

# Déjà Vu

Declarative  
Assembly of Web  
Applications from  
Predefined  
Concepts

**Santiago Perez De Rosso**

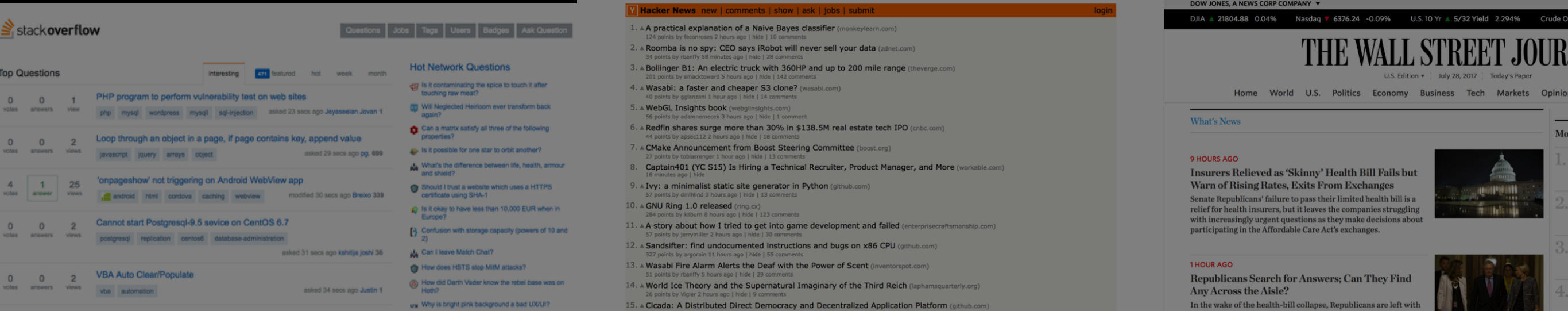
*SPLASH 2019 · Oct 24, 2019*

Work done in collaboration with Daniel Jackson,  
Maryam Archie, Czarina Lao, and Barry A. McNamara III



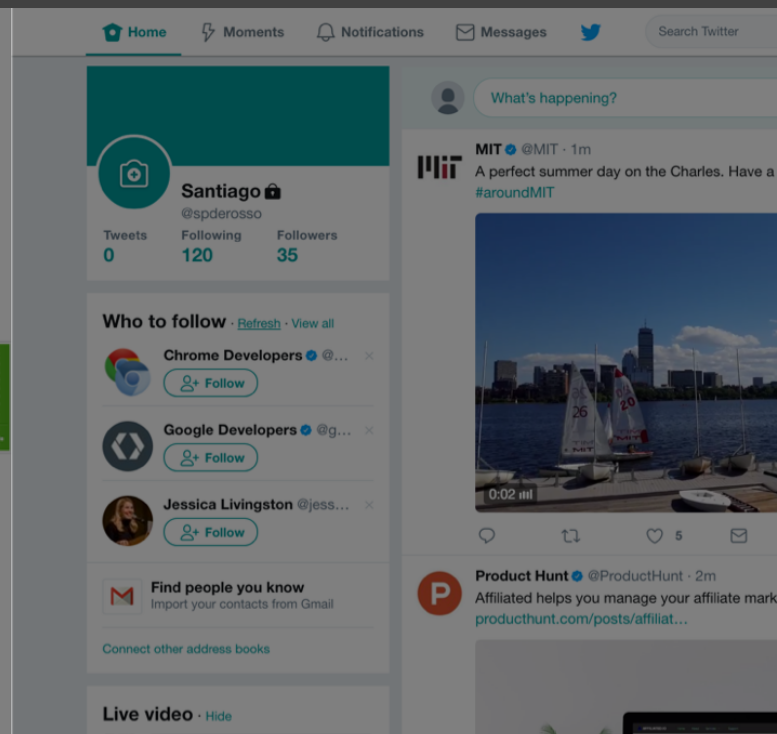
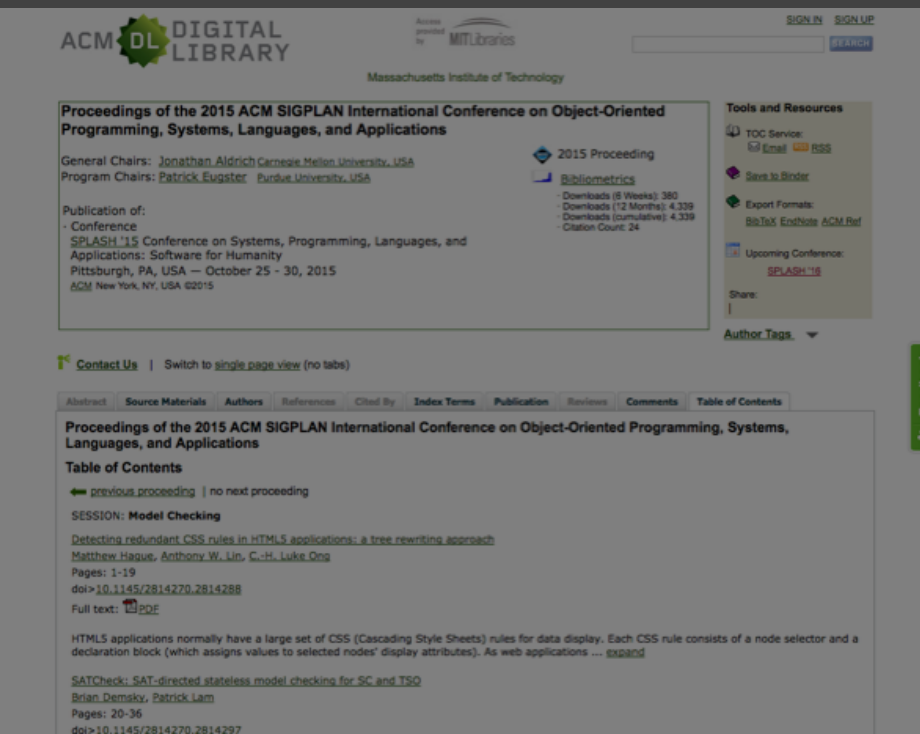
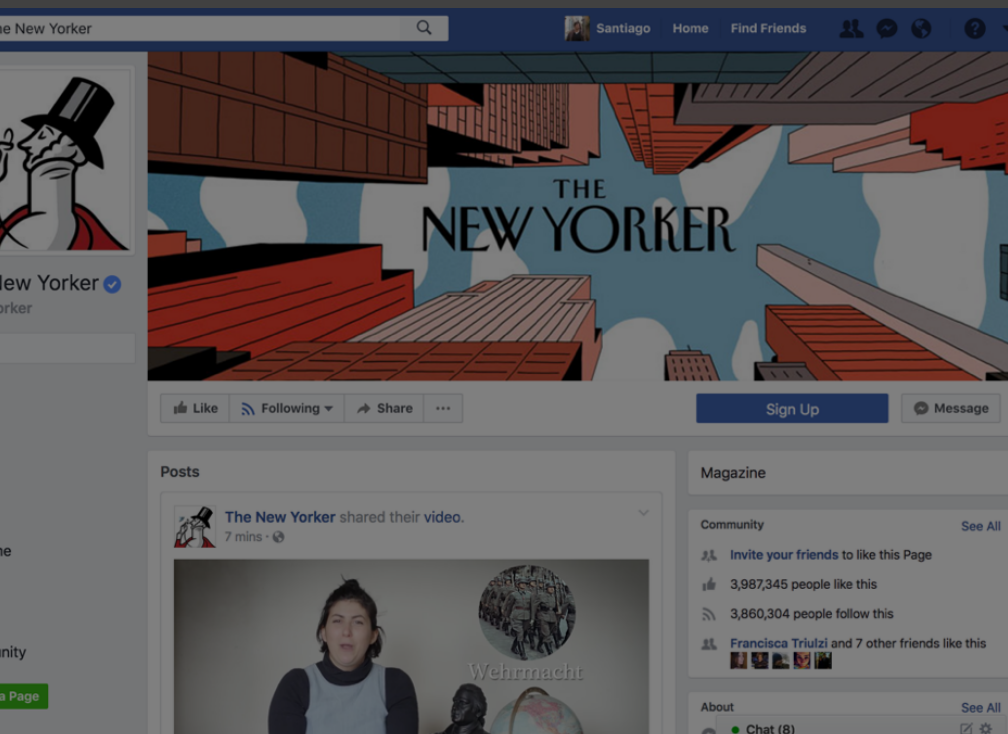
Funded by





# Have you ever noticed most apps are fundamentally the same?

They are mostly combining pre-existing concepts in new ways





HomeMomentsNotificationsMessages

Search Twitter

Tweet

Santiago

@spderosso

Tweets0Following120Followers35

Who to follow · Refresh · View all

Chrome Developers

Follow

Google Developers

Follow

Jessica Livingston

@jess...

Follow

Find people you know

Import your contacts from Gmail

Connect other address books


Live video · Hide

What's happening?

MIT

@MIT · 1m

A perfect summer day on the Charles. Have a great weekend! #FridayFeeling #aroundMIT



0:02

5

Product Hunt

@ProductHunt · 2m

Affiliated helps you manage your affiliate marketing beautifully

producthunt.com/posts/affiliat...

Hacker News

new | comments | show | ask | jobs | submit

login

1. A practical explanation of a Naive Bayes classifier (monkeylearn.com)

124 points by feconroses 2 hours ago | hide | 10 comments

2. Roomba is no spy: CEO says iRobot will never sell your data (zdnet.com)

34 points by rbanffy 58 minutes ago | hide | 28 comments

3. Bollinger B1: An electric truck with 360HP and up to 200 mile range (theverge.com)

201 points by smacktoward 5 hours ago | hide | 142 comments

4. Wasabi: a faster and cheaper S3 clone? (wasabi.com)

40 points by gglanzani 1 hour ago | hide | 14 comments

5. WebGL Insights book (webglinsights.com)

56 points by adamnemecek 3 hours ago | hide | 1 comment

6. Redfin shares surge more than 30% in \$138.5M real estate tech IPO (cnbc.com)

44 points by apsec112 2 hours ago | hide | 18 comments

7. CMake Announcement from Boost Steering Committee (boost.org)

27 points by tobiasrenger 1 hour ago | hide | 13 comments

8. Captain401 (YC S15) Is Hiring a Technical Recruiter, Product Manager, and More (workable.com)

16 minutes ago | hide

9. Ivy: a minimalist static site generator in Python (github.com)

57 points by dmlhlnd 3 hours ago | hide | 13 comments

10. GNU Ring 1.0 released (ring.cx)

284 points by kilburn 8 hours ago | hide | 123 comments

11. A story about how I tried to get into game development and failed (enterprisecraftsmanship.com)

57 points by jerrymiller 2 hours ago | hide | 30 comments

12. Sandsifter: find undocumented instructions and bugs on x86 CPU (github.com)

327 points by argorain 11 hours ago | hide | 55 comments

13. Wasabi Fire Alarm Alerts the Deaf with the Power of Scent (inventorspot.com)

51 points by rbanffy 5 hours ago | hide | 29 comments

14. World Ice Theory and the Supernatural Imaginary of the Third Reich (laphamsquarterly.org)

26 points by Vigier 2 hours ago | hide | 9 comments

15. Cicada: A Distributed Direct Democracy and Decentralized Application Platform (github.com)

92 points by dustingetz 3 hours ago | hide | 80 comments

16. Future of Fake News (futureoffakenews.com)

14 points by mllllike 2 hours ago | hide | 4 comments

17. AMD Threadripper Delidded, with a Multi-Core Surprise Under the Hood (extremetech.com)

78 points by pantalaimon 2 hours ago | hide | 31 comments

18. How I've Lived the 4-Hour Work Week for a Decade (indiehackers.com)

24 points by yvoschaap2 54 minutes ago | hide | 13 comments

19. Satirical maps of the world (bl.uk)

16 points by Thevet 3 hours ago | hide | discuss

DOW JONES, A NEWS CORP COMPANY

DJIA 21804.88 0.04% Nasdaq 6376.24 -0.09% U.S. 10 Yr 5/32 Yield 2.294% Crude Oil 49.58 1.10% Euro 1.1746 0.60%

THE WALL STREET JOURNAL.

U.S. Edition July 28, 2017 Today's Paper

Home World U.S. Politics Economy Business Tech Markets Opinion Arts Life Real Estate

What's News

9 HOURS AGO

Insurers Relieved as 'Skinny' Health Bill Fails but Warn of Rising Rates, Exits From Exchanges

Senate Republicans' failure to pass their limited health bill is a relief for health insurers, but it leaves the companies struggling with increasingly urgent questions as they make decisions about participating in the Affordable Care Act's exchanges.

1 HOUR AGO

Republicans Search for Answers; Can They Find Any Across the Aisle?

In the wake of the health-bill collapse, Republicans are left with no obvious path forward. Except perhaps, Gerald F. Seib writes, to try to find a way to work with Democrats.

10 HOURS AGO

'Skinny' Repeal of Obamacare Fails in Senate

The slimmed-down health-law repeal was the Republicans' last-

Most Popular Videos

1. Sen. Graham: The 'Skinny' Bill Is a Disaster

2. John McCain Casts Deciding Vote to Defeat Health Bill

3. Dan Neil Drives VW's Atlas


4. Who and What Is Covered by Medicaid?

5. How Luxury Home Builders Build for Themselves

Most Popular Articles

The New Yorker

Santiago Home Find Friends




The New Yorker

@newyorker

Home Posts About Photos Events Videos Welcome Notes Groups Community

Create a Page




Like Following Share

Sign Up Message

Posts

The New Yorker shared their video.

7 mins



Magazine

Community

Invite your friends to like this Page

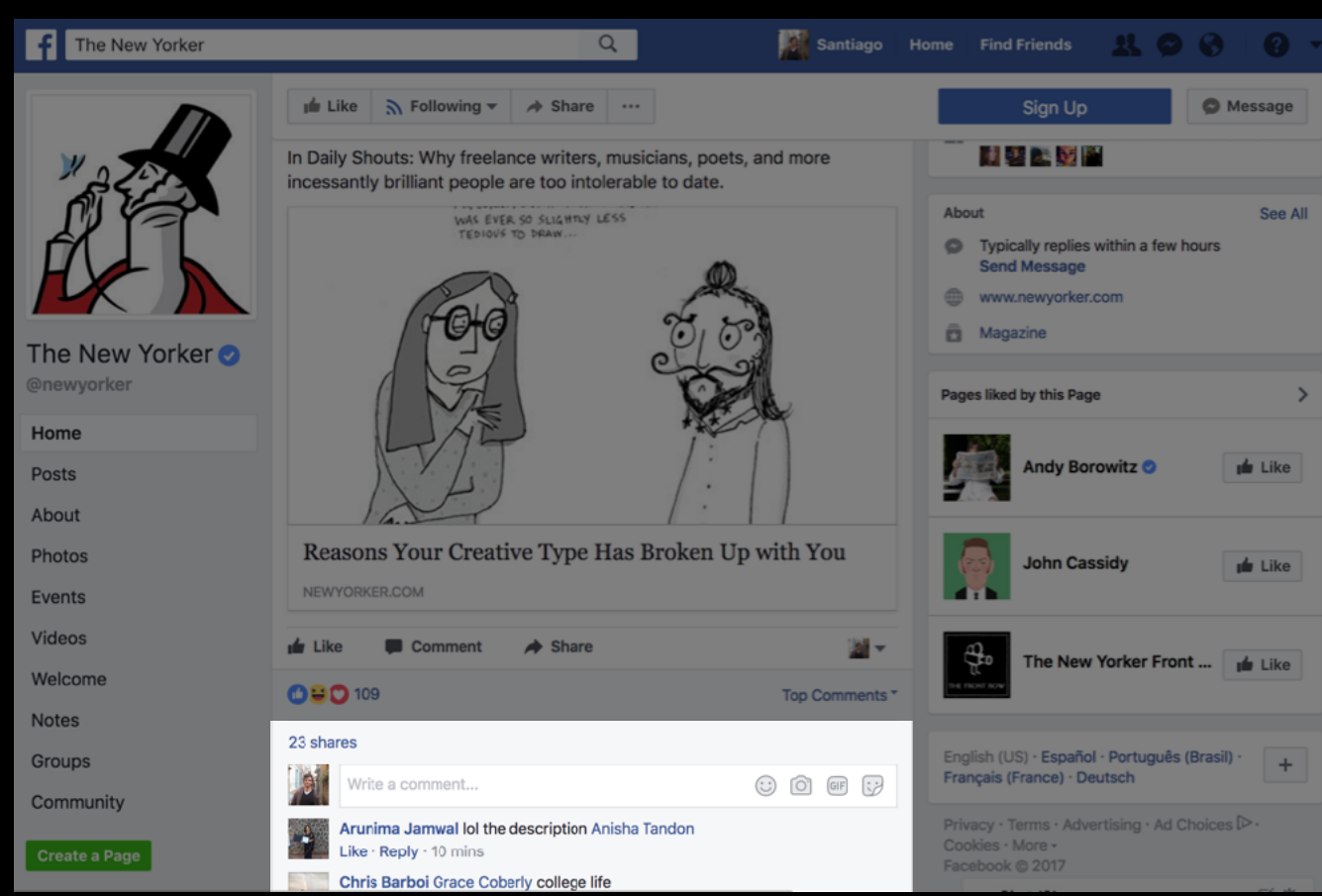
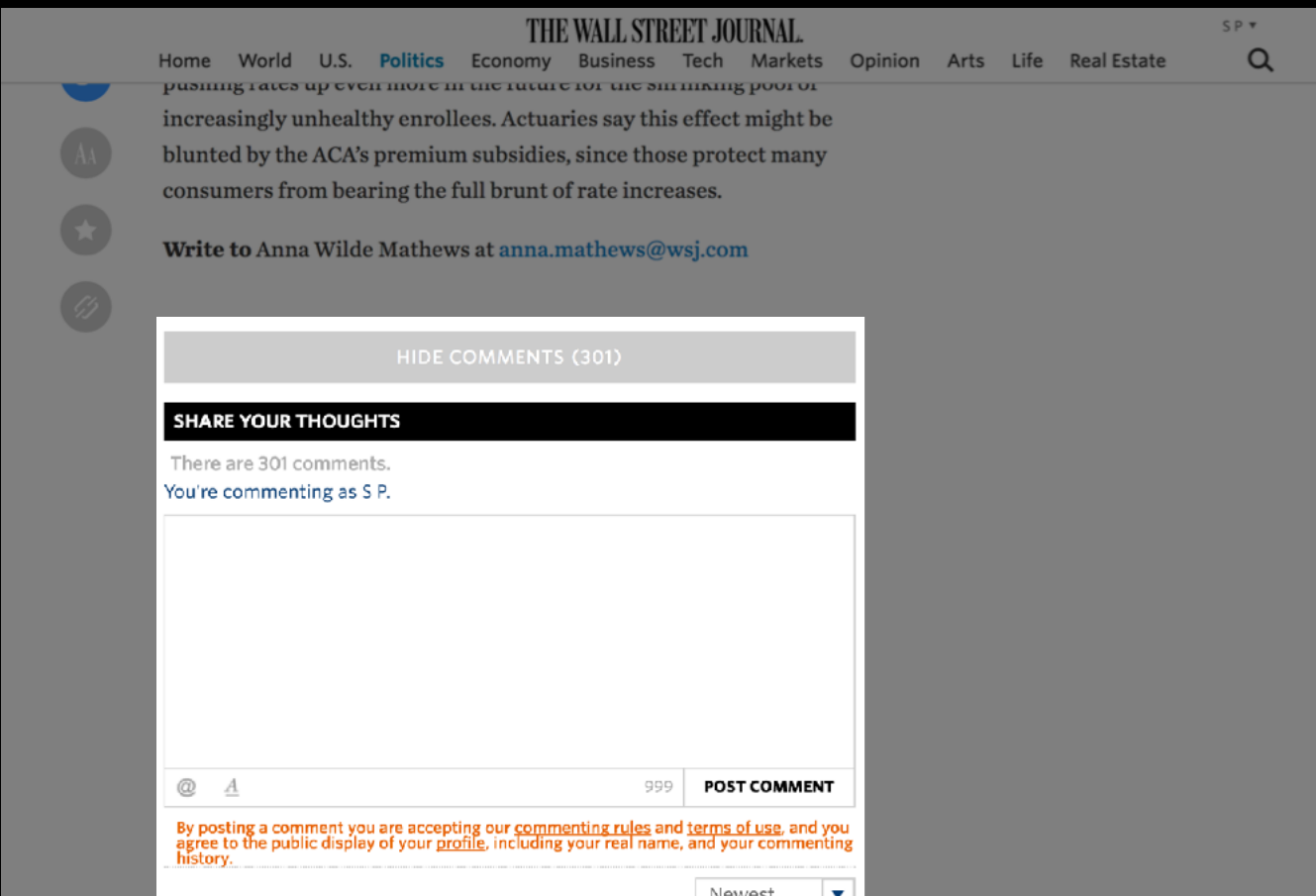
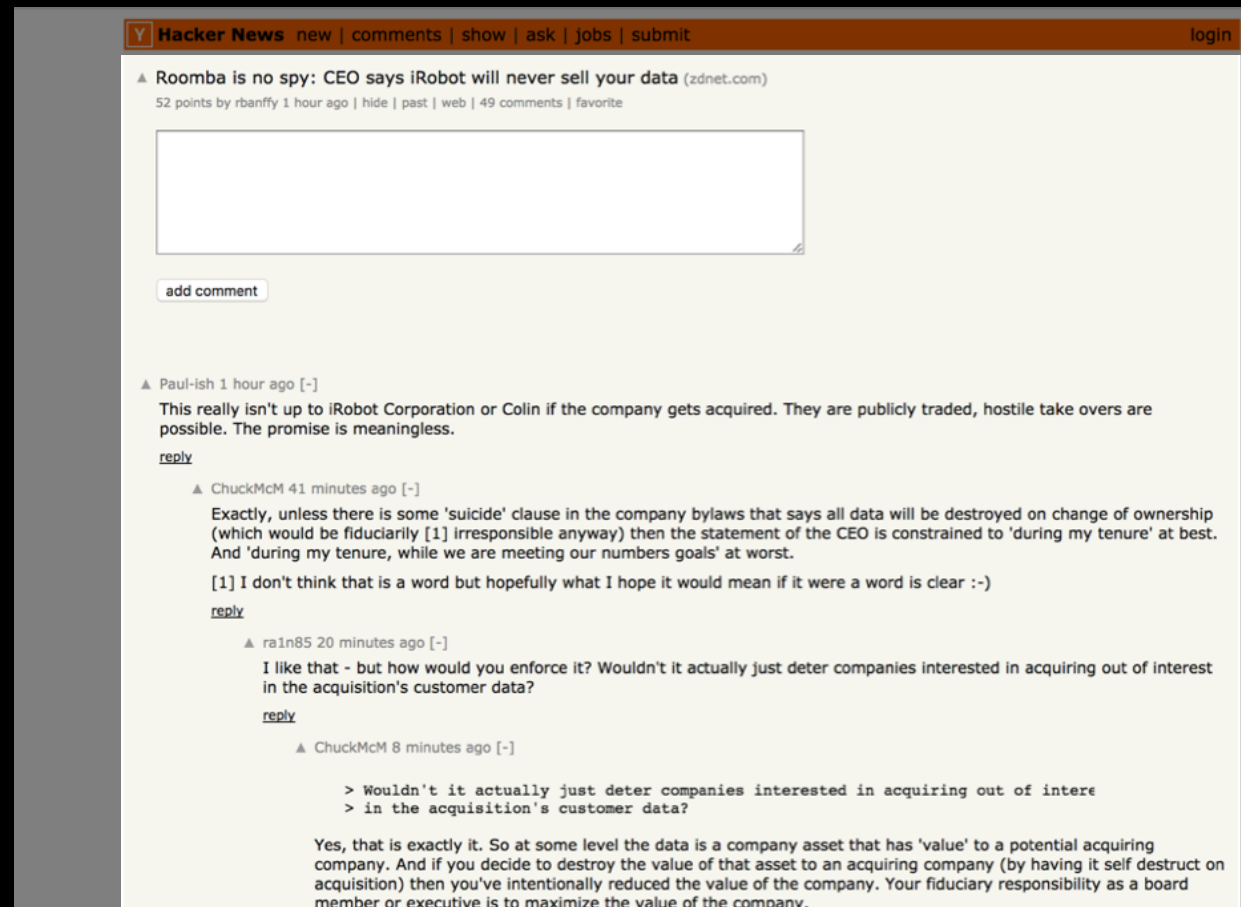
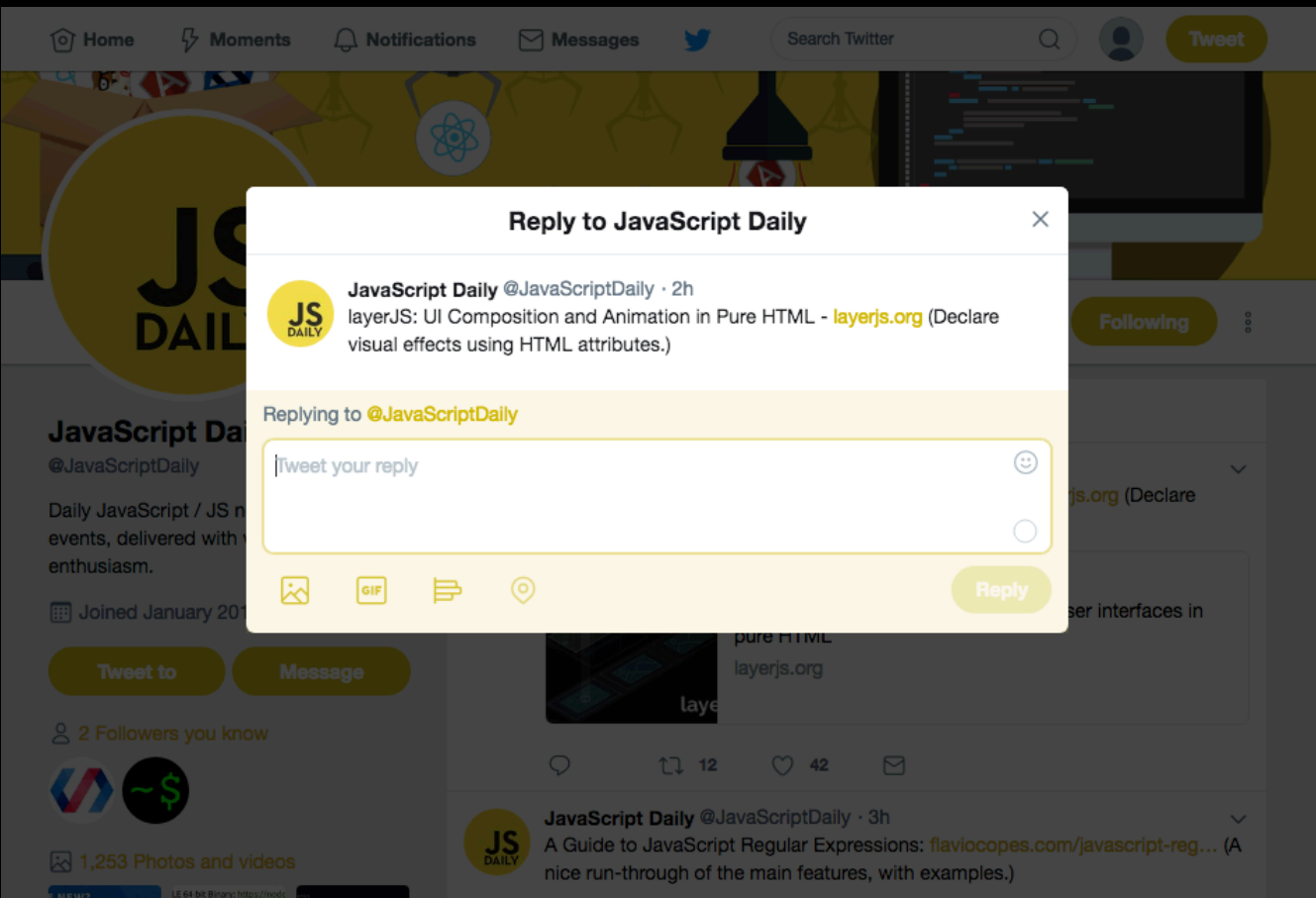
3,987,345 people like this

3,860,304 people follow this

Francisca Trulzi and 7 other friends like this

About

Chat (8)





Home

Moments

Notifications

Messages

Search Twitter

Tweet

JS

DAILY

Tweets

Following

Followers

Likes

Moments

Following

JavaScript Daily

@JavaScriptDaily

Daily JavaScript / JS news, links and events, delivered with wit and enthusiasm.

Joined January 2012

Tweet to

Message

2 Followers you know

1,253 Photos and videos

Tweets

Tweets & replies

Media

JavaScript Daily @JavaScriptDaily · 2h

layerJS: UI Composition and Animation in Pure HTML - [layerjs.org](#) (Declare visual effects using HTML attributes.)

layerJS

layerJS: build animated layered web user interfaces in pure HTML

layerjs.org

Like

42

JavaScript Daily @JavaScriptDaily · 3h

A Guide to JavaScript Regular Expressions: [flaviocopes.com/javascript-reg...](#) (A nice run-through of the main features, with examples.)

Hacker News

new | comments | show | ask | jobs | submit

login

Roomba is no spy: CEO says iRobot will never sell your data (zdnet.com)

52 points by rbanffy 1 hour ago | hide | past | web | 49 comments | favorite

add comment

Paul-ish 1 hour ago [-]

This really isn't up to iRobot Corporation or Colin if the company gets acquired. They are publicly traded, hostile take overs are possible. The promise is meaningless.

reply

ChuckMcM 41 minutes ago [-]

Exactly, unless there is some 'suicide' clause in the company bylaws that says all data will be destroyed on change of ownership (which would be fiduciarilly [1] irresponsible anyway) then the statement of the CEO is constrained to 'during my tenure' at best. And 'during my tenure, while we are meeting our numbers goals' at worst.

[1] I don't think that is a word but hopefully what I hope it would mean if it were a word is clear :-)

reply

ra1n85 20 minutes ago [-]

I like that - but how would you enforce it? Wouldn't it actually just deter companies interested in acquiring out of interest in the acquisition's customer data?

reply

ChuckMcM 8 minutes ago [-]

> Wouldn't it actually just deter companies interested in acquiring out of intere

> in the acquisition's customer data?

Yes, that is exactly it. So at some level the data is a company asset that has 'value' to a potential acquiring company. And if you decide to destroy the value of that asset to an acquiring company (by having it self destruct on acquisition) then you've intentionally reduced the value of the company. Your fiduciary responsibility as a board member or executive is to maximize the value of the company.

THE WALL STREET JOURNAL

SP

Home

World

U.S.

Politics

Economy

Business

Tech

Markets

Opinion

Arts

Life

Real Estate

federal governments, for their people.

38 minutes ago

Like

3

Rustem Sharipov

★

Elitist and reactionary US government is completely disconnected from the realities of American life

39 minutes ago

Like

3

Steven Scharf

★

@Rustem Sharipov What an insightful input. Thanks.

14 minutes ago

Like

3

Wayne Gregersen

★

I do not understand at all you "Freedom" advocating right wingers who seem to understand that everyone needs to buy auto insurance to keep auto insurance affordable for all of us, but not understand the need to have universal health care for the same reason. The problem with health insurance is if you do not buy health insurance you still get free care if you have a catastrophic illness or injury. We don't let people die in the streets. So for those

The New Yorker

Santiago

Home

Find Friends

Sign Up

Message

Like

Following

Share

...

In Daily Shouts: Why freelance writers, musicians, poets, and more incessantly brilliant people are too intolerable to date.

Reactions

Comments

Shares

Reasons Your Creative Type Has Broken Up with You

NEWYORKER.COM

Like

Comment

Share

109

Top Comments

23 shares

Write a comment...

Write a comment...

Arunima Jamwal lol the description Anisha Tandon

Like · Reply · 10 mins

Chris Barboi Grace Coberly college life

About

See All

Typically replies within a few hours

Send Message

www.newyorker.com

Magazine

Pages liked by this Page

Andy Borowitz

Like

John Cassidy

Like

The New Yorker Front ...

Like

English (US) · Español · Português (Brasil) · Français (France) · Deutsch

Privacy · Terms · Advertising · Ad Choices · Cookies · More

Facebook © 2017



Language: English ▾

Have an account? [Log in ▾](#)

## Join Twitter today.

Full name

Phone or Email

Password

☒ Personalize Twitter based on where you've seen Twitter content on the web. [Learn more.](#)

Sign up

By signing up, you agree to the [Terms of Service](#) and [Privacy Policy](#), including [Cookie Use](#). Others will be able to find you by email or phone number when provided.

[Advanced options](#)

### Login

username:

password:

[login](#)

[Forgot your password?](#)

### Create Account

username:

password:

[create account](#)

## THE WALL STREET JOURNAL.

### Sign In

Email or username

Password

 [SHOW](#)

☒ Keep me signed in

SIGN IN



Sign in with Facebook



Sign in with Google

[Forgot your username or password?](#)  
Not a member? [Click here to learn more!](#)

POWERED BY DOW JONES

# facebook

Email or Phone

Password

[Log In](#)

[Forgot account?](#)

## Sign Up

It's free and always will be.

First name

Last name

Mobile number or email

New password

Birthday

Jul ▾

31 ▾

1999 ▾

[Why do I need to provide my birthday?](#)

☐ Female

☐ Male

By clicking Create Account, you agree to our [Terms](#) and that you have read our [Data Policy](#), including our [Cookie Use](#). You may receive SMS Notifications from Facebook and can opt out at any time.

Create Account

[Create a Page](#) for a celebrity, band or business.

## Connect with friends and the world around you on Facebook.



See photos and updates from friends in News Feed.



Share what's new in your life on your Timeline.



Find more of what you're looking for with Facebook Search.



# Low-level Reuse

- ▶ web frameworks: Angular, React, Vue, ...
- ▶ HTTP: Express.js, ...
- ▶ API: GraphQL, ...
- ▶ various libraries: bcryptjs, jsonwebtoken, ...

**what about concept reuse?**

# Problem

- ▶ wasted effort (in concept design & impl)
- ▶ errors (in concept design & impl)





## your shopping bag

### Item Details

Leaves warehouse in 1 - 2 full bus. days.


[Argentina Training Top](#)

Color: Pride Ink/White

Size: L

Article#: F89022

Qty: 
[UPDATE CART](#)

Item price \$68.00

[Remove Item](#) | [Move to Wishlist](#)

Leaves warehouse in 1 - 2 full bus. days.


[Argentina Track Jacket](#)

Color: Bahia Lt Blue/Navy

Size: L

Article#: G77787

Qty: 

! Quantity can not be zero.

[UPDATE CART](#)

Item price \$59.00

[Remove Item](#) | [Move to Wishlist](#)

### Order Summary

Merchandise Subtotal	\$127.00
Shipping and Handling	\$5.00
Free Shipping on \$49 or more ( <a href="#">See Promotion Details</a> ):	-\$5.00

Pre-Tax Total **\$127.00**
[checkout](#)

### Using a Promo Code? (Optional)

☒ No ☐ Yes

[checkout](#)

OR

Check Out with 



## your shopping bag

### Item Details

Leaves warehouse in 1 - 2 full bus. days.



[Argentina Training Top](#)

Color: Pride Ink/White

Size: L

Article#: F89022

Qty:

[UPDATE CART](#)

Item price \$68.00

[Remove Item](#) | [Move to Wishlist](#)

Leaves warehouse in 1 - 2 full bus. days.



[Argentina Track Jacket](#)

Color: Bahia Lt Blue/Navy

Size: L

Article#: G77787

Qty:



Quantity can not be zero.

[UPDATE CART](#)

Item price \$59.00

[Remove Item](#) | [Move to Wishlist](#)

### Order Summary

Merchandise Subtotal \$127.00

Shipping and Handling \$5.00

Free Shipping on \$49 or more ([See Promotion Details](#)): -\$5.00

Pre-Tax Total \$127.00

[checkout](#)

### Using a Promo Code? (Optional)



No



Yes


[checkout](#)

OR

Check Out with 



# Learning Web Development Circa 2019

**6.170 Software Studio**  
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

[LOGIN](#)

[Course : » Course 6 : » Fall 2018 : » 6.170 : »Homepage](#)

[Class Home](#)

[Materials](#)

[Calendar](#)

[Gradebook Module](#)

[Piazza](#)

[Membership](#)

[Staff List](#)

[Search](#)

[STELLAR HELP ?](#)

[Contact Help Desk](#)

## 6.170 Software Studio

### Fall 2018

**Instructors:** Daniel Jackson, Arvind Satyanarayan

**TAs:** Maryam Archie, Zhaozheng Jin, Barry A McNamara, Mary G Thielking

**Lecture:** MW2.30-4 (32-123)

**Information:**

**Recitation Times (Thursdays)**

10am (38-166) - Maryam  
11am (38-166) - Maryam  
12pm (34-301) - Mary  
1pm (34-301) - Mary  
1pm (34-303) - Alice  
2pm (34-303) - Barry  
3pm (34-303) - Barry

**Lecturer Office Hours**

Daniel Jackson (32-G704): Thursday 2-3pm  
Starting Nov. 1, office hours are by appointment. Please email me and I'm happy to find time to chat.

Arvind Satyanarayan (32-G706): Wednesday 4-5pm  
Starting Oct. 31, office hours are by appointment. Please email me and I'm happy to find time to chat.

# Learning Web Development Circa 2019

## HTML



Basic Syntax  
DOM HTML5 &  
Semantic Tags

## CSS



Basic Syntax  
Selectors  
Box Model Flexbox

## JavaScript



Data Types Variables & Scope  
First-class Functions  
Closures List Functionals  
Mixins Prototypes  
DOM Manipulation  
Event Loop Event Handling

## HTTP



URLs, IPs & DNS  
Methods Sessions  
Headers Cookies

## Node.js



Express Routes  
Templates Middleware  
Sessions & Cookies

## Web APIs

Ajax REST  
XMLHttpRequestObject  
Sync vs Async  
Dealing with Async:  
Continuations, Side-effects,  
callback-hell, promises

# Learning Web Development

## Circa 2019

### HTML

Basic Syntax  
DOM  
Semantics

### Databases

NoSQL Relational DBs  
Intro to SQL Foreign keys  
Normalization  
Referential integrity  
Integrity constraints

### CSS



### Abstract Data Design

Basics of data modeling  
Schemas from models  
Design heuristics  
Separation of concerns  
Names & Scope  
Prototypes  
Error Handling

JS

### HTTP

URLs, IPs  
Methods  
Headers

### Reactive Programming

Dataflow Programming  
Functional Reactive  
Programming  
Event Streams Operators  
Reactive variables

### Vue.js



Model-View-Controller  
Model-View-ViewModel

Ajax REST  
XMLHttpRequest  
Sync vs Async

Dealing with Async:  
Continuations, Side-effects,  
callback-hell, promises

## Usability

Easy to use Prototyping  
Learnability Efficiency Visibility User testing  
Error-tolerance Safety Consistency  
Accessibility Usable security Constraints  
Design thinking Fitts's law Heuristic Evaluation  
Heuristics Information scent

## Security

Policy design  
Injection XSS  
CSRF Replay attack  
Man in the middle  
Same Origin Policy

## Databases

NoSQL Relational DBs  
Intro to SQL Foreign keys  
Normalization  
Referential integrity  
Integrity constraints

Basics of data modeling  
Schemas from models  
Design heuristics  
Separation of concerns

## HTTP

URLs, IPs  
Methods  
Headers

## Reactive Programming

Dataflow Programming  
Functional Reactive  
Programming  
Event Streams Operators  
Reactive variables

## Vue.js

Model-View-Controller  
Model-View-ViewModel

Ajax REST  
XMLHttpRequest  
Sync vs Async

Dealing with Async:  
Continuations, Side-effects,  
callback-hell, promises



# Final Projects

# Final Projects

App	Purpose
Accord	Support musical bands in the selection of setlists
ChoreStar	Make it easy for parents to assign chores to children
EasyPick	Recommend classes to college students
GroceryShip	Facilitate peer grocery delivery between students
Lingua	Develop language skills by chatting with native speakers
Listify	Crowdsource opinion-based rankings of anything
LiveScorecard	Provide a live leaderboard for climbing competitions
MapCampus	Allow students to plan events on campus
Phoenix	Help people discuss mental health and make friends
Potluck	Help people plan parties where guests bring supplies
Rendezvous	Plan public events on campus
SweetSpots	Mark spots on a map and review spots added by others

# Final Projects

App	Purpose
Accord	Support musical bands in the selection of setlists
ChoreStar	Make it easy for parents to assign chores to children
EasyPick	Recommend classes to college students
GroceryShip	Facilitate peer grocery delivery between students
Lingua	Develop language skills by chatting with native speakers
Listify	Crowdsource opinion-based rankings of anything
LiveScorecard	Provide a live leaderboard for climbing competitions
MapCampus	Allow students to plan events on campus
Phoenix	Help people discuss mental health and make friends
Potluck	Help people plan parties where guests bring supplies
Rendezvous	Plan public events on campus
SweetSpots	Mark spots on a map and review spots added by others

# Final Projects

App	Purpose
Accord	Support musical bands in the selection of setlists
ChoreStar	Make it easy for parents to assign chores to children
EasyPick	Recommend classes to college students
GroceryShip	Facilitate peer grocery delivery between students
Lingua	Develop language skills by chatting with native speakers
Listify	Crowdsource opinion-based rankings of anything
LiveScorecard	Provide a live leaderboard for climbing competitions
MapCampus	Allow students to plan events on campus
Phoenix	Help people discuss mental health and make friends
Potluck	Help people plan parties where guests bring supplies
Rendezvous	Plan public events on campus
SweetSpots	Mark spots on a map and review spots added by others



# Final Projects

App	Purpose
Accord	Support musical bands in the selection of setlists
ChoreStar	Make it easy for parents to assign chores to children
EasyPick	Recommend classes to college students
GroceryShip	Facilitate peer grocery delivery between students
Lingua	Develop language skills by chatting with native speakers
Listify	Crowdsource opinion-based rankings of anything
LiveScorecard	Provide a live leaderboard for climbing competitions
MapCampus	Allow students to plan events on campus
Phoenix	Help people discuss mental health and make friends
Potluck	Help people plan parties where guests bring supplies
Rendezvous	Plan public events on campus
SweetSpots	Mark spots on a map and review spots added by others

Boston Rock

Add Setlist

Setlists:

Queen Classics

Punk Rock Classics

Punk Rock Classics ⚙

Fro the gig at the pub

Add Proposal

Boulevard of Broken Dreams

by Green Day

View

Added by: Ben Bitdiddle

Your rating: ★☆☆☆☆

Average rating: 1

Select

Self Esteem

by The Offspring

View

Added by: Ben Bitdiddle

Your rating: ★★★★★

Average rating: 5

Select

Pet Sematary

by The Ramones

View

Select

Accord

ben

Rating: ★★★★★

Native Language: English

Learning Language: French

Chat



Lingua

Team Dancing Torpedo // MIT 6.170 Final Project, Fall 2016

Add a spot:

< Back to home

Name

great spot

Category

study space

Floor

3

Review

★★★★☆

meh spot

Submit Spot



SweetSpots

Boston Rock

Punk Rock Classics ⚙

Add Setlist

Setlists:

Queen Classics

Punk Rock Classics

Fro the gig at the pub

Add Proposal

Boulevard of Broken Dreams

by Green Day

View

Added by: Ben Bitdiddle

Your rating: ★☆☆☆☆

Average rating: 1

Select

Self Esteem

by The Offspring

View

Added by: Ben Bitdiddle

Your rating: ★★★★★

Average rating: 5

Select

Pet Sematary

by The Ramones

View

Added by: Ben Bitdiddle

Your rating: ★☆☆

Average rating: 3

Select

Accord

ben

Rating: ★★★★★

Native Language: English

Learning Language: French

Chat

Lingua

Team Dancing Torpedo // MIT 6.170 Final Project, Fall 2016

Add a spot:

< Back to home

Name

great spot

Category

study space

Floor

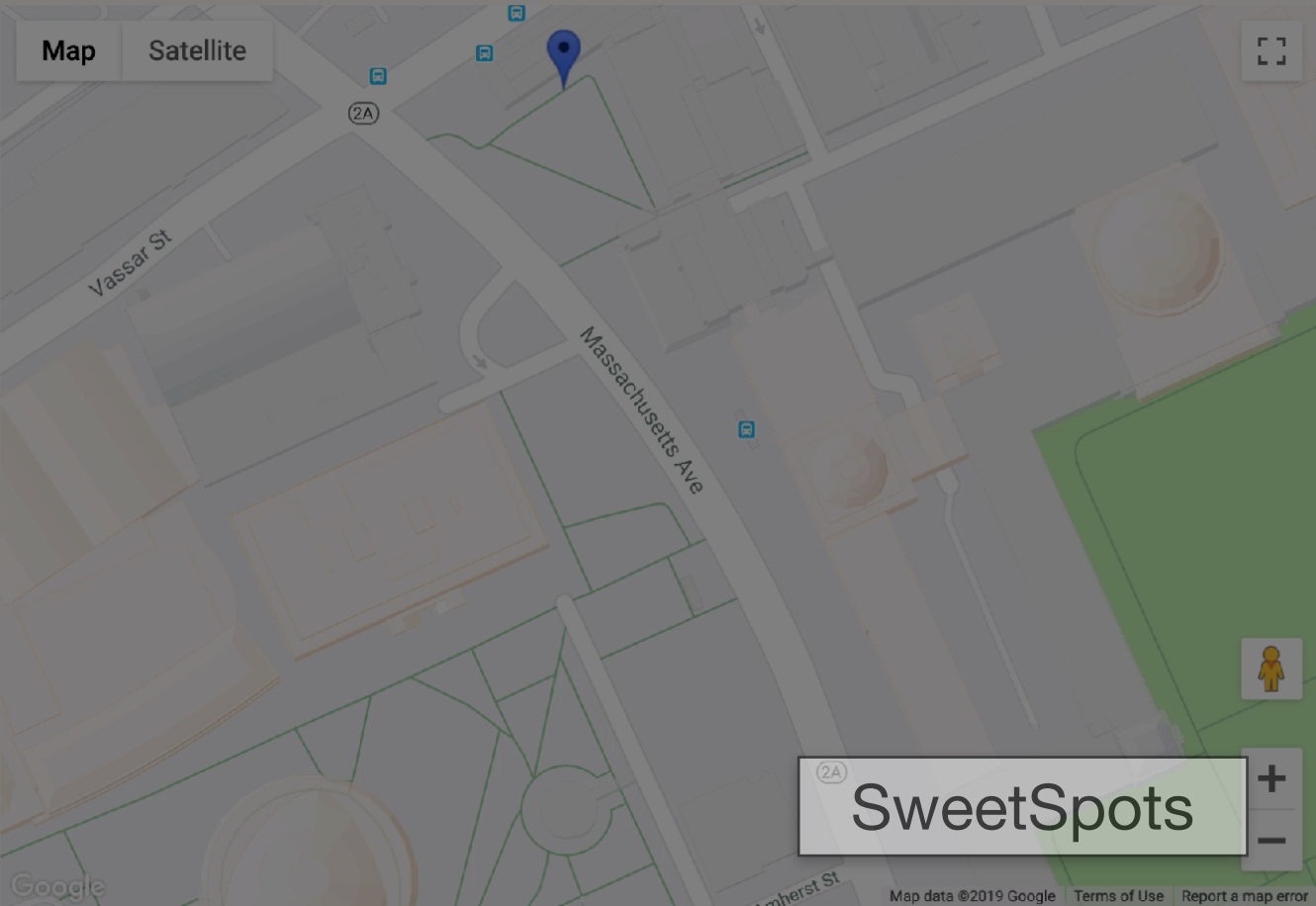
3

Review

★★★★★

meh spot

Submit Spot



SweetSpots

RENDEZVOUS

test

Hosted By sperezde@mit.edu

Attendees 0

Relevant Interests Sports Food & Drink Theater

Location	Date	Start Time	End Time
3B2 Market, Phase 3B2, Sector 60, Sahibzada Ajit Singh Nagar, Punjab, India	11/01/2019	03:00 PM	04:00 PM

Event Details

a description

YOU ARE THE

Rendezvous

Comments

Potluck

ben@bitdiddle.com

Log Out

Create a New Party

Title	Ben's Birthday
Location	My place
Date	03/12/1990, --:-- --
Description	Come celebrate my bday! yay

Guest Name

Guest Email

Add to Guest List

Item

Quantity

Unit

Add Item

Guest List

Name	Email
------	-------

Supply List

Item	Quantity	Unit
------	----------	------

Create Party

Potluck



RENDEZVOUS

test

Hosted By sperezde@mit.edu

Attendees 0

Relevant Interests Sports Food & Drink Theater

Location	Date	Start Time	End Time
3B2 Market, Phase 3B2, Sector 60, Sahibzada Ajit Singh Nagar, Punjab, India	11/01/2019	03:00 PM	04:00 PM

Event Details

a description

YOU ARE THE HOST

Rendezvous

Comments

Potluck

ben@bitdiddle.com

Log Out

Create a New Party

Title	Ben's Birthday		
Location	My place		
Date	03/12/1990, --:-- --		
Description	Come celebrate my bday! yay		

Guest Name

Guest Email

Add to Guest List

Item

Quantity

Unit

Add Item

Guest List



Name	Email
------	-------



Supply List

Item	Quantity	Unit
------	----------	------

Create Party

Potluck

climb #	points	category	
<input type="text" value="climb #"/>	<input type="text" value="points"/>	<div><input type="radio"/> Beginner <input type="radio"/> Intermediate <input checked="" type="radio"/> Advanced <input type="radio"/> Open</div>	
20	20	Advanced	

status	climb info		sent	falls
	#20	A: 20	 0	0 +
<hr/>				
witness 1		<input type="text" value="enter climber code"/>		
witness 2		<input type="text" value="enter climber code"/>		

## Add a New Chore



For Child \*

alice

Name \*

clean bathroom

Star Value \*

10

☐ Expires on

mm/dd/yyyy

Repeats

Never

# Hi alice!

StarBank Total:



## Chores

### Uncompleted Chores

clean bathroom



Expires Sep 25th



### Chores Pending Approval

### Completed and Approved Chores

## Rewards

### Rewards You Can Buy

### Rewards You Cannot Buy Yet

cookies

2 ★

# Opportunity

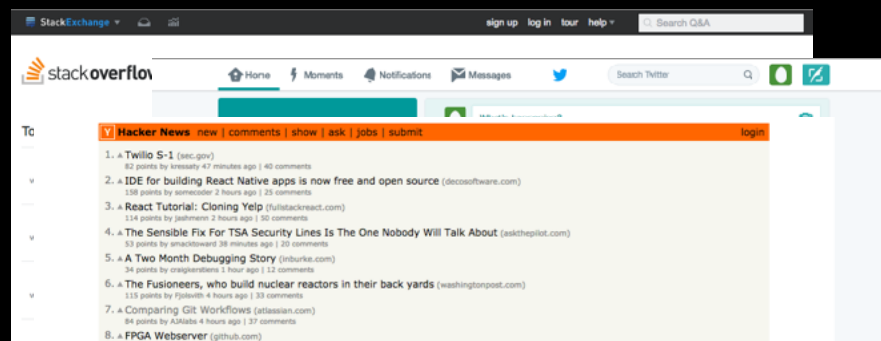
- ▶ concept reuse + simple assembly
- ▶ reduce the amount of knowledge & effort required to build web apps
- ▶ allow end-users to build more sophisticated apps themselves



# The Déjà Vu Platform



```
<dv.component name="submit-post">
  <sn.navbar /> ...
  <dv.tx>
    <dv.gen-id /> ...
    <post.create-object id=dv.gen-id.id
      initialValue={ author: sn.navbar.user.username }
      showExclude=["author"] buttonLabel="Submit"
      newObjectSavedText="Post submitted" />
    <scoreposts.create-score targetId=dv.gen-id.id
      sourceId=sn.navbar.user.username
      value=0 hidden=true /> ...
  </dv.tx> ...
</dv.component>
```



## *catalog of concepts*

- ▶ new modularity: concepts
- ▶ capture and reuse concept design & implementation

## *concept composition mechanism*

- ▶ in HTML, no JS or server-side code
- ▶ join concept actions for aggregate behavior

## *web apps*

- ▶ complex behavior
- ▶ rich graphical user interfaces

# Key Ideas

# Concepts as Modules

Concept: *Scoring*

Purpose: Score items

Score \*

15

Create

create-score

...

ben

2bca5702-fcc2-44be-87c1-eb85804430fd

15

alyssa

ben

Total: 15

show-target

# Concepts as Modules

Concept: *Scoring*  
Purpose: Score items

self-contained,  
reusable increment  
of functionality

Score \*

15

Create

...

ben

2bca5702-fcc2-44be-87c1-eb85804430fd

15

alyssa

ben

Total: 15

create-score

show-target

# Concepts as Modules

Concept: *Scoring*  
Purpose: Score items

motivated by  
end-user  
purpose

self-contained,  
reusable increment  
of functionality

Score \*

15

Create

create-score

...

ben

2bca5702-fcc2-44be-87c1-eb85804430fd

15

alyssa

ben

Total: 15

show-target

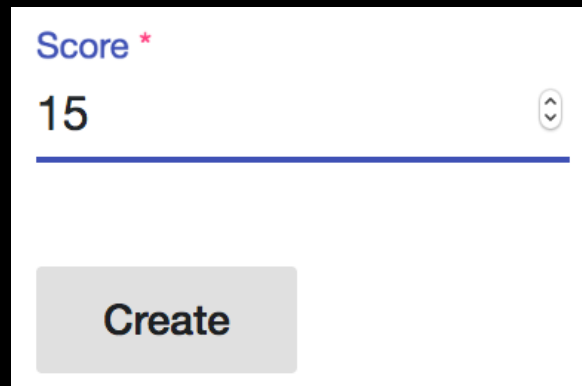


# Concepts as Modules

Concept: *Scoring*  
Purpose: Score items

motivated by  
end-user  
purpose

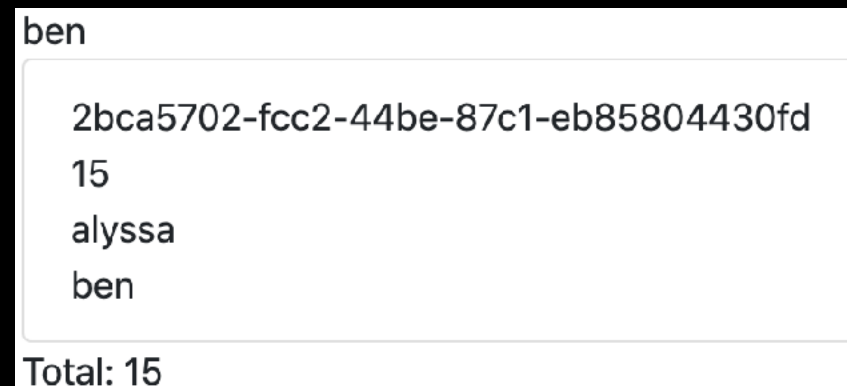
self-contained,  
reusable increment  
of functionality



A UI form for creating a score. It features a label 'Score \*' in blue, a text input field containing the number '15', and a 'Create' button below the input field.

create-score

...



A UI for showing a target. It has a title 'ben' and a list of items: a long alphanumeric string '2bca5702-fcc2-44be-87c1-eb85804430fd', the number '15', the name 'alyssa', and the name 'ben'. At the bottom, it says 'Total: 15'.

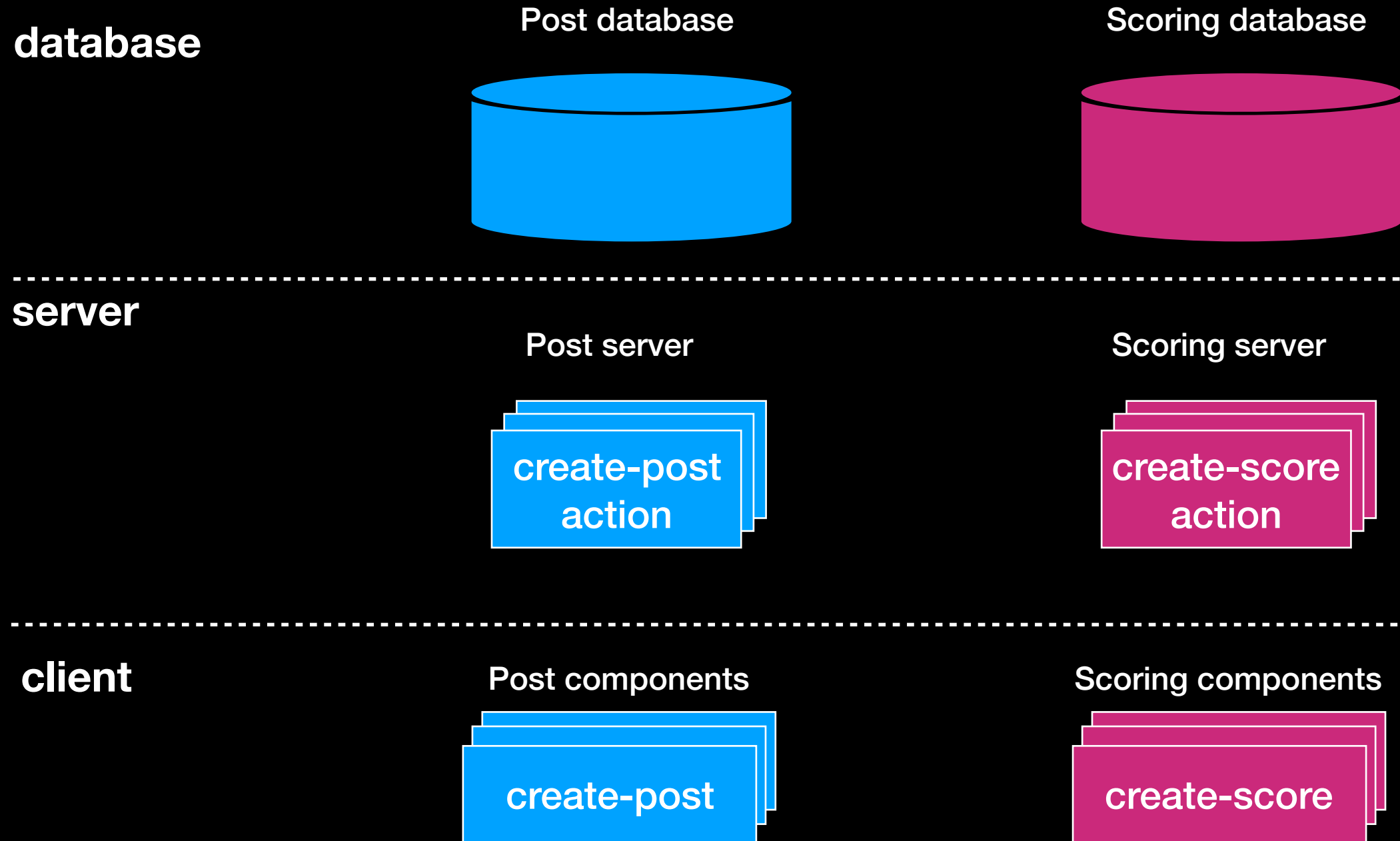
show-target

exports  
components

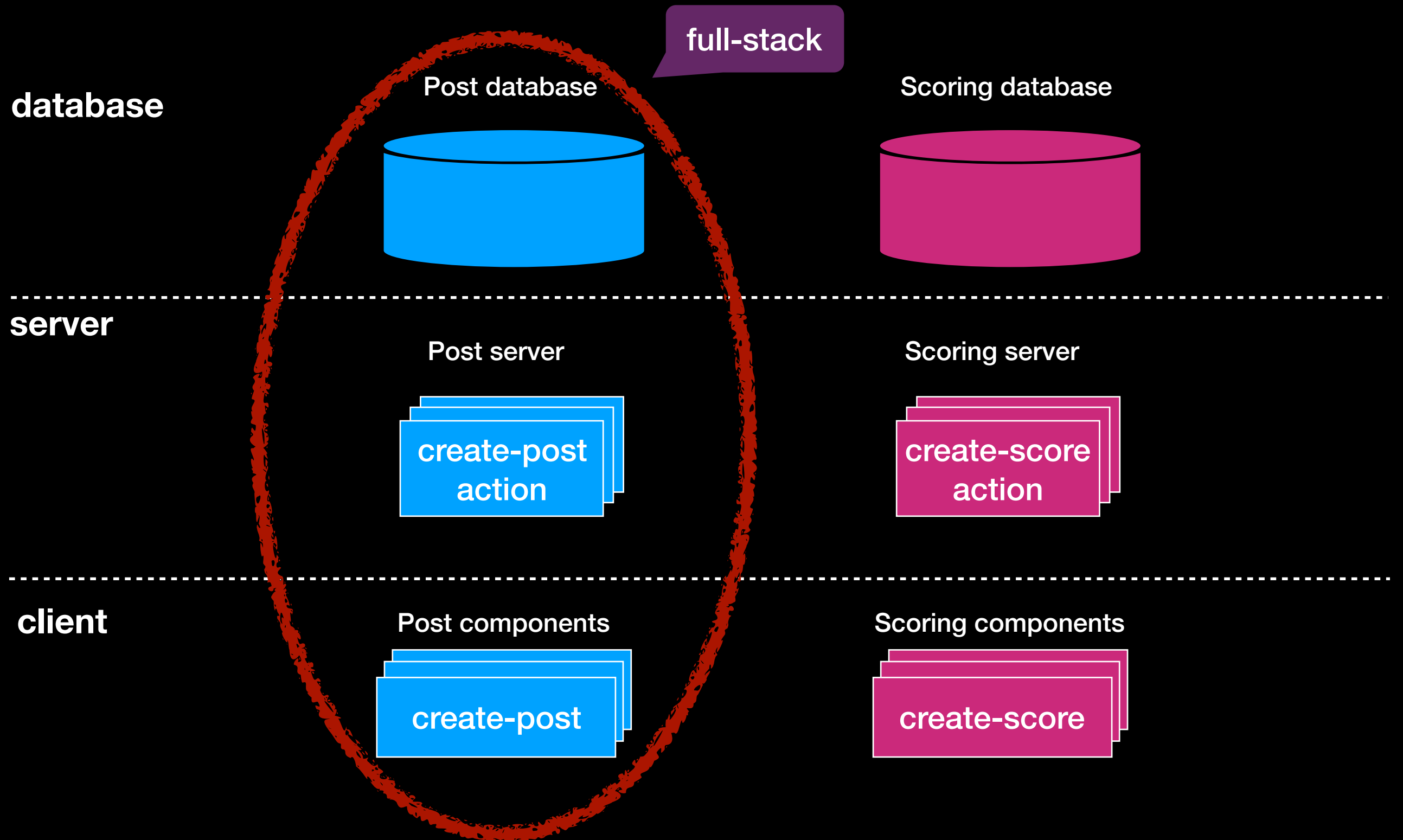
Component:

- GUI element
- reactive I/O properties
- server-side action

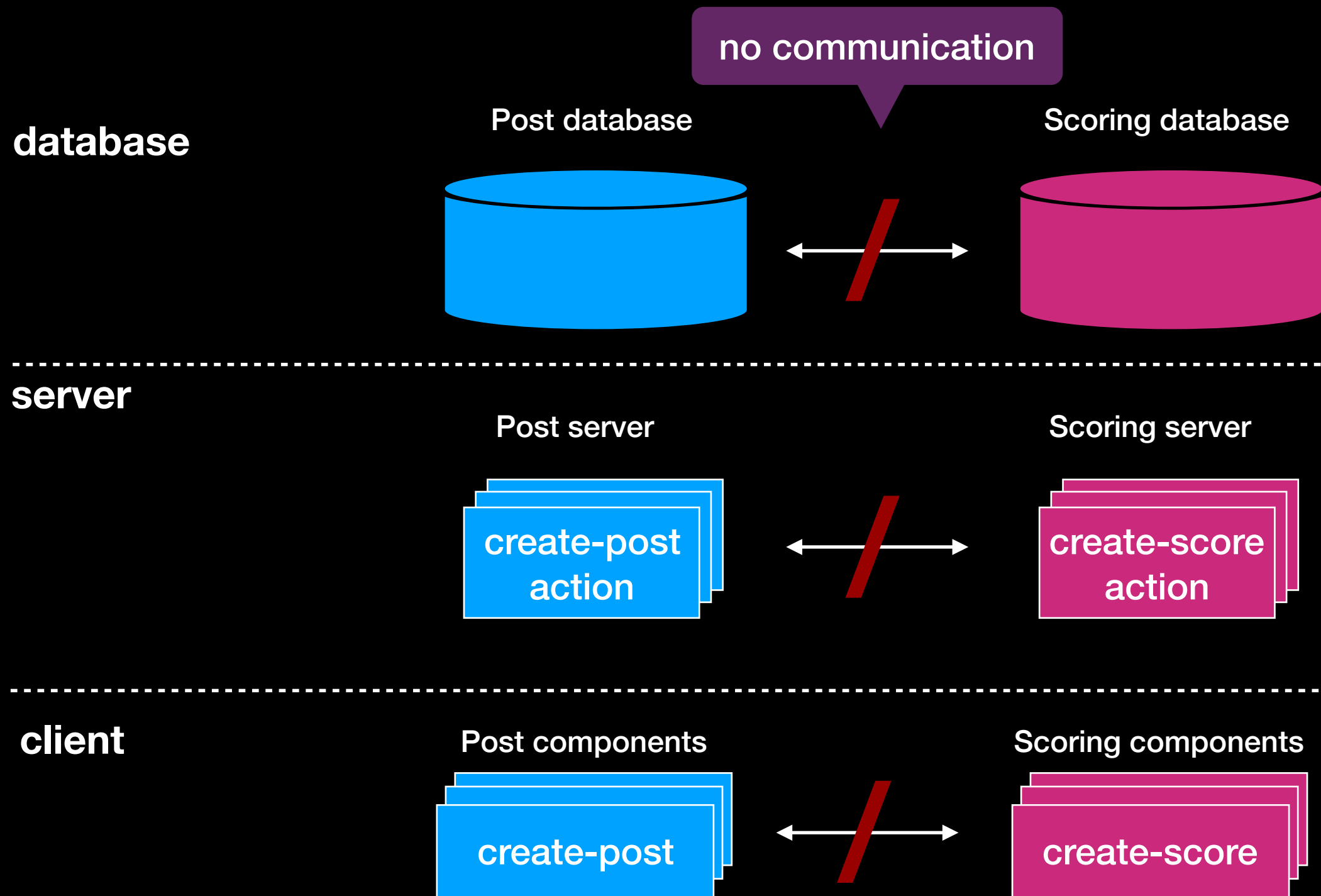
# Full-stack Modules



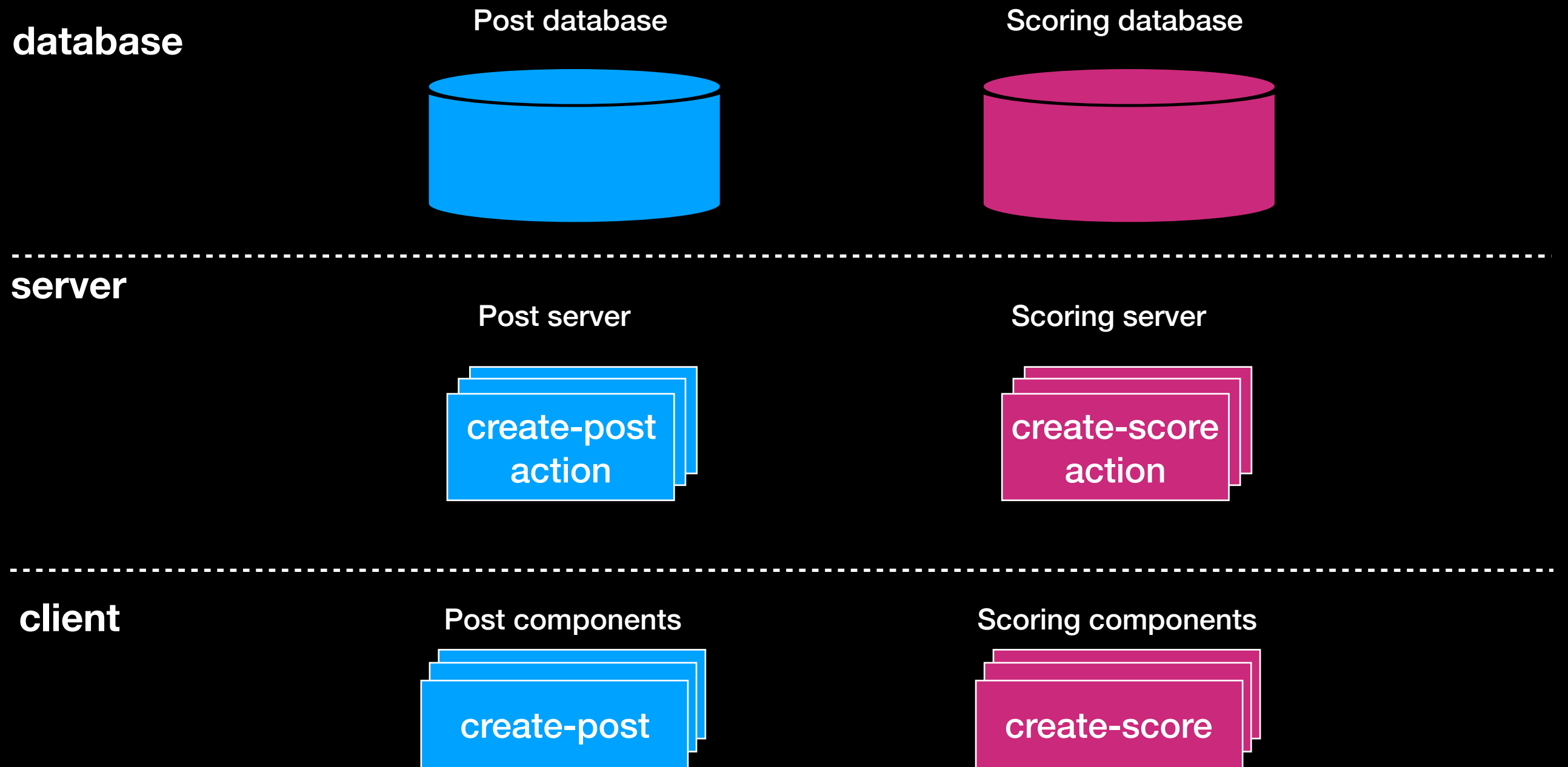
# Full-stack Modules



# Full-stack Modules

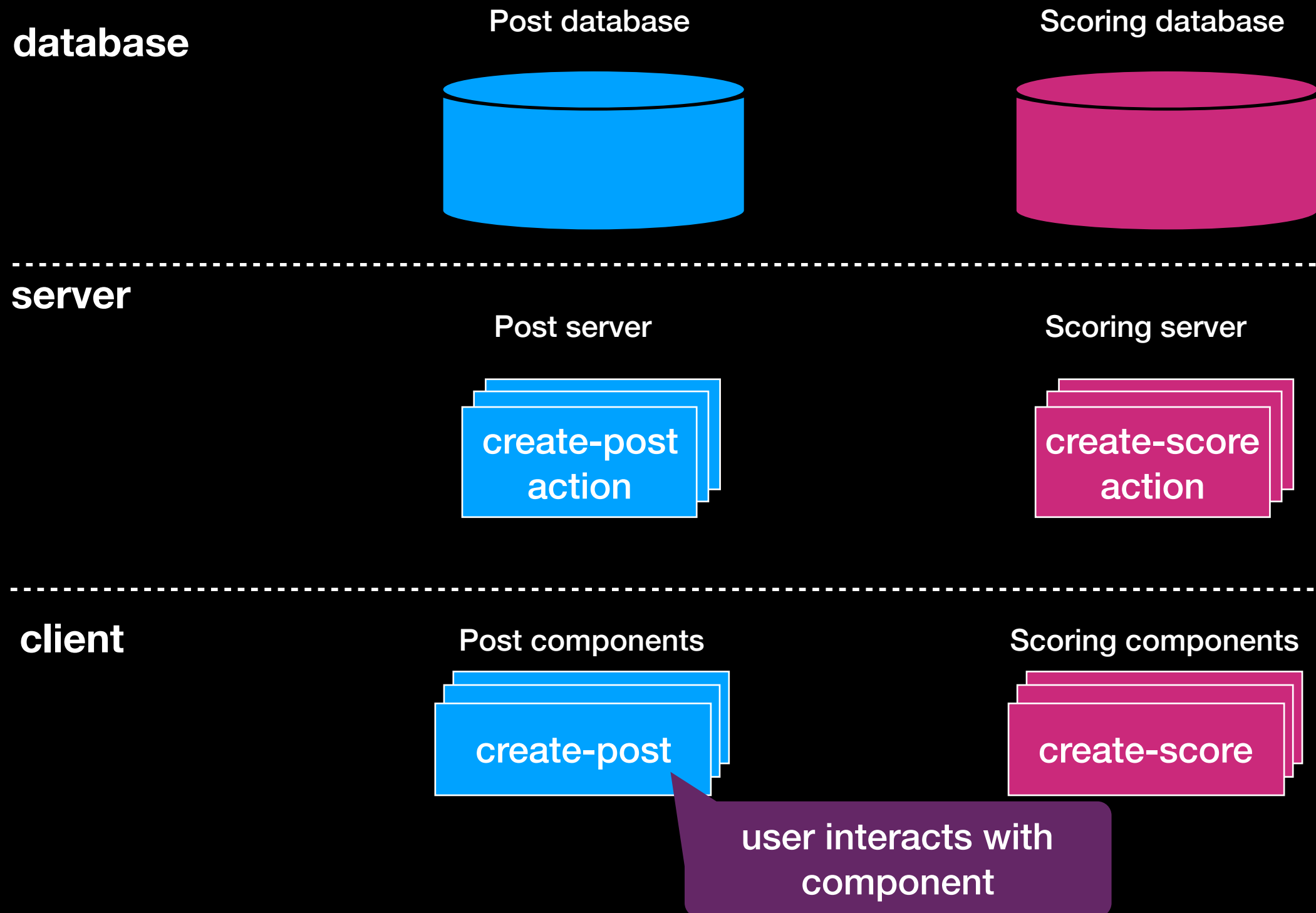


# Full-stack Modules

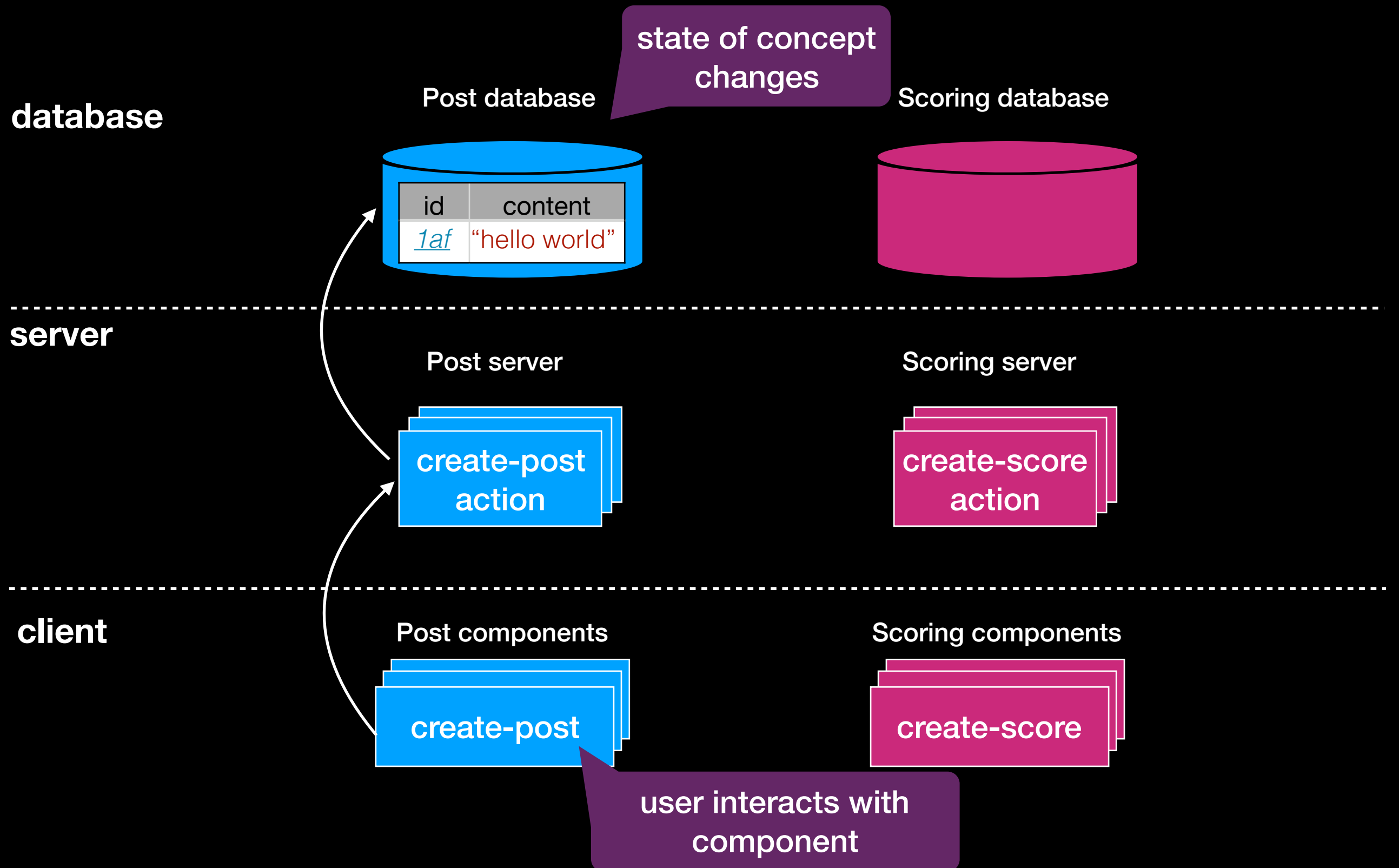




# Full-stack Modules



# Full-stack Modules



# Create Posts with a Score?

# Create Posts with a Score?

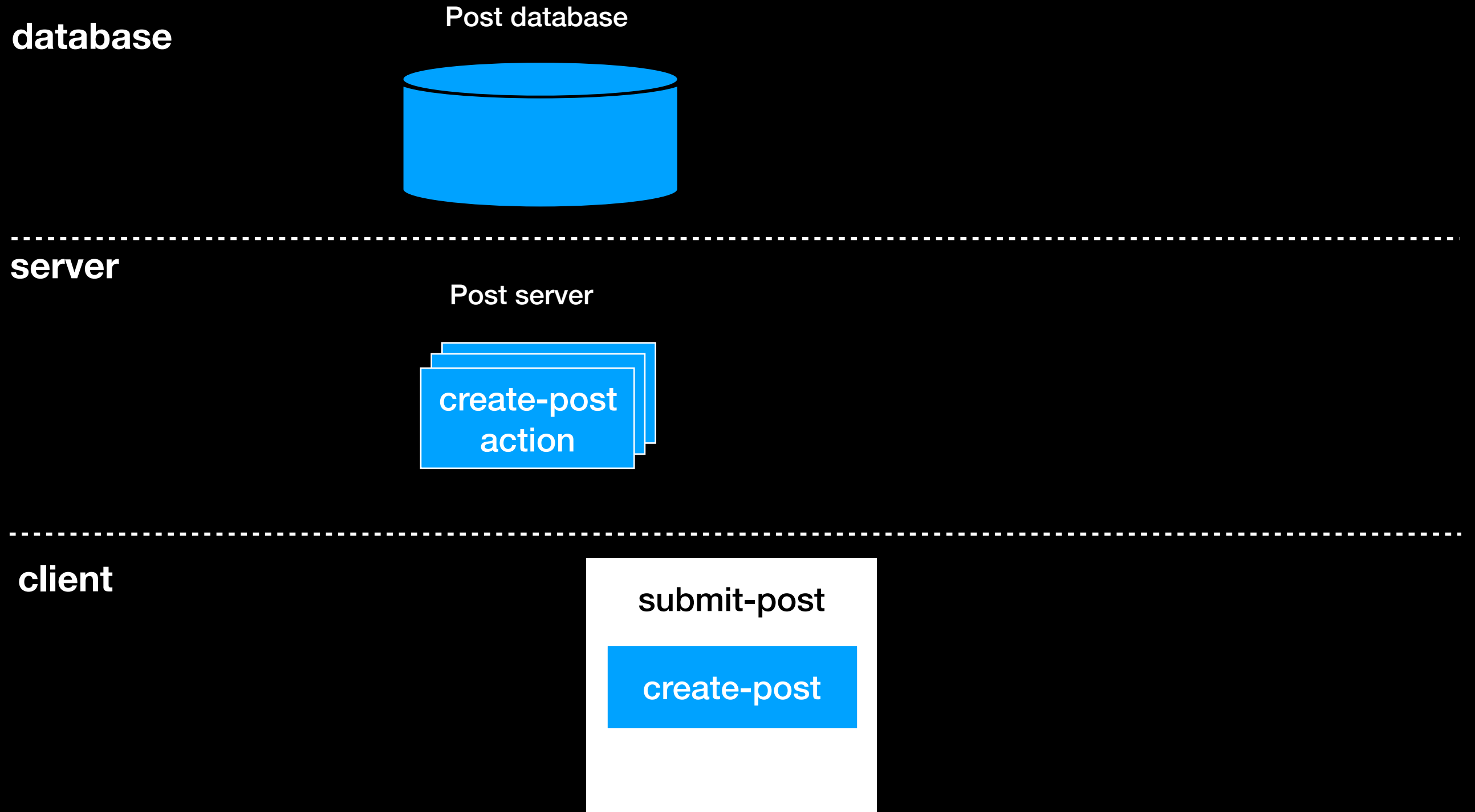
**database**

-----  
**server**

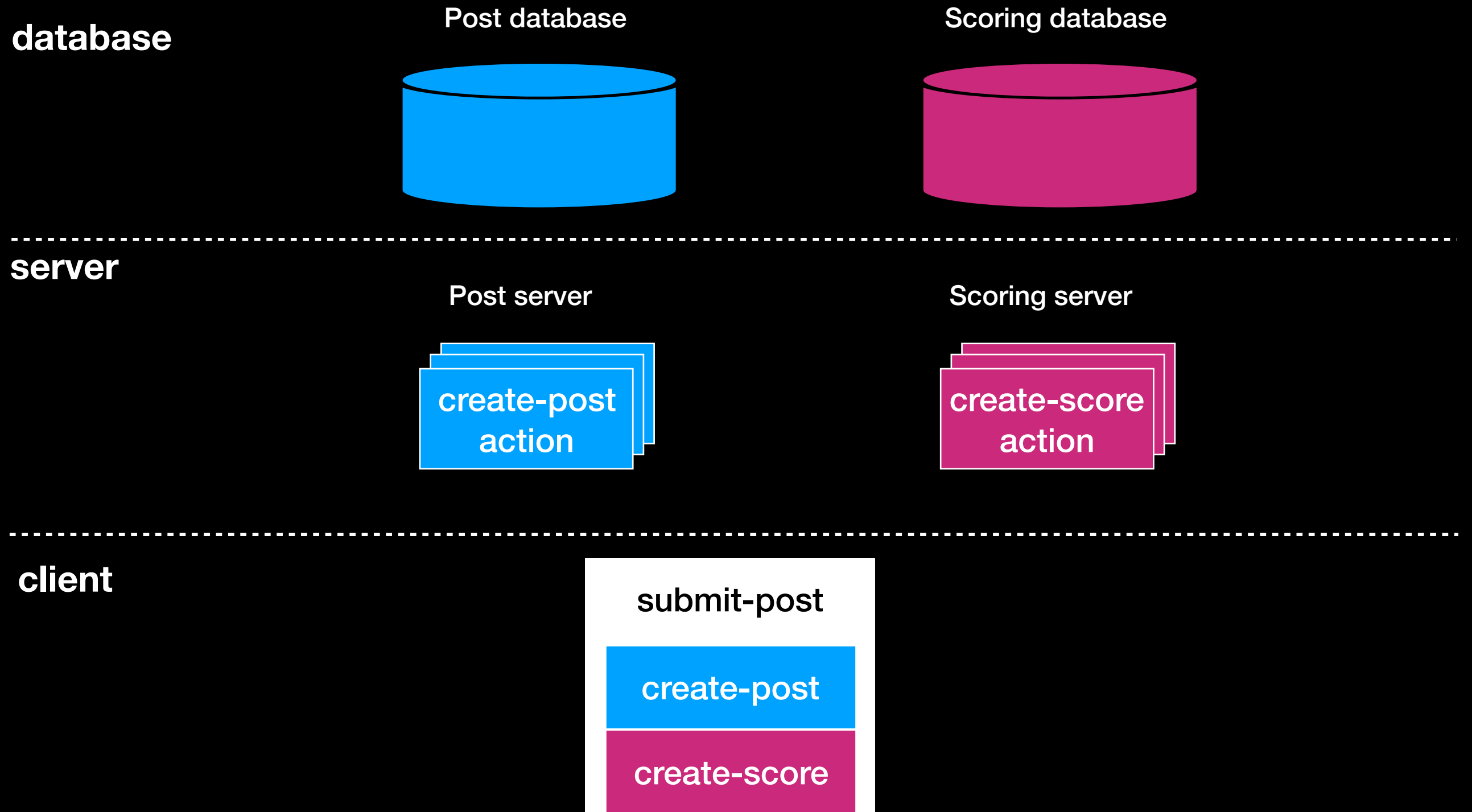
-----  
**client**

submit-post

# Create Posts with a Score?

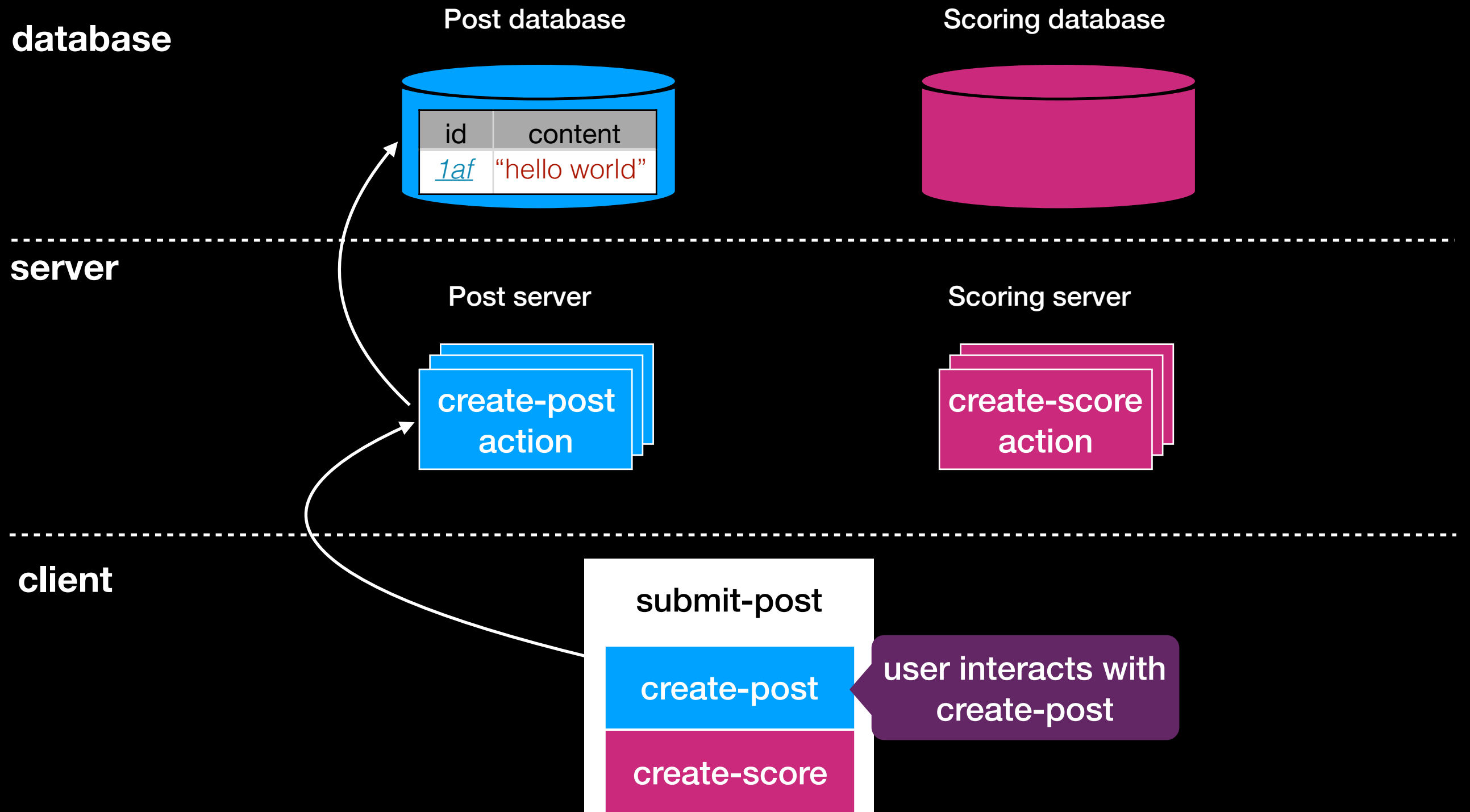


# Create Posts with a Score?

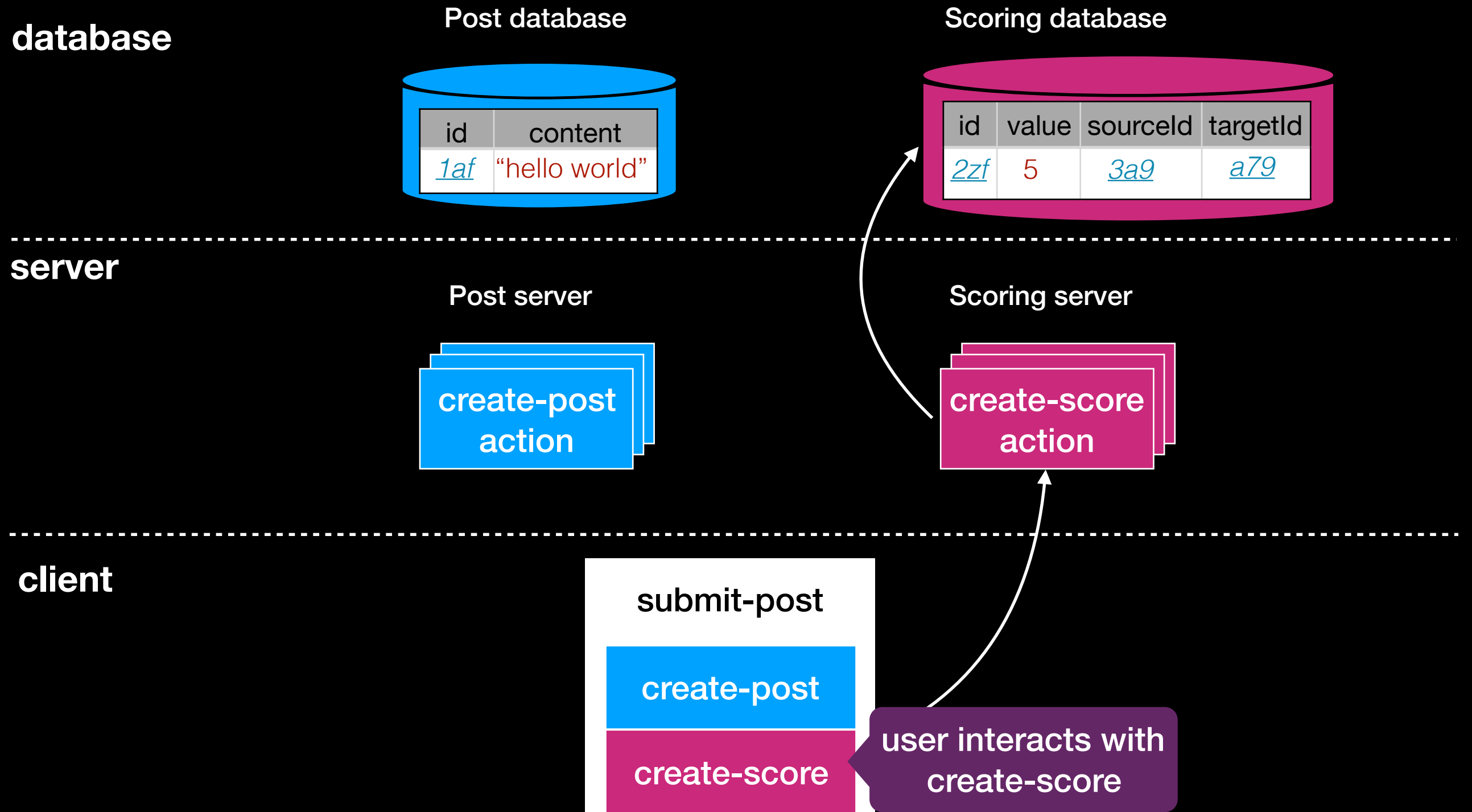




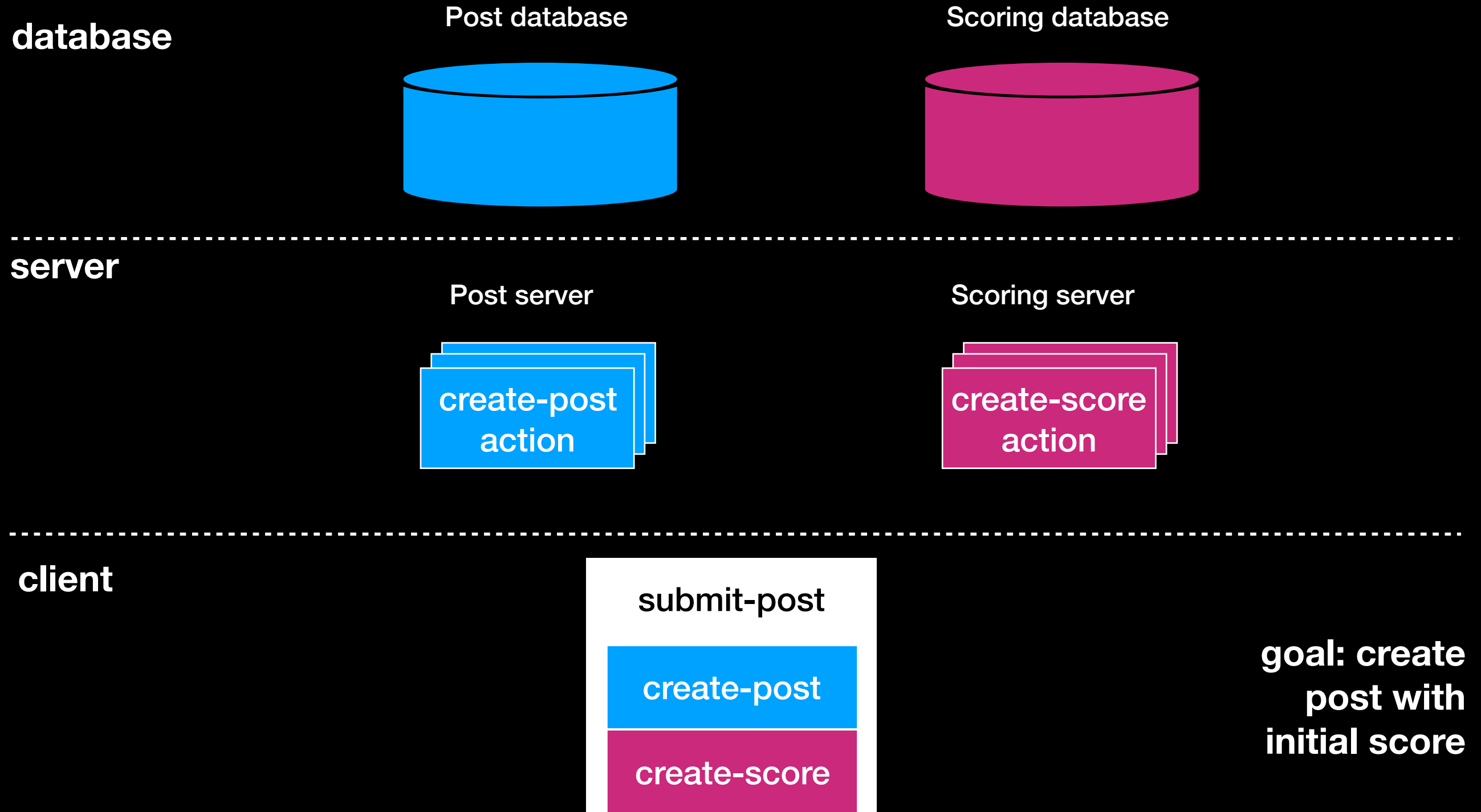
# Create Posts with a Score?



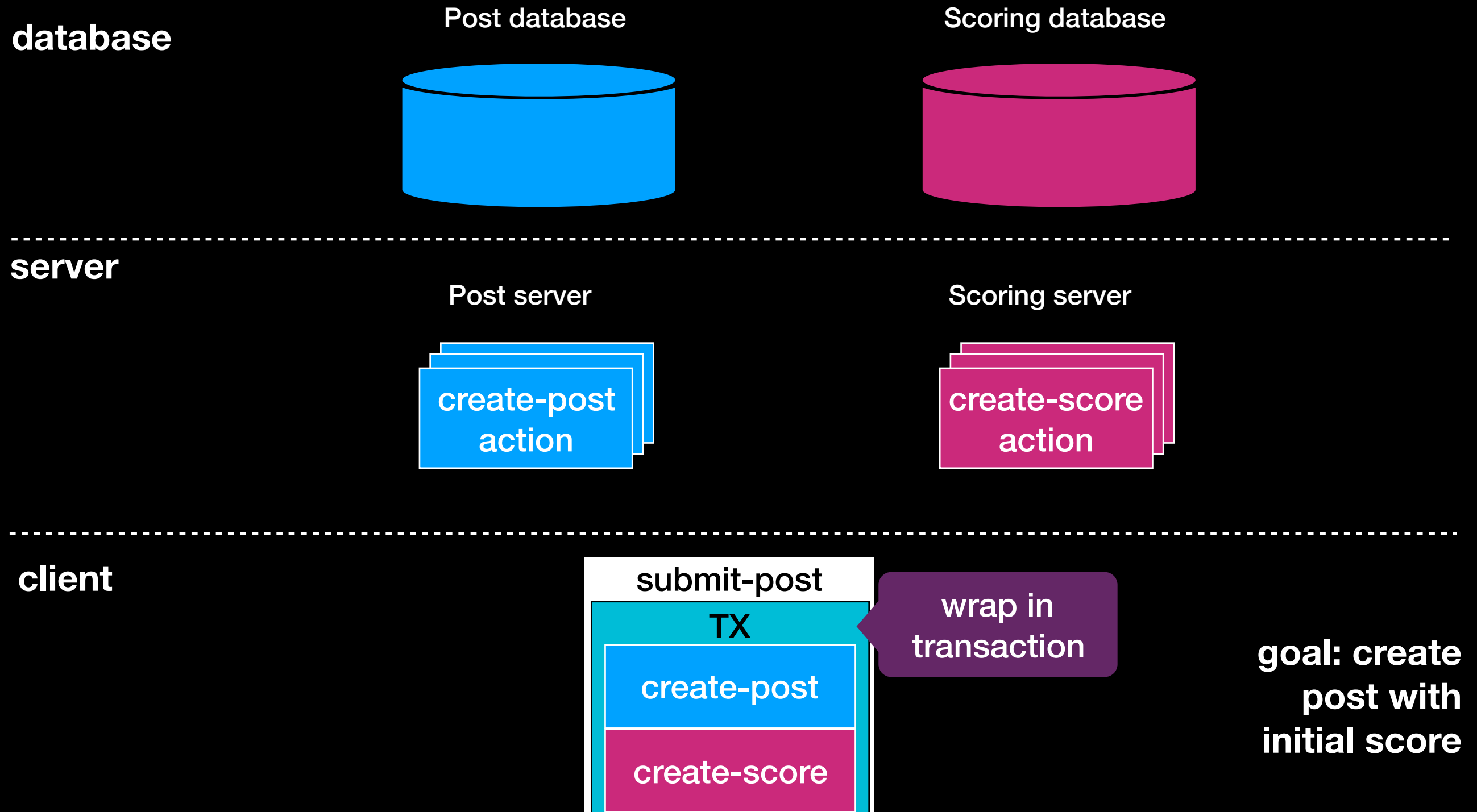
# Create Posts with a Score?



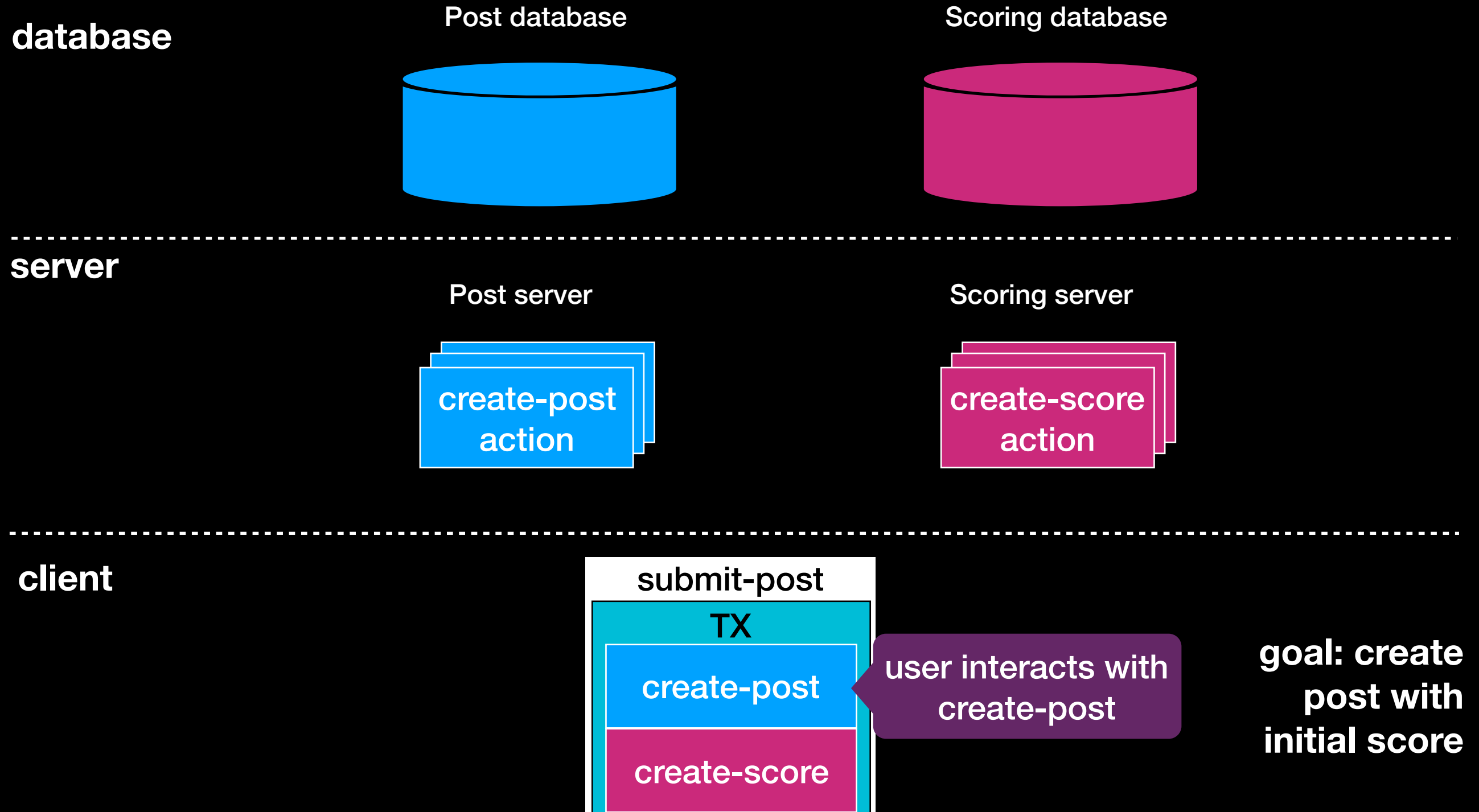
# Declarative Sync



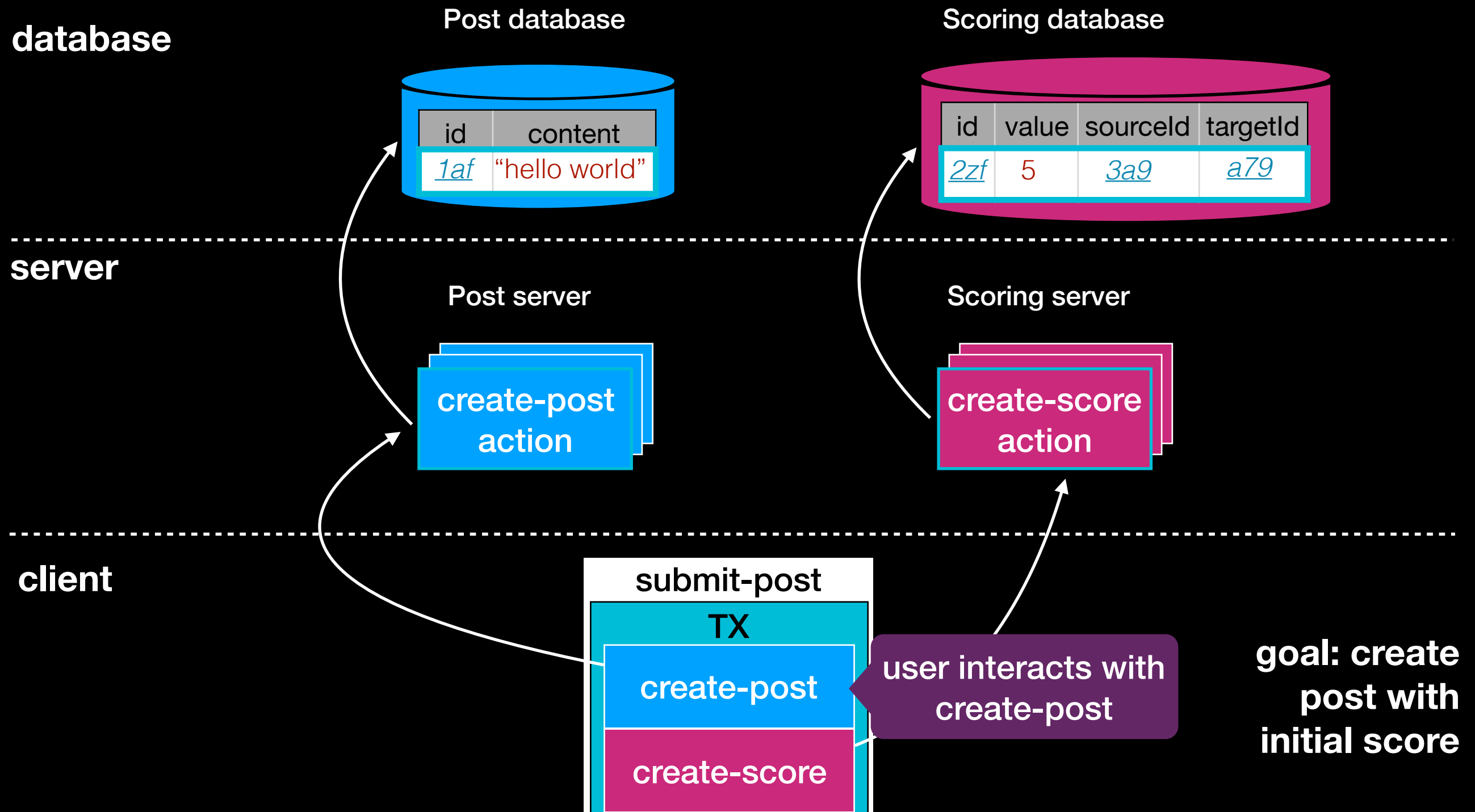
# Declarative Sync



# Declarative Sync

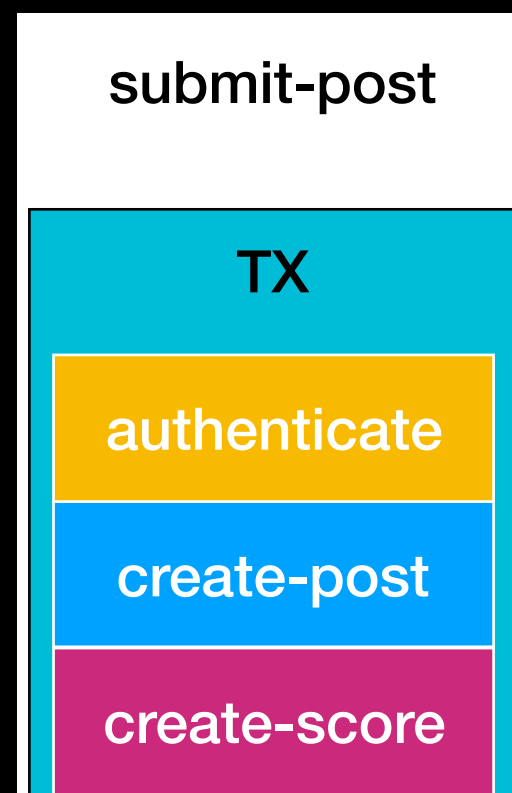
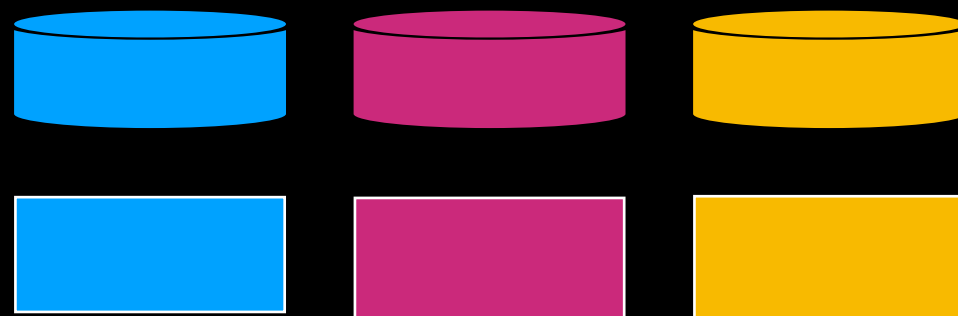


# Declarative Sync

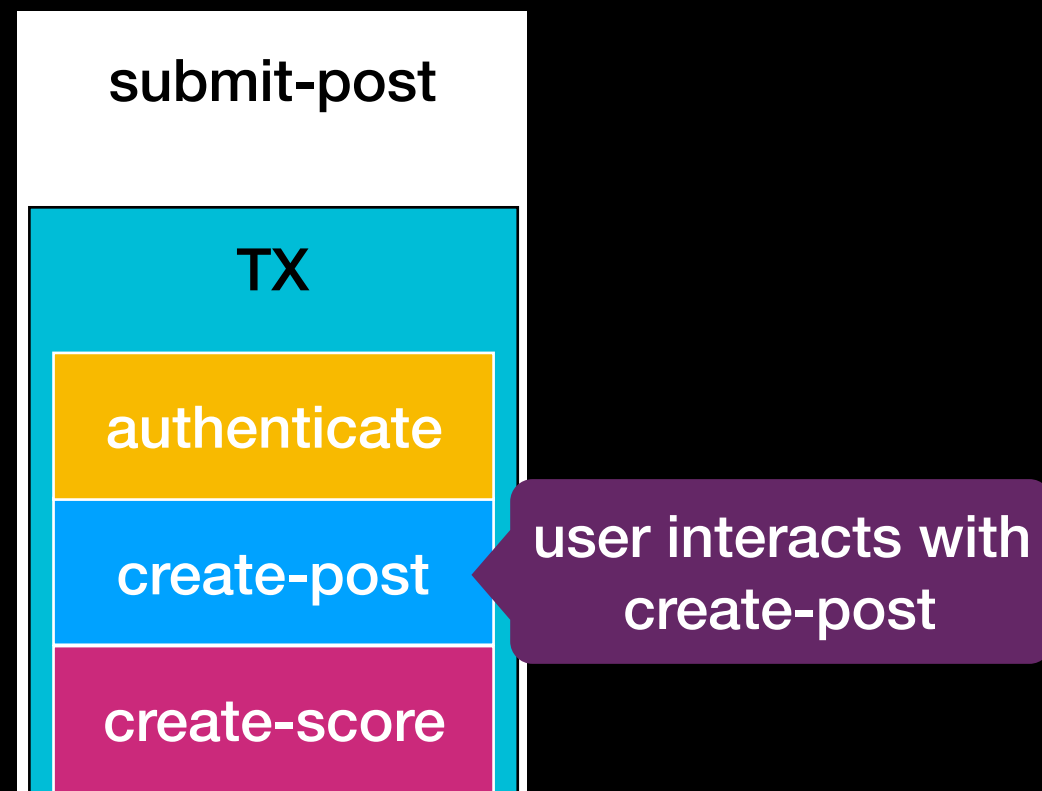
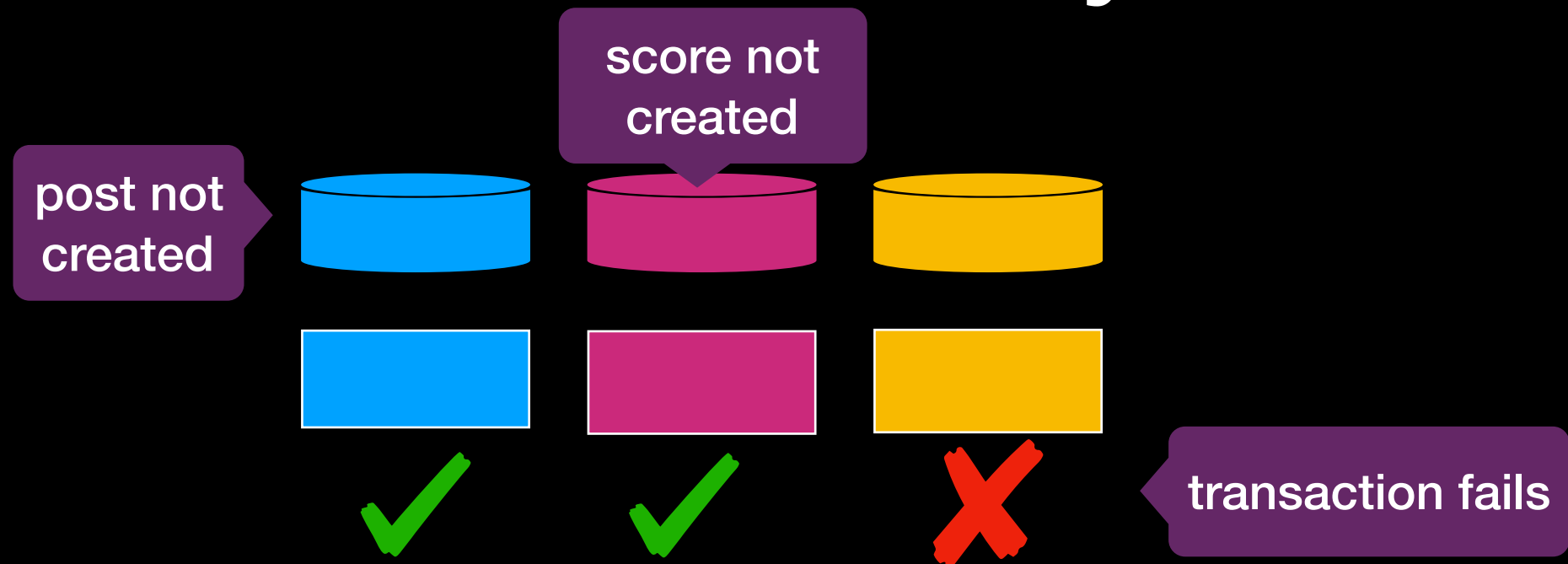




# Declarative Sync



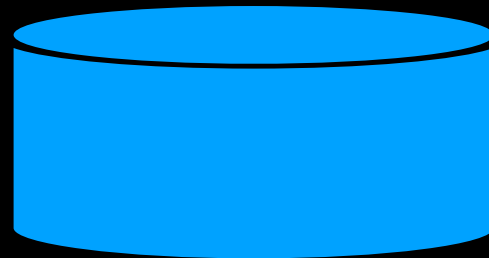
# Declarative Sync



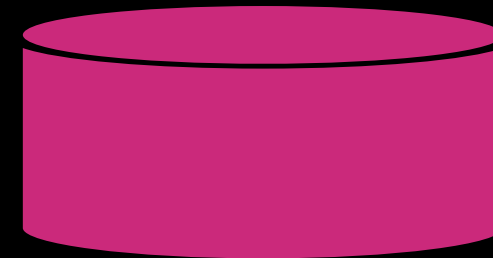
# ID Sharing

**database**

Post database



Scoring database



**server**

Post server

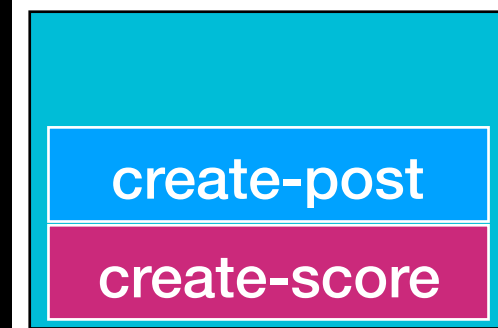


Scoring server



**client**

submit-post

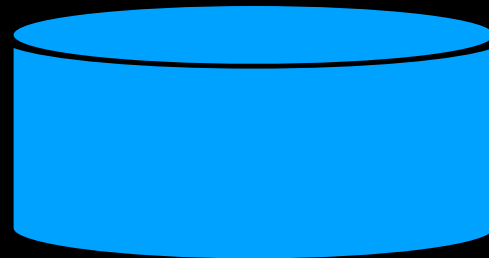


**goal: create  
post with  
initial score**

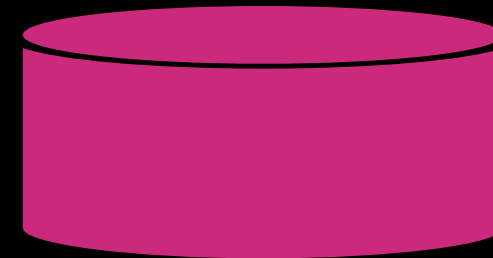
# ID Sharing

database

Post database



Scoring database



server

Post server

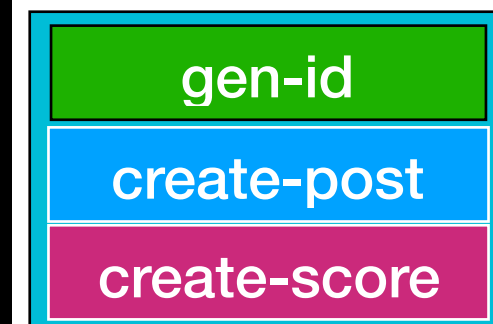


Scoring server



client

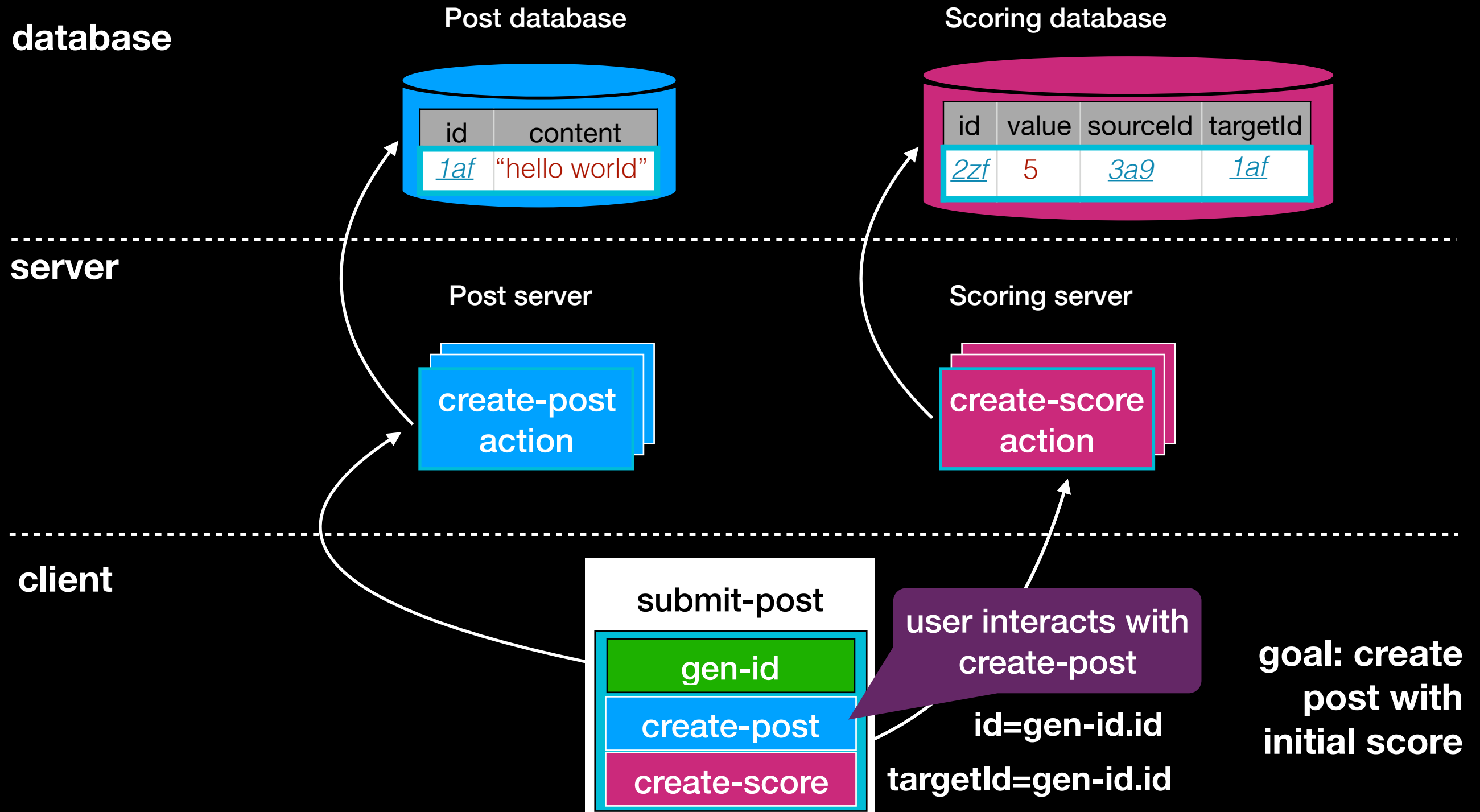
submit-post



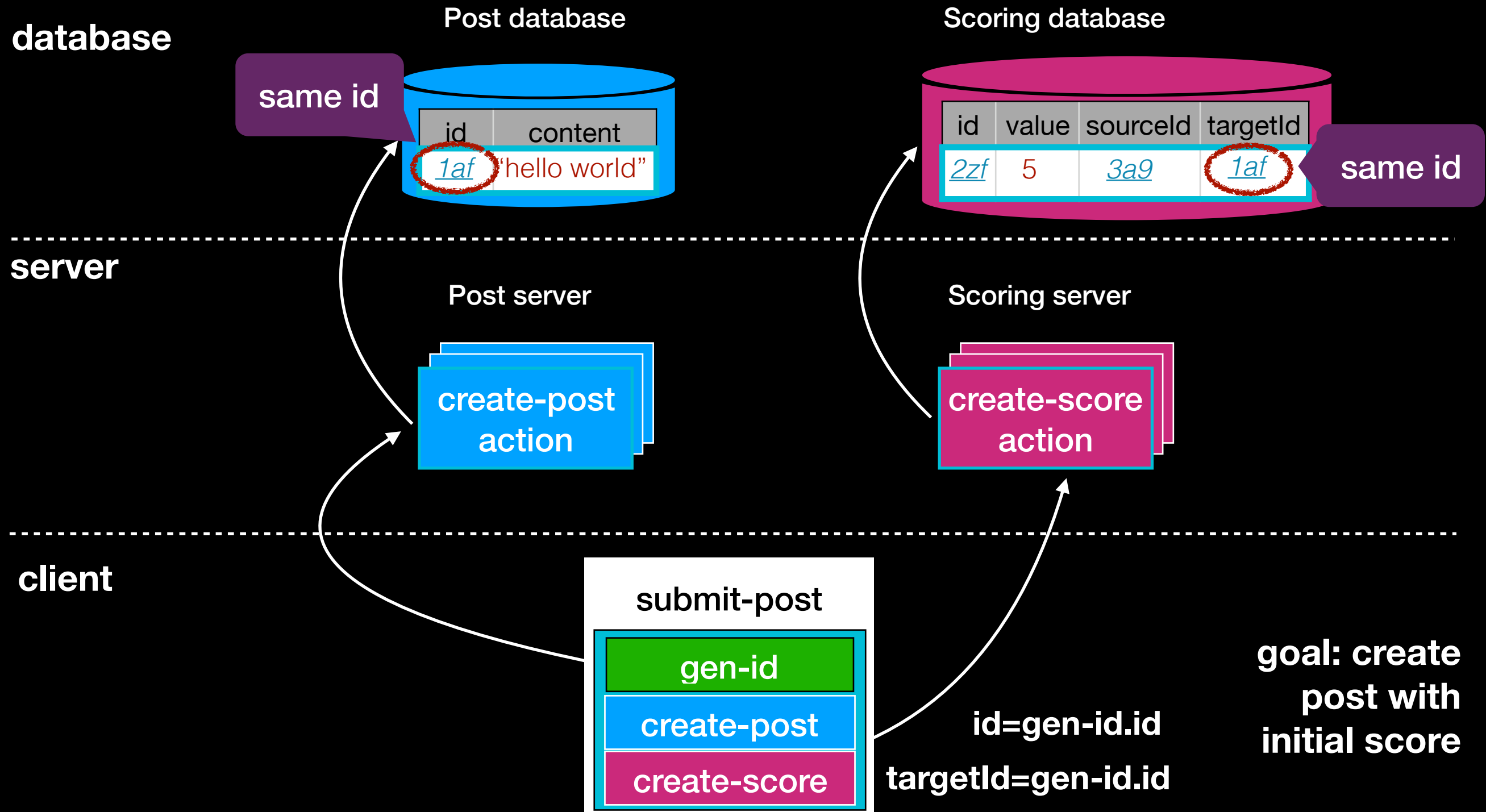
id=gen-id.id  
targetId=gen-id.id

goal: create  
post with  
initial score

# ID Sharing



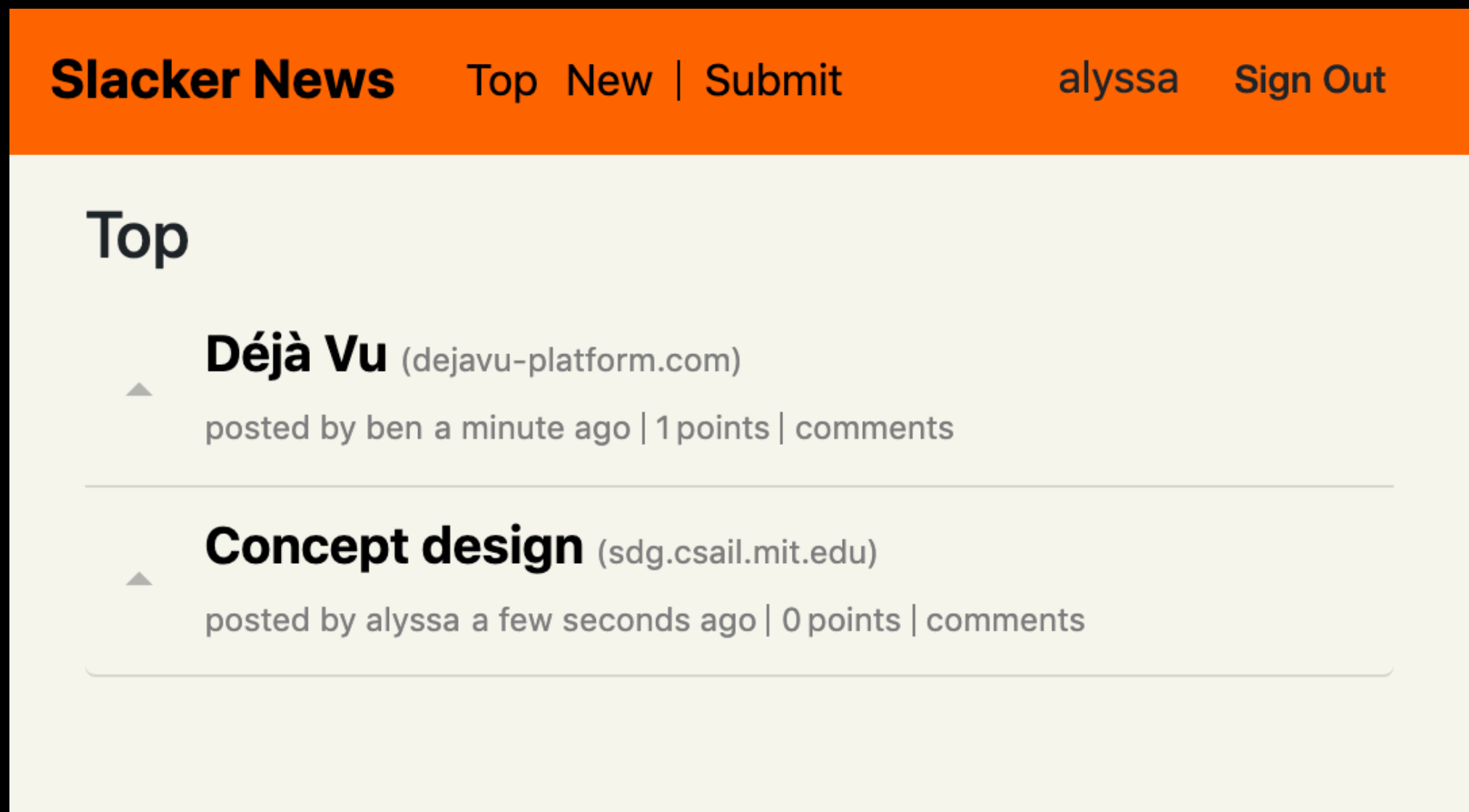
# ID Sharing



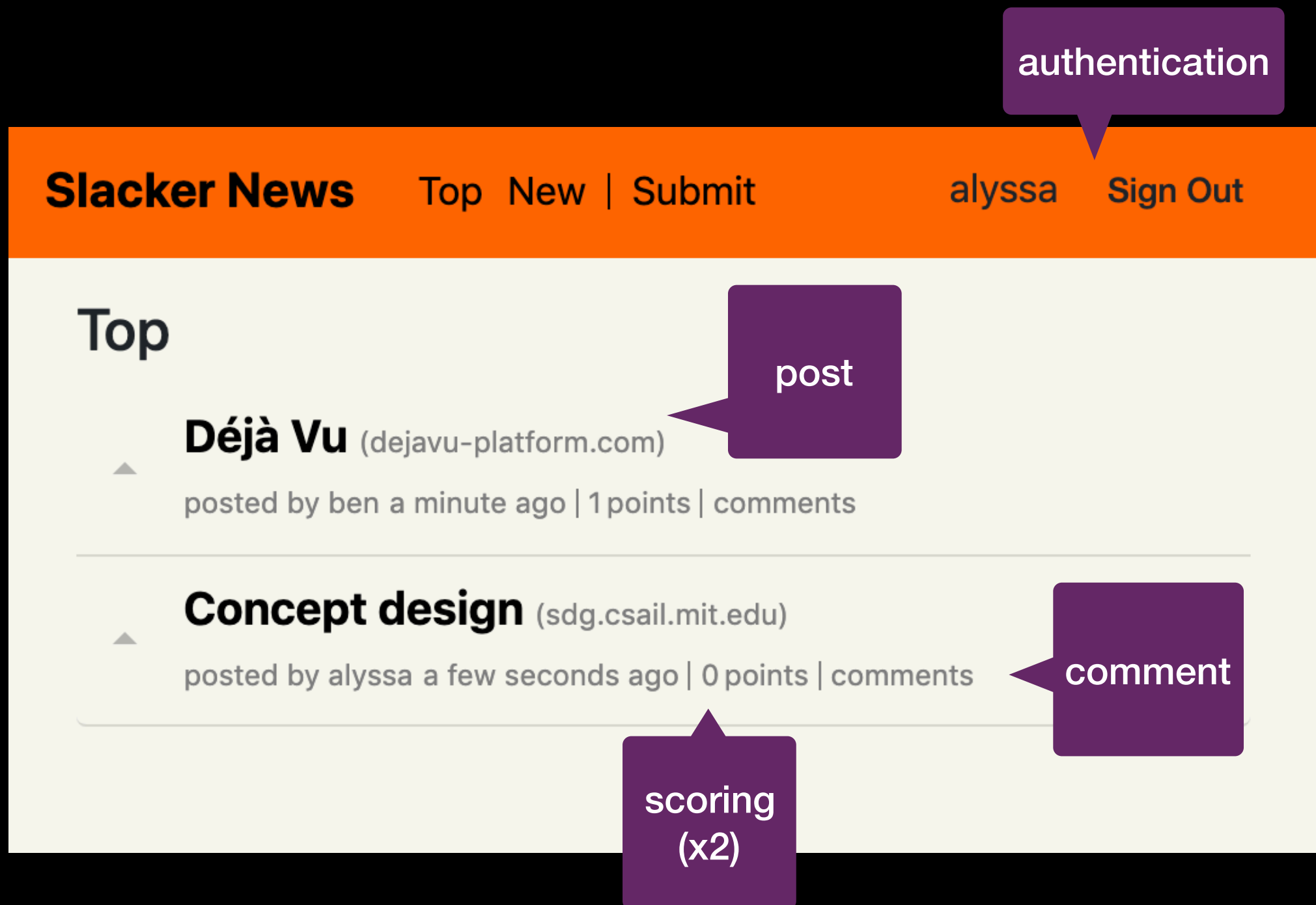
# Building Apps with Déjà Vu



# A Sample Déjà Vu App



# A Sample Déjà Vu App



# Top

No posts yet

Top



**Déjà Vu** (deja-vu-platform.com)

posted by ben a few seconds ago | 0 points | comments

## Top

 **Déjà Vu** (deja-vu-platform.com)

posted by ben 3 minutes ago | 1 points | [comments](#)

---

# Catalog

Concept	Purpose
Authentication	Verify a user's identity with a username and password
Authorization	Control access to resources
Chat	Exchange messages in real time
Comment	Share reactions to items
Event	Schedule events
Follow	Receive updates from sources
Geolocation	Locate points of interest
Group	Organize members into groups so that they can be handled in aggregate
Label	Label items so that they can be found later
Match	Connect users after they both agree
Passkey	Verify a user's identity with a code
Property	Describe an object with properties that have values
Ranking	Rank items
Rating	Crowdsourcing evaluation of items
Schedule	Find a time to meet
Scoring	Keep track of scores
Task	Keep track of pieces of work to be done
Transfer	Transfer money or items between accounts

# Catalog

Concept	Purpose
Authentication	Verify a user's identity with a username and password
Authorization	Control access to resources
Chat	Exchange messages in real time
Comment	Share reactions to items
Event	Schedule events
Follow	Receive updates from sources
Geolocation	Locate points of interest
Group	Organize members into groups so that they can be handled in aggregate
Label	Label items so that they can be found later
Match	Connect users after they both agree
Passkey	Verify a user's identity with a code
Property	Describe an object with properties that have values
Ranking	Rank items
Rating	Crowdsourcing evaluation of items
Schedule	Find a time to meet
Scoring	Keep track of scores
Task	Keep track of pieces of work to be done
Transfer	Transfer money or items between accounts



# Catalog

Concept	Purpose
Authentication	Verify a user's identity with a username and password
Authorization	Control access to resources
Chat	Exchange messages in real time
Comment	Share reactions to items
Event	Schedule events
Follow	Receive updates from sources
Geolocation	Locate points of interest
Group	Organize members into groups so that they can be handled in aggregate
Label	Label items so that they can be found later
Match	Connect users after they both agree
Passkey	Verify a user's identity with a code
Property	Describe an object with properties that have values
Ranking	Rank items
Rating	Crowdsourcing evaluation of items
Schedule	Find a time to meet
Scoring	Keep track of scores
Task	Keep track of pieces of work to be done
Transfer	Transfer money or items between accounts

for  
posts

```

{
  "name": "sn",
  "usedConcepts": {
    "authentication": {},
    "comment": {},
    "post": {
      "name": "Property",
      "config": {
        "schema": {
          "title": "Post", "type": "object",
          "properties": {
            "author": { "type": "string" },
            "title": { "type": "string" },
            "url": { "type": "string", "format": "url" }
          },
          "required": [ "author", "title", "url" ]
        }
      }
    },
    "scoreposts": { "name": "Scoring" },
    "scorecomments": { "name": "Scoring" }
  },
  "routes": [
    { "path": "", "component": "home" },
    { "path": "/login", "component": "login" },
    { "path": "/post", "component": "show-post-details" },
    ...
  ]
}

```

dvconfig.json

used  
concepts

```
{
  "name": "sn",
  "usedConcepts": {
    "authentication": {},
    "comment": {},
    "post": {
      "name": "Property",
      "config": {
        "schema": {
          "title": "Post", "type": "object",
          "properties": {
            "author": { "type": "string" },
            "title": { "type": "string" },
            "url": { "type": "string", "format": "url" }
          },
          "required": [ "author", "title", "url" ]
        }
      }
    },
    "scoreposts": { "name": "Scoring" },
    "scorecomments": { "name": "Scoring" }
  },
  "routes": [
    { "path": "", "component": "home" },
    { "path": "/login", "component": "login" },
    { "path": "/post", "component": "show-post-details" },
    ...
  ]
}
```

dvconfig.json

used  
concepts

concept

instance  
config

alias

```
{
  "name": "sn",
  "usedConcepts": {
    "authentication": {},
    "comment": {},
    "post": {
      "name": "Property",
      "config": {
        "schema": {
          "title": "Post", "type": "object",
          "properties": {
            "author": { "type": "string" },
            "title": { "type": "string" },
            "url": { "type": "string", "format": "url" }
          },
          "required": [ "author", "title", "url" ]
        }
      }
    },
    "scoreposts": { "name": "Scoring" },
    "scorecomments": { "name": "Scoring" }
  },
  "routes": [
    { "path": "", "component": "home" },
    { "path": "/login", "component": "login" },
    { "path": "/post", "component": "show-post-details" },
    ...
  ]
}
```

dvconfig.json

used  
concepts

concept

instance  
config

alias

schema  
definition

```
{
  "name": "sn",
  "usedConcepts": {
    "authentication": {},
    "comment": {},
    "post": {
      "name": "Property",
      "config": {
        "schema": {
          "title": "Post", "type": "object",
          "properties": {
            "author": { "type": "string" },
            "title": { "type": "string" },
            "url": { "type": "string", "format": "url" }
          },
          "required": [ "author", "title", "url" ]
        }
      }
    },
    "scoreposts": { "name": "Scoring" },
    "scorecomments": { "name": "Scoring" }
  },
  "routes": [
    { "path": "", "component": "home" },
    { "path": "/login", "component": "login" },
    { "path": "/post", "component": "show-post-details" },
    ...
  ]
}
```

dvconfig.json

used  
concepts

concept

instance  
config

alias

schema  
definition

routes

```
{
  "name": "sn",
  "usedConcepts": {
    "authentication": {},
    "comment": {},
    "post": {
      "name": "Property",
      "config": {
        "schema": {
          "title": "Post", "type": "object",
          "properties": {
            "author": { "type": "string" },
            "title": { "type": "string" },
            "url": { "type": "string", "format": "url" }
          },
          "required": [ "author", "title", "url" ]
        }
      }
    },
    "scoreposts": { "name": "Scoring" },
    "scorecomments": { "name": "Scoring" }
  },
  "routes": [
    { "path": "", "component": "home" },
    { "path": "/login", "component": "login" },
    { "path": "/post", "component": "show-post-details" },
    ...
  ]
}
```

dvconfig.json

```

<dv.component name="submit-post">
  <sn.navbar /> ...
  <dv.tx>
    <dv.gen-id /> ...
    <post.create-object id=dv.gen-id.id
      initialValue={ author: sn.navbar.user.username }
      showExclude=["author"] buttonLabel="Submit"
      newObjectSavedText="Post submitted" />
    <scoreposts.create-score targetId=dv.gen-id.id
      sourceId=sn.navbar.user.username
      value=0 hidden=true /> ...
  </dv.tx> ...
</dv.component>

```

submit-post.html

Slacker News
Top New | Submit
ben
Sign Out

sn.navbar

Submit

Title \*

Url \*

Submit

post.create-object

app name

component  
name

instance name

```
<dv.component name="submit-post">
  <sn.navbar /> ...
  <dv.tx>
    <dv.gen-id /> ...
    <post.create-object id=dv.gen-id.id
      initialValue={ author: sn.navbar.user.username }
      showExclude=["author"] buttonLabel="Submit"
      newObjectSavedText="Post submitted" />
    <scoreposts.create-score targetId=dv.gen-id.id
      sourceId=sn.navbar.user.username
      value=0 hidden=true /> ...
  </dv.tx> ...
</dv.component>
```

submit-post.html

Slacker News Top New | Submit ben Sign Out

*sn.navbar*

Submit

Title \*

Url \*

Submit

*post.create-object*



```

<dv.component name="submit-post">
  <sn.navbar /> ...
  <dv.tx>
    <dv.gen-id /> ...
    <post.create-object id=dv.gen-id.id
      initialValue={ author: sn.navbar.user.username }
      showExclude=["author"] buttonLabel="Submit"
      newObjectSavedText="Post submitted" />
    <scoreposts.create-score targetId=dv.gen-id.id
      sourceId=sn.navbar.user.username
      value=0 hidden=true /> ...
  </dv.tx> ...
</dv.component>

```

submit-post.html

**Slacker News**
Top New | Submit
ben Sign Out

*sn.navbar*

Submit

Title \*

Url \*

Submit

*post.create-object*

```

<dv.component name="submit-post">
  <sn.navbar /> ...
  <dv.tx>
    <dv.gen-id /> ...
    <post.create-object id=dv.gen-id.id
      initialValue={ author: sn.navbar.user.username }
      showExclude=["author"] buttonLabel="Submit"
      newObjectSavedText="Post submitted" />
    <scoreposts.create-score targetId=dv.gen-id.id
      sourceId=sn.navbar.user.username
      value=0 hidden=true /> ...
  </dv.tx> ...
</dv.component>

```

hide

submit-post.html

**Slacker News**
Top New | Submit
 ben Sign Out

*sn.navbar*

# Submit

Title \*

Url \*

Submit

*post.create-object*

```

<dv.component name="submit-post">
  <sn.navbar /> ...
  <dv.tx>
    <dv.gen-id /> ...
    <post.create-object id=dv.gen-id.id
      initialValue={ author: sn.navbar.user.username }
      showExclude=["author"] buttonLabel="Submit"
      newObjectSavedText="Post submitted" />
    <scoreposts.create-score targetId=dv.gen-id.id
      sourceId=sn.navbar.user.username
      value=0 hidden=true /> ...
  </dv.tx> ...
</dv.component>

```

submit-post.html

**Slacker News**
Top New | Submit
 ben Sign Out

*sn.navbar*

# Submit

Title \*

Url \*

Submit

*post.create-object*

```

<dv.component name="submit-post">
  <sn.navbar /> ...
  <dv.tx> transaction
    <dv.gen-id /> ...
    <post.create-object id=dv.gen-id.id
      initialValue={ author: sn.navbar.user.username }
      showExclude=["author"] buttonLabel="Submit"
      newObjectSavedText="Post submitted" />
    <scoreposts.create-score targetId=dv.gen-id.id
      sourceId=sn.navbar.user.username
      value=0 hidden=true /> ...
  </dv.tx> ...
</dv.component>

```

submit-post.html

**Slacker News**
Top New | Submit
ben Sign Out

*sn.navbar*

## Submit

Title \*

Url \*

Submit

*post.create-object*

generate id

```
<dv.component name="submit-post">
  <sn.navbar /> ...
  <dv.tx>
    <dv.gen-id /> ...
    <post.create-object id=dv.gen-id.id
      initialValue={ author: sn.navbar.user.username }
      showExclude=["author"] buttonLabel="Submit"
      newObjectSavedText="Post submitted" />
    <scoreposts.create-score targetId=dv.gen-id.id
      sourceId=sn.navbar.user.username
      value=0 hidden=true /> ...
  </dv.tx> ...
</dv.component>
```

id used

id used

submit-post.html

**Slacker News**   Top   New | Submit   ben   Sign Out

*sn.navbar*

## Submit

Title \*

Url \*

Submit

*post.create-object*

```

<dv.component name="submit-post">
  <sn.navbar /> ...
  <dv.tx>
    <dv.gen-id /> ...
    <post.create-object id=dv.gen-id.id
      initialValue={ author: sn.navbar.user.username }
      showExclude=["author"] buttonLabel="Submit"
      newObjectSavedText="Post submitted" />
    <scoreposts.create-score targetId=dv.gen-id.id
      sourceId=sn.navbar.user.username
      value=0 hidden=true /> ...
  </dv.tx> ...
</dv.component>

```

submit-post.html

**Slacker News**
Top New | Submit
 ben Sign Out

*sn.navbar*

## Submit

Title \*

Url \*

Submit
 *post.create-object*

user clicks on  
button

trigger *run*

```
<dv.component name="submit-post">
  <sn.navbar /> ...
  <dv.tx>
    <dv.gen-id /> ...
    <post.create-object id=dv.gen-id.id
      initialValue={ author: sn.navbar.user.username }
      showExclude=["author"] buttonLabel="Submit"
      newObjectSavedText="Post submitted" />
    <scoreposts.create-score targetId=dv.gen-id.id
      sourceId=sn.navbar.user.username
      value=0 hidden=true /> ...
  </dv.tx> ...
</dv.component>
```

submit-post.html

**Slacker News**    Top   New | Submit    ben   Sign Out

*sn.navbar*

Submit

Title \*

Url \*

Submit

*post.create-object*

user clicks on  
button

```

<dv.component name="submit-post">
  <sn.navbar />
  <dv.tx>
    <dv.gen-id /> ..
    <post.create-object id=dv.gen-id.id
      initialValue={ author: sn.navbar.user.username }
      showExclude=["author"] buttonLabel="Submit"
      newObjectSavedText="Post submitted" />
    <scoreposts.create-score targetId=dv.gen-id.id
      sourceId=sn.navbar.user.username
      value=0 hidden=true /> ...
  </dv.tx> ...
</dv.component>

```

run create-post action

trigger run

run create-score action

submit-post.html

**Slacker News**
Top New | Submit
 ben Sign Out

*sn.navbar*

## Submit

Title \*

Url \*

Submit
 *post.create-object*

user clicks on button



```

<dv.component name="show-post"> ...
  <post.show-object id=$id hidden=true /> ...
  <dv.if condition=post.show-object.loadedObject>
    <sn.upvote id=$id user=$user />
    <a href=post.show-object.loadedObject.url>
      {{post.show-object.loadedObject.title}}
    </a> ...
    (<post.show-url showBaseUrlOnly=true
      url=post.show-object.loadedObject.url />)
    posted by {{post.show-object.loadedObject.author}}
    <dv.show-date format='time-ago'
      date=post.show-object.loadedObject.timestamp /> ...
    <scoreposts.show-target
      id=$id showId=false
      showScores=false totalLabel="" /> points
    <dv.link href="/post" params={ id: $id }>
      comments
    </dv.link>
  </dv.if>
</dv.component>

```

show-post.html



sn.upvote

**SPLASH 2019** (2019.splashcon.org)

dv.show-date

dv.link

posted by ben an hour ago | 2 points | comments

scoreposts.show-target

load post

component input

```
<dv.component name="show-post" ...
  <post.show-object id=$id hidden=true /> ...
  <dv.if condition=post.show-object.loadedObject>
    <sn.upvote id=$id user=$user />
    <a href=post.show-object.loadedObject.url>
      {{post.show-object.loadedObject.title}}
    </a> ...
    (<post.show-url showBaseUrlOnly=true
      url=post.show-object.loadedObject.url />)
    posted by {{post.show-object.loadedObject.author}}
    <dv.show-date format='time-ago'
      date=post.show-object.loadedObject.timestamp /> ...
    <scoreposts.show-target
      id=$id showId=false
      showScores=false totalLabel="" /> points
    <dv.link href="/post" params={ id: $id }>
      comments
    </dv.link>
  </dv.if>
</dv.component>
```

show-post.html



sn.upvote

**SPLASH 2019** (2019.splashcon.org)

posted by ben

dv.show-date

an hour ago

2

points

dv.link

comments

scoreposts.show-target

```

<dv.component name="show-post"> ...
  <post.show-object id=$id hidden=true /> ...
  <dv.if condition=post.show-object.loadedObject>
    <sn.upvote id=$id user=$user />
    <a href=post.show-object.loadedObject.url>
      {{post.show-object.loadedObject.title}}
    </a> ...
    (<post.show-url showBaseUrlOnly=true
      url=post.show-object.loadedObject.url />)
    posted by {{post.show-object.loadedObject.author}}
    <dv.show-date format='time-ago'
      date=post.show-object.loadedObject.timestamp /> ...
    <scoreposts.show-target
      id=$id showId=false
      showScores=false totalLabel="" /> points
    <dv.link href="/post" params={ id: $id }>
      comments
    </dv.link>
  </dv.if>
</dv.component>

```

post url

post title

post author

show-post.html



sn.upvote

**SPLASH 2019** (2019.splashcon.org)

dv.show-date

dv.link

posted by ben an hour ago | 2 points | comments

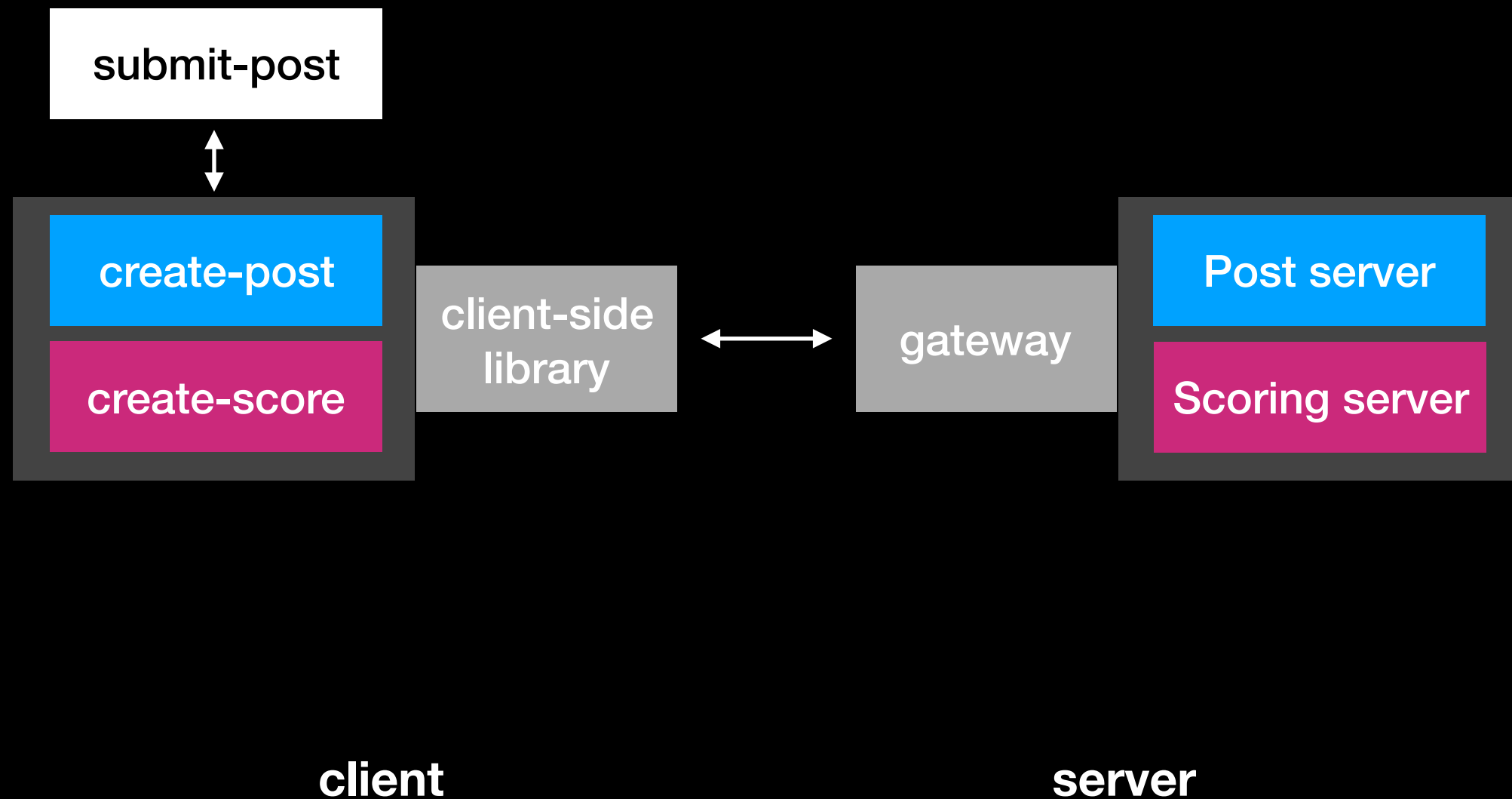
scoreposts.show-target

# Building Apps with Déjà Vu

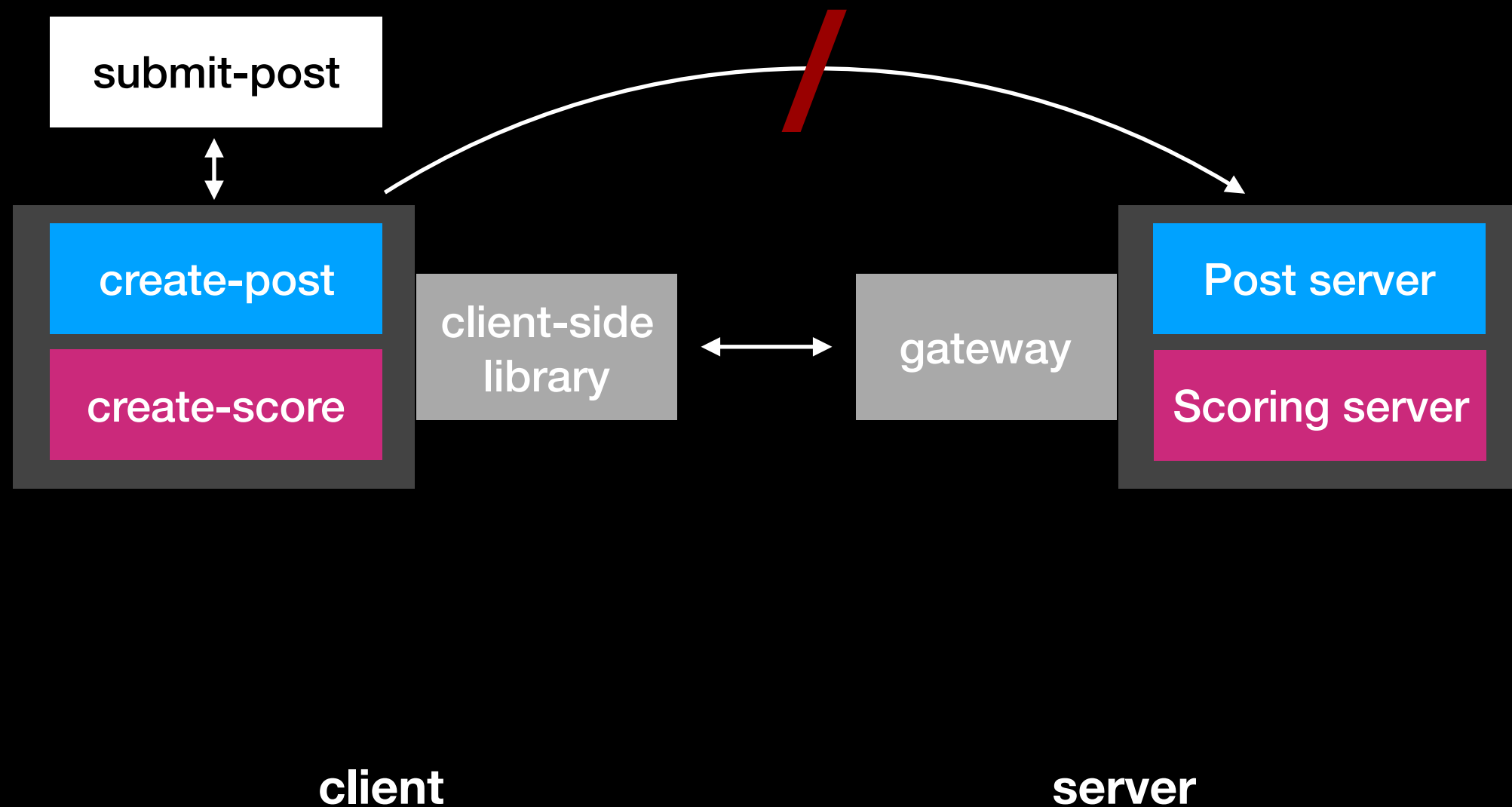
- ▶ choose, include and configure concepts
- ▶ include concept components, sync & bind ids

# Platform Implementation

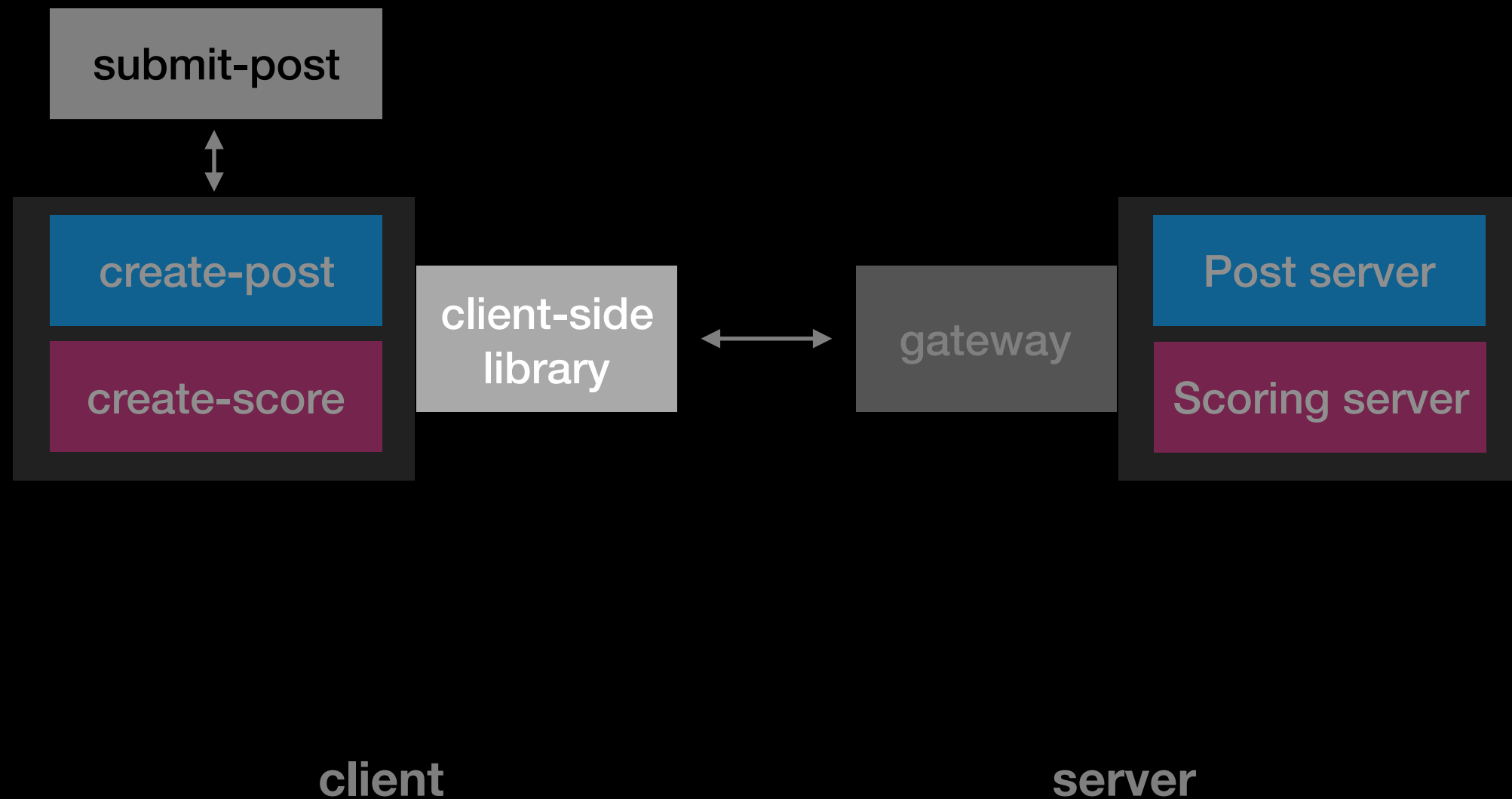
# Architecture



# Architecture

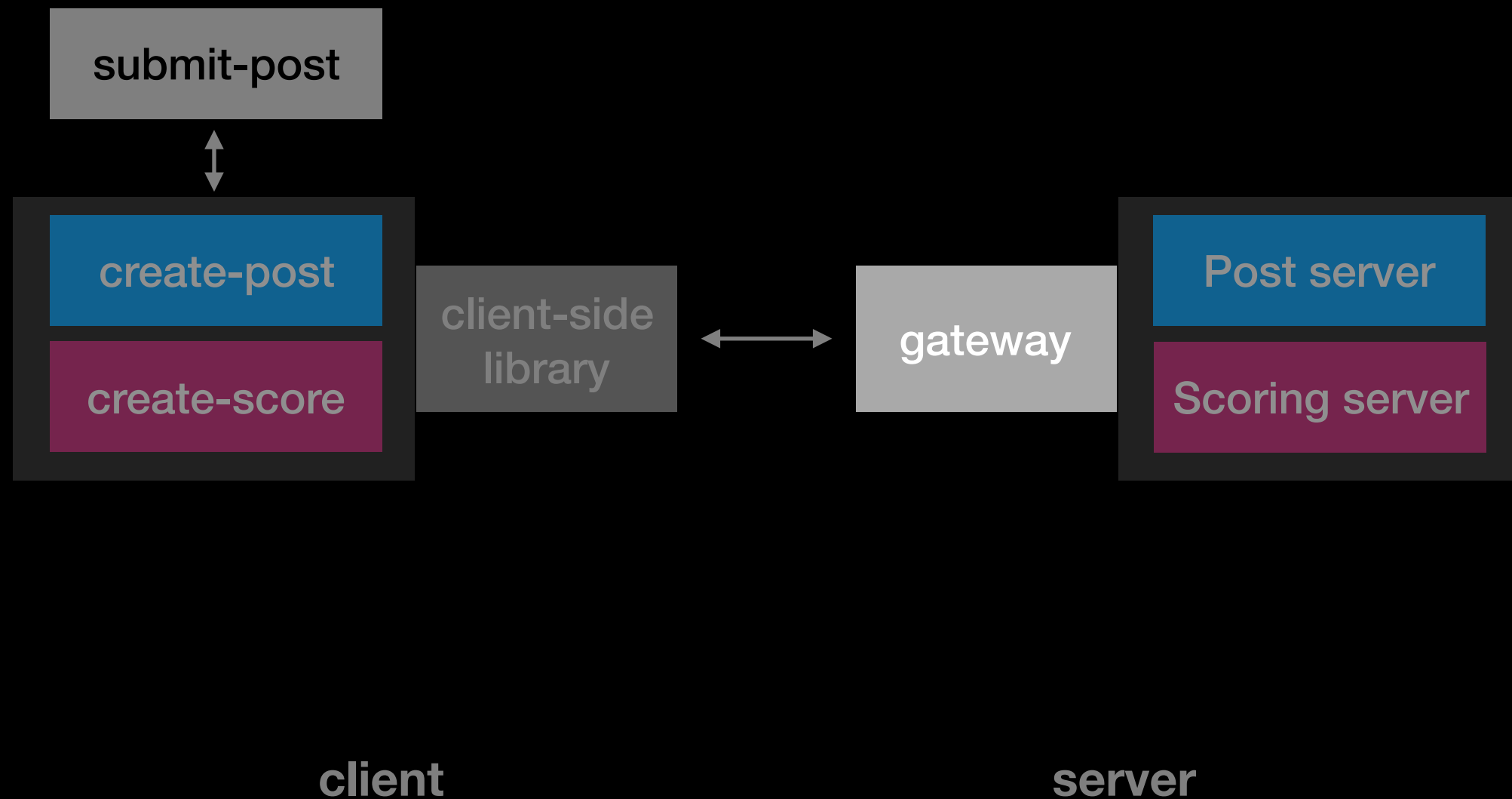


# Architecture

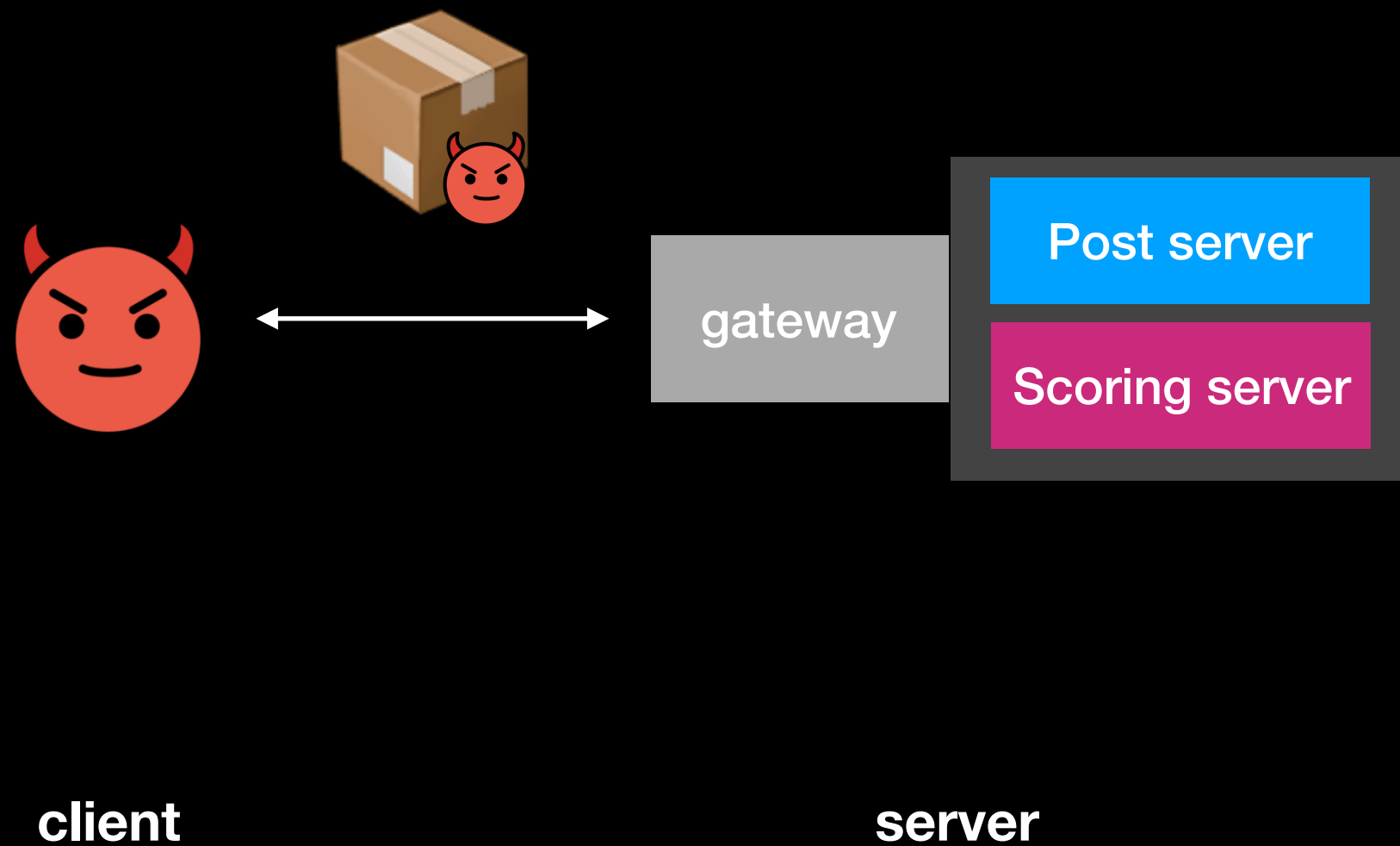




# Architecture



# Security



# Security

```
<dv.component name="show-post"> ...  
<post.show-object id=$id hidden=true /> ...  
</dv.component>
```

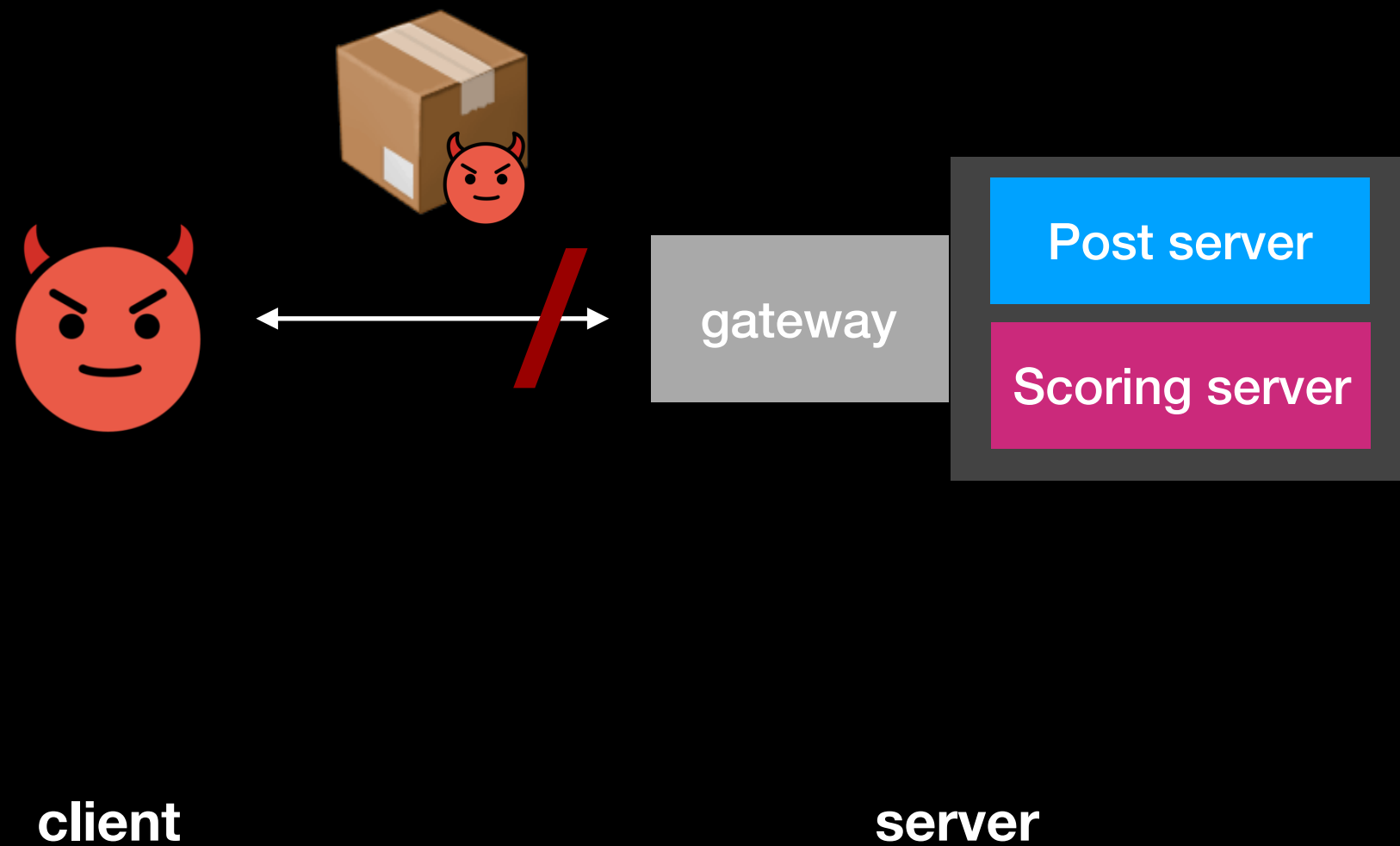
```
<dv.component name="submit-post">  
  <sn.navbar /> ...  
</dv.component>
```

```
{  
  "name": "sn",  
  "usedConcepts": {  
    "authentication": {},  
    "comment": {},  
    "post": {  
      "name": "Property",  
      "config": {  
        "schema": {  
          "title": "Post", "type": "object",  
          "properties": {  
            "author": { "type": "string" },  
            "title": { "type": "string" },  
            "url": { "type": "string", "format": "url" }  
          },  
          "required": [ "author", "title", "url" ]  
        }  
      }  
    },  
    "scoreposts": { "name": "Scoring" },  
    "scorecomments": { "name": "Scoring" }  
  },  
  "routes": [  
    { "path": "", "component": "home" },  
    { "path": "/login", "component": "login" },  
    { "path": "/post", "component": "show-post-details" },  
    ...  
  ]  
}
```



gateway

# Security



# Evaluation

# 1. Comparison with Standard Approach

# Post with Rating

# Post with Rating

**Sweet Spots**

Find

Search

Hello, sperezde@mit.edu!

Logout

Profile

## Add a spot:

[< Back to home](#)

**Name**

**Category**

**Floor**

**Review**

★ ★ ★ ★ ★

★ ★ ★ ★ ★

Submit Spot

MapSatellite

Google

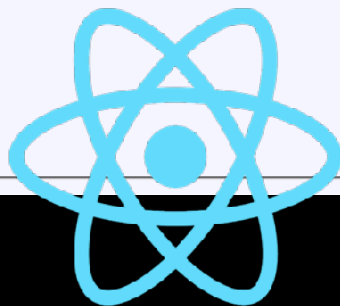
Map data ©2019 Google Terms of Use Report a map error



# Standard Approach

```
class SubmitPost extends React.Component {
  constructor(props) {
    super(props);
    this.state = { p: new Post(), r: 0 };
    this.handleChange = this.handleChange.bind(this);
    this.handleSubmit = this.handleSubmit.bind(this);
  }
  handleChange(e) {
    this.setState({ [e.target.name]: e.target.value });
  }
  handleSubmit(e) {
    PostService.savePost(this.state.p, this.state.r);
    e.preventDefault();
  }
  render() {
    return (
      <form onSubmit={this.handleSubmit}>
        <PostInput onChange={this.handleChange} />
        <RatingInput onChange={this.handleChange} />
        <button type="submit">Submit</button>
      </form>
    );
  }
}
```

client



```
function savePost(p: Post, r: number) {
  if (Post.isValid(p) && Rating.isValid(r)) {
    db.save(p);
  }
}
```

monolith

```
function savePost(p: Post, r: number) {
  if (PostService.isValid(p) &&
    RatingService.isValid(r)) {
    const newP = PostService.newPost(p);
    RatingService.newRating(newP.id, r);
  }
}
```

microservices

server

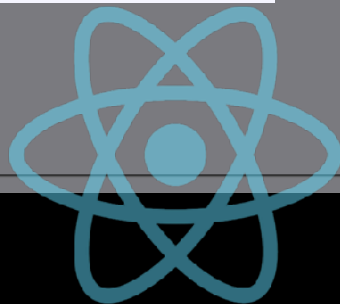
# Standard Approach

```
class SubmitPost extends React.Component {
  constructor(props) {
    super(props);
    this.state = { p: new Post(), r: 0 };
    this.handleChange = this.handleChange.bind(this);
    this.handleSubmit = this.handleSubmit.bind(this);
  }
  handleChange(e) {
    this.setState({ [e.target.name]: e.target.value });
  }
  handleSubmit(e) {
    PostService.savePost(this.state.p, this.state.r);
    e.preventDefault();
  }
  render() {
    return (
      <form onSubmit={this.handleSubmit}>
        <PostInput onChange={this.handleChange} />
        <RatingInput onChange={this.handleChange} />
        <button type="submit">Submit</button>
      </form>
    );
  }
}
```

subscribe to events

```
<form onSubmit={this.handleSubmit}>
  <PostInput onChange={this.handleChange} />
  <RatingInput onChange={this.handleChange} />
  <button type="submit">Submit</button>
</form>
```

client



```
function savePost(p: Post, r: number) {
  if (Post.isValid(p) && Rating.isValid(r)) {
    db.save(p);
  }
}
```

monolith

```
function savePost(p: Post, r: number) {
  if (PostService.isValid(p) &&
    RatingService.isValid(r)) {
    const newP = PostService.newPost(p);
    RatingService.newRating(newP.id, r);
  }
}
```

microservices

server

# Standard Approach

```
class SubmitPost extends React.Component {
  constructor(props) {
    super(props);
    this.state = { p: new Post(), r: 0 };
    this.handleChange = this.handleChange.bind(this);
    this.handleSubmit = this.handleSubmit.bind(this);
  }

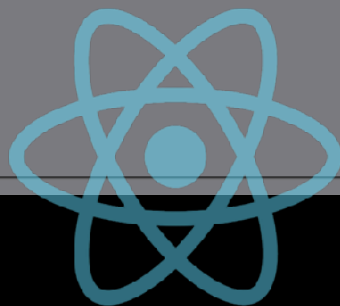
  handleChange(e) {
    this.setState({ [e.target.name]: e.target.value });
  }

  handleSubmit(e) {
    PostService.savePost(this.state.p, this.state.r);
    e.preventDefault();
  }

  render() {
    return (
      <form onSubmit={this.handleSubmit}>
        <PostInput onChange={this.handleChange} />
        <RatingInput onChange={this.handleChange} />
        <button type="submit">Submit</button>
      </form>
    );
  }
}
```

write event handler

client



```
function savePost(p: Post, r: number) {
  if (Post.isValid(p) && Rating.isValid(r)) {
    db.save(p);
  }
}
```

monolith

```
function savePost(p: Post, r: number) {
  if (PostService.isValid(p) &&
    RatingService.isValid(r)) {
    const newP = PostService.newPost(p);
    RatingService.newRating(newP.id, r);
  }
}
```

microservices

server

# Standard Approach

```
class SubmitPost extends React.Component {
  constructor(props) {
    super(props);
    this.state = { p: new Post(), r: 0 };
    this.handleChange = this.handleChange.bind(this);
    this.handleSubmit = this.handleSubmit.bind(this);
  }
  handleChange(e) {
    this.setState({ [e.target.name]: e.target.value });
  }
  handleSubmit(e) {
    PostService.savePost(this.state.p, this.state.r);
    e.preventDefault();
  }
  render() {
    return (
      <form onSubmit={this.handleSubmit}>
        <PostInput onChange={this.handleChange} />
        <RatingInput onChange={this.handleChange} />
        <button type="submit">Submit</button>
      </form>
    );
  }
}
```

aggregate request data

client



```
function savePost(p: Post, r: number) {
  if (Post.isValid(p) && Rating.isValid(r)) {
    db.save(p);
  }
}
```

monolith

```
function savePost(p: Post, r: number) {
  if (PostService.isValid(p) &&
    RatingService.isValid(r)) {
    const newP = PostService.newPost(p);
    RatingService.newRating(newP.id, r);
  }
}
```

microservices

server

# Standard Approach

```
class SubmitPost extends React.Component {
  constructor(props) {
    super(props);
    this.state = { p: new Post(), r: 0 };
    this.handleChange = this.handleChange.bind(this);
    this.handleSubmit = this.handleSubmit.bind(this);
  }
  handleChange(e) {
    this.setState({ [e.target.name]: e.target.value });
  }
  handleSubmit(e) {
    PostService.savePost(this.state.p, this.state.r);
    e.preventDefault();
  }
  render() {
    return (
      <form onSubmit={this.handleSubmit}>
        <PostInput onChange={this.handleChange} />
        <RatingInput onChange={this.handleChange} />
        <button type="submit">Submit</button>
      </form>
    );
  }
}
```

client



```
function savePost(p: Post, r: number) {
  if (Post.isValid(p) && Rating.isValid(r)) {
    db.save(p);
  }
}
```

monolith

```
function savePost(p: Post, r: number) {
  if (PostService.isValid(p) &&
    RatingService.isValid(r)) {
    const newP = PostService.newPost(p);
    RatingService.newRating(newP.id, r);
  }
}
```

microservices

server



# Standard Approach

```
class SubmitPost extends React.Component {
  constructor(props) {
    super(props);
    this.state = { p: new Post(), r: 0 };
    this.handleChange = this.handleChange.bind(this);
    this.handleSubmit = this.handleSubmit.bind(this);
  }
  handleChange(e) {
    this.setState({ [e.target.name]: e.target.value });
  }
  handleSubmit(e) {
    PostService.savePost(this.state.p, this.state.r);
    e.preventDefault();
  }
  render() {
    return (
      <form onSubmit={this.handleSubmit}>
        <PostInput onChange={this.handleChange} />
        <RatingInput onChange={this.handleChange} />
        <button type="submit">Submit</button>
      </form>
    );
  }
}
```

client



```
function savePost(p: Post, r: number) {
  if (Post.isValid(p) && Rating.isValid(r)) {
    db.save(p);
  }
}
```

validate request

monolith

```
function savePost(p: Post, r: number) {
  if (PostService.isValid(p) &&
    RatingService.isValid(r)) {
    const newP = PostService.newPost(p);
    RatingService.newRating(newP.id, r);
  }
}
```

validate request

microservices

server

# Standard Approach

```
class SubmitPost extends React.Component {
  constructor(props) {
    super(props);
    this.state = { p: new Post(), r: 0 };
    this.handleChange = this.handleChange.bind(this);
    this.handleSubmit = this.handleSubmit.bind(this);
  }
  handleChange(e) {
    this.setState({ [e.target.name]: e.target.value });
  }
  handleSubmit(e) {
    PostService.savePost(this.state.p, this.state.r);
    e.preventDefault();
  }
  render() {
    return (
      <form onSubmit={this.handleSubmit}>
        <PostInput onChange={this.handleChange} />
        <RatingInput onChange={this.handleChange} />
        <button type="submit">Submit</button>
      </form>
    );
  }
}
```

save to  
database

integrate  
services

```
function savePost(p: Post, r: number) {
  if (Post.isValid(p) && Rating.isValid(r)) {
    db.save(p);
  }
}
```

monolith

```
function savePost(p: Post, r: number) {
  if (PostService.isValid(p) &&
    RatingService.isValid(r)) {
    const newP = PostService.newPost(p);
    RatingService.newRating(newP.id, r);
  }
}
```

microservices



client

server

# Déjà Vu

```
<dv.component name="submit-post">  
  <dv.tx>  
    <dv.gen-id />  
    <property.create-object id=dv.gen-id.id />  
    <rating.rate-target targetId=dv.gen-id.id />  
    <dv.button>Submit</dv.button>  
  </dv.tx>  
</dv.component>
```



# Déjà Vu

```
<dv.component name="submit-post">  
  <dv.tx> synchronize  
    <dv.gen-id />  
    <property.create-object id=dv.gen-id.id />  
    <rating.rate-target targetId=dv.gen-id.id />  
    <dv.button>Submit</dv.button>  
  </dv.tx>  
</dv.component>
```

# Déjà Vu

```
<dv.component name="submit-post">
  <dv.tx>
    <dv.gen-id />
    <property.create-object id=dv.gen-id.id />
    <rating.rate-target targetId=dv.gen-id.id />
    <dv.button>Submit</dv.button>
  </dv.tx>
</dv.component>
```

bind ids

# Déjà Vu

```
<dv.component name="submit-post">
  <dv.tx>
    <dv.gen-id />
    <property.create-object id=dv.gen-id.id />
    <rating.rate-target targetId=dv.gen-id.id />
    <dv.button>Submit</dv.button>
  </dv.tx>
</dv.component>
```

Déjà Vu subscribes to run events, runs component handlers, aggregates the data client-side and sends request, unpacks the request server-side and coordinates the calls to the back-end services of each concept so that they happen in a tx if necessary

## 2. Case Study

# Student Projects

App	Purpose
Accord	Support musical bands in the selection of setlists
ChoreStar	Make it easy for parents to assign chores to children
EasyPick	Recommend classes to college students
GroceryShip	Facilitate peer grocery delivery between students
Lingua	Develop language skills by chatting with native speakers
Listify	Crowdsource opinion-based rankings of anything
LiveScorecard	Provide a live leaderboard for climbing competitions
MapCampus	Allow students to plan events on campus
Phoenix	Help people discuss mental health and make friends
Potluck	Help people plan parties where guests bring supplies
Rendezvous	Plan public events on campus
SweetSpots	Mark spots on a map and review spots added by others

AccordDashboardAbout

Music Lovers

Rock Music

Country Music

Jazz

Classics

Mozart Symphony No.3

by Mozart

Lyrics

(none)

Media

youtube.com/AZ3rDksE

Rubinstein's piano version

youtube.com/fBoeJD9sk

2010 Berlin Performance

youtube.com/JLldj8DE

2011 Munich Performance

Comments

No comments to show

Write your comment \*

Create Comment

Added By Yuni1

Your Rating: ★★★★★

Average Rating: ★★★★★ 1 vote

ChoreStar

eddySign Out

StarBank Total: ★0

Chores

Uncompleted Chores

No uncompleted chores

Chores Pending Approval

No chores pending approval

Completed and Approved Chores

No approved chores

Rewards

Rewards You Can Buy

Rewards You Cannot Buy Yet

EasyPick

Logout

Filters

☒ HASS

☐ REST

☒ CIH

☐ CIHW

Grading

★☆☆☆☆ 1 & up

★☆☆☆☆ 2 & up

★☆☆☆☆ 3 & up

★☆☆☆☆ 4 & up

Pace

★☆☆☆☆ 1 & up

★☆☆☆☆ 2 & up

★☆☆☆☆ 3 & up

★☆☆☆☆ 4 & up

Contents

★☆☆☆☆ 1 & up

★☆☆☆☆ 2 & up

★☆☆☆☆ 3 & up

★☆☆☆☆ 4 & up

Overall

★☆☆☆☆ 1 & up

★☆☆☆☆ 2 & up

★☆☆☆☆ 3 & up

★☆☆☆☆ 4 & up

Classes

24,900 Intro to Linguistics

HASS CIH CIHW

Pace ★★★★★ 3 votes

Grading ★★★★★ 3 votes

Contents ★★★★★ 3 votes

Overall ★★★★★ 3 votes

Show Comments

You are delivering

Delivering to: unisushi

Pick Up Time: 5pm

Pick Up Location: new house

Items

Delivered

Your Requests

You don't have any non in-delivery requests.

To Deliver

	Item Name	Quantity	Price Estimate	Due	Pickup Location	Tips
✓	toothbrush	1	2	8/9	Senior House	0.5
✓	apples	5	5	8/9	EC	1
✓	bananas	5	5	8/9	EC	1
✓	pork belly	2	15	8/9	EC	2

Deliver Now

Lingua

Find Pen Pals119Logout

Chats:

Brett Yang

Eddy Chen

Hilary Hahn

Conversation:

Hilary Hahn : ★★★★★

Brett Yang : ★★★★★

Rate Your Conversation: ★★★★★

Brett Yang : Aug 7, 2019 3:19 PM

Sure!!

Correct Words

Hilary Hahn : Aug 7, 2019 3:19 PM

Hi Brett! I'm learning French right now! I saw that you are practicing English. Shall we talk in each other's language?

Correct Words

Content \*

Create Message

Corrections:

No Corrections

Listify

pikachuSign Out

Who is the best cat?

Created by pikachuClose Voting

1. Niuniu

2. Puru

3. Luna

4. Maple

5. Loki

6. Gina

7. Eva

Welcome to Live ScoreCard

I am a ...

Host

Climber

Spectator

username \*

password \*

LOGIN

Don't have an account? Sign up here

mapMiT

HomeMy EventsMy GroupsLogged in as: pikachuSign Out

My Events

Create Event

Fire spinning spinjam

Host:

Description:

Time:

Location:

Room Number:

Location Description:

Spin fire together!  
Thu Aug 8, 2019 11:11 AM - 12:11 PM  
Ames Street, East Cambridge, Cambridge, Middlesex County, Massachusetts, 02142, USA  
EC courtyard  
EC courtyard

Cancel Event

Coffee night

Host:

Description:

Time:

Location:

Room Number:

Location Description:

let's brew coffee together and drink them  
Wed Aug 21, 2019 12:11 AM - 1:11 AM  
McDermott Court, East Cambridge, Cambridge, Middlesex County, Massachusetts, USA  
H511  
EC H511 - my room!

Cancel Event

Dashboard

Personal

Journal

FAQ

Availability

Week 32 in 2019

Sun

Mon

Tue

Wed

Thu

Fri

Sat

00:00

01:00

02:00

03:00

04:00

05:00

06:00

07:00

08:00

09:00

10:00

11:00

12:00

13:00

14:00

15:00

16:00

17:00

18:00

19:00

20:00

21:00

22:00

23:00

About Jsnov

✓ Enable Updates!

Description

I don't know what I'm talking about things.

201 Crow

807-82

Location Rules

1392

Save Personal Info

Topics

Already Chosen Topics

No topics selected

Choose Topics

newalitygenderfamilymapgendervery

about

Pollbook

benSign Out

Your Parties

Create a New Party

My birthday

Date: Thu Oct 24, 2019 3:00 PM - Sat Oct 26, 2019 4:00 PM

Host: ben

Location: My place

come celebrate my bday!

Guest List

No members

Add Guest

Choose Guest

Add to Guest List

Supply List

Item

Quantity

Unit

oreo cookies1bag

Count \*

Claim

No claims

RENDEZVOUS

My ProfileLogout

Create Event

Hosted By jsnow

Cambridge

Massachusetts USA

Sweet Spots

Hello, ben!Logout

quiet study room

Reviews

Rating: 6 out of 6

"very quiet"

By ben (Rep: 1)

Concept/App	Accord	Chorestar	EasyPick	GroceryShip	Lingua	Listify	LiveScorecard	MapCampus	Phoenix	Potluck	Rendezvous	SweetSpots	#Apps	#Instances
Authentication	1	2	1	1	1	1	1	1	1	1	1	1	12	13
Authorization	1	1	1	1	1	1	1	2	1	1	1	1	12	13
Chat	0	0	0	0	1	0	0	0	0	0	0	0	1	1
Comment	1	0	1	0	0	0	0	0	1	0	1	1	5	5
Event	0	0	0	0	0	0	1	1	0	1	1	0	4	4
Follow	0	0	0	0	0	0	0	0	0	0	0	1	1	1
Geolocation	0	0	0	0	0	0	0	1	1	0	1	1	4	4
Group	1	0	0	0	2	1	3	1	0	1	1	0	7	10
Label	0	0	0	0	0	0	0	0	1	0	1	1	3	3
Match	0	0	0	0	0	0	0	0	1	0	0	0	1	1
Passkey	0	0	0	0	0	0	2	0	0	0	0	0	1	2
Property	5	3	3	3	2	2	4	3	2	2	2	1	12	32
Ranking	0	0	0	0	0	1	0	0	0	0	0	0	1	1
Rating	1	0	4	1	1	0	0	0	0	0	0	1	5	8
Schedule	0	0	0	0	0	0	0	0	1	0	0	0	1	1
Scoring	0	0	0	0	0	1	2	0	0	0	0	2	3	5
Task	0	1	0	0	0	0	1	0	0	0	0	0	2	2
Transfer	0	1	0	0	0	0	0	0	0	1	0	0	2	2
# Concept Types	6	5	5	4	6	6	8	6	8	6	8	9		
# Concept Instances	10	8	10	6	8	7	15	9	9	7	9	10		

Concept/App	Accord	Chorestar	EasyPick	GroceryShip	Lingua	Listify	LiveScorecard	MapCampus	Phoenix	Potluck	Rendezvous	SweetSpots	#Apps	#Instances
Authentication	1	2	1	1	1	1	1	1	1	1	1	1	12	13
Authorization	1	1	1	1	1	1	1	2	1	1	1	1	12	13
Chat	0	0	0	0	1	0	0	0	0	0	0	0	1	1
Comment	1	0	1	0	0	0	0	0	1	0	1	1	5	5
Event	0	0	0	0	0	0	1	1	0	1	1	0	4	4
Follow	0	0	0	0	0	0	0	0	0	0	0	1	1	1
Geolocation	0	0	0	0	0	0	0	1	1	0	1	1	4	4
Group	1	0	0	0	2	1	3	1	0	1	1	0	7	10
Label	0	0	0	0	0	0	0	0	1	0	1	1	3	3
Match	0	0	0	0	0	0	0	0	1	0	0	0	1	1
Passkey	0	0	0	0	0	0	2	0	0	0	0	0	1	2
Property	5	3	3	3	2	2	4	3	2	2	2	1	12	32
Ranking	0	0	0	0	0	1	0	0	0	0	0	0	1	1
Rating	1	0	4	1	1	0	0	0	0	0	0	1	5	8
Schedule	0	0	0	0	0	0	0	0	1	0	0	0	1	1
Scoring	0	0	0	0	0	1	2	0	0	0	0	2	3	5
Task	0	1	0	0	0	0	1	0	0	0	0	0	2	2
Transfer	0	1	0	0	0	0	0	0	0	1	0	0	2	2
# Concept Types	6	5	5	4	6	6	8	6	8	6	8	9		
# Concept Instances	10	8	10	6	8	7	15	9	9	7	9	10		



Concept/App	Accord	Chorestar	EasyPick	GroceryShip	Lingua	Listify	LiveScorecard	MapCampus	Phoenix	Potluck	Rendezvous	SweetSpots	#Apps	#Instances
Authentication	1	2	1	1	1	1	1	1	1	1	1	1	12	13
Authorization	1	1	1	1	1	1	1	2	1	1	1	1	12	13
Chat	0	0	0	0	1	0	0	0	0	0	0	0	1	1
Comment	1	0	1	0	0	0	0	0	1	0	1	1	5	5
Event	0	0	0	0	0	0	1	1	0	1	1	0	4	4
Follow	0	0	0	0	0	0	0	0	0	0	0	1	1	1
Geolocation	0	0	0	0	0	0	0	1	1	0	1	1	4	4
Group	1	0	0	0	2	1	3	1	0	1	1	0	7	10
Label	0	0	0	0	0	0	0	0	1	0	1	1	3	3
Match	0	0	0	0	0	0	0	0	1	0	0	0	1	1
Passkey	0	0	0	0	0	0	2	0	0	0	0	0	1	2
Property	5	3	3	3	2	2	4	3	2	2	2	1	12	32
Ranking	0	0	0	0	0	1	0	0	0	0	0	0	1	1
Rating	1	0	4	1	1	0	0	0	0	0	0	1	5	8
Schedule	0	0	0	0	0	0	0	0	1	0	0	0	1	1
Scoring	0	0	0	0	0	1	2	0	0	0	0	2	3	5
Task	0	1	0	0	0	0	1	0	0	0	0	0	2	2
Transfer	0	1	0	0	0	0	0	0	0	1	0	0	2	2
# Concept Types	6	5	5	4	6	6	8	6	8	6	8	9		
# Concept Instances	10	8	10	6	8	7	15	9	9	7	9	10		

Concept/App	Accord	Chorestar	EasyPick	GroceryShip	Lingua	Listify	LiveScorecard	MapCampus	Phoenix	Potluck	Rendezvous	SweetSpots	#Apps	#Instances
Authentication	1	2	1	1	1	1	1	1	1	1	1	1	12	13
Authorization	1	1	1	1	1	1	1	2	1	1	1	1	12	13
Chat	0	0	0	0	1	0	0	0	0	0	0	0	1	1
Comment	1	0	1	0	0	0	0	0	1	0	1	1	5	5
Event	0	0	0	0	0	0	1	1	0	1	1	0	4	4
Follow	0	0	0	0	0	0	0	0	0	0	0	1	1	1
Geolocation	0	0	0	0	0	0	0	1	1	0	1	1	4	4
Group	1	0	0	0	2	1	3	1	0	1	1	0	7	10
Label	0	0	0	0	0	0	0	0	1	0	1	1	3	3
Match	0	0	0	0	0	0	0	0	1	0	0	0	1	1
Passkey	0	0	0	0	0	0	2	0	0	0	0	0	1	2
Property	5	3	3	3	2	2	4	3	2	2	2	1	12	32
Ranking	0	0	0	0	0	1	0	0	0	0	0	0	1	1
Rating	1	0	4	1	1	0	0	0	0	0	0	1	5	8
Schedule	0	0	0	0	0	0	0	0	1	0	0	0	1	1
Scoring	0	0	0	0	0	1	2	0	0	0	0	2	3	5
Task	0	1	0	0	0	0	1	0	0	0	0	0	2	2
Transfer	0	1	0	0	0	0	0	0	0	1	0	0	2	2
# Concept Types	6	5	5	4	6	6	8	6	8	6	8	9		
# Concept Instances	10	8	10	6	8	7	15	9	9	7	9	10		

Concept/App	Accord	Chorestar	EasyPick	GroceryShip	Lingua	Listify	LiveScorecard	MapCampus	Phoenix	Potluck	Rendezvous	SweetSpots	#Apps	#Instances
Authentication	1	2	1	1	1	1	1	1	1	1	1	1	12	13
Authorization	1	1	1	1	1	1	1	2	1	1	1	1	12	13
Chat	0	0	0	0	1	0	0	0	0	0	0	0	1	1
Comment	1	0	1	0	0	0	0	0	1	0	1	1	5	5
Event	0	0	0	0	0	0	1	1	0	1	1	0	4	4
Follow	0	0	0	0	0	0	0	0	0	0	0	1	1	1
Geolocation	0	0	0	0	0	0	0	1	1	0	1	1	4	4
Group	1	0	0	0	2	1	3	1	0	1	1	0	7	10
Label	0	0	0	0	0	0	0	0	1	0	1	1	3	3
Match	0	0	0	0	0	0	0	0	1	0	0	0	1	1
Passkey	0	0	0	0	0	0	2	0	0	0	0	0	1	2
Property	5	3	3	3	2	2	4	3	2	2	2	1	12	32
Ranking	0	0	0	0	0	1	0	0	0	0	0	0	1	1
Rating	1	0	4	1	1	0	0	0	0	0	0	1	5	8
Schedule	0	0	0	0	0	0	0	0	1	0	0	0	1	1
Scoring	0	0	0	0	0	1	2	0	0	0	0	2	3	5
Task	0	1	0	0	0	0	1	0	0	0	0	0	2	2
Transfer	0	1	0	0	0	0	0	0	0	1	0	0	2	2
# Concept Types	6	5	5	4	6	6	8	6	8	6	8	9		
# Concept Instances	10	8	10	6	8	7	15	9	9	7	9	10		

Concept/App	Accord	Chorestar	EasyPick	GroceryShip	Lingua	Listify	LiveScorecard	MapCampus	Phoenix	Potluck	Rendezvous	SweetSpots	#Apps	#Instances
Authentication	1	2	1	1	1	1	1	1	1	1	1	1	12	13
Authorization	1	1	1	1	1	1	1	2	1	1	1	1	12	13
Chat	0	0	0	0	1	0	0	0	0	0	0	0	1	1
Comment	1	0	1	0	0	0	0	0	1	0	1	1	5	5
Event	0	0	0	0	0	0	1	1	0	1	1	0	4	4
Follow	0	0	0	0	0	0	0	0	0	0	0	1	1	1
Geolocation	0	0	0	0	0	0	0	1	1	0	1	1	4	4
Group	1	0	0	0	2	1	3	1	0	1	1	0	7	10
Label	0	0	0	0	0	0	0	0	1	0	1	1	3	3
Match	0	0	0	0	0	0	0	0	1	0	0	0	1	1
Passkey	0	0	0	0	0	0	2	0	0	0	0	0	1	2
Property	5	3	3	3	2	2	4	3	2	2	2	1	12	32
Ranking	0	0	0	0	0	1	0	0	0	0	0	0	1	1
Rating	1	0	4	1	1	0	0	0	0	0	0	1	5	8
Schedule	0	0	0	0	0	0	0	0	1	0	0	0	1	1
Scoring	0	0	0	0	0	1	2	0	0	0	0	2	3	5
Task	0	1	0	0	0	0	1	0	0	0	0	0	2	2
Transfer	0	1	0	0	0	0	0	0	0	1	0	0	2	2
# Concept Types	6	5	5	4	6	6	8	6	8	6	8	9		
# Concept Instances	10	8	10	6	8	7	15	9	9	7	9	10		

Concept/App	Accord	Chorestar	EasyPick	GroceryShip	Lingua	Listify	LiveScorecard	MapCampus	Phoenix	Potluck	Rendezvous	SweetSpots	#Apps	#Instances
Authentication	1	2	1	1	1	1	1	1	1	1	1	1	12	13
Authorization	1	1	1	1	1	1	1	2	1	1	1	1	12	13
Chat	0	0	0	0	1	0	0	0	0	0	0	0	1	1
Comment	1	0	1	0	0	0	0	0	1	0	1	1	5	5
Event	0	0	0	0	0	0	1	1	0	1	1	0	4	4
Follow	0	0	0	0	0	0	0	0	0	0	0	1	1	1
Geolocation	0	0	0	0	0	0	0	1	1	0	1	1	4	4
Group	1	0	0	0	2	1	3	1	0	1	1	0	7	10
Label	0	0	0	0	0	0	0	0	1	0	1	1	3	3
Match	0	0	0	0	0	0	0	0	1	0	0	0	1	1
Passkey	0	0	0	0	0	0	2	0	0	0	0	0	1	2
Property	5	3	3	3	2	2	4	3	2	2	2	1	12	32
Ranking	0	0	0	0	0	1	0	0	0	0	0	0	1	1
Rating	1	0	4	1	1	0	0	0	0	0	0	1	5	8
Schedule	0	0	0	0	0	0	0	0	1	0	0	0	1	1
Scoring	0	0	0	0	0	1	2	0	0	0	0	2	3	5
Task	0	1	0	0	0	0	1	0	0	0	0	0	2	2
Transfer	0	1	0	0	0	0	0	0	0	1	0	0	2	2
# Concept Types	6	5	5	4	6	6	8	6	8	6	8	9		
# Concept Instances	10	8	10	6	8	7	15	9	9	7	9	10		

# Implementing Non-trivial Behavior

# Welcome to **ChoreStar**

## Parent Login

Username \*

Password \*

Sign In

## Child Login

Username \*

Password \*

Sign In

## Parent Registration

Username \*

Password \*

Retype Password \*

ChoreStar (clone)

# Welcome to CHORESTAR

## Parent Login

Username \*

Password \*

Sign In

## Child Login

Username \*

Password \*

Sign In

## Parent Registration

Username \*

Password \*

Retype Password \*

Register User

ChoreStar (clone)



# Implementing Non-trivial Behavior

```
{ ...  
  "usedConcepts": { ...,  
    "parentauthentication": { "concept": "Authentication" },  
    "childauthentication": { "concept": "Authentication" }  
  }, ...  
}
```

dvconfig.json

```
<dv.component name="parent-home"> ...  
  <chorestar.parent-navbar /> ...  
  <dv.tx>  
    <h2>Add a New Child</h2>  
    <dv.gen-id />  
    <dv.status savedText="Child created" />  
    <parentauthentication.authenticate  
      user=chorestar.parent-navbar.user hidden=true />  
    <childauthentication.register-user  
      id=dv.gen-id.id signIn=false  
      showOptionToSubmit=false /> ...  
    <dv.button>Create Child</dv.button>  
  </dv.tx>  
</dv.component>
```

parent-home.html

# Implementing Non-trivial Behavior

auth x2

```
{ ...  
  "usedConcepts": { ...,  
    "parentauthentication": { "concept": "Authentication" },  
    "childauthentication": { "concept": "Authentication" }  
  }, ...  
}
```

dvconfig.json

```
<dv.component name="parent-home"> ...  
  <chorestar.parent-navbar /> ...  
  <dv.tx>  
    <h2>Add a New Child</h2>  
    <dv.gen-id />  
    <dv.status savedText="Child created" />  
    <parentauthentication.authenticate  
      user=chorestar.parent-navbar.user hidden=true />  
    <childauthentication.register-user  
      id=dv.gen-id.id signIn=false  
      showOptionToSubmit=false /> ...  
    <dv.button>Create Child</dv.button>  
  </dv.tx>  
</dv.component>
```

parent-home.html

# Implementing Non-trivial Behavior

```
{ ...  
  "usedConcepts": { ...,  
    "parentauthentication": { "concept": "Authentication" },  
    "childauthentication": { "concept": "Authentication" }  
  }, ...  
}
```

dvconfig.json

```
<dv.component name="parent-home"> ...  
  <chorestar.parent-navbar /> ...  
  <dv.tx>  
    <h2>Add a New Child</h2>  
    <dv.gen-id />  
    <dv.status savedText="Child created" />  
    <parentauthentication.authenticate  
      user=chorestar.parent-navbar.user hidden=true />  
    <childauthentication.register-user  
      id=dv.gen-id.id signIn=false  
      showOptionToSubmit=false /> ...  
    <dv.button>Create Child</dv.button>  
  </dv.tx>  
</dv.component>
```

parent-home.html

register  
child

# Implementing Non-trivial Behavior

```
{ ...  
  "usedConcepts": { ...,  
    "parentauthentication": { "concept": "Authentication" },  
    "childauthentication": { "concept": "Authentication" }  
  }, ...  
}
```

dvconfig.json

auth  
parent

```
<dv.component name="parent-home"> ...  
  <chorestar.parent-navbar /> ...  
  <dv.tx>  
    <h2>Add a New Child</h2>  
    <dv.gen-id />  
    <dv.status savedText="Child created" />  
    <parentauthentication.authenticate  
      user=chorestar.parent-navbar.user hidden=true />  
    <childauthentication.register-user  
      id=dv.gen-id.id signIn=false  
      showOptionToSubmit=false /> ...  
    <dv.button>Create Child</dv.button>  
  </dv.tx>  
</dv.component>
```

parent-home.html

logged  
in parent

# Implementing Non-trivial Behavior

```
{ ...  
  "usedConcepts": { ...,  
    "parentauthentication": { "concept": "Authentication" },  
    "childauthentication": { "concept": "Authentication" }  
  }, ...  
}
```










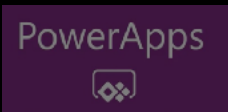


dvconfig.json

```
<dv.component name="parent-home"> ...  
  <chorestar parent-navbar /> ...  
  <dv.tx> synchronize  
    <h2>Add a New Child</h2>  
    <dv.gen-id />  
    <dv.status savedText="Child created" />  
    <parentauthentication.authenticate  
      user=chorestar.parent-navbar.user hidden=true />  
    <childauthentication.register-user  
      id=dv.gen-id.id signIn=false  
      showOptionToSubmit=false /> ...  
    <dv.button>Create Child</dv.button>  
  </dv.tx>  
</dv.component>
```










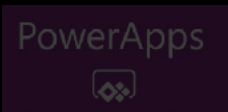


parent-home.html

# Conclusion

# Web App Development














- ▶ **General Purpose Tools**     
  - ▶ flexible, but libraries often just client- or server-side
  - ▶ full-stack components typically require server code  
- ▶ **Content Management Systems**  
  - ▶ plug-ins are full-stack, but composition is ad-hoc
- ▶ **Low-code Tools**   
  - ▶ high level, but you have to write most end-user behavior

# Web App Development













- ▶ **General Purpose Tools**     
  - ▶ flexible, but libraries often just client- or server-side
  - ▶ full-stack components typically require server code  
- ▶ **Content Management Systems**  
  - ▶ plug-ins are full-stack, but composition is ad-hoc
- ▶ **Low-code Tools**   
  - ▶ high level, but you have to write most end-user behavior



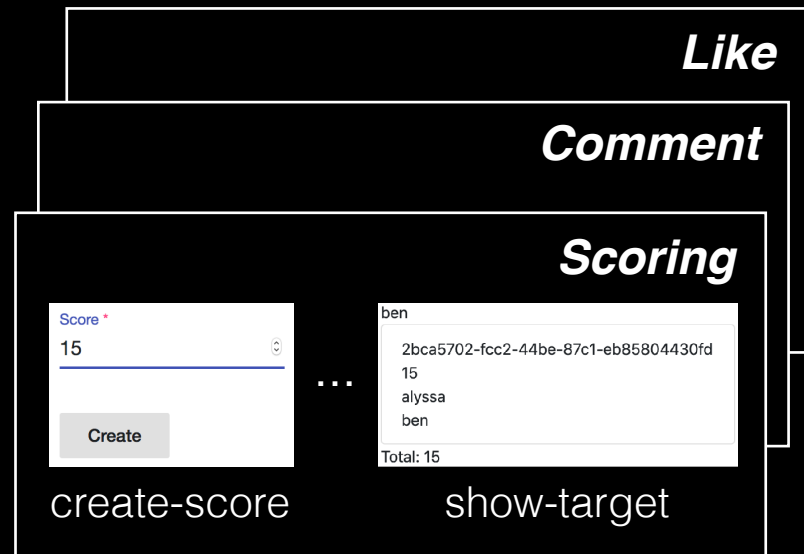
# Web App Development

- ▶ **General Purpose Tools**     
  - ▶ flexible, but libraries often just client- or server-side
  - ▶ full-stack components typically require server code  
- ▶ **Content Management Systems**  
  - ▶ plug-ins are full-stack, but composition is ad-hoc
- ▶ **Low-code Tools**    
  - ▶ high level, but you have to write most end-user behavior

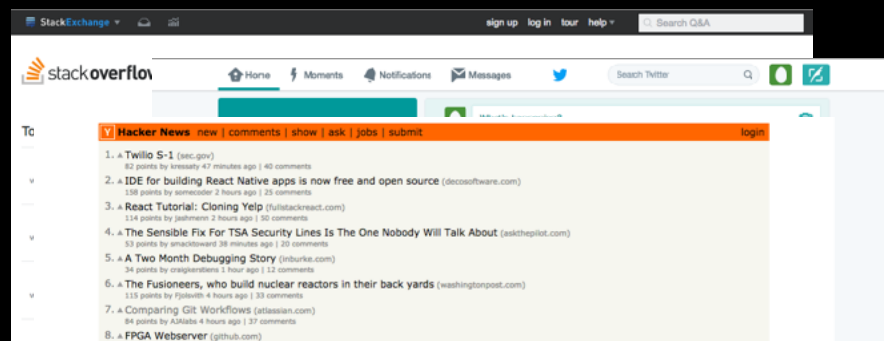
# Web App Development

- ▶ **General Purpose Tools**     
  - ▶ flexible, but libraries often just client- or server-side
  - ▶ full-stack components typically require server code  
- ▶ **Content Management Systems**  
  - ▶ plug-ins are full-stack, but composition is ad-hoc
- ▶ **Low-code Tools**   
  - ▶ high level, but you have to write most end-user behavior

# The Déjà Vu Platform



```
<dv.component name="submit-post">
  <sn.navbar /> ...
  <dv.tx>
    <dv.gen-id /> ...
    <post.create-object id=dv.gen-id.id
      initialValue={ author: sn.navbar.user.username }
      showExclude=["author"] buttonLabel="Submit"
      newObjectSavedText="Post submitted" />
    <scoreposts.create-score targetId=dv.gen-id.id
      sourceId=sn.navbar.user.username
      value=0 hidden=true /> ...
  </dv.tx> ...
</dv.component>
```



## catalog of concepts

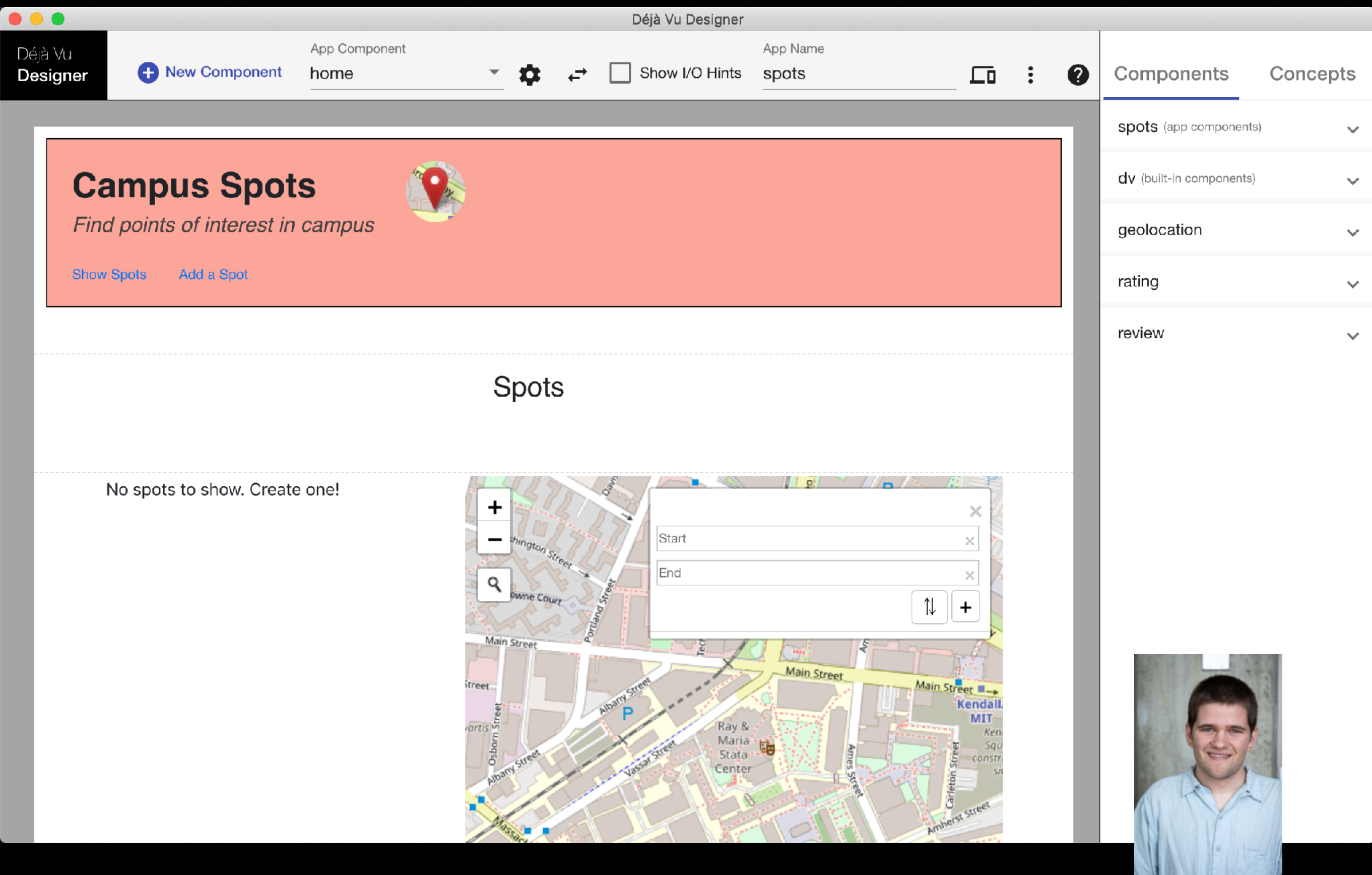
- ▶ new modularity: concepts
- ▶ capture and reuse concept design & implementation

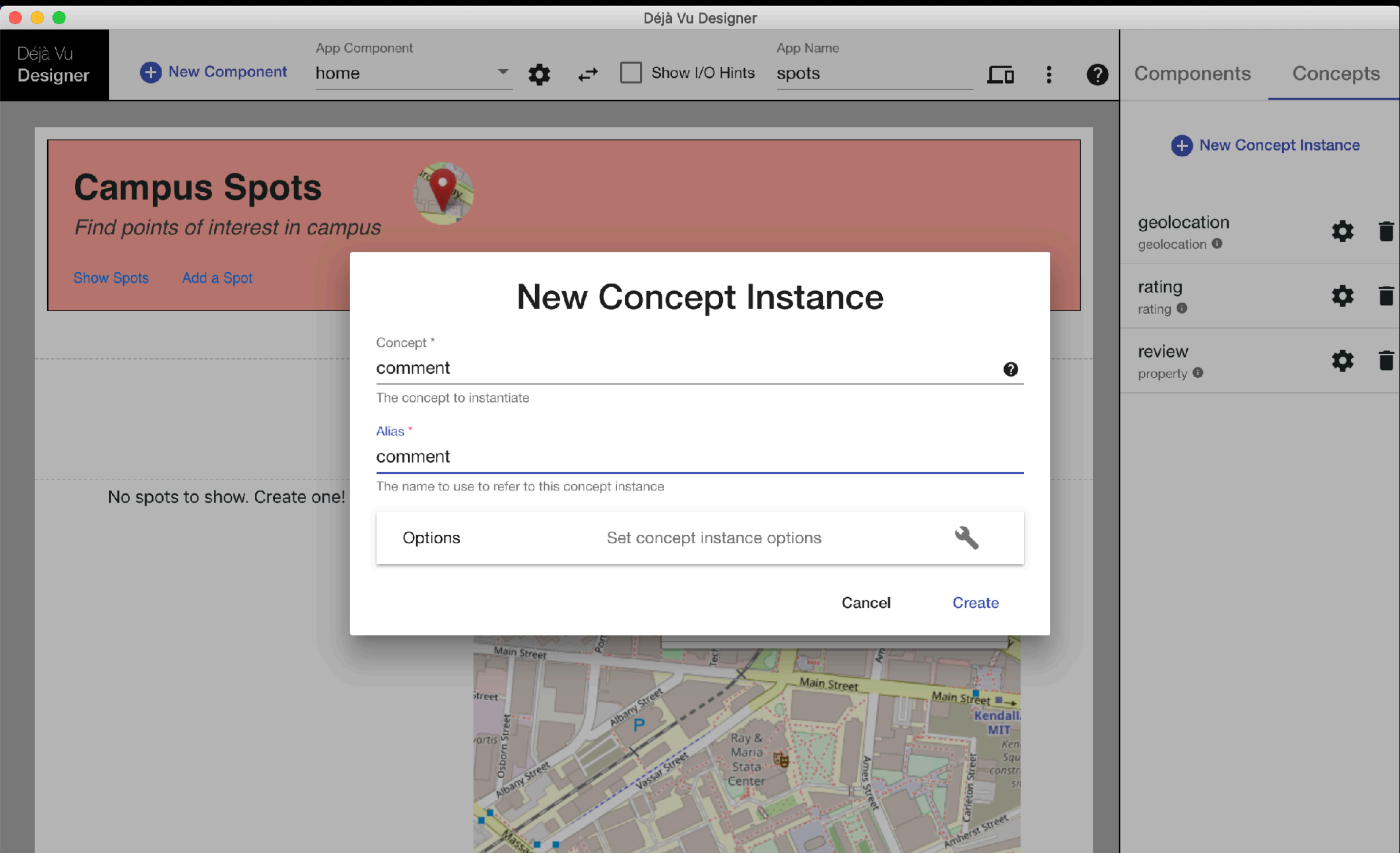
## concept composition mechanism

- ▶ in HTML, no JS or server-side code
- ▶ join concept actions for aggregate behavior

## web apps

- ▶ complex behavior
- ▶ rich graphical user interfaces





Déjà Vu Designer

New Component

App Component

create-spot

⚙️

↔️

☐ Show I/O Hints

App Name

spots

🖥️

⋮

?

Components

Concepts

+ New Concept Instance

geolocation

geolocation ⓘ

⚙️

🗑️

rating

rating ⓘ

⚙️

🗑️

review

property ⓘ

⚙️

🗑️

Review \*

great spot!

↔️

🗑️

Create Spot



# Try Déjà Vu:

[deja-vu-platform.com](http://deja-vu-platform.com)



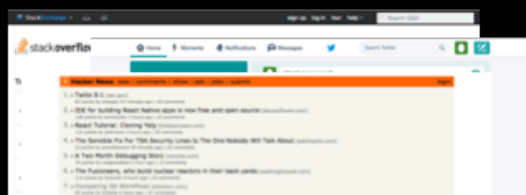
## *catalog of concepts*

- ▶ new modularity: concepts
- ▶ capture and reuse concept design & implementation

```
<dv.component name="submit-post">
  <sn.navbar /> ...
  <dv.tx>
    <dv.gen-id /> ...
    <post.create-object id=dv.gen-id.id
      initialValue={ author: sn.navbar.user.username }
      showExclude=["author"] buttonLabel="Submit"
      newObjectSavedText="Post submitted" />
    <scoreposts.create-score targetId=dv.gen-id.id
      sourceId=sn.navbar.user.username
      value=0 hidden=true /> ...
  </dv.tx> ...
</dv.component>
```

## *concept composition mechanism*

- ▶ in HTML, no JS or server-side code
- ▶ join concept actions for aggregate behavior



## *web apps*

- ▶ complex behavior
- ▶ rich graphical user interfaces