xoriant
Imagination Realized

DevOps Tools and Process Transformations Journey

Customer Challenges

Lack of Developer Productivity Tools

On-premise SVN repositories lack essential developer productivity tools.

Complex Repository Structure

Over 90 application codes are added to a single folder in a trunk-based repository.

This structure causes complexity, requiring developers to download approximately 15GB of code locally, even for specific application work.

Inefficient Workflow for Single Projects

Developers must download the entire repository, including all 90+ projects, to work on a single project.

Manual Deployment Process

Deployment for over 20 customer sites is entirely manual, making the process tedious and error-prone.

Each site deployment, including database changes and customizations, takes approximately 4-5 hours.

Complex Rollback Process

Rollback procedures are manual and complex, with numerous checkpoints.

Lack of Security Checks

There are no security checks at the code level, such as Static Application Security Testing (SAST) and Dynamic Application Security Testing (DAST).

In Scope / Out of Scope Activities

In Scope

SVN to GitHub Migration

- Migrate on-premise SVN repository for 90+ applications to SaaS GitHub Enterprise version
- Develop 25+ CI/CD Azure DevOps Pipelines to automated deployment and rollback process for .NET, Angular, ASP.NET and Python Stack

GitHub Advance Security Management

- Secure GitHub organization, repositories and branches using best practices
- Advance code level security (Dependency Graph, Dependabot, Dependabot alerts and SonarQube)

Code Segmentation

- Repository Segmentation
- Environment-specific branching (Dev, QA, UAT and PROD)
- Implement Release Tagging and Access Control
- Implement Automated Security Checks (Dependency Graph, Dependabot, Dependabot alerts and SonarQube)
- Modify existing Azure DevOps Pipeline with new repository structure

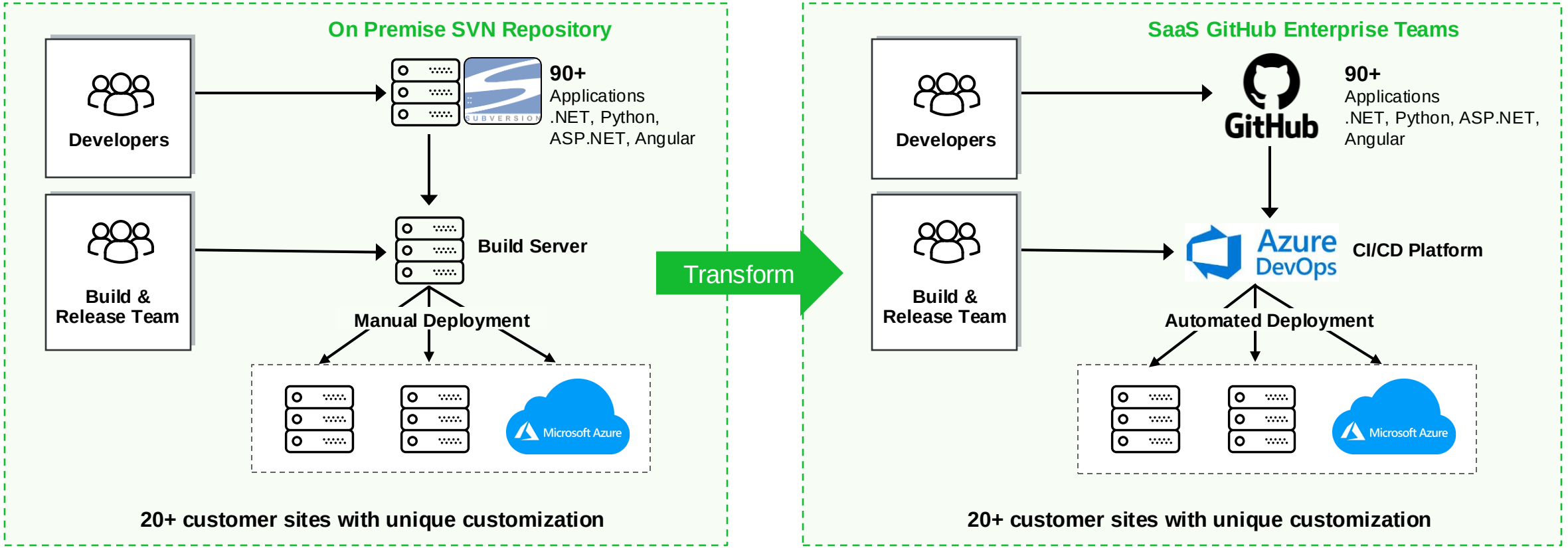
Code Obfuscation

- Provide consultancy on code obfuscation tools.
- The code obfuscation to be implemented using open-source tool ConfuserEx for 5-10 files max

Out of Scope

- Application Development
- Code Refactoring
- Application Performance Management

DevOps Tools and Process Transformation



Challenges

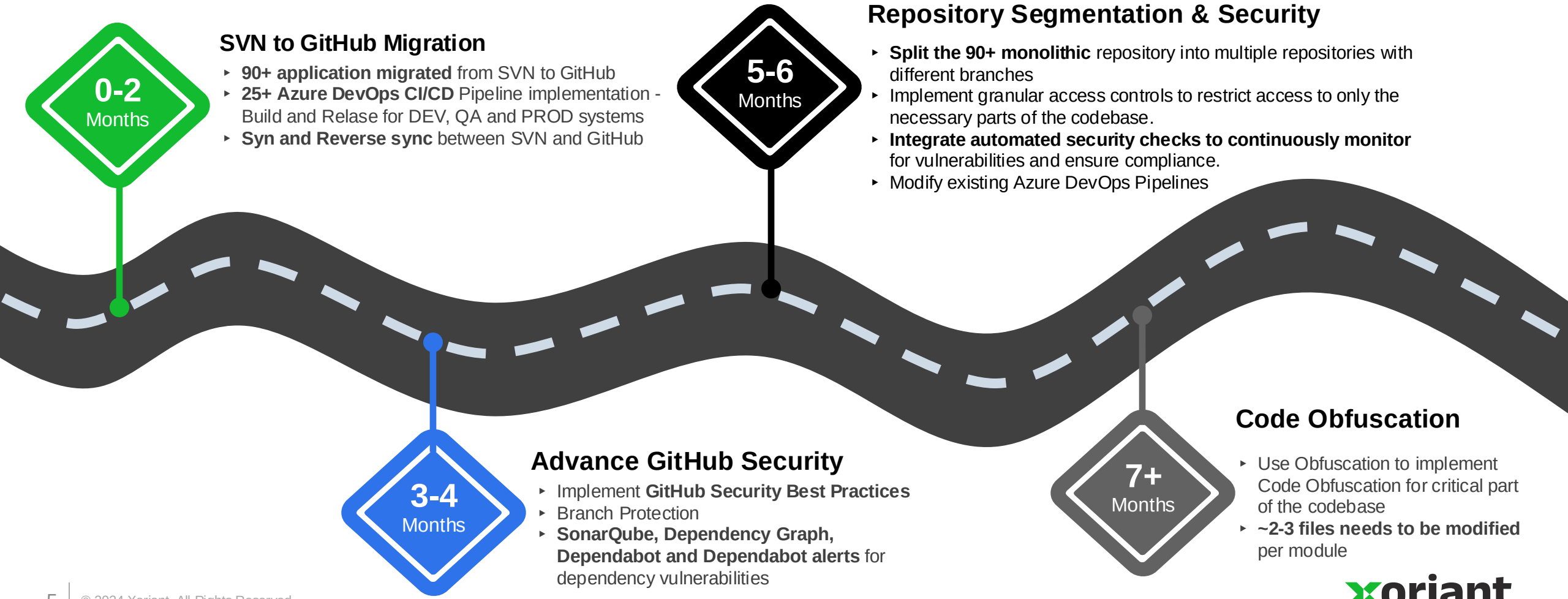
- **Manual deployment on over 20 customer sites** with unique customizations
- Deployment rollbacks are tedious
- Lack of security checks during application deployments
- Each site deployment, including database changes, **takes approximately 4-5 hours**
- On-premise SVN tools lack developer productivity features

Benefits

- Automated deployment across over 20 customer sites with unique customizations
- **Enhanced developer productivity** with integrated tools in Azure DevOps
- **Advance security controls and checks** in CI/CD Pipeline
- Streamlined and simplified deployment rollback processes
- Reduced deployment time per site, including database changes, to **significantly less than 20-30 minutes**

Migrating from SVN to GitHub DevOps Journey

90+	25+	5
Total Application	**CI/CD Pipelines**	**Security Tools**



0-2 Months

SVN to GitHub Migration

- **90+ application migrated** from SVN to GitHub
- **25+ Azure DevOps CI/CD** Pipeline implementation - Build and Relase for DEV, QA and PROD systems
- **Syn and Reverse sync** between SVN and GitHub

3-4 Months

Advance GitHub Security

- Implement **GitHub Security Best Practices**
- Branch Protection
- **SonarQube, Dependency Graph, Dependabot and Dependabot alerts** for dependency vulnerabilities

5-6 Months

Repository Segmentation & Security

- **Split the 90+ monolithic** repository into multiple repositories with different branches
- Implement granular access controls to restrict access to only the necessary parts of the codebase.
- **Integrate automated security checks to continuously monitor** for vulnerabilities and ensure compliance.
- Modify existing Azure DevOps Pipelines

7+ Months

Code Obfuscation

- Use Obfuscation to implement Code Obfuscation for critical part of the codebase
- **~2-3 files needs to be modified** per module

xoriant

www.xoriant.com

Migrating from SVN to GitHub DevOps Journey

90+	25+	5
Total Application	CI/CD Pipelines	Security Tools

0–2 Months

SVN to GitHub Migration

- **90+ application migrated** from SVN to GitHub
- **25+ Azure DevOps CI/CD** Pipeline implementation - Build and Relase for DEV, QA and PROD systems
- **Syn and Reverse sync** between SVN and GitHub

3–4 Months

Advance GitHub Security

- Implement **GitHub Security Best Practices**
- Branch Protection
- **SonarQube, Dependency Graph, Dependabot and Dependabot alerts** for dependency vulnerabilities

5–6 Months

Repository Segmentation & Security

- **Split the 90+ monolithic** repository into multiple repositories with different branches
- Implement granular access controls to restrict access to only the necessary parts of the codebase.
- **Integrate automated security checks to continuously monitor** for vulnerabilities and ensure compliance.
- Modify existing Azure DevOps Pipelines

7+ Months

Code Obfuscation

- Use Obfuscation to implement Code Obfuscation for critical part of the codebase
- **~2-3 files needs to be modified** per module

xoriant