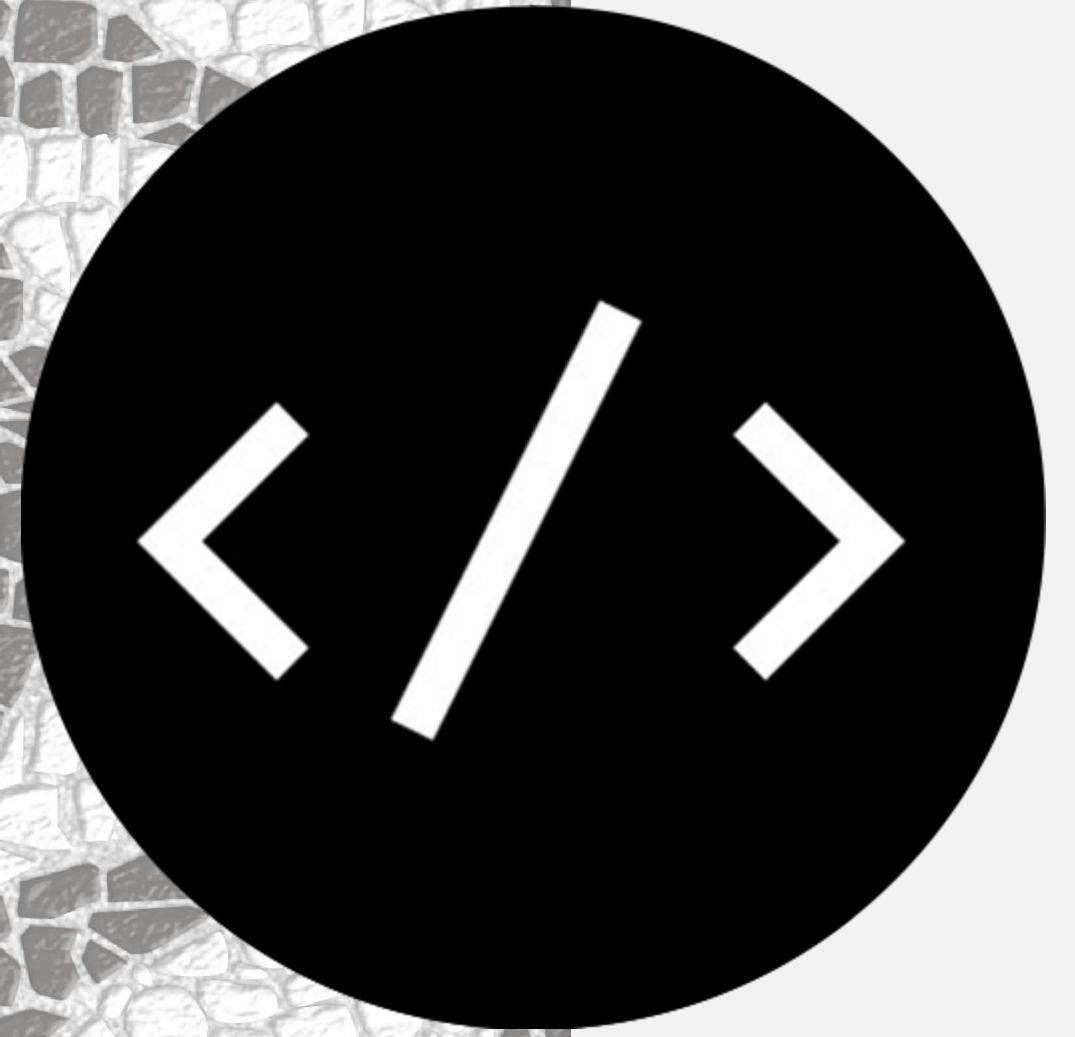


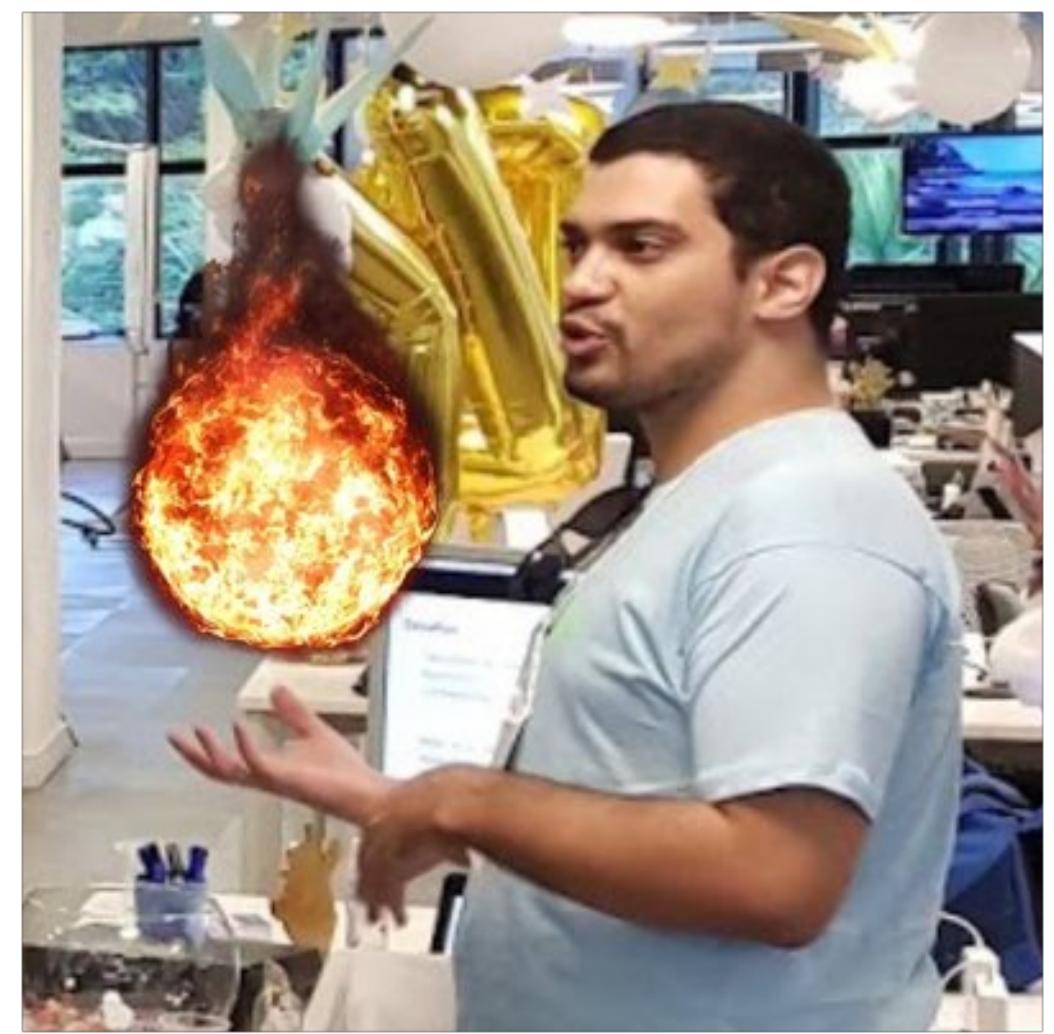
<coders> in </Rio>



Processando milhares de
arquivos para ajudar
comunidades de tradução
usando Go

Arthur Rodrigues Xavier da Silva

Gamer, amante do RPG, apaixonado por OSS e Goleiro nas horas vagas.



Titulos:

- Mantainer do Tsuru PaaS (tsuru.io)
- Estudante de Eng. Computação na UVA
- Pessoa que contribuiu para mais traduções do jogo Rimworld :D

Profissão:

- Estag de desenvolvimento no Feed @ globo.com



@arxdsilva



arxdsilva



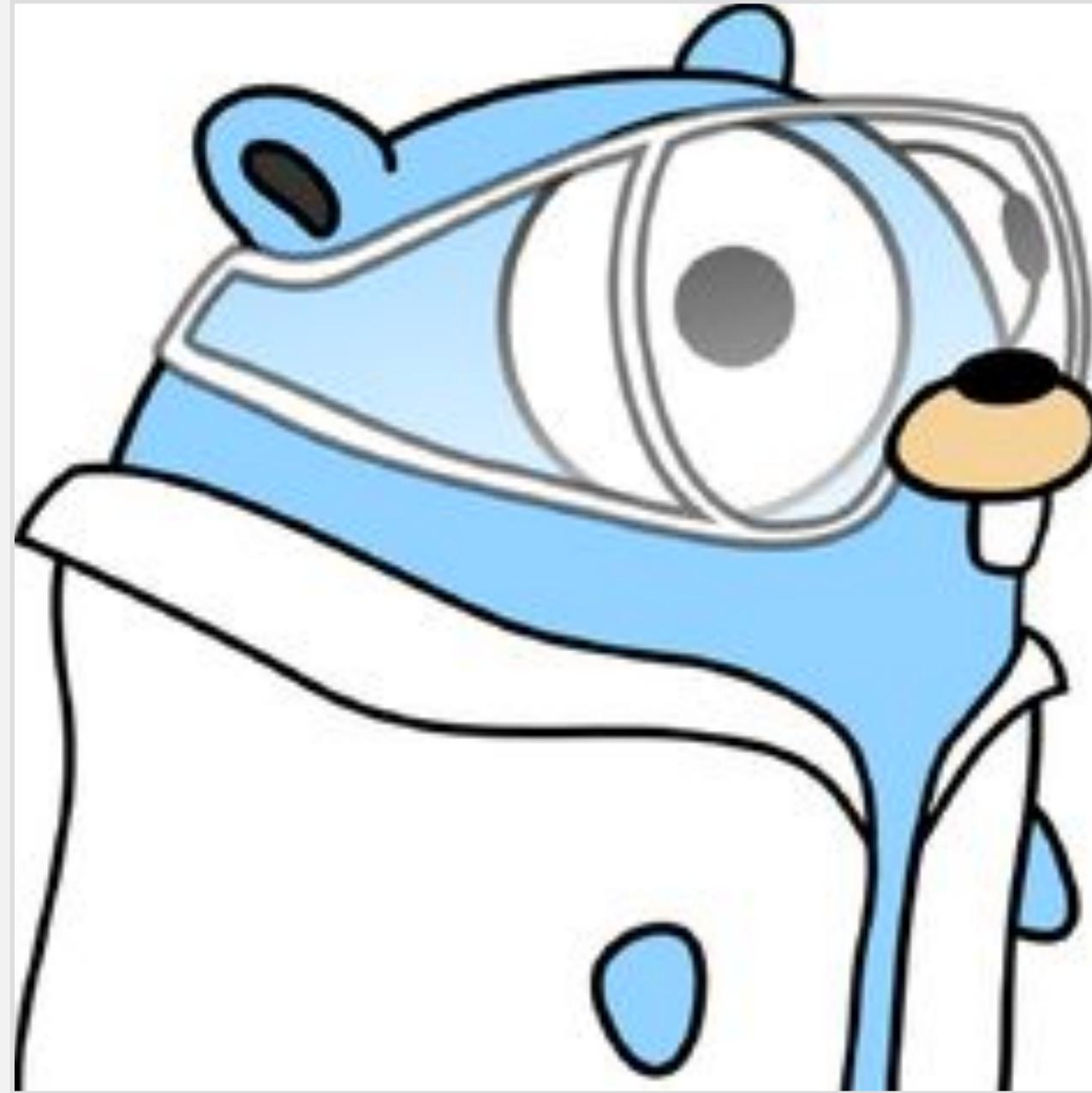
arxdsilva@gmail.com

<Coders>in </Rio>

Pauta

- Golang
- Rimworld
- OSS
- Process
- Tool
- How? Problems?
- Improvements
- Demo Gods please

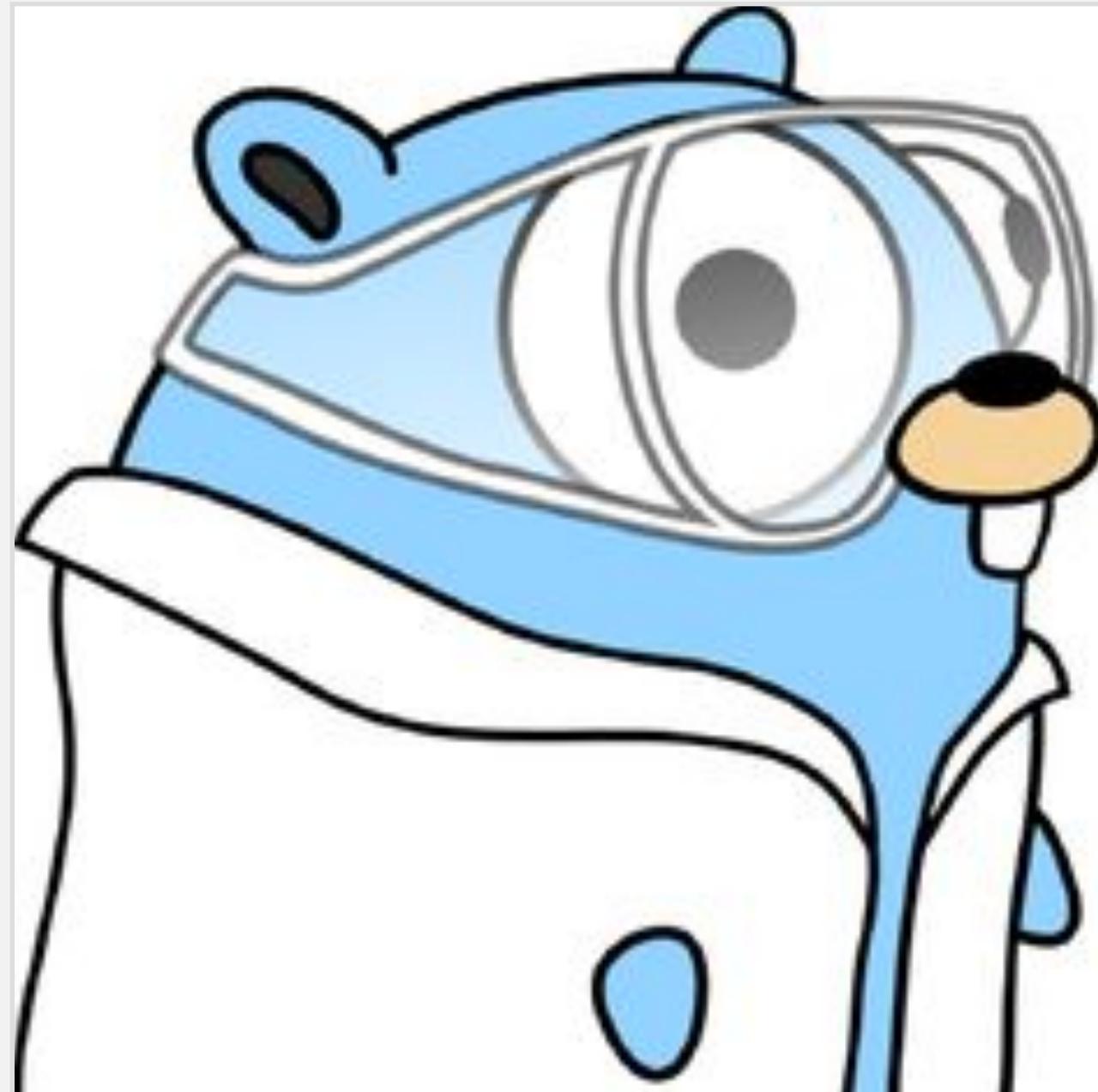




O que é Go?

- Linguagem criada pelo Google (2009)
 - Multiplataforma
 - Compilada
 - Fortemente tipada
 - Concisa, limpa e eficiente
-
- <https://golang.org/doc/faq>

Simulador de colônia ambientado em ficção científica 'dirigido' por uma IA



Por que Go?

- Open Source
- Legibilidade
- Cross compile fácil
- Resiliente
- Performance
- Concorrência
- Compila rápido
- Garbage collector
- Focado em multithread
- Tsuru <3

Simulador de colônia ambientado em ficção científica 'dirigido' por uma IA



Quem usa Go?

- Google
- globo.com <3 (Cartola, [tsuru](#), feed, ...)
- [Twitch.tv](#)
- [Twitter](#)
- [Dropbox](#)
- [Docker](#)
- [Uber](#)

Simulador de colônia ambientado em ficção científica 'dirigido' por uma IA

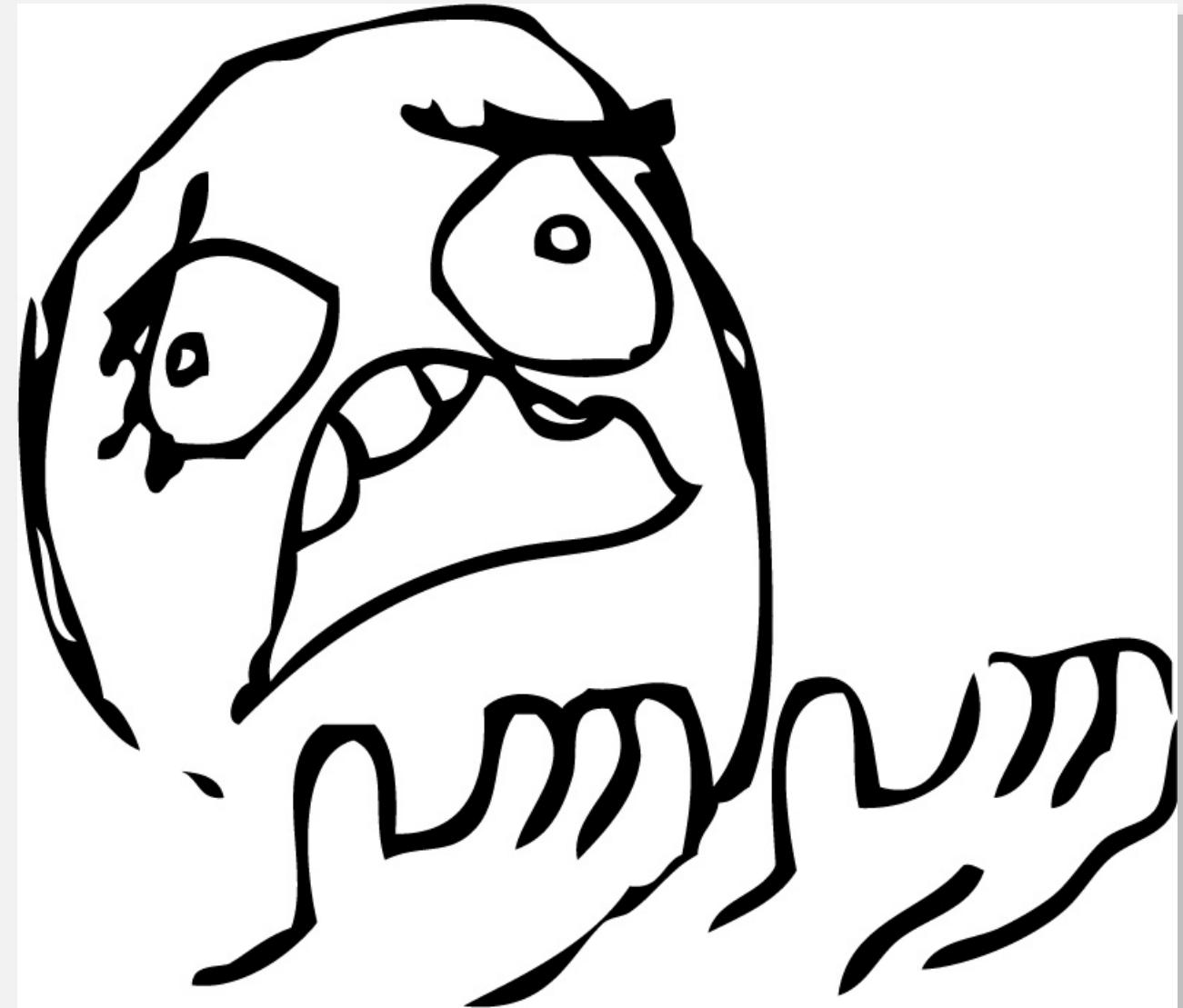
Rimworld



Simulador de colônia ambientado em ficção científica 'dirigido' por uma IA

Open Source Software

<\>



WHY?

Contribuidores, ater-se a legibilidade, menos complexidade. Existe um produto por trás?



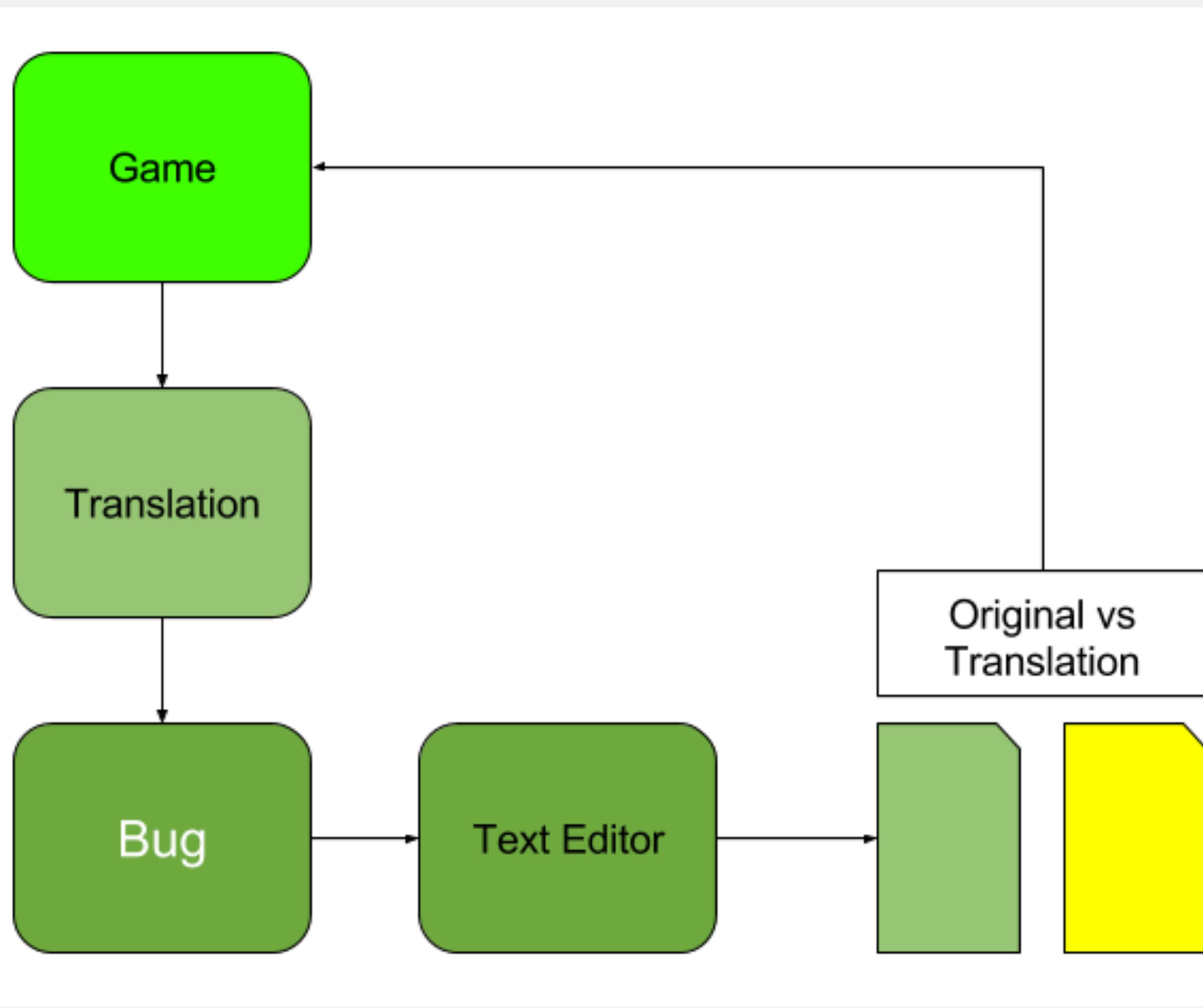
WHY?

- Pre-Alpha Game
- 440+ tradutores
- Mudanças constantes
- Código ‘legado’ (tags inexistentes)
- Alto tempo para tradução
- Tradução dependente de comparações constantes
- Baixo conhecimento tecnológico dos tradutores

Process



- Github
- 2 text editors
- line to line comparison



Tool



XML-Comp / XML-Comp

[View Repository](#) [Unwatch](#) 1 [Star](#) 7 [Fork](#) 6

[Code](#) [Issues 9](#) [Pull requests 1](#) [Projects 1](#) [Wiki](#) [Settings](#) [Insights](#)

Compare ANY markup documents. [Edit](#)

[xml](#) [rimworld](#) [xml-comparer](#) [translation](#) [go](#) [golang](#) [compare](#) [file](#) [Manage topics](#)

112 commits 3 branches 15 releases 8 contributors MIT

Branch: master [New pull request](#) [Create new file](#) [Upload files](#) [Find file](#) [Clone or download](#)

File	Message	Date
	Arthur Silva committed with ArxdSilva bump to 0.4	Latest commit c3c502f on Sep 3
	comparer main: enable 1:N comparison	16 days ago
	.gitignore comparer: add test coverage, fix bug	8 months ago
	.travis.yml travis: fixes coveralls	3 months ago
	LICENSE LICENSE & README.md update	11 months ago
	README.md docs: add how to use comparer package	7 months ago
	main.go bump to 0.4	16 days ago

<Coders>in </Rio>

Tool



A screenshot of the GitHub repository page for XML-Comp. The repository has 102 commits, 1 branch, 12 releases, and 8 contributors. The description states: "Helps you compare ANY markup documents". The repository is built with Go and includes links for XML, XML-to-JSON conversion, XML-to-CSV, and XML-to-YAML. The commit history shows recent updates to the comparer package and Travis CI configuration.

XML-Comp (github.com/xml-comp/xml-comp)

OSS, 100% Go, CLI, Lib;
Born: xml-to-xml comparer;
Now: anything comparer;
Comparer: N:N relation



XML-Comp

Baixo conhecimento tecnológico

Alta eficiência

Rápida

Confiável

XML

How?



PoC: XML-Comp (2 horas)

Inputs: pasta Origem (EN) e Tradução (~)

Entrar na pasta

Abrir arquivos

Guardar tag's das linhas

Comparar Origem-Tradução

Gerar diff's

How?

</>



PoC: XML-Comp (2 horas)

Interface c/
usuário

Navegação no
projeto

Leitura

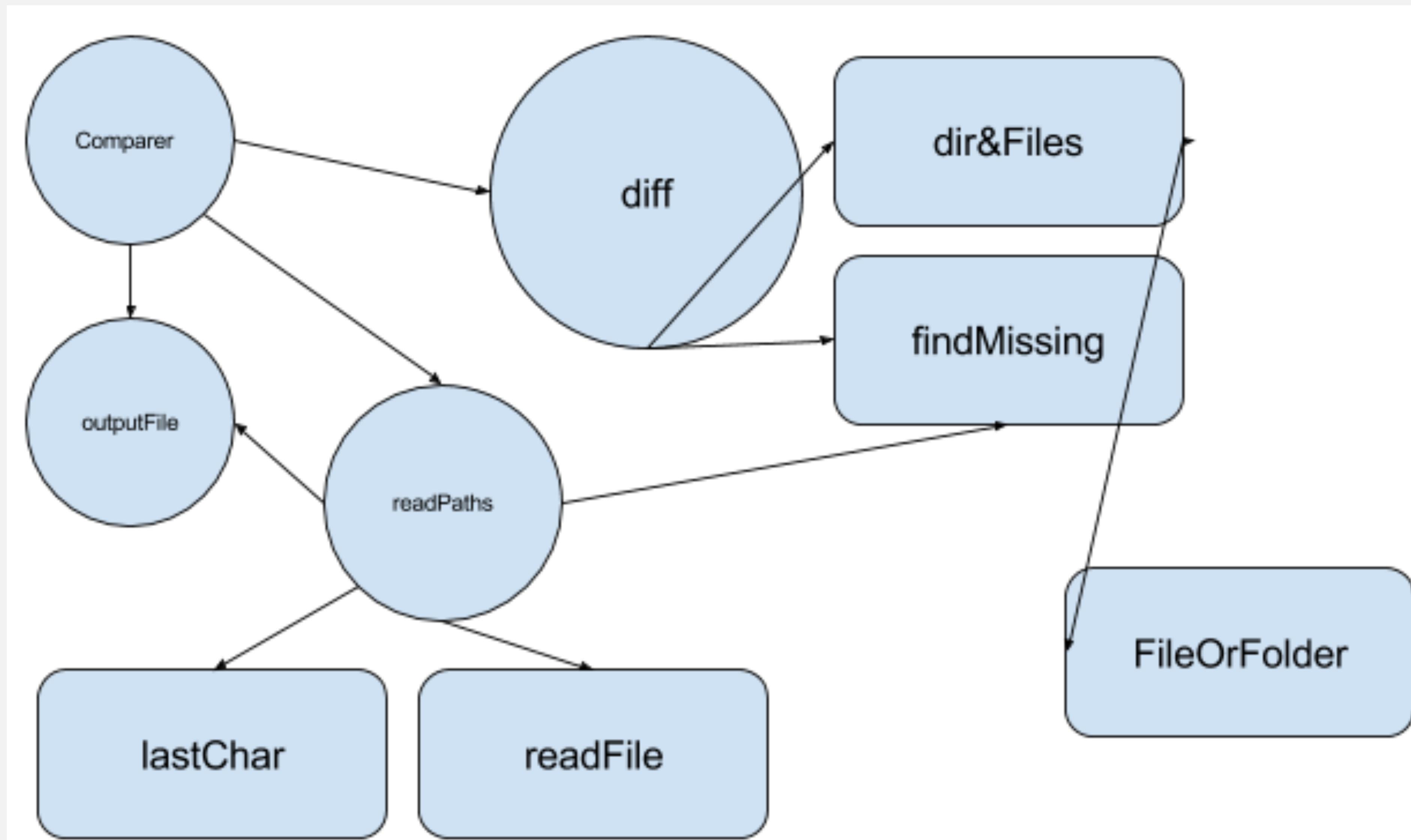
Comparação

Criação de
arquivos

Criação de
diretórios

How? Problems

</>



How? Problems

<\>



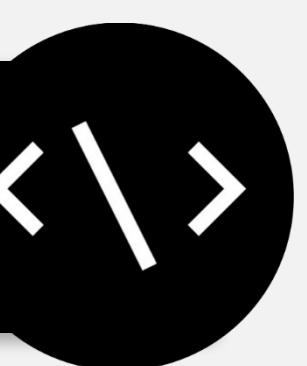


PoC: XML-Comp (2 horas)

Comparava 1 pasta:1 pasta

Impossível de escalar (comparar N:N)

Improvements



tsuru ✓



Apps Services Hosts Deploys Dashboard Events

Apps
966

Units
2,131

Nodes
251

Units per node
8

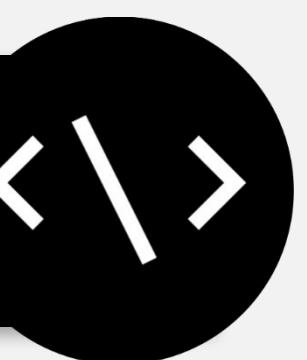
Healings (last 24h)

0

Deploys with error
(last 24h)

21

Improvements



[tsuru / tsuru](#)

[Watch ▾](#) 204 [Unstar](#) 2,319 [Fork](#) 304

[Code](#) [Issues 174](#) [Pull requests 5](#) [Projects 0](#) [Pulse](#) [Graphs](#) [Settings](#)

Open source, extensible and Docker-based Platform as a Service (PaaS). <https://tsuru.io> [Edit](#)

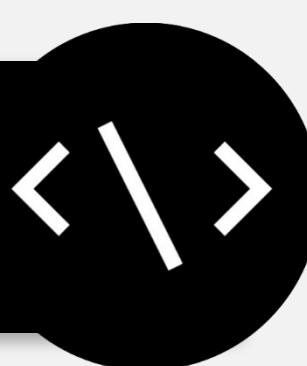
[paas](#) [go](#) [docker](#) [containers](#) [devops](#) [swarm](#) [kubernetes](#) [Manage topics](#)

12,787 commits 2 branches 113 releases 81 contributors BSD-3-Clause

Branch: master ▾ [New pull request](#) [Create new file](#) [Upload files](#) [Find file](#) [Clone or download ▾](#)

ggarnier integration: update deprecated client commands Latest commit 0447756 4 days ago

Improvements

[Code](#)[Issues 174](#)[Pull requests 4](#)[Projects 0](#)[Pulse](#)[Graphs](#)[Settings](#)

tsuru-client: add support to ignore files in app-deploy #1512

Closed

ggarnier opened this issue on Dec 8, 2016 · 4 comments



ggarnier commented on Dec 8, 2016

Owner



If you run `tsuru app-deploy .`, it tries to deploy the whole directory, including `.git`, which may be huge and probably unwanted.

It would be nice if `tsuru app-deploy` command supported a `.gitignore`-style file (maybe `.tsuruignore`?). Besides `.git`, it could list test files, documentation and local environment setup files, making the app image smaller.



3

Assignees

No one—ass

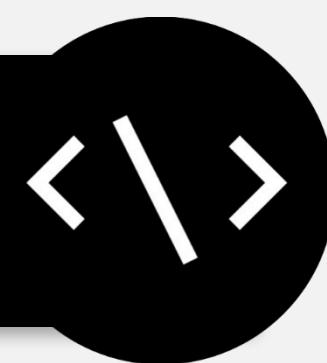
Labels

brainstorm

client

Projects

Improvements



.tsuruignore

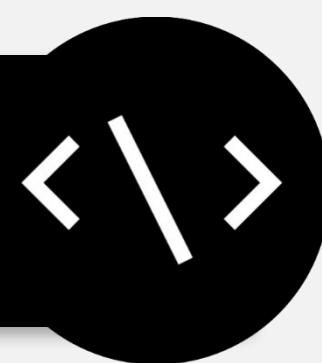
Deploy:

Ignorar arquivos/pastas

Menor tamanho de imagem da app

(-) Disco

Improvements

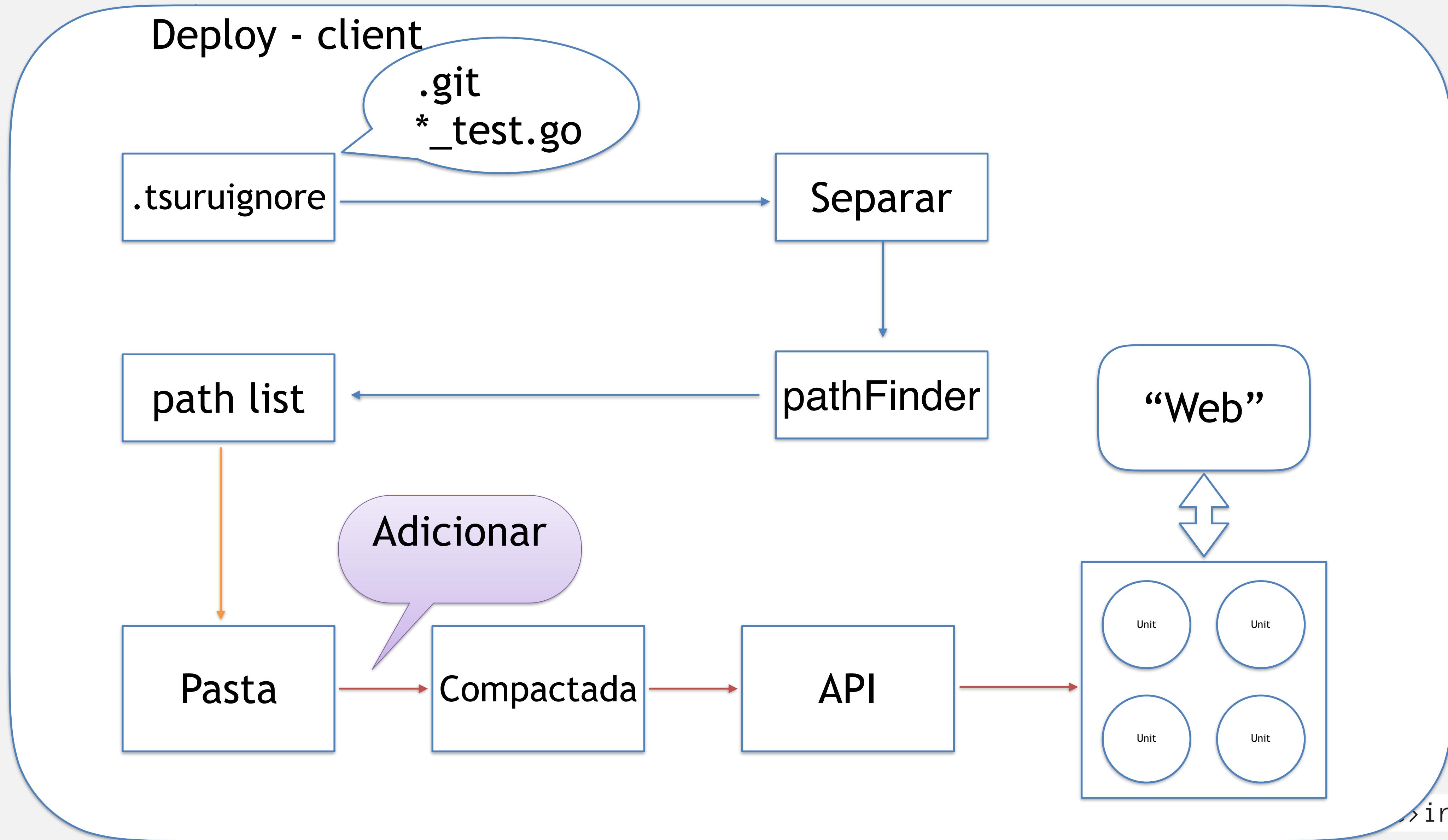


.tsuruignore processos

- > Ler padrões
- > Gerar padrões a serem ignorados
- > Leitura do repositório
- > Guardar locais de match com padrões
- > Deploy + inteligente

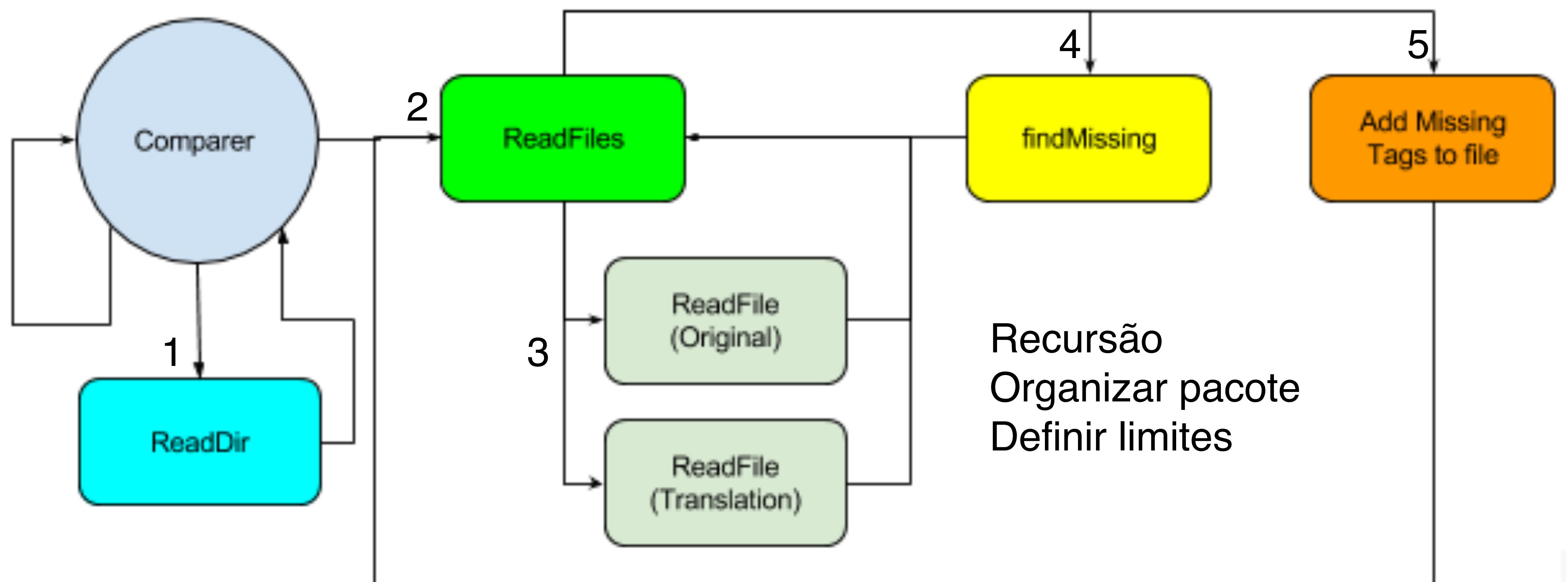
Improvements

</>

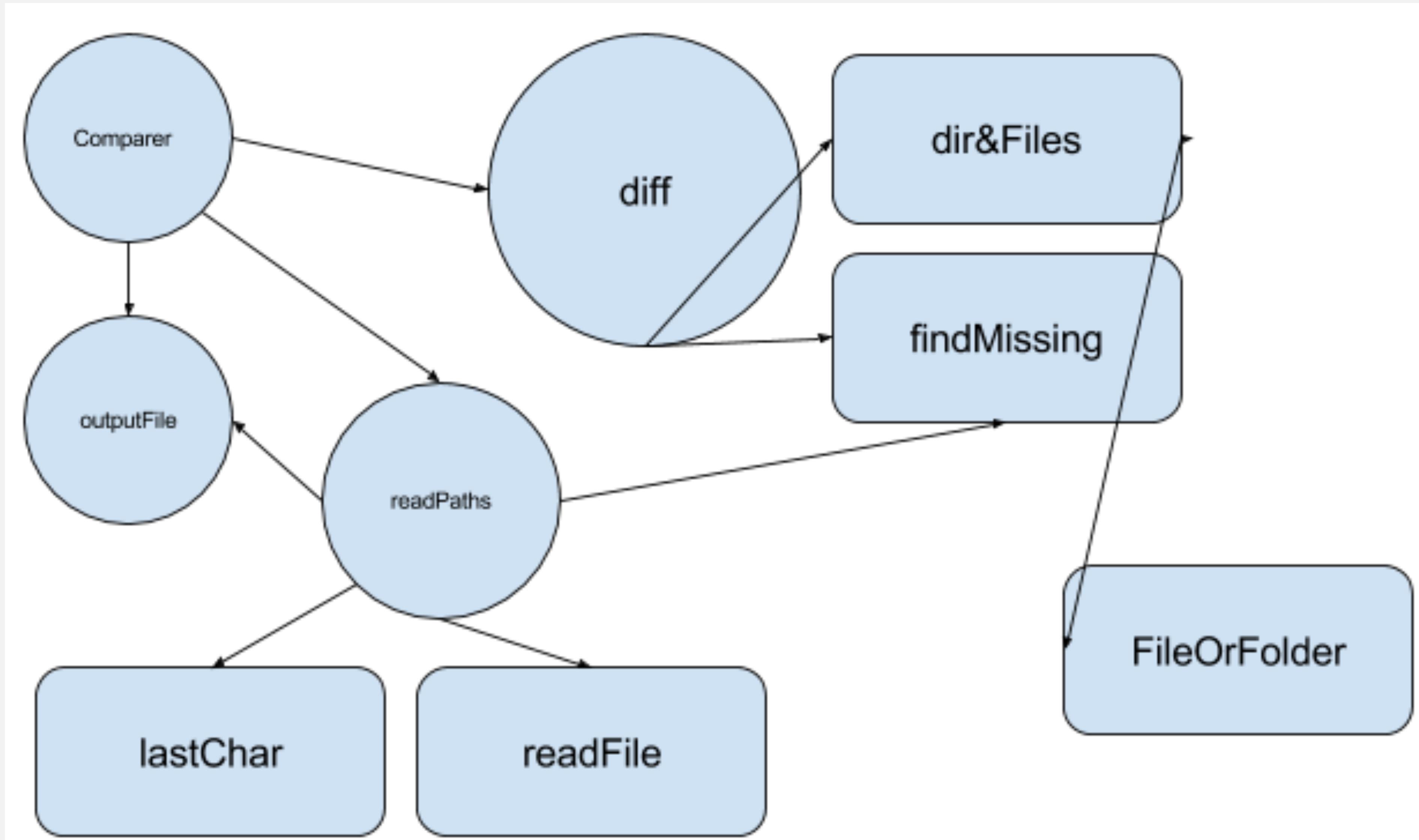
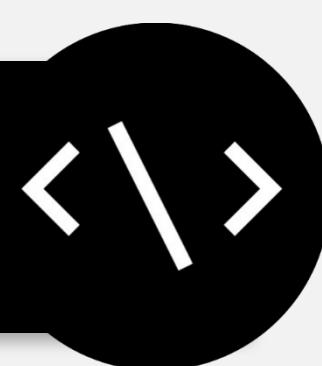


Improvements

<\>



Improvements



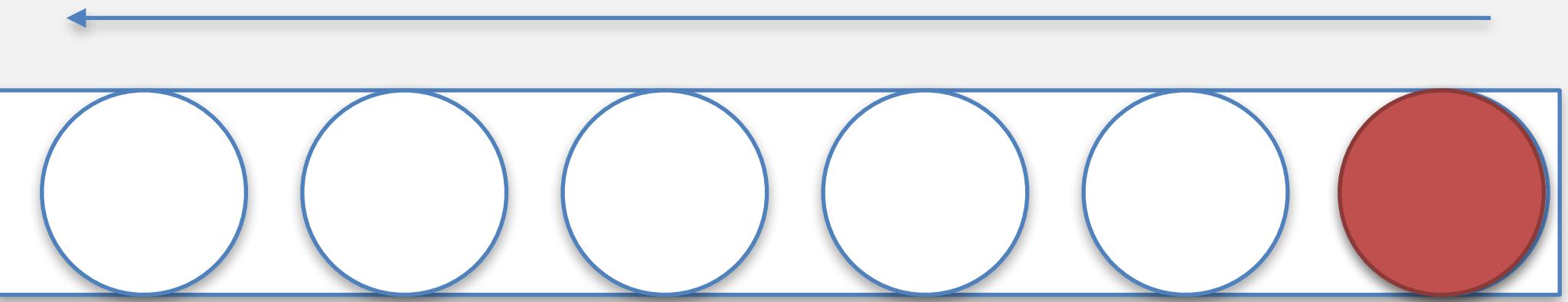
Code: Slice vs Maps



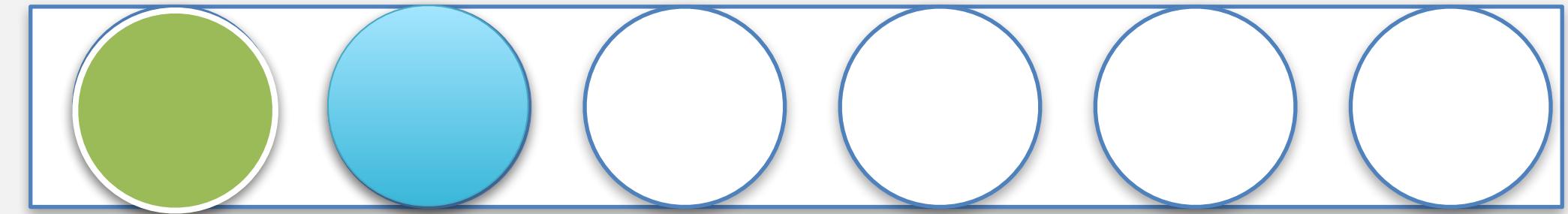
```
func findMissing(fileFolderA, fileFolderB []string) []string {
    sort.Strings(fileFolderA)
    sort.Strings(fileFolderB)
    if reflect.DeepEqual(fileFolderA, fileFolderB) {
        return nil
    }
    for i := len(fileFolderA) - 1; i >= 0; i-- {
        for _, vD := range fileFolderB {
            if fileFolderA[i] == vD {
                fileFolderA = append(fileFolderA[:i], fileFolderA[i+1:]...)
                break
            }
        }
    }
    return fileFolderA
}
```

Code: Slice vs Maps

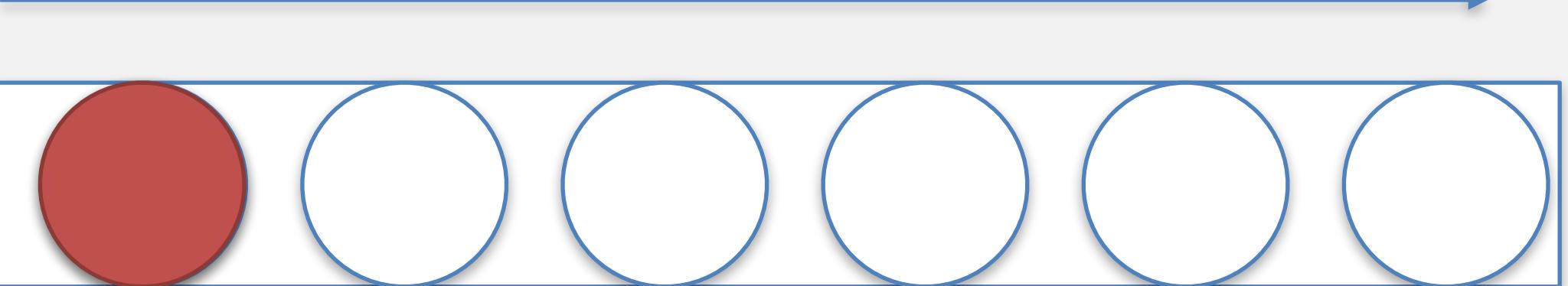
</>



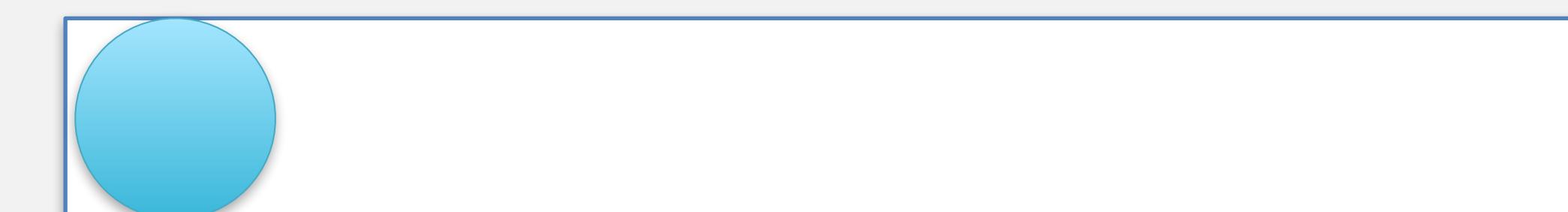
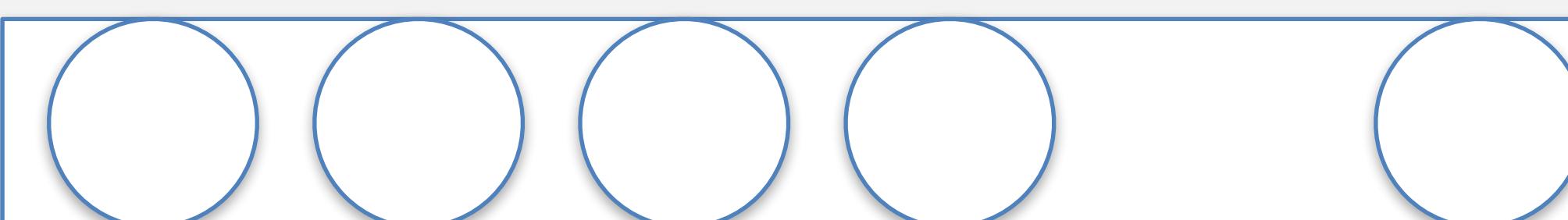
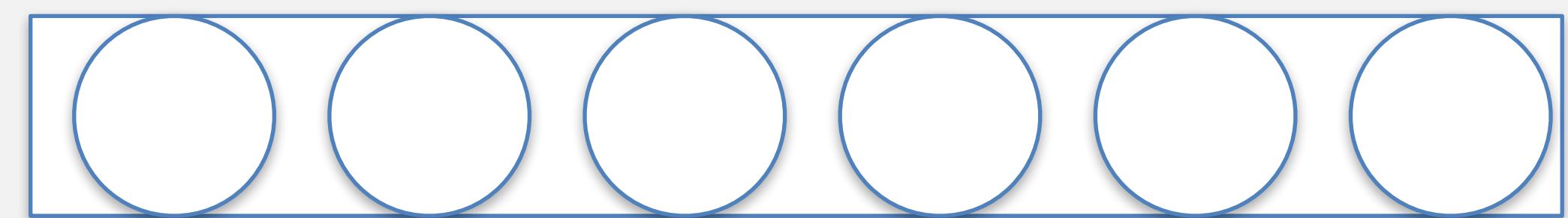
Match!



Match?



Missing



Code: Slice vs Maps



```
func findMissing(original, translation map[string]string) map[string]string {
    missing := make(map[string]string)
    if reflect.DeepEqual(original, translation) {
        return nil
    }
    for k, v := range original {
        if _, ok := translation[k]; !ok {
            missing[k] = v
        }
    }
    return missing
}
```



```
→ keynotes git:(master) ✘ xml-comp -original /Users/arthur/Library/Application\ Support/Steam/steamapps/common/RimWorld/RimWorldMac.app/Mods/Core/Languages/English -translation /Users/arthur/work/src/github.com/Ludeon -multiple
```

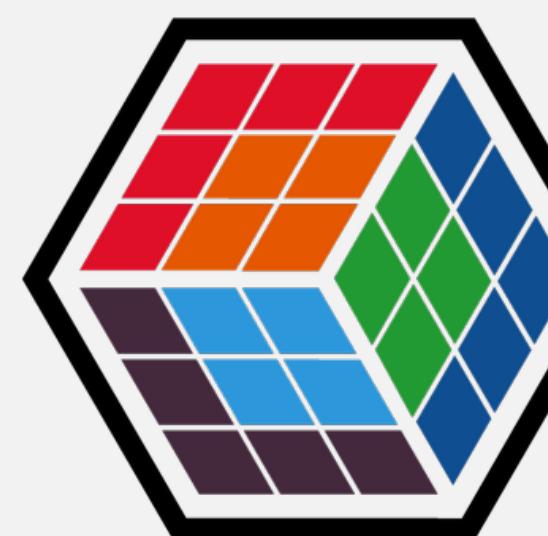
Referências

- N:N : <https://github.com/XML-Comp/XML-Comp/pull/49>
- .tsuruignore : <https://github.com/tsuru/tsuru-client/pull/25>
- Recursão: <https://github.com/XML-Comp/XML-Comp/pull/36>
- Map: <https://github.com/XML-Comp/XML-Comp/pull/44>



Agradecimentos

<Coders>in </Rio>





Obrigado!

