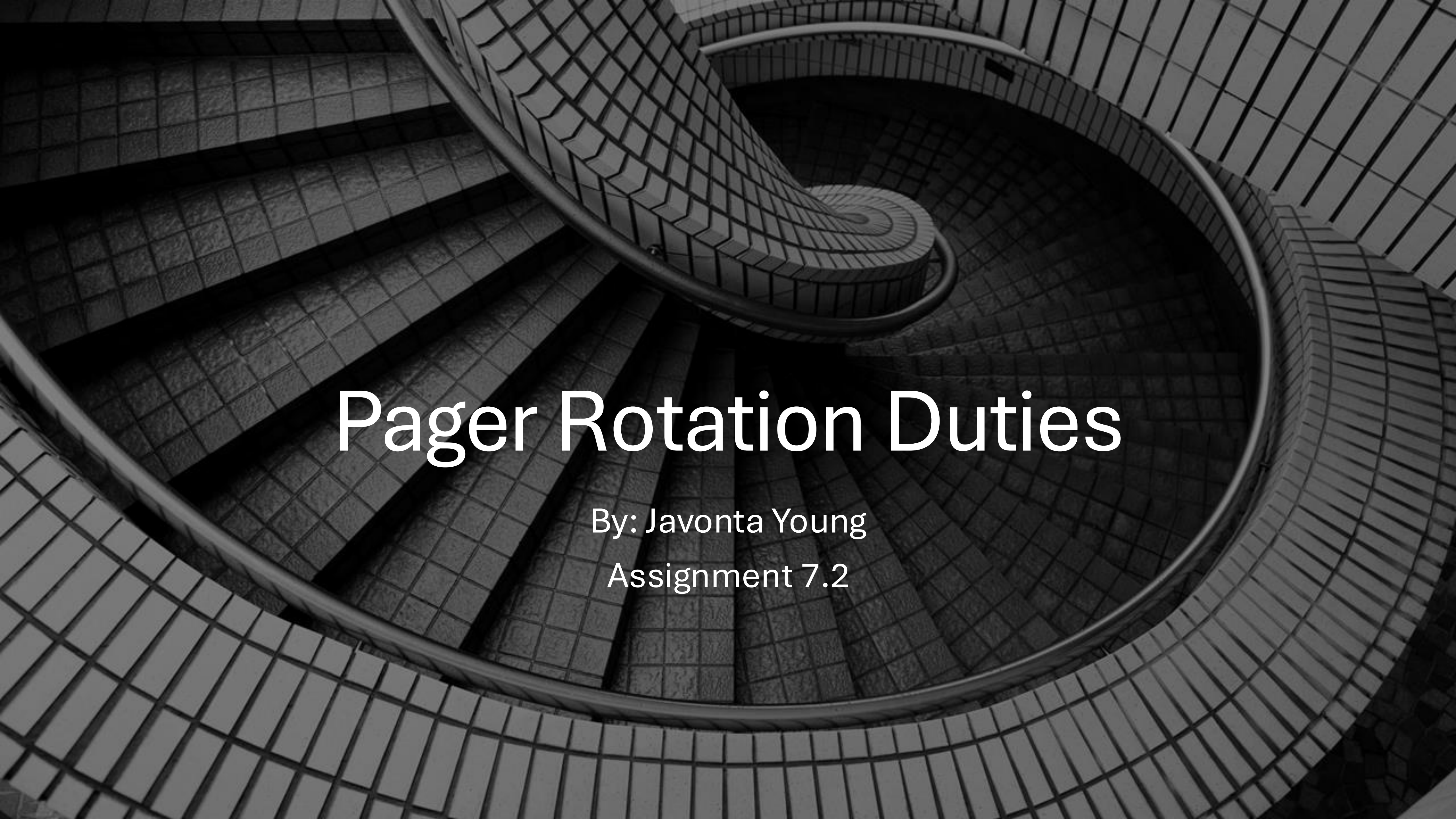


Pager Rotation Duties

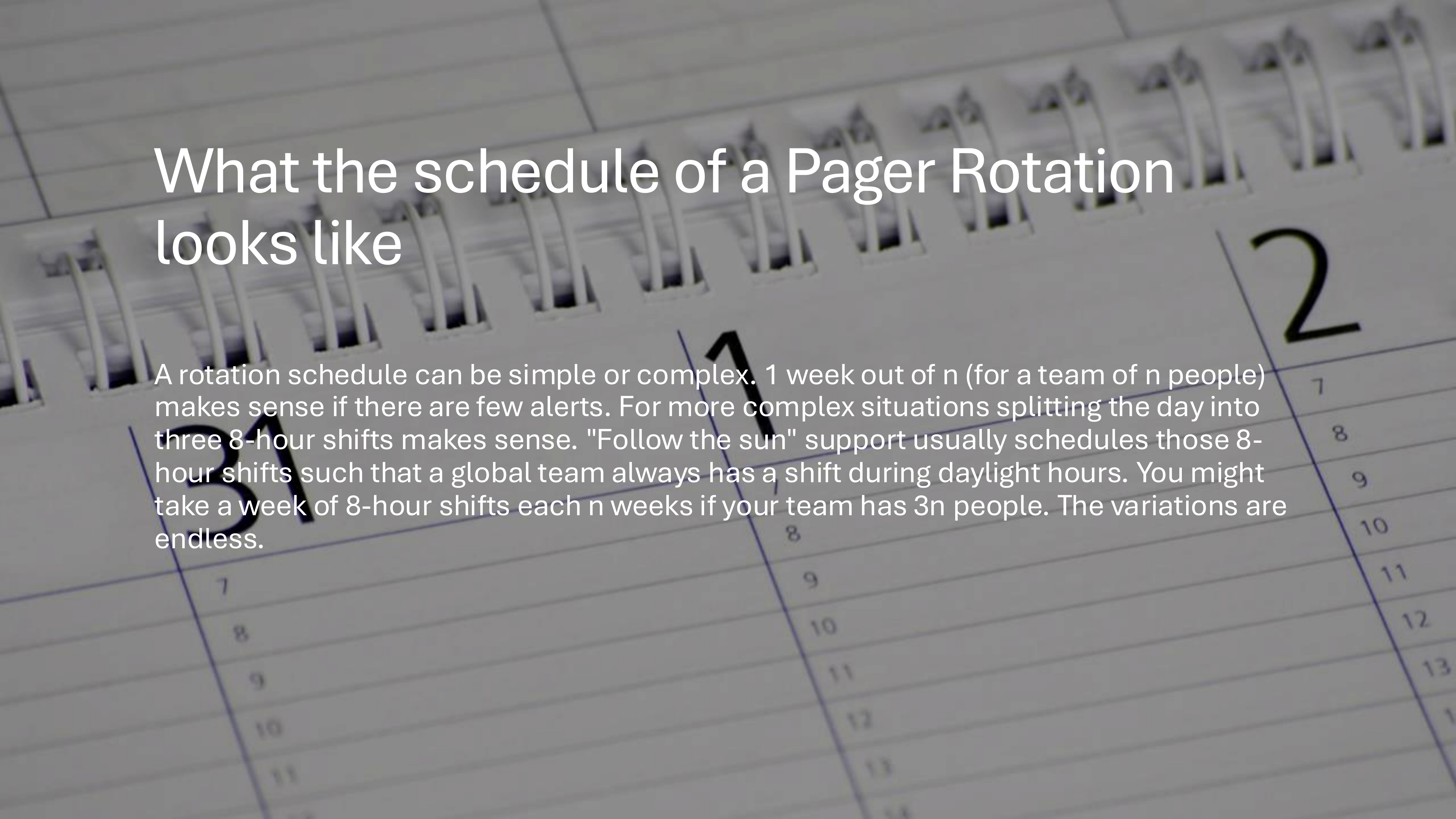
By: Javonta Young

Assignment 7.2

What is a Pager Rotation

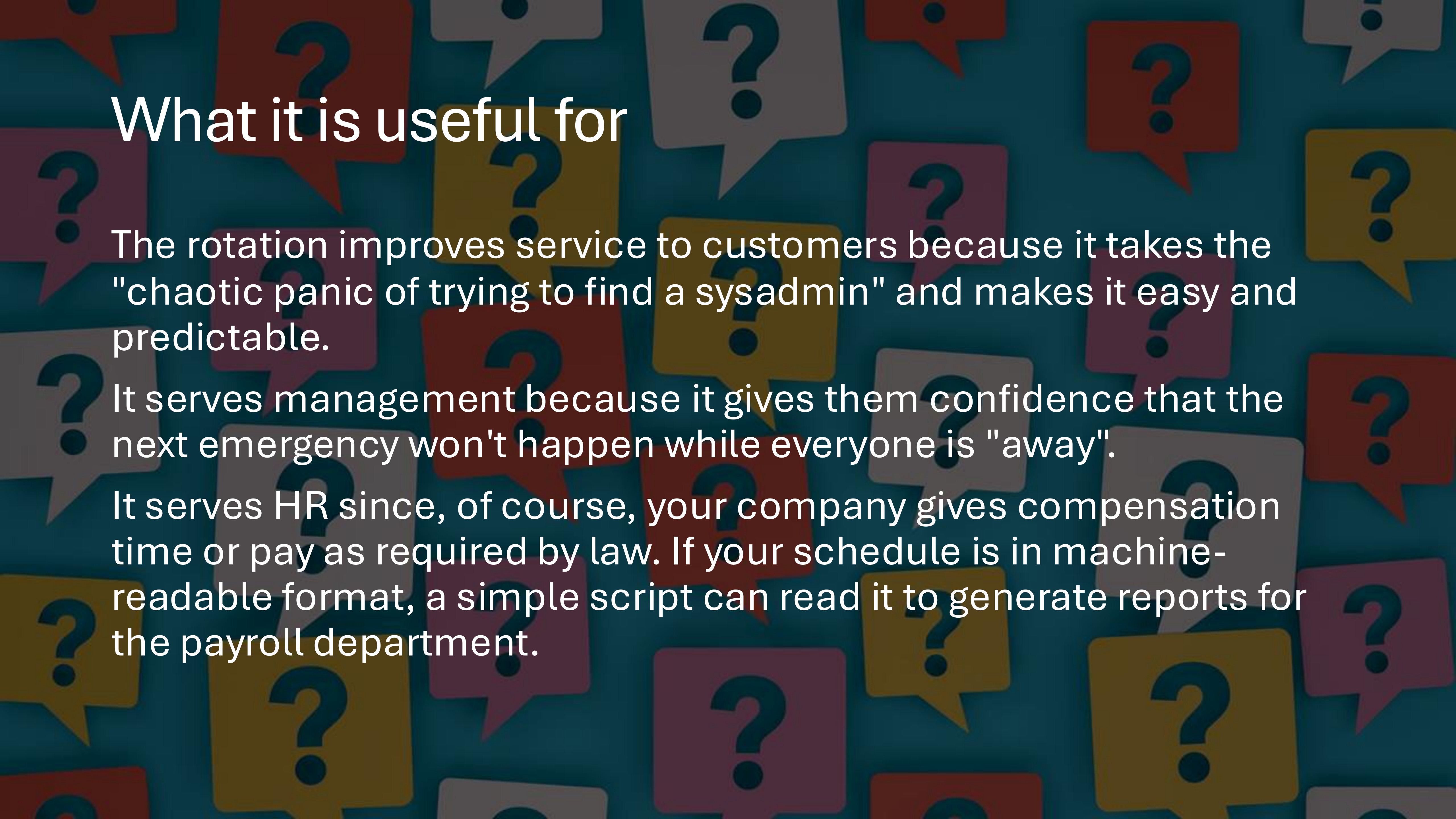
An on-call rotation schedule documents who is "carrying the pager" (or responsible for alerts and emergencies) at which times.

You might literally "pass the pager", handing it to the next person periodically, or everyone might have their own pager and your monitoring system consults a schedule to determine who to page. It is best to have a generic email address that goes to the current person so that customers don't need to know the schedule.



What the schedule of a Pager Rotation looks like

A rotation schedule can be simple or complex. 1 week out of n (for a team of n people) makes sense if there are few alerts. For more complex situations splitting the day into three 8-hour shifts makes sense. "Follow the sun" support usually schedules those 8-hour shifts such that a global team always has a shift during daylight hours. You might take a week of 8-hour shifts each n weeks if your team has 3n people. The variations are endless.



What it is useful for

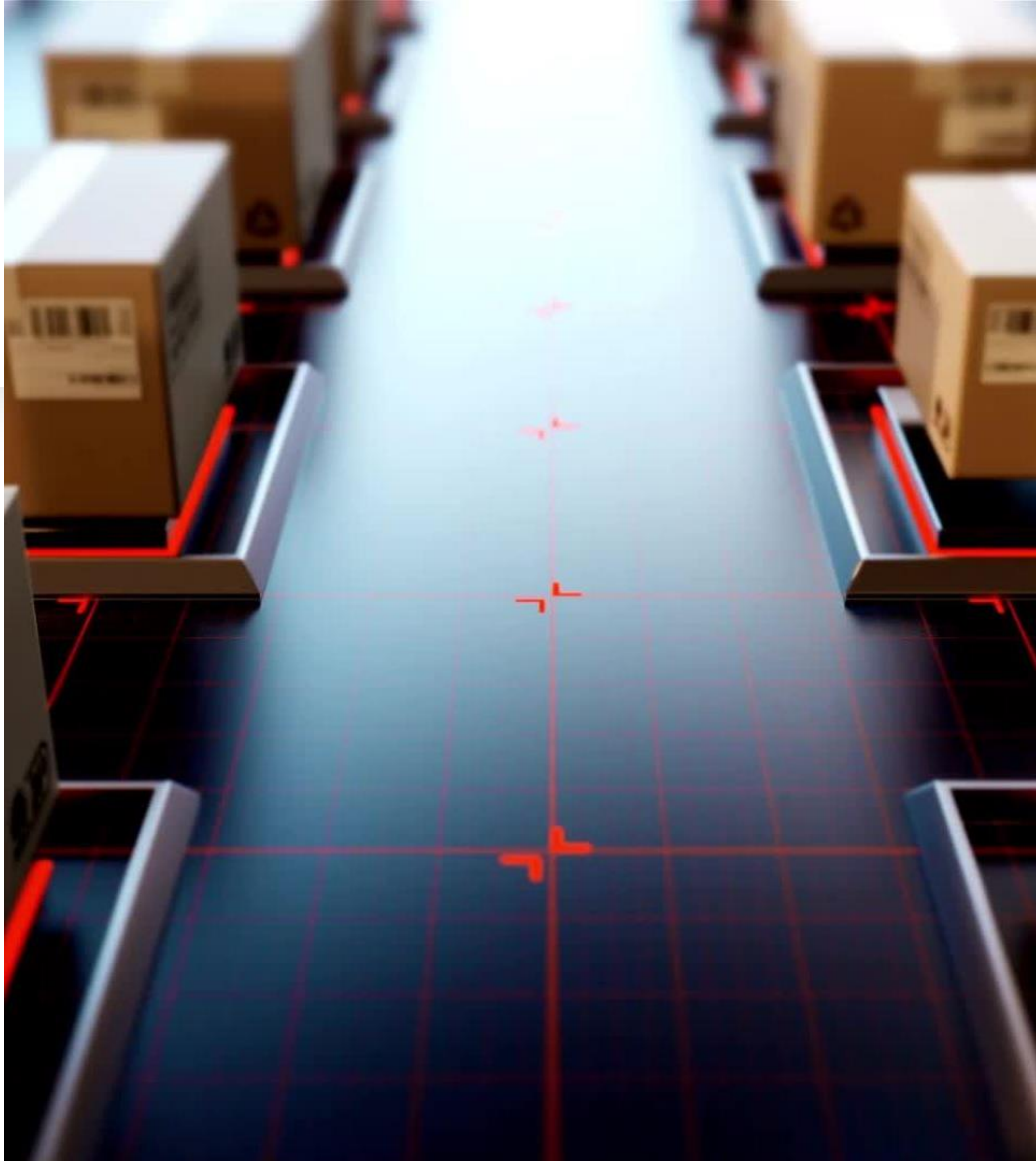
The rotation improves service to customers because it takes the "chaotic panic of trying to find a sysadmin" and makes it easy and predictable.

It serves management because it gives them confidence that the next emergency won't happen while everyone is "away".

It serves HR since, of course, your company gives compensation time or pay as required by law. If your schedule is in machine-readable format, a simple script can read it to generate reports for the payroll department.

Features are not "done"

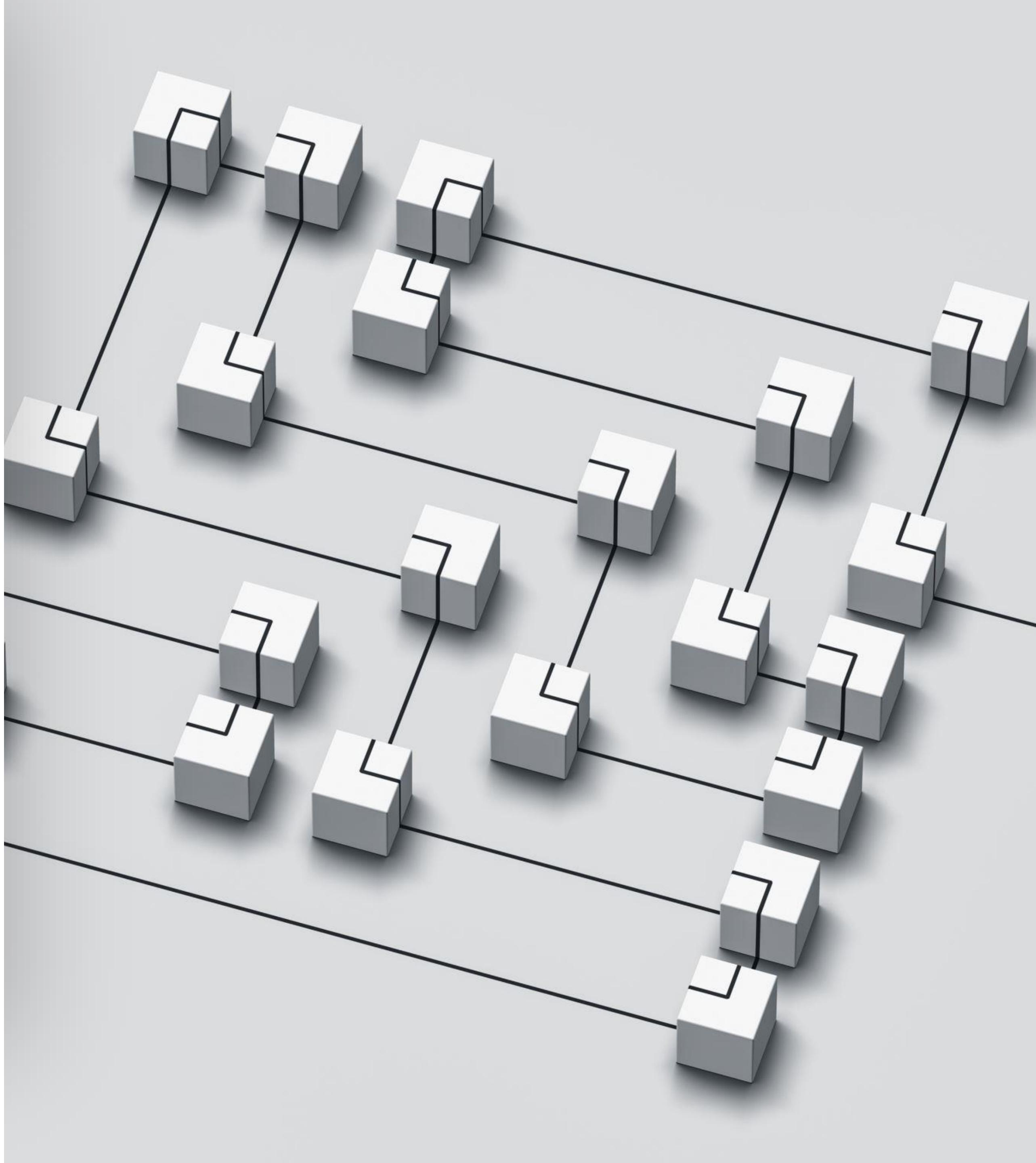
Features are only done when it is performing as designed in production, without causing excessive escalations or unplanned work for either dev or ops. ITIL defines warranty as when a service can run in production reliably without intervention for a predefined period of time. This definition of warranty should ideally be integrated into our collective definition of done.



THE PRINCIPLES OF FLOW

The principles of flow are a summary of all the relevant Lean, Agile principles and practices described more or less in Kanban, Scrum, Value Stream Mapping, and Theory of Constraints in order to take control and manage effectively software product delivery.

- Reduce WIP
- Make Work Visible
- Reduce Batch Sizes
- Reduce the number of hands-off
- Continually Identify and Elevate Constraints
- Eliminate Hardships and Waste in the Value Stream



THE PRINCIPLES OF FEEDBACK

The principles of feedback are a summary of all the relevant feedback principles from the Fifth Discipline (by Peter Senge), Kanban, Scrum, and The Toyota Way (Jeffrey K. Liker.)

- Working Safely with Complex Systems
- See problems as they occur
- Keep pushing quality closer to the source
- Enable Optimizing for Downstream Work Centers

THE PRINCIPLES OF CONTINUAL LEARNING AND EXPERIMENTATION

The principles of continual learning and experimentation are a summary of all the relevant learning culture principles from the Fifth Discipline (by Peter Senge), Kanban, Scrum, and The Toyota Way (Jeffrey K. Liker.)

Enable Organizational Learning and Safety Culture

Institutionalize the Improvement of Daily Work

Transform Local Discoveries into Global Improvements

Inject Resilience Patterns into the Daily Work

Leaders Reinforce a Learning Culture

Sources

- https://medium.com/@nikolaos.raptis83/clean-devops-back-to-basics-1feaf938322b
- https://pabis.eu/devops-handbook/Segment47.html
- ttps://opsreportcard.com/section/13#:~:text=An%20on%2Dcall%20rotation%20schedule,to%20determine%20who%20to%20page
.