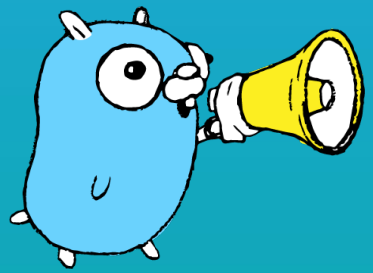
Taipei 2023/03/23

Microservices Scaffolding

Gaston Chiu

Crypto.com

Backend Engineer



Today’s (glorious) blather.

SECTION ONE

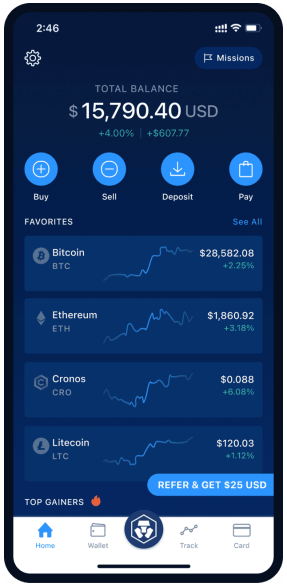
Background

crypto.com products

- Main App
- DeFi wallet
- NFT
- Exchange

Background for main app

- Ruby -> Go
- From Monolithic to Microservices



The challenge of Monolithic

- Slow development, testing, release
- Resilience
- Scalability
- Security

The challenge of microservice

- Cost of learning new programing language
- Duplication
- Quality

SECTION TWO

Solutions

Define the practice (reference-guide)

- Coding Style
- Technical Selection
- Tracing
- Testing
- Logging

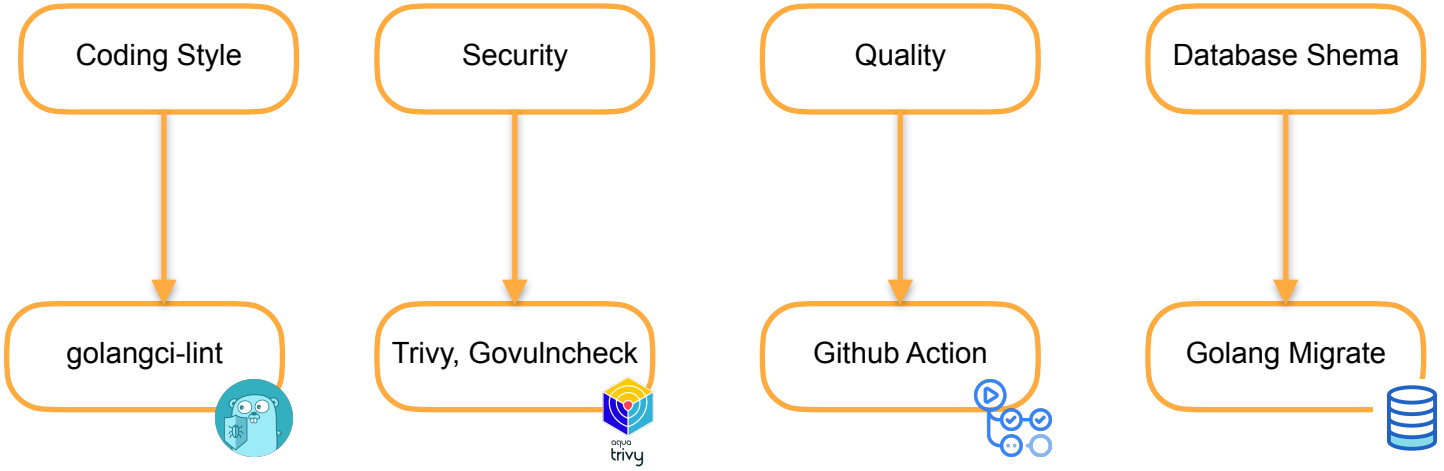
Share Library

- DB connection
- Configuration management
- Graceful shutdown
- Middlewares (contextlogger, requestId)
- Tracing

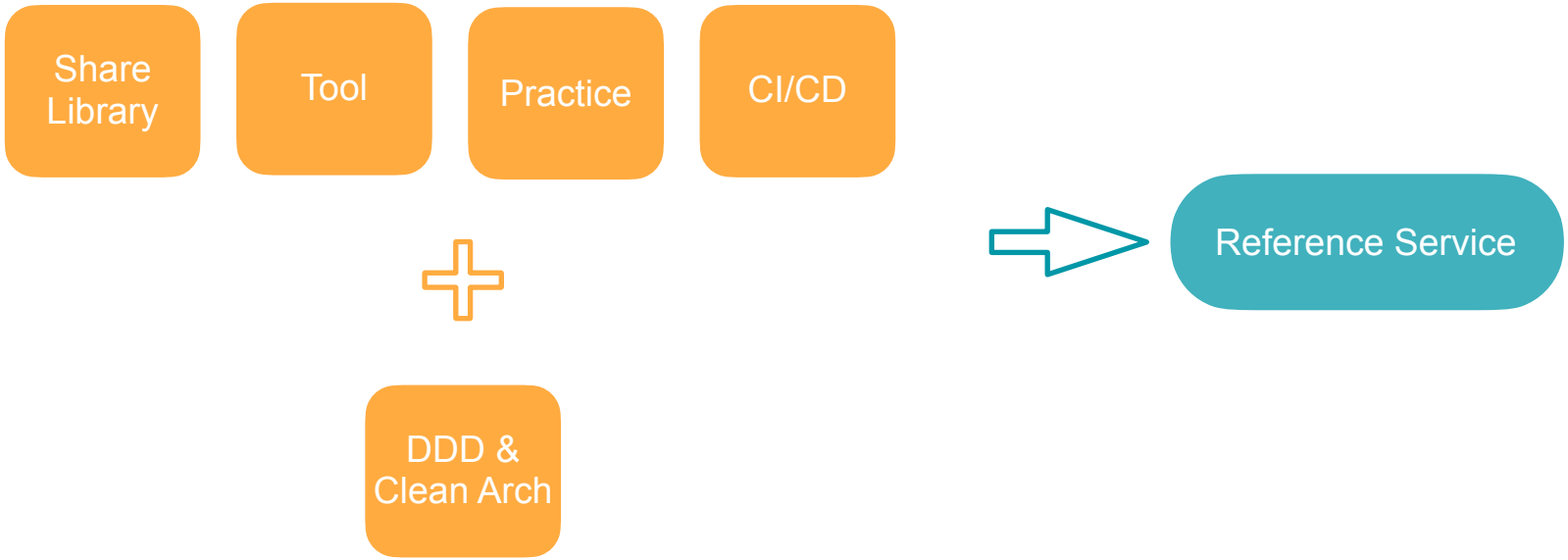
Open source microservice framework

- Less Customization
- Slow support

Tools


Coding Style
golangci-lint
Security
Trivy, Govulncheck
aqua
trivy
Quality
Github Action
Database Shema
Golang Migrate

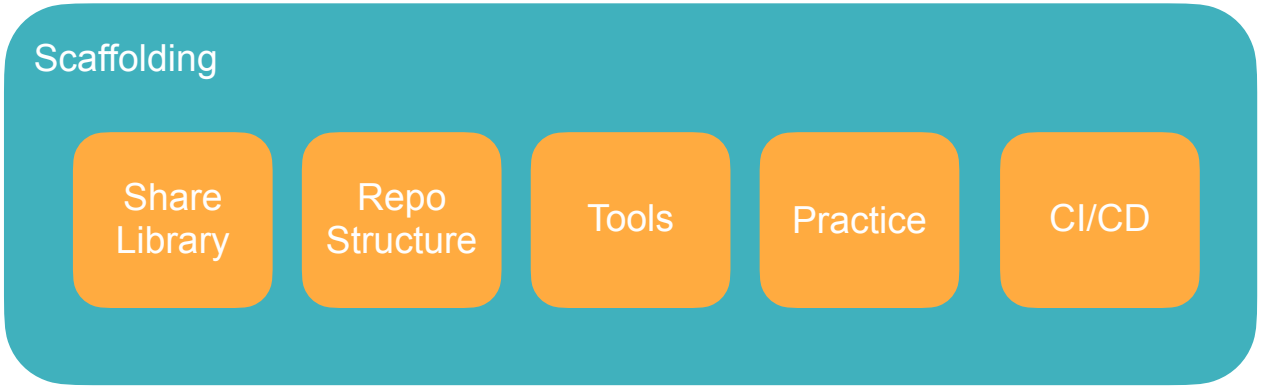
Reference Service



Apply reference service to different team

- Create: Copy and modify from reference service
- Update: Weekly sync meeting on new update and manually copy paste

Scaffolding CLI



Update scaffolding

- Download the latest version and run cli with update cmd

Scaffolding

Characteristic / Solution	Customization	Init new project	Update and apply new practice	Tool building cost
Template Repository				
Open Source Skeleton Generator				
Skeleton Generator				

SECTION THREE

Implementation

Create new project by text template

- placeholder for dynamic data
- for loop
- if else statement

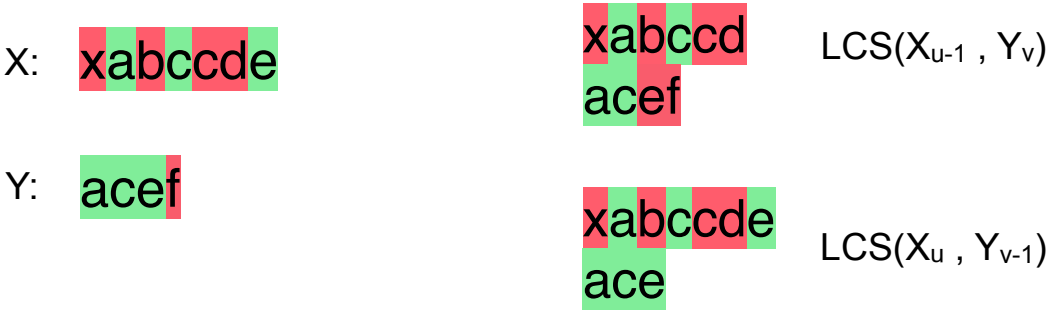
Testing generated result

- Create expected result and compare the generated result with expected one

Testing Challenge

- Find out difference between the generate result and expected result.

Build text differentiator (Longest common subsequence)



- $LCS(X_u , Y_v) = 1 + LCS(X_{u-1} , Y_{v-1})$ -> (for equal characters)
- $LCS(X_u , Y_v) = \max(LCS(X_u , Y_{v-1}), LCS(X_{u-1} , Y_v))$ -> (for unequal characters)

Construct DP table

		X	A	B	C	C	D	E
	0	0	0	0	0	0	0	0
A	0							
C	0							
E	0							
F	0							

$LCS(X_u , Y_v) = 1 + LCS(X_{u-1} , Y_{v-1})$ -> (for equal characters)

$LCS(X_u , Y_v) = \max(LCS(X_u , Y_{v-1}), LCS(X_{u-1} , Y_v))$ -> (for unequal characters)

Construct DP table

		X	A	B	C	C	D	E
	0	0	0	0	0	0	0	0
A	0	0						
C	0							
E	0							
F	0							

$LCS(X_u , Y_v) = 1 + LCS(X_{u-1} , Y_{v-1})$ -> (for equal characters)

$LCS(X_u , Y_v) = \max(LCS(X_u , Y_{v-1}), LCS(X_{u-1} , Y_v))$ -> (for unequal characters)

Construct DP table

		X	A	B	C	C	D	E
	0	0	0	0	0	0	0	0
A	0	0	1					
C	0							
E	0							
F	0							

$LCS(X_u , Y_v) = 1 + LCS(X_{u-1} , Y_{v-1})$ -> (for equal characters)

$LCS(X_u , Y_v) = \max(LCS(X_u , Y_{v-1}), LCS(X_{u-1} , Y_v))$ -> (for unequal characters)

Construct DP table

		X	A	B	C	C	D	E
	0	0	0	0	0	0	0	0
A	0	0	1	1				
C	0							
E	0							
F	0							

$LCS(X_u , Y_v) = 1 + LCS(X_{u-1} , Y_{v-1})$ -> (for equal characters)

$LCS(X_u , Y_v) = \max(LCS(X_u , Y_{v-1}), LCS(X_{u-1} , Y_v))$ -> (for unequal characters)

Construct DP table

		X	A	B	C	C	D	E
	0	0	0	0	0	0	0	0
A	0	0	1	1	1	1	1	1
C	0	0	1	1	2			
E	0							
F	0							

$LCS(X_u , Y_v) = 1 + LCS(X_{u-1} , Y_{v-1})$ -> (for equal characters)

$LCS(X_u , Y_v) = \max(LCS(X_u , Y_{v-1}), LCS(X_{u-1} , Y_v))$ -> (for unequal characters)

Construct DP table

		X	A	B	C	C	D	E
	0	0	0	0	0	0	0	0
A	0	0	1	1	1	1	1	1
C	0	0	1	1	2	2	2	2
E	0	0	1	1	2	2	2	3
F	0	0	1	1	2	2	2	3

$LCS(X_u , Y_v) = 1 + LCS(X_{u-1} , Y_{v-1})$ -> (for equal characters)

$LCS(X_u , Y_v) = \max(LCS(X_u , Y_{v-1}), LCS(X_{u-1} , Y_v))$ -> (for unequal characters)

Backtrace DP table

		X	A	B	C	C	D	E
	0	0	0	0	0	0	0	0
A	0	0	1	1	1	1	1	1
C	0	0	1	1	2	2	2	2
E	0	0	1	1	2	2	2	3
F	0	0	1	1	2	2	2	3

$LCS(X_u , Y_v) = 1 + LCS(X_{u-1} , Y_{v-1})$ -> (for equal characters)

$LCS(X_u , Y_v) = \max(LCS(X_u , Y_{v-1}), LCS(X_{u-1} , Y_v))$ -> (for unequal characters)

Backtrace DP table

		X	A	B	C	C	D	E
	0	0	0	0	0	0	0	0
A	0	0	1	1	1	1	1	1
C	0	0	1	1	2	2	2	2
E	0	0	1	1	2	2	2	3
F	0	0	1	1	2	2	2	3

$LCS(X_u , Y_v) = 1 + LCS(X_{u-1} , Y_{v-1})$ -> (for equal characters)

$LCS(X_u , Y_v) = \max(LCS(X_u , Y_{v-1}), LCS(X_{u-1} , Y_v))$ -> (for unequal characters)

Backtrace DP table

		X	A	B	C	C	D	E
	0	0	0	0	0	0	0	0
A	0	0	1	1	1	1	1	1
C	0	0	1	1	2	2	2	2
E	0	0	1	1	2	2	2	3
F	0	0	1	1	2	2	2	3

$LCS(X_u , Y_v) = 1 + LCS(X_{u-1} , Y_{v-1})$ -> (for equal characters)

$LCS(X_u , Y_v) = \max(LCS(X_u , Y_{v-1}), LCS(X_{u-1} , Y_v))$ -> (for unequal characters)

Backtrace DP table

		X	A	B	C	C	D	E
	0	0	0	0	0	0	0	0
A	0	0	1	1	1	1	1	1
C	0	0	1	1	2	2	2	2
E	0	0	1	1	2	2	2	3
F	0	0	1	1	2	2	2	3

$LCS(X_u , Y_v) = 1 + LCS(X_{u-1} , Y_{v-1})$ -> (for equal characters)

$LCS(X_u , Y_v) = \max(LCS(X_u , Y_{v-1}), LCS(X_{u-1} , Y_v))$ -> (for unequal characters)

Update Challenge

- Separate scaffolding code and application code
- Scaffolding code with DO NOT EDIT annotation

Update confirmation

- Show diff and ask for confirmation

Q&A

https://github.com/gastonqiu/PublicTalk/tree/main/golang_meetup_220323

https://www.linkedin.com/in/gaston-chiu-60824aa4/

Thank you