

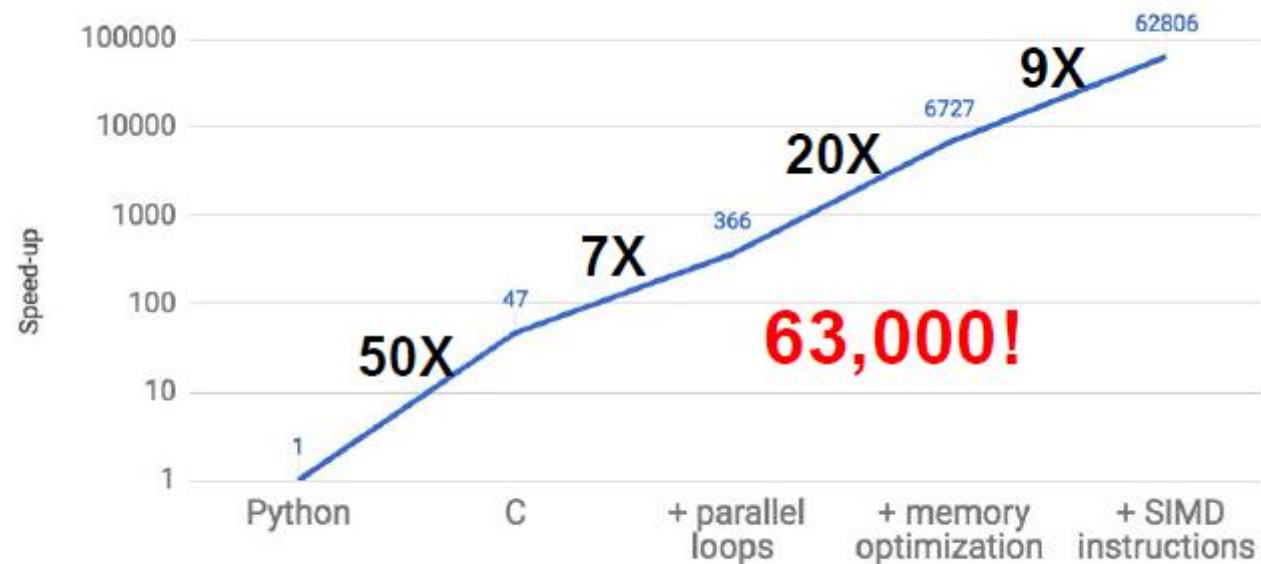


Addressing the EDA Roadblocks for Domain-specific Compilers: An Industry Perspective

Alireza Kaviani, Ph.D.
Senior Fellow of Engineering
AEAI CTO office

The Age of Domain Specific Architectures

Matrix Multiply Speedup Over Native Python



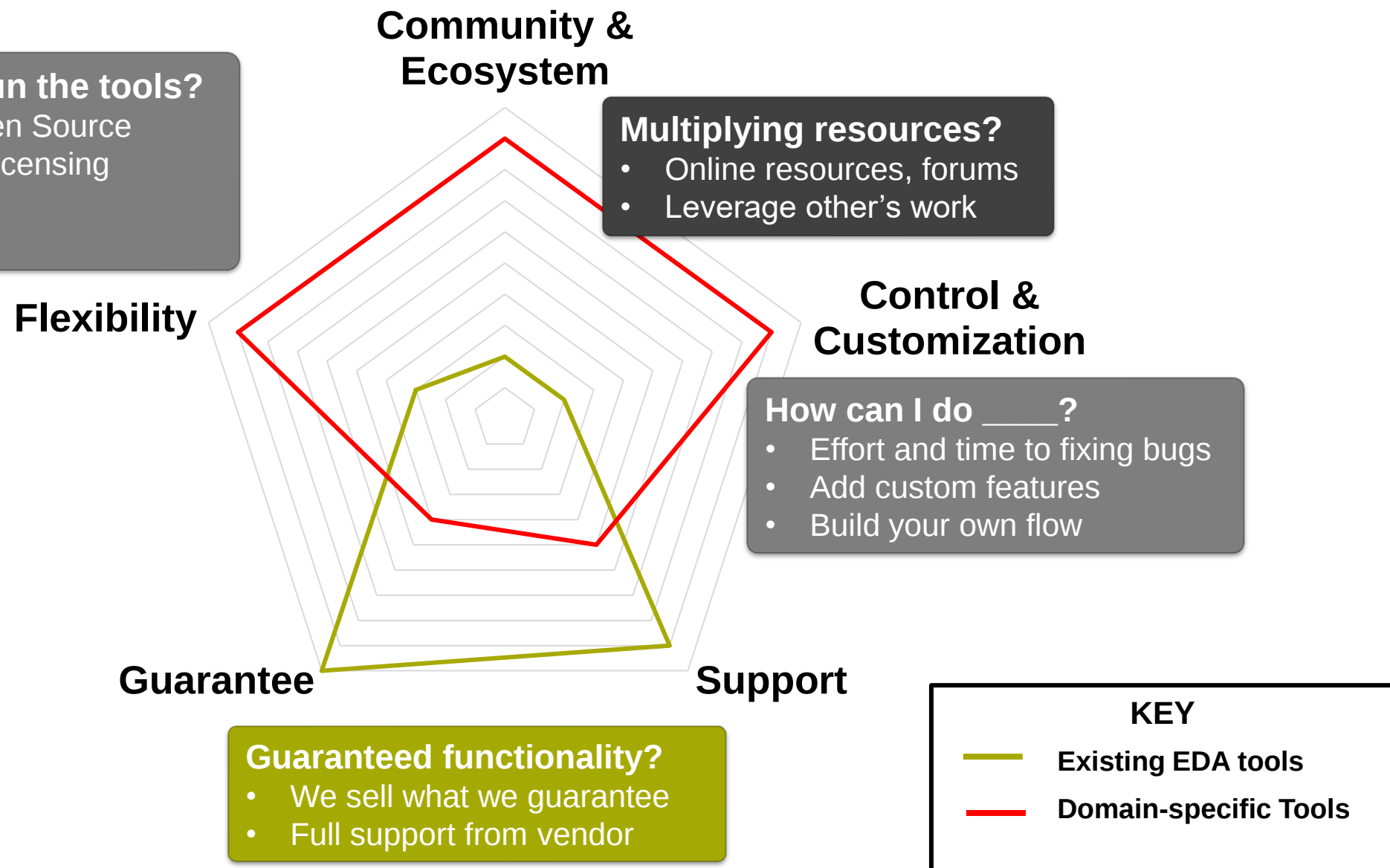
End of Dennard scaling
and slowing Moore's law

Customization
opportunity

from: "There's Plenty of Room at the Top," Leiserson, et. al., to appear.

- ▶ Achieve higher efficiency by tailoring the architecture to characteristics of the domain
 - More effective parallelism for a specific domain, More effective use of memory bandwidth
 - Domain specific programming language

Towards Domain-Specific EDA tooling



Domain Specific Era Needs Domain Specific Backends

Application Domains



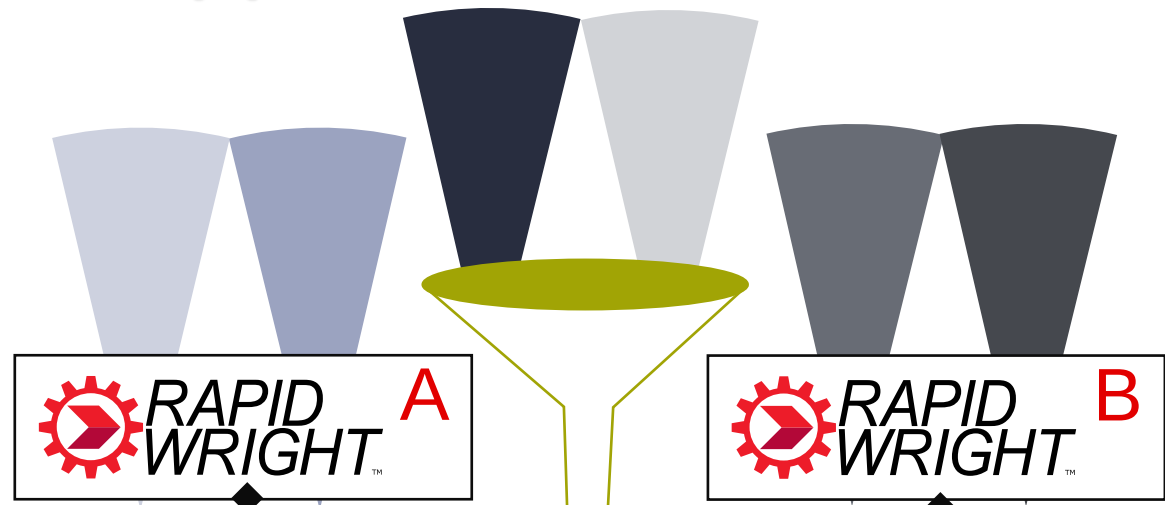
VIVADO



Vivado must **generalize** solutions

- ▶ Exploit domain-specific traits for:
 - Higher performance
 - Faster compile time
 - Timing closure predictability

Application Domains



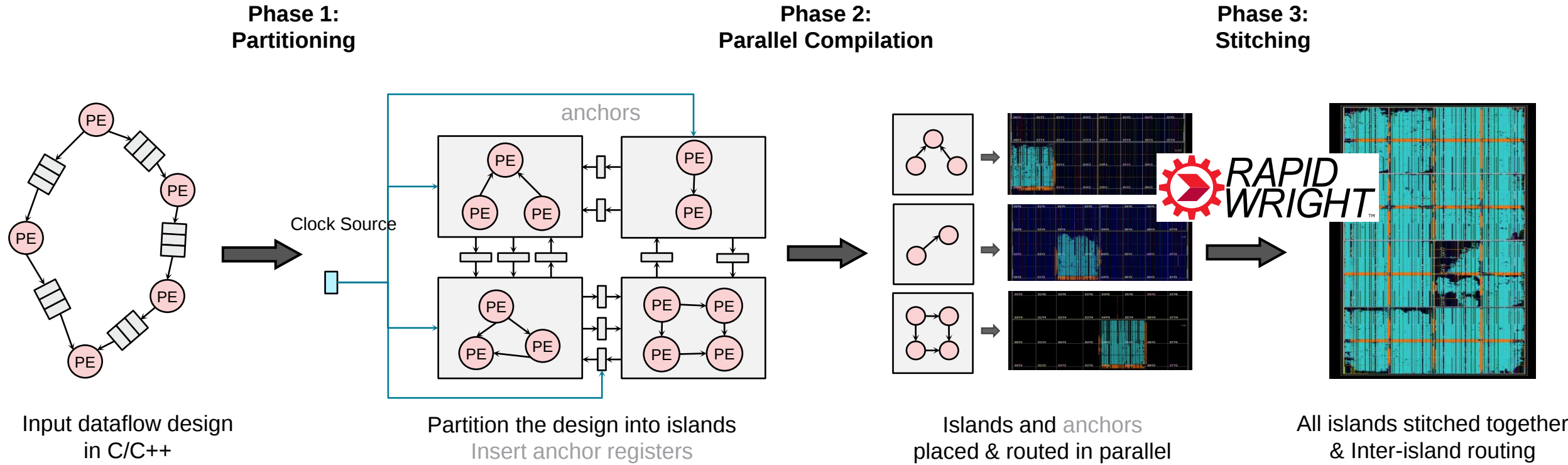
VIVADO



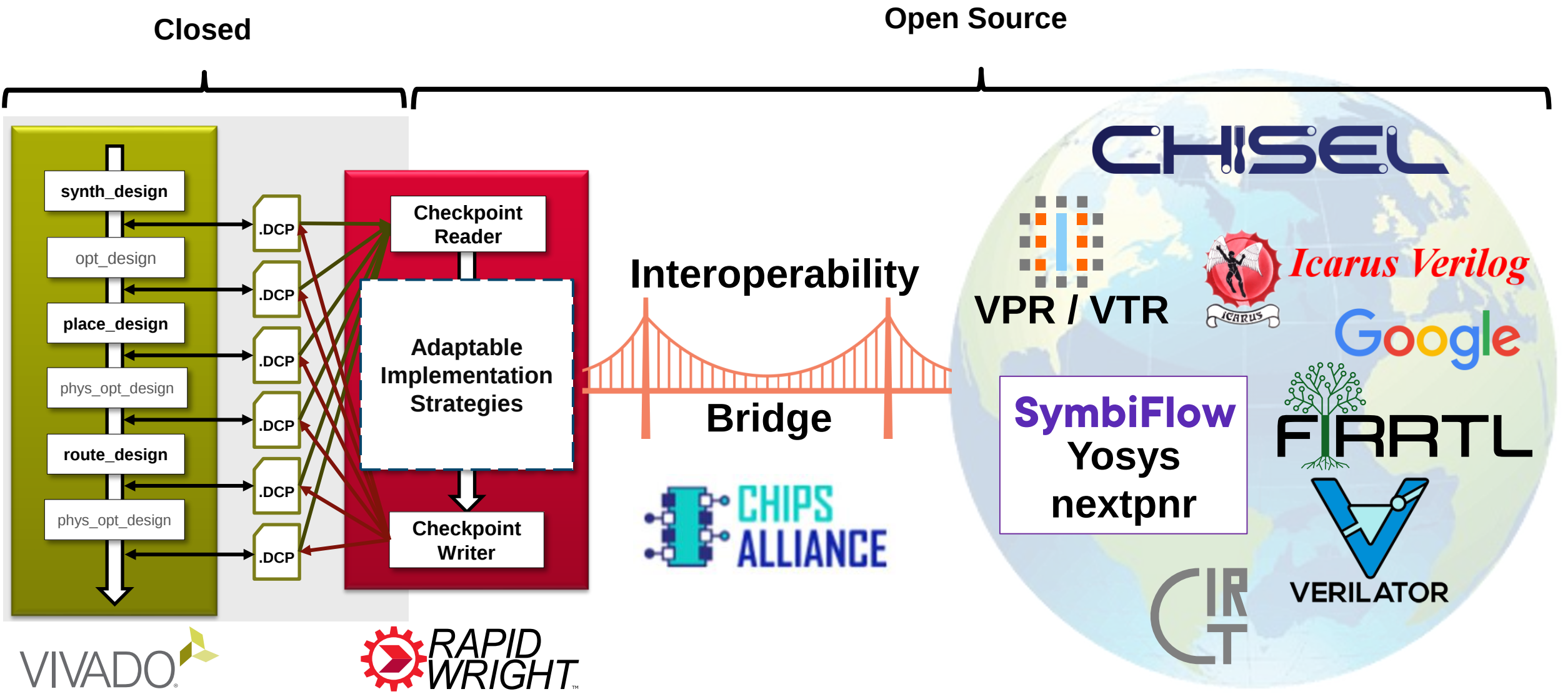
RapidWright can **specialize**

www.rapidwright.io

Domain-specific Fast P&R Using Split-Compilation

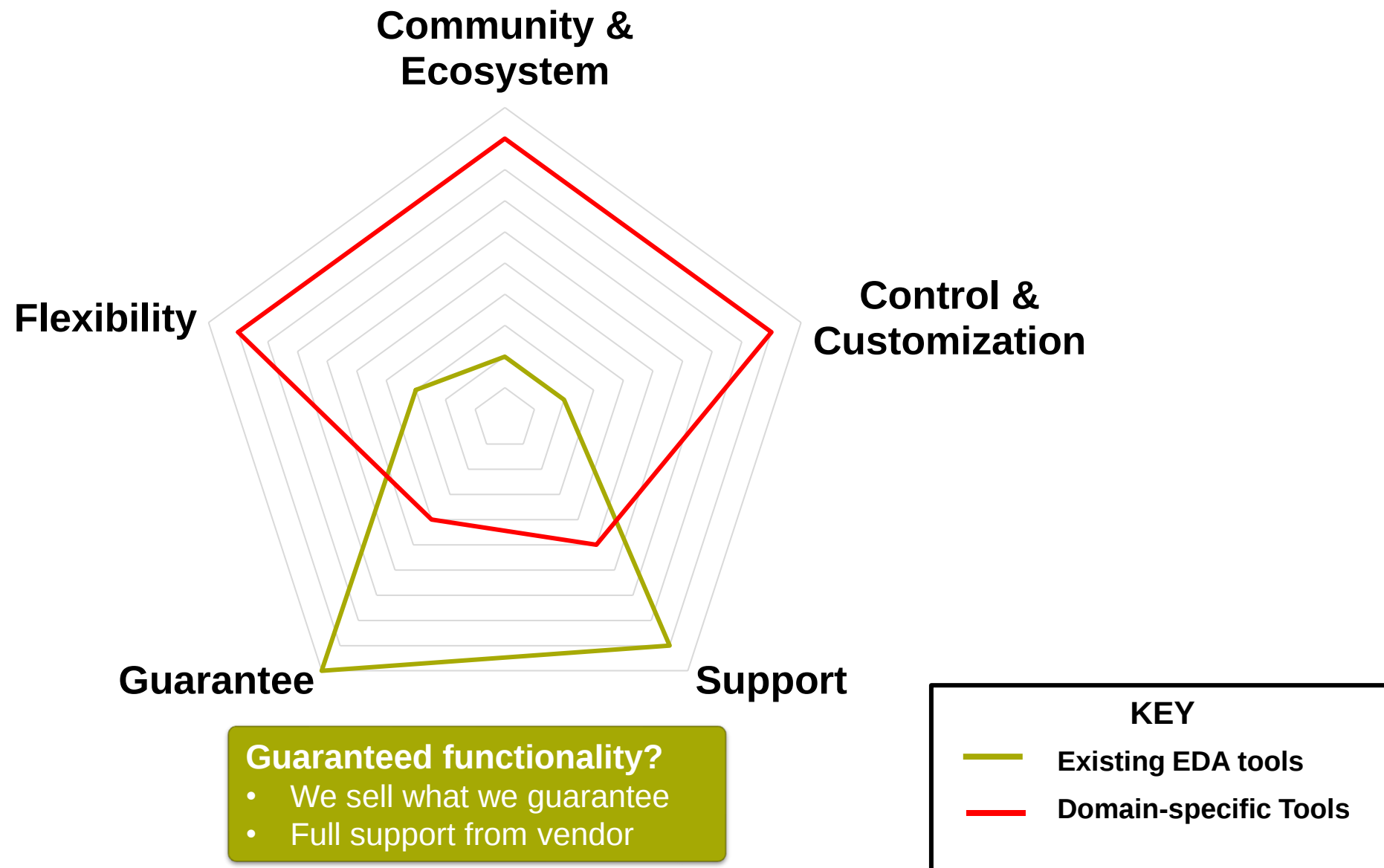


Interchange Project: The Bridge to Open Source

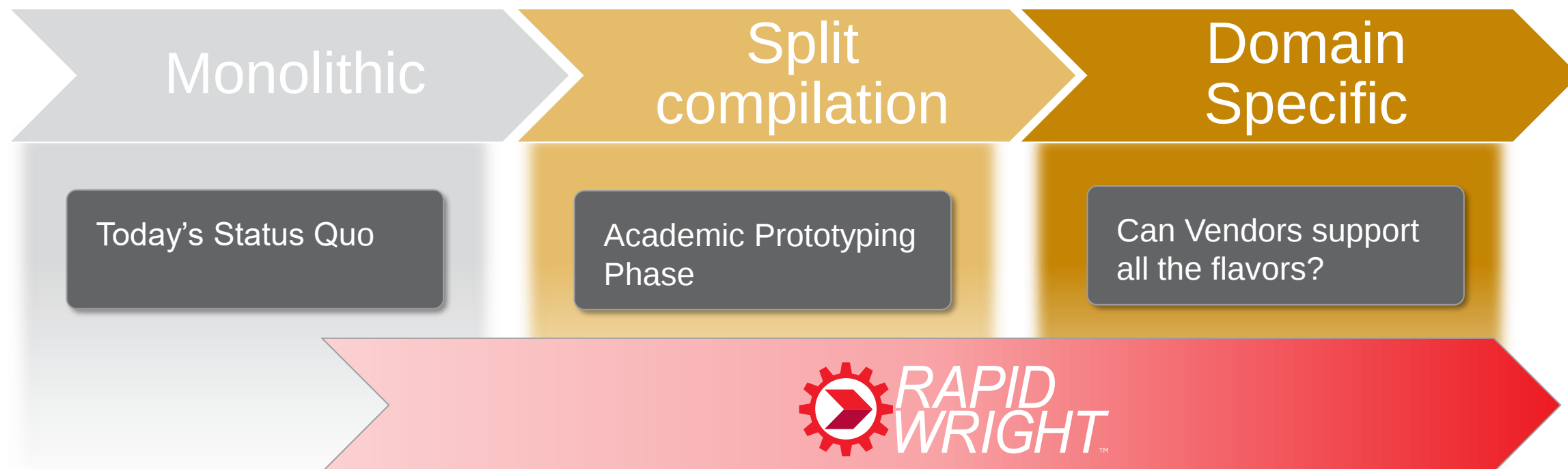


<https://github.com/chipsalliance/fpga-interchange-schema/>

Towards Domain-Specific EDA tooling



Macro Trends of Compilation Strategies

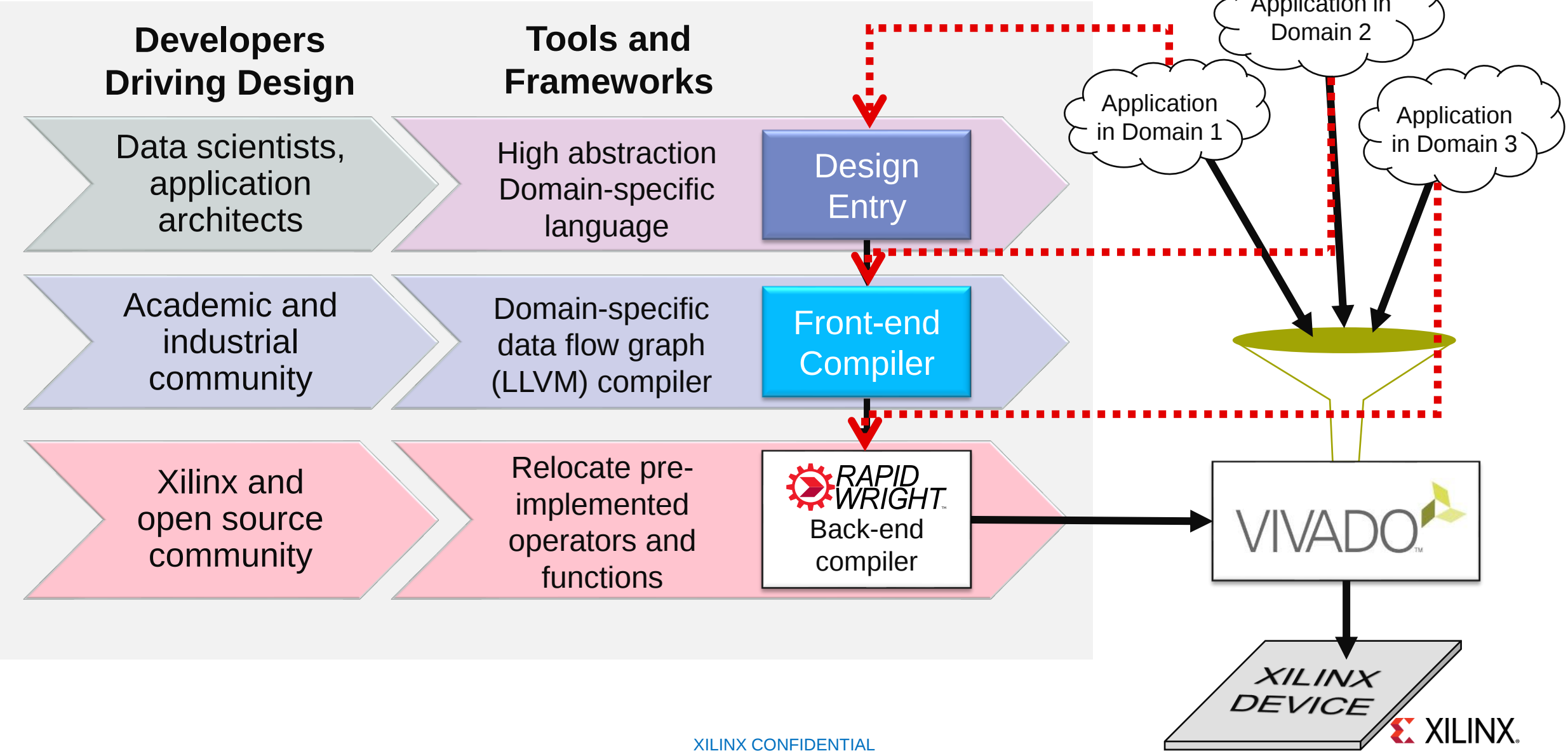


Future flows will require advanced customization capabilities:

- Open-source companion platforms are the best pathway for domain-specific technology
- We need new ways of providing support and guarantee

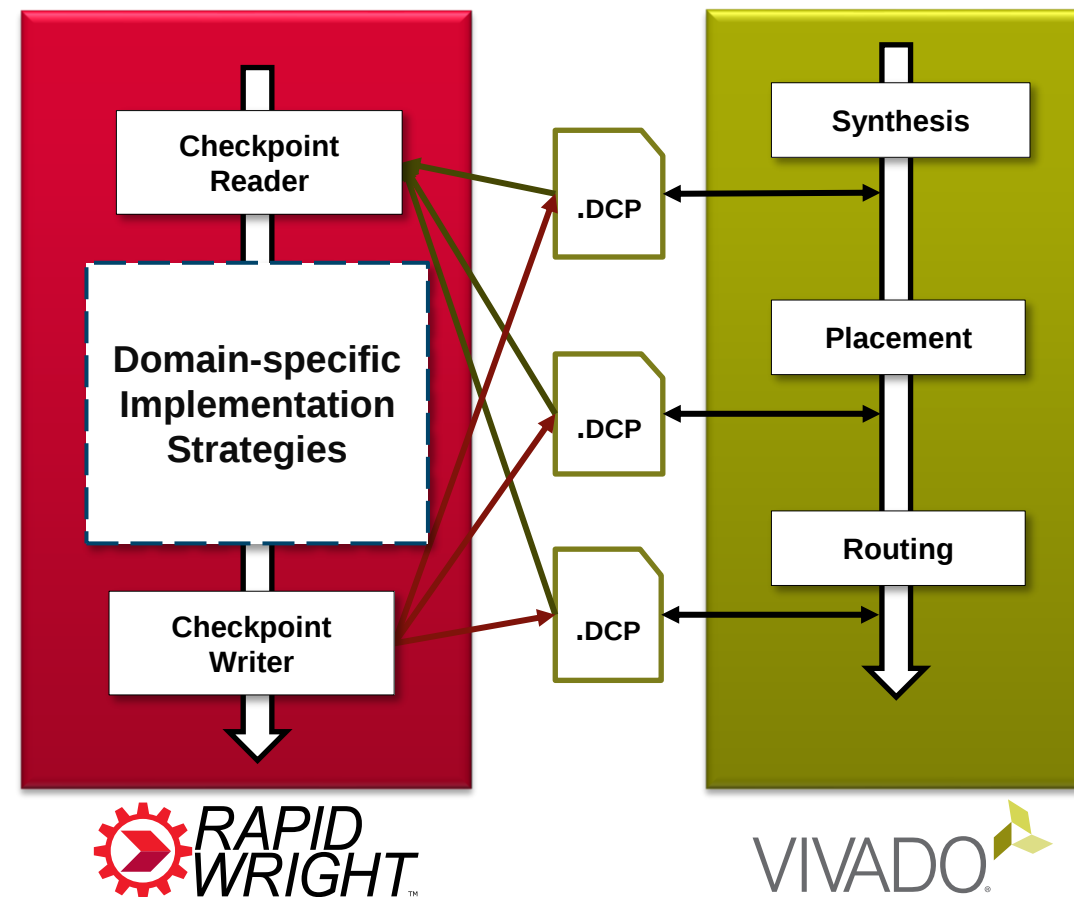


Domain Compilers



What is RapidWright?

- ▶ Companion framework for Vivado
 - Fast, light-weight, open source
 - Source code independent of Vivado
 - www.rapidwright.io
- ▶ Enables targeted solutions
 - “Build-your-own” flow
 - “Copy and Paste” implementations
 - Domain-specific fast compile flows
 - Customized bottom-up flows
 - Rapid prototyping of CAD concepts



DCP = Design Checkpoint (FPGA compilation snapshot)