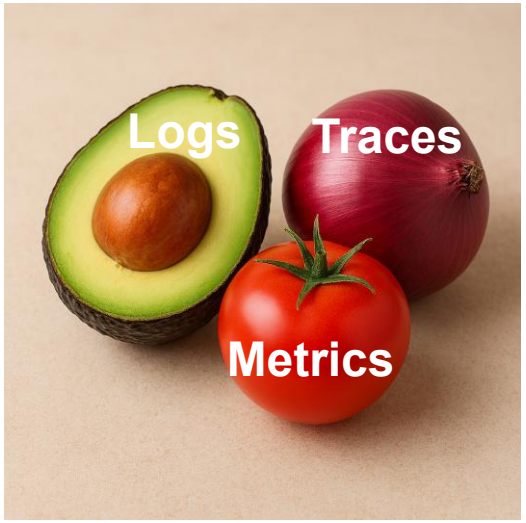
Du Jour Research Themes with Ciena: Innovating Observability and Security

François Tétreault

Observability



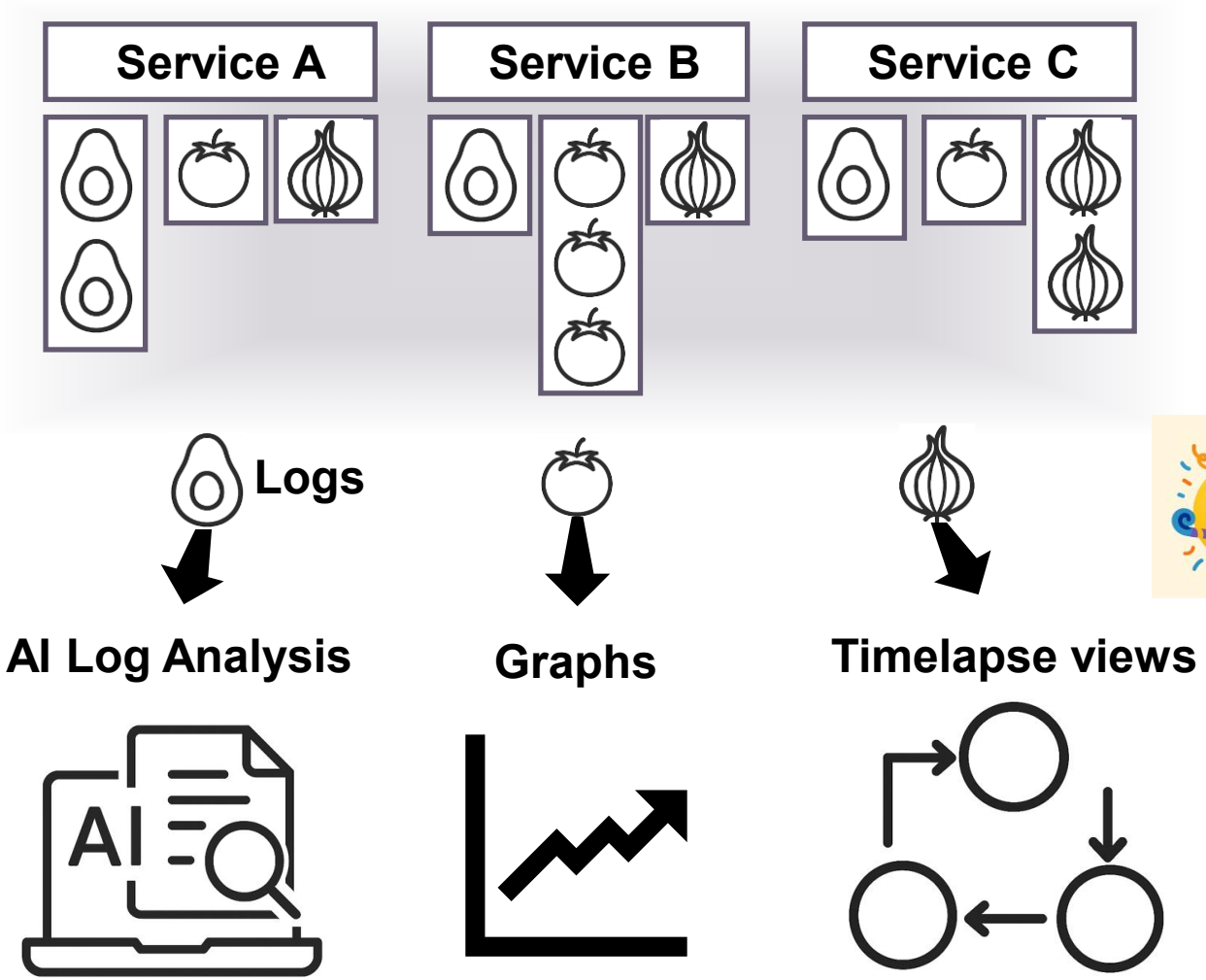
Unorganized Logs = Guacamole

Challenges

- Hard to see logs, metrics, traces from the mix
- Hard to engineer
- Hard to enforce guidelines

ciena

Observability



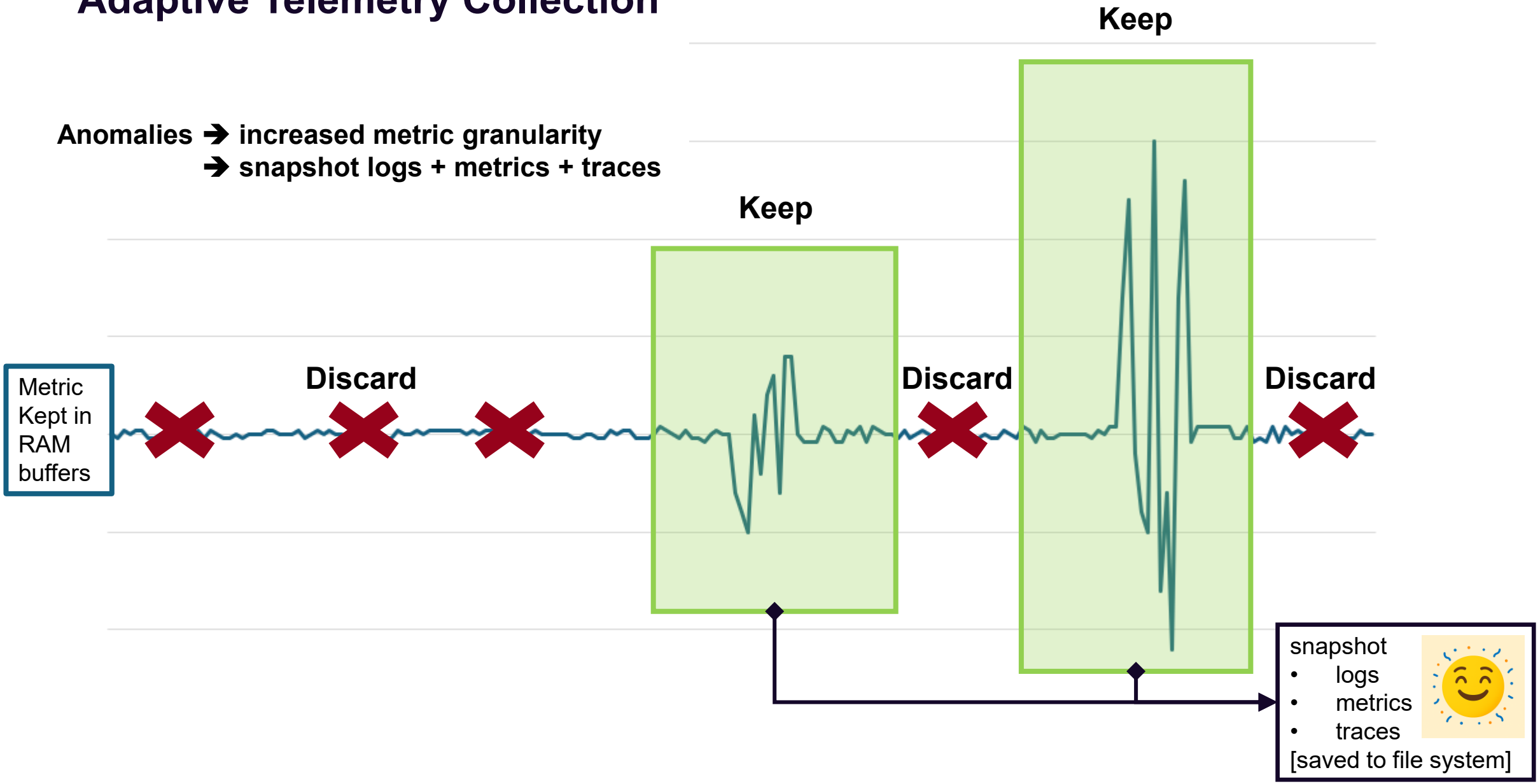
Benefits

- Telemetry data from each service are separated
- Logs, Metrics, Traces, are separated
- Easy to engineer
- Easy to ensure guidelines are followed

- Easy to export to APM solutions (e.g. Granafa Labs OpenTelemetry APM)
- Easy to generate graphs and visualisations
- Easier to analyse

Adaptive Telemetry Collection

Anomalies ➔ increased metric granularity
➔ snapshot logs + metrics + traces



Tools Security Risk Exposures

Code-Level Security Techniques

Are the tools in our toolbox prepared for that, are they still working ?

- Stack Canaries
- Obfuscation
- CFI: Control Flow Integrity
- ASLR: Address Space Layout Randomization
- DEP: Data Execution Prevention
- Code Signing
- Runtime Integrity Checks
- Memory Safety Techniques

Vulnerabilities, Bugs, Security Flaws

Are our tools secure? Could hidden vulnerabilities like Log4J exist?

- Fuzz Testing
- Penetration Testing
- Input Validation
- Attack Surface
- Secure Code Practices
- Threat Modeling
- Vulnerability Management
- Encryption and Data Protection
- Zero Trust Architecture

ciena

Tracing Summit 2025 presentation by Argonne National Laboratory

Tracepoint Factory: Automated LTTng Instrumentation & Babeltrace2 Plugin Generation from header files

Can this approach be used to:

- Determine which part of a library of package is used ? ➔ Compile out unused features
  - Decrease load size
  - Decrease attack surface

- Determine how an application uses an API ? ➔ Validate that the API is used correctly
  - Remove latent bugs
  - Help investigate crashes caused by misused APIs

ciena

Thank you!