redhat.

Accelerate DevOps with Red Hat

Jaen Swart
Senior Solution Architect
Red Hat

PRODUCT MANAGERS

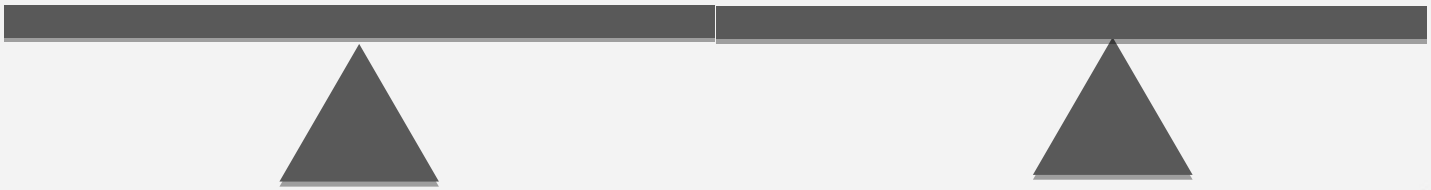
Lines of business

DEVELOPERS

Rapid development

OPERATIONS

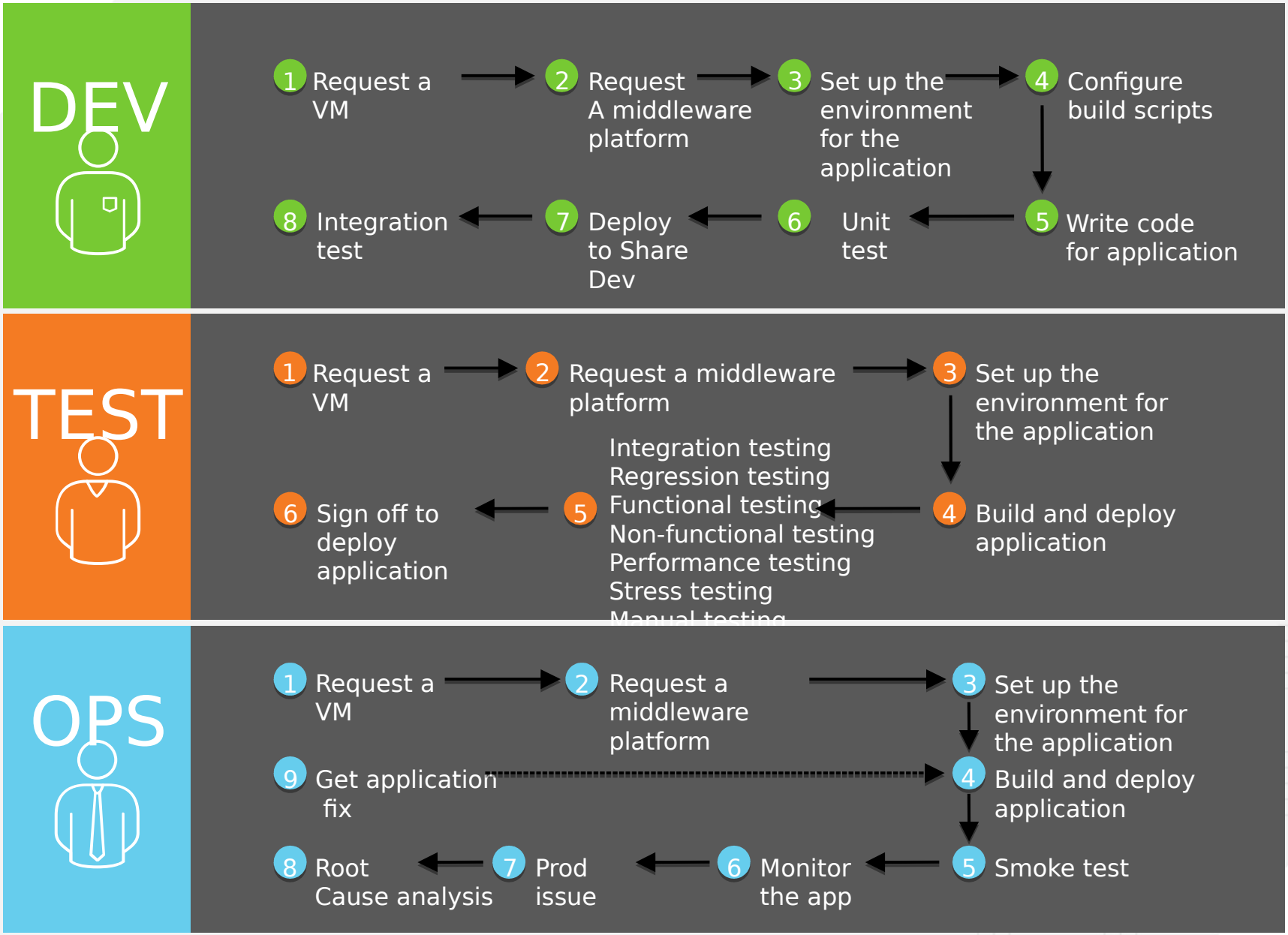
Stability

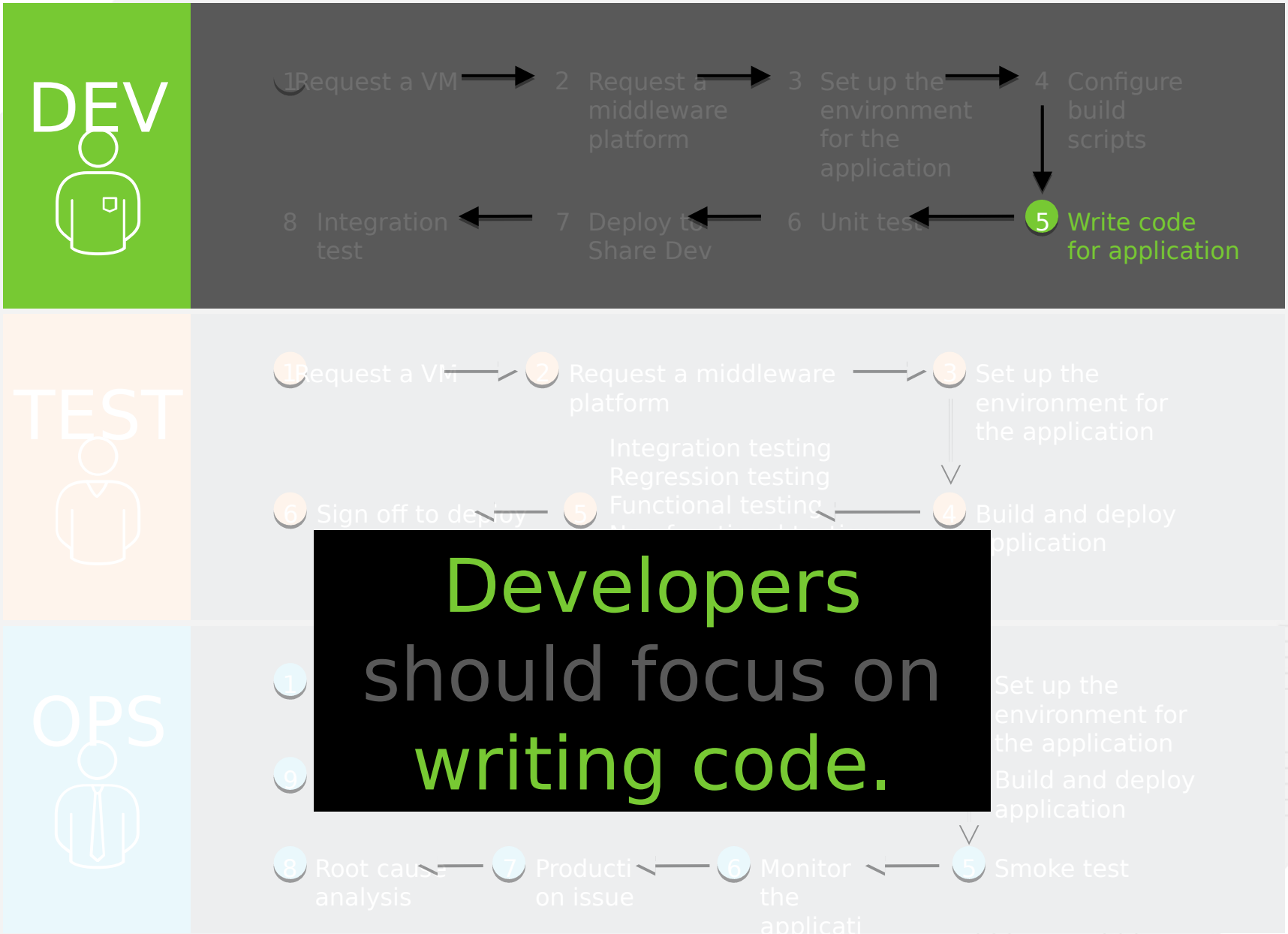


Agile

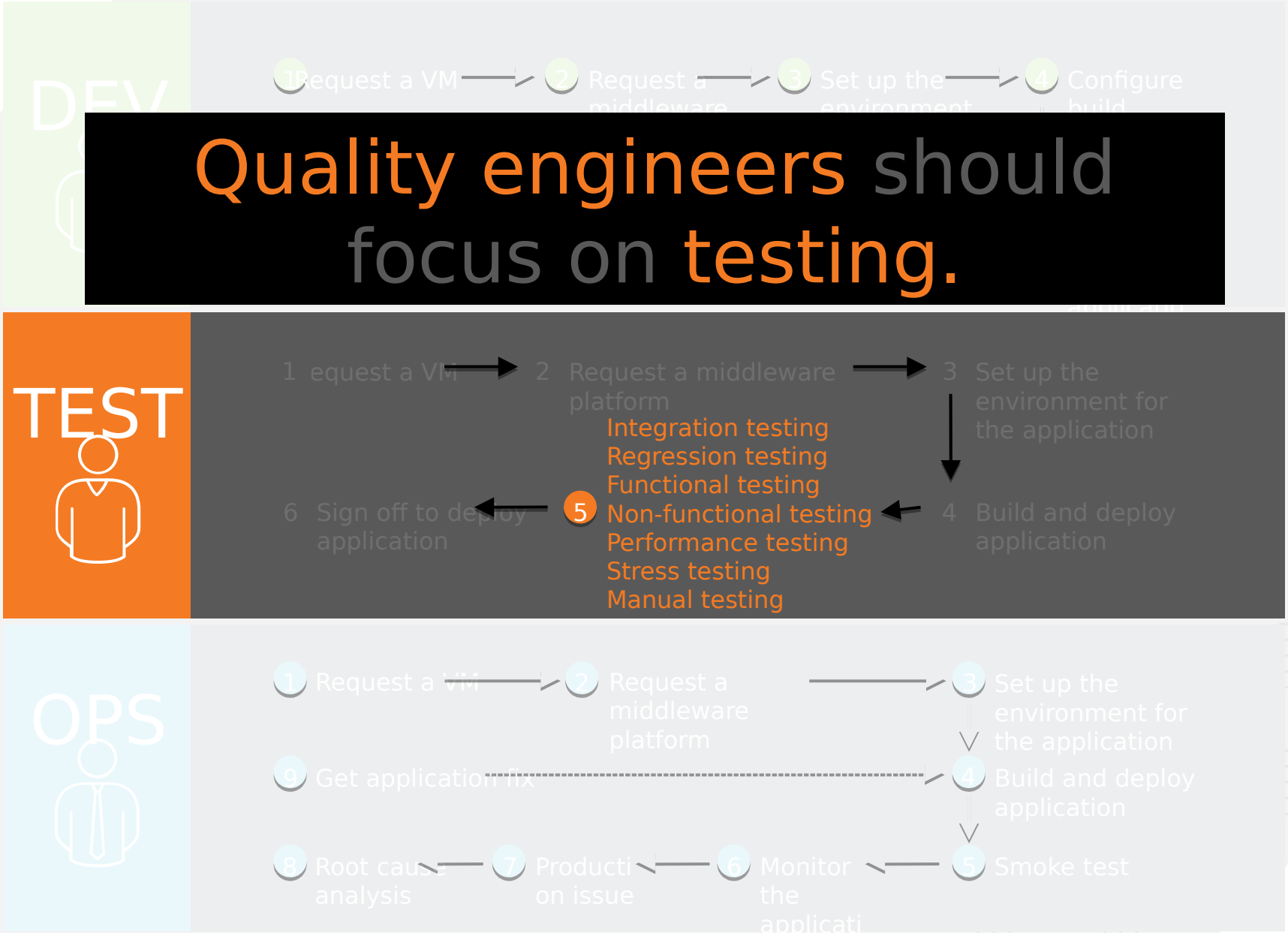
DevOps

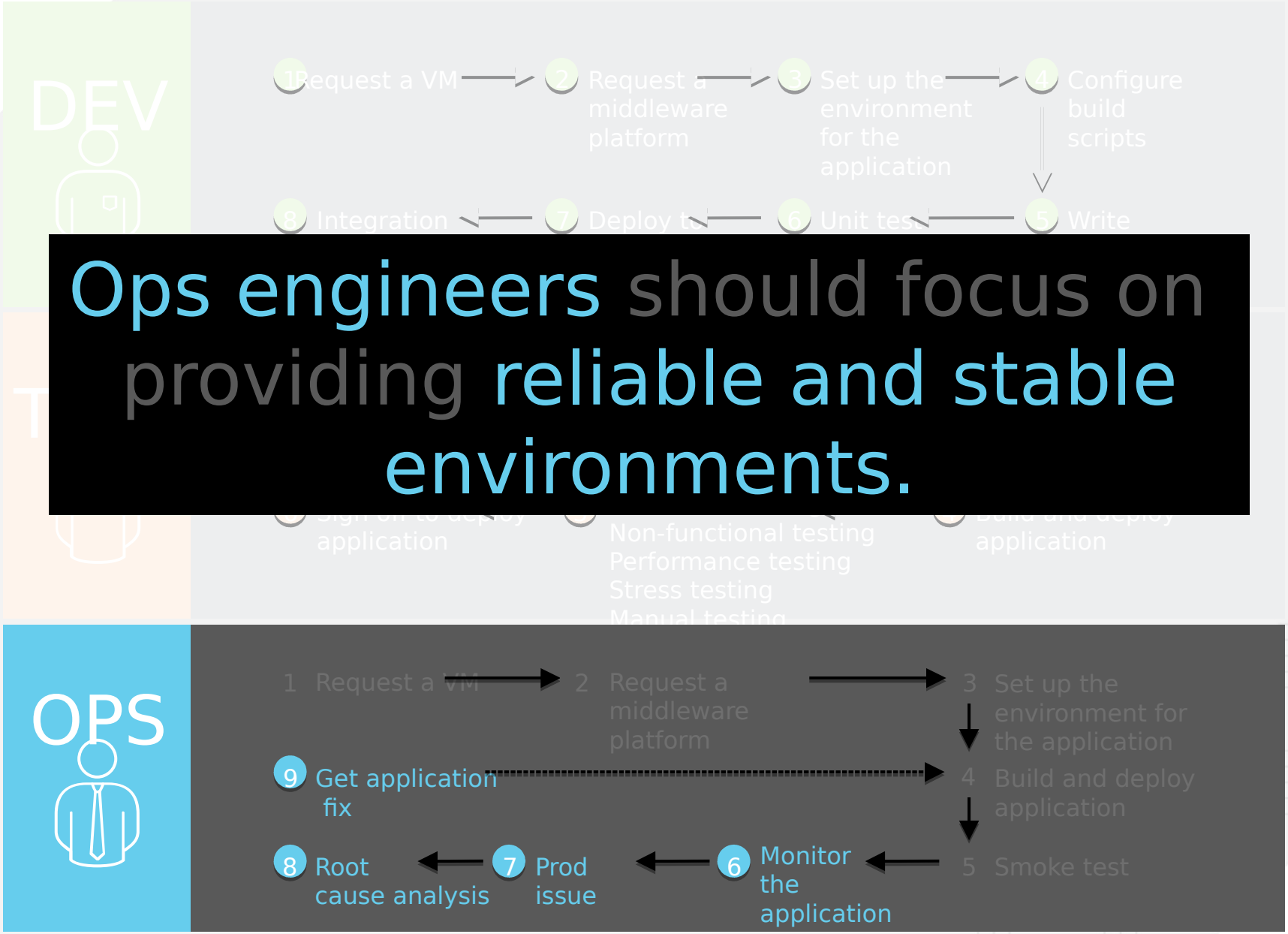
ALL TEAMS ARE THERE TO ENABLE THE BUSINESS


DEV
1 Request a VM
2 Request A middleware platform
3 Set up the environment for the application
4 Configure build scripts
5 Write code for application
6 Unit test
7 Deploy to Share Dev
8 Integration test
TEST
1 Request a VM
2 Request a middleware platform
3 Set up the environment for the application
4 Build and deploy application
5 Integration testing
Regression testing
Functional testing
Non-functional testing
Performance testing
Stress testing
Manual testing
6 Sign off to deploy application
OPS
1 Request a VM
2 Request a middleware platform
3 Set up the environment for the application
4 Build and deploy application
5 Smoke test
6 Monitor the app
7 Prod issue
8 Root Cause analysis
9 Get application fix


DEV
1 Request a VM
2 Request a middleware platform
3 Set up the environment for the application
4 Configure build scripts
5 Write code for application
6 Unit test
7 Deploy to Share Dev
8 Integration test
TEST
1 Request a VM
2 Request a middleware platform
3 Set up the environment for the application
4 Build and deploy application
5 Functional testing
Integration testing
Regression testing
6 Sign off to deploy
OPS
Set up the environment for the application
Build and deploy application
5 Smoke test
6 Monitor the application
7 Production issue
8 Root cause analysis
Developers
should focus on
writing code.

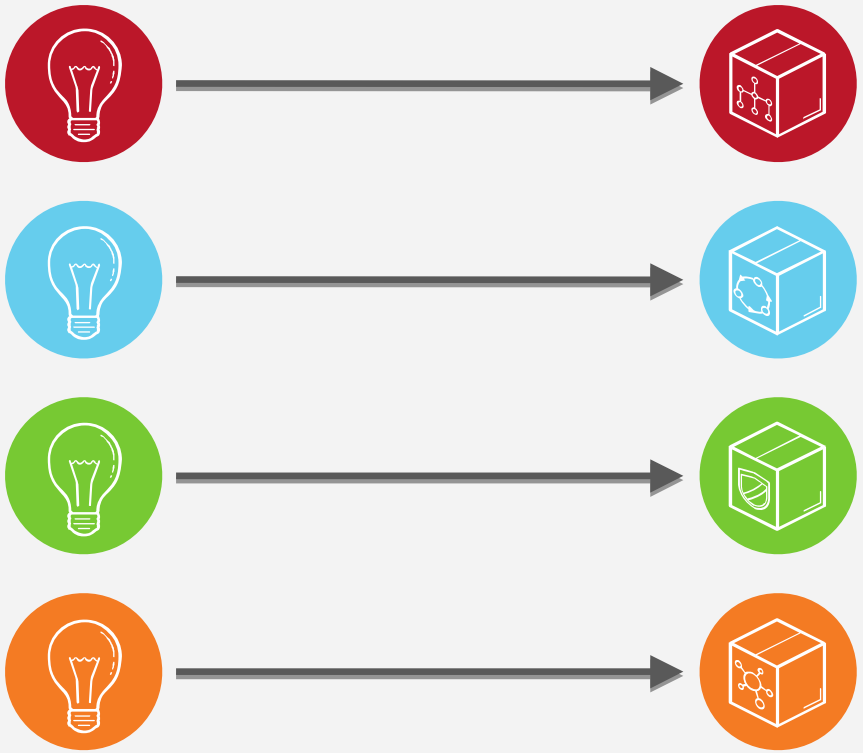
Quality engineers should focus on testing.





Ops engineers should focus on providing reliable and stable environments.

IDEA TO PRODUCT



Increased quality

Rapid delivery of product features and service

Doing more with less

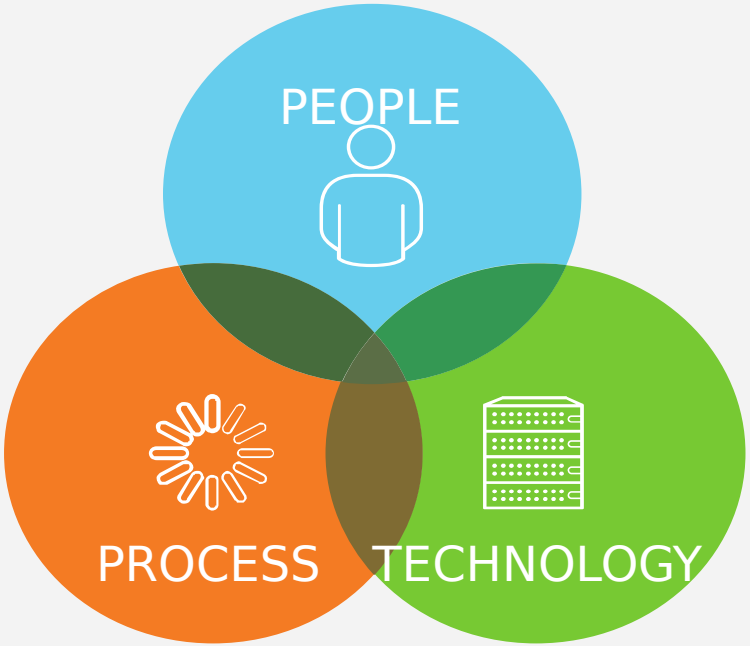
HOW DO WE ENABLE FOCUSED WORK STREAMS

TWO DIMENSIONS OF DEVOPS

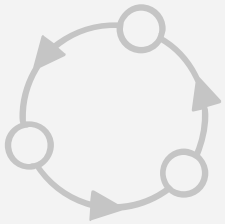
STANDARDIZATION

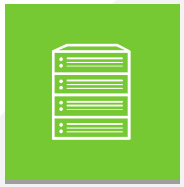
AUTOMATION

CONTINUOUS IMPROVEMENT



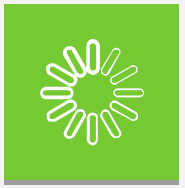
STANDARDIZATION





STANDARDIZE TECHNOLOGY

- Operating systems (with patch levels)
- Application servers
- Java/JDK/JRE
- Common libraries

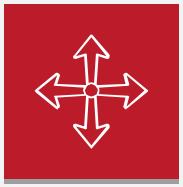


STANDARDIZE PROCESSES

- SDLC
- Release management
- Monitoring
- Escalation management

THREE LEVELS OF AUTOMATION

APPLICATION LIFE CYCLE AUTOMATION
Application



MIDDLEWARE PLATFORM AUTOMATION
Web/app servers | Libraries

INFRASTRUCTURE AUTOMATION
Virtualization | OS | Bare metal

THREE LEVELS OF AUTOMATION

APPLICATION LIFE CYCLE AUTOMATION
Application

MIDDLEWARE PLATFORM AUTOMATION
Web/app servers | Libraries

INFRASTRUCTURE AUTOMATION
Virtualization | OS | Bare metal

Provisioning resources operating system and down

- Operating systems
- Network
- Disk and storage
- CPU, RAM, and compute

Typically provided by IaaS capabilities such as OpenStack, RHEV and CloudForms

Typical use cases

- Developers, testers, and ops teams requesting VMs
- Allocating compute power to your
- applications during peak load times
- Dynamically adding storage based on consumption
- Compute governance policies and automatic
- set up and tear down of resources
- Utility-based consumption models, pay what you use
- Does not include application platforms (only VM and down)
- Standard operating environment

THREE LEVELS OF AUTOMATION

APPLICATION LIFE CYCLE AUTOMATION
Application

MIDDLEWARE PLATFORM AUTOMATION
Web/app servers | Libraries

Provisioning middleware platforms

- Load balancers
- Application servers
- Java/JDK environments
- Stand-alone frameworks

Typically provided by PaaS capabilities such as OpenShift

Typical use cases

- Developers, testers, and ops teams requesting middleware platforms
- Auto-scaling
- Compute governance policies and automatic set up and tear down of resources
- Resource optimization
- Standard operating environment

INFRASTRUCTURE AUTOMATION
Virtualization | OS | Bare metal

THREE LEVELS OF AUTOMATION

APPLICATION LIFE CYCLE AUTOMATION
Application

Application life cycle

- Software features, enhancements, versions
- Release management version control, build, release management, IDE, continuous
- Integration frameworks, common frames of references for monitoring, configuration management

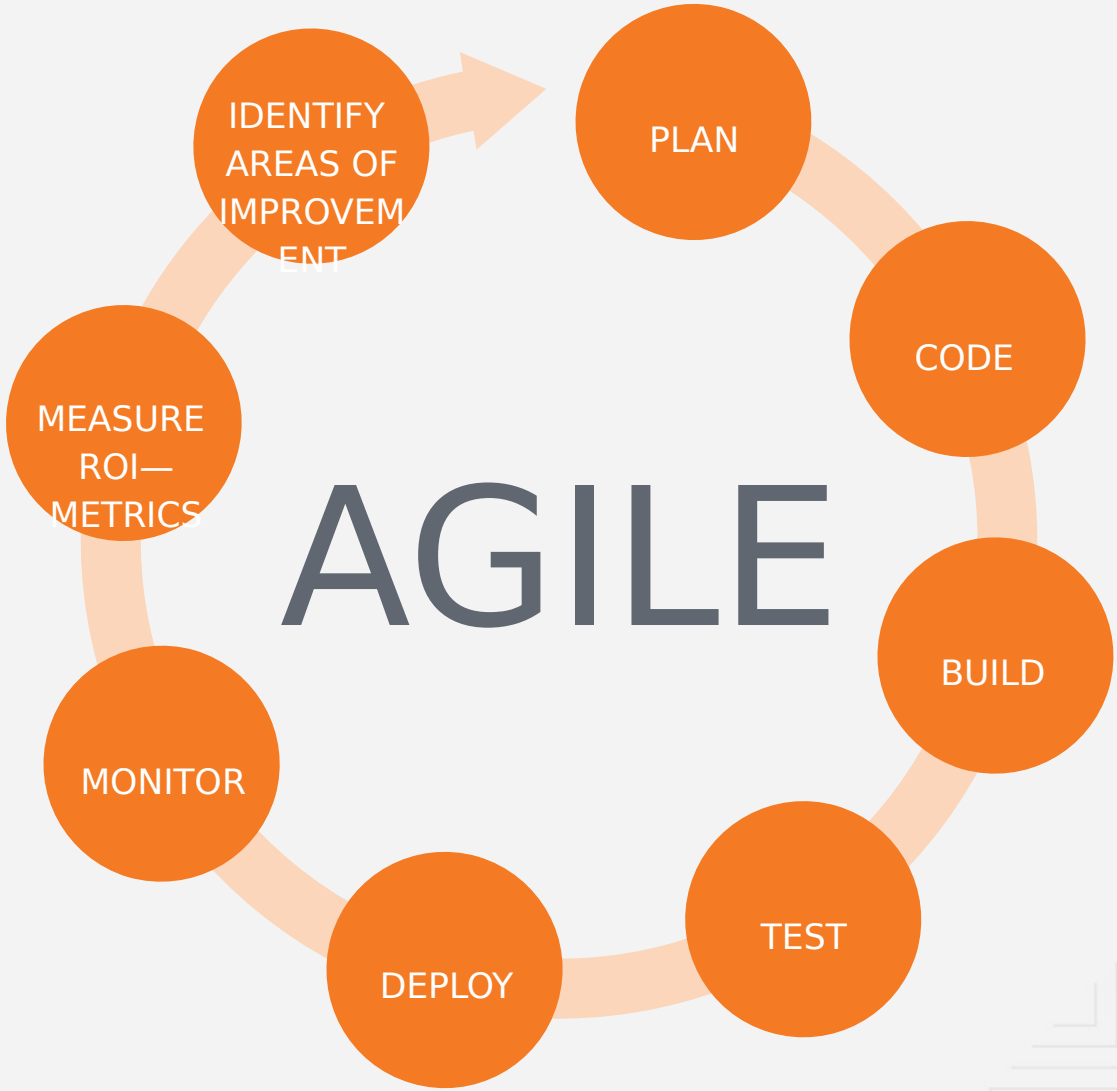
Typical use cases

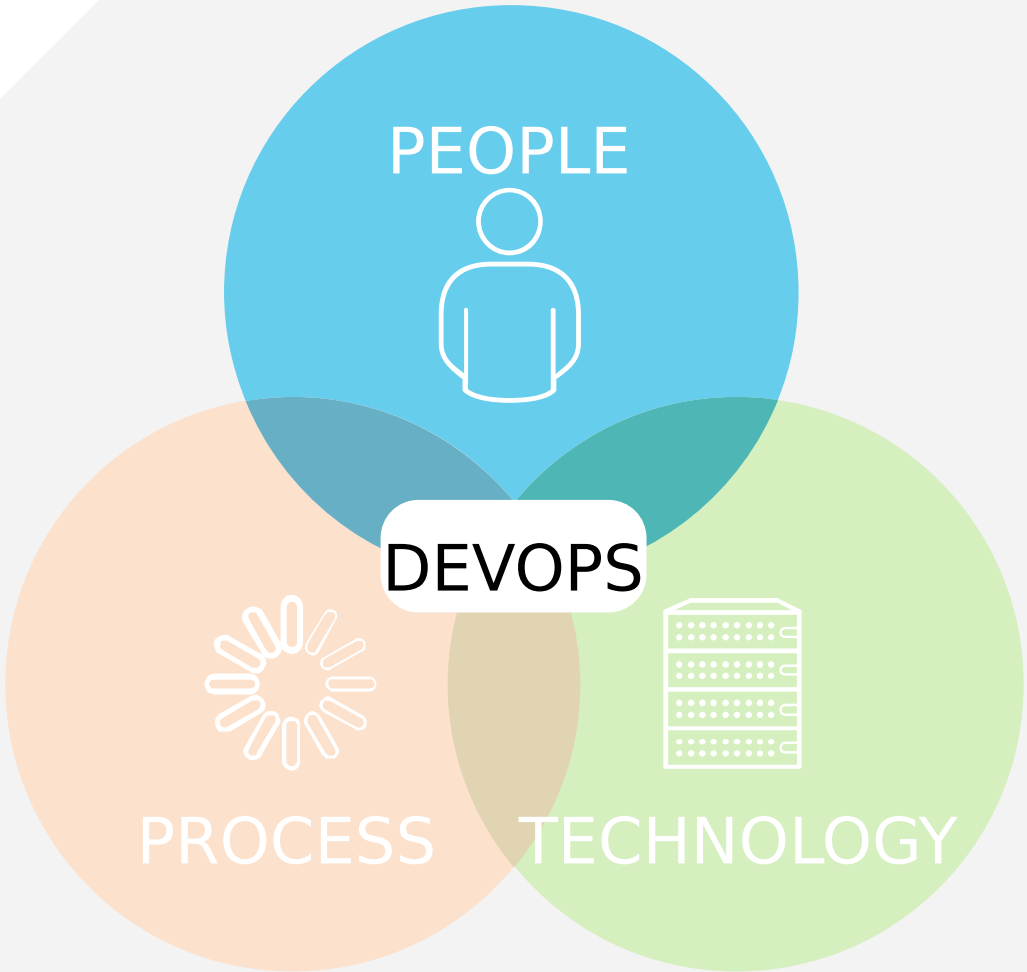
- Continuous integration
- Continuous delivery
- Automated testing

MIDDLEWARE PLATFORM AUTOMATION
Web/app servers | Libraries

INFRASTRUCTURE AUTOMATION
Virtualization | OS | Bare metal

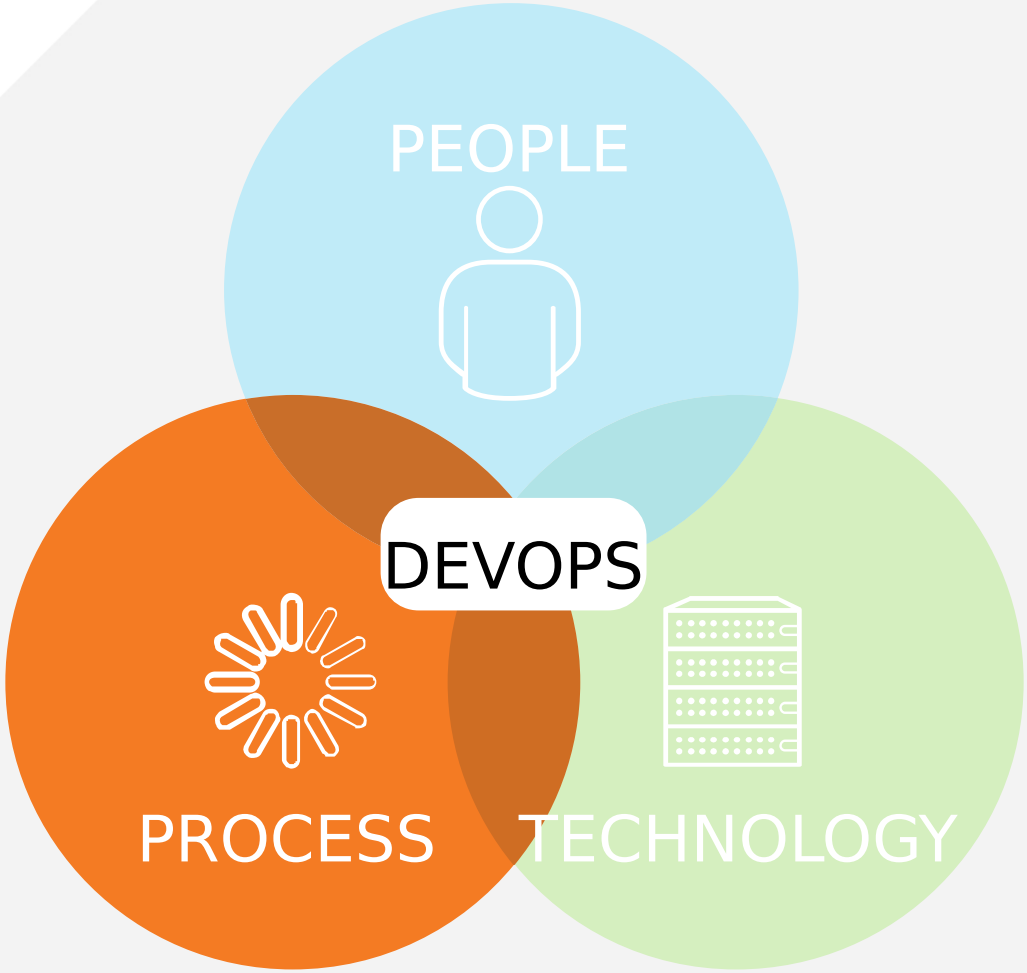
CONTINUOUS IMPROVEMENT





PEOPLE

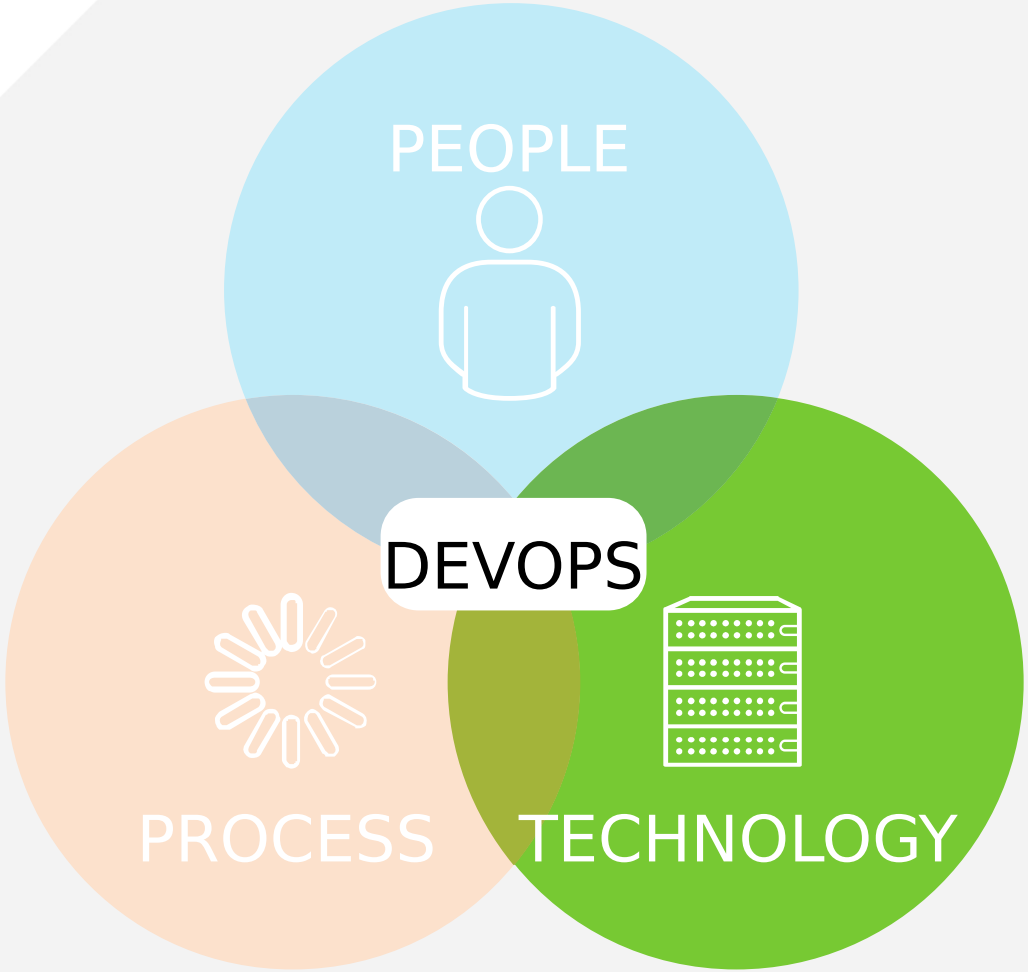
- Cultural paradigm shift
- Cross-training of skills
- Collaboration and involvement of teams across all aspects from designing through monitoring of application
- The question everyone should ask is “Is my application driving business value based on the state it is in now?”
- Short-lived and interim DevOps Enablement Team can be created in organizations



PROCESS

- Agile methodologies
- Governance and continuous feedback loops to reduce and eliminate technical debt
- Define metrics for measure
- Project is not done until the application is driving value for

  the customers and business
- Automate everything
- If something breaks, don’t hack. Fix the automation script and start over.
- Common frames of reference (for dev, qa and ops) for application monitoring in production
- Open access
- Developers on call

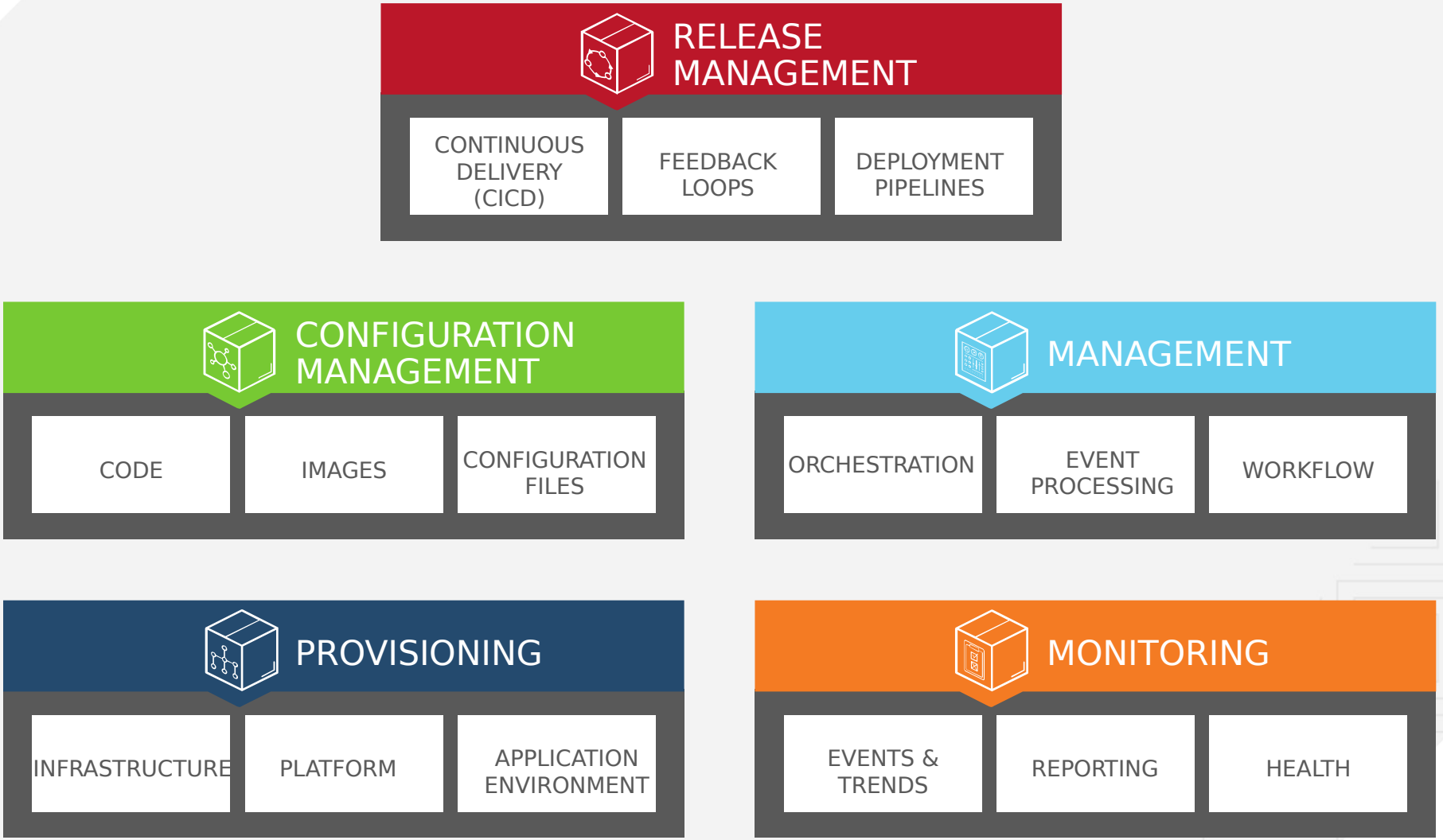


TECHNOLOGY

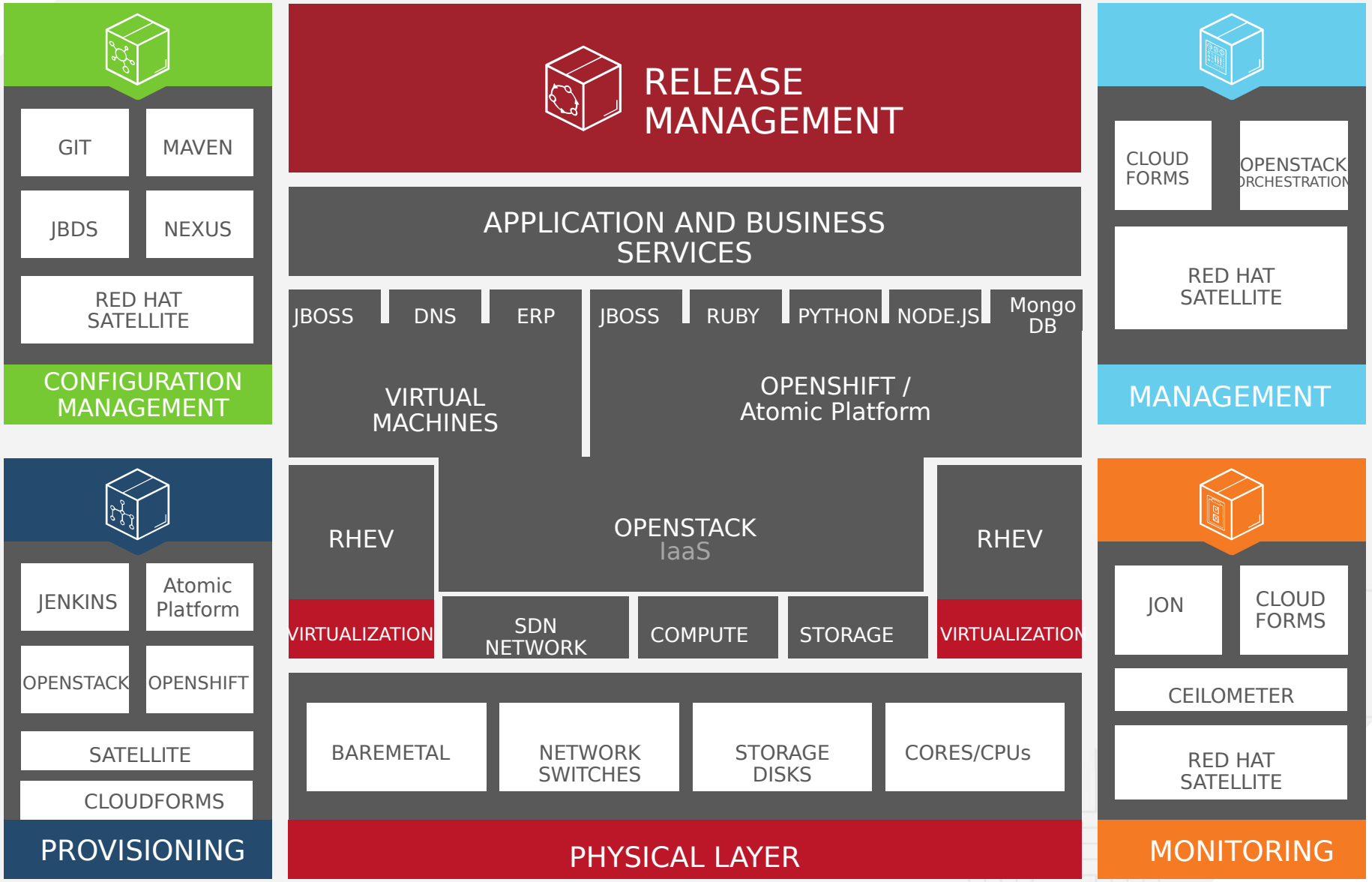
- Automation is key. OpenStack and OpenShift provide lots of required automation capabilities out-of-the-box.
- Standardize software versions, patch levels, and provisioning mechanisms
- Faster application environment provisioning, root cause analysis
- Notifications and pro-active monitoring

DEVOPS FUNCTIONAL MODEL

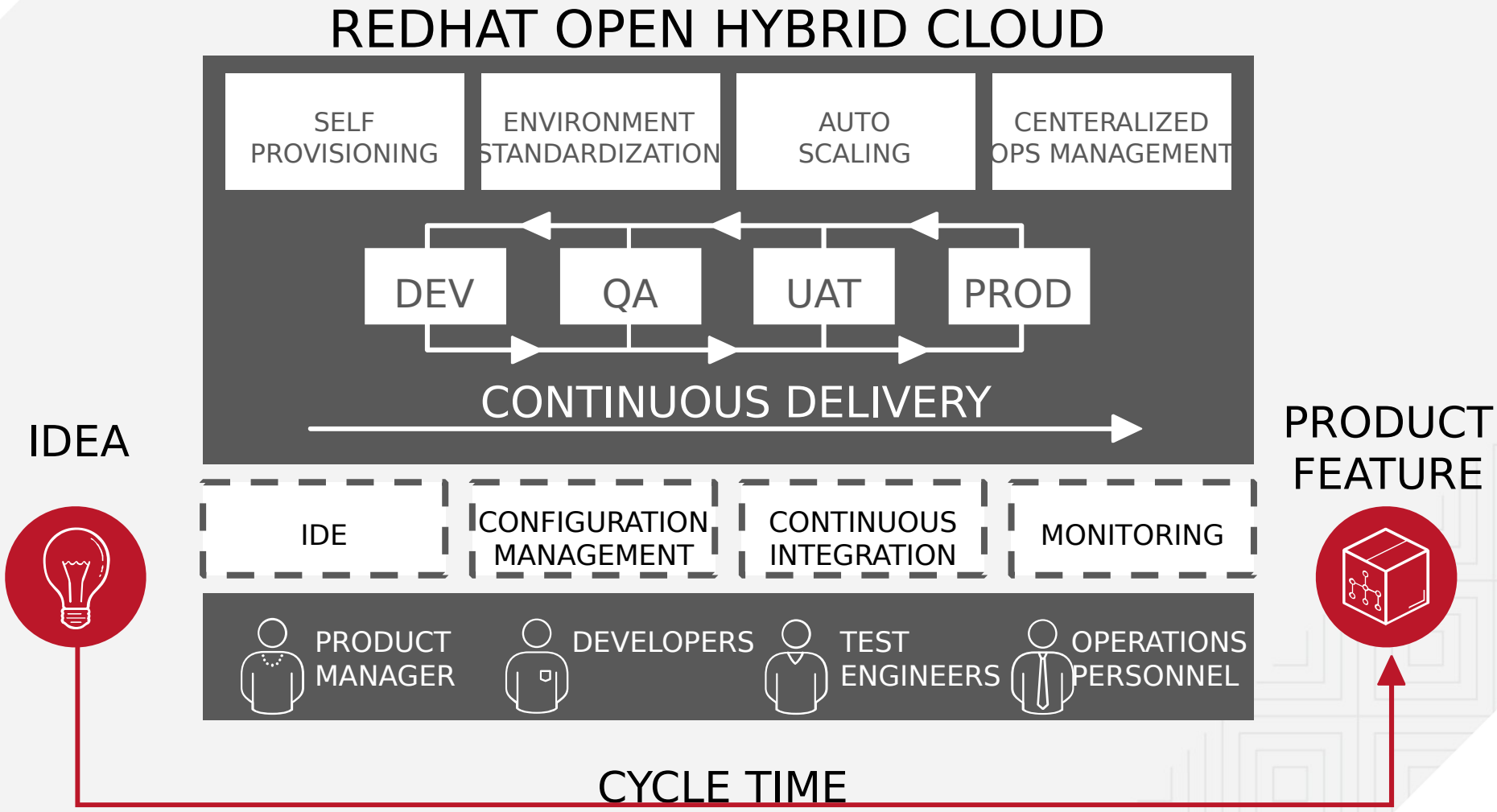
DEVOPS FUNCTIONAL MODEL



ARCHITECTURE



CONTINUOUS DELIVERY



TYPICAL USE CASES / SCENARIOS

ORGANIZATIONAL MATURITY

- Templatized architecture tiers and complex environments

- Advanced automation capabilities
- Multiple deployments a day
- Capabilities as a Service

- Templatize your individual infrastructure tiers
- Auto-Scaling, Pro-active monitoring and automatic management

- Continuous Deployment and push-button application deployments
- One deployment a day

Self-provisioning of

- Virtual Machines
- Platforms
- Application Environments

- Continuous Integration
- Infrastructure as code
- Basic automation & auto-scaling

- Migration of VMs to OpenStack (IaaS)
- Development of greenfield applications in OpenShift (PaaS)
- Basic systems provisioning through standardization

HOW RED HAT ACCELERATES DEVOPS

Automation is a cornerstone of DevOps practices.

APPLICATION LIFE CYCLE AUTOMATION

Application

Red Hat's Open Hybrid Cloud integrates well with various tools and frameworks to drive ALM and be an accelerator to implementing DevOps.

MIDDLEWARE PLATFORM AUTOMATION

Web/app servers | Libraries

OpenShift and JBoss Middleware provide advanced platform automation. Does not replace DevOps collaboration, but provides a framework for it, so you don't have to roll your own.

INFRASTRUCTURE AUTOMATION

Virtualization | OS | Bare metal

OpenStack, RHEV and CloudForms. OpenStack provides advanced infrastructure automation, provides standard way to define infrastructure as code so environments can be provisioned in a predictable and repeatable manner

CLOUD-ENABLED DEVOPS TRANSFORMATION SUCCESS STORY SUMMARY

INSURANCE

Location: EUROPE

INSURANCE SERVICES

SOFTWARE AND SERVICES

eNovance Red Hat Consulting
OpenStack
OpenShift
RHEL
Jenkins
GitHub
Ansible
Puppet

CHALLENGE

- Externalized IT
- Intense competition to provide quality IT services
- Shadow IT (AWS) and non-standard infrastructures
- Hug data compliance risks

SOLUTION

- Built an internal cloud based on OpenStack for IaaS and OpenShift to provide PaaS capabilities
- Self-service portals for Admins, Developers and Operations personnel
- Continuous Feedback loops were enabled using proven methodologies such as SCRUM
- Established common templates for application and middleware creation
- Organizational transformation through demonstrable successes within projects vs. top down mandates

BENEFITS

- Reduced time to market from weeks to days
- Enhanced reporting for quick decision making
- Aligned IT with business

RED HAT CONSULTING SERVICE OFFERINGS AND METHODOLOGY

Service Offering	Description	LOE	Deliverables
Assessment Workshop	Requirements and use case gathering session	2 days	Journal describing business and technical priorities and our proposed solution
Architecture Review & Mentoring Service	Cloud Discovery Workshop plus basic software installation use case	2 weeks	Current state assessment, mentoring assessment, software installation and basic mentoring
Pilot Implementation	Architecture Review & mentoring Service plus design and implementation of a pilot use case for an application. LOE estimates are given in increments of 2 weeks per our SCRUM process	~ 6 weeks. Requires additional scoping	All of the above plus working pilot implementation
Solution Implementation and Optimization	Phased implementation of solution, production readiness and optimization of key processes, architectures and implementations. LOE estimates are given in increments of 2 weeks per our SCRUM process	~ 3 months and over. Requires additional scoping	Production readiness, design and code artifacts and optimized working solution

Skills: Iaas, Paas, Middleware, Enterprise Architecture, SDLC, Agile, DevOps Process Architect

SUMMARY & TAKEAWAYS

Standardize, Automate and Continuous Improvement
People, Process and Technology

Without Automation, you don’t have DevOps. Build an efficient Deployment Pipeline

Think about DevOps Tool Chains, Infrastructure Automation Engineers, DevOps Tools Team

Infrastructure as Code, Environment as Code

Embrace Agile. Release early , release often

Red Hat is your partner. We can help accelerate your DevOps

QUESTIONS?


redhat