11. Analytics & Reporting

Summary

- Embed end-to-end observability by instrumenting user actions, system events and infrastructure metrics with OpenTelemetry, choosing appropriate granularity vs. retention for GDPR and cost control.
- Centralize analytics in a scalable, schema-on-read data lake (e.g. ADLS Gen2 + Synapse), enforce encryption, RBAC and exportable raw data for compliance.
- Standardize on tools via infrastructure-as-code, regularly review costs and avoid lock-in.

Pending Decisions

- Choose primary analytics platform for system related data → maybe initially let's go with Datadog.
- Choose primary analytics platform for business related data.
- Data retention policies: exact durations for raw, aggregated, log and trace data.
- Sampling rates and aggregation levels to balance fidelity, cost and performance.
- Export/archival approach for long-term or escrowed datasets.

Open Items

- Define a unified event/metric taxonomy and naming conventions.
- Create infrastructure-as-code templates for analytics resources (Data Lake, Synapse, monitoring stacks).
- Instrument all services with OpenTelemetry and validate end-to-end trace correlation.
- Build cost-alerting dashboards on data ingestion volumes and query usage.
- Draft standard reporting dashboards and template queries for key stakeholders.
- Document GDPR data-export, modification and deletion workflows within analytics.
- Integrate external analytics exports into the central data lake for unified BI.

Overview

Define what success looks like for the platform and establish clear metrics to monitor progress. Ensure analytics and reporting support both technical and business stakeholders and are designed with scalability, security, and compliance (e.g. GDPR) in mind.

Data Collection

- **observability by design:**

    - Integrate observability into development practices from the start. Ensure all critical user actions, system events, and errors are logged with sufficient granularity
    - Collect performance metrics at both application and infrastructure levels
    - see also 7. Development & Testing | Monitor
- **granularity & retention:**
  Decide per use case if aggregated data suffices or if raw records are needed. Define and document data retention periods in line with regulatory compliance and business needs.
    - see also 3. Data Storage & Management
- **comprehensive logging:**
  Track logs, errors, and traces. Use a combination of *event sourcing* and *audit logging* where appropriate for traceability and compliance.
    - see also 2. Security & Compliance & 3. Data Storage & Management
    - a data lake type store can be so cheap, that from the pure storage perspective dumping stuff into it is a no brainer (compute & bandwith still needs to be calculated, e.g CPU is still required to log extra stuff...)
- **user behavior tracking:**
  Clearly define which user actions are important to track (*who, what, when, how*). Ensure tracking respects privacy and regulatory requirements.
    - again, see 2. Security & Compliance
- **open standards, platform neutrality base preference:**
  Prefer OpenTelemetry for system data collection, enabling flexibility in backend analytics platforms which can be ingested by various platforms.
- **trace correlation:**
  Ensure end-to-end trace correlation is possible for distributed systems.
    - see also 4. Microservices & Scalability & Performance & Reliability
- **external analytics tools:**
  When using third-party tools (e.g., Google Analytics, Amplitude), regularly assess their value and ensure raw data can be exported to our own storage for unified analysis, compliance and backup.

Data storage & processing

- **centralized analytics storage:**
  Use a scalable, schema-on-read data lake (e.g., Azure Data Lake Storage Gen2 + Synapse Analytics) for historical, batch, and BI reporting.
- **data privacy & security:**
  Ensure all analytics data is handled in compliance with GDPR
- **access control:**
  RBAC

Example tooling overview

Tool/Platform	Main Use	Data Model	Best For	Typical Users	Rough pricing estimate
Google Analytics	Web/marketing analytics	Session-based	Website traffic	Marketing / Product	• **GA4 Standard**: free with limits; pay-with-your-visitor's-data • **GA4 360**: nobody has that much money
Amplitude	Product analytics	Event-based	User behavior & retention	Product / UX	• **Starter**: free, up to 50,000 Monthly Tracked Users

					(MTUs) • **Plus**: from $49/month (up to 300,000 MTUs, more features) in May 2025, we may have had more than 400k unique visitors (calculation may be inaccurate) ⇒ more than $2,520/month • **Growth & Enterprise**: Custom pricing, contact sales
<app hosting platform default> e.g. Azure Application Insights	App monitoring / APM (Azure Application Insights also provides some GA/Amplitude features, but not so product/UX friendly)	Telemetry/events	App health & performance	Dev / DevOps (SRE)	• Azure Application Insights: ◦ **usage-based** (data ingested, retention, export, etc.). Example: $2.76 / GB of data ingested ◦ its not the best option for long-term large-scale logs. It is better for short-mid-term performance monitoring with some logs
Grafana	Visualization / APM / Logs / IRM	Any	Custom dashboards	Any	• **Free**: $0, limited usage (e.g., 10k metrics, 50GB logs, 3 users) • **Pro**: from $19/month + usage-based (some are included but then $8 per 1k metrics, $0.50/GB logs, $8/user) • **Advanced**: From $299/month, higher limits and support

Datadog & New Relic	Observability / APM / Logs / IRM	Metrics/events/logs/traces	Infra & app monitoring	Dev/ DevOps (SRE)	• Datadog: ◦ ~~**Free**: $0, up to 5 hosts, 1-day retention~~ (useless) ◦ **Pro**: $15/host/month (billed annually), $18 on-demand ◦ **Enterprise**: $23/host/month (billed annually), $27 on-demand ◦ Add-ons: Custom metrics, containers, etc. are extra. • New Relic: similar to Grafana, usage-based + user seats
<data lake> e.g. Azure Data Lake + Synapse / Data Explorer	Centralized analytics	Schema-on-read	Historical & batch analytics	Data Engineers / Business-with-tech-skills	• **Azure Data Lake Storage**: usage-based, e.g., $0.039/GB/month for storage (500 GB × $0.039/GB = $19.50/month) • **Azure Synapse Analytics**: Pay for reserved or on-demand SQL/data processing, e.g., $5/hour for a DWU100c SQL pool

Pricing example

Scenario 1

Azure Container Apps Environment (i.e. Kubernetes), 4 nodes

200 GB logs / month

5,000 active metric series (CPU, memory, disk, etc.) [hard to estimate ahead of time]

5 users (in the monitoring tool)

Tool	Logs (200GB)	Metrics	Users (5)	Total (est.)
Azure Application Insights	$552	Incl.	Incl.	$552
Grafana Cloud	$75	Free	$16	$91
New Relic	$35	Incl.	$396	$431
Datadog	$20	Incl.	Incl.	$204

Azure Application Insights

- **Data ingestion:** 200GB × $2.76/GB = **$552/month**
- **Users:** No extra charge for users

- **Total: $552/month**

Grafana Cloud (as a SaaS)

- **Logs:** (200GB - 50GB free) × $0.50/GB = $75/month
- **Metrics:** Free (under 10,000 series) *[hard to estimate ahead of time]*
- **Users:** 2 users above free tier × $8 = $16/month
- **Total: $91/month**

New Relic

- **First 100GB:** Free
- **Next 100GB:** 100 × $0.35 = $35
- **Users:** 4 additional full users × $99 = $396/month (Standard plan)
- **Total:** $35 + $396 = **$431/month**

Datadog

- **Hosts:** 4 nodes × ($15 Infra + $31 APM (traces, performance, errors, etc.)) = $184/month
- **Logs:** 200GB × $0.10/GB = $20/month (standard log retention)
- **Users:** No extra charge for users
- **Total:** $184 + $20 = **$204/month**

Scenario 2

Azure Container Apps Environment 4 nodes

300 GB logs / month

8000 metric series (CPU, memory, disk, etc.) [hard to estimate ahead of time]

5 users (in the monitoring tool)

+ 1 SQL database, 1 Redis cache, and 1 Load Balancer

Tool	Logs (300GB)	Metrics (8,000)	Users (5)	Hosts	Total (est.)
Azure Application Insights	$828	Incl.	Incl.	Any	$828
Grafana Cloud	$125	Free	$16	Any	$141
New Relic	$70	Incl.	$396	Any	$466
Datadog	$30	Incl.	Incl.	7	$352

Best Practices

- **scalability:**
  Design analytics pipelines and storage to handle growth in data volume and user base
- **open source & vendor neutrality:**
  Prefer open standards and tools (e.g., OpenTelemetry, perhaps Grafana) to avoid vendor lock-in and flexibility
- **cost management:**
  Regularly review analytics infrastructure costs
- **infrastructure-as-code:**
  Favor to use infrastructure-as-code (e.g. Terraform has providers for everything) for the provisioning of resources reproducibility and easier pivot (especially with AI boost)

Notes

- Grafana vs Datadog / New Relic**:**
  Grafana is open-source and flexible but requires more setup; Datadog / New Relic is proprietary, easier to start, but can be costly. Ever more reason to start from scratch with infra-as-code to have a semi-generic blueprint first, then the actual platform can come
- usage based vs host based: e.g. Grafana vs Datadog. See pricing overviews. Usage can be very attractive but do keep in mind the nature of highly scalable cloud environments: when a flood comes and things are not written and controlled correctly, credit cards will experience it too