NanoCMOS:

Simulating Nanometre-Scale Transistors and Circuits on the NGS

Clive Davenhall
National e-Science Centre
(acd@staffmail.ed.ac.uk)

4 February 2010

NGS Roadshow at the University of Hull

NanoCMOS

- Characterising the variability inherent in modern CMOS chip design.
- Addressing some of the fundamental problems in designing these devices.
- Multi-institution (and multi-discipline) collaboration.
- Principal Investigator: Prof. Asen Asenov, Department of Electronics and Electrical Engineering, University of Glasgow.
- The project will run for 4 years: 2007 to 2010.
- Funded by funded by the UK Engineering and Physical Sciences Research Council (EPSRC) with additional contributions from industrial partners.
- Total funding of £5.3 million.

NanoCMOS

Complementary Metal-Oxide Semiconductor

- Characterising the variability inherent in modern CMOS chip design.
- Addressing some of the fundamental problems in designing these devices.
- Multi-institution (and multi-discipline) collaboration.
- Principal Investigator: Prof. Asen Asenov, Department of Electronics and Electrical Engineering, University of Glasgow.
- The project will run for 4 years: 2007 to 2010.
- Funded by funded by the UK Engineering and Physical Sciences Research Council (EPSRC) with additional contributions from industrial partners.
- Total funding of £5.3 million.

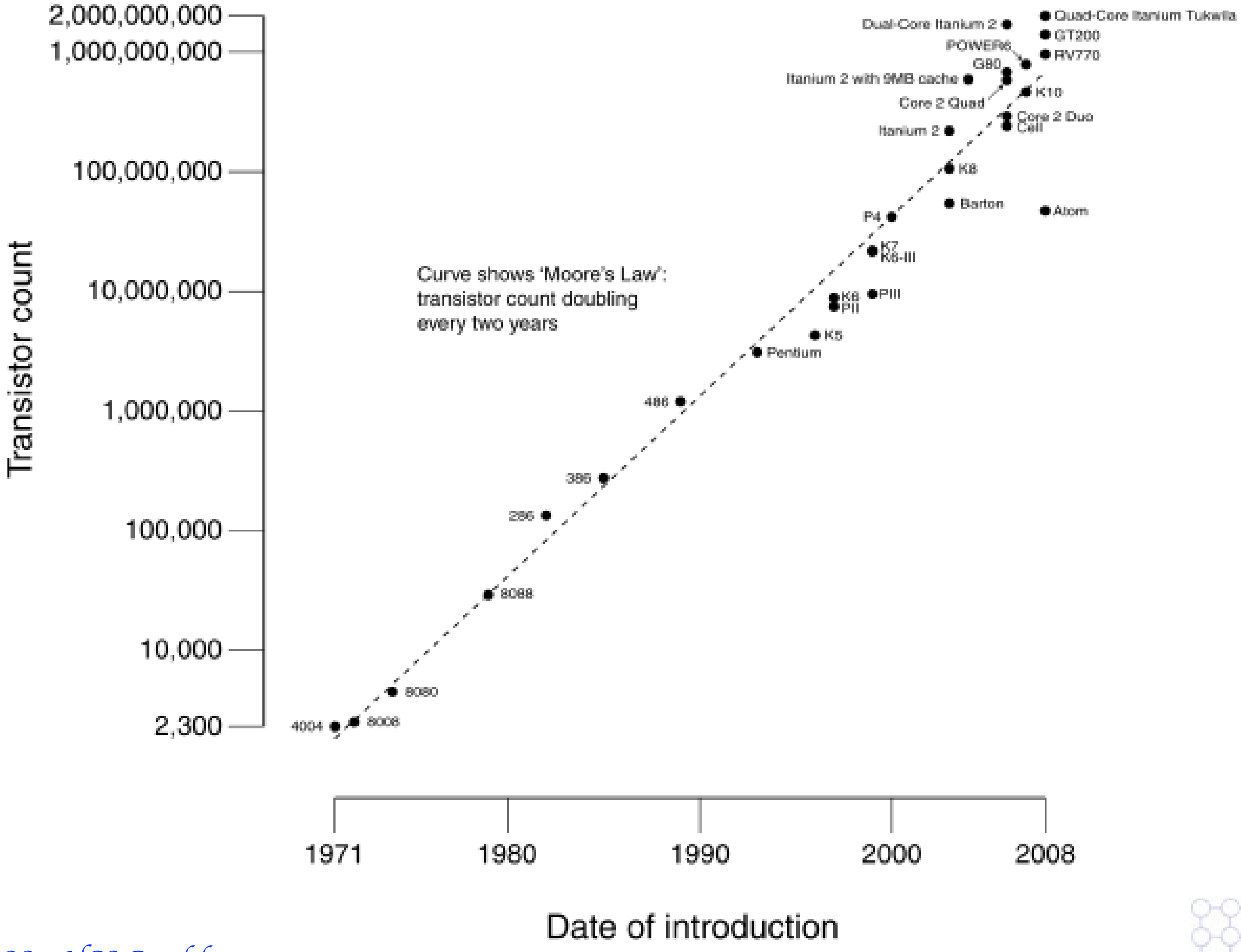
Partners

- University of Glasgow
  - Department of Electronics and Electrical Engineering
  - National eScience Centre
- University of Manchester
  - School of Computing Science
- University of Southampton
  - School of Electronics & Computer Science
- University of Edinburgh
  - Department of Electronics and Electrical Engineering
  - National eScience Centre
- University of York
  - Department of Electronics
- Industrial Partners
  - Fujitsu, ARM, Wolfson Microelectronics, Freescale Semiconductor, Synopsis and National Semiconductor

The Problem (1)

- A commonplace that smaller and more capable CMOS devices become available as time progresses,
  - consequently computers become more powerful.
- Captured in 'Moore's Law':
  - loosely applied since the integrated circuit was invented in 1958,
  - 'happy scaling' that has led to the enormous success of the computer industry,
  - may be coming to an end.
- Integrated circuit design is now in the nano-CMOS regime:
  - 40 nm MOSFET transistors are already in mass-production,
  - transistors smaller than 10 nm anticipated by 2018.

CPU Transistor Counts 1971-2008 & Moore's Law

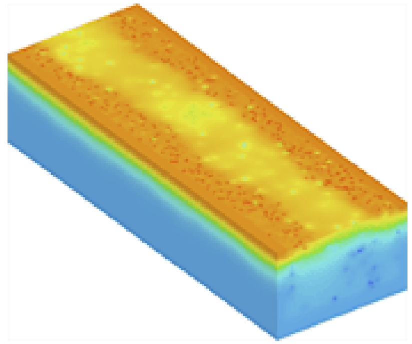


NanoCMOS – NGS Roadshow.

National e-Science Centre

The Problem (2)

- Major challenges in further reducing the size of transistors:
  - variability in device characteristics,
  - need to introduce novel architectures.
- Variability between devices:
  - several sources,
  - some fundamental to the quantisation (or discreteness) of charge and matter,
  - has a statistical impact on device behaviour,
  - and is major source of device variability
- Consequently device variations:
  - are stochastic,
  - cannot be reduced by better process control.



Random discrete dopants and line edge roughness in a 35 nm MOSFET representative of the transistors that are now in mass production.

The Problem (3)

- This variability requires changes in the way that circuits and systems are designed.
  - the variability must be fully characterised,
  - strong links needed between the fundamental device technology and circuit and system design,
  - design methods need to change to accommodate the increased variability.
- Variability is assessed by monte-carlo simulations:
  - multiple runs of already lengthy calculations,
  - the problem has an inherent coarse-grained parallelism…
  - … which is ideally suited to super-computers / clusters,
  - NanoCMOS uses a variety of resources, including the NGS.
- Distributed teams need to share results,
  - eg, Glasgow runs device simulations which are made available to other groups.
- Security considerations: some designs are proprietary and commercially-sensitive.

Level of Abstraction and Complexity

Spinaker

Manchester

systems

Edinburgh

genetic algorithms

York

aurora

circuits

ngspice
hspice
RandomSpice

Southampton

Glasgow

physical devices

geronimo

eScience Components

- The main components of the eScience infrastructure are:
  - wrapper for the automated submission of jobs to remote clusters,
  - data management system to store and retrieve metadata about the jobs run,
  - AFS distributed file system, for sharing results.
  - security middleware.

eScience Components

- The main components of the eScience infrastructure are:
  - wrapper for the automated submission of jobs to remote clusters,
  - data management system to store and retrieve metadata about the jobs run,
  - AFS distributed file system, for sharing results.
  - security middleware.
- Briefly discuss the AFS and the job submission wrapper individually…

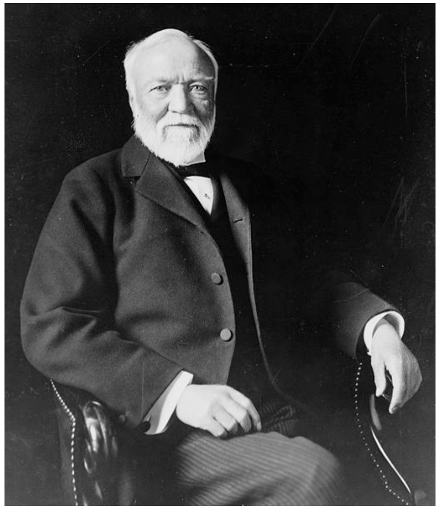
eScience Components

- The main components of the eScience infrastructure are:
  - wrapper for the automated submission of jobs to remote clusters,
  - data management system to store and retrieve metadata about the jobs run,
  - AFS distributed file system, for sharing results.
  - security middleware.
- Briefly discuss the AFS and the job submission wrapper individually…

Keeps track of the jobs run by automatically storing details of the jobs run, the location (in AFS) of the files of results and optionally of details extracted from the results files. All these details can subsequently be queried and retrieved.

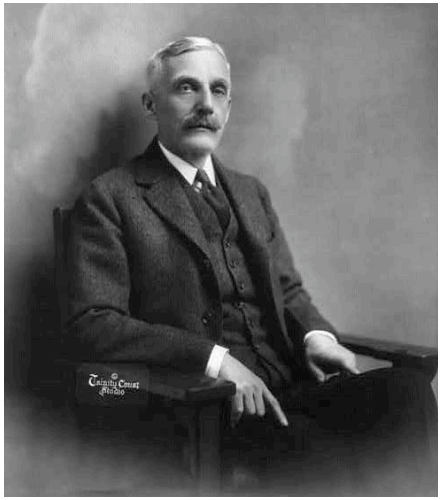
AFS: Andrew File System

- The Andrew File System was originally developed in 1983 at Carnegie-Mellon University:
  - ✦ There are several implementations; NanoCMOS is using the OpenAFS open-source version.
- AFS is a distributed file system:
  - ✦ the clusters write files of results into the AFS directories,
  - ✦ users can access these files from their local home machines.
- Client-server model:
  - ✦ cluster of machines; mixture of clients and servers,
  - ✦ servers store files,
  - ✦ clients allow users to access these files.
- AFS and local file space:
  - ✦ AFS files can coexist with a standard Unix file system,
  - ✦ AFS files exist in their own file hierarchy,
  - ✦ by convention this hierarchy starts in directory /afs



Andrew Carnegie

Andrew Mellon



eScience Components

- The main components of the eScience infrastructure are:
    - wrapper for the automated submission of jobs to remote clusters,
    - data management system to store and retrieve metadata about the jobs run,
    - AFS distributed file system, for sharing results.
    - security middleware.

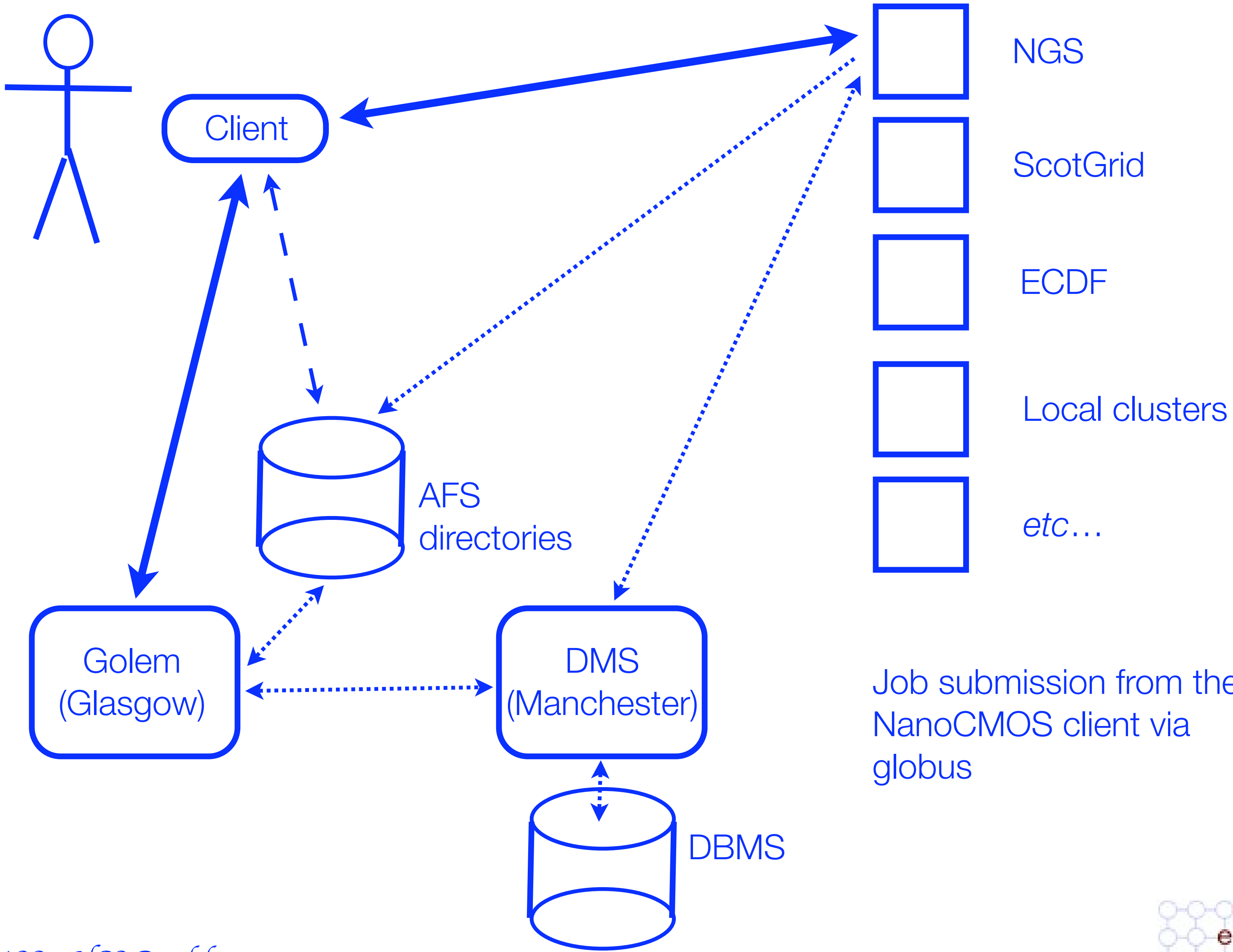
Job Submission Wrapper

- NanoCMOS has developed a job submission wrapper,
  - ✦ which simplifies submitting jobs to remote clusters.
- The wrapper submits a job to run an application (geronimo, *spice, [aurora]) on a designated compute resource:
  - ✦ NGS, ScotGrid, ECDF, various local clusters *etc*…
- It simplifies and unifies job submission:
  - ✦ easier to use than native job submission systems, and tailored to running NanoCMOS jobs,
  - ✦ the same procedure is used to submit to all remote clusters.
- The user must install a client on his local machine,
  - ✦ but this is simpler than trying to install a native job submission system, eg. globus.

Job Submission Wrapper (2)

- The wrapper handles authentication and authorisation on behalf of the user.
- It stages in input files and stages out output files to the AFS directories.
- Details of the job are added to the data management system for subsequent retrieval.
- Submission is a two-stage process:
  - preparation; setting up the job *etc*,
  - submitting the job.
- Job submission is usually by Globus and JSDL,
  - For some resources submission by SGI (Sun Grid Engine) is also available.

Job Submission Wrapper (3)



Job submission from the NanoCMOS client via globus

Submitting a Job with the Client

- Creating the job:

```
./randomspice -a create -i spice.xml
```

(returns a unique job identifier, `<job-id>`)

- Preparing a resource prior to submission:

```
./sam -a prep
```

- Submitting the job:

```
./sam -a submit -j <job-id>
```

- Checking the progress of the job:

```
./sam -a status -j <job-id>
```

Submitting a Job with the Client

- Creating the job:

```
./randomspice -a create -i spice.xml
```

(returns a unique job identifier, `<job-id>`)

- Preparing a resource prior to submission:

```
./sam -a prep
```

- Submitting the job:

```
./sam -a submit -j <job-id>
```

- Checking the progress of the job:

```
./sam -a status -j <job-id>
```

Setting up a Cluster to Run NanoCMOS Jobs

- Some configuration is required.
- Obviously the user must have registered to use the cluster, obtained a username *etc*.
- The NanoCMOS executables need to be available;
    - ✦ it may be possible to use them from the NanoCMOS AFS directories.
- An AFS client must be installed on the cluster.
- Firewall rules may need changing to allow traffic on some ports used by AFS.
- So, some assistance from the cluster's sys admins is required…
- … the NGS staff were very helpful, particularly Jason Lander at Leeds.

NanoCMOS:
Simulating Nanometre-Scale Transistors and Circuits on the NGS

Clive Davenhall
National e-Science Centre
(acd@staffmail.ed.ac.uk)

4 February 2010

NGS Roadshow at the University of Hull