High Level Design & ESL

How design cost is driving innovation in system-level designs?

Rajesh Gupta
University of California, San Diego

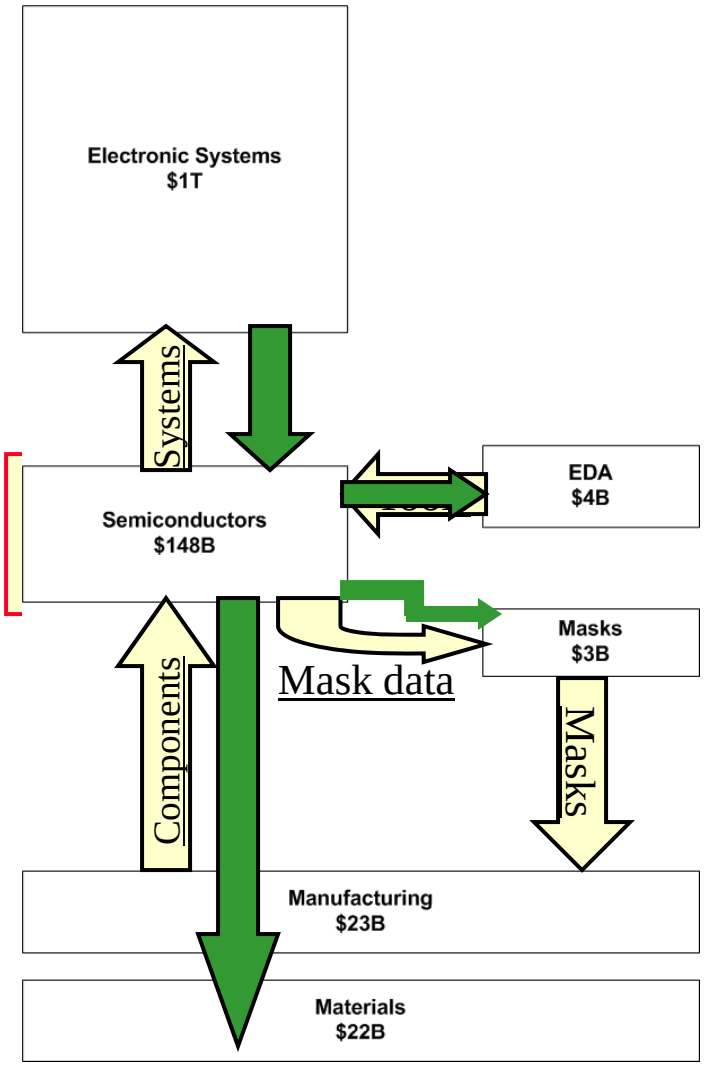
FMCAD, Portland, Nov. 17, 2008

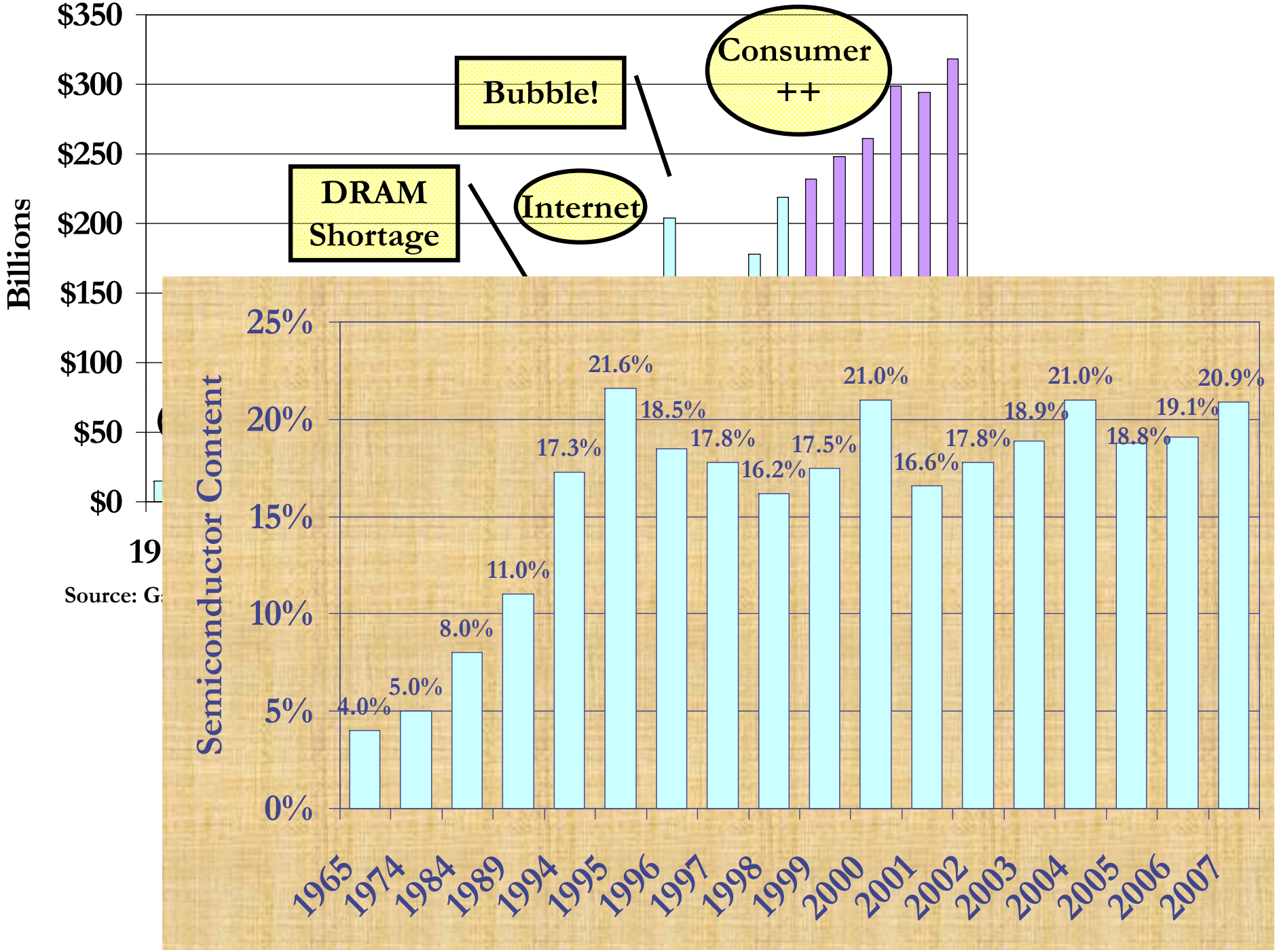
MESL . UCSD . EDU

My main point

- At various time VLSI design has been driven by
  - Area, timing, power, reliability, manufacturing variability
- Cost of design is likely to be the driver for future innovations in how we architect, design and implement future ICs in each of these areas:
  - Tools, Methods
  - Architectures
  - Programming models and methods

The Technology and Its Industry

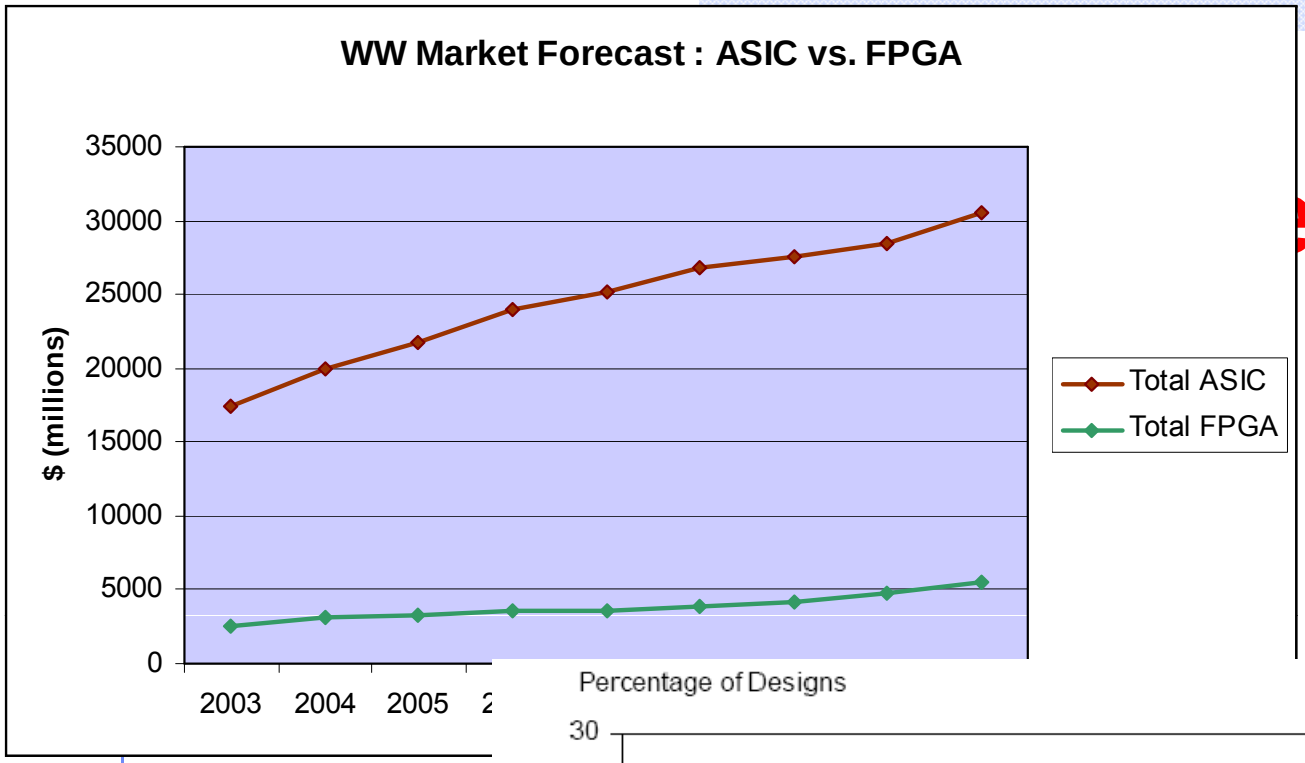



Billions
$350
$300
$250
$200
$150
$100
$50
$0
Bubble!
Consumer ++
DRAM Shortage
Internet
Source: G
Semiconductor Content
25%
20%
15%
10%
5%
0%
4.0%
5.0%
8.0%
11.0%
17.3%
21.6%
18.5%
17.8%
16.2%
17.5%
21.0%
16.6%
17.8%
18.9%
21.0%
18.8%
19.1%
20.9%
1965
1974
1984
1989
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007

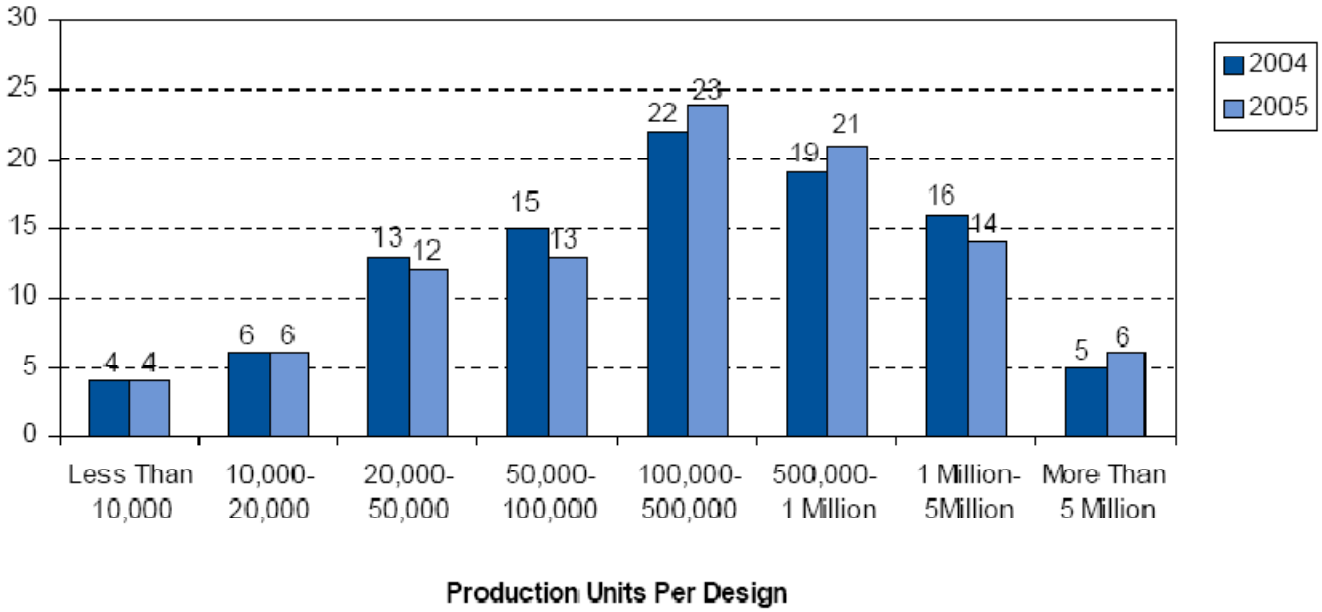
More Silicon to More Boxes…

- Of the 72 distinct application markets that rely on value added IC designs (ASIC, ASSP, FPGA, SOC)
  - over 50% are less than $500M, 75% are less than $1B
- The rising fabless, fablite
  - The US has 56% of over 1K design houses…
  - …and accounts for 76% of industry revenues
  - (Wireless 27%, networking 25%, consumer 20%)
- Cost is increasingly the driver for fabless
  - Only 17% of designs above 500 MHz
    - 67% of ASIC designs are 299 MHz and lower
  - Sizes pretty much evenly distributed from 100K to 5M gates

Source: IBS



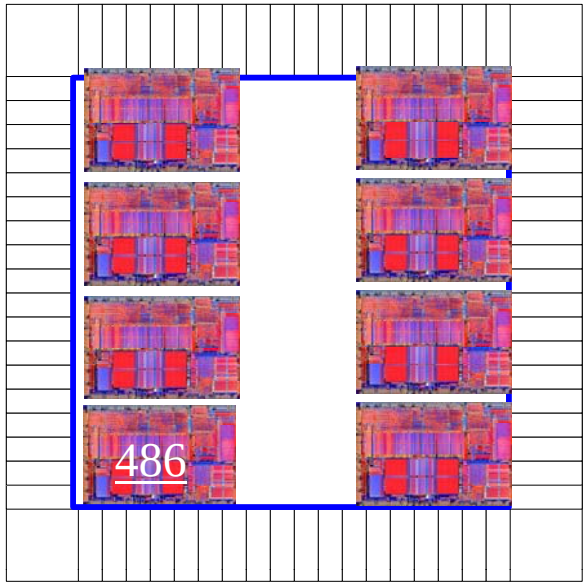
Source: Gartner Dataquest "ASIC an



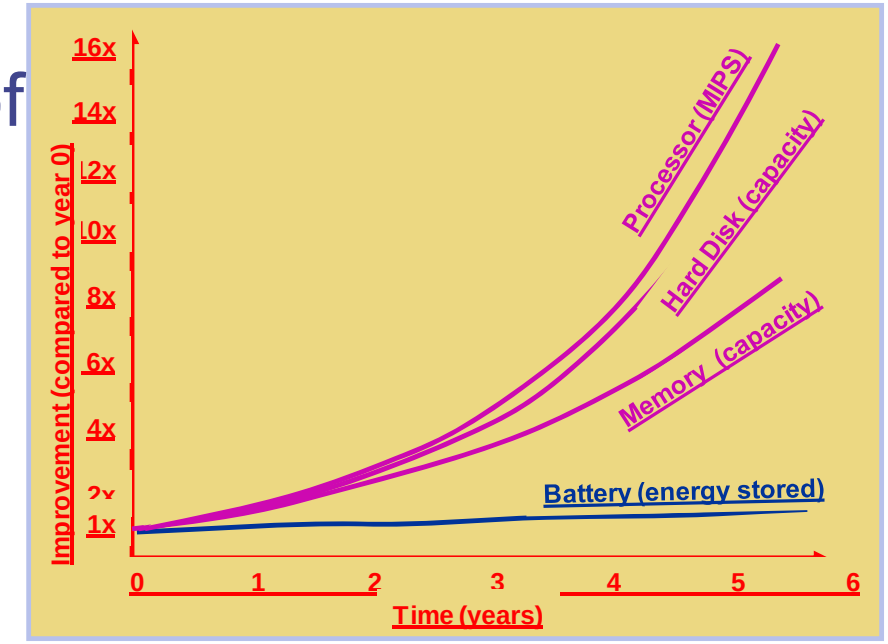
Source: Gartner Dataquest (August 2006)

More & Moore

Pad limited die:
200 pins
52 mm^2



- Most things in real-life do not scale anywhere close to this
  - Battery energy, power sources
  - Size, Space, Spectrum
  - Design time.
- Dealing with the effects of Moore
  - "Embedded Systems"



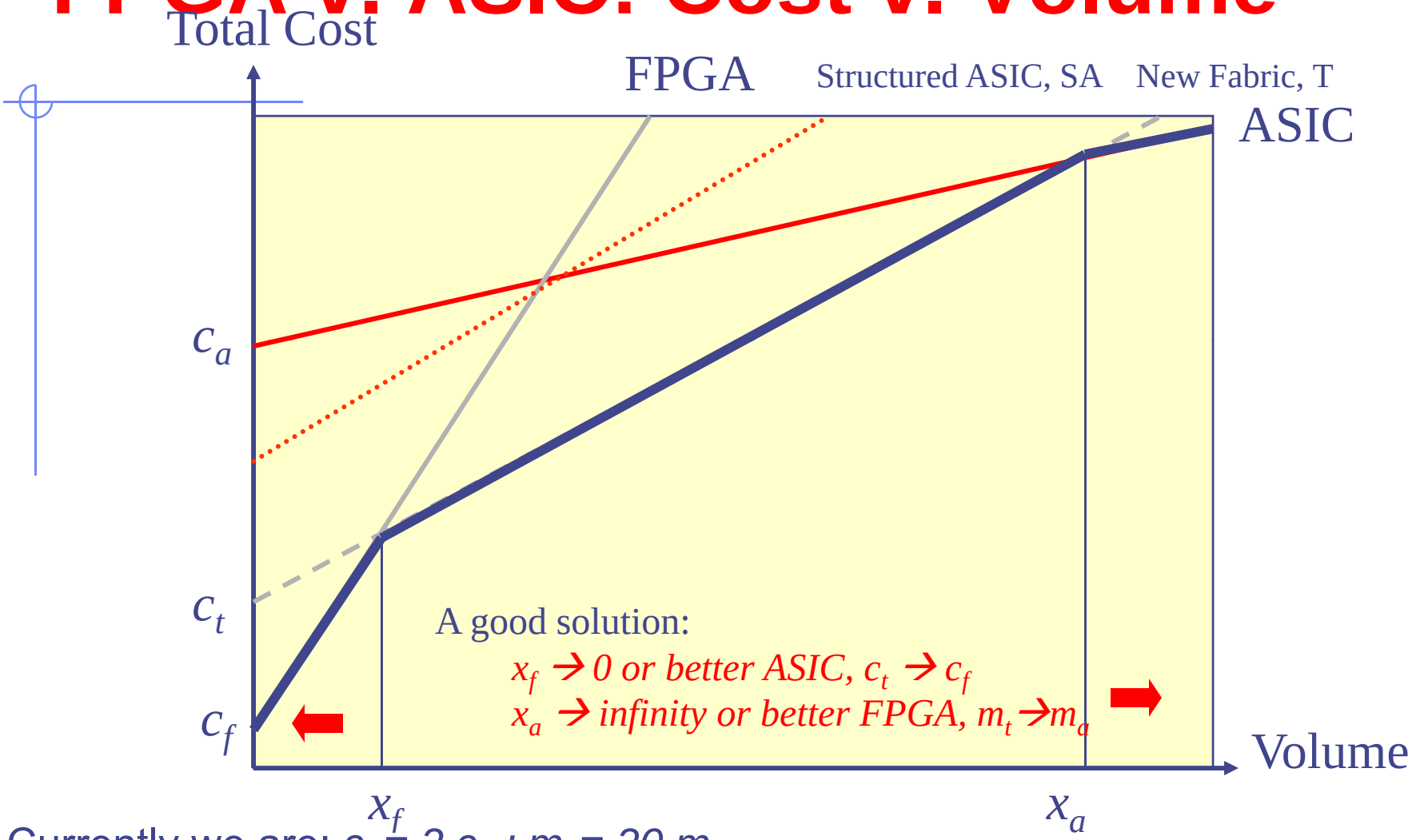
A Tale of Two Consequence

1. EDA: Raise abstractions
   - Raising abstraction has always been part of the solution strategy to lower design costs.
   - In design modeling, design synthesis, design verification

2. Architecture: Raise programmability
   - Holy Grail: ASIC efficiency with CPU programmability.
   - The tremendous space of architectural innovations between ASIC and FPGA

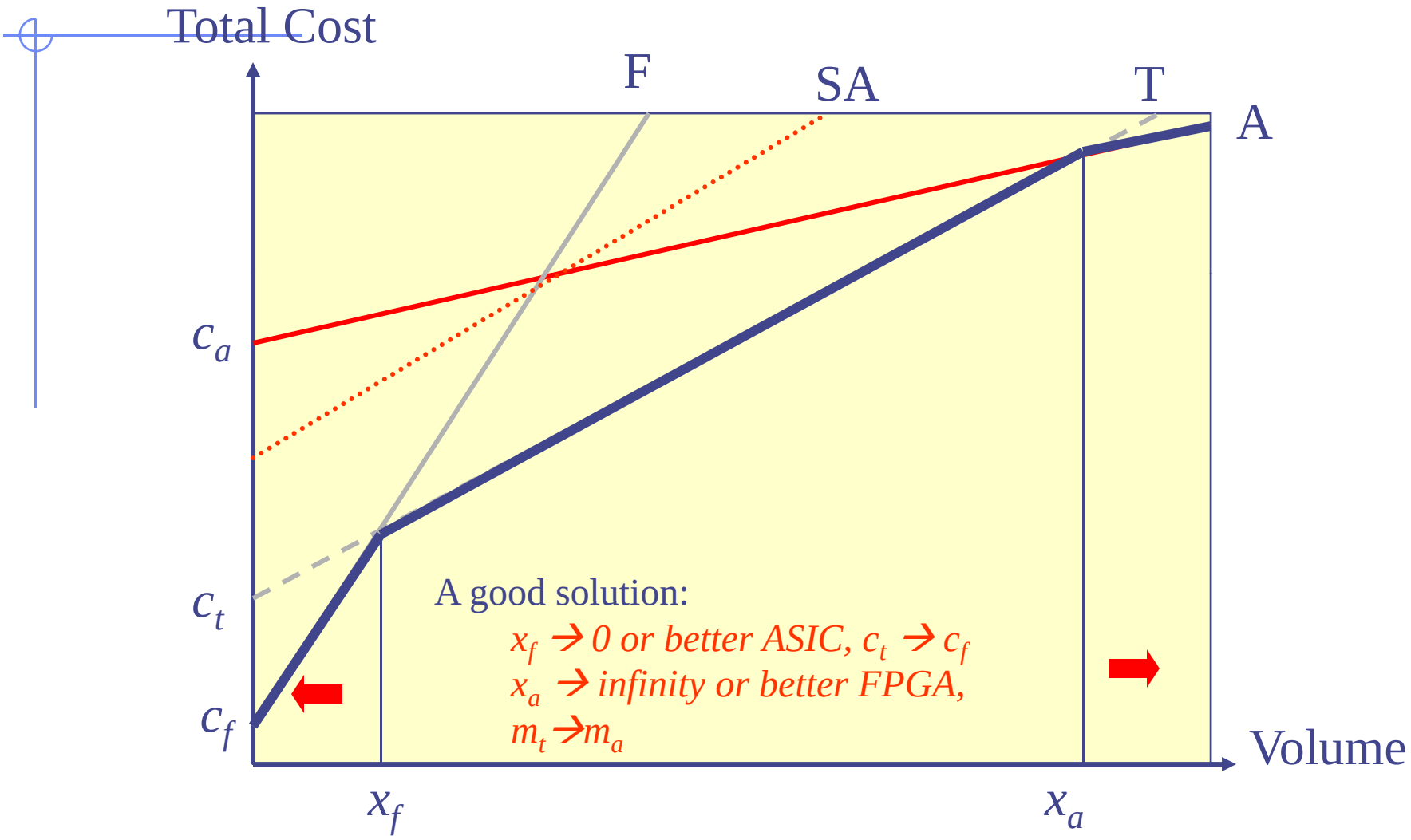
► Let us take a look at the two sides from a familiar perspective

FPGA v. ASIC: Cost v. Volume

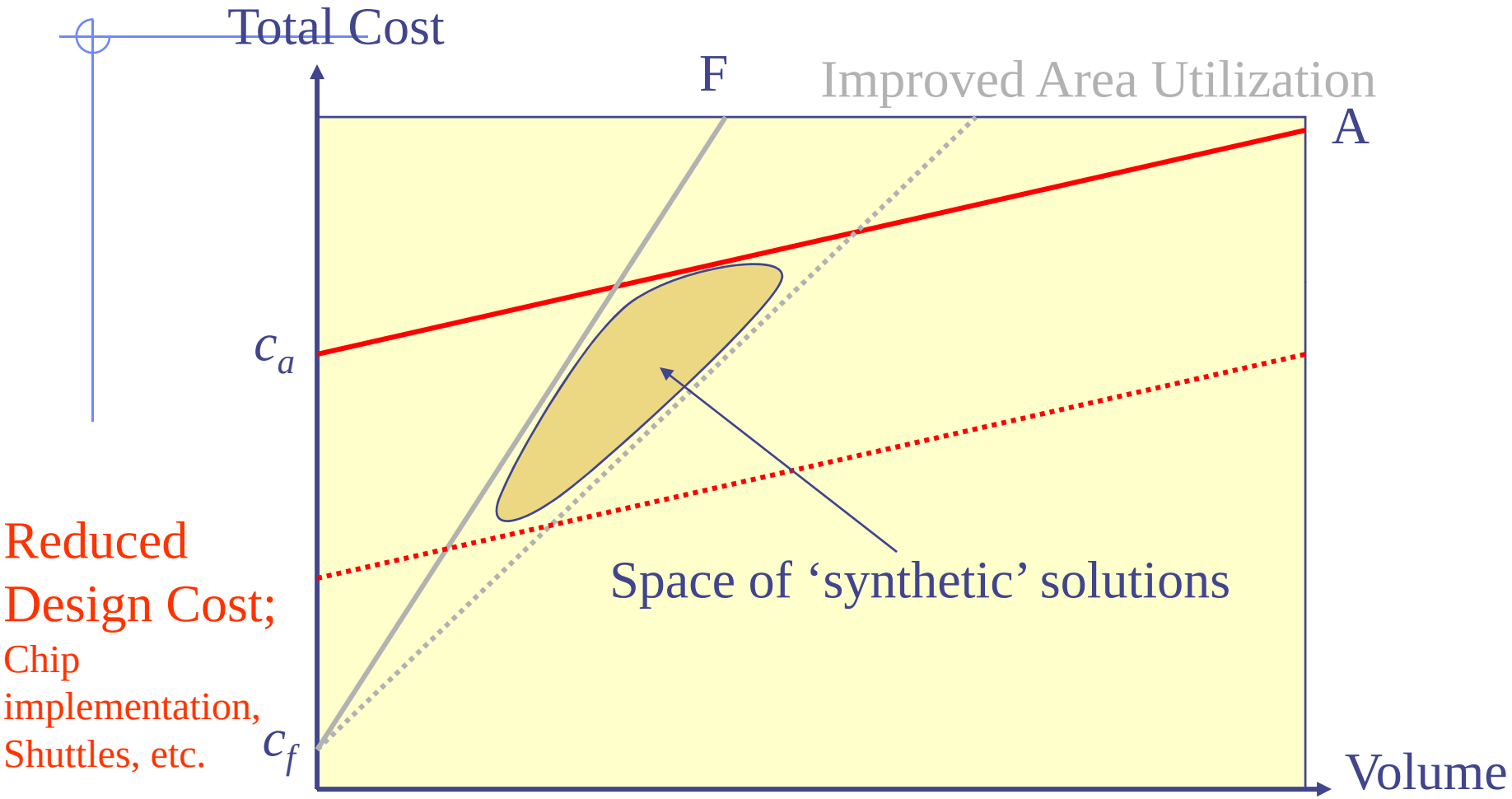


- Currently we are: $c_f = 2\ c_a$; $m_f = 20\ m_a$
  - Fixed cost of FPGA design = 2 * ASIC design costs
  - Per part cost of FPGAs rises 20x cost of ASIC.
  - Current crossover point at 100K units.

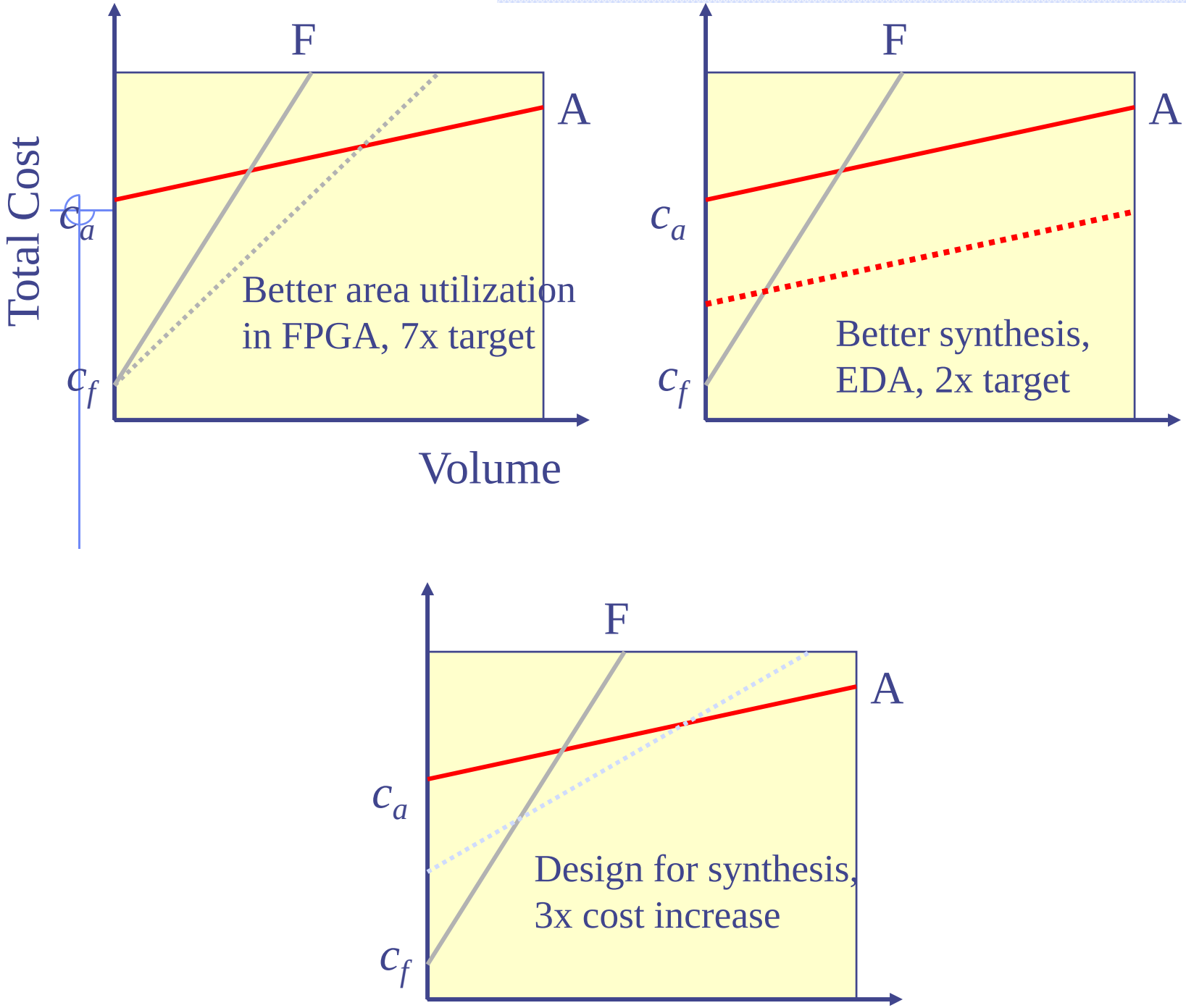
ASIC/FPGA Tradeoff



Better ASIC or Better FPGA?



Reduced Design Cost;
Chip implementation, Shuttles, etc.


Total Cost
F
A
c_a
c_f
Better area utilization
in FPGA, 7x target
Volume
F
A
c_a
c_f
Better synthesis,
EDA, 2x target
F
A
c_a
c_f
Design for synthesis,
3x cost increase

Technical Dimensions of the Problem

- SE: Silicon Efficiency
  - Inherently better circuit implementation styles, levels, logic: Asynchronous, GALS
- AE: Architectural Efficiency
  - Inherently improved application-level performance or performance independent of mapping methods
- PA: Programmer Accessibility
  - Use existing programming models/methods to ensure IP availability and integration.
- DP: Designer Productivity

ITRS, last updated 2006

Designer Productivity is Challenge #1

Verification

Predictable Implementation

Embedded SW

Distributed design, AMS

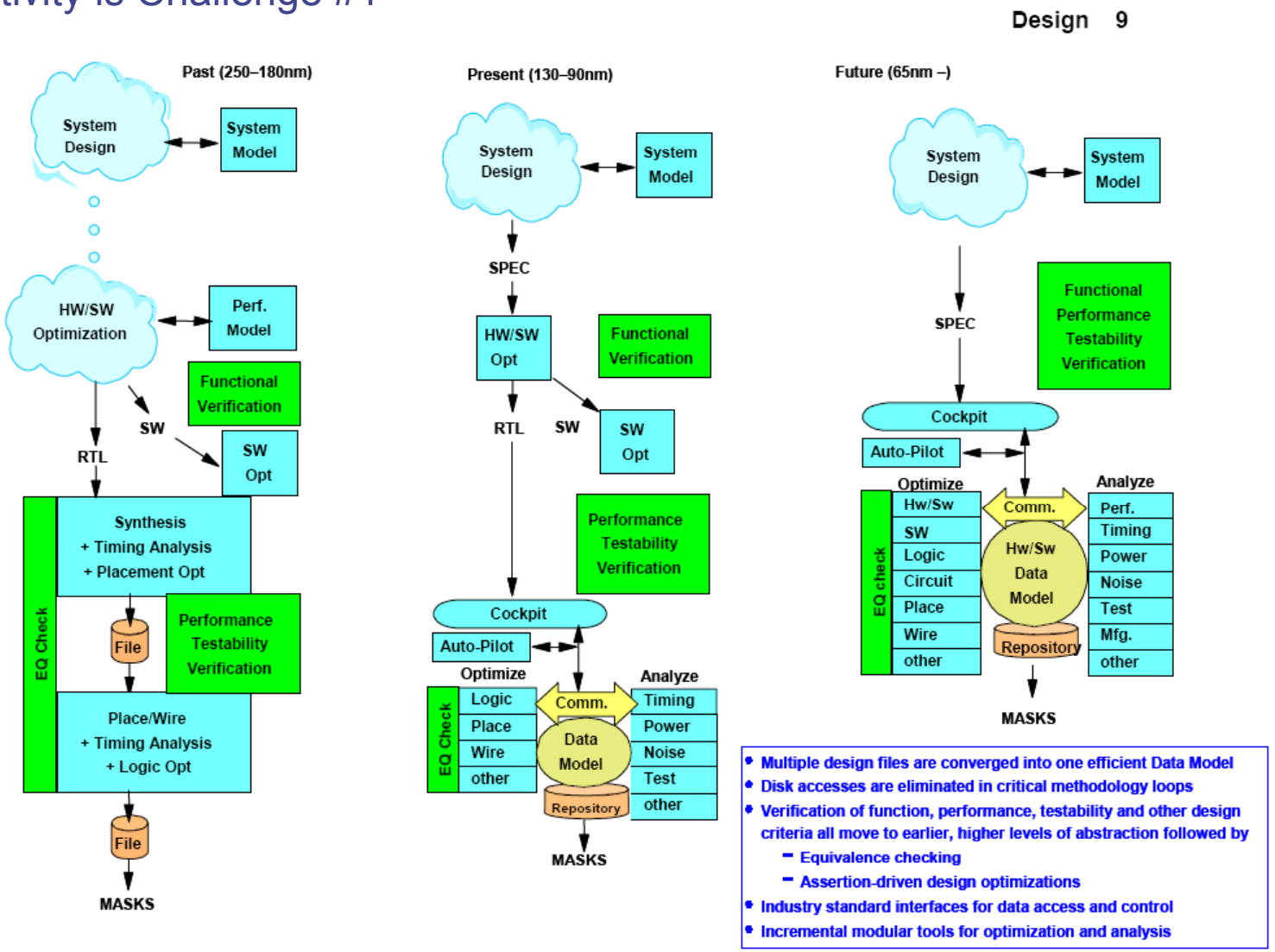


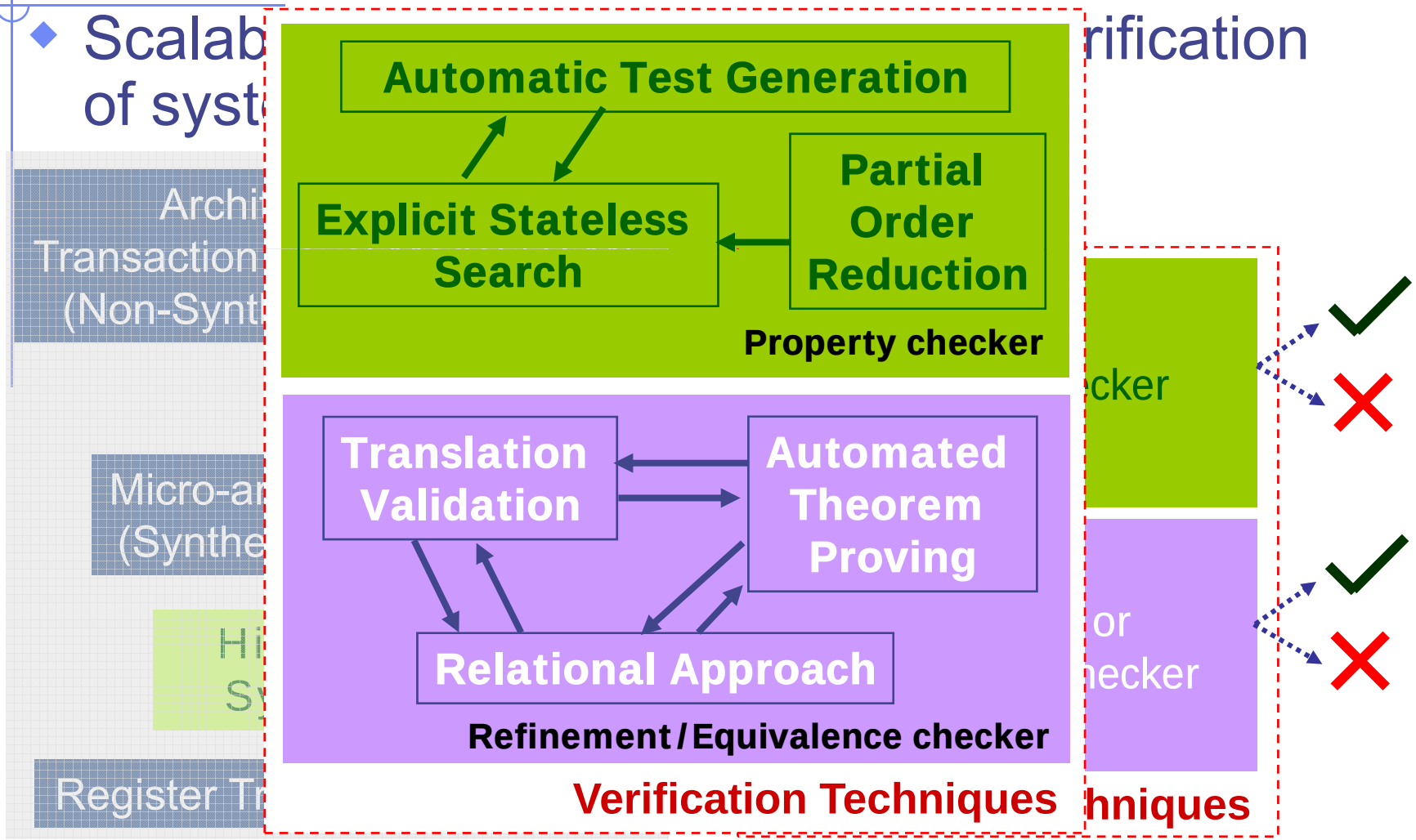
Figure 17 Required Evolution of Design System Architecture

Impact on Designer Productivity

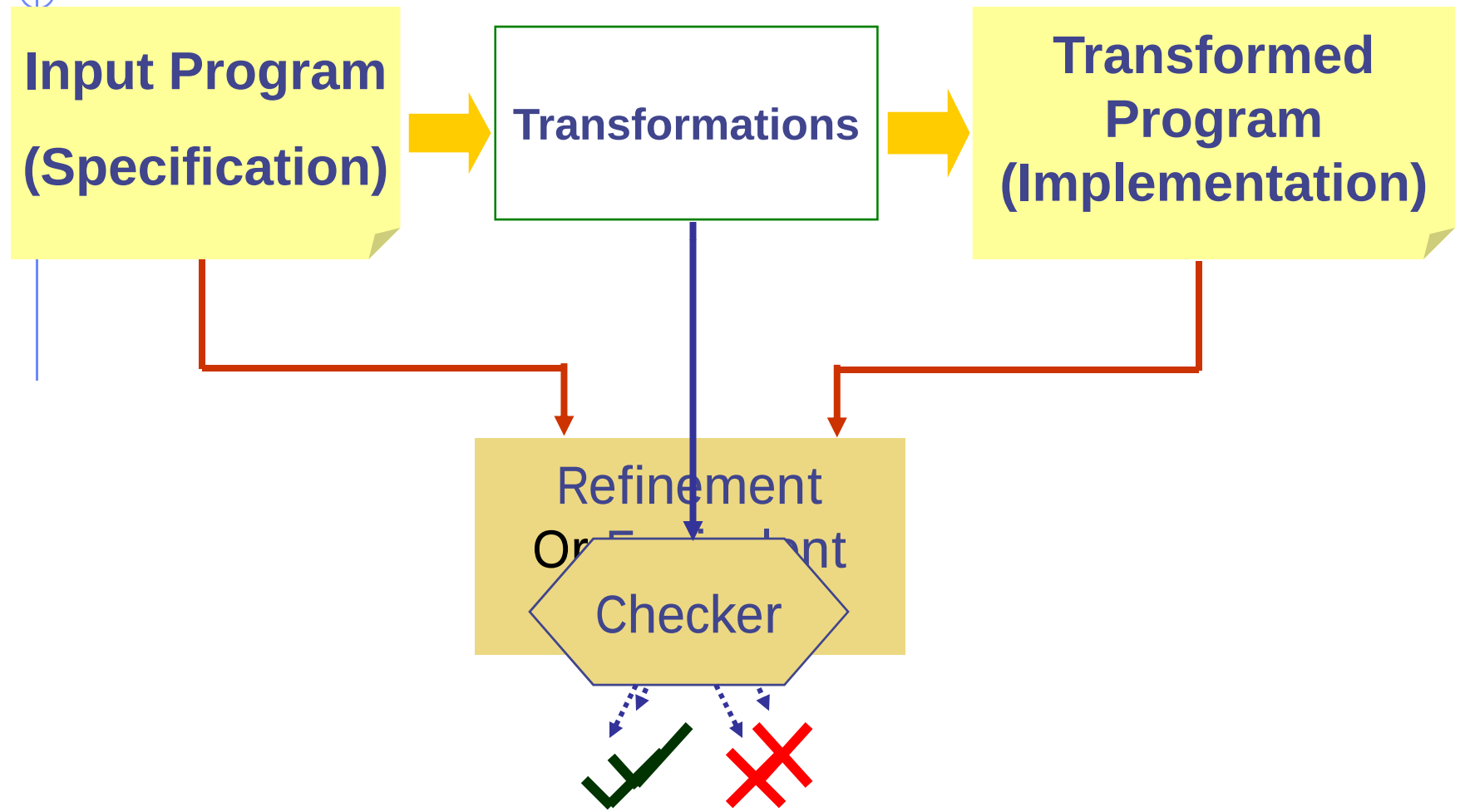
Design Technology	Year	Productivity Delta gates/DY	Comments
Physical Design (APR)	1993	38.9% 5.55K	PD integration
Tall-thin Engineer	1995	63.6% 9.1K	Chip/circuit/PD/Verif.
Small block reuse	1997	340% 40K	2.5K-75K gates
Large block reuse	1999	38.9% 56K	75K-1M gates
IC implementation suits	2001	63.6% 91K	RTL-GDSII integration
RTL functional verification	2003	37.5% 125K	SW development verif.
ES Methodology	2005	60% 200K	Behavioral above RTL
Very large block reuse	2007	200% 600K	>1M gates, IP cores
Homogenous parallel processing	2009	100-200% 1.2M	Many identical cores around a main processor
Intelligent test bench	2011	37.5%2.4M	Automation of verification partitioning
Concurrent SW compiler	2013	60% 3.3M	Enables SW in parallel SOCs
Heterogenous massive parallel processing	2015	100-200% 5.3M	Specialized cores around a main processor
System-level DA and executable specs	2017-19	100-200% 10.5M	On/off-chip integration of functions.
Total		264,000%	

Raising Verification

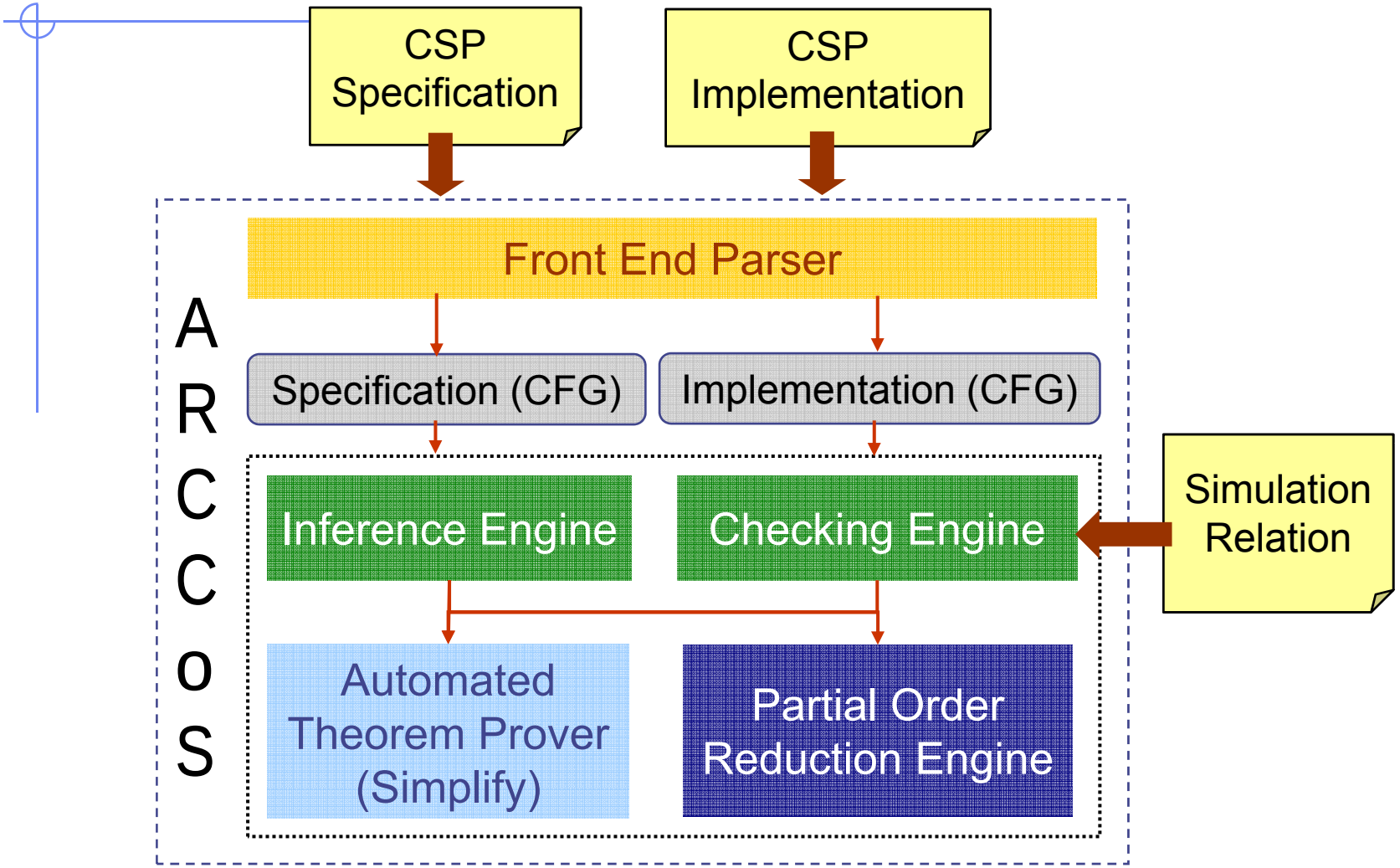
- Scalab … rification of syst…



Refinement Checking



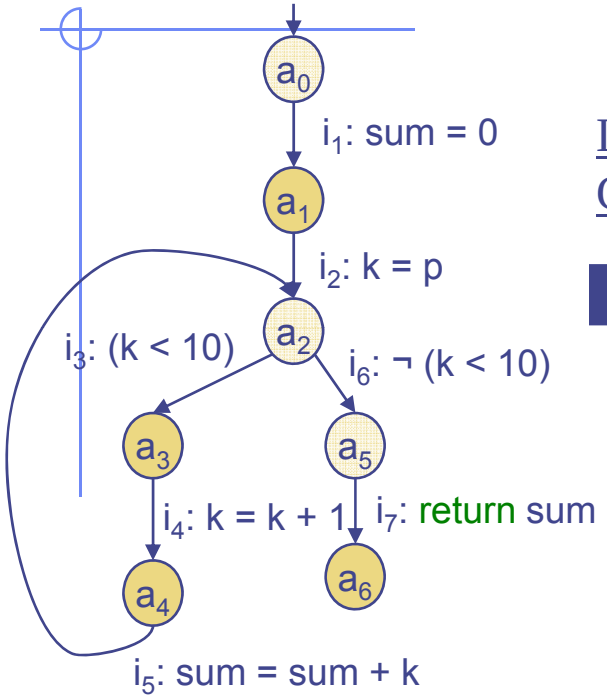
Prototype Implementation - ARCCoS



Results from ARCCoS

Descriptions	#Process			Time (no PO) (min:sec)	Time (PO) (min:sec)
	Spec	Impl	Total		
Simple buffer	3	4	7	00:00	00:00
Simple vending machine	1	1	2	00:00	00:00
Cyclic scheduler	3	3	6	01:01	00:49
College student tracking system	1	2	3	00:01	00:01
Single communication link	3	8	11	00:01	00:01
2 parallel communication links	6	12	18	01:28	00:04
3 parallel communication links	9	16	25	514:52	00:21
4 parallel communication links	12	20	32	DNT	01:11
5 parallel communication links	15	24	39	DNT	02:32
6 parallel communication links	18	28	46	DNT	08:29
7 parallel communication links	21	32	53	DNT	37:28
Hardware refinement	3	5	8	00:00	00:00
EP2 System	1	2	3	01:51	01:47

Example

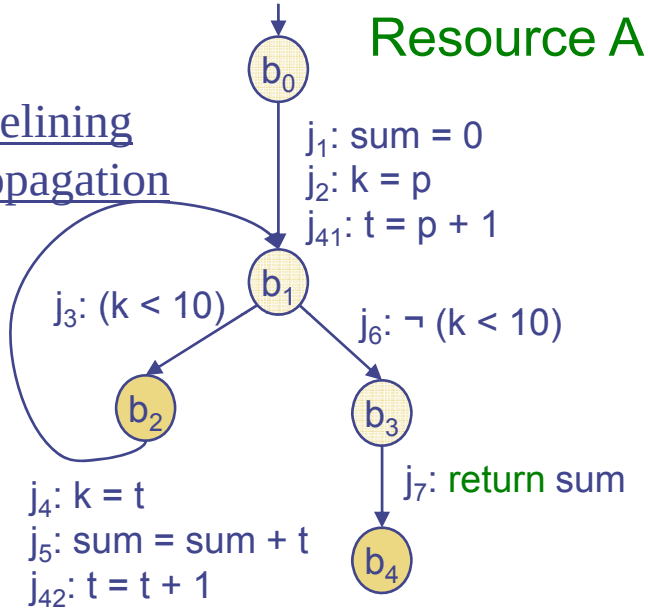


(a) Specification

$$sum = \sum_{p+1}^{10} i$$

Loop pipelining
Copy propagation

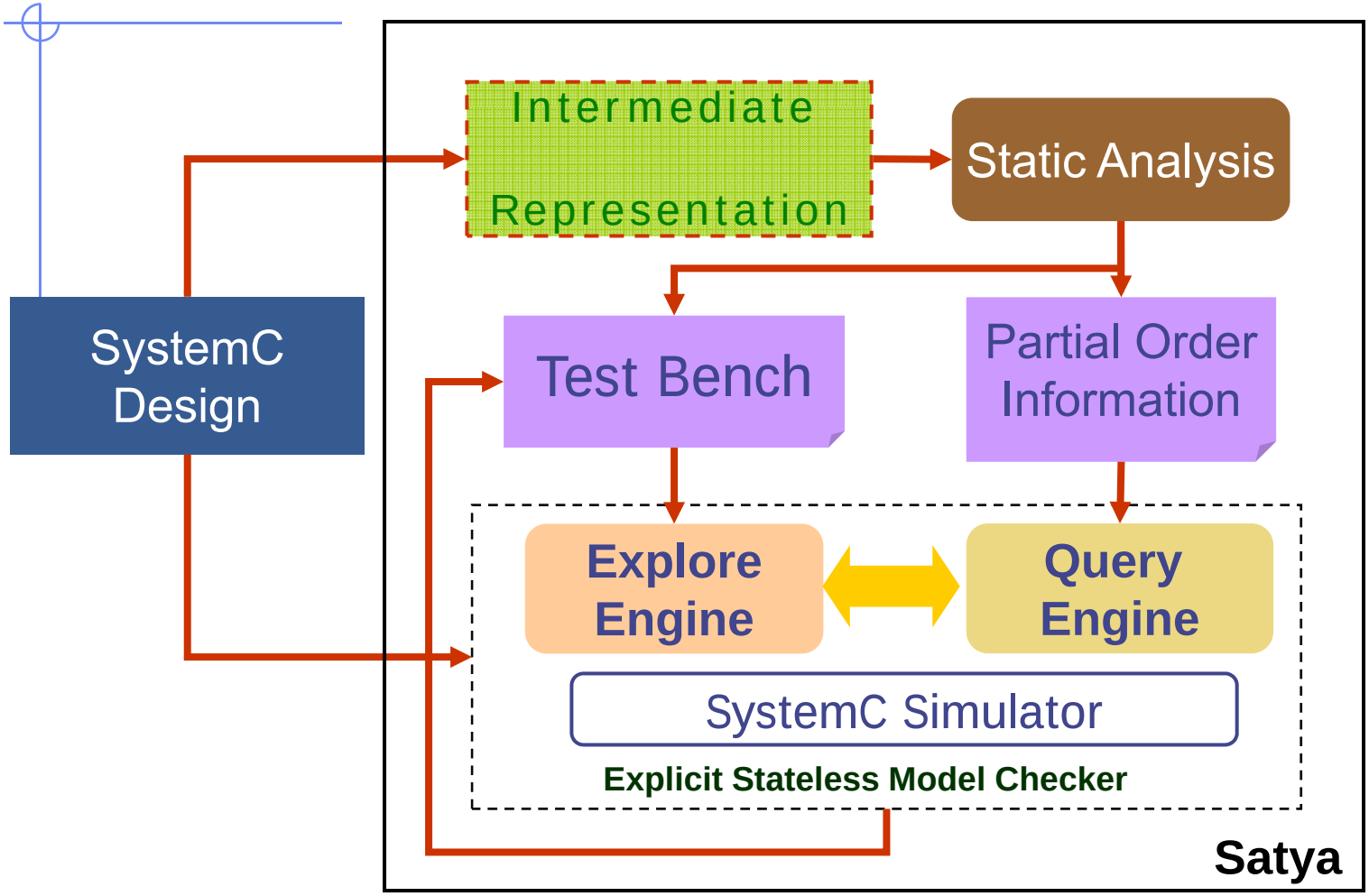
Resource Allocation: + + <



(b) Implementation

(I_1, I_2)	1st Pass	2nd Pass
1. (a_0, b_0)	$p_s = p_i$	$p_s = p_i$
2. (a_2, b_1)	$k_s = k_i$	$k_s = k_i \wedge sum_s = sum_i \wedge (k_s + 1) = t_i$
3. (a_5, b_3)	$sum_s = sum_i$	$sum_s = sum_i$

On going work



Closing Thoughts

- ASIC design cost is the new driver
  - Solution space is expanded to include not only tools but also architectures
- A time for tremendous creativity