Transform Mainframe Testing with Open Source Tools

Adam Munawar Rahman
Staff Software Developer @ IBM
Medium: @msradam

Marist Computing Conference 2025

What Are Mainframes?

Every credit card swipe, ATM withdrawal, airline booking.

30+ billion transactions per day. 90% of credit card transactions.
92 of the top 100 banks use IBM mainframes.

Problem: Testing tools stuck in the 1980s.

The Gap

Mainframe testing:

- Archaic interfaces, sparse documentation
- Jerry-rigged configurations
- No CI/CD integration

Modern load testing (Locust, k6):

- Modern scripting, CI/CD native
- Massive communities, excellent docs

Can we bring industry-standard tools to mainframes?

The Bridge: z/OS REST APIs

z/OS components now expose HTTP endpoints:

z/OSMF - System management (jobs, datasets, consoles)
z/OS Connect - CICS, IMS, Db2, batch programs
Zowe API Mediation Layer - Unified API gateway
CICS native - Direct REST/JSON support in CICS TS

This means: If a tool can make HTTP requests, it can test z/OS.

The door is open. Now we just need the right tools.

The Solution: Adapt, Don't Reinvent

Two proven tools. Two adaptations.

🐍 **Locust** (Python) - EA/DICE, AWS, Learnosity
→ Extended with py3270 plugin for terminal testing

⚡ **k6** (Go) - GitLab, Carvana, fuboTV, Olo
→ Ported to run natively on z/OS

The insight: Modern tools already exist.
We just made them work on mainframes.

How This Works: Two Patterns

Pattern 1: External control node (Locust, py3270)

```
Your Laptop/Jenkins  --HTTP-->  z/OS (z/OSMF, CICS, Zowe)
                     --Telnet->  z/OS (3270 terminals)
```

Pattern 2: Native on z/OS (k6)

```
z/OS USS (k6 binary)  --localhost-->  z/OSMF, CICS, Zowe
                                      (same LPAR)
```

External: Run from CI/CD. **Native:** Mainframe tests itself.

No agents. No mainframe installation for external tools.

Why These Tools?

Scale: Simulate millions of concurrent users
Flexibility: Dynamic load shapes that change throughout tests
Distributed: Run across multiple machines for massive load
Extensible: Rich plugin ecosystems for both tools
Observable: Real-time metrics and performance monitoring

14-30x less memory than JMeter. Modern, event-driven architecture.

Locust: Python Ecosystem

```
class MainframeUser(HttpUser):
    @task
    def submit_job(self):
        response = self.client.put("/zosmf/restjobs/jobs",
                                   data=jcl)
        assert response.json()["jobid"]
```

Used by EA/DICE for Battlelog (Battlefield). Endorsed by Flask creator Armin Ronacher.

Built py3270 plugin for 3270 terminal testing (locust-plugins PR #206).

k6: Native on z/OS ⚡

The challenge: Performance + portability. Run tests 24/7 on the mainframe itself.

The solution: Ported k6 to z/OS (added build flags, fixed dependencies).

```
export default function() {
  http.post('https://localhost/zosmf/restjobs/jobs', jobData);
}
```

```
$ k6 run test.js --vus 1000 --duration 24h
```

Compiled Go binary. Runs natively. No external dependencies.

Ported upstream - grafana/k6 PR #2892

Why This Matters

Your team already knows these tools.

Before:

- $50K+/year vendor licenses
- Mainframe specialists only
- Separate CI/CD pipelines

After:

- Open source: $0
- Any Python/JavaScript/Go developer
- One pipeline for everything

The real win: Transferable skills. Lower costs. No vendor lock-in.

Real-World Impact at IBM Z Test 🎯

Deployed in production:

- IBM Wazi as a Service - Locust testing z/OSMF APIs
- System testing - Terminal workflows with py3270
- Customer simulation - k6 running natively on z/OS

Testing capabilities:
REST APIs • 3270 Terminals • Batch Jobs • Mixed Workloads • CI/CD Gates

The proof: Three production environments. Zero vendor tools.

Try It Yourself

Install:

```
pip install locust py3270
# or
brew install k6  # macOS
# or
go install go.k6.io/k6@latest  # Any platform with Go
```

Read the journey:
Medium: @msradam

- "Swarming Stressed Servers" (Locust + z/OSMF)
- "Ticks by Telnet" (py3270 plugin)
- "Go-ing Native" (k6 porting process)

Start small: Pick one vendor test. Rewrite it. See the difference.

Questions?

Adam Munawar Rahman
Staff Software Developer @ IBM
M.S. Computer Engineering Student @ NYU Tandon

🌐 adamr.io
✉️ Medium: @msradam
💻 github.com/msradam

"Don't reinvent the wheel—install one through pip."