



Connect

From Code to Outcomes

OpenShift, the Platform for the Agentic SDLC

Siamak Sadeghianfar

Director of Product Management

Hybrid Platforms, Red Hat

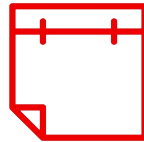


AI is transforming how we **build software**



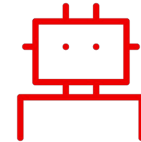
85%

of developers use
AI coding agents



51%

use AI coding agents
every day

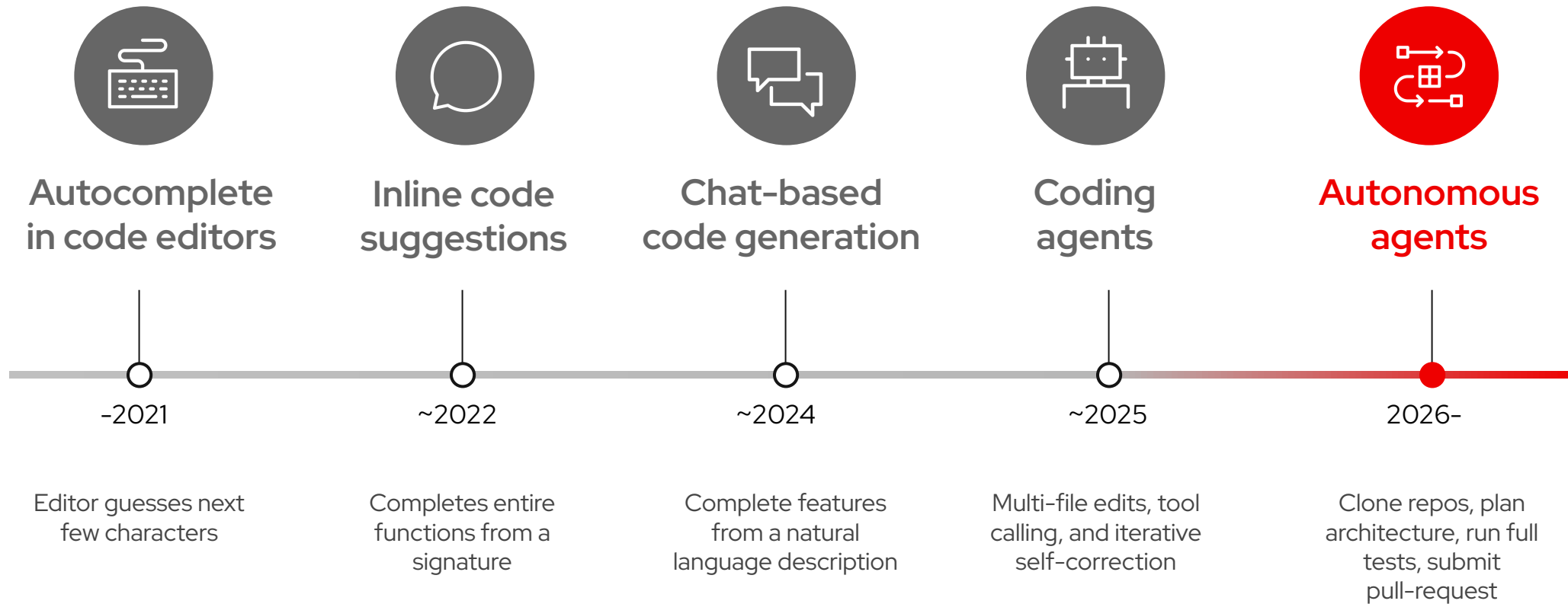


41%

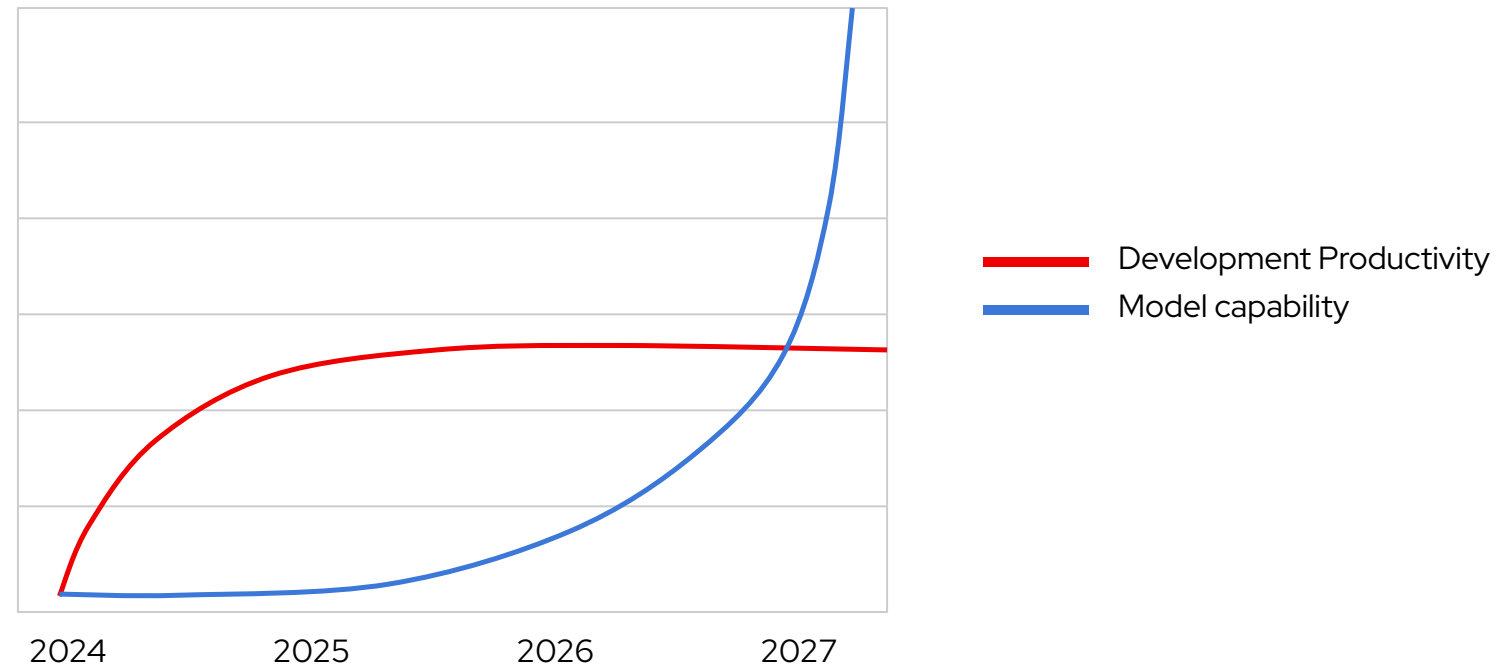
of new code is
AI-generated



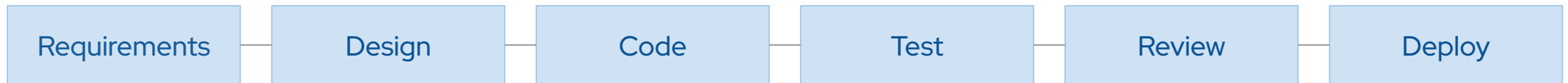
Evolution of AI Coding Assistants



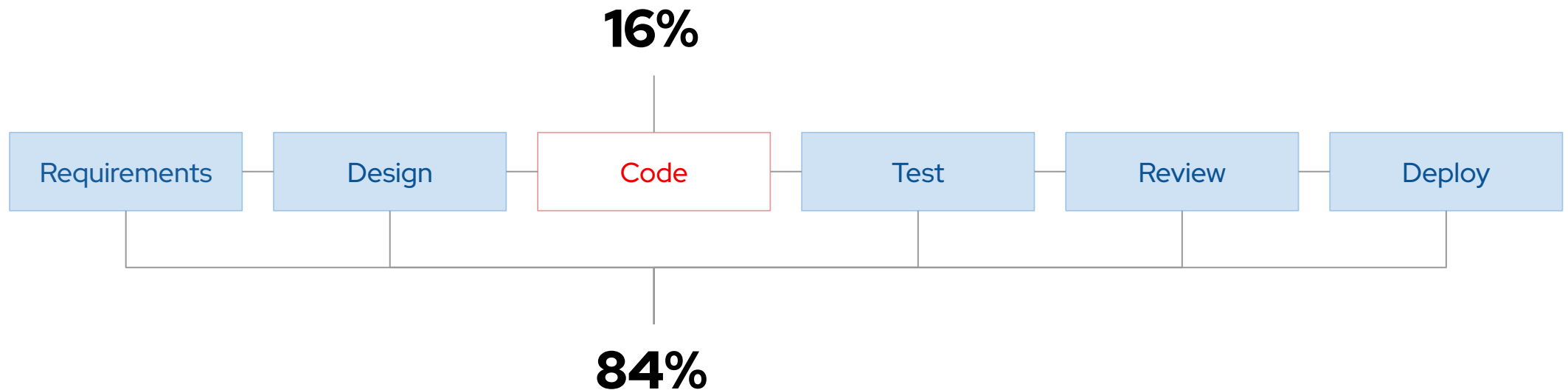
Model capabilities has increased exponentially but productivity gains in software development has hit a plateau



The Software Development Lifecycle (SDLC)



Code is no longer the bottleneck



...but the bottleneck was never coding, it was everything that comes before and after



How you use AI matters more than **whether** you use it

Vibe coding

Prompt, accept, ship. No tests, no review, no guardrails

Agentic engineering

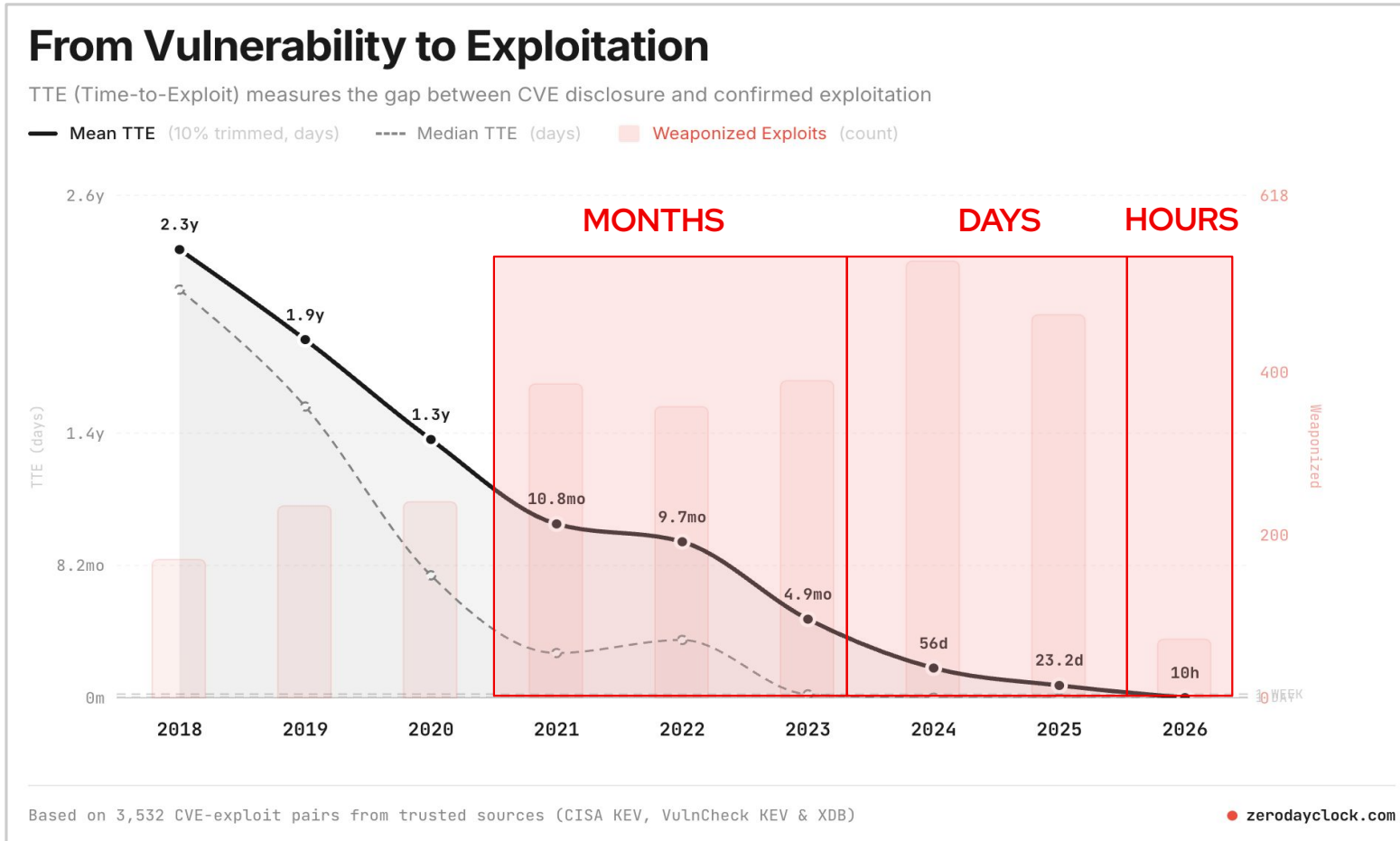
AI implements features within organization constraints, verified at every step

What's different?

Verification and human judgment surrounding the AI's output



AI is also reshaping the **security** landscape



+170%

New CVEs published last 3 months, against the same period last year



The New Scale of Software Development



Human Speed

Dozens of pull-requests per day

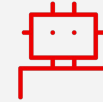
Static CI capacity planning

Manual approval gates

Trust by process and ownership

Handful of CVEs

45-90 days to patch CVEs



Machine Speed

Hundreds of pull-requests per day

Elastic on-demand CI

Automated policy-driven

Trust by verifications and provenance

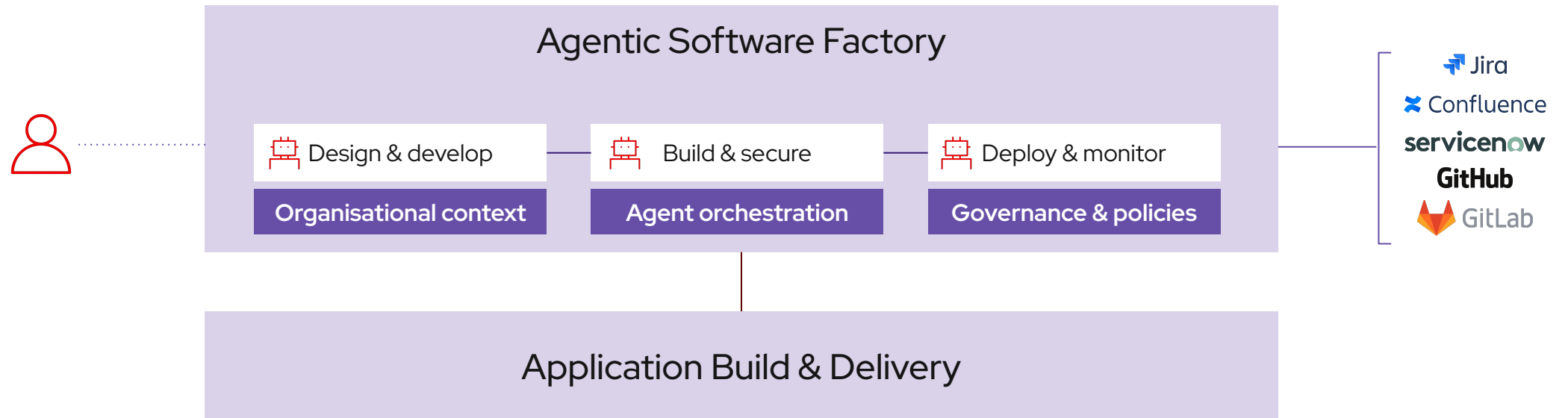
Thousand of CVEs

>14 days to patch CVEs



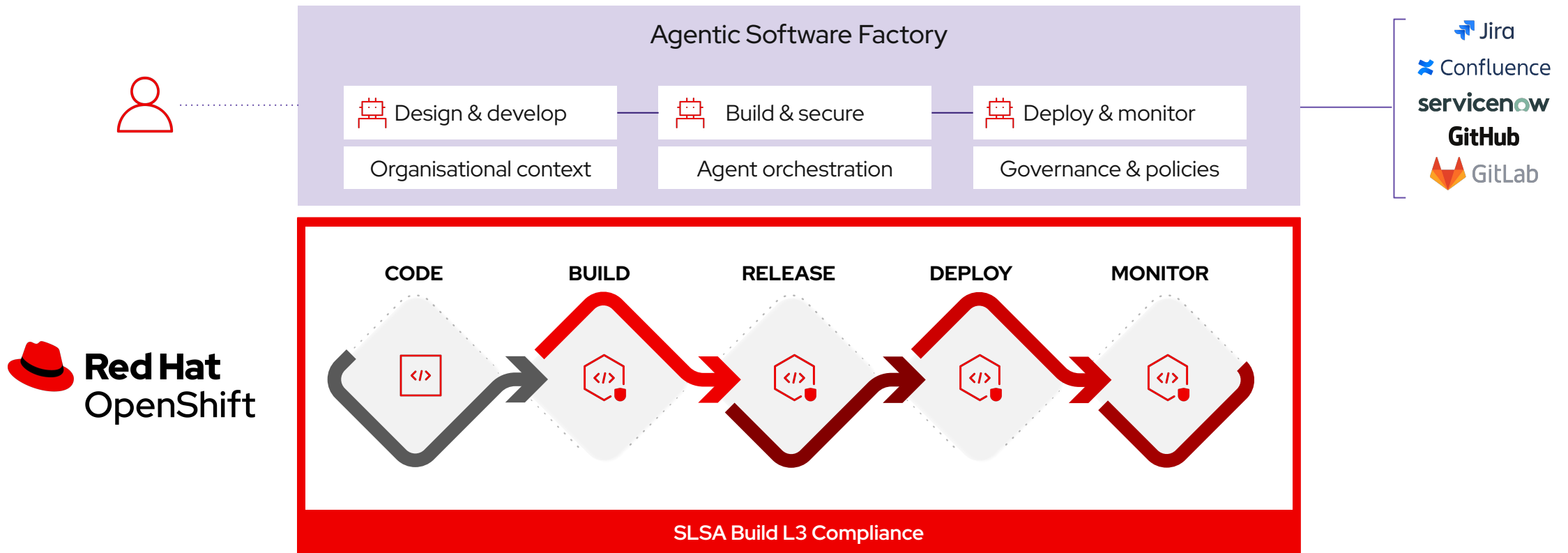
The System is Called **Software Factory**

Delivering software at machine speed



The System is Called **Software Factory**

Delivering software at machine speed

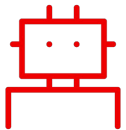


Design and Develop



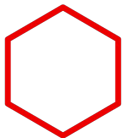
Define intent, commit the plan

Self-service template pre-configures repo, CI, and deploy targets with the developer's approved design in the Git repo



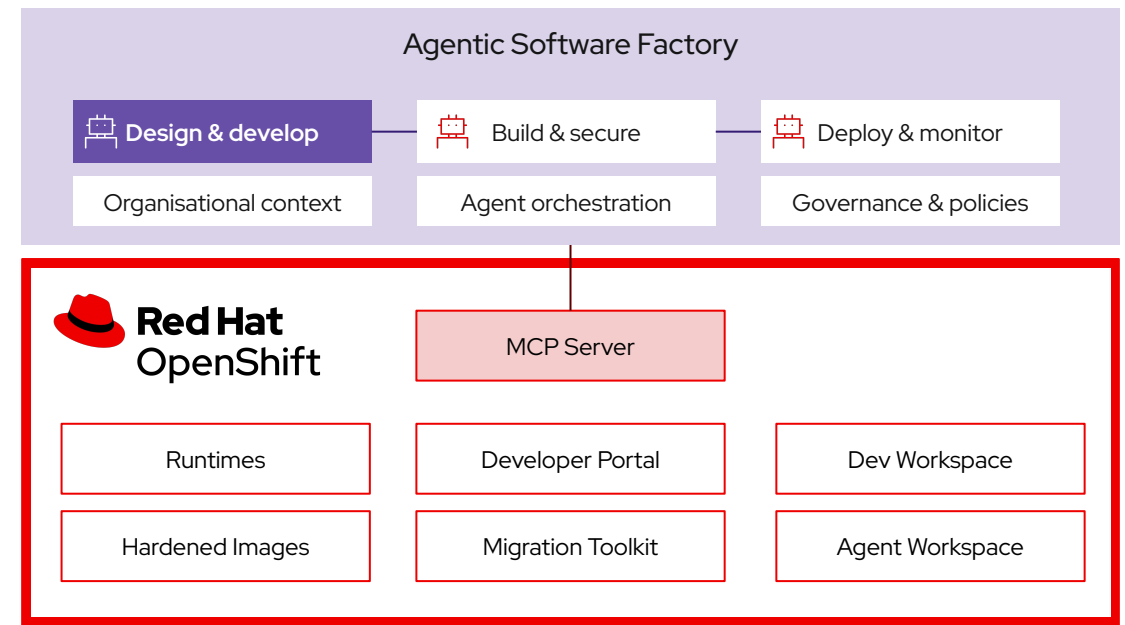
Implement

Agent generates the implementation in a consistent sandbox environment using trusted images and libraries



Submit

Code, tests, and send a pull-request to the application code repository for review



Build and Secure



Build and verify

Build pipelines test, run evals of agent output, scan for vulnerabilities and security policies



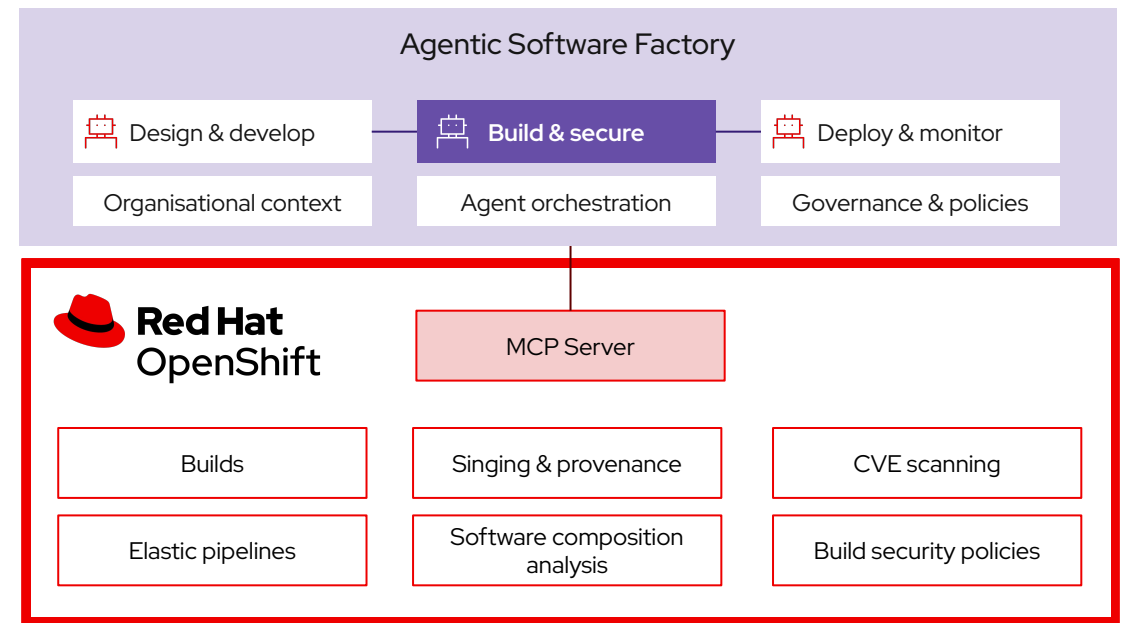
Sign and store

Images, pipelines, and security scan results are signed and store in tamper-proof record for provenance checks



Approve and gate

Security policies check provenance and compliance, a human approves high risk deployments



Deploy and Monitor



Canary deploy

Fraction of traffic first and roll out to entire traffic, with automated promotion and rollback



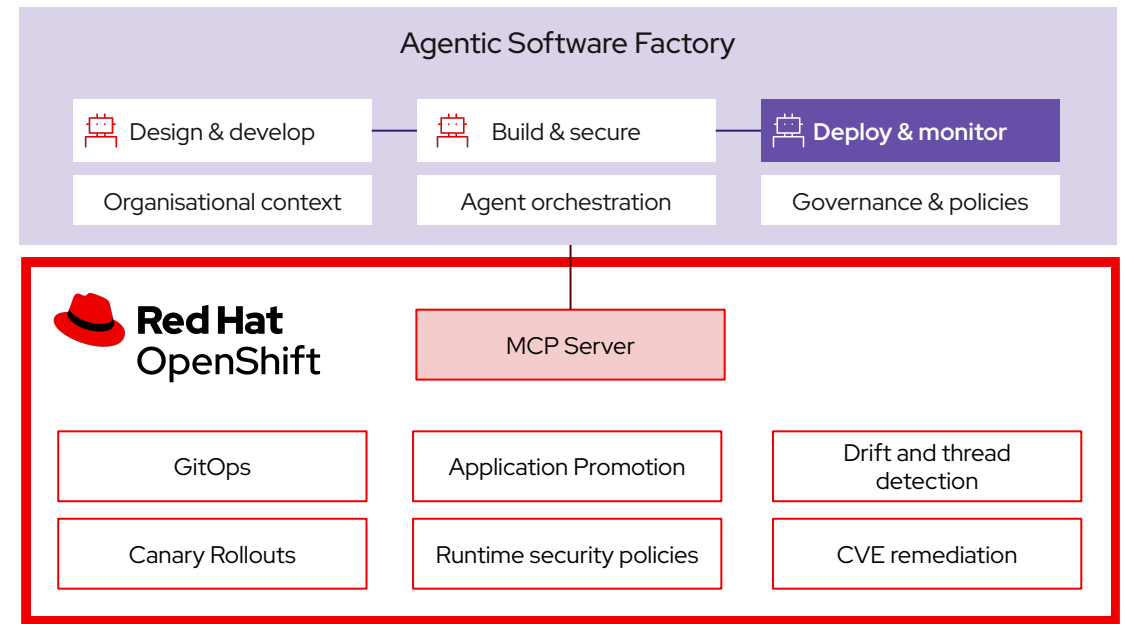
Monitor and act

Monitor application health and security posture, identify drift and security threats and take action



Maintain & remediate

A CVE lands for a dependence, Lightwell remediates, the same pipeline rebuilds, re-signs, redeploys.





Start with the factory.
Not the agents!



 red.ht/playbook





Connect

Thank you



linkedin.com/company/red-hat



facebook.com/redhatinc



youtube.com/user/RedHatVideos



x.com/RedHat



Agentic Software Factory

Delivering software at machine speed

