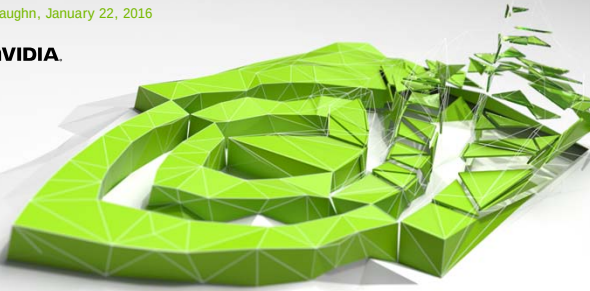


CS-314 Panel Discussion

Ron Vaughn, January 22, 2016



Communicating customer “wants”

Our software is often “low level”, e.g. video driver

- Nvidia highly depends on industry standard specifications
 - We contribute to the specs
 - Then follow up and implement the specs
- We receive requirements directly from partners
 - If you are partnered with Google working on Android, they'll communicate what they want
 - Need to pass conformance test suites (which specifically tests a spec)
- Bleeding edge: The customer may not know what they want



Techniques to understand and solve problems

- “Problem solving” is difficult to teach. And difficult to learn on your own. Learn from your mentor and teammates.
- Over time you will grow your “tool belt” of problems you can recognize and solve.
- There are classes of problems. Eventually you quickly recognize the problem class and know your approach to solving the problem.
 - Race conditions
 - Memory corruption
 - Multi-threaded issues
 - Performance issues
 - Tool chain or infrastructure issue
 - Security or permission violation
 - Cascading errors
 - Fence post error

What teamwork skills do you expect?

- Become an Nvidiaian
- Help teammates in need (look at their code, help debug, brainstorm)
- Come through in crunch time
- Most engineers are naturally good teammates
- Senior engineers are not necessarily technically better - they are leaders
- If you are going to be a lone-wolf, you better be exceptionally good



Collaborative tools

Learning tools is very important

Source control: internal git, perforce  

Code review: gerrit 

Bug tracking: internal system (think bugzilla)

Project management: Jira 

Build farm: internal system

Automated test farm: internal system

All of these systems are tied together. And there are many more.

All of these systems are way more complicated than new-hires are used to.

You must master the collaboration tools to become an effective engineer



Collaborative tools

Using git in college is not like using git at Nvidia

- A single source code tree:
 - 650 individual git repositories
 - Each repository at the correct location, each at the correct branch
 - 50G (before you compile)

On a scale of 1 to 10, you use git at a ‘4’. We go to 11.



Communication skills at NVidia

Top 5

- Email
- Email
- Email
- Email
- Email

7  NVIDIA

Communication skills

OK... There is more to communication than just email

- Concise emails are invaluable.
- Maintaining wiki pages. Detailed accurate instructions.
- Running a meeting, with an agenda and good meeting notes.
- Concise PowerPoint presentations
 - Project overview
 - Technical presentation
 - Schedules and roadmaps

Good communication skills dramatically increase your impact as an engineer.
Good communication skills are required to move into management.

8  NVIDIA

