Practical Operability Techniques for Teams

Matthew Skelton
Skelton Thatcher Consulting
skeltonthatcher.com / @SkeltonThatcher

Today

What is operability?
Modern logging
Run Book dialogue sheets
Endpoint healthchecks
Correlation IDs
User Personas for dashboards
You
Software Developer
Tester / QA
DevOps Engineer
Team Leader
Head of Department
Operability:
use modern logging, Run Book dialogue sheets, endpoint healthchecks, correlation IDs, and user personas as team collaboration techniques
About us

Matthew Skelton

Rob Thatcher

Co-founders at Skelton Thatcher Consulting
Team-first digital transformation
30+ organisations
UK, US, EU, India, China
We build modern capabilities by mentoring your teams
Team Guide to Software Operability
Matthew Skelton & Rob Thatcher
skeltonthatcher.com/publications
Download a free sample chapter
Practical Operability Techniques for Teams
What is operability?
Operability

making software work well in Production
Logging with Event IDs
logging with Event IDs

reduce time to detect problems
increase team engagement
increase configurability
enhance collaboration

#operability
<table>
<thead>
<tr>
<th>Date/Time</th>
<th>Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>3/21/2015</td>
<td>Delivered</td>
</tr>
<tr>
<td>10:25 am</td>
<td>Package delivered by U.S. Postal Service</td>
</tr>
<tr>
<td>8:48 am</td>
<td>At U.S. Postal Service facility</td>
</tr>
<tr>
<td>3/20/2015</td>
<td>Delivered</td>
</tr>
<tr>
<td>9:14 am</td>
<td>In transit</td>
</tr>
<tr>
<td>7:30 am</td>
<td>In transit to U.S. Postal Service</td>
</tr>
<tr>
<td>4:14 am</td>
<td>Departed FedEx location</td>
</tr>
<tr>
<td>3/19/2015</td>
<td>Delivered</td>
</tr>
<tr>
<td>4:47 pm</td>
<td>Arrived at FedEx location</td>
</tr>
<tr>
<td>5:04 pm</td>
<td>Shipment information sent to FedEx</td>
</tr>
<tr>
<td>3/16/2015</td>
<td>Delivered</td>
</tr>
</tbody>
</table>

Event ID: {Delivered, InTransit, Arrived}

search by event
**Transaction Trace**

**Correlation ID**

61299995896239476896

---

<table>
<thead>
<tr>
<th>Date/Time</th>
<th>Activity</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>3/21/2015 - Sat</td>
<td>Delivered</td>
<td>LOS ANGELES, CA</td>
</tr>
<tr>
<td>10:25 am</td>
<td>Package delivered by U.S. Postal Service</td>
<td></td>
</tr>
<tr>
<td>8:48 am</td>
<td>At U.S. Postal Service facility</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Arrived at local Post Office - Allow one to two additional days for delivery</td>
<td></td>
</tr>
<tr>
<td>3/20/2015 - Fri</td>
<td>In transit</td>
<td>LOS ANGELES, CA</td>
</tr>
<tr>
<td>9:14 am</td>
<td>In transit to U.S. Postal Service</td>
<td></td>
</tr>
<tr>
<td>7:30 am</td>
<td>Shipment information sent to U.S. Postal Service</td>
<td></td>
</tr>
<tr>
<td>4:14 am</td>
<td>Departed FedEx location</td>
<td>FEDEX SMARTPOST CHINO, CA</td>
</tr>
<tr>
<td>3/19/2015 - Thu</td>
<td>In transit</td>
<td>LOS ANGELES, CA</td>
</tr>
<tr>
<td>4:47 pm</td>
<td>Arrived at FedEx location</td>
<td>FEDEX SMARTPOST CHINO, CA</td>
</tr>
<tr>
<td>3/16/2015 - Mon</td>
<td>Shipment information sent to FedEx</td>
<td></td>
</tr>
</tbody>
</table>

Select time zone Local Scan Time
How many distinct event types (state transitions) in your application?
represent distinct states
enum

Human-readable sets: unique values, sparse, immutable

C#, Java, Python, node
(Ruby, PHP, ...)
public enum EventID
{
    // Badly-initialised logging data
    NotSet = 0,
    // An unrecognised event has occurred
    UnexpectedError = 10000,

    ApplicationStarted = 20000,
    ApplicationShutdownNoticeReceived = 20001,

    MessageQueued = 40000,
    MessagePeeked = 40001,

    BasketItemAdded = 60001,
    BasketItemRemoved = 60002,

    CreditCardDetailsSubmitted = 70001,

    // ...
}
BasketItemAdded = 60001
BasketItemRemoved = 60002
log using Event IDs with a modern ‘structured logging’ library
example:

https://github.com/EqualExperts/opslogger

Sean Reilly
@seanjreilly
SUPERCHARGED VIDEO ADVERTISING DELIVERY

Faster delivery and better collaboration for brands, broadcasters and everyone in between.
Example: video processing

On-demand processing of TV and mobile streaming adverts

Ad-agency → TV broadcaster

High throughput

Glitch-free video & audio
Example: video processing

Discover processing bottlenecks
Trigger alerts via LogEntries / HostedGraphite
Report on KPIs
Target areas for improvement
Honeycomb

http://honeycomb.tv/
Run Book dialogue sheets
Run Book dialogue sheets

Checklists for typical operational considerations

Team-friendly exploration
System characteristics

Hours of operation

During what hours does the service or system actually need to operate? Can portions or features of the system be unavailable at times if needed?

Hours of operation - core features

(e.g. 03:00-01:00 GMT+0)

Hours of operation - secondary features

(e.g. 07:00-23:00 GMT+0)

Data and processing flows

How and where does data flow through the system? What controls or triggers data flows?
(e.g. mobile requests / scheduled batch jobs / inbound IoT sensor data )

...
Run Book / System Operation Manual

Service or system overview

Business overview
What business need is met by this service or system? What expectations do we have about availability and performance? (e.g. Provides reliable automated reconciliation of logistics transactions from the previous 24 hours)

Technical overview
What kind of system is this? Web-connected order processing? Back-end batch system? Internal HTTP-based API? ETL control system?
Endpoint health checks
WAXING
LEAST PAINFULL
METHOD
endpoint health checks

Every runnable app/service/daemon exposes `/status/health`

An HTTP GET to the endpoint returns:
- 200 – "I am healthy"
- 500 – "I am sick"
Each component is responsible for determining its own health and viability – this is very contextual
endpoint healthchecks

Use JSON as a response type – parsable by both machines and humans!
/status/health
endpoint healthchecks

For databases and other non-HTTP components, run a lightweight HTTP service in front of the component 200 / 500 responses
Infrastructure status

- Tutum
- CloudFront
- EC2 US East
- Production - Healthcheck

No response from healthcheck
Correlation IDs
‘Unique-ish’ identifier for each request
Passed through downstream layers
Synchronous HTTP:

X-HEADER e.g. \texttt{X-trace-id}
\texttt{X-trace-id: 348e1cf8}

If header is present, pass it on

(Yes, \texttt{RFC6648}, but this is internal only)
Asynchronous (queues, etc.):

Message Attributes, *name*:value pair

* e.g. "trace-id":"348e1cf8"

AWS SQS: `SendMessage()` / `ReceiveMessage()`

Log the Correlation ID if present
Example: electronic trading

High speed, low latency
Trading options & derivatives
Connected to stock exchanges
Sub-millisecond timings
> £1 million per day traded
Correlations IDs for trading

Evidence for timely operation
Help identify bottlenecks
Target areas for perf tuning
Identify race conditions
Increase operability
Example: OpenTracing / PCF

3 tracing elements:
TraceID, SpanID, ParentSpan
"X-B3-TraceId" "X-B3-SpanId"
"X-B3-ParentSpan"
Example: OpenTracing / PCF

Always log the TraceID as-is
Log calling SpanID as ParentSpan
Log new SpanID
Trace

<table>
<thead>
<tr>
<th>Services &amp; Spans</th>
<th>milliseconds</th>
</tr>
</thead>
<tbody>
<tr>
<td>user</td>
<td>11,151.5</td>
</tr>
<tr>
<td>stock</td>
<td>23.6</td>
</tr>
<tr>
<td>order</td>
<td>64.9</td>
</tr>
<tr>
<td>order</td>
<td>63.2</td>
</tr>
<tr>
<td>order</td>
<td>9.6</td>
</tr>
<tr>
<td>money</td>
<td>72.5</td>
</tr>
<tr>
<td>money</td>
<td>107.6</td>
</tr>
<tr>
<td>money</td>
<td>58.3</td>
</tr>
<tr>
<td>money</td>
<td>10,007.2</td>
</tr>
<tr>
<td>confirmations</td>
<td></td>
</tr>
<tr>
<td>confirmations</td>
<td></td>
</tr>
<tr>
<td>stock</td>
<td></td>
</tr>
<tr>
<td>stock</td>
<td></td>
</tr>
</tbody>
</table>

Span

ParentSpan

Logs

Tue Jan 17 2017 [APP/0] APP user OUT 2017-01-17 20:00:11.425 "INFO [tracer,6d2d6cf9f6424f52,6d2d6cf9f6424f52,6d2d6cf9f6424f52,6d2d6cf9f6424f52,true]"
14  --- [nio-0000-exec-1] com.metrics.SomController : finished
delay=78
Lightweight user personas
Stand by me
Lightweight user personas:

- Ops Engineer
- Test Engineer
- Build & Deployment Engineer
- Service Owner
Lightweight user personas:

Consider the User Experience (UX) of engineers and team members using and working with the software.
Clark Andrews

**Motivations**
- Incentive
- Fear
- Achievement
- Growth
- Power
- Social

**Goals**
- To cut down on unhealthy eating and drinking habits
- To measure multiple aspects of life more scientifically
- To set goals and see and make positive impacts on his life

**Frustrations**
- Unfamiliar with wearable technology
- Saturated tracking market
- Manual tracking is too time consuming

**Bio**
Aaron is a systems software developer, a "data junkie" and for the past couple years, has been very interested in tracking aspects of his health and performance. Aaron wants to track his mood, happiness, sleep quality and how his eating and exercise habits affects his well being. Although he only drinks occasionally with friends on the weekend, he would like to cut down on alcohol intake.

**Brands**

**Technology**
- IT & Internet
- Software
- Mobile Apps
- Social Networks

**Personality**
- Extrovert
- Introvert
- Sensing
- Intuition
- Thinking
- Feeling
- Judging
- Perceiving

http://www.keepitousable.com/blog/?tag=alan-cooper
Lightweight user personas:

What data does the User Persona need visible on a dashboard in order to make decisions rapidly & safely?
Summary
Operability

making software work well in Production
logging with Event IDs

use `enum`-based Event IDs to explore runtime behaviour and fault conditions
Run Book dialogue sheets

explore and establish operational requirements as a team, around a physical table, together
endpoint health checks

HTTP 200 / 500 responses to /status/health call with JSON details — good for tools and humans
Correlation IDs

trace execution using correlation IDs:
synchronous (HTTP x-trace-id)
async (SQS MessageAttribute)
lightweight user personas explore the UX and needs of different roles for rapid decisions via dashboards
Operability
use modern logging, Run Book dialogue sheets, endpoint healthchecks, correlation IDs, and user personas as team collaboration techniques
Resources

• Training: Practical Operability for Developers and Testers – led by Matthew Skelton and Rob Thatcher – 1-day workshop – http://www.unicom.co.uk/practical-operability-for-developers-and-testers.html


• Run Book template & Run Book dialogue sheets http://runbooktemplate.info/
Questions?

Twitter: @SkeltonThatcher | #operability
email: questions@skeltonthatcher.com
thank you

@SkeltonThatcher
skeltonthatcher.com