

DriftlessAF

An open source agentic reconciler framework proven at scale

Manfred Moser, Sr Principal DevRel Engineer

Manfred Moser

Victoria, BC, Canada

Sr. Principal DevRel Engineer at Chainguard

Open source hacker and advocate, author,
teacher, presenter, and host

manfred.moser@chainguard.dev





DriftlessAF

- Agentic reconciler framework
- Open source at <https://driftlessaf.ai/>
- Driftless means - no deviation from the desired state
- Agentic framework - AI-powered and traditional bots
- Reconciler like Kubernetes control plane for deployment and YAML-defined state

Reconciliation examples

Content of GitHub pull requests and content in GitHub repositories

- Ensure new release of a package is used
- Verify PR title and description
- Enforce test coverage and build success
- Improve source code and configuration files

Why we built DriftlessAF?

- Our event-driven pipeline buckled under the load.
- Every dropped event became a missed package or failed build.
- Engineers were on call for work that should have been automated.
- Open source ships faster every year — we couldn't keep up by hand.
- Security urgency made hours-not-months the new bar.
- We needed reconciliation, not retries.

Agents: not magic, not marketing – real benefits

- Not chatbots. Not autocomplete. Not “ask it for a script.”
- Workflows with state and goals
- Plan → Act → Verify → Learn → Repeat
- Executing real tasks, in real systems, continuously
- From prompts to pipelines

That's how reconcilers in DriftlessAF work.

Essential lesson: Humans in the loop and accountable

- Systems verify
- Agents propose
- Humans approve
- Trust through visibility
- Accountability isn't removed – it's amplified

What is included with DriftlessAF

- Complete operating framework
- Terraform recipes for Google Cloud deployment
- Work queue, management tools, observability, dashboards
- Agentic AI framework
- Numerous reconciler bots and examples
- All code written in Go

Workqueue - architecture notes

- URLs as task key, status data as JSON objects
- Stored in GCS buckets
- Very resilient, per region, optionally sharded
- queued, in-progress and dead-letter states as prefix
- Prioritization and automatic retry support
- Dispatcher to hand work to reconciler bots for

concurrent processing

Agentic AI infrastructure – models and prompts

- Meta-agent layer for Gemini, Claude, and OpenAI
- Executors manage conversation flow with LLM
- Secure prompt builder
 - Parameterized
 - Immutable
 - Support for dynamic binding of JSON, XML, YAML

Agentic AI infrastructure – tools and evals

- Robust tool-calling mechanism
 - Abstracts model specifics
- Agent evaluation framework for optimal choice and historic tracking including timing, cost, response quality, ...
- Judge – LLM evaluation of LLM output

Agentic AI infrastructure – advanced use

- Support RAG for improvements based on historical data and engineering input
- Observability with tracing and monitoring tools
 - Including metrics for model interactions

GitHub reconcilers - operations

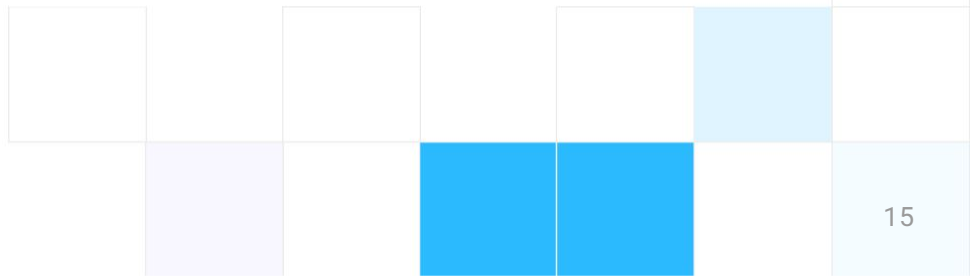
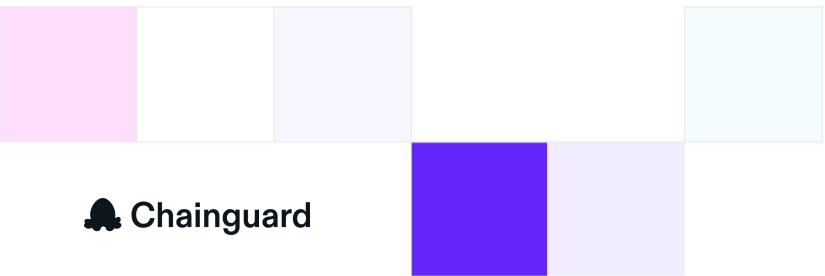
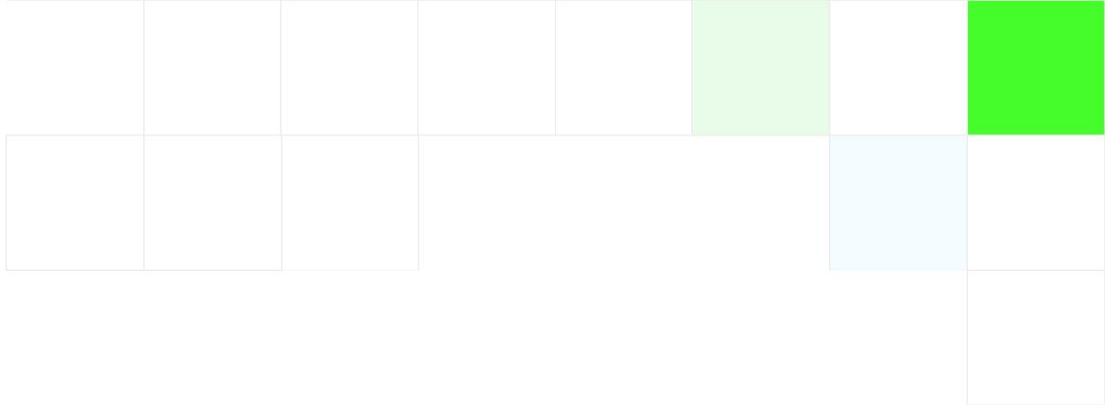
- Robust authentication and commit signing
- Supports pull requests and issues
- Find, create, update, close operations
- Embedded, structured metadata from Go templates in description
- Optimized , pooled clone management for git operations

GitHub reconcilers - file changes

- Supports file path analysis and file modifications
 - Source code
 - Test source code
 - Config files
- Status reporting via GitHub check runs

Other reconcilers

- Linear issues
- APK Linux packages
- OCI container images



Example reconcilers

- github-pr-validator - validate title and description
- github-pr-autofix - improve title and description
- github-issue-materializer - create issue
- github-path-modernizer - improve Go code using agent

It's demo time! Head on over to

<https://github.com/mosabua/driftlessaf-example>

Go modernization

Use `net/http` Status Code Constants

Use named constants from the `net/http` package instead of hardcoded numeric HTTP status codes. Named constants are self-documenting and less error-prone.

✓ Correct:

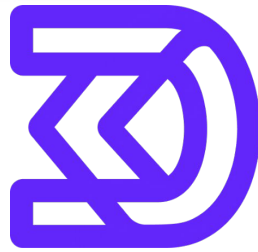
```
import "net/http"

http.Error(w, "not found", http.StatusNotFound)
w.WriteHeader(http.StatusOK)
if resp.StatusCode == http.StatusUnauthorized {
    // handle unauthorized
}
```

✗ Wrong:

```
http.Error(w, "not found", 404)
w.WriteHeader(200)
if resp.StatusCode == 401 {
    // handle unauthorized
}
```

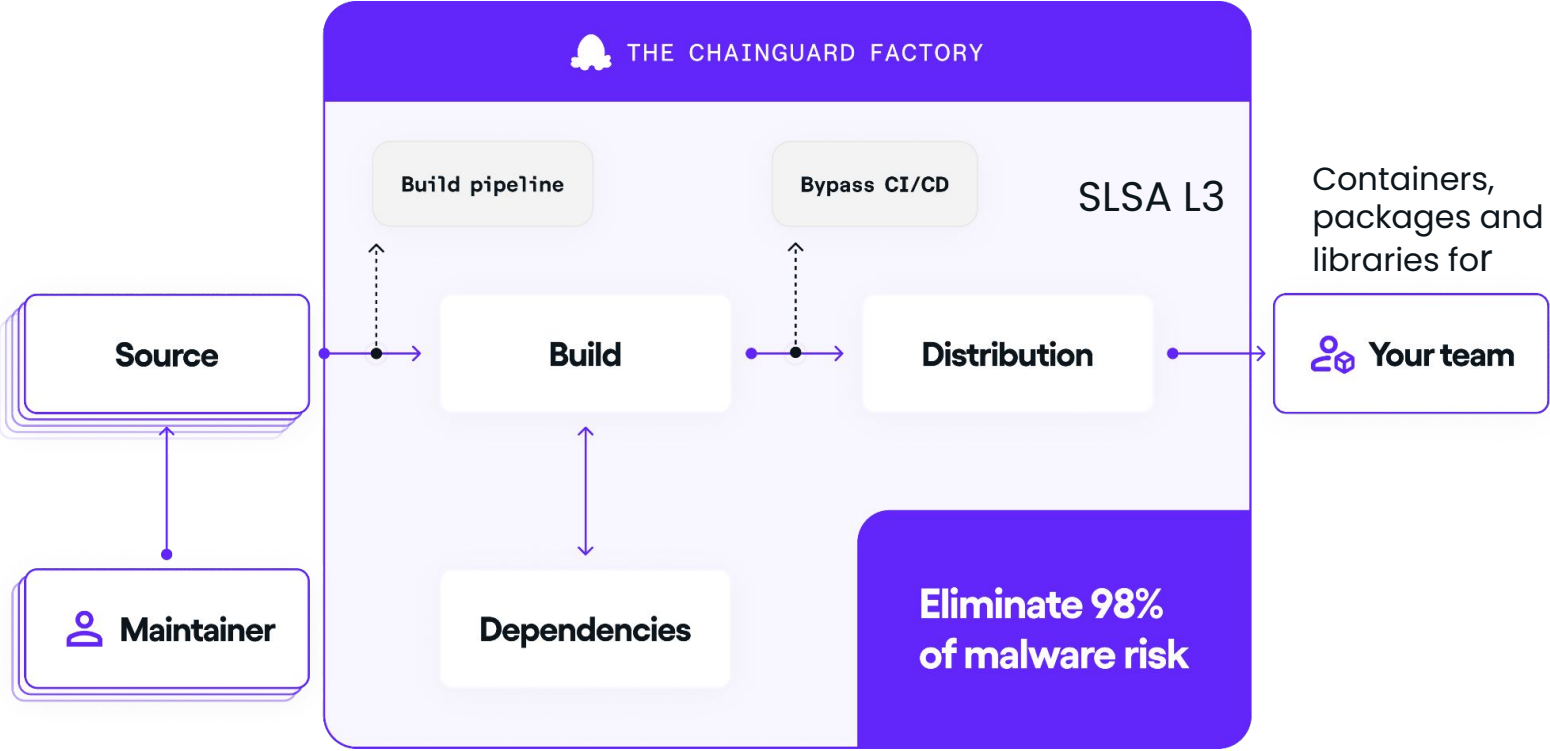
Audit: Search `.go` files for hardcoded status codes like `\b(200|201|204|301|302|400|401|403|404|405|409|500|502|503)\b` in contexts where `net/http` constants should be used instead.



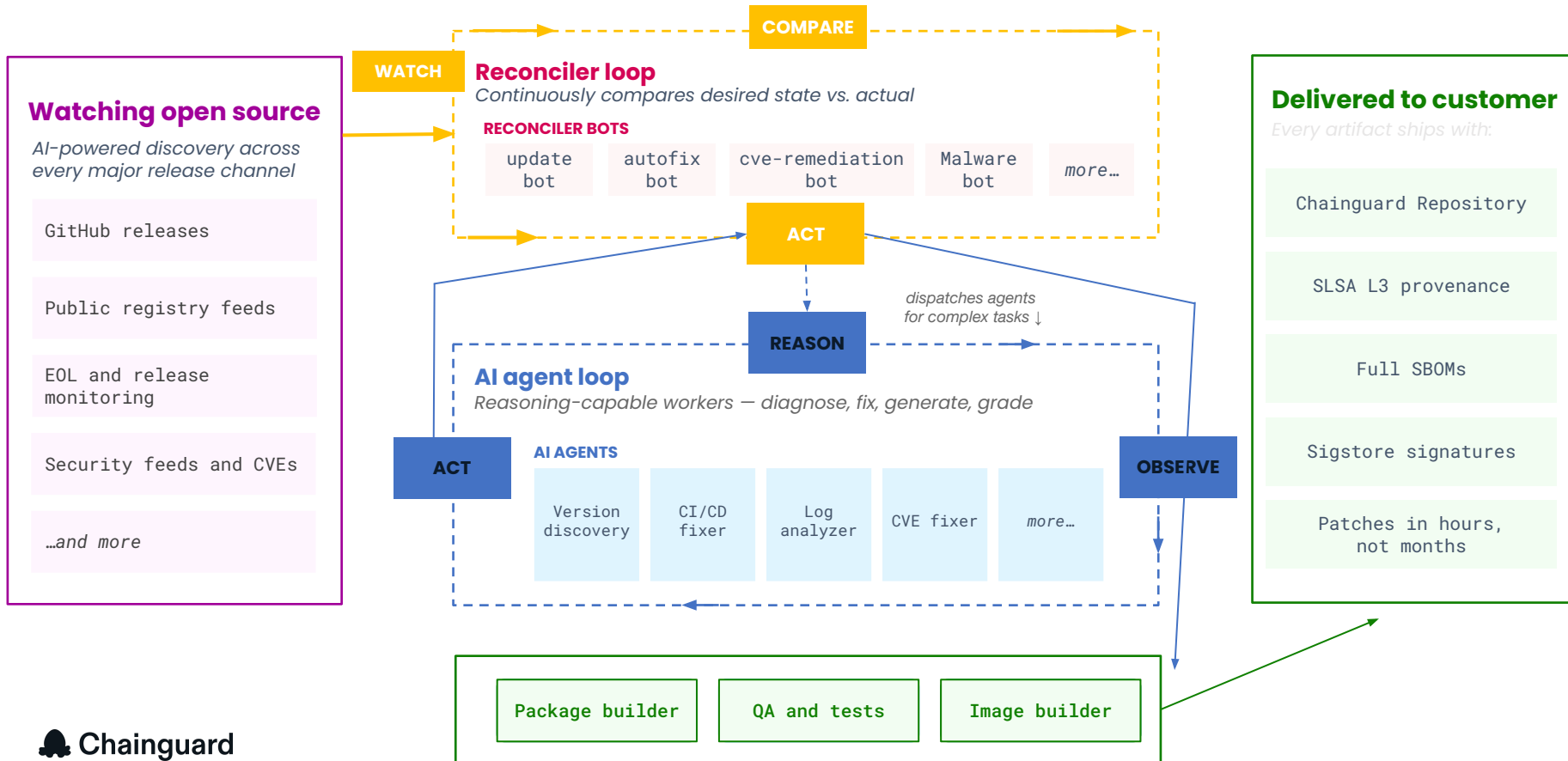
DriftlessAF

What we do at Chainguard..

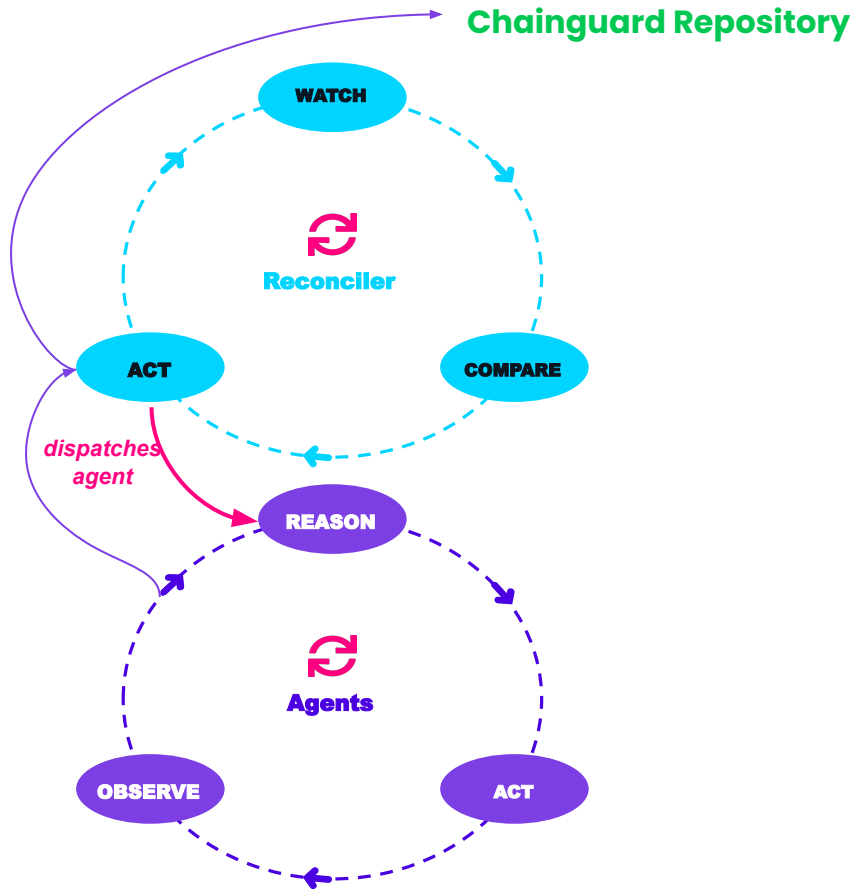
Chainguard Agentic Factory



The Chainguard Factory



DriftlessAF powering Chainguard Factory



⚙️ Reconciler bots

Specialized workers — each watches a specific type of drift

update bot

Bumps versions

autofix bot

Patches broken builds

cve-remediation bot

Patches CVEs

qackage bot

Improves tests

🤖 AI agents

Reasoning-capable — diagnose, fix, generate code, grade quality

Version discovery

Finds new releases

Fixer

Diagnoses and fixes builds

Log analyzer

Reads failure logs

Qackage improve

Writes better tests

Scale with DriftlessAF at Chainguard

- Over 2300 container images kept current 24x7
- 238,000+ container versions
- 610M+ container builds
- 1.5 million CVEs remediated
- Thousands of APK packages, npm packages, Python wheels, and Java packages every day

Wins from DriftlessAF

- Increased stability and throughput
 - Python fixer landed **>2100 PRs at <7500\$**
- Improved test quality and coverage
 - Initial use triggered **900+ PRs** for test improvements
- Faster, more reliable version discovery
 - **From 51% to 100%** coverage
- Build fixes for more libraries and packages from source

What else could it do?

- Deployments on prem or different cloud providers
- GitLab and others backend systems
- Support for other issue trackers
- Reconcilers for other languages
- More powerful code generation for upgrades
- Reconcile your docs against code and testing

Projects using DriftlessAF

- Chainguard Factory
- [code-reviewer](#)
- [zen](#)
- Your project next?

Resources

Source

- driftlessaf.ai
- <https://github.com/driftlessaf/go-driftlessaf>
- <https://github.com/driftlessaf/terraform-infra-reconcilers>

Additional docs with agent helper:

- <https://codewiki.google/github.com/driftlessaf/go-driftlessaf>
- <https://codewiki.google/github.com/driftlessaf/terraform-infra-reconcilers>

We are hiring for  **DriftlessAF** engineering and beyond



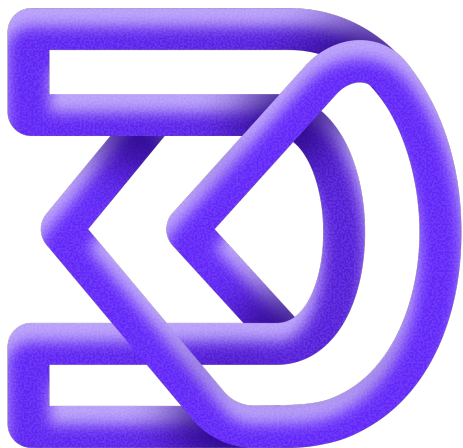
chainguard.dev/careers



Q&A

Ask now or
drop by at the Chainguard booth.





Thank you

Manfred Moser

manfred.moser@chainguard.dev

<https://driftlessaf.ai/>