LCFG Workshop
Edinburgh University 20/12/06

9.30 - 10.00	Paul Anderson, System Configuration & LCFG
10.00 - 10.45	Kenny MacDonald, LCFG in EUCS
10.45 - 11.15	Break
11.15 - 12.00	Alastair Scobie, Stephen Quinney & Chris Cooke, The LCFG distribution & LCFG in Informatics
12.00 - 12.20	Sean McGeever - LCFG in EPCC
12.20 - 12.40	Michael Gordon - LCFG in SEE
12.40 - 13.00	Discussion/Questions

System Configuration and LCFG

a "bottom-up" approach

Paul Anderson <dcspaul@inf.ed.ac.uk>

- What is "Configuration" and how does LCFG fit in ?
- A simple "bottom-up" example.
- More advanced use - "proscriptive" configuration.
- Current and future LCFG development.

Levels of configuration

"Copy this disk image onto these machines"

⇨

"Put these files on these machines"

⇨

"Put this line in sendmail.cf on this machine"

⇨

"Configure machine X as a mail server"

⇨

"Configure machine X as a mail server for this cluster"
(and the clients will automatically be configured to match)

⇨

"Configure any suitable machine as a mail server for this cluster"
(and the clients will automatically be configured to match)

⇨

Configure enough mail servers to guarantee an SMTP response
time of X seconds

Levels of configuration

"Copy this disk image onto these machines"

⇨

"Put these files on these machines"

⇨

"Put this line in sendmail.cf on this machine"

⇨

"Configure machine X as a mail server"

⇨

LCFG

"Configure machine X as a mail server for this cluster"
(and the clients will automatically be configured to match)

⇨

"Configure any suitable machine as a mail server for this cluster"
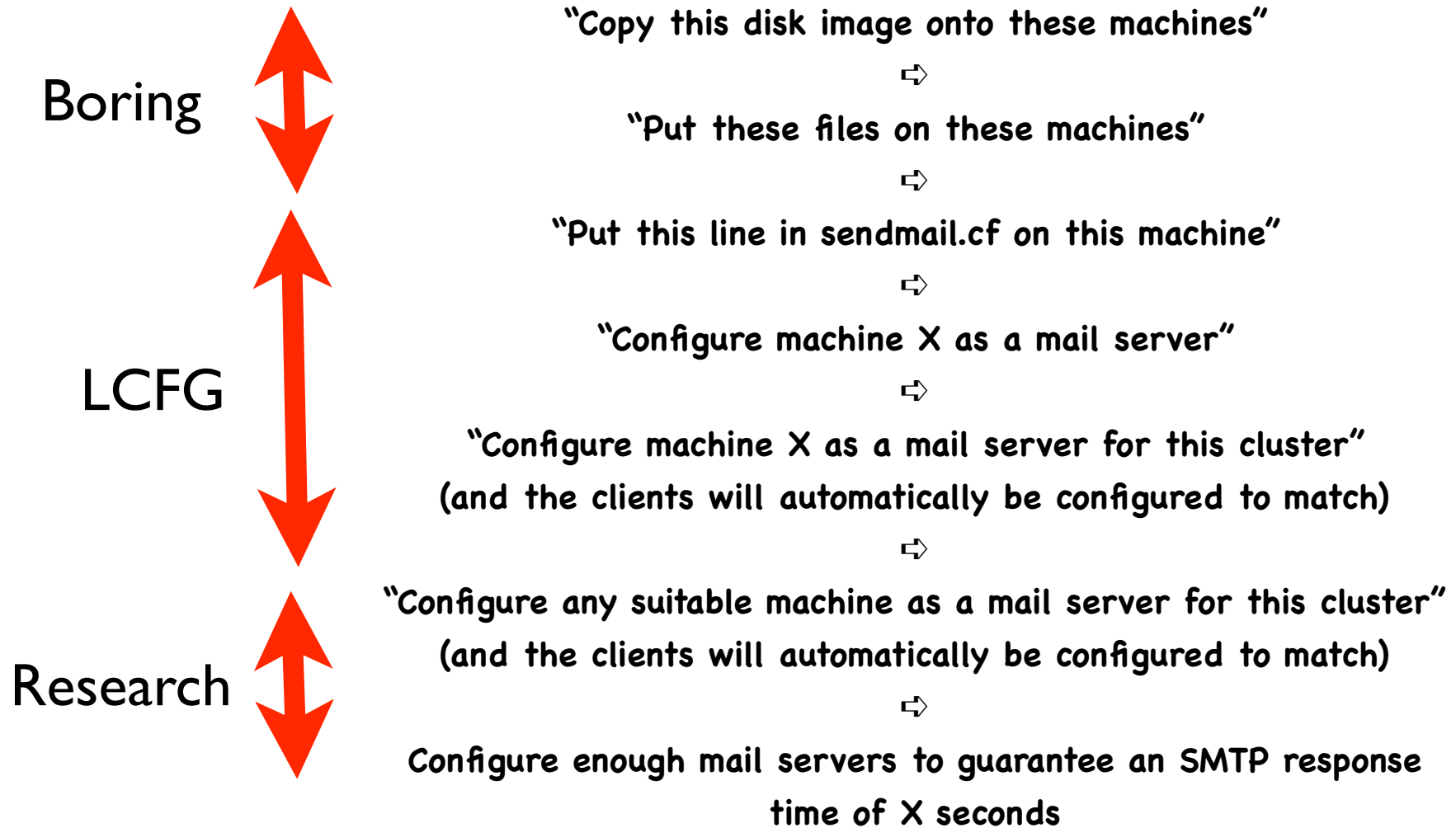(and the clients will automatically be configured to match)

⇨

Configure enough mail servers to guarantee an SMTP response
time of X seconds

Levels of configuration

Boring

LCFG

Research

"Copy this disk image onto these machines"
⇨
"Put these files on these machines"
⇨
"Put this line in sendmail.cf on this machine"
⇨
"Configure machine X as a mail server"
⇨
"Configure machine X as a mail server for this cluster"
(and the clients will automatically be configured to match)
⇨
"Configure any suitable machine as a mail server for this cluster"
(and the clients will automatically be configured to match)
⇨
Configure enough mail servers to guarantee an SMTP response time of X seconds

LCFG

- Started 12 years ago (LISA 8 paper) in the Computer Science department.

- Production tool (1000+ machines) and research testbed.

- Capable of very comprehensive configuration but (traditionally) steep entry curve.

- See www.lcfg.org

- The SAGE configuration booklet is here: http://www.sage.org/pubs/14_sysconfig/

- **LCFG is usually presented from the "top down" -**

  - LCFG manages the whole installation (including servers and relationships).

  - LCFG is the *only* thing which changes the configuration of any machine (*proscriptive*).

  - Clients run ~50 "components" ...

- **LCFG is capable of this, and this approach has a lot of benefits, but ...**

  - This is very committing for people starting out.

  - It is difficult to learn the system starting in this way.

- **It is possible to use LCFG in a very "lightweight way" -**

  - One or two "components" run on each client to manage specific configuration files.

  - Other parts of the configuration are managed in some other way (manually, or with some other tool).

- **This can gradually be extended to encompass more of the configuration as and when required**

  - Unlike simpler tools, this is possible without completely changing the tool.

- **A simple "bottom-up" example -**
  Let's configure the "message of the day file" -

`/etc/motd`

```
Welcome to the Blah computing service
```

- **The LCFG "file" component creates files from templates -**

```
Welcome to the <%dept%> computing ...
```

- **The values to substitute in the template are given as "resources" -**
  `dept=Informatics`

- **The template is common across all machines and does not change very often**

  - This is distributed in some other way

- **The resources specify the things which are different between machines (or which change frequently)**

Template → File Component → Configuration file

Resources

`dept=Informatics`
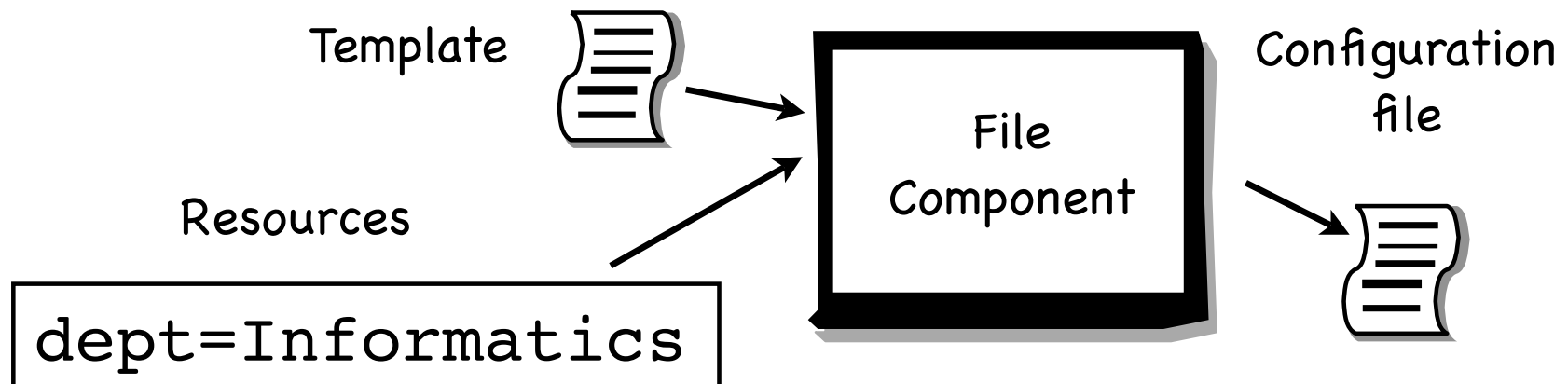
- **Where do the resources come from?**

  - The resources are stored in a simple (DBM) file on the client.

- **The LCFG "client" component ..**

  - Watches for change notifications from a server.

  - Fetches the resources from a web server (in a simple XML format called the "profile").

  - Tells the components whose resources have changed, so that they can "reconfigure".

  - The file component will regenerate the target file(s) when it reconfigures.

LCFG Architecture

Source
Specifications
(aspects)

"Components"

XML "profiles"

Compiler

Mail

DNS

...

Mail

DNS

...
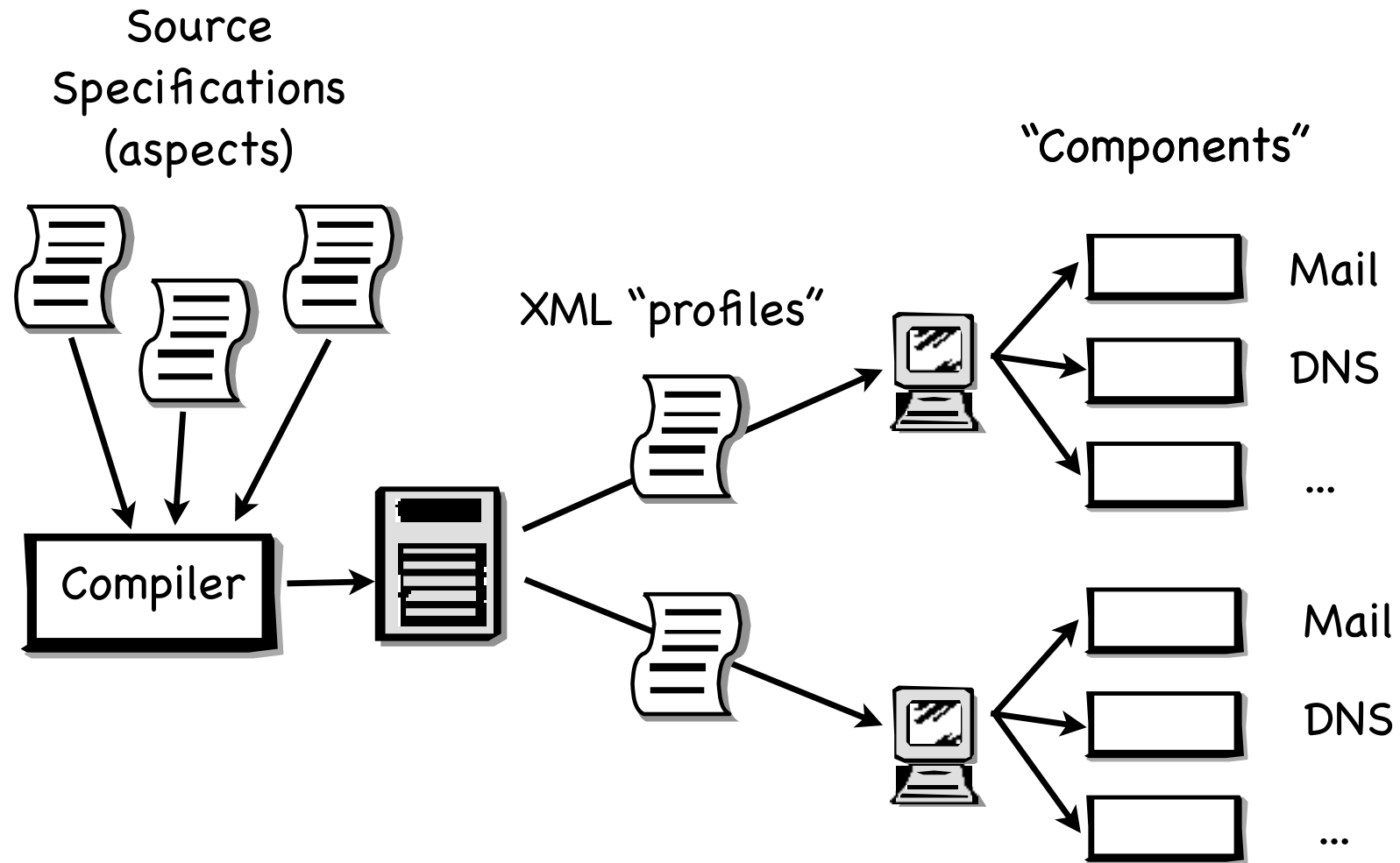
- **What does the LCFG server do?**

  - Make the XML profile by processing simple text files containing the resources.

  - Handles "classing" to specify resources for groups. ("include" files).

  - Handles (some) conflicts between resource specifications for different groups.

  - More complex things like relating resources between client and server machines (spanning maps).

- **We now have some useful properties ..**

  - Notice that all resources for all hosts come from the central LCFG (web) server.

  - This machine holds all the information about the configuration of all machines.

  - Whenever things are changed on the server, the clients pick up the change as soon as possible (usually immediately).

  - We can change the configuration of whole groups of machines by changing common files.

- **The whole configuration task is now reduced to managing the configuration data on the server.**

- **Summary**

  - Install (and start) the client component.

  - Install (and start) one or more other components (eg. file component) on the client.

  - Run a server.

  - When we change one file on the server, all the files will be reconfigured

- **Structuring and managing the configuration data on the server is the real problem!! -**

  - This defines what your machines look like!

- **Where next?**
  - Special components are needed to manage things which involve more than just creating a file - eg. stopping/restarting daemons.

  - One component usually manages the software packages on the machine.

  - One component handles component start/stop at boot/shutdown.

  - One component handles installing machines from "bare metal".

  - All of these components are "optional" and can be replaced or extended by your own ....

- **Current developments with LCFG ...**
  - The LCFG core is old, but very stable and well-tested. We do not expect this to change much.

  - Components will continue to evolve and increase.

  - We would like to improve portability (Solaris & Mac).

  - We are working on improving the web site (including software distribution) and the documentation.

- **We would like to promote collaboration to share some of these developments.**