WHY OUR FUTURE DEPENDS ON INTEROPERABILITY

(AND WHY EMBEDDED SOFTWARE DEVELOPERS ARE AWESOME)

Current Analysis Group

Gary Barnett
@thinkovation
Agenda

• Hello!
  • Why am I passionate about Embedded Systems

• Why is interoperability so important?
  • ... or why the world will break if we don’t address the issue

• Some challenges
  • Why we have a lot of work to do

• Homework
  • Three tasks to take home
  • And one to complete while you are here!
Hello!
Two projects I have worked on

http://www.weatherfile.com/

http://www.airsenssa.org/
Air Quality Monitoring
Air Quality Monitoring
The future of the planet depends on interop...

**WHY IS INTEROPERABILITY SO IMPORTANT?**
In the good old days

CC BY-SA 3.0 Jonathunder wikipedia
In the good old days

Lines of code = 0
Programming tools were simple

Public Domain via Lynn at publicdomainpictures.net
Today

Lines of code = >20 million
The size of software is growing

- Modern Car: 100 million lines of code
- Facebook: 80 million lines of code
- Windows Vista: 60 million lines of code
- Large Hadron Collider: 40 million lines of code
- Boeing 787: 30 million lines of code
- Android: 20 million lines of code
- Google Chrome: 10 million lines of code
- Linux Kernel: 5 million lines of code
- Mars Curiosity Rover: 2 million lines of code
- Hubble Space Telescope: 1 million lines of code
- F-22 Fighter: 0.5 million lines of code
- Space Shuttle: 0.2 million lines of code
- Apollo 12: 0.1 million lines of code
To put this into perspective

Public Domain – via Nasa

By DJI technologies - http://www.dji.com/, CC BY-SA 3.0
But it is not just LOC...

Different systems

Different sources

That interact
But it is not just LOC...

Software accounts for a growing proportion of the cost of a car
But it is not just LOC...

Software and electronics account for a growing proportion of the cost of a car

>50% by 2030?
But it is not just LOC...

Firmware is already > 50% of the development cost for many IOT devices
By addressing interoperability Crystal will help to make safety critical software and systems that are:

- Higher quality
- Lower cost
- Delivered more quickly
Holy Grail of Software Engineering

Increase speed of development
Holy Grail of Software Engineering

Increase quality of development

CC04-BY-SA – Wikipedia Andrzej Barabasz (Chepry) - Own work
Lower cost of development

CC0 – Free to use
We have a lot of work to do

SOME CHALLENGES
CHALLENGE #1

EXTEND COVERAGE TO OTHER DOMAINS
The four application domains represent a significant step forward

- Aerospace
- Automotive
- Health
- Rail
But in the future we will be surrounded by safety critical systems.

- As we travel
  - Road management
- At home
  - Appliances, smoke/gas detectors, central heating systems
- At work
  - HVAC, computers, robots....
CHALLENGE #2

WORK TO INTRODUCE GOOD PRACTICE TO NON-TRADITIONAL EMBEDDED S/W DEVELOPERS
Many of these systems will be developed by experts like you...
But many will be developed by muppets like me.....
My nightmare
Is this a safety critical system?

- It can heat two litres of water to boiling point
- It can be controlled remotely
- The firmware was developed by muppets
CHALLENGE #3

WORK TO EXPLAIN THE LONG TERM VALUE OF QUALITY TO OUR BOSSES AND CUSTOMERS
“Why does it take so long????”

CC0 – Free to use
“Why does it take so long????”

Public Domain via Project Gutenberg
FINAL CHALLENGE!
Make a new friend while you’re here

Public Domain via pixabay
THANK YOU!

Gary Barnett - @thinkovation

This work is licensed under a Creative Commons Attribution 4.0 International License.

For more information on Creative Commons Licensing please visit https://creativecommons.org/