



Introducing Jazz

T-Mobile's Serverless Development Platform

github.com/tmobile/jazz

Satish Malireddi
Principal Cloud Architect, T-Mobile

About T-Mobile

- As America's Un-carrier, T-Mobile US, Inc. is redefining the way consumers and businesses buy wireless services through leading product and service innovation.
- NASDAQ traded public company – TMUS
- Operating two flagship brands: T-Mobile and MetroPCS
- Based in Bellevue, Washington



TECHNOLOGY



Jazz Serverless



Conductor



T-Vault



PacMan



Cloud Insights



Next Directory
(formerly Hyper Directory)



Adaptive
Security

PEOPLE



Recognition



Training



Tech.
Community



Open
Source

PROCESS



Compliance



NoOps



Self-Service



Federated
Admin



Serverless

...

- Serverless computing allows you to build and run applications and services without thinking about servers
 - No server management
 - Scales on demand
 - High availability
 - Don't pay for idle capacity



Jazz...

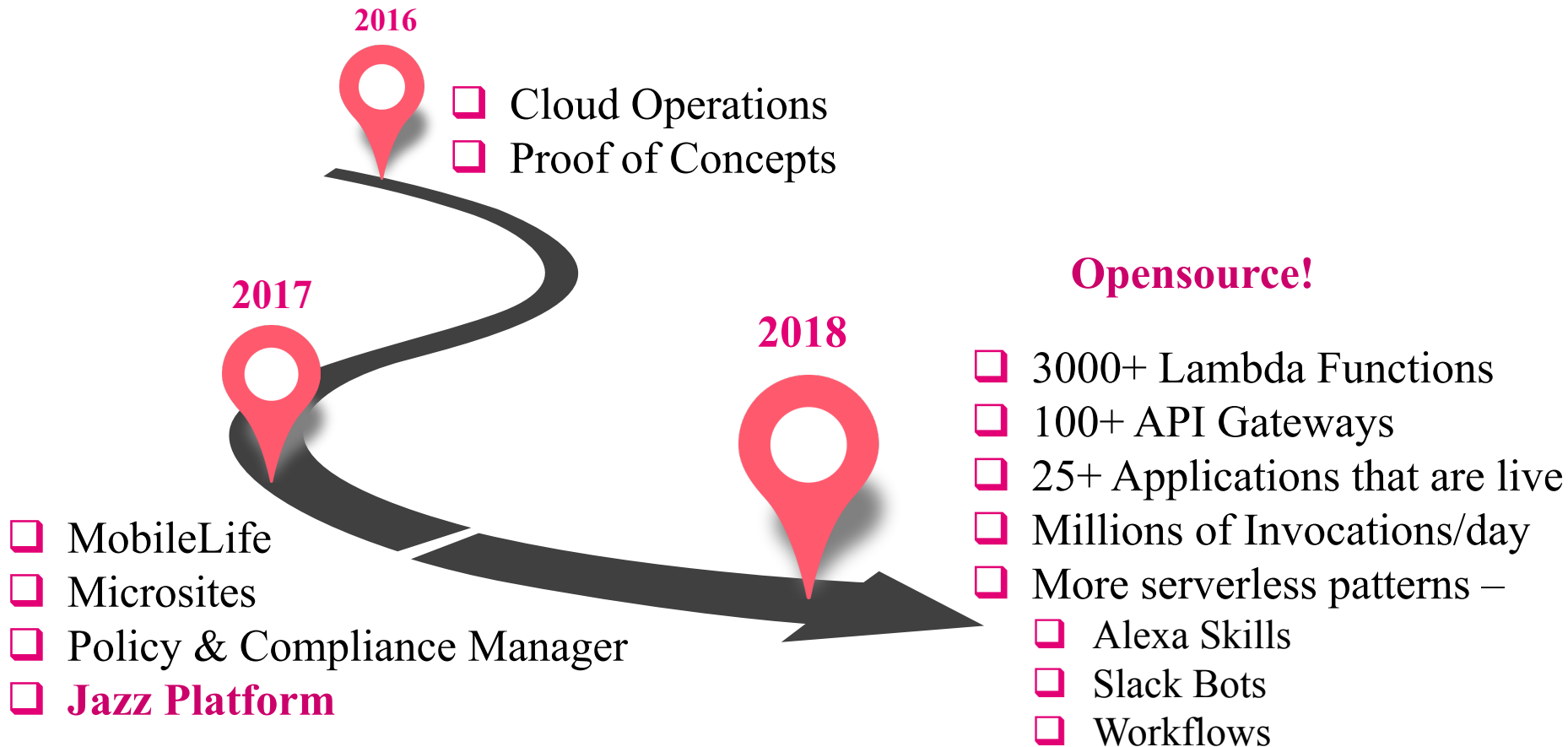
- A Serverless Development Platform
- Jazz provides developers a fast on-ramp to build & manage their serverless applications
 - It is not another FaaS implementation. Rather, it enhances the usability of existing FaaS systems
 - Jazz has well-built interfaces designed to let developers quickly self-start and focus on code

Serverless Usecases at T-Mobile...

- ❑ APIs/Microservices
- ❑ Static Websites/Single Page Applications
- ❑ Mobile Backends & Serverless DataStores
- ❑ Timer-based processing (Cron Jobs)
- ❑ Event-based processing
 - S3 Upload
 - Payment Failed
 - Monitoring Alert Triggered
- ❑ Realtime stream processing
 - IoT Events

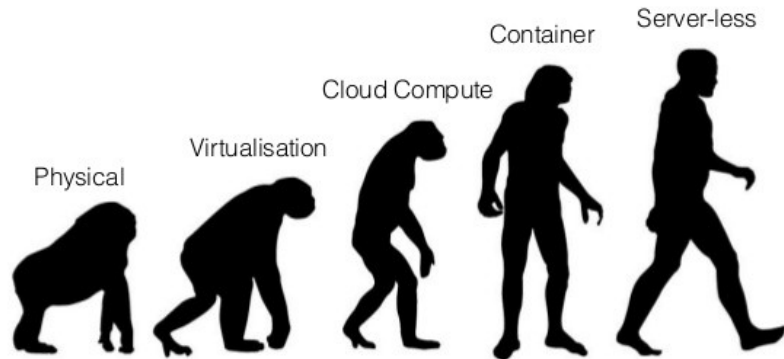


Our Serverless Journey...



Motivation

- ❖ Packaging & deploying (CI/CD)
- ❖ Multi-tenancy
- ❖ Local testing
- ❖ Logging & monitoring
- ❖ Service Sprawl
- ❖ Secret management
- ❖ Security, auditing & best practices
- ❖ Integration with enterprise services

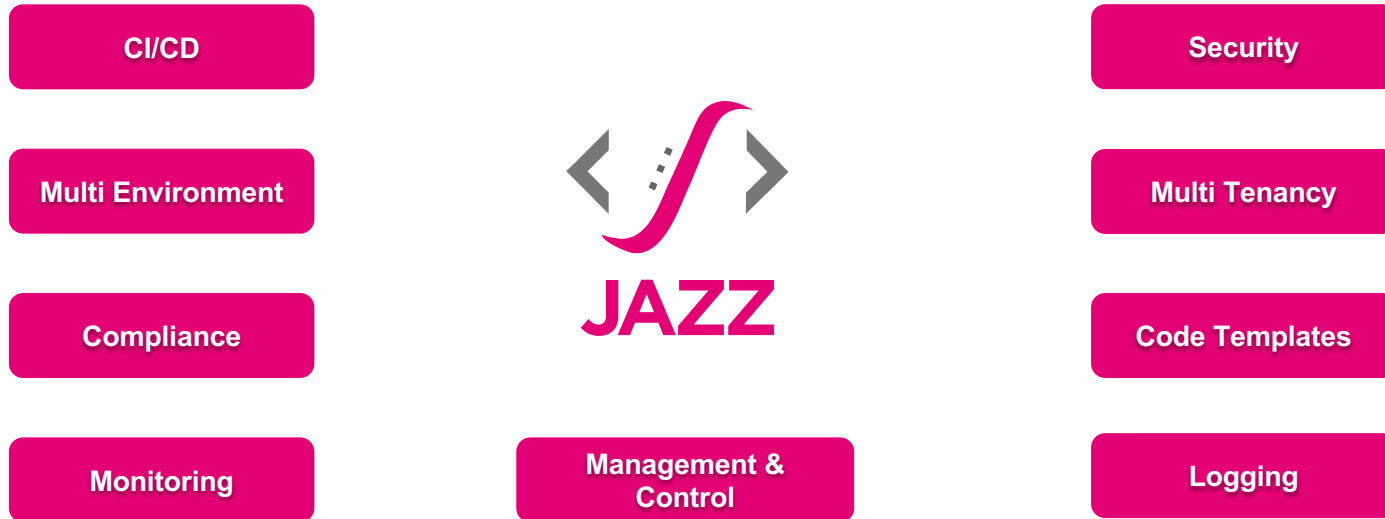


Now, how can we drive adoption within T-Mobile?

- ❑ Serverless is just like Cloud 6 years ago. We've seen that Cloud worked and we've realized that serverless works!
- ❑ FaaS is great, but can we build production ready applications & operate them at scale?
- ❑ Can we make it simpler for enterprise users to use serverless?
- ❑ How can we drive ADOPTION internally in T-Mobile?

Jazz – Our solution to drive serverless adoption

Focus on code, Jazz takes care of everything else!

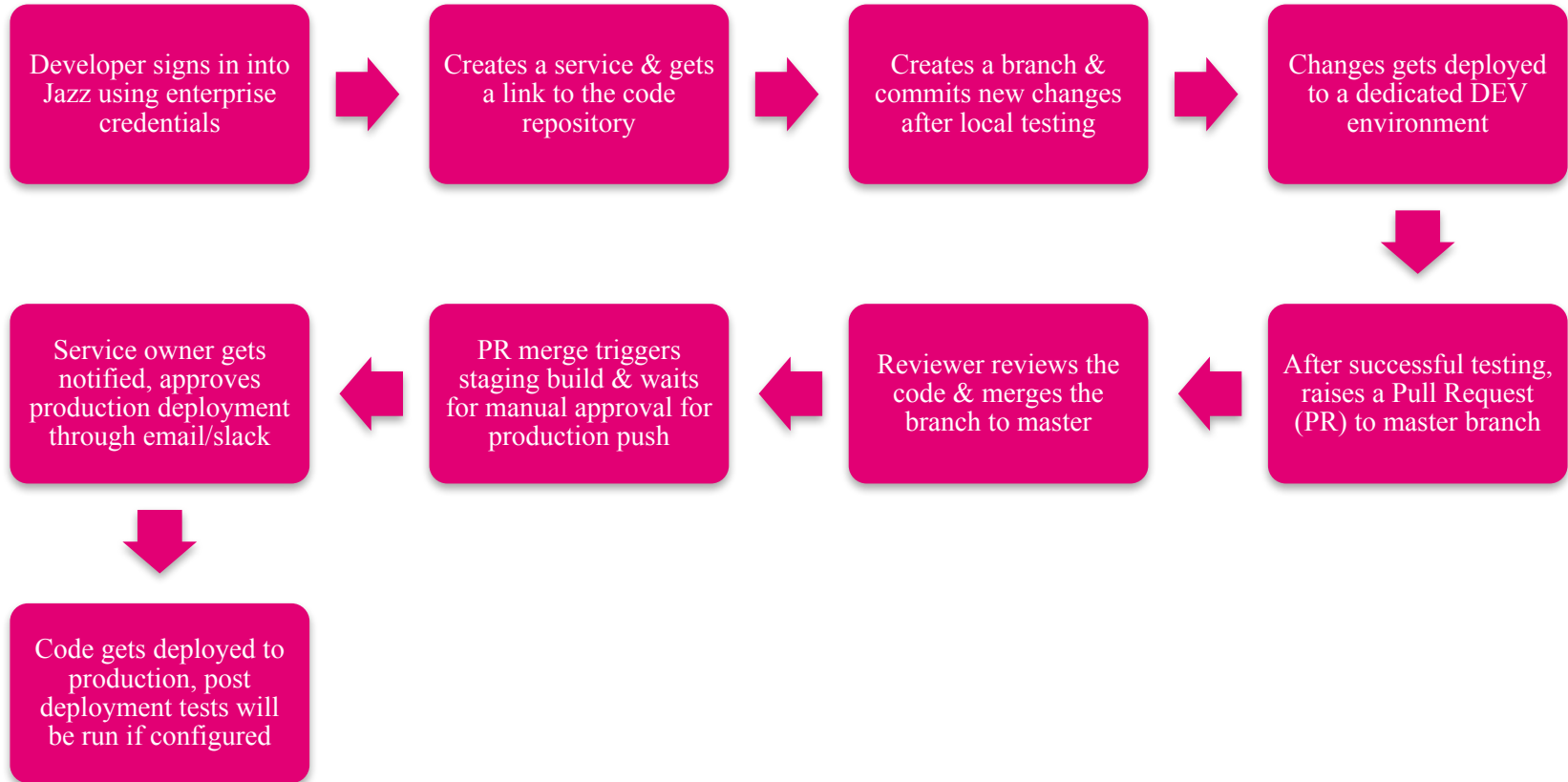


<https://github.com/tmobile/jazz>

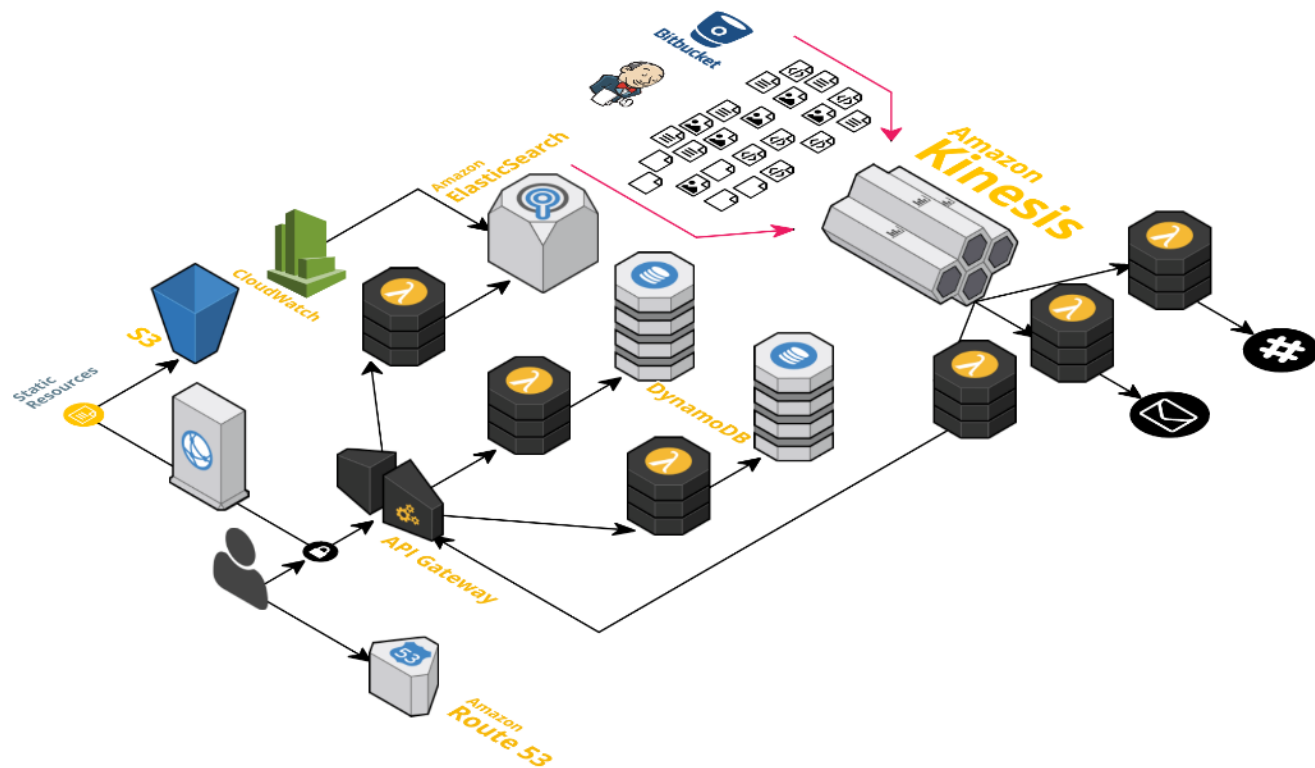
Jazz – Developer Experience



Jazz – Developer Journey



Jazz Architecture...



**Jazz core itself is
serverless—and written
in Jazz!**

Jazz Components...



Bitbucket/Gitlab:

- NodeJS/Python/Java/Go*
- Commit hooks for CI/CD
- Code/Swaggers



Jenkins:

- CI/CD
- Package & deploy
- Code & Dependency scan
- Test integration



Amazon ES



Kinesis



CloudFront



AWS KMS



S3



DynamoDB

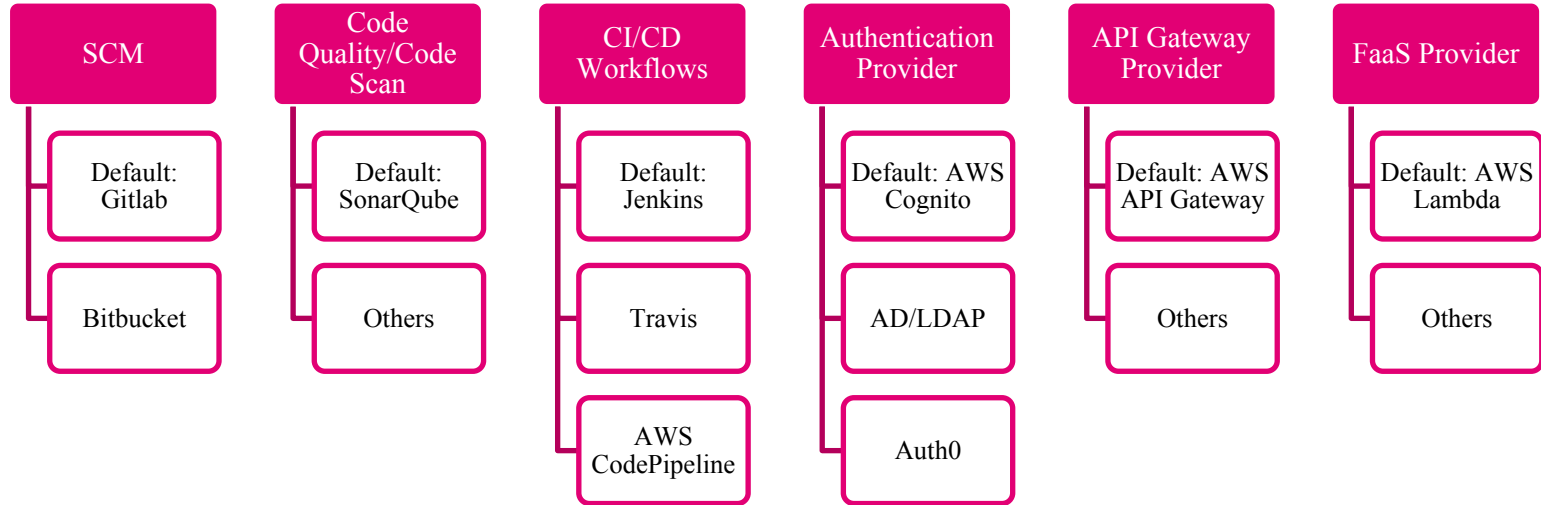


API Gateway



AWS Lambda

Jazz – Modules & Customization



Jazz – CI/CD

- Integration with Enterprise Delivery Pipeline & Branching Strategy
- Integration with Enterprise tools - Bitbucket/Jenkins/SonarQube/Artifactory*
- In-built compliance controls (SOX)
- Approval Workflow baked into the pipeline
- Static Code Scan & Vulnerability Scan before deployment
- Slack/Email Integration
- Dynamic Multi-Environment support for parallel development
- Instant Reporting on code quality metrics, deployments etc.
- Multi account/region deployments*

Jazz – Security

- Integration with Enterprise AD for centralized Access Control
- Single Sign-on
- ACL for Code/Service access via security groups
- Integration with OWASP Dependency Check before code gets deployed
- Single click provisioning for public vs private endpoints
- IP level access control for services
- AWS specific controls –
 - IAM roles - Default roles following **Least Privilege** principle
 - VPC access
 - VPC/Subnet/SG Customization
 - Custom Authorizers
- Secret Management
 - T-Vault/AWS Secrets Manager Integration*

How Jazz helped us with adoption within T-Mobile

- ❑ Serverless microservices are being created in minutes
- ❑ Applications are going live without teams managing a single server
- ❑ Active users exploring serverless increasing each month
- ❑ 3000+ Lambda Functions deployed & managed via the platform
- ❑ Code templates made it easy to share best practices
- ❑ CI/CD, Compliance & Security comes with platform
- ❑ No server patching; significant overall cost reductions

What's Next...

- ❖ **Serverless-First** approach for Cloud Native Applications
- ❖ **Opensource-First** development for Jazz & community engagement
- ❖ Support new serverless patterns, service types & runtimes
- ❖ ACL & Enhanced Security Controls
- ❖ API Versioning, Rollbacks, B/G & Canary deployments
- ❖ SAM & Serverless Application Repository Integration
- ❖ Local testing and debugging
- ❖ External plugin development
- ❖ What do you need?

<https://github.com/tmobile/jazz>



Helpful Stuff...



- **GitHub:** <https://github.com/tmobile/jazz>
- **OSS:** <https://opensource.t-mobile.com/>
- **Slack:** <https://tmo-oss-getinvite.herokuapp.com/>
- **Email:** serverless@t-mobile.com
- **Try Jazz:** <http://tiny.cc/tryjazz>
 - For **Reg. Code**, Slack us!



Q& A

T-Mobile®

THE UN-CARRIER®



Try it yourself...

<http://tiny.cc/tryjazz>

<http://try.tmo-jazz.net>

For **Reg. Code**, slack/email us!