

GET THE FUTURE YOU WANT WITH OUR LARGE S/4HANA TRANSFORMATION PROGRAM (LTP) METHOD



DAVID LAWSON
HEAD OF GLOBAL SAP COE



BARRY NEAVES
SAP TECHNICAL SOLUTIONS ARCHITECT



MARTINA ALEKSIEVA
METHOD DELIVERY LEAD



KISAN SHIRKE
QT&E GTM LEAD



ROBERT SHERWOOD
SR. ENTERPRISE ARCHITECT
TRICENTIS

01 OPERATE THE RENEWABLE
ENTERPRISE IN THE CONTEXT OF
LTP

02 THE NEED FOR AND IMPORTANCE
OF CONTINUOUS DELIVERY

03 THE CONTINUOUS DELIVERY
CHALLENGES

04 OUR RECIPE FOR SUCCESS

05 CONTINUOUS DELIVERY
TOOLCHAIN IN COLLABORATION
WITH TRICENTIS



SITUATIONS TO WHICH WE NEED TO RESPOND

COMPANIES ARE UNDER GREATER PRESSURES TO RESPOND TO NEW CUSTOMER EXPECTATIONS, MARKET OPPORTUNITIES, EVOLVING INDUSTRY DYNAMICS AND NEW TECHNOLOGY DISRUPTIONS

WHAT DRIVES SAP CLIENTS TO TRANSFORM AND INNOVATE ?

1 Innovation & Sustainability



- ▶ Need to incorporate new technologies into business processes
- ▶ Ever-increasing focus on sustainability throughout Supply Chain
- ▶ Market demands for new digital business models and product as a service
- ▶ Sustainability

2 Flexibility & Agility



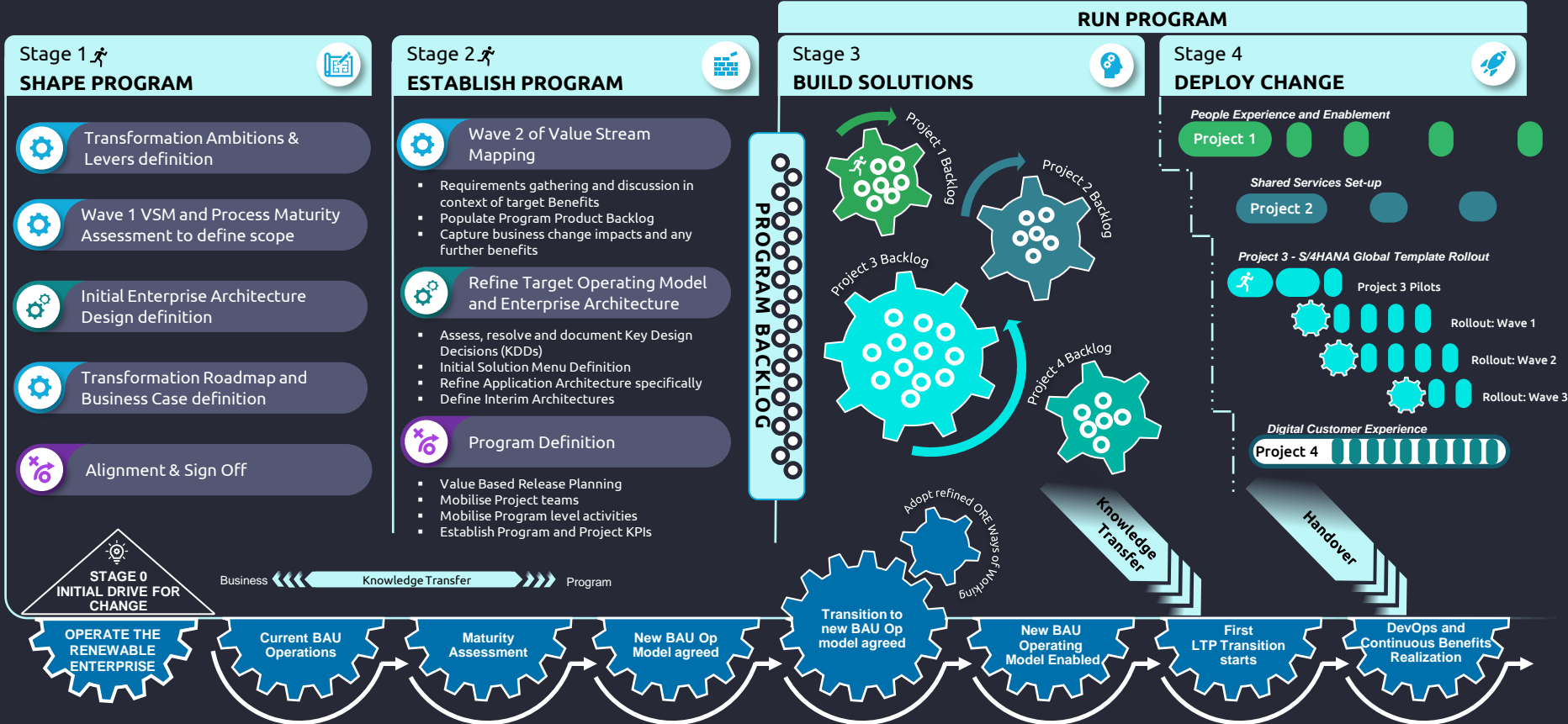
- ▶ Constantly changing customer demands
- ▶ Fast-paced unpredictable world (pandemic, climate, war, etc.)
- ▶ Company's internal processes not conducive to agile mode of business process delivery
- ▶ Clean Core
- ▶ Data Foundation modernization

3 Landscape Manageability



- ▶ Increasingly complex IT infrastructure
- ▶ New technological advancements every year
- ▶ Multiple applications required per role per department
- ▶ Technical debts
- ▶ Cybersecurity concerns across multi-cloud landscapes

WE DELIVER SUSTAINABLE BUSINESS CHANGE THAT GENERATES BUSINESS VALUE



THE 10 'MUST-HAVE' INGREDIENTS FOR SUCCESS

- 01 Continued effort investment into creating transparency and trust amongst different communities in the organisation
- 02 Product based organizational structure
- 03 Common governance and ways of working
- 04 Start early and engage the business throughout
- 05 Appropriate contractual construct
- 06 Shared backlog of requirements linked to business benefits, KPIs and change impacts
- 07 Effective release, dependency and change management
- 08 Composable Architecture
- 09 Right skillset and mix of expertise
- 10 Right mix of tooling

MOVE FROM PROJECT CENTRIC TO PRODUCT CENTRIC DELIVERY MODEL FOR DEVOPS

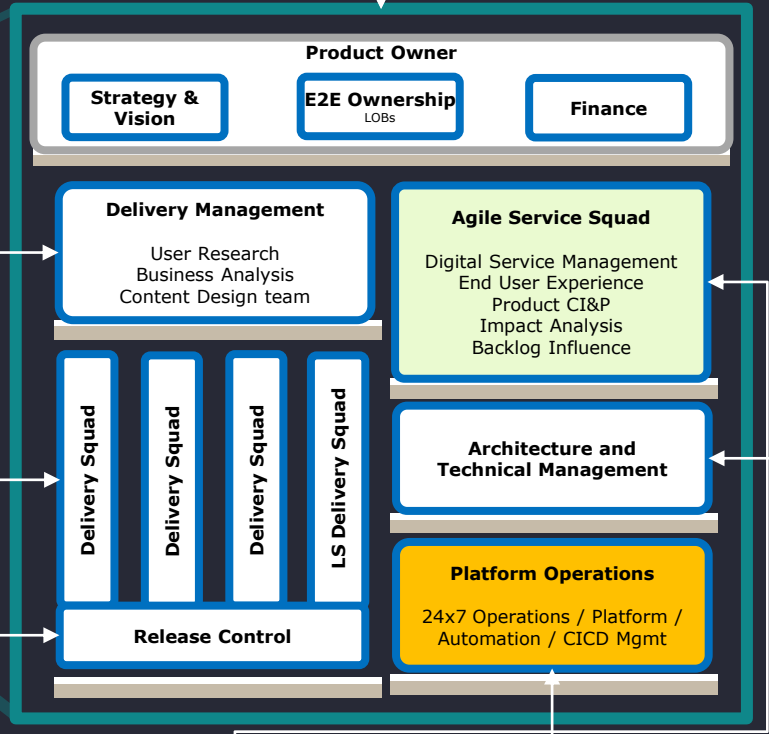
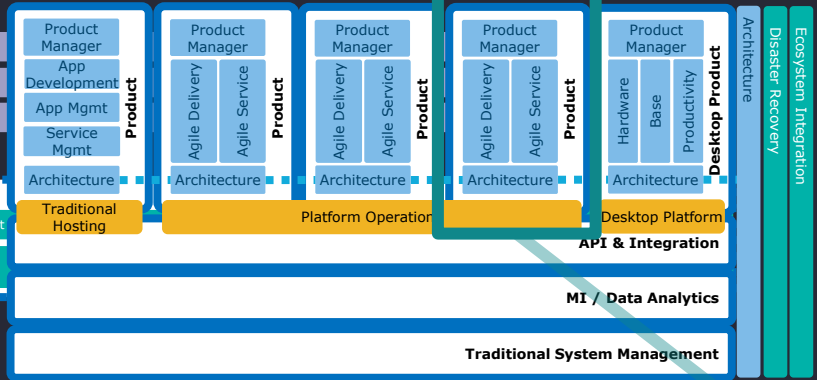
Single product owner that ensures the product is developed and managed correctly

Strategy and vision is combined and owned by the product owner along with the business relationship

DevOps is managed by the Agile Delivery function

(micro)services are developed and supported by the core Delivery Squad making the changes

- End User Contact
- Demand Management (Common Backlog)
- Communities of Practice
- Central Digital Service Mgmt
- Digital Operations Management
- Cross Product Operations Incident, Change and Release



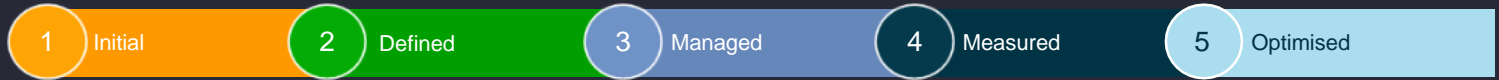
Product changes are managed through a common backlog & are combined into a release function to ensure consistency of deployment

The platform team is embedded within the product also to provide support and guide technology direction

Backlogs for development are influenced by the service/product owner to ensure CSI is maintained and the product remains sustainable

Architecture and technical management are embedded with the product using an "L" shaped role

DEVOPS MATURITY MODEL - OVERVIEW



DELIVERY ORGANISATION AND METHODOLOGY

1 Release scope poorly defined; subject to ad hoc change requests

2 Releases duration exceeds business need; releases face disruptive change

3 Release cadences well defined but exceeds business need; Requirements are stable;

4 Release on demand, multi-speed releases; time-box meets to business need (eg: monthly)

5 Small changes pushed through the pipeline ; Continuous deployment enables innovation

LEAN DELIVERY GOVERNANCE & PROCESSES

1 No defined or consistent applied delivery approach

2 Delivery approach is well defined, but largely manually governed

3 Industrialised project delivery; governance integrated into delivery and product management established

4 Blame free culture embedded in governance that uses real-time data rather than manual inputs

5 Continuous Delivery / Deployment supporting agile business change with automated governance

AUTOMATED RELEASE OF SOFTWARE

1 Mostly manual deployments

2 Some CIs automated, environment tailoring required, no enterprise tools

3 Fully automated Single-touch deployments into environments.

4 Functioning environments can be build from nothing programmatically.

5 Zero-touch zero-downtime deployments

CONTINUOUS INTEGRATION

1 No consistent use of version control – builds cannot be traced back to source code

2 Source code consistently managed in VCS; releases traceable to source

3 Developers integrate changes by checking into trunk on regular basis (daily)

4 Build is typically green – if build breaks developers do not make other changes until resolved

5 Build is typically green – if build breaks the CI tooling automatically reverses the failed change

CONTINUOUS DELIVERY (INCL AUTOMATED QA)

1 Fully manual test scripts

2 Testers run a harness / suite

3 Test harness / suite run automatically for some envs

4 Automated test suites enforce a quality gate

5 Tests run as functional monitoring

AUTOMATED OPERATIONS

1 No monitoring tools

2 Tools in place, but not configured beyond basic OS checks

3 Functional monitoring in place and DR is available

4 Environment and application health monitored and proactively managed

5 Service level monitoring (perf, usage) integrated with infrastructure and self-healing available

SW DEFINED INFRASTRUCTURE & CLOUD

1 Environments are managed ad-hoc without consistent blueprint

2 Environments follow standards but are manually created

3 Environments can be created automatically and follow common blueprint

4 SW defined infrastructure that closely resembles Production throughout SDLC

5 Utility based computing that leverages auto-scaling for several SDLC aspects

PLATFORM/ APPLICATION ARCHITECTURE

1 Big monolithic application architecture

2 Architecture based on platforms, business requires IT for every change

3 Resilient Architecture minimising single point of failures

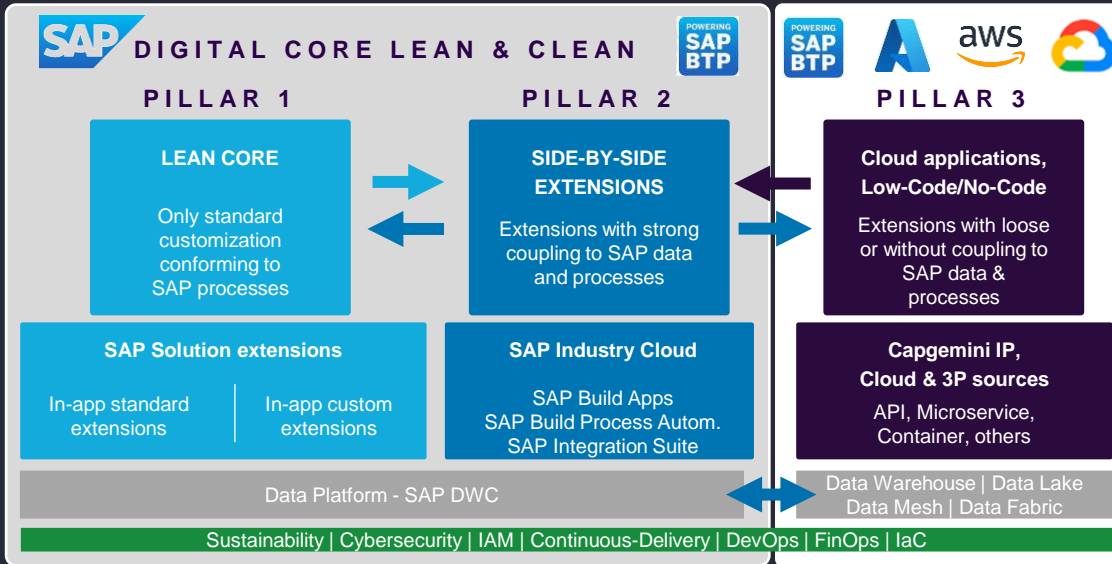
4 Modular and scalable architecture, that allows for some business self service configuration

5 Microservice based architecture supported by end-to-end ownership in teams

OUR MPSA AGILE ARCHITECTURE FITS CLEAN CORE APPROACH



Other than leveraging on SAP Packages, the proposed architectural approach allows to leverage on a fully extensible and customizable technology platform. We foresee to adopt our Capgemini's approach called **Multi-pillar Architecture (MPSA)**



1 Define your business domains

Decide in which domains we can build business advantages at the right place and on the right platform

2 Categorize them into pillars

Categorize applications into pillars, moving business differentiators and previous customization outside, so processes remain agile and minimize disruption

3 Carve out services

Carve out existing custom code during the transition to keep the core as lean and clean as possible

4 Move innovation to the edge

Build new custom applications at the edge using a combination of SAP, Cloud and Capgemini solutions

It's best to use standard functions of the core platform and move extensions to the SAP Business Technology Platform and Hyperscaler, while customized applications live outside the core all together

1

Architecture definition logic is fully compliant with SAP suggestions and guidelines

2

SAP Core is cleaned from customizing in order to keep it as much standard as possible

3

The innovation and customization are outside the core and leverage on SAP BTP & Hyperscaler (IaaS, PaaS, SaaS)

4

Built to allow change to be introduced, while protecting mission critical applications that run the business

WHY CLIENTS NEED CONTINUOUS DELIVERY?



Faster time to market



Improved Quality and Stability



Reduced Downtime impact



Increased Agility



Enhanced collaboration



Keep the Momentum

WHAT ARE THE CHALLENGES TO ACHIEVING CONTINUOUS DELIVERY?



**Change Control
and Impact
Assessment**



**Dependency and
Configuration
Management**



Build <=> Enhance <=> Fix



**Governance vs
Tooling**



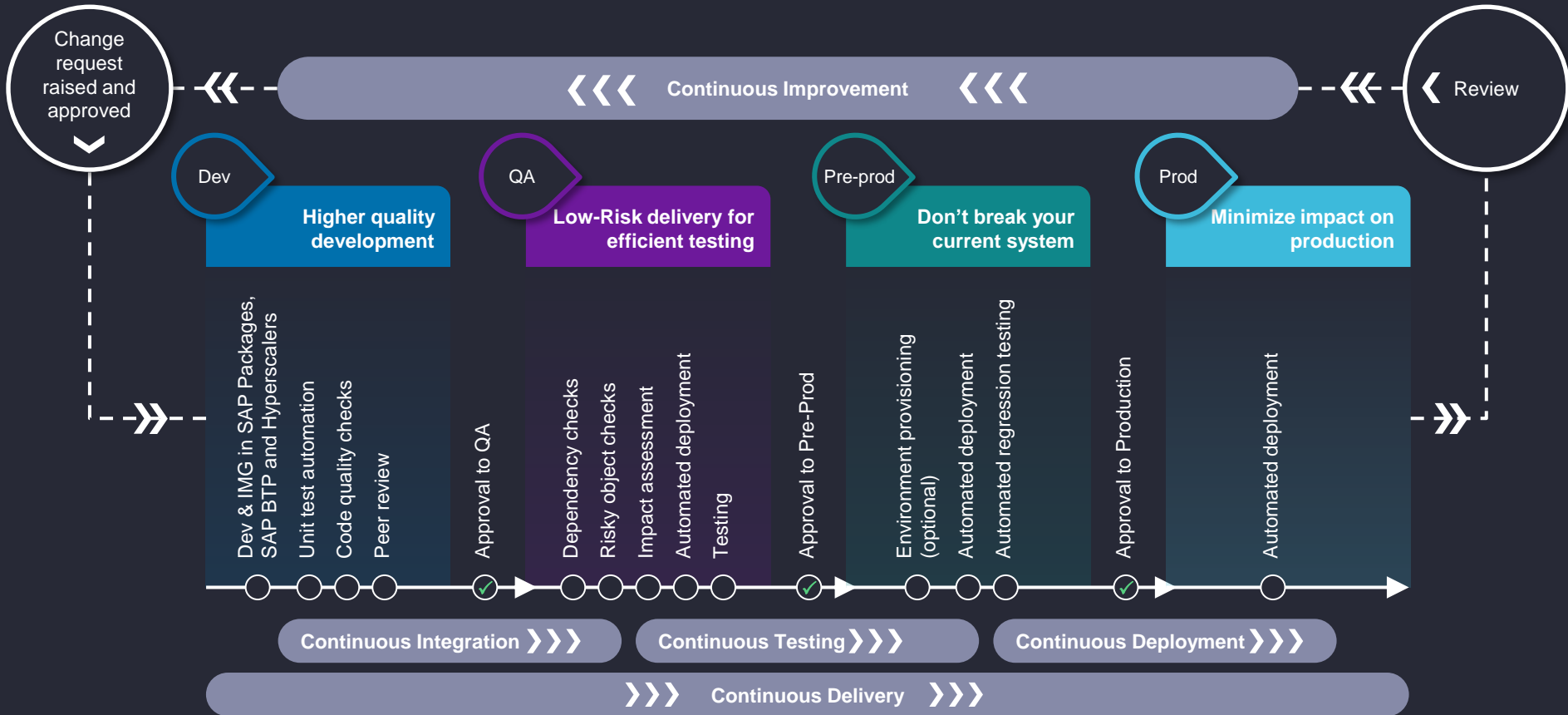
**Multi-vendor
Environment**



Skills and Mindset

CAPGEMINI CONTINUOUS DELIVERY

THE SAP DEVOPS DEVELOPMENT LIFECYCLES



SAFEGUARD CONTINUOUS DELIVERY

HIGH-QUALITY APPLICATIONS FASTER AND WITH LOWER RISK

Where is the risk?

Smart Impact
Analysis

Risk-based
optimization

Does it work?

AI-powered, end-to-end
Test Automation of
Enterprise & custom apps

Does it scale?

Enterprise
Performance
Testing

Are the decisions we making correct?

End-to-end financial
and analytical data
validation

Is it ready?

Test Management &
Analytics

Compliance audit trail

Digital validation for
health & life sciences



Safeguard Continuous Delivery



SAP Application Change

Impact Analysis

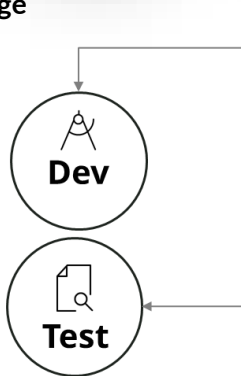
E2E Testing

Test Management

Go Live

Visibility of Change

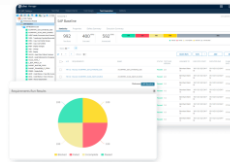
Where is the Risk?



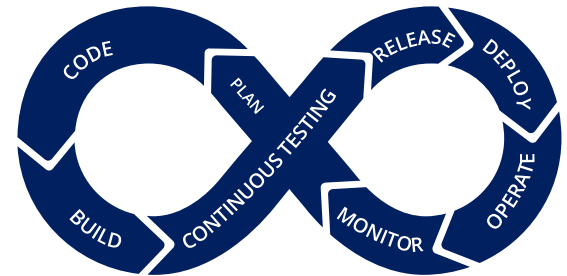
Does it Work?



Does it Scale?



Are we Ready?



IT IS NOT EASY TO ACHIEVE... AND IT IS ALL POSSIBLE

FEELING OF UNCERTAINTY, CONFUSION AND NERVOUSNESS OF
HOW COMPLICATED FOSTERING SAP DEVOPS IS



To get
from here...



To here...



DEVOPS

FEELING OF CONFIDENCE GROWTH, CAPABILITY IMPROVEMENT AND
ENABLING ENVIRONMENT FOR SAP DEVOPS

YOU NEED TO:

- Put in place the right mix of governance and tooling
- Question whether your contractual arrangements support your goals
- Ensure that Business and IT stakeholders are aligned around the change
- Be prepared to make some mistakes at the beginning, "Fail faster to succeed sooner"
- Be ready to encourage, sponsor and enable change in mindset, organisation and team skillsets
- Remember that there is no simple answers to complex problems and there will always be room for further improvement



**GET THE
FUTURE
YOU WANT**