What if the promise of AIOps were true?

niallm@relyabilit.ie

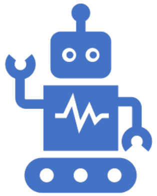
https://www.relyabilit.ie

@niallm

Acknowledgements

- Thanks to Todd Underwood (Google), Steve Ross (Google), Danyel Fisher (Honeycomb), and many other thinkers in the AI/Ops space whose thoughts I plundered for this

A few definitions



AI: Artificial Intelligence

ML: Machine Learning

MLOps: Running machine learning infrastructure

AIOps: Running production infrastructure with AI (~ML)

(Actually, AIOps is more like “Big Data applies to production too, you know”)

Claim Chowder

- “Most experts consider AIOps to be the future of IT operations management.” – IBM, https://www.ibm.com/cloud/learn/aiops
- “Gartner Says There Is No Future of IT Operations That Does Not Include AIOps.” – www.sciencelogic.com
- “AIOps can be viewed as CI/CD for core IT functions.” – Wikipedia
- “But there’s another type of AI—an algorithmic approach to intelligence—that is smart and is emerging as the type of AI that IT organizations of all types could start implementing very soon. And there’s nothing artificial about it.” – Richard Whitehead, https://dataconomy.com/2017/03/aiops-type-ai-nothing-artificial/
- “For developers, one of the fears around DevOps is that in “picking up the pager,” they are opening themselves up to a constant stream of irrelevant alerts. Old Ops hands nod wryly at this, or maybe try to share their personal approach to filtering the noise. Instead, AIOps ensures that only valid, actionable alerts get as far as being shown to a human being and taking up their valuable time.” – Dominic Wellington, https://devops.com/aiops-second-law-ops/
- “Today, with AIOps, you can automate the monitoring and correlation of activities in the data pipeline and can uncover the myriad issues that can arise […].” – Bala Venkatrao, https://thenewstack.io/how-aiops-conquers-performance-gaps-on-big-data-pipelines/

Photo by Adrien Sala on Unsplash

Saying the quiet bit out loud

A Digital Advisor

“In an age when IT has become too overwhelmed with data to function **under traditional management**, AIOps opens the door to a brighter future. With swift **cross-silo** data gathering and analytics, it saves time, lowers costs, foresees problems, and solves many of them before you even know to ask.

“One customer told us it’s like having **an additional engineer they don’t have to pay for**,” Fondekar says.”

https://www.cio.com/article/3620073/simplifying-it-with-aiops.html

Photo by Kristina Flour on Unsplash

The Opposing Positions

- The Underwood/Ross Stance
  - (as per talks at SRECons *passim*)
  - **AIOps doesn't make much sense because:**
    - Maths for effective ML doesn't work out
    - ROI for model generation doesn't work out
    - Faster/cheaper to just do stats
  - Techniques will improve, so perhaps some specific tasks will get better, but certainly not a revolution

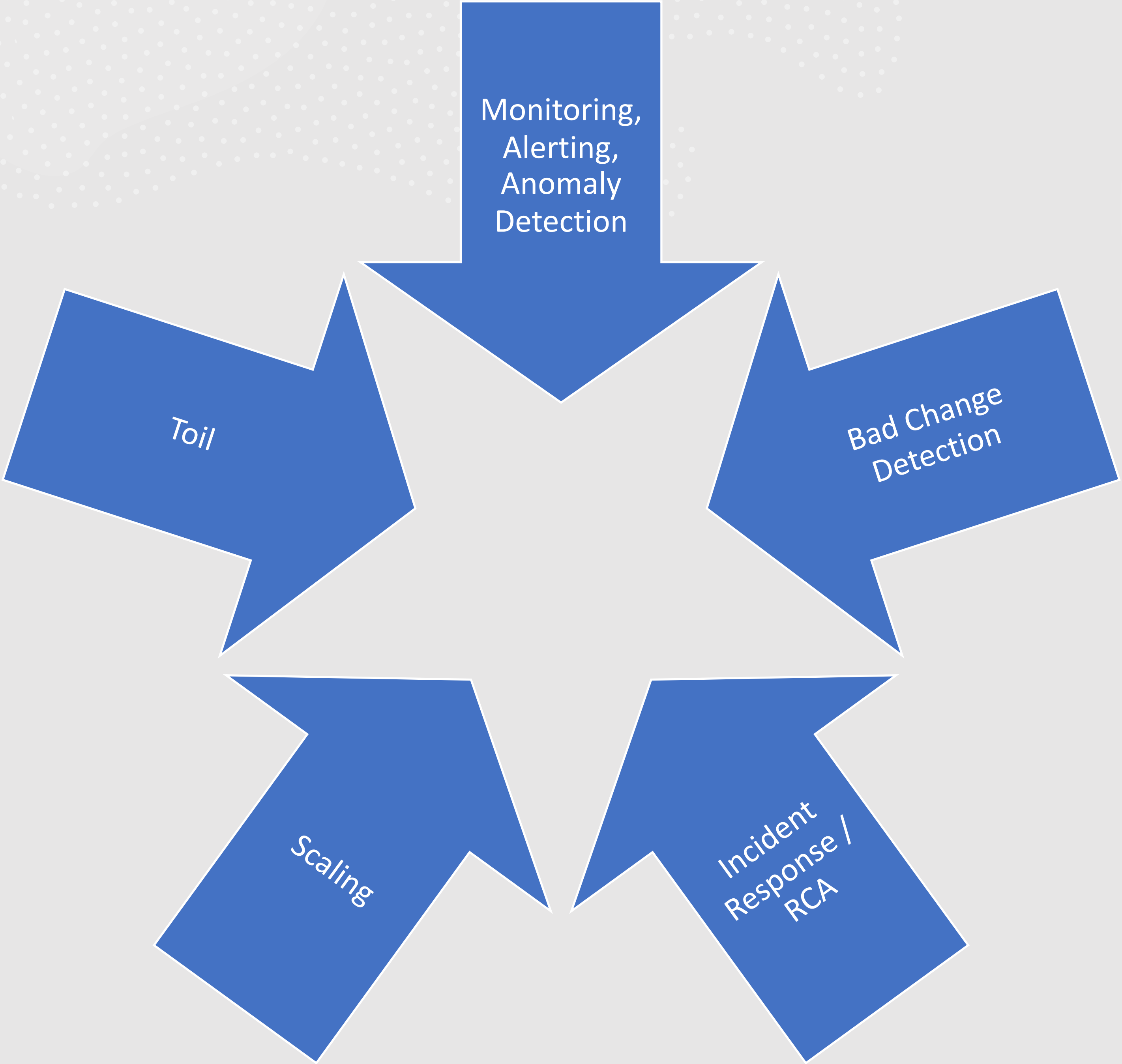
The Strong AIOps Stance

- **AIOps will dominate in (5-10) years**

The Weak AIOps Stance

- **ML can be really useful in Ops contexts**
  - Deploy in narrow circumstances with relatively context-free application (e.g. anomaly detection)
  - Silos will often accede to data being shared with machines but not other people
  - Stats requires stats expertise

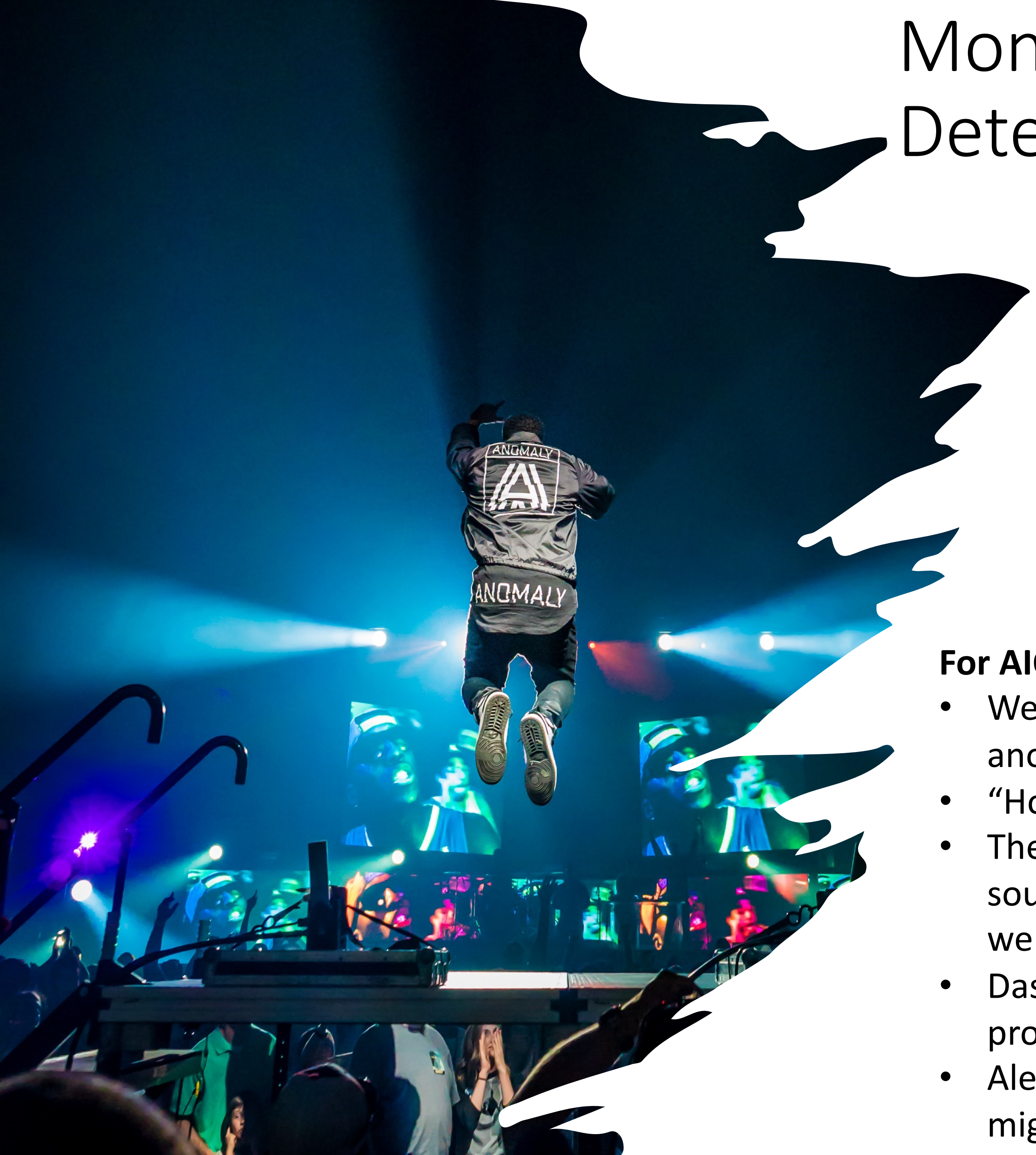
Analysis of Applicability



Analysis of Applicability

Actually, it’s all about the data

(and creativity and supervision)



Monitoring, Alerting, Anomaly Detection

SRE Beliefs

- Monitoring is a dynamic thing.
- You often discover you're not monitoring everything you need to, and need to add metrics.
- You sometimes discover you're monitoring too many things (dashboarding).
- Alerting noise is a problem solvable by tuning & pruning alerts.
- Anomaly definitions are highly context sensitive, in the limit.

For AIOps to be true/useful

- We only need some kind of Pareto relationship for signals, anomalies, and outages to be useful.
- "Hoofbeats are more likely to be horses than zebras."
- The software may have access to more data (join across more sources) than the people do. We'll make progress with the sources we can get.
- Dashboarding is a human interface, we don't have to fix that problem.
- Alert tuning relies on expertise and time, neither of which your org might have.

Bad Change Detection

SRE Beliefs

- Changes are a combination of really high risk and completely necessary to do, and if you don't do them sometimes the risk gets higher.
- Validating the change worked ok can be on a spectrum from completely manual to completely automated.
- Rollback is the most powerful tool for addressing bad changes. Just do it if you have any suspicion at all.
- Rollback is tricky in e.g. schema change cases, and some companies don't even get to roll back because their environment doesn't support it.

For AIOps to be true/useful

- We don't need to solve roll-back, we just need to be useful at detection.
- Testing-in-prod allows user behaviour/SLOs to be part of the decision-making.
- GitHub CoPilot might help you to write tests, but AIOps isn't going to. AIOps can look at metrics and train on previous data, possibly even automatically labelled.


Photo by Emma Gossett on Unsplash

Incident Response / Root Cause Analysis

SRE Beliefs

- You can’t automate incident response. It’s too fundamentally chaotic / creative.
- Root cause work is a) extremely complicated and b) primarily a social activity.
- Ideally, root causes don’t recur because you put the work in to make sure they don’t.

For AIOps to be true/useful

- Root causes recur all the time. With enough repetition, you can train on it.
- Incident response can be divided into known-knowns and everything else; we only need a Pareto-style structure to make useful suggestions.
- Big Red Button style approaches (i.e. context-free mitigation actions, like drain a DC) make it easy for AIOps to contribute value.

The Johari Window/Quadrant Model (On-call)

Known Knowns (Risks we're aware of, and know how to fix)	**Known Unknowns** (Risks we're aware can happen, but don't know how to fix)
Unknown Knowns (Risks we're not aware can happen, but we know how to fix)	**Unknown Unknowns** (Risks we're not even aware can exist, and we don't know how to fix)

Photo by Josh Blanton on Unsplash

Scaling / Toil / General Ops

SRE Beliefs

- We very much hope scaling is auto-scaling, except when we don't.
- Scaling is usually easy except when non-linear effects happen.
- Toil is supremely automatable. But incident response isn't.

For AIOps to be true/useful

- Scaling is one of the clearest use cases for matching input to actions - even a non-linear relationship is okay as long as the back-off is too. But this is close to a solved problem for a number of key use cases; therefore AIOps only helps with edge cases.
- Toil is automatable but at the moment it's hard to see how AIOps can meaningfully exceed human performance here.
- We can trigger actions on metrics today; don't need ML for that. For AIOps to be useful, we'd need something more sophisticated.

Photo by [Tony Tran](#) on [Unsplash](#)

Conclusion

- AIOps "use definition" seems to be "big data can help operations". Hard to object.
- In particular, help CI/TOs compensate for organizational dysfunction of various kinds. This is a real effect.
- For activities which match "known known" patterns, AIOps is plausible today.
- Applicability will improve generally, particularly if sample sizes can improve dramatically, standardization increases, and unsupervised learning becomes tractable.
- But fundamental limits, particularly around {un}known-unknowns in incident resolution, cost/benefit, and anomaly detection would appear likely to exist.
- "Attacks only get better." We should beware of complacency about what value we provide.

Photo by Niall Murphy

What if the promise of AIOps were true?

niallm@relyabilit.ie

https://www.relyabilit.ie

@niallm