Distributed Systems at GitHub

CSE 452 (23wi)

Taylor Blau

Taylor Blau

Staff Software Engineer at GitHub

Now: upstream Git, “Git scale”, etc.
Past: UW CSE ‘20

CSE 452

- Took this class in undergrad
- Extremely fond memories of lab3 (not so much lab4)
- Great mix of concepts and practical details
- Today’s goal: highlight some examples from lecture(s) and connect them to real systems at GitHub

Today’s agenda

Sharding / routing

Repository replication, routing, and balancing

Remote procedure calls

Querying Git content as part of a web request

3-phase commit (3PC)

Linearizing reference (branch/tag) updates across repository replicas

Sharding / routing

Repository replication, routing, and balancing

Sharding / routing

- GitHub stores Git repositories by their `.git` directory (AKA a “bare” repository)
- Plain-old Git on the backend (mostly)
- Several copies of a repository are stored
  - Usually 5, across 3 data-centers
  - Sometimes more for read- or write-heavy repositories
- Each replica has a “read weight” (sums to 100 within a DC)

Sharding / routing

```
ttaylorr@spokes-shell-6ec44b1.ac4-iad(prd) ~ $ spokesctl dat github/github
+----------------------------------------+--------+-----+----------+------+
|                 HOST                   | STATE  | R W | REPL SUM |  ?   |
+----------------------------------------+--------+-----+----------+------+
| github-dfs-a1d46af.ac4-iad.github.net  | ACTIVE | 50  | 5:141ff  | [OK] |
| github-dfs-a7baec3.ac4-iad.github.net  | ACTIVE | 50  | 5:141ff  | [OK] |
| github-dfs-455915e.va3-iad.github.net  | ACTIVE | 33  | 5:141ff  | [OK] |
| github-dfs-542aa72.va3-iad.github.net  | ACTIVE | 33  | 5:141ff  | [OK] |
| github-dfs-7e18f50.va3-iad.github.net  | ACTIVE | 33  | 5:141ff  | [OK] |
| github-dfs-32f9372.ash1-iad.github.net | ACTIVE | 50  | 5:141ff  | [OK] |
| github-dfs-b94d3d5.ash1-iad.github.net | ACTIVE | 50  | 5:141ff  | [OK] |
+----------------------------------------+--------+-----+----------+------+

repo: 3/3 (github/github)
repo_sum: 5:141ff92bd1eb53355f3cab7569fe68facf30b91b
path: /data/repositories/e/nw/ec/cb/c8/3/3.git
storage policy: Read-Heavy (created: 2020-11-30T15:10:32Z updated:
2022-01-27T09:05:04Z)
```


Sharding / routing

```
ttaylorr@spokes-shell-6ec44b1.ac4-iad(prd) ~ $ spokesctl dat github/github
+-----------------------------------------+--------+-----+----------+------+
|                  HOST                   | STATE  | R W | REPL SUM |  ?   |
+-----------------------------------------+--------+-----+----------+------+
| github-dfs-a1d46af.ac4-iad.github.net   | ACTIVE | 50  | 5:141ff  | [OK] |
| github-dfs-a7baec3.ac4-iad.github.net   | ACTIVE | 50  | 5:141ff  | [OK] |
| github-dfs-455915e.va3-iad.github.net   | ACTIVE | 33  | 5:141ff  | [OK] |
| github-dfs-542aa72.va3-iad.github.net   | ACTIVE | 33  | 5:141ff  | [OK] |
| github-dfs-7e18f50.va3-iad.github.net   | ACTIVE | 33  | 5:141ff  | [OK] |
| github-dfs-32f9372.ash1-iad.github.net  | ACTIVE | 50  | 5:141ff  | [OK] |
| github-dfs-b94d3d5.ash1-iad.github.net  | ACTIVE | 50  | 5:141ff  | [OK] |
+-----------------------------------------+--------+-----+----------+------+

repo: 3/3 (github/github)
repo_sum: 5:141ff92bd1eb53355f3cab7569fe68facf30b91b
path: /data/repositories/e/nw/ec/cb/c8/3/3.git
storage policy: Read-Heavy (created: 2020-11-30T15:10:32Z updated:
2022-01-27T09:05:04Z)
```


Sharding / routing

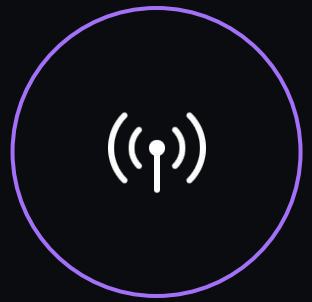
```
ttaylorr@spokes-shell-6ec44b1.ac4-iad(prd) ~ $ spokesctl dat github/github
+-----------------------------------------+--------+-----+----------+------+
|                  HOST                   | STATE  | R W | REPL SUM |  ?   |
+-----------------------------------------+--------+-----+----------+------+
| github-dfs-a1d46af.ac4-iad.github.net   | ACTIVE | 50  | 5:141ff  | [OK] |
| github-dfs-a7baec3.ac4-iad.github.net   | ACTIVE | 50  | 5:141ff  | [OK] |
| github-dfs-455915e.va3-iad.github.net   | ACTIVE | 33  | 5:141ff  | [OK] |
| github-dfs-542aa72.va3-iad.github.net   | ACTIVE | 33  | 5:141ff  | [OK] |
| github-dfs-7e18f50.va3-iad.github.net   | ACTIVE | 33  | 5:141ff  | [OK] |
| github-dfs-32f9372.ash1-iad.github.net  | ACTIVE | 50  | 5:141ff  | [OK] |
| github-dfs-b94d3d5.ash1-iad.github.net  | ACTIVE | 50  | 5:141ff  | [OK] |
+-----------------------------------------+--------+-----+----------+------+

repo: 3/3 (github/github)
repo_sum: 5:141ff92bd1eb53355f3cab7569fe68facf30b91b
path: /data/repositories/e/nw/ec/cb/c8/3/3.git
storage policy: Read-Heavy (created: 2020-11-30T15:10:32Z updated:
2022-01-27T09:05:04Z)
```


Sharding / routing

```
ttaylorr@spokes-shell-6ec44b1.ac4-iad(prd) ~ $ spokesctl dat github/github
+-----------------------------------------+--------+-----+----------+------+
|                  HOST                   | STATE  | R W | REPL SUM |  ?   |
+-----------------------------------------+--------+-----+----------+------+
| github-dfs-a1d46af.ac4-iad.github.net   | ACTIVE | 50  | 5:141ff  | [OK] |
| github-dfs-a7baec3.ac4-iad.github.net   | ACTIVE | 50  | 5:141ff  | [OK] |
| github-dfs-455915e.va3-iad.github.net   | ACTIVE | 33  | 5:141ff  | [OK] |
| github-dfs-542aa72.va3-iad.github.net   | ACTIVE | 33  | 5:141ff  | [OK] |
| github-dfs-7e18f50.va3-iad.github.net   | ACTIVE | 33  | 5:141ff  | [OK] |
| github-dfs-32f9372.ash1-iad.github.net  | ACTIVE | 50  | 5:141ff  | [OK] |
| github-dfs-b94d3d5.ash1-iad.github.net  | ACTIVE | 50  | 5:141ff  | [OK] |
+-----------------------------------------+--------+-----+----------+------+

repo: 3/3 (github/github)
repo_sum: 5:141ff92bd1eb53355f3cab7569fe68facf30b91b
path: /data/repositories/e/nw/ec/cb/c8/3/3.git
storage policy: Read-Heavy (created: 2020-11-30T15:10:32Z updated:
2022-01-27T09:05:04Z)
```

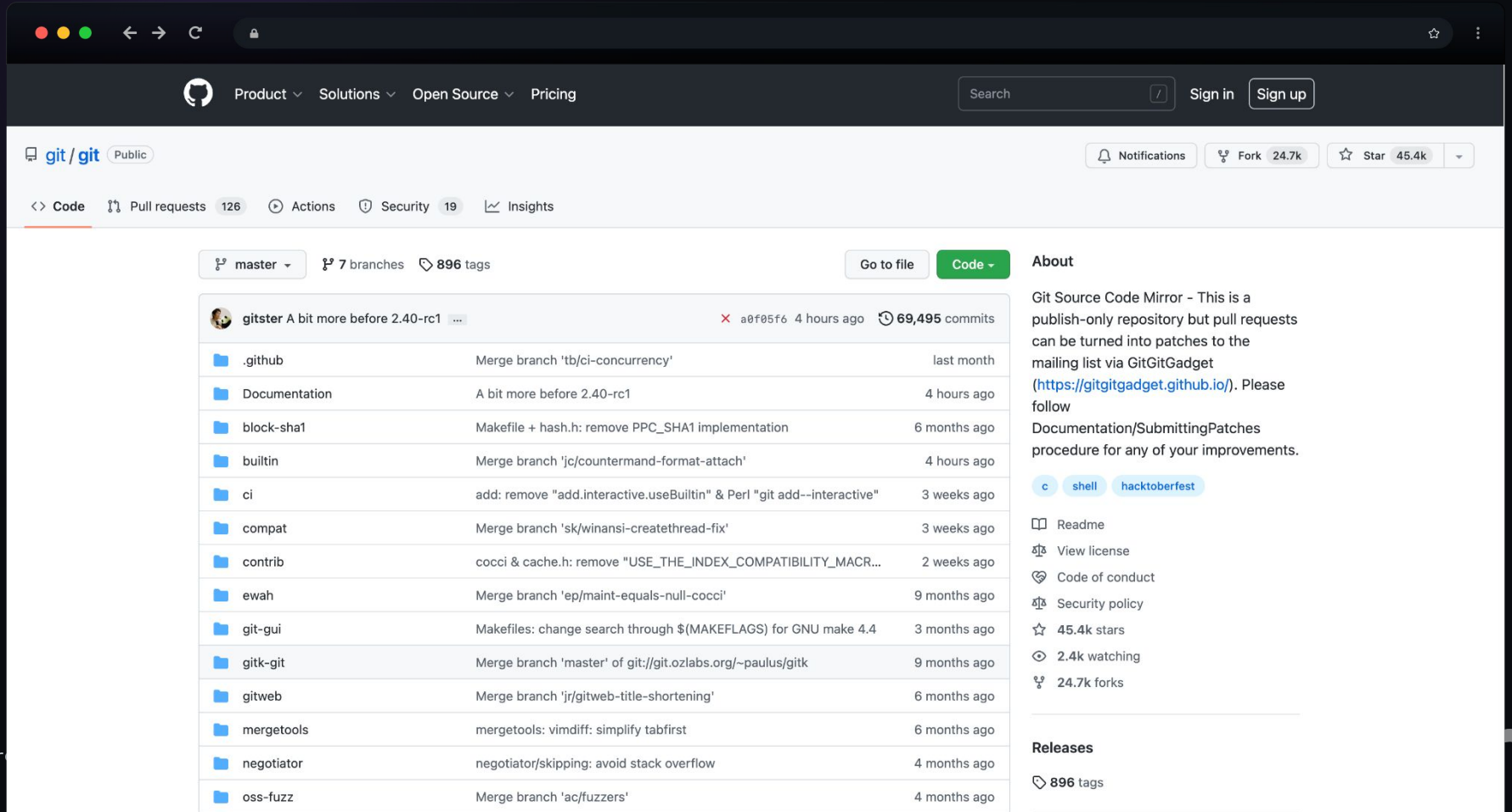
Sharding / routing

- Routing is oblivious for read requests: replicas are kept in sync so choosing which replica to read from is arbitrary
  - First step: pick a datacenter
  - Second step: pick a replica within that datacenter based on the healthy replicas' read weights
- Write requests are proxied among all replicas (healthy or not)
- Visible updates (e.g., the reference update(s) tied to a push) are carried out using a separate process



Remote procedure calls

Querying Git content as part of a web request



Product Solutions Open Source Pricing

Search / Sign in Sign up

git / git Public

Notifications Fork 24.7k Star 45.4k

Code Pull requests 126 Actions Security 19 Insights

master 7 branches 896 tags Go to file Code

gitster A bit more before 2.40-rc1 ... a0f05f6 4 hours ago 69,495 commits

.github	Merge branch 'tb/ci-concurrency'	last month
Documentation	A bit more before 2.40-rc1	4 hours ago
block-sha1	Makefile + hash.h: remove PPC_SHA1 implementation	6 months ago
builtin	Merge branch 'jc/countermand-format-attach'	4 hours ago
ci	add: remove "add.interactive.useBuiltin" & Perl "git add--interactive"	3 weeks ago
compat	Merge branch 'sk/winansi-createthread-fix'	3 weeks ago
contrib	cocci & cache.h: remove "USE_THE_INDEX_COMPATIBILITY_MACR...	2 weeks ago
ewah	Merge branch 'ep/maint-equals-null-cocci'	9 months ago
git-gui	Makefiles: change search through $(MAKEFLAGS) for GNU make 4.4	3 months ago
gitk-git	Merge branch 'master' of git://git.ozlabs.org/~paulus/gitk	9 months ago
gitweb	Merge branch 'jr/gitweb-title-shortening'	6 months ago
mergetools	mergetools: vimdiff: simplify tabfirst	6 months ago
negotiator	negotiator/skipping: avoid stack overflow	4 months ago
oss-fuzz	Merge branch 'ac/fuzzers'	4 months ago

About

Git Source Code Mirror - This is a publish-only repository but pull requests can be turned into patches to the mailing list via GitGitGadget (https://gitgitgadget.github.io/). Please follow Documentation/SubmittingPatches procedure for any of your improvements.

c shell hacktoberfest

Readme

View license

Code of conduct

Security policy

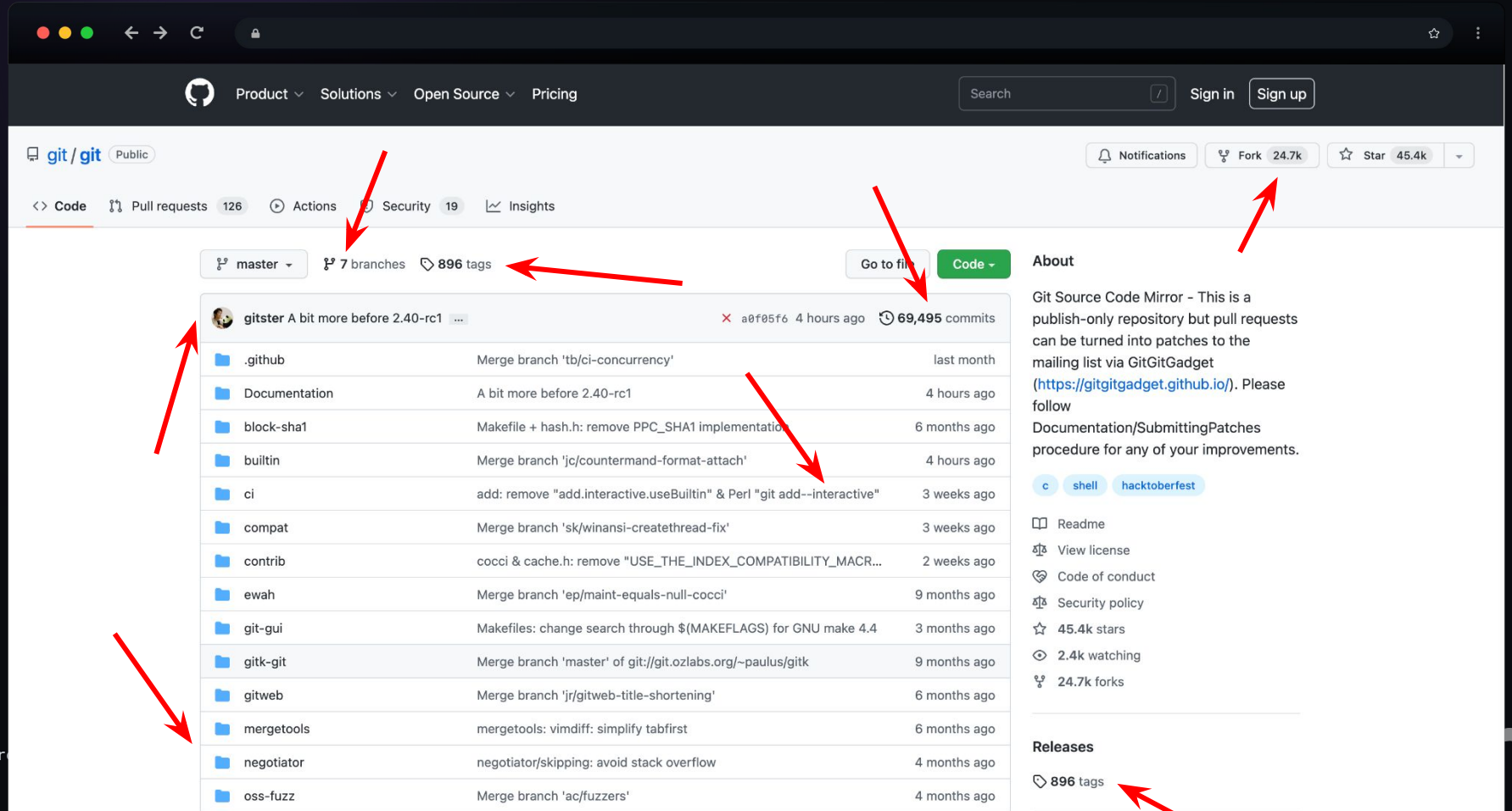
45.4k stars

2.4k watching

24.7k forks

Releases

896 tags


Product
Solutions
Open Source
Pricing
Search
Sign in
Sign up
git / git
Public
Notifications
Fork 24.7k
Star 45.4k
Code
Pull requests 126
Actions
Security 19
Insights
master
7 branches
896 tags
Go to file
Code
gitster A bit more before 2.40-rc1 ...
a0f05f6 4 hours ago
69,495 commits
.github Merge branch 'tb/ci-concurrency' last month
Documentation A bit more before 2.40-rc1 4 hours ago
block-sha1 Makefile + hash.h: remove PPC_SHA1 implementation 6 months ago
builtin Merge branch 'jc/countermand-format-attach' 4 hours ago
ci add: remove "add.interactive.useBuiltin" & Perl "git add--interactive" 3 weeks ago
compat Merge branch 'sk/winansi-createthread-fix' 3 weeks ago
contrib cocci & cache.h: remove "USE_THE_INDEX_COMPATIBILITY_MACR... 2 weeks ago
ewah Merge branch 'ep/maint-equals-null-cocci' 9 months ago
git-gui Makefiles: change search through $(MAKEFLAGS) for GNU make 4.4 3 months ago
gitk-git Merge branch 'master' of git://git.ozlabs.org/~paulus/gitk 9 months ago
gitweb Merge branch 'jr/gitweb-title-shortening' 6 months ago
mergetools mergetools: vimdiff: simplify tabfirst 6 months ago
negotiator negotiator/skipping: avoid stack overflow 4 months ago
oss-fuzz Merge branch 'ac/fuzzers' 4 months ago
About
Git Source Code Mirror - This is a publish-only repository but pull requests can be turned into patches to the mailing list via GitGitGadget (https://gitgitgadget.github.io/). Please follow Documentation/SubmittingPatches procedure for any of your improvements.
c
shell
hacktoberfest
Readme
View license
Code of conduct
Security policy
45.4k stars
2.4k watching
24.7k forks
Releases
896 tags

Remote procedure calls

- Web backends (Unicorn, Ruby on Rails) are on separate machines from Git repositories
- Isolation of data, can tailor machine hardware based on demand
- Need to be able to synchronously “ask” for Git data from web backends

Remote procedure calls: GitRPC

- GitRPC: set of library methods / interfaces for requesting data about a Git repository
  - How many references?
  - Tree-level blame
  - Fetch the content of this blob (e.g., “README.md”)
- Custom serialization format (BERT-RPC) published by GitHub
  - https://github.blog/2009-10-20-introducing-bert-and-bert-rpc/
- Body consists of:
  - Method to call
  - Arguments to that method
  - Fileserver identifier and path

Remote procedure calls: GitRPC

- RPCs are tagged as `rpc_reader` or `rpc_writer`
- Choice is made according to whether or not the RPC can modify the “hard state” of the repository (more on this later)
- Can send messages synchronously (`send_message`) or asynchronously (`async_send_message`)
- Can send messages to one (`send_single`) or more than one recipient (`send_multiple`)

Remote procedure calls: GitRPC

- Simple idea: one of the first topics covered by CSE452
- Powerful concept:
  - Distribute “data” away from “compute”
  - Tailor hardware choices to accommodate your workload
- Lots of details:
  - Making sure messages make their way to desired recipient(s)
  - Serializing / deserializing message contents
  - Performance is critical: tiny amounts of overhead add up

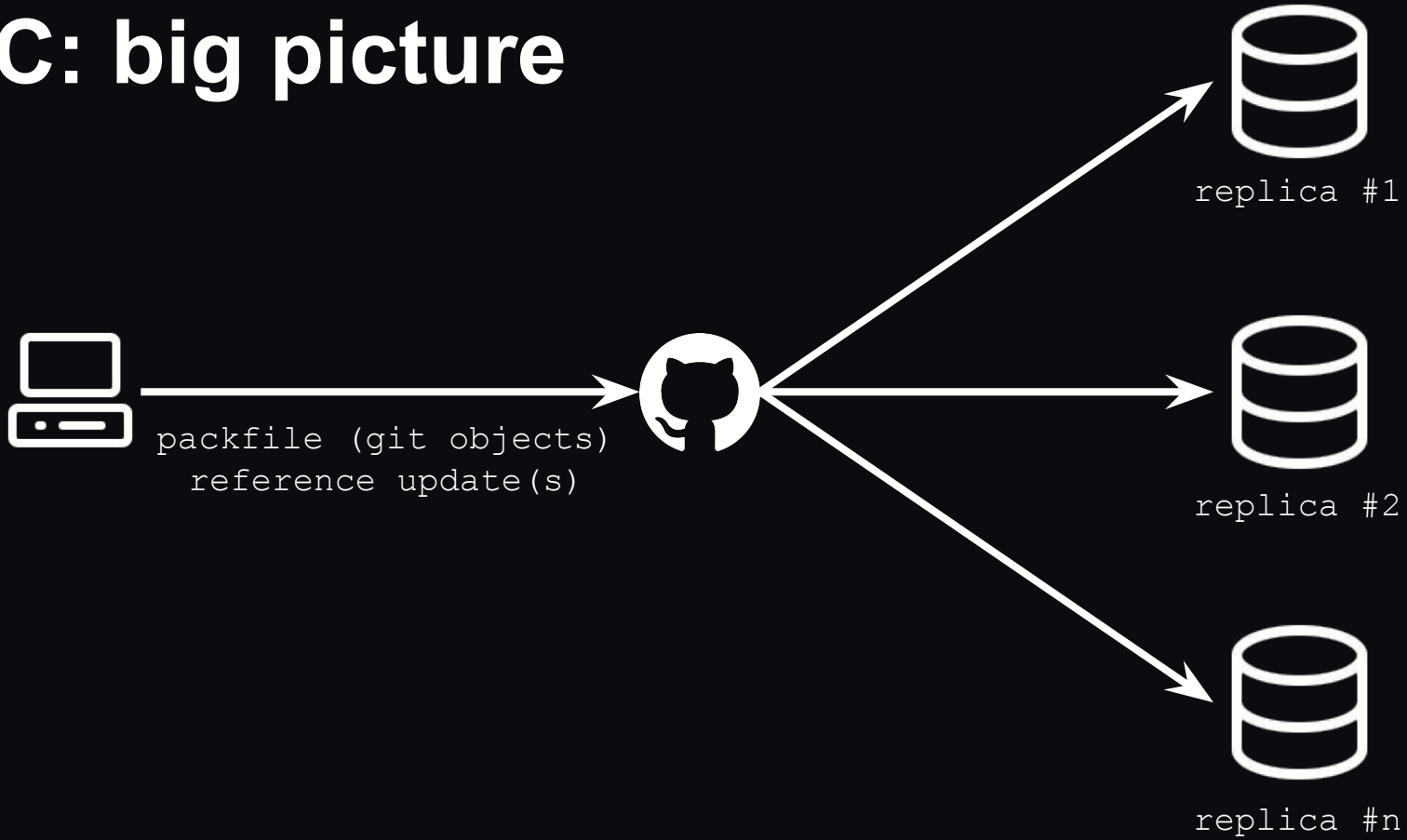
3-phase commit (3PC)

Linearizing reference (branch/tag) updates across repository replicas

3PC

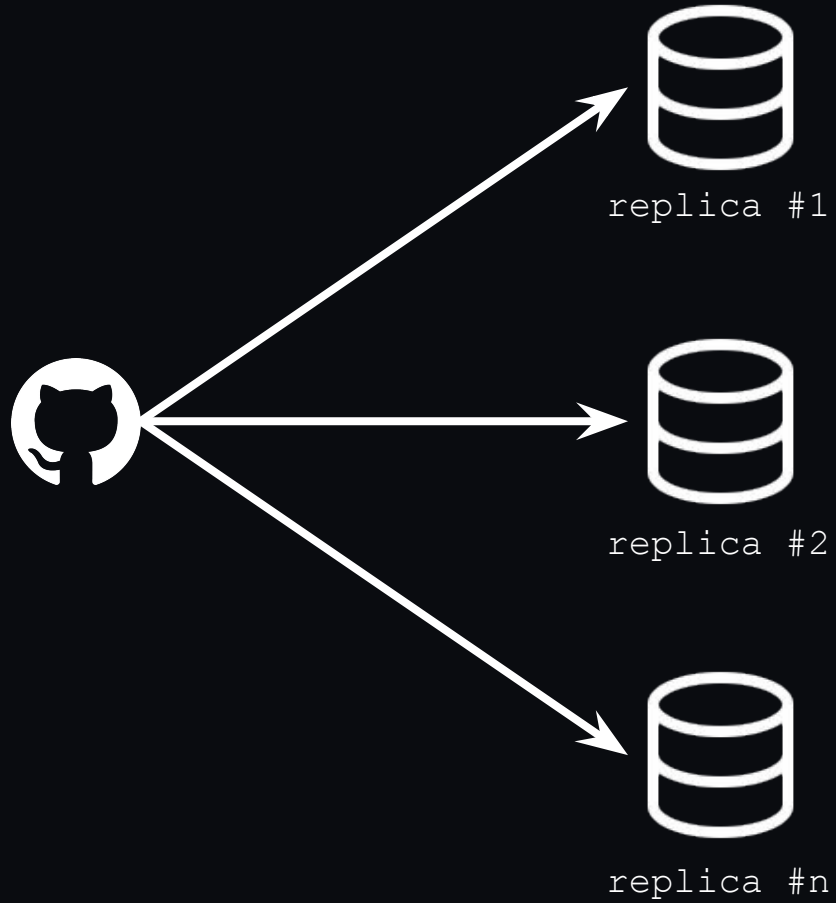
- 3-phase commit (3PC) is a type of atomic commitment protocol
  - Atomic commit protocol (ACP):
    - consistency: all nodes agree on the proposed value
    - stability: once a value is agreed upon, it never changes
- We use 3PC(-ish) to linearize reference updates among Git replicas

3PC: big picture



3PC: big picture

- Must keep repository replicas in sync
- “In sync”: same set of references in a snapshot
- Replicating a push proceeds in two steps:
  - First, proxy all data to all replicas
  - Then, use 3PC to coordinate the reference update(s)



audit_log

- “In sync”: same set of (reference name, SHA-1) pairs
  - Or: `git clone`-ing from an arbitrary replica yields the same content
- Reference updates are written serially to an `audit_log` file tracking which reference updates have been performed

```
[gitro@github-dfs-11e04ce » ttaylorr/dotfiles]$ tail -1 audit_log
refs/heads/work-gh bc14198c05cb78276b0d0c66712033ceb20e0346
4336ad33ee19b56b1fe6a9e3f9064981c5ff5098 Taylor Blau
<ttaylorr@github.com> 1677168429 +0000    { … }
```


audit_log

- “In sync”: same set of (reference name, SHA-1) pairs
  - Or: `git clone`-ing from an arbitrary replica yields the same content
- Reference updates are written serially to an `audit_log` file tracking which reference updates have been performed

```
[gitro@github-dfs-11e04ce » ttaylorr/dotfiles]$ tail -1 audit_log
refs/heads/work-gh bc14198c05cb78276b0d0c66712033ceb20e0346
4336ad33ee19b56b1fe6a9e3f9064981c5ff5098 Taylor Blau
<ttaylorr@github.com> 1677168429 +0000     { … }
```


audit_log

- “In sync”: same set of (reference name, SHA-1) pairs
  - Or: `git clone`-ing from an arbitrary replica yields the same content
- Reference updates are written serially to an `audit_log` file tracking which reference updates have been performed

```
[gitro@github-dfs-11e04ce » ttaylorr/dotfiles]$ tail -1 audit_log
refs/heads/work-gh bc14198c05cb78276b0d0c66712033ceb20e0346
4336ad33ee19b56b1fe6a9e3f9064981c5ff5098 Taylor Blau
<ttaylorr@github.com> 1677168429 +0000    { … }
```


audit_log

- “In sync”: same set of (reference name, SHA-1) pairs
  - Or: `git clone`-ing from an arbitrary replica yields the same content
- Reference updates are written serially to an `audit_log` file tracking which reference updates have been performed

```
[gitro@github-dfs-11e04ce » ttaylorr/dotfiles]$ tail -1 audit_log
refs/heads/work-gh bc14198c05cb78276b0d0c66712033ceb20e0346
4336ad33ee19b56b1fe6a9e3f9064981c5ff5098 Taylor Blau
<ttaylorr@github.com> 1677168429 +0000     { … }
```

replica #n

audit_log

- “In sync”: same set of (reference name, SHA-1) pairs
  - Or: `git clone`-ing from an arbitrary replica yields the same content
- Reference updates are written serially to an `audit_log` file tracking which reference updates have been performed

```
[gitro@github-dfs-11e04ce » ttaylorr/dotfiles]$ tail -1 audit_log
refs/heads/work-gh bc14198c05cb78276b0d0c66712033ceb20e0346
4336ad33ee19b56b1fe6a9e3f9064981c5ff5098  Taylor Blau
<ttaylorr@github.com> 1677168429 +0000 { … }
```


audit_log

- “In sync”: same set of (reference name, SHA-1) pairs
  - Or: `git clone`-ing from an arbitrary replica yields the same content
- Reference updates are written serially to an `audit_log` file tracking which reference updates have been performed

```
[gitro@github-dfs-11e04ce » ttaylorr/dotfiles]$ tail -1 audit_log
refs/heads/work-gh bc14198c05cb78276b0d0c66712033ceb20e0346
4336ad33ee19b56b1fe6a9e3f9064981c5ff5098 Taylor Blau
<ttaylorr@github.com> 1677168429 +0000     { … }
```

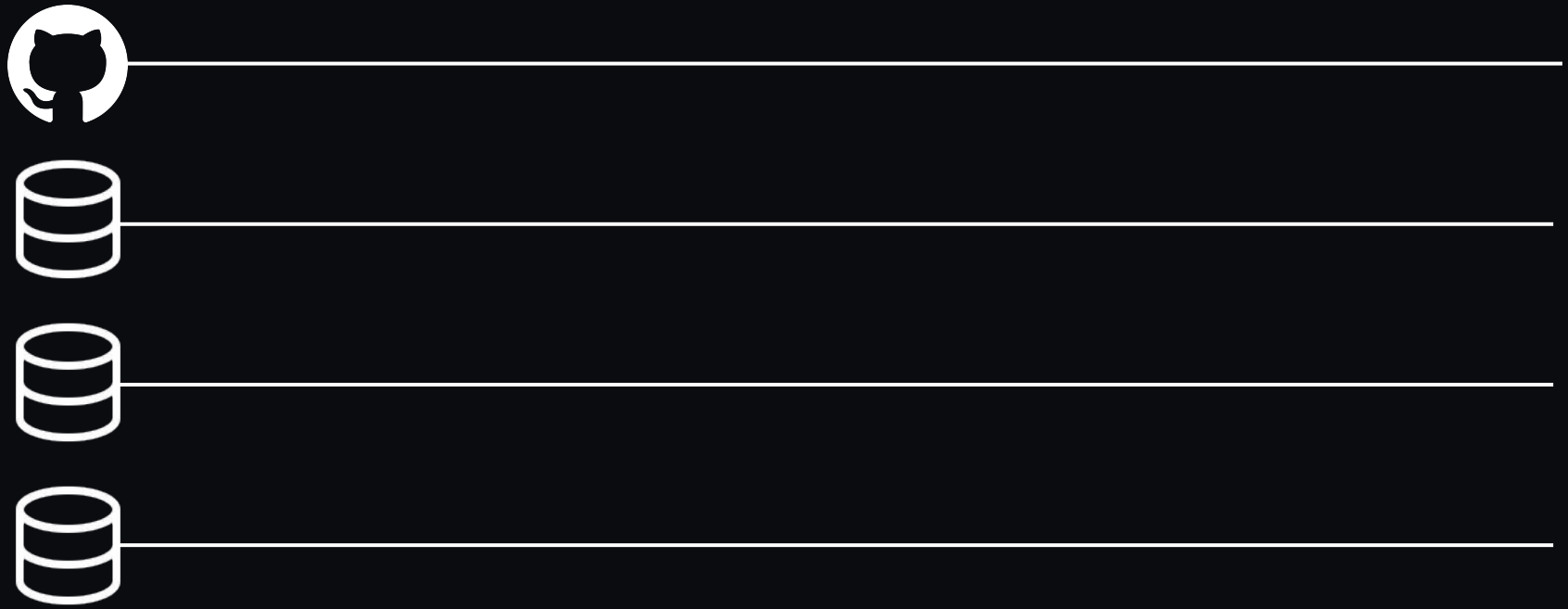

audit_log

- “In sync”: same set of (reference name, SHA-1) pairs
  - Or: `git clone`-ing from an arbitrary replica yields the same content
- Reference updates are written serially to an `audit_log` file tracking which reference updates have been performed

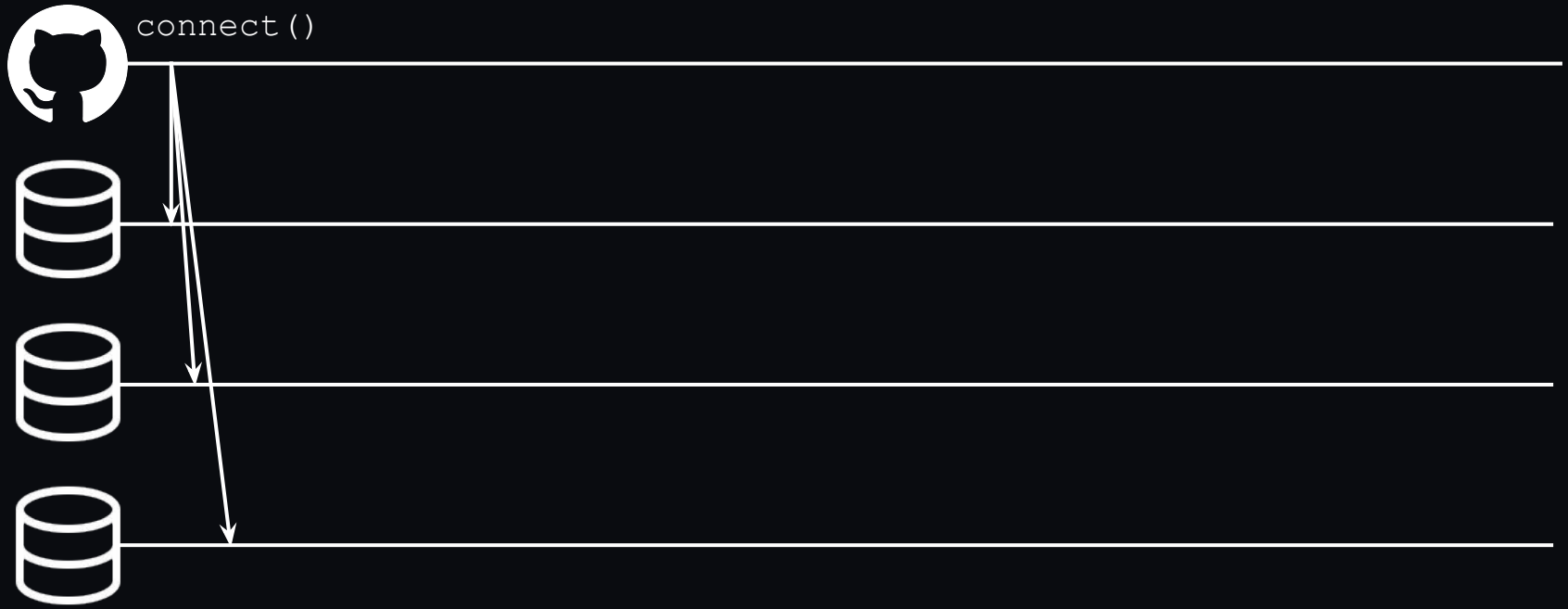
```
[gitro@github-dfs-11e04ce » ttaylorr/dotfiles]$ tail -1 audit_log
refs/heads/work-gh bc14198c05cb78276b0d0c66712033ceb20e0346
4336ad33ee19b56b1fe6a9e3f9064981c5ff5098 Taylor Blau
<ttaylorr@github.com> 1677168429 +0000    { … }

[gitro@github-dfs-11e04ce » ttaylorr/dotfiles]$ cat dgit-state
5:62e932c0976b050fff510dae4235085d157eba8a
audit-log-checksum 2b0c6953e1799e8fafa6a42308276e5c4d37d21a
audit-log-partial-state block-sha1-blockwise 462720 e97f4d63 81fd63c9
e09d949b fe38d052 d3714666
```

replica #n

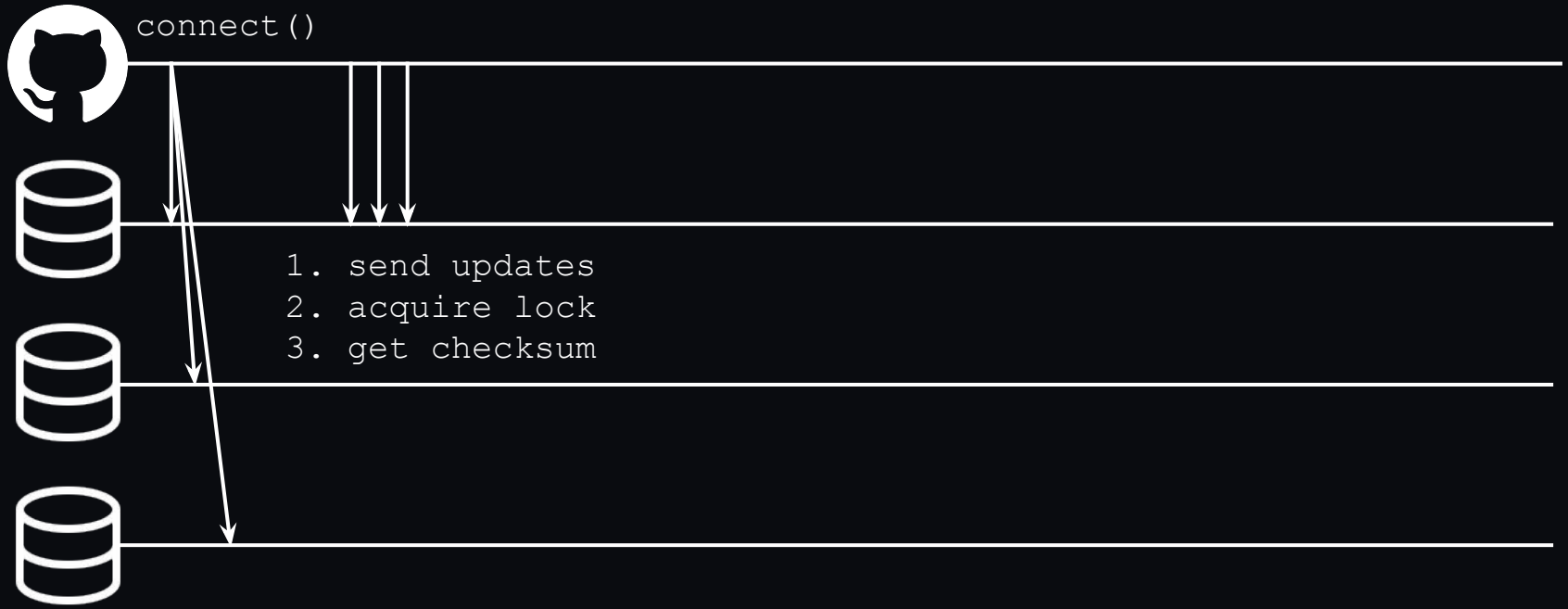
3PC: small picture



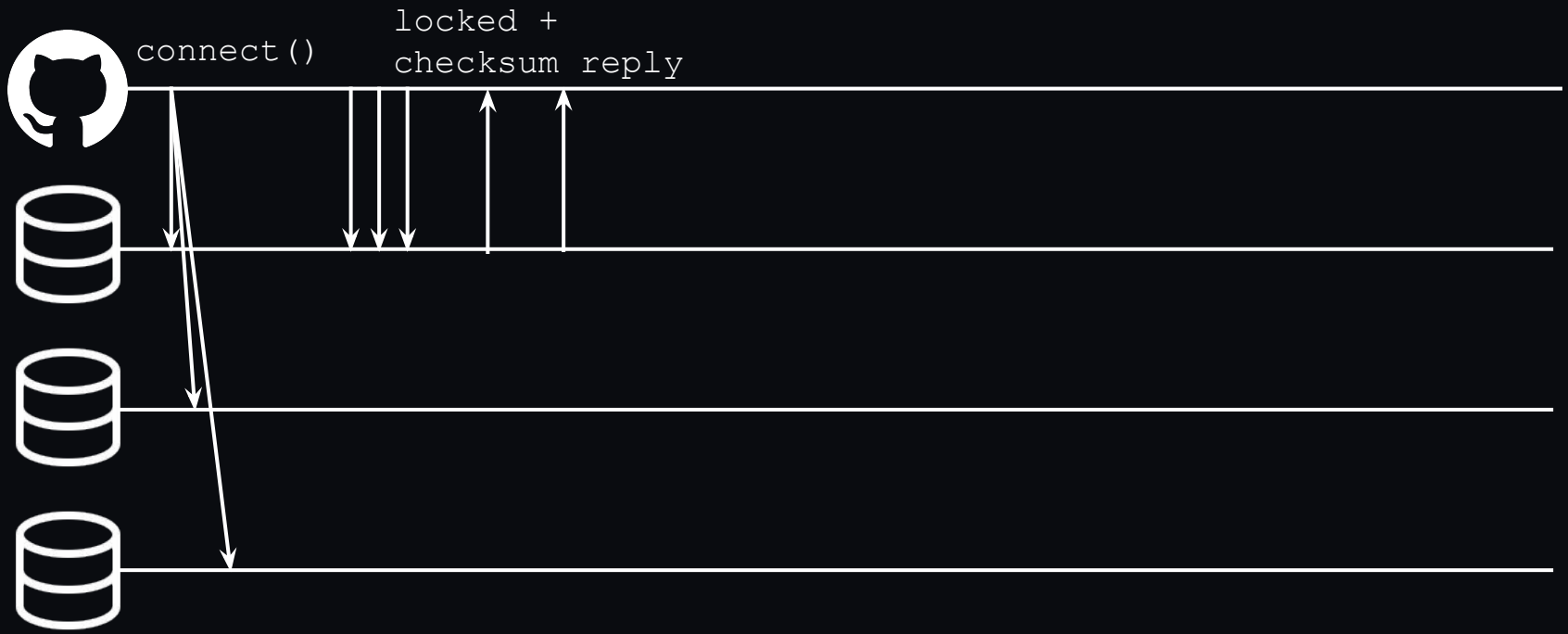
3PC: small picture



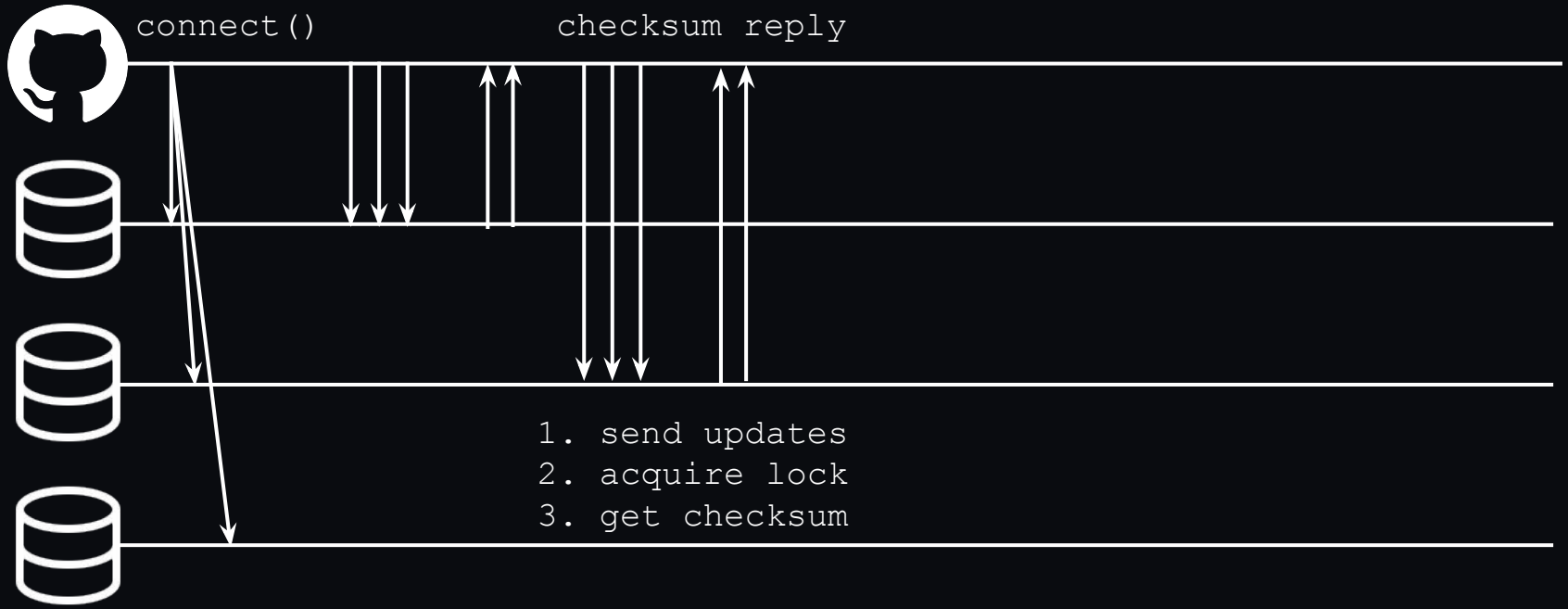
3PC: small picture



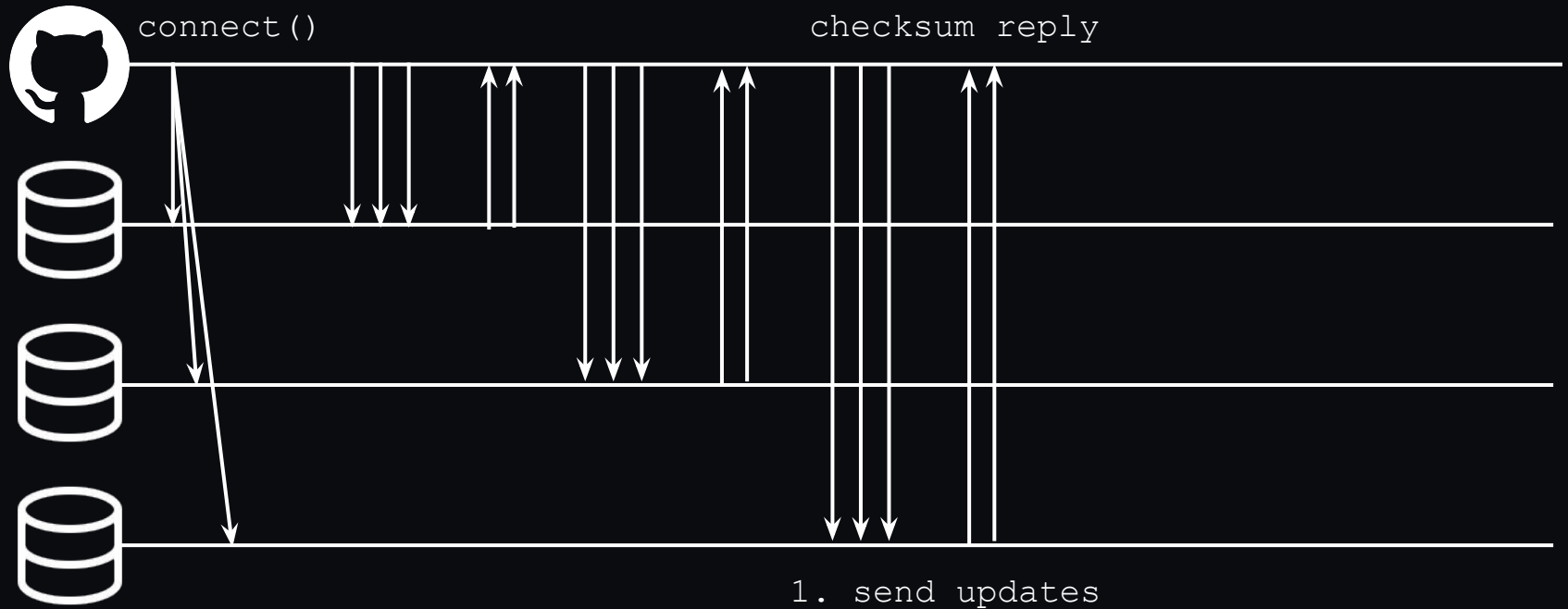
3PC: small picture



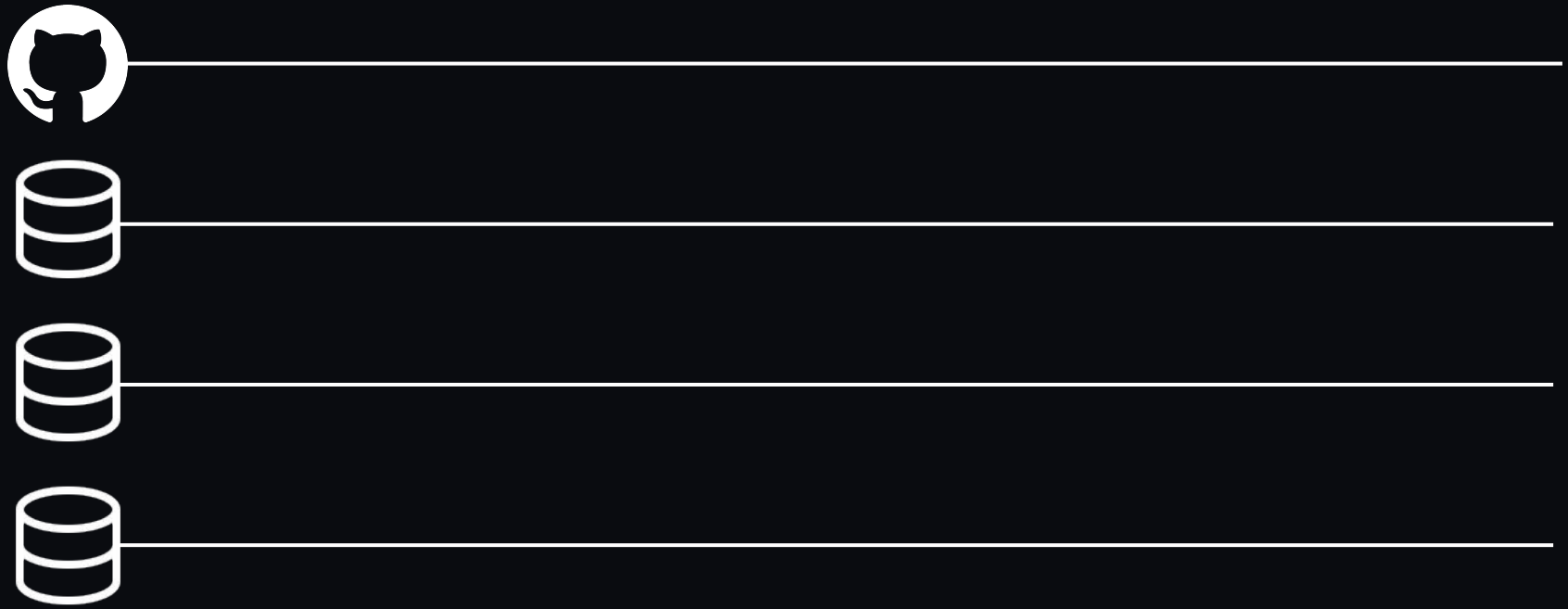
3PC: small picture



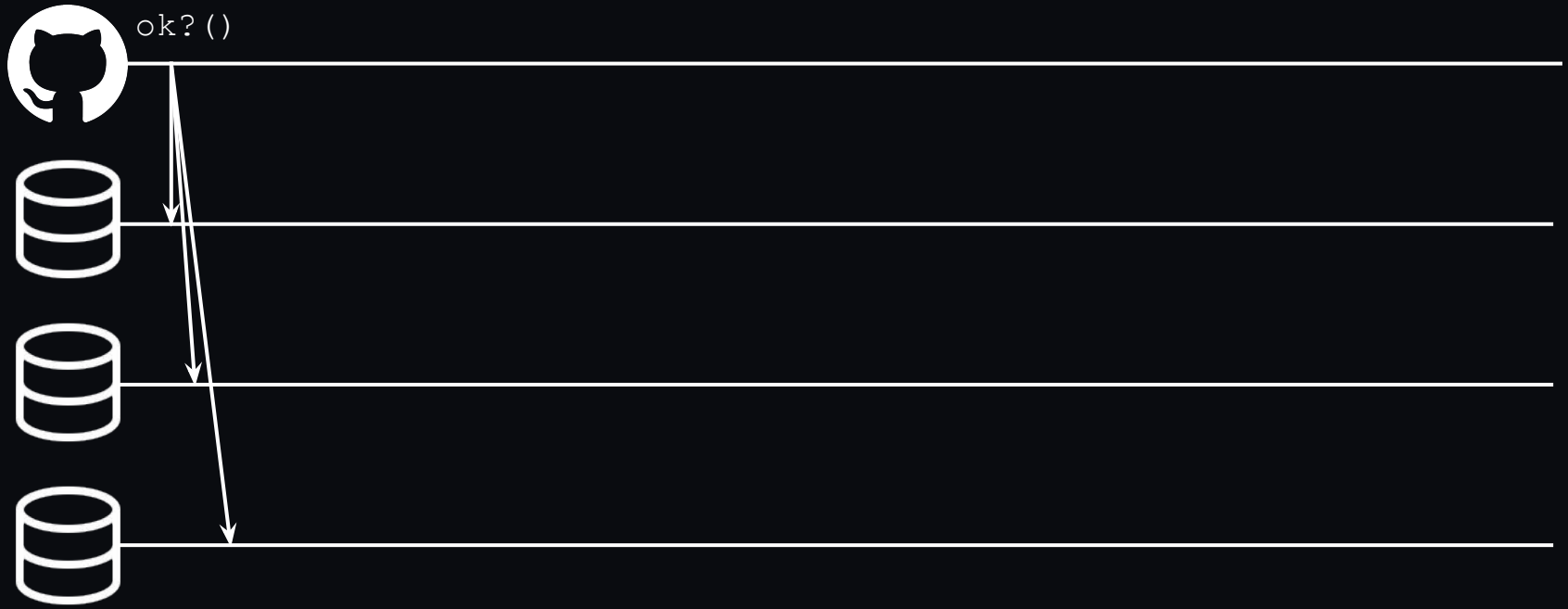
3PC: small picture



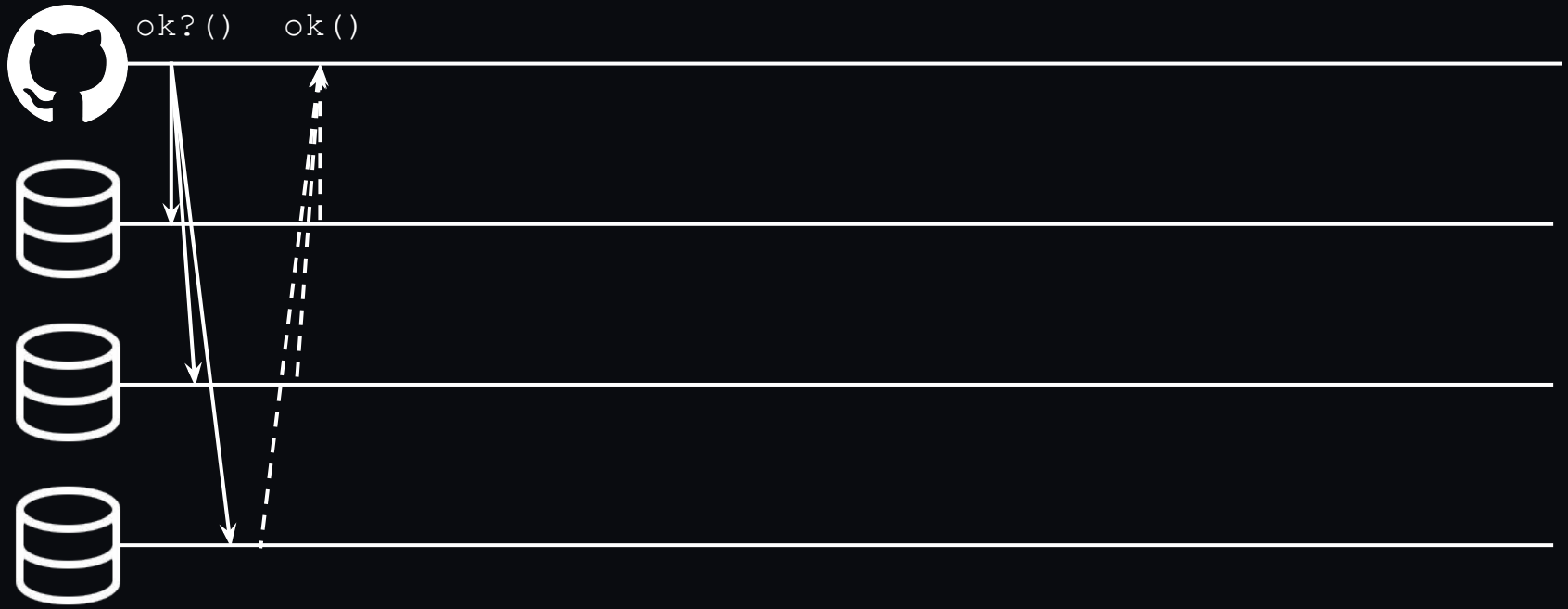
3PC: small picture



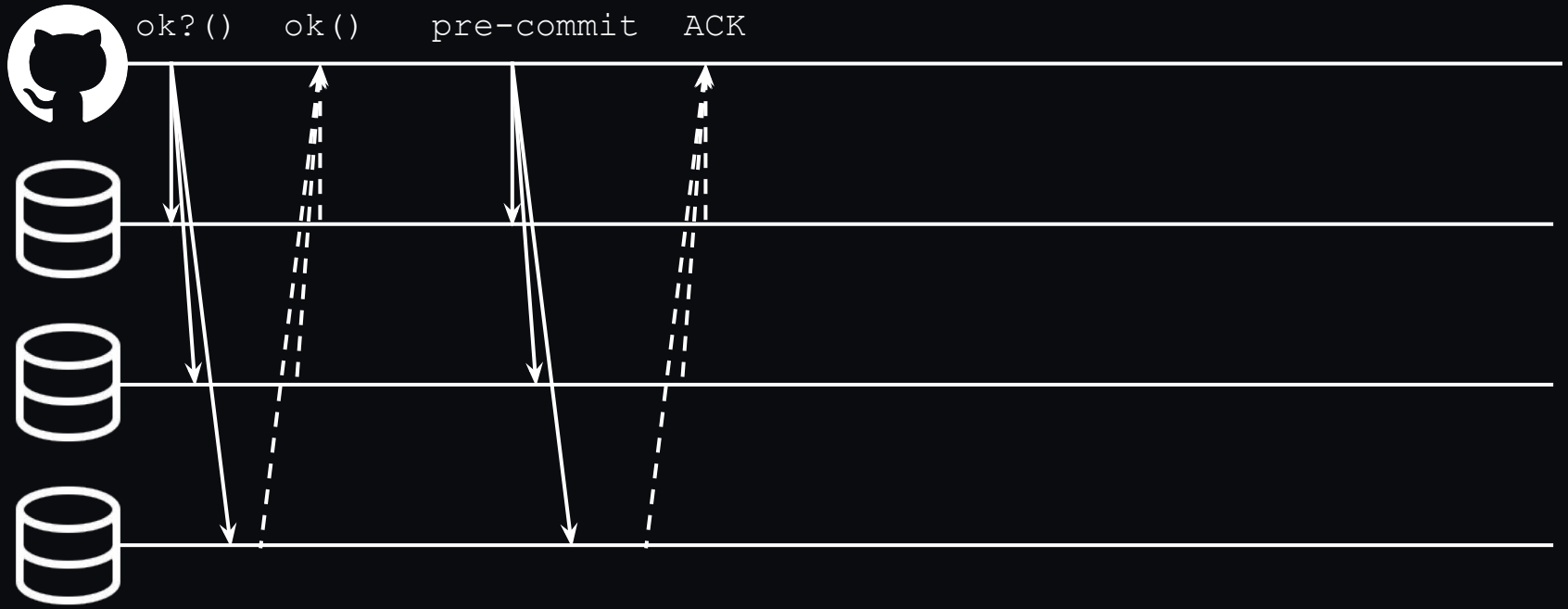
3PC: small picture



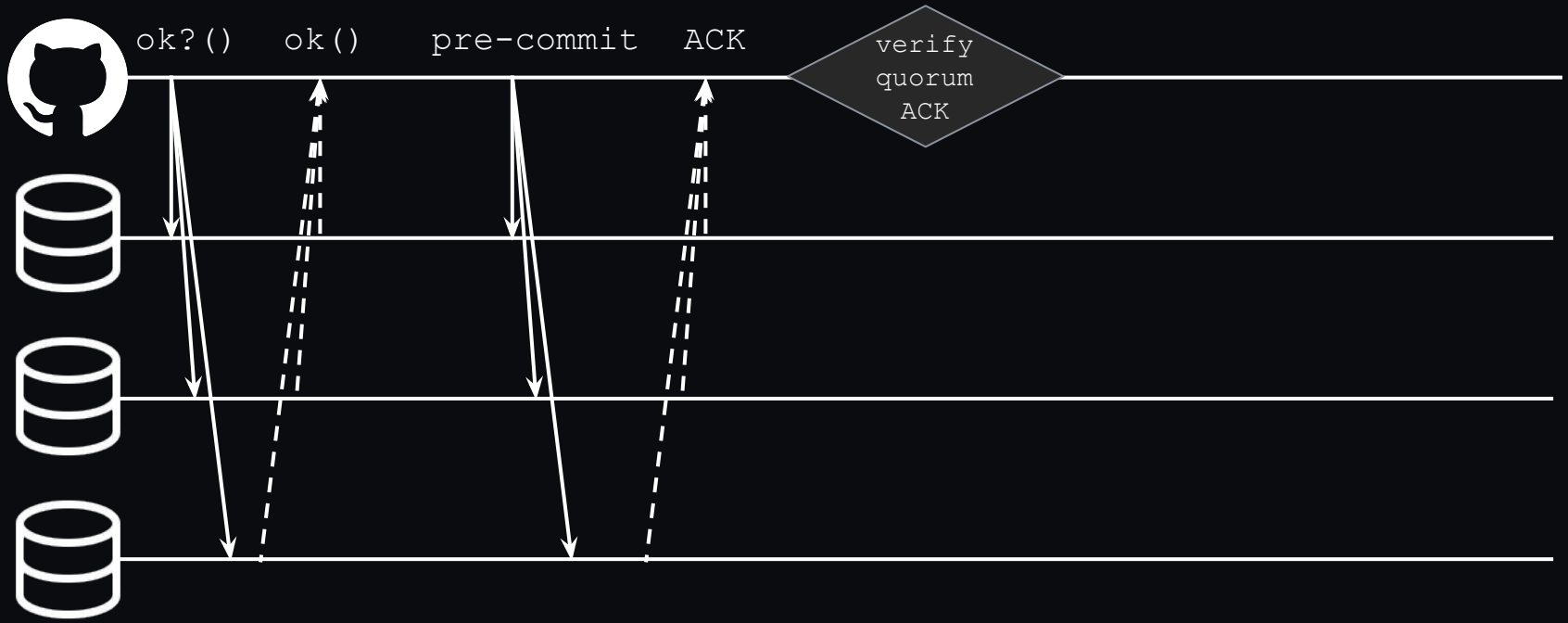
3PC: small picture



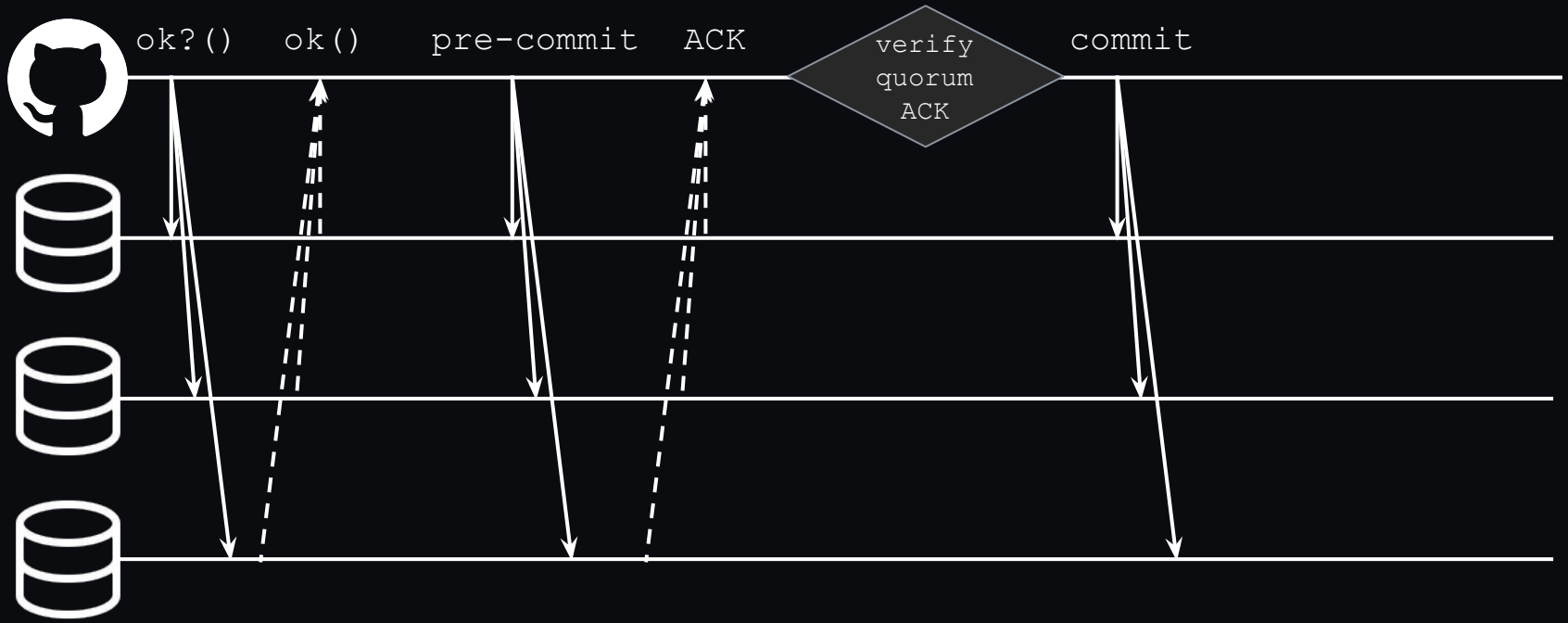
3PC: small picture



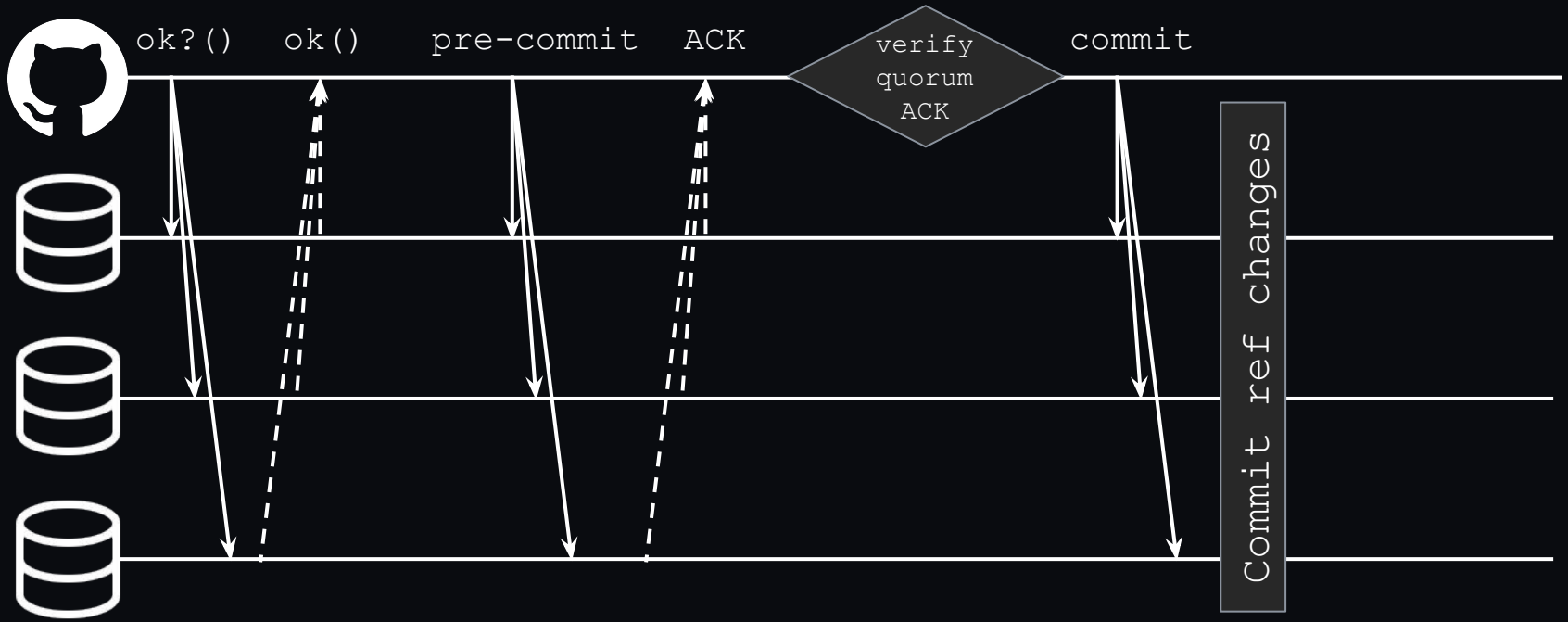
3PC: small picture



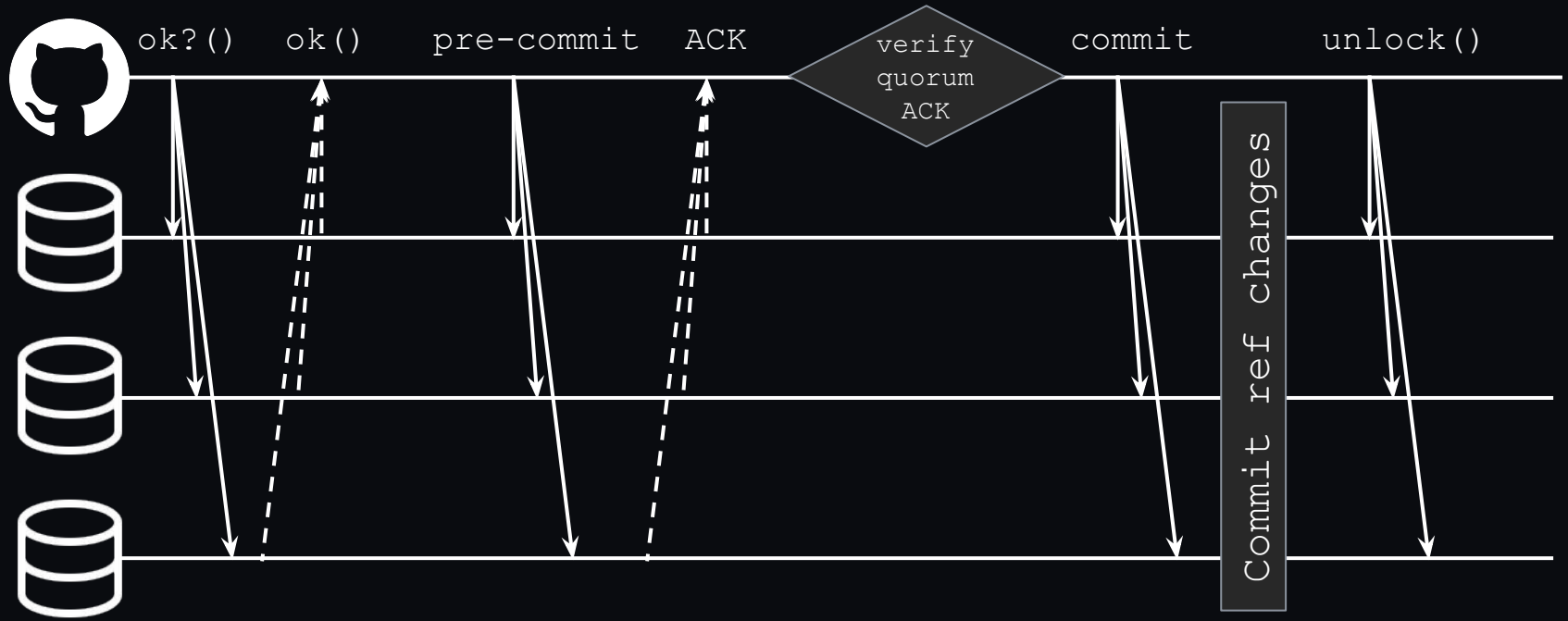
3PC: small picture



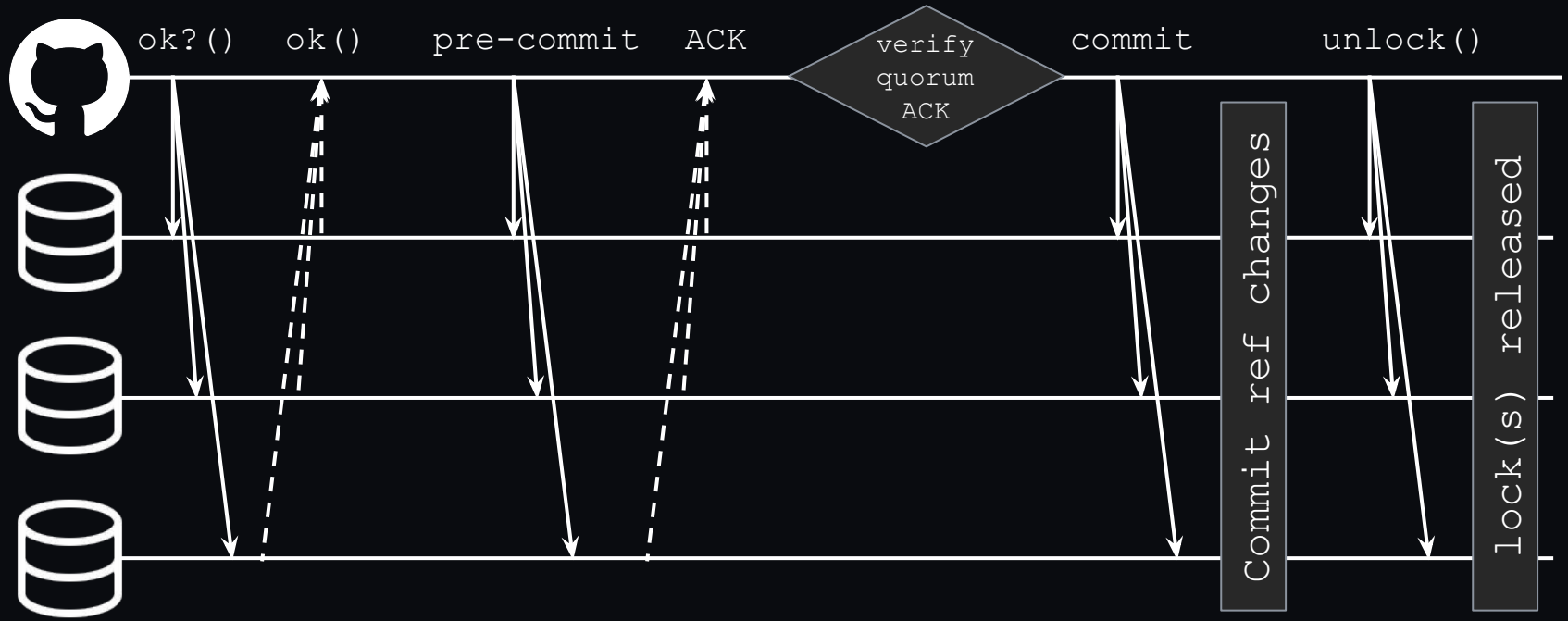
3PC: small picture



3PC: small picture



3PC: small picture



3PC: pros/cons

- Pro: completely serial (linearizable) set of reference transactions
  - Read-your-writes
- Con: requires many network round-trips
  - Difficult to scale across multiple datacenters without fast/reliable links

3PC: other details

- Non-voting replicas: optimization we made to handle additional replicas at geo-distributed sites
  - NV replicas can be locked in parallel without worrying about the Dining Philosophers problem
  - NV replicas can be unlocked before the committed checksum is written back to the database
- Not-quite-3PC: OK to have a minority of replicas dissent/NAK or disappear
- Repairs: ~`git fetch` on damaged replicas

(etc)

- Off-site repository backups (gitbackups)
- Rate limiting (gitmon)
- Caching clones (lariat)
- …Lots more :-)

Q&A