
Workshop on Essential Abstractions in GCC	People interested in deploy… enhanc…	Explaining the essential	Lectures, demonstrations, …d practicals (experiements and assignments)	Three days
Tutorial on Essential Abstractions in GCC	People interested in knowing about issues in deploying or enhancing GCC	Explaining the essential abstractions in GCC to ensure a quick ramp up into GCC Internals	Lectures and demonstrations	One day
Workshop on Compiler Construction with Introduction to GCC	College teach…	…d	Lectures, …onstrations, …racticals …periements and assignments)	Seven days
Tutorial on Demystifying GCC Compilation	Students	Explaining the translation sequence of GCC through gray box probing (i.e. by examining the dumps produced by GCC)	Lectures and demonstrations	Half day

3, 4, and 5 July, 2009
IIT Bombay, Mumbai

10 to 17 July, 2009
Rajagiri School of Engg. and Tech., Cochin

Part 2

Motivation Behind this Workshop

Motivation Behind this Workshop

- To understand GCC well :-)
- Reasonably quickly

Why is Understanding GCC Difficult?

Some of the obvious reasons:

- *Comprehensiveness*
  GCC is a production quality framework in terms of completeness and practical usefulness.
- *Open development model*
  Leads to heterogeneity of the design.
  However now the main plan is vetted by the steering committee.
- *Rapid versioning*
  GCC maintenance is a race against time.

July 09 Introduction: Motivation Behind this Workshop 12/28

Why is Understanding GCC Difficult?

Notes

Essential Abstrations in GCC GCC Resource Center, IIT Bombay

Comprehensiveness of GCC 4.3.1: Wide Applicability

- Input languages supported:
  C, C++, Objective-C, Objective-C++, Java, Fortran, and Ada
- Processors supported in standard releases:
  - Common processors:
    Alpha, ARM, Atmel AVR, Blackfin, HC12, H8/300, IA-32 (x86), x86-64, IA-64, Motorola 68000, MIPS, PA-RISC, PDP-11, PowerPC, R8C/M16C/M32C, SPU, System/390/zSeries, SuperH, SPARC, VAX
  - Lesser-known target processors:
    A29K, ARC, ETRAX CRIS, D30V, DSP16xx, FR-30, FR-V, Intel i960, IP2000, M32R, 68HC11, MCORE, MMIX, MN10200, MN10300, Motorola 88000, NS32K, ROMP, Stormy16, V850, Xtensa, AVR32
  - Additional processors independently supported:
    D10V, LatticeMico32, MeP, Motorola 6809, MicroBlaze, MSP430, Nios II and Nios, PDP-10, TIGCC (m68k variant), Z8000, PIC24/dsPIC, NEC SX architecture

July 09 Introduction: Motivation Behind this Workshop 13/28

Comprehensiveness of GCC 4.3.1: Wide Applicability

Notes

Essential Abstrations in GCC GCC Resource Center, IIT Bombay

Comprehensiveness of GCC 4.3.1: Size

- Pristine compiler sources (download tarball)
  - Lines of C code : 2122047
  - Lines of MD code : 245933
  - Lines of total code : 2367980
  - Total authors (approx) : 65
  - Backend directories : 35
- Generated source for i386 (input language: c)
  - Total lines of code : 439703
  - Total lines of .c files code : 334855
  - Total number of .c files : 16
  - Total lines of .h files : 104848
  - Total number of .h files : 274

July 09 Introduction: Motivation Behind this Workshop 14/28

Comprehensiveness of GCC 4.3.1: Size

Notes

Essential Abstrations in GCC GCC Resource Center, IIT Bombay

Open Source and Free Software Development Model

- The Cathedral and the Bazaar
  Eric S Raymond, 1999.
- Cathedral: Total Centralized Control
  Design, implement, test, release
  - + Facilitates a homogeneous, coherent, and well-planned design
  - − Could be restricted by the vision of a few people
- Bazaar: Total Decentralization
  Release early, release often, let users fix bugs
  - − Could lead to heterogeneity in the design
  - + Is not restricted by the vision of a few people

July 09 Introduction: Motivation Behind this Workshop 15/28

Open Source and Free Software Development Model

Notes

Essential Abstrations in GCC GCC Resource Center, IIT Bombay

The Bazaar Approach

Release early, release often, let users fix bugs

- Brooks' law (The Mythical Man Month, 1975)
  - 12 man month effort
  - 1 person working for 12 months
    OR
    12 persons working for 1 month?
- Bazaar approach believes that the two somewhat equivalent in internet-based distributed development.
- "Given enough eyeballs, all bugs are shallow".
  Code errors, logical errors, and architectural errors.

A combination of the two seems more sensible

July 09 Introduction: Motivation Behind this Workshop 16/28

The Bazaar Approach

Notes

Essential Abstrations in GCC GCC Resource Center, IIT Bombay

The Current Development Model of GCC

- GCC Steering Committee: Free Software Foundation has given charge
  - Major policy decisions
  - Handling Administrative and Political issues
- Release Managers:
  - Coordination of releases
- Maintainers:
  - Usually area/branch/module specific
  - Responsible for design and implementation

July 09 Introduction: Motivation Behind this Workshop 17/28

The Current Development Model of GCC

Notes

Essential Abstrations in GCC GCC Resource Center, IIT Bombay

The Current Development Model of GCC

- Proposing changes
  - Extensive discussions over mailing lists
  - Submissions to `gcc-patches@gcc.gnu.org`
  - Major changes are forked off as an independent branch which is later merged with the main code
- Reviwers:
  - Can be general/global or area/branch/module specific
  - Can approve changes suggested by others
  - Need approval of other reviewers for their own changes
- Maintainers:
  - Can approve changes suggested by others
  - Do not need approval of reviewers for their own changes in their area/branch/module

July 09 Introduction: Motivation Behind this Workshop 18/28

The Current Development Model of GCC

Notes

Essential Abstrations in GCC GCC Resource Center, IIT Bombay

Why is Understanding GCC Difficult?

Deeper reason: GCC is not a *compiler* but a *compiler generation framework*

There are two distinct gaps that need to be bridged:

- Input-output of the generation framework: The target specification and the generated compiler
- Input-output of the generated compiler: A source program and the generated assembly program

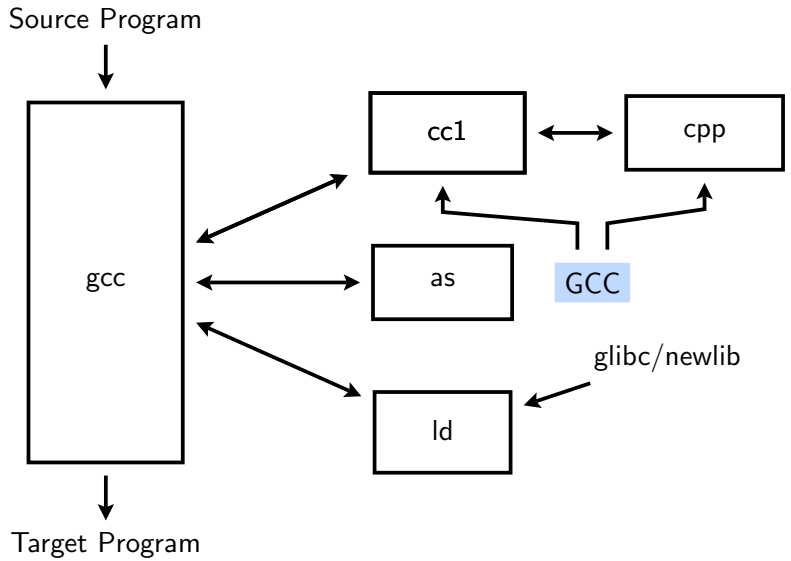
July 09 Introduction: Motivation Behind this Workshop 19/28

Why is Understanding GCC Difficult?

Notes

Essential Abstrations in GCC GCC Resource Center, IIT Bombay

The Gnu Tool Chain



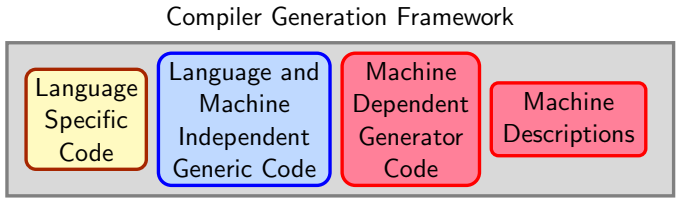
July 09 Introduction: Motivation Behind this Workshop 20/28

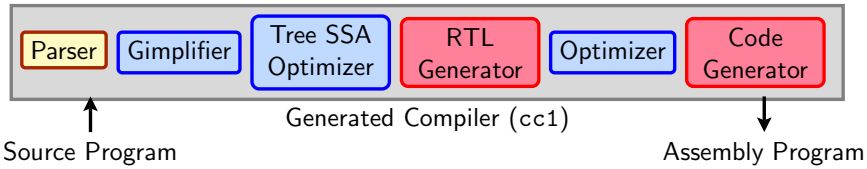
The Gnu Tool Chain

Notes

Essential Abstrations in GCC GCC Resource Center, IIT Bombay

The Architecture of GCC





Essential Abstrations in GCC GCC Resource Center, IIT Bombay

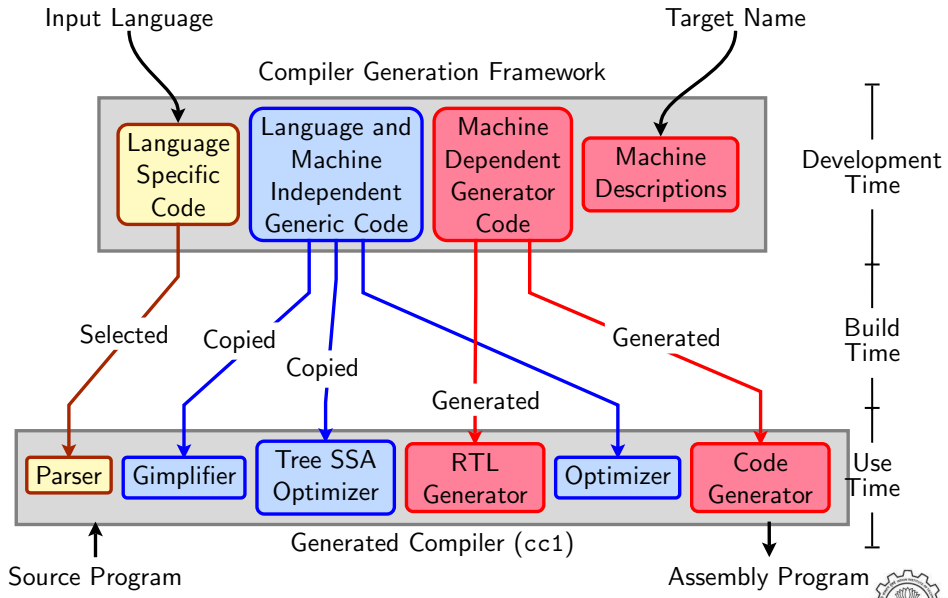
July 09 Introduction: Motivation Behind this Workshop 21/28

The Architecture of GCC

Notes

Essential Abstrations in GCC GCC Resource Center, IIT Bombay

The Architecture of GCC



The Architecture of GCC

Notes

Part 3

Our Philosophy and Pedogogy

Philosophy and Pedogogy

Twin goals of this workshop:

- Learning how to learn GCC
- Striking a balance between theory and practice

Philosophy and Pedogogy

- We will
  - Explain configuration and building of GCC
  - Explain essential abstractions related to compilation
    The key intermediate representations and their manipulations
  - Explain essential abstractions related to generation of a compiler
    The machine descriptions and their influence on compilation
- You will
  - Build and run GCC
  - Examine various IR dumps produced by GCC
  - Add a new machine description and systematically enhance it

July 09 Introduction: Our Philosophy and Pedogogy 23/28

Philosophy and Pedogogy

Notes

Essential Abstrations in GCC GCC Resource Center, IIT Bombay

Takeaways from this Workshop

- A programmer will get a better compiler
- A compiler professional will be able to deploy and enhance GCC much more easily.
- A compiler researcher will be able to use GCC for research much better.
- A compiler teacher will be able to strike a better balance between theory and practice.
- A compiler student will be exposed to issues in real compilers.

Schedule: Day 1

09:00 to 09:30	Registration
09:30 to 10:00	Introduction and opening remarks
10:00 to 11:15	Getting started with GCC: Configuration and building
11:15 to 11:30	Tea break
11:30 to 13:00	(Lab) Configuration and building
13:00 to 14:00	Lunch
14:00 to 15:00	Gray Box Probing of GCC An introduction to IRs
15:00 to 15:45	Introduction to Gimple IR
15:45 to 16:00	Tea break
16:00 to 17:30	(Lab) Adding a gimple pass
19:00 to 20:00	Inspecting and debugging gcc code (Optional Session) (use of cscope, ctags, gdb etc.)
20:00 to 21:00	Dinner

July 09 Introduction: Our Philosophy and Pedogogy 25/28

Schedule: Day 1

Notes

Essential Abstrations in GCC GCC Resource Center, IIT Bombay

Schedule: Day 2

09:30 to 10:15	Introduction to RTL
10:15 to 11:00	An overview of retargetability and an introduction to machine descriptions
11:00 to 11:15	Tea break
11:15 to 13:00	(Lab) Adding an RTL pass
13:00 to 14:00	Lunch
14:00 to 15:30	Spim machine descriptions: Level 0 and 1
15:30 to 15:45	Tea break
15:45 to 17:30	(Lab) spim machine descriptions
18:30 to 20:00	(Optional session) An Overview of research projects in GCC Resource Center
20:00 to 21:00	Dinner

July 09 Introduction: Our Philosophy and Pedogogy 26/28

Schedule: Day 2

Notes

Essential Abstrations in GCC GCC Resource Center, IIT Bombay

Schedule: Day 3

09:30 to 10:45	spim machine descriptions levels 2 and 3
10:45 to 11:00	Tea break
11:00 to 13:00	(Lab) spim machine descriptions
13:00 to 14:00	Lunch
14:00 to 15:30	The Generic Data Flow Analyzer in GCC
15:30 to 15:45	Tea break
15:45 to 16:15	The retargetability model of GCC
16:15 to 17:30	Concluding session

July 09 Introduction: Our Philosophy and Pedogogy 27/28

Schedule: Day 3

Notes

Essential Abstrations in GCC GCC Resource Center, IIT Bombay

Questions?