Introduction to Software Architecture

17-313 Spring 2024

Foundations of Software Engineering

https://cmu-313.github.io

Michael Hilton and Eduardo Feo Flushing

Administrivia

- “Regrade requests can be submitted via Gradescope. The regrade period is open for one week after grades have been released for a particular assignment.”

Smoking Section

- Last **two** full rows

Learning Goals

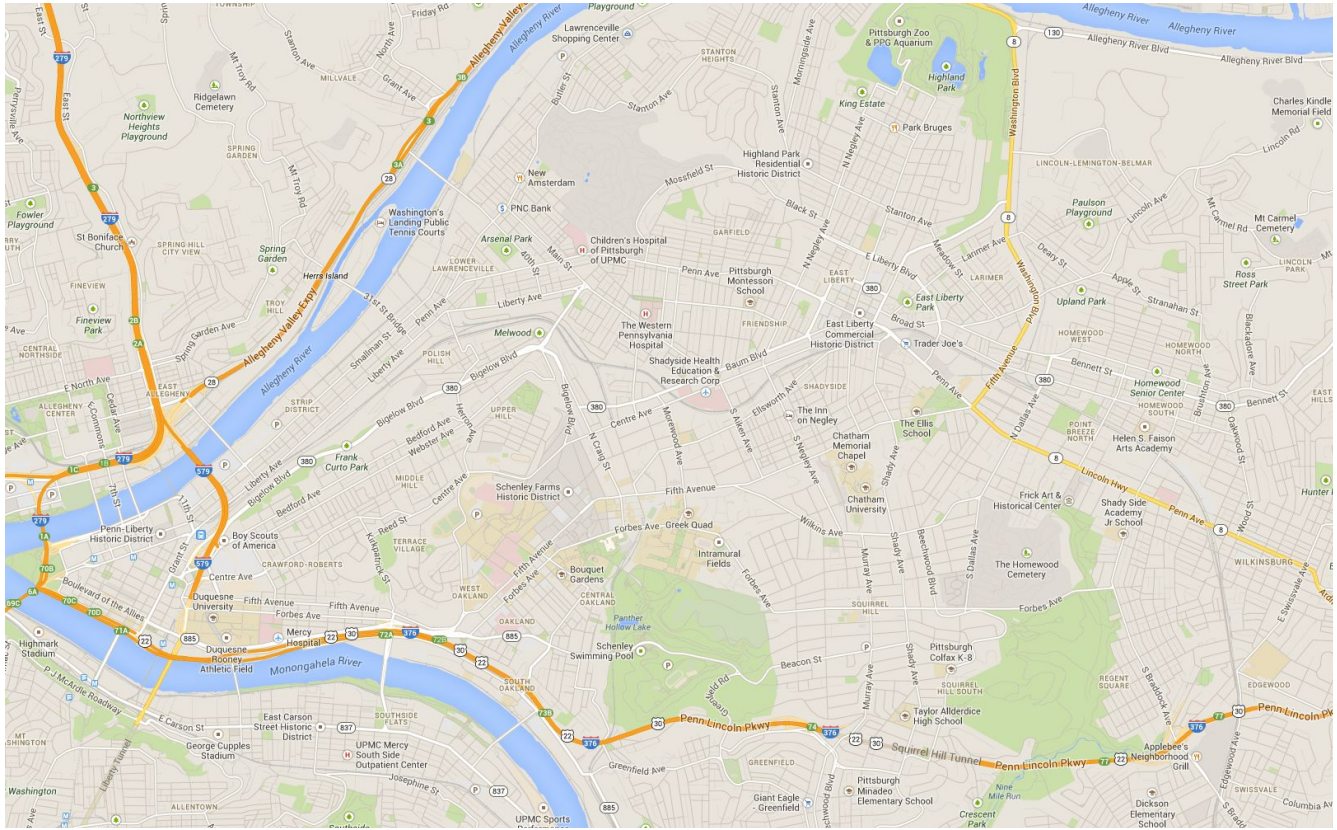
- Understand the abstraction level of architectural reasoning
- Appreciate how software systems can be viewed at different abstraction levels
- Distinguish software architecture from (object-oriented) software design
- Explain the importance of architectural decisions
- Integrate architectural decisions into the software development process
- Document architectures clearly, without ambiguity

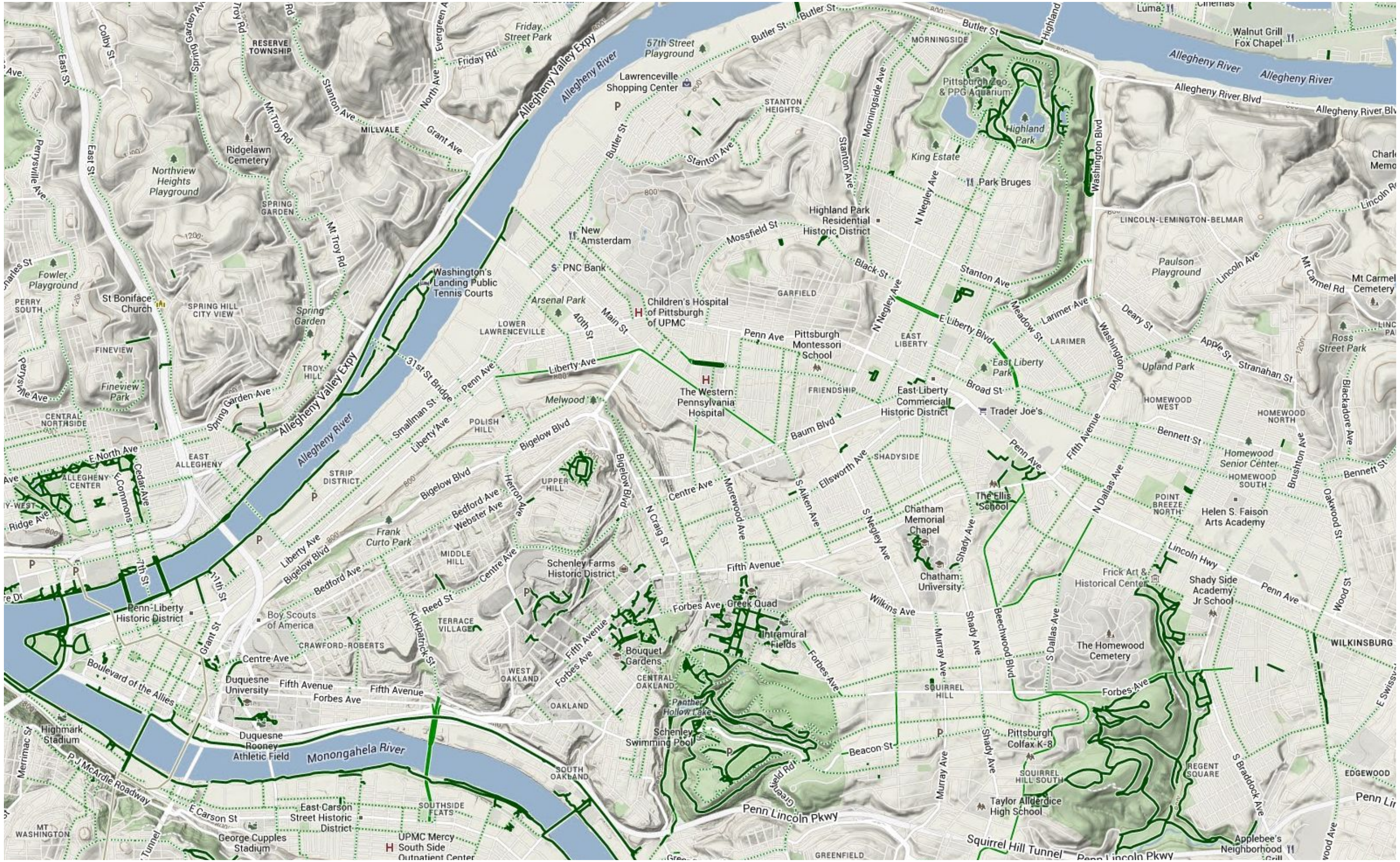
Outline

- Views and Abstraction
- Case Study: Autonomous Vehicles
- Software Architecture
  - Definitions, Importance
  - Software Design vs. Software Architecture
- Architecting software
  - Integrating Architectural Decisions into the SW Development Process
  - Common Software Architectures
  - Documentation

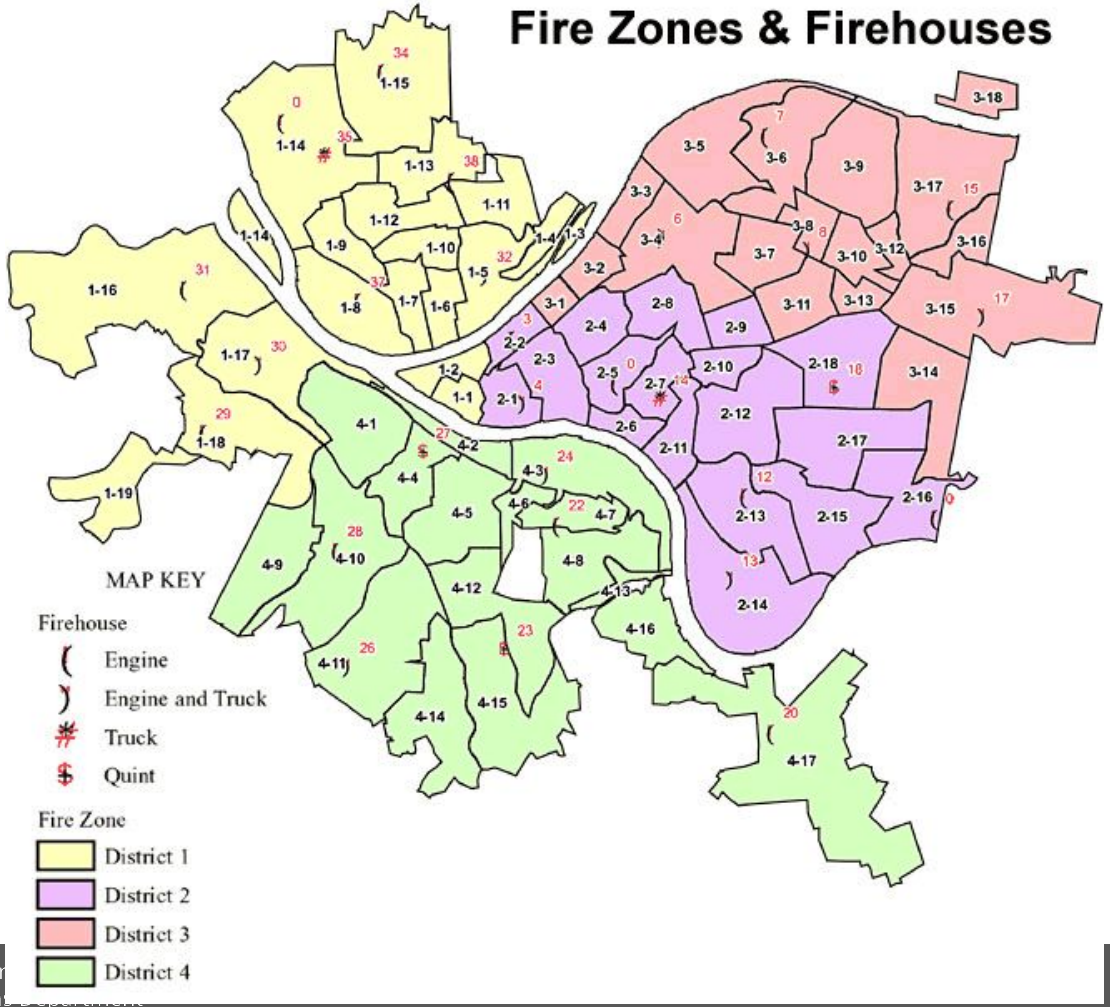
Outline

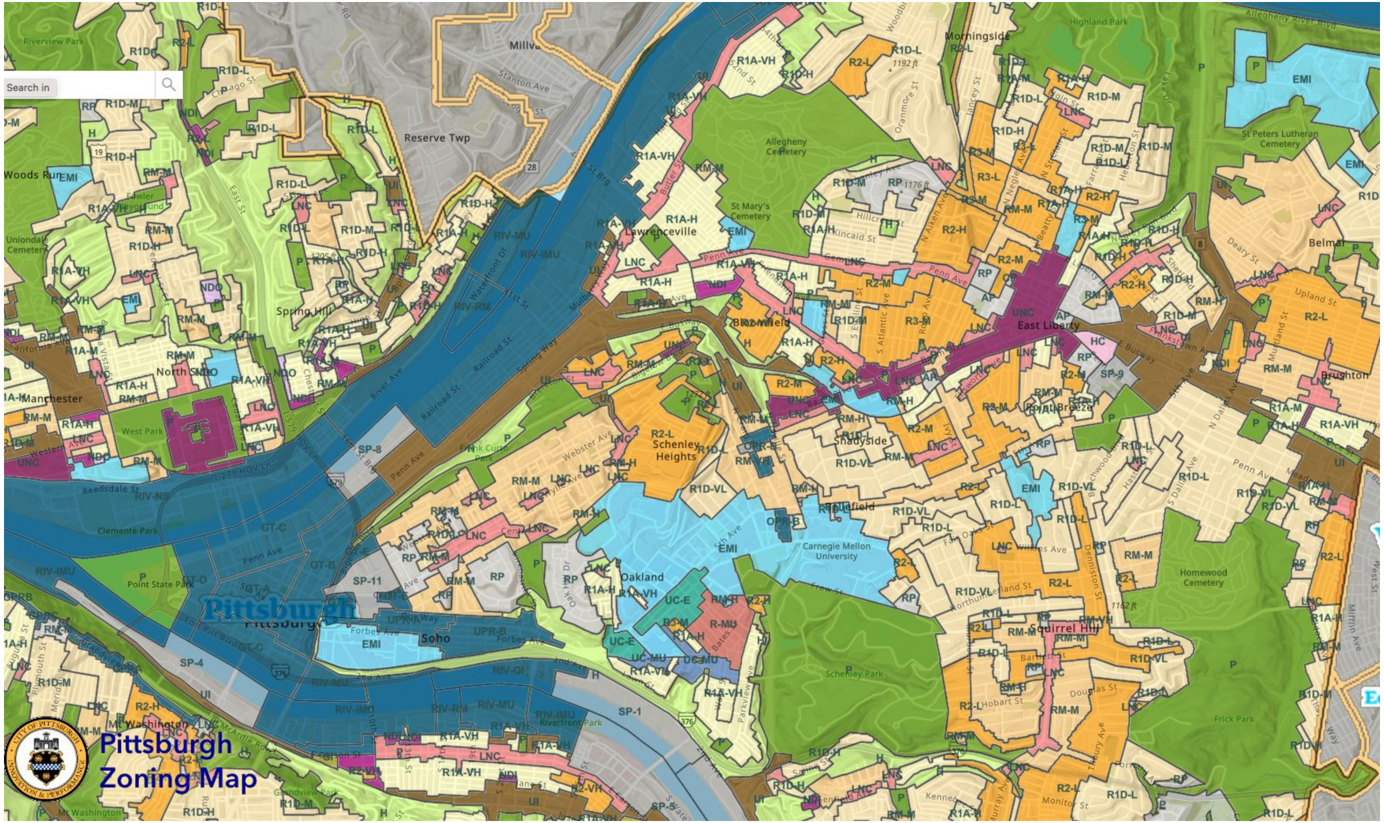
- Views and Abstraction
- Case Study: Autonomous Vehicles
- Software Architecture
  - Definitions, Importance
  - Software Design vs. Software Architecture
- Architecting software
  - Integrating Architectural Decisions into the SW Development Process
  - Common Software Architectures
  - Documentation

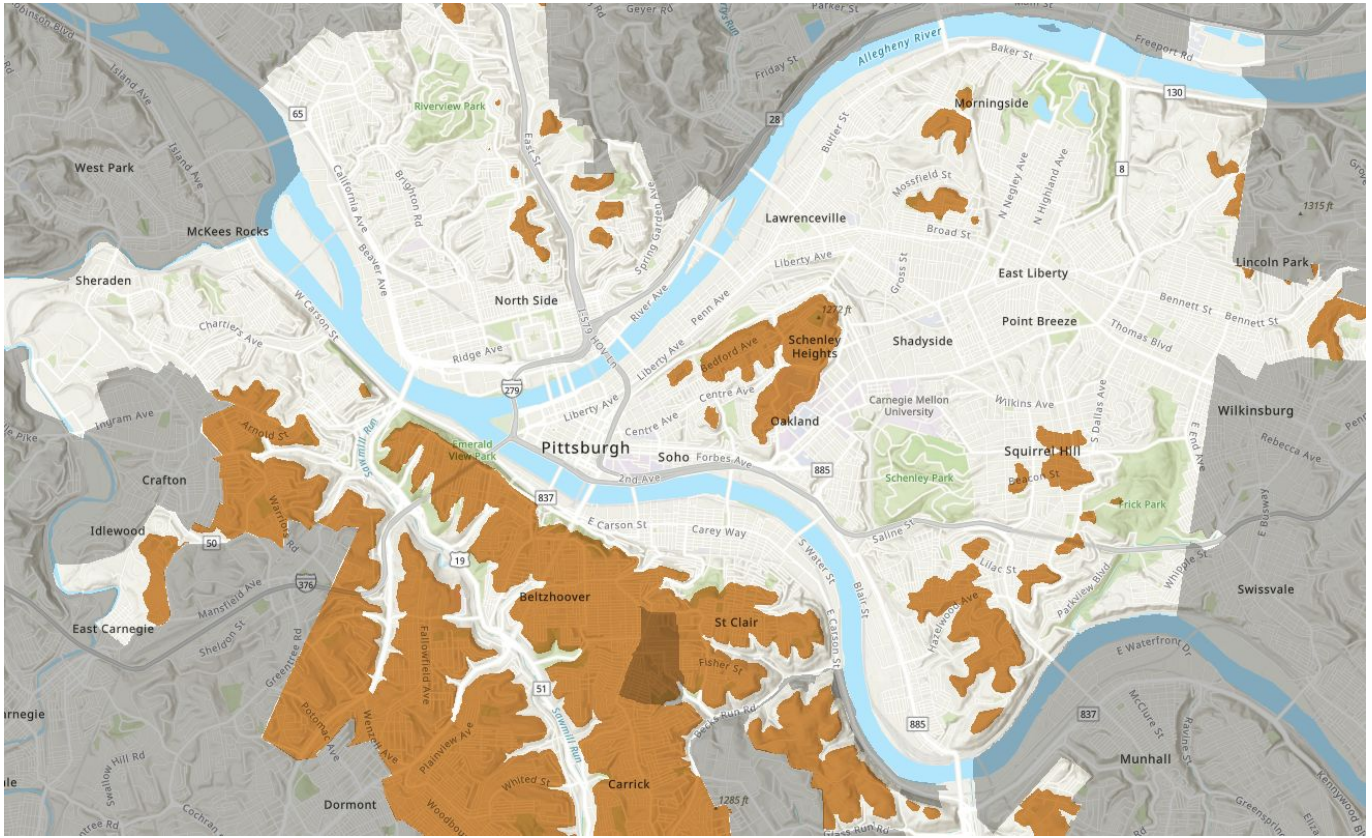

Allegheny River
Allegheny River Blvd
Pittsburgh Zoo & PPG Aquarium
Highland Park
Charles Kindle Memorial Field
Lawrenceville Shopping Center
Northview Heights Playground
Ridgelawn Cemetery
King Estate
Park Bruges
Highland Park Residential Historic District
Washington Blvd
LINCOLN-LEMINGTON-BELMAR
Fowler Playground
St Boniface Church
Washington's Landing Public Tennis Courts
New Amsterdam
PNC Bank
Arsenal Park
Children's Hospital of Pittsburgh of UPMC
Paulson Playground
Mt Carmel Cemetery
Spring Garden
Herrs Island
Pittsburgh Montessori School
Upland Park
Ross Street Park
Fineview Park
Allegheny Valley Expy
Liberty Ave
Penn Ave
Melwood
The Western Pennsylvania Hospital
East Liberty Park
East Liberty Commercial Historic District
Trader Joe's
Shadyside Health Education & Research Corp
Baum Blvd
Bigelow Blvd
Fifth Avenue
Homewood Senior Center
Frank Curto Park
The Inn on Negley
The Ellis School
Chatham Memorial Chapel
Helen S. Faison Arts Academy
Schenley Farms Historic District
Chatham University
Frick Art & Historical Center
Shady Side Academy Jr School
Lincoln Hwy
Penn Ave
Penn-Liberty Historic District
Boy Scouts of America
Forbes Ave
Greek Quad
Intramural Fields
Bouquet Gardens
The Homewood Cemetery
Wilkins Ave
Boulevard of the Allies
Duquesne University
Fifth Avenue
Panther Hollow Lake
Mercy Hospital
Schenley Swimming Pool
Pittsburgh Colfax K-8
Highmark Stadium
Duquesne Rooney Athletic Field
Monongahela River
Beacon St
Taylor Allderdice High School
Penn-Lincoln Pkwy
Squirrel Hill Tunnel
Applebee's Neighborhood Grill
East Carson Street Historic District
George Cupples Stadium
UPMC Mercy South Side Outpatient Center
Greenfield Ave
Giant Eagle - Greenfield
Pittsburgh Miradeo Elementary School
Dickson Elementary School
UPMC Sports


Allegheny River
Allegheny Valley Expy
Monongahela River
Butler St
Lawrenceville Shopping Center
57th Street Playground
Friday Street Park
Friday Rd
RESERVE TOWNSHIP
MILLVALE
Ridgelawn Cemetery
Northview Heights Playground
SPRING GARDEN
Stanton Ave
Grant Ave
North Ave
Mt Troy Rd
MORNINGSIDE
STANTON HEIGHTS
Pittsburgh Zoo & PPG Aquarium
Highland Park
King Estate
Park Bruges
Highland Park Residential Historic District
Washington Blvd
LINCOLN-LEMINGTON-BELMAR
Paulson Playground
Lincoln Ave
Mt Carmel Cemetery
New Amsterdam
PNC Bank
Mossfield St
Black St
GARFIELD
Washington's Landing Public Tennis Courts
Arsenal Park
LOWER LAWRENCEVILLE
Children's Hospital of Pittsburgh of UPMC
Penn Ave
Pittsburgh Montessori School
EAST LIBERTY
E Liberty Blvd
LARIMER
Larimer Ave
Upland Park
Stranahan St
Ross Street Park
Fowler Playground
St Boniface Church
PERRY SOUTH
SPRING HILL CITY VIEW
Spring Garden
FINEVIEW
Fineview Park
TROY HILL
Liberty Ave
Melwood
The Western Pennsylvania Hospital
FRIENDSHIP
East Liberty Commercial Historic District
Trader Joe's
East Liberty Park
Broad St
HOMEWOOD WEST
HOMEWOOD NORTH
Bennett St
Homewood Senior Center
HOMEWOOD SOUTH
CENTRAL NORTHSIDE
E North Ave
EAST ALLEGHENY
ALLEGHENY CENTER
POLISH HILL
Bigelow Blvd
Baum Blvd
SHADYSIDE
Ellsworth Ave
Fifth Avenue
STRIP DISTRICT
UPPER HILL
Centre Ave
Chatham Memorial Chapel
The Ellis School
POINT BREEZE NORTH
Helen S. Faison Arts Academy
Frank Curto Park
MIDDLE HILL
Schenley Farms Historic District
Chatham University
Lincoln Hwy
Frick Art & Historical Center
Shady Side Academy Jr School
Penn-Liberty Historic District
Boy Scouts of America
TERRACE VILLAGE
CRAWFORD-ROBERTS
Forbes Ave
Greek Quad
Intramural Fields
Wilkins Ave
The Homewood Cemetery
WILKINSBURG
Duquesne University
Bouquet Gardens
WEST OAKLAND
CENTRAL OAKLAND
OAKLAND
SQUIRREL HILL
Panther Hollow Lake
Schenley Swimming Pool
Beacon St
Pittsburgh Colfax K-8
REGENT SQUARE
Highmark Stadium
Duquesne Rooney Athletic Field
SOUTH OAKLAND
SQUIRREL HILL SOUTH
Taylor Allderdice High School
EDGEWOOD
MT WASHINGTON
E Carson St
East Carson Street Historic District
SOUTHSIDE FLATS
George Cupples Stadium
UPMC Mercy South Side Outpatient Center
Penn Lincoln Pkwy
Squirrel Hill Tunnel
GREENFIELD
Applebee's Neighborhood Grill

Fire Zones & Firehouses




Pittsburgh
Zoning Map
Search in
Reserve Twp
Millvale
Lawrenceville
Allegheny Cemetery
St Mary's Cemetery
Morningside
Highland Park
East Liberty
Bloomfield
Shadyside
Schenley Heights
Oakland
Carnegie Mellon University
Schenley Park
Squirrel Hill
Point Breeze
Homewood Cemetery
Frick Park
Belmar
Brushton
Spring Hill
North Side
Manchester
West Park
Point State Park
Soho
Mt Washington
Woods Run
Riverview Park
Uniondale Cemetery
St Peters Lutheran Cemetery



Source: Pittsburgh Zoning Map (https://gis.pittsburghpa.gov/pghzoning/)

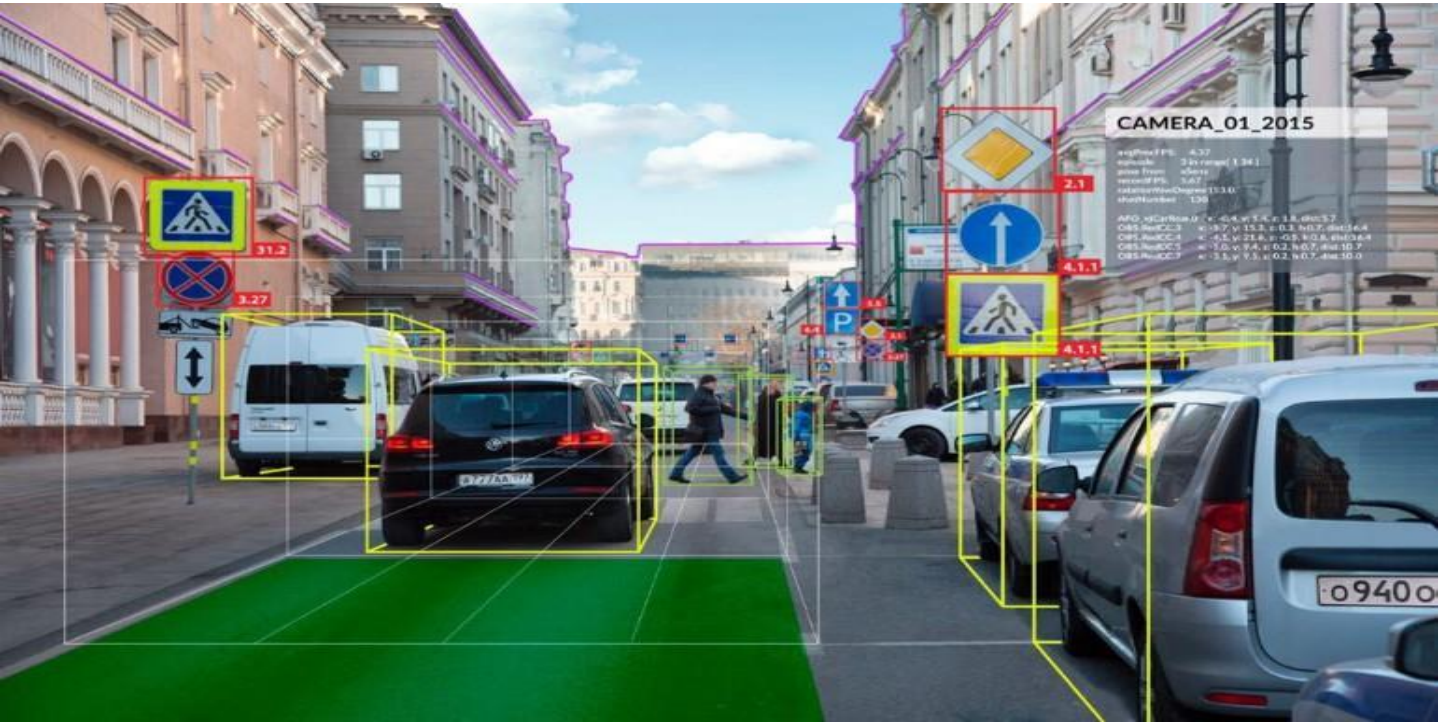
Abstracted views focus on conveying specific information

- They have a well-defined purpose
- Show only necessary information
- Abstract away unnecessary details
- Use legends/annotations to remove ambiguity
- Multiple views of the same object tell a larger story

Outline

- Views and Abstraction
- **Case Study: Autonomous Vehicles**
- Software Architecture
  - Definitions, Importance
  - Software Design vs. Software Architecture
- Architecting software
  - Integrating Architectural Decisions into the SW Development Process
  - Common Software Architectures
  - Documentation

Case Study: Autonomous Vehicle Software



Case Study: Apollo

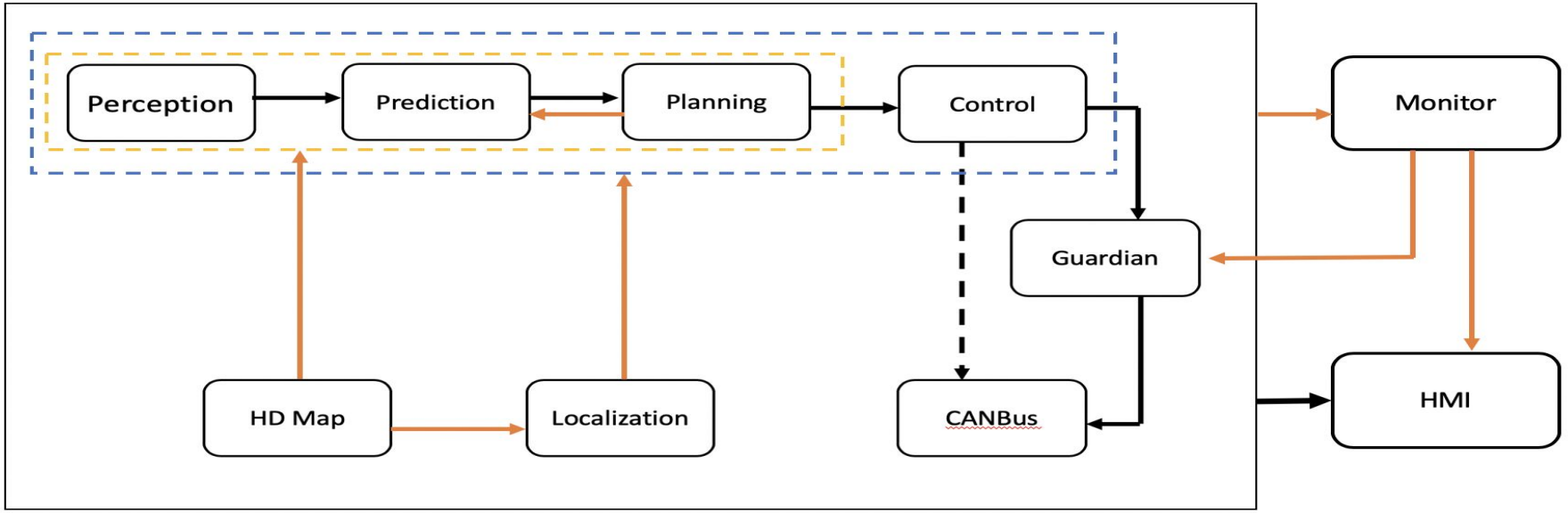
Check out the “side pass” feature from the video:
https://www.youtube.com/watch?v=BXNDUtNZdM4

- Discuss in teams of 4 what parts are associated with the **side pass feature**

Source: https://github.com/ApolloAuto/apollo

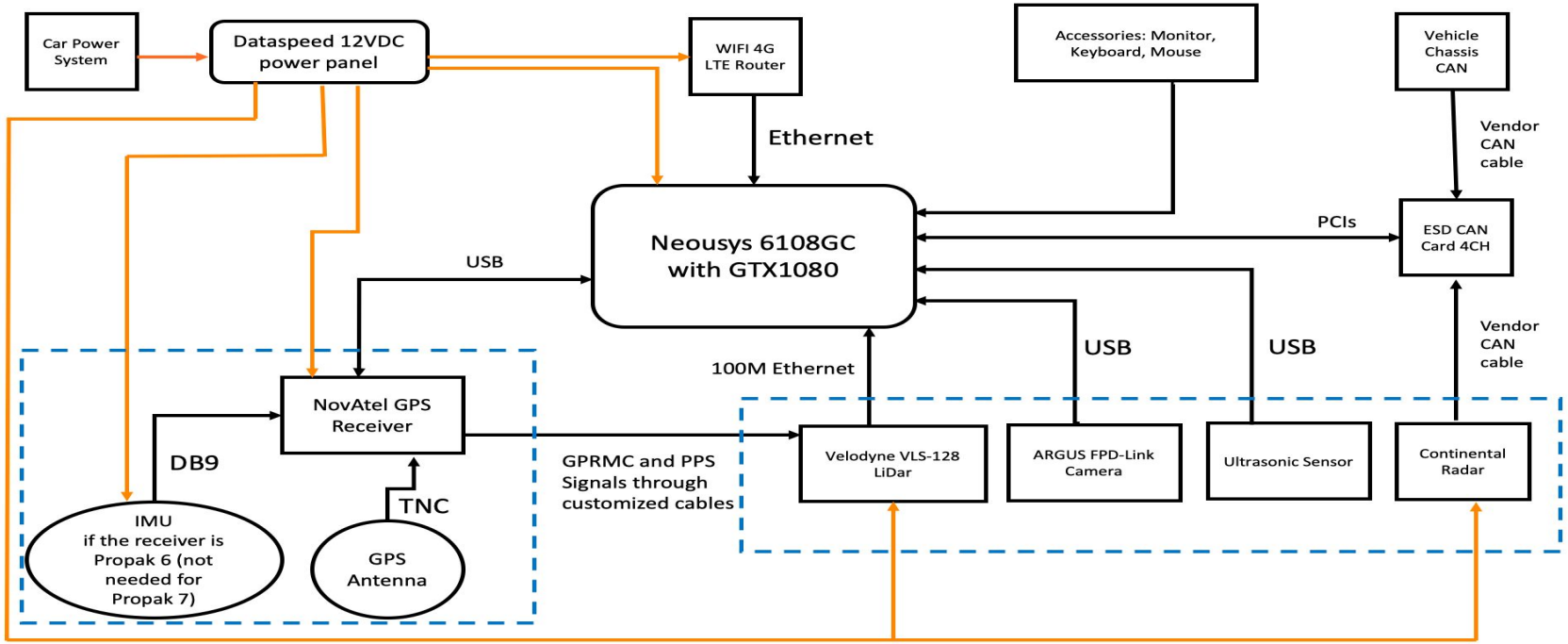
Doxygen:
https://hidetoshi-furukawa.github.io/apollo-doxygen/index.html

Apollo Software Architecture



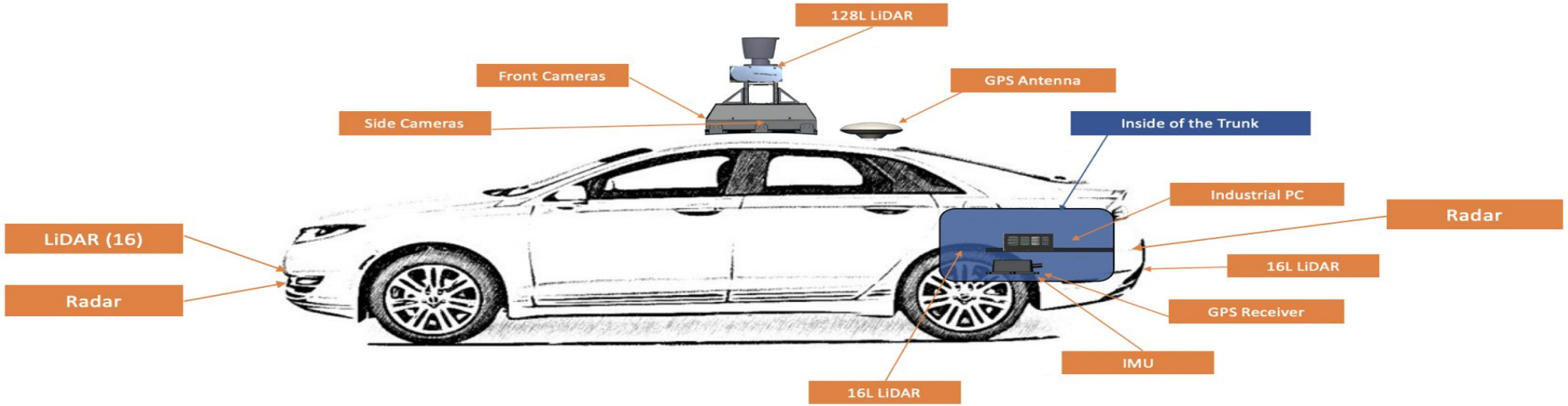
Source: https://github.com/ApolloAuto/apollo/blob/v6.0.0/docs/specs/Apollo_5.5_Software_Architecture.md

Apollo Hardware Architecture



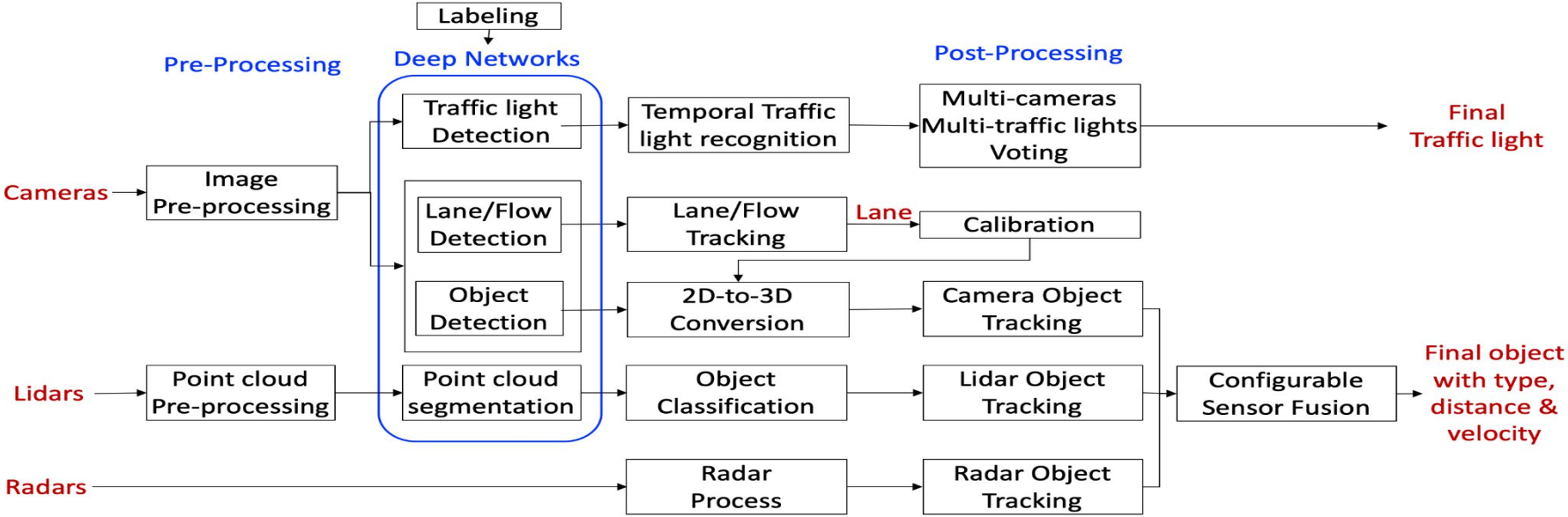
Source: https://github.com/ApolloAuto/apollo/blob/v6.0.0/README.md

Apollo Hardware/Vehicle Overview

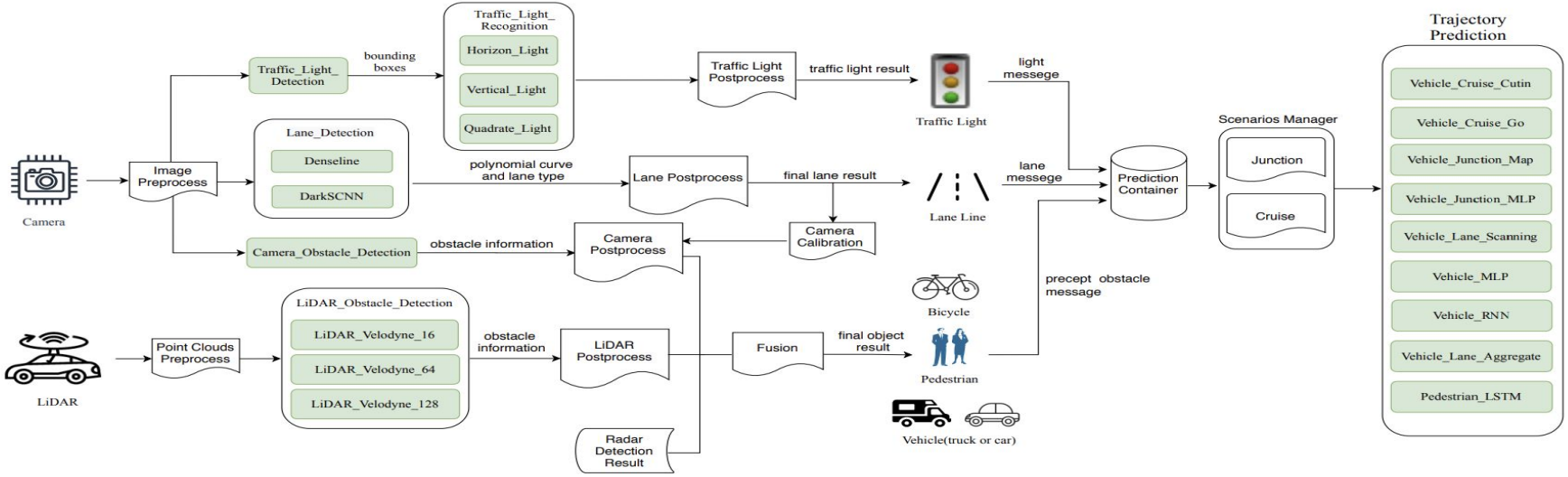


Source: https://github.com/ApolloAuto/apollo/blob/v6.0.0/README.md

Apollo Perception Module

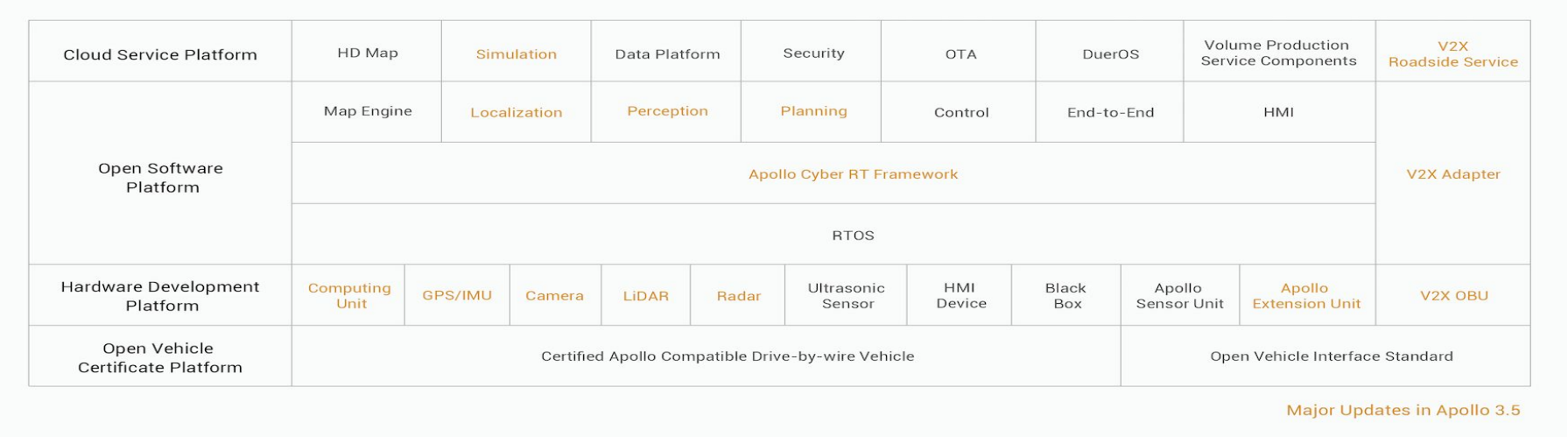


Apollo ML Models



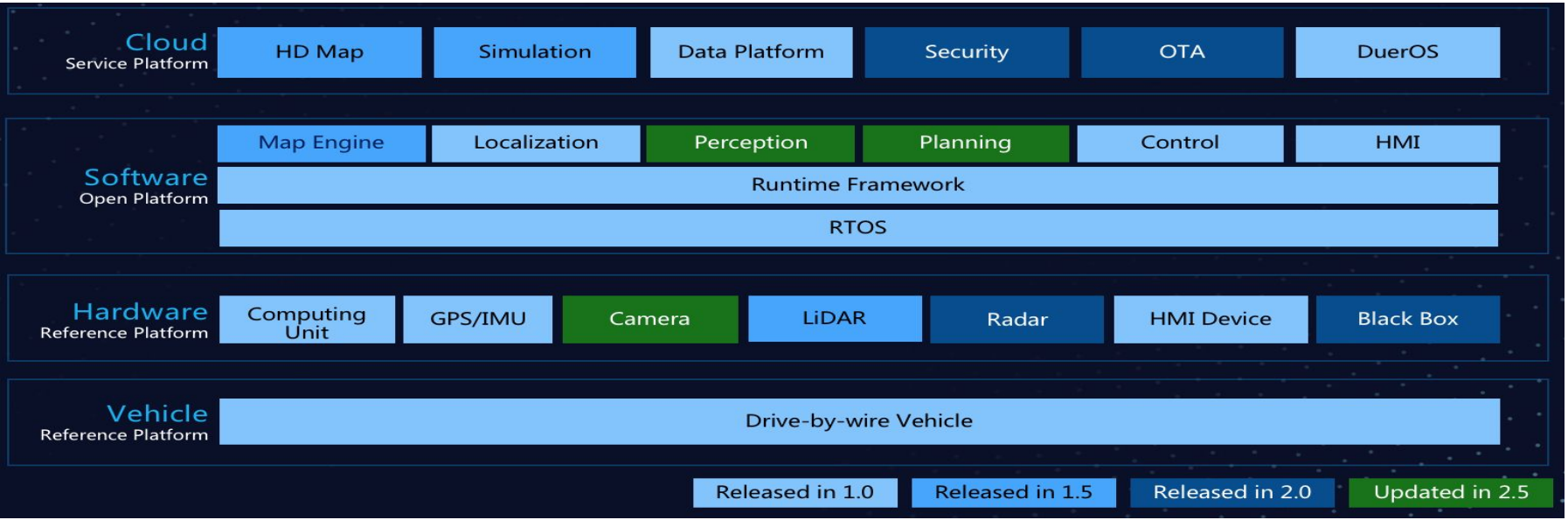
Source: Zi Peng, Jinqiu Yang, Tse-Hsun (Peter) Chen, and Lei Ma. 2020. A First Look at the Integration of Machine Learning Models in Complex Autonomous Driving Systems: A Case Study on Apollo. In Proceedings of the 28th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE '20), https://doi.org/10.1145/ 3368089.3417063

Apollo Software Stack



Source: https://github.com/ApolloAuto/

Feature Evolution (Software Stack View)



Source: https://github.com/ApolloAuto/apollo

Outline

- Views and Abstraction
- Case Study: Autonomous Vehicles
- **Software Architecture**
  - **Definitions, Importance**
  - **Software Design vs. Software Architecture**
- Architecting software
  - Integrating Architectural Decisions into the SW Development Process
  - Common Software Architectures
  - Documentation

Software Architecture

The software architecture of a program or computing system is the structure or structures of the system, which comprise software elements, the externally visible properties of those elements, and the relationships among them.

[Bass et al. 2003]

Note: this definition is ambivalent to whether the architecture is known or whether it's any good!

Software Architecture

- Abstraction
- Elements: roles, responsibilities, behaviors, properties
- Relationships between elements
- Relationships to non-software elements
  - Hardware, external systems
- Described from many different perspectives (views)

Software Architecture: Motivation

- Facilitates internal and external communication
- Describes design decisions and prescribes implementation constraints
- Relates to organizational structure
- Permits/precludes achieving non-functional requirements
- Control complexity
- Reason about and manage change
- Good basis for effort estimation
- …

Architecting Software the SEI Way - Software Architecture Fundamentals: Technical, Business, and Social Influences. Robert Wojcik. 2012

Software Architecture

“Architecture is about the important stuff. Whatever that is.”

Ralph Johnson

design

Editor: Martin Fowler ■ ThoughtWorks ■ fowler@acm.org

Who Needs an Architect?

Martin Fowler

Wandering down our corridor a while ago, I saw my colleague Dave Rice in a particularly grumpy mood. My brief question caused a violent statement, “We shouldn’t interview anyone who has ‘architect’ on his resume.” At first blush, this was an odd turn of phrase, because we usually introduce Dave as one of our leading architects.

The reason for his title schizophrenia is the fact that, even by our industry’s standards, “architect” and “architecture” are terribly overloaded words. For many, the term “software architect” fits perfectly with the smug controlling image at the end of *Matrix Reloaded*. Yet even in firms that have the greatest contempt for that image, there’s a vital role for the technical leadership that an architect such as Dave plays.

What is architecture?

When I was fretting over the title for *Patterns of Enterprise Application Architecture* (Addison-Wesley, 2002), everyone who reviewed it agreed that “architecture” belonged chitect.) However, as so often occurs, inside the blighted cynicism is a pinch of truth. Understanding came to me after reading a posting from Ralph Johnson on the Extreme Programming mailing list. It’s so good I’ll quote it all.

A previous posting said

> The RUP, working off the IEEE definition, defines architecture as “the highest level concept of a system in its environment. The architecture of a software system (at a given point in time) is its organization or structure of significant components interacting through interfaces, those components being composed of successively smaller components and interfaces.”

Johnson responded:

> I was a reviewer on the IEEE standard that used that, and I argued uselessly that this was clearly a completely bogus definition. There is no highest level concept of a system. Customers have a different concept than developers. Customers do not care at all about the structure of significant components. So, perhaps on architecture is the highest level concept that developers have of a

Software Design vs. Architecture

Levels of Abstraction

- Requirements
  - high-level “what” needs to be done
- Architecture (High-level design)
  - high-level “how”, mid-level “what”
- OO-Design (Low-level design, e.g. design patterns)
  - mid-level “how”, low-level “what”
- Code
  - low-level “how”

Design vs. Architecture

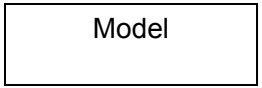
Design Questions

- How do I add a menu item in NodeBB?
- How can I make it easy to create posts in NodeBB?
- What lock protects this data?
- How does Google rank pages?
- What encoder should I use for secure communication?
- What is the interface between objects?

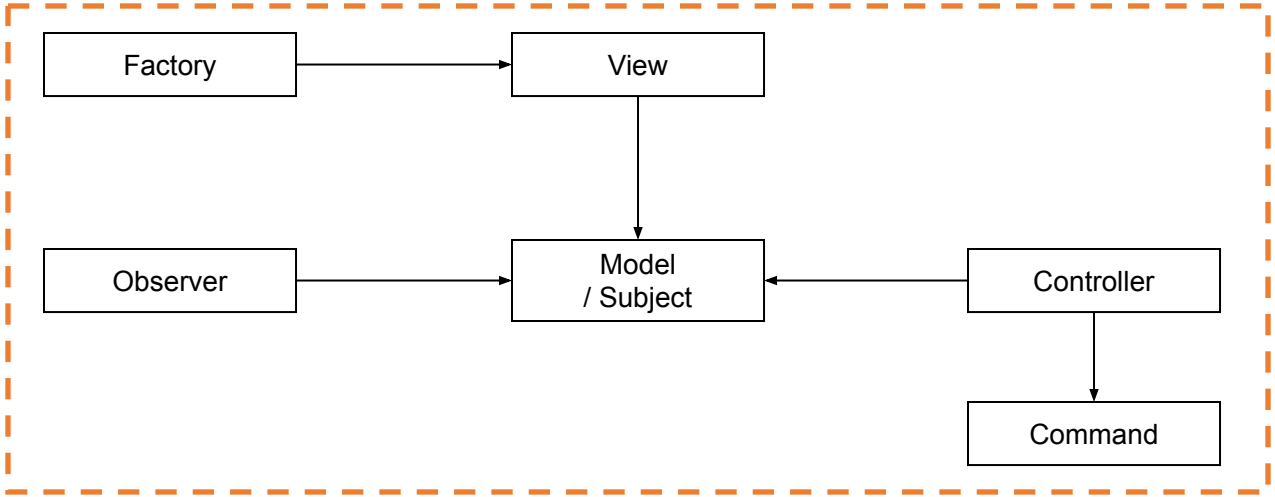
Architectural Questions

- How do I extend NodeBB with a plugin?
- What threads exist and how do they coordinate?
- How does Google scale to billions of hits per day?
- Where should I put my firewalls?
- What is the interface between subsystems?

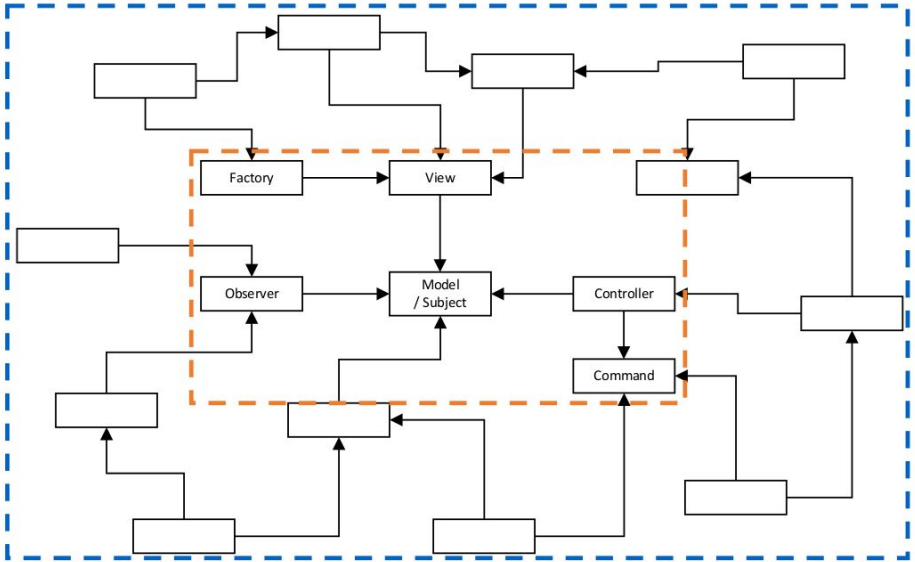
Objects



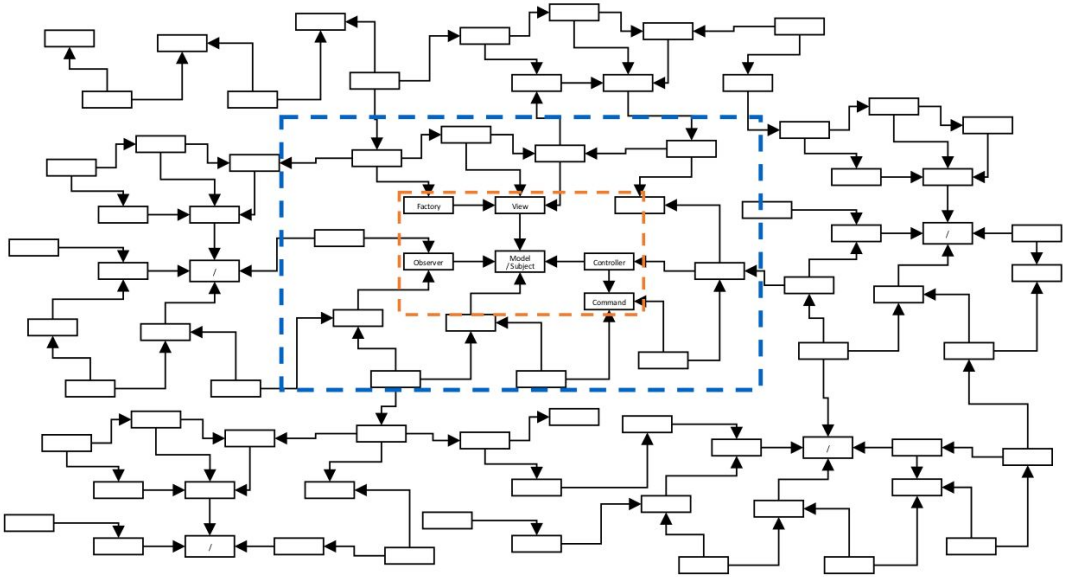
Design Patterns



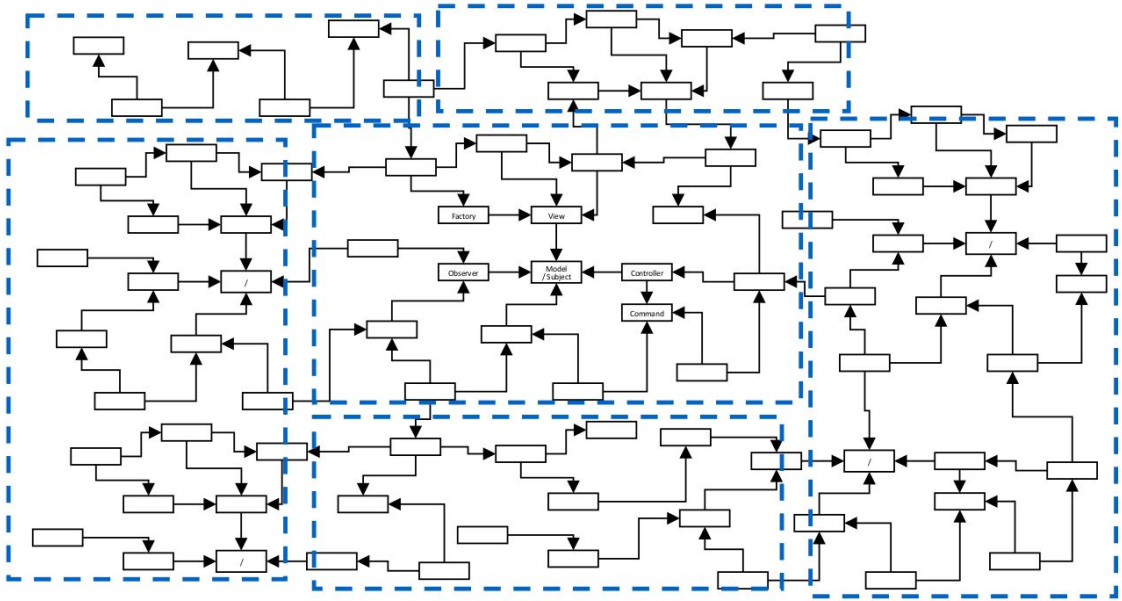
Design Patterns



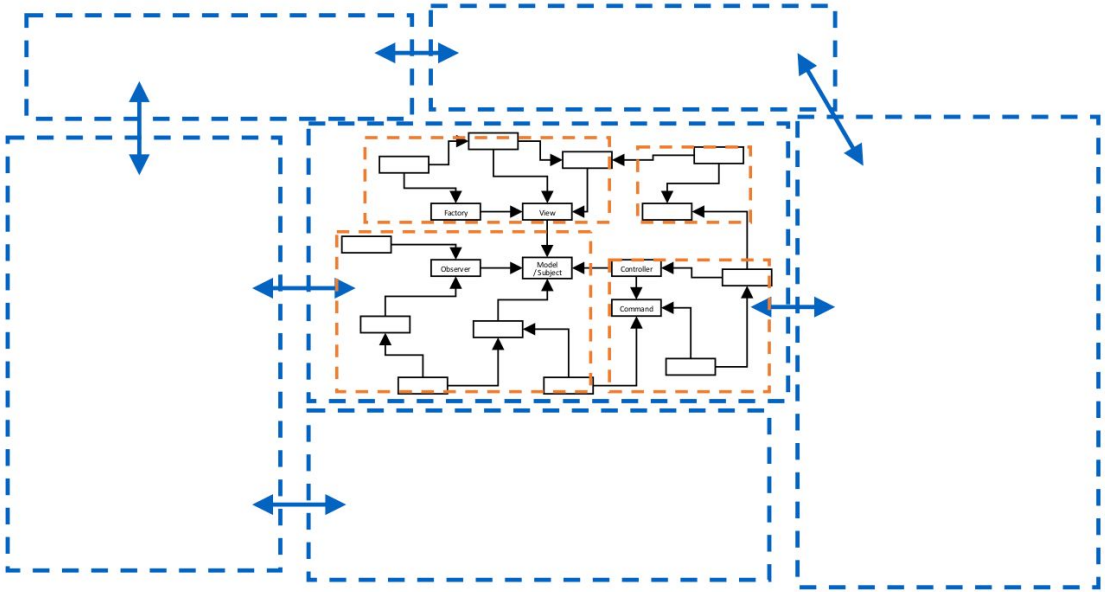
Design Patterns



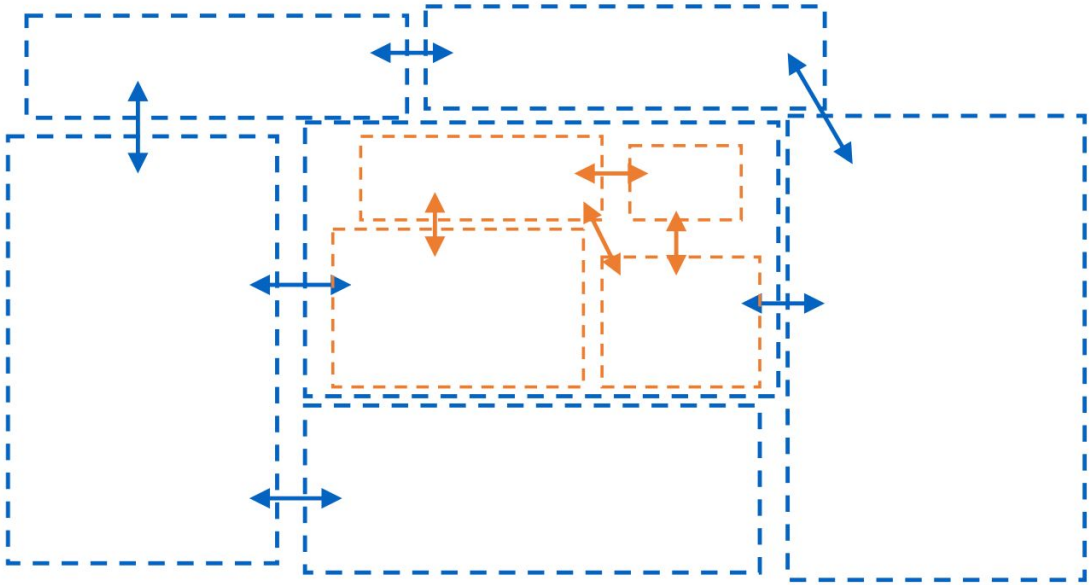
Architecture



Architecture



Architecture



Outline

- Views and Abstraction
- Case Study: Autonomous Vehicles
- Software Architecture
  - Definitions, Importance
  - Software Design vs. Software Architecture
- **Architecting software**
  - **Integrating Architectural Decisions into the SW Development Process**
  - **Common Software Architectures**
  - **Documentation**

https://www.archdaily.com/

https://www.instagram.com/architectanddesign

https://www.mykonosceramica.com/

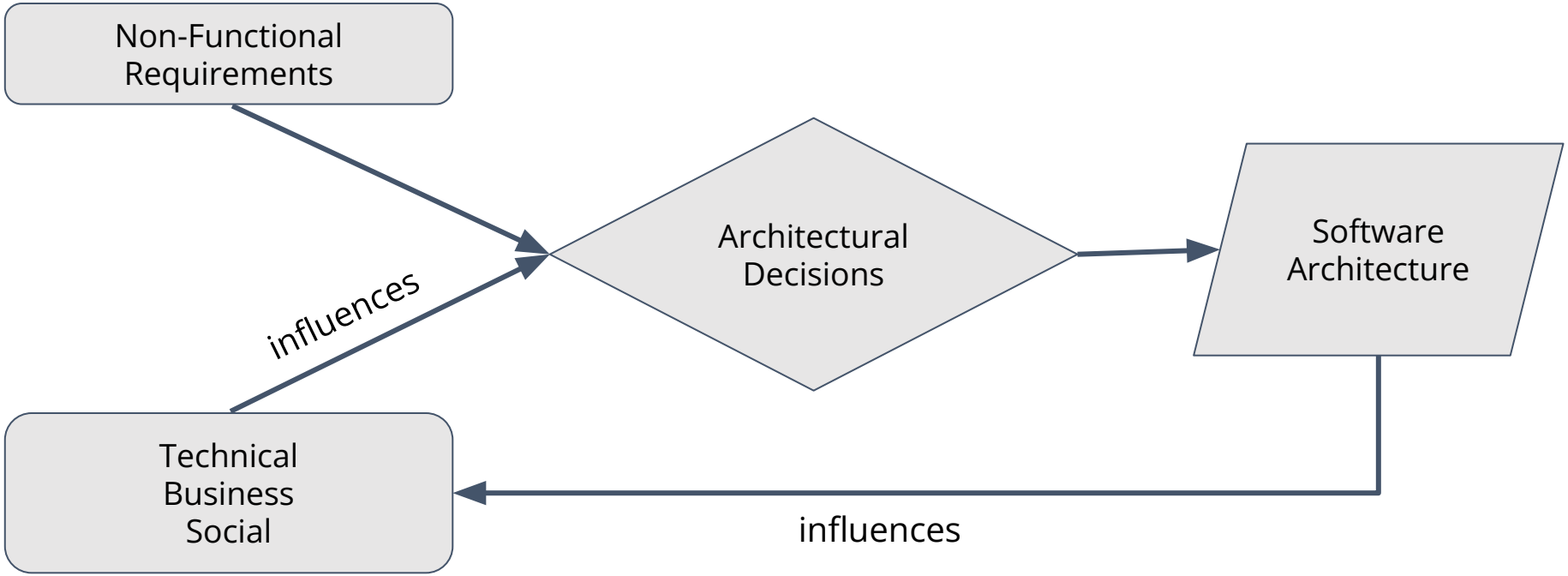
www.over-view.com

Every software system has an architecture

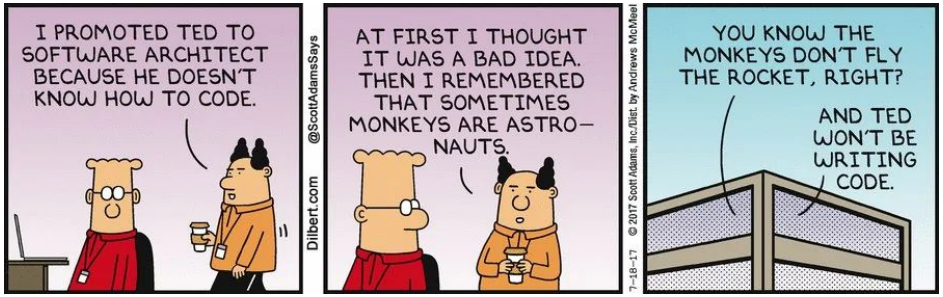
- Whether you know it or not
- Whether you like it or not
- Whether it’s documented or not

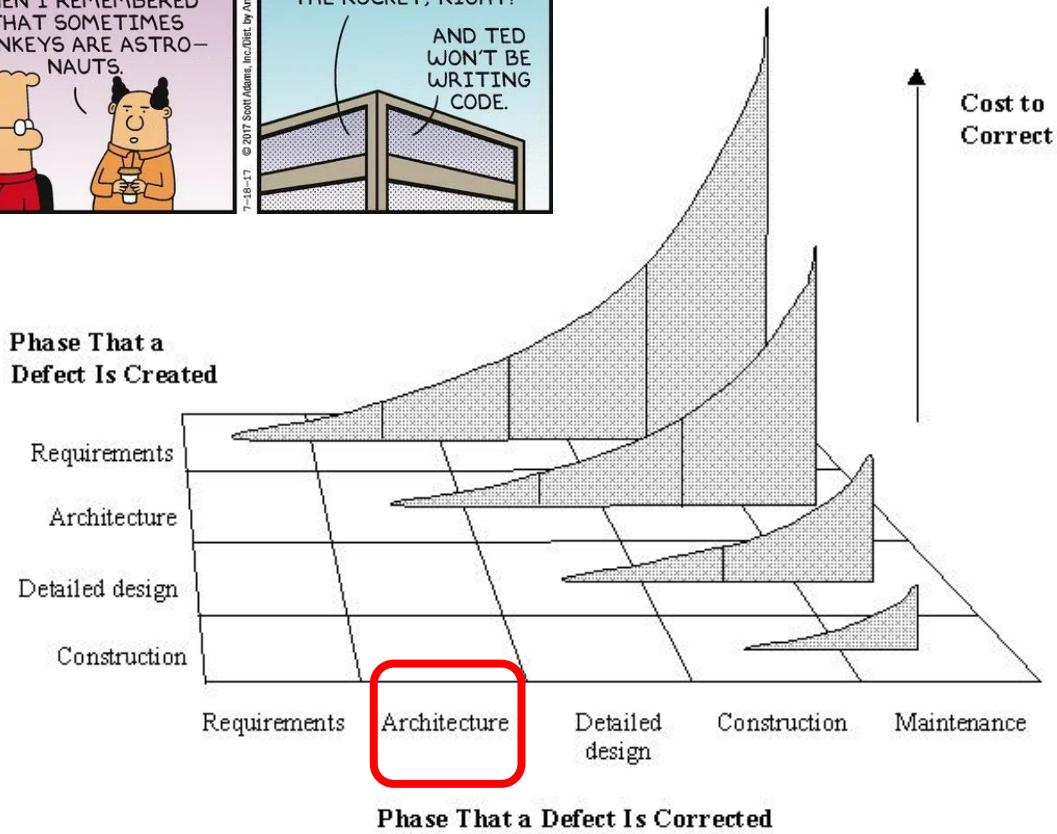
If you don’t consciously elaborate the architecture, it will evolve by itself!

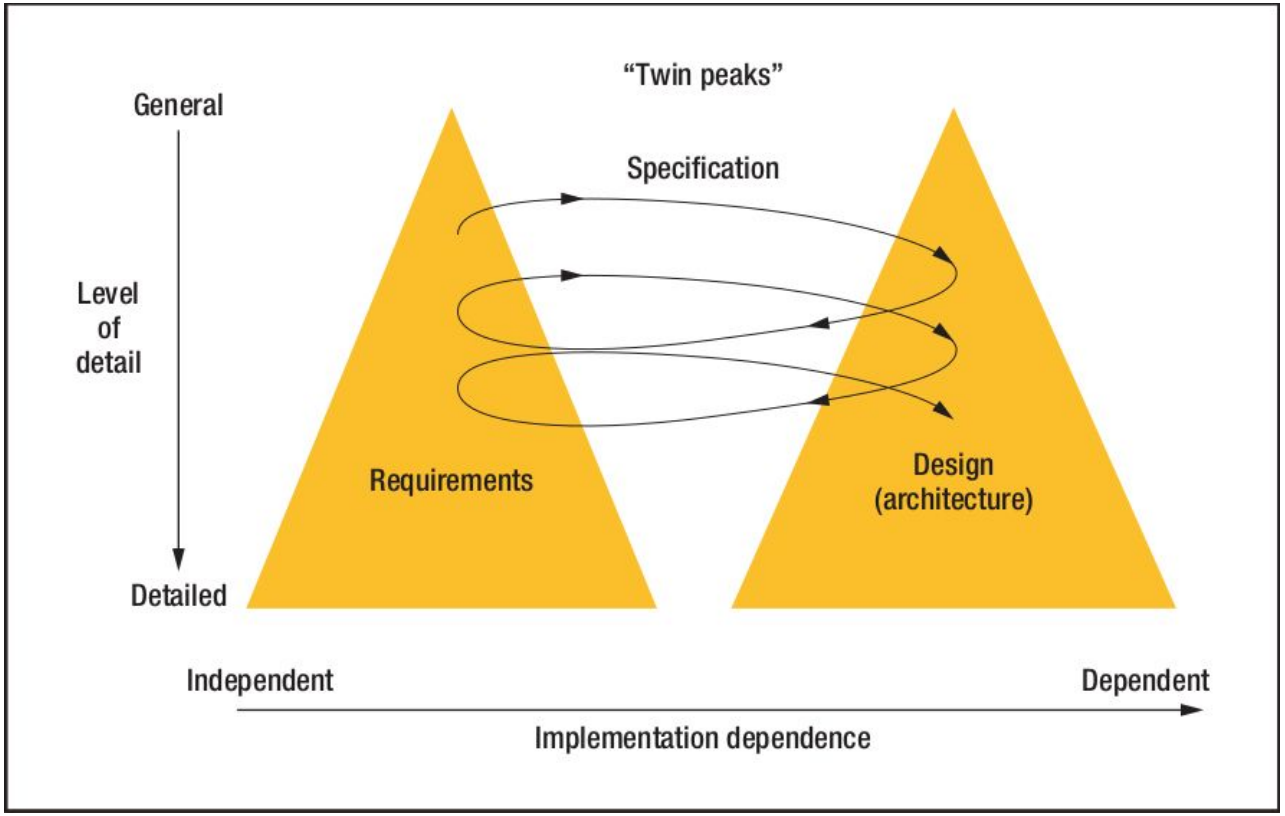
Architecting Software the SEI Way - Software Architecture Fundamentals: Technical, Business, and Social Influences. Robert Wojcik. 2012



Architecting Software the SEI Way - Software Architecture Fundamentals: Technical, Business, and Social Influences. Robert Wojcik. 2012







B. Nuseibeh, "Weaving together requirements and architectures". 2001

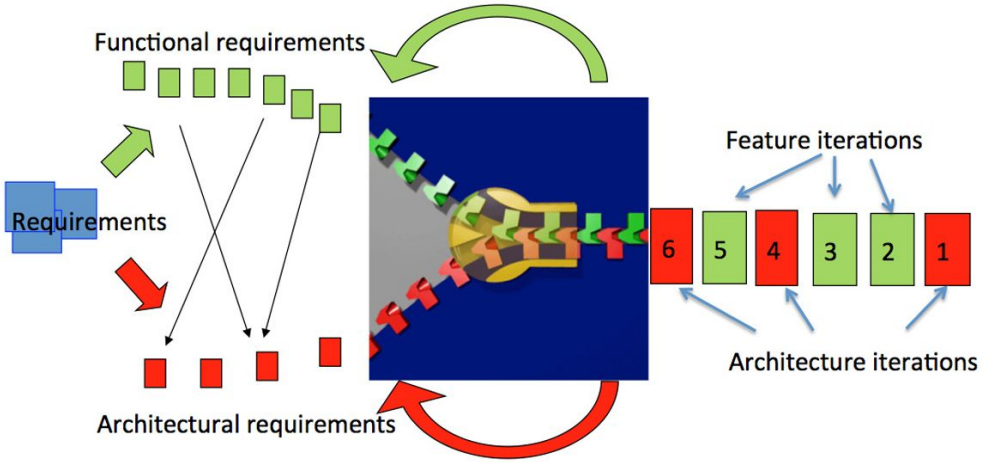
Agile and Architecture

“The best architectures, requirements, and designs emerge from self-organizing teams”

The Zipper Model

How to Agilely Architect an Agile Architecture

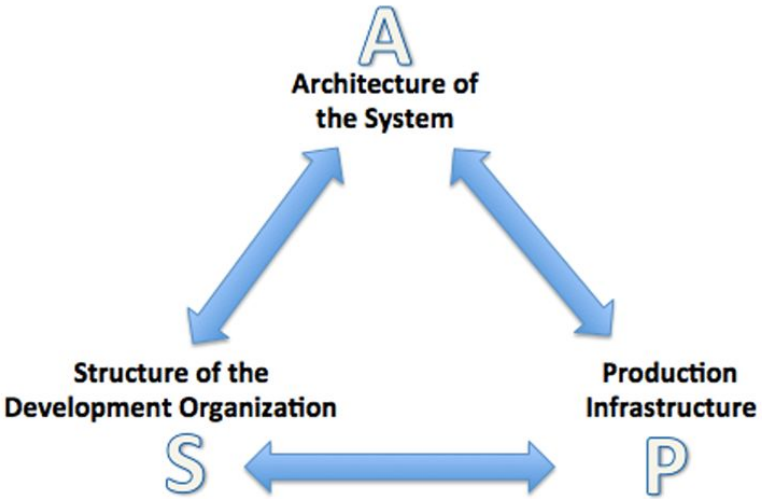
by Stephany Bellomo, Philippe Kruchten, Robert L. Nord, and Ipek Ozkaya



Agile in Distress: Architecture to the Rescue

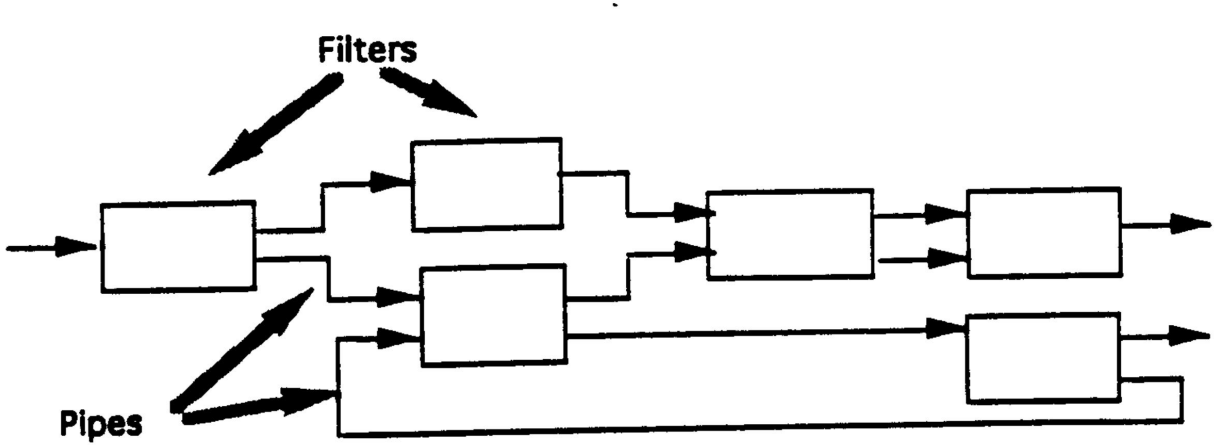
Robert L. Nord[1], Ipek Ozkaya[1], and Philippe Kruchten[2]

[1] Carnegie Mellon Software Engineering Institute, Pittsburgh, PA, USA
{rn,ozkaya}@sei.cmu.edu
[2] Electrical & Computer Engineering, University of British Columbia, Vancouver, Canada
pbk@ece.ubc.ca



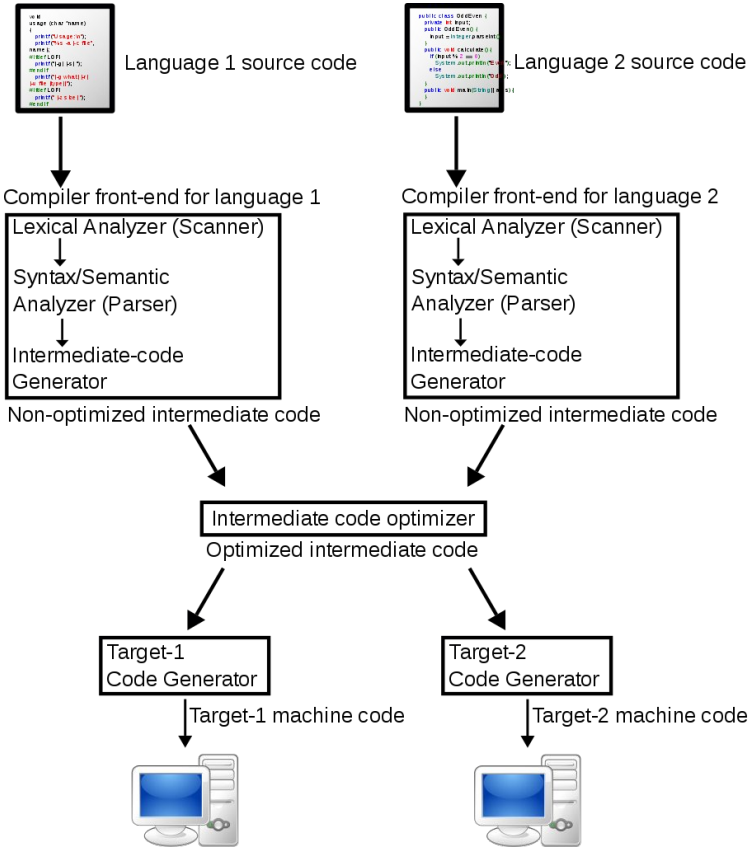
Common Software Architectures

1. Pipes and Filters

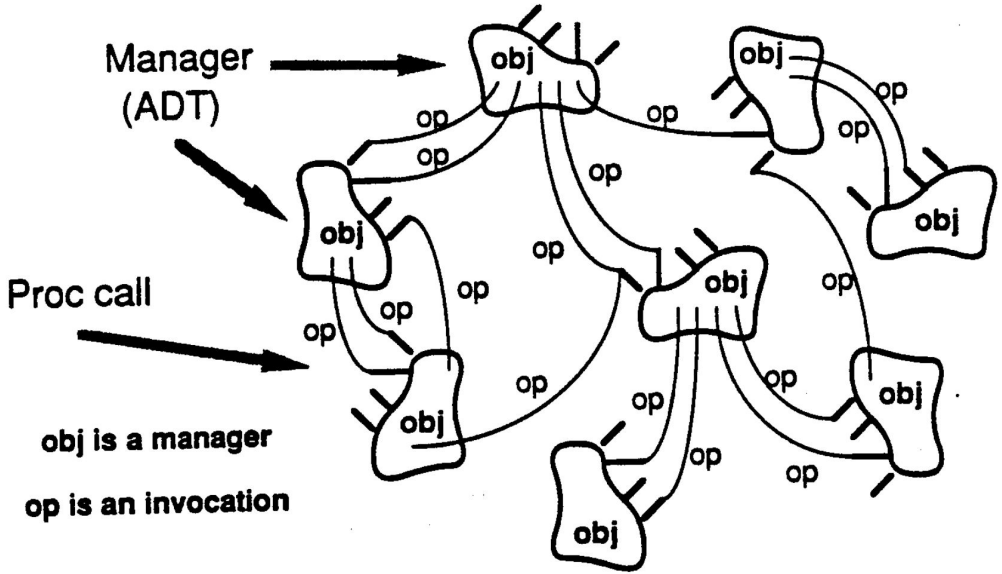


© David Garlan and Mary Shaw, CMU/SEI-94-TR-021

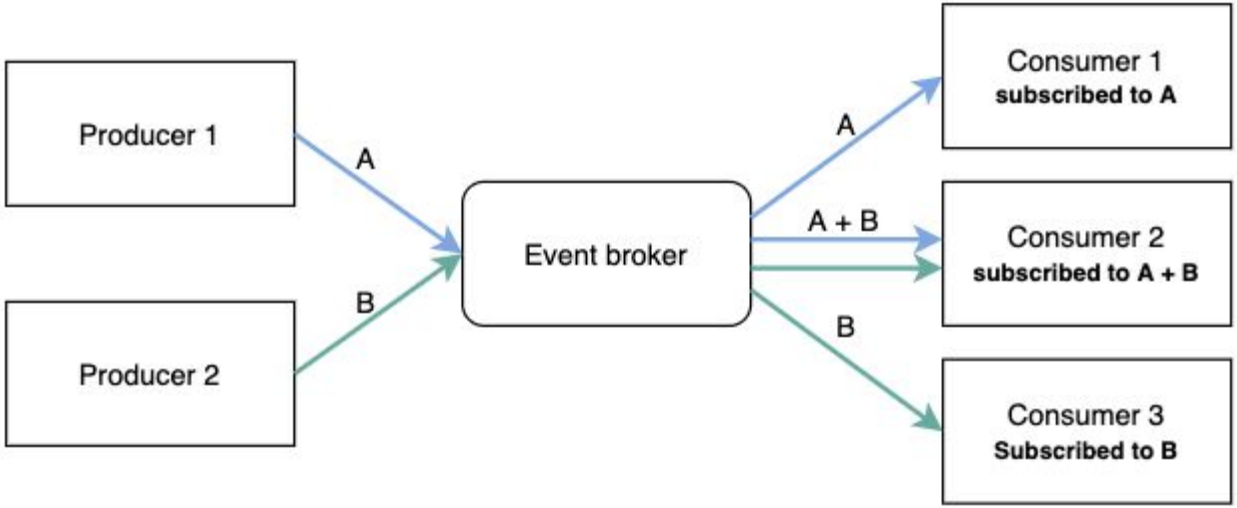
Example: Compilers



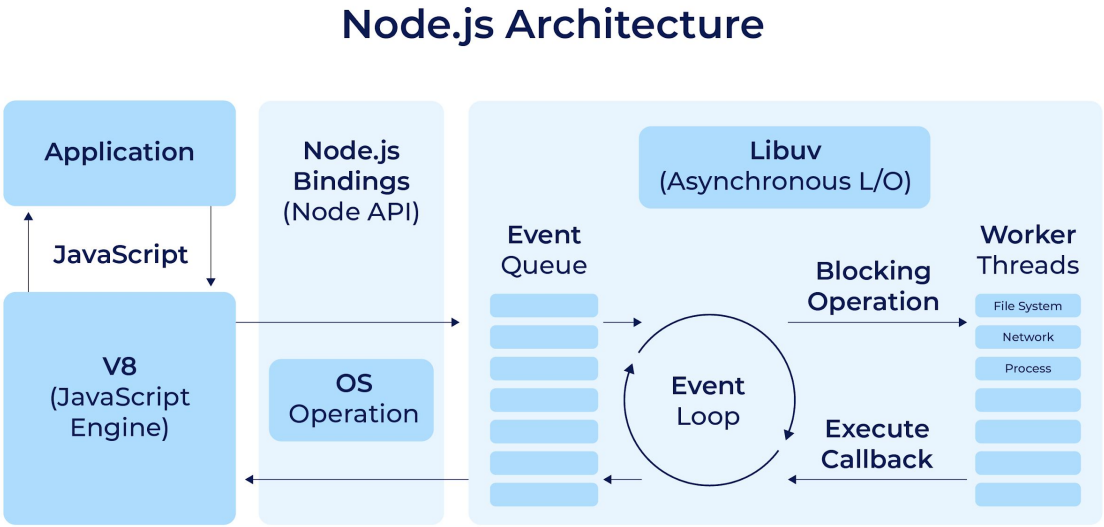
2. Object-Oriented Organization



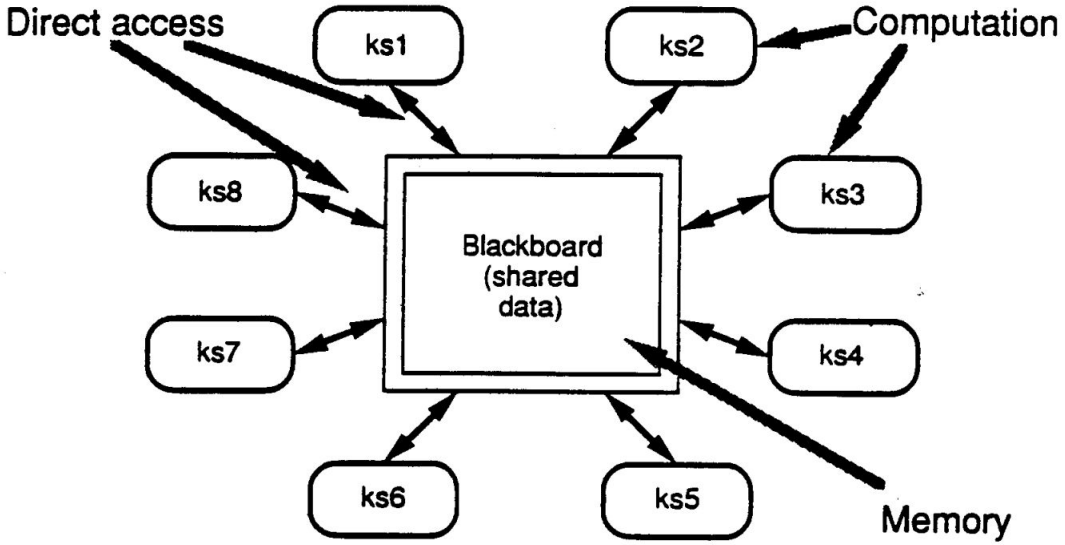
3. Event-Driven Architecture



Example: Node.js

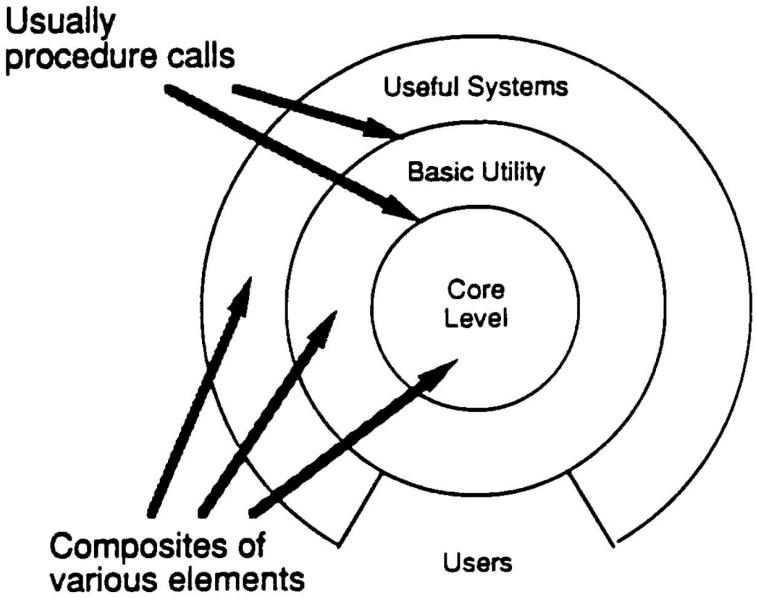


4. Blackboard Architecture



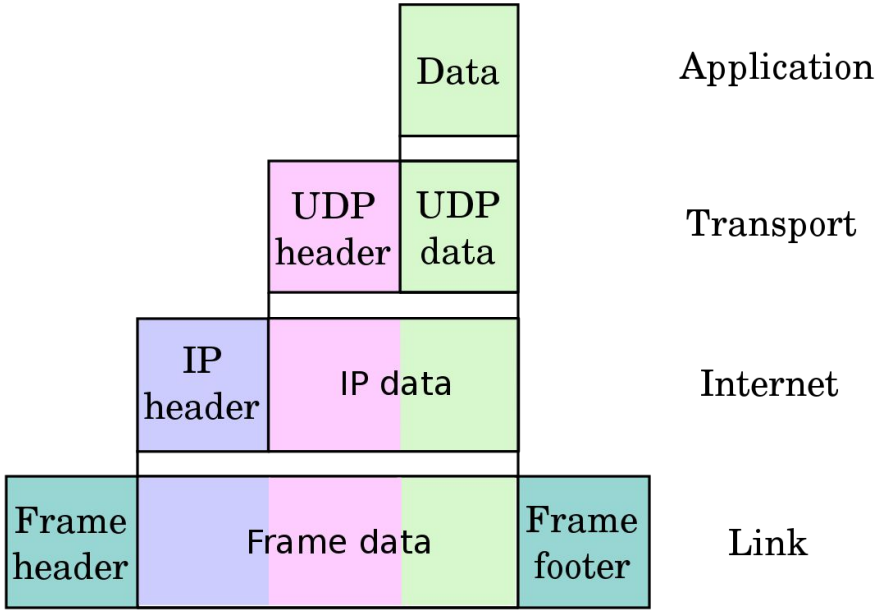
© David Garlan and Mary Shaw, CMU/SEI-94-TR-021

5. Layered Systems



© David Garlan and Mary Shaw, CMU/SEI-94-TR-021

Example: Internet Protocol Suite



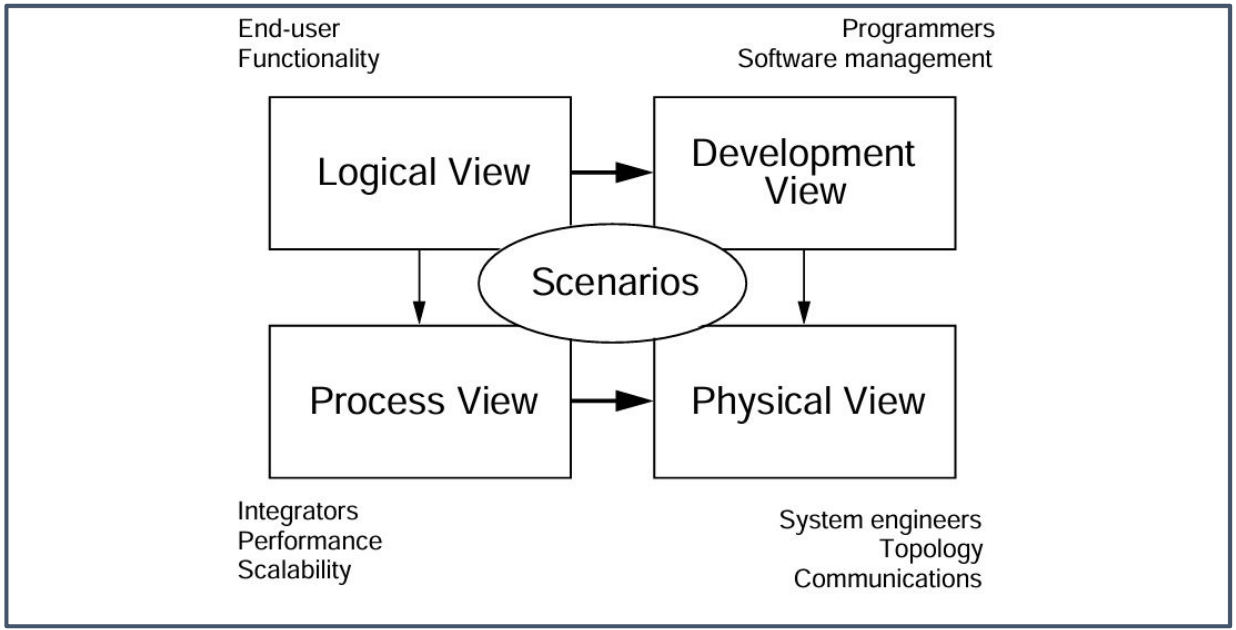
Why Document Architecture?

- Blueprint for the system
  - Artifact for early analysis
  - Primary carrier of quality attributes
  - Key to post-deployment maintenance and enhancement
- Documentation speaks for the architect, today and 20 years from today
  - As long as the system is built, maintained, and evolved according to its documented architecture
- Support traceability.

Views and Purposes

- Every view should align with a purpose
- Views should only represent information relevant to that purpose
  - Abstract away other details
  - Annotate view to guide understanding where needed
- Different views are suitable for different reasoning aspects (different quality goals), e.g.,
  - Performance
  - Extensibility
  - Security
  - Scalability
  - …

The “4+1” view model

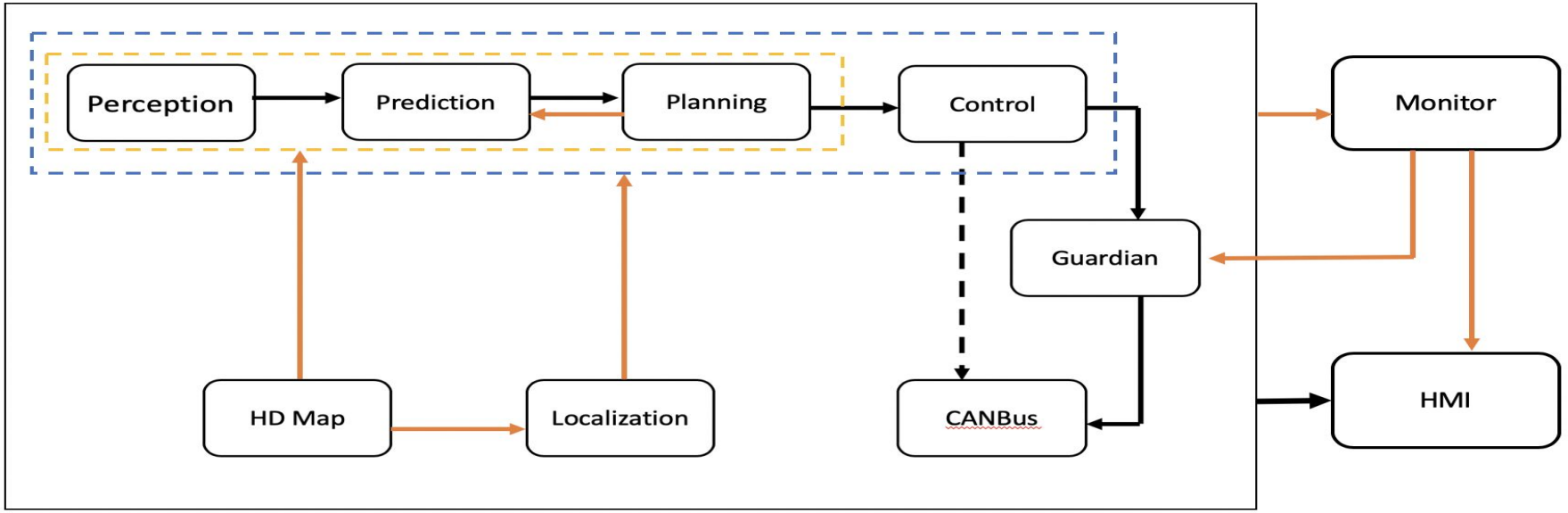


Philippe Kruchten, Architectural Blueprints—The “4+1” View Model of Software Architecture[

Common Views in Documenting Software Architecture

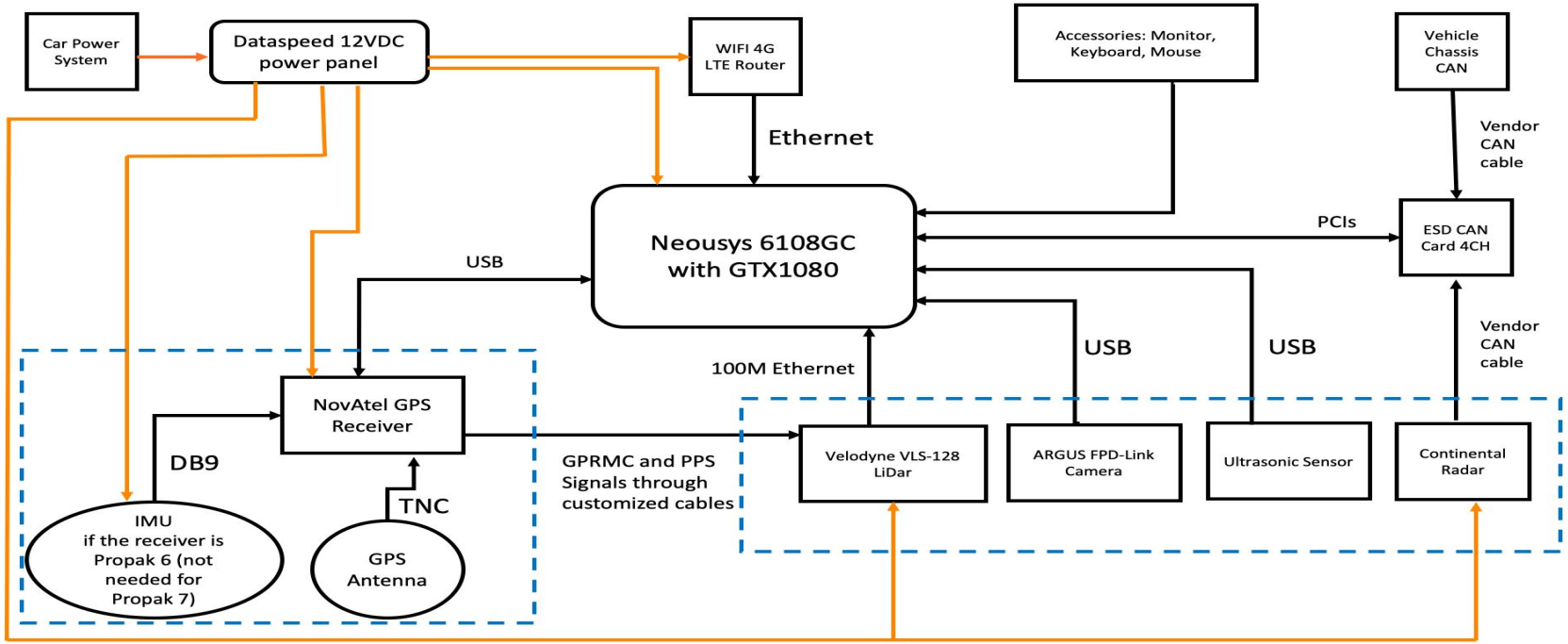
- Logical View (End user)
  - Functionality
  - Subsystems, structures and their relations (dependencies, …)
- Process View (System Integration)
  - Non-functional aspects
  - Components (processes, runnable entities) and connectors (messages, data flow, …)
- Development View (Developers)
  - Software modularity / decomposition
- Physical View (System Engineer/DevOps)
  - Hardware structures and their connections
  - Deployment
- Scenarios (All)
  - Outline tasks/use cases
  - Sequences of interactions between objects and processes

Apollo Software Architecture



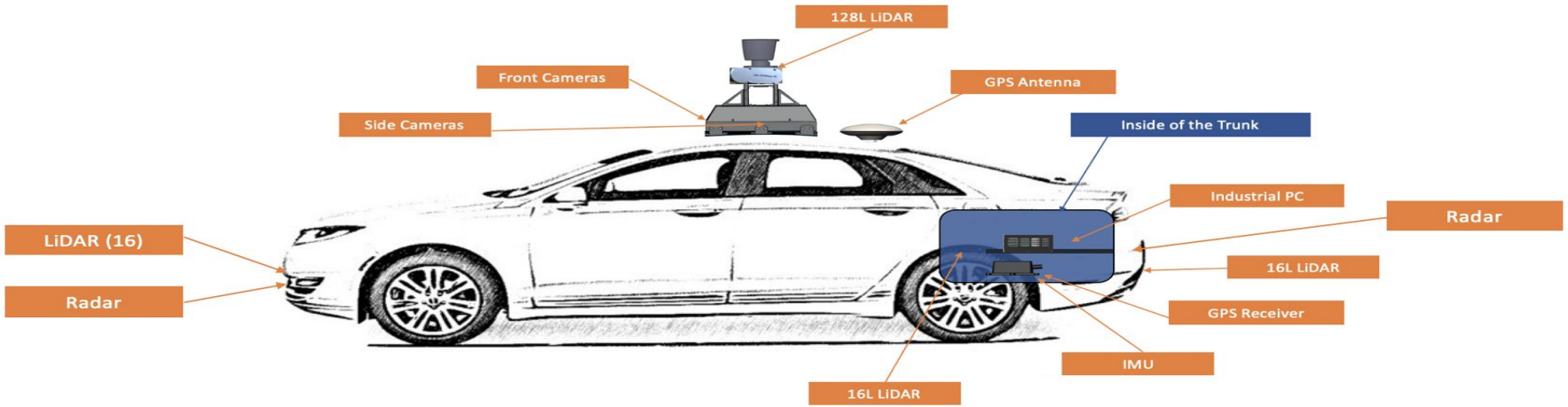
Source: https://github.com/ApolloAuto/apollo/blob/v6.0.0/docs/specs/Apollo_5.5_Software_Architecture.md

Apollo Hardware Architecture



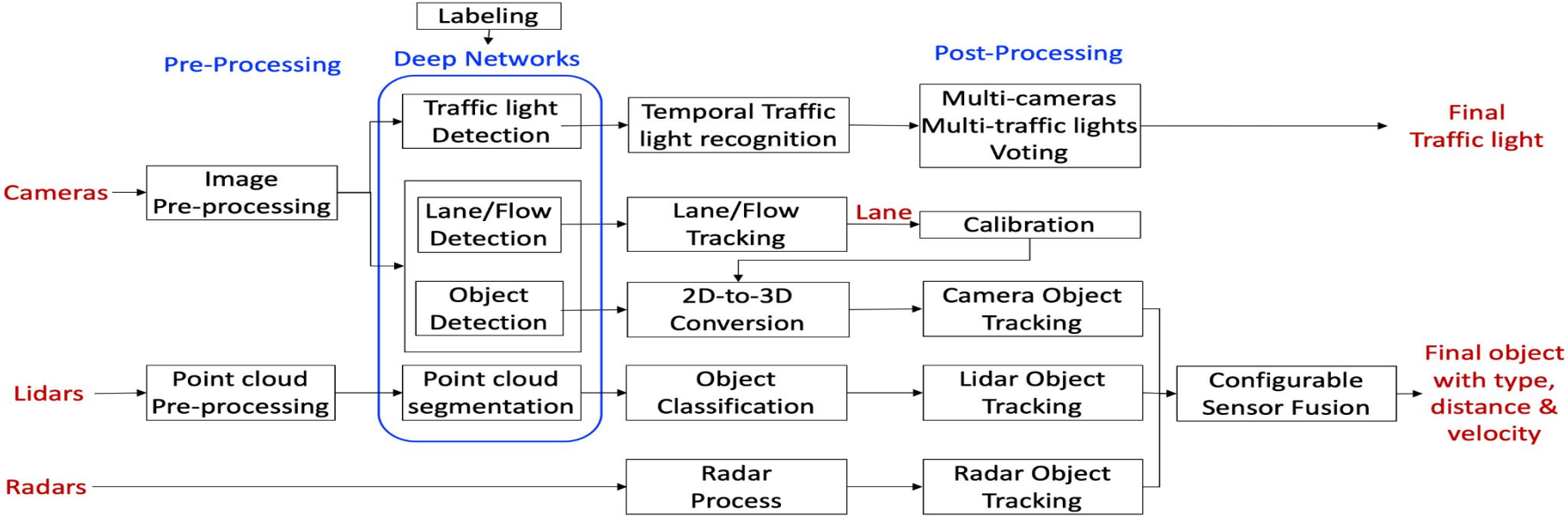
Source: https://github.com/ApolloAuto/apollo/blob/v6.0.0/README.md

Apollo Hardware/Vehicle Overview

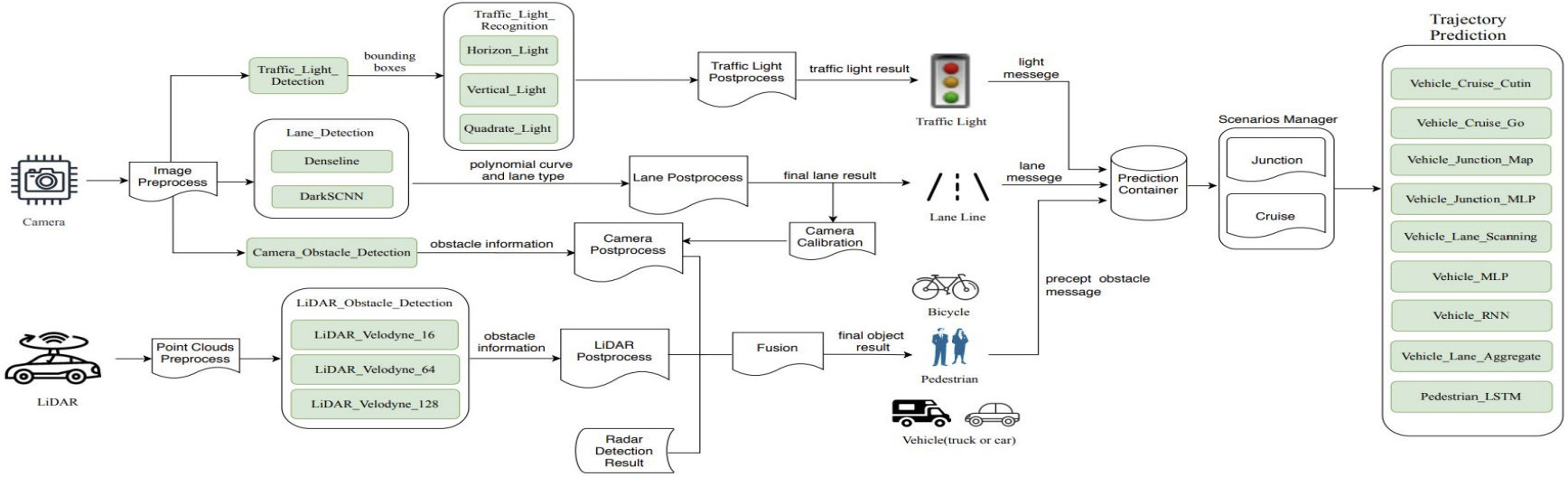


Source: https://github.com/ApolloAuto/apollo/blob/v6.0.0/README.md

Apollo Perception Module

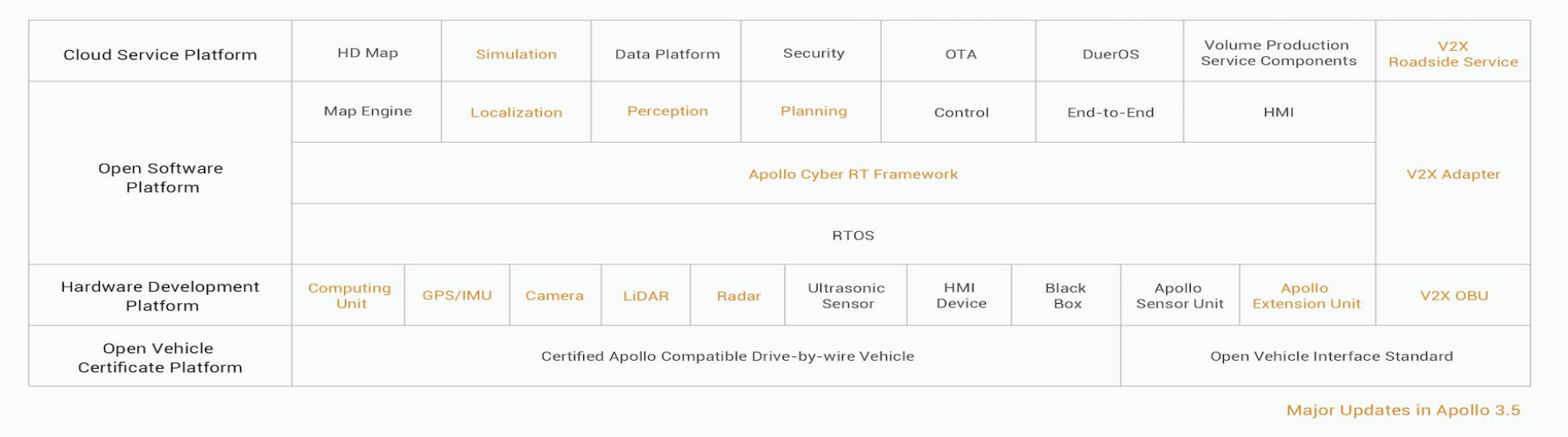


Apollo ML Models



Source: Zi Peng, Jinqiu Yang, Tse-Hsun (Peter) Chen, and Lei Ma. 2020. A First Look at the Integration of Machine Learning Models in Complex Autonomous Driving Systems: A Case Study on Apollo. In Proceedings of the 28th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE '20), https://doi.org/10.1145/ 3368089.3417063

Apollo Software Stack



Source: https://github.com/ApolloAuto/

Elon Musk ✓
@elonmusk

Btw, I'd like to apologize for Twitter being super slow in many countries.
App is doing >1000 poorly batched RPCs just to render a home timeline!

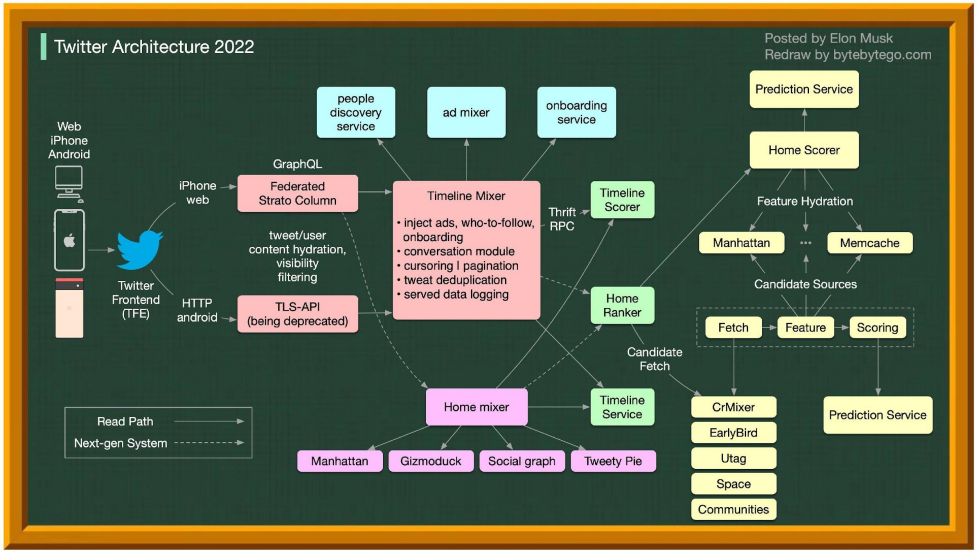
1:00 PM · Nov 13, 2022

Elon Musk ✓
@elonmusk

Just leaving Twitter HQ code review

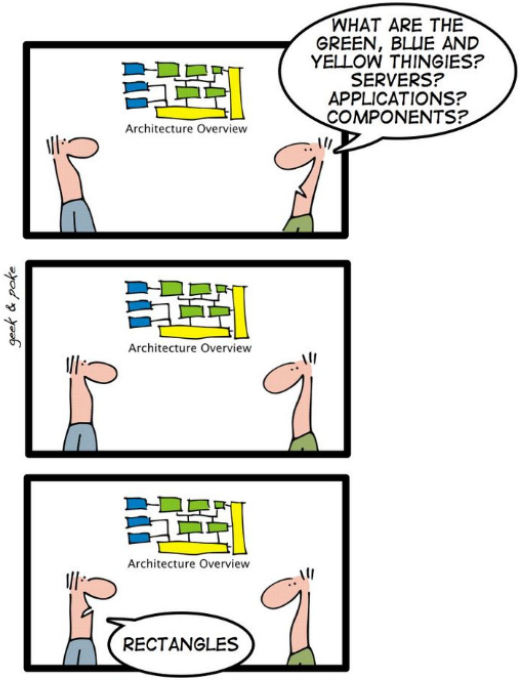
4:28 AM · Nov 19, 2022

36.9K Retweets **16.1K** Quote Tweets **464K** Likes



Guidelines for selecting a notation

- Suitable for purpose
- Often visual for compact representation
- Usually, boxes and arrows
- UML possible (semi-formal), but possibly constraining
  - Note the different abstraction level – Subsystems or processes, not classes or objects
- Formal notations available
- Decompose diagrams hierarchically and in views
- Always include a legend
- Define precisely what the boxes mean
- Define precisely what the lines mean
- Do not try to do too much in one diagram
  - Each view of architecture should fit on a page
  - Use hierarchy



Learning Goals

- Understand the abstraction level of architectural reasoning
- Appreciate how software systems can be viewed at different abstraction levels
- Distinguish software architecture from (object-oriented) software design
- Explain the importance of architectural decisions
- Integrate architectural decisions into the software development process
- Document architectures clearly, without ambiguity