THE LINUX FOUNDATION
OPEN SOURCE SUMMIT
NORTH AMERICA

Embedded Linux Conference

Platform Engineering:

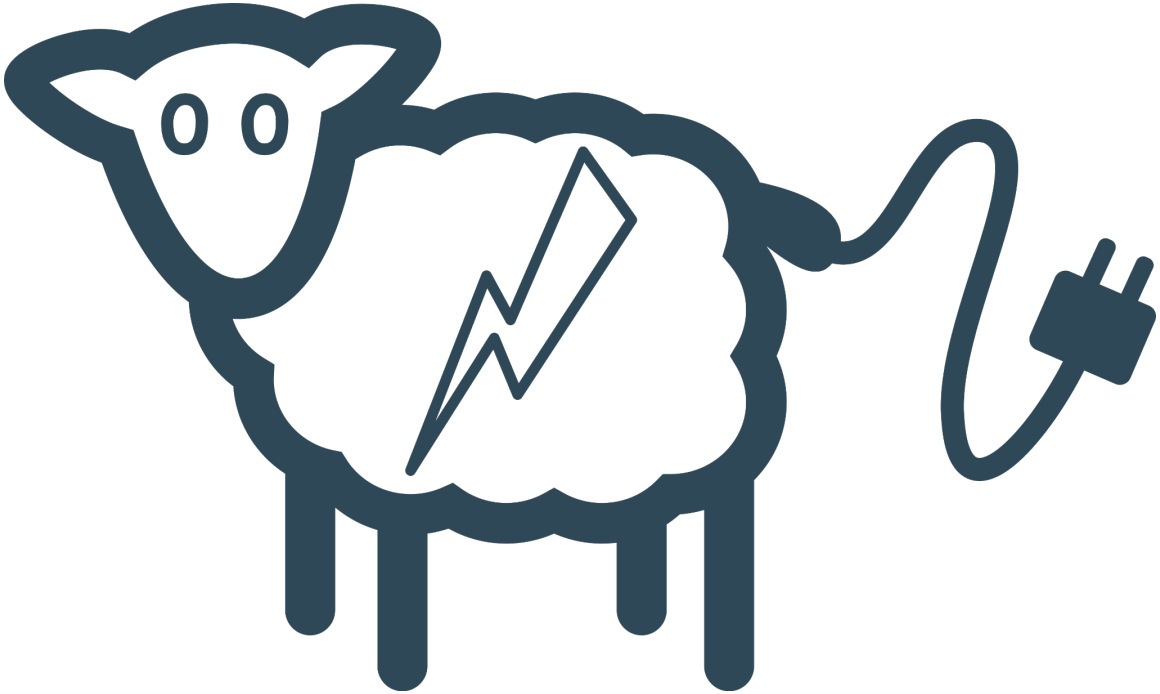
Herding the Electric Sheep

Brett Smith
<bc.smith@sas.com>
20260519

Platform Engineering

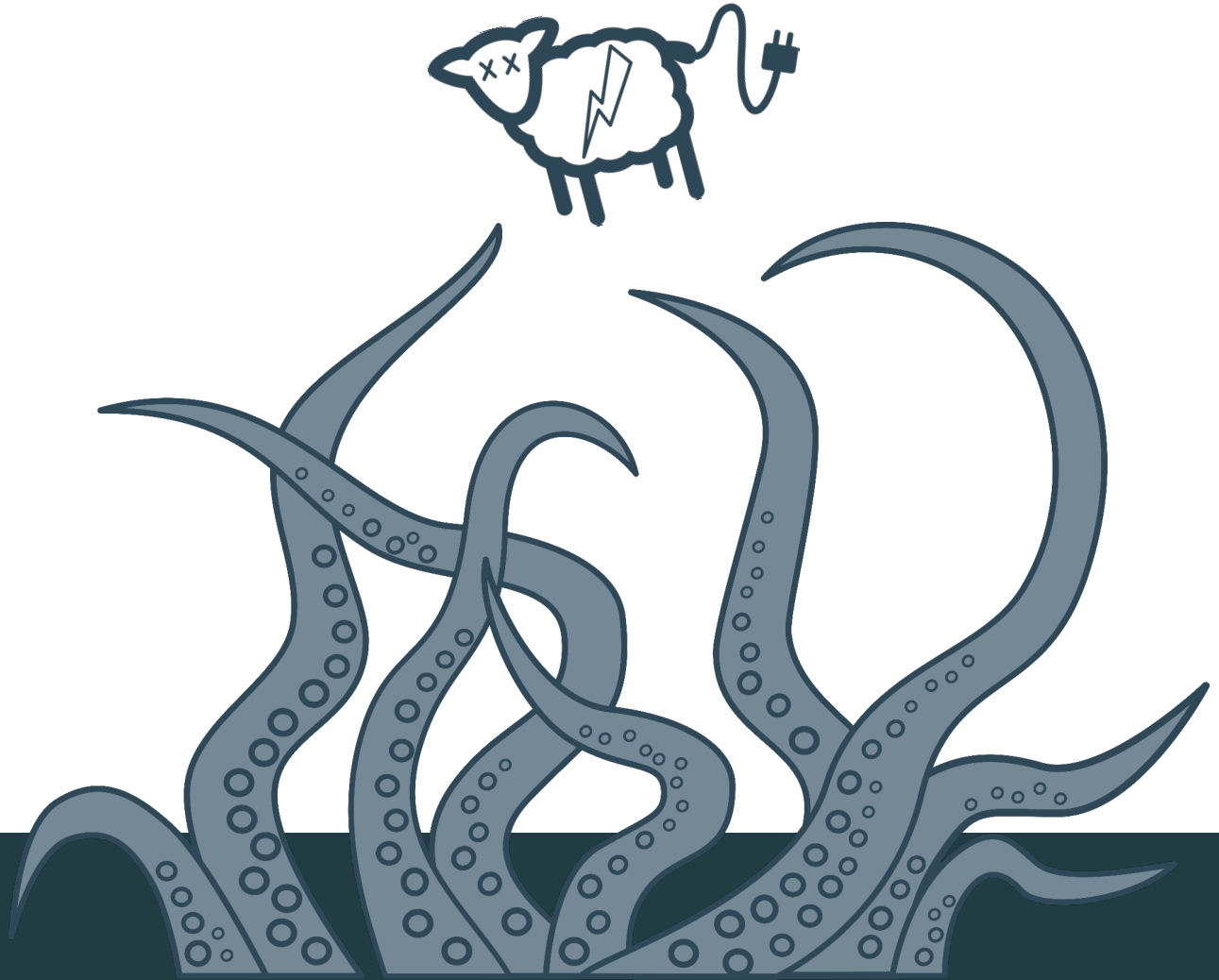
Internal Developer Platform

Electric Sheep

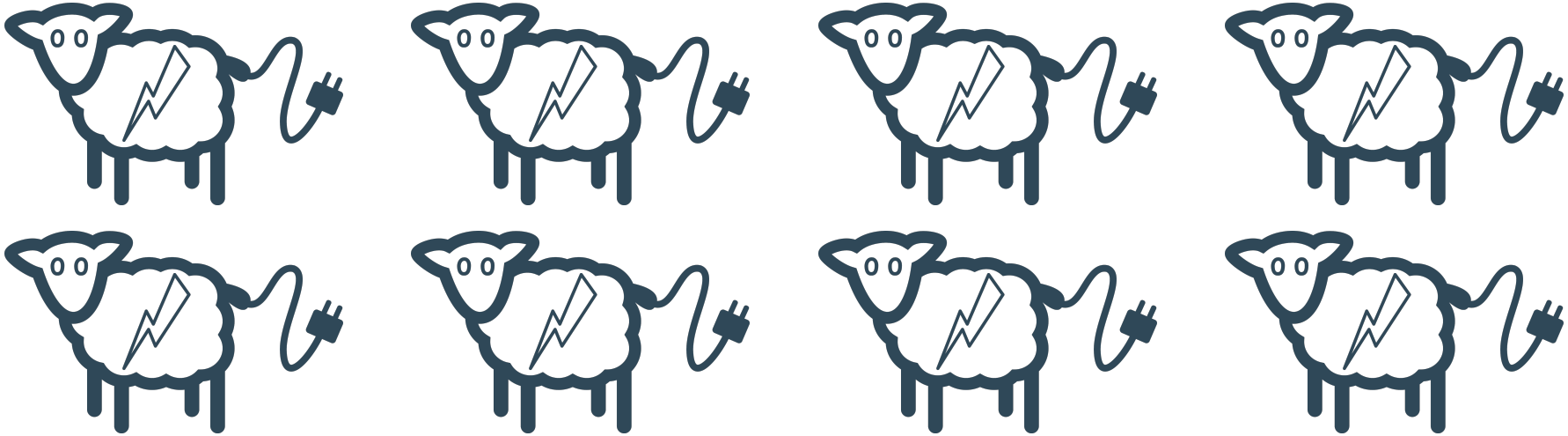


DevOps

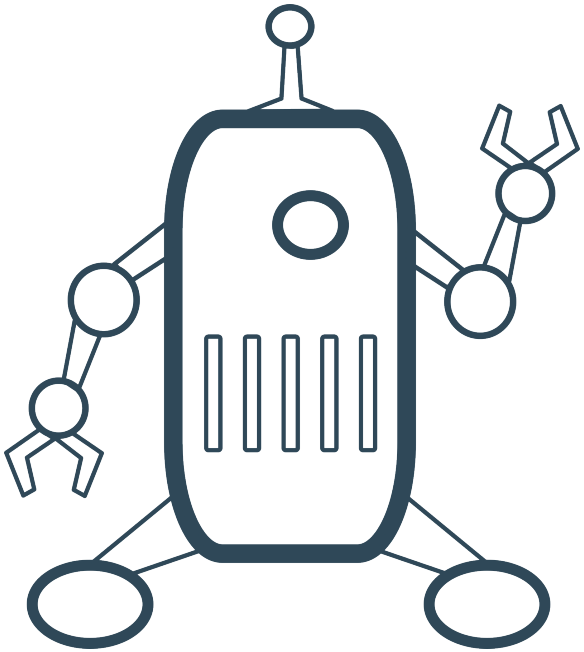
Also DevOps

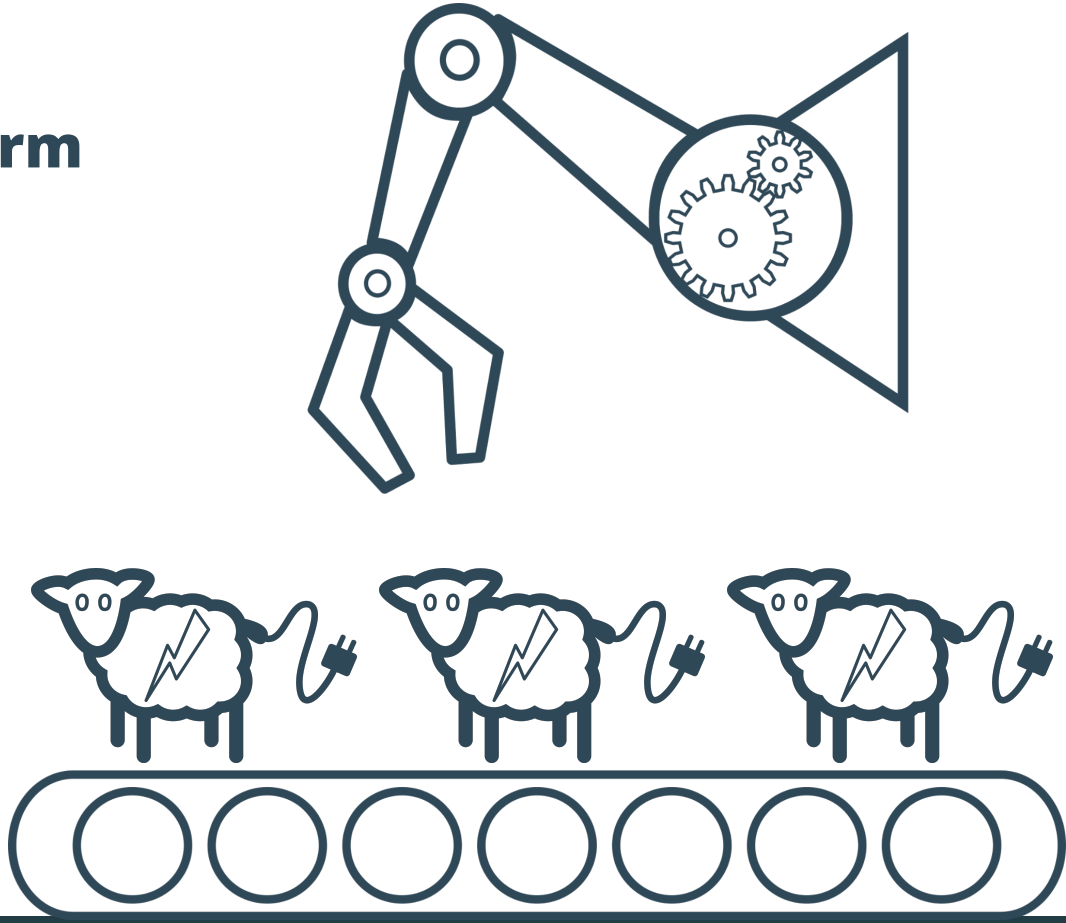


Platform Engineering



Internal Developer Platform

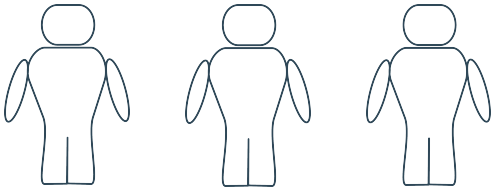


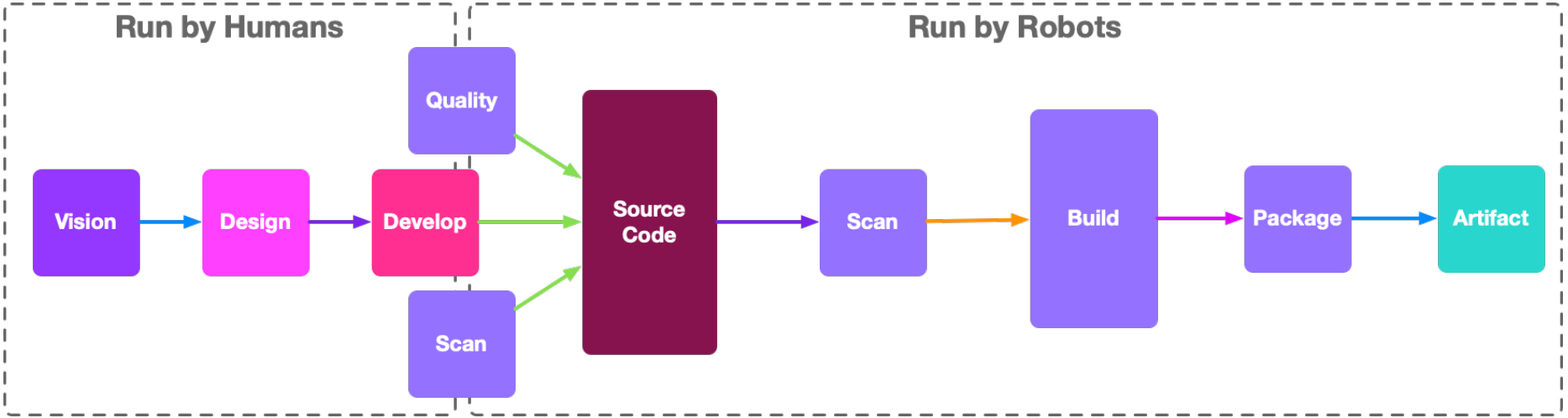


IDP

Who Benefits from an IDP and How?

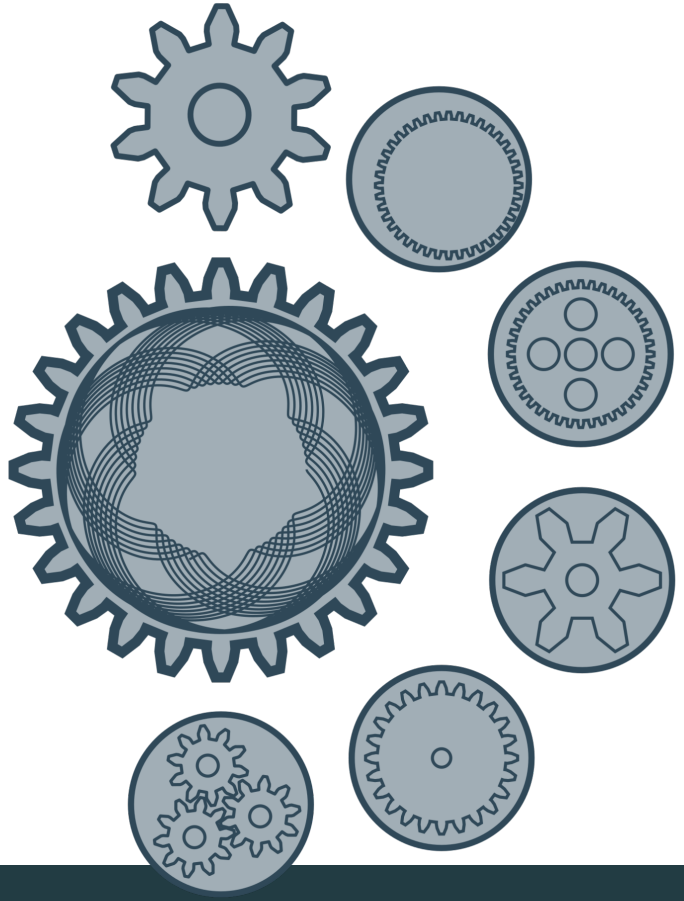
Shift Left - Automate Right



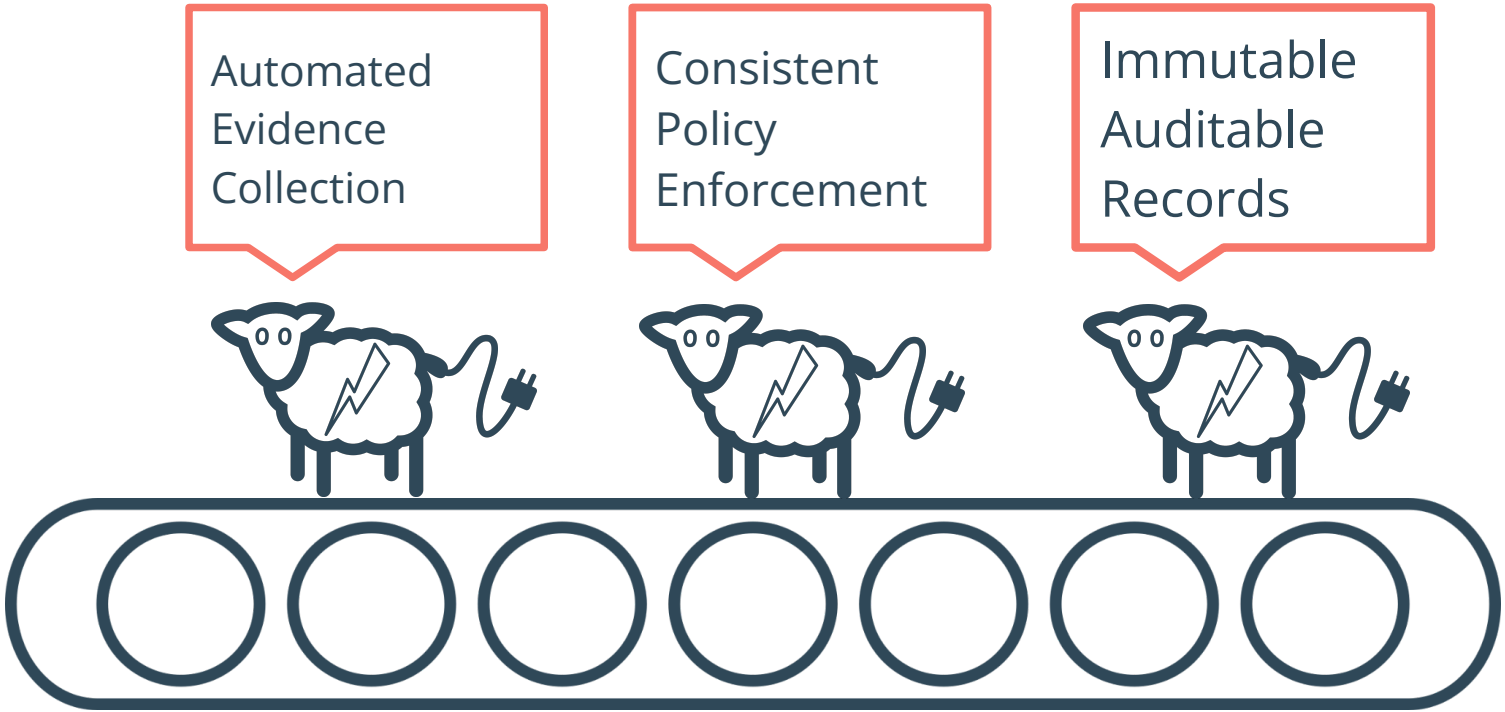


SBOM

Software Bill of Materials



Compliance


Automated
Evidence
Collection
Consistent
Policy
Enforcement
Immutable
Auditable
Records

SLSA

Source Track

Track/Level	Requirements	Focus
Source L1	Use a version control system	First steps towards operational maturity
Source L2	History and controls for protected branches & tags	Preserve history and ensure the process has been followed
Source L3	Signed provenance	Tampering by the source control system
Source L4	Code review	Tampering by project contributors

Build Track

Track/Level	Requirements	Focus
Build L0	(none)	(n/a)
Build L1	Provenance showing how the package was built	Mistakes, documentation
Build L2	Signed provenance, generated by a hosted build platform	Tampering after the build
Build L3	Hardened build platform	Tampering during the build

Policy as Code (PaC)

Security is codified, automated, open, and accessible to all stakeholders

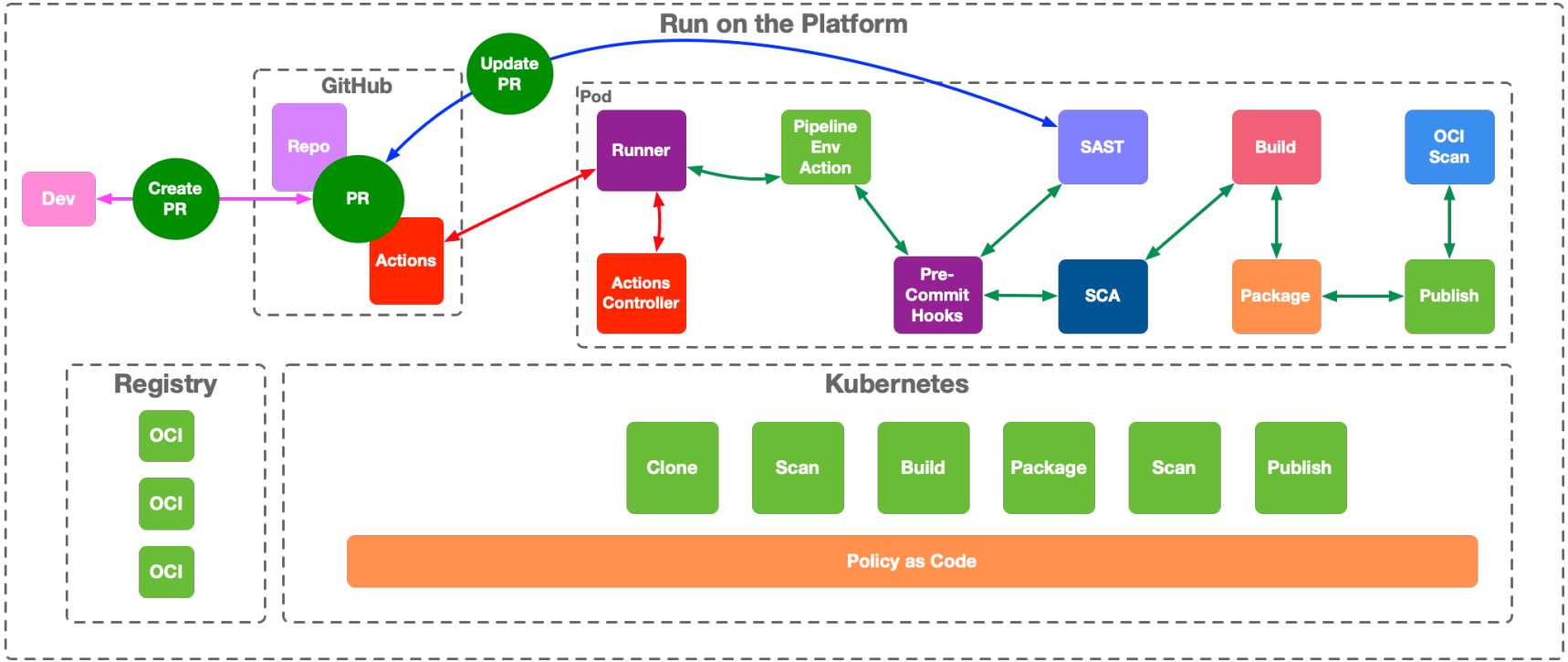
Key Features:

- Codification
- Automation
- Integration
- Consistency

Benefits:

- Improved Security
- Enhanced Compliance
- Increased Efficiency
- Reduced Errors
- Greater Agility
- Improved Collaboration

Shift-Down Security



Where to Start?

- Assess existing tools and services across teams
- Identify patterns and consolidation opportunities
- Visualize findings for a clear landscape overview
- Collaborate on unique needs for potential platform improvements
- Design platform around main use cases first
- Prioritize the main 80% - 10% - 10%, and handle exceptions
- Use core solutions to drive broader adoption
- Address uncommon scenarios later

Potential Pitfalls

Outliers

Complexity

Over-Engineering

Exceptions

Fear of the Unknown

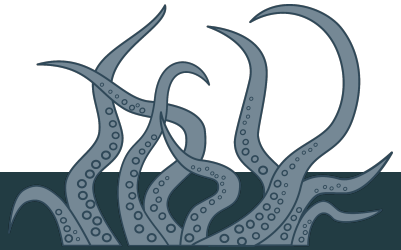
One-Off Solutions

Resistance to Change

Ignoring Feedback

Lack of Product Mindset

Vendor Lock-In



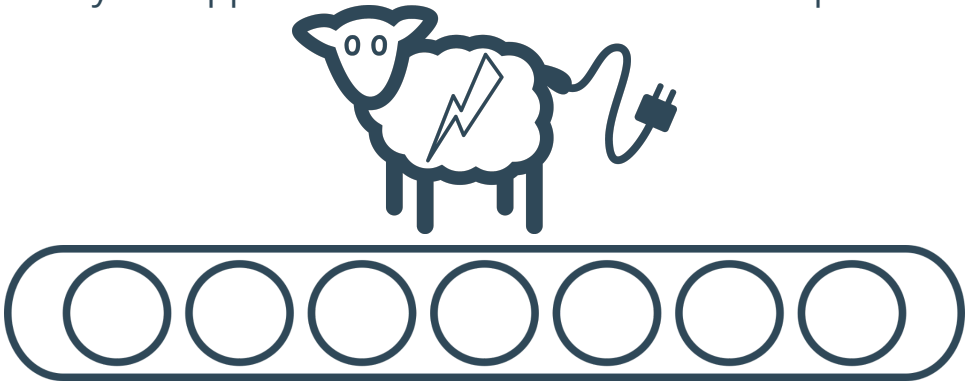
Is Platform Engineering Right for You?

Ask Yourself:

- How many electric sheep do you produce?
- How many developers do you have?
- How many teams do you support?

Do you have:

- A complex software development process?
- A lot of different tools and services to integrate?
- Extensive compliance and security requirements?



I am Smitty and I am afraid of Robots

Brett Smith <bc.smith@sas.com>

THE LINUX FOUNDATION
OPEN SOURCE SUMMIT
NORTH AMERICA
Embedded Linux Conference