Hey. I'm G/.

@guillaumedotid

And this is where I live...

My application

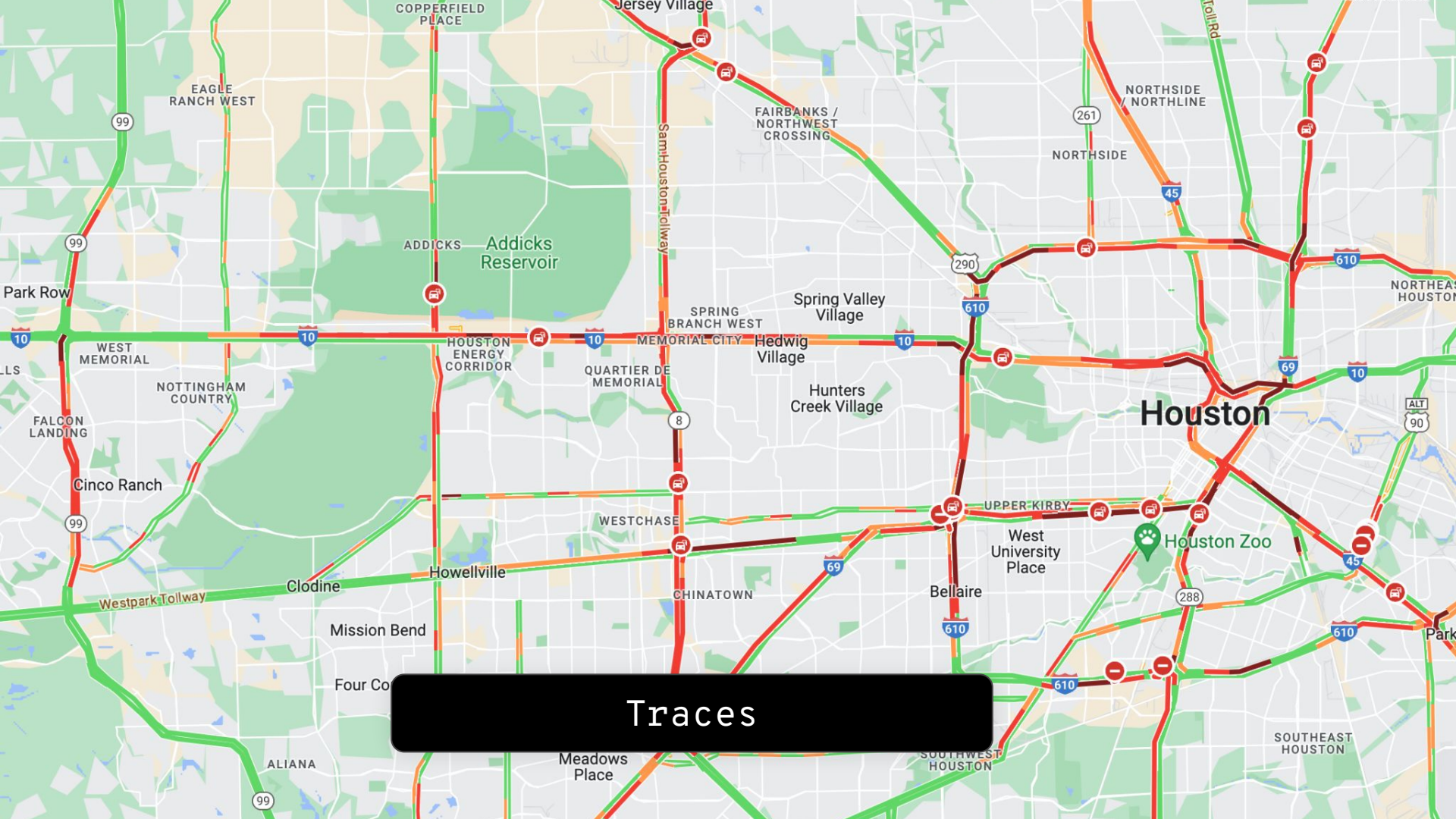
Can't we just add more lanes?

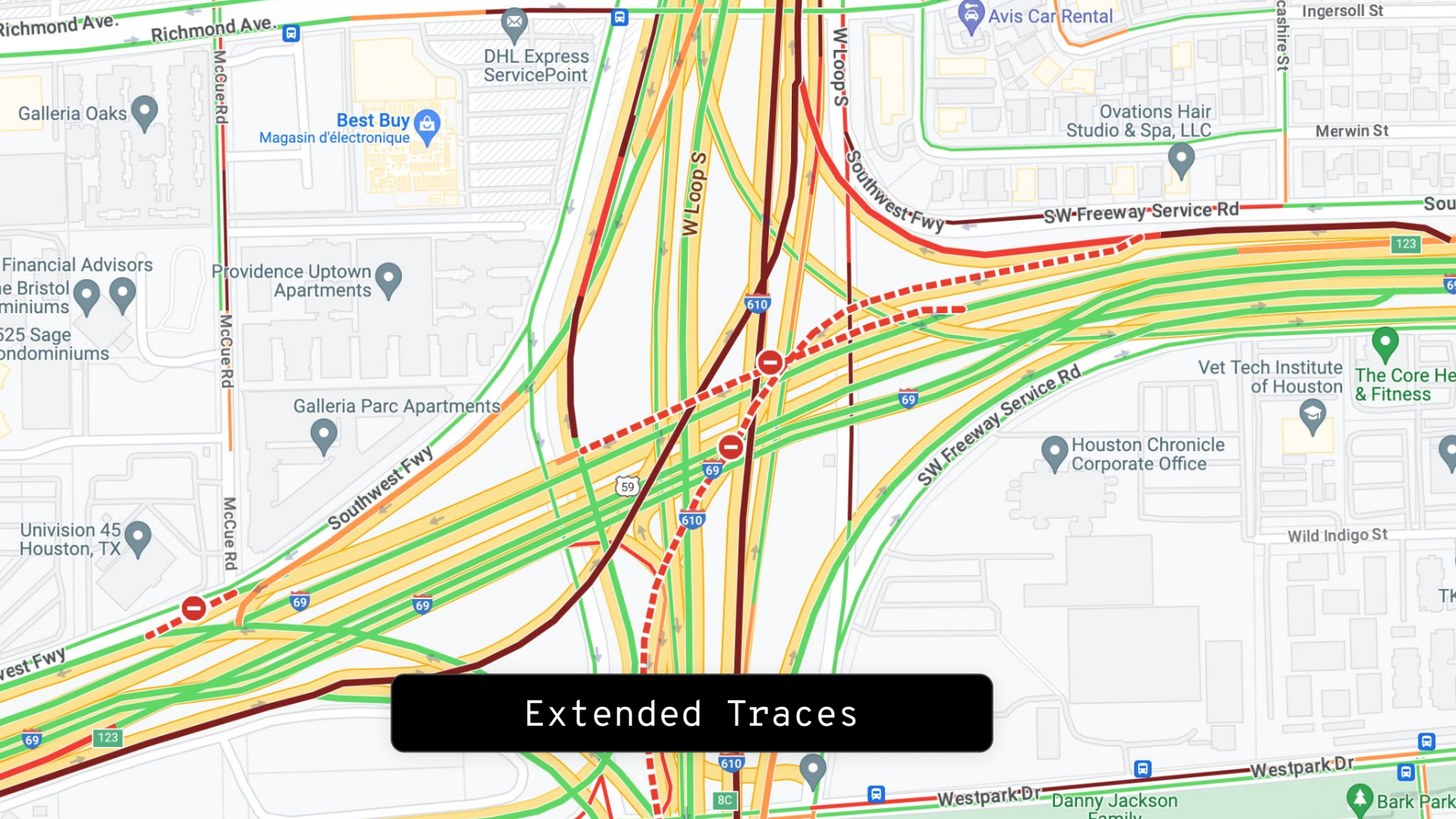
(servers and processes)

Short answer: **You should not.**

Can we **detect** and **fix issues** in production quickly?

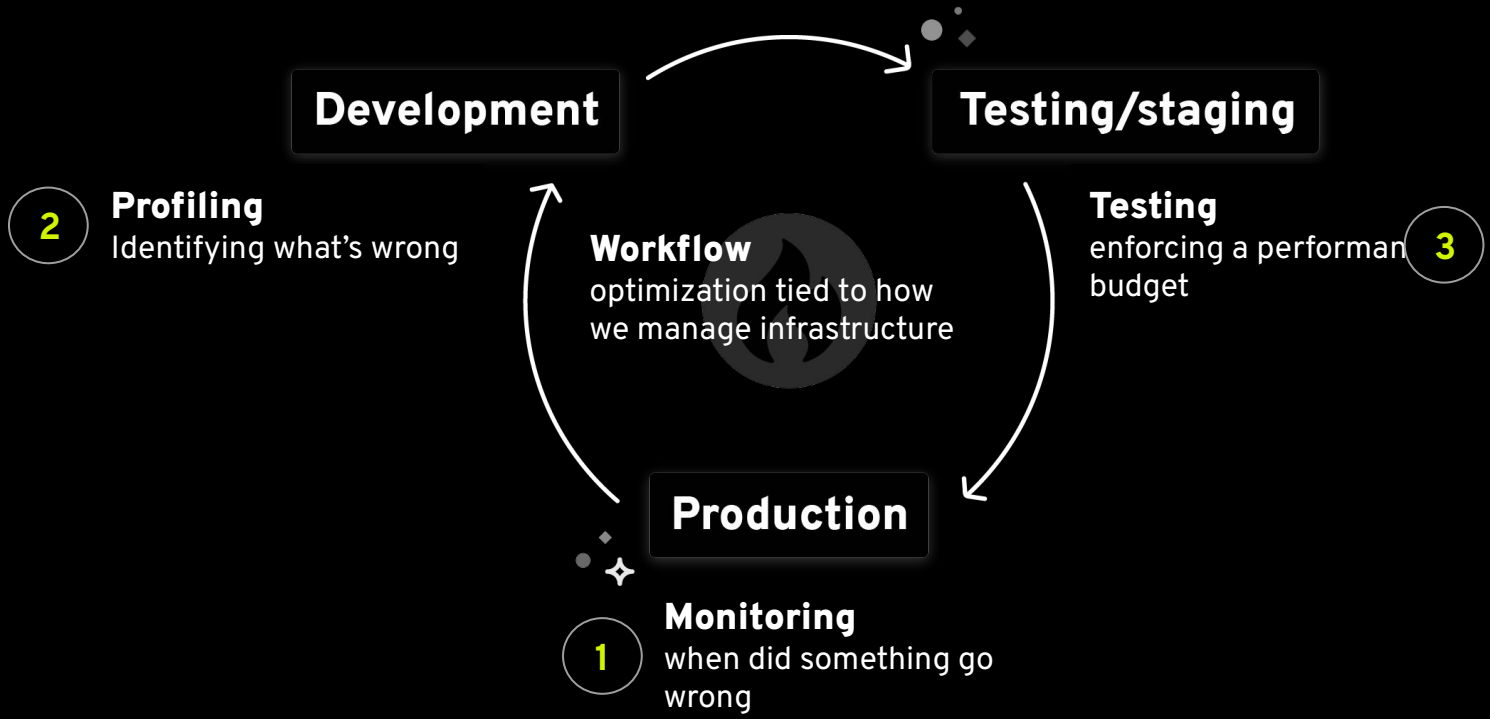
Can we **prevent** new work that contains **performance regressions** from promotion?


Traces
COPPERFIELD
Jersey Village
EAGLE RANCH WEST
FAIRBANKS / NORTHWEST CROSSING
NORTHSIDE / NORTHLINE
NORTHSIDE
Sam Houston Tollway
ADDICKS
Addicks Reservoir
Park Row
SPRING BRANCH WEST
Spring Valley Village
WEST MEMORIAL
HOUSTON ENERGY CORRIDOR
MEMORIAL CITY
Hedwig Village
QUARTIER DE MEMORIAL
NOTTINGHAM COUNTRY
Hunters Creek Village
Houston
FALCON LANDING
Cinco Ranch
WESTCHASE
UPPER KIRBY
West University Place
Houston Zoo
Howellville
Clodine
Westpark Tollway
CHINATOWN
Bellaire
Mission Bend
Four Co
ALIANA
Meadows Place
SOUTHWEST HOUSTON
SOUTHEAST HOUSTON


Extended Traces

IH-69 Southwest Southbound Connector Ramp To IH-610 WEST LOOP southbound.
Closed Until Further Notice from Friday, April 29, 2022 9:00 PM to Monday, April 29, 2024 3:00 PM
RAMP CLOSED
TUNDRA
Profiling

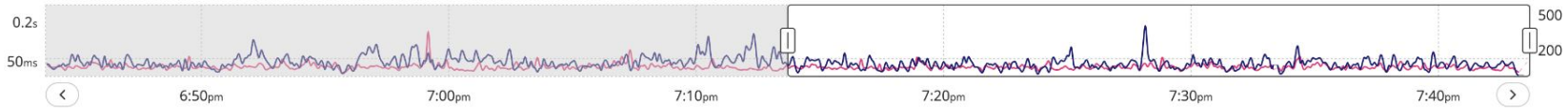
Building a continuous observability strategy



blackfire.io
A Platform.sh product

platform.sh

Context Web HTTP Host (All) HTTP Method (All) Host (All) Runtime (All) Period Today 7:13 pm - 7:43 pm LIVE

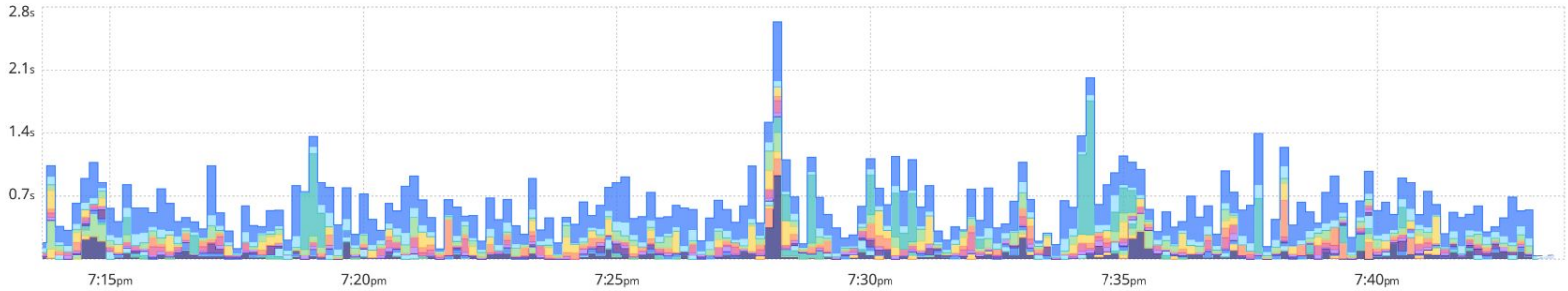


Time Memory Events Alerts Zoom RPMs Load Output

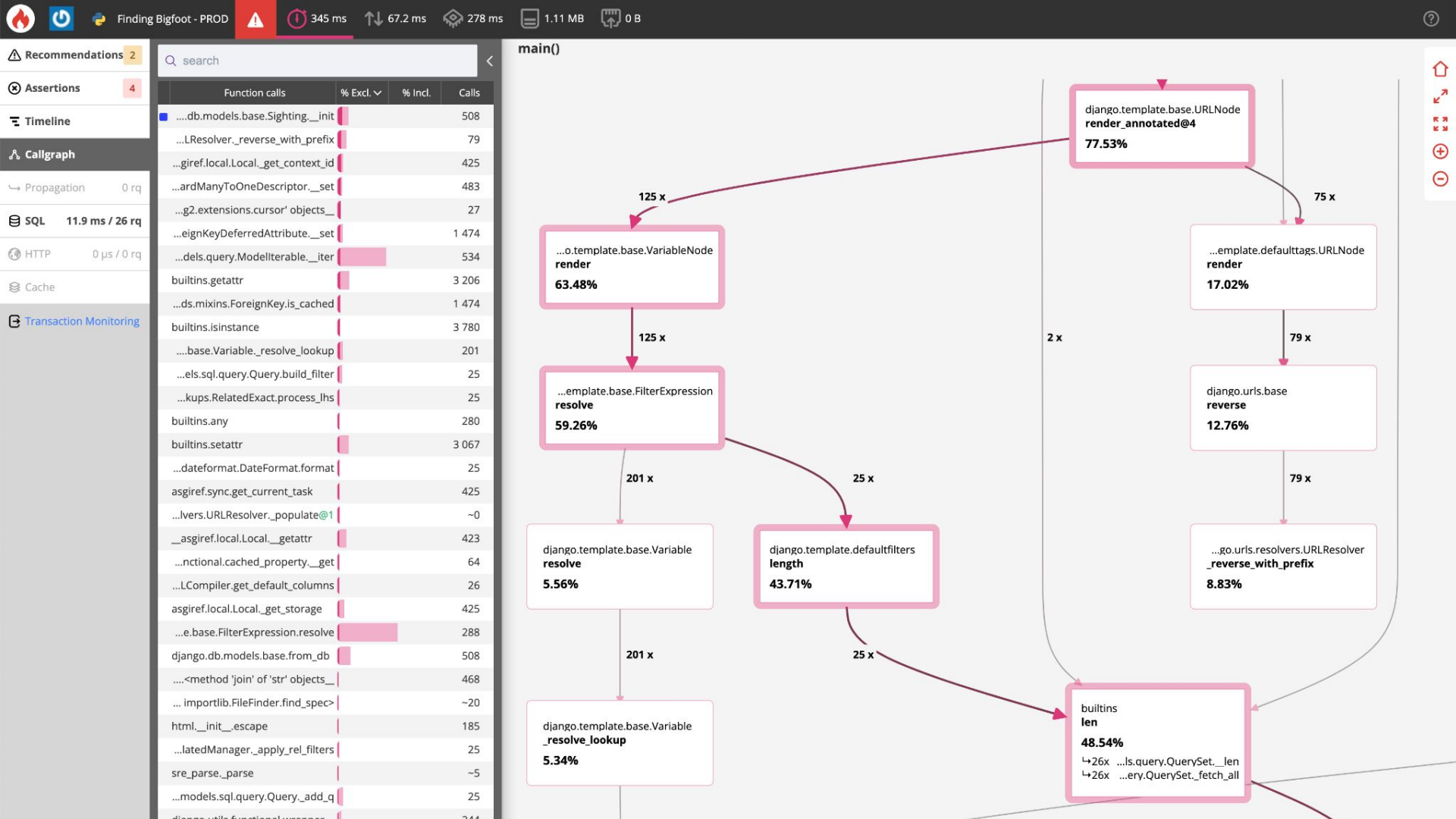
10 µs average

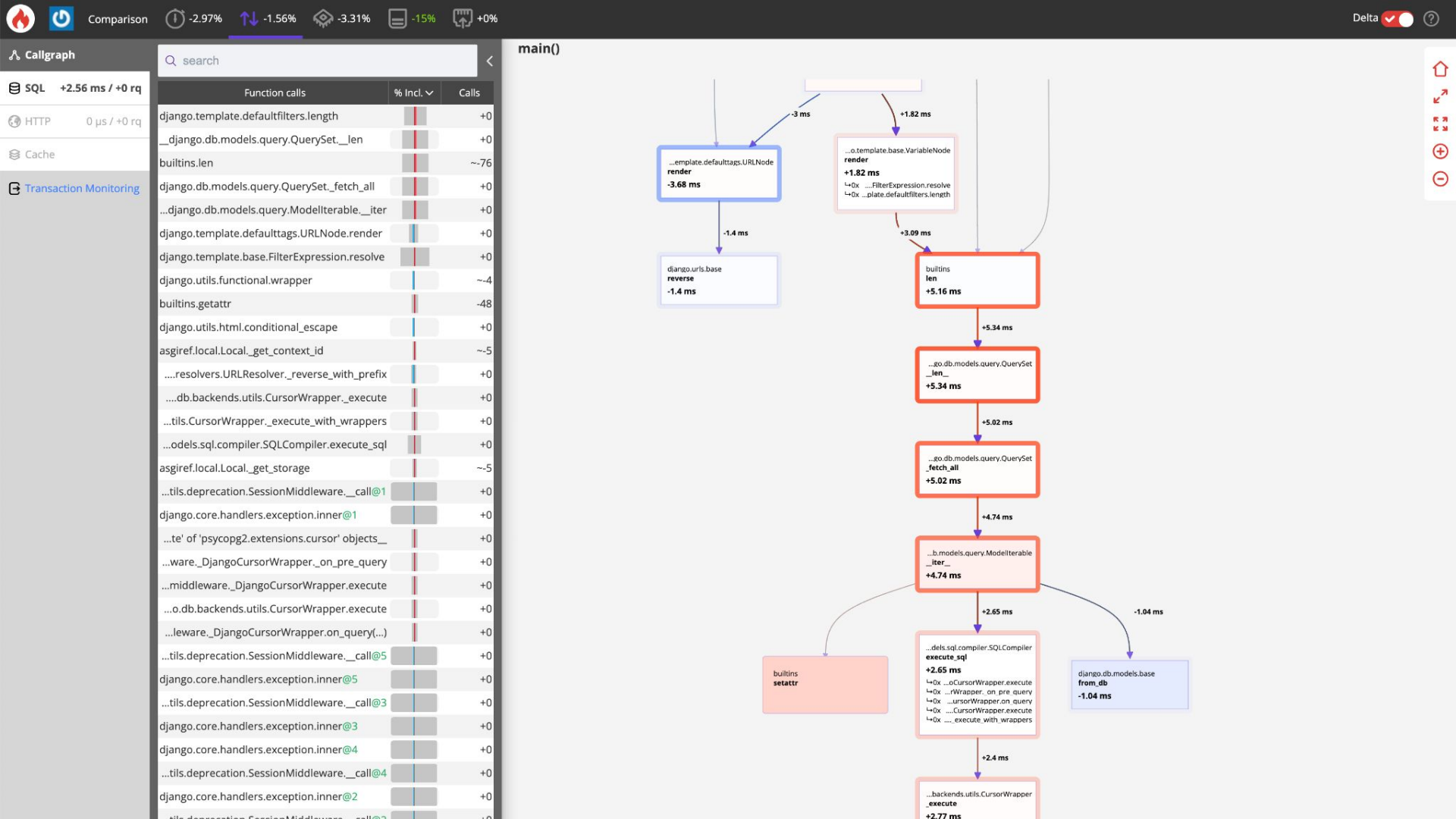
Top Transactions | Top Services

Transactions Wall-Time Wall-Time Consumed



Name	Impact	Requests	Response Time	96th Percentile	Memory	Errors
App\Controller\DocController::docAction	37.2%	742	55.2 ms	72.5 ms	6.78 MB	-
App\Controller\SlnController::indexAction	9.56%	642	16.4 ms	22.7 ms	2.28 MB	-
App\Component\SearchComponent::__invoke	8.2%	88	102 ms	138 ms	2.39 MB	-
App\Controller\ContentController::homepageAction	6.64%	218	33.5 ms	44.4 ms	3 MB	-
App\Controller\BlogController::indexAction	6.61%	76	95.6 ms	138 ms	3.36 MB	-


Finding Bigfoot - PROD
345 ms
67.2 ms
278 ms
1.11 MB
0 B
Recommendations 2
Assertions 4
Timeline
Callgraph
Propagation 0 rq
SQL 11.9 ms / 26 rq
HTTP 0 µs / 0 rq
Cache
Transaction Monitoring
search
Function calls
% Excl.
% Incl.
Calls
....db.models.base.Sighting.__init 508
...LResolver._reverse_with_prefix 79
...giref.local.Local._get_context_id 425
...ardManyToOneDescriptor.__set 483
...g2.extensions.cursor' objects__ 27
...eignKeyDeferredAttribute.__set 1 474
...dels.query.ModelIterable.__iter 534
builtins.getattr 3 206
...ds.mixins.ForeignKey.is_cached 1 474
builtins.isinstance 3 780
....base.Variable._resolve_lookup 201
...els.sql.query.Query.build_filter 25
...kups.RelatedExact.process_lhs 25
builtins.any 280
builtins.setattr 3 067
...dateformat.DateFormat.format 25
asgiref.sync.get_current_task 425
...lvers.URLResolver._populate@1 ~0
__asgiref.local.Local.__getattr 423
...nctional.cached_property.__get 64
...LCompiler.get_default_columns 26
asgiref.local.Local._get_storage 425
...e.base.FilterExpression.resolve 288
django.db.models.base.from_db 508
....<method 'join' of 'str' objects__ 468
... importlib.FileFinder.find_spec> ~20
html.__init__.escape 185
...latedManager._apply_rel_filters 25
sre_parse._parse ~5
...models.sql.query.Query._add_q 25
main()
django.template.base.URLNode
render_annotated@4
77.53%
125 x
75 x
...o.template.base.VariableNode
render
63.48%
...emplate.defaulttags.URLNode
render
17.02%
125 x
2 x
79 x
...emplate.base.FilterExpression
resolve
59.26%
django.urls.base
reverse
12.76%
201 x
25 x
79 x
django.template.base.Variable
resolve
5.56%
django.template.defaultfilters
length
43.71%
...go.urls.resolvers.URLResolver
_reverse_with_prefix
8.83%
201 x
25 x
django.template.base.Variable
_resolve_lookup
5.34%
builtins
len
48.54%
↳26x ...ls.query.QuerySet.__len
↳26x ...ery.QuerySet._fetch_all


Comparison
-2.97%
-1.56%
-3.31%
-15%
+0%
Delta
Callgraph
SQL +2.56 ms / +0 rq
HTTP 0 µs / +0 rq
Cache
Transaction Monitoring
search
Function calls
% Incl.
Calls
django.template.defaultfilters.length +0
__django.db.models.query.QuerySet.__len +0
builtins.len ~-76
django.db.models.query.QuerySet._fetch_all +0
...django.db.models.query.ModelIterable.__iter +0
django.template.defaulttags.URLNode.render +0
django.template.base.FilterExpression.resolve +0
django.utils.functional.wrapper ~-4
builtins.getattr -48
django.utils.html.conditional_escape +0
asgiref.local.Local._get_context_id ~-5
....resolvers.URLResolver._reverse_with_prefix +0
....db.backends.utils.CursorWrapper._execute +0
...tils.CursorWrapper._execute_with_wrappers +0
...odels.sql.compiler.SQLCompiler.execute_sql +0
asgiref.local.Local._get_storage ~-5
...tils.deprecation.SessionMiddleware.__call@1 +0
django.core.handlers.exception.inner@1 +0
...te' of 'psycopg2.extensions.cursor' objects__ +0
...ware._DjangoCursorWrapper._on_pre_query +0
...middleware._DjangoCursorWrapper.execute +0
...o.db.backends.utils.CursorWrapper.execute +0
...leware._DjangoCursorWrapper.on_query(...) +0
...tils.deprecation.SessionMiddleware.__call@5 +0
django.core.handlers.exception.inner@5 +0
...tils.deprecation.SessionMiddleware.__call@3 +0
django.core.handlers.exception.inner@3 +0
django.core.handlers.exception.inner@4 +0
...tils.deprecation.SessionMiddleware.__call@4 +0
django.core.handlers.exception.inner@2 +0
main()
-3 ms
+1.82 ms
...emplate.defaulttags.URLNode
render
-3.68 ms
...o.template.base.VariableNode
render
+1.82 ms
0x ....FilterExpression.resolve
0x ...plate.defaultfilters.length
-1.4 ms
+3.09 ms
django.urls.base
reverse
-1.4 ms
builtins
len
+5.16 ms
+5.34 ms
...go.db.models.query.QuerySet
__len__
+5.34 ms
+5.02 ms
...go.db.models.query.QuerySet
_fetch_all
+5.02 ms
+4.74 ms
...b.models.query.ModelIterable
__iter__
+4.74 ms
+2.65 ms
-1.04 ms
builtins
setattr
...dels.sql.compiler.SQLCompiler
execute_sql
+2.65 ms
0x ...oCursorWrapper.execute
0x ...rWrapper._on_pre_query
0x ...ursorWrapper.on_query
0x ....CursorWrapper.execute
0x ...._execute_with_wrappers
django.db.models.base
from_db
-1.04 ms
+2.4 ms
...backends.utils.CursorWrapper
_execute
+2.77 ms

Tests: blocking regressions

Enforce a performance budget for known user scenarios with performance tests.

Prevent performance regressions and take up dedicated performance motivated sprints to make optimization a primary objective.

```
tests:
    'The homepage should be fast':
        path:
          - '/'
        assertions:
            - 'main.wall_time <= 250ms'

    "The homepage should not consume too much memory":
        path: "/"
        assertions:
            - "main.peak_memory < 2mb"

    "The number of created entities should be reasonable":
        path: "/.*"
        assertions:
            - "metrics.entities.created.count < 50"

    "The homepage should have a limited number of SQL queries":
        path: "/"
        assertions:
            - "metrics.sql.queries.count <= 5"
scenarios: |
    #!blackfire-player

    name: "Anonymous visits"

    scenario
        visit url('/user/login')
            expect status_code() == 200
        visit url('/user/password')
            expect status_code() == 200
        visit url('/contact')
            expect status_code() == 200
```

#2 Prod Triggered Build

View in Admin

Finished *less than a minute ago* - Build triggered by Webhook

DETAILS OUTPUT NOTIFICATIONS

2 recommendations are common to all profiles

Basic Visit scenario

Homepage

Show Profile

1 successful and 4 failing assertions

https://main-bvxea6i-colxodint4t7g.ca-1.platformsh.site/

Assertions

✗	Comment count should be annotated to Sighting entity	`metrics.app.comment_count_annotation.count 0 > 0`
✗	The homepage should be fast	`main.wall_time 313ms < 100ms`
✗	The homepage should have a limited number of SQL queries	`metrics.sql.queries.count 26 <= 5`
✓	The homepage should not consume too much memory	`main.peak_memory 565kB < 2mb`
✗	The number of created entities should be reasonable	`metrics.entities.created.count 508 < 50`

First sighting page

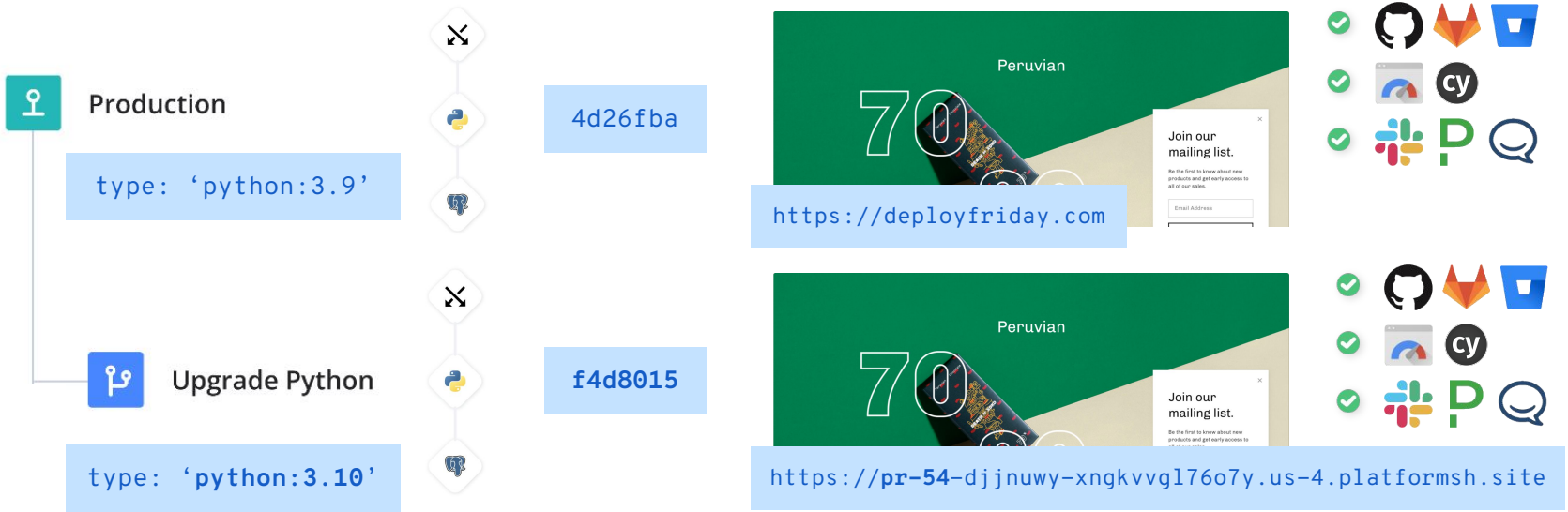
Show Profile

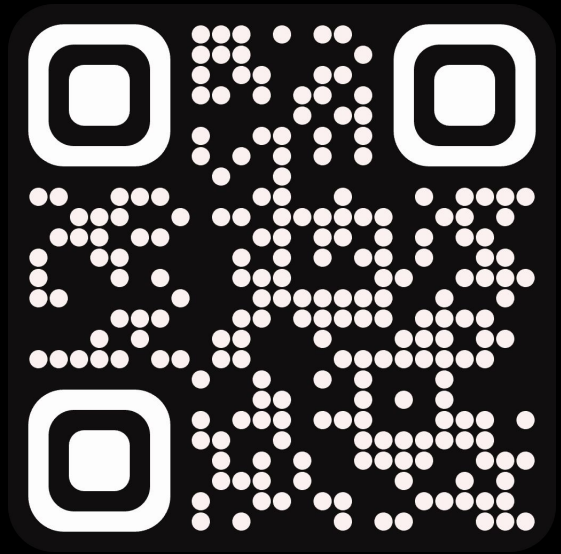
2 failing assertions

https://main-bvxea6i-colxodint4t7g.ca-1.platformsh.site/sighting/3/

True staging environments for every PR

Isolated, byte-level copies of code, infrastructure, and production data with **every** git branch and PR.



upsun

Register for early access.