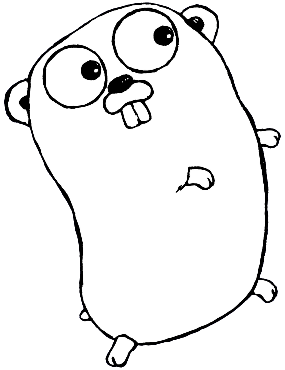
Modules and Authentication



Katie Hockman, Google
Go Open Source
github.com/katiehockman
@katie_hockman

github.com/katiehockman/puppies

1. Non-reproducible builds
2. Disappearing source code
3. Dangerous downloads

Some things we might consider

- Don’t have any dependencies

- Vendoring

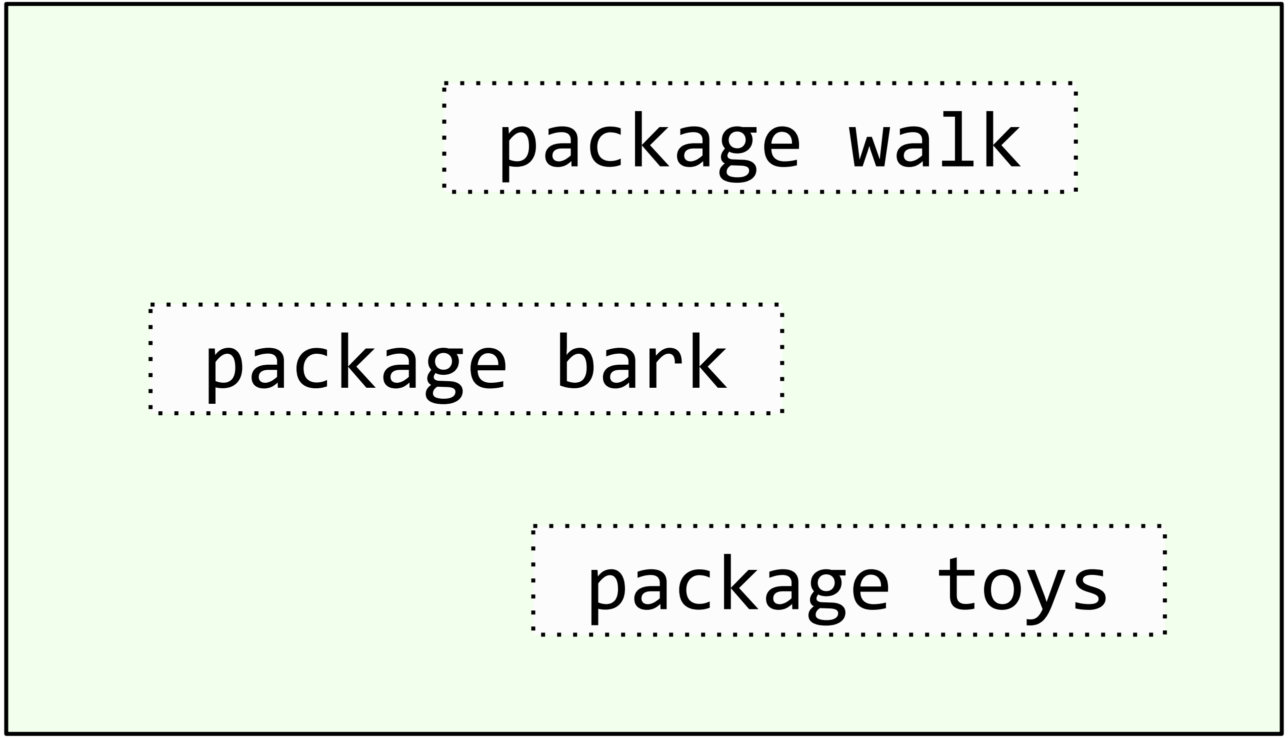
- Do nothing

Better solutions

1. Reproducible builds	Modules
2. Persistent dependencies	Mirror
3. Trustworthy fetches	Checksum Database

1. Modules

github.com/katiehockman/puppies

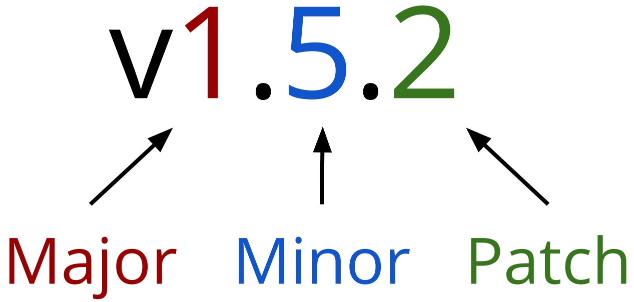


package walk

package bark

package toys

Semantic versioning



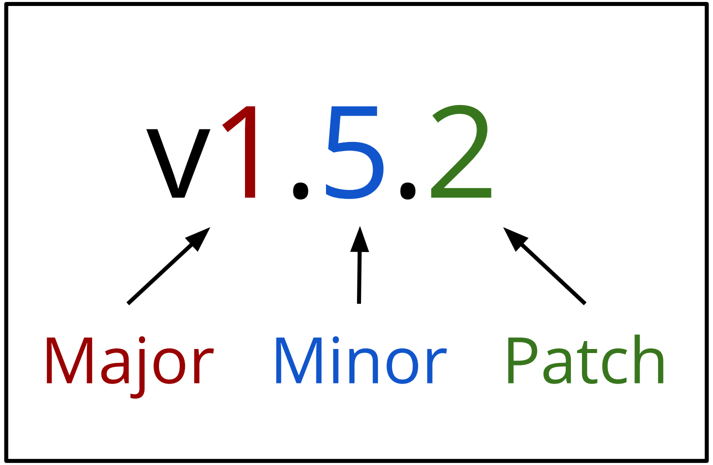
v1.5.2

Major Minor Patch

Semantic versioning

Immutable!

Backwards compatible!



v1.5.2

Major Minor Patch

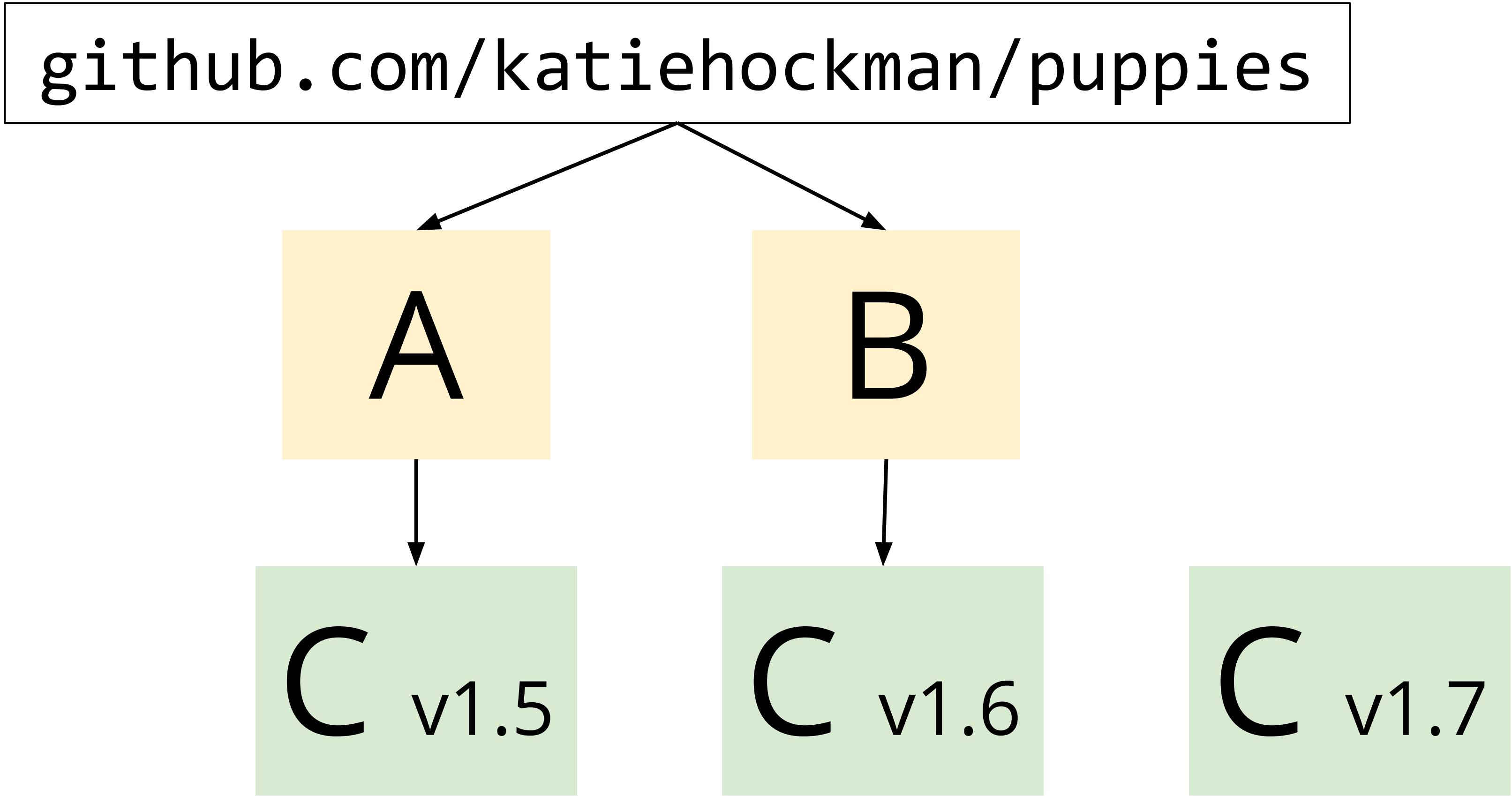
go.mod

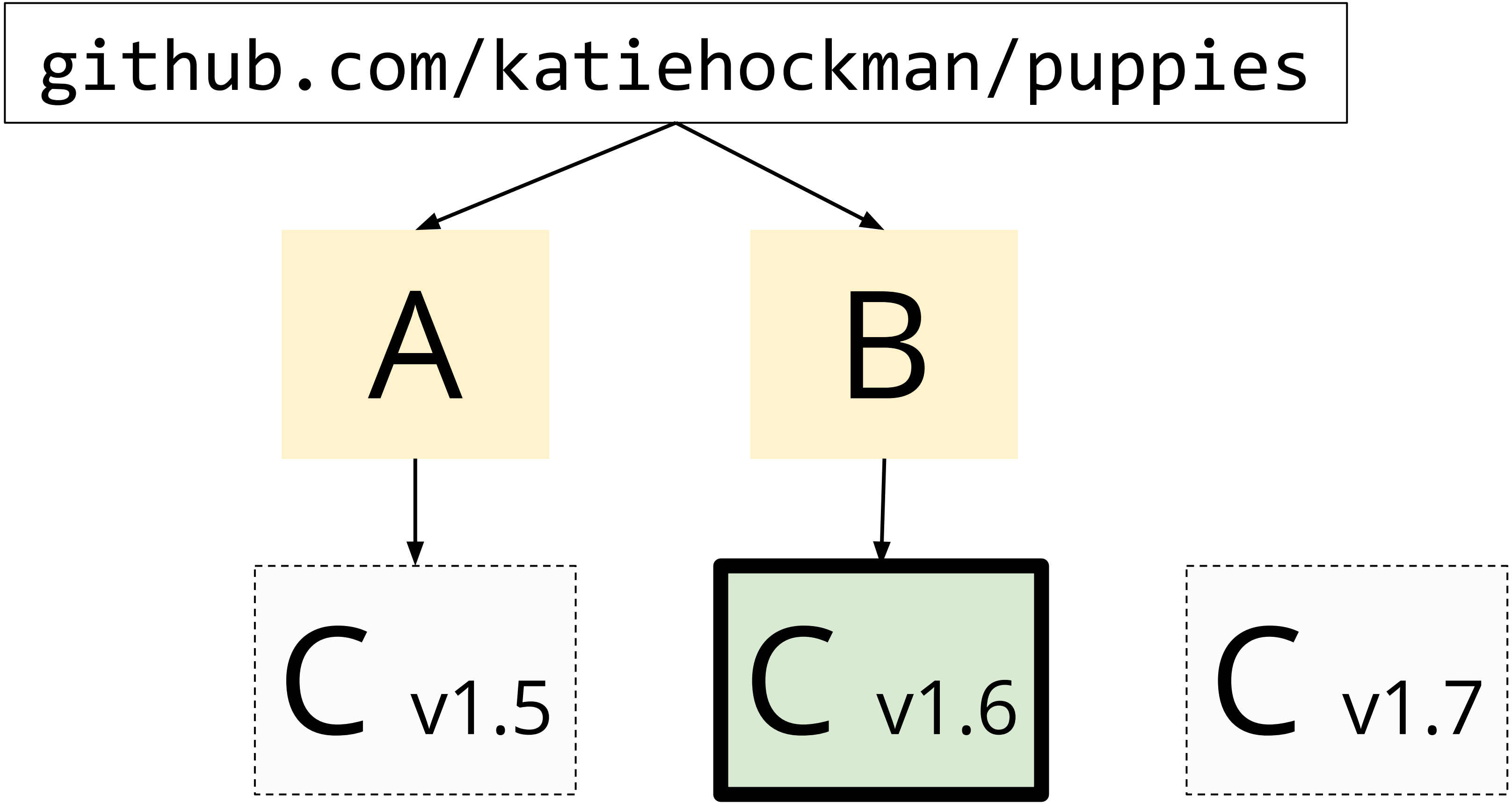
```
module github.com/katiehockman/puppies

require (
    github.com/maps/neighborhood v1.4.1
    github.com/audio/dogs v0.19.2
    golang.org/x/crypto v0.0.0-20190308221718-c2843e01d9a2
)
```

go.mod

```
module github.com/katiehockman/puppies

require (
    github.com/maps/neighborhood v1.4.1
    github.com/audio/dogs v0.19.2
    golang.org/x/crypto v0.0.0-20190308221718-c2843e01d9a2
)
```

Minimal version selection


github.com/katiehockman/puppies
A
B
C v1.5
C v1.6
C v1.7



github.com/katiehockman/puppies

A

B

C v1.5

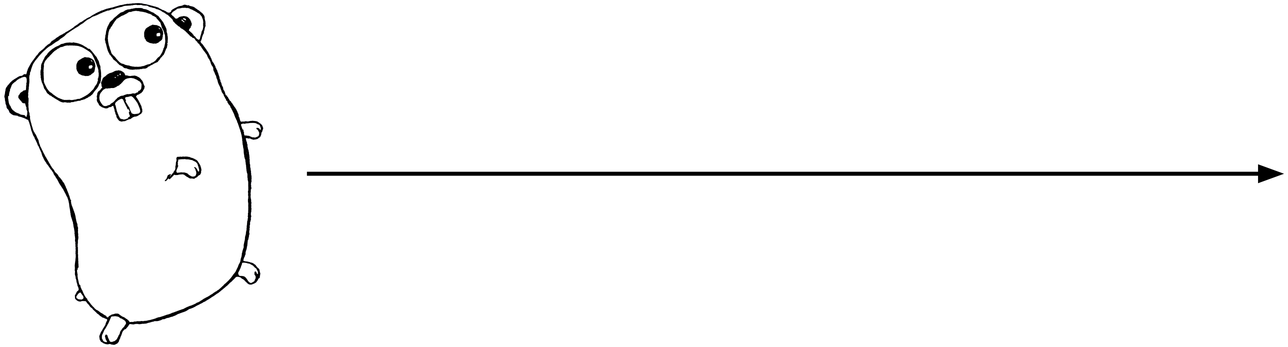
C v1.6

C v1.7

Better solutions

✓ Reproducible builds	Modules
2. Persistent dependencies	Mirror
3. Trustworthy fetches	Checksum Database

2. Mirror/Proxy



Origin server
(ie. GitHub)

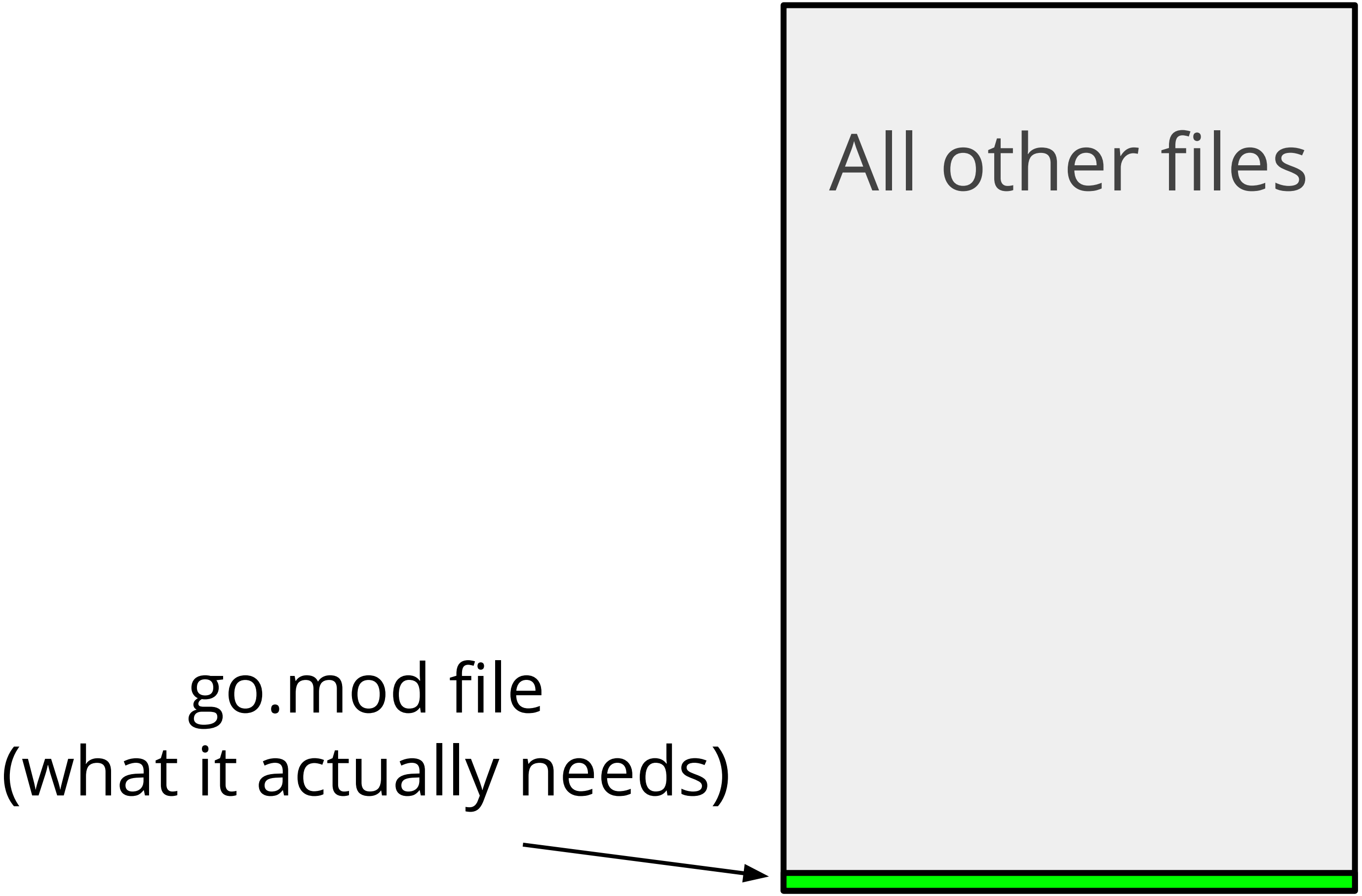
go command

go get

1. Fetch the source code
2. Resolve dependencies

Lots of files

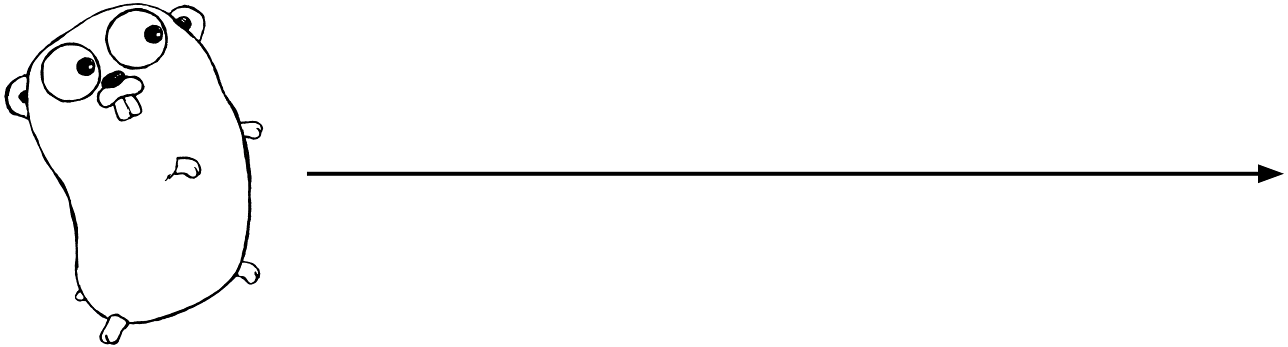
Mirror



All other files

go.mod file
(what it actually needs)

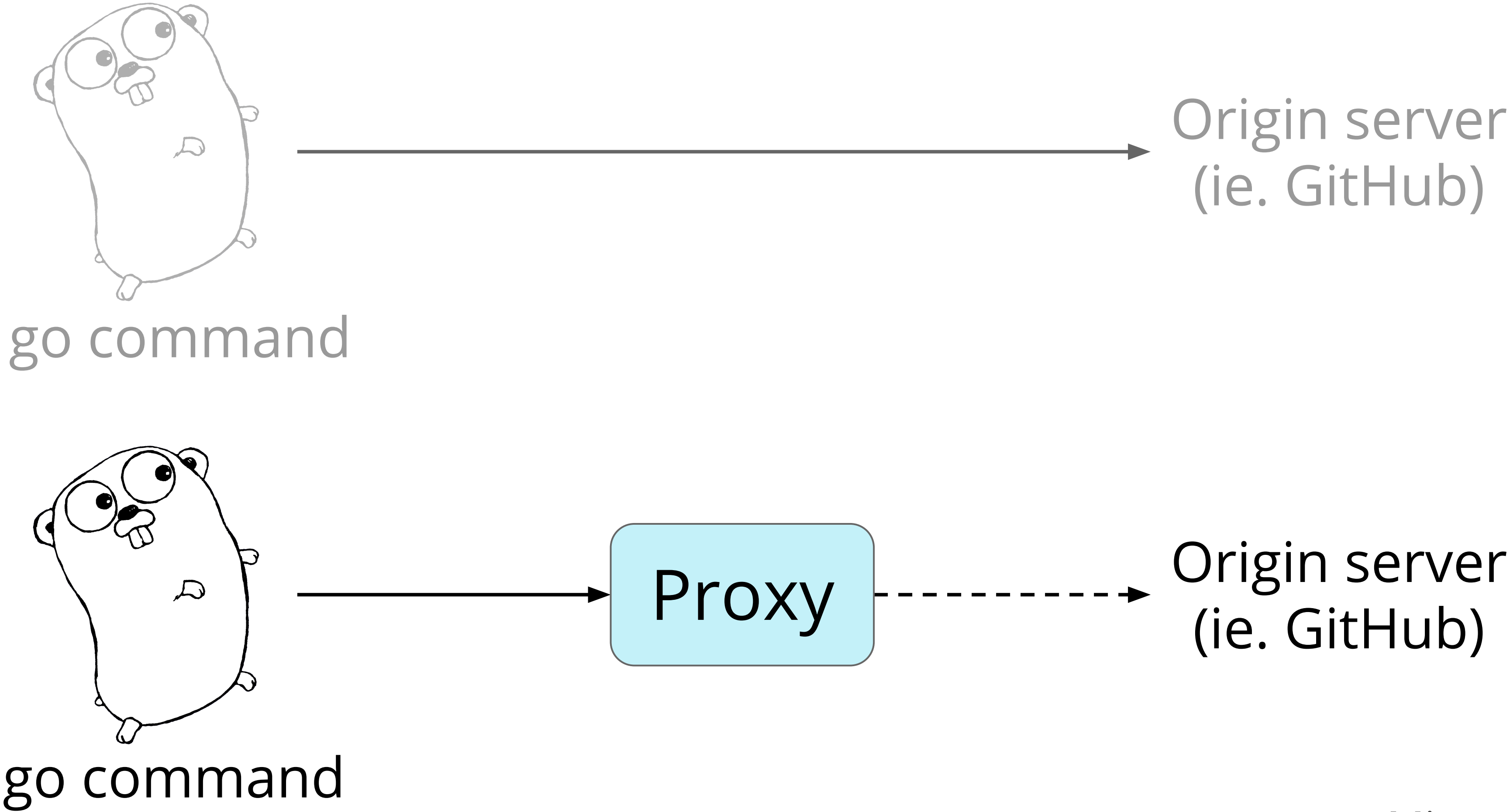
Mirror



go command

Origin server
(ie. GitHub)

Mirror



go command

Origin server
(ie. GitHub)

go command

Proxy

Origin server
(ie. GitHub)

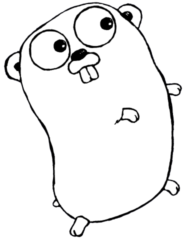
Mirror

go command

Proxy

Mirror

```
go get go.dog/breeds
```

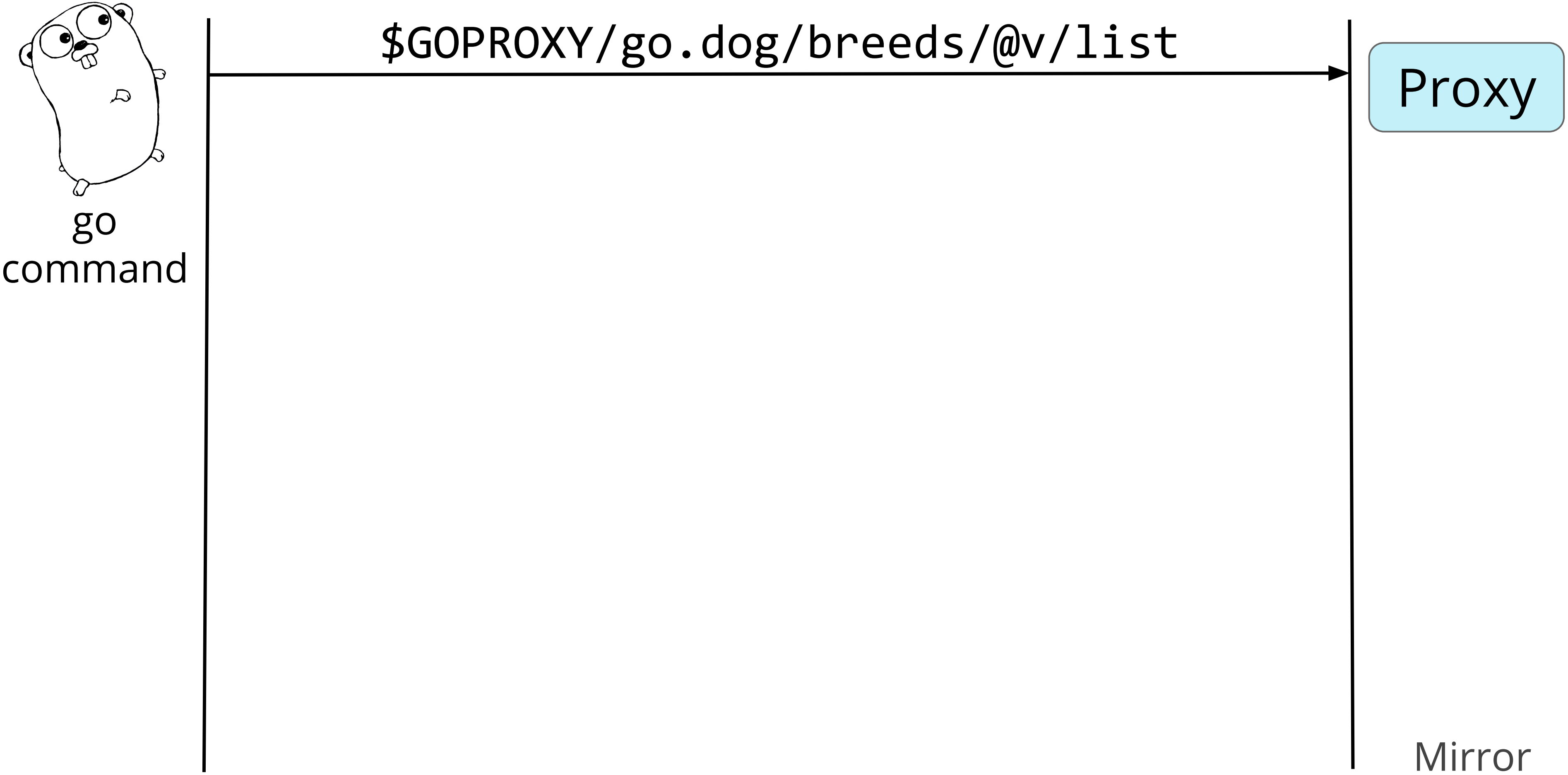


go command

Proxy

Mirror

```
go get go.dog/breeds
```



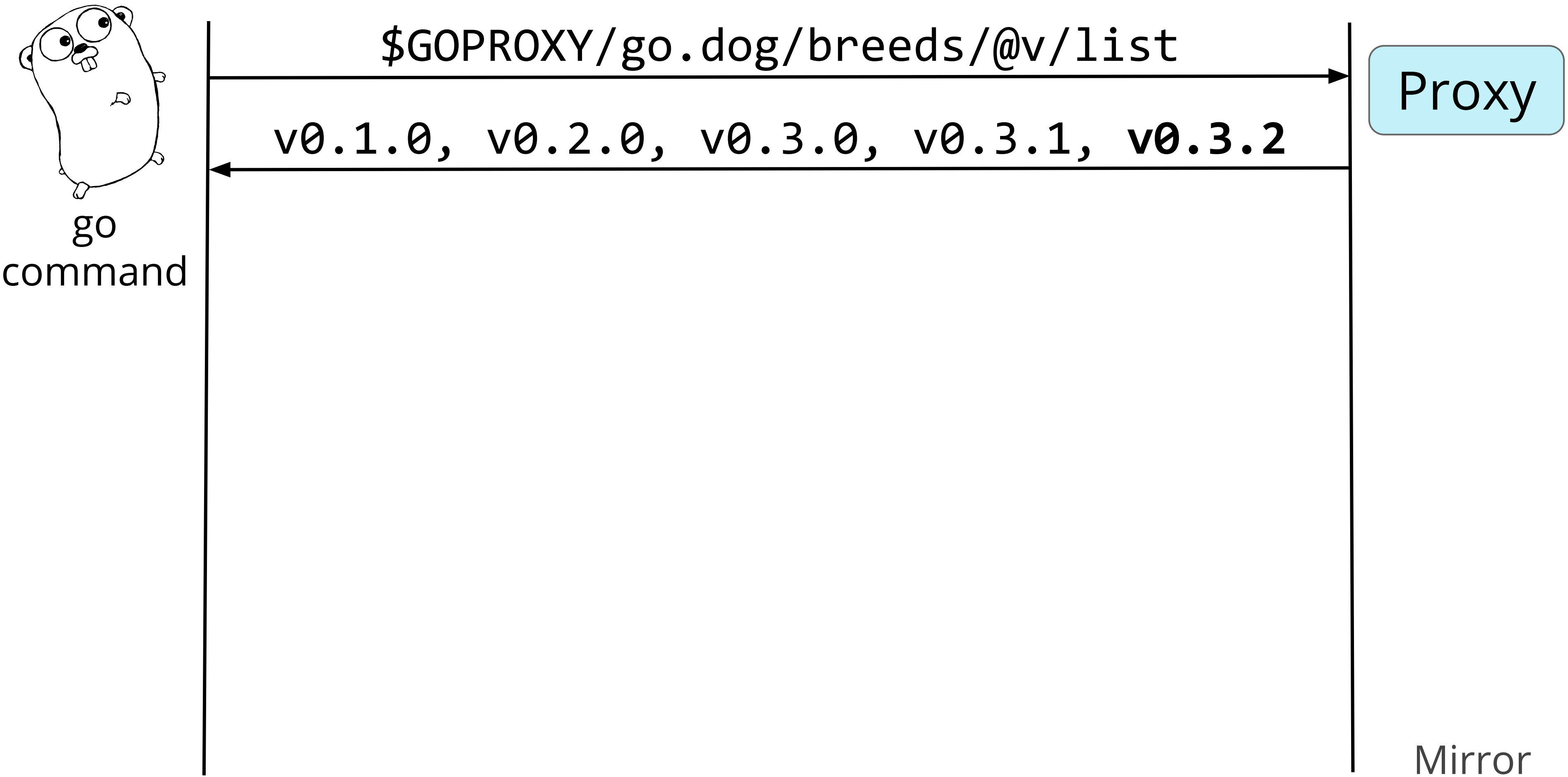
go command

$GOPROXY/go.dog/breeds/@v/list

Proxy

Mirror

```
go get go.dog/breeds
```



go command

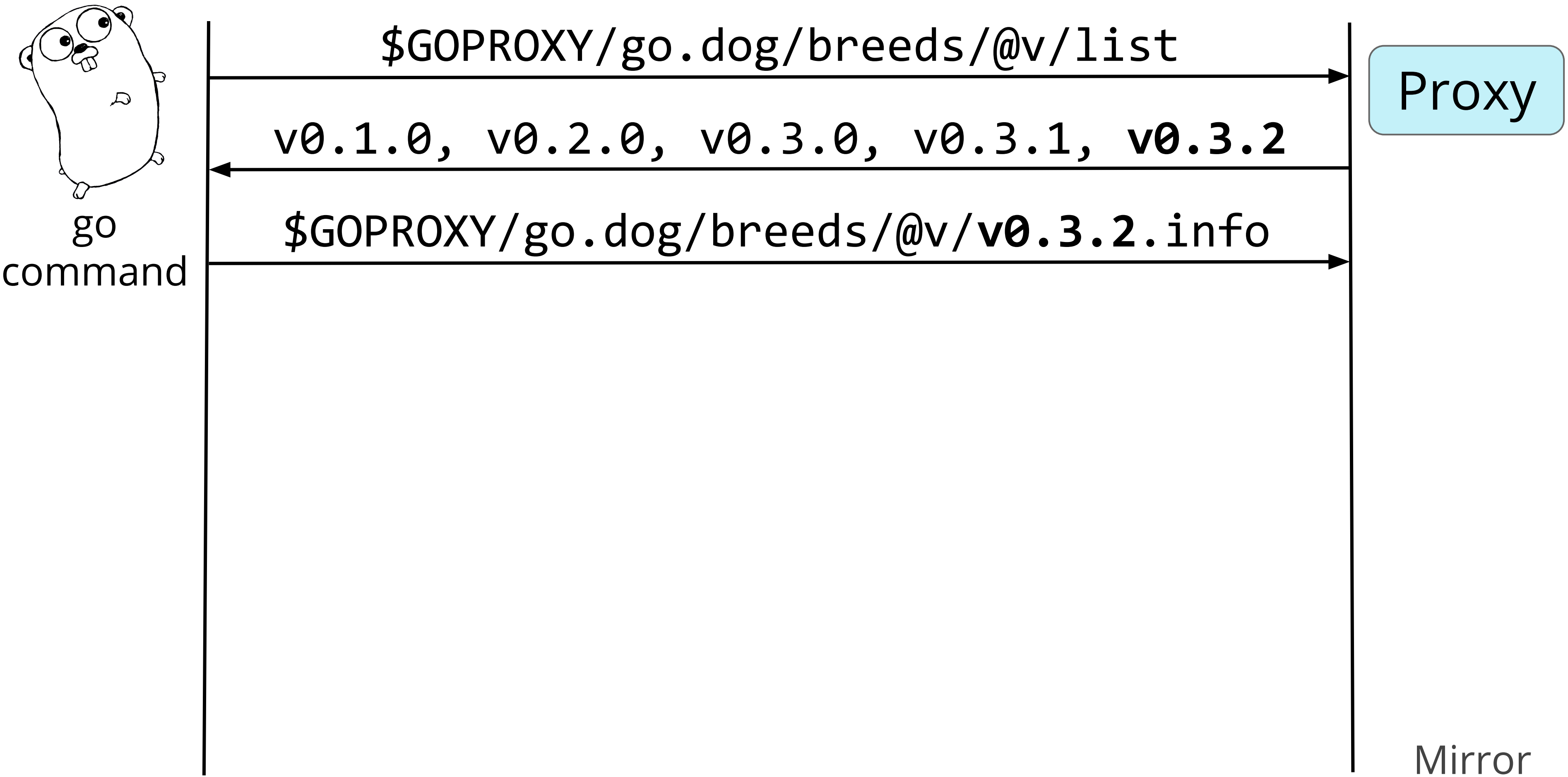
$GOPROXY/go.dog/breeds/@v/list

v0.1.0, v0.2.0, v0.3.0, v0.3.1, **v0.3.2**

Proxy

Mirror

`go get go.dog/breeds`



go command

`$GOPROXY/go.dog/breeds/@v/list`

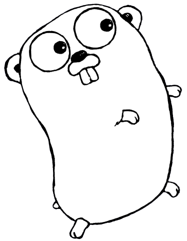
Proxy

v0.1.0, v0.2.0, v0.3.0, v0.3.1, **v0.3.2**

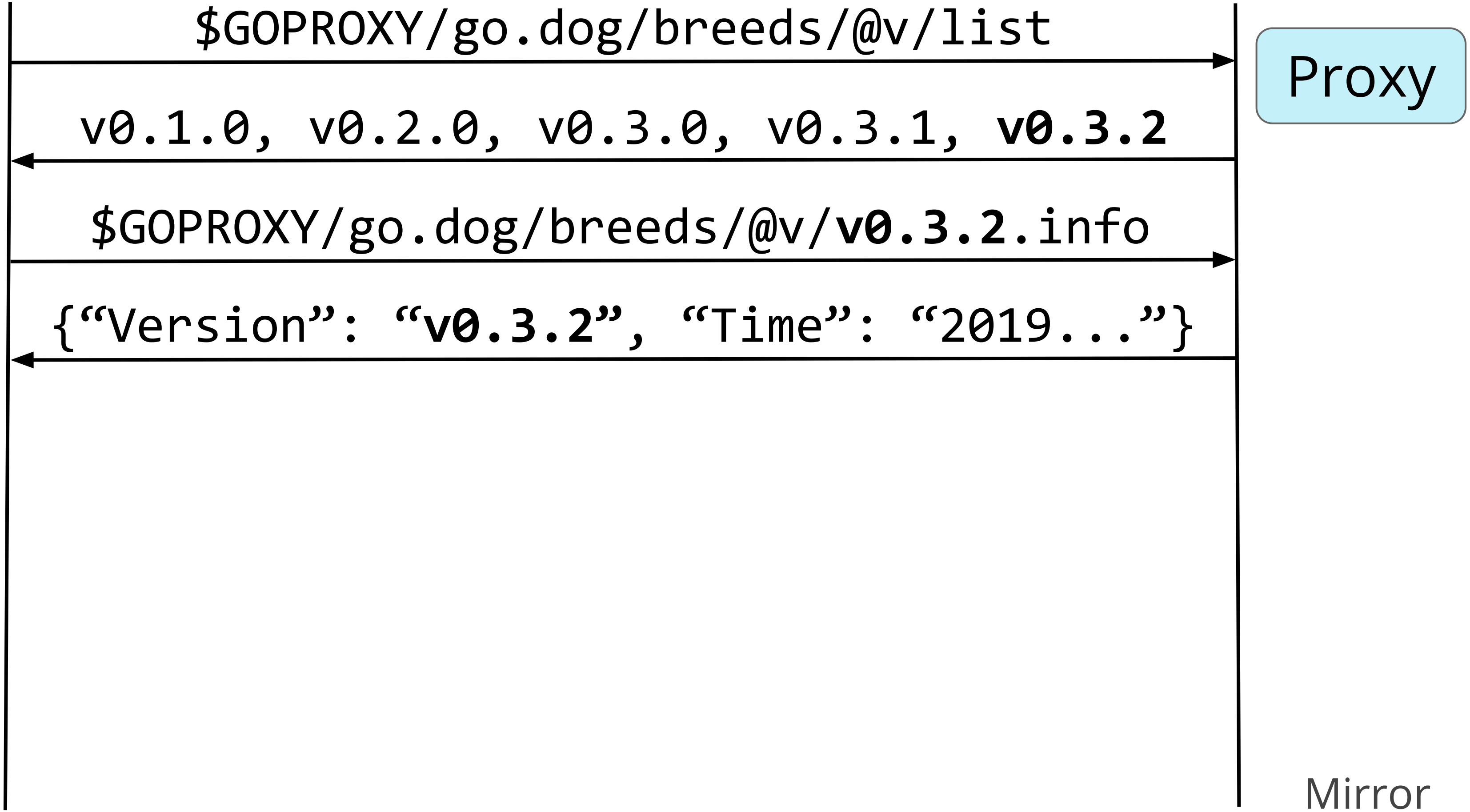
$GOPROXY/go.dog/breeds/@v/**v0.3.2**.info

Mirror

go get go.dog/breeds



go
command



$GOPROXY/go.dog/breeds/@v/list

v0.1.0, v0.2.0, v0.3.0, v0.3.1, **v0.3.2**

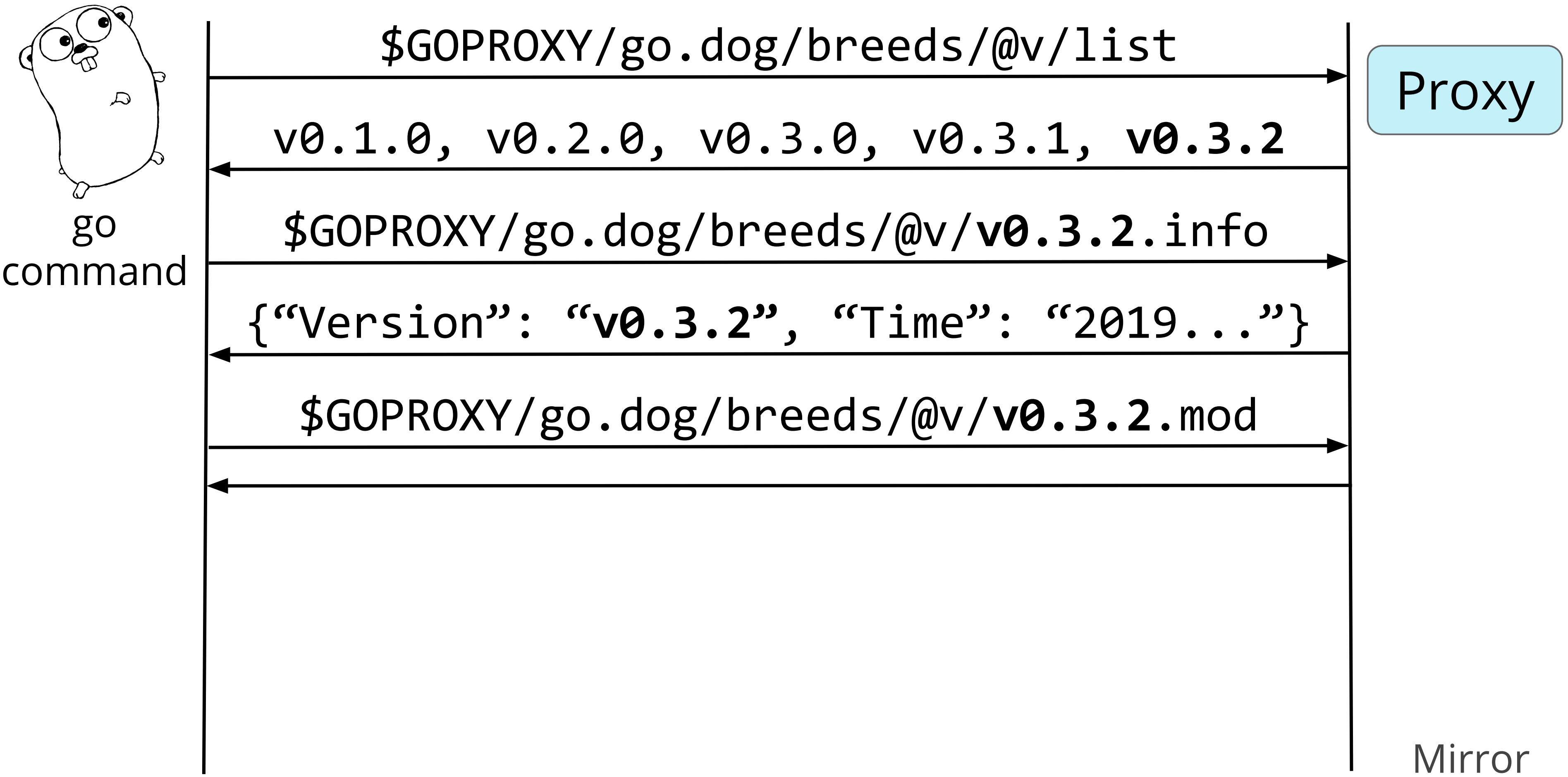
$GOPROXY/go.dog/breeds/@v/**v0.3.2**.info

{“Version”: “**v0.3.2**”, “Time”: “2019...”}

Proxy

Mirror

go get go.dog/breeds



go command

Proxy

$GOPROXY/go.dog/breeds/@v/list

v0.1.0, v0.2.0, v0.3.0, v0.3.1, **v0.3.2**

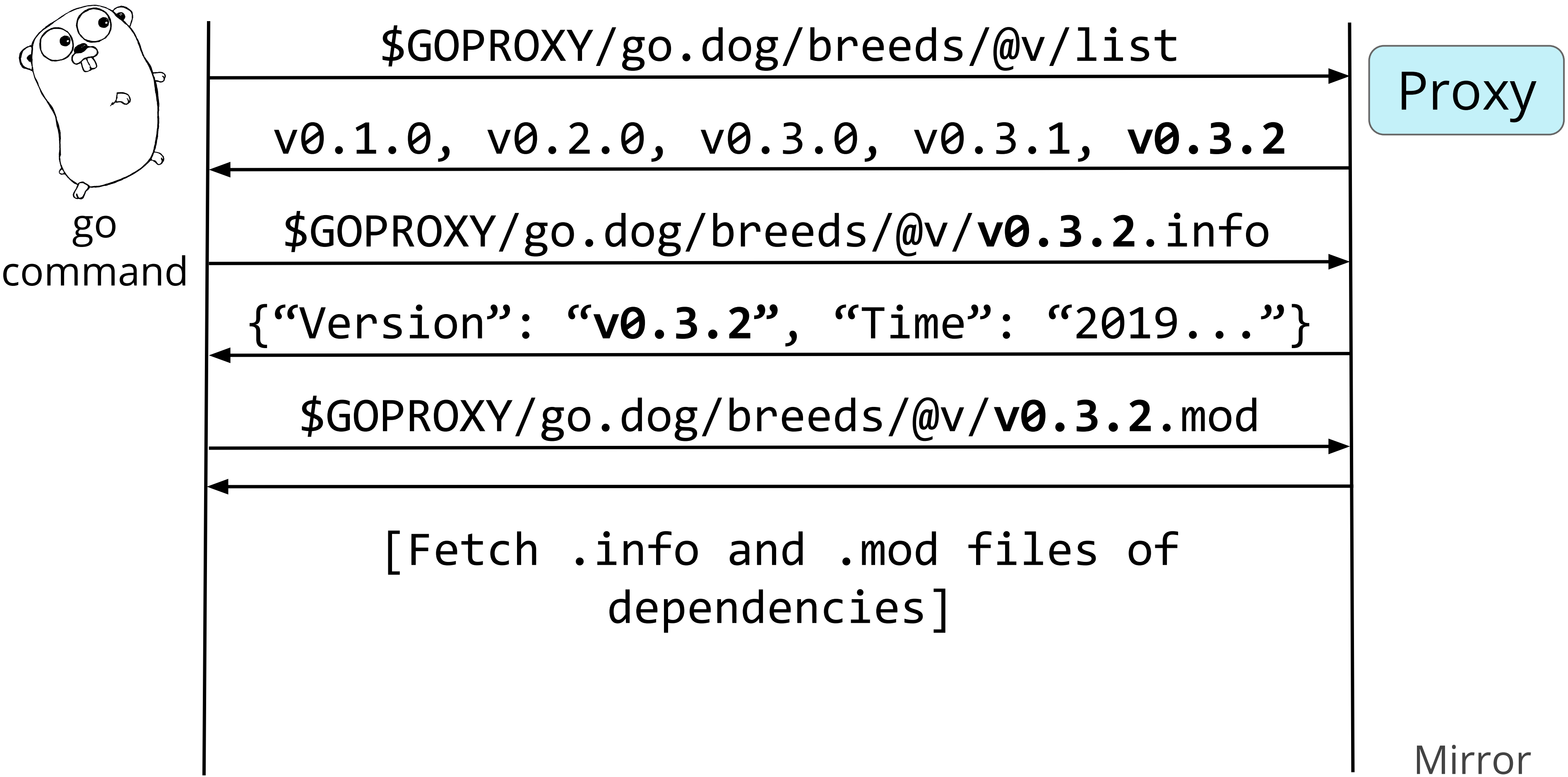
$GOPROXY/go.dog/breeds/@v/**v0.3.2**.info

{“Version”: “**v0.3.2**”, “Time”: “2019...”}

$GOPROXY/go.dog/breeds/@v/**v0.3.2**.mod

Mirror

```
go get go.dog/breeds
```



$GOPROXY/go.dog/breeds/@v/list

v0.1.0, v0.2.0, v0.3.0, v0.3.1, **v0.3.2**

$GOPROXY/go.dog/breeds/@v/**v0.3.2**.info

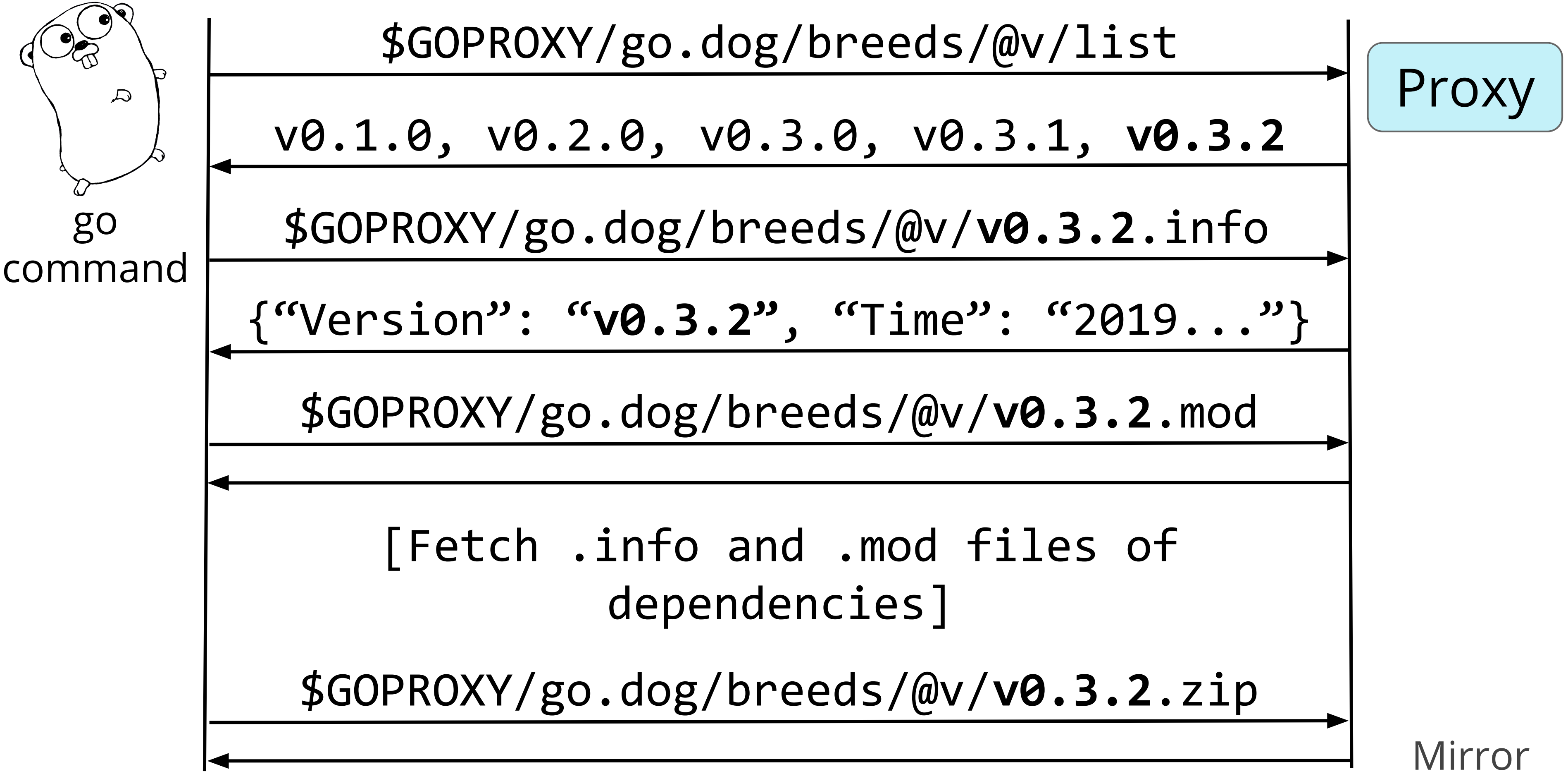
{“Version”: “**v0.3.2**”, “Time”: “2019...”}

$GOPROXY/go.dog/breeds/@v/**v0.3.2**.mod

[Fetch .info and .mod files of dependencies]

Mirror

go get go.dog/breeds



go command

Proxy

$GOPROXY/go.dog/breeds/@v/list

v0.1.0, v0.2.0, v0.3.0, v0.3.1, **v0.3.2**

$GOPROXY/go.dog/breeds/@v/**v0.3.2**.info

{“Version”: “**v0.3.2**”, “Time”: “2019...”}

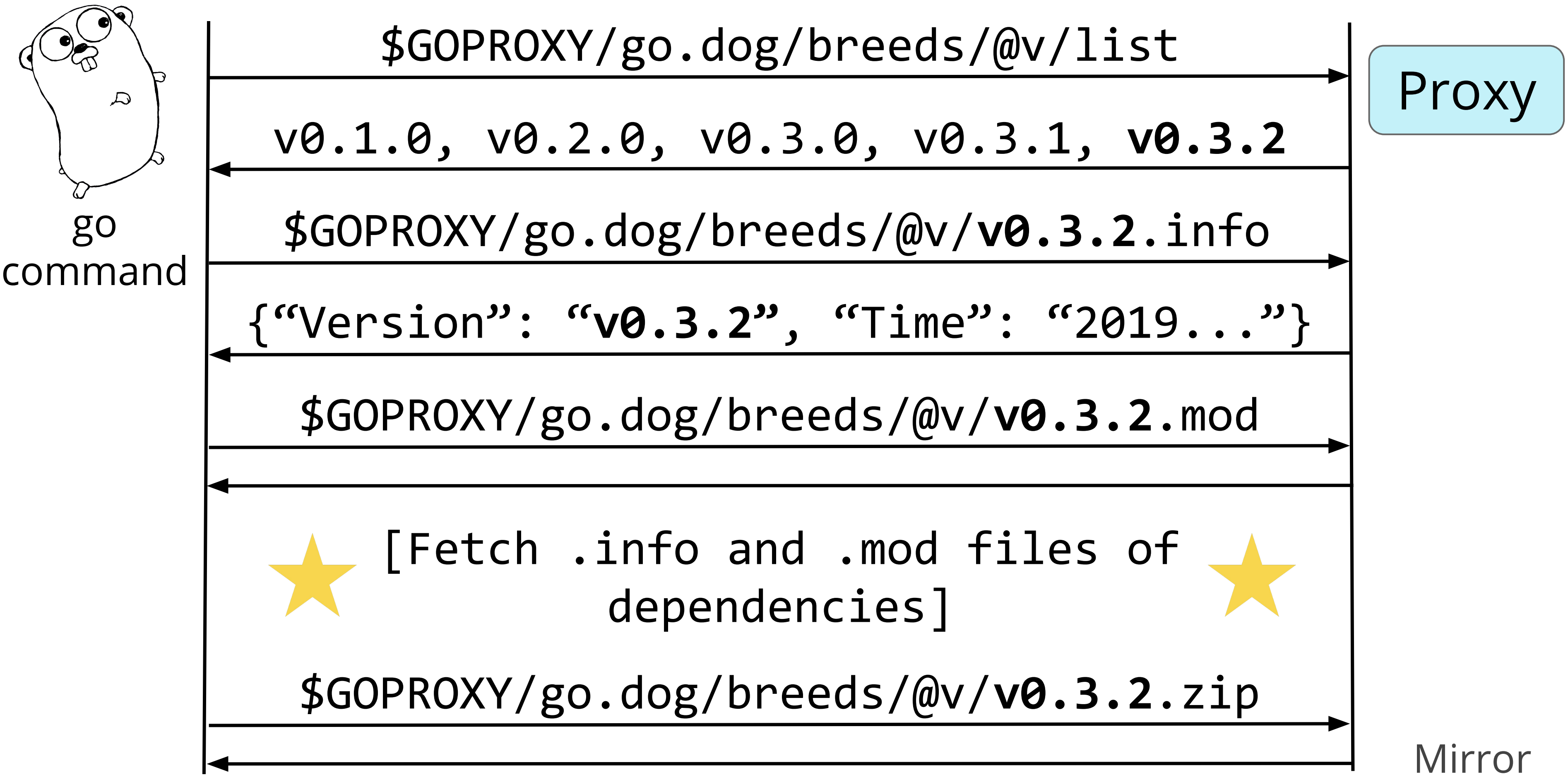
$GOPROXY/go.dog/breeds/@v/**v0.3.2**.mod

[Fetch .info and .mod files of dependencies]

$GOPROXY/go.dog/breeds/@v/**v0.3.2**.zip

Mirror

go get go.dog/breeds



go command

Proxy

$GOPROXY/go.dog/breeds/@v/list

v0.1.0, v0.2.0, v0.3.0, v0.3.1, **v0.3.2**

$GOPROXY/go.dog/breeds/@v/**v0.3.2**.info

{“Version”: “**v0.3.2**”, “Time”: “2019...”}

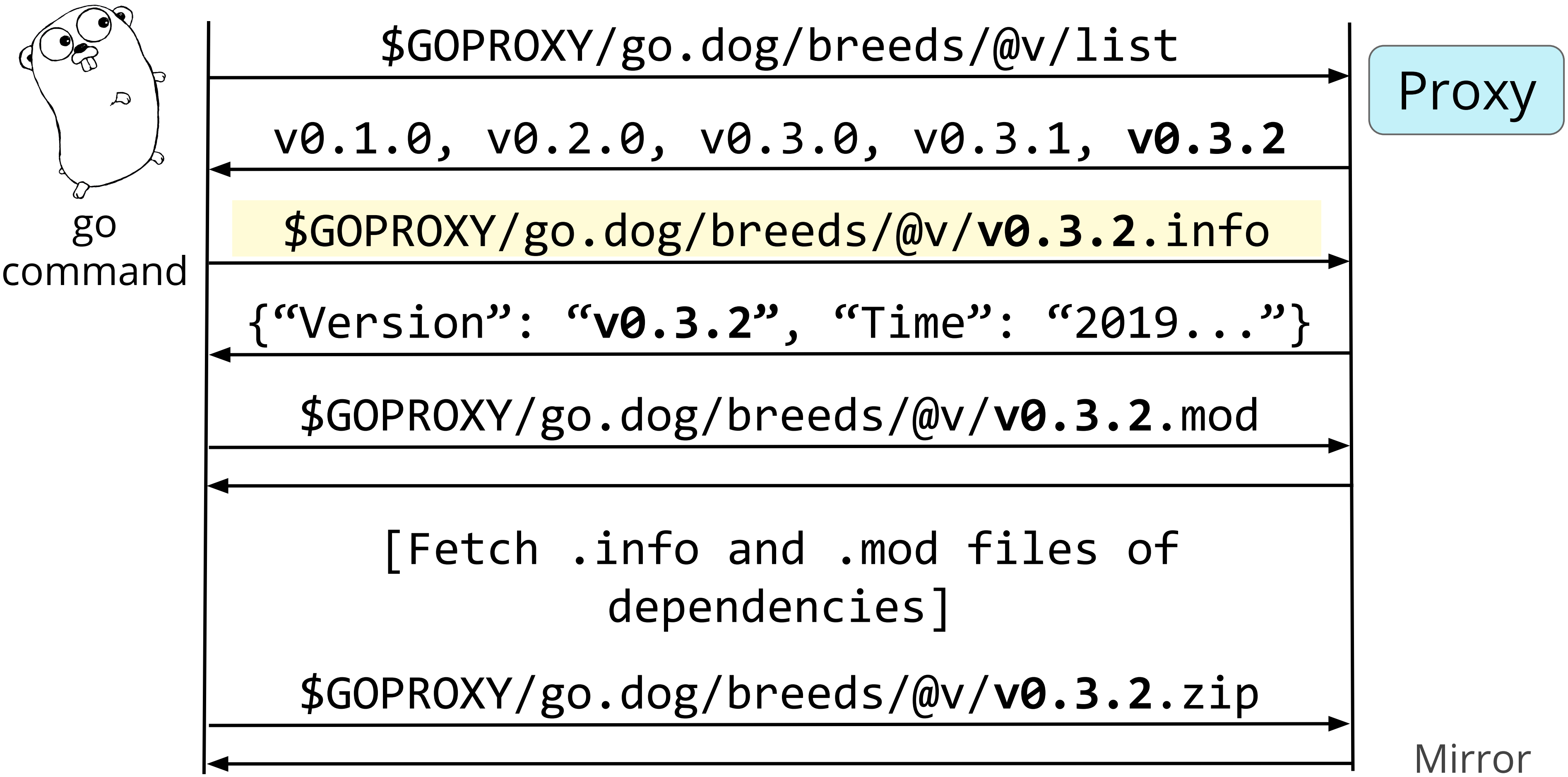
$GOPROXY/go.dog/breeds/@v/**v0.3.2**.mod

[Fetch .info and .mod files of dependencies]

$GOPROXY/go.dog/breeds/@v/**v0.3.2**.zip

Mirror

go get go.dog/breeds



go command

Proxy

$GOPROXY/go.dog/breeds/@v/list

v0.1.0, v0.2.0, v0.3.0, v0.3.1, **v0.3.2**

$GOPROXY/go.dog/breeds/@v/**v0.3.2**.info

{“Version”: “**v0.3.2**”, “Time”: “2019...”}

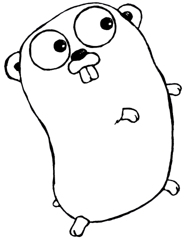
$GOPROXY/go.dog/breeds/@v/**v0.3.2**.mod

[Fetch .info and .mod files of dependencies]

$GOPROXY/go.dog/breeds/@v/**v0.3.2**.zip

Mirror

```
go get go.dog/breeds@master
```

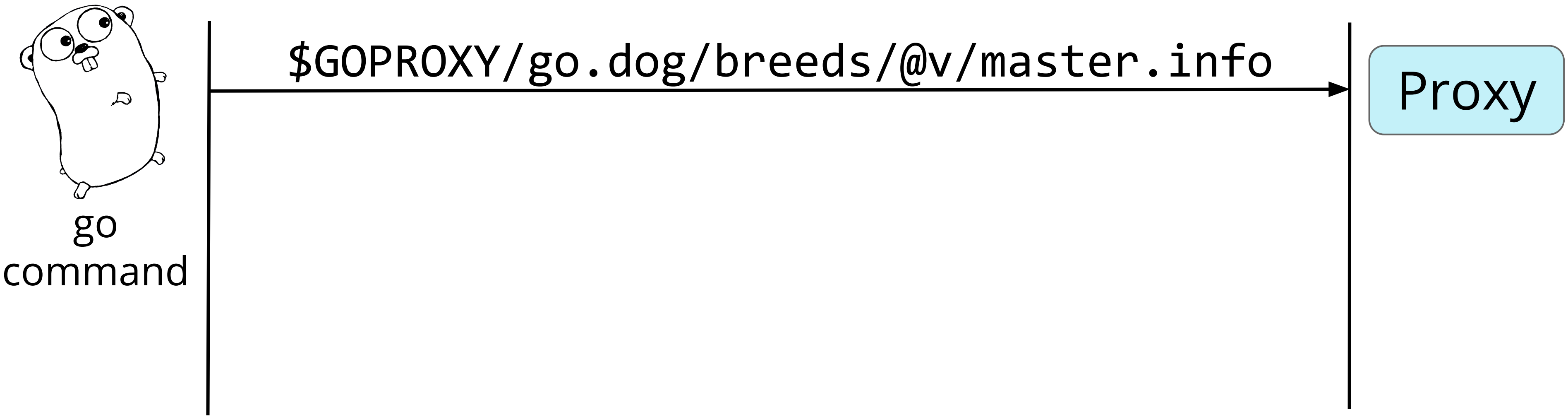


go
command

Proxy

Mirror

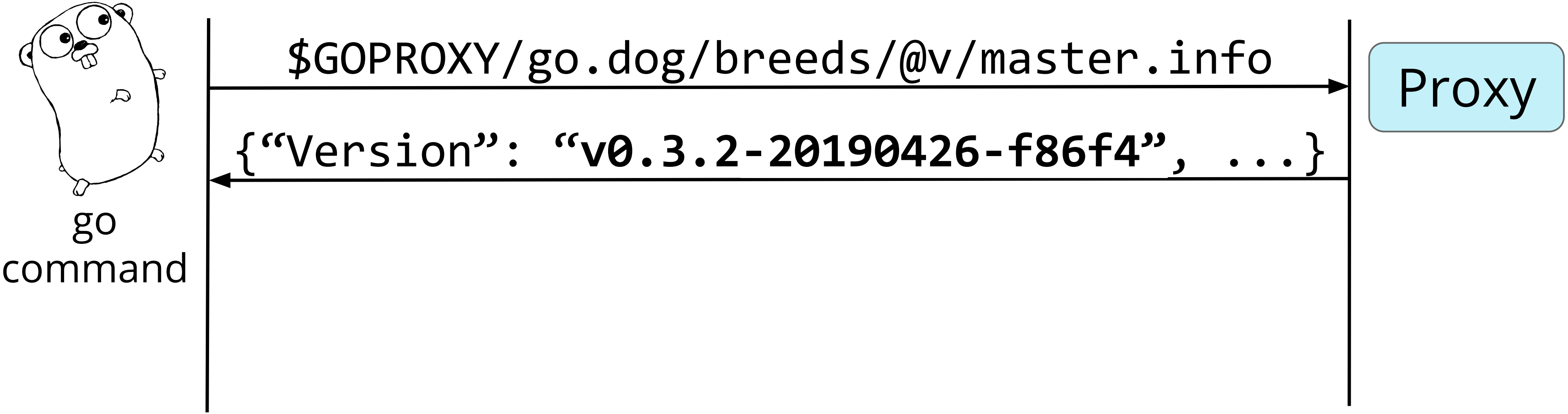
```
go get go.dog/breeds@master
```



```
$GOPROXY/go.dog/breeds/@v/master.info
```

Mirror

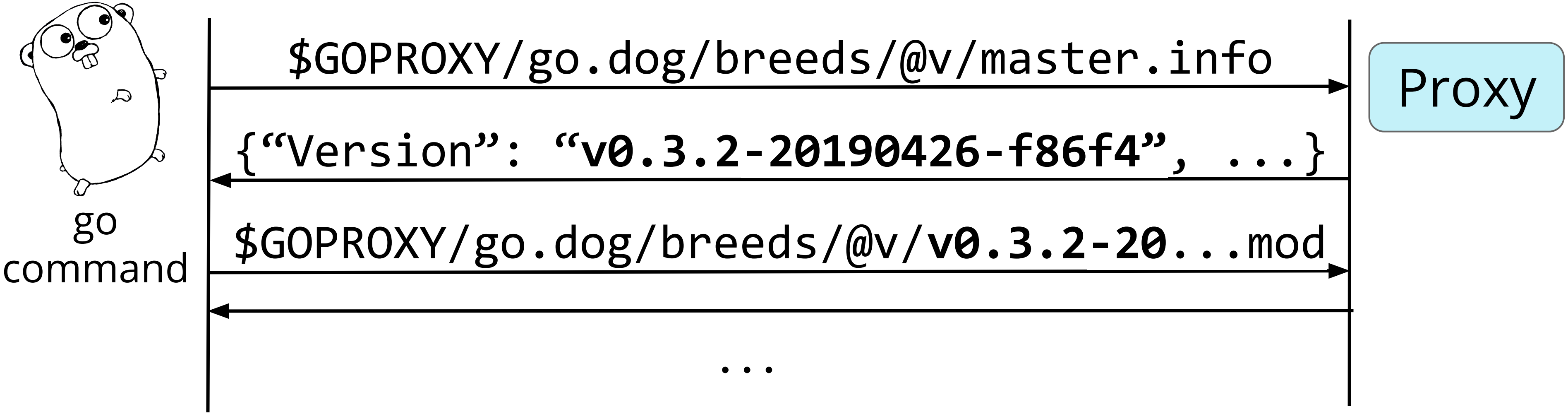
```
go get go.dog/breeds@master
```



```
$GOPROXY/go.dog/breeds/@v/master.info
```

```
{"Version": "v0.3.2-20190426-f86f4", ...}
```

go get go.dog/breeds@master



go command

Proxy

$GOPROXY/go.dog/breeds/@v/master.info

{“Version”: “**v0.3.2-20190426-f86f4**”, ...}

$GOPROXY/go.dog/breeds/@v/**v0.3.2-20**...mod

...

Mirror

Proxy spec (go help goproxy)

`GET $GOPROXY/<module>/@v/list` returns a list of all known versions of the given module

`GET $GOPROXY/<module>/@v/<version>.info` returns JSON metadata about that version of the given module.

`GET $GOPROXY/<module>/@v/<version>.mod`

`GET $GOPROXY/<module>/@v/<version>.zip`

Mirror = Proxy (+ caching)

Why use a mirror?

Why use a mirror?

- ✓ No more disappearing dependencies

Why use a mirror?

✓ No more disappearing dependencies

✓ Faster downloads

Why use a mirror?

- ✓ No more disappearing dependencies
- ✓ Faster downloads
- ✓ Less storage use

Why use a mirror?

- ✓ No more disappearing dependencies
- ✓ Faster downloads
- ✓ Less storage use
- ✓ Easy to use

Better solutions

✓ Reproducible builds	Modules
✓ Persistent dependencies	Mirror
3. Trustworthy fetches	Checksum Database

3. Checksum Database

The old way… direct HTTPS

The new way… go.sum files

go.sum for github.com/katiehockman/puppies

```
github.com/maps/neighborhood v1.4.1 h1:g24URVg0OFbNUTx9qq...
github.com/maps/neighborhood v1.4.1/go.mod h1:zAC/RDZ24gD...

github.com/audio/dogs v0.19.2 h1:ZZpq6xI6kv/LuE/5s5UQvBU...
github.com/audio/dogs v0.19.2/go.mod h1:NO/8qkisMZLZ1FCs...

golang.org/x/crypto v0.0.0-20190325154230-a5d413f7728c h1:V...
golang.org/x/crypto v0.0.0-20190325154230-a5d413f7728c/go.mod
```

Check in your module’s go.sum file!

My module’s go.sum file

```
github.com/maps/neighborhood v1.4.1 h1:g24URVg0OFbNUTx9...
github.com/maps/neighborhood v1.4.1/go.mod h1:zAC/RDZ24g...
github.com/audio/dogs v0.19.2 h1:ZZpq6xI6kvLuE5s5U...
github.com/audio/dogs v0.19.2/go.mod h1:NO/8qkisMZ...
```

The generated go.sum lines

```
github.com/audio/dogs v0.19.2 h1:ABfk0xE3fdLeS8s3W...
github.com/audio/dogs v0.19.2/go.mod h1:HI/0dfjwDS...
```

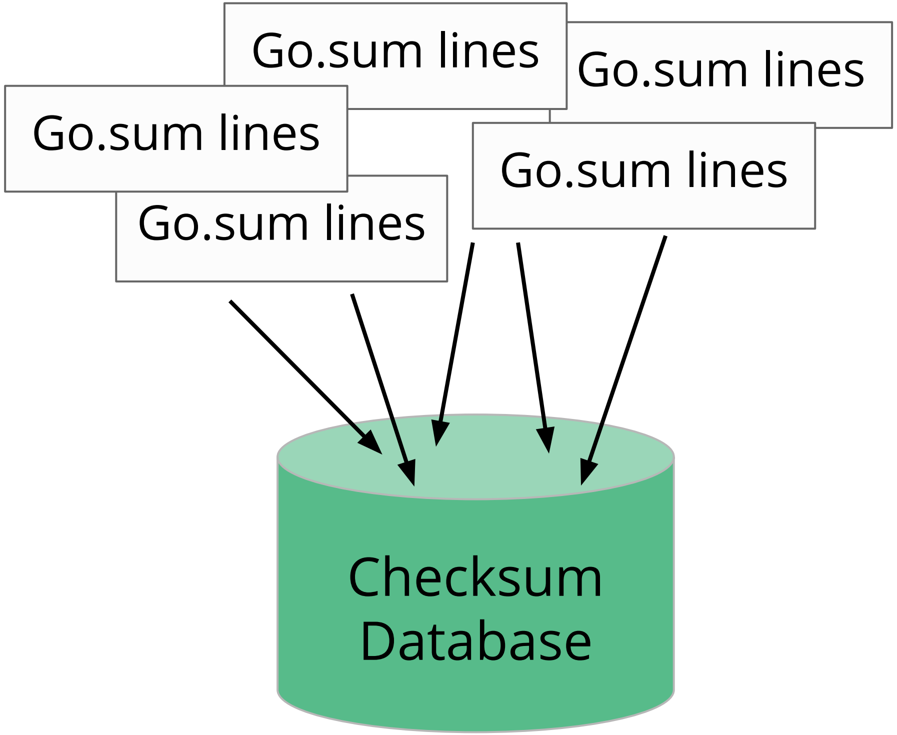
“Trust on ***your*** first use”

What about proxies?

The best possible scenario...

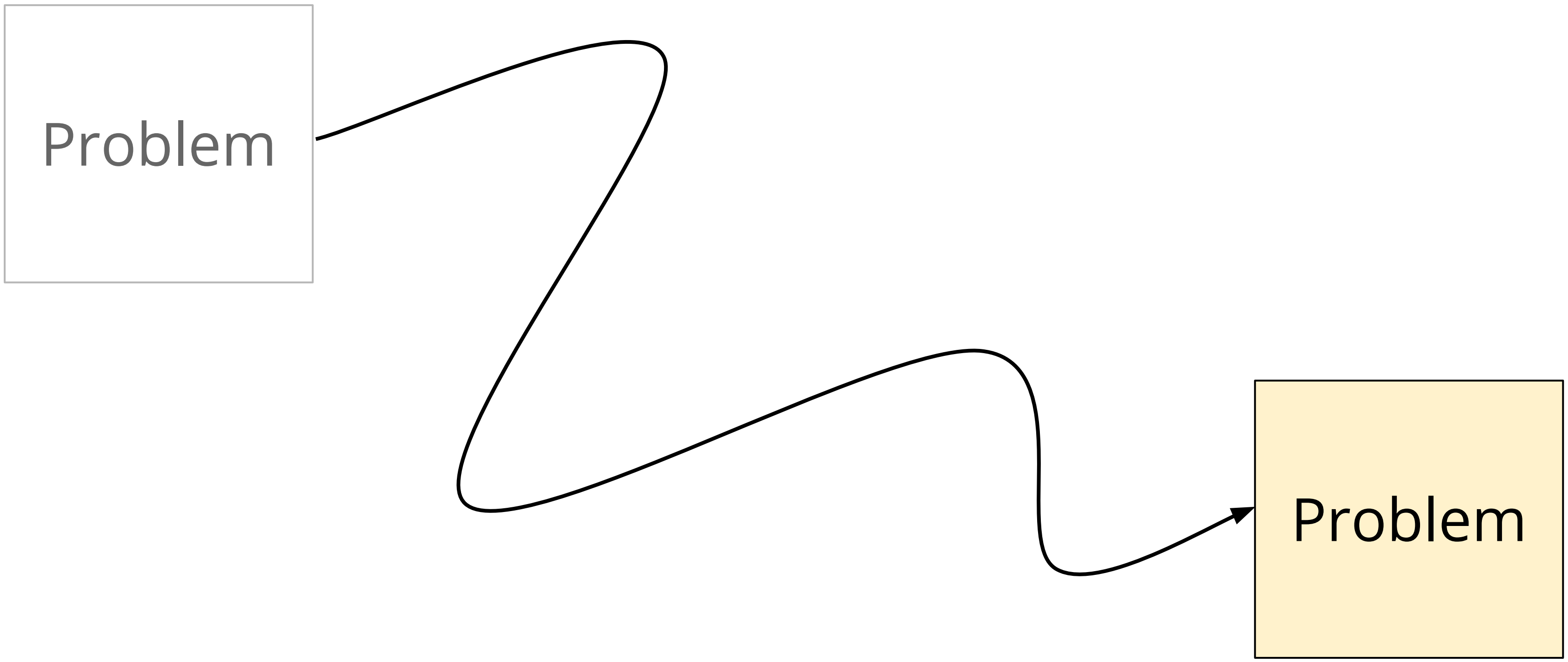
The **newest** way...

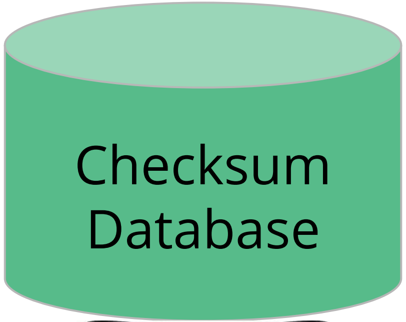
The **newest** way...



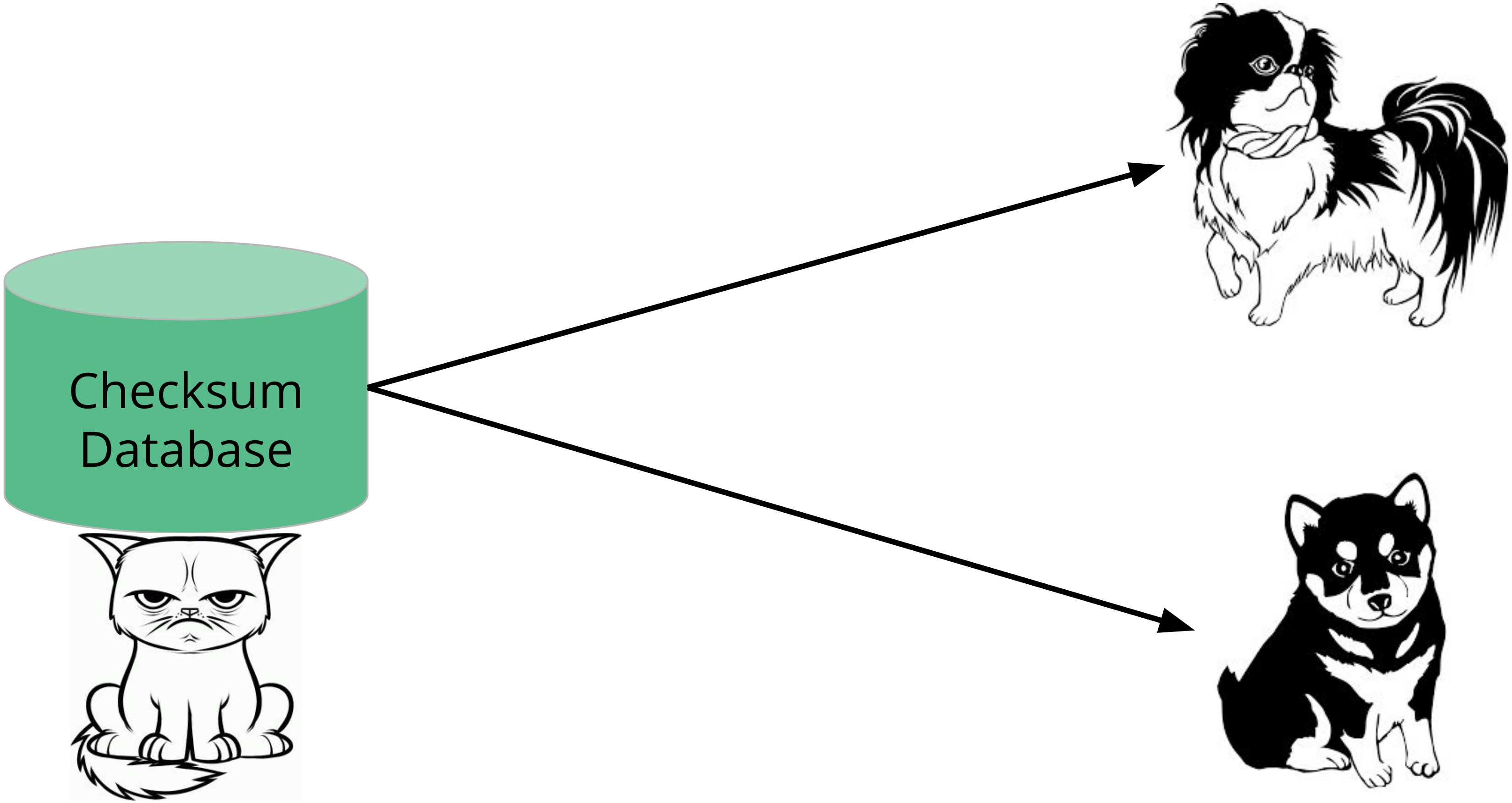
How about a database?

Problem


Problem
Problem

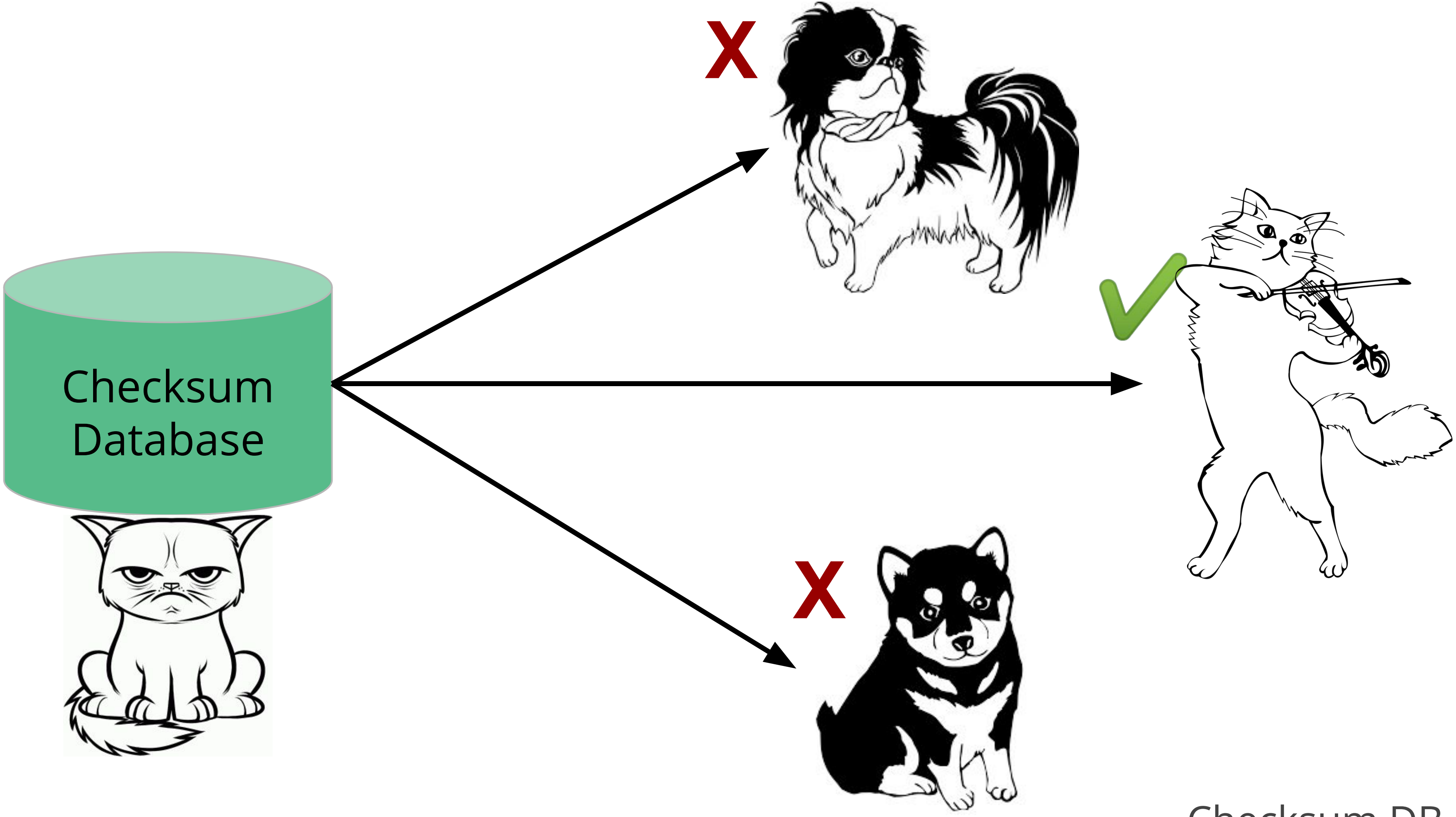


Checksum DB



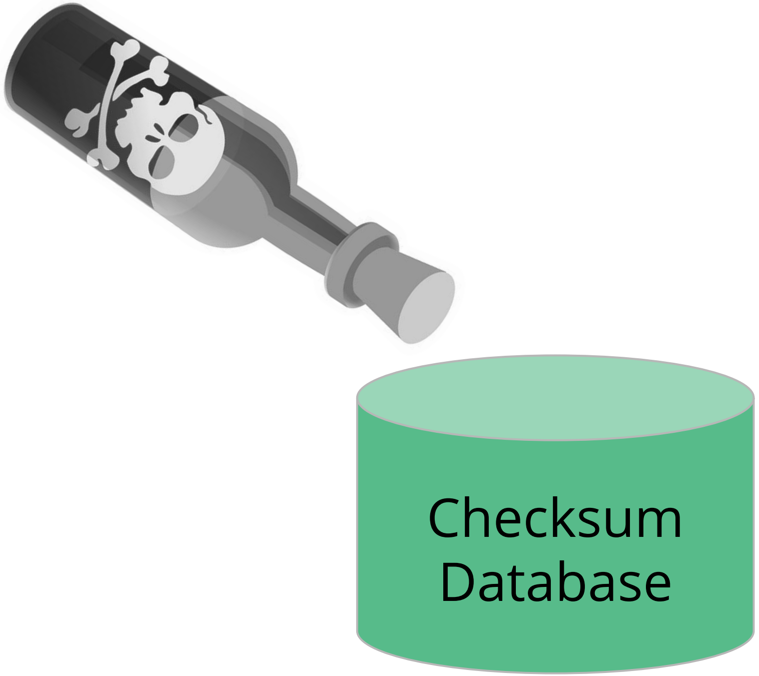
Checksum Database

Checksum DB



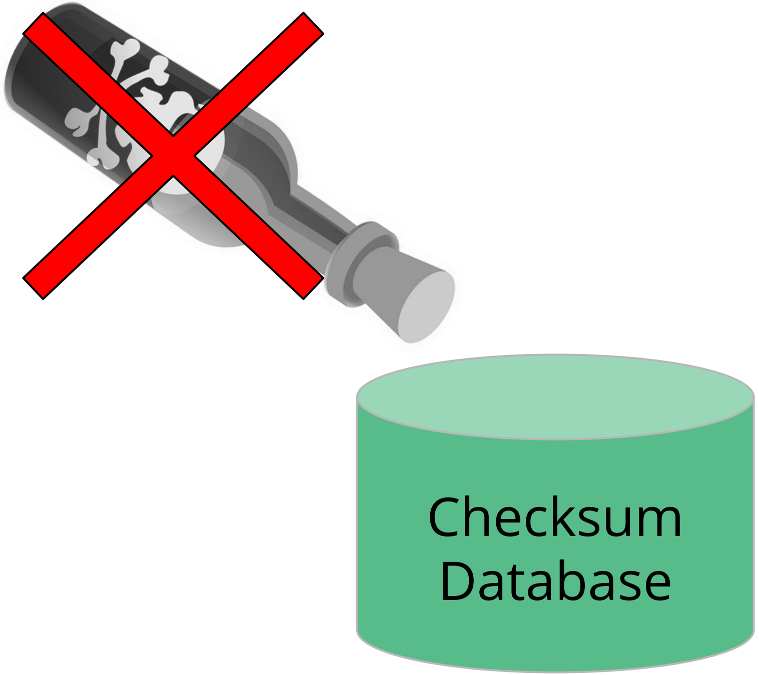
Checksum Database

Checksum DB



Checksum
Database

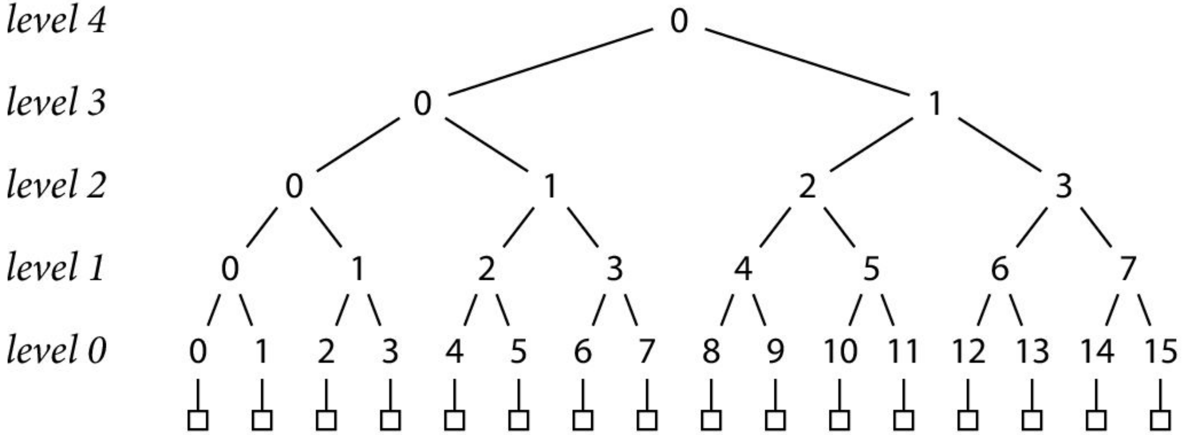
Checksum DB



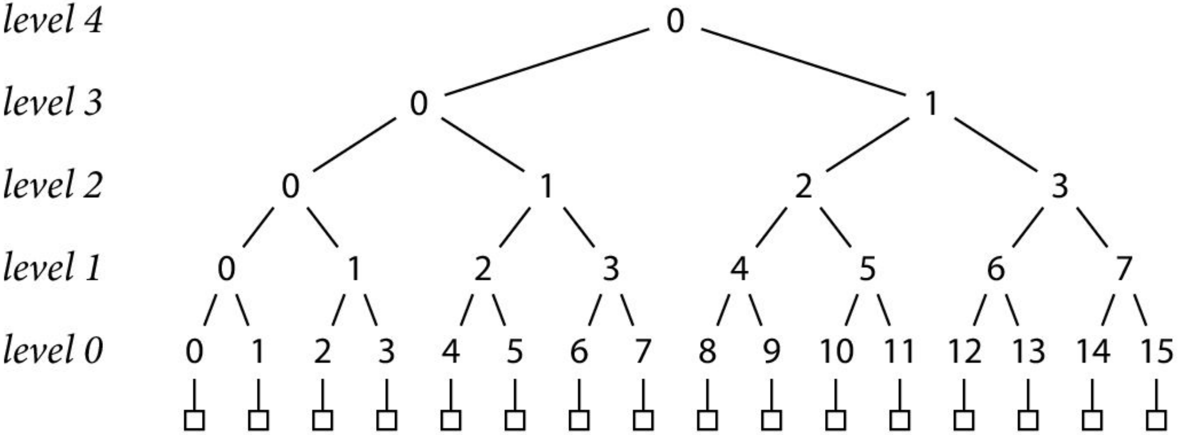
Checksum Database

Checksum DB

Transparent Log

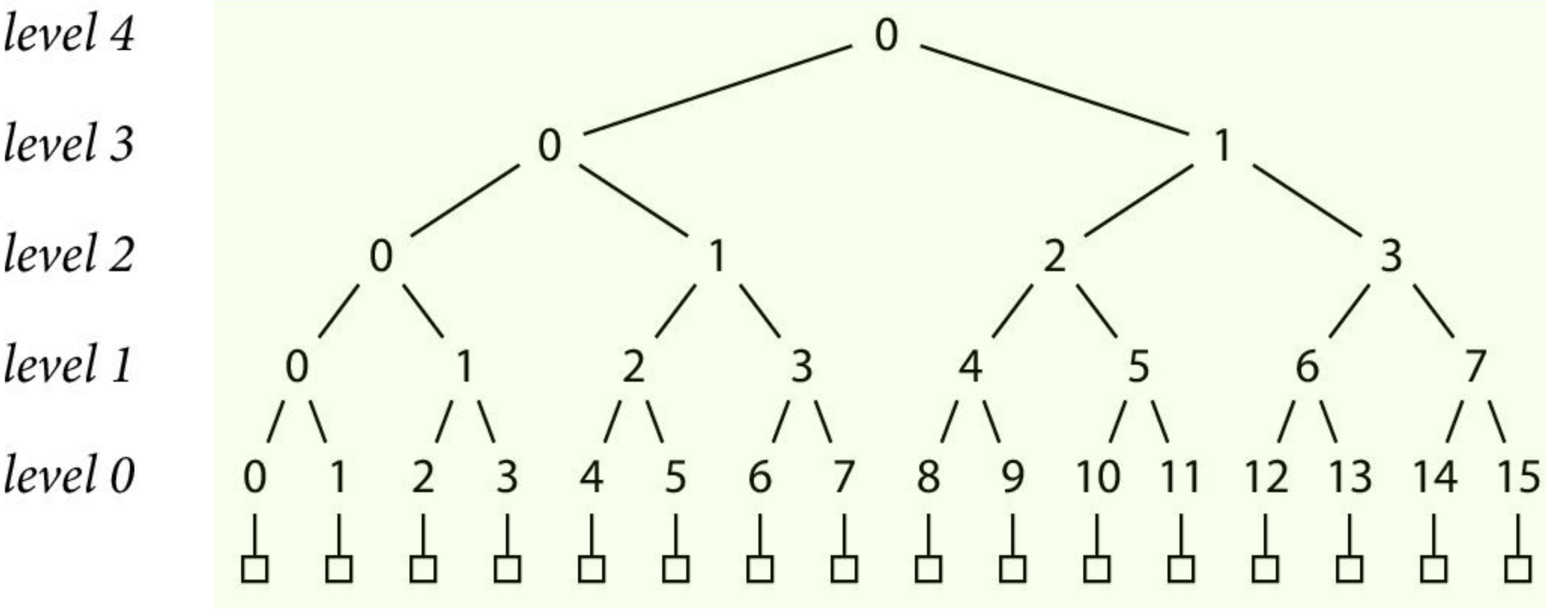


Transparent Log (merkle tree)

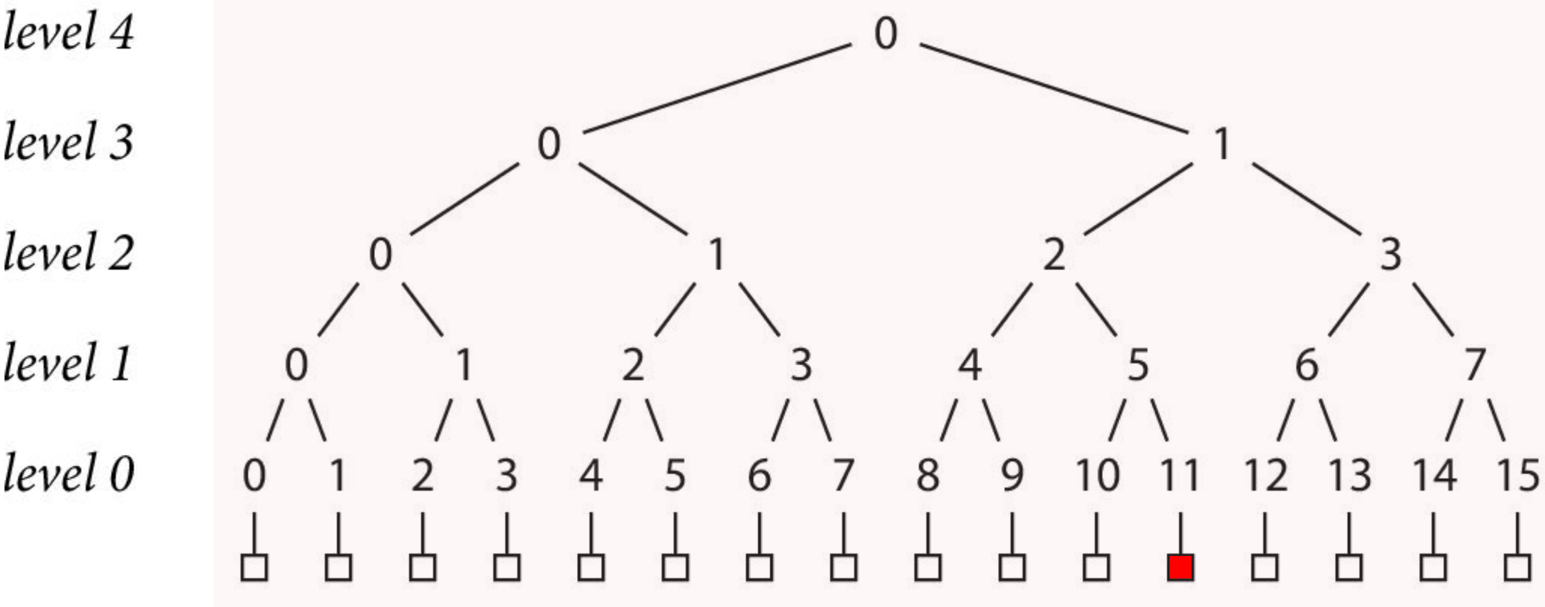


What about a merkle tree?

✓ Tamper-proof

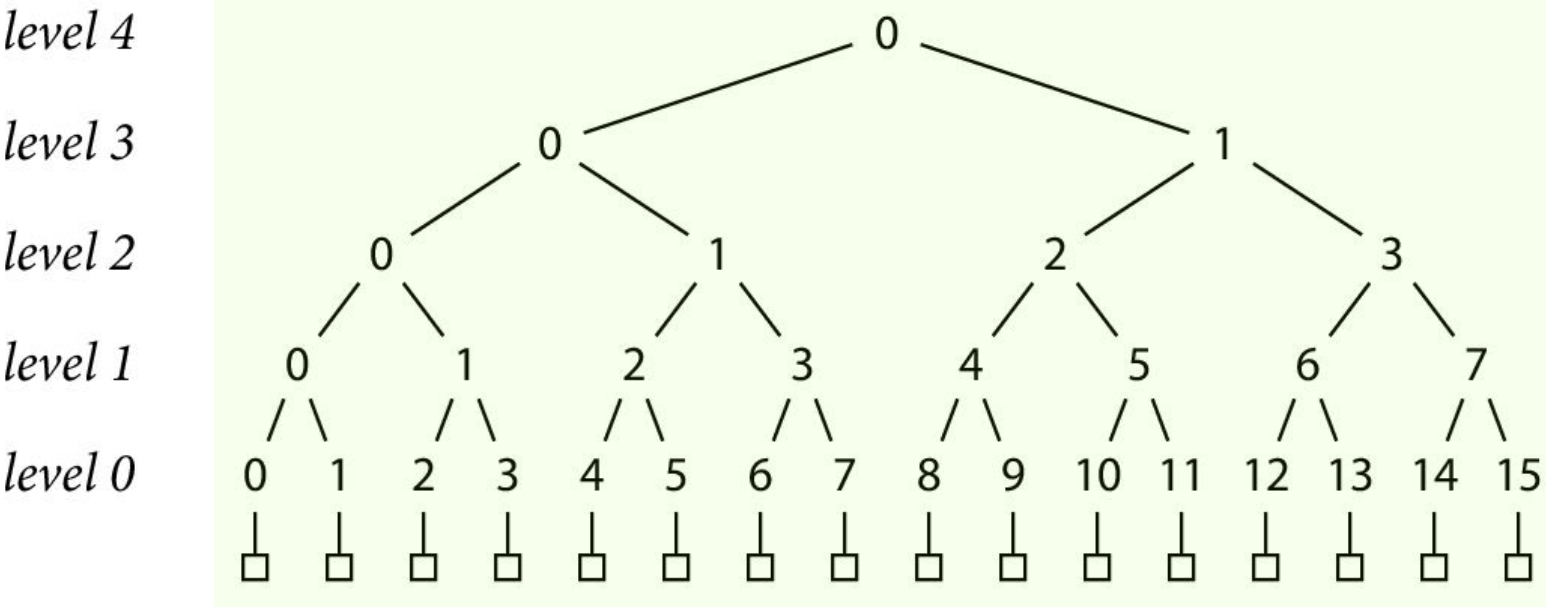

level 4
level 3
level 2
level 1
level 0
0
0 1
0 1 2 3
0 1 2 3 4 5 6 7
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

Checksum DB


level 4
level 3
level 2
level 1
level 0
0
0 1
0 1 2 3
0 1 2 3 4 5 6 7
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

Checksum DB

✓


level 4
level 3
level 2
level 1
level 0
0
0 1
0 1 2 3
0 1 2 3 4 5 6 7
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

Checksum DB

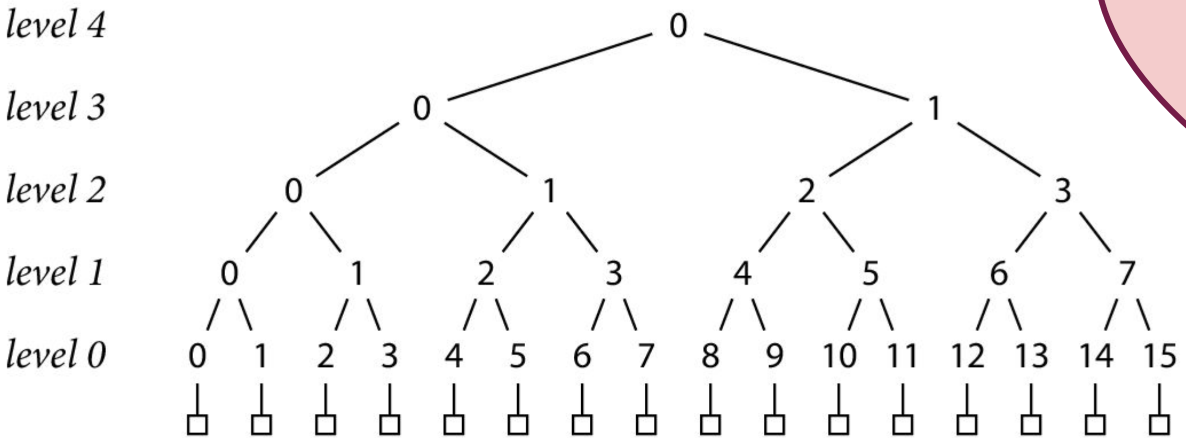
We all want the “correct” code

We all want the “correct” code

=

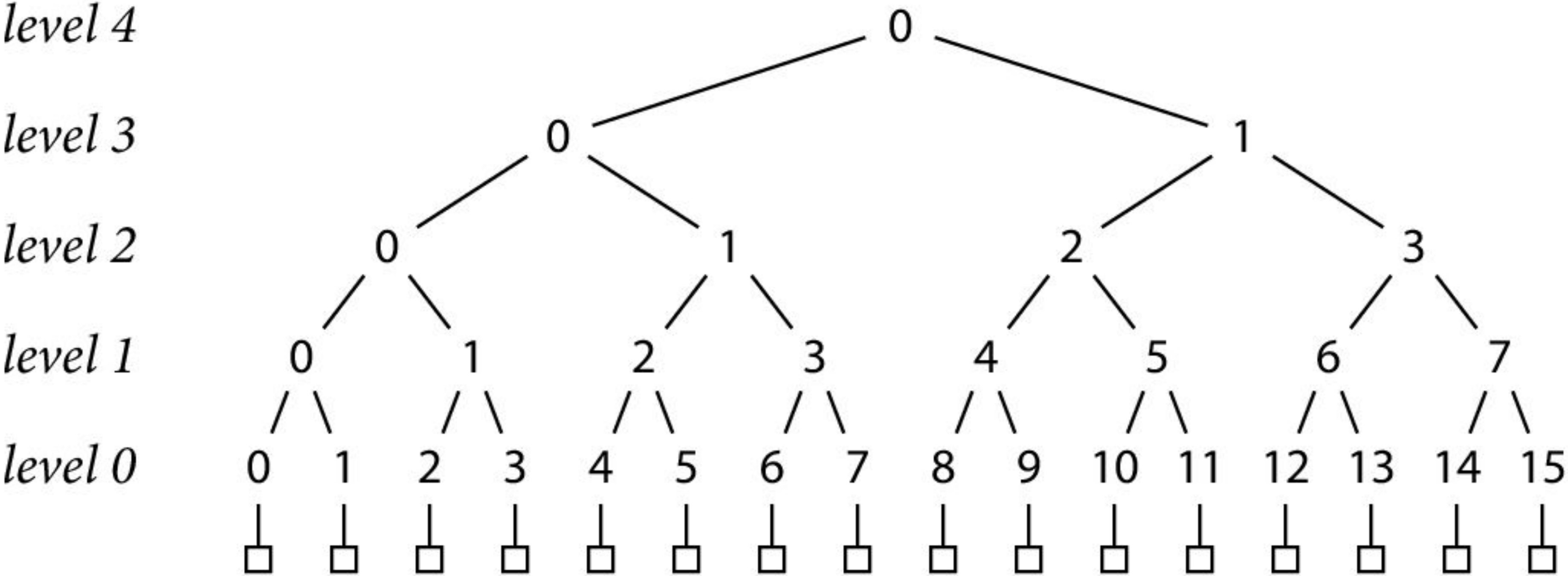
We all want the “same” code

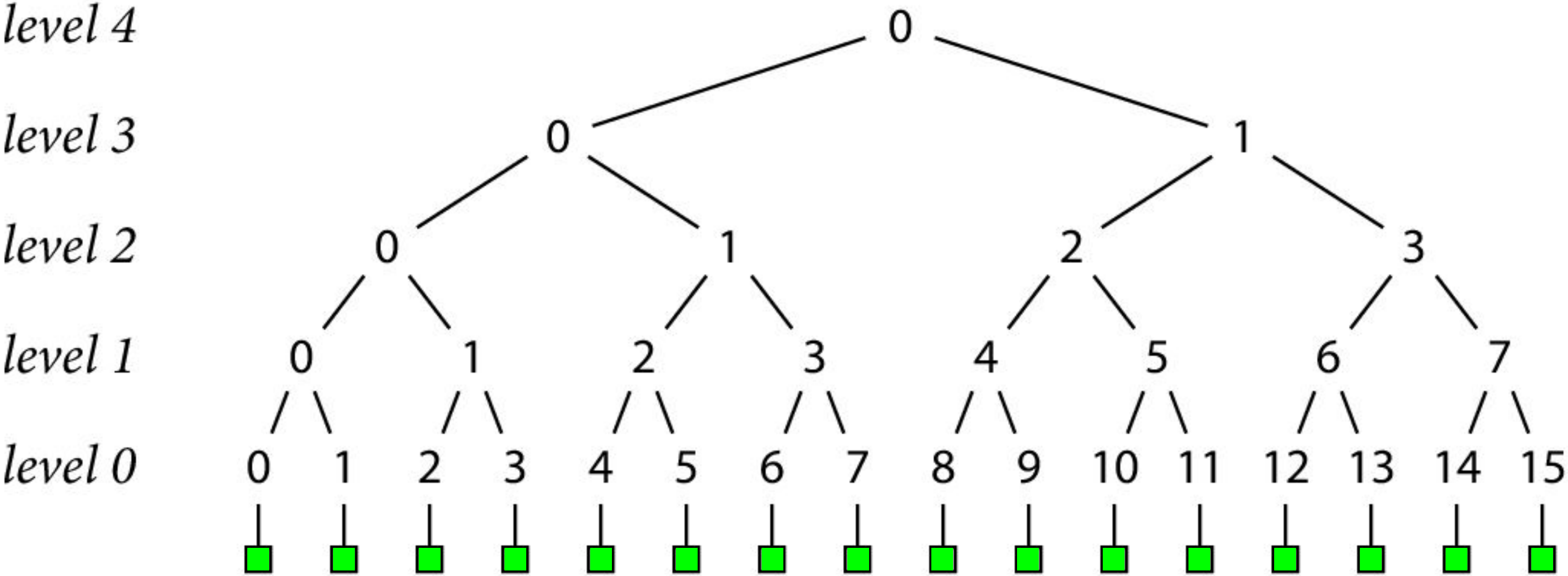
math

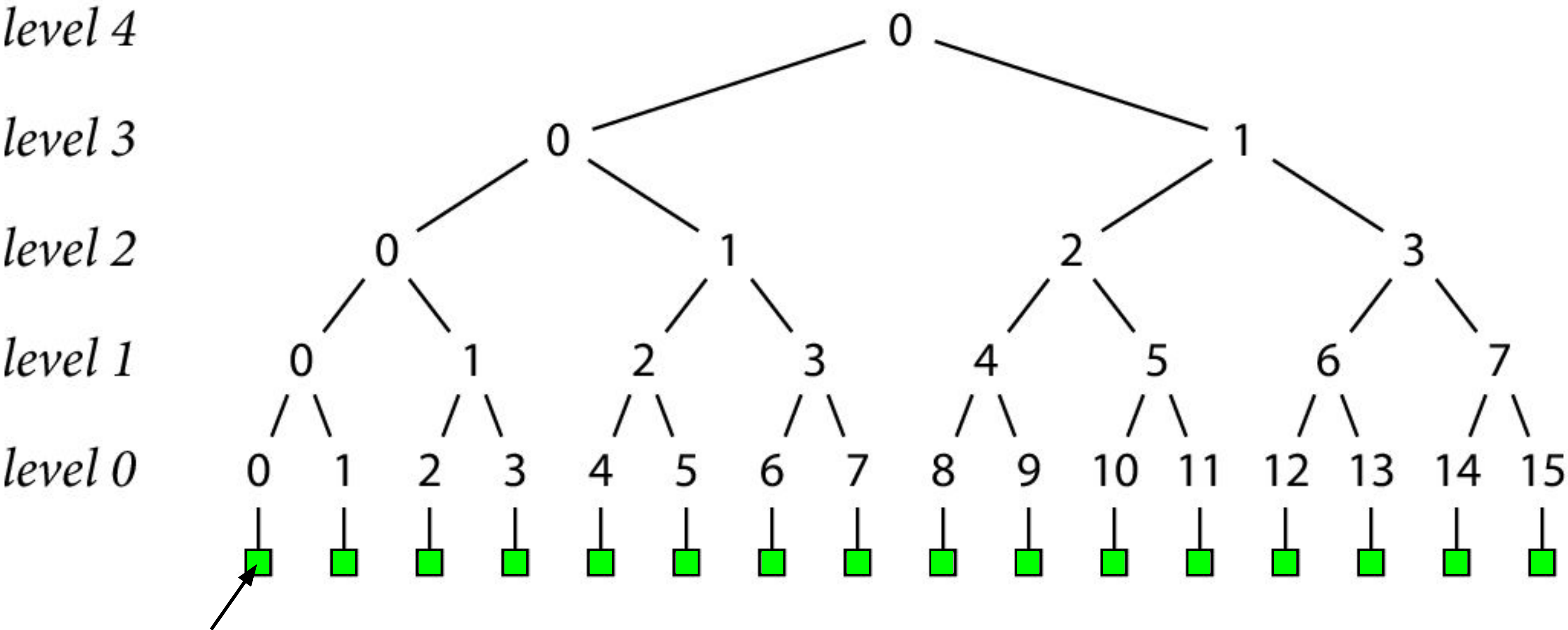


Inclusion proof ✅

Consistency proof ✅

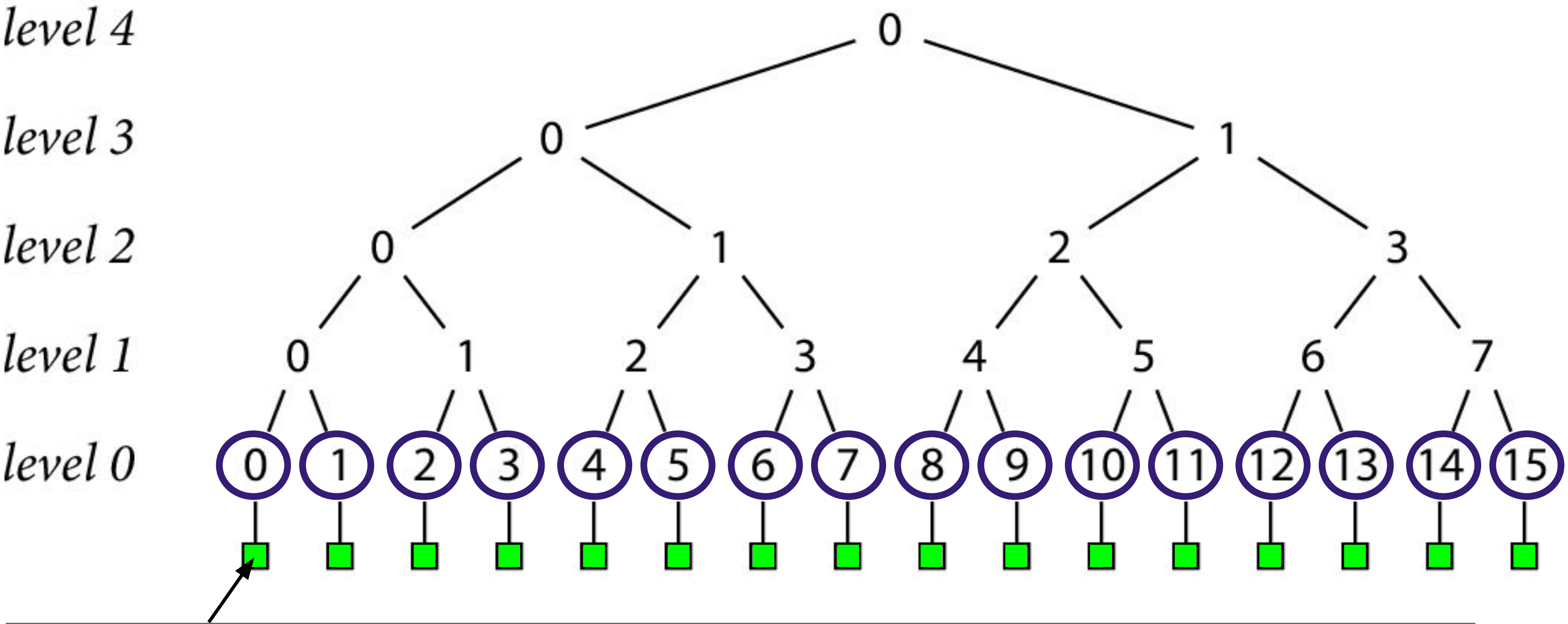

level 4
level 3
level 2
level 1
level 0
0
0
1
0
1
2
3
0
1
2
3
4
5
6
7
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15


level 4
level 3
level 2
level 1
level 0
0
0
1
0
1
2
3
0
1
2
3
4
5
6
7
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15



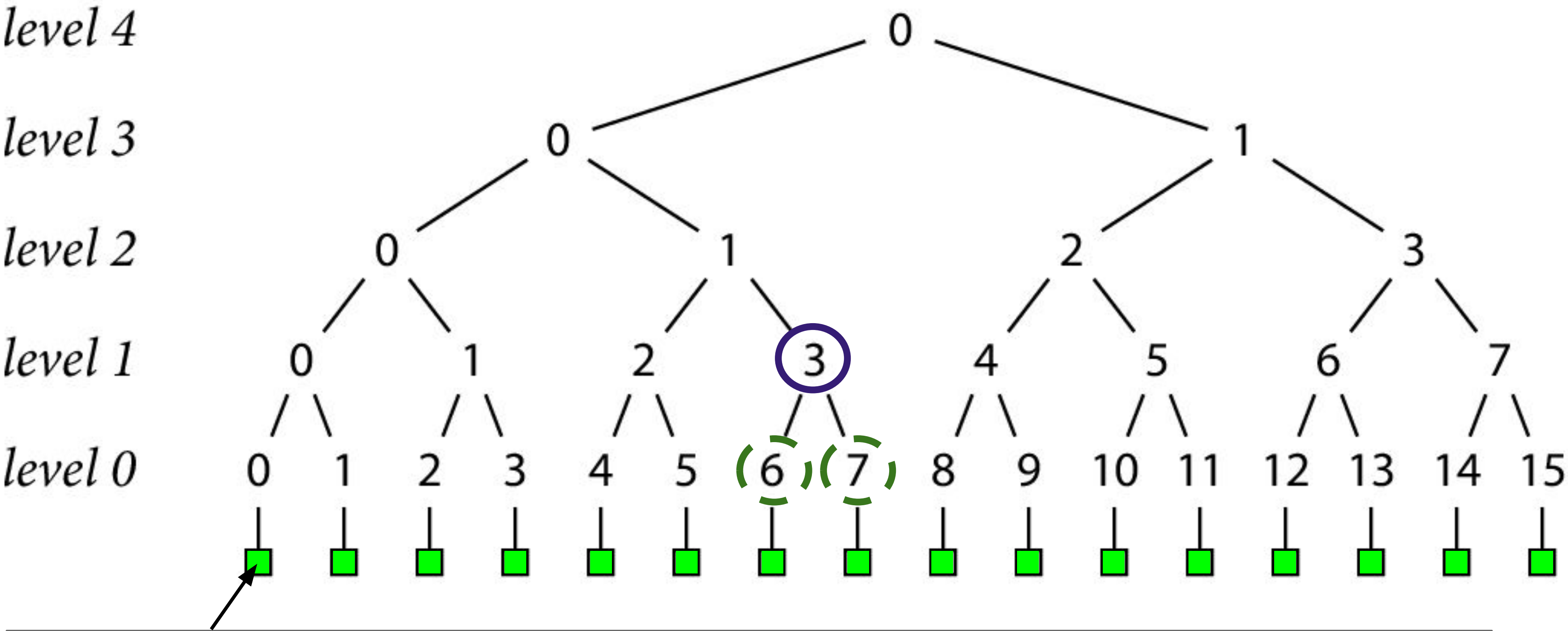
```
go.opencensus.io v0.19.2 h1:ZZpq6xI6kv/LuE/5s5UQvB...
go.opencensus.io v0.19.2/go.mod h1:NO/8qkisMZLZ1FC...
```

Checksum DB



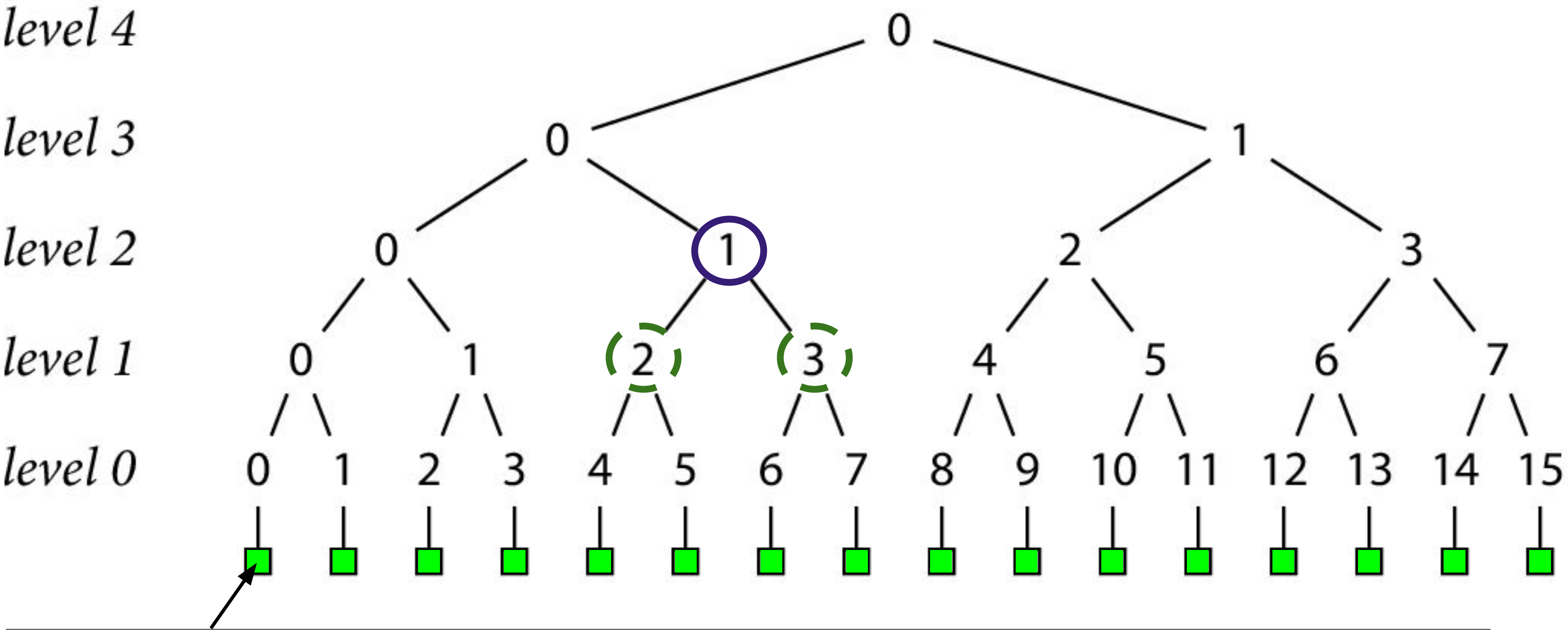
```
go.opencensus.io v0.19.2 h1:ZZpq6xI6kv/LuE/5s5UQvB...

go.opencensus.io v0.19.2/go.mod h1:NO/8qkisMZLZ1FC...
```

Checksum DB



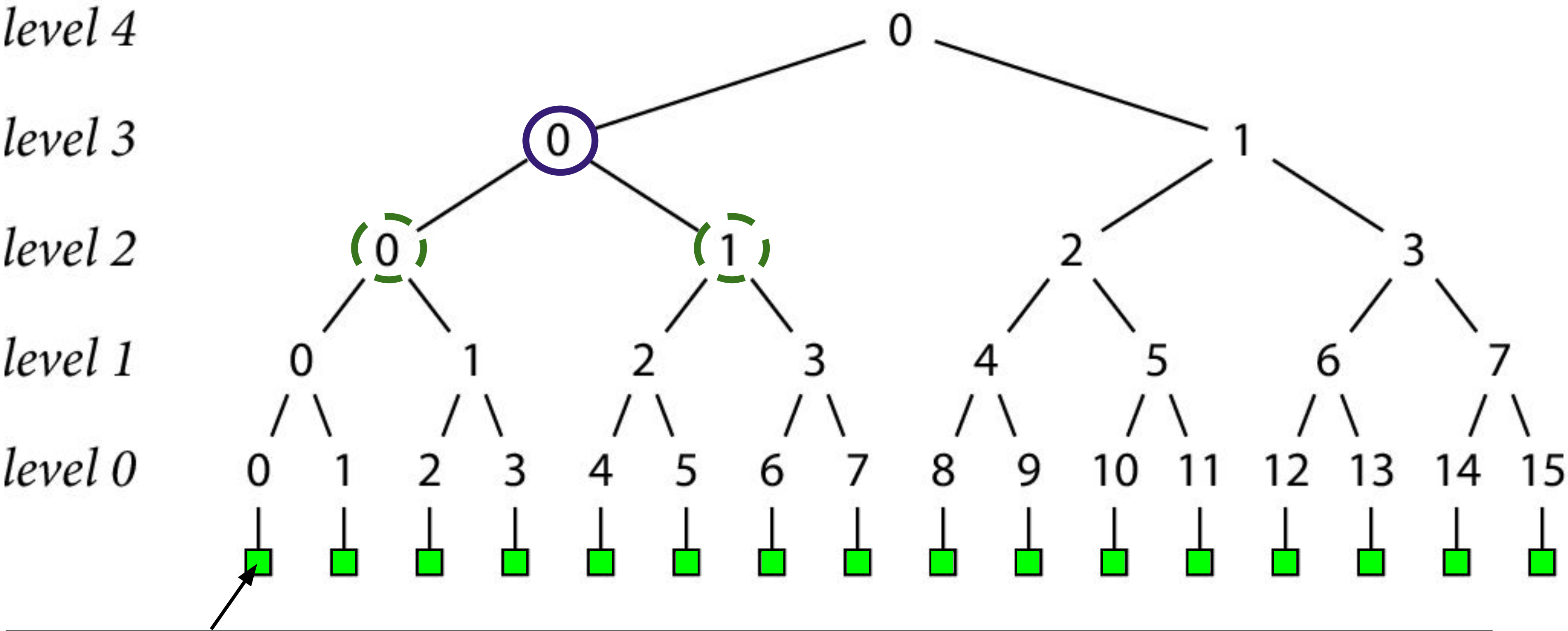
```
go.opencensus.io v0.19.2 h1:ZZpq6xI6kv/LuE/5s5UQvB...

go.opencensus.io v0.19.2/go.mod h1:NO/8qkisMZLZ1FC...
```

Checksum DB



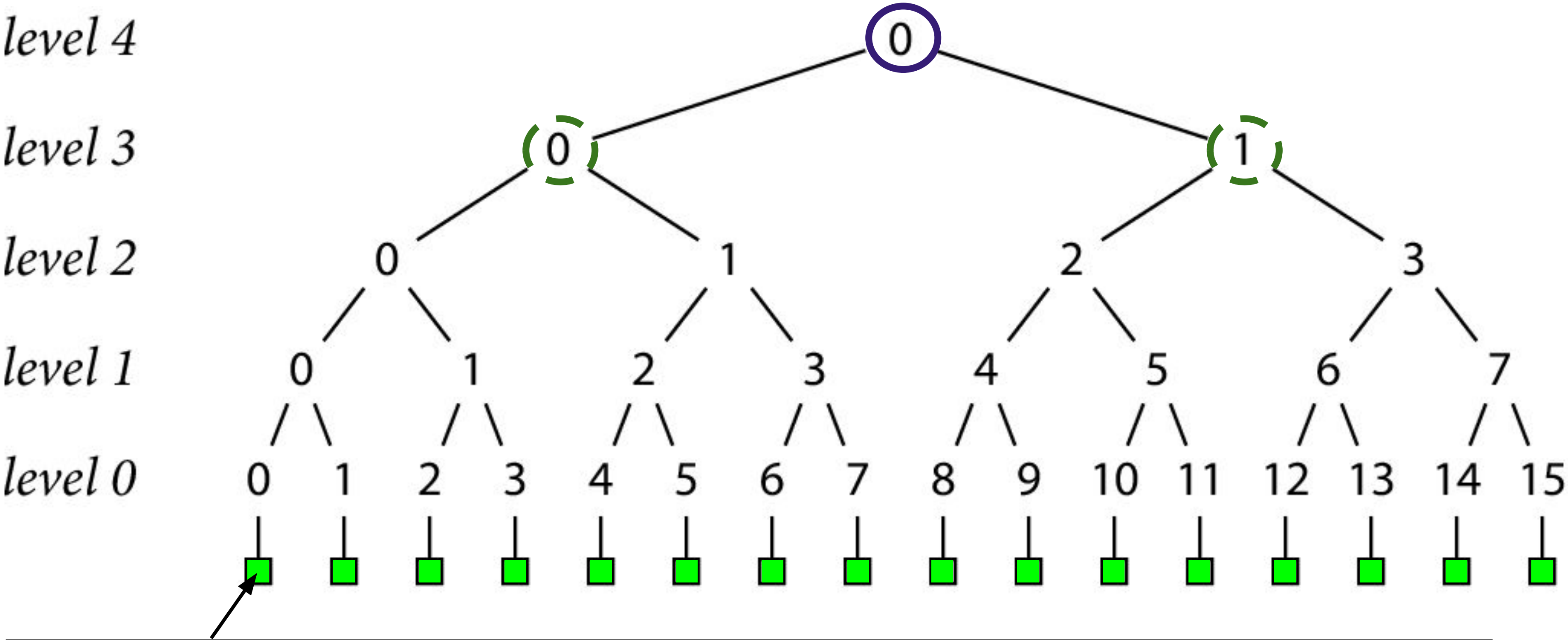
```
go.opencensus.io v0.19.2 h1:ZZpq6xI6kv/LuE/5s5UQvB...
go.opencensus.io v0.19.2/go.mod h1:NO/8qkisMZLZ1FC...
```

Checksum DB



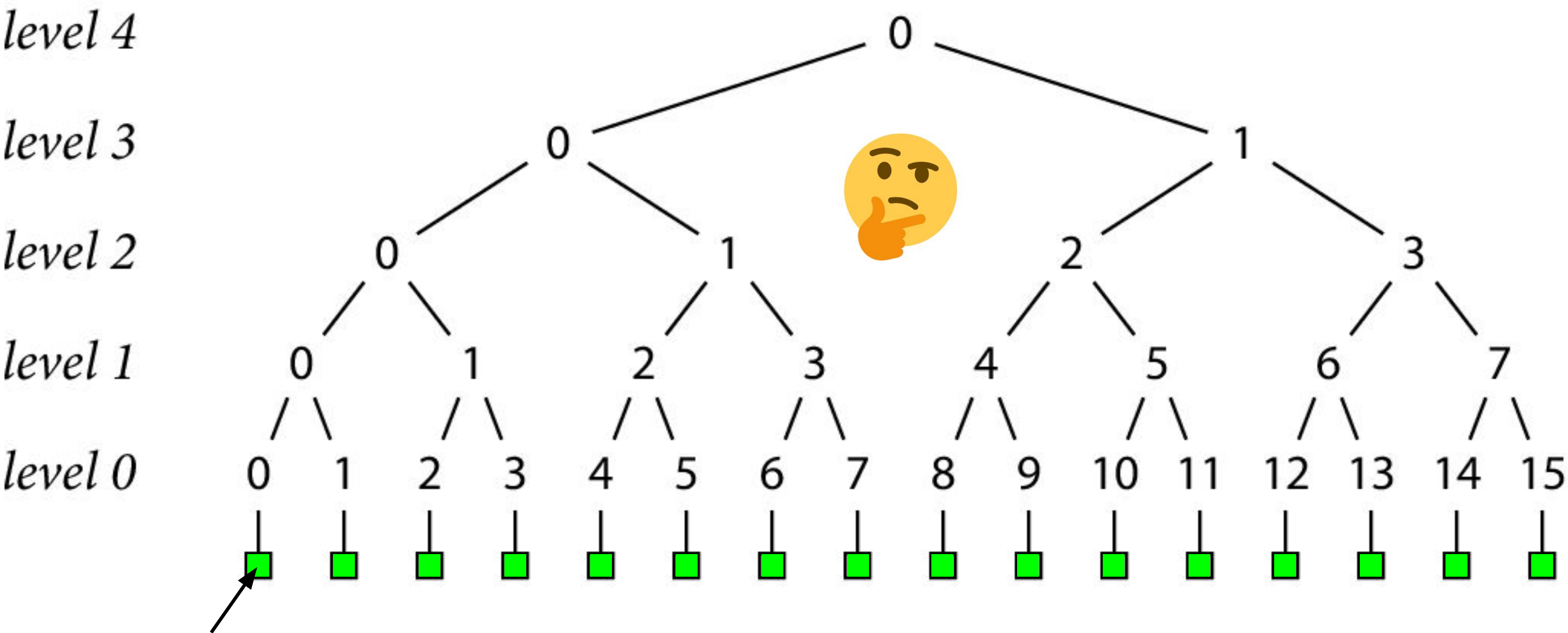
```
go.opencensus.io v0.19.2 h1:ZZpq6xI6kv/LuE/5s5UQvB...
go.opencensus.io v0.19.2/go.mod h1:NO/8qkisMZLZ1FC...
```

Checksum DB



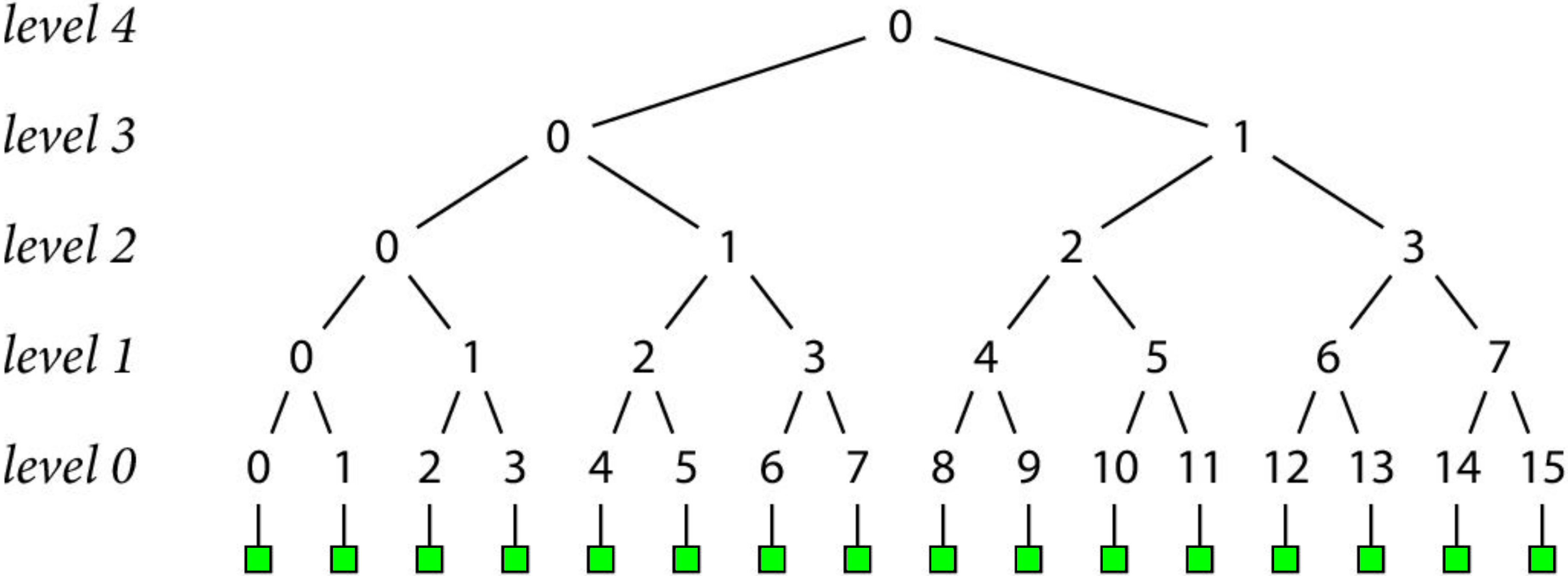
```
go.opencensus.io v0.19.2 h1:ZZpq6xI6kv/LuE/5s5UQvB...
go.opencensus.io v0.19.2/go.mod h1:NO/8qkisMZLZ1FC...
```

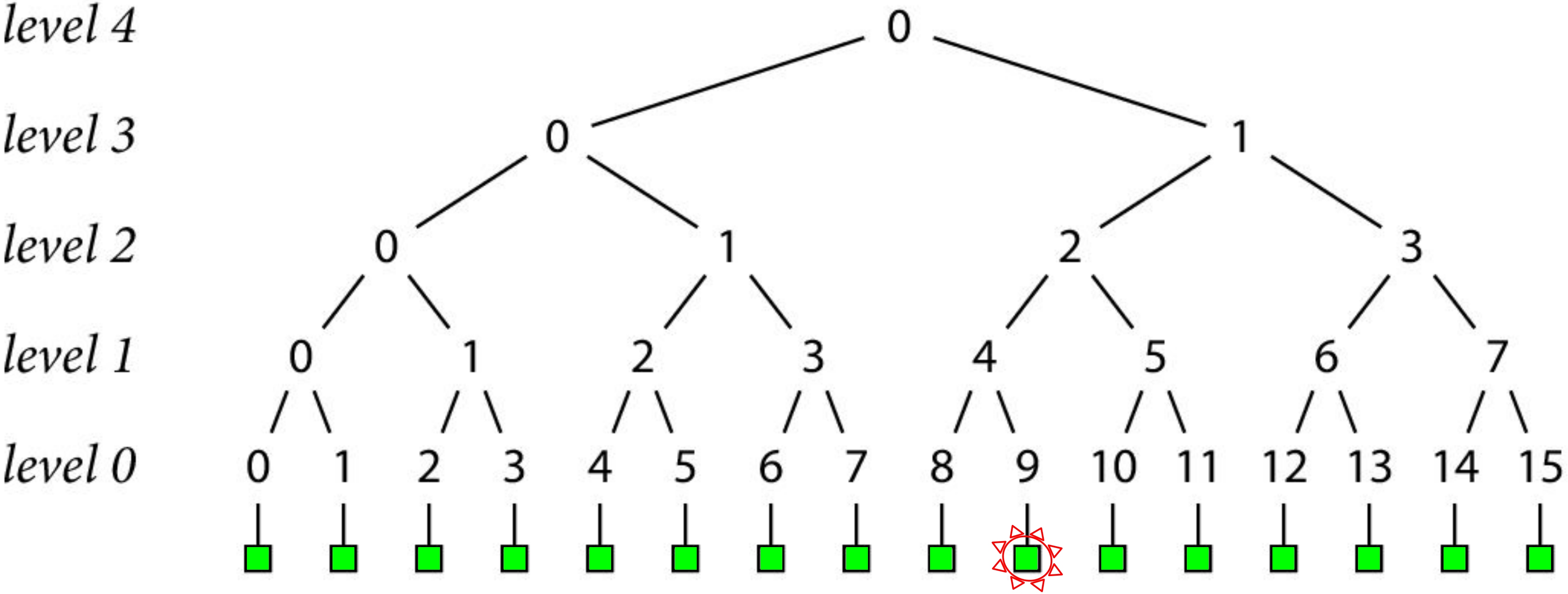
Checksum DB



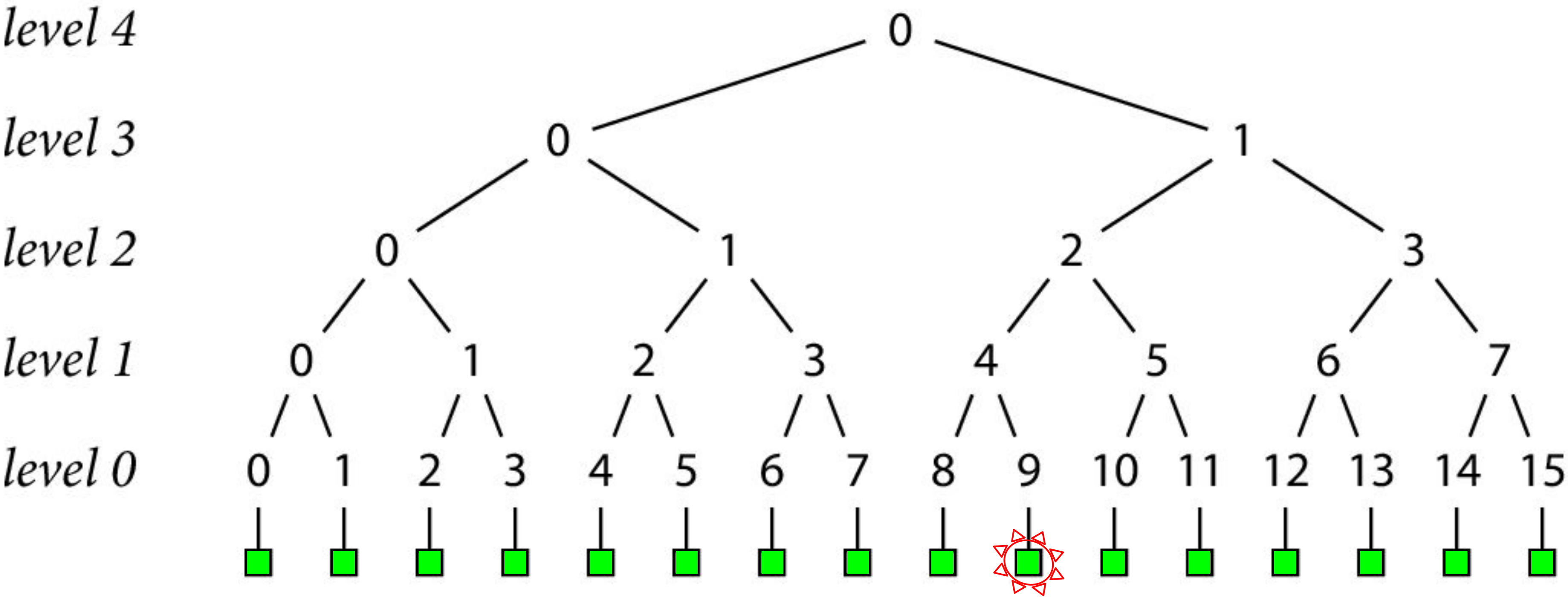
```
go.opencensus.io v0.19.2 h1:ZZpq6xI6kv/LuE/5s5UQvB...
go.opencensus.io v0.19.2/go.mod h1:NO/8qkisMZLZ1FC...
```

Checksum DB


level 4
0
level 3
0
1
level 2
0
1
2
3
level 1
0
1
2
3
4
5
6
7
level 0
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15


level 4
level 3
level 2
level 1
level 0
0
0
1
0
1
2
3
0
1
2
3
4
5
6
7
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15

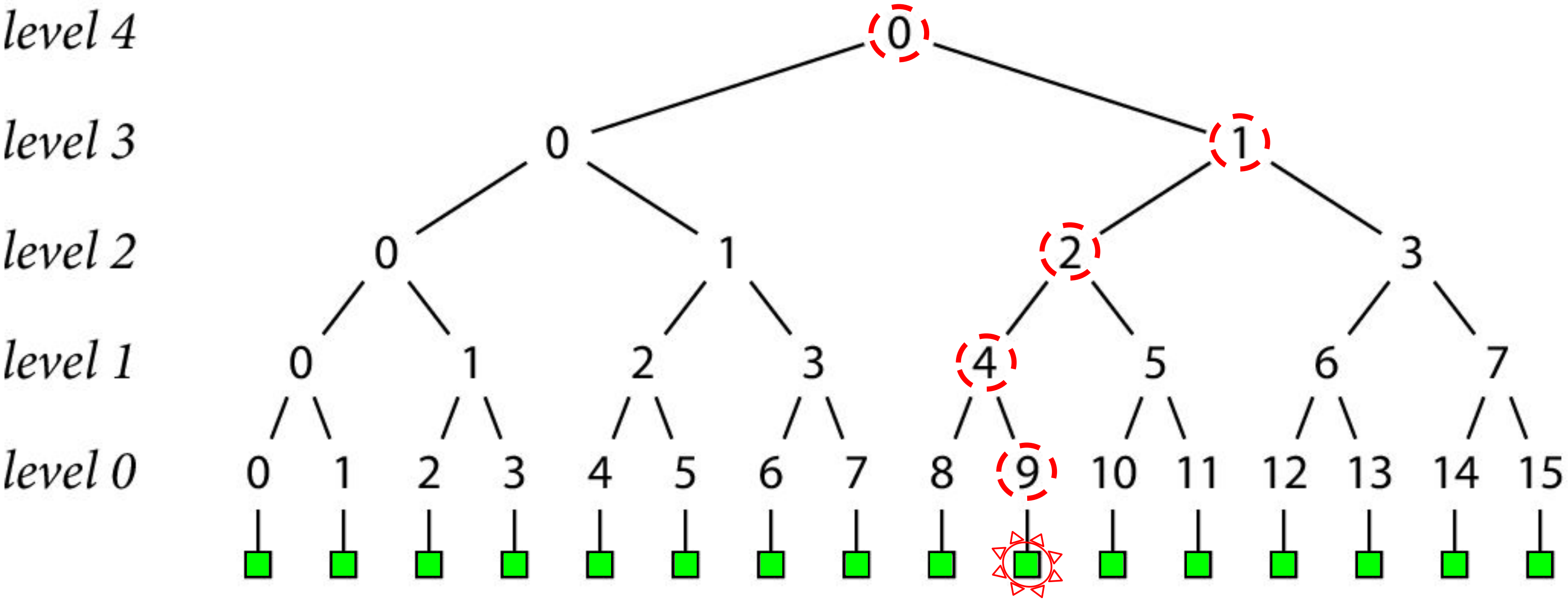
Checksum DB



`/lookup/go.dog/breeds@v0.3.2`

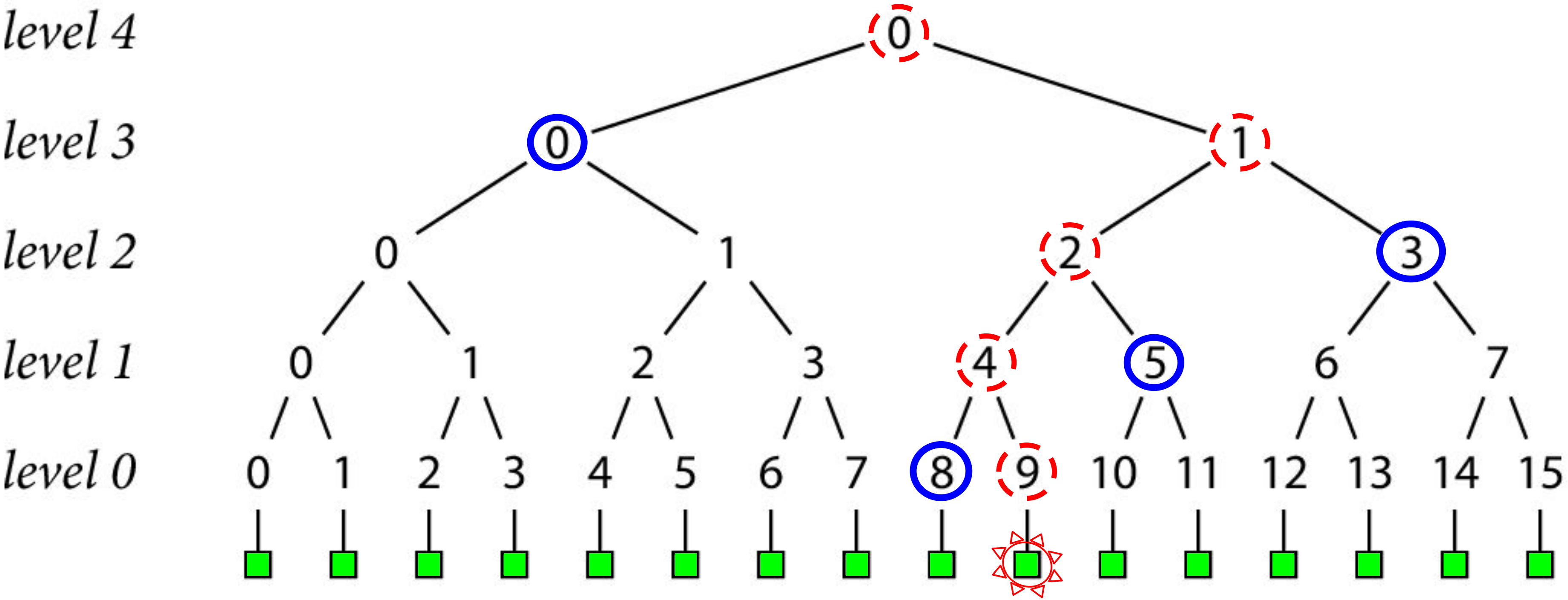
1. record number (9)
2. go.sum lines
3. tree head

Checksum DB



`/lookup/go.dog/breeds@v0.3.2`

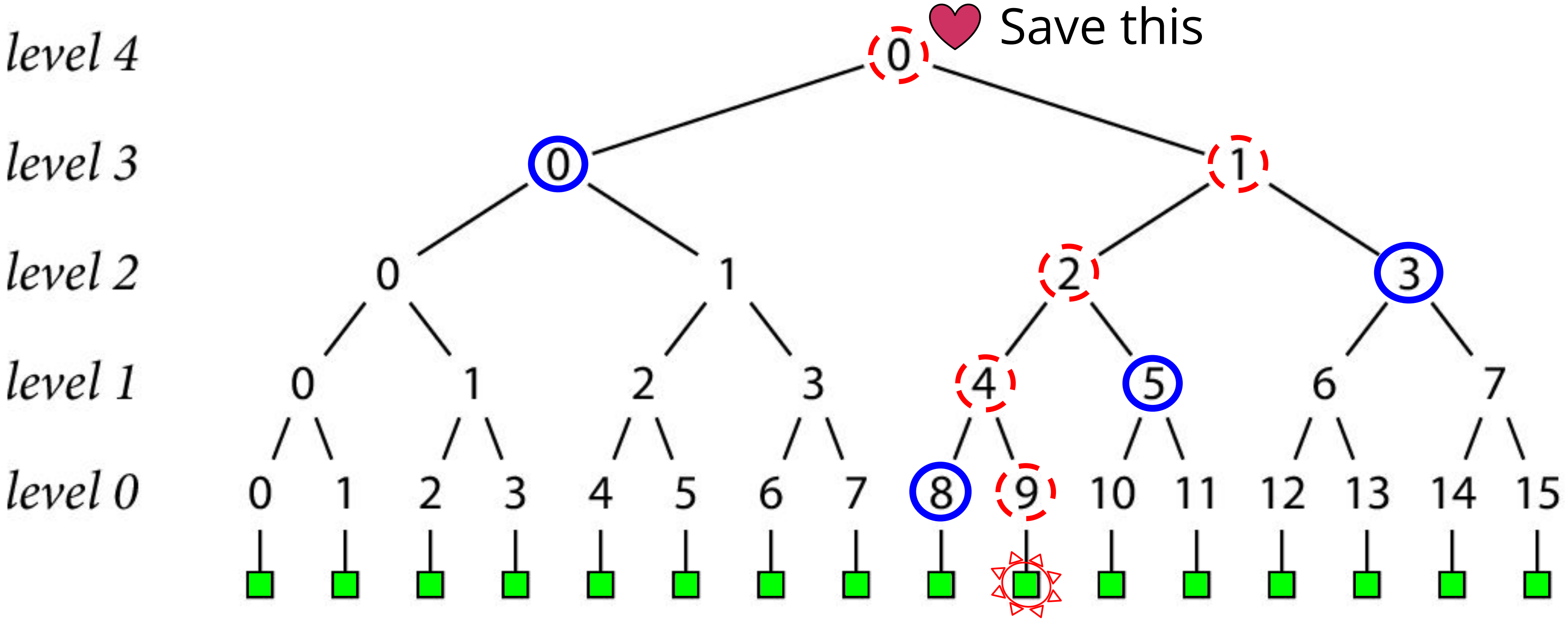
1. record number (9)
2. go.sum lines
3. tree head



/lookup/go.dog/breeds@v0.3.2

1. record number (9)
2. go.sum lines
3. tree head

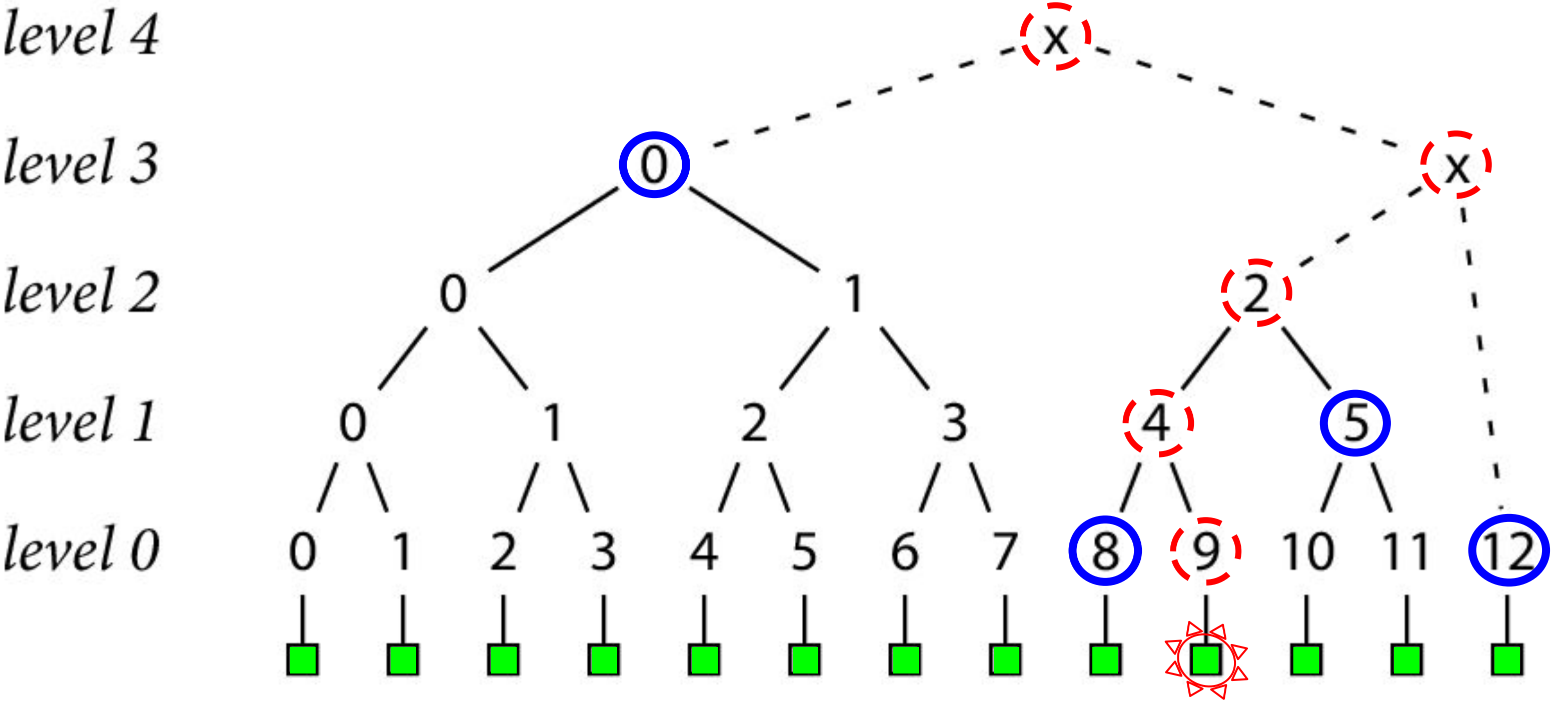
Checksum DB



/lookup/go.dog/breeds@v0.3.2

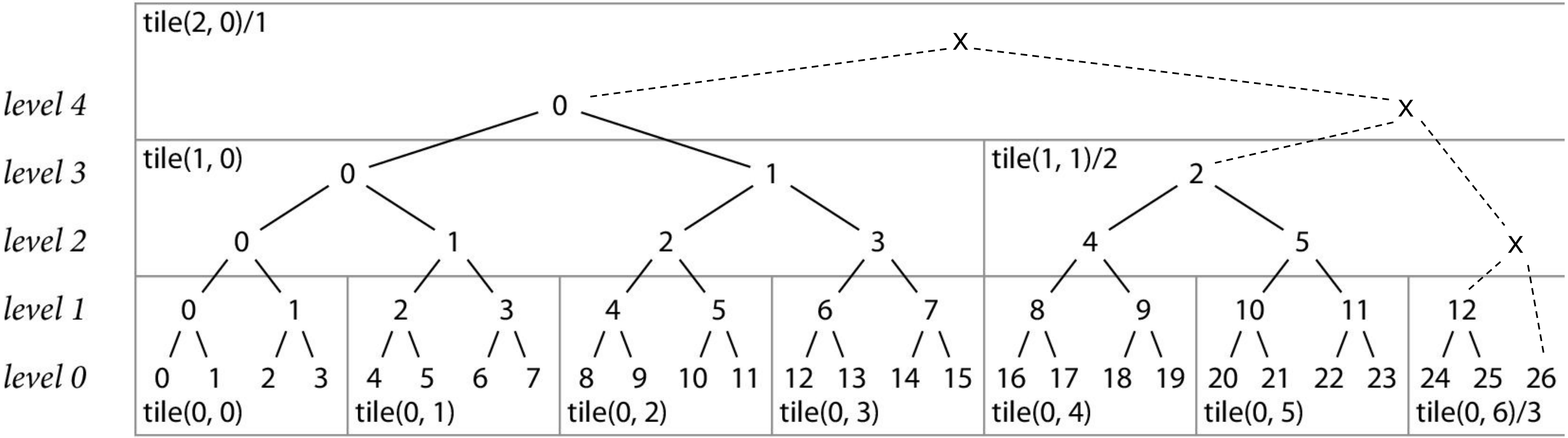
1. record number (9)
2. go.sum lines
3. tree head

Checksum DB

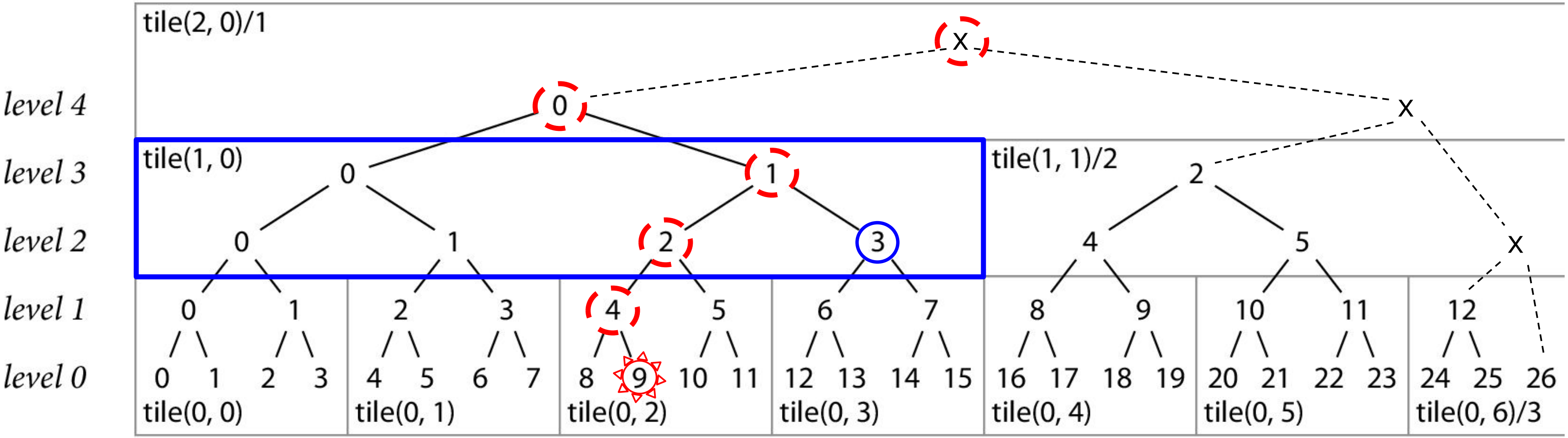

level 4
level 3
level 2
level 1
level 0
x
0
x
0
1
2
0
1
2
3
4
5
0
1
2
3
4
5
6
7
8
9
10
11
12

Checksum DB

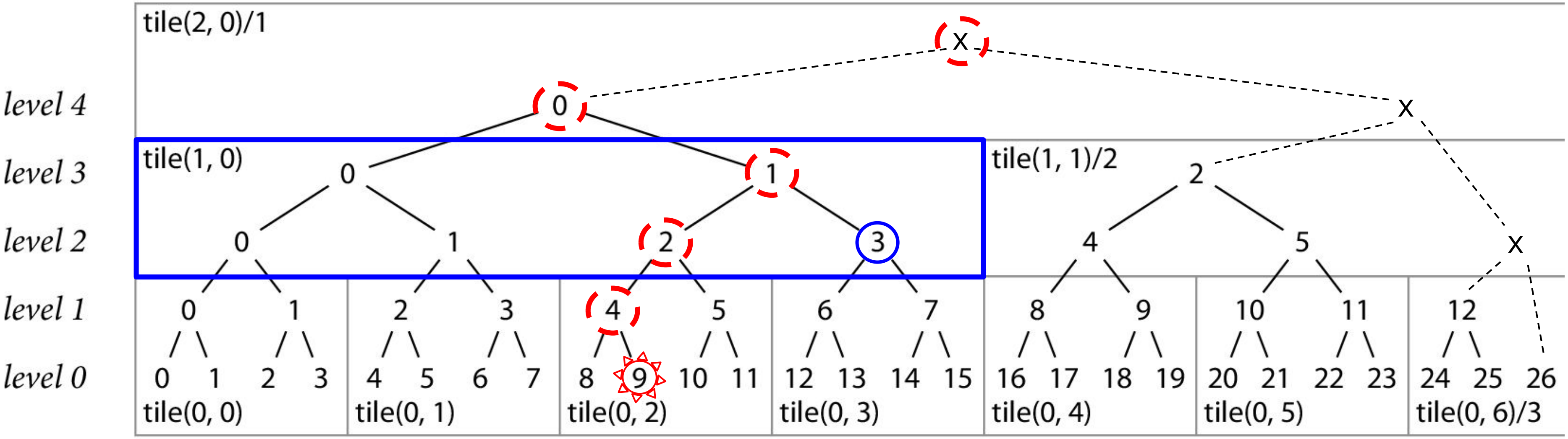
Tiles!


tile(2, 0)/1
x
level 4
0
x
level 3
tile(1, 0)
0
1
tile(1, 1)/2
2
level 2
0
1
2
3
4
5
x
level 1
0
1
2
3
4
5
6
7
8
9
10
11
12
level 0
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
tile(0, 0)
tile(0, 1)
tile(0, 2)
tile(0, 3)
tile(0, 4)
tile(0, 5)
tile(0, 6)/3

Checksum DB


tile(2, 0)/1
x
level 4
0
x
level 3
tile(1, 0)
0
1
tile(1, 1)/2
2
level 2
0
1
2
3
4
5
x
level 1
0
1
2
3
4
5
6
7
8
9
10
11
12
level 0
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
tile(0, 0)
tile(0, 1)
tile(0, 2)
tile(0, 3)
tile(0, 4)
tile(0, 5)
tile(0, 6)/3

Checksum DB



✓ Cache friendly

✓ Less storage

Checksum Database spec

`GET $GOSUMDB/lookup/M@V` returns the 1) record # of the module version, 2) its go.sum lines, and 3) a tree head.

`GET $GOSUMDB/tile/H/L/K[.p/W]`

`GET $GOSUMDB/tile/H/data/K[.p/W]` returns record data

<u>For auditors:</u> `GET $GOSUMDB/latest` returns the latest signed tree head for a log.

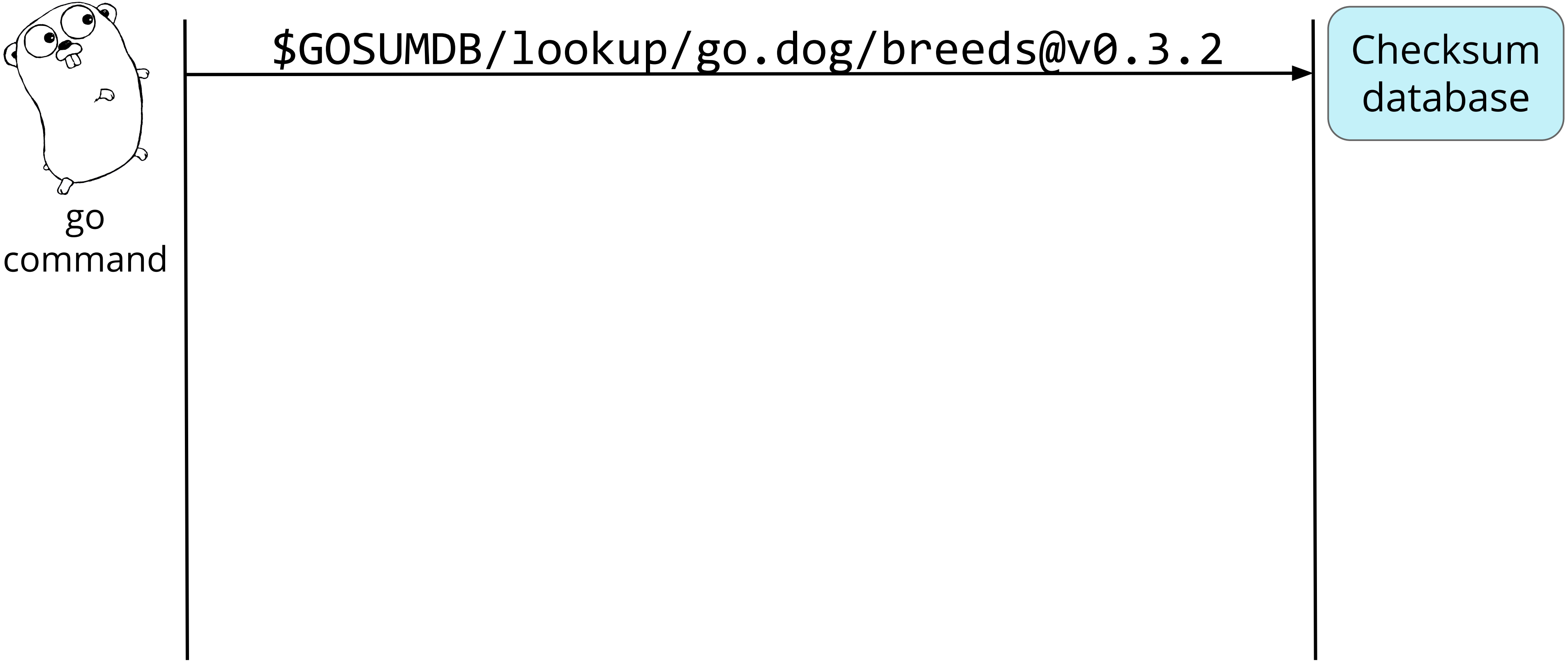
/latest

```
go.sum database tree
11131
uJ+UzAfOs0Gi22MuICyemMge4fp8gI3+5+wZF5t1aXU=

— sum.golang.org
Az3grjQmIfBoZv0eHleRf4gu7VNSS8LvsdfECddEnVIA1drPr
g29LDmVliD3dshfo9ZYr1Gq0HWyVWXJ96ZjDpiSvgY=
go get go.dog/breeds
```

go command

Checksum database

```
go get go.dog/breeds
```

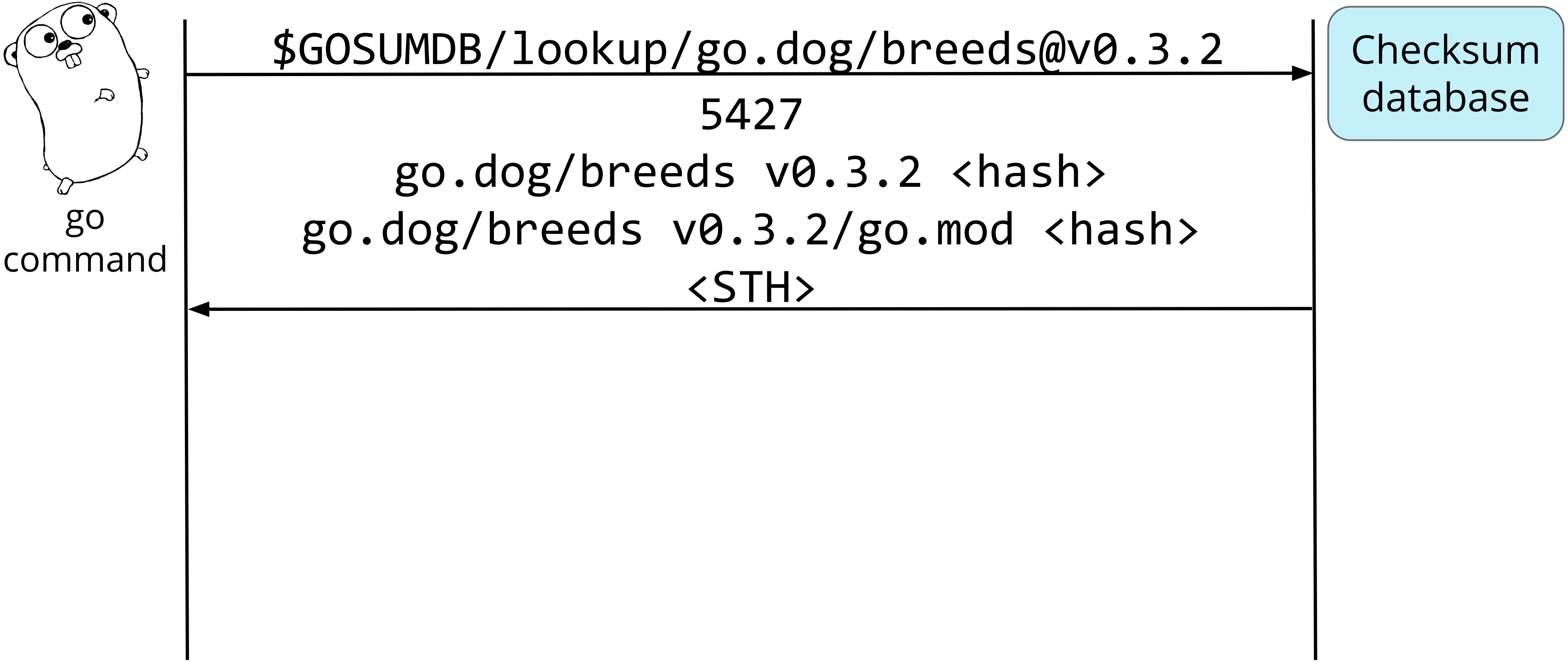


go command

`$GOSUMDB/lookup/go.dog/breeds@v0.3.2`

Checksum database

go get go.dog/breeds



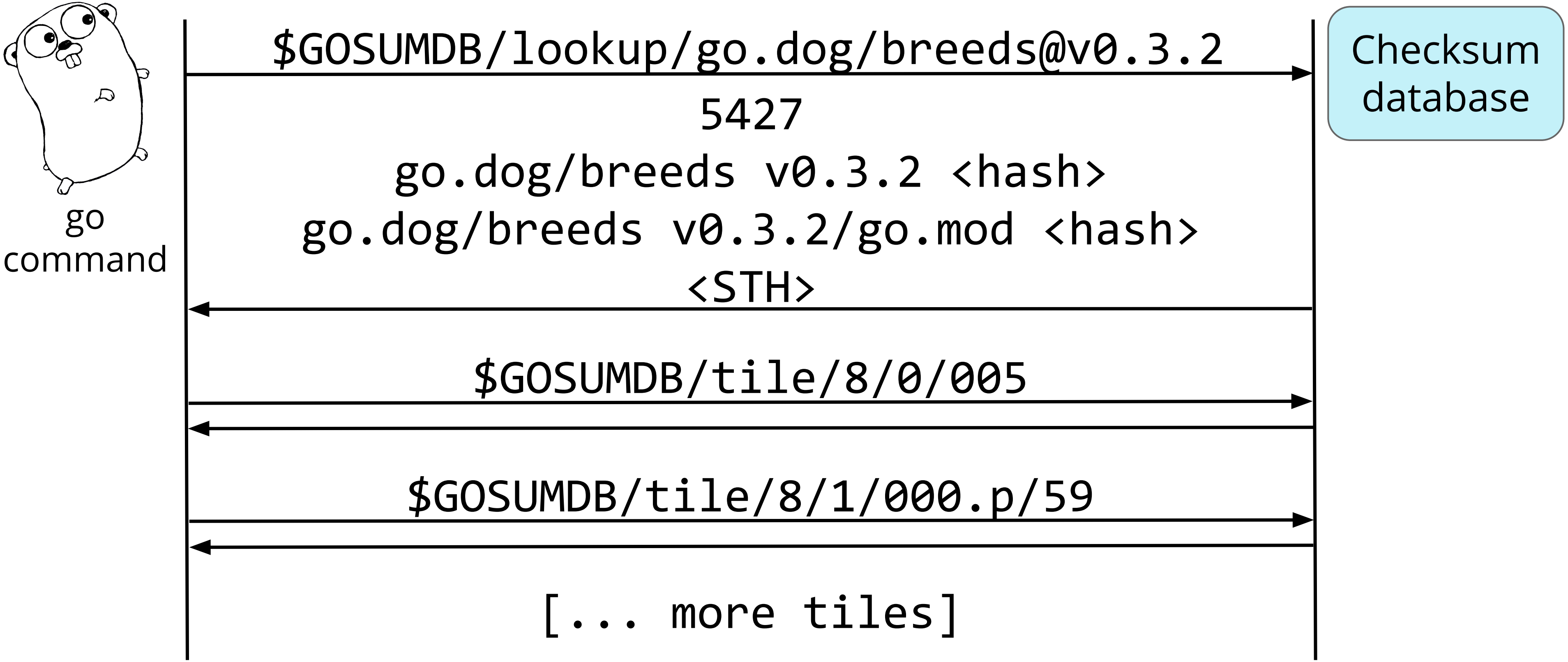
go command

Checksum database

$GOSUMDB/lookup/go.dog/breeds@v0.3.2

5427
go.dog/breeds v0.3.2 <hash>
go.dog/breeds v0.3.2/go.mod <hash>
<STH>

go get go.dog/breeds



go command

Checksum database

$GOSUMDB/lookup/go.dog/breeds@v0.3.2

5427
go.dog/breeds v0.3.2 <hash>
go.dog/breeds v0.3.2/go.mod <hash>
<STH>

$GOSUMDB/tile/8/0/005

$GOSUMDB/tile/8/1/000.p/59

[... more tiles]

“Trust on ~~your~~ first use”

anyone’s

Why use the checksum database?

- ✓ Validates untrusted proxies

Why use the checksum database?

- ✓ Validates untrusted proxies
- ✓ Mitigates hacked origins

Why use the checksum database?

✓ Validates untrusted proxies

✓ Mitigates hacked origins

✓ Prevents author error

Why use the checksum database?

✓ Validates untrusted proxies

✓ Mitigates hacked origins

✓ Prevents author error

✓ Easy to use

Better solutions

✓	Reproducible builds	Modules
✓	Persistent dependencies	Mirror
✓	Trustworthy fetches	Checksum Database

Setting up your environment

On by default in Go 1.13

Environment variables

- GO111MODULE
    - GO111MODULE="on"
- GOPROXY
    - GOPROXY="https://proxy.golang.org,direct"

For private modules

- GOPRIVATE
  - GOPRIVATE="*.corp.google.com,rsc.io/private"

Trillian

https://github.com/google/trillian

A transparent, highly scalable and cryptographically verifiable data store.

One more cool thing

index.golang.org/index

- Feed of new modules discovered by proxy.golang.org

https://index.golang.org/index?since=2019-03-04T18:00:15.161182-07:00

The future of Go

The future of Go