ECS 160

Software Engineering

Instructor: Tapti Palit

Teaching Assistant: Gabe Bai

Agenda

- Instructor and TA introductions
- Syllabus and course policies
- Modern software engineering overview
- OO revision + design patterns

Instructor details

- Name: Tapti Palit
- Background: Software security, program analysis, systems
- Contact details
  - Email: tpalit@ucdavis.edu
  - Email subject must contain "**W26 ECS 160**" followed by the actual subject
  - Office hours: Wednesday 2 PM – 4 PM (Zoom* / TBD)

TA details

- Name: Gabe Bai
- Background: PhD student in the CS department
- Email: gabbai@ucdavis.edu
- Office hours: TBD

Course information

- Class timings
  - Lectures - MWF 9 AM – 9:50 AM in Veihmeyer Hall 212
  - Discussion – 12:10 PM - 1:00 PM W Wellman Hall 234
- Lectures and discussion will be recorded and uploaded on Canvas
- Course page: https://github.com/davsec-teaching/W26_ECS_160
- Piazza link: https://piazza.com/ucdavis/winter2026/ecs160winter2026/home
- Textbooks:
  - Required: None
  - Recommended: check the course website

Course components

- 1 midterm, 1 final
- 3-4 HWs done individually
- 5-6 reading/video discussion on Perusall*
- 5 **in-class** quizes

Component	Weightage
Midterm	30
Final	35
HW	20
In-class quiz	10
Reading/video reflections	5

Grade cutoffs

- Standard UC Davis grade cutoffs

Percentage	Grade
93%	A
90%	A-
87%	B+
83%	B
80%	B-
77%	C+
73%	C
70%	C-
67%	D+
63%	D
60%	D-

- Might be curved on top of these cutoffs *per instructor's discretion*
- No extra credit assignments

Academic integrity and AI policies

- Inter-group collaboration and discussion is permitted
  - Copying code is not permitted
- Using AI to understand concepts, exceptions, or compilation errors is permitted
- Using AI as a “search engine” is allowed
- Using any sort of AI-integration in IDEs is not permitted (Copilot, Cursor, etc)
- Turning in AI-generated code **without proper attribution** is not permitted

Logistics

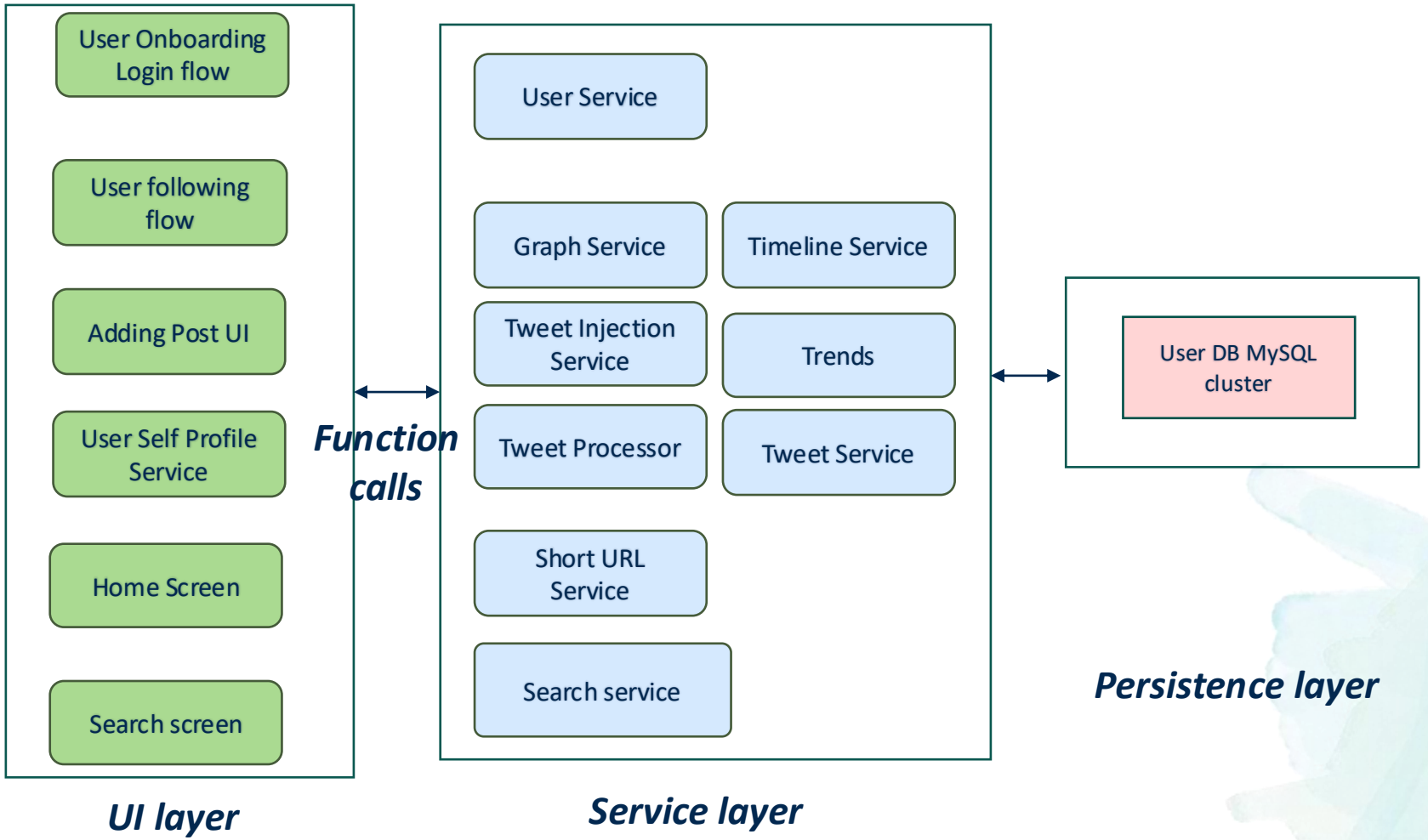
- Enrollment cap will likely not increase
- Limited by room size
- … ***Questions?***

Modern software engineering

Modern software engineering

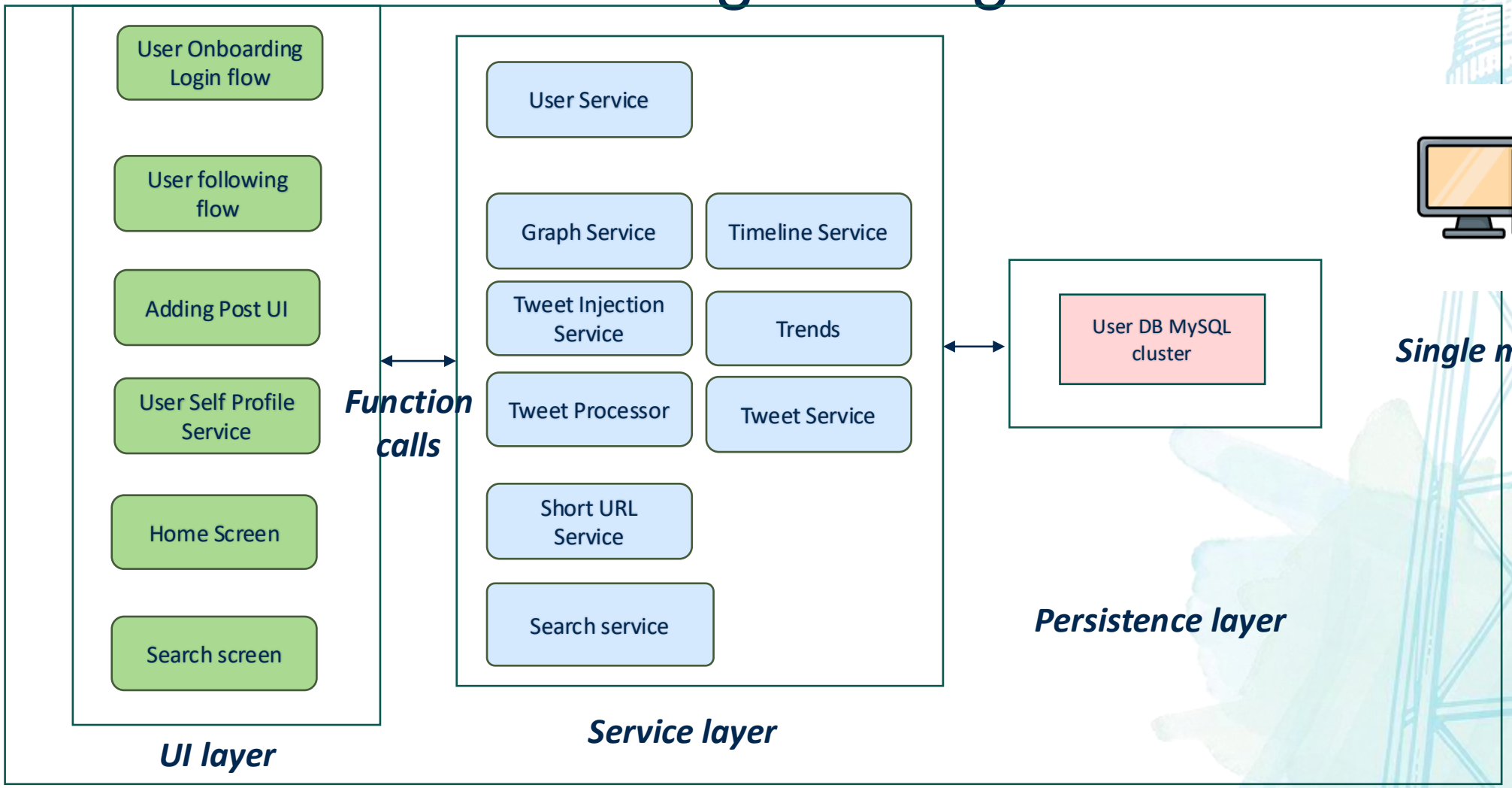
- Is distributed
- Is systems engineering
- Is rapidly evolving
- LLMs and automation raise the bar for humans
  - ***… ECS 160 takes these factors into account***

Traditional software engineering was monolithic

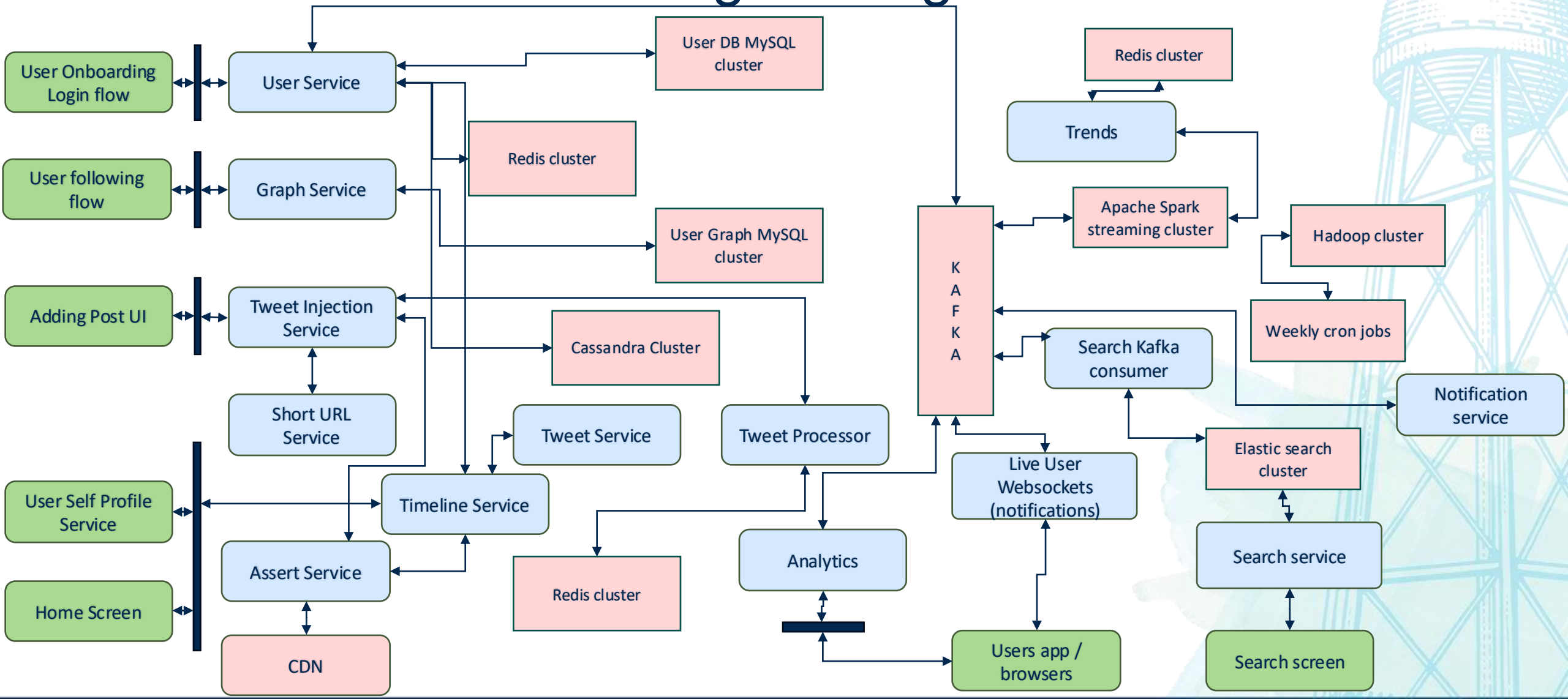


Social Media Platform

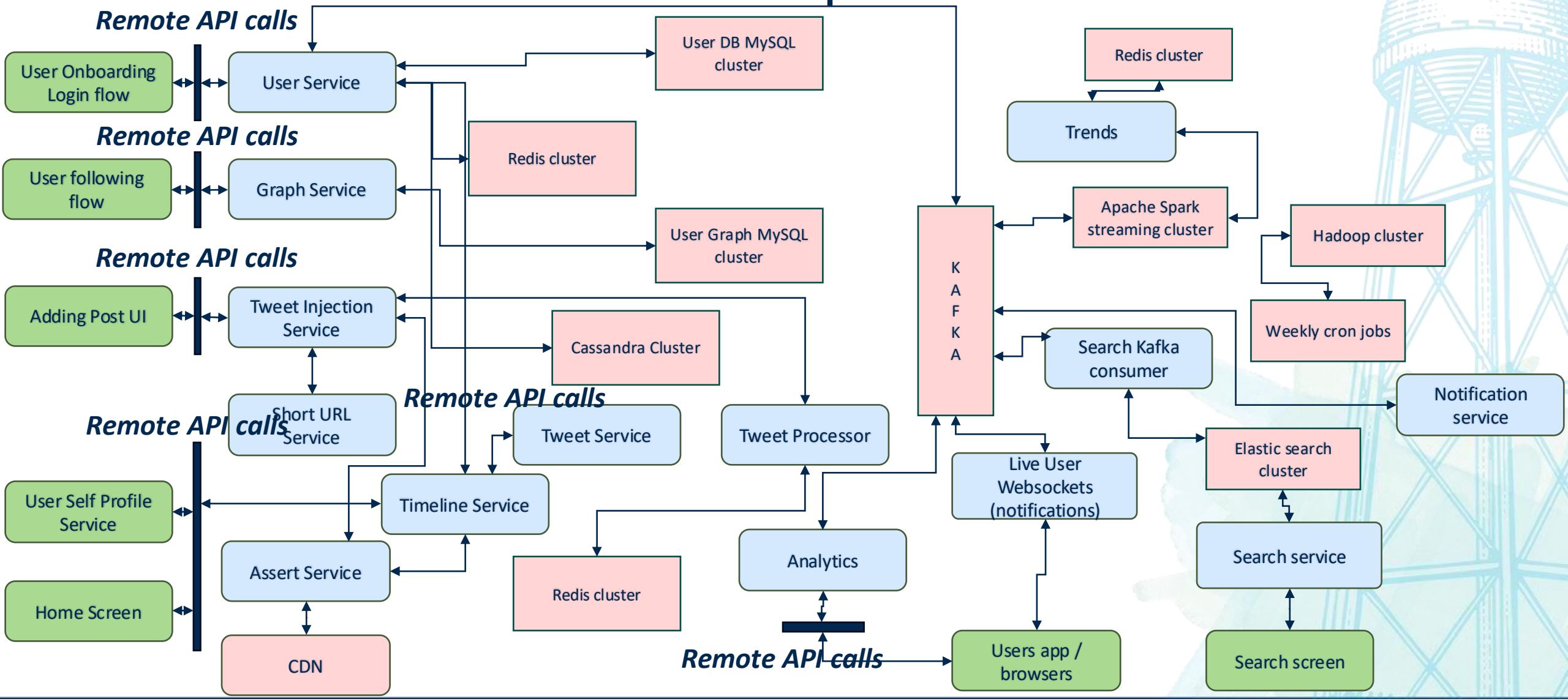
Traditional software engineering was monolithic



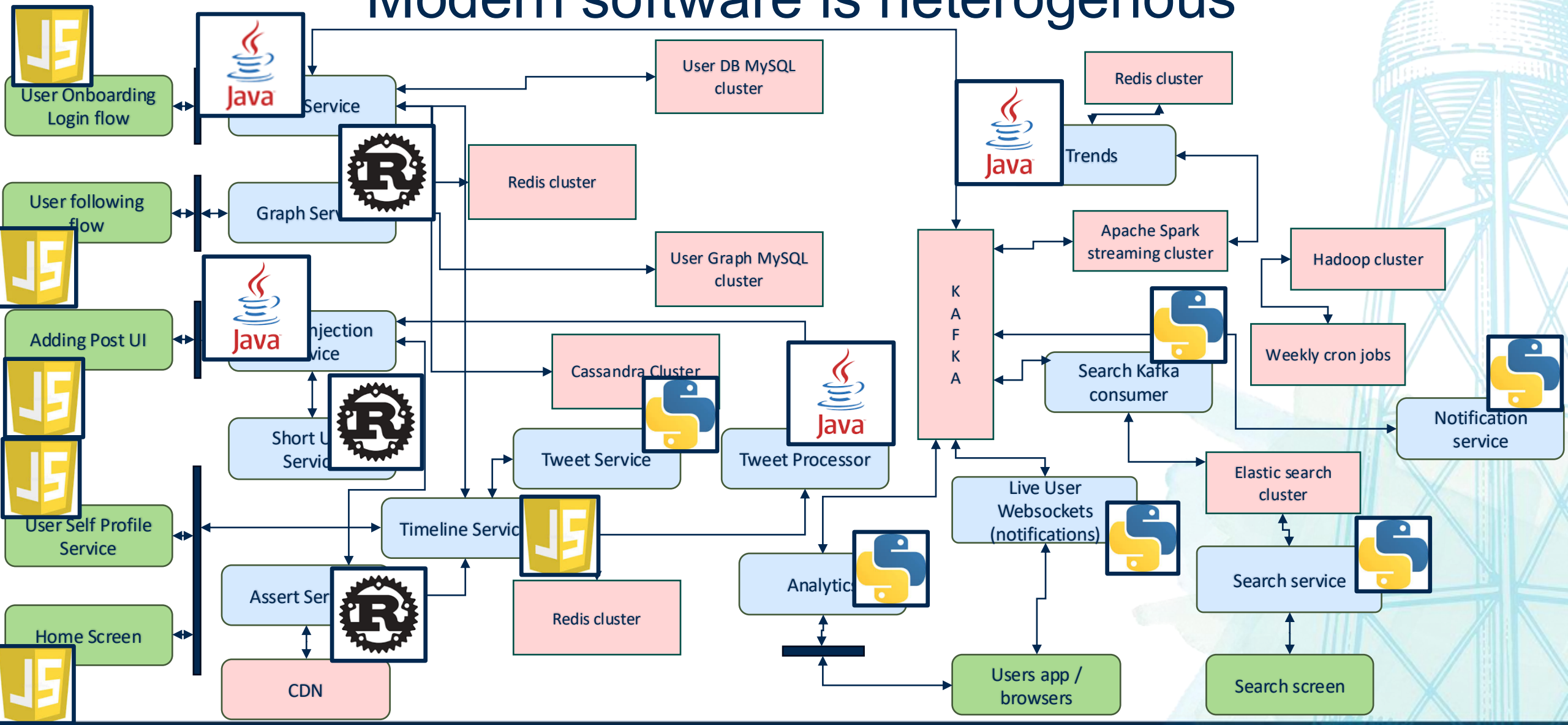
Modern software engineering is non-monolithic



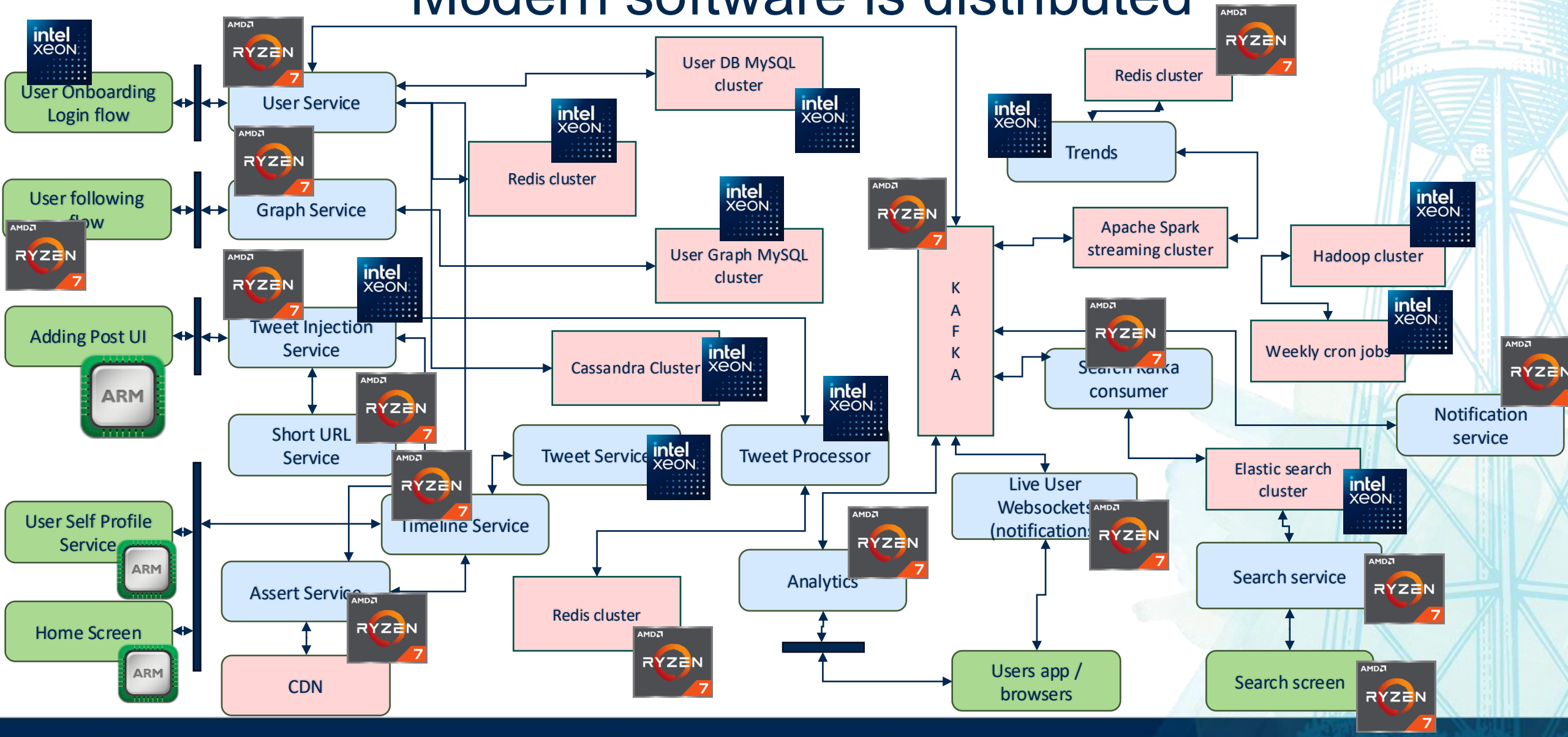
Modern software is decomposed into microservices



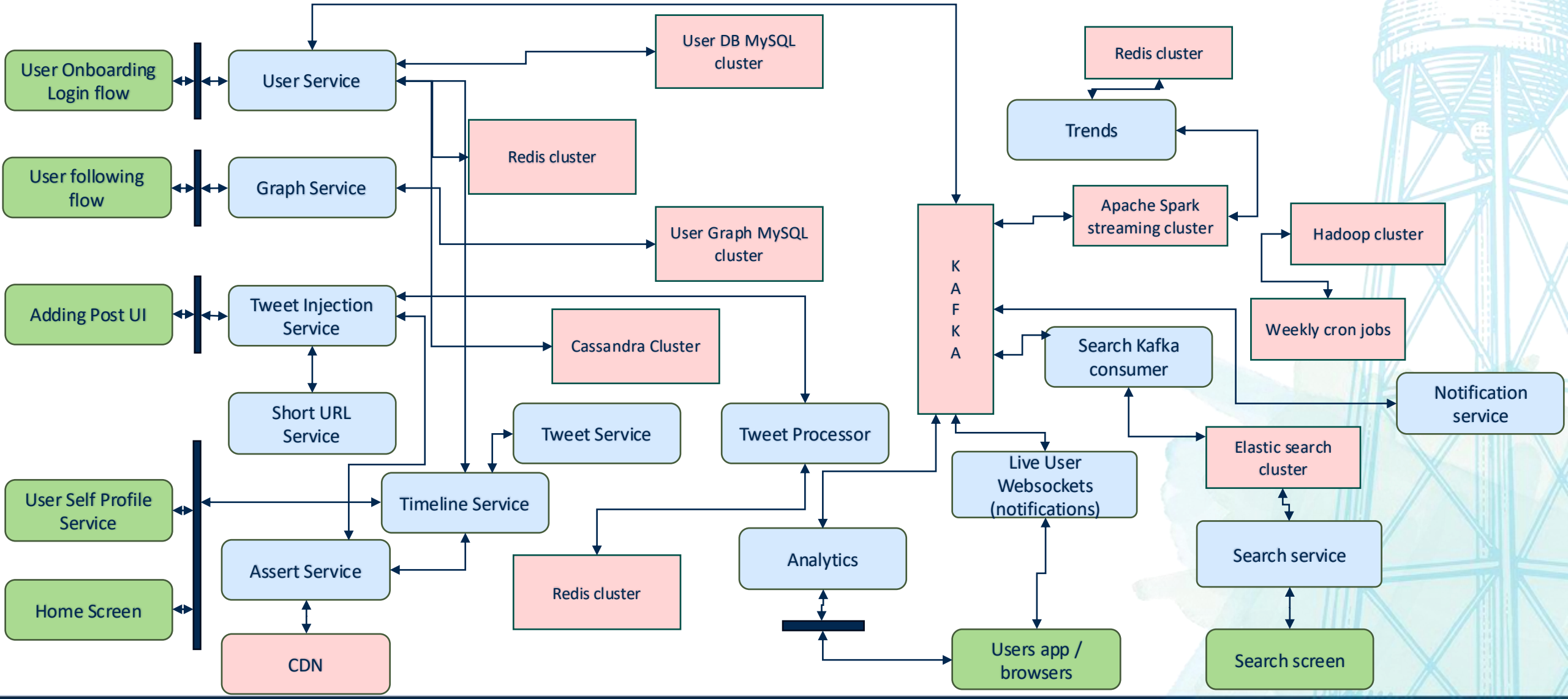
Modern software is heterogenous



Modern software is distributed



Data is decentralized



Challenges

- How to decompose monolithic software into microservices?
- How to handle distributed state?
- What are ideal communication patterns?
- How to orchestrate different machines on the cluster?
- How to handle failure?

ECS 160 modules

- Building components
- Composing systems
- Validating components

Syllabus overview

Module	Topic	Lectures	Weightage
Building components	- Design patterns - Metaprogramming using reflection and annotations	10	36%
Composing systems	- Microservices - Message queue, pub/sub - Event-driven design using Kafka - Orchestration using Kubernetes - Serverless and Function-as-a-Service	12	44%
Validating components	- Property testing - Fuzz testing and sanitizers - Symbolic execution - Abstract interpretation	6	18%
		28	

HW assignments

- Apache Cassandra persistence layer using Java annotation and reflection
- Social media data analysis using Spring Boot micro-services and Kafka
- Deployment of microservices on Google Cloud and Kubernetes
- [Tentative] Fuzz testing a provided library with custom mutators

Who should take this course?

- Experience with “app development”
- Knowledge of networking fundamentals – ports, IP addresses, some socket programming experience
- Knowledge of OS fundamentals – userspace vs kernelspace, processes vs. threads, etc.
- Experience with Java is useful, but not necessary
- We will try to cover some background during the discussion

Questions

- Any questions?