



ARCTIQ

A Benchmark Company

➤ Assessment One Pager

➤ Consultation-as-a-Service (DevOps/DevSecOps)

Prepared for External Customers

2023



DEVOPS EQ

DevOps Consultation-as-a-Service

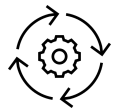
DevOps Maturity Assessment - focused on development effectiveness and efficiency



Understand the motivation and engagement level within your teams



Identify the human effort in your Software Delivery Life Cycle

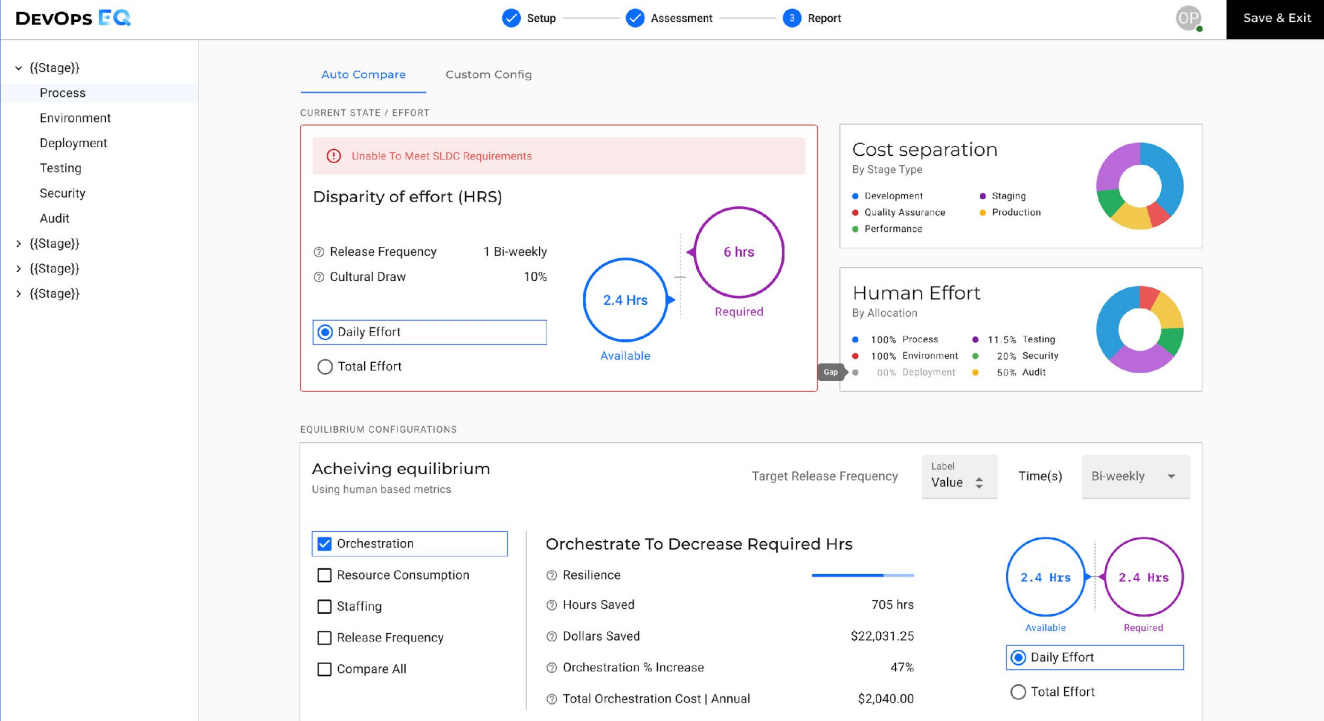


Identify process improvements to remove your talent heroics

CHALLENGE: Organizations are unable to sustain and measure DevOps adoption

- Failing DevOps adoption due to wrong vision and lack of talent management
- Stagnation and lack of optimized approach
- Disparate patterns and maturity due to DIY (Do It Yourself)
- Introducing Best In Class (BIC) tools but without cohesivity/orchestration

We help you pick the best patterns and help you mature with the existing toolset



Minimizes Risk

of hours of downtime in production

Minimizes Errors

% of human errors in deployments to production

Minimizes Effort

of production epics and stories orchestrated – unbreakable pipelines



Let us implement your roadmap with our elite DevOps Engineers. To book your assessment, please contact robert.tran@arctiq.ca



> Assessment Flow / Level of Effort (per offering)

To conduct the assessment, we'll have the following sessions with the teams, these sessions will be scattered over 3-4 weeks based on your team's availability

Sessions	Time	Participants
Kick Off – Level Set Objectives	30 mins	Key stakeholder of the SDLC process – team leads (Dev, Ops, Sec, Infra)
Working Session 1 – Mapping out the current state using a visual drawing workflow	90 mins	Tech and team leads responsible for SDLC process, maximum participants from QA teams, development, change, security, operations, infrastructure etc.
Working Session 2 – Running Arctiq applications	60-90 mins	Same participants as session 1
Report Review / Executive Summary Walk Through (top 4 priorities)	30 mins	Client to decide
High level roadmap / Proposal (Duration / Scope Order)	30 mins	Client to decide
SOW/RFP content for implementation	30 mins	Stakeholders per outcome / Procurement

Assessment Assets (Artifact Register)

Providing you the DO portion for your next journey steps

Working Session

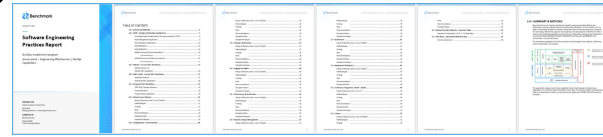
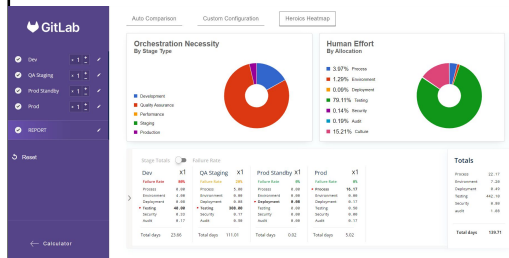
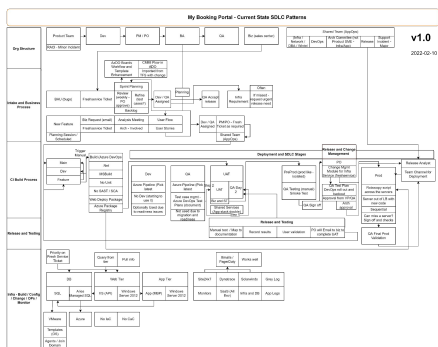
Mapping your SDLC/AppSec/SRE from Intake to Production

Working Session 2

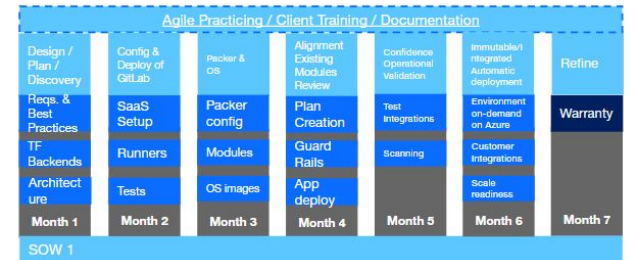
Measurement of Human Effort/Observability capabilities/Security coverage 8 Risks

Review / Briefing / Next Steps Sessions

- Optimized - CI/CD SDLC flow/ Observability architecture / DevSecOps design - including client desired direction
- Providing a comprehensive and optimized report with recommendations, risks, findings, catered archetypes and artifacts
- Executive briefing of priorities and reasoning for stakeholder support
- HLP/Roadmap - Mapping 12 months ahead the actual lift (and support where required)
- SOW - Supporting the stakeholders to achieve the first and most valuable outcome



- Rating of Maturity • Methodologies • Findings • Risks • Recommendation • Example Artifact for your use case • Example Archetype



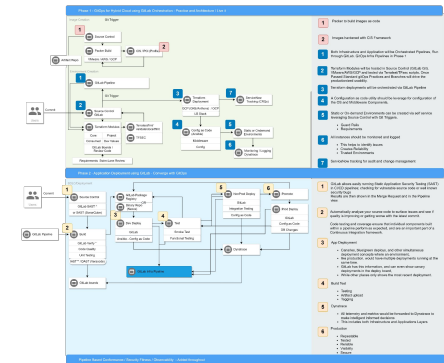
Live Drawing (draw.io)

2 Process + DevSecOps (AppSec)

Metrics PDF • Soon to release SaaS account

Findings Requiring Executive Attention

Concern	Impact and Risk
1 Governance / Security is required but entirely a gap	No security controls are being implemented (0) for any effort (Infosec or AppSec). Complete gaps on the artifact build side. This is considered a large PCI challenge and requires immediate action. This cannot be delegated to the transaction provider.
2 Testing is entirely manual causing failures and user experience issues (coverage and consistency gaps)	This is the highest level of burden (60% of the total) and allows releases to fail throughout the SDLC. Dev stages do not have enough testing which causes 2/3 ratio of repeat failures in QA. Testing can be automated and be validated using Dynatrace and integrated testing frameworks.
3 Inefficient process and disparate delivery methods without verification	Development and production deliver using different deployment methods. This produces a lack of consistency and reliability. Due to the lack of testing and consistent deployment approaches process is heavy to prevent failures.
4 Infrastructure/Configuration is created manually	The environments have drift due to hand crafting. Some environments can't be created, and others become shared due to the inability to rebuild. This reduces confidence in releases as testing becomes less related to prod-like systems and the environments to improve confidence can't be stood up.



*Benchmark Confidential 7