Streamlining DevEx

The Power of CI/CD Standardization and Interoperability

FOSDEM'24

@IAmJerdog

Jeremy Meiss

Co-Founder

DevEx Startup

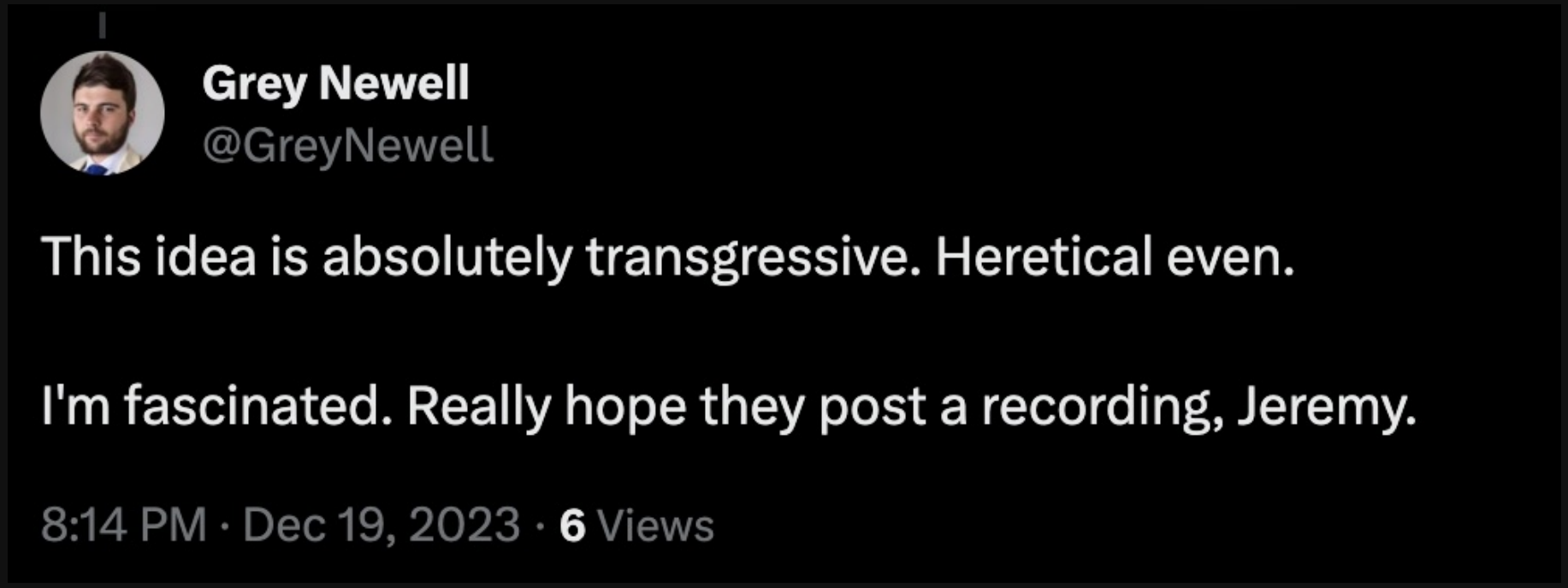
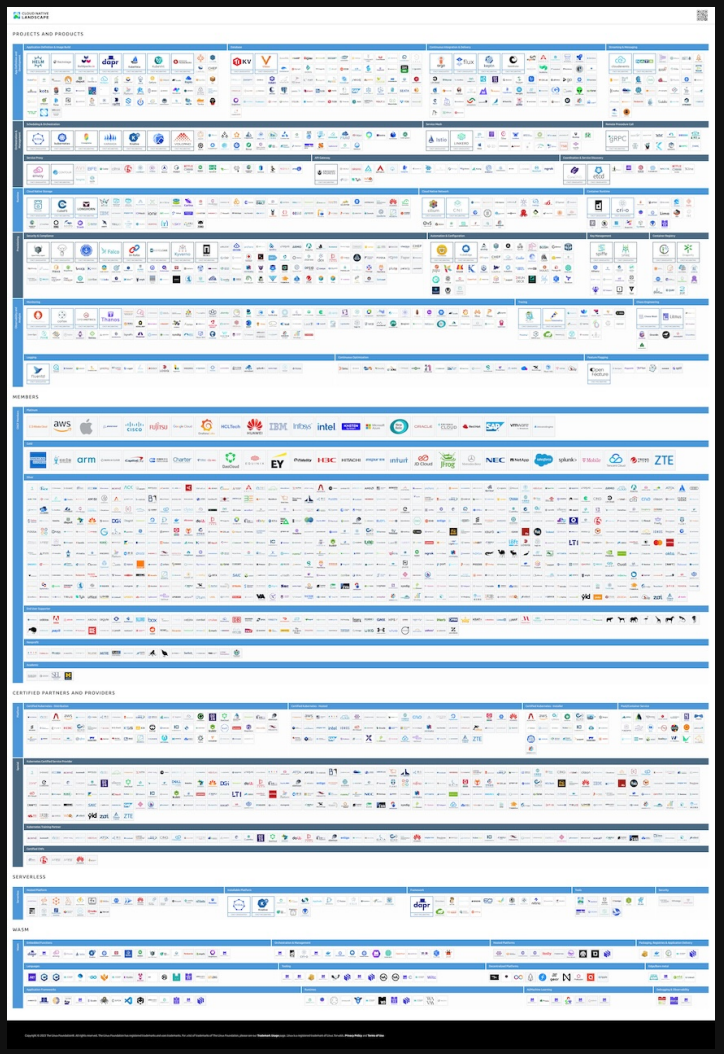
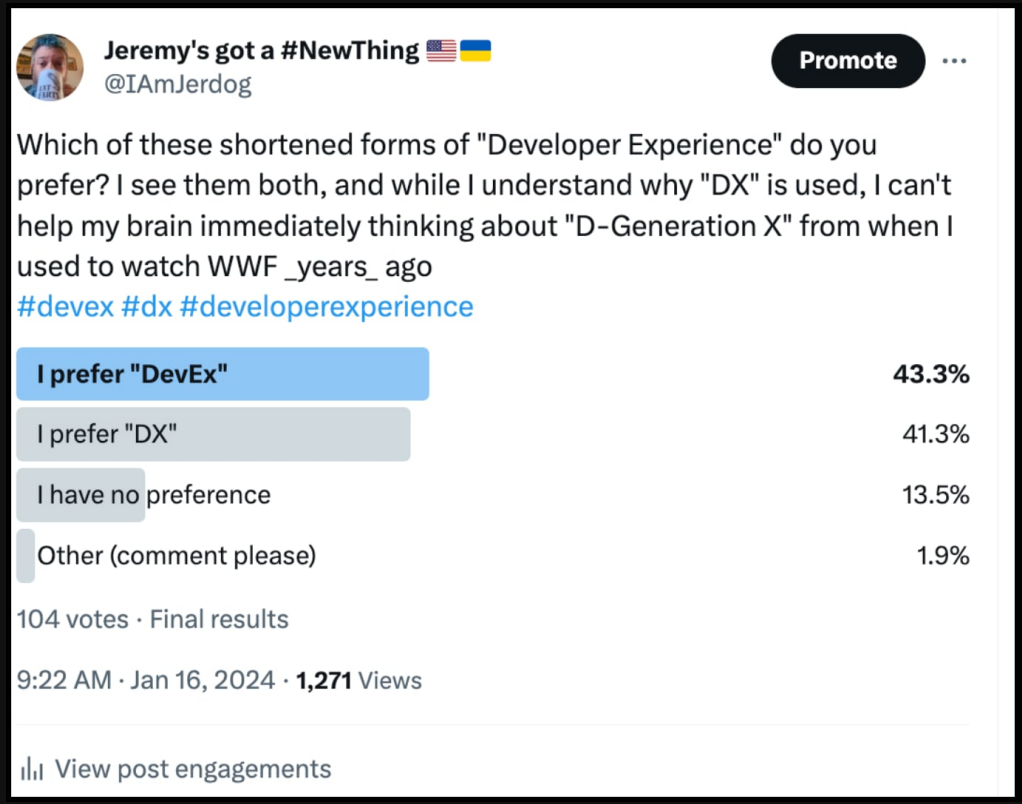


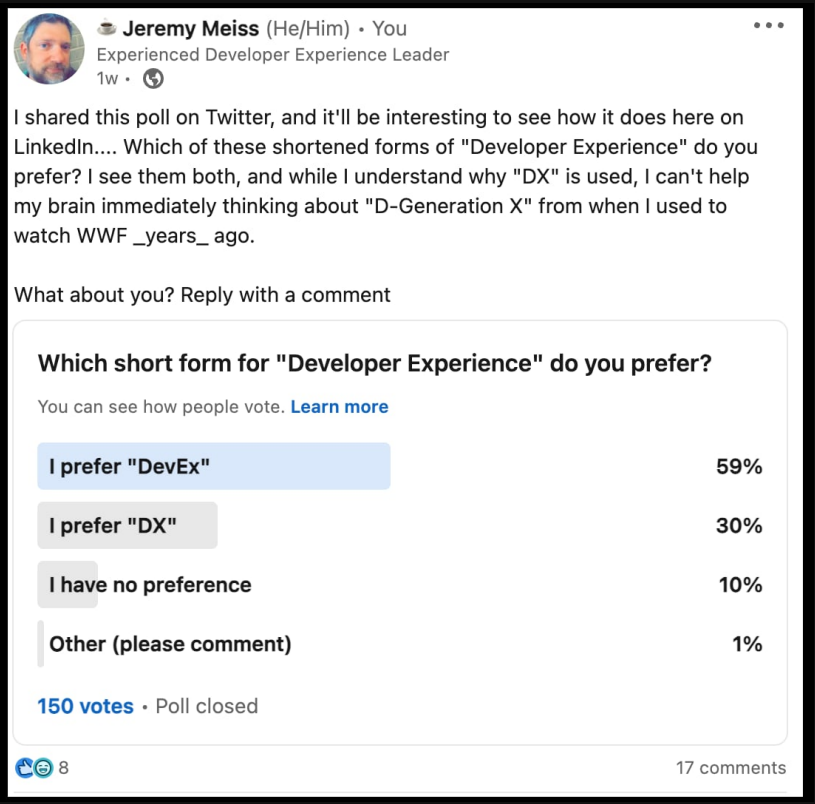
Image: @GreyNewell on Twitter



REF: CNCF Landscape, 29-Jan-2024

Developer Experience





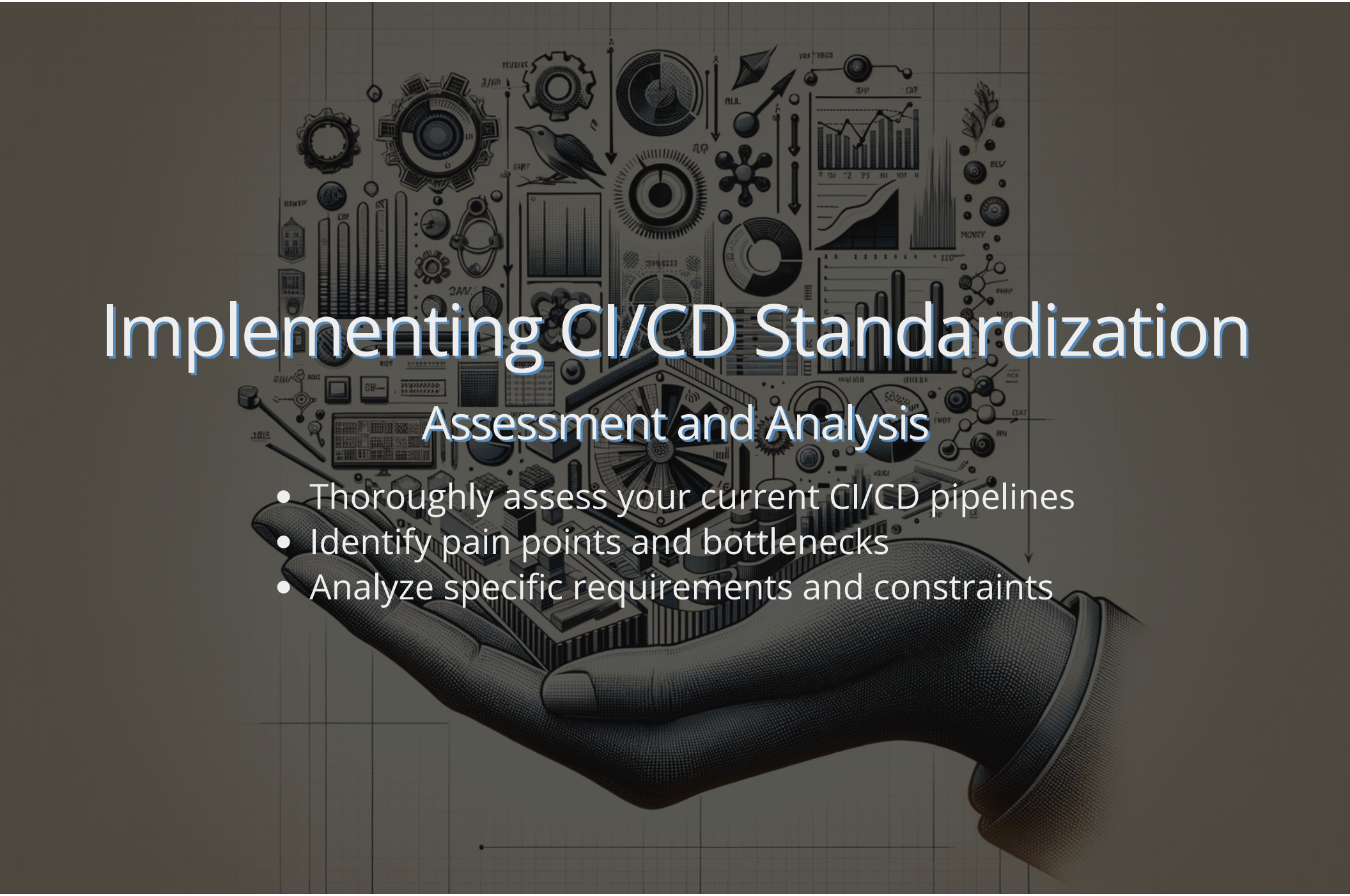
52% of respondents said “DevEx” (34% “DX”)


Developer Experience (DevEx)...
...encompasses the journey of developers as they learn and deploy technology. When successful, it focuses on eliminating obstacles that hinder a developer or practitioner from achieving success in their endeavors.


CI/CD Standardization


CI/CD Interoperability


Implementing CI/CD
Standardization



Implementing CI/CD Standardization

Assessment and Analysis

- Thoroughly assess your current CI/CD pipelines
- Identify pain points and bottlenecks
- Analyze specific requirements and constraints

Implementing CI/CD Standardization

Define Standardization Goals

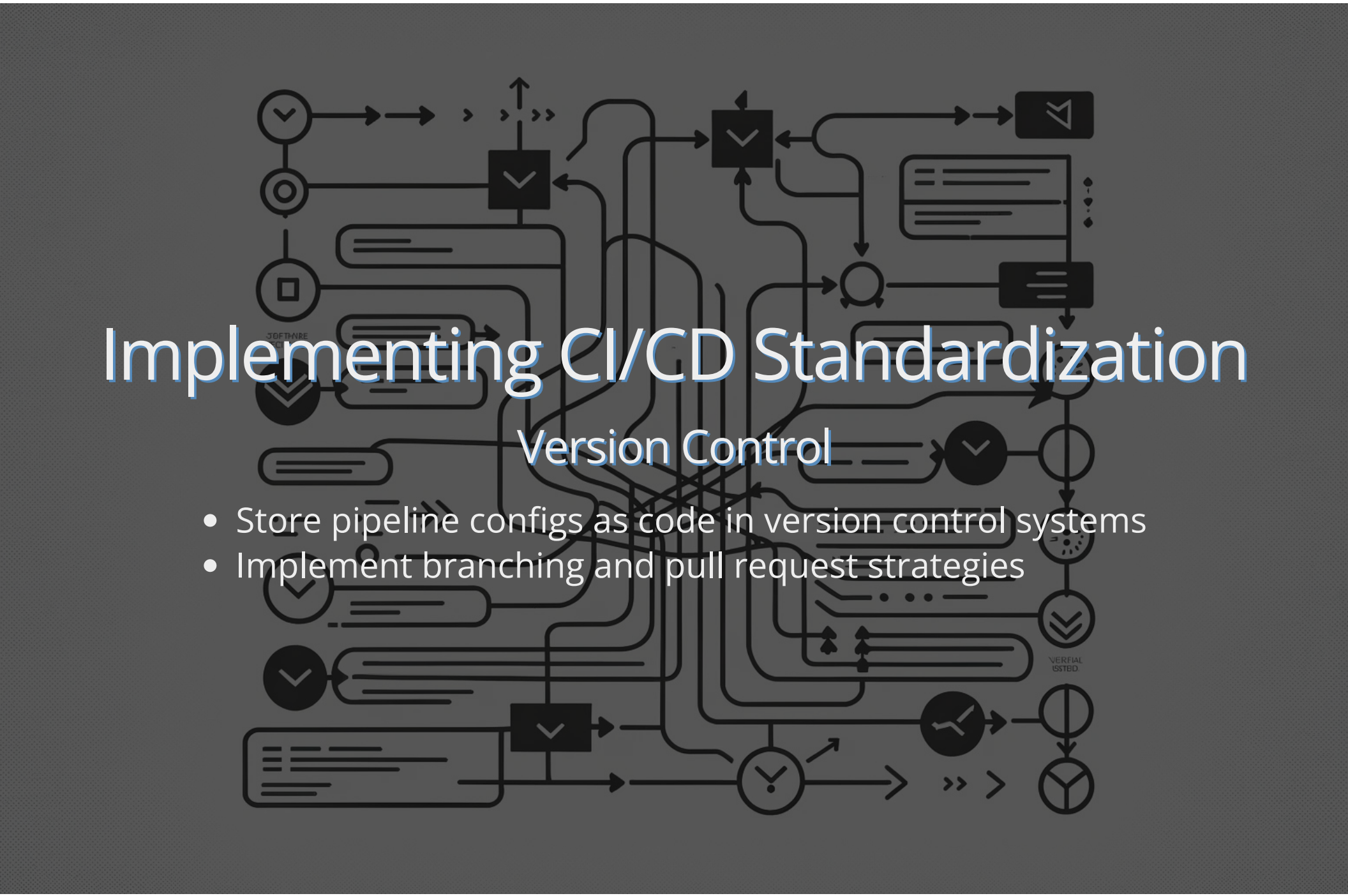
- Define goals and objectives, align with strategy and objectives
- Determine success, like reduced deployment times / error rates


Implementing CI/CD Standardization
Select Standardization Tools and Practices
• Choose tools & practices aligned with organization needs, goals
• Establish standard templates and configurations for pipelines
• Enforce coding standards for consistency and readability

Implementing CI/CD Standardization

Documentation and Training

- Create comprehensive docs for processes, configs, best practices
- Provide training to ensure understanding and effective use



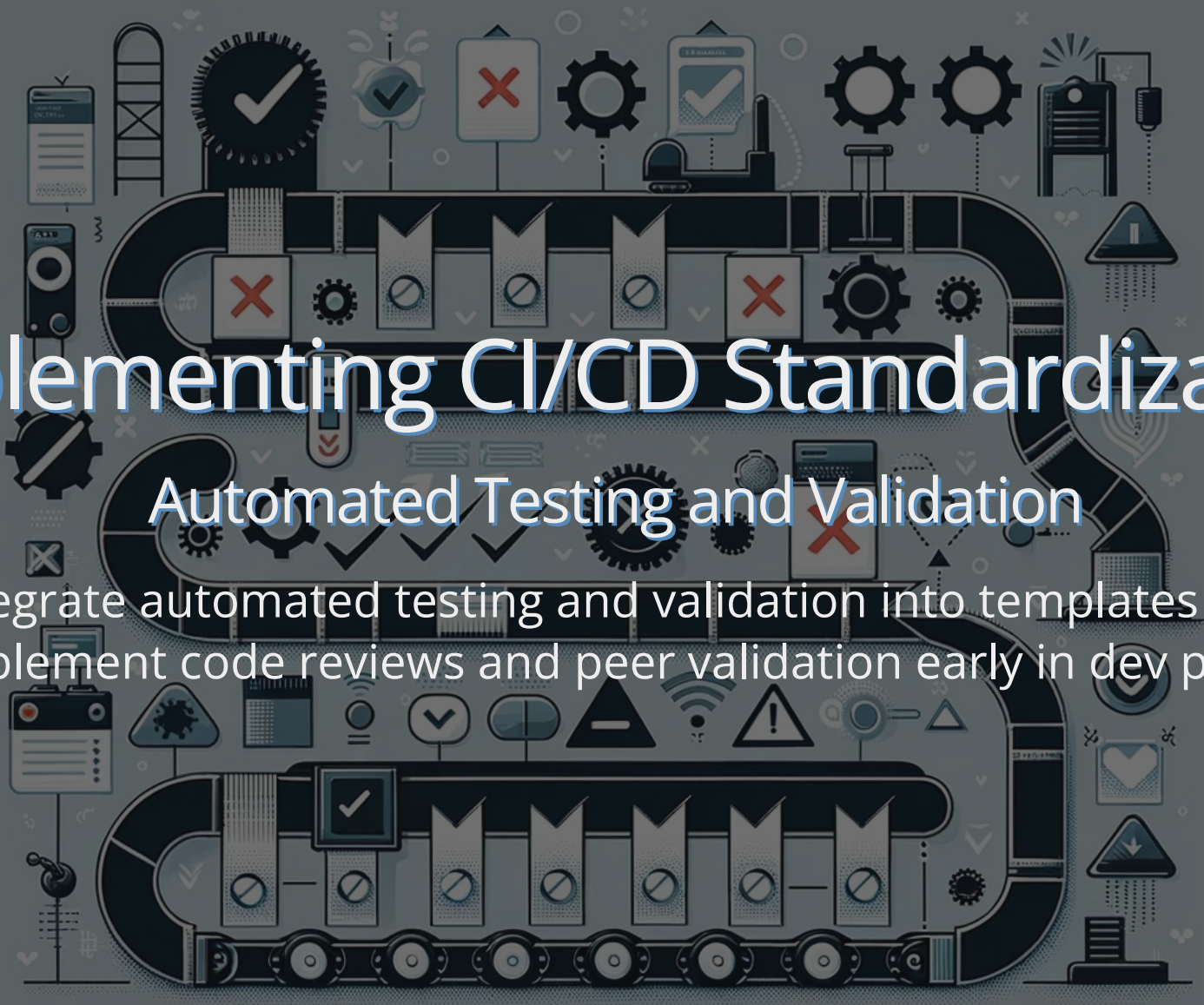
Implementing CI/CD Standardization

Version Control

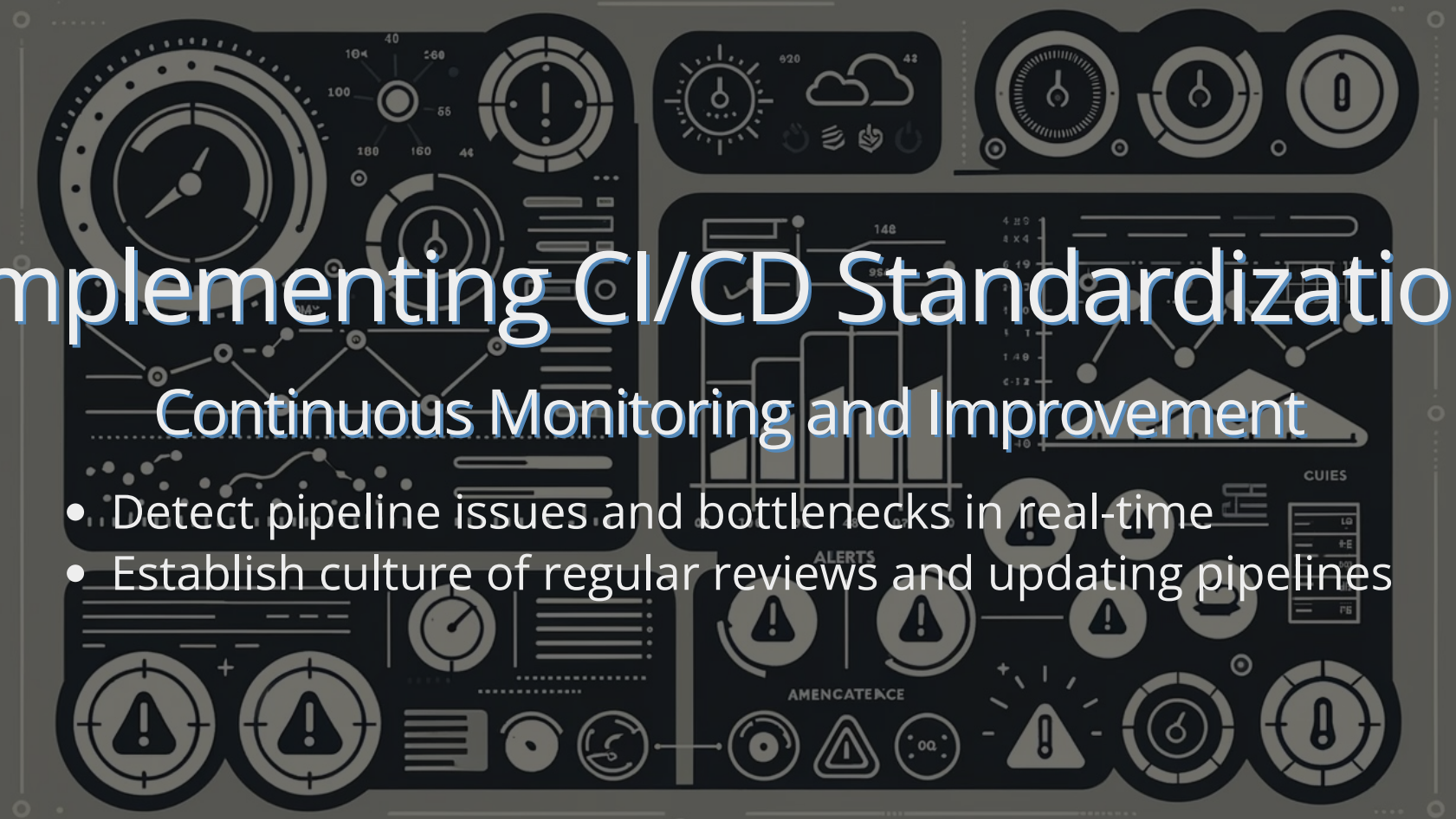
- Store pipeline configs as code in version control systems
- Implement branching and pull request strategies

Implementing CI/CD Standardization

Automated Testing and Validation



- Integrate automated testing and validation into templates
- Implement code reviews and peer validation early in dev process



Implementing CI/CD Standardization

Continuous Monitoring and Improvement

- Detect pipeline issues and bottlenecks in real-time
- Establish culture of regular reviews and updating pipelines

Implementing CI/CD Standardization

Governance and Compliance

- Implement governance policies to enforce pipeline standards
- Validate compliance with industry regulations / internal standards
- Regularly audit and assess adherence to standardized practices

Implementing CI/CD Standardization

Scaling and Adaptation

- Ensure standardized templates can scale and adapt
- Maintain flexibility to accommodate unique project requirements

Implementing CI/CD Standardization

Feedback Loop and Collaboration

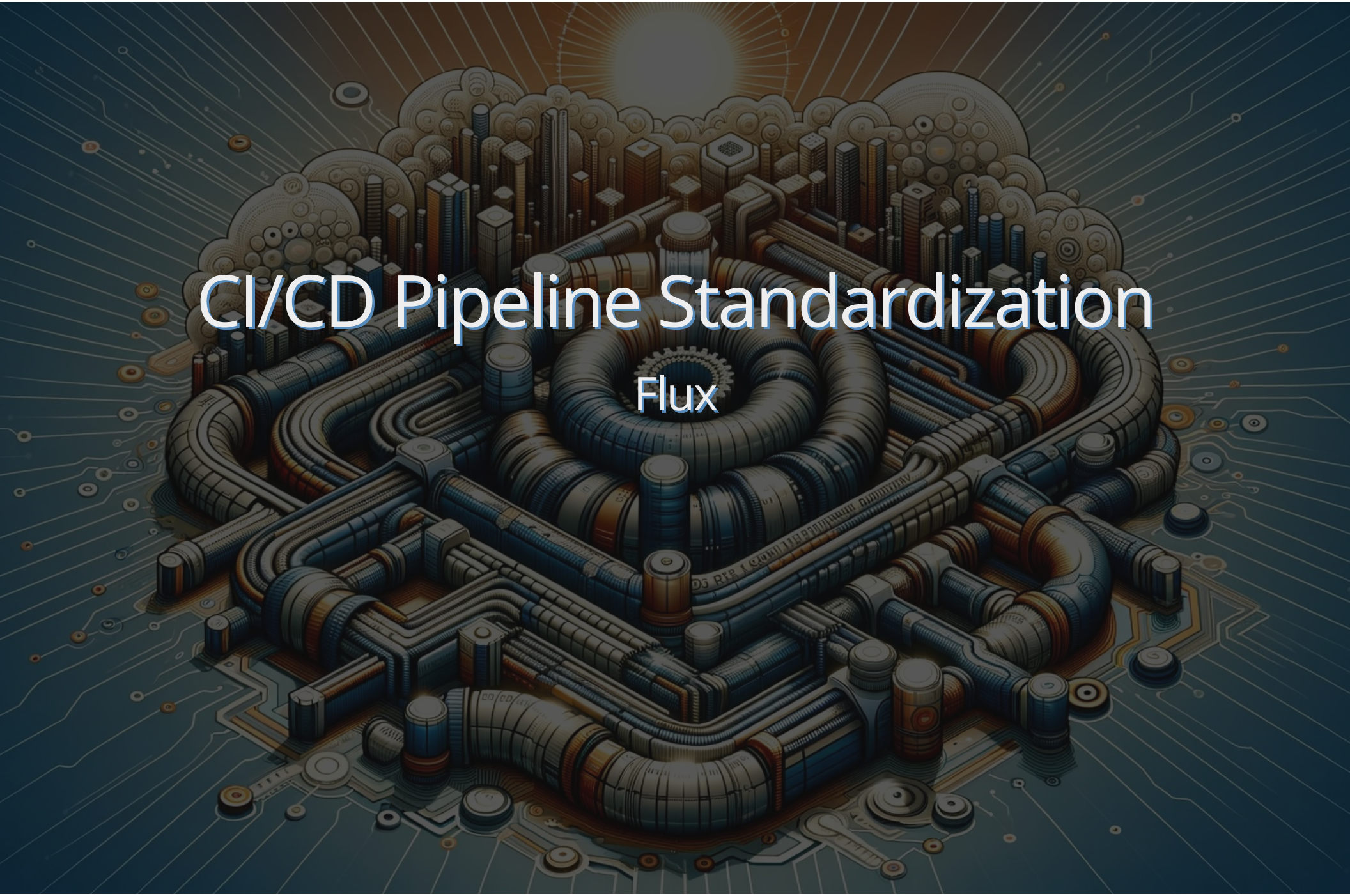
- Foster collaborative environments where feedback & contributions encouraged
- Continuously communicate benefits of standardized pipelines & celebrate successes

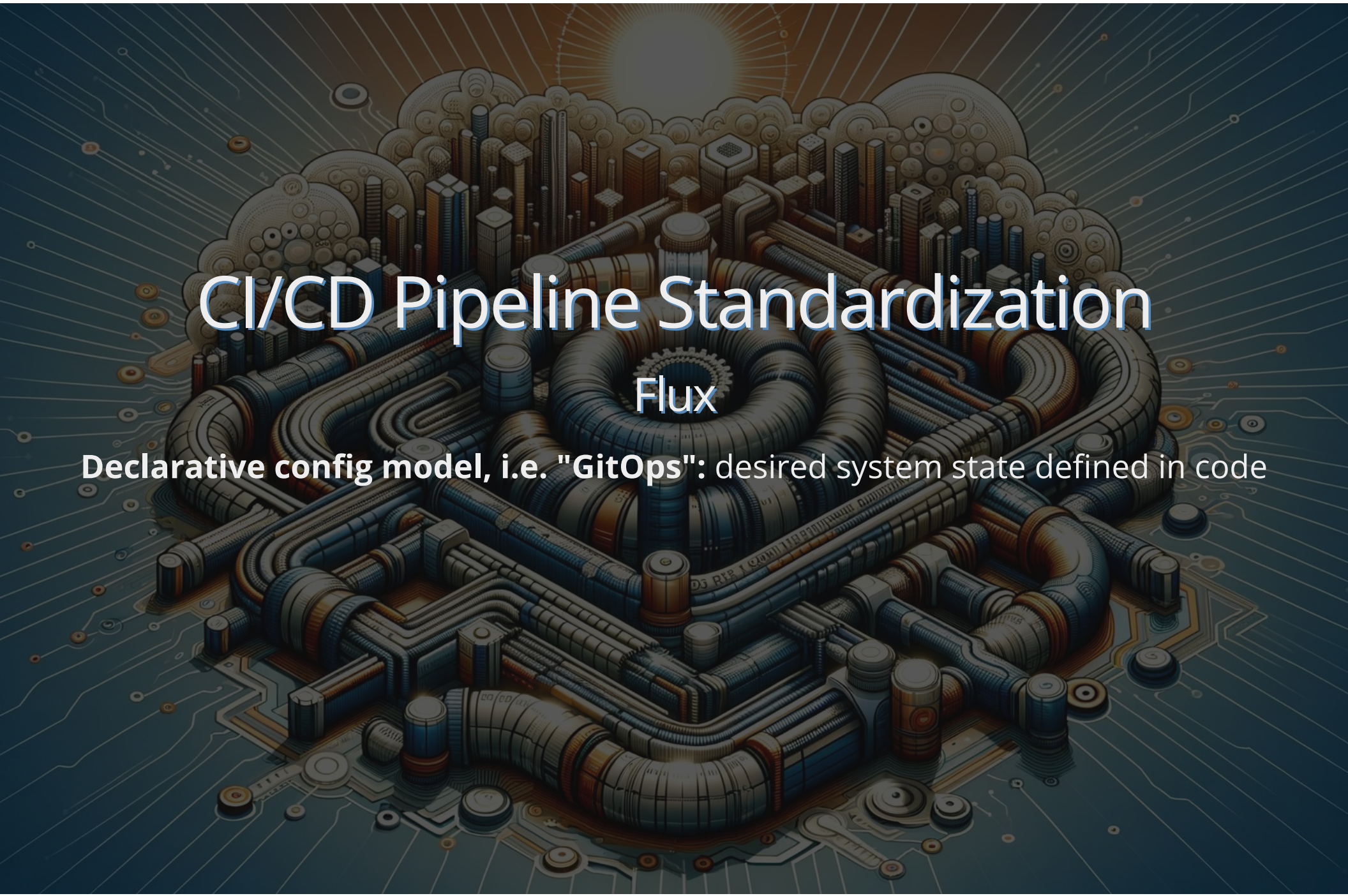

CI/CD Pipeline Standardization
Argo


CI/CD Pipeline Standardization
Argo
Reusable workflows: orgs define reusable workflow templates

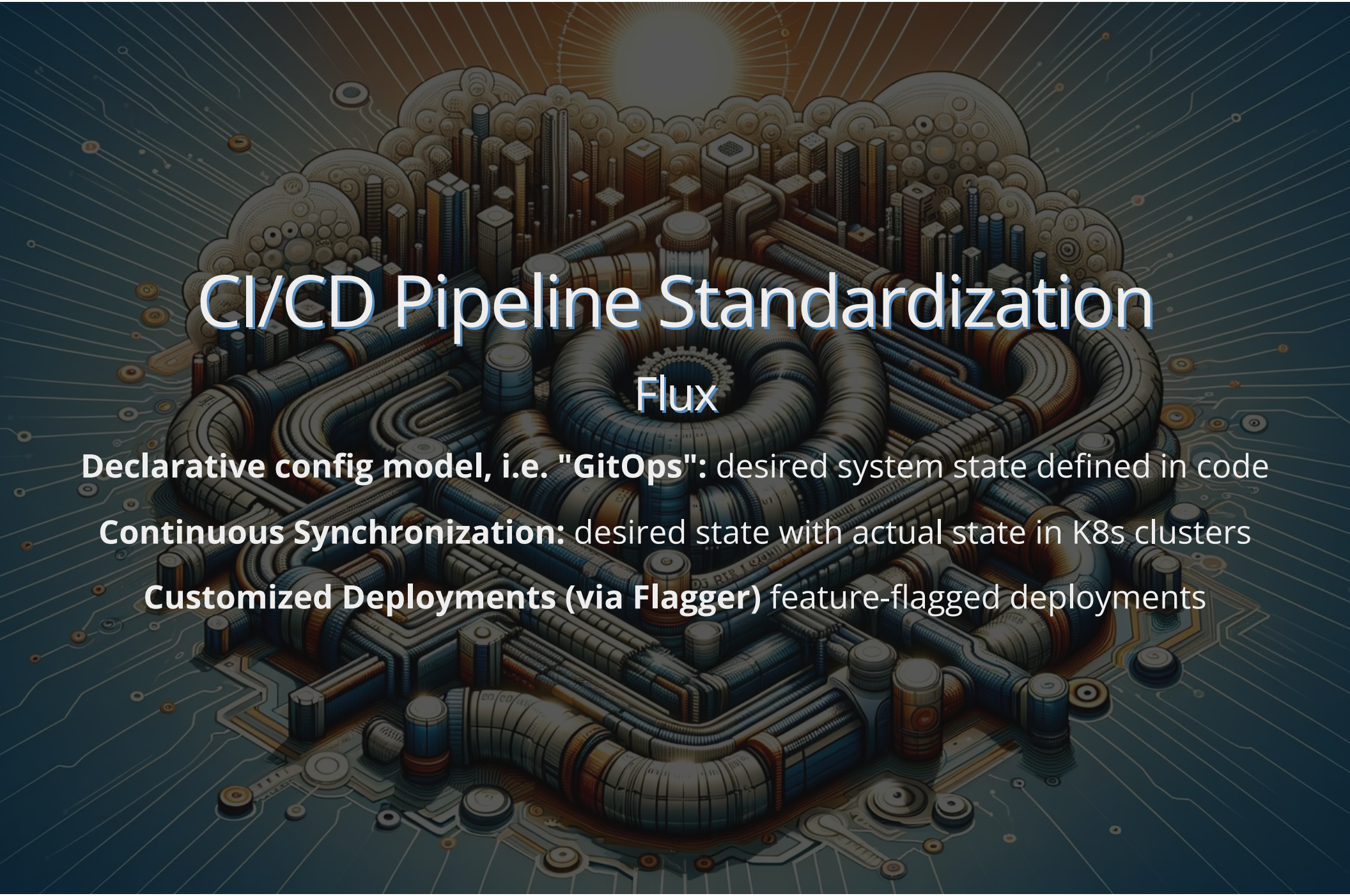

CI/CD Pipeline Standardization
Argo
Reusable workflows: orgs define reusable workflow templates
GitOps principles: CI/CD configs & workflows managed as code


CI/CD Pipeline Standardization
Argo
Reusable workflows: orgs define reusable workflow templates
GitOps principles: CI/CD configs & workflows managed as code
Artifact Management Support: artifacts managed & stored for consistency


CI/CD Pipeline Standardization
Flux


CI/CD Pipeline Standardization
Flux
Declarative config model, i.e. "GitOps": desired system state defined in code


CI/CD Pipeline Standardization
Flux
Declarative config model, i.e. "GitOps": desired system state defined in code
Continuous Synchronization: desired state with actual state in K8s clusters


CI/CD Pipeline Standardization
Flux
Declarative config model, i.e. "GitOps": desired system state defined in code
Continuous Synchronization: desired state with actual state in K8s clusters
Customized Deployments (via Flagger) feature-flagged deployments


Achieving Standardized Workflows

Achieving Standardized Workflows

Argo & Flux: encourage standardized templates/definitions

Achieving Standardized Workflows

Argo & Flux: encourage standardized templates/definitions

VCS & CI/CD Integrations: ensures configs maintained and accessible to all

Achieving Standardized Workflows

Argo & Flux: encourage standardized templates/definitions

VCS & CI/CD Integrations: ensures configs maintained and accessible to all

Documentation & Training: responsibility of org for devs understanding process

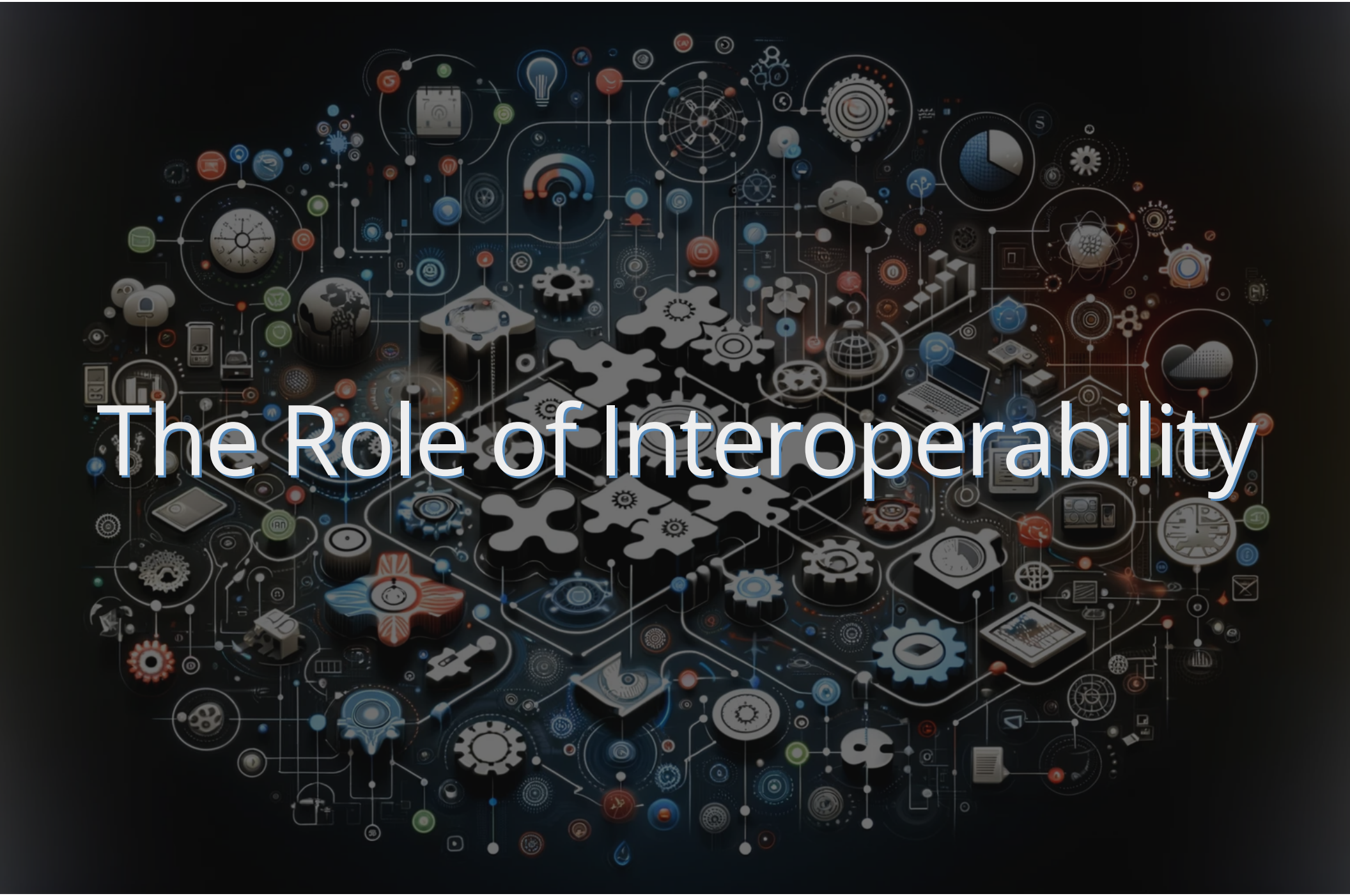
Achieving Standardized Workflows

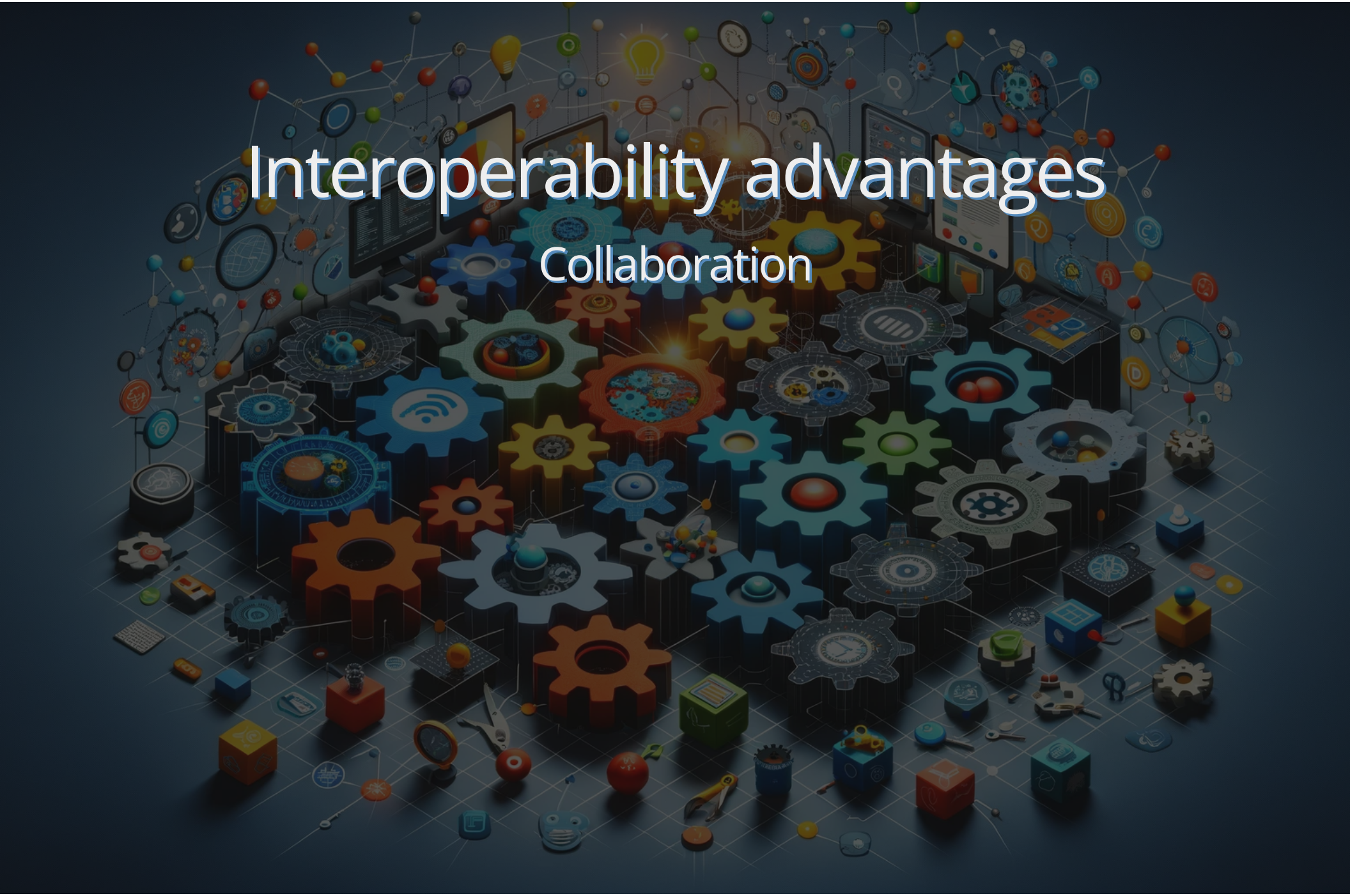
Argo & Flux: encourage standardized templates/definitions

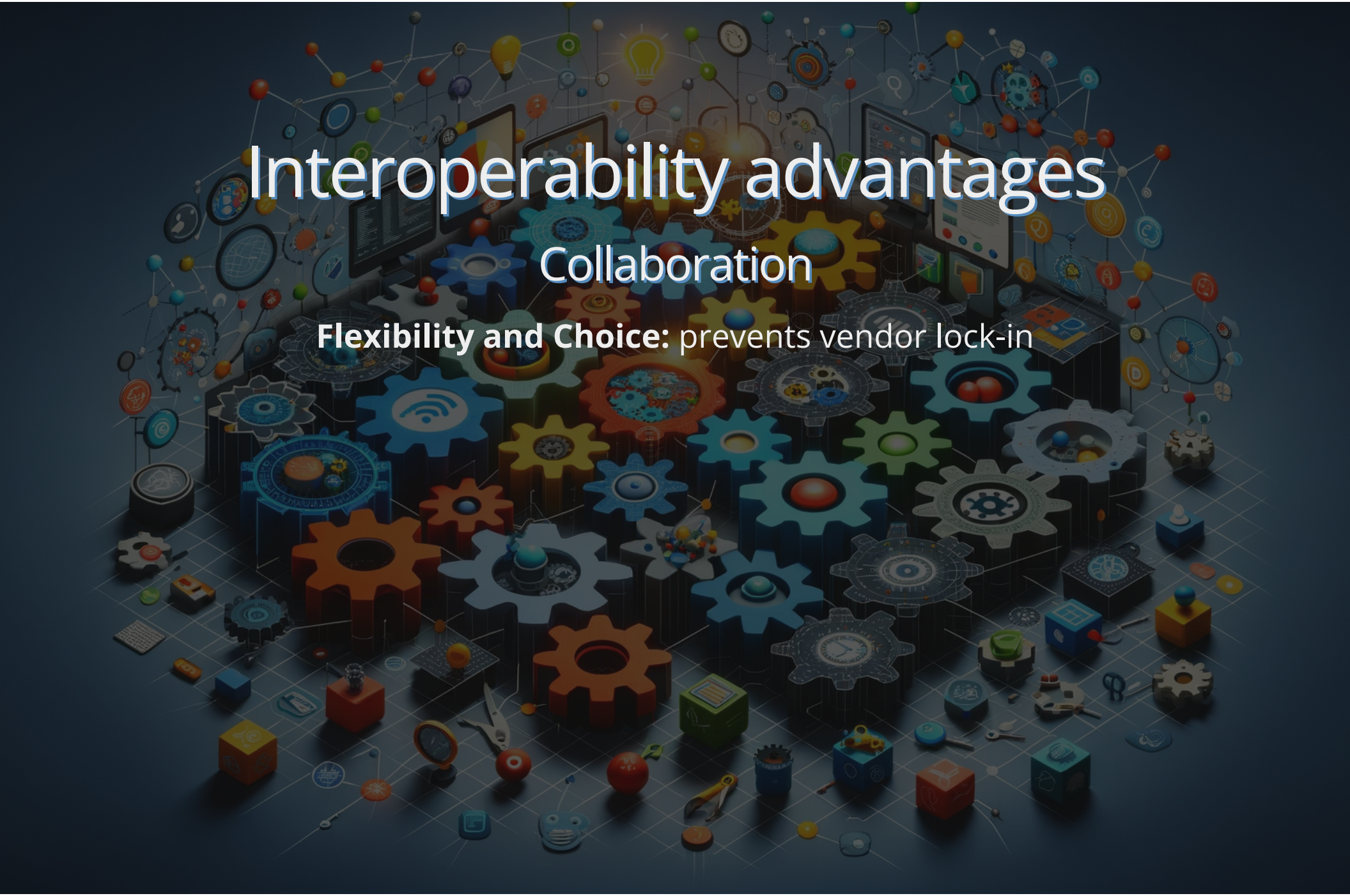
VCS & CI/CD Integrations: ensures configs maintained and accessible to all

Documentation & Training: responsibility of org for devs understanding process

Continuous Improvement: foster continual improvement & gathering feedback


The Role of Interoperability

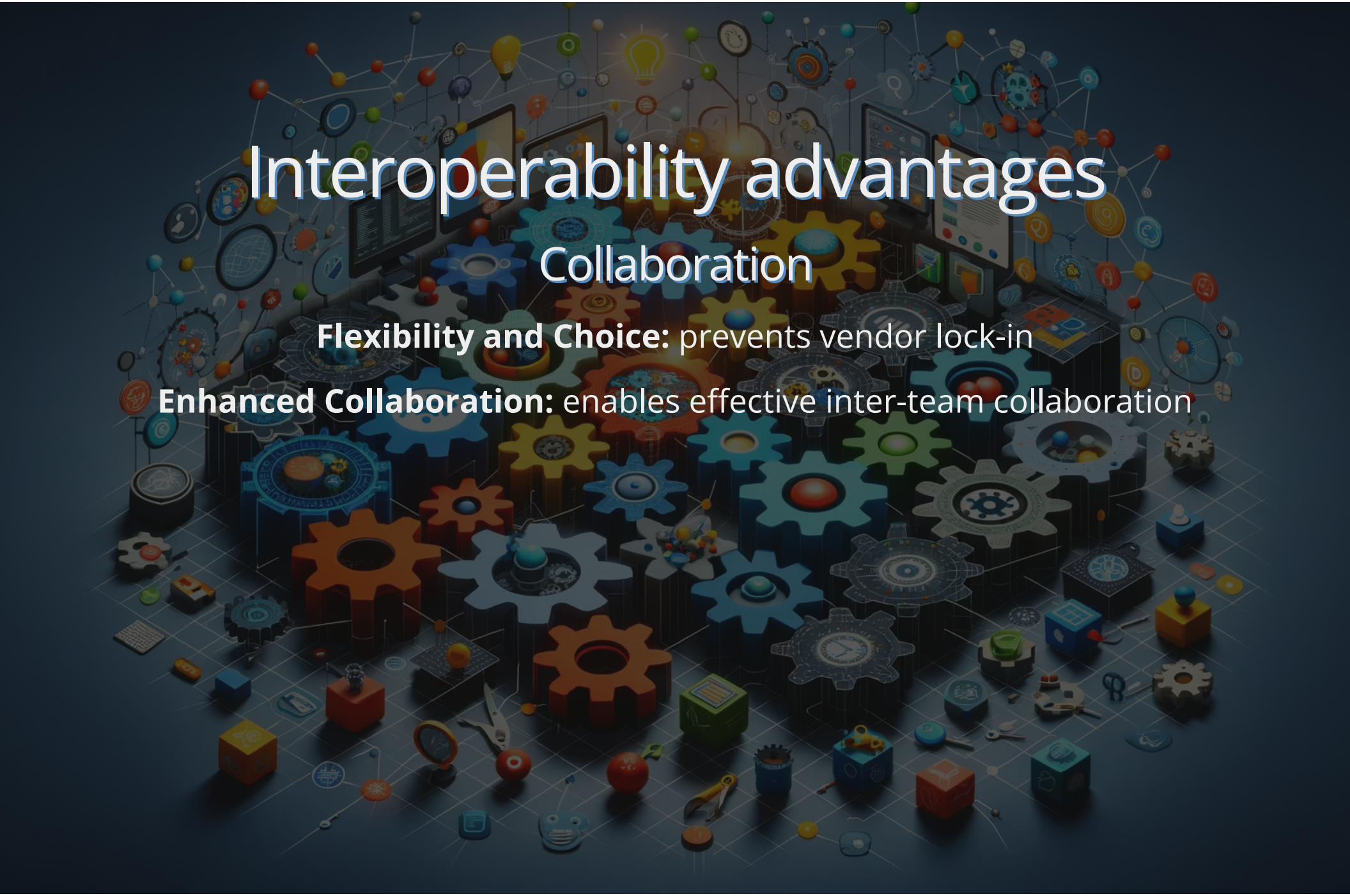

Interoperability advantages
Collaboration



Interoperability advantages

Collaboration

Flexibility and Choice: prevents vendor lock-in

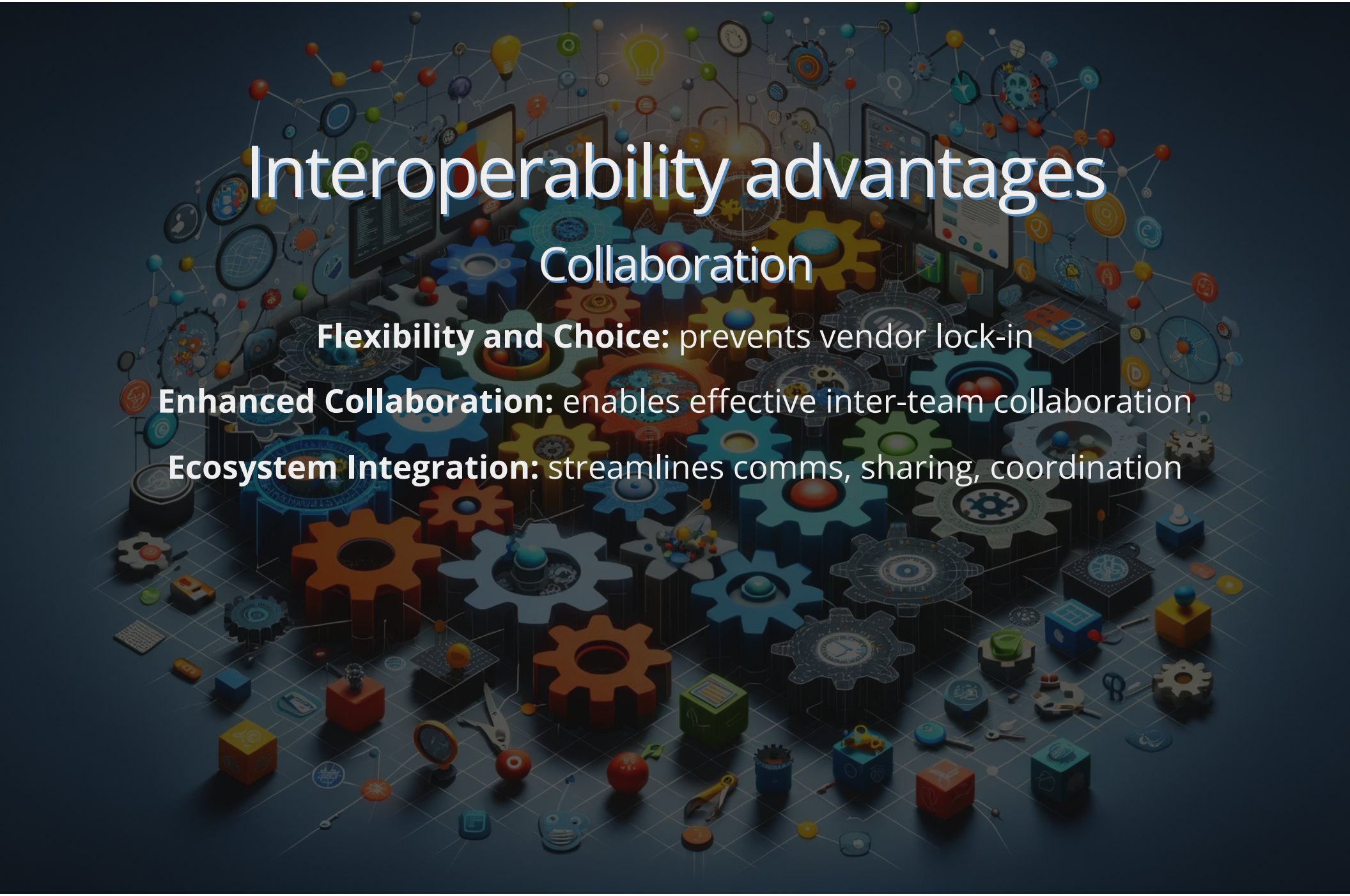


Interoperability advantages

Collaboration

Flexibility and Choice: prevents vendor lock-in

Enhanced Collaboration: enables effective inter-team collaboration



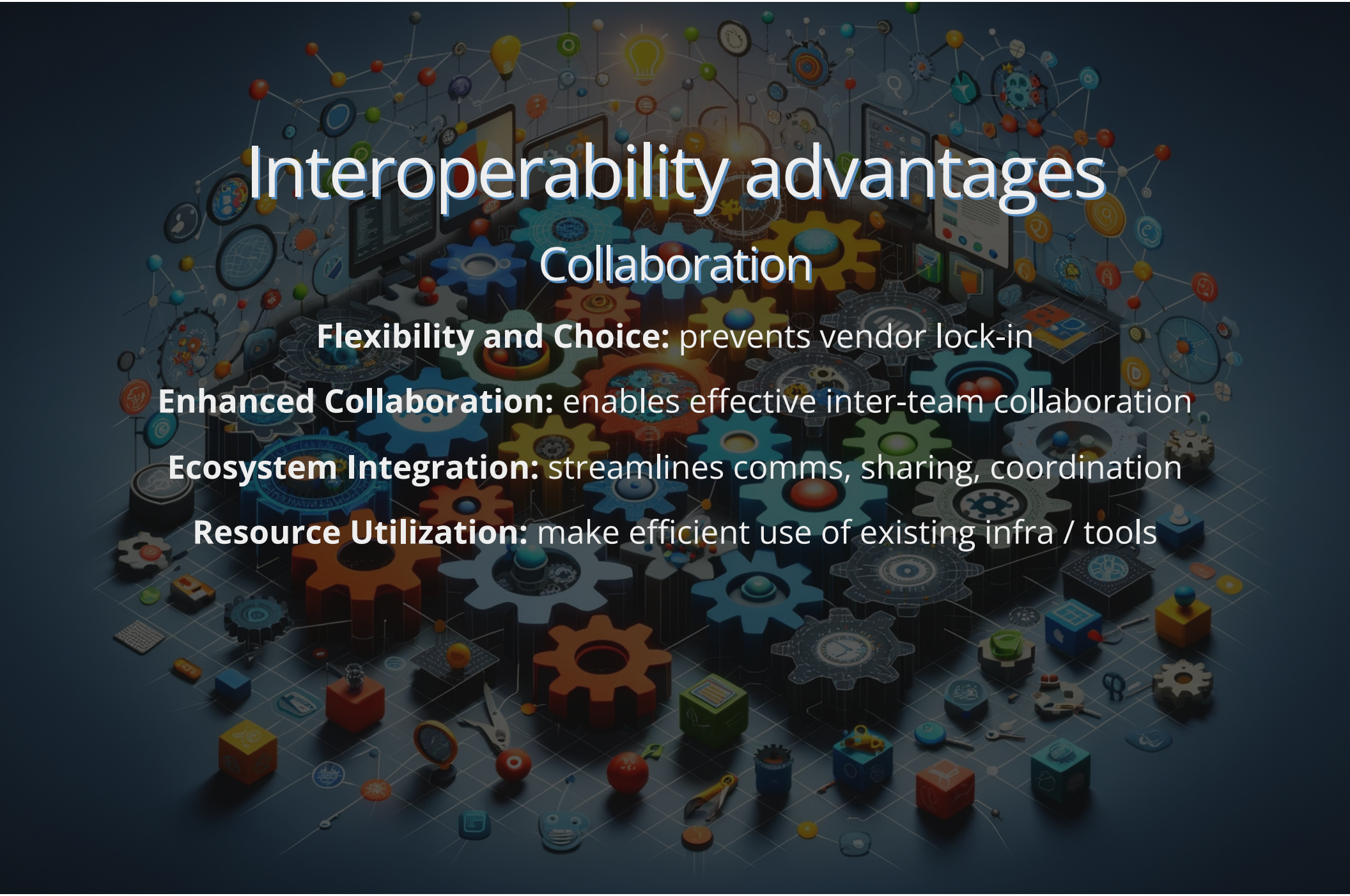
Interoperability advantages

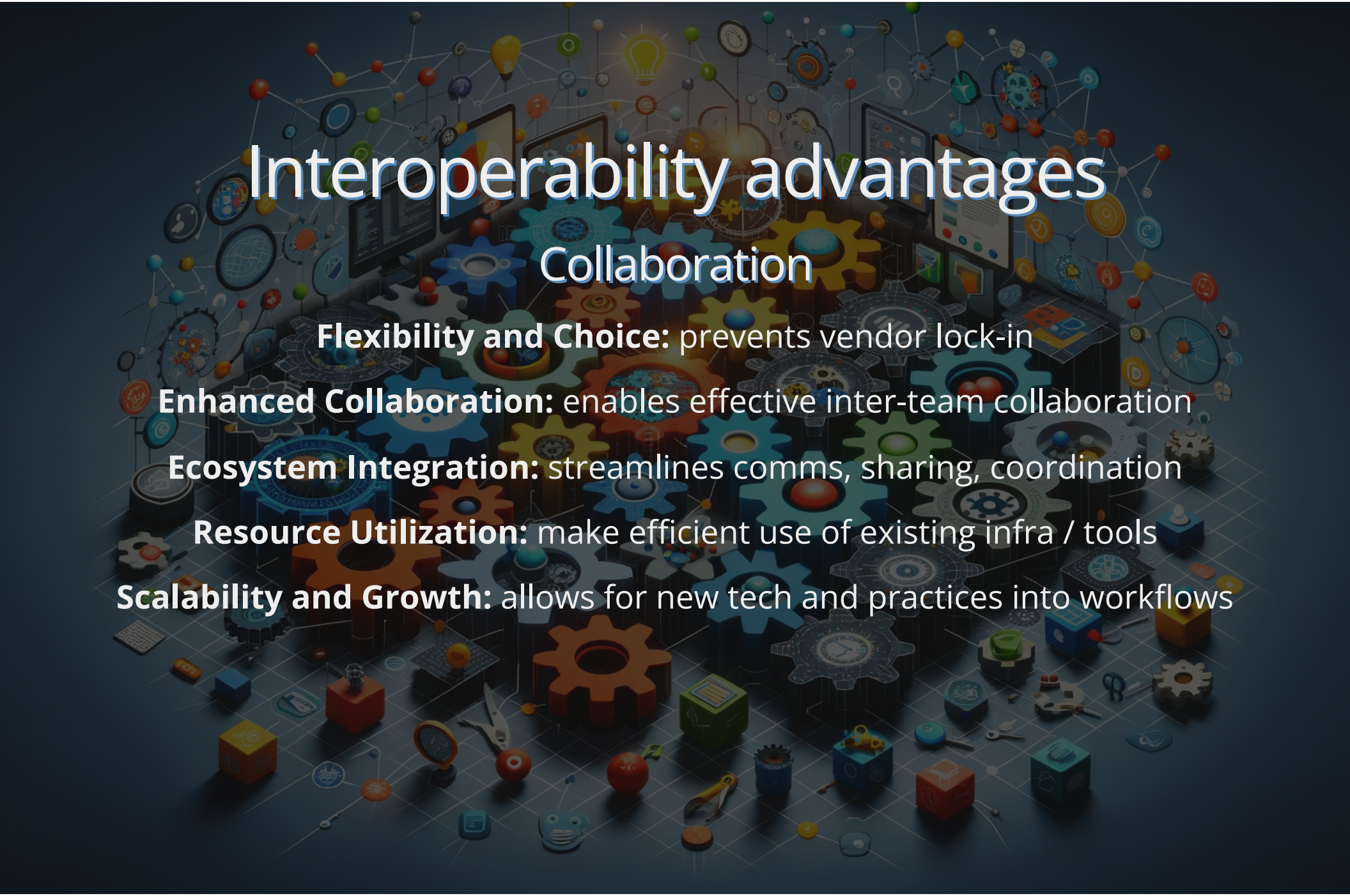
Collaboration

Flexibility and Choice: prevents vendor lock-in

Enhanced Collaboration: enables effective inter-team collaboration

Ecosystem Integration: streamlines comms, sharing, coordination


Interoperability advantages
Collaboration
Flexibility and Choice: prevents vendor lock-in
Enhanced Collaboration: enables effective inter-team collaboration
Ecosystem Integration: streamlines comms, sharing, coordination
Resource Utilization: make efficient use of existing infra / tools



Interoperability advantages

Collaboration

Flexibility and Choice: prevents vendor lock-in

Enhanced Collaboration: enables effective inter-team collaboration

Ecosystem Integration: streamlines comms, sharing, coordination

Resource Utilization: make efficient use of existing infra / tools

Scalability and Growth: allows for new tech and practices into workflows

Interoperability advantages

Collaboration

Flexibility and Choice: prevents vendor lock-in

Enhanced Collaboration: enables effective inter-team collaboration

Ecosystem Integration: streamlines comms, sharing, coordination

Resource Utilization: make efficient use of existing infra / tools

Scalability and Growth: allows for new tech and practices into workflows

Cross-Platform Deploys: promotes unified deployment / infra mgmt approach

Interoperability advantages

Collaboration

Flexibility and Choice: prevents vendor lock-in

Enhanced Collaboration: enables effective inter-team collaboration

Ecosystem Integration: streamlines comms, sharing, coordination

Resource Utilization: make efficient use of existing infra / tools

Scalability and Growth: allows for new tech and practices into workflows

Cross-Platform Deploys: promotes unified deployment / infra mgmt approach

Troubleshooting and Debugging: enables better incident response


Interoperability
Spinnaker & Backstage

Interoperability: Spinnaker


Interoperability: Spinnaker
Integration with Cloud Providers: broad integration options


Interoperability: Spinnaker
Integration with Cloud Providers: broad integration options
VCS Integrations: enable trigger deployment pipelines automation

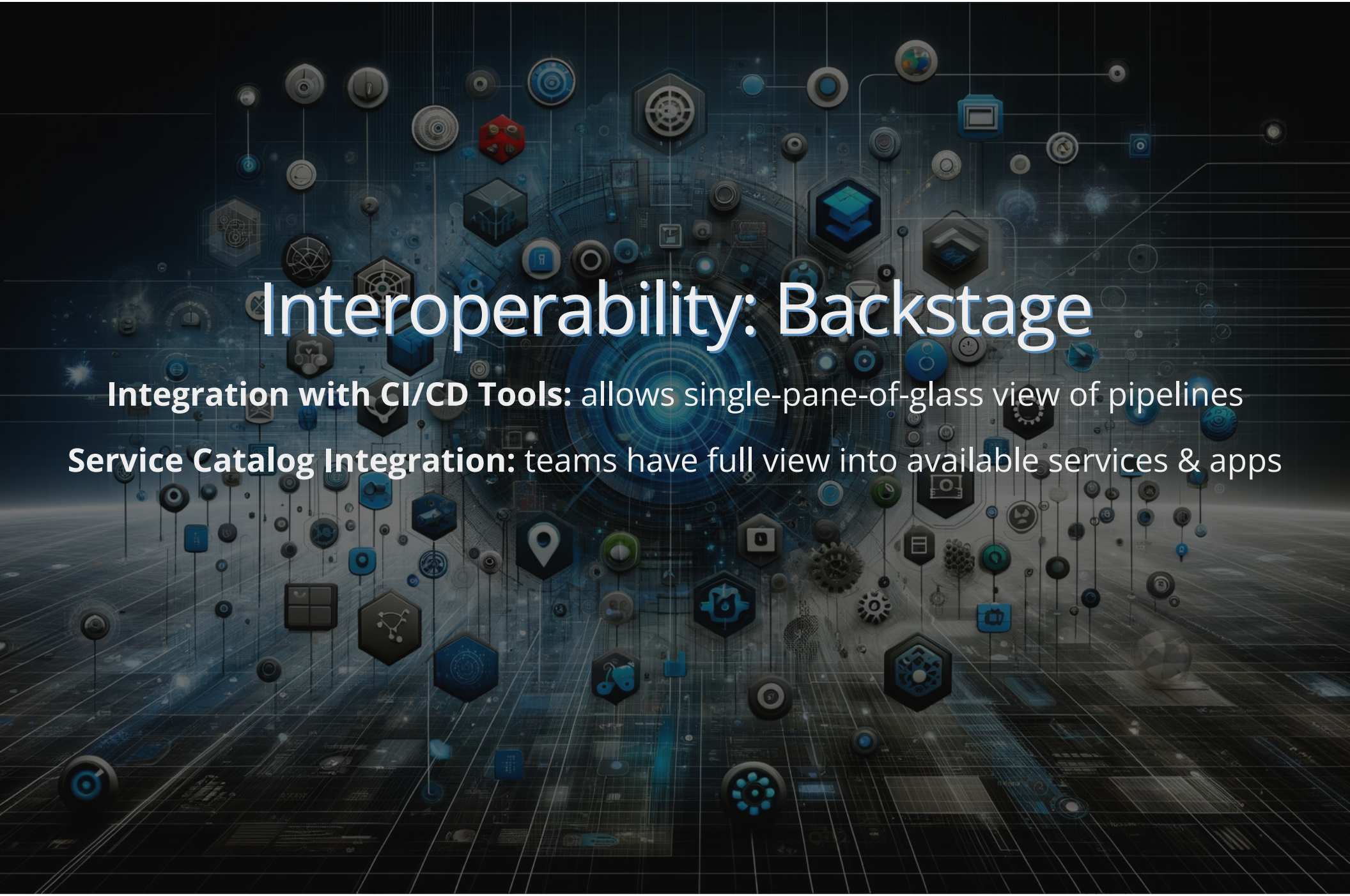

Interoperability: Spinnaker
Integration with Cloud Providers: broad integration options
VCS Integrations: enable trigger deployment pipelines automation
Extensibility: useful integrations, like monitoring, incident mgmt, etc.

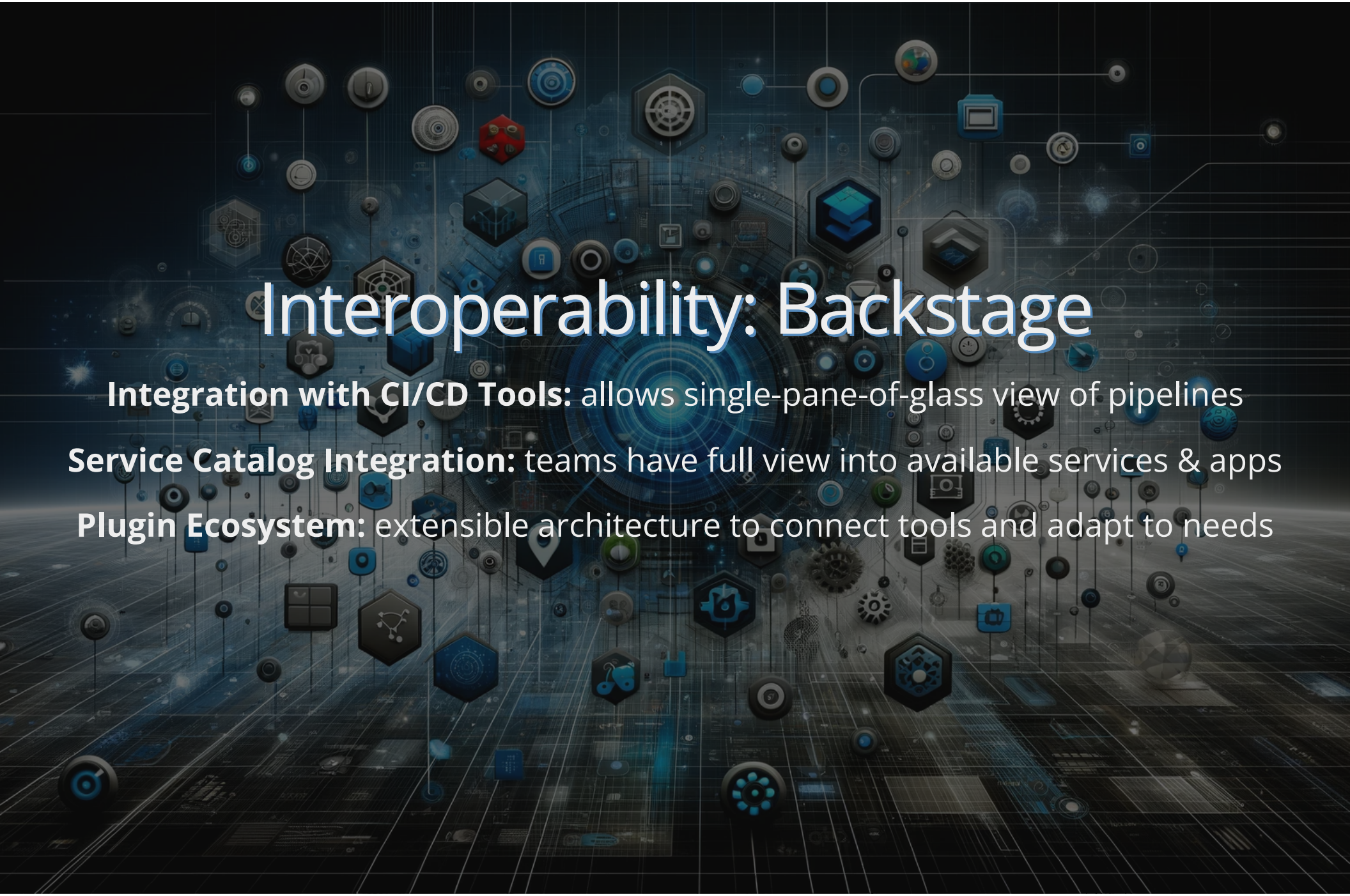

Interoperability: Spinnaker
Integration with Cloud Providers: broad integration options
VCS Integrations: enable trigger deployment pipelines automation
Extensibility: useful integrations, like monitoring, incident mgmt, etc.
Artifact Management: deploy the right artifacts, enhancing reliability

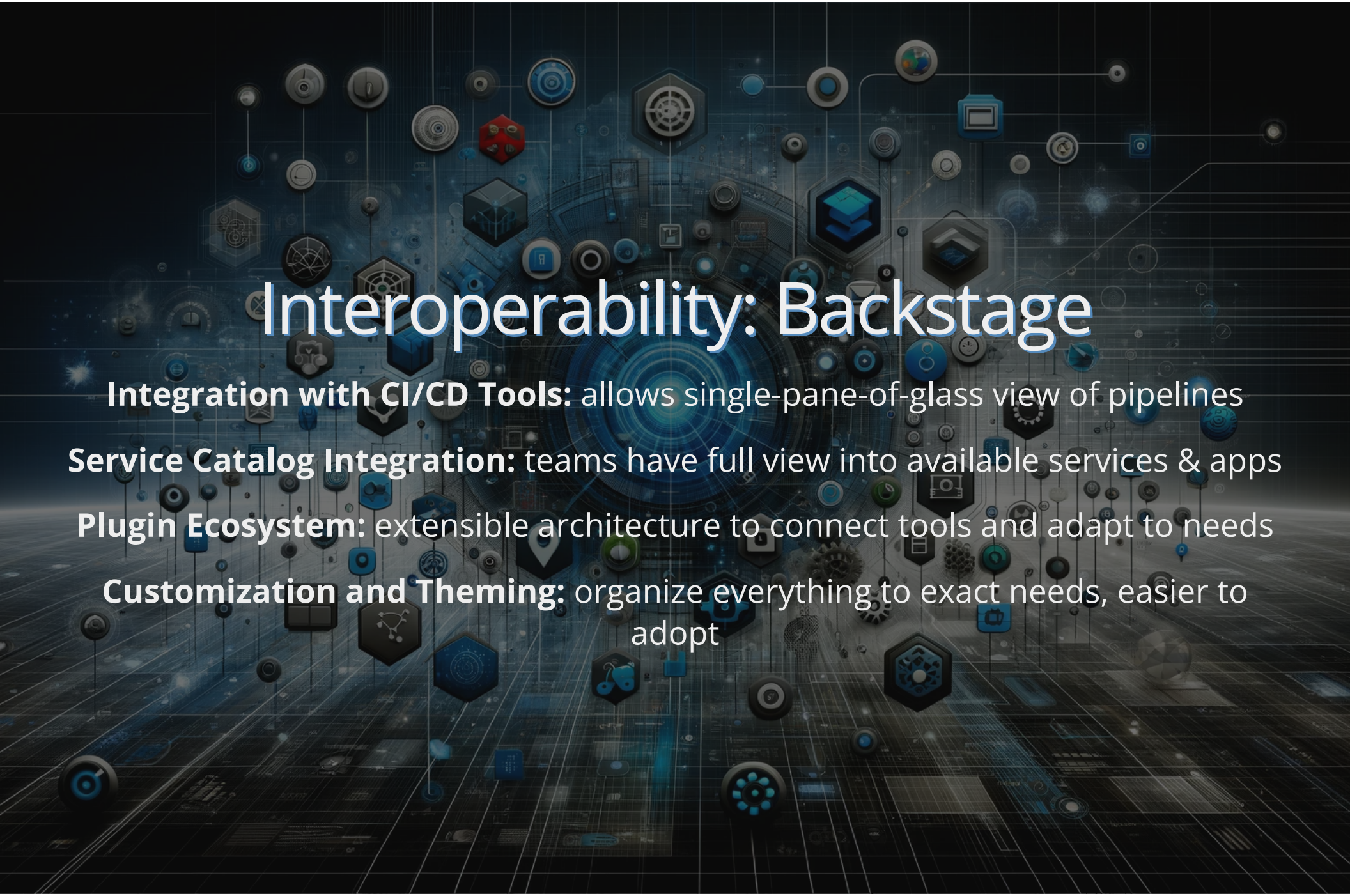

Interoperability: Spinnaker
Integration with Cloud Providers: broad integration options
VCS Integrations: enable trigger deployment pipelines automation
Extensibility: useful integrations, like monitoring, incident mgmt, etc.
Artifact Management: deploy the right artifacts, enhancing reliability
Pipeline Abstraction: flexible / adaptable process as reqs evolve


Interoperability: Backstage


Interoperability: Backstage
Integration with CI/CD Tools: allows single-pane-of-glass view of pipelines


Interoperability: Backstage
Integration with CI/CD Tools: allows single-pane-of-glass view of pipelines
Service Catalog Integration: teams have full view into available services & apps


Interoperability: Backstage
Integration with CI/CD Tools: allows single-pane-of-glass view of pipelines
Service Catalog Integration: teams have full view into available services & apps
Plugin Ecosystem: extensible architecture to connect tools and adapt to needs


Interoperability: Backstage
Integration with CI/CD Tools: allows single-pane-of-glass view of pipelines
Service Catalog Integration: teams have full view into available services & apps
Plugin Ecosystem: extensible architecture to connect tools and adapt to needs
Customization and Theming: organize everything to exact needs, easier to adopt


Challenges implementing interoperability
STOP
STOP
STOP


Challenges implementing interoperability
Diverse Toolsets & Ecosystems: seamless integration challenging
STOP


Challenges implementing interoperability
Diverse Toolsets & Ecosystems: seamless integration challenging
Data Format & Schema Differences: disrupt compatibility and communication
STOP

Challenges implementing interoperability

Diverse Toolsets & Ecosystems: seamless integration challenging

Data Format & Schema Differences: disrupt compatibility and communication

Authentication and Authorization: complexity with methods and permissions

Challenges implementing interoperability

Diverse Toolsets & Ecosystems: seamless integration challenging

Data Format & Schema Differences: disrupt compatibility and communication

Authentication and Authorization: complexity with methods and permissions

Versioning and Compatibility: breaking changes require ongoing maintenance

Challenges implementing interoperability

Diverse Toolsets & Ecosystems: seamless integration challenging

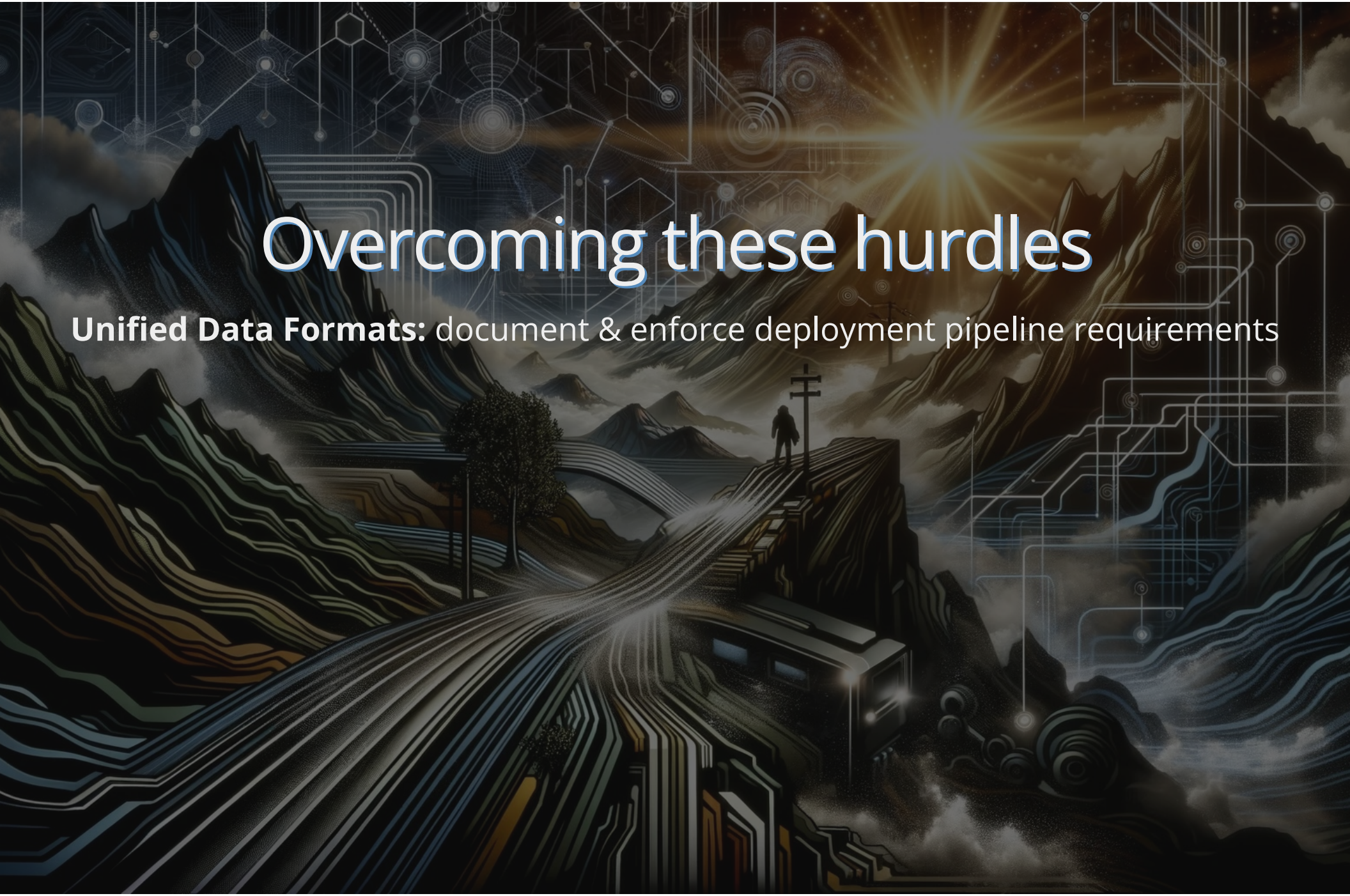
Data Format & Schema Differences: disrupt compatibility and communication

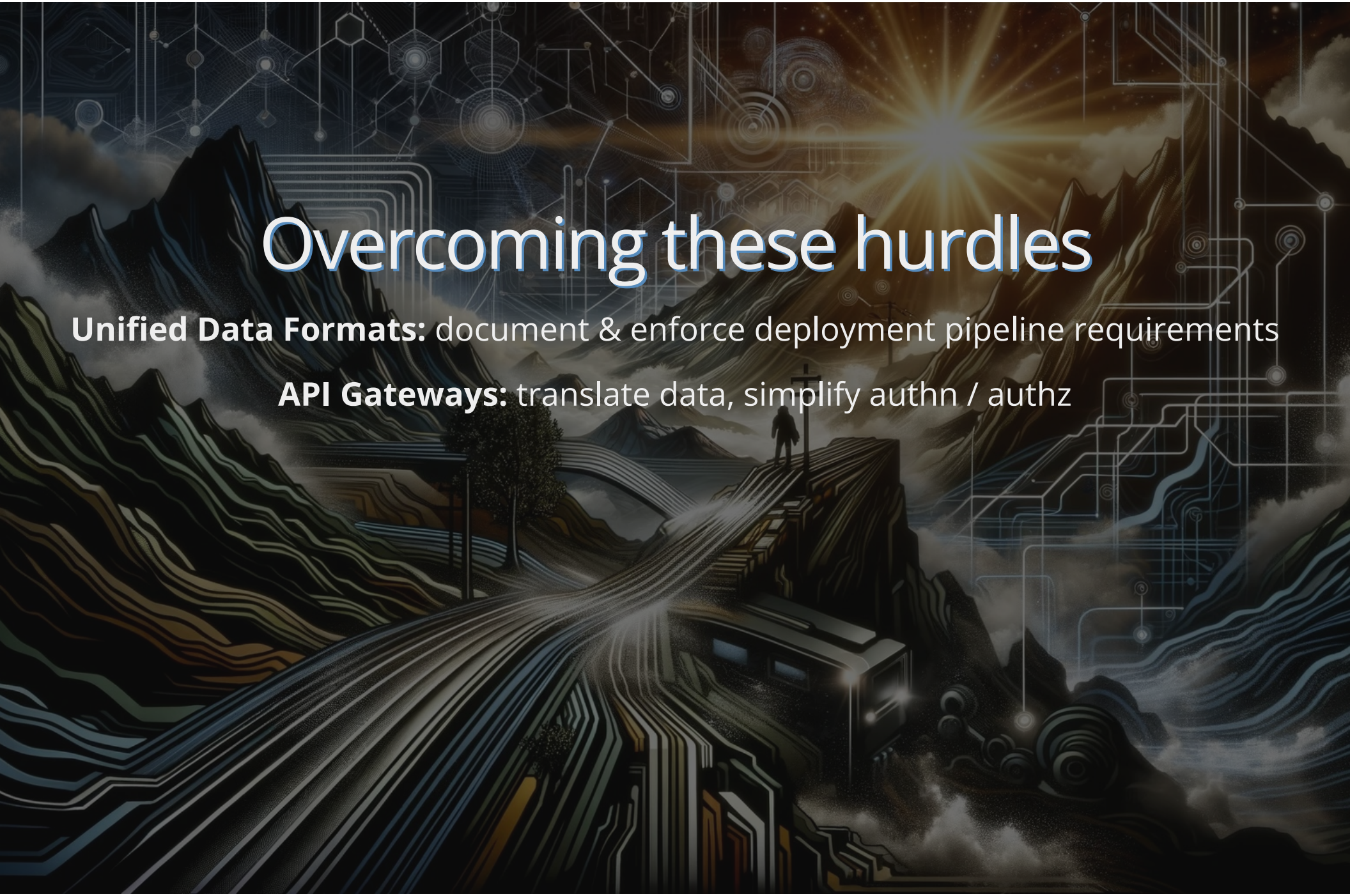
Authentication and Authorization: complexity with methods and permissions

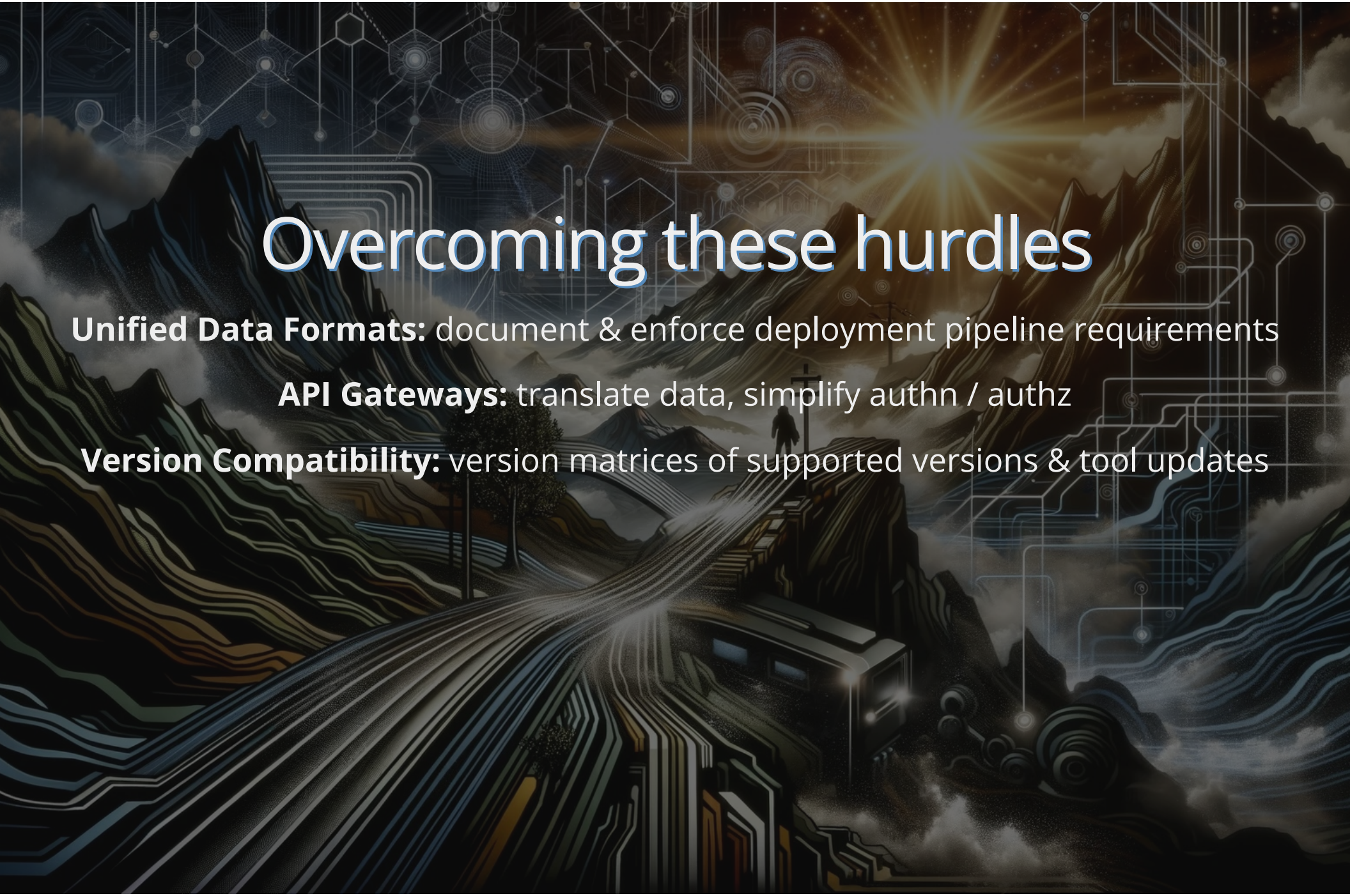
Versioning and Compatibility: breaking changes require ongoing maintenance

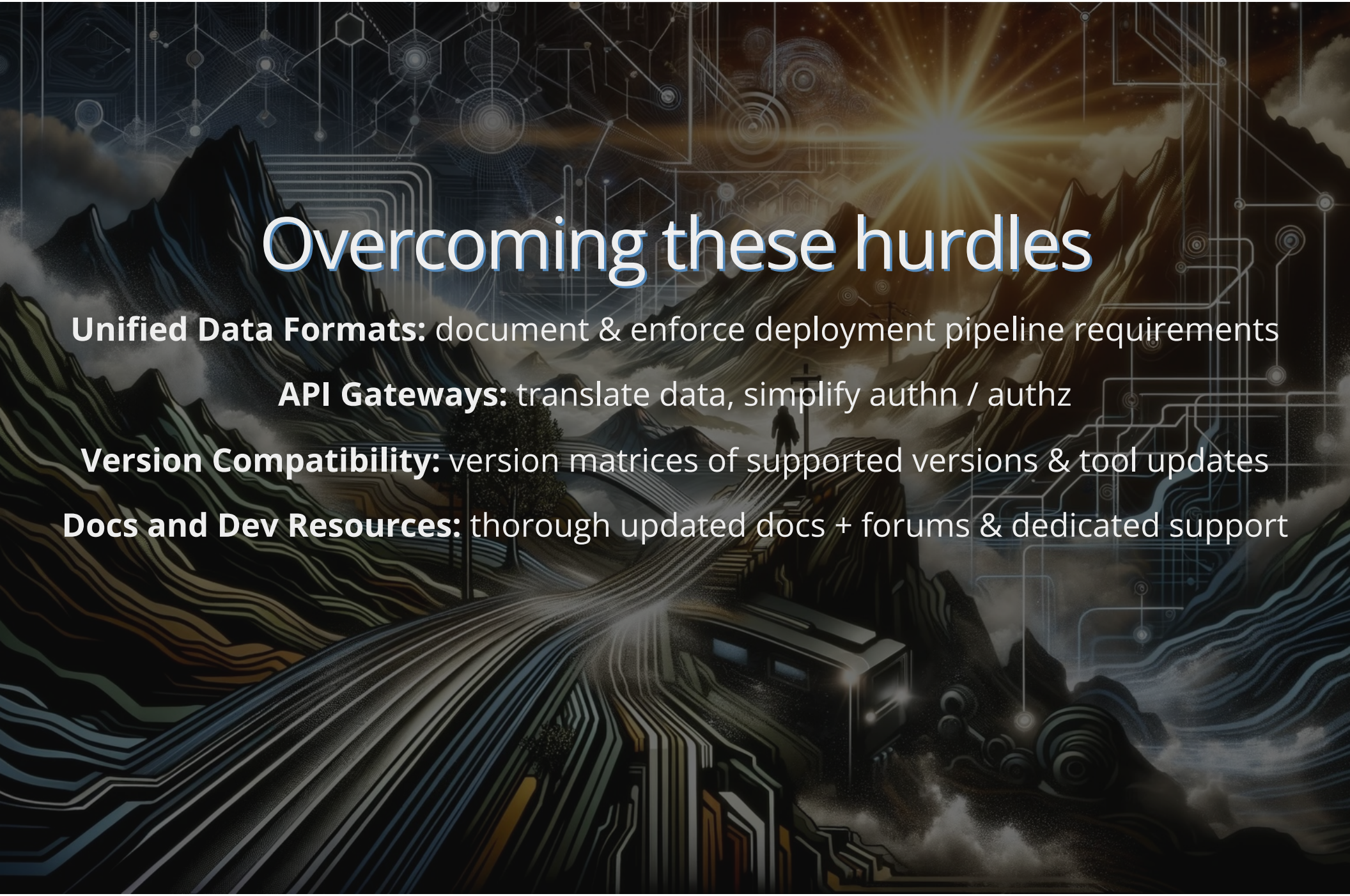
Lack of Documentation: insufficient or outdated docs = common roadblock


Overcoming these hurdles


Overcoming these hurdles
Unified Data Formats: document & enforce deployment pipeline requirements


Overcoming these hurdles
Unified Data Formats: document & enforce deployment pipeline requirements
API Gateways: translate data, simplify authn / authz


Overcoming these hurdles
Unified Data Formats: document & enforce deployment pipeline requirements
API Gateways: translate data, simplify authn / authz
Version Compatibility: version matrices of supported versions & tool updates


Overcoming these hurdles
Unified Data Formats: document & enforce deployment pipeline requirements
API Gateways: translate data, simplify authn / authz
Version Compatibility: version matrices of supported versions & tool updates
Docs and Dev Resources: thorough updated docs + forums & dedicated support


Overcoming these hurdles
Unified Data Formats: document & enforce deployment pipeline requirements
API Gateways: translate data, simplify authn / authz
Version Compatibility: version matrices of supported versions & tool updates
Docs and Dev Resources: thorough updated docs + forums & dedicated support
Continuous Testing: automate testing of integrations between all tools + pipelines


Overcoming these hurdles
Unified Data Formats: document & enforce deployment pipeline requirements
API Gateways: translate data, simplify authn / authz
Version Compatibility: version matrices of supported versions & tool updates
Docs and Dev Resources: thorough updated docs + forums & dedicated support
Continuous Testing: automate testing of integrations between all tools + pipelines
Community Collab: share successes & challenges within tooling communities


“ruthlessly eliminating barriers (and blockers) that keep your developers from being successful”

Thank You.

https://bit.ly/DevExTalk

/in/jeremymeiss

@IAmJerdog

@jerdog

@jerdog@hachyderm.io