

Agentic AI for Next-Generation Chip Verification

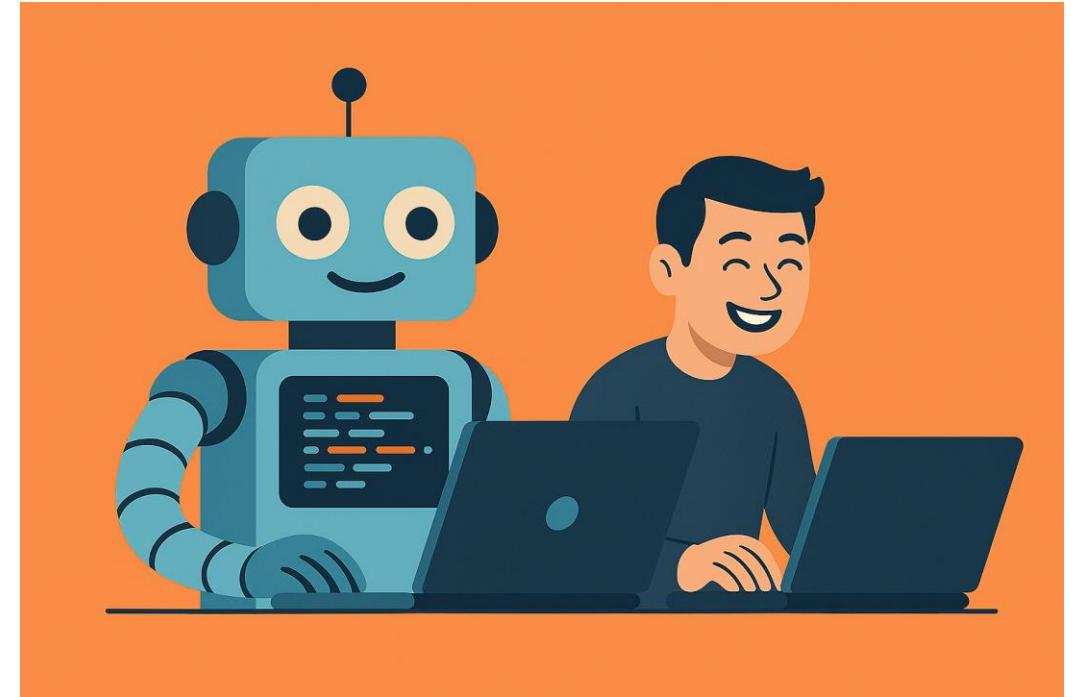
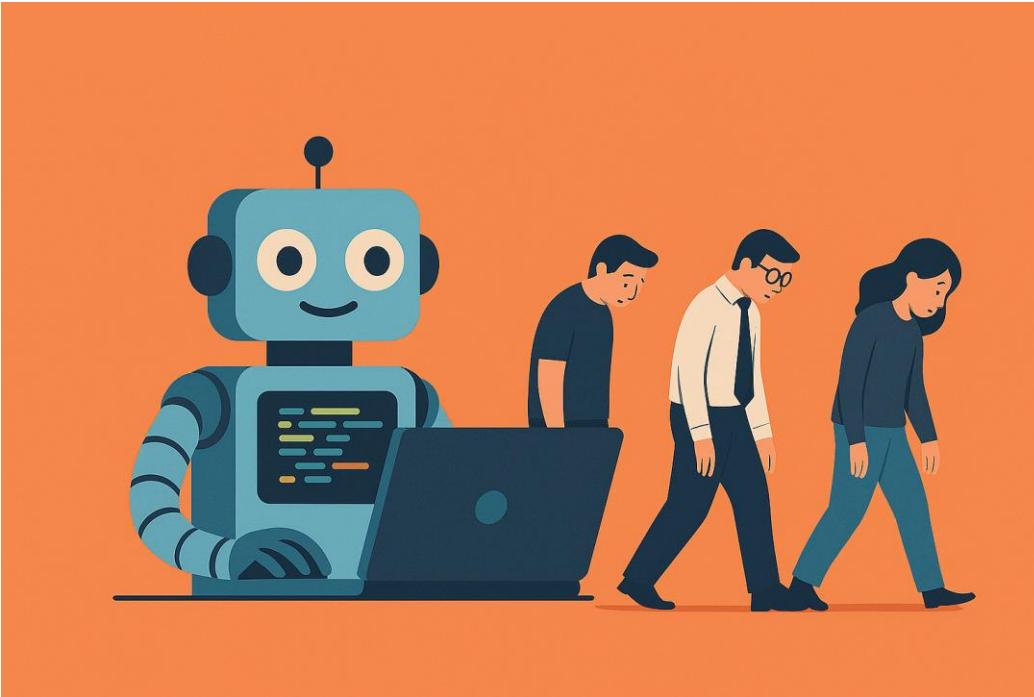
Verisium Powered by JedAI

Anika Sunda – Director Product Management, System Verification Group
December 16, 2025

cadence®

Engineers won't lose their jobs to AI.

They will lose their jobs to another engineer using AI. – Jensen Huang



AI Coding – Making the world a better place?

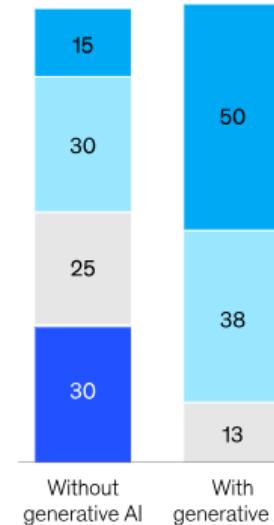
- Automate tedious tasks
 - Boiler plate and syntax
 - Unit tests and edge cases
 - Debugging
 - Documentation
- Focus engineering time on creative problem-solving
- Improve morale and engagement – “Happy bees make sweeter honey.”

Generative AI tools have potential to improve the developer experience.

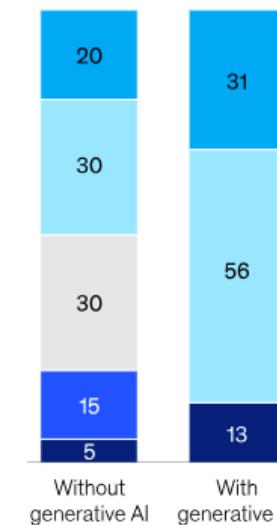
Agreement with statement,
% of respondents

Strongly
disagree Somewhat
disagree Neither agree
or disagree Somewhat
agree Strongly
agree

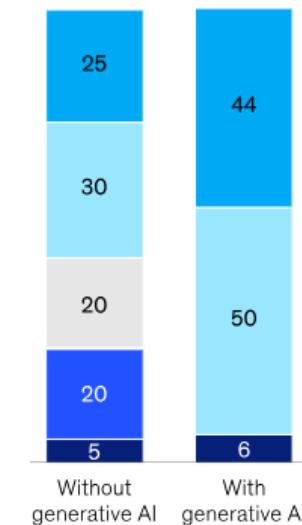
I felt happy



I was able to focus on satisfying and meaningful work



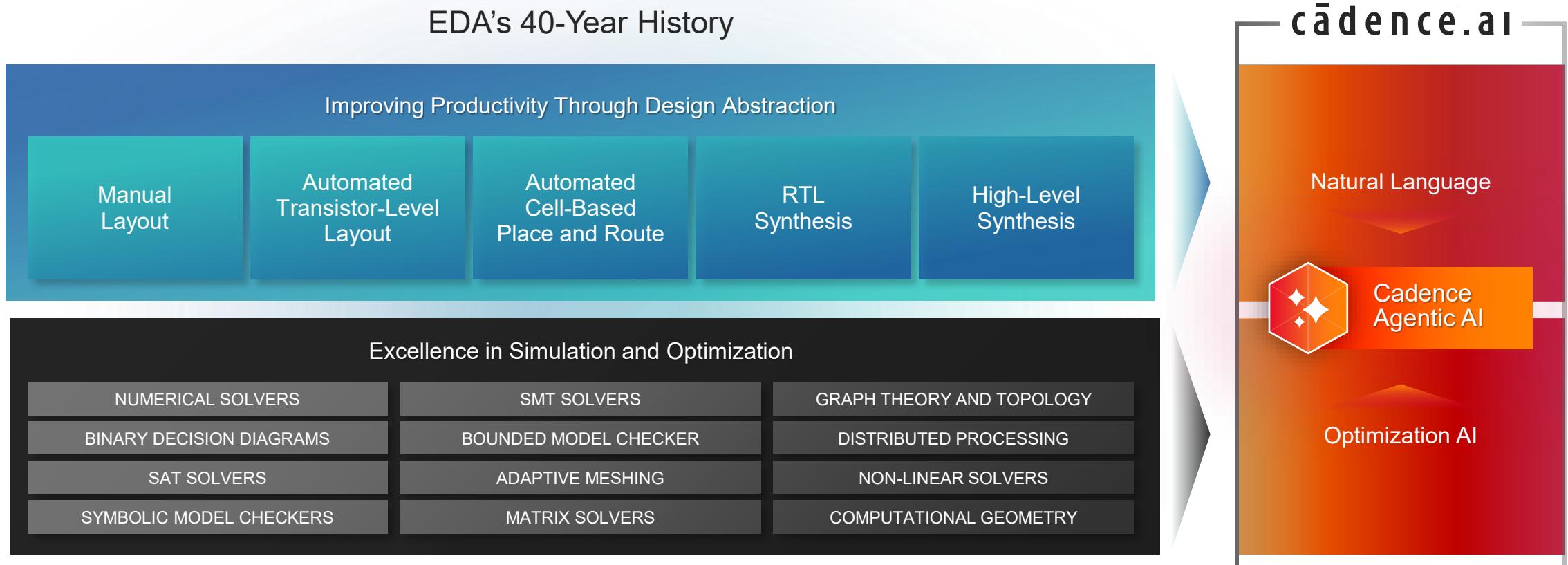
I was in a 'flow' state



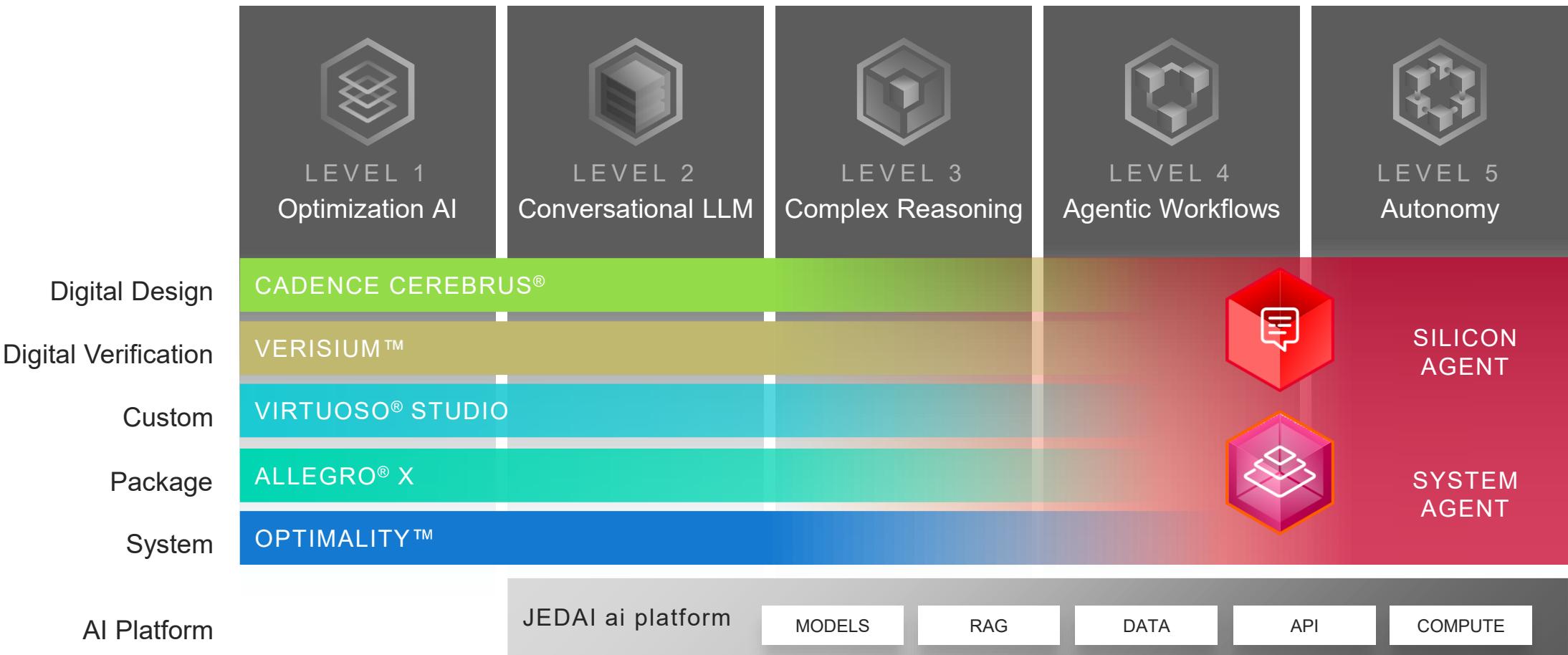
Note: Figures may not sum to 100%, because of rounding.

McKinsey & Company

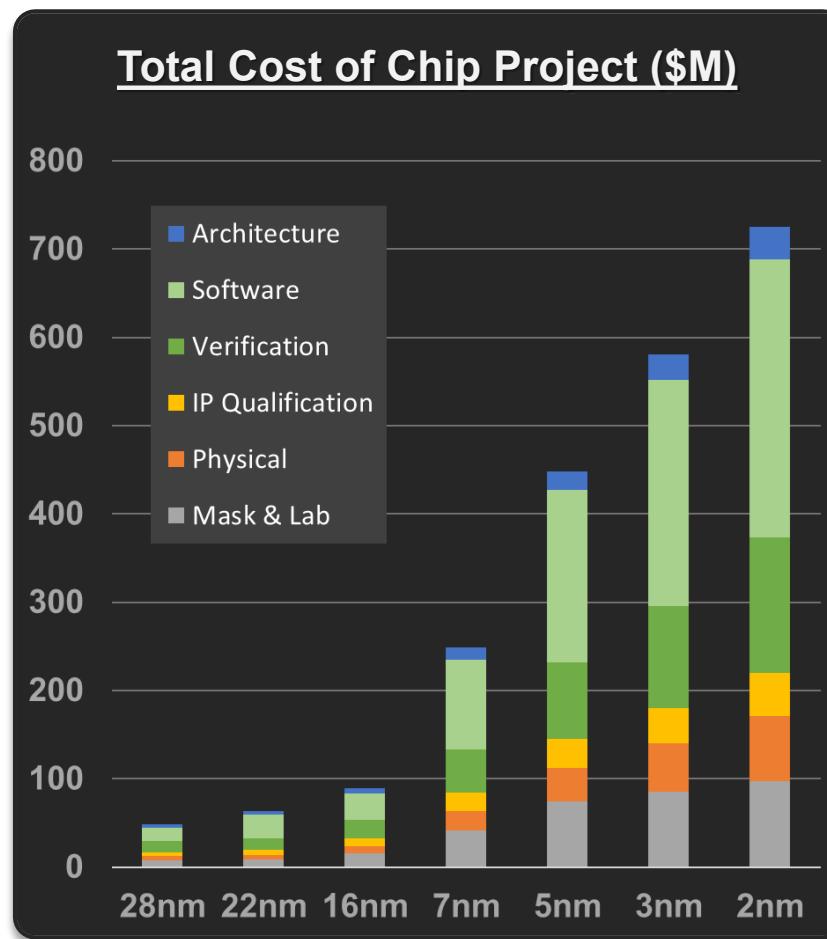
Agentic AI: Optimize Engines & Optimize Engineers



The Journey to Autonomous Design



Verification is Still the Infinite Problem

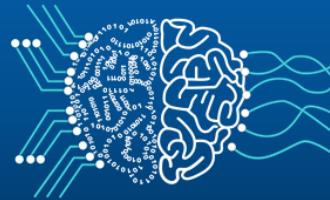


IBS, Global Semiconductor Industry Service Report

- Verification largest cost in IC dev
- Advanced node and 3D-IC means more gates, more states, more potential bugs
- Dev schedules continue to tighten
- Engineering resource gap is growing

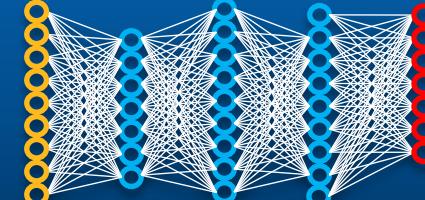
Agentic AI Journey – Level1 - Optimization AI

Digital Implementation (Cerebrus)



Reinforcement Learning

Digital Simulation (SimAI)



Reinforcement Learning

Digital Formal Verification (Formal AI - Jasper)



Reinforcement Learning

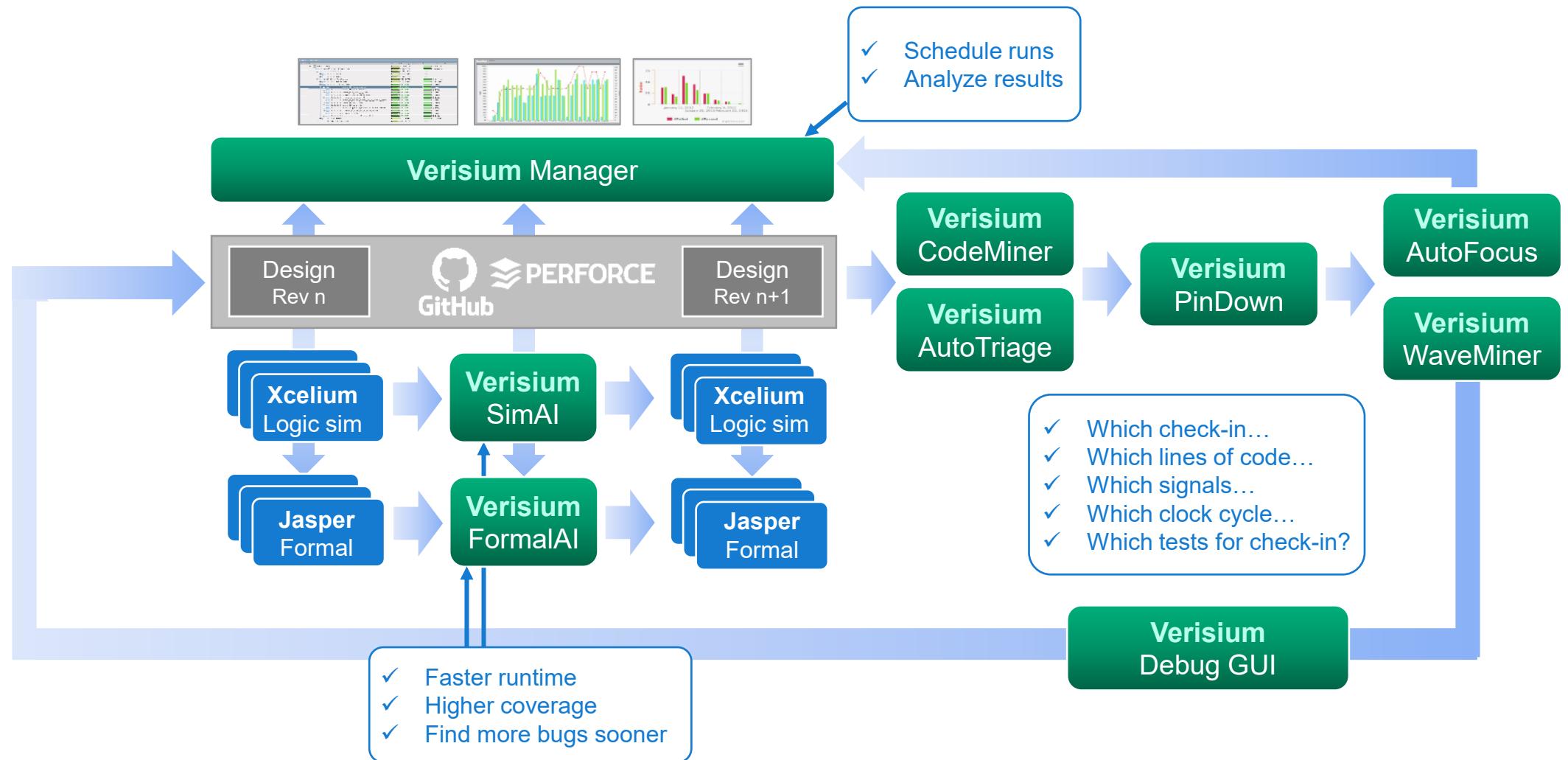
Node	Productivity	PPA
4nm	5x	8% leakage
4nm	5x	10% leakage
6nm	5x	20% PPA
5nm	5x	14% Fmax
7nm	5x	8% power
5nm	5x	15% power
N12	4x	13% PPA
5nm	4x	25% power

Design Type	Speed Up
Modem IP	20x
Mobile SoC	7x
AI IP	8x
AI IP	3x
Memory IP	5x
Base Station	18x
NOC IP	5x
Automotive SoC	5x
CPU IP	3x

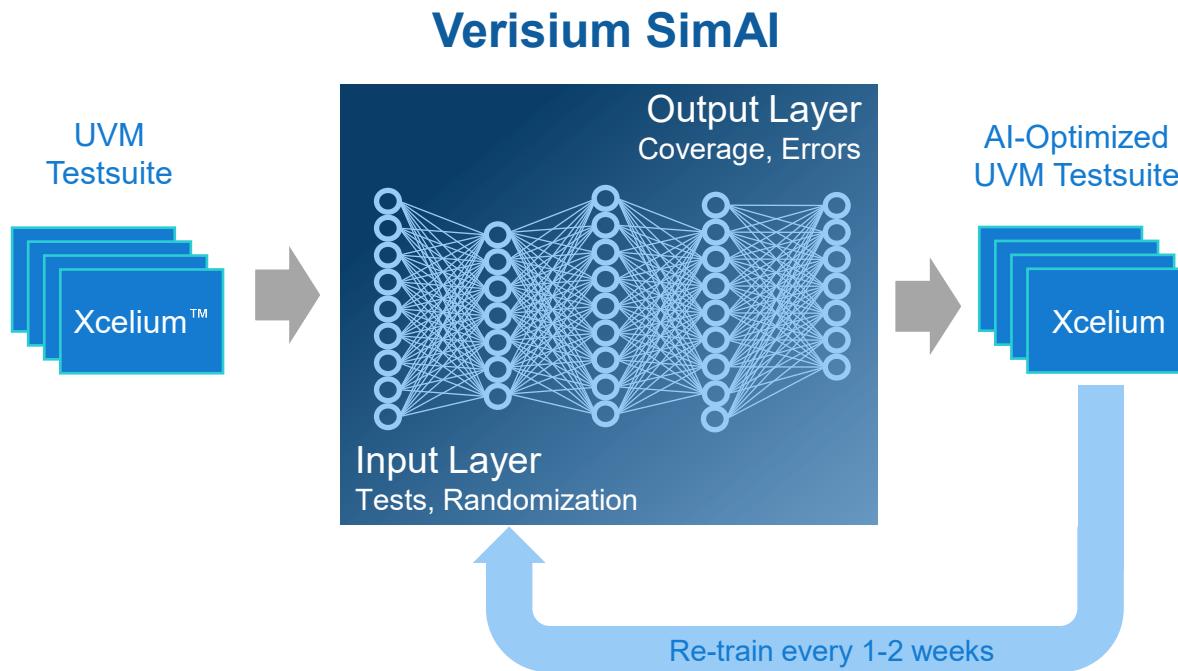
Design	Proof
Block A	1.2x
Block B	2.1x
Block C	1.9x
Block D	1.6x
Block E	1.5x
Block F	2.5x
Block G	3.1x
Block H	1.6x
Block I	3.6x

Optimization of core algorithms and engines with ML (AI Inside)
Using tools data to improve the core algorithms of engines and flows.

Verisium in Action



Xcelium SimAI

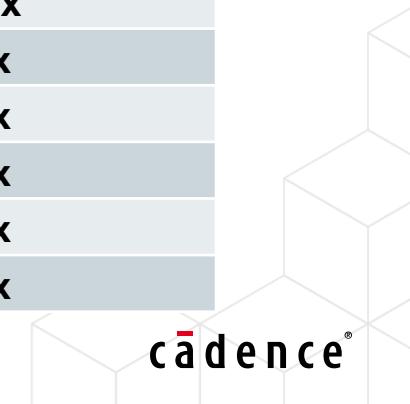


Used by 18 of the top 20 Semis

AI Mode	AI Mode Goal
Compression Mode	Provides compressed test list achieving similar coverage as original full regression
Failure Amp Mode	Amplifies a rare hard to hit failure so it can be predictably reproduced and get fixed quickly.
Coverage Amp Mode	Finds new bugs by amplifying rarely hit on bins
Closure Mode	Hit cover points that have not been hit yet. Works best when run iteratively (run-refine-run-refine...)
AutoFocus Mode*	Automatically generates a customized sanity/check-in gate testuite based on a specific RTL/TB changelist.

* December 2025

Design Type	Speed Up (Compression Mode)
Modem IP	20x
Mobile SoC	7x
AI IP	8x
AI IP	3x
Memory IP	5x
Base Station	18x
NOC IP	5x
Automotive SoC	5x
CPU IP	3x
Network IP	4x
CPU IP	9x



Verisium AI Focused Regression Flow

Automation

Automate regression generation and execution based on user modifications

Efficiency

Minimize regression size and runtime by focusing on the relevant subset of changes

Integration

Unified experience leveraging both SimAI and AutoFocus functionalities

Regression Health Dashboard

New Tab for Regression Health Dashboard

Verisium Manager

Regression Analysis Planning Composer

My Sessions* Views Global Operations Scripts

Launch Import Collect Runs Refresh Scripts Manager Export Export Merge Stop Stop Auto. Suspend Resume Set as completed Delete Relocate Open dir Session Info Recalc UDA Chart Filter Selected

Flow Sessions

Session Status	Name	Total Runs	#Passed
completed	session_AB_sim_args.shuri.25_10_06_01_2...	324	144
completed	session_AB_sim_args.shuri.25_10_06_01_2...	224	98
completed	session_AB_sim_args.shuri.25_10_06_01_2...	224	104
completed	session_AB_sim_args.shuri.25_10_06_01_2...	224	96
completed	session_AB_sim_args.shuri.25_10_06_01_1...	224	85
completed	session_AB_sim_args.shuri.25_10_06_01_1...	224	103
completed	session_AB_sim_args.shuri.25_10_06_01_1...	220	87
completed	session_AB_sim_args.shuri.25_10_06_01_1...	23	9
completed	session_AB_sim_args.shuri.25_10_06_01_0...	23	15
in_progress	crc_demo.shuri.25_10_06_01_03_26_6453	97	0
in_progress	crc_demo.shuri.25_10_06_00_55_38_5076	97	0
completed	crc_demo.shuri.25_10_06_00_46_58_7057	60	60
completed	crc_demo.shuri.25_10_06_00_41_23_9454	60	60
completed	crc_demo.shuri.25_09_29_22_05_28_7424	60	60
completed	crc_demo.shuri.25_09_29_21_57_02_7015	60	60
completed	crc_demo.shuri.25_09_29_20_18_24_6570	60	60
completed	session_AB_sim_args.shuri.25_09_29_20_0...	16	6
completed	session_AB.shuri.25_09_29_30_51_1174	13	5
completed	session_AB_sim_args.shuri.25_09_29_18_3...	16	8
completed	session_AB.shuri.25_09_29_18_24_21_4483	13	5
completed	session_AB_sim_args.shuri.25_09_29_18_2...	16	8
completed	session_AB.shuri.25_09_29_18_16_05_3494	13	5

Showing 22 out of 22 items, 1 item selected

Health dashboard Errors Warnings/Info

Run Anomalies

Duration (sec.)	CPU Time (ms.)	Max Memory Usage (MB)	Average Memory Usage (MB)	Max Runs In Parallel
170	54	185	185	200
0	0	0	0	Max Requested
0	0	39	39	188
0	0	39	39	Max Actual

None Major Minor Unknown

Actual Index	Index	Name	CPU Time (ms.)	Estimated CPU Time (ms.)	Duration (sec.)	Estimated Duration (sec.)	Max Memory Usage (MB)	Estimated Max Memory Usage (MB)	Average Memory Usage (MB)
(no filter)	(no filter)	(no filter)	(no filter)	(no filter)	(no filter)	(no filter)	(no filter)	(no filter)	(no filter)
2	1	/group_A/A1_test	3689	2759	11	3	237	215	127.0
3	2	/group_A/A1_test	3541	2759	11	3	205	215	137.0
4	3	/group_A/A1_test	3684	2759	12	3	202	215	121.0
5	4	/group_A/A1_test	3772	2759	13	3	228	215	137.0
6	5	/group_A/A1_test	3690	2759	12	3	202	215	124.0
7	6	/group_A/A1_test	3786	2759	12	3	250	215	144.0
8	7	/group_A/A1_test	3625	2759	14	3	270	215	137.0
9	8	/group_A/A1_test	3671	2759	12	3	229	215	139.0

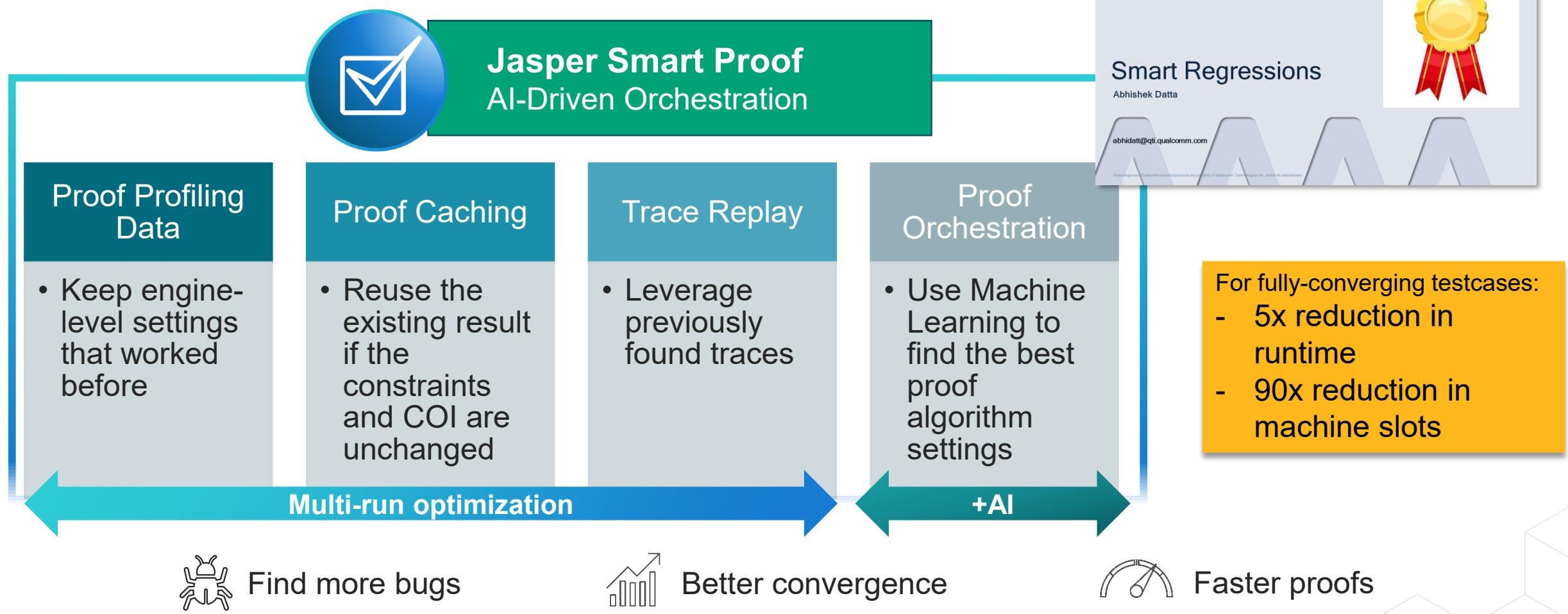
Showing 224 items, 1 item selected

cadence® Regression Center Messages

11 © 2025 Cadence Design Systems, Inc. Cadence confidential.

Optimization AI in Jasper

Parallel runs find more bugs and achieve deeper proofs

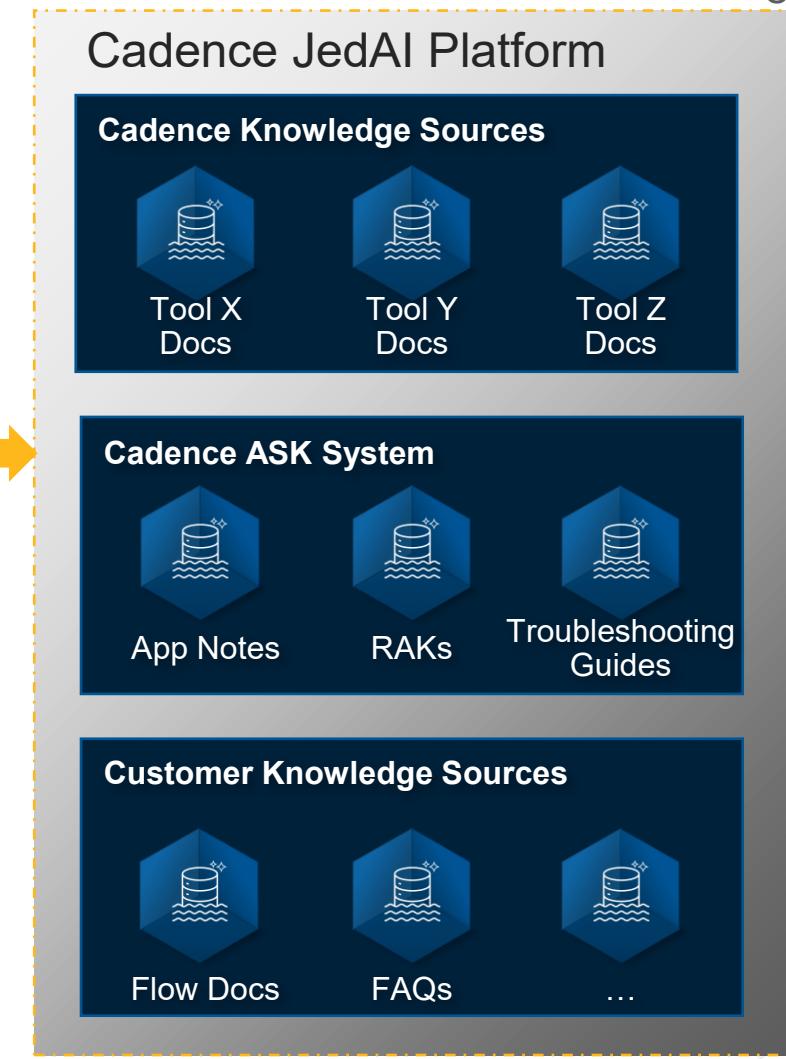
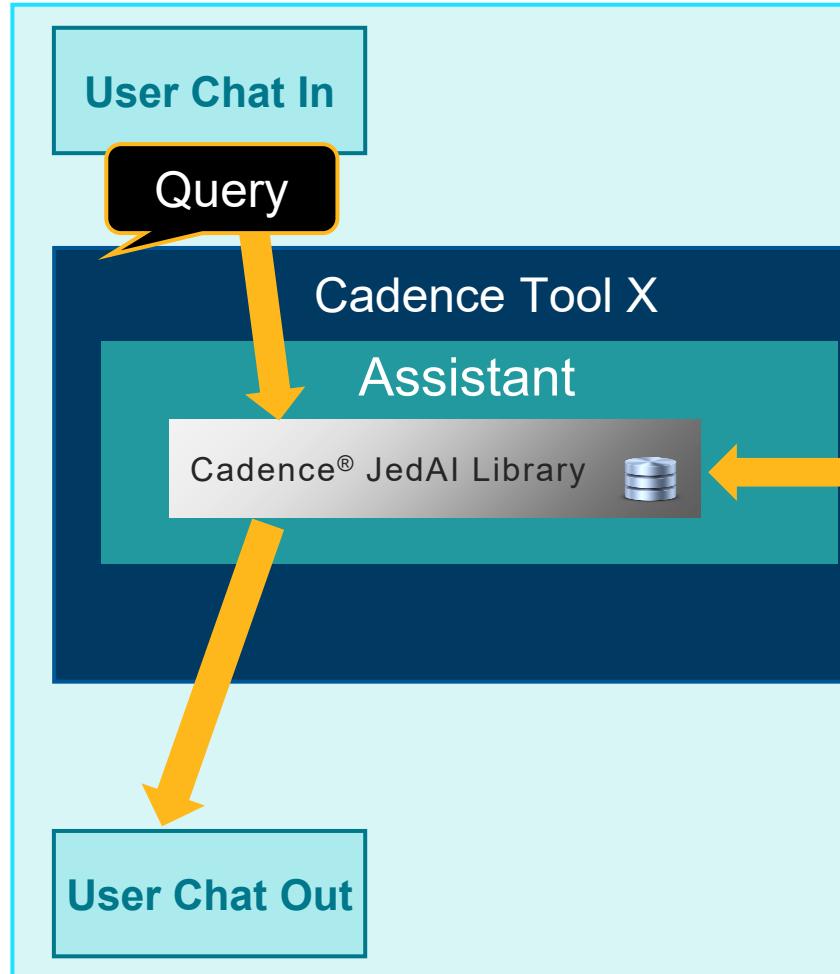
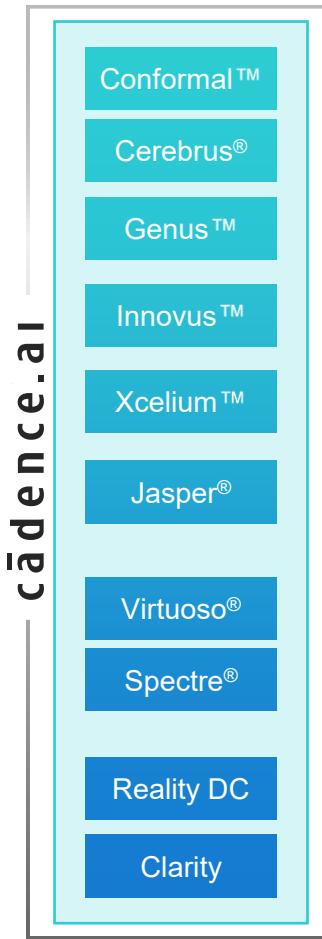


Conversational LLM Agents Across the Software Portfolio



LEVEL 2

Conversational LLM



AI Assistants - Verisium Debug AI Assistant



LEVEL 2

Conversational LLM

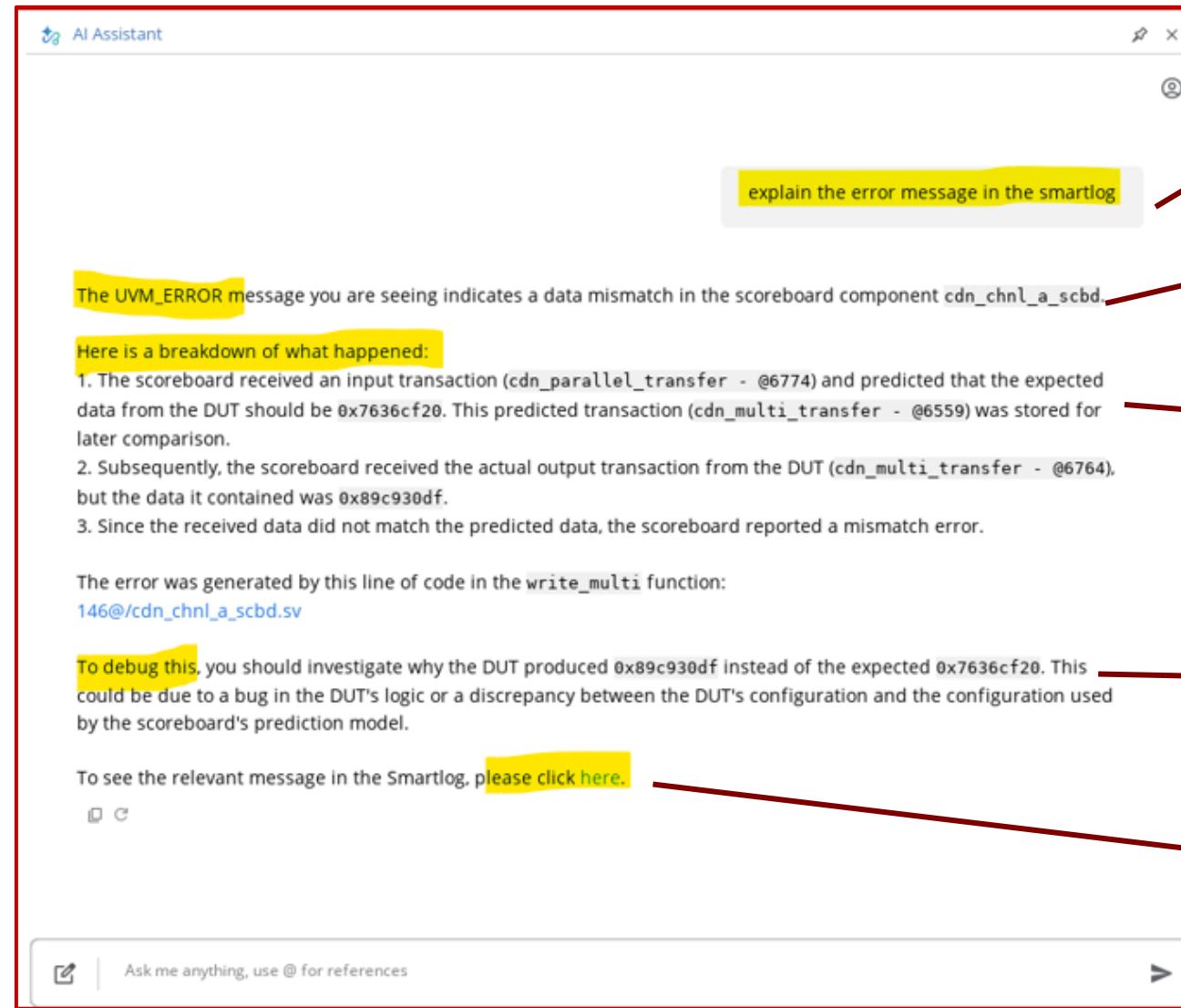
AI Chat

- **Ask and Retrieve** information from your design (Design, TB)
- **Operate** Verisium Debug and summarizing information
- **Answers** from Tool help and Setting options

AI-Powered Tools

- Code inspection - Read code and suggest potential issues – **Autonomous**
- Explain signal - collect data about signal and summarize it's functionality
- Complex waveform expression
- Randomization contradiction AI summary

AI Assistants - Verisium Debug AI Assistant -SmartLog, Source Search



The AI Assistant interface is shown with a red border. The main content area contains a conversation log with several yellow-highlighted text snippets. Red lines connect these snippets to yellow callout boxes on the right.

- Prompt:** A yellow box containing the text "explain the error message in the smartlog".
- Explain the error message:** A yellow box containing the text "The UVM_ERROR message you are seeing indicates a data mismatch in the scoreboard component cdn_chnl_a_scbd".
- Explain what leads to the error:** A yellow box containing the text "Here is a breakdown of what happened:
 1. The scoreboard received an input transaction (cdn_parallel_transfer - @6774) and predicted that the expected data from the DUT should be 0x7636cf20. This predicted transaction (cdn_multi_transfer - @6559) was stored for later comparison.
 2. Subsequently, the scoreboard received the actual output transaction from the DUT (cdn_multi_transfer - @6764), but the data it contained was 0x89c930df.
 3. Since the received data did not match the predicted data, the scoreboard reported a mismatch error.The error was generated by this line of code in the write_multi function:
146@/cdn_chnl_a_scbd.sv
- Suggest next steps:** A yellow box containing the text "To debug this, you should investigate why the DUT produced 0x89c930df instead of the expected 0x7636cf20. This could be due to a bug in the DUT's logic or a discrepancy between the DUT's configuration and the configuration used by the scoreboard's prediction model.
- Links to related SmartLog messages:** A yellow box containing the text "To see the relevant message in the Smartlog, please click [here](#)".

At the bottom of the interface, there is an "Ask me anything, use @ for references" input field and a ">>" button.

AI Assistants - Verisium Debug AI Assistant – *RTL exploration*

AI Assistant

find the RTL code that inverts the data

The data inversion is caused by the RTL code in the file /vols/verisium_builds/sanity_ci/compass_greg/rflow_test/designs/uvm/clab_uvm/dut/audio.v at line 168.

The specific line of code is:

```
assign cha_mux_out_data = (mux_out == 1'b0 ? ({32{{cha_inverter_en}}} ^ cha_limiter_data) : chb_limiter_data);
```

This code inverts the cha_limiter_data when the signal cha_inverter_en is high. The inversion is performed using a bitwise XOR with a 32-bit vector of all ones.

You can view the code here: [168@/audio.v](#).

The cha_inverter_en signal is enabled by the test environment. In the file /vols/verisium_builds/sanity_ci/compass_greg/rflow_test/designs/uvm/clab_uvm/tests/test_lib.sv, the test mismatch_test sets this signal to 1 through a configuration object constraint:

```
chnl_a.inverter_en == 1'b1;
```

You can view this code here: [301@/test_lib.sv](#).

This configuration is then written to a register, which drives the cha_inverter_en signal in the DUT, as seen in /vols/verisium_builds/sanity_ci/compass_greg/rflow_test/designs/uvm/clab_uvm/dut/regfile.vhd at line 46:

```
cha_inverter_en = chnl_a_conf(3);
```

You can view this code here: [46@/regfile.vhd](#).

Prompt

Found the RTL logic, explain the logic and provide links

Details about the TB code that enabled the inversion

Details on the Register that enables the inversion

AI Assistants - Jasper AI Assistant – Tool Docs Q&A



LEVEL 2
Conversational LLM

The screenshot shows the Cadence Jasper AI Assistant interface. The top window is a 'Formal Property V...' session showing a 'Property Table' with various assertions and covers. The bottom-left window is a command-line interface for tool queries, showing a session named 'session_0' with a history of commands like #, #prove -wait, #cover -remove, #check_cov -measure, and analyze -sv12. The bottom-right window is a 'Jasper Assistant' chat window where the AI is responding to questions about tool commands and analysis.

Formal Property V... 4/4

Design Hierarchy

- pp (pp:(DATA MEM SIZE=8))
 - ig_parse (pp_ig_parse)
 - route_ctrl (pp_route_ctrl:(PTR_W=3))
 - payload_mem (pp_payload_mem:(PTR_W=3))
 - thr_mem (pp_thr_mem:(PTR_W=3))
 - stall (pp_stall:(PTR_W=3))
 - drop_ctrl (pp_drop_ctrl:(PTR_W=3))
 - list_ctrl (pp_list_ctrl:(PTR_W=3))
 - cfg (pp_cfg:(PTR_W=3))
 - intr_ctrl (pp_intr:(PTR_W=3))
 - fv_env (fv_env:(DATA_MEM_SIZE=8,MAX_PKT_LEN=6))
- Packages
- fv_pkg (fv_pkg)
- Compilation Units

Property Table

Type	Name	Engine	Bound	Target Bound	Traces	Time
Assert	pp.payload_mem.mem.no_simultaneous_wri...	?	1-	N/A	0	
Cover (related)	pp.payload_mem.mem.no_simultaneous_wri...	?	1-	N/A	0	
Assert	pp.thr_mem.mem.no_simultaneous_writes_t...	?	1-	N/A	0	
Cover (related)	pp.thr_mem.mem.no_simultaneous_writes_t...	?	1-	N/A	0	
Cover	pp.fv_env.ig_agent.COV_use_all_avail	?	1-	N/A	0	
Cover	pp.fv_env.ig_agent.COV_b2b_pkts	?	1-	N/A	0	
Assert	pp.fv_env.ig_agent.AST_M_valid	?	1-	N/A	0	
Cover (related)	pp.fv_env.ig_agent.AST_M_valid:witness1	?	1-	N/A	0	
Assert	pp.fv_env.ig_agent.AST_M_no_gaps	?	1-	N/A	0	
Cover (related)	pp.fv_env.ig_agent.AST_M_no_gaps:witness1	?	1-	N/A	0	
Assert	pp.fv_env.ig_agent.AST_M_no_msa	?	1-	N/A	0	

Design Hierarchy Task Tree Proof Structure

session_0

```
## #prove -wait
##
## #cover -remove {*} {get_property_list -include {name {tenable|fsm|pattern} :.*$} } -regexp
##
## #check_cov -measure -refresh
2025.09-20250915 64 bits
[embedded] % assistant -tool -query {What is the command to analyze the design using System Verilog version 2012 in Jasper? }
The command to analyze a design using SystemVerilog version 2012 in Jasper is:
analyze -sv12 <design_file>
Here, '<design_file>' is the name of the SystemVerilog file that you want to analyze.

[embedded] %
```

Jasper Assistant

Tool Q&A

Hello! How can I help you?

What is the command to analyze the design using System Verilog version 2012 in Jasper?

The command to analyze a design using SystemVerilog version 2012 in Jasper is:
analyze -sv12 <design_file>
Here, <design_file> is the name of the SystemVerilog file that you want to analyze.

[1] jasper_command_reference.pdf [2] Solution-Workaround_for_compiling_a_design_in_Jasper_if_enable_portfaults_xmv

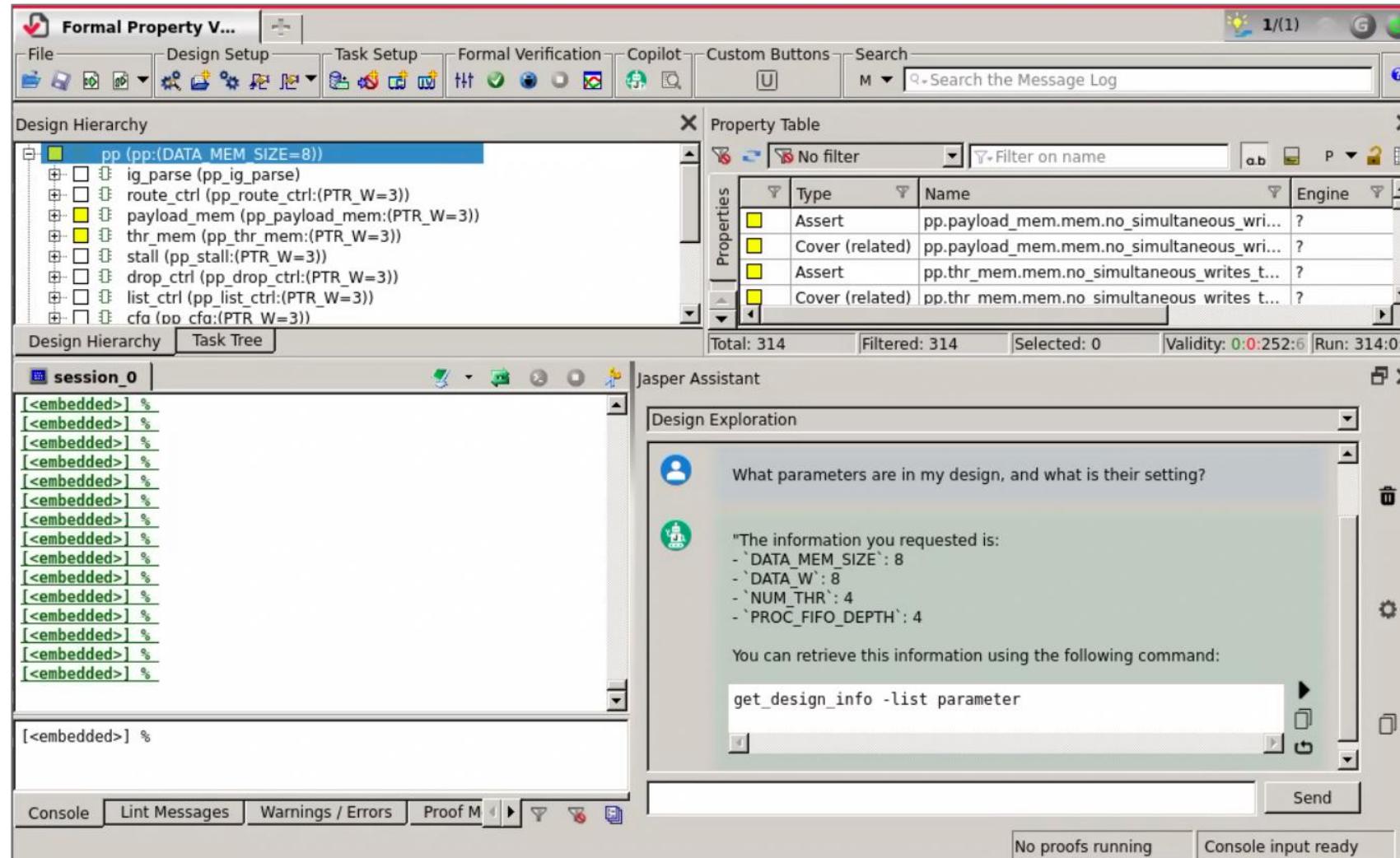
Console Lint Messages Warnings / Errors

- ✓ **Tool documentation search and command querying from NL**
- ✓ Connection to Cadence or customer-provided LLM via JedAI
- ✓ Open queries to streamline general routines
- ✓ History

Query examples

- How to convert the assertion `pp.fv_env.fl_chk.AST_no_duplicate` to an `assume`?
- Give me an example of **commands** to run `hunt cycle swarm`
- How to save the DHO database
- What is target bound?
- How to **enable multiple traces** for an assertion

AI Assistants - Jasper AI Assistant – Design Exploration

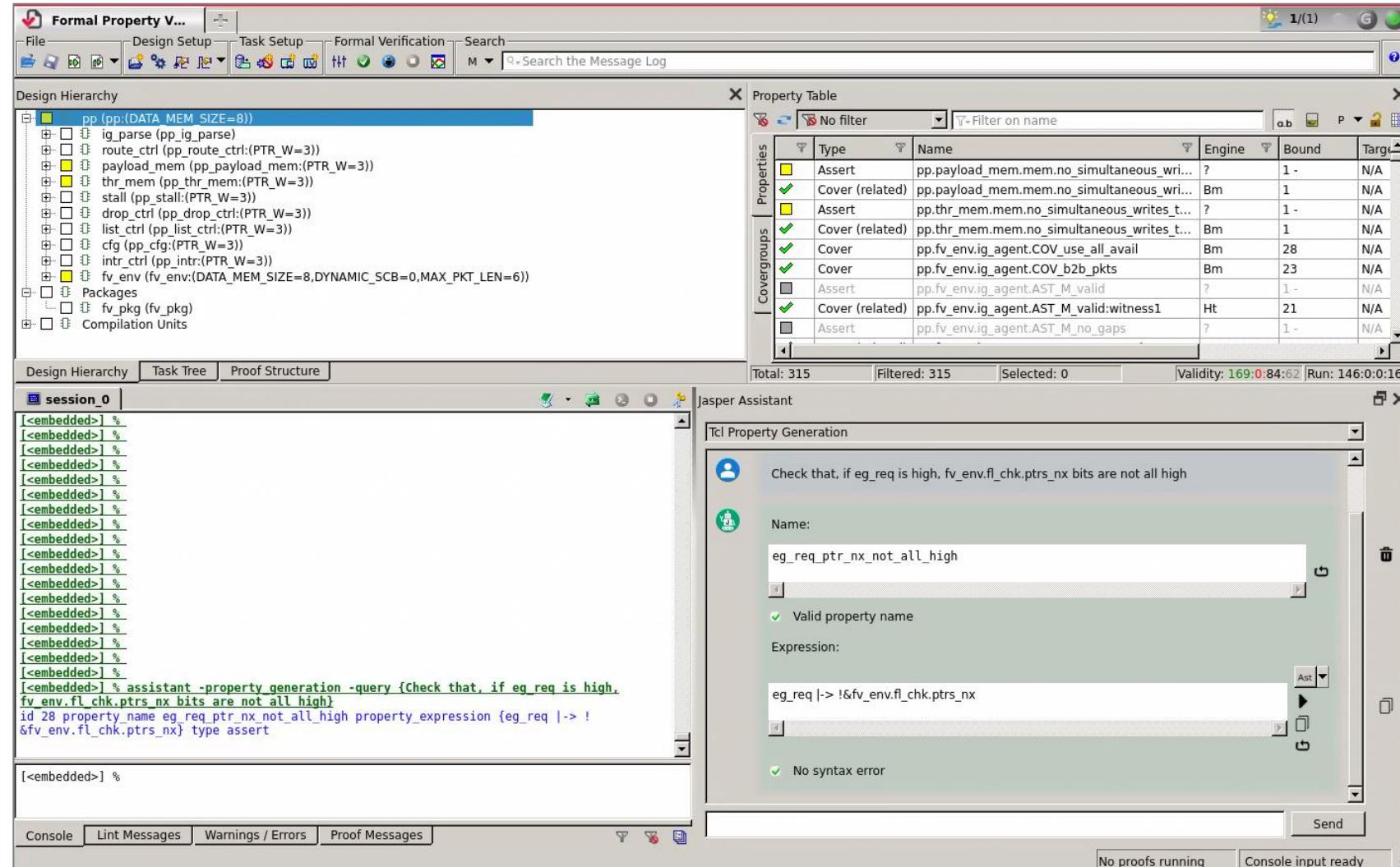


- ✓ Design exploration from natural language
- ✓ Tcl code editable in case of required adjustments

Query examples

- List all IOs of instance <instance>
- What are the instances with parameter PTR_W set to 3?
- Which assertions are disabled?
- What are the FIFOs of instance <instance>?
- What are the instances of module <module>?

AI Assistants - Jasper AI Assistant – Tcl Property Generation

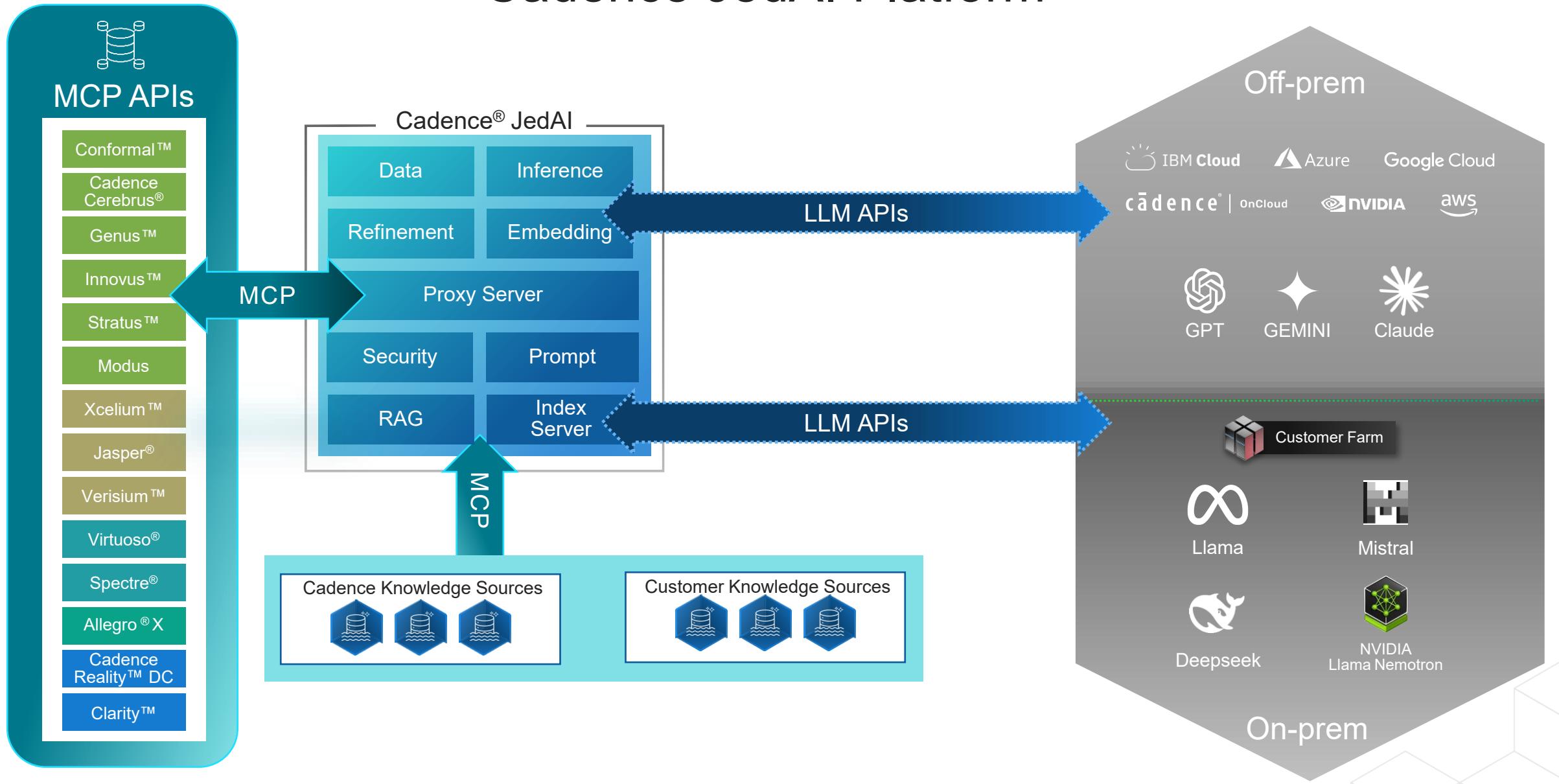


- ✓ **Tcl property generation** from natural language
- ✓ Specialized prompt for SVA generation
- ✓ Instant syntax checking, including signal names
- ✓ Toggle between different property types

Query examples

- **Check that** `sig_a` is equal to `DATA_MEM_SIZE` if initialized and all bits of `sig_b` are high
- **Check that** `sig_a` is different from `DATA_MEM_SIZE` if `sig_b` is high and any bit of `sig_c` is high, but not all bits
- **Check that** `sig_a` is always equal or more than one and `sig_b` is equal or less than four

Cadence JedAI Platform



New Protocols Are Enabling Plug-and-Play Agentic Systems

Anthropic



Model Context Protocol (MCP)

Google

A2A protocol

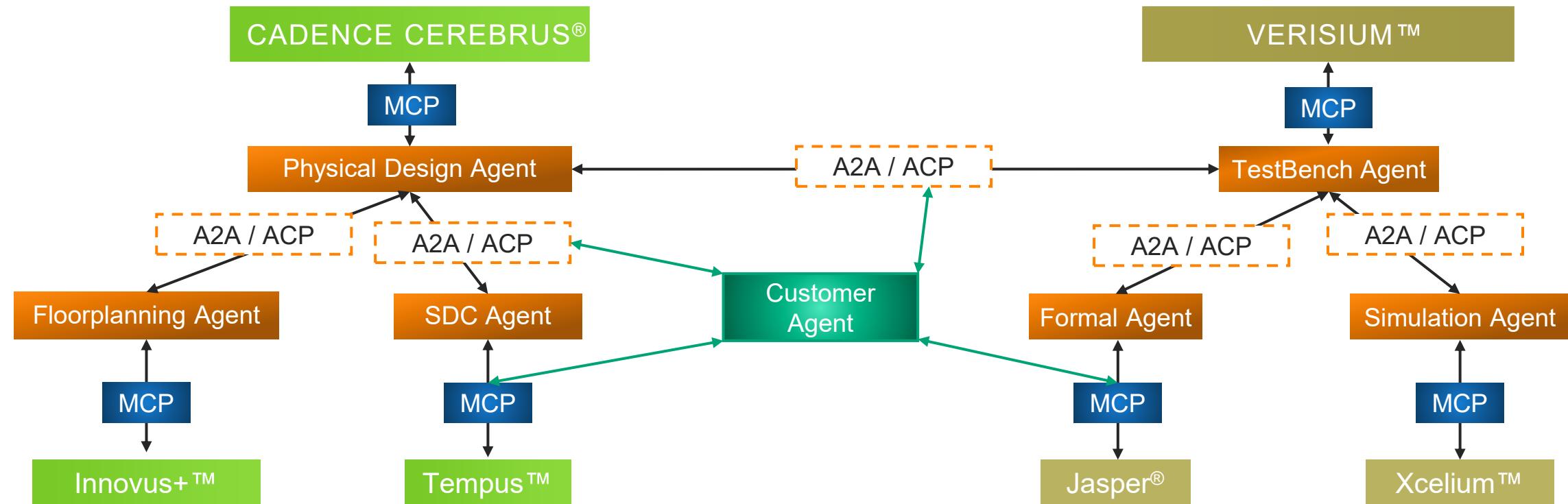


Agent2Agent (A2A)

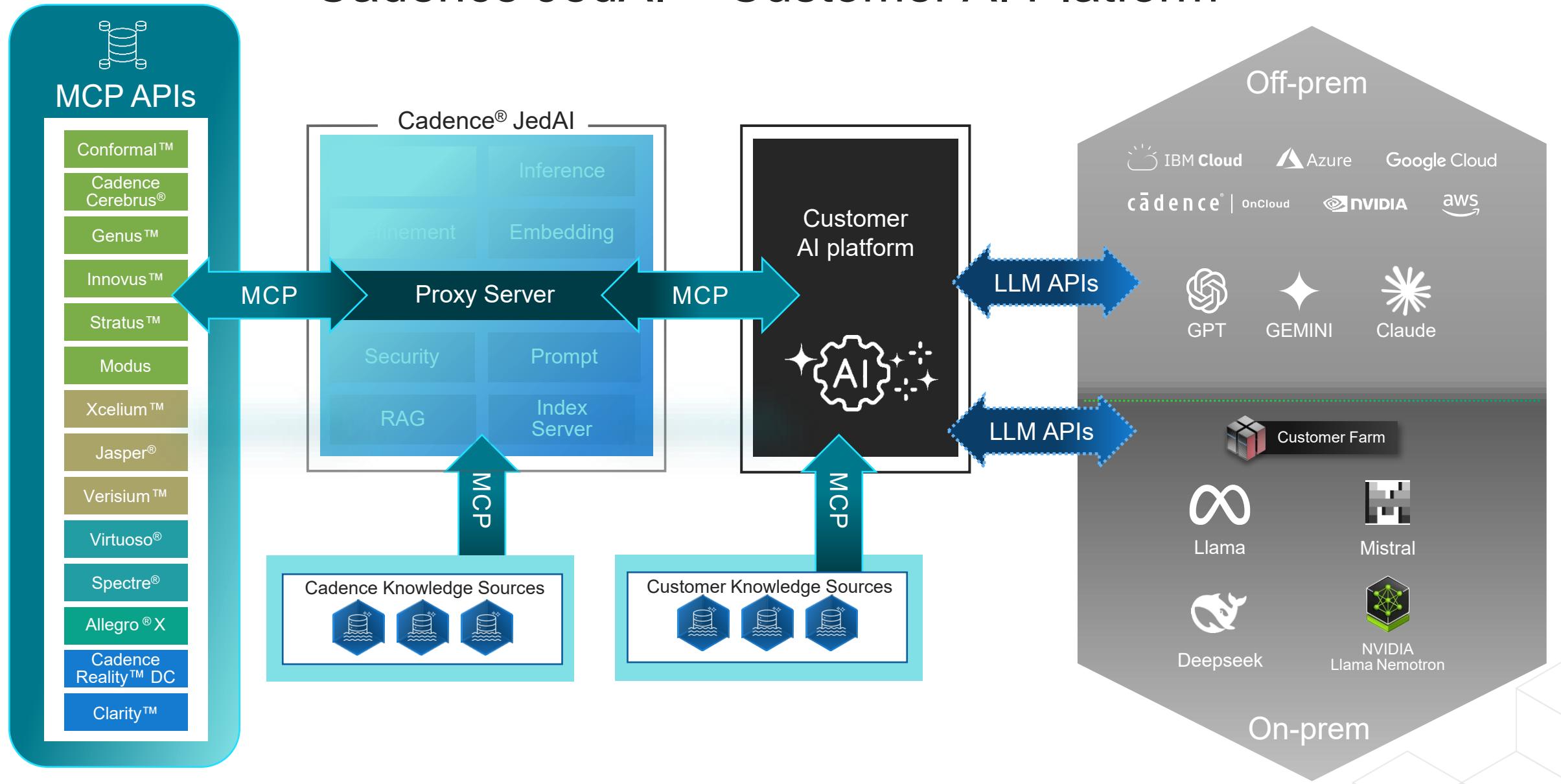
IBM



Agent Communication Protocol (ACP)



Cadence JedAI + Customer AI Platform





Jasper Assistant Script Generation Agent

LEVEL 3

Complex Reasoning

The screenshot shows the Jasper Assistant interface with the following components:

- Design Hierarchy:** A tree view showing a project structure with nodes like `pp (pp:(DATA_MEM_SIZE=8))`, `ig_parse (pp_ig_parse)`, `route_ctrl (pp_route_ctrl:PTR_W=3)`, etc.
- Property Table:** A table showing a list of properties with columns for Type, Name, Engine, Bound, Traces, and Tries.
- jedAI:** A chat window with a history of interactions, including a user request to prove liveness assertions and a generated Tcl script.
- Console:** A terminal window showing command-line interactions and logs.

- ✓ Jasper Tcl script generation from NL
- ✓ Under the hood reasoning about command formation
- ✓ Edit return
- ✓ Execute on the fly
- ✓ History



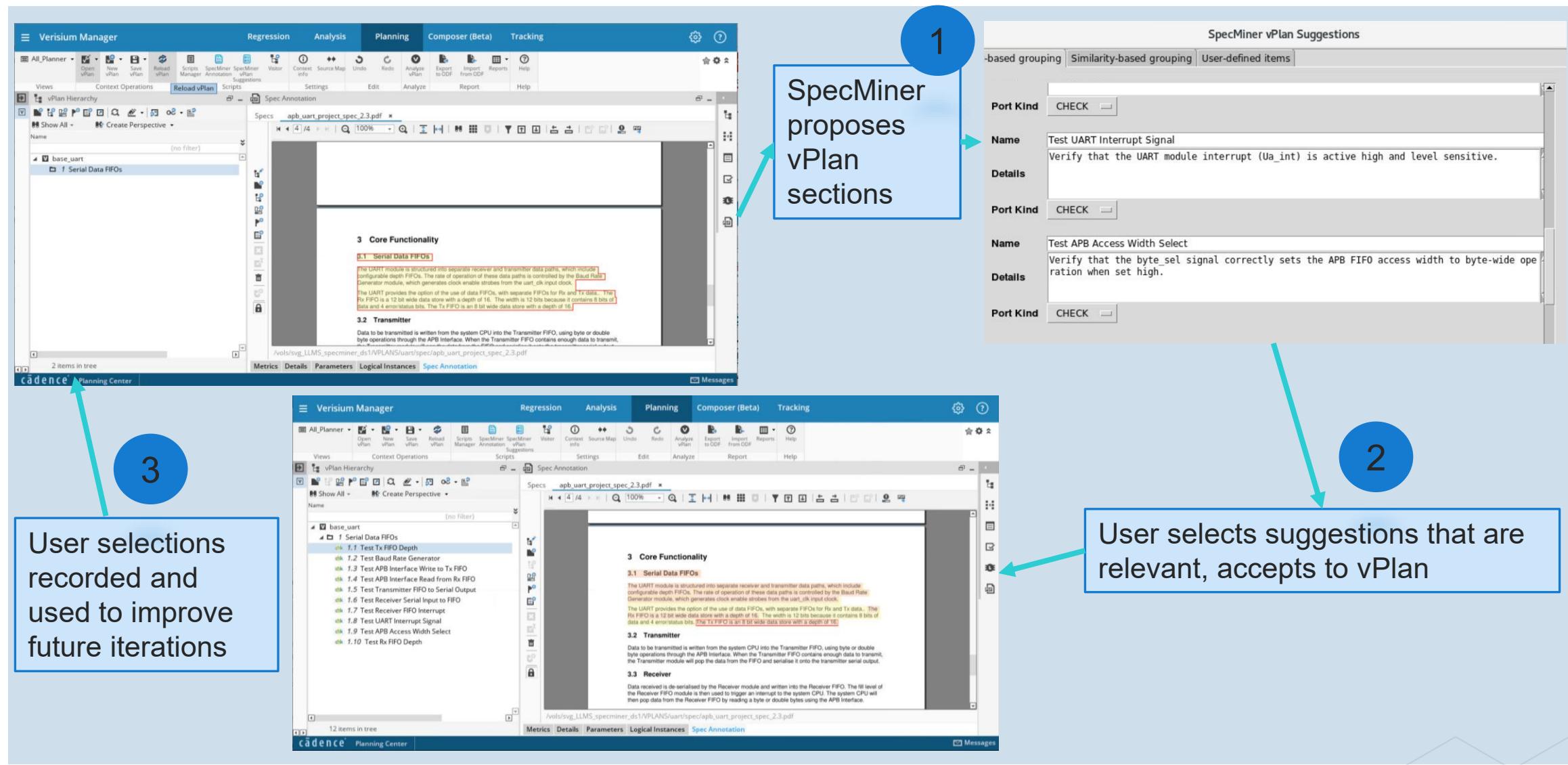
Reasoning

Edit

Execute

This slide contains forward-looking statements regarding Cadence's business or products. Actual results may differ materially from the information presented here.

Agentic AI in Verisium – Verification Plan Gen



1 SpecMiner proposes vPlan sections

2 User selects suggestions that are relevant, accepts to vPlan

3 User selections recorded and used to improve future iterations

SpecMiner vPlan Suggestions

- Port Kind:** CHECK
- Name:** Test UART Interrupt Signal
- Details:** Verify that the UART module interrupt (Ua_int) is active high and level sensitive.
- Port Kind:** CHECK
- Name:** Test APB Access Width Select
- Details:** Verify that the byte_sel signal correctly sets the APB FIFO access width to byte-wide operation when set high.
- Port Kind:** CHECK

Agentic Workflow Example



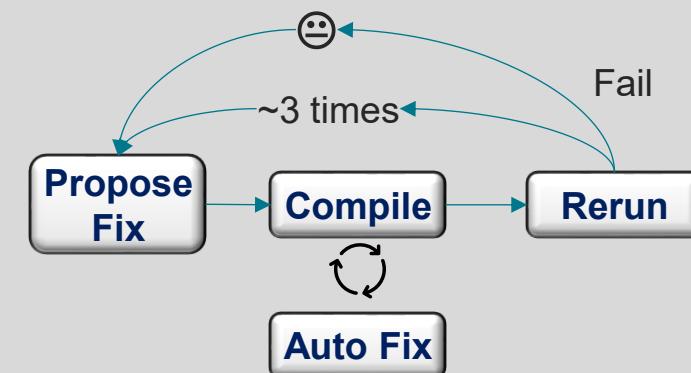
vManager AI Assistant – group failures and launch agents to root cause and fix.



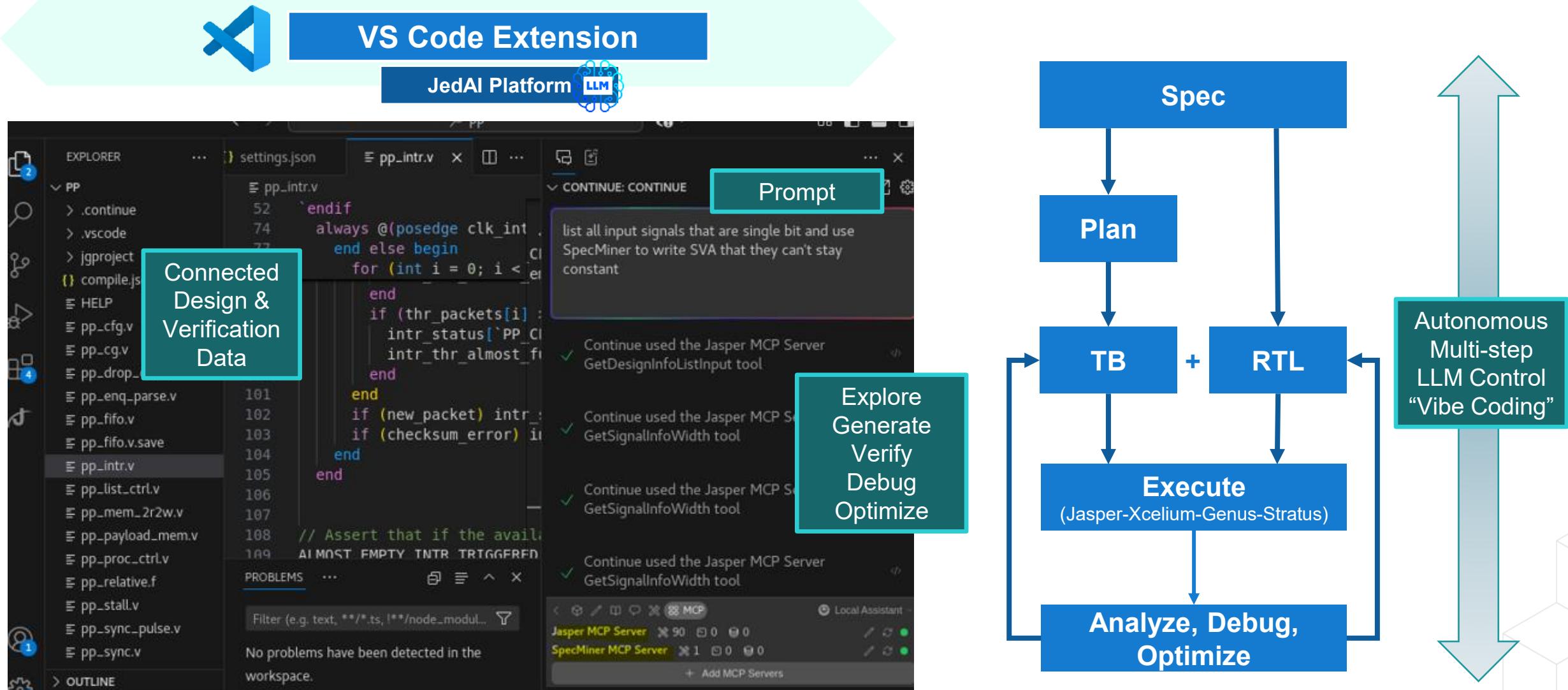
vDebug AI Assistant - root-cause the failure and generate summary of potential issues.



Frontend Silicon Agent – receives the issue description, proposes fix, applies fix, and loops until it is resolved.



Cadence Frontend Agent for RTL and Testbench Creation



Cadence Welcome's Chipstack

The World's First AI Verification Engineer

Business Impact

10x Faster

Testbench and coverage generation

60-80% Automation

Eliminates manual DV tasks

Production Ready

Deployed at leading customers
battle tested at scale

Enterprise Security

On-prem or cloud,
your IP stays secure



ChipStack AI Platform

Design
Understanding

Formal
Verification

Unit
Testing

UVM
Generation

AI Chat
Assistant

Coverage
Analysis

Agentic • Integrated • Automated

✓ **FULL BATCH MODE CLI**

Cadence Integration

Native tool integration



Xcelium

Execution Manager, VIPs



Jasper Formal

Property verification



vDebug

Debug and root cause



vManager

Verification management



VWDB / UCM/UCD

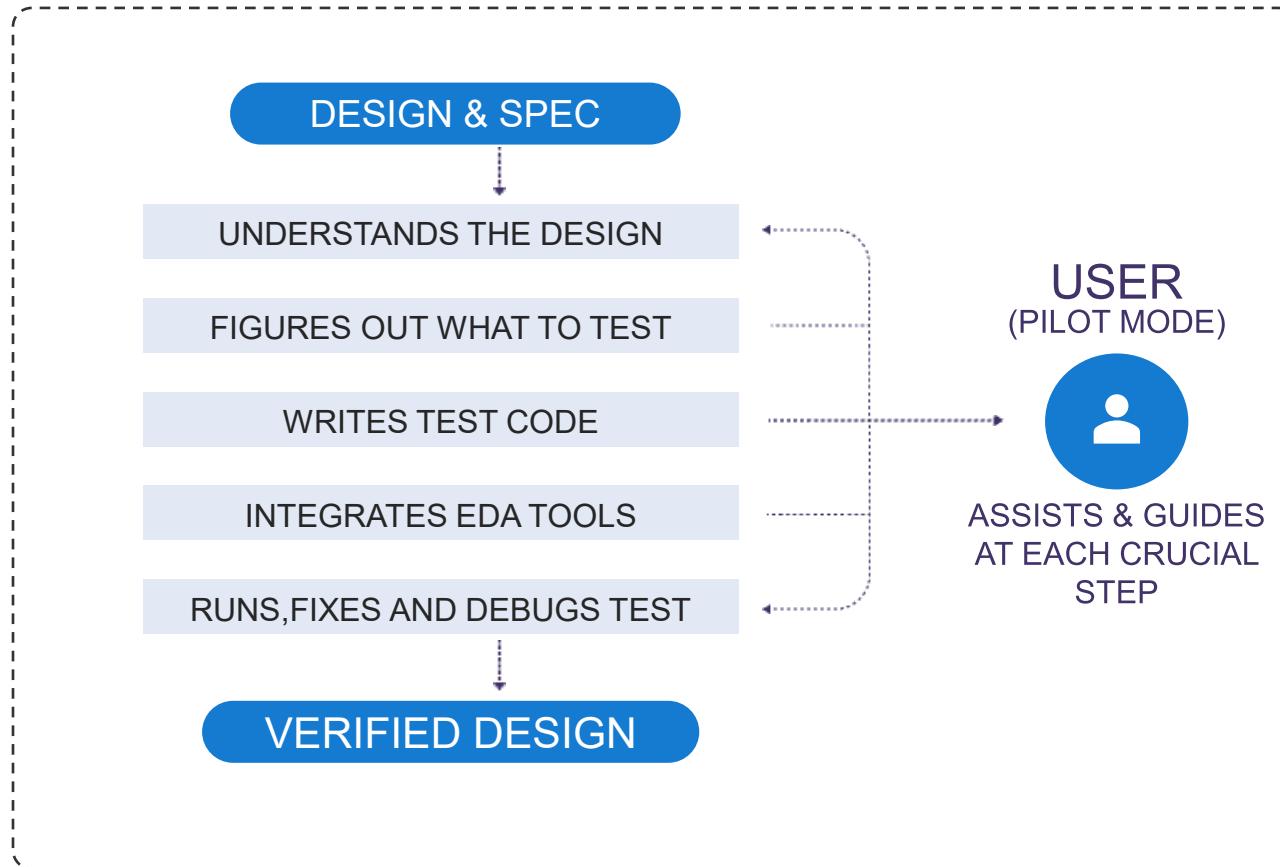
Waveform and coverage

From Design Understanding to Signoff — Fully Automated, Deeply Integrated

Reduce verification time by 10× • Scale your team instantly • Deploy in weeks, not months

Meet The World's First AI Chip Verification Engineer

Go from Design to 80% Coverage in Minutes!



Simulation Test Plan Scenarios

For all scenarios: [Generate](#)

Shared Code Testpoints

Select all [Add scenarios](#)

BasicPushAndRead

Description	Code	Results
BasicPushAndRead	Syntax Check	Success
UpdateAfterReadWithHaz		Failure
ShrinkOperationAndCredi		Failure
ConcurrentPushAndUpda		Warning
BackpressureAndFlowCor		Failure
ResetOutputsDefaultChec		Failure
ResetNoSpuriousActivity		Failure

BasicPushAndRead

Description: Code: Results: CODE: Syntax Check

Expand module

```
196 196 task automatic test_BasicPushAndRead
197 197 fork
198 198 begin : timeout_block
199 199 #(PER_TASK_TIMEOUT);
200 200 $display("Time: %0t",
201 201 Stimuli is not observed in the
202 202 test.", $time);
203 203 failed_tests.push_back();
204 204 end : timeout_block
205 205 begin : test_block
206 206 // Purpose: Verify the
207 207 -push and indexed read operation
208 208 -credit-based flow control,
209 209 -accurate credit reporting and
210 210 // Purpose: Verify the
211 211 -push and indexed read
212 212 -credit-based flow control
213 213 -accurate credit reporting
214 214
215 215
216 216
217 217
218 218
219 219
220 220
221 221
222 222
223 223
224 224
225 225
226 226
227 227
228 228
229 229
230 230
231 231
232 232
233 233
234 234
235 235
236 236
237 237
238 238
239 239
240 240
241 241
242 242
243 243
244 244
245 245
246 246
247 247
248 248
249 249
250 250
251 251
252 252
253 253
254 254
255 255
256 256
257 257
258 258
259 259
260 260
261 261
262 262
263 263
264 264
265 265
266 266
267 267
268 268
269 269
270 270
271 271
272 272
273 273
274 274
275 275
276 276
277 277
278 278
279 279
280 280
281 281
282 282
283 283
284 284
285 285
286 286
287 287
288 288
289 289
290 290
291 291
292 292
293 293
294 294
295 295
296 296
297 297
298 298
299 299
300 300
301 301
302 302
303 303
304 304
305 305
306 306
307 307
308 308
309 309
310 310
311 311
312 312
313 313
314 314
315 315
316 316
317 317
318 318
319 319
320 320
321 321
322 322
323 323
324 324
325 325
326 326
327 327
328 328
329 329
330 330
331 331
332 332
333 333
334 334
335 335
336 336
337 337
338 338
339 339
340 340
341 341
342 342
343 343
344 344
345 345
346 346
347 347
348 348
349 349
350 350
351 351
352 352
353 353
354 354
355 355
356 356
357 357
358 358
359 359
360 360
361 361
362 362
363 363
364 364
365 365
366 366
367 367
368 368
369 369
370 370
371 371
372 372
373 373
374 374
375 375
376 376
377 377
378 378
379 379
380 380
381 381
382 382
383 383
384 384
385 385
386 386
387 387
388 388
389 389
390 390
391 391
392 392
393 393
394 394
395 395
396 396
397 397
398 398
399 399
400 400
401 401
402 402
403 403
404 404
405 405
406 406
407 407
408 408
409 409
410 410
411 411
412 412
413 413
414 414
415 415
416 416
417 417
418 418
419 419
420 420
421 421
422 422
423 423
424 424
425 425
426 426
427 427
428 428
429 429
430 430
431 431
432 432
433 433
434 434
435 435
436 436
437 437
438 438
439 439
440 440
441 441
442 442
443 443
444 444
445 445
446 446
447 447
448 448
449 449
450 450
451 451
452 452
453 453
454 454
455 455
456 456
457 457
458 458
459 459
460 460
461 461
462 462
463 463
464 464
465 465
466 466
467 467
468 468
469 469
470 470
471 471
472 472
473 473
474 474
475 475
476 476
477 477
478 478
479 479
480 480
481 481
482 482
483 483
484 484
485 485
486 486
487 487
488 488
489 489
490 490
491 491
492 492
493 493
494 494
495 495
496 496
497 497
498 498
499 499
500 500
501 501
502 502
503 503
504 504
505 505
506 506
507 507
508 508
509 509
510 510
511 511
512 512
513 513
514 514
515 515
516 516
517 517
518 518
519 519
520 520
521 521
522 522
523 523
524 524
525 525
526 526
527 527
528 528
529 529
530 530
531 531
532 532
533 533
534 534
535 535
536 536
537 537
538 538
539 539
540 540
541 541
542 542
543 543
544 544
545 545
546 546
547 547
548 548
549 549
550 550
551 551
552 552
553 553
554 554
555 555
556 556
557 557
558 558
559 559
560 560
561 561
562 562
563 563
564 564
565 565
566 566
567 567
568 568
569 569
570 570
571 571
572 572
573 573
574 574
575 575
576 576
577 577
578 578
579 579
580 580
581 581
582 582
583 583
584 584
585 585
586 586
587 587
588 588
589 589
590 590
591 591
592 592
593 593
594 594
595 595
596 596
597 597
598 598
599 599
600 600
601 601
602 602
603 603
604 604
605 605
606 606
607 607
608 608
609 609
610 610
611 611
612 612
613 613
614 614
615 615
616 616
617 617
618 618
619 619
620 620
621 621
622 622
623 623
624 624
625 625
626 626
627 627
628 628
629 629
630 630
631 631
632 632
633 633
634 634
635 635
636 636
637 637
638 638
639 639
640 640
641 641
642 642
643 643
644 644
645 645
646 646
647 647
648 648
649 649
650 650
651 651
652 652
653 653
654 654
655 655
656 656
657 657
658 658
659 659
660 660
661 661
662 662
663 663
664 664
665 665
666 666
667 667
668 668
669 669
670 670
671 671
672 672
673 673
674 674
675 675
676 676
677 677
678 678
679 679
680 680
681 681
682 682
683 683
684 684
685 685
686 686
687 687
688 688
689 689
690 690
691 691
692 692
693 693
694 694
695 695
696 696
697 697
698 698
699 699
700 700
701 701
702 702
703 703
704 704
705 705
706 706
707 707
708 708
709 709
710 710
711 711
712 712
713 713
714 714
715 715
716 716
717 717
718 718
719 719
720 720
721 721
722 722
723 723
724 724
725 725
726 726
727 727
728 728
729 729
730 730
731 731
732 732
733 733
734 734
735 735
736 736
737 737
738 738
739 739
740 740
741 741
742 742
743 743
744 744
745 745
746 746
747 747
748 748
749 749
750 750
751 751
752 752
753 753
754 754
755 755
756 756
757 757
758 758
759 759
760 760
761 761
762 762
763 763
764 764
765 765
766 766
767 767
768 768
769 769
770 770
771 771
772 772
773 773
774 774
775 775
776 776
777 777
778 778
779 779
780 780
781 781
782 782
783 783
784 784
785 785
786 786
787 787
788 788
789 789
790 790
791 791
792 792
793 793
794 794
795 795
796 796
797 797
798 798
799 799
800 800
801 801
802 802
803 803
804 804
805 805
806 806
807 807
808 808
809 809
810 810
811 811
812 812
813 813
814 814
815 815
816 816
817 817
818 818
819 819
820 820
821 821
822 822
823 823
824 824
825 825
826 826
827 827
828 828
829 829
830 830
831 831
832 832
833 833
834 834
835 835
836 836
837 837
838 838
839 839
840 840
841 841
842 842
843 843
844 844
845 845
846 846
847 847
848 848
849 849
850 850
851 851
852 852
853 853
854 854
855 855
856 856
857 857
858 858
859 859
860 860
861 861
862 862
863 863
864 864
865 865
866 866
867 867
868 868
869 869
870 870
871 871
872 872
873 873
874 874
875 875
876 876
877 877
878 878
879 879
880 880
881 881
882 882
883 883
884 884
885 885
886 886
887 887
888 888
889 889
890 890
891 891
892 892
893 893
894 894
895 895
896 896
897 897
898 898
899 899
900 900
901 901
902 902
903 903
904 904
905 905
906 906
907 907
908 908
909 909
910 910
911 911
912 912
913 913
914 914
915 915
916 916
917 917
918 918
919 919
920 920
921 921
922 922
923 923
924 924
925 925
926 926
927 927
928 928
929 929
930 930
931 931
932 932
933 933
934 934
935 935
936 936
937 937
938 938
939 939
940 940
941 941
942 942
943 943
944 944
945 945
946 946
947 947
948 948
949 949
950 950
951 951
952 952
953 953
954 954
955 955
956 956
957 957
958 958
959 959
960 960
961 961
962 962
963 963
964 964
965 965
966 966
967 967
968 968
969 969
970 970
971 971
972 972
973 973
974 974
975 975
976 976
977 977
978 978
979 979
980 980
981 981
982 982
983 983
984 984
985 985
986 986
987 987
988 988
989 989
990 990
991 991
992 992
993 993
994 994
995 995
996 996
997 997
998 998
999 999
1000 1000
```

Accelerating Verification Schedules By Over 70%

Deployed at leading customer sites, used on complex, industry-standard project

Understanding large designs & complex specifications



Mental Model Agent

Enable engineers to deeply understand the design intent

CHIP DESIGNERS

Deploying formal verification at scale for design/DV



Formal Verification Agent

Augment formal verification engineers to hunt harder-to-reach bugs

DESIGNERS FV ENGINEERS

Designer-driven unit testing to reduce early bugs



Unit Testing Agent

Enable RTL design teams to deliver high-quality RTL via ease of unit testing

DESIGNERS

Accelerating DV workflows and coverage closure



UVM Agent

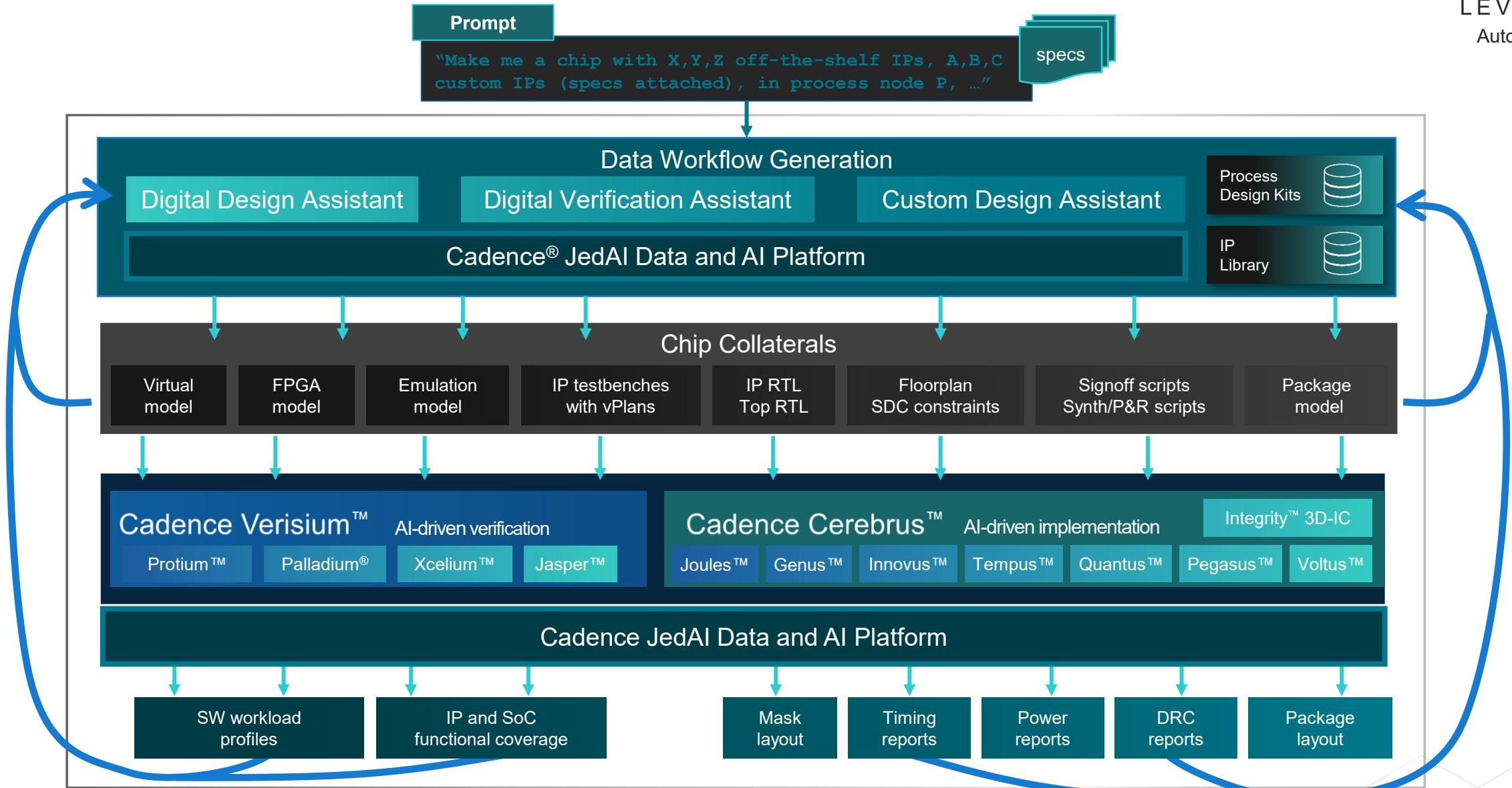
Testplans, functional covers, debugging, coverage closure, and more.

DV ENGINEERS AND DESIGNERS



LEVEL 5
Autonomy

Cadence Total Agentic AI for Silicon Design





cadence®

© 2025 Cadence Design Systems, Inc. All rights reserved worldwide. Cadence, the Cadence logo, and the other Cadence marks found at <https://www.cadence.com/go/trademarks> are trademarks or registered trademarks of Cadence Design Systems, Inc. Accellera and SystemC are trademarks of Accellera Systems Initiative Inc. All Arm products are registered trademarks or trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere. All MIPI specifications are registered trademarks or service marks owned by MIPI Alliance. All PCI-SIG specifications are registered trademarks or trademarks of PCI-SIG. All other trademarks are the property of their respective owners.