# Co-Design Pain Points, Tooling Gaps, and Lessons from the Past

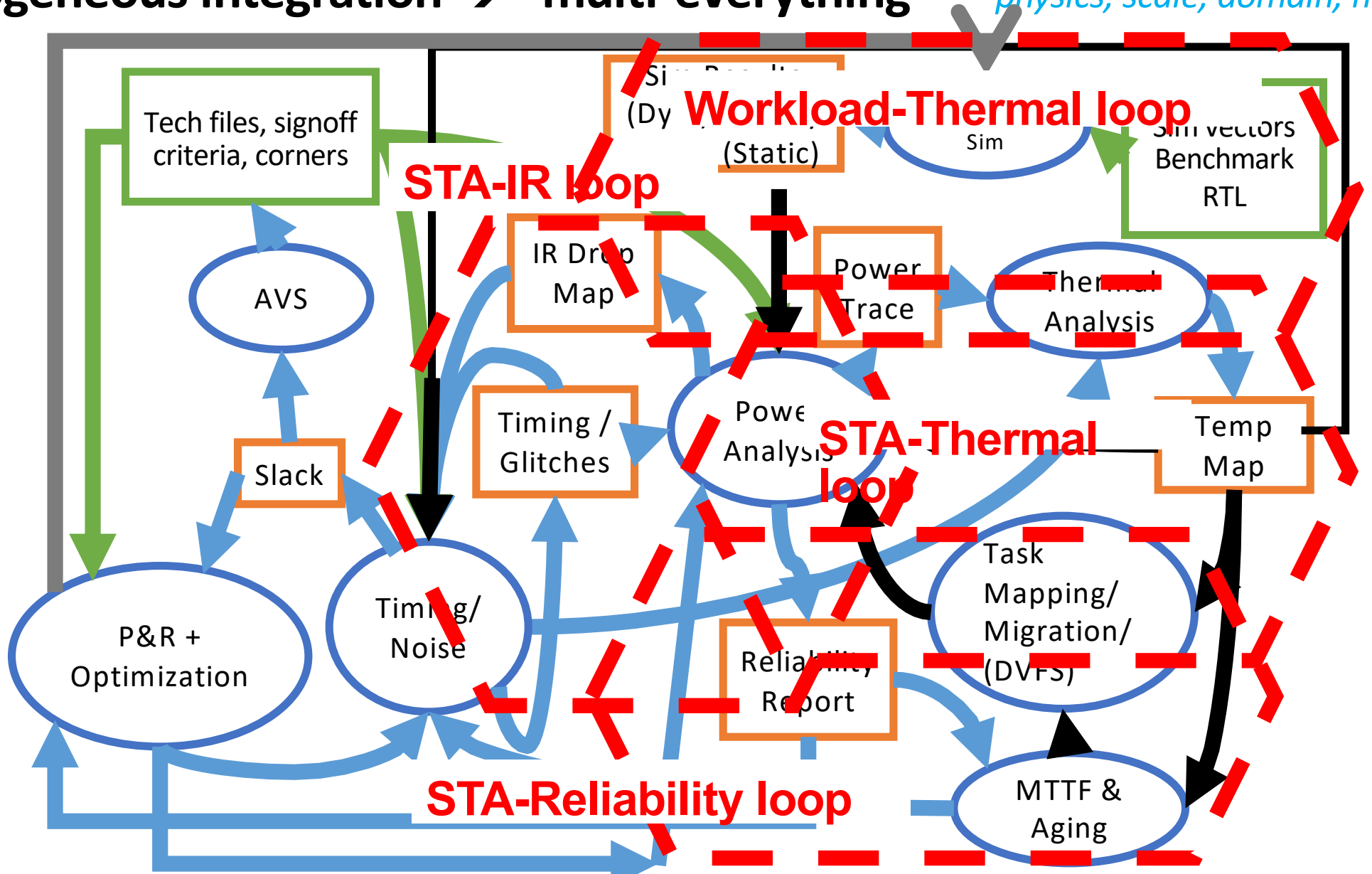
Andrew B. Kahng, UC San Diego CSE and ECE Depts.

http://vlsicad.ucsd.edu/~abk

ECTC Evening Panel, May 29, 2018

# Pain

- **Heterogeneous integration → "multi-everything"** *physics, scale, domain, hierarchy …*



# Pain

- **Heterogeneous integration → "multi-everything"** *physics, scale, domain, hierarchy …*
- **Imagine the pain of "system-on-chip" design …**

https://developer.arm.com/products/system-design/system-guidance/system-guidance-for-infrastructure

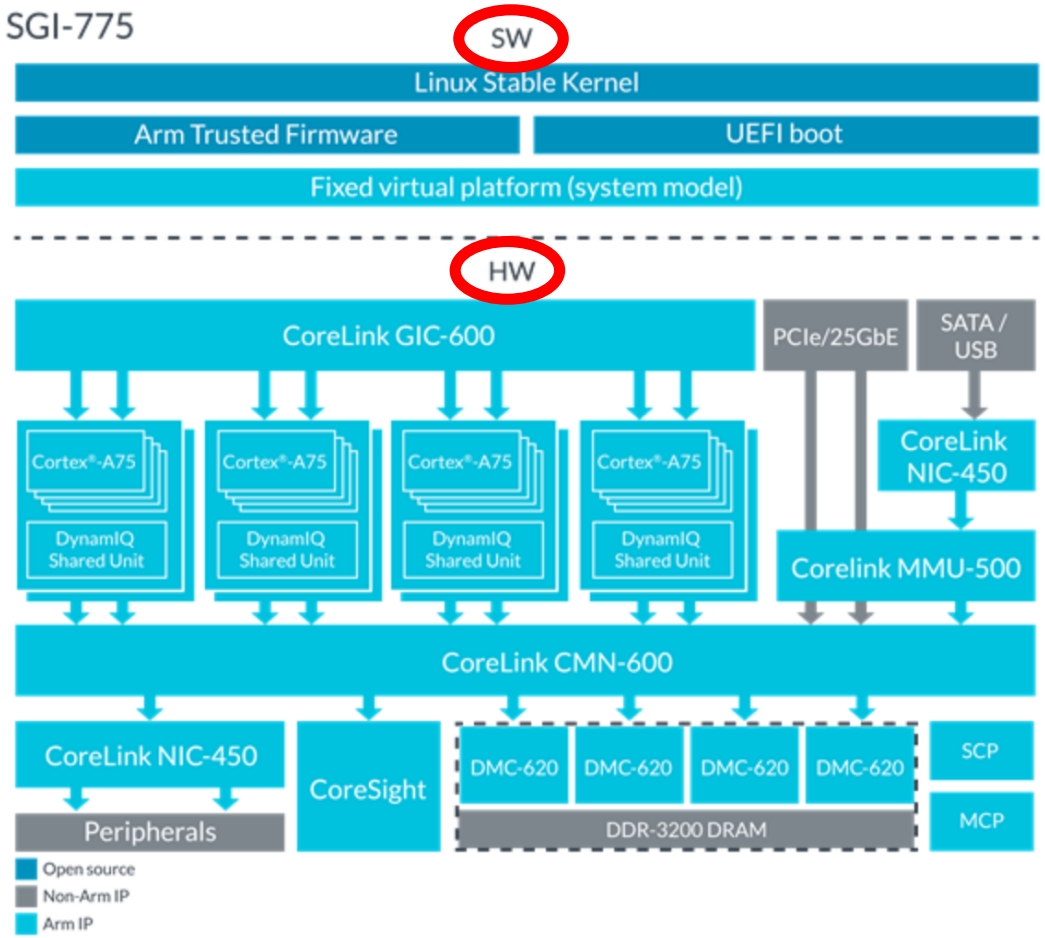
SGI-775 is the collection of resources to guide partners looking to design a Server SoC, using Arm Cortex-A75 and CoreLink CMN-600.

Hardware details:

- Server Based System Architecture (SBSAv3)
- 32x Cortex-A75 with private L2
- DynamIQ with L3 Cache options
- System Level Cache options
- Up to 4x DDR4-3200 (DMC-620)

Software stack:

- Standard Platform interfaces with Arm Trusted Firmware
- Linux Kernel



# Pain

- **Heterogeneous integration → "multi-everything"** *physics, scale, domain, hierarchy …*
- **Imagine the pain of "system-on-chip" design, for which guidance looks like:**
  https://developer.arm.com/products/system-design/socrates-system-builder

Arm Socrates System Builder is a new tool that guides a user through the selection, configuration and creation of Arm IP, and system assembly in weeks, not months:

- Intelligently configure Arm IP and reduce the time to assemble Arm-based systems to weeks, not months.
- Get the system performance you expect, through Arm IP that is configured, built, and integrated right first time.
- Simplify configuration of Arm CoreSight and CoreLink IP and supersedes AMBA Designer. It automatically creates a CoreSight system or CoreLink interconnect that is right first time.

**CoreLink creation**

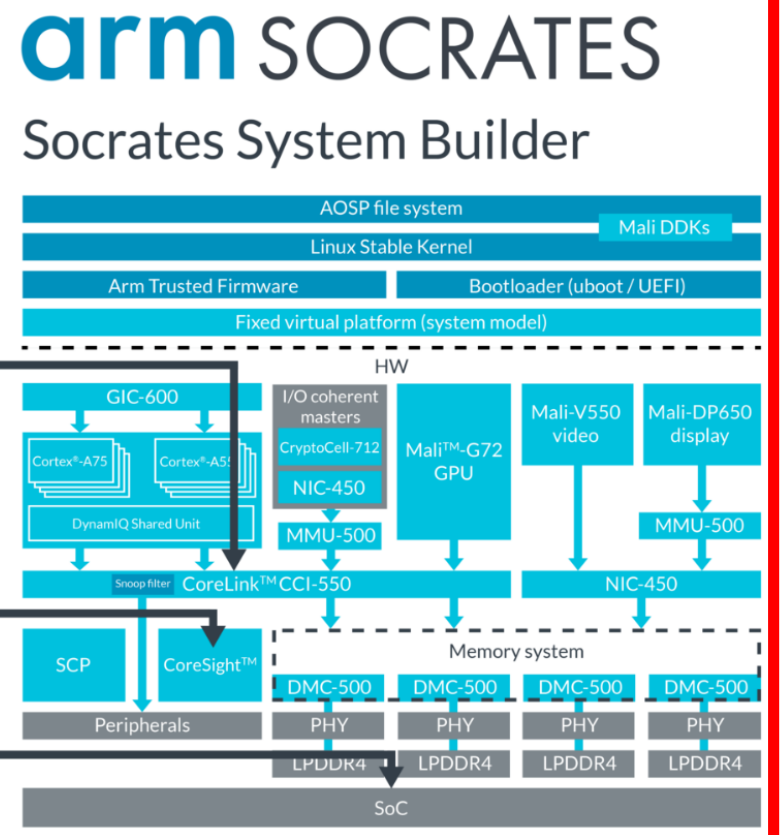
Makes configuration of CoreLink IP simple and automatically creates your CoreLink interconnect system that is right first time

**CoreSight creation**

Makes configuration of CoreSight IP simple and automatically creates your CoreSight system that is right first time

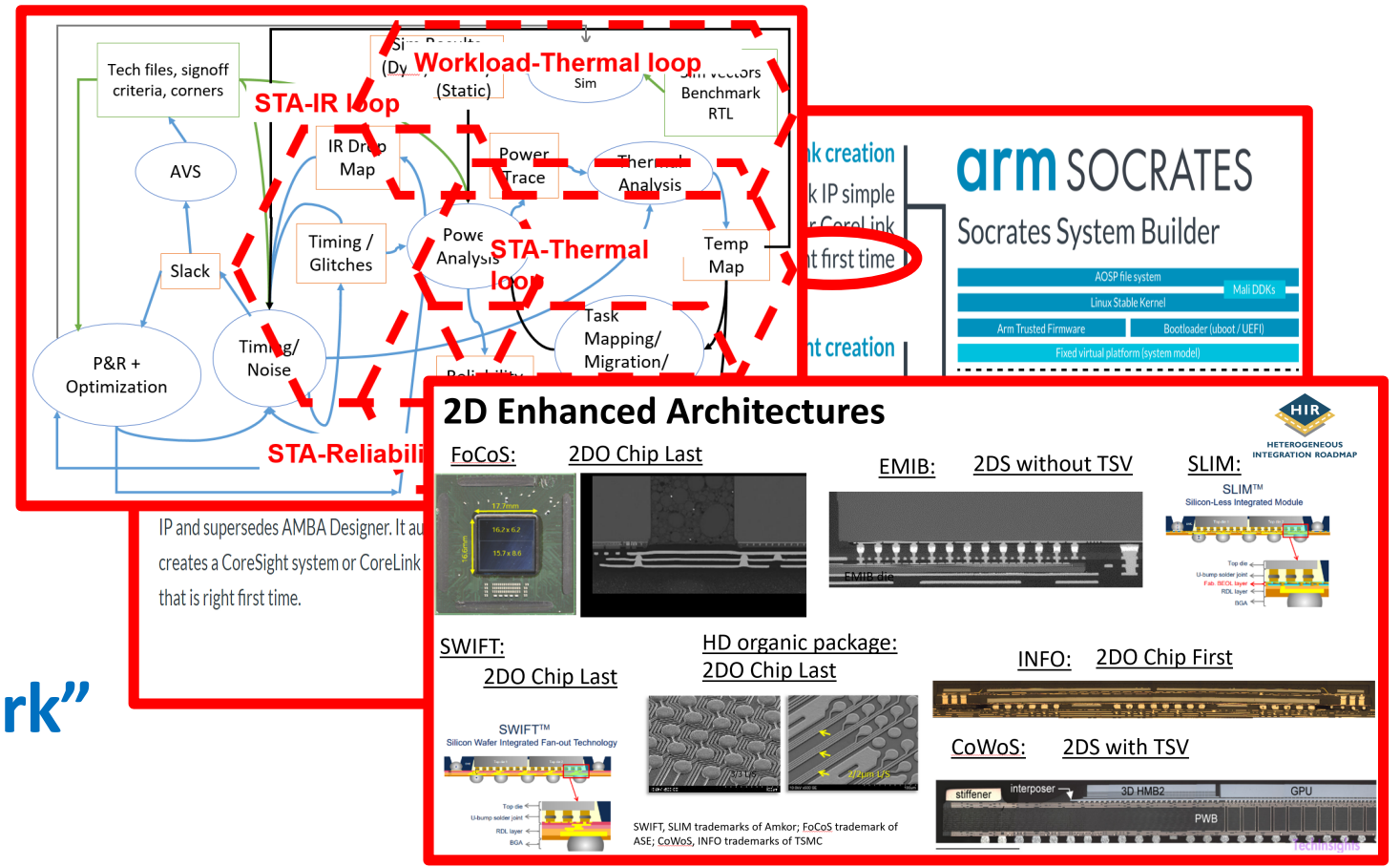
**IP and system creation**

Enables intelligent configuration and integration of Arm IP to quickly and easily build Arm-based systems



# Extreme Pain

- **Heterogeneous integration → "multi-everything"** *physics, scale, domain, hierarchy …*
- **Imagine the pain of "system-on-chip" design …**
- **Now imagine the pain of system-in-package, "2.xD / 3D", "heterogeneous integration" … co-design**
  **= today's panel ☺**



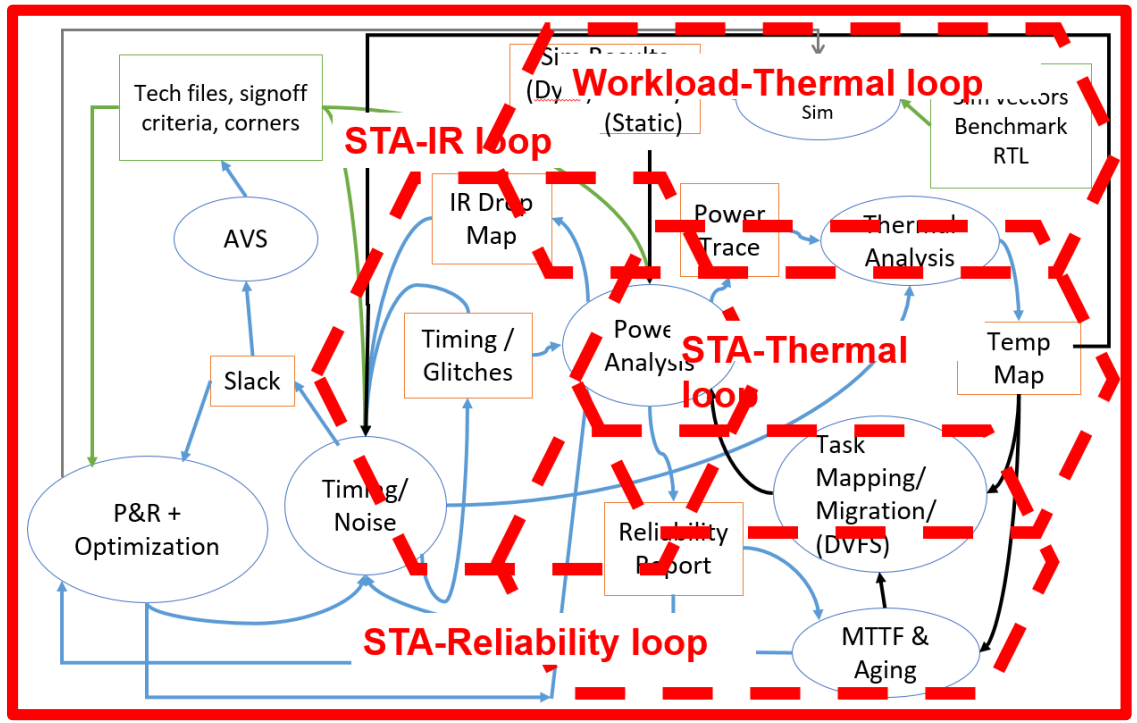

## MISSING:

- **Pathfinding (tech,system,design)**
- **Design Space Exploration**
- **Optimization vs. Checking**
- **"What is possible" vs. "Will it work"**

# Pain Points ... = Tool Gaps

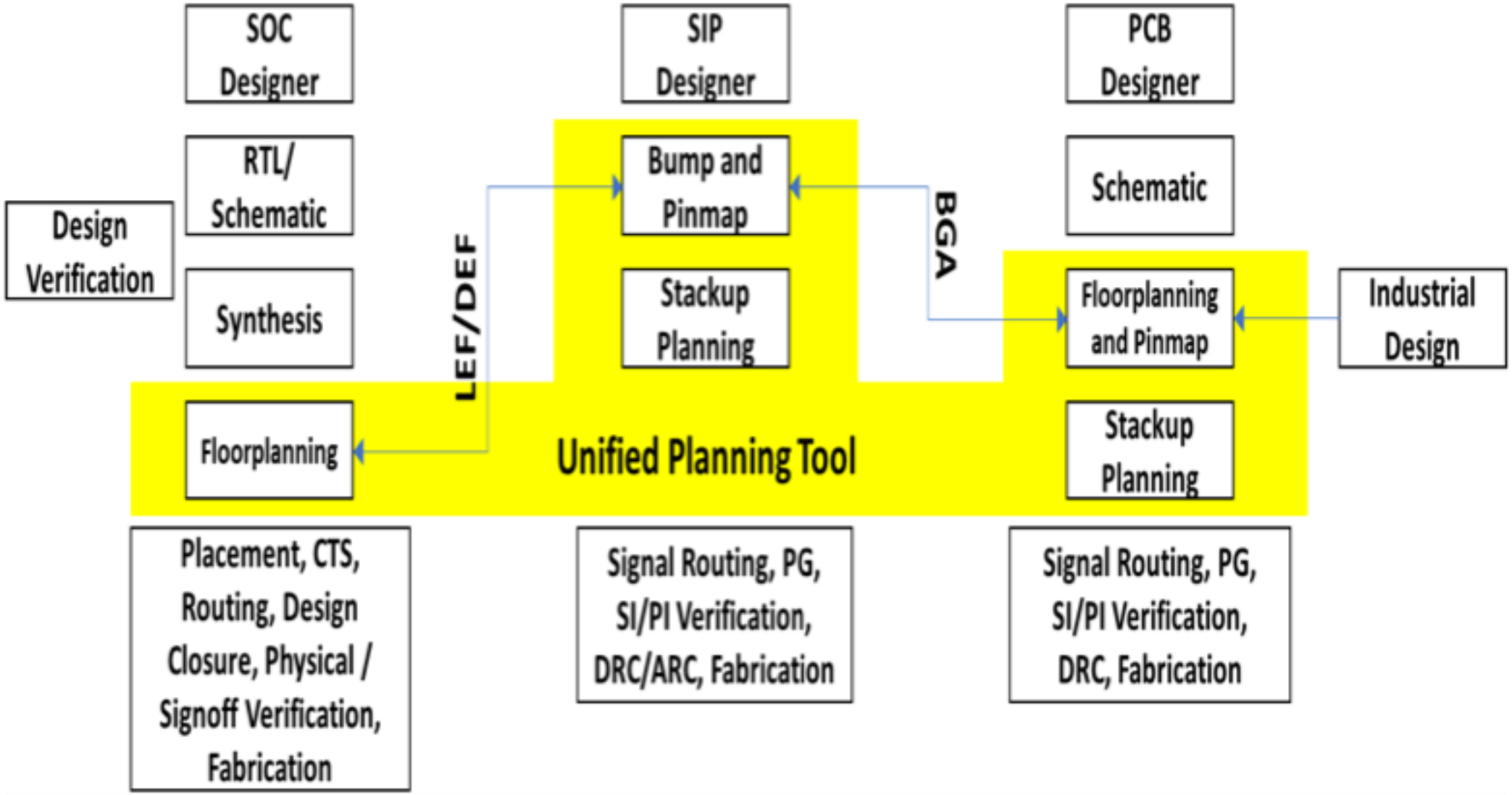
- **IC, Package, Board tools: different worlds** (R&D $, usage, …)
  - Many cross-domain integrations: power delivery, timing signoff, cost …
- **High-quality system partitioning, layout; Physics signoff with less margin**
- **HI co-design adds system, architecture** to layout, signoff
  - System designers are not like layout engineers ! → DSE, Pathfinding, Architecture
- **Specification, synthesis, verification at system / HI level**
  - Mix-and-match of chiplets to meet system spec
  - What chiplets should be available for co-integration?
- **Encapsulation of co-integration technology usable by design tools**
- **“Interoperability wrapper”** for co-integration within package

# 10000x Gaps: Not Business As Usual !



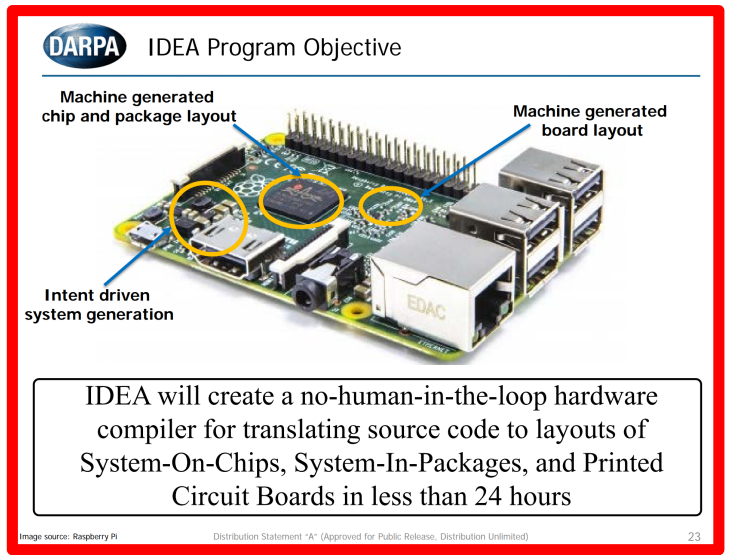

- **"Multiphysics Checker" has 10000x Speed Gap vs. "Optimizable Objective Function"**
- **Analysis ≠ Optimization**
- **Accuracy ≠ Fidelity**
- **Machine Learning (predictors, optimizable models) likely essential !!!**

# Must Have Starting Die-Pkg-PCB “Backplane”



# Lessons From the Past

- Fact: Non-emergence of commercial EDA solutions for Co-Design
  - Long history of multi-die integration: MCM, SIP, 3-D / 2.5-D, …
  - Small available market mismatches commercial EDA model
  - Lots of chicken-egg interlocks ☹
- Fact: Co-design tools and flows exist today (evidenced by products)
  - Based on existing IC design standards, tool flows, methodologies
  - **Which system products will drive fundamentally new co-design tools/methods?**
- **Could a free, open-source software (FOSS) ecosystem address HI co-design tool needs?**
- **See: DARPA "IDEA" (!)**




Andreas Olofsson, DARPA
ISPD-2018 keynote
March 27, 2018
http://www.ispd.cc/slides/2018/k2.pdf

# Avi's Three Questions

- **What is the state of the art in co-design?**
  - **Unchanged preoccupation with analyses/checkers: +physics, +scale, +layers**
  - **Lack of pathfinding, DSE, optimization has unknown cost**
  - **But, harms design schedule, system optimizations (cross-layer, multiphysics, …)**
- **What are the key challenges that need to be overcome?**
  - **Lack of well-defined co-design enablements, methodologies, automation of design synthesis and optimization**
- **What needs to happen for these challenges to be overcome?**
  - **Not sure…. !**
  - **History: heart attacks ("Pentium bug", ASP competition) drive tool R&D investment**
  - **Tool specs, benchmarks, problem statements can be enormously influential !**
  - **Invest in machine learning (e.g., to span 10000x gaps in optimization vs. analysis)**
  - **Enable open-source tool ecosystem for co-design optimization (= outsource beyond commercial tool providers)**