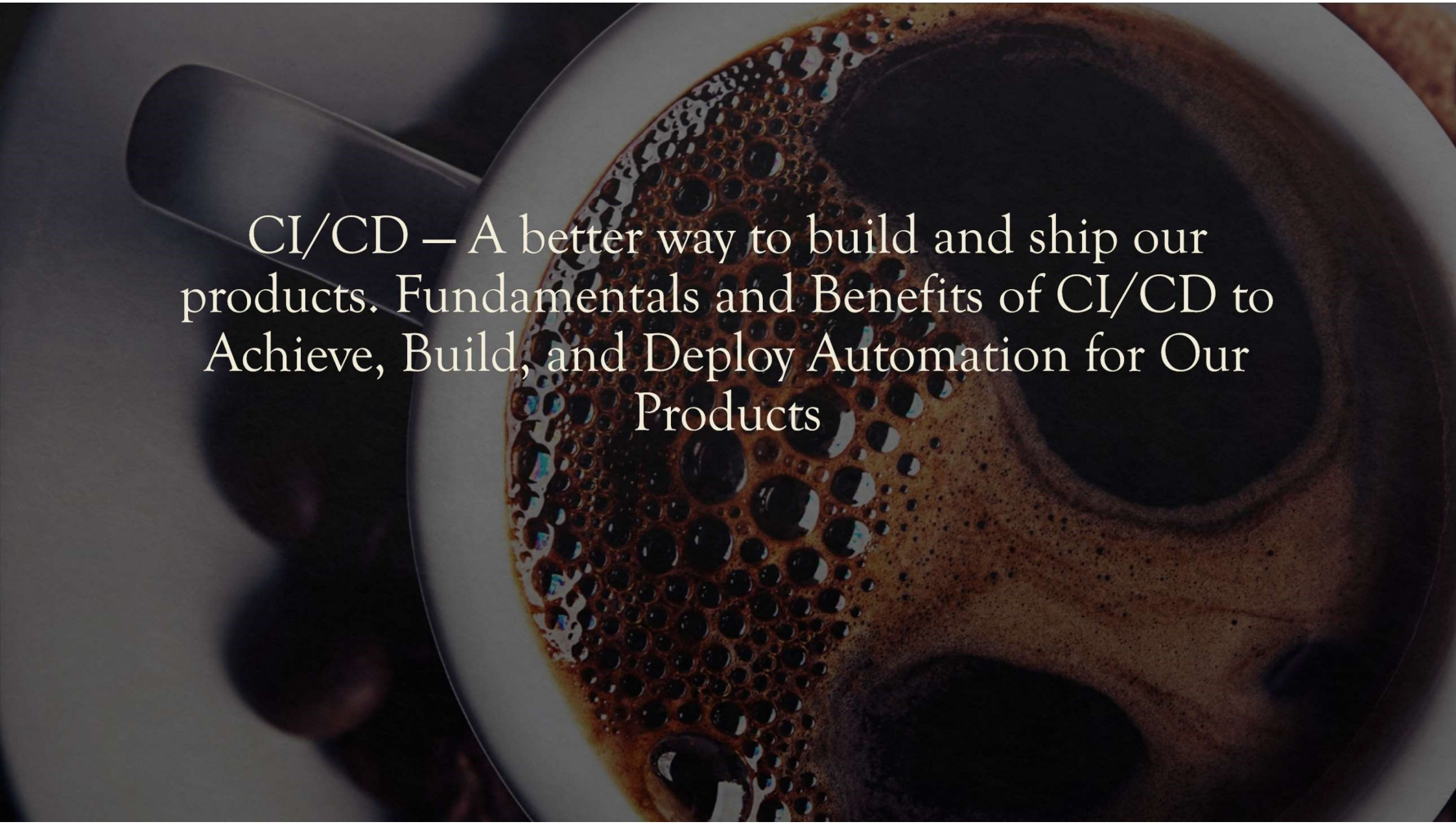
CI/CD – A better way to build and ship our products. Fundamentals and Benefits of CI/CD to Achieve, Build, and Deploy Automation for Our Products

The practice of merging all developers' working copies to a shared mainline several times a day to avoid conflicts in the code in the future. It's the first step towards ensuring that we have a high quality, deployable artifact. Some of the steps in this stage include: compiling, testing, running static analysis, checking for vulnerabilities in the our dependencies and storing the code artifacts

Continuous Deployment

- This is the process by which verified changes in codebase or system architecture are deployed to production as soon as they are ready and without human input. Some steps in this stage include: setting up infrastructure, provisioning servers, copying files, smoke testing, promoting to production and even rolling back a change if something did not look right

Benefits of CI/CD to our business

- We would have these benefits when we set up our CI/CD pipeline: - Automated Smoke Tests: This would protect our revenue by reducing downtime caused by deploy-related crash or bugs. - Catch Unit Test Failures: Having less bug in our live app and spending less time doing manual testing would help us to avoid cost. - Faster and More Frequent Production Deployment: We would get more revenue by shipping value generating features more frequently to the customers, this would also help us to get feedback early and

Benefits of CI/CD to our business contd.

◈- Detect Security Vulnerabilities: This would enable us to easily detect serious security flaws that would be embarrassing if it had made it to the public. This would save us money trying to win back the customers' trust and rebuilding our image. - Deploy to Production Without Manual Checks: Less time to market would help us to increase our revenue. I would leave with this: A penny saved is a penny earned