

Promoting friendship and collaboration between projects

Peter Clifton

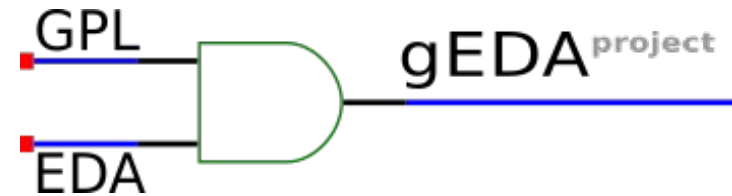
Peter.Clifton@clifton-electronics.co.uk

Diplomacy for developers and communities.

Finding common ground and ways to collaborate - whilst retaining project identities.

Some of the FOSS EDA projects

KiCad



PCB
BCB

gnucap



KTechlab

GHDL



Icarus Verilog



GTKwave

gwave



Oregano

gaw

Motivations

Questions to address for each project...

- What problem is our project trying to solve?
- Who is our target audience?
- How are we solving this problem?
- What do we want to achieve?

Some assumptions

- What problem is our project trying to solve?
 - Helping people to design electronic circuits and devices
 - Lowering barriers to collaboration (e.g. at CERN)
 - Promoting open-source software ideologies?
 - Promoting education in electronics?

Some assumptions

- Who is our target audience?
 - Students
 - Academics
 - Scientists
 - Professional engineers
 - Amateurs / hobbyists
 - A wide-spectrum of the above?

Some assumptions

- How are we solving this problem?
 - Developing software design tools for schematic capture, circuit simulation, HDL simulation, and PCB layout design.
 - Creating a community around these tools where users can interact and help each other
 - Using the open-source model to reduce barriers to entry, and costs to the user

Some assumptions

- What do we want to achieve?
 - A reduced dependency on commercial tools with high associated costs
 - Increased freedom for users to work productively on electronic designs
 - Enable truly open “Open Hardware”

Why do we work on our projects?

- Personal necessity
- Intellectual challenge
- KUDOS / recognition
- Social interaction with our communities
- Giving back to the community
- Money

Our competition...

Our competition...

- Other FOSS projects?
 - Kicad vs. gEDA
 - gnucap vs. ngspice vs. QUCS vs....

Our competition...

- Other FOSS projects?
 - Kicad vs. gEDA
 - gnucap vs. ngspice vs. QUCS vs....
- NO. These projects should be our allies...

Our competition...

- Other FOSS projects?
 - Kicad vs. gEDA
 - gnuicap vs. ngspice vs. QUCS vs....
- NO. These projects should be our allies...
 - We should not tolerate members of our community who attempt to detract from our fellow FOSS EDA projects.

Our competition...

- Our real competition comes from the give-away or “freemium” tools
 - Eagle (Farnell / Newark / Element14)
 - Design spark (RS Components)
 - Did we miss an opportunity here?
- Traditional commercial tools...
 - Our capabilities are catching up to these

Why collaborate?

- Assuming common goals...
 - Can we better reach them by working together than in isolation?
- Bringing projects with expertise in different domains together as a tool-kit
 - E.g. Does every simulation tool need its own schematic capture system?
- Broadens the scope of expertise we can draw upon when solving problems in our own projects
- Working with others is fun!

However...

- Working together can be difficult!
- Even within the umbrella of gEDA, we have a wide spectrum of differing opinions, design philosophies and approaches.
- Even finding agreement within our project community is sometimes quite difficult.
- Managing this is an ongoing challenge!

Community dynamics

- Absence of clear direction leads and leadership has lead to problems for the gEDA project
- Uncertainty over project future can create concern and mistrust of changes
- A lack of respect for developers and their efforts can detract from their motivation
- Projects need strength of identity

Working together

- Can projects find more ways to work together, and interact – whilst preserving their individual identities?
- Does this mean forming a larger “umbrella” community – would that work?
- How do we encourage increased communication?
- Are our goals sufficiently aligned?

Working together

- What does the geography of these projects look like?
 - USA
 - Europe (mainland, UK etc..)
 - Asia
 - Australia
- Consider hosting multi-project code-sprints where developers can socialise, and work on their projects, and share ideas
- Attend FOSEDM etc..!

Parallels in the desktop communities

- “Desktop summit” conferences (KDE + Gnome)
- freedesktop.org
 - X window system related desktops
 - Code hosting for relevant software projects
 - standards developed between projects
 - Solve things like where where cache + config files live under your home directory
 - Common specifications for menu / launcher entries shared across desktops
- D-Bus

An example of repeated effort

- Data creation
 - Schematic symbols
 - Package footprints
 - 3D models (KiCAD + future)
- Aside from stylistic convention, this data should be universal!
- EDACore proposal from FOSDEM 2015?
- Is this best implemented a software library, file-format(s), or a combination of the two?

Barriers to a shared resource

- Different data-models
 - Do all tools share enough commonality to use the same underlying data?
- File-formats
 - “n” x “n” converters?
 - A common intermediate format?
 - Existing standard – e.g. ISO10303-210?
 - Develop our own standard (or pick an existing format?)
- Stylistic differences (symbols etc..)
 - A problem, even when sharing within a project

HOW STANDARDS PROLIFERATE: (SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC)



<https://xkcd.com/927/>

(Do we want this to be us?)

Plug-in vs. core functionality?

- Make use of the plug-in support in our projects
 - A route to experiment with new ideas for data-sharing, and interoperability
 - Avoids the need to redesign the core of each software project on top of a common library
 - (Helper library vs. core library?)
- Specification + reference implementation?

Barriers to shared code

- Language choice (C/C++/Python/Scheme/...)
- Coding styles
- Licence (GPL3 vs. GPL2+ or other)
- Release schedule / reliability
- API stability
- Control / management of shared resources

Some ideas...

- Develop a common model for representation of the underlying netlist
 - Let different tools share a common view of the low-level electrical connectivity
- API (via D-Bus?) for sharing selection data between different programs?
 - Facilitate cross-probing between schematic and layout, attach waveform views to a simulation etc..

Some ideas...

- Difficulties to be overcome
 - A suitably generic representation that it works across many tools
 - How to express the context of the design / project (gEDA doesn't really have a “project” concept)

Some ideas...

- Possible gEDA / gschem roadmap
 - Remove special-case hard-coded functionality
 - Re-introduce as “personality” mode modules
 - More net-list awareness at schematic capture time
 - Allows richer interaction with other tools
 - Should allow introducing new, work-flow specific behaviours

Some ideas...

- Educational market
 - Need for low barriers to entry
 - Desire for workflow integration
 - Schematic capture
 - Simulation (analogue / HDL / firmware)
 - Board layout (low-end mostly)
 - Low cost (University budgets may be tight)
 - Potential for funded development effort here though
 - Find / create tomorrow's FOSS EDA developers?

Some ideas...

- Shared advocacy efforts
 - FOSS EDA tools present a credible alternative to expensive commercial tools in many areas
 - Don't forget to acknowledge other projects. Some may better fit a given user's needs than your own
 - Promote the common goals we share, as well as any unique points about your own project(s)

Thank-you

Any questions?