Andrew: Bus Principle, i.e. the need for SMEs	Better runbooks/SOPs, and time to work on it.	To properly scale any team, work needs to be done smart so as to not have the need for an SME every time something goes wrong. This means having a proper procedure to go through went things go wrong. Even for SMEs, it's difficult to remember every single thing about every job/service's alerts. Proper Runbooks allow you to save mental space and work on other things that require attention, rather than having to remember every single thing ever.
Andrew: Oversaturation of tickets	"Pass the pager back". If jobs are unstable, they should be given back to the development team.	Reliability is not just our responsibility, it's the entire company's. If teams know they have to be accountable for their work, they won't just throw it over the wall. Engineering work requires careful thought and focus. If we keep getting inundated by tickets, and for that matter, meetings and other toil, the engineering work outside of it will struggle and fail.
Andrew: Communication issues	Stop using emails. Put people first.	Teams channels with communication lines with the other teams and the "public" such as us. Within the channel, a post should be made for an incident, and communication should be handled there. This allows us to keep everything in one place for incidents. This also allows us or the other teams to not mess up communications by forgetting to hit "send all", instead of "send". 90 day org policy of removing messages either needs to be removed or the conversations archived somewhere for later viewing.
Andrew: Too many meetings	Daily standup and scrum is micromanaging. Remove it.	Time is the most valuable resource to any engineer. Daily standups, in the classical sense, are for teams of relatively large size (~10), that can't all pair/meet adhoc, that work heavily on connected things, and have a need to potentially pivot.

Andrew Sison: Bad allerts

Remove bad alerts.

An alert has to be 2 things, actionable and impacting. If it's not actionable, why should the on-call be alerted when they need to focus on issues that are actionable. Any non actionable alerts need to be given proper investigation from an engineering standpoint , i.e. long term solutions. If it's not impacting, then it doesn't matter.