Unifying Multiple Tools to Achieve High Performance SoC Design

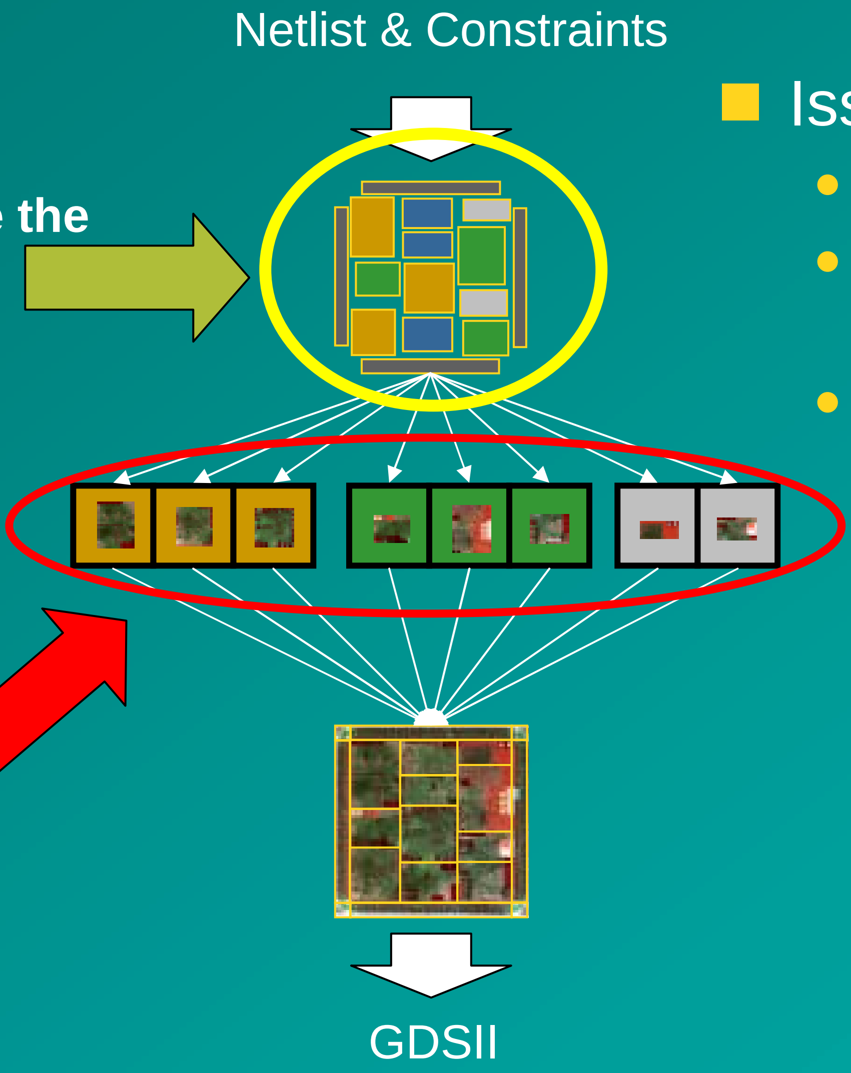
Mark Bales

ReShape, Inc.

April 26, 2004

ReShape

Physical Design Paradox



Chip-level decisions have the biggest impact

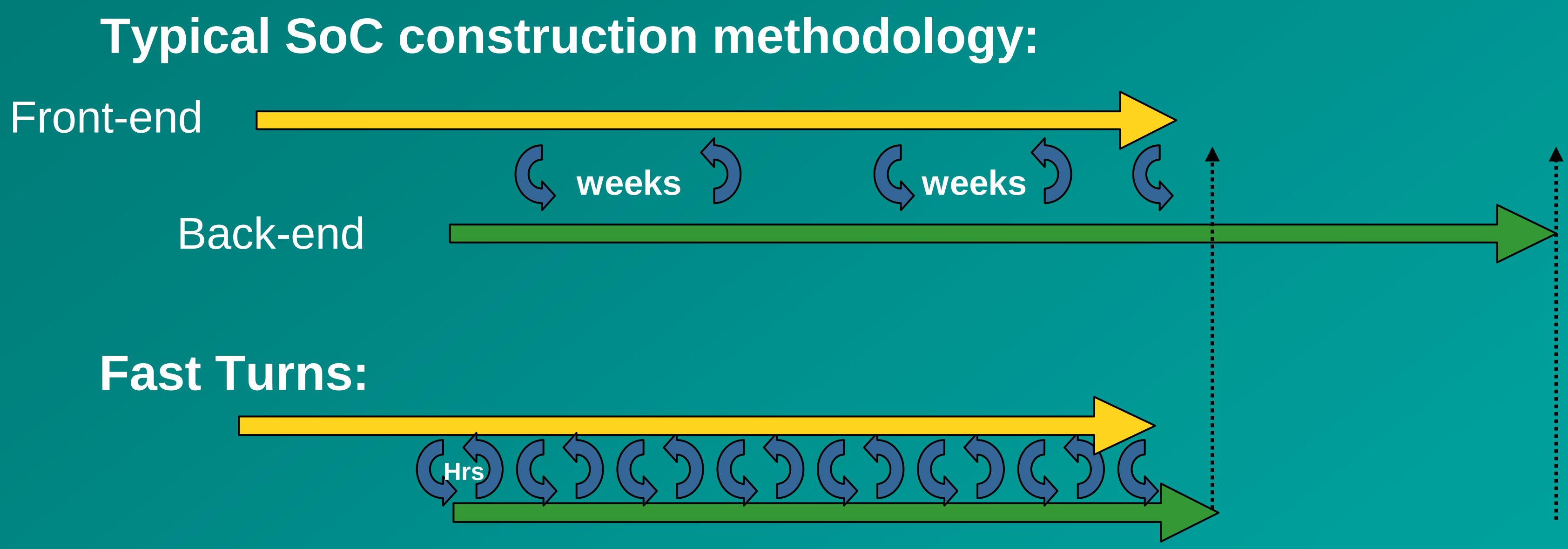
80% of time & effort is spent at the block-level

- Issues:
  - Iterations are slow: weeks
  - Full chip not available until late in the design cycle
  - Changes at top level highly disruptive

Problem: Scripting Mundane & Complex

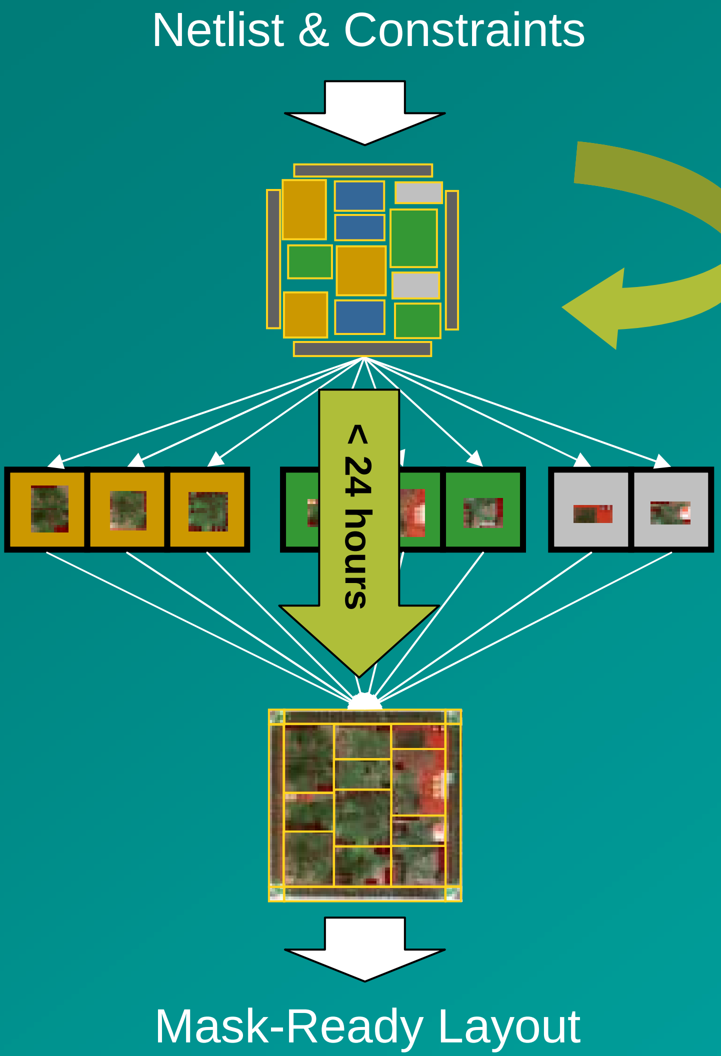
- **Physical design intent captured in scripts**
  - 20 year old design paradigm
  - 100,000's of lines required to specify a design
- **Scripts consume ~50% physical design effort**
  - Manual data preparation
  - Monitoring script progress
  - Recovering from scripting errors and tool failures
  - Managing machine and license availability
  - Determining next steps after the script completes
  - Optimizing the script for the specific design
- Scripts are inherently brittle
  - Binding: floorplan, netlist, library, tool versions, tool settings
  - Debug and maintenance is a chore

Ideal: Fast Concurrent Design



- Fast design exploration
- Optimize RTL and physical design in parallel
- Uncover full-chip issues early
- Yields schedule predictability

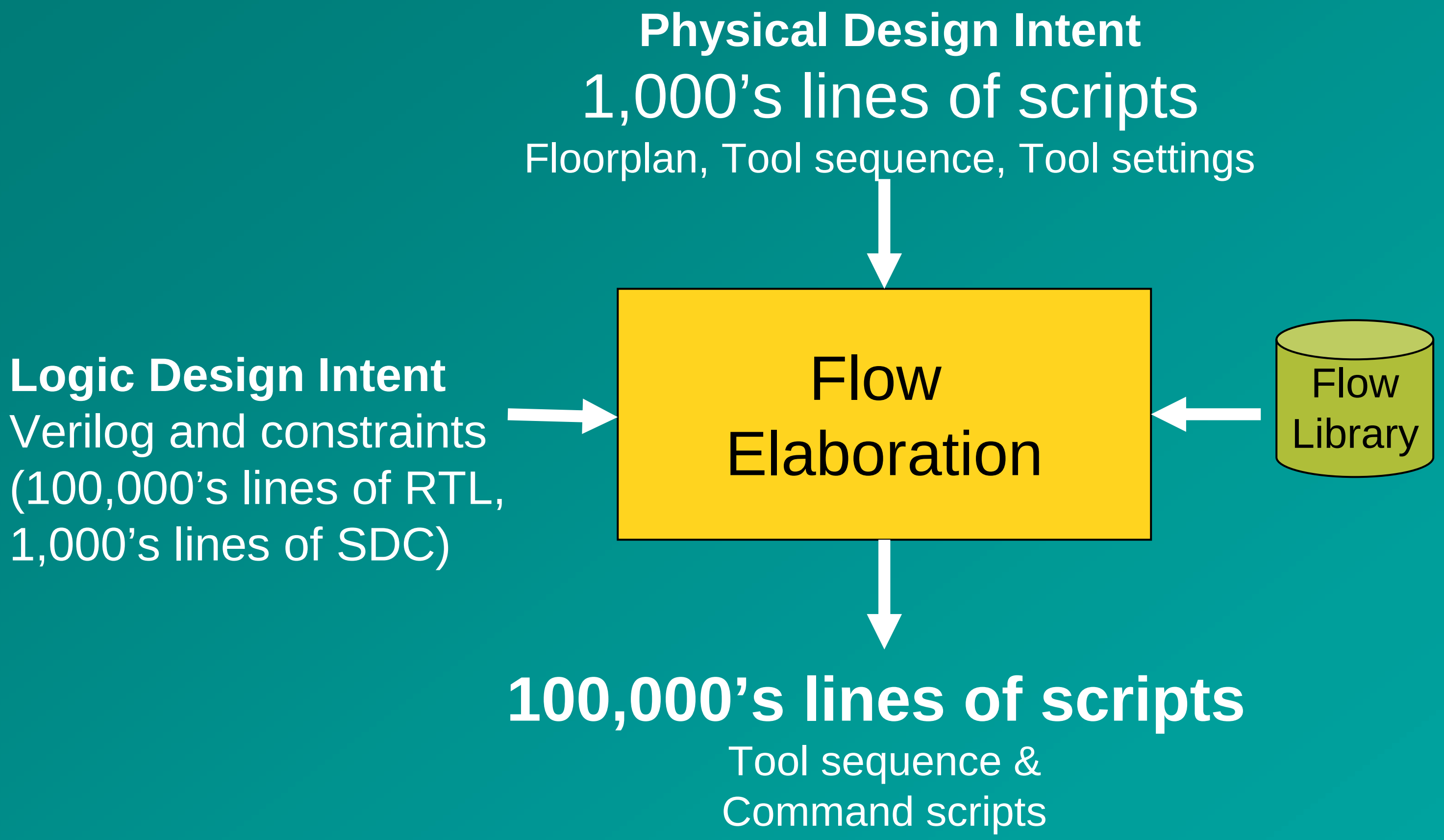
Chip-Level Design Automation



- **Automatic Builds**
  - *Fast* design exploration: 24 hours from netlist to mask layout
  - Faster / more predictable tapeouts: 3 week tapeouts
  - More predictable schedules: 100% first time success
- **Chip-Level Optimization**
  - Smaller chips: 15% average reduction in die area
  - Faster chips: 30% average reduction in global wire length

ReShape

Key Technology: Flow Elaboration



Physical Design Intent

1,000’s lines of scripts

Floorplan, Tool sequence, Tool settings

Logic Design Intent

Verilog and constraints
(100,000’s lines of RTL,
1,000’s lines of SDC)

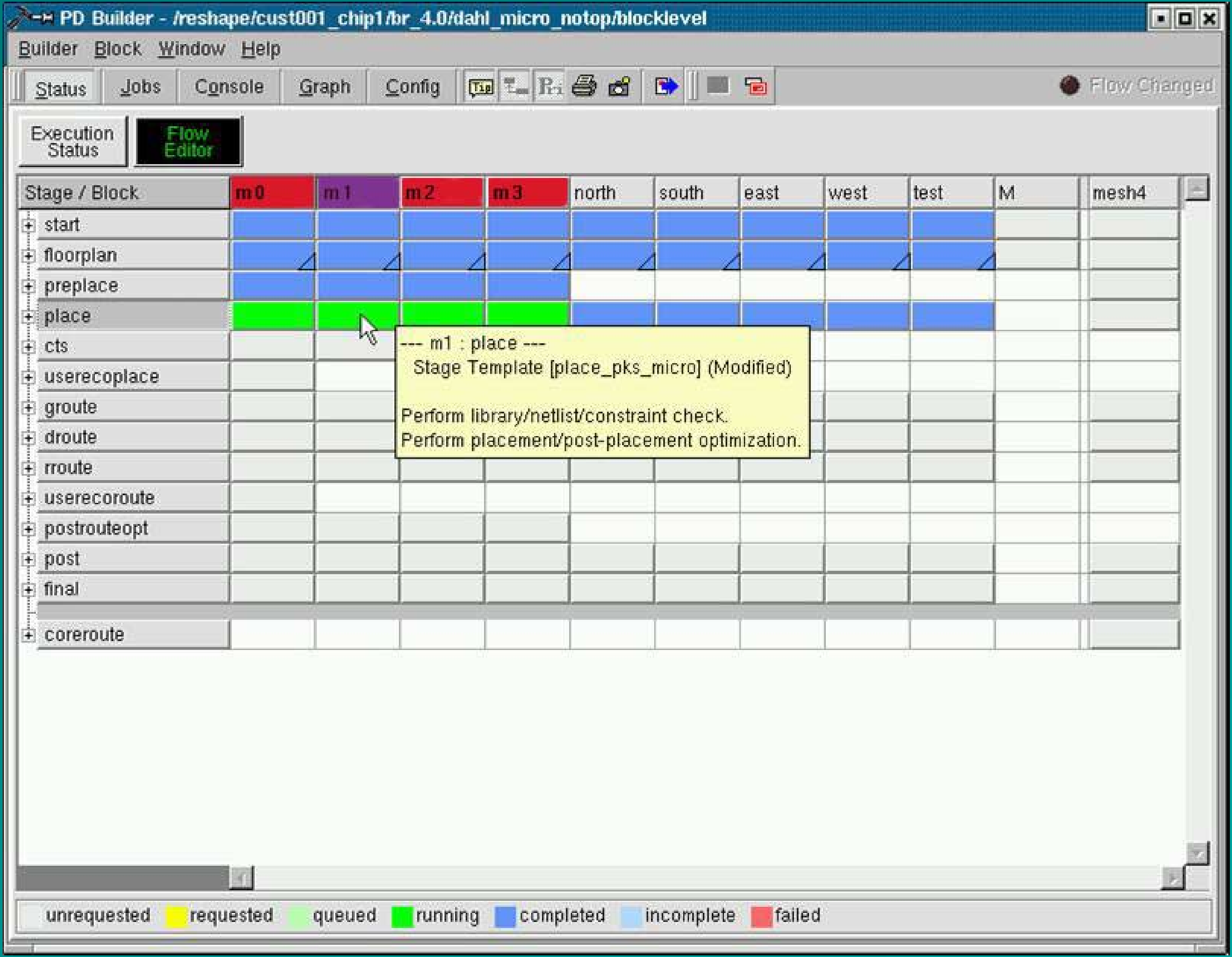
Flow Elaboration

Flow Library

100,000’s lines of scripts

Tool sequence &
Command scripts

PD Builder™ – Automated Block Builds



Stage / Block	m0	m1	m2	m3	north	south	east	west	test	M	mesh4
start	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦		
floorplan	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦		
preplace	🟦	🟦	🟦	🟦							
place	🟩	🟩	🟩	🟩	🟦	🟦	🟦	🟦	🟦		
cts											
userecoplace											
groute											
droute											
rroute											
userecoroute											
postrouteopt											
post											
final											
coreroute											

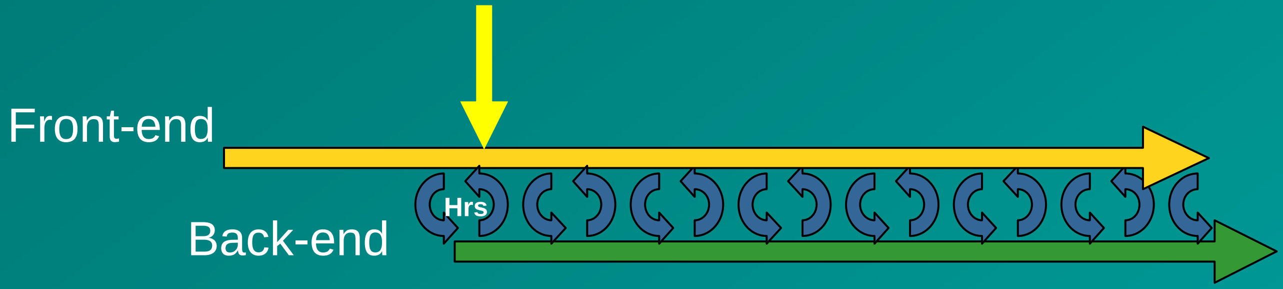
--- m1 : place ---
Stage Template [place_pks_micro] (Modified)

Perform library/netlist/constraint check.
Perform placement/post-placement optimization.

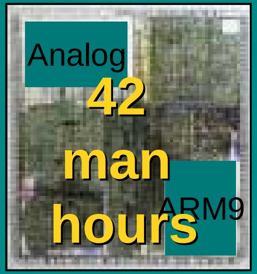
unrequested 🟨 requested queued 🟩 running 🟦 completed incomplete 🟥 failed

Results

10X Productivity: First Full-Chip Build

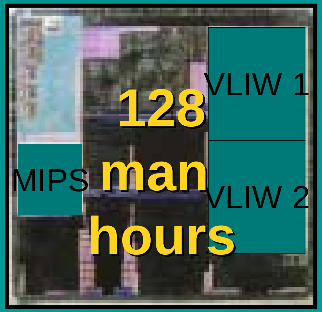


- **First full-chip build**
  - **Scripted flow takes 5 engineers at least 42 days**
  - **ReShape one engineer < 17 days**



5 Blocks
720K instances
0.18u, 9x8mm

5 Blocks
920K instances
0.13u, 6x7mm



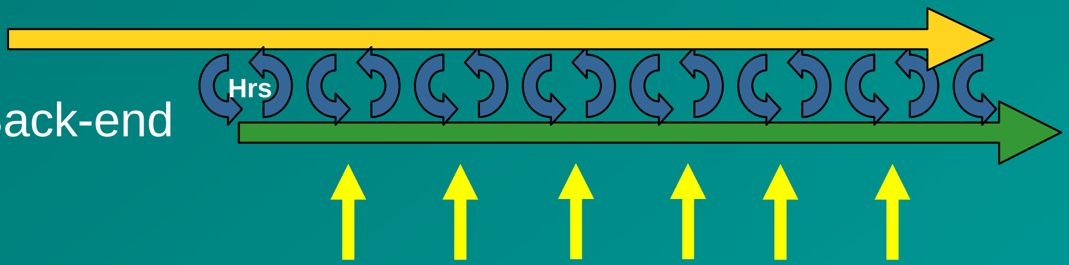
12 Blocks
1,500K instances
0.13u, 10x10mm

98 man hours

9 Blocks
1,220K instances
0.13u, 12x12mm

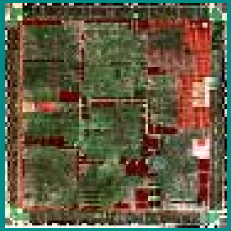
ReShape

40X Productivity: Chip Rebuilds



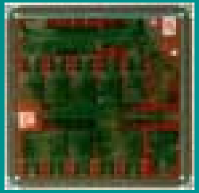
- **Fast design iterations**
  - **Scripted flow: 8 days X 5 engineers**
  - **ReShape: <24 hours, one engineer**

18 Hours



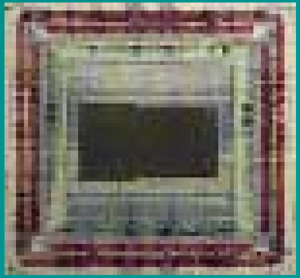
7.5M Gates
14 blocks
11 x 12 mm

11 Hours



5.5 M gates
13 blocks
8 x 8 mm

18 Hours



1200 sig. pins
2600 pwr/grd.
14 X 14 mm

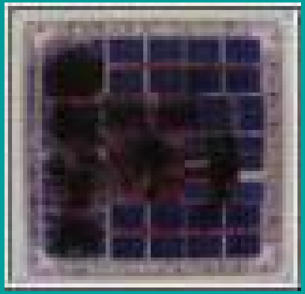
8 Hours

5 blocks
5 X 6 mm

9 Hours

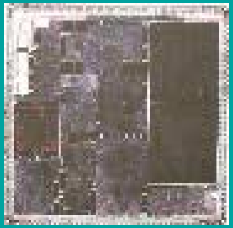
3 blocks
ARM core
7 X 7 mm

23 Hours



5M gates
15 blocks
16 X 16 mm

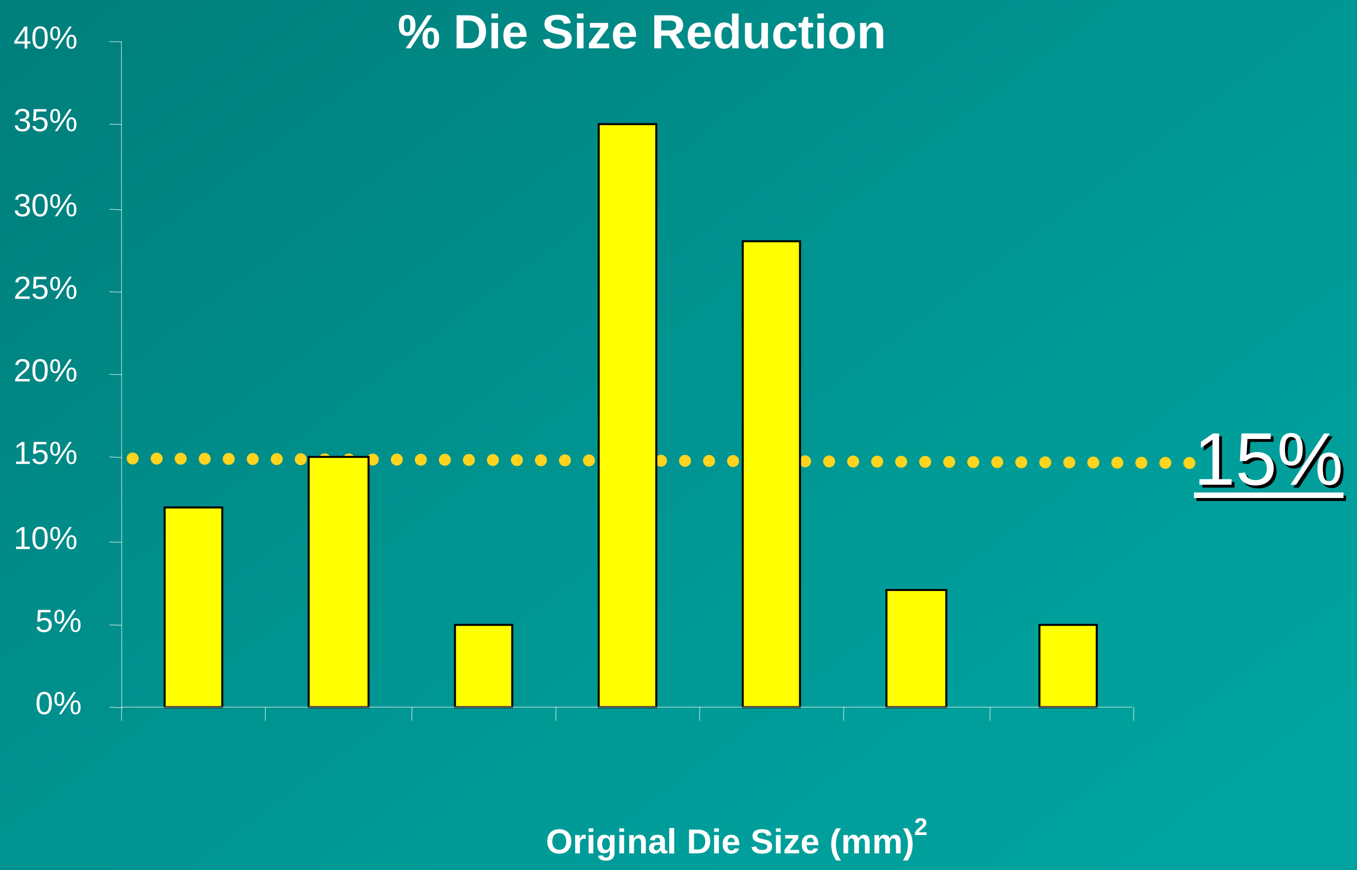
21 Hours



10M gates
10 blocks
10 X 10 mm

QoR: Smaller SoCs

- **15%** average die reduction across 7 benchmarks

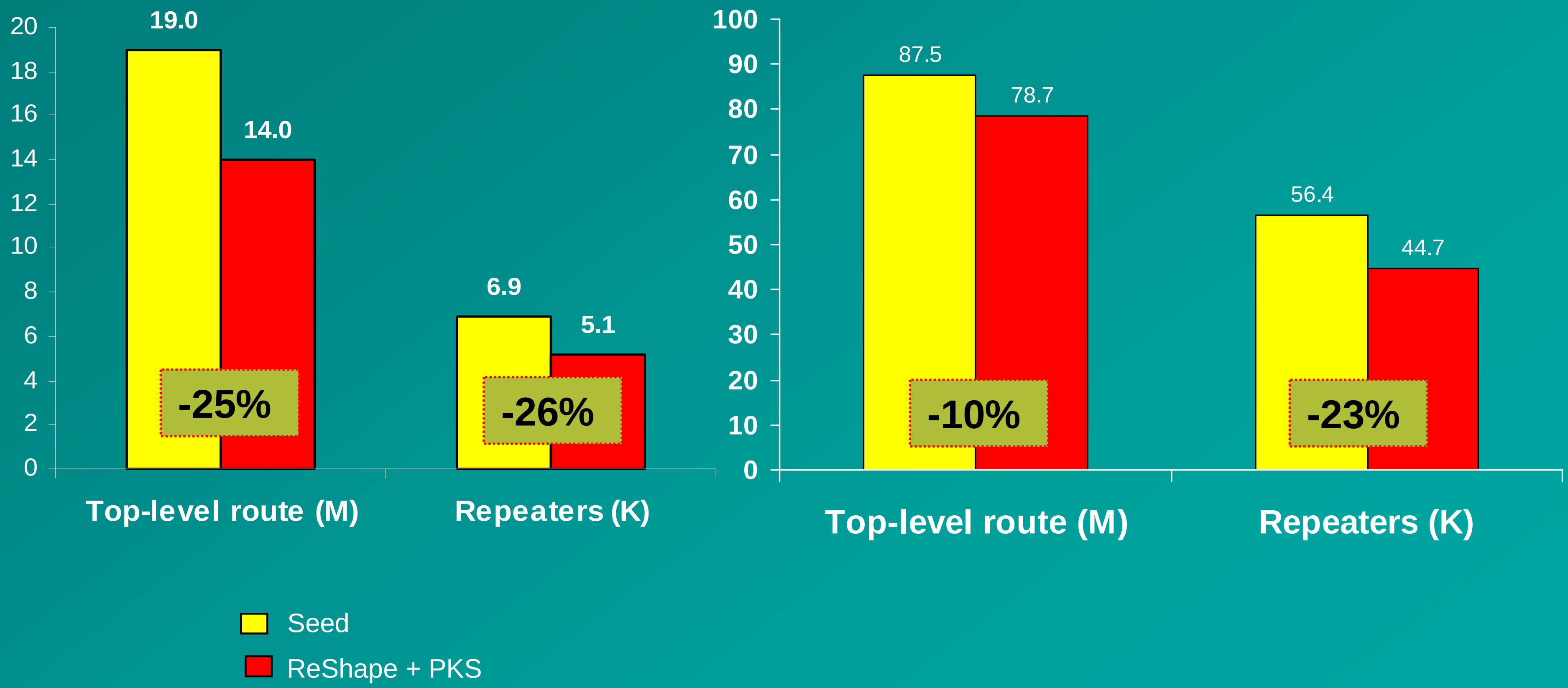


ReShape

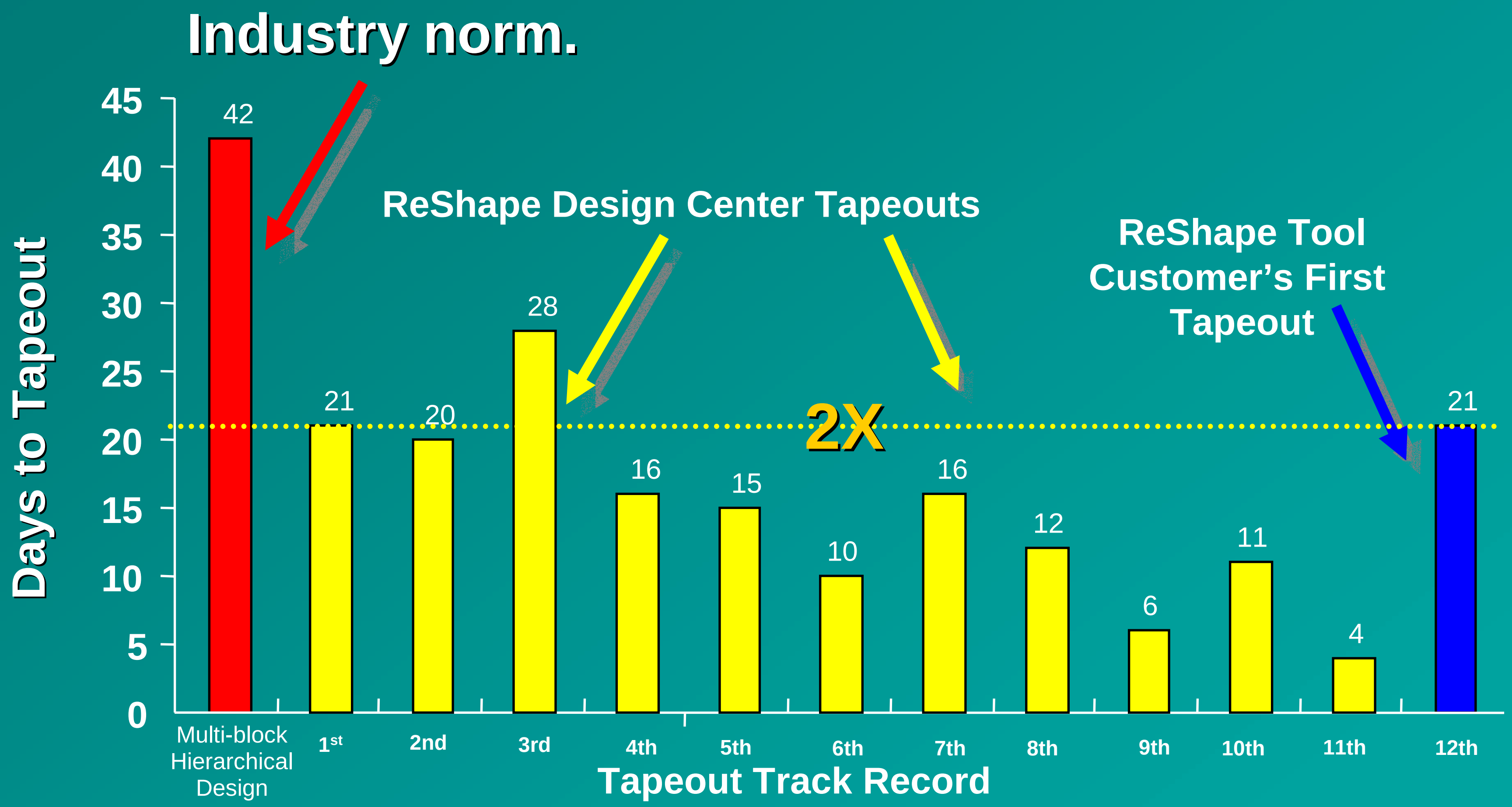
QoR: Faster Chips

1.1M Instance Design
After One Optimization Pass

4.6M Instance Design
After One Optimization Pass



2X Shorter Tapeouts

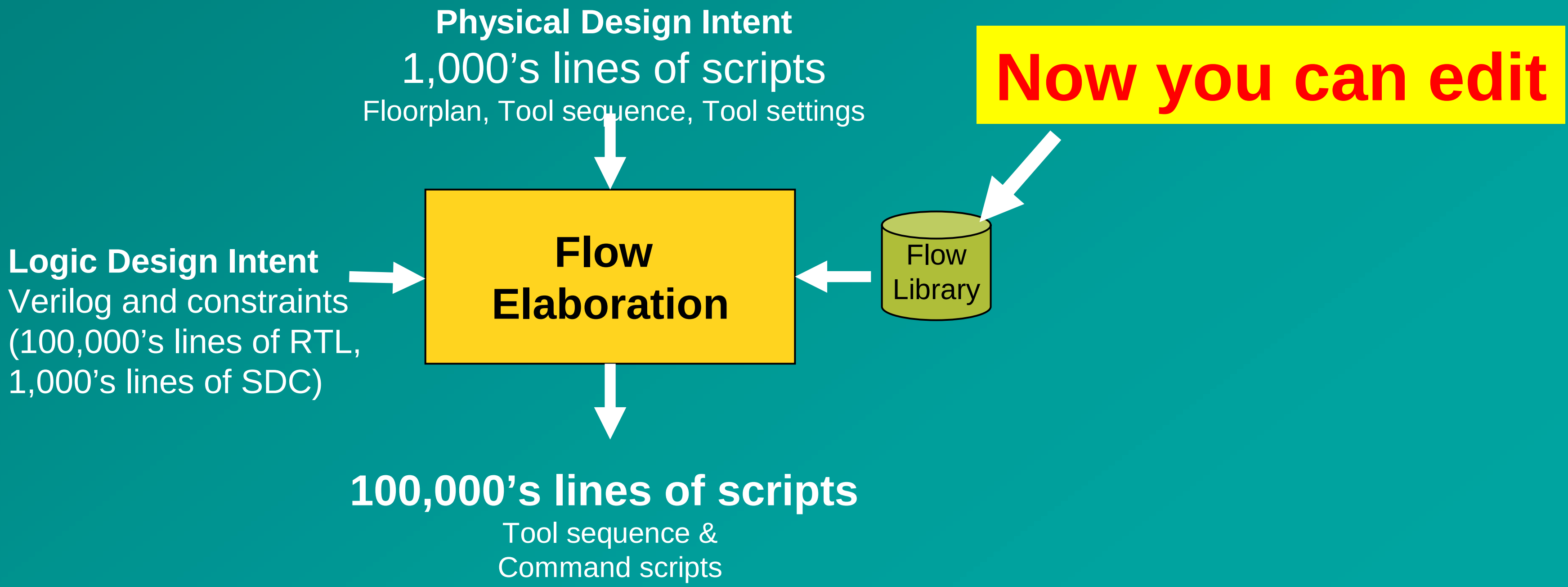


Time is from "final" deliverables to tape shipped to mask shop

Time includes accommodating 2 to 3 ECOs over the time period

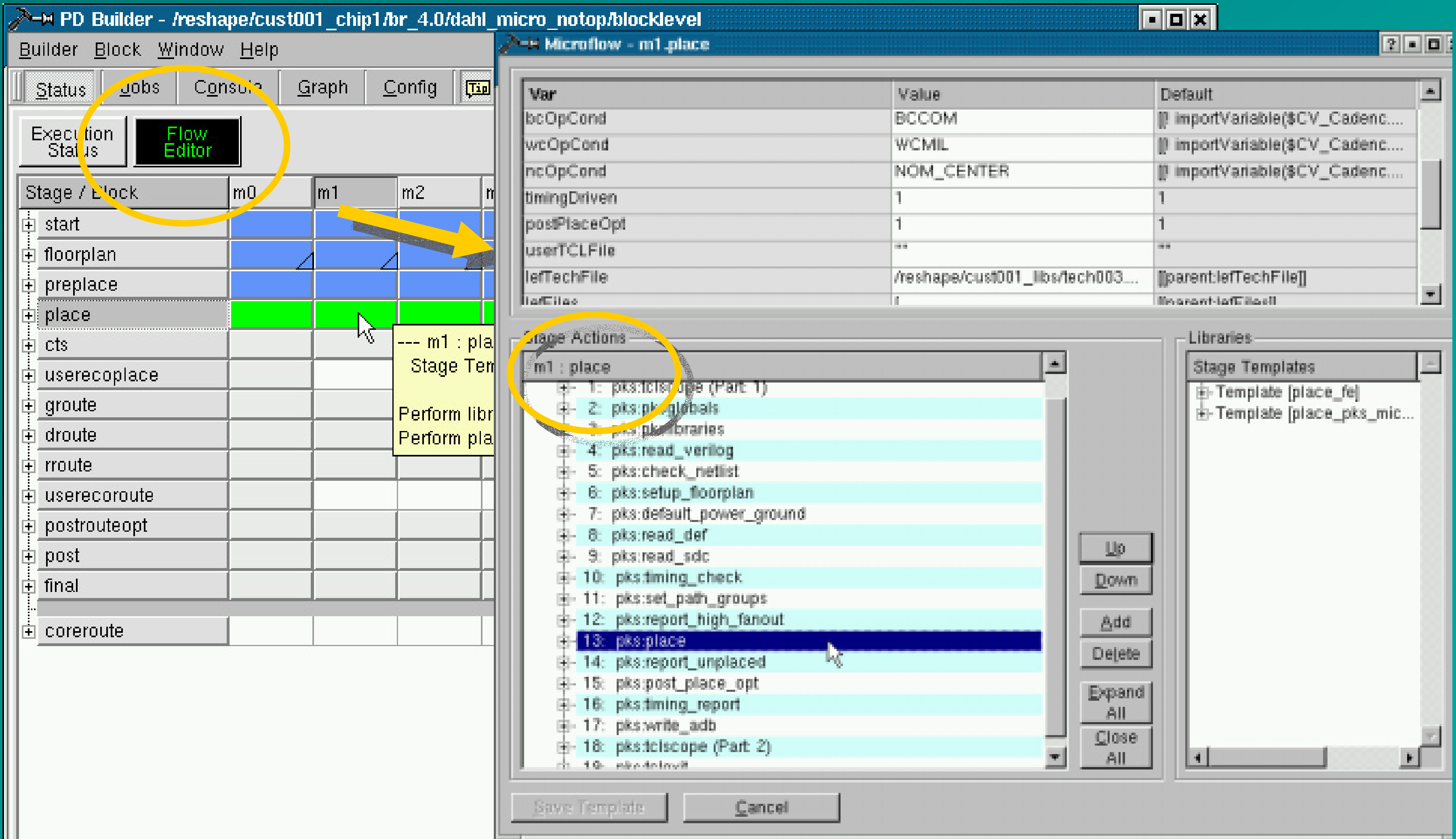
Announcing: ReShape Open Flow

- Makes SoC construction recipes Reusable
  - Leverage on the next design
  - Leverage across the design community



ReShape's Open Flow Editor

- Stage actions are exposed



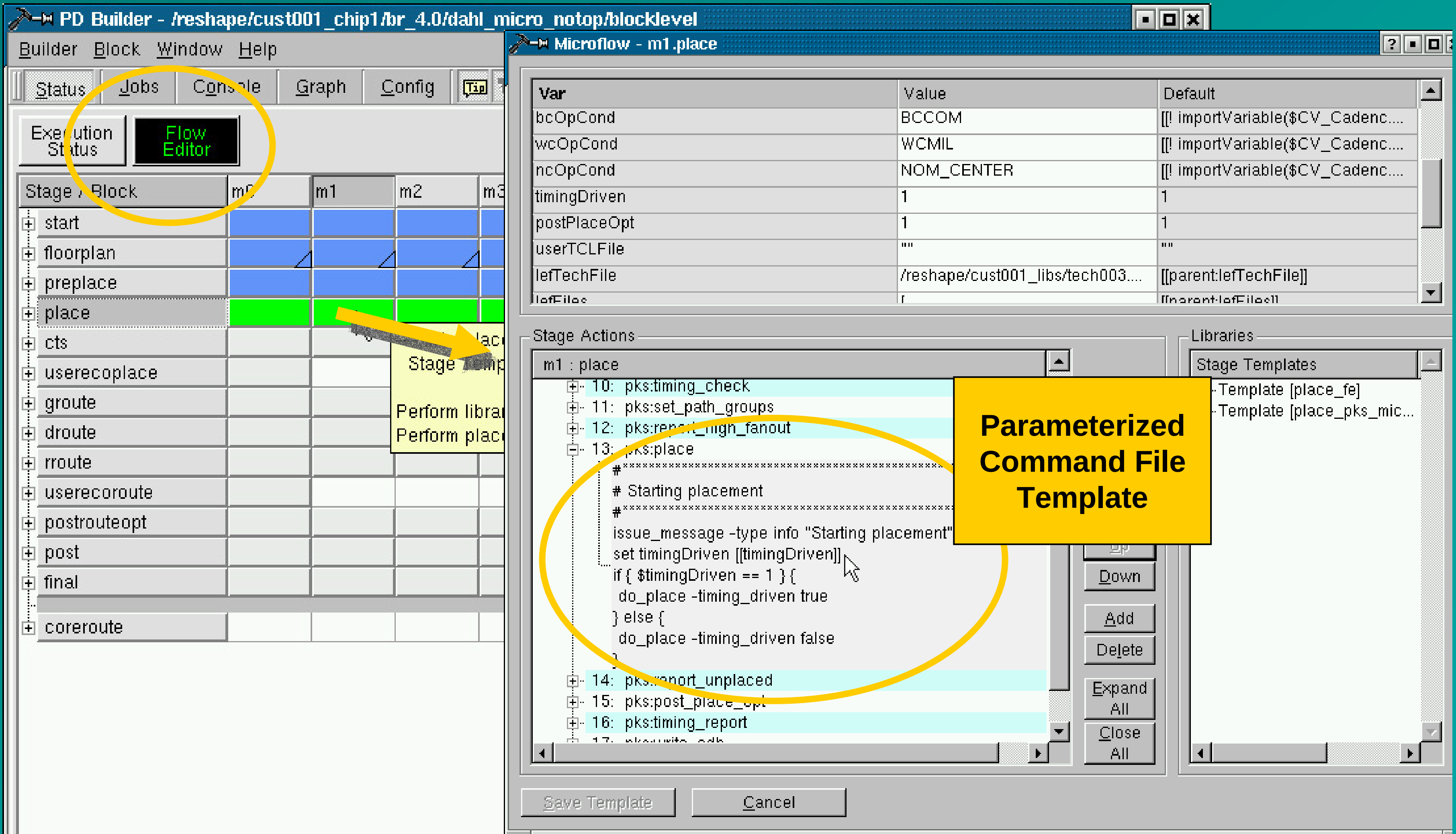
Var	Value	Default
bcOpCond	BCCOM	[[importVariable($CV_Cadenc....
wcOpCond	WCMIL	[[importVariable($CV_Cadenc....
ncOpCond	NOM_CENTER	[[importVariable($CV_Cadenc....
timingDriven	1	1
postPlaceOpt	1	1
userTCLFile	""	""
lefTechFile	/reshape/cust001_libs/tech003....	[[parent:lefTechFile]]
lefFiles	[	[[parent:lefFiles]]

m1 : place

1: pks:tclscope (Part 1)
2: pks:pkglobals
3: pks:pklibraries
4: pks:read_verilog
5: pks:check_netlist
6: pks:setup_floorplan
7: pks:default_power_ground
8: pks:read_def
9: pks:read_sdc
10: pks:timing_check
11: pks:set_path_groups
12: pks:report_high_fanout
13: pks:place
14: pks:report_unplaced
15: pks:post_place_opt
16: pks:timing_report
17: pks:write_adb
18: pks:tclscope (Part 2)
19: pks:tclexit

Open Flow

- Flow template is editable



Stage / Block:

- start
- floorplan
- preplace
- place
- cts
- userecoplace
- groute
- droute
- rroute
- userecoroute
- postrouteopt
- post
- final
- coreroute

Var	Value	Default
bcOpCond	BCCOM	[[! importVariable($CV_Cadenc....
wcOpCond	WCMIL	[[! importVariable($CV_Cadenc....
ncOpCond	NOM_CENTER	[[! importVariable($CV_Cadenc....
timingDriven	1	1
postPlaceOpt	1	1
userTCLFile	""	""
lefTechFile	/reshape/cust001_libs/tech003....	[[parent:lefTechFile]]
lefFiles	[	[[parent:lefFiles]]

m1 : place

```
10:  pks:timing_check
11:  pks:set_path_groups
12:  pks:report_high_fanout
13:  pks:place
      #**********************************************
      # Starting placement
      #**********************************************
      issue_message -type info "Starting placement"
      set timingDriven [[timingDriven]]
      if { $timingDriven == 1 } {
       do_place -timing_driven true
      } else {
       do_place -timing_driven false
      }
14:  pks:report_unplaced
15:  pks:post_place_opt
16:  pks:timing_report
17:  pks:write_adb
```

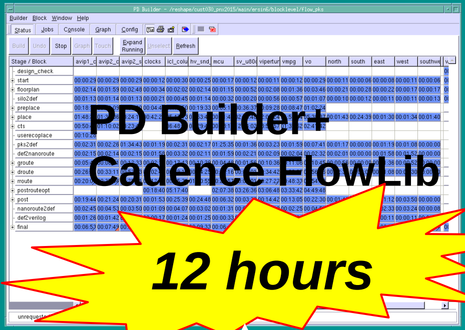
Stage Templates

- Template [place_fe]
- Template [place_pks_mic...

Parameterized Command File Template

Multi Vendor Support Achieved

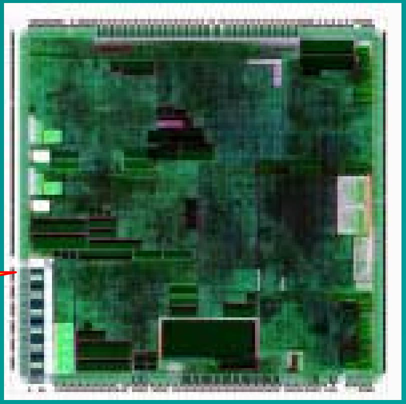
Cadence



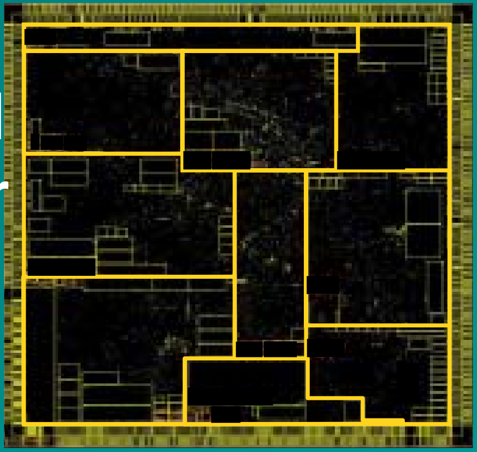
Stage / Block	avip1_c	avip2_c	avip2_s	clocks	icl_colu	hv_snd_	mcu	sv_u80	vipertur	vmpg	vo	north	south	east	west	southwe	v
design_check																	00
start	00:00:29	00:00:29	00:00:29	00:00:12	00:00:30	00:00:25	00:00:17	00:00:12	00:00:11	00:00:12	00:00:29	00:00:11	00:00:06	00:00:08	00:00:11	00:00:06	00
floorplan	00:02:14	00:01:59	00:02:48	00:00:34	00:02:02	00:02:14	00:01:15	00:00:52	00:02:08	00:01:36	00:03:46	00:00:21	00:00:28	00:00:22	00:00:17	00:00:17	00
silo2def	00:01:13	00:01:14	00:01:13	00:00:21	00:00:45	00:01:14	00:00:32	00:00:20	00:00:56	00:00:57	00:01:07	00:00:10	00:00:12	00:00:11	00:00:11	00:00:13	00
preplace	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	00:19:33	[illegible]	00:36:37	00:09:28	00:08:47	01:02:24						
place	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	00:01:43	00:24:39	00:01:30	00:01:34	00:01:40	
cts	[illegible]	01:10:02	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	02:41:42						
userecoplace	00:10:26																
pks2def	00:02:31	00:02:26	01:34:43	00:01:19	00:02:31	00:02:17	01:25:35	00:01:36	00:03:23	00:01:59	00:07:41	00:01:17	00:00:00	00:01:19	00:01:08	00:00:00	
def2nanoroute	00:02:15	00:02:14	00:02:15	00:01:58	00:03:32	00:02:11	00:01:59	00:02:21	00:02:09	00:02:04	00:02:32	00:02:01	00:00:00	00:01:58	00:01:52	00:00:00	
groute	[illegible]	[illegible]	00:12:33	[illegible]	00:17:43	[illegible]	00:04:48	00:01:56	00:10:36	[illegible]	[illegible]	00:00:56	00:00:00	[illegible]	00:00:52	00:00:00	
droute	[illegible]	00:33:11	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	
rroute	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]						
postrouteopt				00:18:40	05:17:40		02:07:38	03:26:36	03:06:48	03:33:42	04:49:48						
post	00:19:44	00:21:24	00:20:31	00:01:53	00:25:39	00:24:48	00:06:32	[illegible]	00:14:42	00:13:05	00:22:30	[illegible]	[illegible]	[illegible]	00:03:50	00:00:00	
nanoroute2def	00:02:45	00:04:53	00:03:50	00:01:09	00:04:07	00:03:02	00:01:31	[illegible]	[illegible]	00:02:25	[illegible]	[illegible]	[illegible]	[illegible]	00:03:24	00:00:08	
def2verilog	00:01:26	00:01:42	[illegible]	[illegible]	00:01:24	00:01:25	00:00:33	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	00:00:11	[illegible]	
final	00:06:53	00:07:49	[illegible]	[illegible]	[illegible]	00:09:33	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	00

PD Builder+
Cadence FlowLib

12 hours



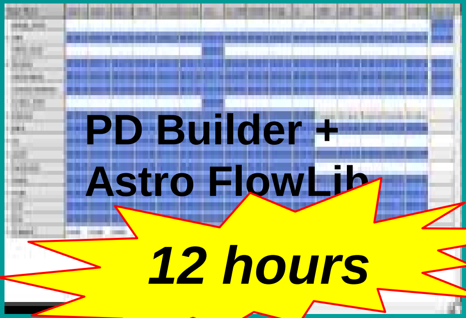
Physical designer Intent



Front-end designer intent

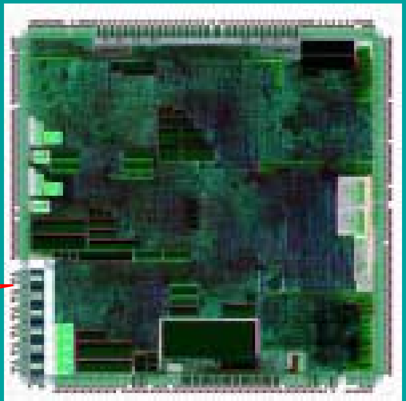
Netlist, constraints, library

Synopsys



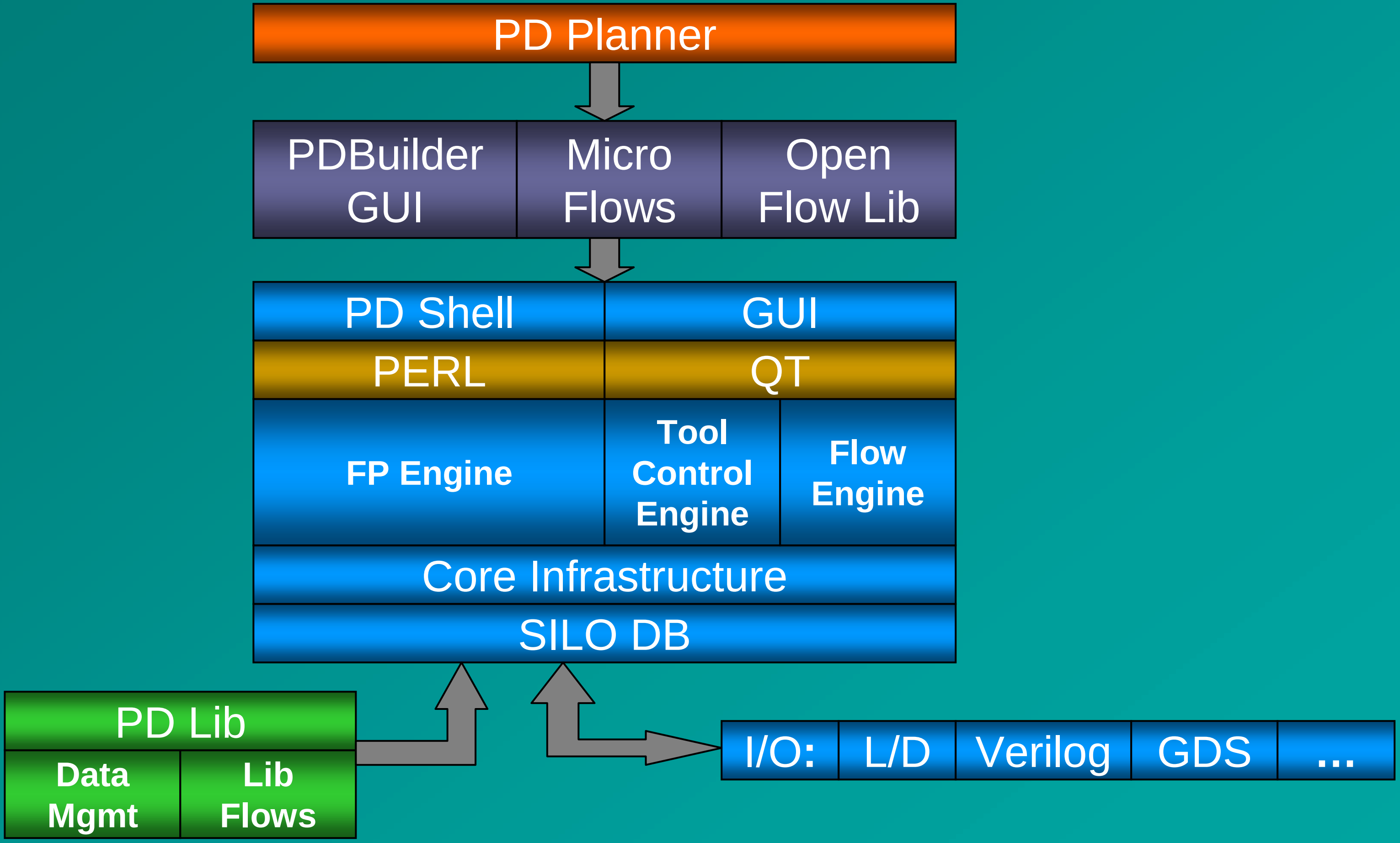
PD Builder +
Astro FlowLib

12 hours



1.1 million instance design courtesy of Philips Semiconductors

ReShape Architecture



ReShape Architecture

- Value-added Components
  - Replayable Floorplan support
  - Flow support infrastructure
  - Huge value of tool-interface stages
  - Block-building/queuing/vendor-tool management support
- Supports Standard/Vendor Formats
  - LEF/DEF/Verilog/SPEF/GDS/OA for all
  - Scheme/Milkyway/SDCs/etc. for Synopsys
  - FE Floorplan/Congestion/etc. for Cadence
- Good Use of Standards
  - QT for graphics
  - PERL/SWIG for high-level language and interfaces
  - XML for Flow support

Summary

- Chip-Level Design Automation next 10X Productivity
  - Automates front-end floorplanning process
  - Fully automates back-end block-generation process
- Lets Design Groups concentrate on global design
  - Tedium of block generation removed
- Faster to first chip build
- *Much* faster turns as design evolves and changes
- First practical customizable flow management
  - Easy to add proprietary/3rd-party tools to flows
- Vendor-neutral flow management

Raising the level of abstraction from cell-level design

ReShape