



Adding value to test tooling

performance testing and test automation

Hoe we performance risico's ook in
een CI/CD wereld de baas blijven

Wie Ben Ik?

- >20 jaar ervaring in IT
- 10 jaar PerformanceArchitecten
- Software engineer -> performance test consultant
- Waarom deze presentatie?

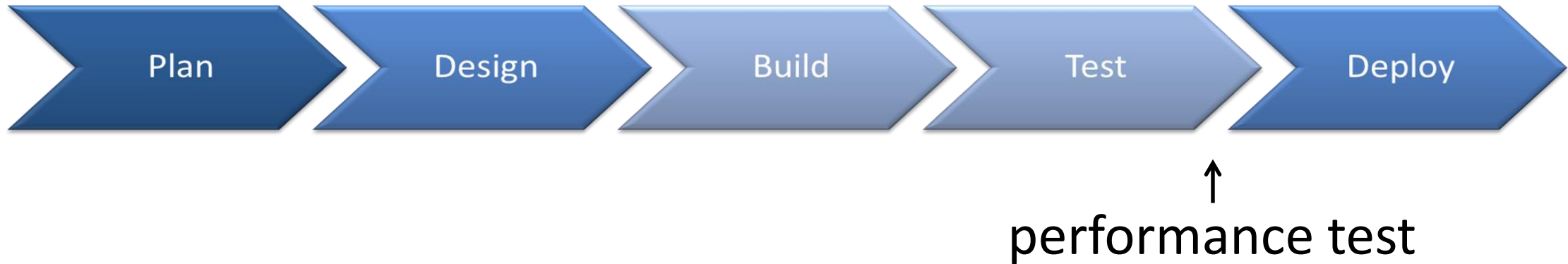
René Meijboom
Tel: 06-22882960
rene@performancearchitecten.nl



Agenda

- Performance testen (Traditioneel)
- Veranderingen door DevOps
- Uitdagingen
- Oplossingen
 - Shift left
 - CI/CD
 - Automatiseren performance testen
 - Continuous Monitoring
- Breder profiel performance engineer
- Kernpunten

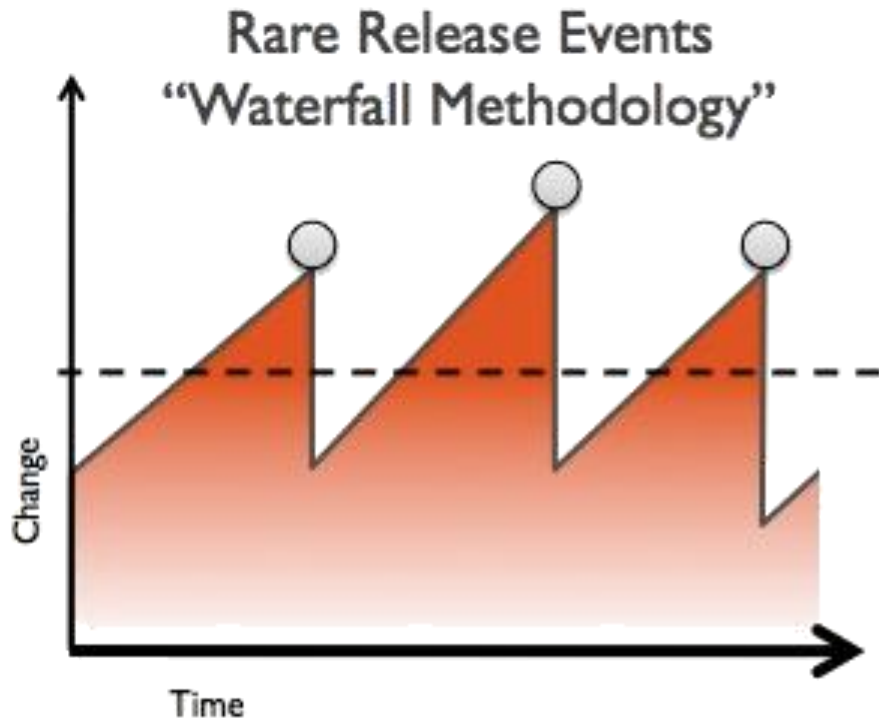
Traditioneel Performance Testen



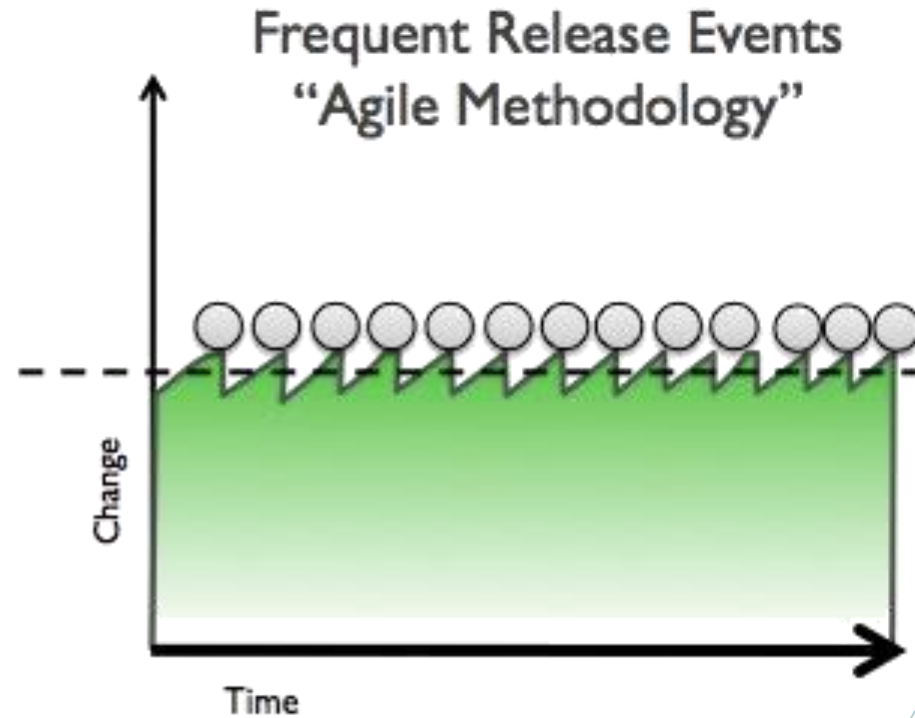
Nadelen Traditioneel Performance Testen

- Issues laat gevonden, hogere herstel kosten
- Bugfix leiden tot nieuwe build en test cyclus
- Nieuwe functionaliteit uitgesteld tot volgende release, of release wordt uitgesteld.
- Acceptatietest omgeving onder druk vanwege project en release testen.
- Geen goede aansluiting bij DevOps

Agile Release Frequentie



Effort Peaks
High Risk



Smoother Effort
Less Risk

Release Frequentie

Waterfall



DevOps



Code



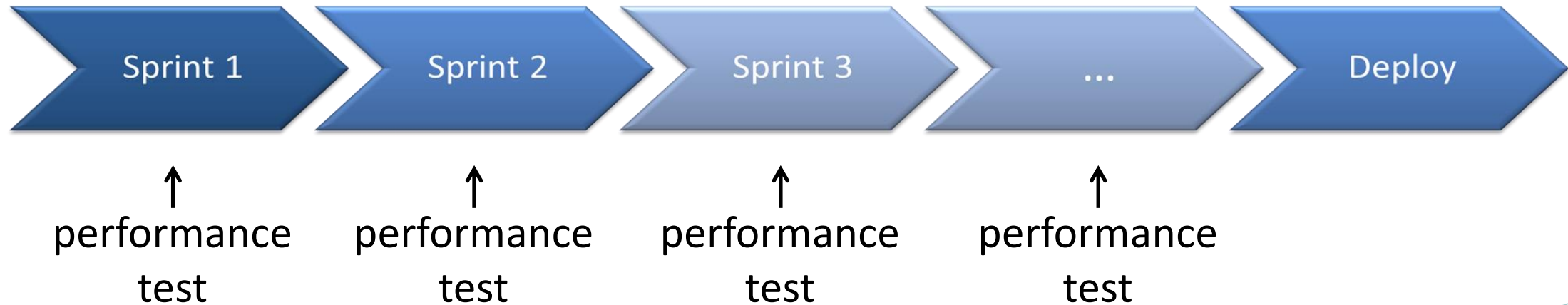
Test



Deploy



Agile/DevOps Performance Testen

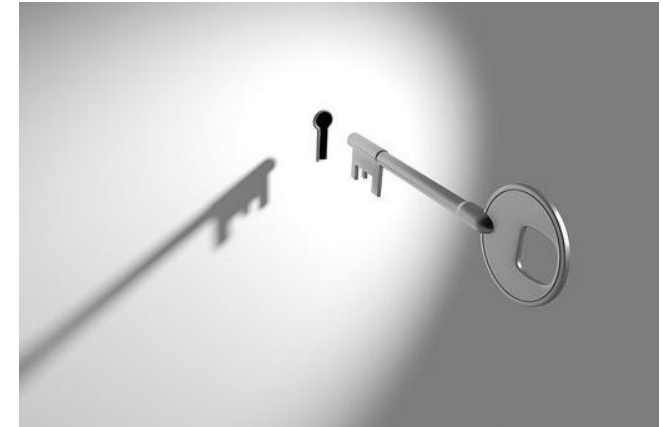


Performance Uitdagingen door DevOps

- Minder tijd voor performance testen
- Geen stabiel en volledig functioneel getest product
- Geen performance test kennis binnen agile/devops team
- Weinig aandacht voor PT (niet op DoD)
- Teams voelen zich niet verantwoordelijk voor performance

Oplossingen

1. Shift left
2. Continuous Integration / Continuous Delivery
3. Automatiseren van performance testen
4. Continuous Monitoring.

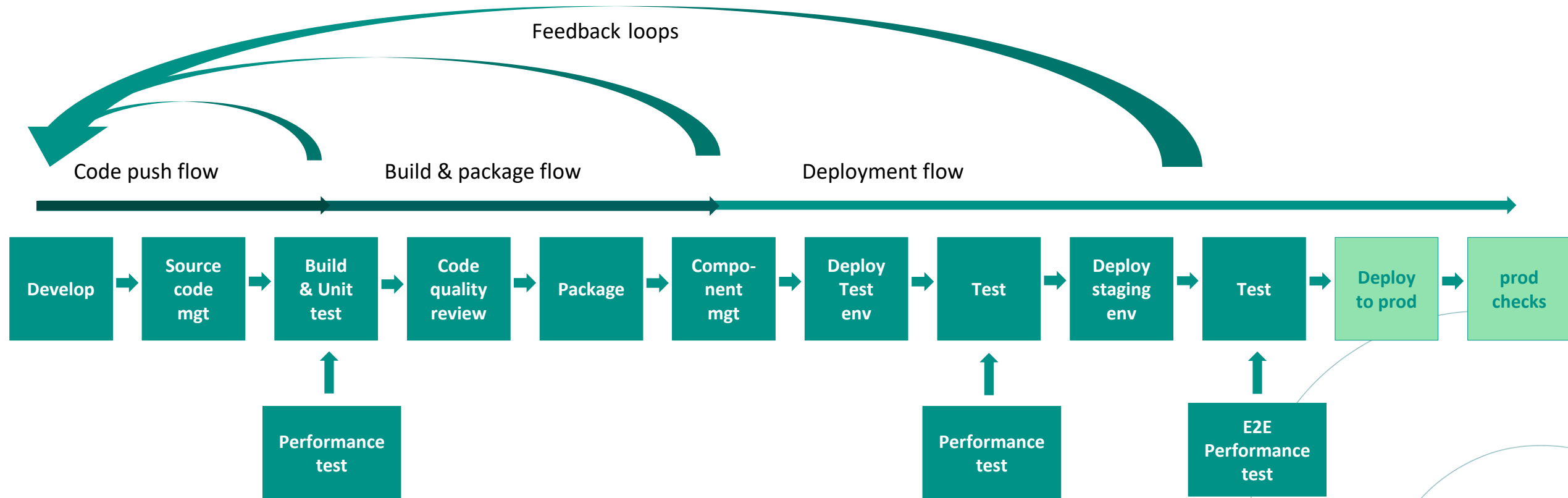


1. Shift Left



- Eerder testen
 - Van acceptatietest omgeving naar test omgeving
 - Van volledig functioneel getest naar functioneel stabiel
- Met als doel:
 - Eerder vinden van bugs
 - Hoger kwaliteit product in acceptatietest omgeving
 - Minder druk op (performance) test in acceptatietest omgeving

Performance Testen en Shift Left



Shift Left in 5 Punten:

1. Component testen

Feedback loops

2. In test ipv acceptatie omgeving

3. Gevirtualiseerde services / backend

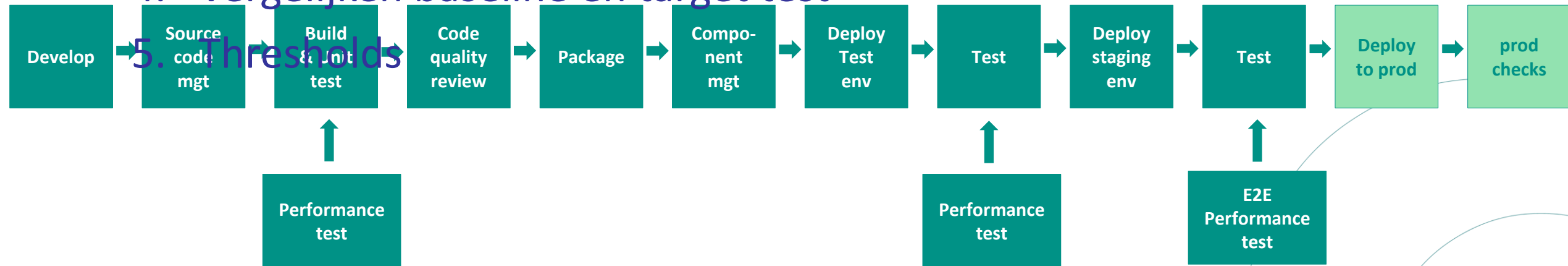
Code push flow

Build & package flow

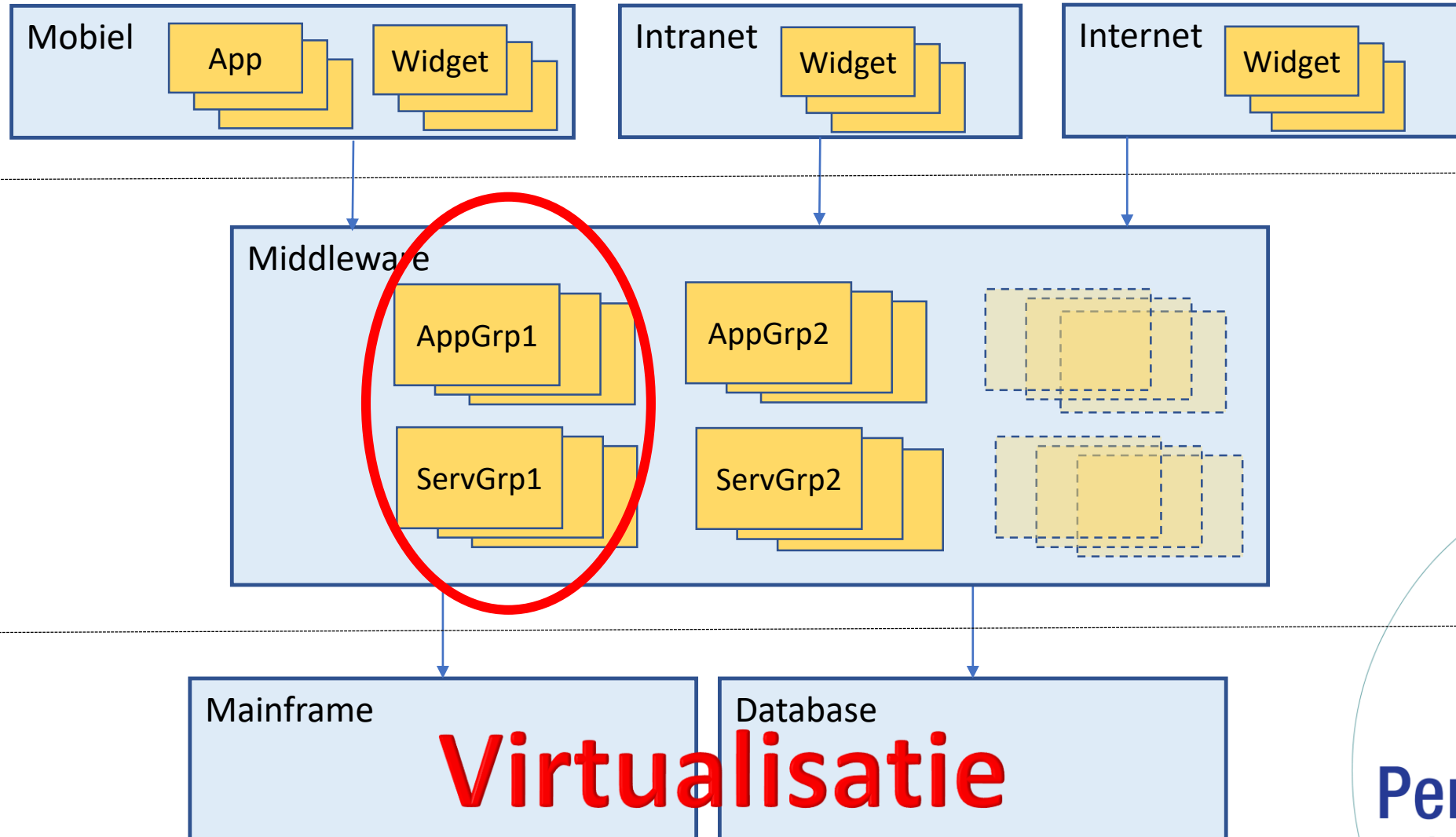
Deployment flow

4. Vergelijken baseline en target test

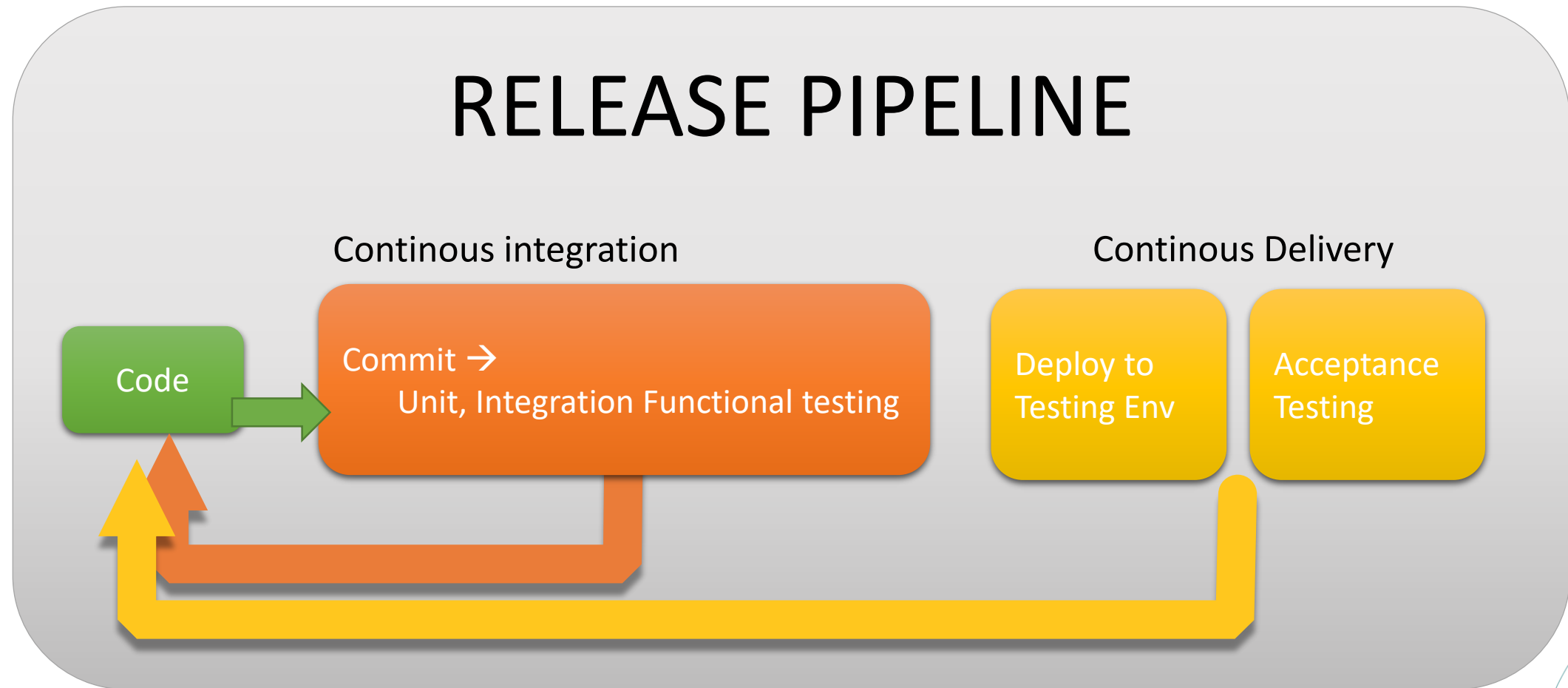
5. Thresholds



High Level Architecture



2. Continuous Integration / Continuous Delivery



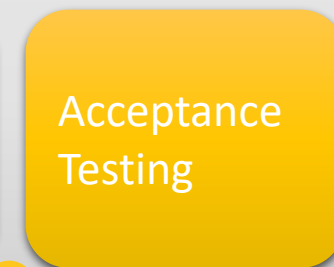
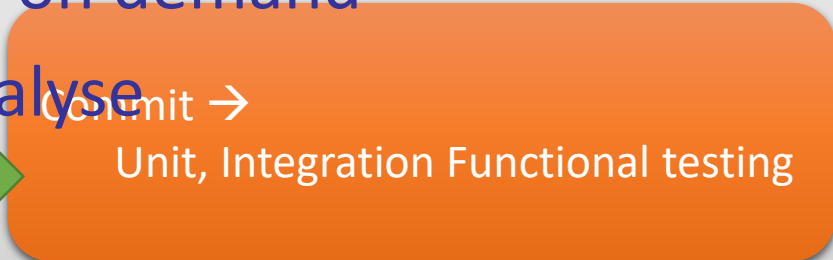
CI/CD in 5 punten:

RELEASE PIPELINE

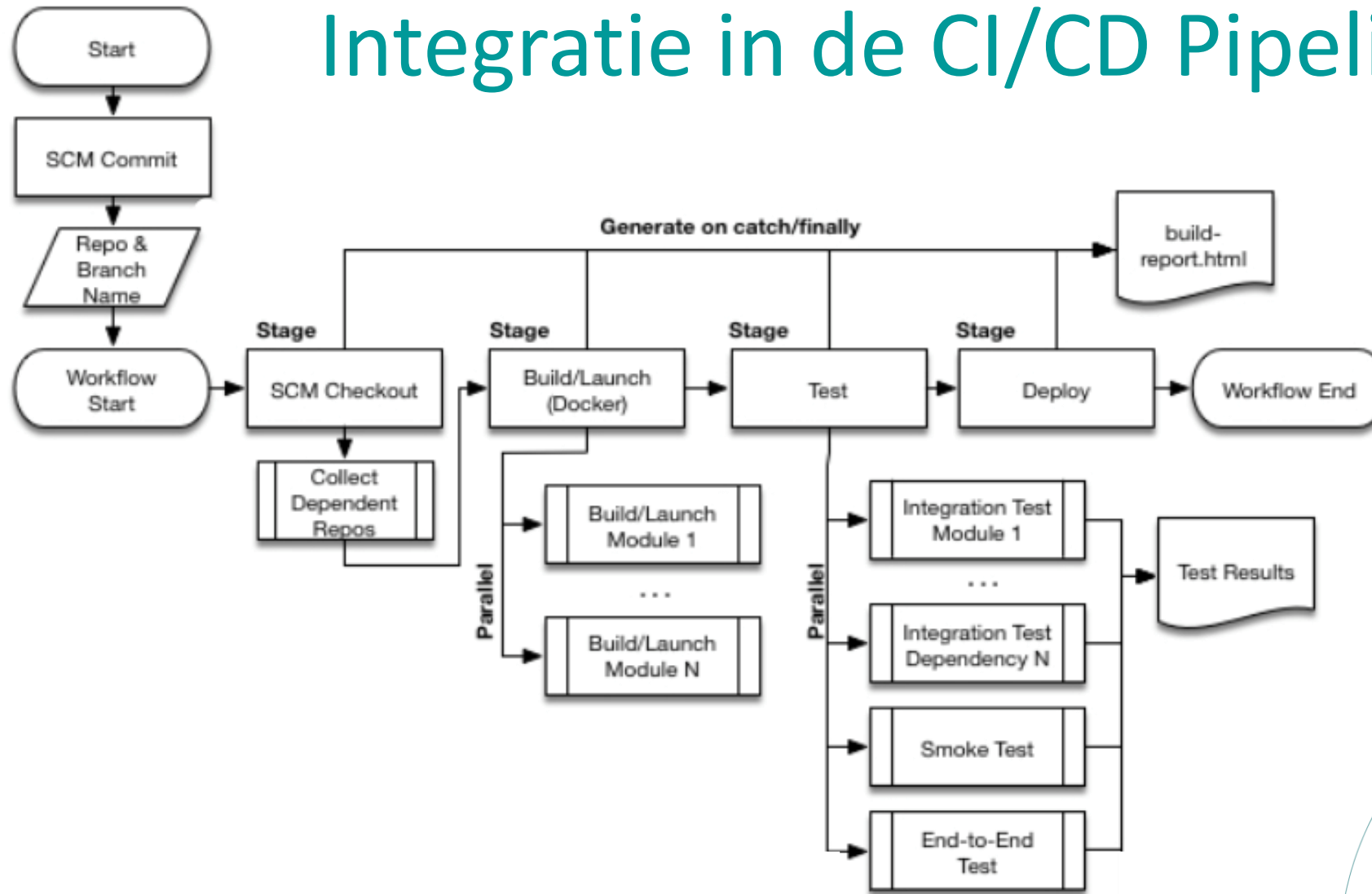
1. Integreer performance test in pipeline
2. Automatiseren performance test
3. Test data on demand
4. Trend analyse
5. Tooling

Continuous integration

Continuous Delivery

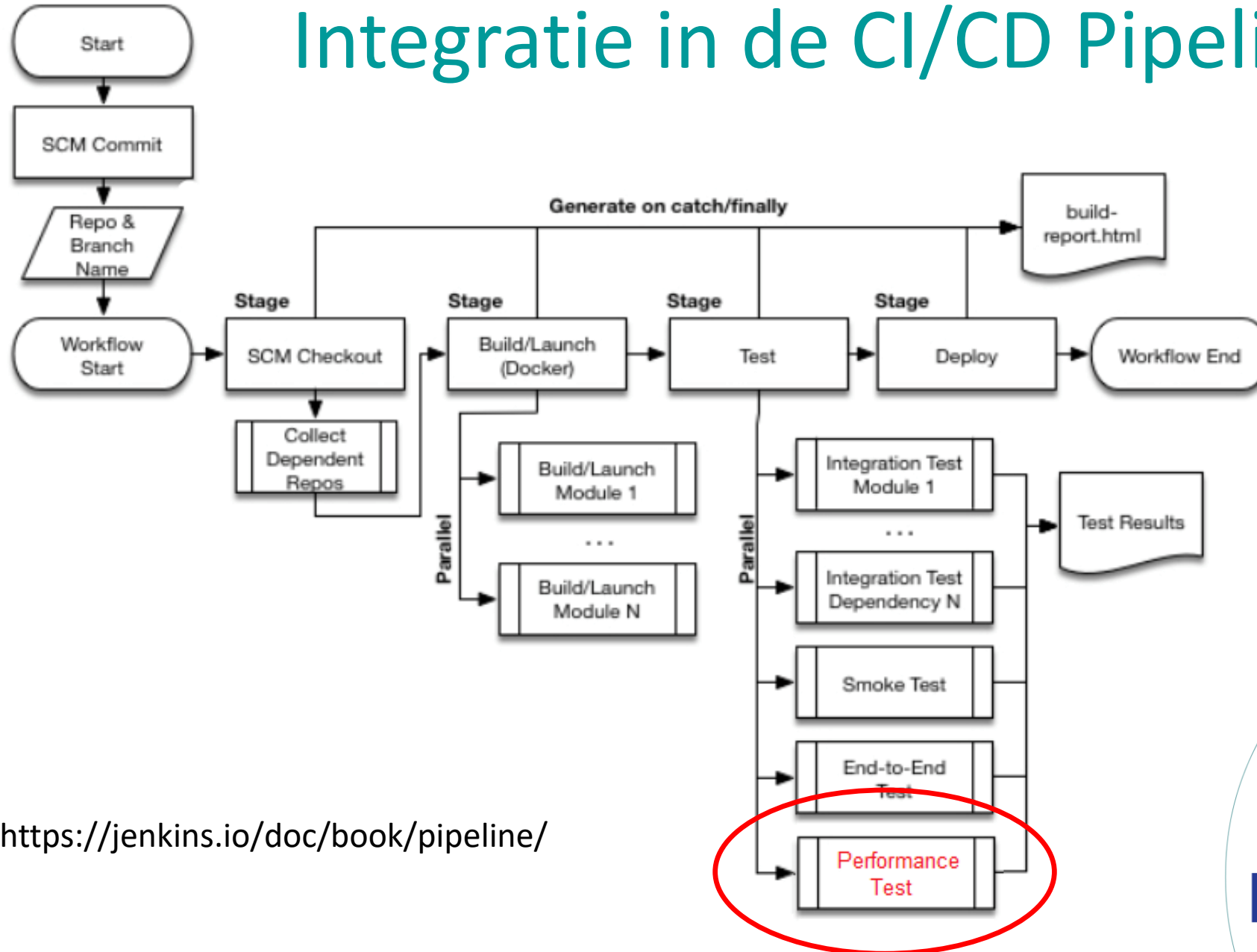


Integratie in de CI/CD Pipeline



<https://jenkins.io/doc/book/pipeline/>

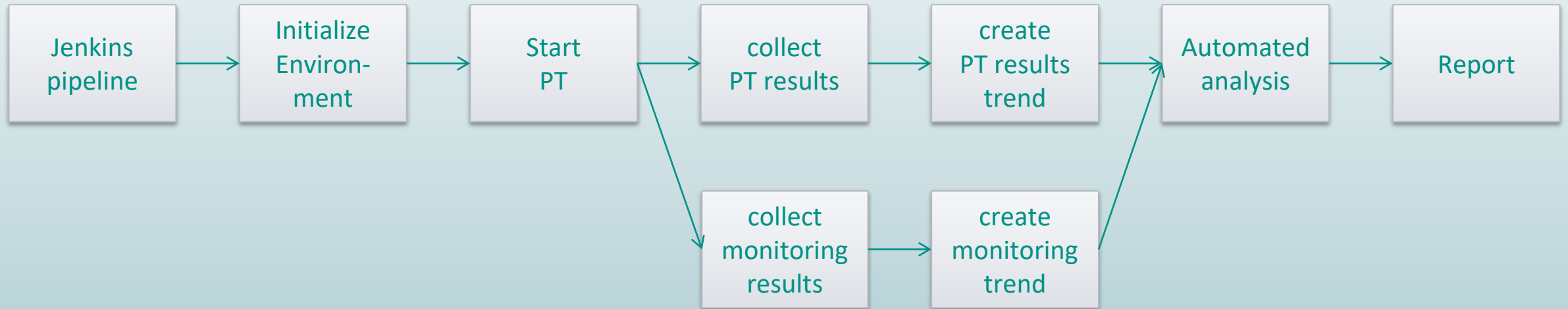
Integratie in de CI/CD Pipeline



<https://jenkins.io/doc/book/pipeline/>

3. Automatiseren van Performance Testen

Automation framework



4. Continuous (Performance) Monitoring

- Feedbackloop van productie na deployment
 - Business metrics
 - Technical metrics
- Monitor veranderingen en bijbehorende impact
- Alerting

Breder Profiel Performance Engineer

- Allround performance specialist
- Automatiseren
- Coach / teacher
- Soft skills
- Teamlid centraal team
- Teamlid Agile/DevOps team
- Guilds

Kernpunten

1. Test (componenten) zo vroeg mogelijk
2. Automatiseer performance test
3. Integreer performance test in CI/CD pipeline
4. Goede monitoring is een vereiste
5. Aanvullend performance testen blijft nodig

René Meijboom
Tel: 06-22882960
rene@performancearchitecten.nl



Zijn er nog vragen?

Performance
Architecten