HARDWARE INDEPENDENCE

YOU'LL NEVER TAKE OUR FREEDOM

JON VICKERS, PRODUCT MANAGER
STEMMER IMAGING
HARDWARE INDEPENDENCE

POSSIBILITIES, CONSIDERATIONS AND IMPLICATIONS

JON VICKERS, PRODUCT MANAGER, STEMMER IMAGING
VISION.
CUTTING-EDGE MACHINE VISION TECHNOLOGY
RIGHT.

COMBINING PASSION AND EXCELLENT SERVICE
1. What?
2. Why?
3. How?
4. What to consider?
BEFORE WE START...

THERE IS A COMMON THEME:
MAX VALUE, MIN RISK
THE VALUE THAT YOU ADD
WHERE IS THE VALUE-ADDED IN YOUR APPLICATIONS?

*It is in your software, your development time, your expertise*
WHAT DO WE MEAN BY ‘HARDWARE INDEPENDENCE?’

We’ll assume we are talking about VISION SYSTEMS

1) Platform-independence
   • The processing platform
   • X86 vs ARM
   • Andreas Rittinger is giving a talk about this:
     • Performance Embedded-Prozessoren
     • So we will not talk about performance in any detail

2) Camera-independence
   • Really the acquisition part of a vision system
   • Cameras, framegrabbers and software
WHAT DO WE MEAN BY ‘HARDWARE INDEPENDENCE?’

We’ll assume we are talking about VISION SYSTEMS

1) Platform-independence
   • The processing platform
   • X86 vs ARM
   • CISC vs RISC
   • Andreas Rittinger is giving a talk about this:
     • Performance Embedded-Prozessoren
     • So we will not talk about performance in any detail

2) Camera-independence
   • Really the acquisition part of a vision system
   • Cameras, framegrabbers and software
WHAT DO WE MEAN BY ‘HARDWARE INDEPENDENCE?’

We’ll assume we are talking about VISION SYSTEMS

1) Platform-independence
   - The processing platform
   - X86 vs ARM
   - CISC vs RISC
   - Andreas Rittinger is giving a talk about this:
     - Performance Embedded-Prozessoren
     - So we will not talk about performance in any detail

2) Camera-independence
   - Really the acquisition part of a vision system
   - Cameras, framegrabbers and software
   - Areascan, linescan, 3D, …
PLATFORM INDEPENDENCE

A STEMMER IMAGING PRODUCT
PCs, embedded and somehow OS gets in there too

- PCs are the standard choice
  - Flexibility (slots, upgrades)
  - Power (CISC, Intel vs AMD)
  - Well-known, well-supported
  - Choice of OS

- “Embedded” (we’ll assume ARM-based SoC)
  - Choices of retail SoC and carriers
  - Self-designed carriers and SoCs.
  - Size/shape advantage.
  - Relatively less power (GHz and RISC)
  - Less power (Watts) too
  - Less well-known, Linux only.
DEVELOPMENT

Generally developing and deploying an embedded solution will be more work than a traditional solution.

Development time and cost is higher.

Duplication cost is lower.

For some applications the equation favours Embedded, sometimes not.
WHY NOT DEVELOP ONCE FOR ALL PLATFORMS? CROSS-PLATFORM DEVELOPMENT

Write once, compile for different platforms

What do you need to consider?

• Software that supports x86 & ARM, Windows & Linux
• Knowledge of the differences between the platforms
• You can take this away by using
  • Visual Studio C++ (IFF there is no GUI)
  • QT for cross-platform GUI applications
  • MONO allows .NET applications under Linux
  • Python is inherently multi-platform
CAMERA (ACQUISITION) INDEPENDENCE
CAMERA INDEPENDENCE

*Why? We know what we want.*

- We can calculate the required resolution and speed
- We can work out if we need mono, colour, NIR, hyperspectral, areascan, linescan, 3D...
- We quickly get to a preferred interface and a small selection of cameras

So, what is the problem?
CAMERA INDEPENDENCE

Why? We know what we want.

• We can calculate the required resolution and speed
• We can work out if we need mono, colour, NIR, hyperspectral...
• From those we quickly get to a preferred interface
• And a small selection of cameras

So, what is the problem?

Find the right area scan camera

Use our clever product search to find the camera for your individual machine task.

Search

Spectral sensitivity

Resolution (MP)

Frame rate (Hz)

Interface

Lens mount

Search products

57 products were found.
WHAT IS THE RISK?

You’ve chosen a camera – just start developing…

Consider an application that can only use one camera model (the worst case)

• Camera Vendor brings out a v2, nothing works
  • You need to rewrite

• Camera Vendor makes the model obsolete or is bought out or goes out of business
  • You need to scope the market AND rewrite

• Camera Vendor 2 brings out a new, better, faster, smaller, cheaper model
  • You wish you didn’t have to rewrite!

“We need to support a new camera”
SO THERE IS A RISK – WHAT CAN BE DONE TO MITIGATE IT?

Standards are a good choice to start

Genicam is common:
- GigE Vision
- USB3Vision
- CoAxPress
- Some CameraLink too

But using a manufacturer’s Genicam-supporting SDK still doesn’t solve the problem of being tied to a manufacturer.

“I have a solution, but…”
INDEPENDENT STANDARDS SUPPORT

Supporting standards but not being tied to a manufacturer

- Genicam Support is taken as a given, but it is doubly-useful if you can support 3rd party GenTLs
  - CameraLink grabbers
  - LMI 3D scanners
  - Intel RealSense stereo cameras
- Beyond standards?
  - Easy changeover of different types of device – areascan to linescan to grabber to 3D
  - Easy can mean one line of code
    - Or one variable if you are clever about it…

“That is a good solution!”
COMMON VISION BLOX FITS ALL OF THESE REQUIREMENTS

CVB 2019, available now!

- Independent of hardware manufacturers
- Supports x86 Windows and Linux, ARM Linux
- “Hardware Abstraction Layer” means that CVB translates standard commands to different hardware, so you don’t need to.
- To support low-power platforms, minimise the acquisition overhead – CVB is famously fast and lightweight

- 1) CVB CameraSuite – free SDK with GenICam cameras from STEMMER IMAGING
- 2) CVB Image Manager, Foundation Package, high-end tools (including 3D and Machine Learning) has a good and increasing coverage on Linux on any platform. This brings processing to the hardware independence*

*Ask Jon for details
THANK YOU FOR YOUR TIME

Your contact person
Jon Vickers

STEMMER IMAGING AG
+44 1252 7800-84
info@stemmer-imaging.de
www.stemmer-imaging.de