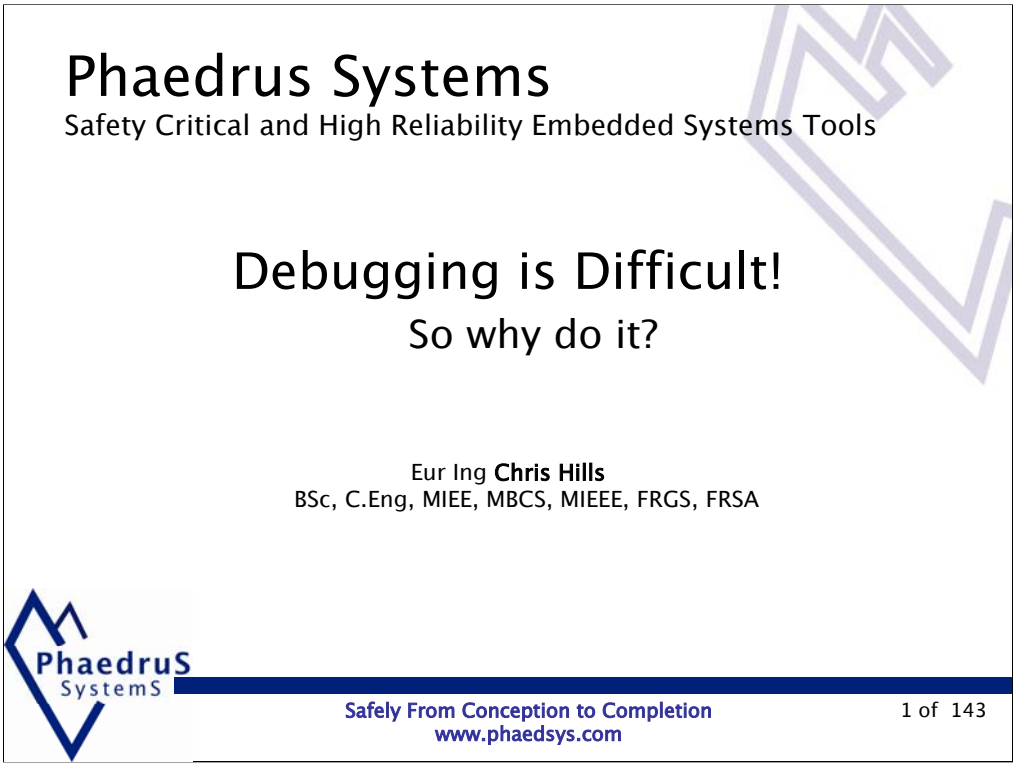
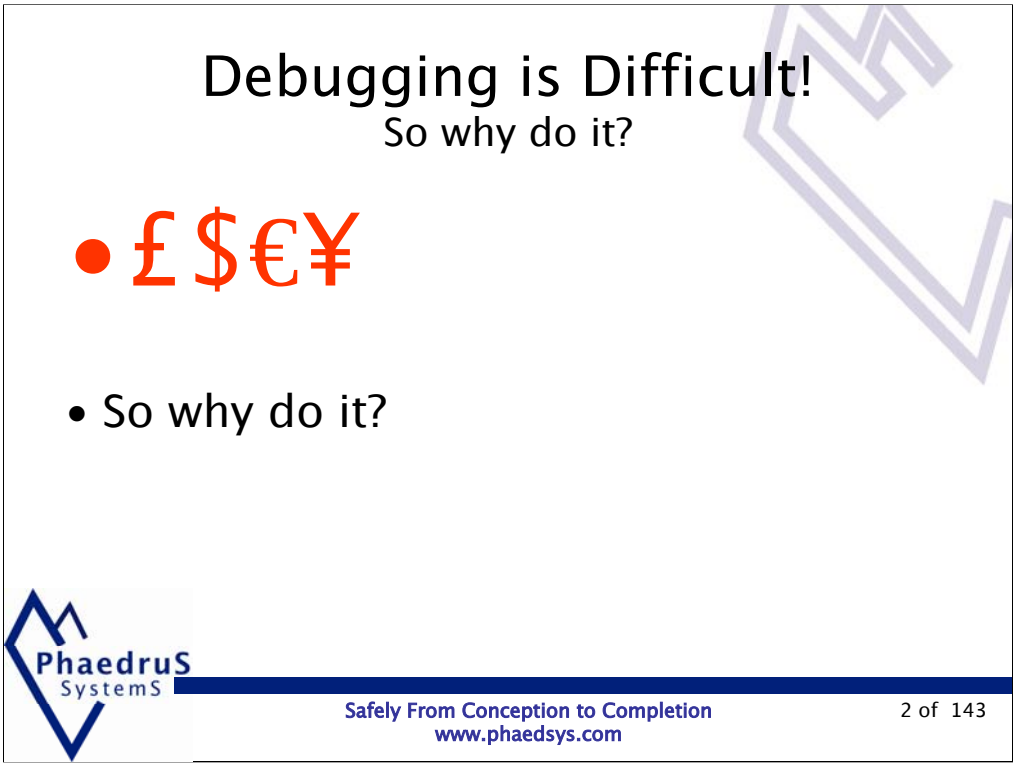



Debugging IS difficult so why to it?




Debugging

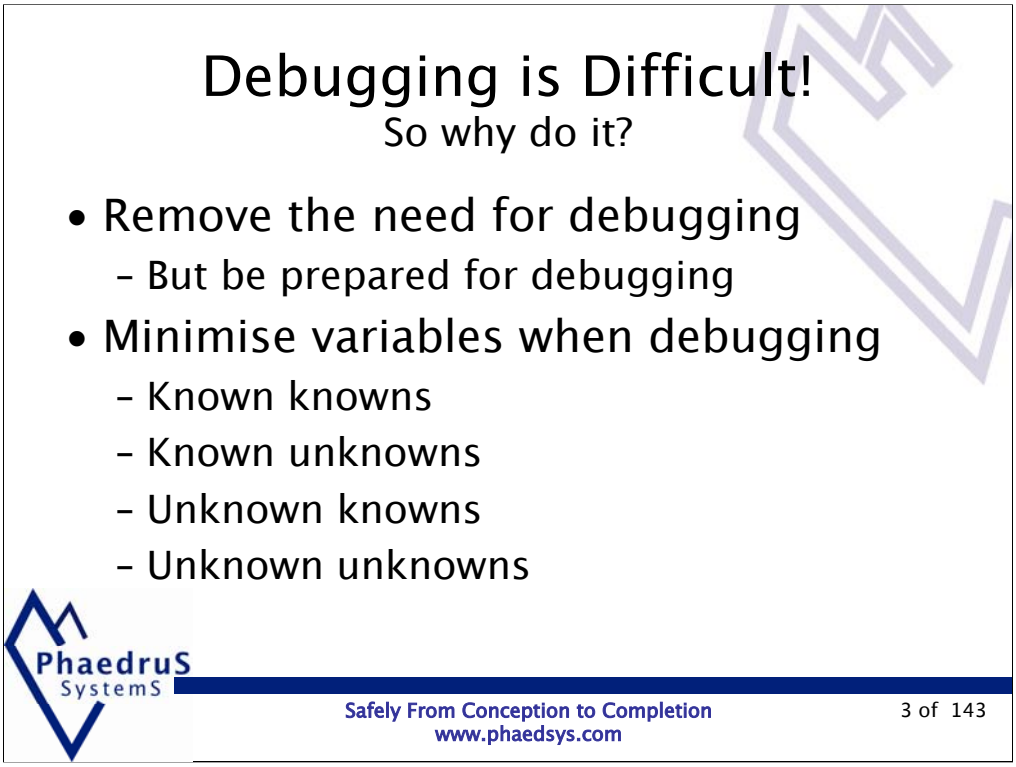
Takes time

Needs tools

Uses resources

COSTS  if you over run deadlines or deliver a buggy product to market it could cost more than money.

Jobs, company reputation, lives



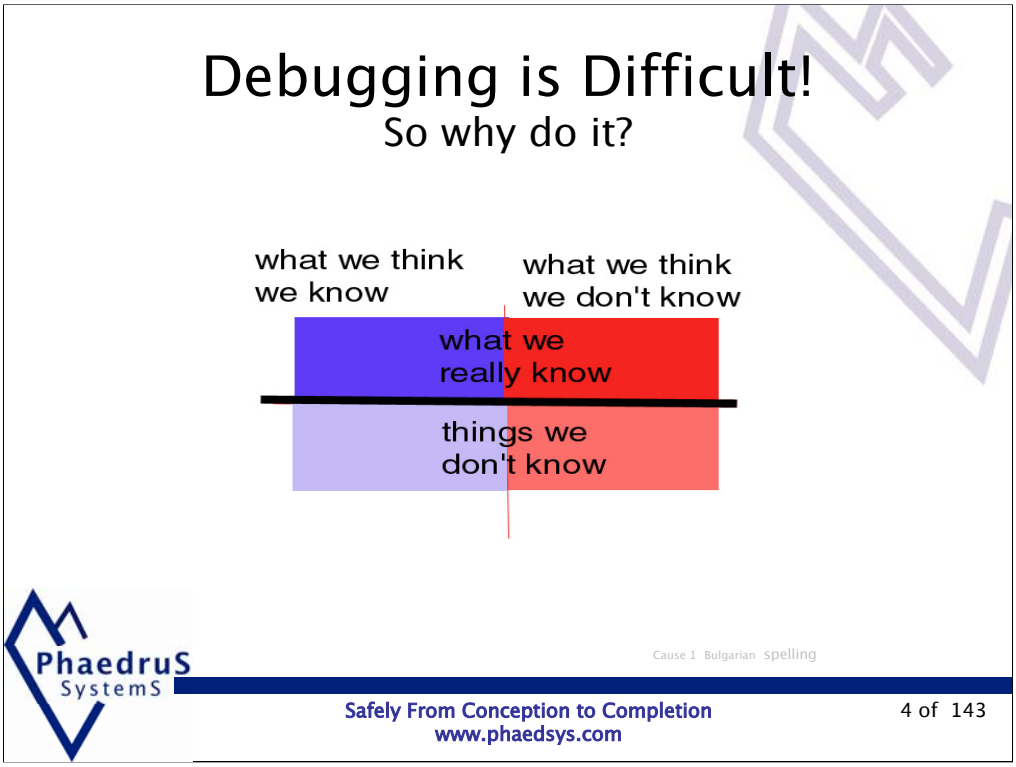

Remove the need for debugging but be better prepared for it.

Rumsfeld……

What do you know, what do you know you are missing

What  don't you realise you do have the answer for?

and vitally which piece of the puzzle don't you know you are missing?



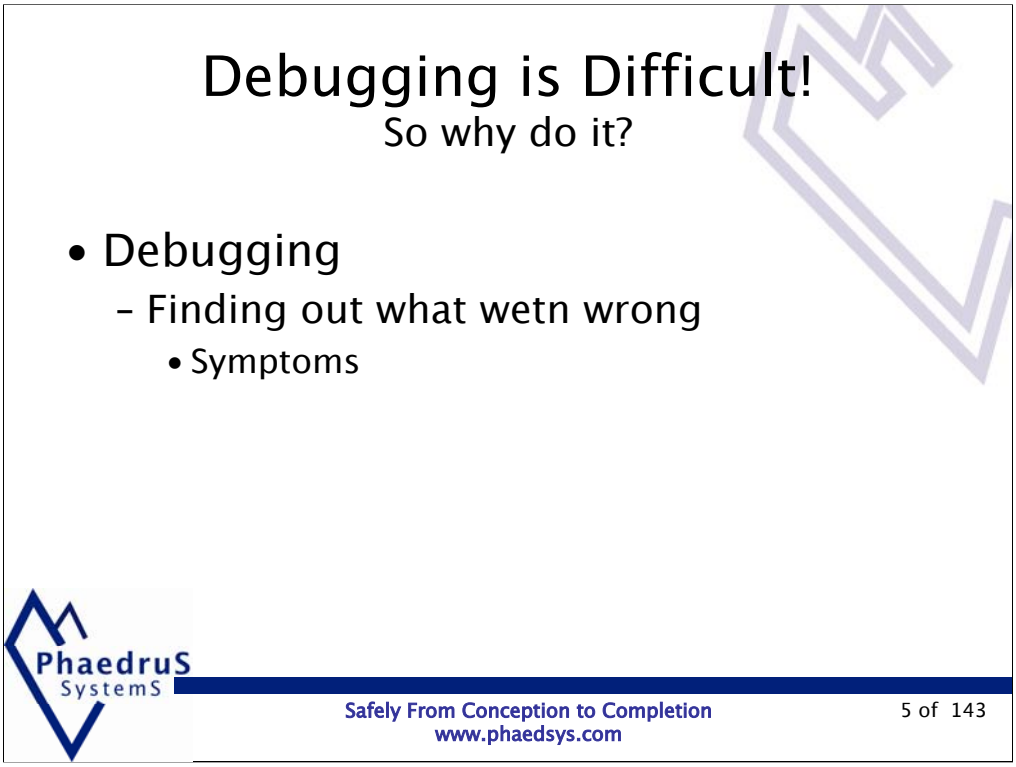

What we think we know and what we really know are often two different things.

Some things are in plain sight but not understood or recognised

What do you know, what do you know you are missing

What  don't you realise you do have the answer for?

and vitally which piece of the puzzle don't you know you are missing?



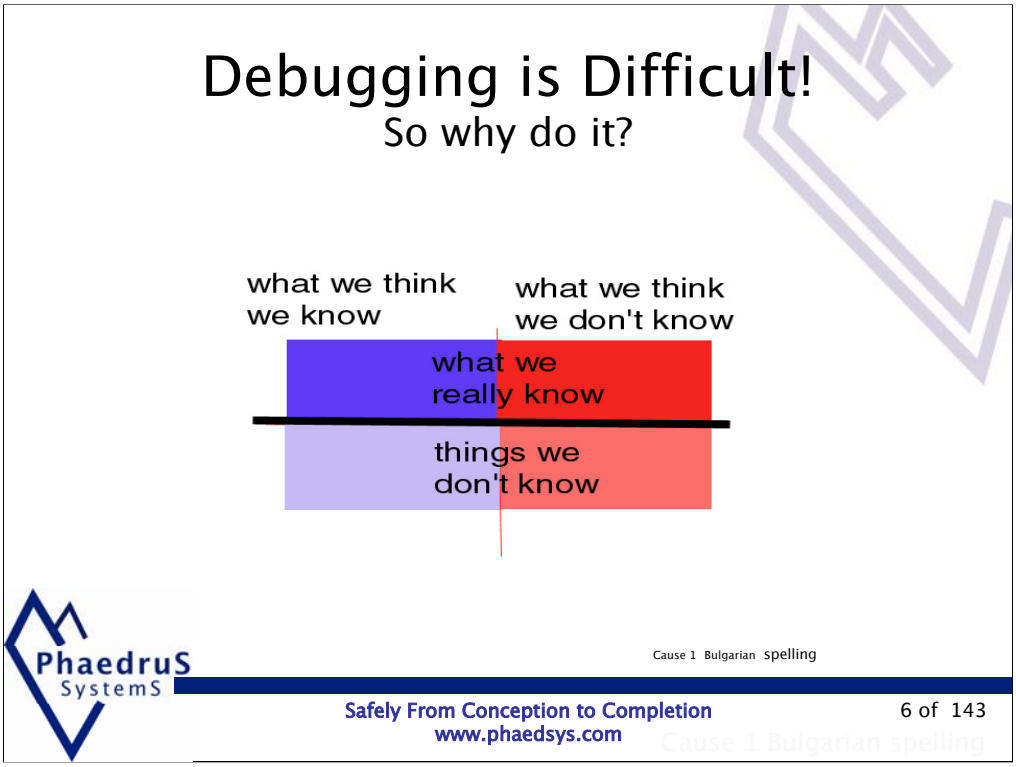

Debugging is working out why the system  is not doing what it should…….

Why is wetn there with no underline?

Bulgarian spell checking is on for that word…

You see Symptoms NOT  causes

EOFf between unix and ms




What we think we know and what we really know are often two different things.

Some things are in plain sight but not understood or recognised

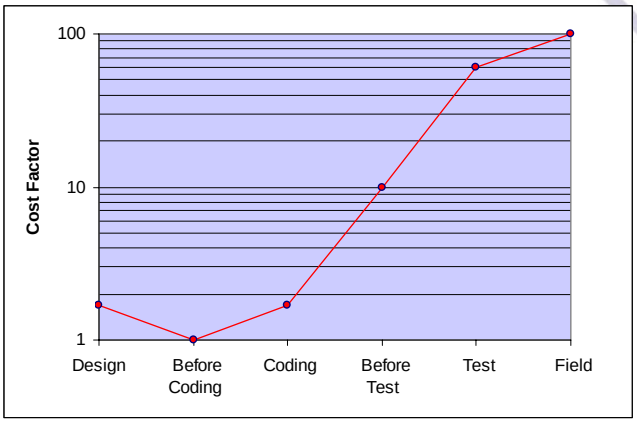
What do you know, what do you know you are missing

What  don't you realise you do have the answer for?

and vitally which piece of the puzzle don't you know you are missing?

# The Cost of Debugging

- Bugs cost more the longer they live




PhaedruS SystemS

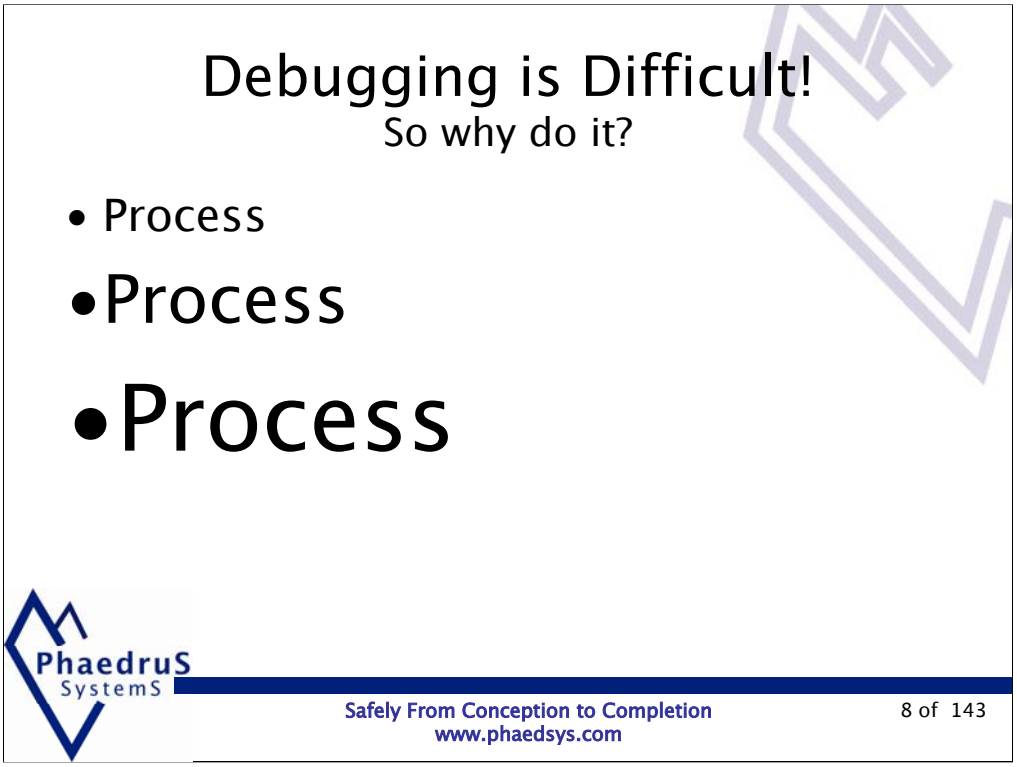
Safely From Conception to Completion
www.phaedsys.com

7 of 143

Source IBM and Programming Research

The sooner after birth you can  remove the bugs the less they cost.

Also the earlier in the project you can remove bugs the lest they cost



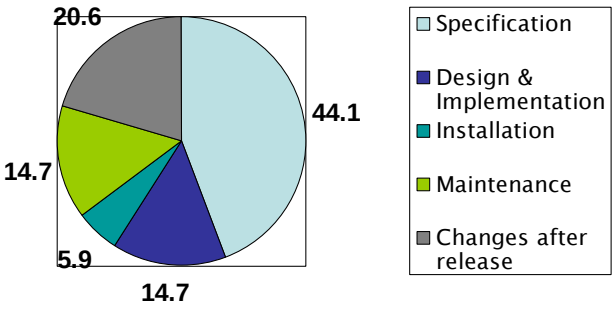

Process things should run on rails

Military process you can think for yourself….. Just don't need to think about the non essential things.

Hunter  crash 04:30  all in tea bar

# Debugging is Difficult!

## So why do it?




PhaedruS SystemS

Safely From Conception to Completion
www.phaedsys.com

9 of 143

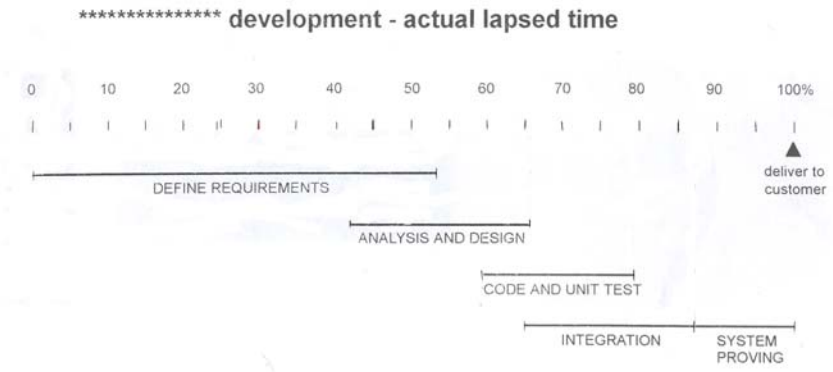
IEE Feb 2000 Over view of 61508 Ron Bell and P A Bennet

Primary causes of failure in safety critical projects

In a GOOD process the main problem is specification. This is because the problems in al the other areas are much lower than “normal.”

# Debugging is Difficult!

## So why do it?



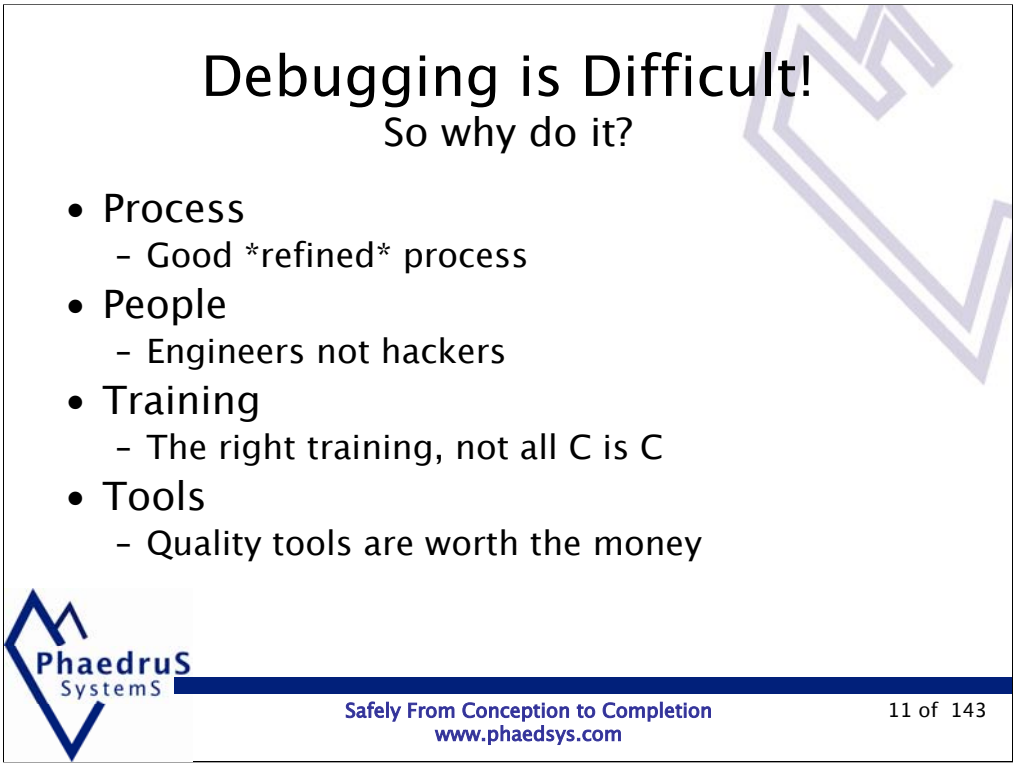

PhaedruS SystemS



Figures from several successful projects in a business division

What many don't realise is the coding is normally, or should normally be, only be a small part of the project.

Programmers need to realise this,




Process needs refining. FEED back try out process before casting in stone. Do CMM to level 3 before ISO9K  CMM-3 has process improvement built in.

Management have one view and the Engineers another. These need to be reconciled honestly. A system built on false information is worse than no system. Dictating a system that has not been properly researched is also fatal.  People should know their own domain managers have a different view than coders or HW designers….. And if the whole lot does not make a profit everyone looses.

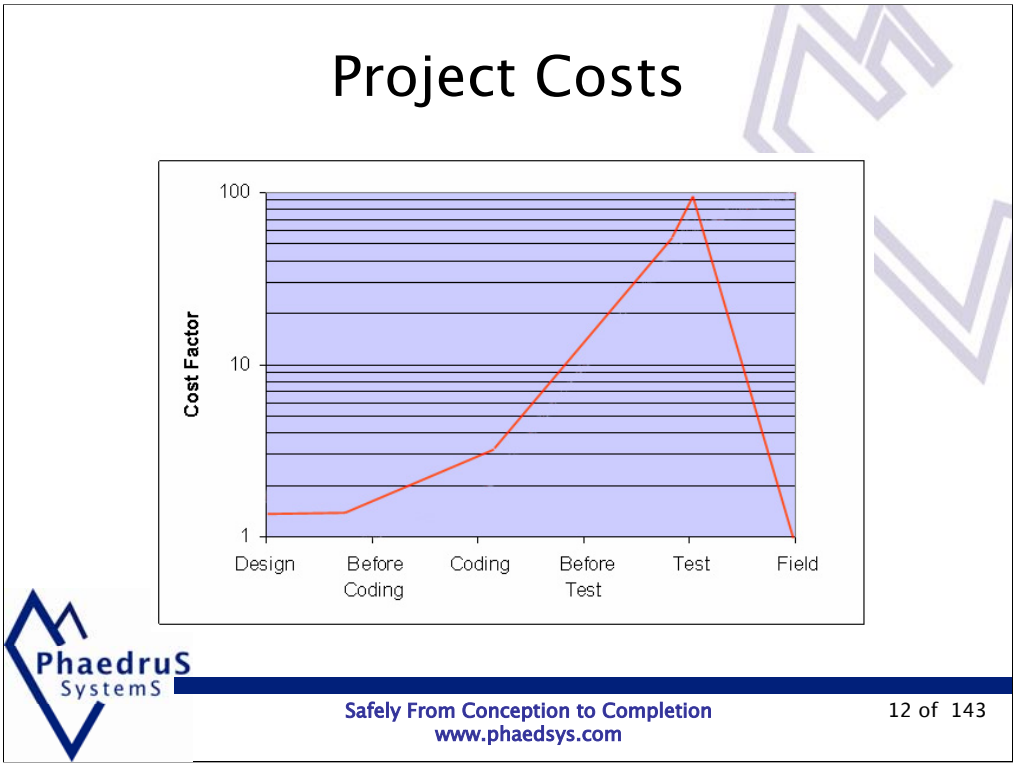
NO SILVER BULLET (see Brooks)  no religion (FOSS)

People Engineers not hacker. Right philosophy

Relevant training… most C and C++ training is for MS Windows VC++ not embedded or real time.

Tools good tools pay for themselves cheap tools usually require a lot of time and hassle for poorer results

Save money on multiple projects or have hassle on multiple projects

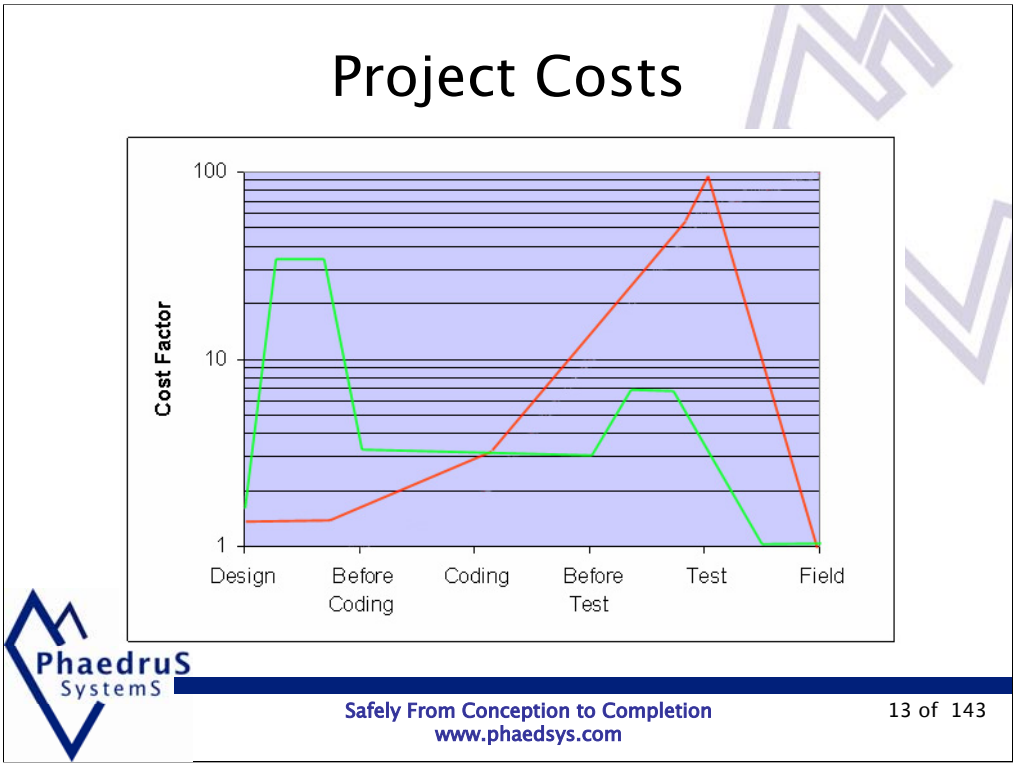



Low cost minimum risk start.

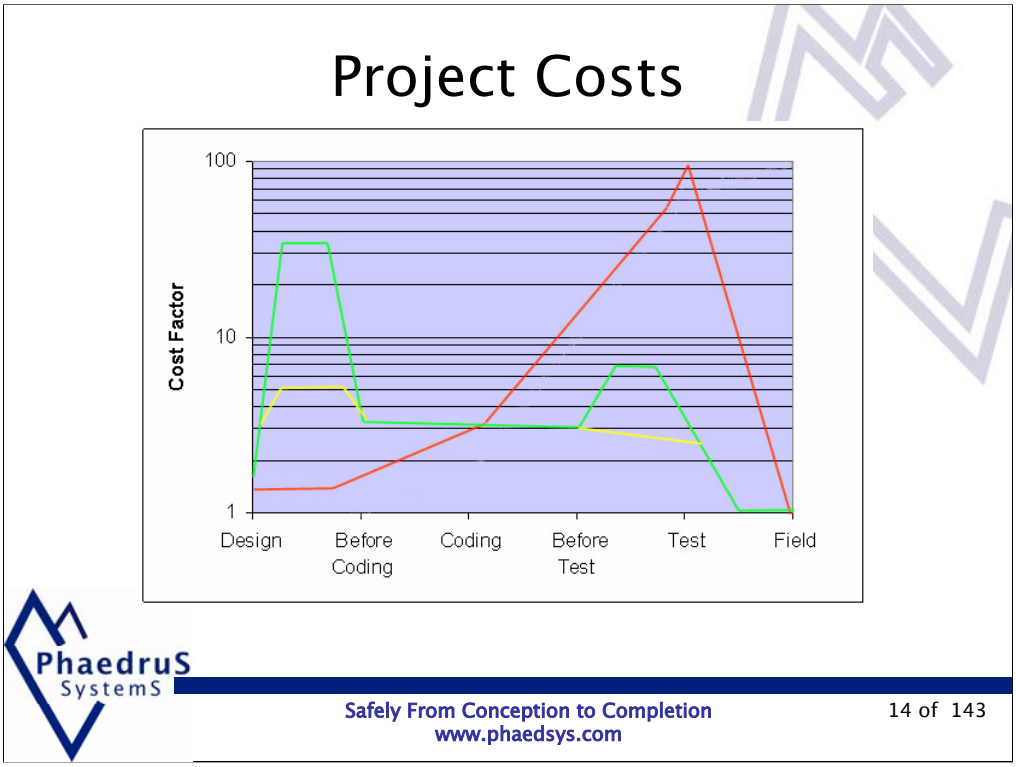
Buy  in tools late as possible

More people for coding as project slips

Lots more time and costs on debugging




Right process and tools higher cost start but lower cost for rest of project.
More likely hood of getting out on time and on budget
Management have to have faith that things will complete on time

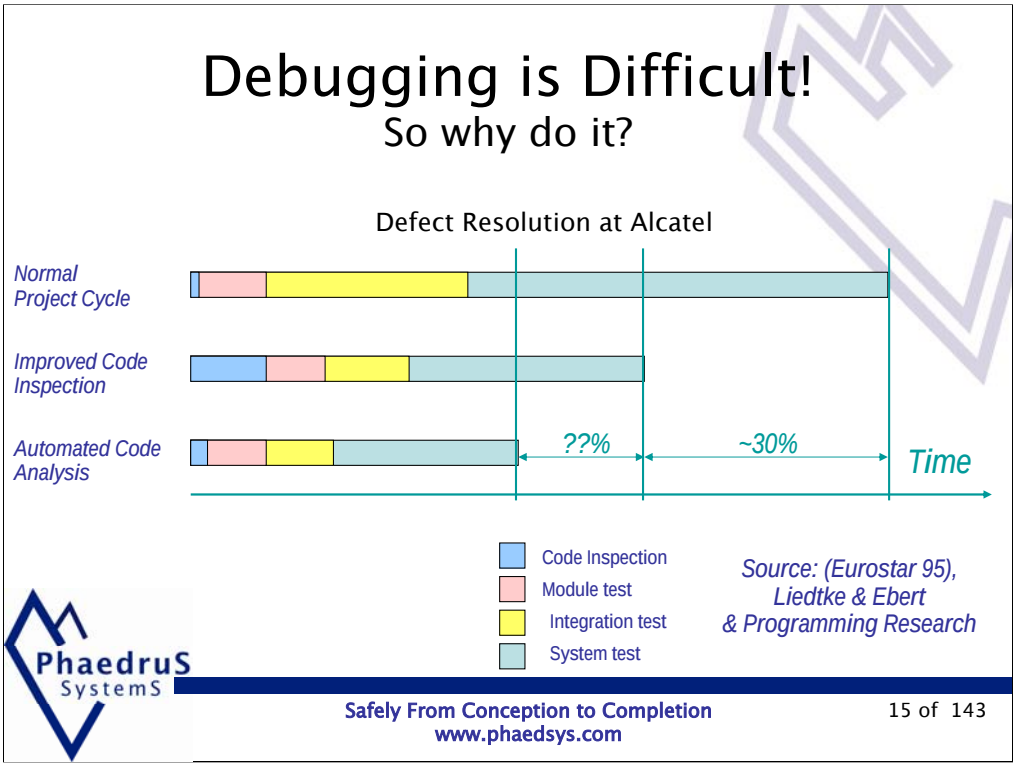



With the right tools already in place the initial costs are greatly reduced and as people are already used to the process less learning and more systems are in place and familiar

More to the point release on time! Running late on a release can cost more than the cost of the project.

More certain of hitting deadlines and finishing project on time

Is this just conjecture? No. many studies



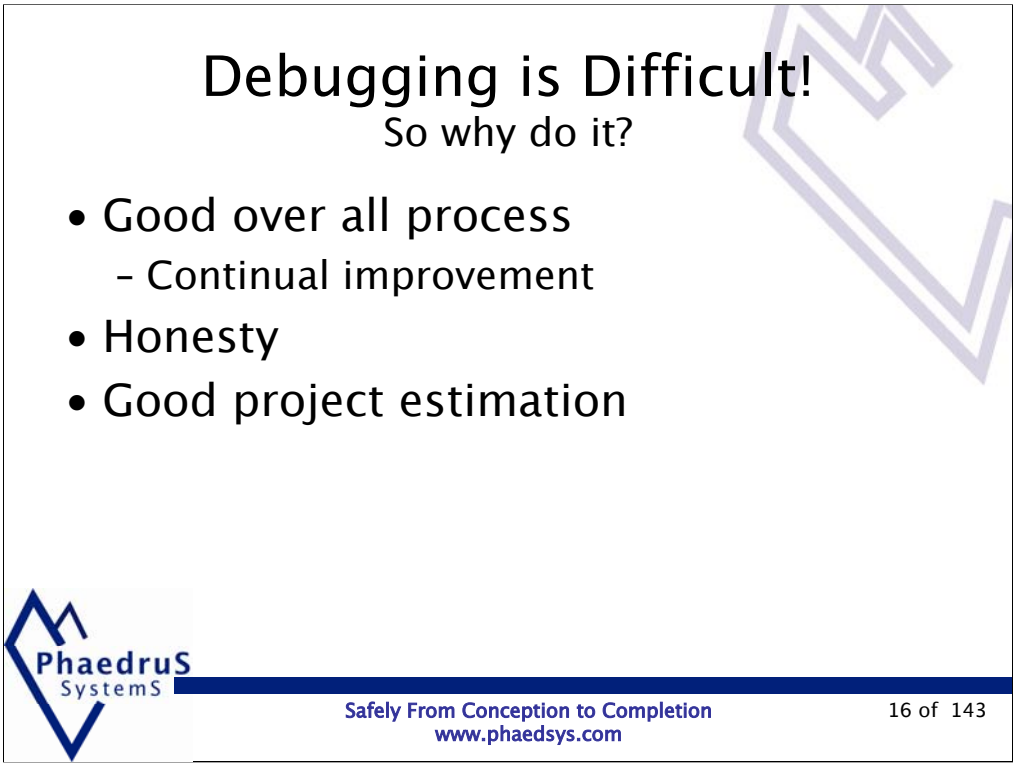

By improving and automating code inspections and static analysis vast amounts of time/resources and money can be saved

Cost of the tool to give these saving 6K

Similar savings in all areas. Starting with good specs and test specs

Some tests such as static analysis don't require specifications

Automate unit and dynamic test on target with emulators




Management have one view and the Engineers another. These need to be reconciled honestly. A system built on false information is worse than no system. Dictating a system that has not been properly researched is also fatal.

Management have one view and the Engineers another. These need to be reconciled honestly. A system built on false information is worse than no system. Dictating a system that has not been properly researched is also fatal. People should know their own domain managers have a different view than coders or HW designers…..

NO SILVER BULLET (see Brooks) no religion (FOSS)

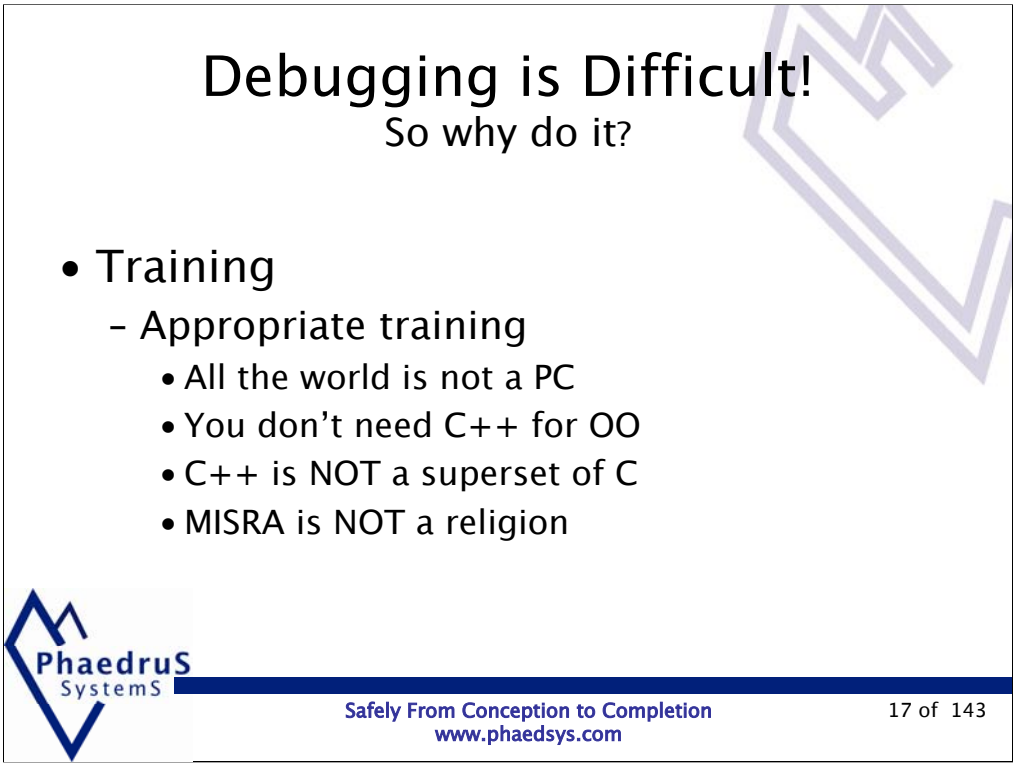
Not being honest saves up more trouble for later. Often the problem is more serious if not addresses running late when you discover everyone knew it would not work but did not say

Project estimation,. Why traditionally -10% + 100% and failure. ?

Lack of honesty in estimation by all sides

Bonus sales /Engineering net result annual 25% turnover of engineering staff.

Good estimation can save a project. Critical path, resources needed. +/- 5%




Many new "programmers" no real idea of embedded, real time etc. Software Engineering not taught but programming is.

Often with the wrong emphasis due to universities being strapped for cash….

Fashion languages

OO == modular

C++ a LOT more complex than C as are the tools

# Debugging is Difficult!

## So why do it?

- Correct Tools
  - Cheap tools often cost less too
- Spend to save
  - Engineer cost 60 GBP per hour
  - Emulator cost 8K GBP
  - In profit if it saves 4 man weeks




Cheap tools cost less too…. ☺

Cut corners,

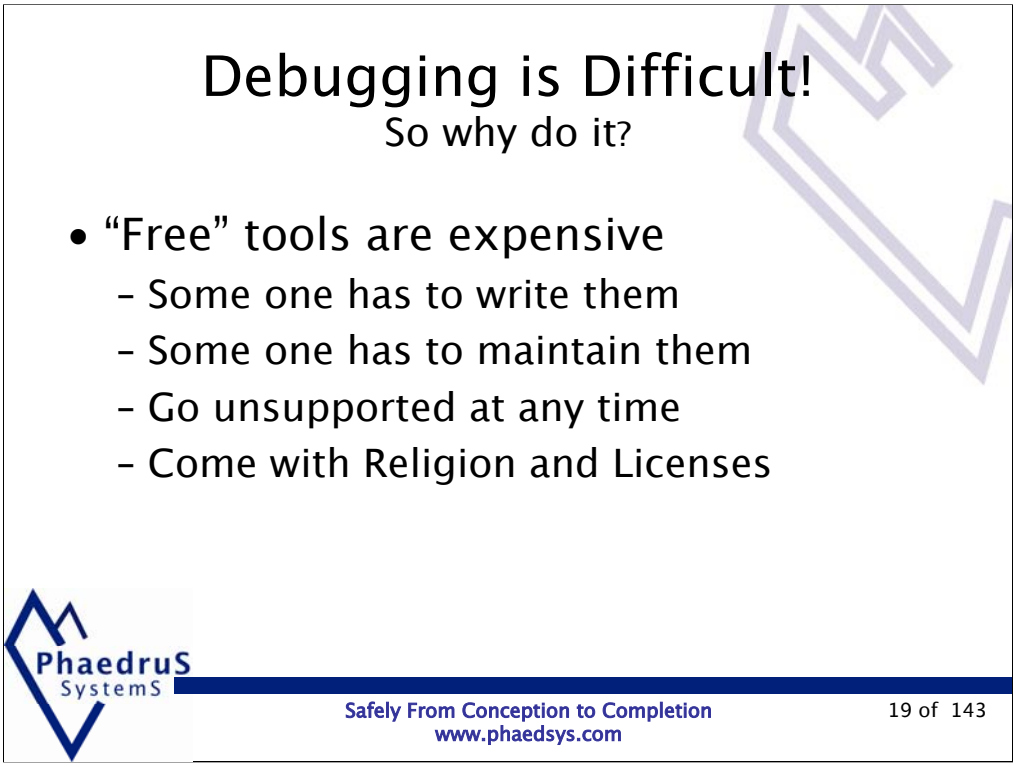
JTAGS for example

Parallel 50 GBP slow and unreliable

USB 150 GBP fast & reliable

8K ICE fast reliable trace, modular and usable on more project and only get more efficient.

Automatged testing



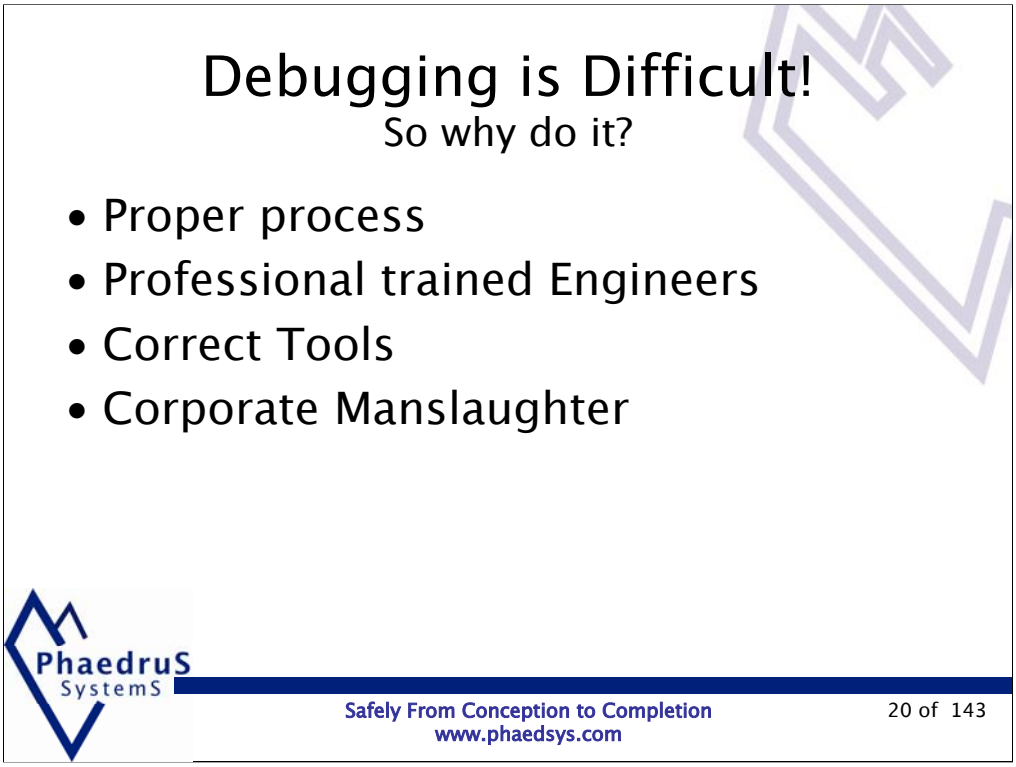

Free tools other than some small utilities essentially one man projects

Free SW not usually supported long term (writers usually get jobs, girl friends, children, diy etc) many projects die out

Linux for example last year's Killer linux on the way out web page listing all the very many versions of Linux. Majority unsupported/obsolete

Licences and publishing code

RELIGION

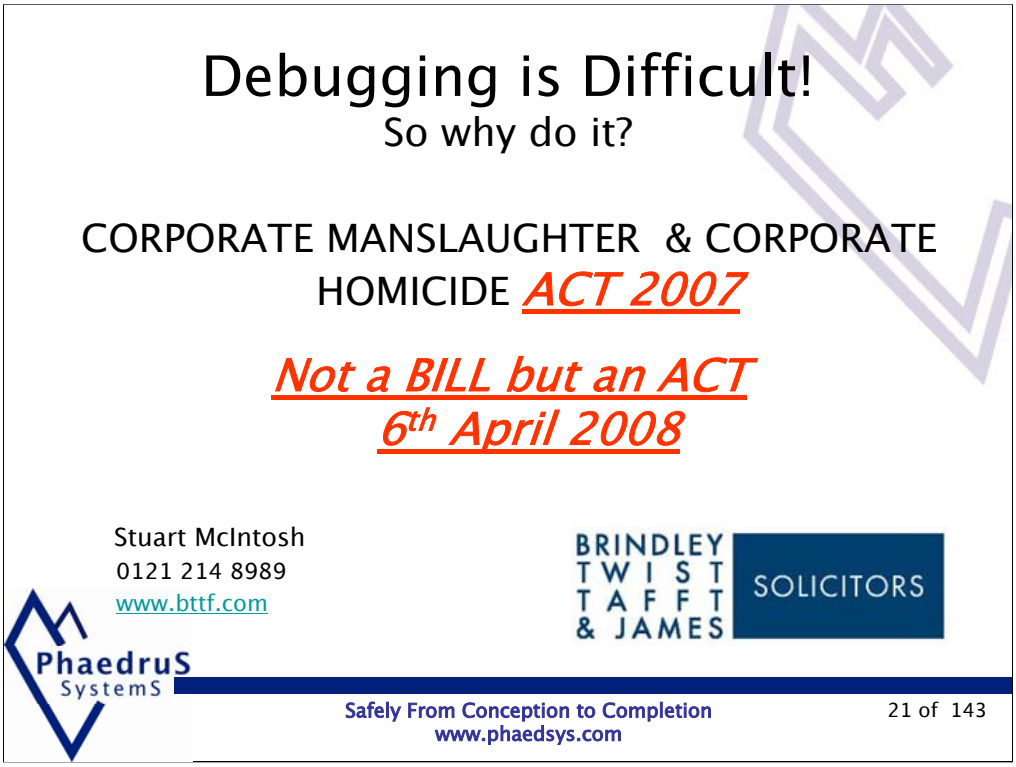



From 2003 I have gone on about Corporate Manslaughter "when the NEW bill comes in " because it has a duty of care rather than controlling mind

In an accident tomorrow if the device you made, car, plane, microwave, control system is involved in a fatal accident the rules will change

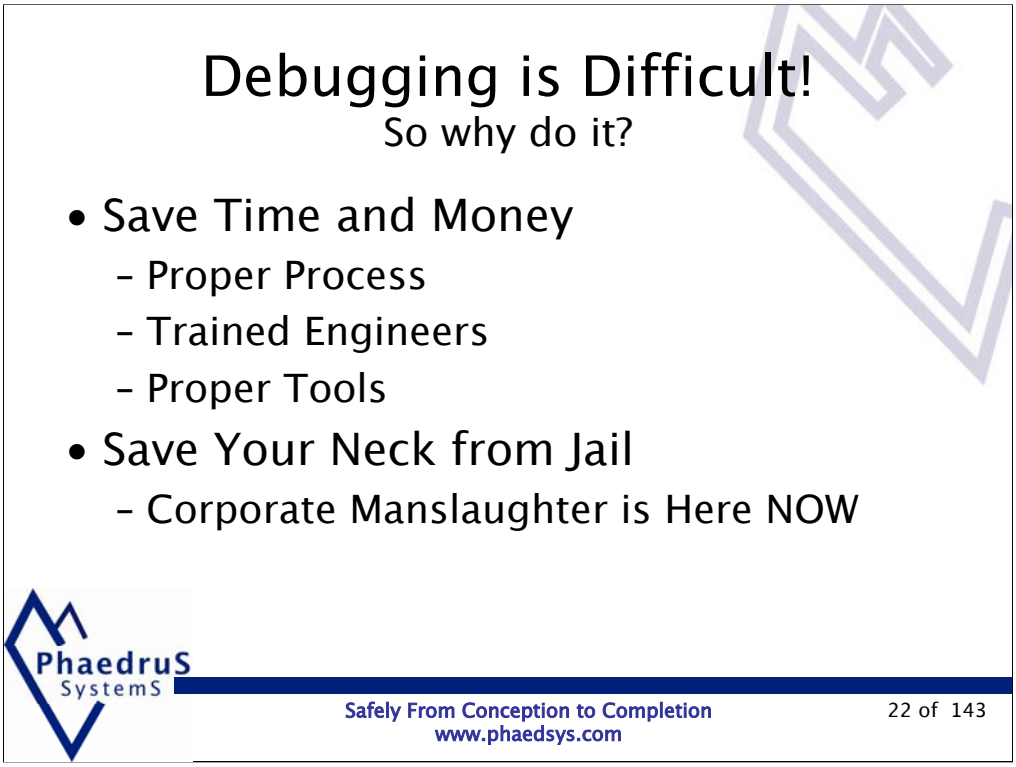
They can look at the process and methods used in it's development and hold the management responsible…

When the act comes in

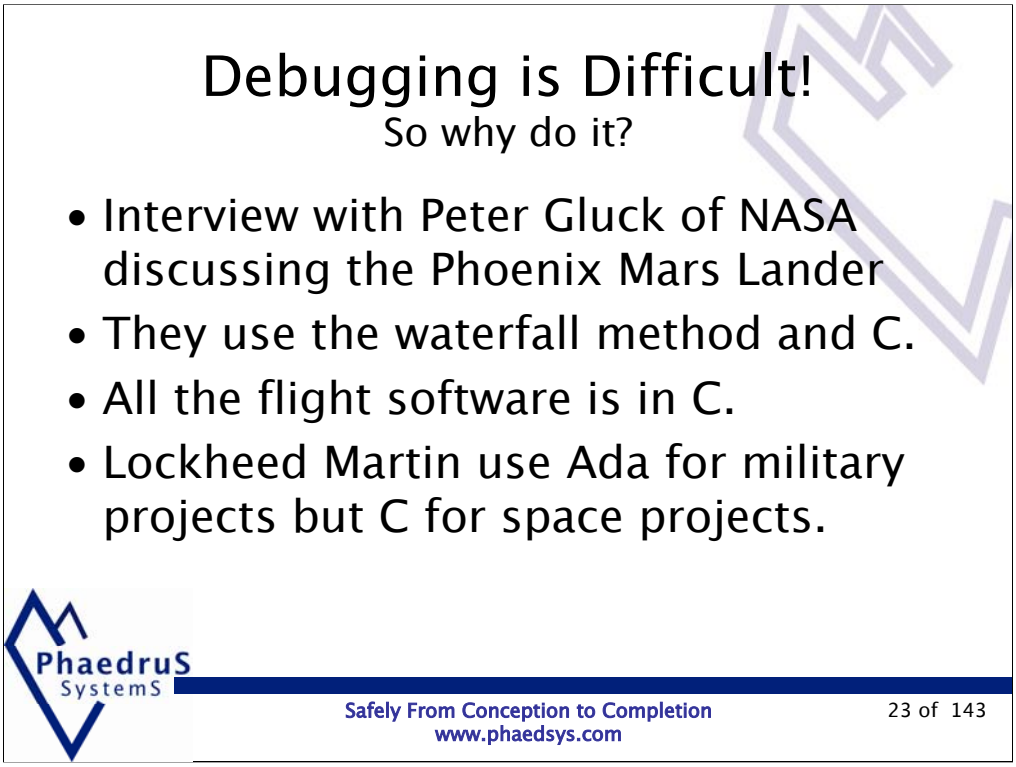



IANAL Came into force 6th April 2008. so I can't give any advice.

Solicitor at BCS meeting covered it last week. CRIMINAL LAW NOT CIVIL




Save time and money hit deadlines reduce bugs, increase testing and keep out of jail




http://news.oreilly.com/2008/07/the-software-behind-the-mars-p.html  The full text is there as well as the audio file. It's worth listening to.

US DoD requires Ada otherwise they would use C

V and Waterfall siimilar

# Debugging is Difficult!

So why do it?

# Any Questions?

# Debugging is Difficult!
## So why do it?

Eur Ing **Chris Hills**
BSc, C.Eng, MIEE, MBCS, MIEEE, FRGS, FRSA

# Phaedrus Systems

Safety Critical and High Reliability Embedded Systems Tools

info@phaedsys.com
www.phaedsys.com