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



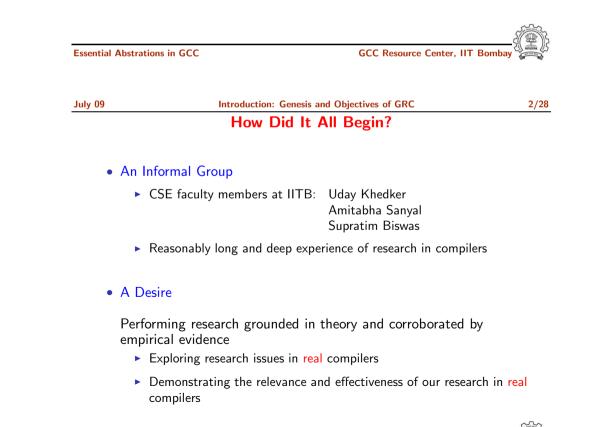
July 2009

Part 1

Genesis and Objectives of GRC

July 09

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



Introduction: Genesis and Objectives of GRC

3/28

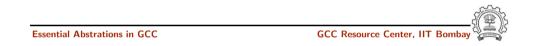
5/28

A Modest Start in 2003...

- 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



July 09

Introduction: Genesis and Objectives of GRC And then in 2008...

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



Introduction: Genesis and Objectives of GRC

4/28

And Then in 2007...

◆	्रि • Google 🍳
ost Visited ▼ 🌘 Getting Started 🔂 Latest Headlines ▼	
Workshop on GCC Ir Organised by <u>Centre for Formal Design and Verifica</u> <u>Dept. of Computer Science</u> <u>IIT Bombay.</u>	ation of Software and
ome Home	
IIOme	News
Dout GCC This workshop is a 3-day instructional ther workshops on CC workshop (and not a forum for contributed presentations) and involves lectures and laboratory exercises aimed at providing details of the internals of GCC which is an acronym for GNU borkshop GCC ake-aways from this orkshop GCC which is an acronym for GNU compiler Collection. It is the de-facto standard compiler generation framework on GNU/Linux and many other variants of Unix/Linux on a wide variety of machines. In the last 20 years of its existence, it has seen a rapid growth and wide acceptability. eaching IIT Bombay The focus of this workshop is different	 (15 Aug 07). Our paper on incremental construction of machine descriptions has been accepted for presentation at the GREPS 2007 workshop. This paper describes the methodology which was taught in our workshop at IIT Bombay. (23 July 07). Slides and other workshop material is available at the downloads page. (10 July 07). There have been some delays in organizing the slides page and in sending certificates. I am tied up with a couple of deadlines. I hope to do the needful soon enough, perhaps on this week end. Will send a mail to all participants once this is done.

uly 09	Introduction: Genesis and Objectives of GRC	5	/28
	And then in 2008		-
	GCC Resource Center - Mozilla Firefox		×
e <u>E</u> dit ⊻iew Hi <u>s</u> tory <u>B</u> ookm	marks Iools Help		ं 👧
🎍 🔶 🕲 🎓 🗖	🙆 http://www.cse.iitb.ac.in/grc	jle 🔇	
Most Visited 👻 🌘 Getting Star	arted 🔂Latest Headlines 🕶		0
.go	CCTC GCC Resource Center Department of Computer Science & Engg.	GCC	
GCC Docs	Welcome to the GCC Resource Center at I.I.T. Bombay.		
Google Search	GCC is an acronym for GNU Compiler Collection. It is the de-facto standard compiler gener for a number on GNU/Linux and many other variants of Unix/Linux on a wide variety of machin		3
lome	the most dominant softwares in the free software community. Although it follows an op- development methodology and its source code is available to all for inspection and modific effort has gone in bridging the gap between standard conceptual structure of a compile implementation.	cation, not much	-
ctivities CC Internals Docs	This web site is an attempt to give you an insight into ideas and concepts that go behind a p strength compiler. Visit the GCC Internals Documents page for more information about the i and operation of GCC that we have uncovered. We also plan to have other activities at the ce	nternal structure	
ownloads	Interesting Aspects of GCC		
'eople T Bombay SE	Historically, GCC has been one of the first projects of the Free Software Foundation (FSF) to compiler for its GNU Operating System. It started as C compiler, and was the acronym for "C in the early days. Over the years, it has been continuously upgraded to support a num machines. Similarly, on the front end side, it has grown to support a number of front end lang Objective C, Java, and Fortran to name a few. As a consequence, it has been renamed as	NU C Compiler" ber of backend guages like C++,	() () () () () () () () () () () () () (
WSC 4.0 WSC CBB	Email: uday[AT]cse[AT]iitb[D0T]ac[D0T]in ©2008, GCC Resource Center Last Site Update: July 27, 2008	0	
] 🧔 Applications Places S	System Prof. Uday Khedkar Sun Jan 25, 7:50 PM	5	T2

Essential Abstrations in GCC

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



Essential Abstrations in GCC

July 09	Introduction: Genesis and
	Objectives of GCC F

roduction: Genesis and Objectives of GRC

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

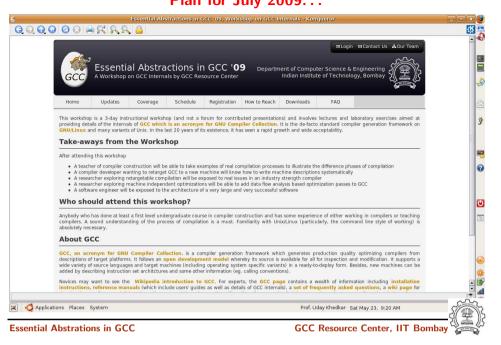
Restructuring GCC and devising methodologies for systematic construction of machine descriptions in GCC.

4. To bridge the gap between academic research and practical implementation

Designing suitable abstractions of GCC architecture

luly 09	
---------	--

Plan for July 2009...



July 09 Introduction: Genesis and Objectives of GRC
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

9/28

GRC Training Programs

Title	Target	Objectives	Mode	Duration
Workshop on	People interested	Explaining the essential	Lectures,	Three
Essential	in deplovir 3.	4, and 5 July, 2009	demonstrations,	days
Abstractions in	enhand	Г Bombay, Mumbai	d practicals	
GCC		Dombay, Wumbar	experiements	
			and assignments)	
Tutorial on	People interested	Explaining the essential	Lectures and	One day
Essential	in knowing about	abstractions in GCC to	demonstrations	
Abstractions in	issues in	ensure a quick ramp up		
GCC	deploying or	into GCC Internals		
	enhancing GCC			
Workshop on	College teach		Lectures,	Seven
Compiler		10 to 17 July, 2009		days
Construction	Rajagi	ri School of Engg. and	racticals	
with Introduction		Tech., Cochin	periements	
to GCC			and assignments)	
Tutorial on	Students	Explaining the translation	Lectures and	Half day
Demystifying		sequence of GCC through	demonstrations	
GCC		gray box probing (i.e. by		
Compilation		examining the dumps		
		produced by GCC)		

Essential Abstrations in GCC

GCC Resource Center, IIT Bombay

July 09

Introduction: Motivation Behind this Workshop

11/28

Motivation Behind this Workshop

Part 2

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.

Notes



• 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	13/2
	Comprehensiveness of GCC 4.3.1: Wide Applicability	



Essential Abstrations in GCC

Essential Abstrations in GCC

Introduction: Motivation Behind this Workshop

July 09

Notes

Introduction: Motivation Behind this Workshop Comprehensiveness of GCC 4.3.1: Size

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 15/28 **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

Introduction: Motivation Behind this Workshop 15/28 **Open Source and Free Software Development Model**







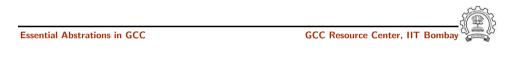
Essential Abstrations in GCC GCC Resource Center, IIT July 09

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
	The Current Development Model of GCC

• GCC Steering Committee: Free Software Foundation has given charge

GCC Resource Center, IIT

- Major policy decisions
- Handling Administrative and Political issues
- Release Managers:
 - Coordination of releases
- Maintainers:
 - Usually area/branch/module specific
 - Responsible for design and implementation

Notes

Essential Abstrations in GCC



July 09	Introduction: Motivation Behind this Workshop	17/
	The Current Development Model of GCC	

Notes





16/28

/28

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

Essential Abstrations in GCC	GCC Resource Center, IIT Bombay

July 09

Introduction: Motivation Behind this Workshop Why is Understanding GCC Difficult?

Deeper reason:	GCC is not a	compiler	but a	compiler generation	
fue and encounts					

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

Notes



Why is Understanding GCC Difficult?

Notes

July 09





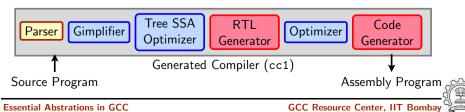
Source Program

July 09

Notes



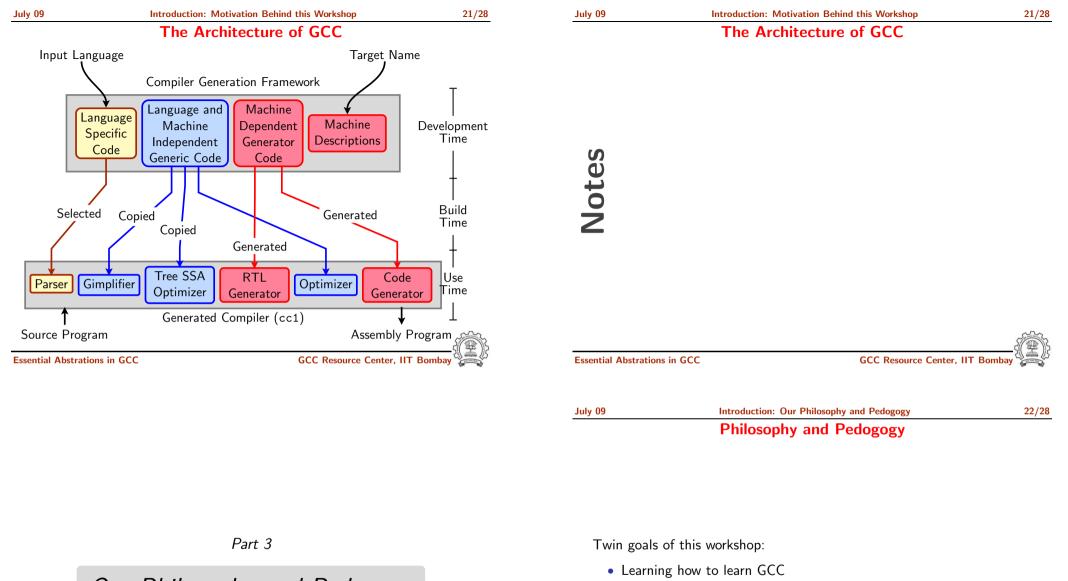
cc1 cpp GCC gcc as glibc/newlib ld Target Program **Essential Abstrations in GCC** GCC Resource Center, IIT Bombay Introduction: Motivation Behind this Workshop 21/28 July 09 The Architecture of GCC Compiler Generation Framework Machine Language and Language Dependent Machine Machine Specific Descriptions Independent Generator Code Generic Code Code



	The Architecture of GCC	
July 09	Introduction: Motivation Behind this Workshop	21/28
Essential Abstrations	in GCC GCC Resource Center, I	T Bombay

Notes





Our Philosophy and Pedogogy

• Striking a balance between theory and practice



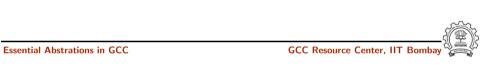
Introduction: Our Philosophy and Pedogogy

Philosophy and Pedogogy

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 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.



Essential Abstrations in GCC

GCC Resource Center, IIT Bombay



24/28

Introduction: Our Philosophy and Pedogogy

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	

Essential Abstrations in GCC

GCC Resource Center, IIT Bombay

GCC Resource Center, IIT Bombay

July 09

Introduction: Our Philosophy and Pedogogy

26/28

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

Essential Abstrations in GCC

July 09

July 09

Notes

Introduction: Our Philosophy and Pedogogy

26/28

25/28

Schedule: Day 2

Introduction: Our Philosophy and Pedogogy

Schedule: Day 1



_____ś

GCC Resource Center, IIT Bomb

Schedule: Day 3

27/28

28/28

July 09

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

Notes

Essential Abstrations in GCC	GCC Resource Center, IIT Bombay

July 09	
---------	--

Introduction: Our Philosophy and Pedogogy Questions? Essential Abstrations in GCC



GCC Resource Center, IIT Bombay