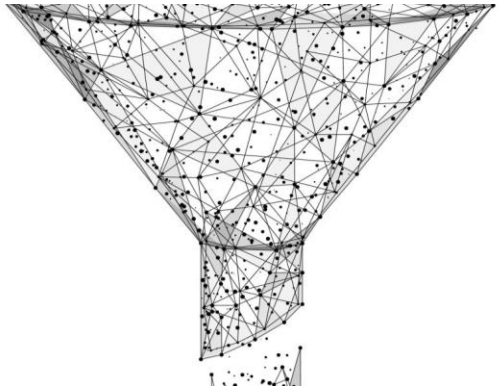
SRE: Value streams and the toil “Funnel”



What is the bread and butter of SRE work? **Preparation, Prevention and Response** (this can be shortened to CAPA) for incidents and toil. To clarify in this context, Toil is defined by Google as *"the kind of work tied to running a production service that tends to be manual, repetitive, automatable, tactical, devoid of enduring value, and that scales linearly as a service grows”*. Incidents require *“mitigating the impact and/or restoring the service to its previous condition”*. In short toil is the annoying kind of work that wastes talent and time, and incidents are stressful events that impact customers and have direct financial consequences. Something they both have in common > lowering morale.

Back to the show.... An SRE’s mind is adept to finding an adequate solution to both the boredom of toil and the panic of incidents. In those moments of downtime of “waiting around for the next incident/ticket” or during MTBF we are preoccupied with preparation and prevention to limit the need for responding. In moments of panic (or possibly annoyance) we are simply responding using our training/tools and after that initial response we fall back to the earlier state.

There is a mutually exclusive collectively exhaustive list of value streams that lead to less toil. All toil can be resolved by one of these 4: observability enhancements, automation, process change or RCA.

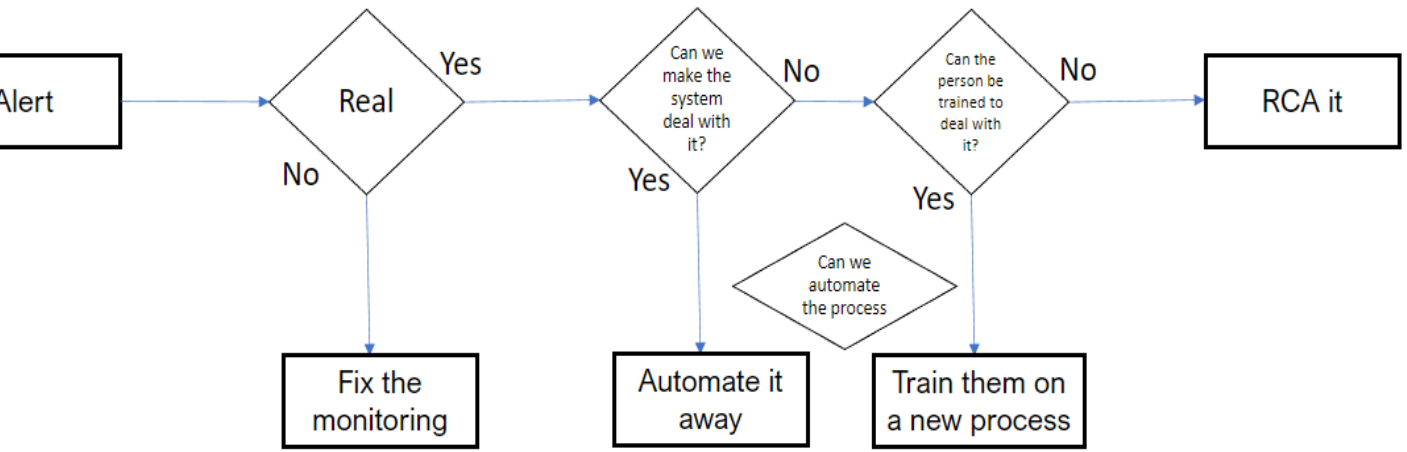
Why? Either the problem is real or fake. If it is fake the monitoring needs to be tweaked. If it is real an engineer must systematically analyze it: Can it be automated away (error interception)? Can we train people to use a new process (avoidance)? Can we automate that “new” process and catch the issue proactively with some monitoring changes? Do we just need to fix it the old fashion like tweaking the system design or updating some code?

SRE must leverage their time to create the greatest impact. If done right, the “greatest hits” will be all the toil and incidents that never happened.

SREs on there down time are either:

	Preparing	Preventing	Responding
Toil	• Unit testing alerts • Capping workload at 50%	• Observability Enhancements • Automation • Process Modifications	FIGURE IT OUT

	• Returning misbehaving services to app team	• RCA	
Incidents	• Operational Readiness Reviews • Fire drills • Chaos testing • DiRT Testing	• Predictive Observability • Bounding Threshold Alerts • Anomaly Detection • Postmortems	GET TO WORK


Alert
Real
Yes
No
Fix the monitoring
Can we make the system deal with it?
No
Yes
Automate it away
Can we automate the process
Can the person be trained to deal with it?
No
Yes
Train them on a new process
RCA it