Yosys and nextpnr Update

David Shah
Griffith Research Ltd

Griffith Research

Yosys

- Open source synthesis framework
- Multiple FPGA families
- ASIC synthesis, formal verification, …

Yosys – new features

- ABC9 (Eddie Hung)
- New interface to ABC optimisation/mapping framework
- Improved timing-driven synthesis
- Sequential (flip-flop) optimisations

Yosys – new features

- cxxrtl (whitequark)
- C++ “backend” similar to Verilator but built into Yosys
- Efficient handling of multiple clock domains, loops, etc
- Can be used for mixed-language sim

Yosys – new features

- GHDL plugin (tgingold)
- Adds an open VHDL frontend
- Still experimental, but good language coverage
- Capable of building Microwatt, MiST SNES core, ...

nextpnr

- Open source FPGA place and route, development started May 2018
- Multi-architecture, aimed at real-world FPGAs including advanced functionality
- Timing driven
- Python API for extensibility and experimentation

Current Status

- Stable support for iCE40 and ECP5 devices
- Experimental Xilinx and generic Python deivce support
- Extensively tested up to ~100kLUT designs
- DDR3 controllers, Gigabit Ethernet, 64-bit SoCs

Current Status

- Analytical and simulated annealing placers
- Two congestion-based routers
- Packing is per-architecture

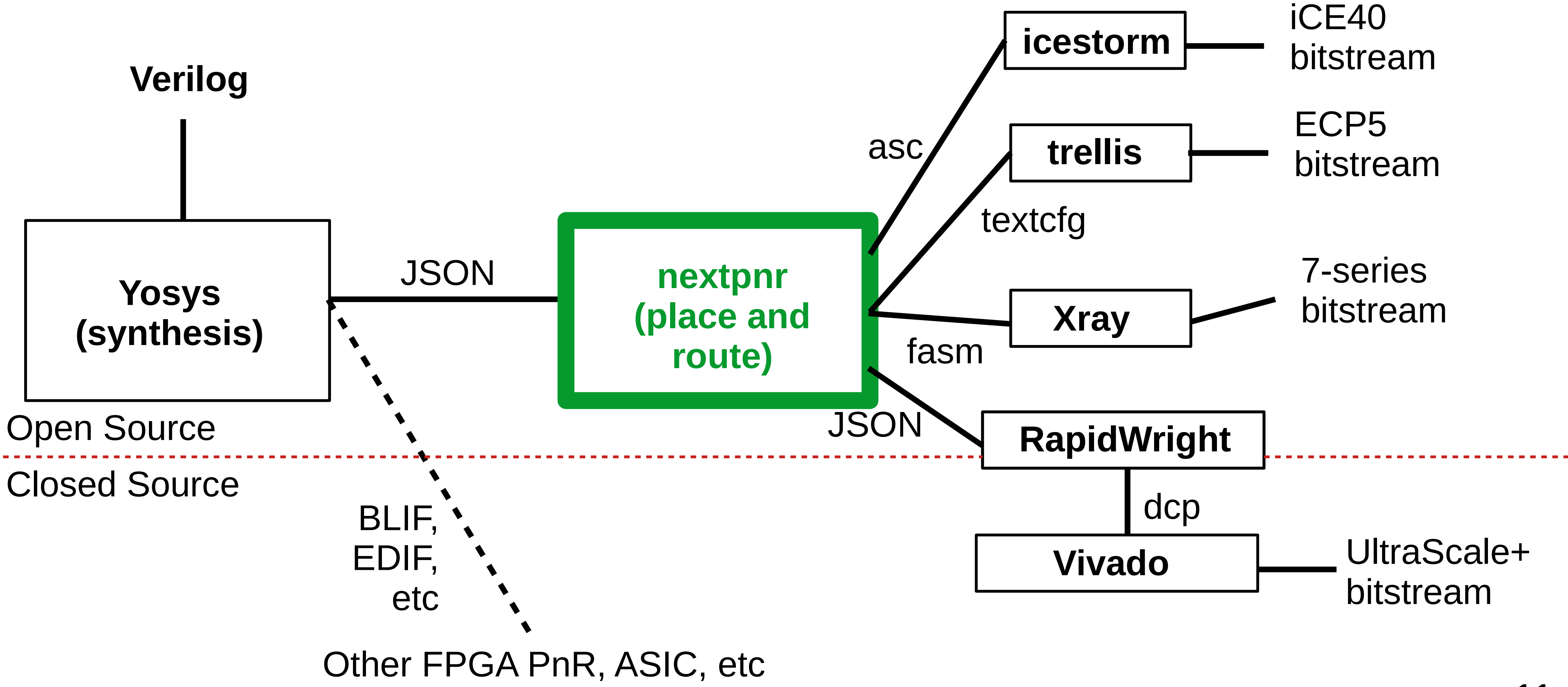
nextpnr-xilinx

- Current ongoing efforts to support Xilinx devices
- Not yet upstream
- RapidWright: UltraScale+, Vivado bitgen
- Project X-Ray: 7-series, FOSS bitgen

nextpnr-xilinx

- Highly experimental, but can build complex designs
- Linux SoC with DDR3
- Main issue is runtime scaling for larger devices

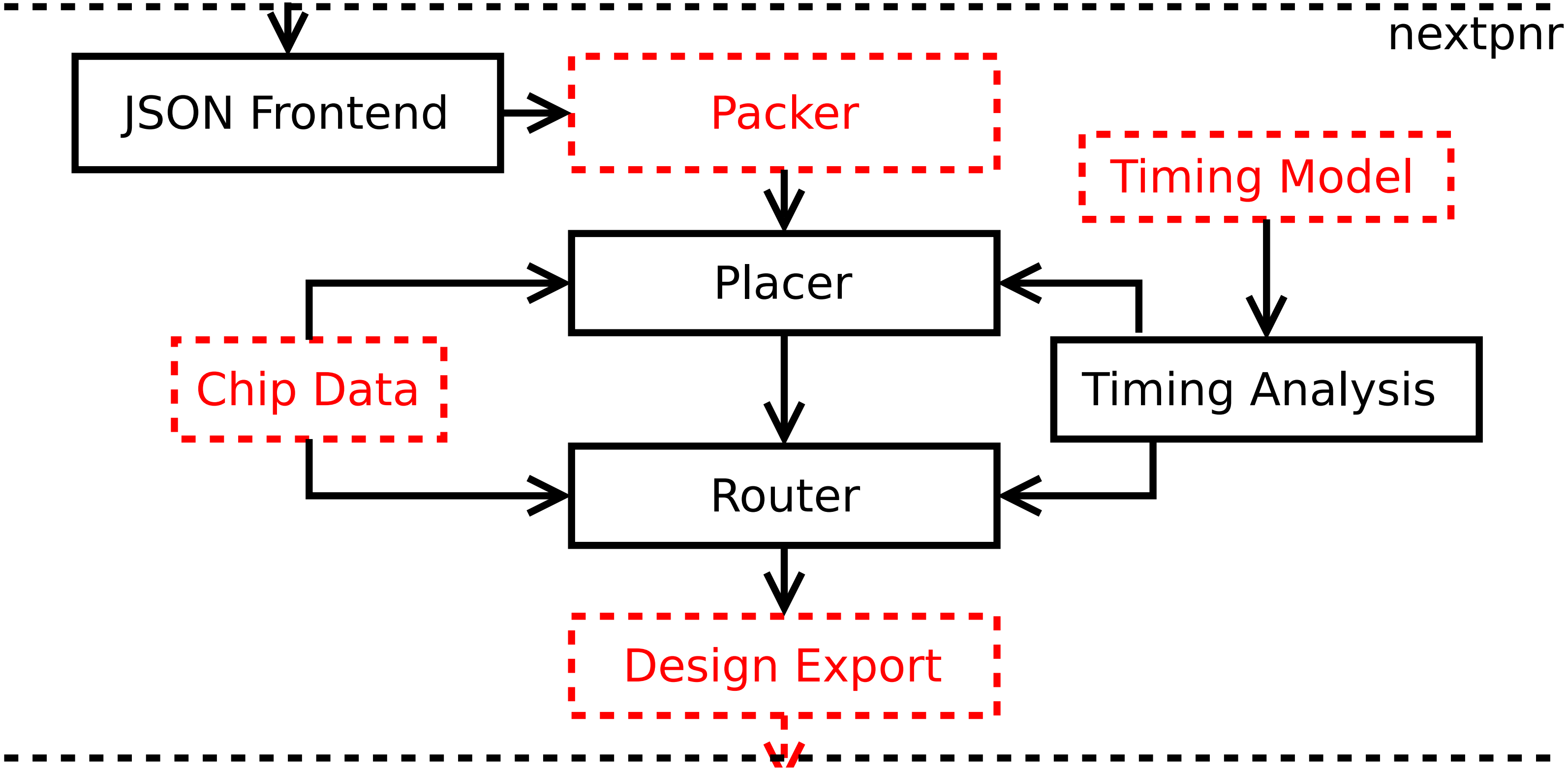
nextpnr flow



Inside nextpnr

- Architectures are code, not data
- Implement arbitrarily complex constraints to support real-world architectures
- Custom packing, DRC, bitstream generation, etc
- Multiple choices of placer and router

Inside nextpnr



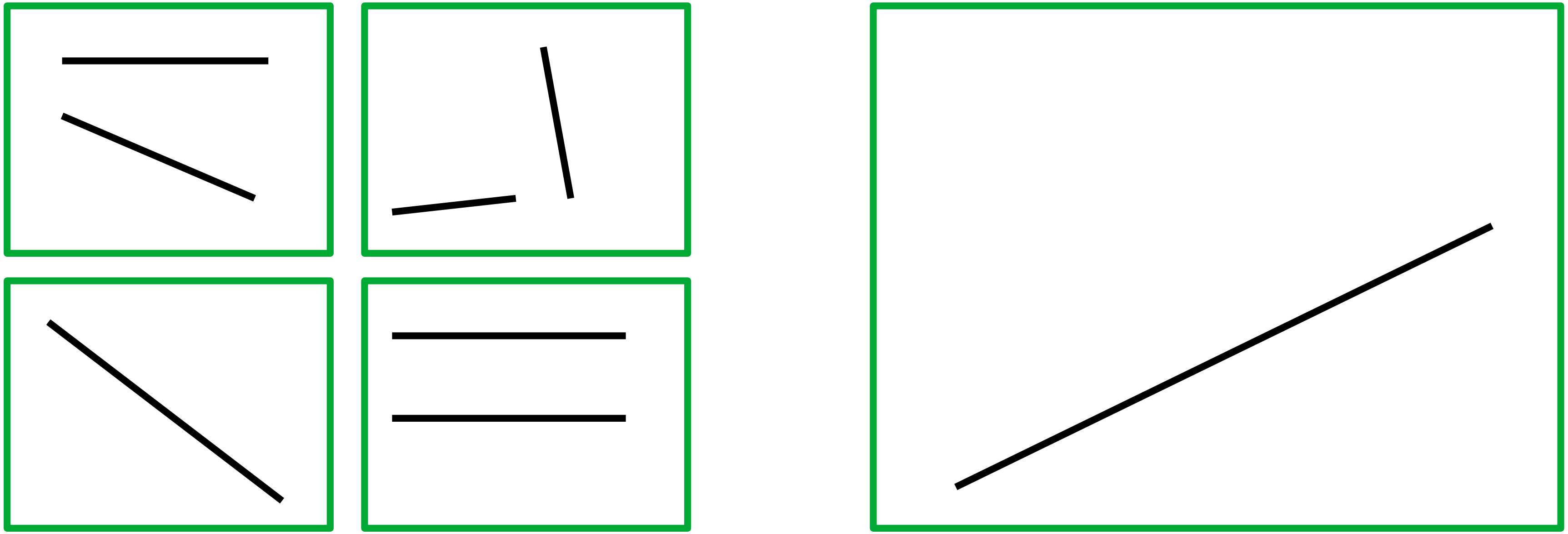
router2

- New congestion driven router based on CRoute (Ghent University)
- Negotiated congestion, overlaps allowed during early phases
- Penalty for most congested wires

router2

- Region-based parallelisation
- Arcs that don’t cross determined regions routed in their own thread
- Long arcs or arcs not containable to a bounding box are single-threaded

router2 - parallelisation



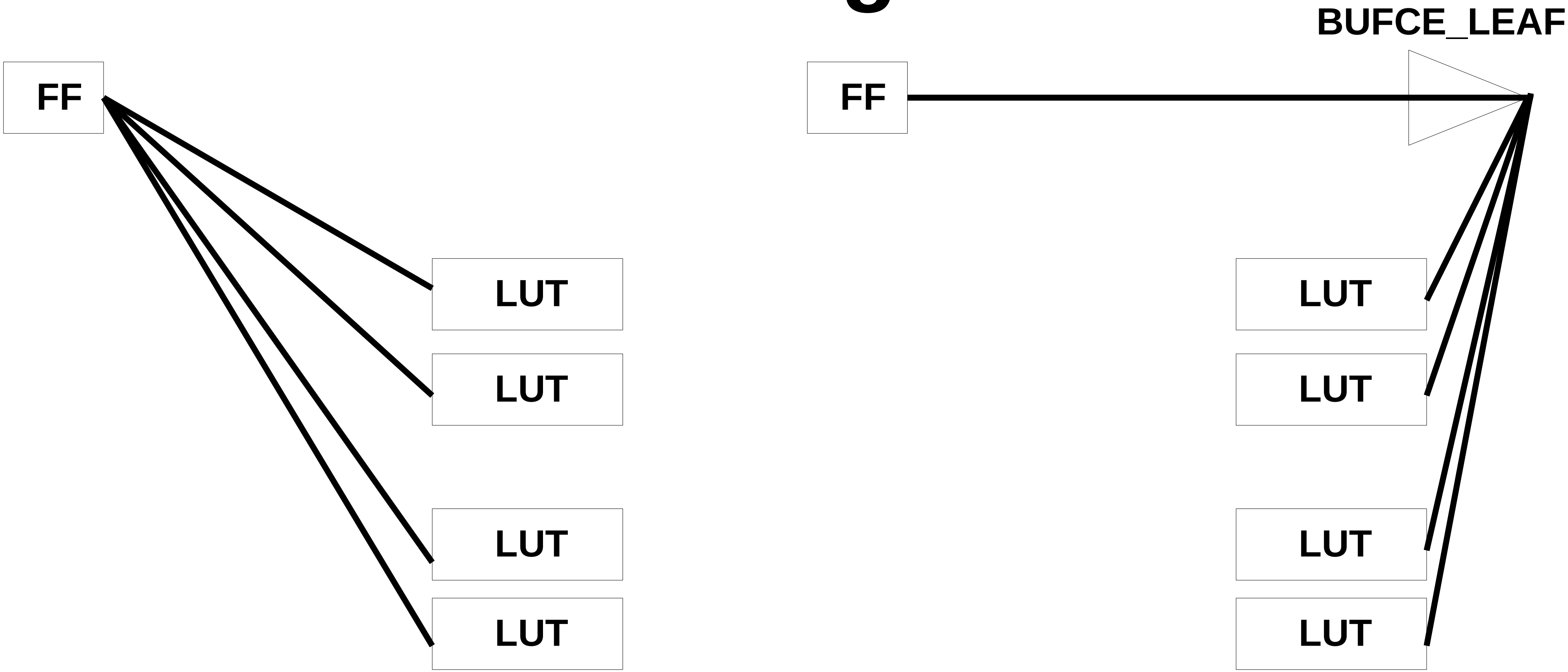
router2

- Parts of the router are arch-overrideable
- Architectures can choose to route (or try to route) some segments themselves
- Used to speed-up constant routing for Xilinx FPGAs
- Scope to replace current pre-route clock routing pass

router2

- Architectures can break arcs into segments to force use of dedicated resources
- Example use cases:
  - UltraScale leaf clock buffers for high fanout signals
  - Cross-SLR routing for SSI FPGAs
  - Special-case IO/IP routing?

router2 - segments



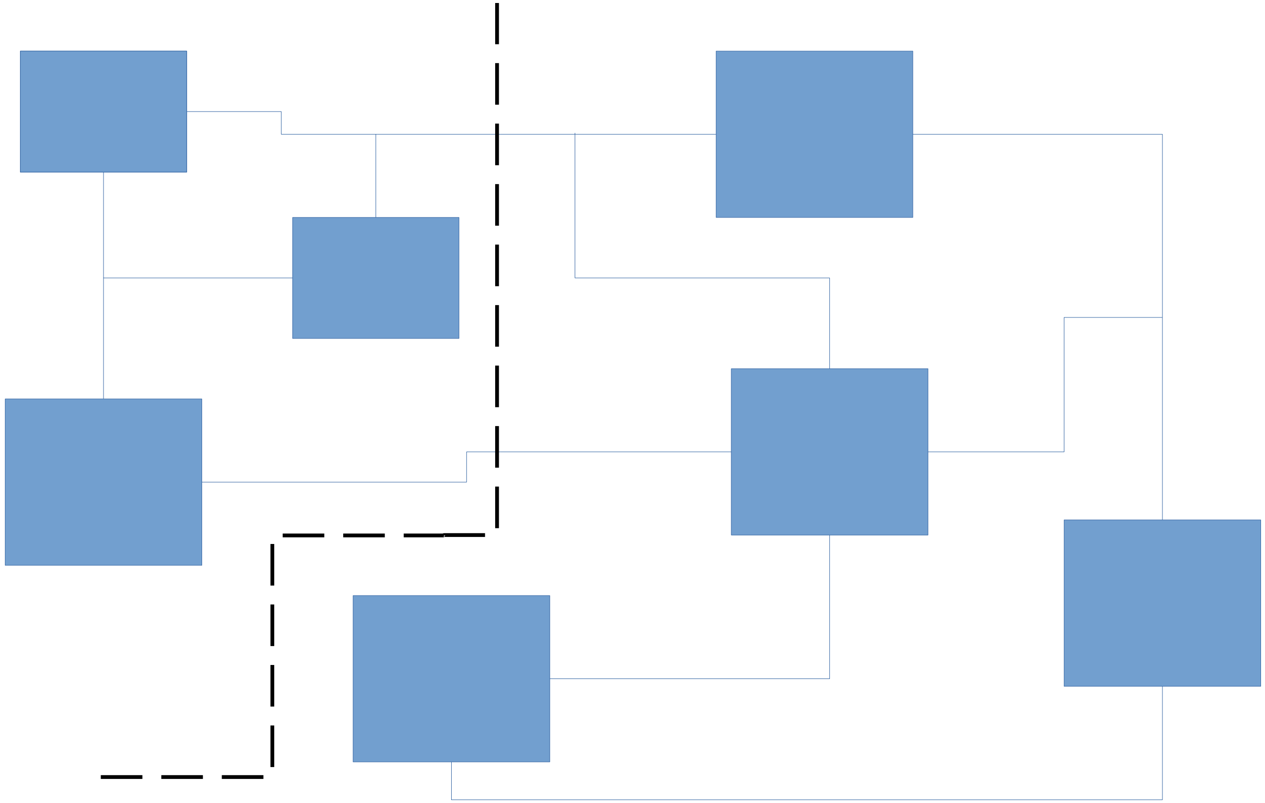
RippleFPGA placer

- Routeability driven analytical placement
- Several heuristics to reduce routing congestion
- Avoids placement in congested areas
- Pack-and-place based on fine grain elements

RippleFPGA placer

- Core similarities to existing HeAP placer
- Assign higher “area” to congested cells to spread them more
- Partition at start to improve initial placement

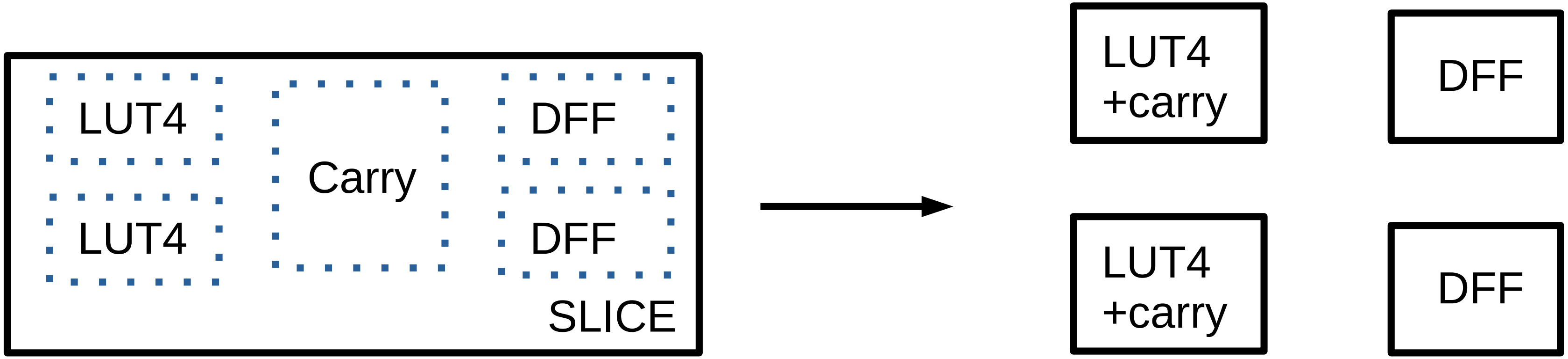
Hypergraph Partitioning



Hypergraph Partitioning

- Not only useful for Ripple
- Logic Regioning (chiplet) partitioning for large FPGAs
- Parellelisation of various algorithms
- Multi-device designs?

Fine-Grain Placement



Future Work

- Finishing Ripple placer
- Improved timing constraints
- Retiming and resynthesis framework