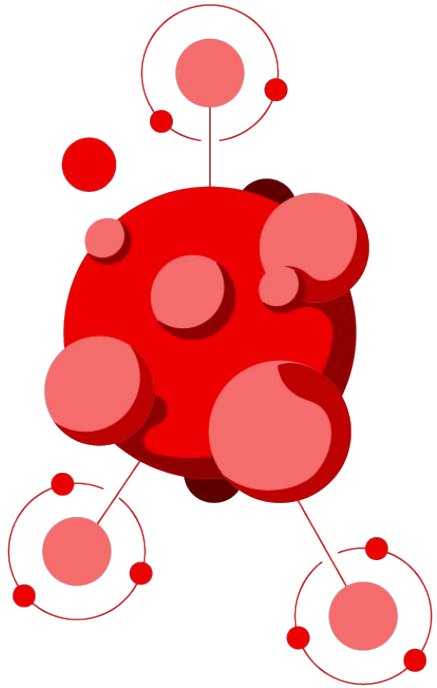
Red Hat | intel

Open Tour

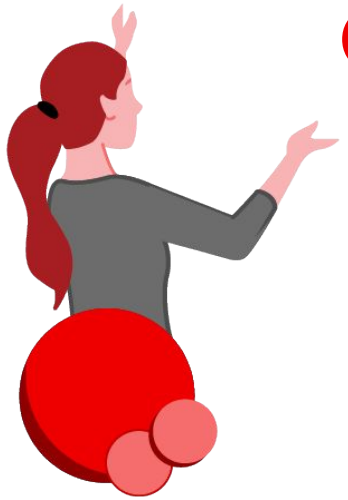
Connecting people and solutions
to accelerate your business

OKRs in Software Development and Project Pelorus



What is it - and why would I care?

What we'll discuss today



- What are those Objectives and Key Results (OKR)?
  - define **what** you want to accomplish
  - and **how** to measure success
- A little organisational context of where we developers live (backed by Research)
- What we always knew - nobody understands us! (5 Pillars of any IT Organisation and communication breakdowns)
- How we can bridge those gaps with meaningful metrics
- How we capture and present those metrics
- How these metrics help us improve

Speak after me...

Software Development work is...

- ...fun
- ...yet difficult to explain to management
- ...often hidden “deep in the trenches” of daily challenges
- ...has a huge impact on the overall organisation

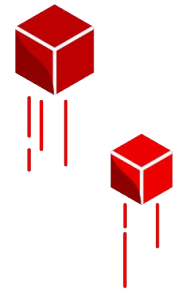
Wait... what?

- ...has a huge impact on on the overall organisation

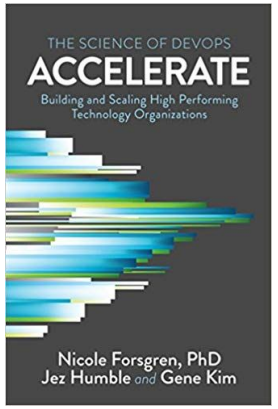
YES, IT HAS!

- ...even beyond the obvious “we deliver apps to our users” impact.

Let’s see some research to back this statement

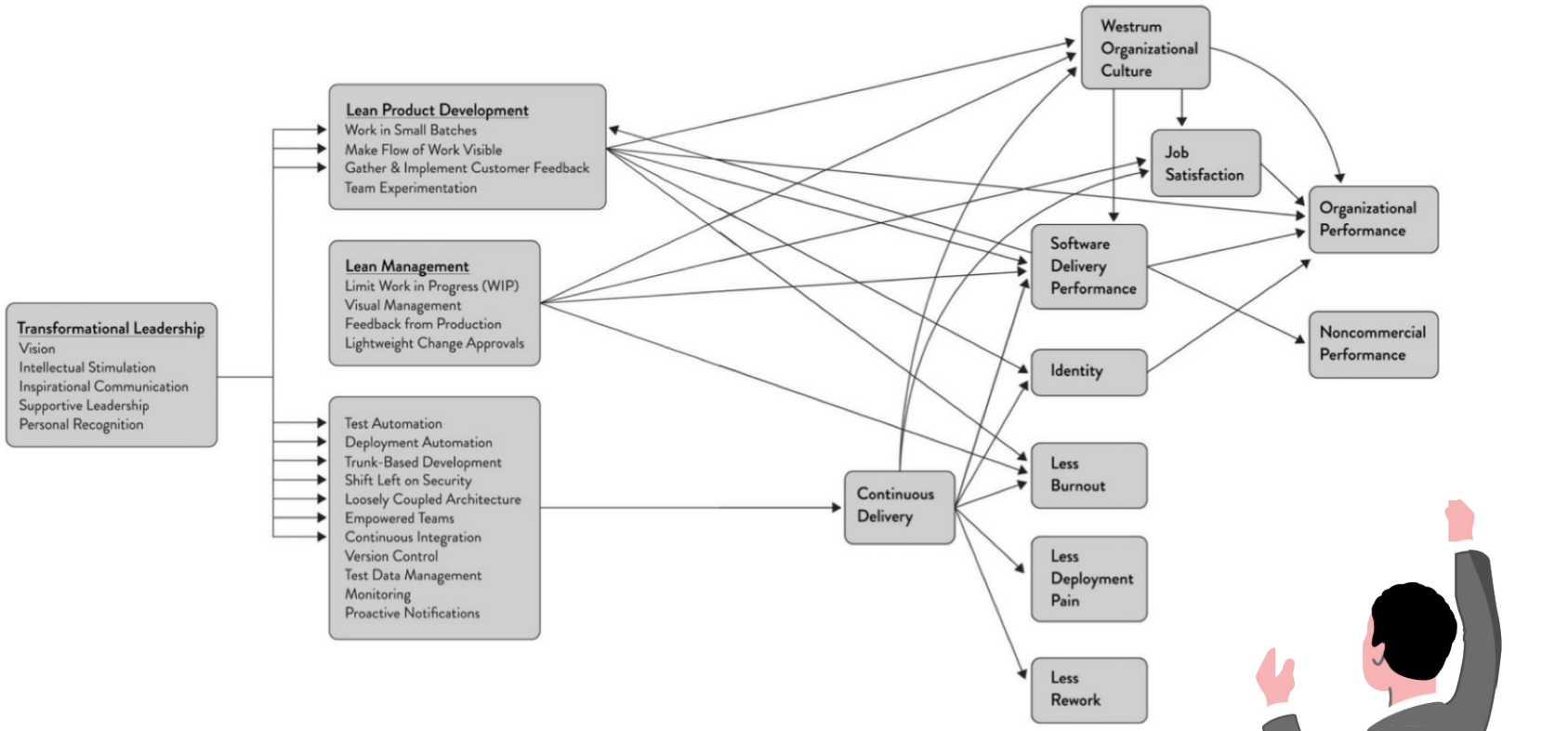


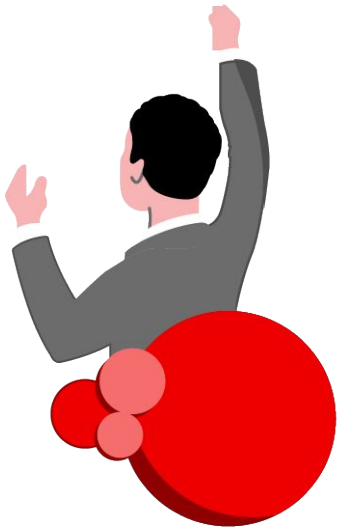
Industry Research



- Published in **Accelerate** and other books
  - Authors: Gene **Kim**, Jez **Humble**, and Nicole **Forsgren, PhD**
  - Published on March 27, 2018
- Software Delivery metrics as a proxy
- Significant impact on organizational outcomes, like profitability, market share, quality, customer satisfaction, and achieving organizational and mission goals.
- Four critical measures defined to improve

Overall Research Program



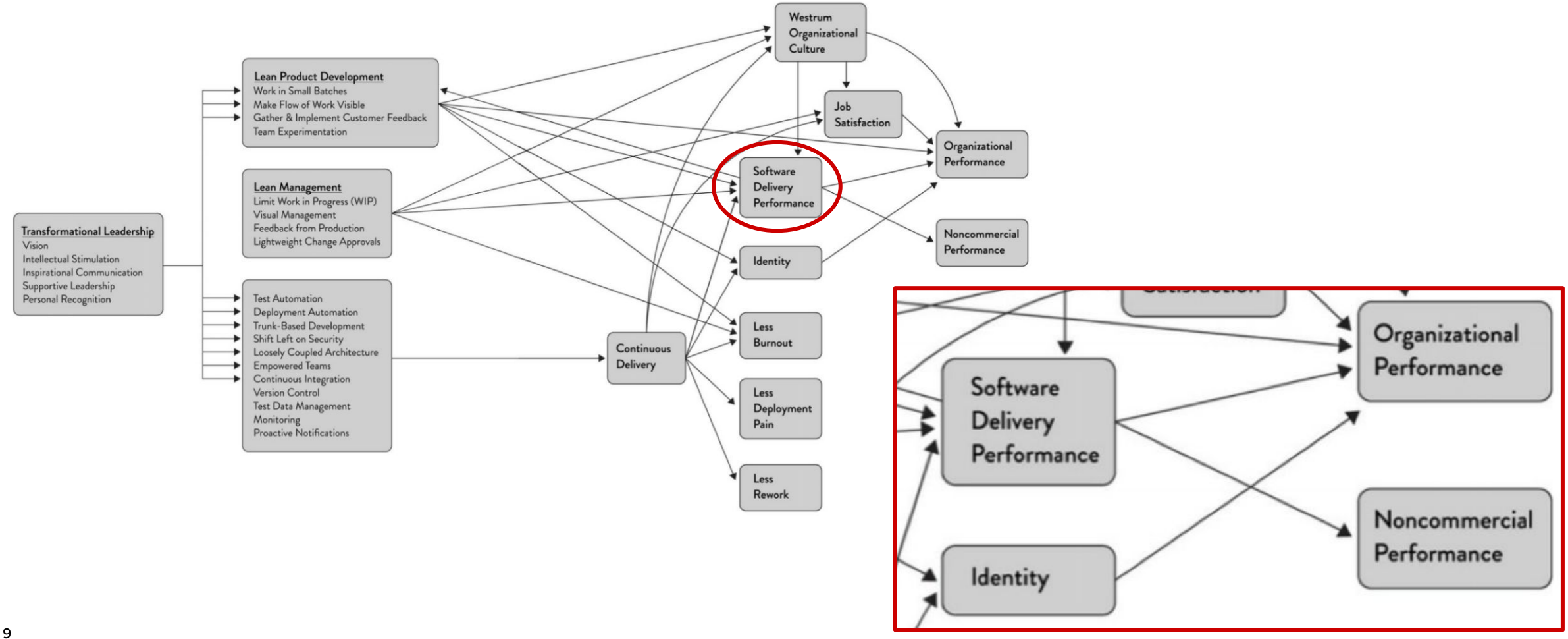


Don't worry!!!

We won't go into too many details here!

Source: https://itrevolution.com/book/accelerate/

We are talking about this...



Source:
https://itrevolution.com/book/accelerate/

A bit more context

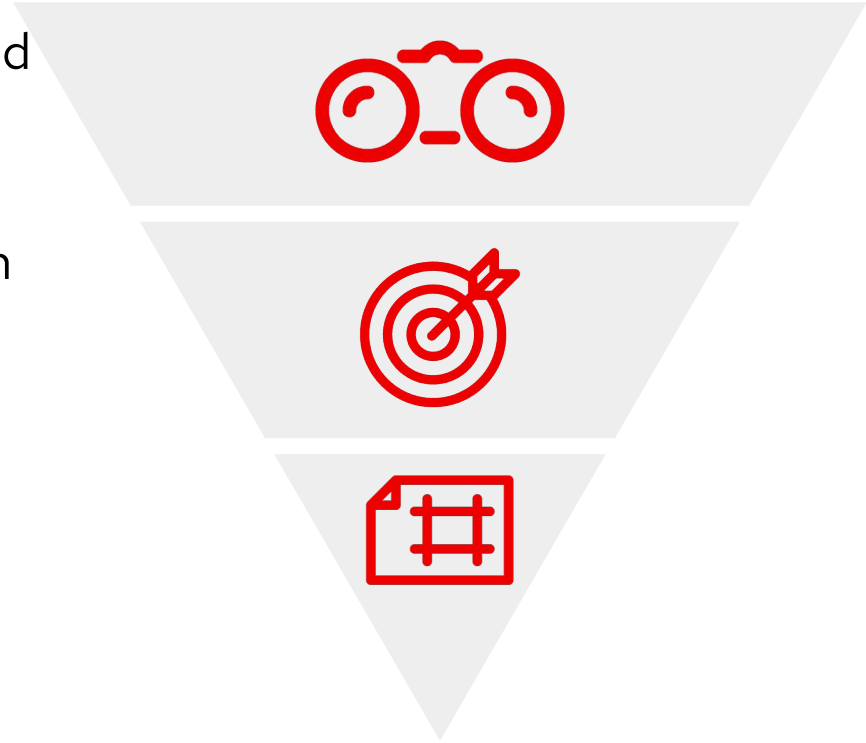
- The current (or should I say constant, recurring) challenge
- The 5 pillars of any IT Organisation…
- …and why there are communication gaps
- How to bridge these gaps
- Why should you stay until the end of the presentation (yes, there's technology coming up!)

The current challenge

Leadership wants... to paint a picture of a ***long-term*** vision, and support that vision with smart investments

Delivery wants... to have a concrete plan they can execute on in the ***short term***

How do we bridge that gap?
Find a common goal to align on in the ***mid-term***, and measure progress towards it continually



Observe the Five Elements within the IT organization

Leadership **Product** **Development** **Architecture** **Operations**

Patterns of failure

What do these failures look like?

Leadership

Prevents change

Product

Builds things that don't matter

Development

Builds wrong things

Architecture

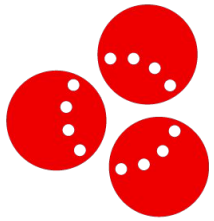
Builds things wrong

Operations

Incidents and outages

Why are there communication gaps?

Shared ***measurable*** outcomes can help bridge these communication gaps.



Functions care about different types of work

This can lead to organizational silos over time. People tend to organically group with those they share function with.

People talk in different scopes of time

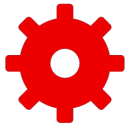
Functions prioritize short-term and long-term impacts differently.

Alignment seems to call for major trade-offs

Tactical vs. Strategy and Change vs. Stability

How might we create a shared goal without major trade-offs?

Opportunities to get things right

Leadership	prevents change			**Leadership** provides executive sponsorship
Product	builds things that don't matter			**Product** creates strategic requirements
Development	builds wrong things	**TO**		**Development** delivers the features
Architecture	builds things wrong			**Architecture** makes the right thing easy
Operations	has incidents and outages			**Operations** keeps systems running

Introducing Bridge Outcomes

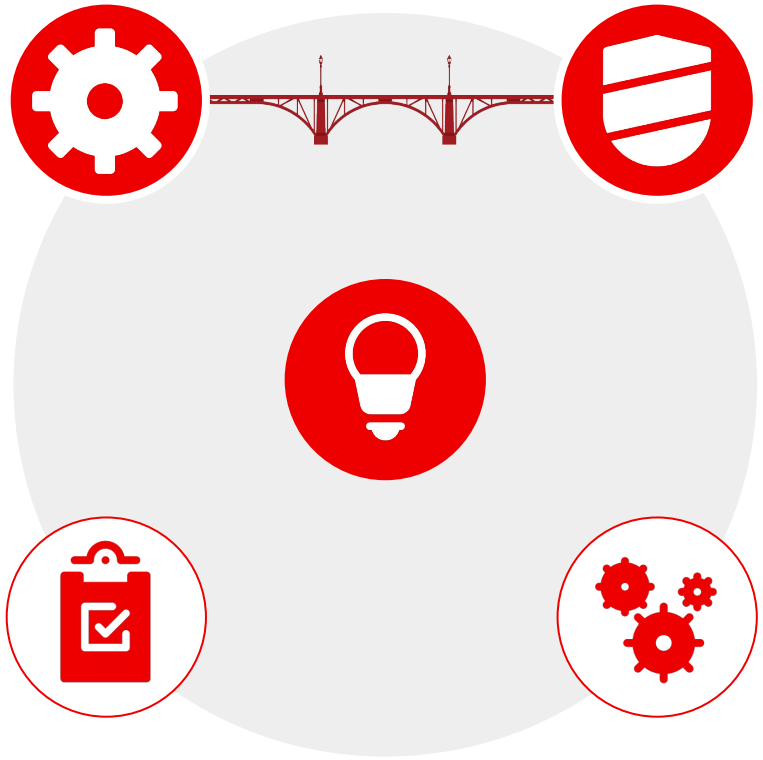
Bridge communication gaps with shared understanding of **outcomes**

A **bridge outcome** is an outcome that:

- Impacts multiple IT functions
- Optimizes flow of value
- Uses SMART metrics
  - (Specific, Measurable, Achievable, Relevant, Time-Bound)
- Measurable in a consistent, repeatable manner

Establish a Shared Measurable Outcome

Between Development and Operations



OUTCOME

Software delivery performance

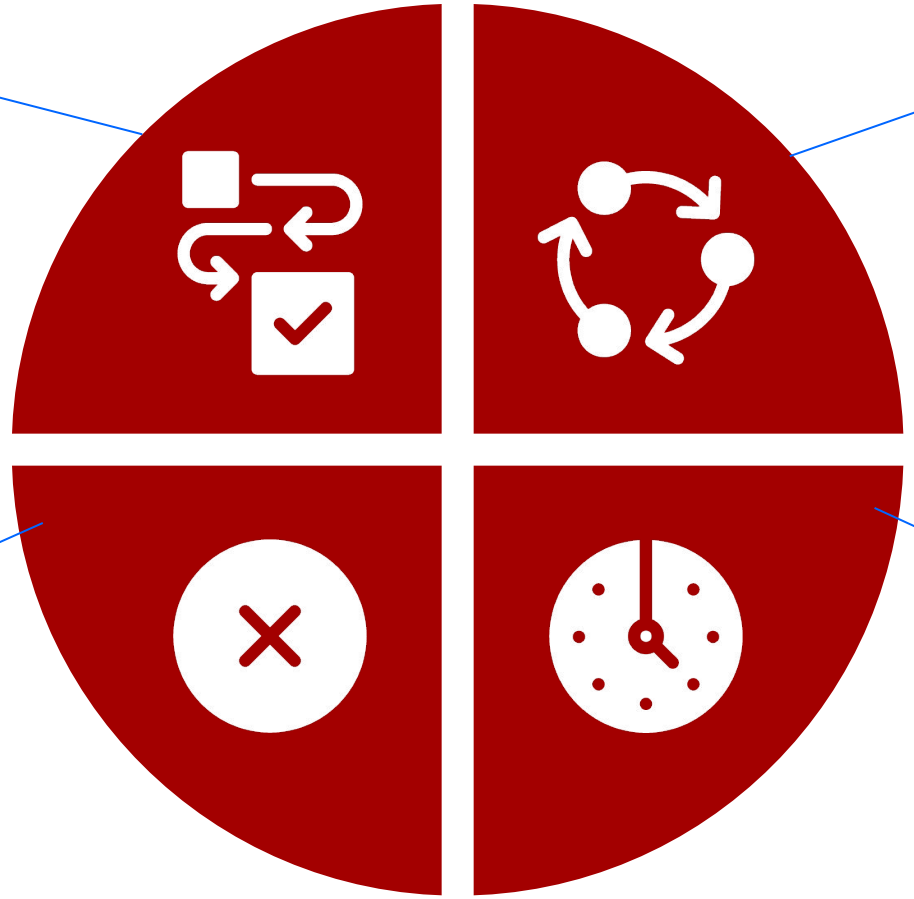
JUSTIFICATION

Measures the ability of an organization to deliver value to its customers

Software Delivery Performance

Lead time to change

Time from code committed to deployed to production



Deployment frequency

How often does an app deploy to production

Change failure rate

Percentage of deployments requiring rollback and/or fixes

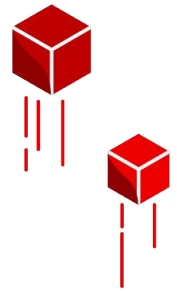
Mean time to restore

How long it takes systems to restore after a failure in production

As defined by Gene Kim, Jez Humble and Nicole Forsgren in ***Accelerate***

There are many more

- Development and Operations: **Availability**
- Operations and Architecture: **Supported Technology Adoption**
- Product and Development: **Value Flow**
- And even more, all with their associated metrics (it needs to be ***measurable***!!!)



...but let's focus on Software Delivery Performance

Why measure?

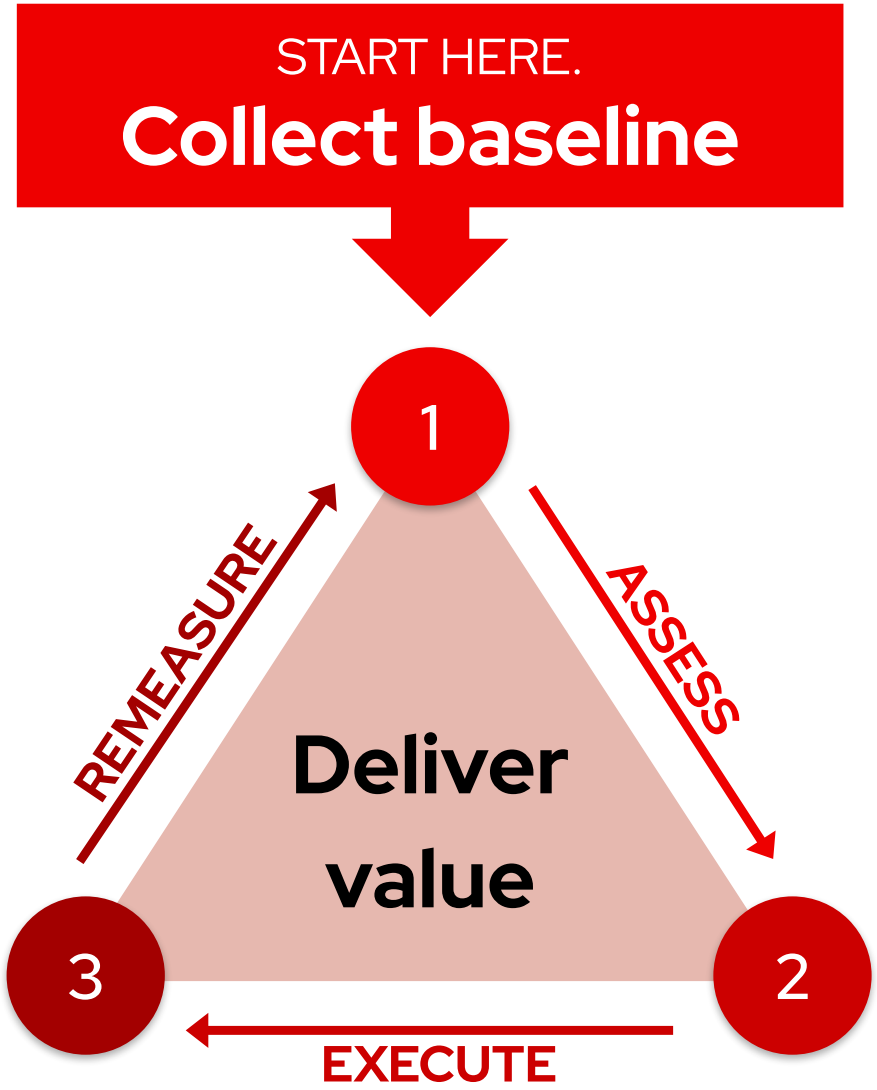
To improve – Using trends to drive decisions

COLLECT. Understand the baseline first.

ASSESS. Analyze current state and identify where you want to improve.

EXECUTE. Define an experiment by focusing on a deliverable with a specific target improvement on the metrics.

REMEASURE. Recapture metrics to validate hypotheses.



Red Hat

Introducing...

pelorus

https://www.konveyor.io/tools/pelorus/

A dashboard for organizational alignment & transparency around trends towards shared outcomes.

Measure bridged outcomes

Current dashboard is designed to capture Software Delivery Performance. Use Pelorus to understand business value delivery for your products. Accumulate products to assess the organization-wide impact.

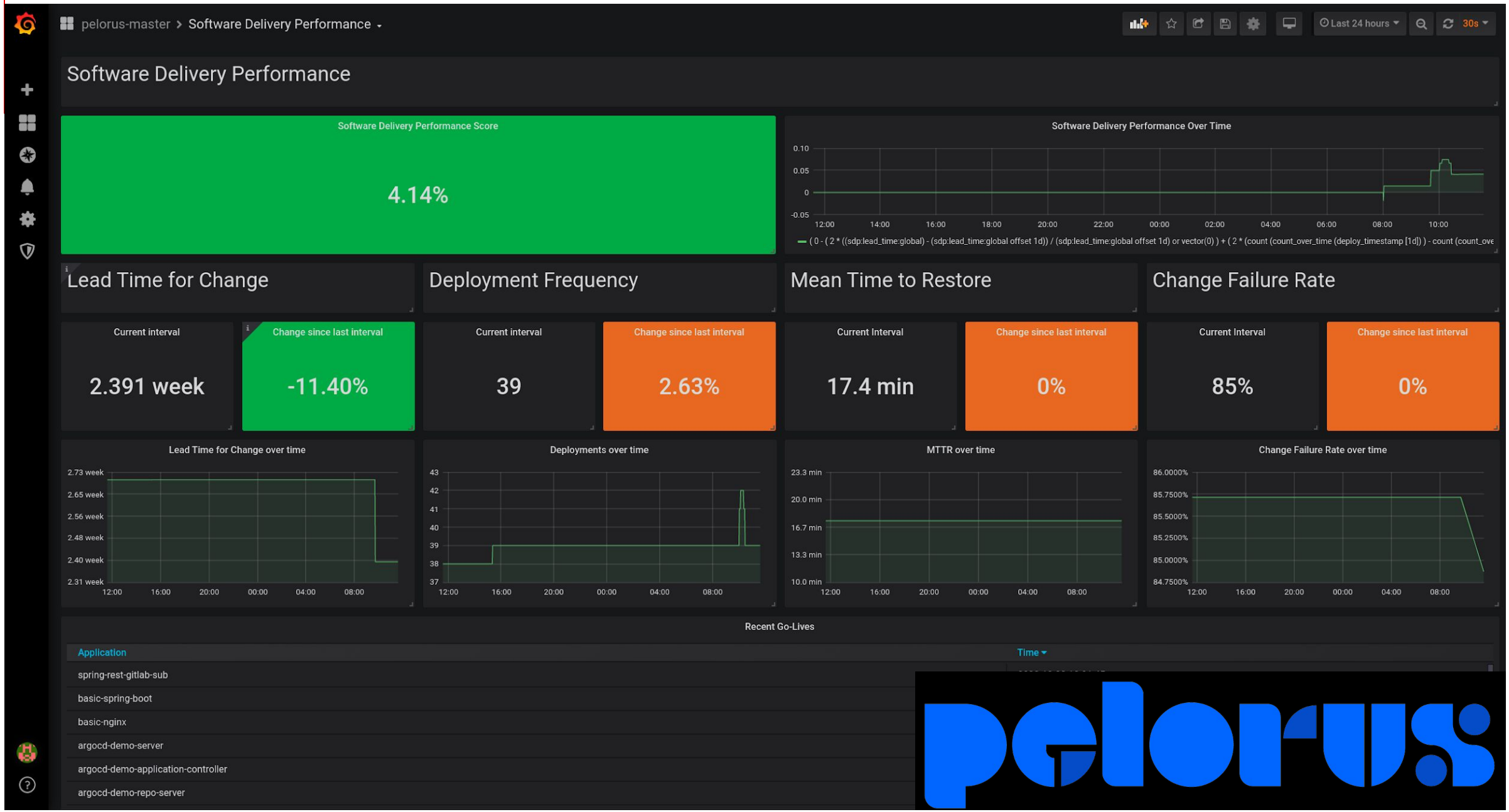
Or create your own, based on your target bridged outcome.

Customize to fit the environment

Set up exporters to use existing data sources as metric inputs to calculate measures.

Use trends to inform IT decisions

Talk about the metrics trends and set shared goals around improvements, teams can achieve those shared goals in specific ways that enables their work



EXPORTERS

Deploy time exporter

Commit time exporters

Failure exporter

PROVIDERS

Red Hat OpenShift

GitHub

ATLASSIAN Bitbucket

Visual Studio Team Foundation Server

GitLab

Coming soon

Jira Software

servicenow

METRICS

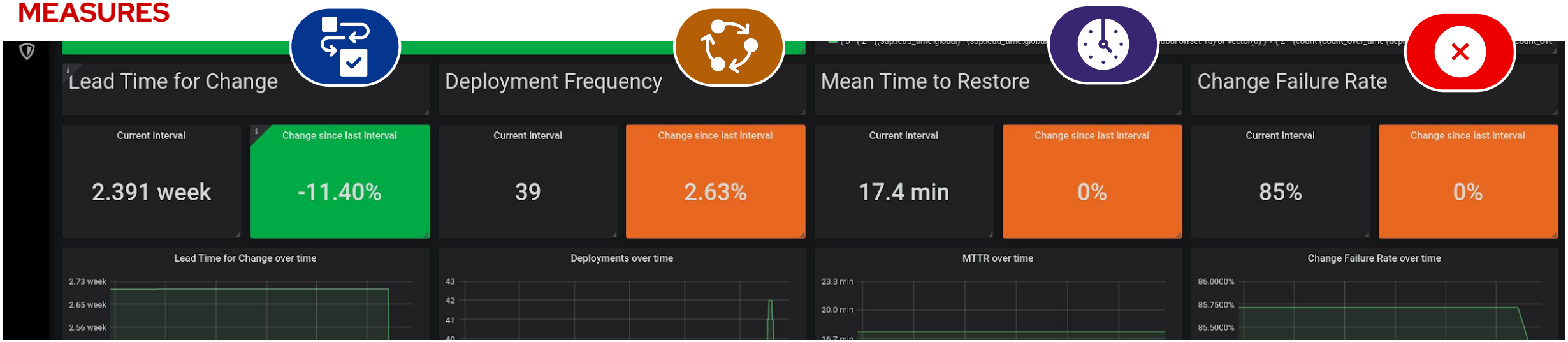
{deploy_time}

{commit_time}

{failure_creation}

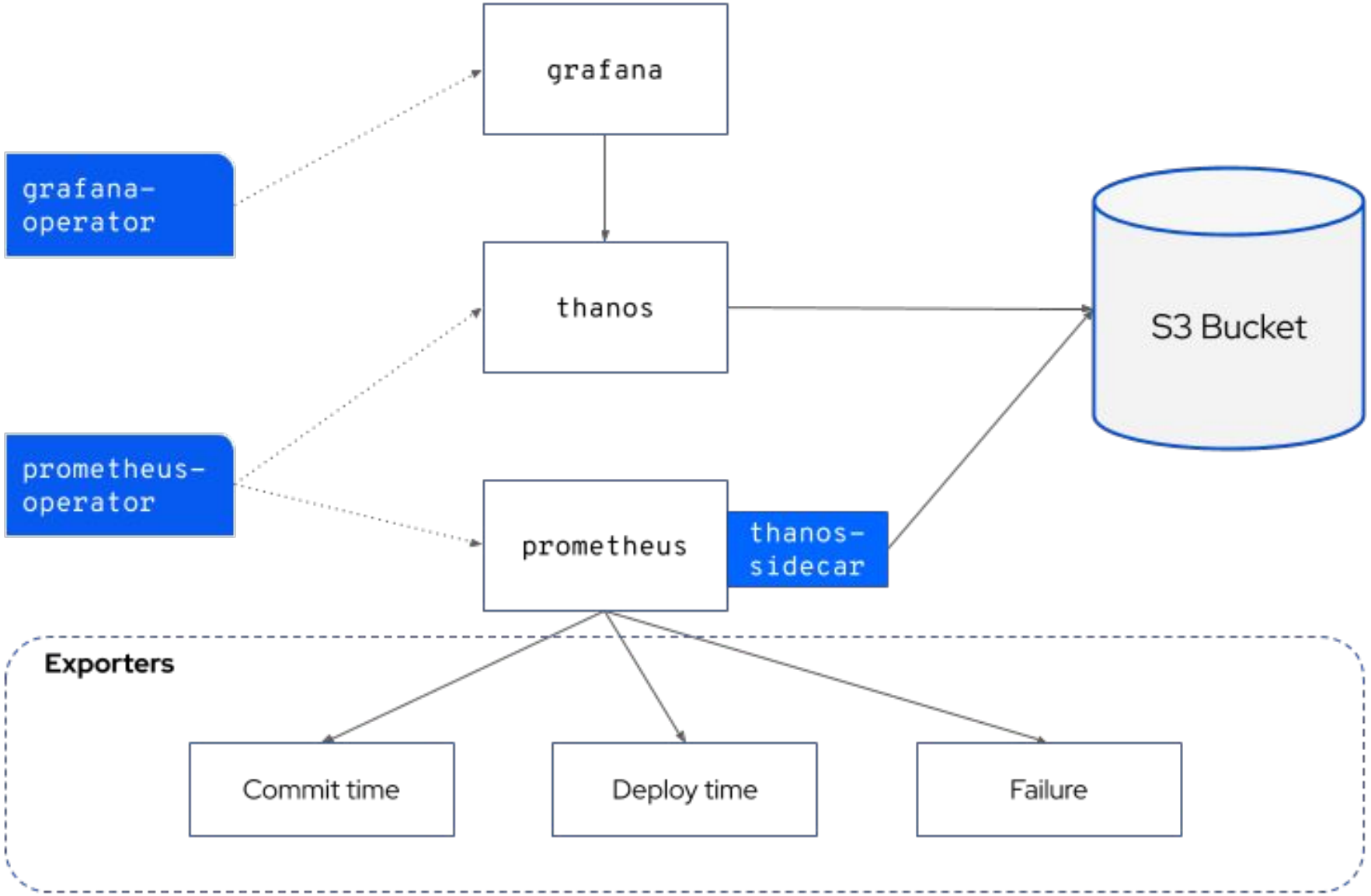
{failure_resolution}

MEASURES



Pelorus is composed of the following open source components:

- Prometheus Operator
- Prometheus
- Thanos (backed by Object Store)
- Grafana Operator
- Grafana
- Pelorus Exporters
  - Commit Time
  - Deploy Time
  - Failure
  - ...your own

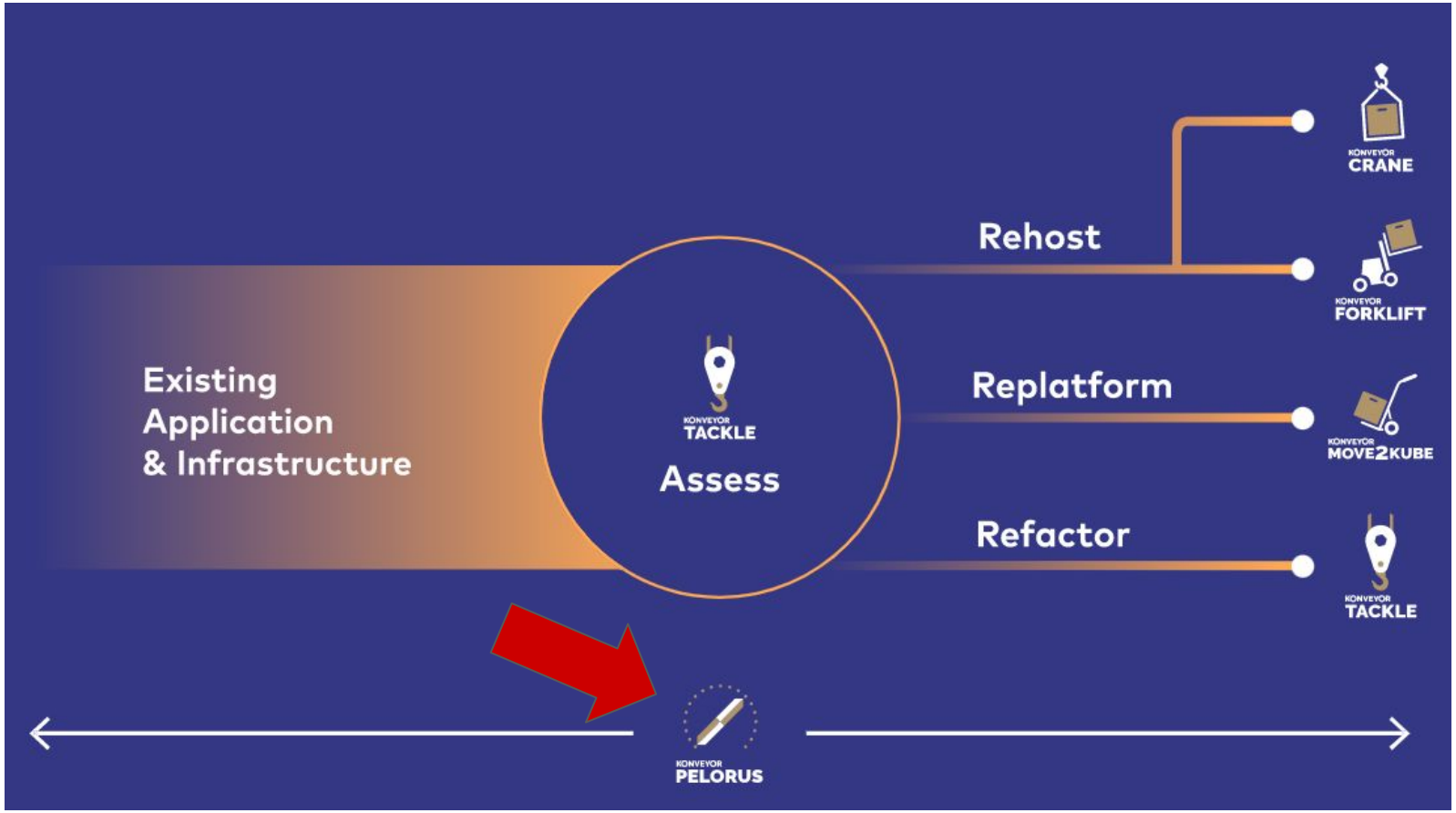


Some notes and recommendations

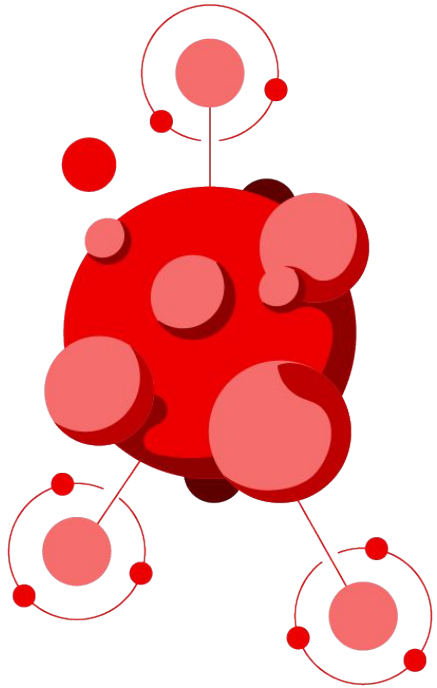
- As shown, you will need to re-evaluate your progress (and sometimes deviations) again and again over time as you are making changes to your processes, tools, architecture, etc.
- If you want to show to management that you have improved (and celebrate your success), create a long-term storage S3 bucket (e.g. via NooBaa or others)
- Pelorus is primarily based on Prometheus and Grafana. So, you can easily write your own exporters in any language or use the huge list of exporters readily available.

- Think about the outcome you'd like to achieve and visualise, define metrics and start building.

https://www.konveyor.io/



Q & A



Red Hat

Red Hat Developer

Join Red Hat Developer.

Build here. Go anywhere.

youtube.com/RedHatDevelopers

linkedin.com/showcase/red-hat-developer

facebook.com/RedHatDeveloperProgram

twitter.com/rhdevelopers

Learn containers, Kubernetes, and OpenShift in your browser.

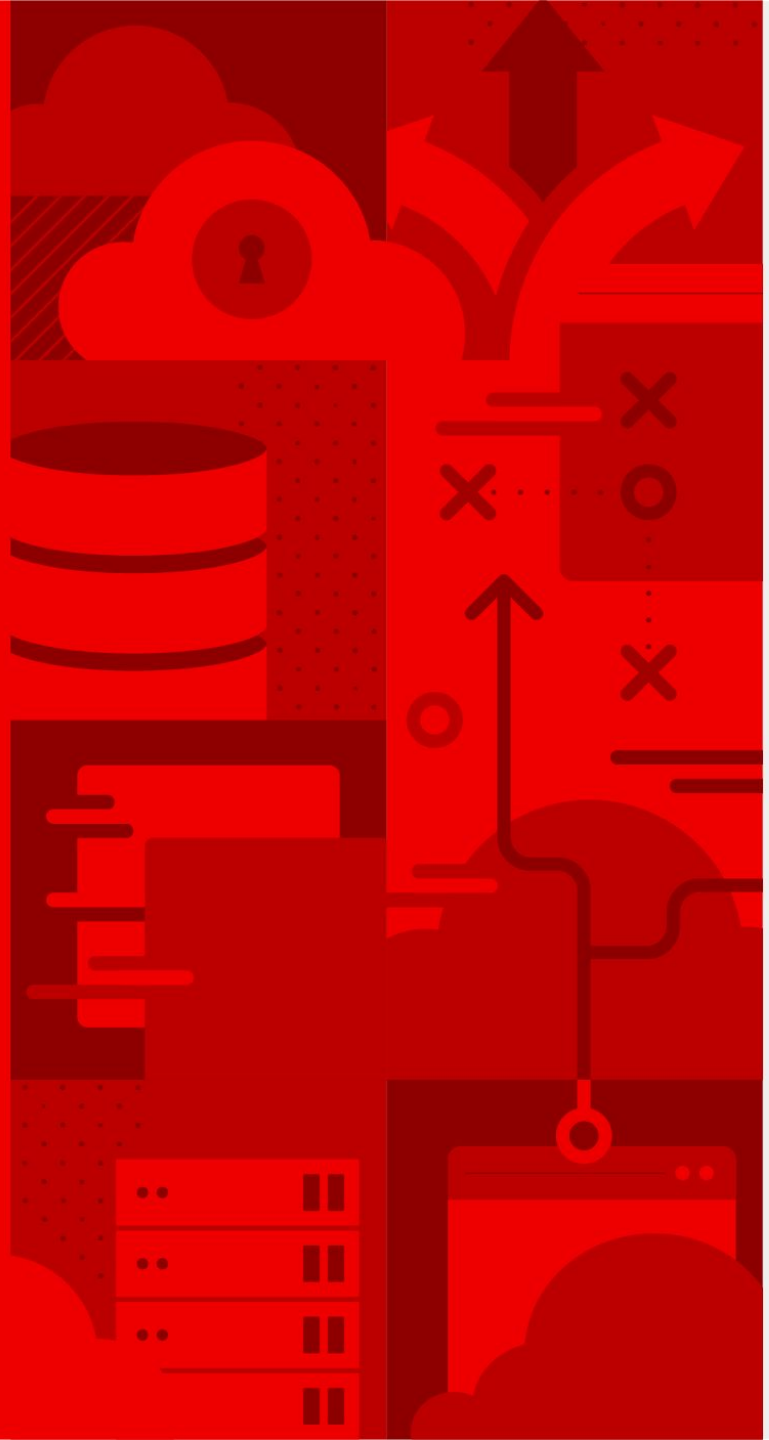
Start exploring in the OpenShift Sandbox.

Try Red Hat's products and technologies without setup or configuration.

developers.redhat.com/developer-sandbox

Red Hat

Red Hat Developer



Thank you!

linkedin.com/showcase/
red-hat-developer

facebook.com/redhatdeveloperprogram

youtube.com/RedHatDevelopers

twitter.com/RHDevelopers