ATRENTA®

# RTL Handoff -
# *Why? & How?*

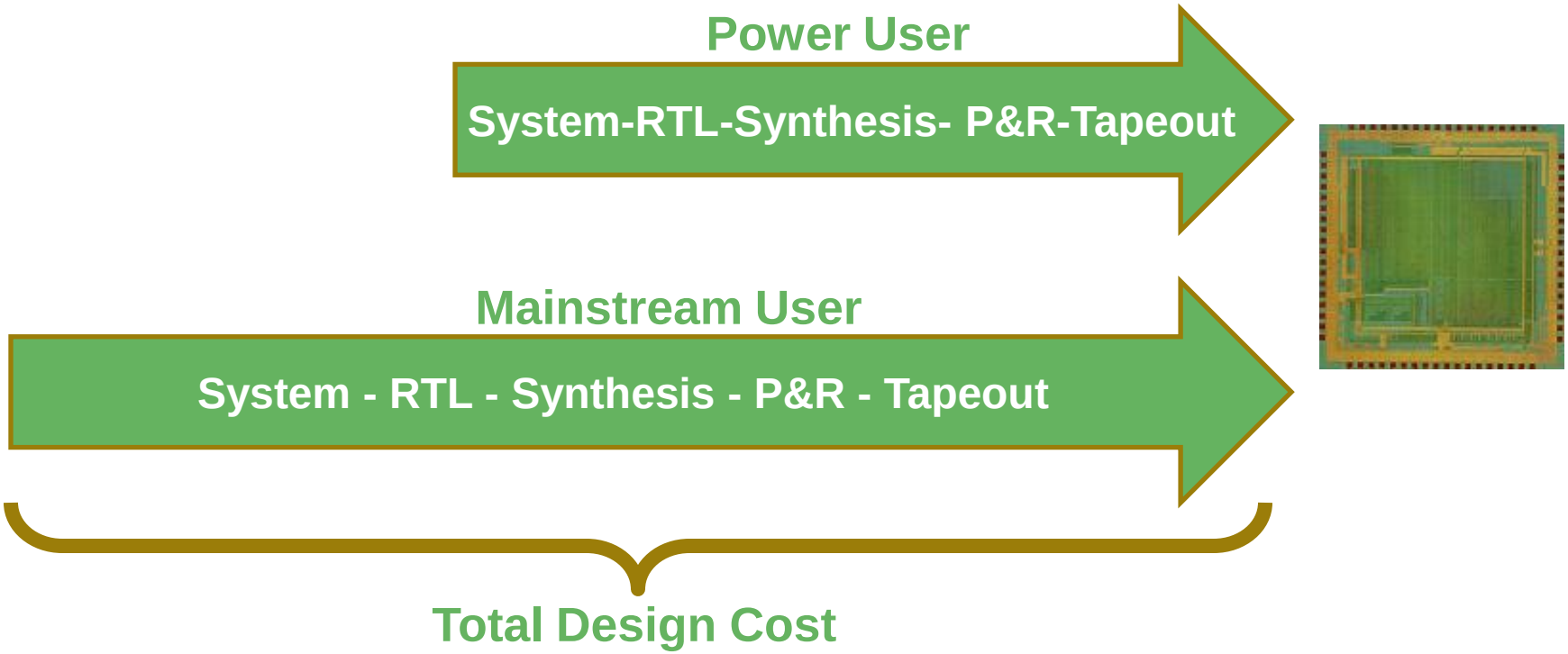
**EDP 2009, Monterey**

**April 10**

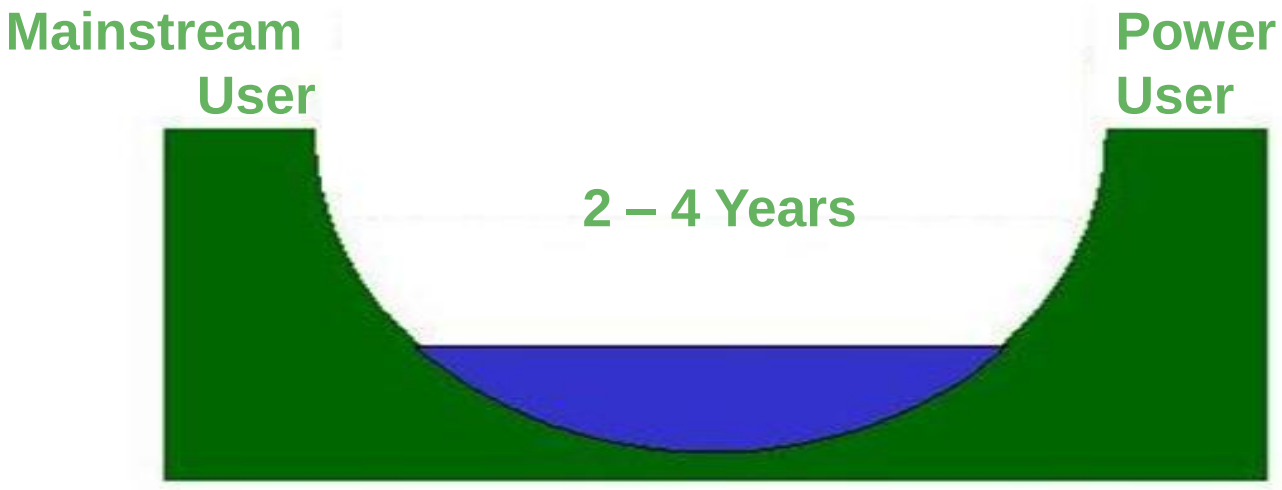
THINK EARLY DESIGN CLOSURE® THINK ATRENTA®

# Cutting Edge SoC - Cost Is A Major Barrier To Entry

- **The design productivity gap limits competition**
  - A ***Power User*** can get an advanced SoC designed for ~$22M
  - A ***Mainstream User*** needs ~$33M
  - ***Mainstream Users*** *can't compete* – *Margins too low*



# Can’t Get There From Here

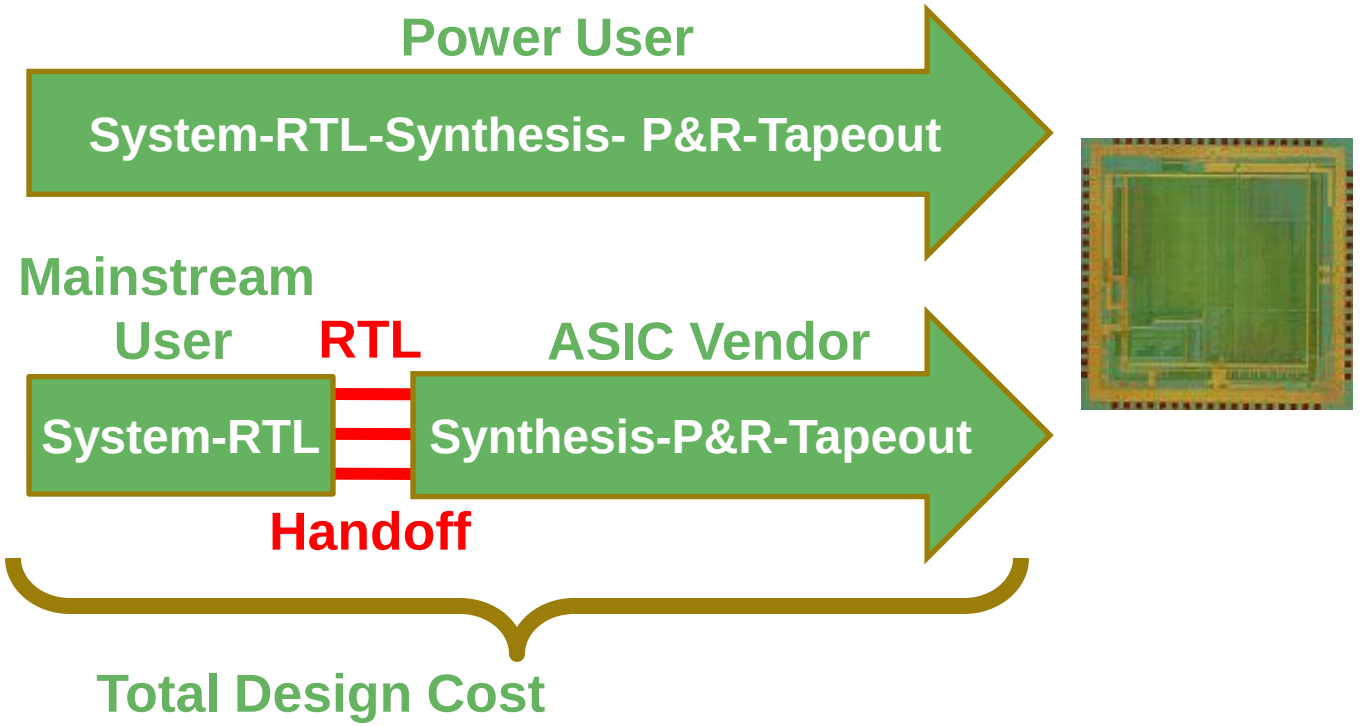



- Experience
- Staff training
- Tool investment
- Infrastructure investment

# RTL Handoff – The Bridge to Higher Margins

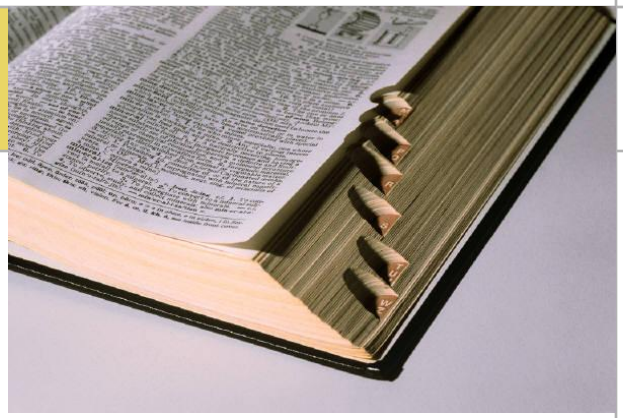

- **Mainstream User focuses on system architecture – RTL**
  - Core competency of the organization
- **Handoff all implementation tasks to Power User**
  - Custom chip supplier (aka ASIC vendor)
  - Core competency of the organization

# Taming the Design Productivity Gap




- **RTL Handoff**
  - Unambiguous definition of ***design intent***
  - Architecture, interfaces, power strategies, routing feasibility, clock domains, timing, false paths, test strategies

# Definitions



## Design Closure

***The complete, unambiguous description of some aspect of a design***

## Virtual Prototype

***A prediction of the power, area, routing congestion or timing behavior of a design <u>before</u> physical implementation is performed***

# Early Design Closure® Enables RTL Handoff

Enter @ architecture

RTL Handoff

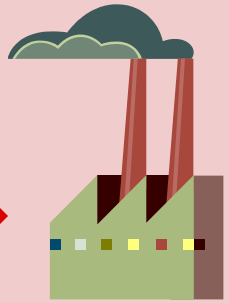
Debug & close @ RTL

## Early Design Closure

## Late Design Closure

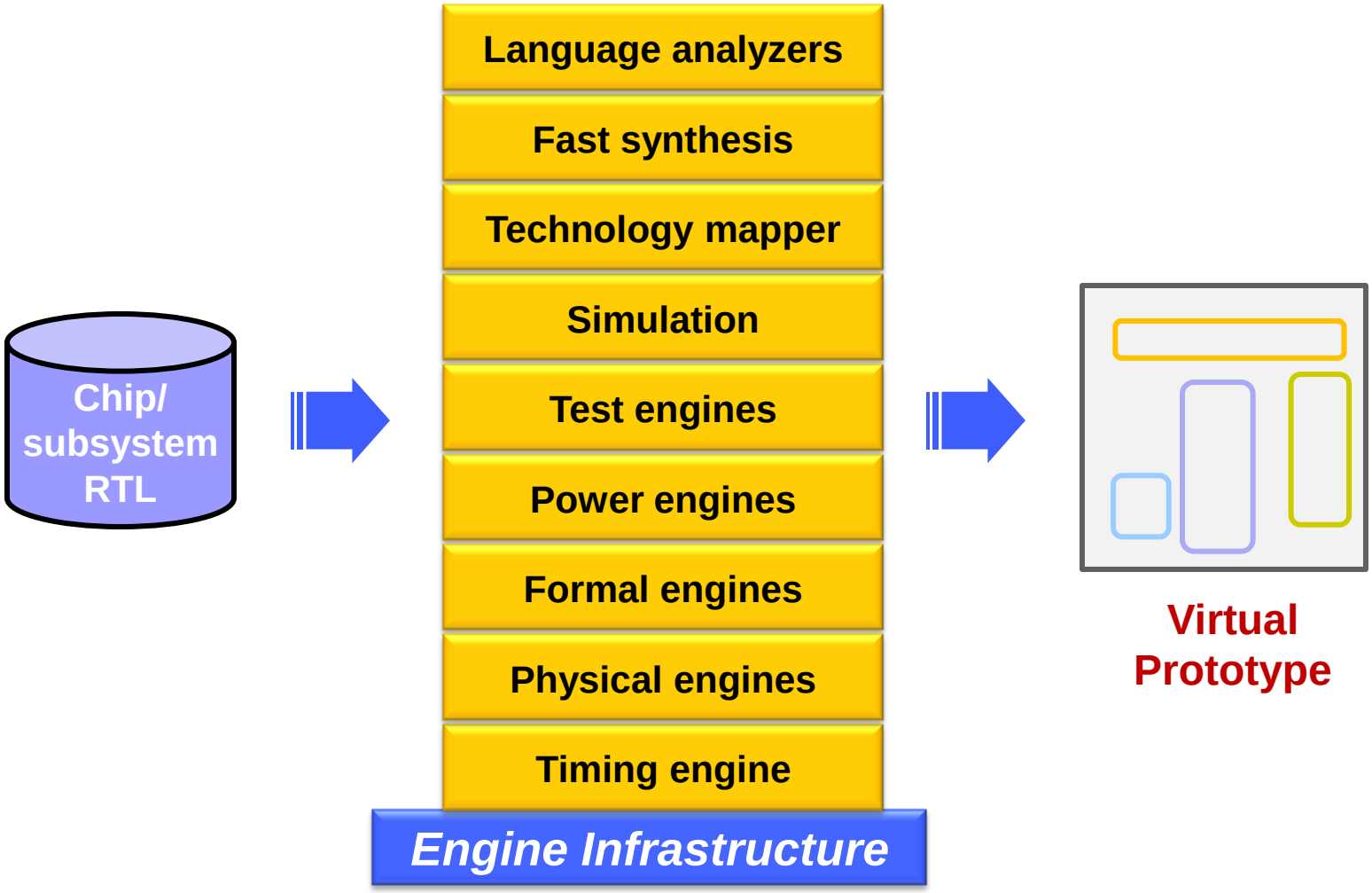
Enter @ RTL

Traditional Method



Debug & close @ implementation

# Virtual Prototype Makes RTL Handoff Realistic



# RTL Handoff – What's Needed?

- **Robust analysis**
  - Power, area, timing, congestion, clock synch, constraints, test, floorplan
- **Correlation to the back end**
  - Not perfect, but "good enough"
- **An unambiguous way to specify design intent to the back-end**
  - Standards can help (on a good day)
- **An environment to assemble the chip, perform the analysis, and deal with iterations from the back-end (*stuff* happens)**

# One Final Thought…

**What defines the boundaries of your SoC design capability?**

**Your ASIC vendor's physical implementation environment?**

**OR**

**You and your architectural vision?**