HENDRIK EBBERS

JAVAFX ENTERPRISE

GUI Garage

OPEN SOURCE UI STUFF

Do they match?

canoo

ABOUT ME

- Hendrik Ebbers

- Senior Java Architect @ Canoo Engineering AG

- Lead of JUG Dortmund

@hendrikEbbers

www.guigarage.com

hendrik.ebbers@web.de

canoo

CONTENT

- JavaFX
- Enterprise Development
- JavaFX Enterprise

some basics

Best of JEE Spec

Let's test the mix

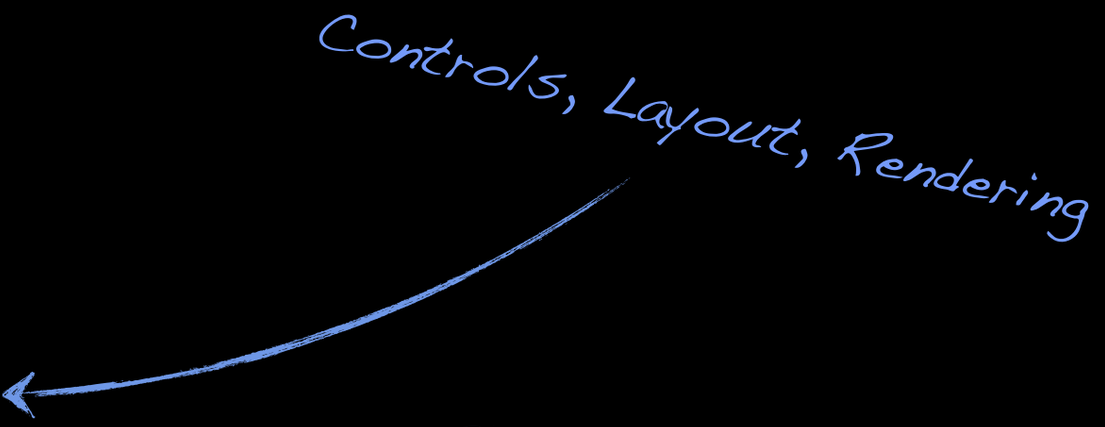
canoo

NEXT GENERATION UI WITH

JAVAFX

BASICS

- Scene Graph
- Property API
- FXML
- CSS



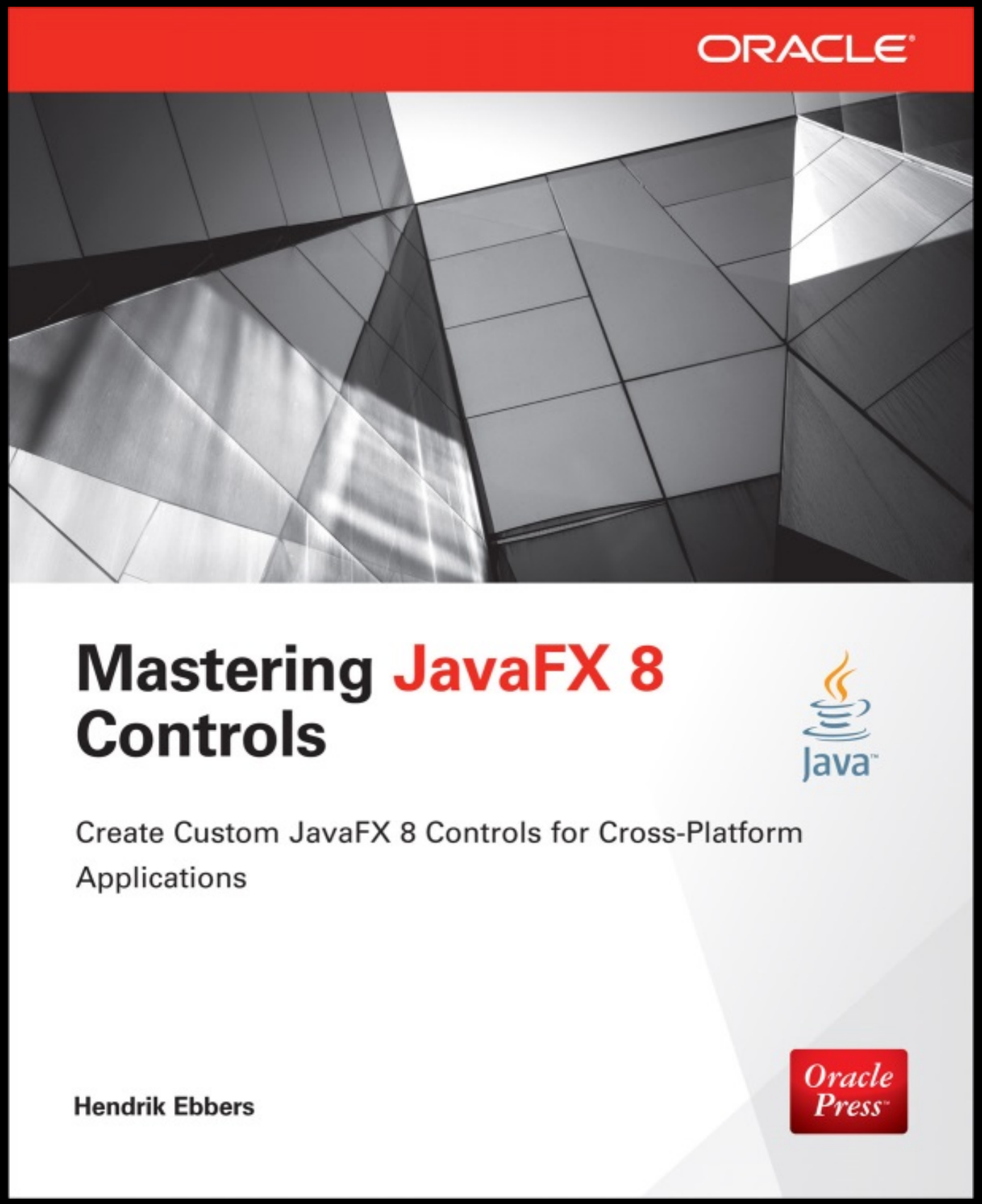
canoo

DEMO

THERE IS A LOT MORE STUFF

- Controls
- Animation
- 3D Support
- Printing
- …

canoo



ORACLE PRESS

MASTERING JAVAFX 8 CONTROLS

BEST PRACTICE

ENTERPRISE DEVELOPMENT

JAVA EE SPECIFICATIONS

- JAX-RS ← Data CRUD operations
- JAX-WS ← bidirectional communication
- JAVA BEAN VALIDATION ← just annotations
- EJB ← Local & Remote
- CDI ← manage the lifecycle & inject the data
- JSF Flow ← structure of view

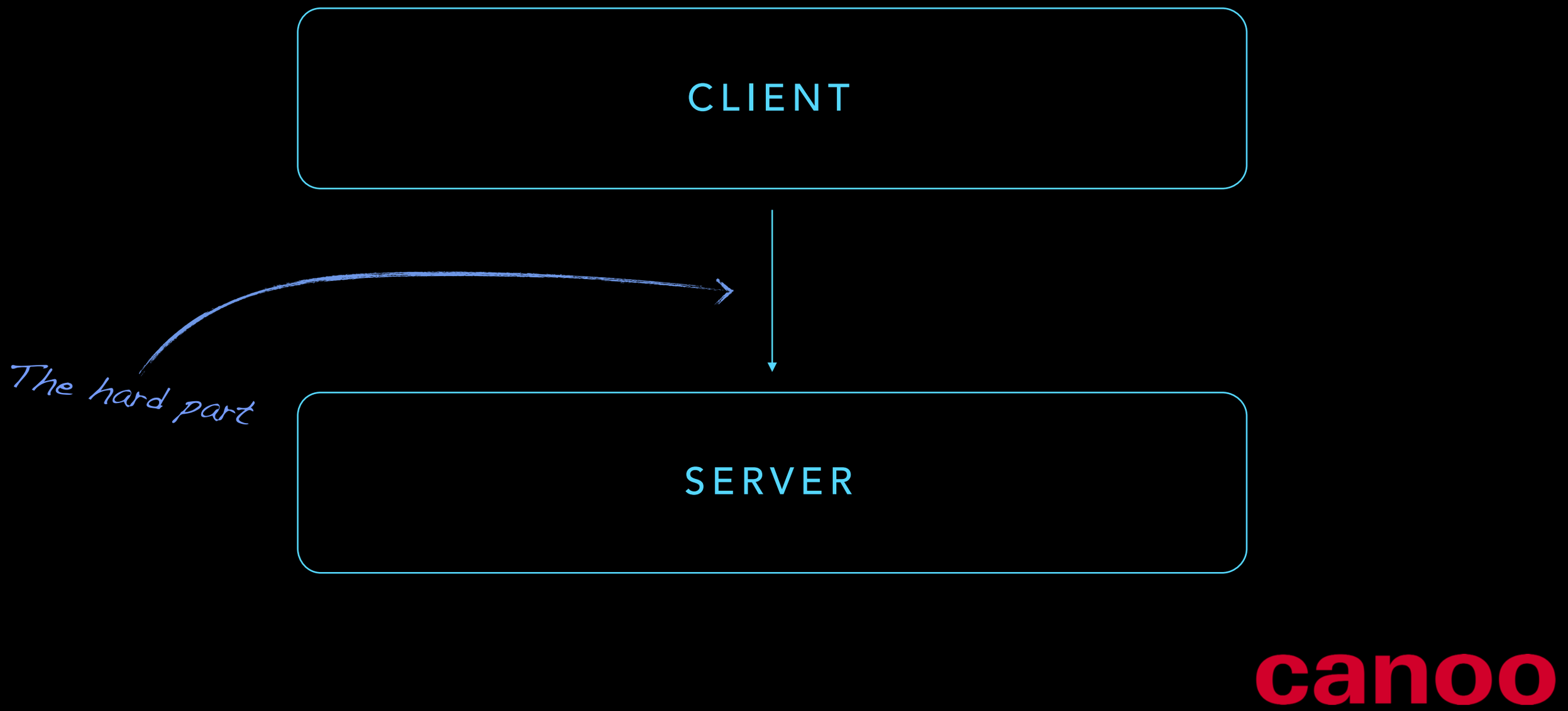
canoo

THE SEXY MIX

JAVAFX ENTERPRISE

LET'S START WITH A

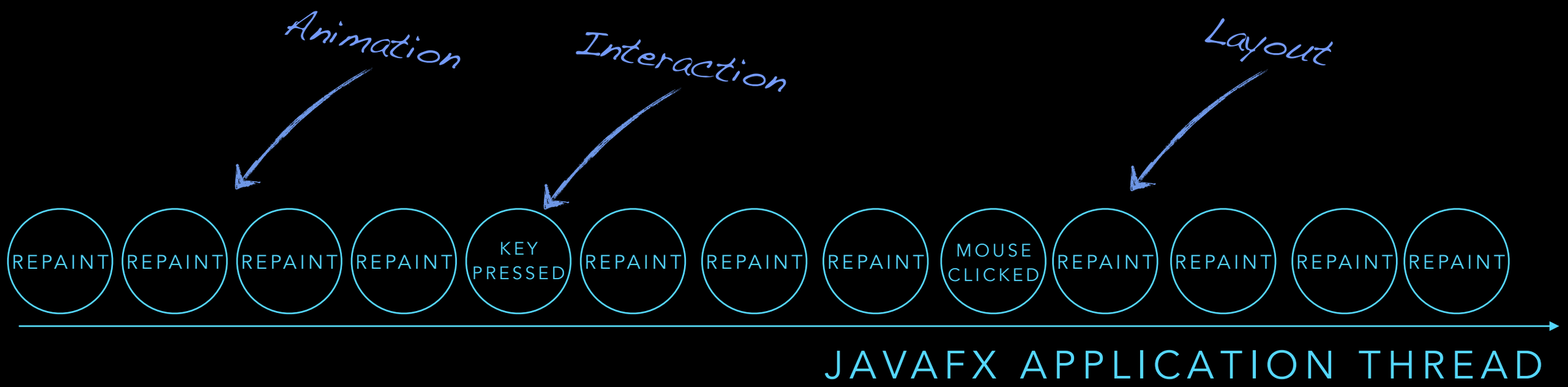
BIG PROBLEM



canoo

DESKTOP DEVELOPERS NEED TO KNOW

MULTITHREADING...

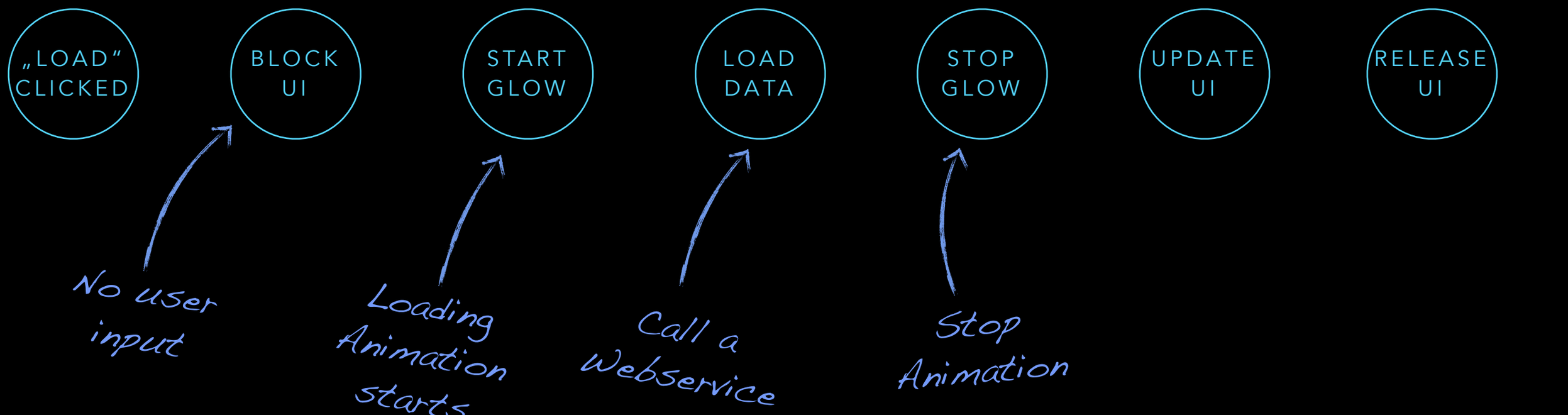


canoo

LET'S HAVE A LOOK AT A

SMALL USE CASE

STORYBOARD



canoo

OH, THAT ONE IS SIMPLE

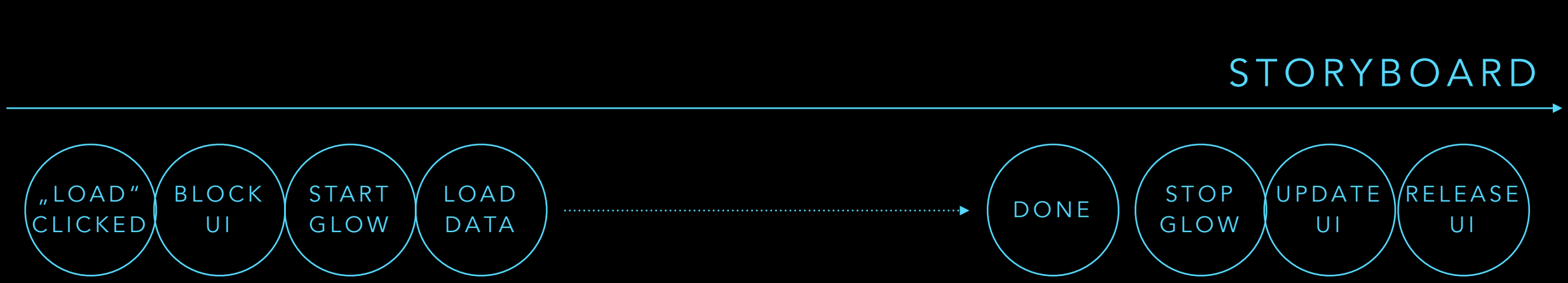
CODE SNIPPET

```
blockUI();
data = loadFromServer();
updateUI(data);
unblockUI();
```

canoo

IT'S WORKING LIKE CHARM IN MY

DEV ENVIRONMENT



canoo

BUT THE CUSTOMER HAS PROBLEMS IN THE

REAL ENVIRONMENT

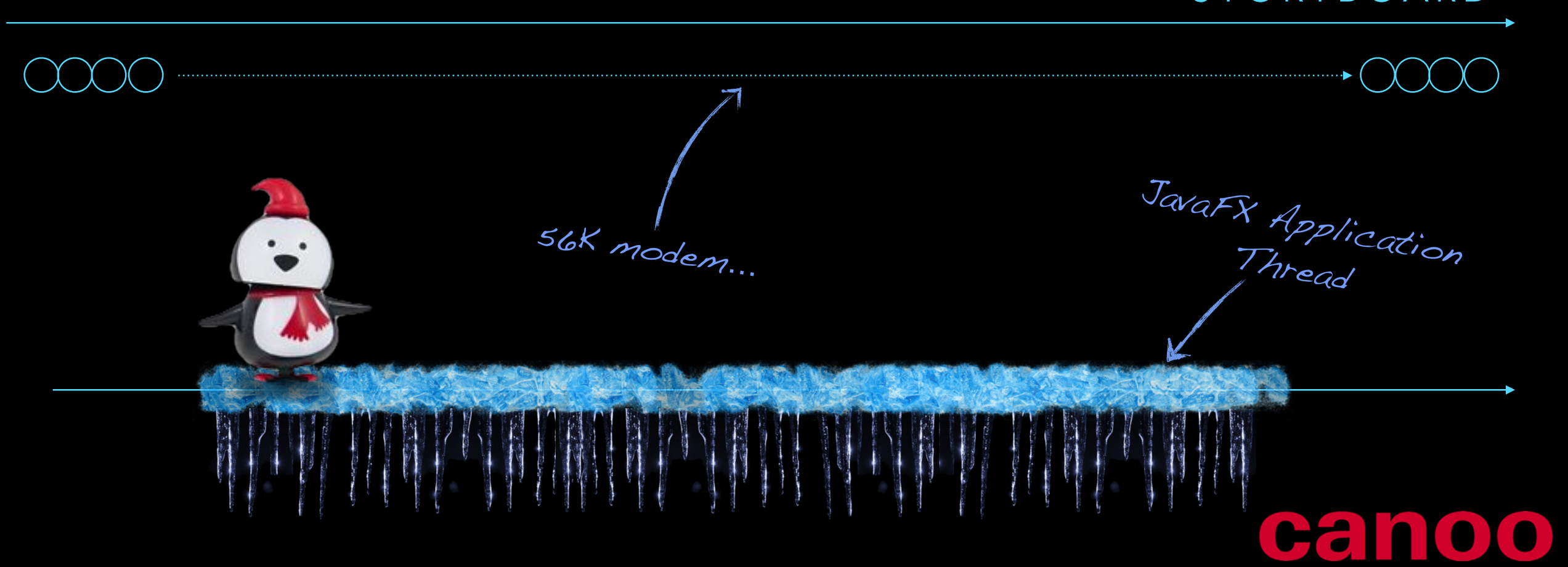
STORYBOARD

canoo

BUT THE CUSTOMER HAS PROBLEMS IN THE

REAL ENVIRONMENT

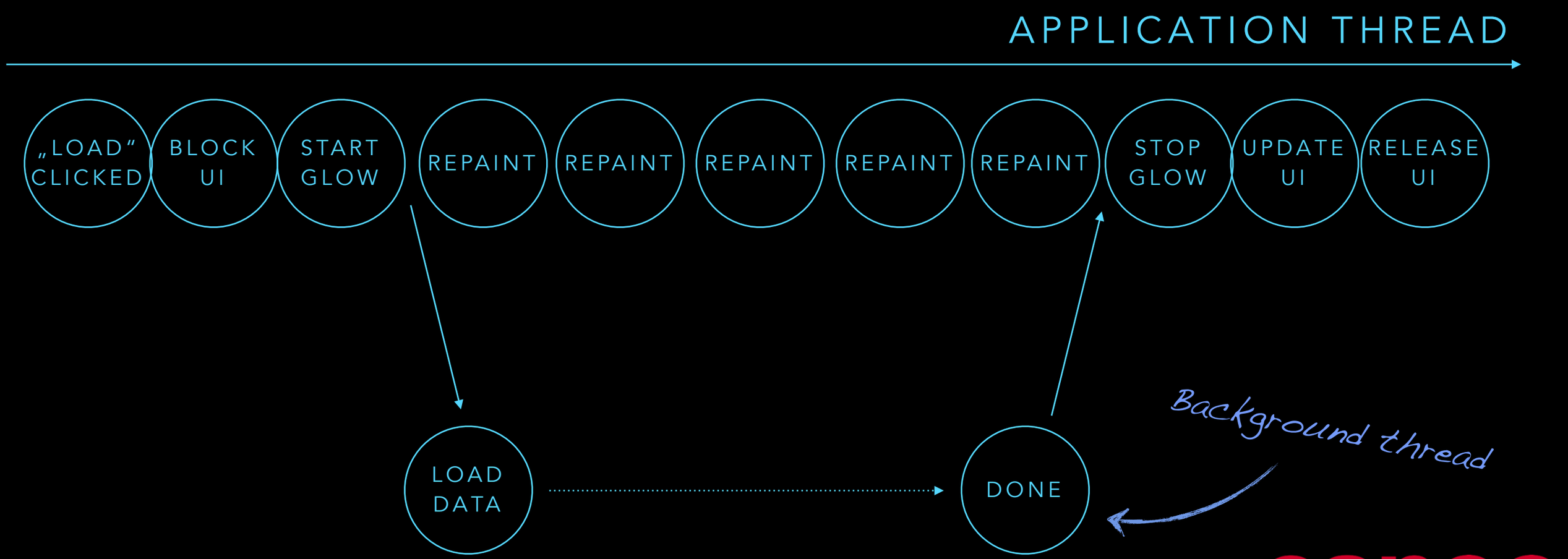
STORYBOARD



canoo

SO WE NEED A

BACKGROUND THREAD



canoo

LET'S START AND CREATE SOME

HACKED CODE

```
blockUI();
new Thread(backgroundRunnable).start();
```

How to go back to application thread?

```
Runnable backgroundRunnable = () -> {
   data = loadFromServer();
   Platform.runLater(() -> {
       updateUI(data);
       unblockUI();
    });
}
```

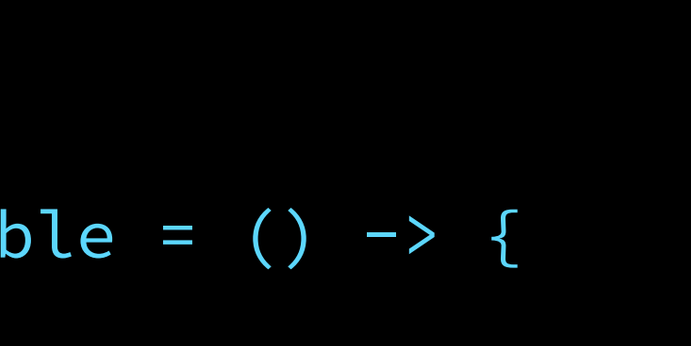
What happens if this throws an Exception?

canoo

WE NEED TO ADD SOME

EXCEPTION HANDLING

```
Runnable backgroundRunnable = () -> {
    try {
        data = loadFromServer();
    } catch(Exception e) {
        Platform.runLater(() -> {
            handleException(e);
        });
    } finally {
        Platform.runLater(() -> {
            updateUI(data);
            unblockUI();
        });
    }
}
```



canoo

WHO WANTS SOME

SPAGHETTI CODE?

```
Runnable backgroundRunnable = () -> {
    try {
        data = loadFromServer();
        Platform.runLater(() -> {
            updateUI(data);
        });
    } catch(Exception e) {
        Platform.runLater(() -> {
            handleException(e);
        });
    } finally {
        Platform.runLater(() -> {
            unblockUI();
        });
    }
}
```

canoo

DON’T LIKE IT? UNTIL LAST YEAR THIS WAS

SPAGHETTI CODE XXL

```
Runnable backgroundRunnable = new Runnable() {

     @Override
     public void run() {
          try {
               data = loadFromServer();
               Platform.runLater(new Runnable() {

                    @Override
                    public void run() {
                         updateUI(data);
                    }
               });
          } catch(Exception e) {
               Platform.runLater(new Runnable() {

                    @Override
                    public void run() {
                         handleException(e);
                    }
               });
          } finally {
               Platform.runLater(new Runnable() {

                    @Override
                    public void run() {
                         unblockUI();
                    }
               });
          }
     }
}
```

Java 8

canoo

WE NEED A

BETTER SOLUTION

DataFX 8

```
ProcessChain.create().
addRunnableInPlatformThread(() -> blockUI()).
addSupplierInExecutor(() -> loadFromServer()).
addConsumerInPlatformThread(d -> updateUI(d)).
onException(e -> handleException(e)).
withFinal(() -> unblockUI()).
run();
```

canoo

DEMO

WE CAN EVEN DO IT BETTER BY USING

REACTIVE PROGRAMMING

```
ListView<Data> listView = new ListView<>();

ProcessChain.create().
addRunnableInPlatformThread(() -> listView.getItems().clear()).
addPublishingTask(() -> listView.getItems(),
                  publisher -> getDataWithCallback(elem -> publisher.publish(elem)).
onException(e -> handleException(e)).
run();
```

canoo

TRY IT TODAY AND ADD THE

DATAFX CORE MODULE

```
<dependency>
  <groupId>io.datafx</groupId>
  <artifactId>core</artifactId>
  <version>X.Y</version>
</dependency>
```

canoo

LET'S PIMP THE APP BY ADDING

TIMEOUT CHECKS

```
class CommandGetData extends HystrixCommand<List<String>> {

    public CommandGetData() {
        super(Config.withExecutionIsolationThreadTimeoutInMilliseconds(6000)
                .withExecutionIsolationThreadInterruptOnTimeout(true)
                .withFallbackEnabled(false)));
    }

    @Override
    protected List<String> run() {
        List<String> list = new ArrayList<>();
        for (int i = 0; i < 10; i++) {
            list.add("Value " + i);
            sleep();
        }
        return list;
    }

}
```

Netflix Hystrix

DEMO

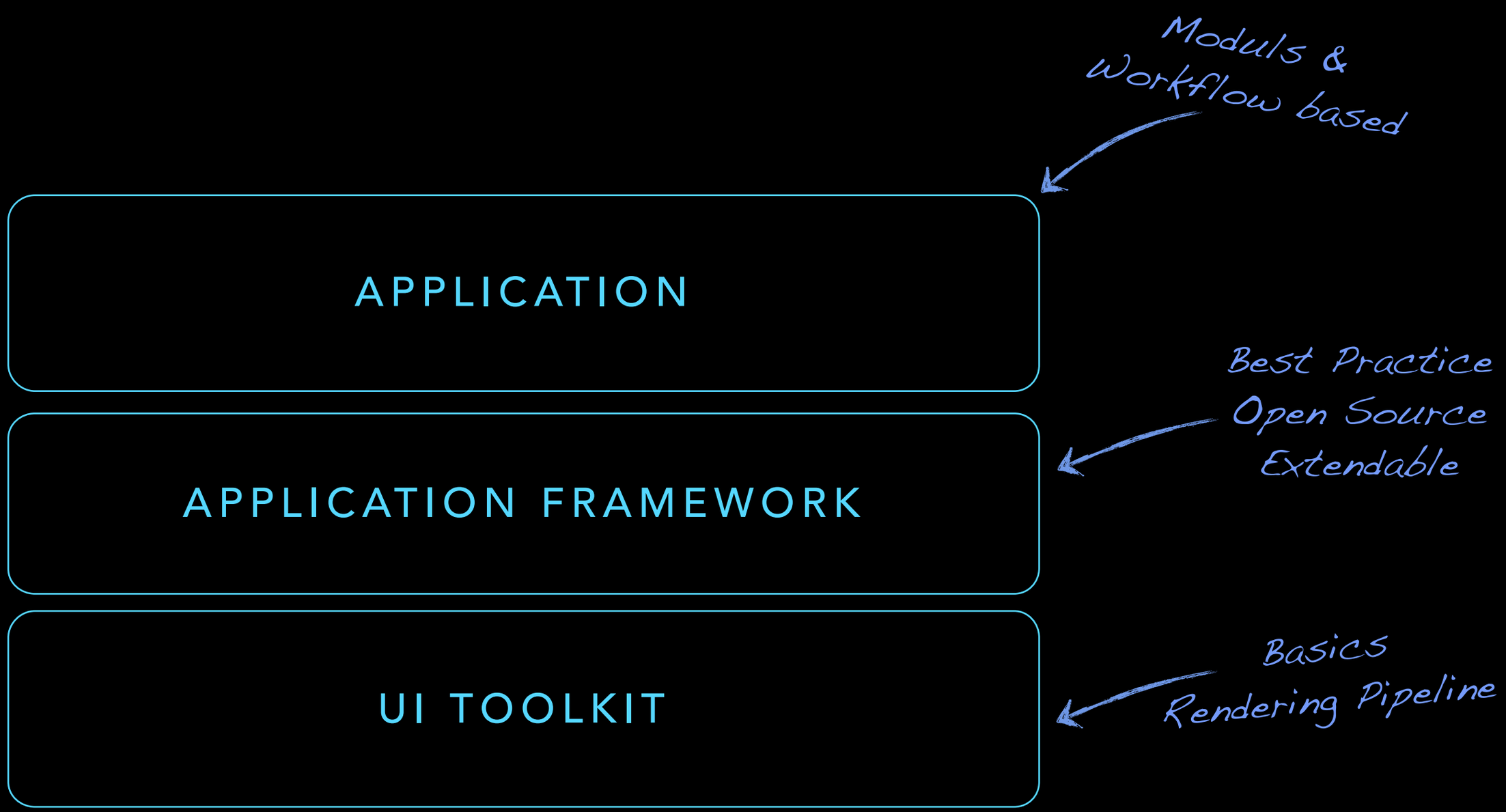
YOU ONLY NEED TO ADD THE

HYSTRIX DEPENDENCY

```
<dependency>
  <groupId>com.netflix.hystrix</groupId>
  <artifactId>hystrix-core</artifactId>
  <version>1.3.18</version>
</dependency>
```

LET'S HAVE A LOOK AT THE

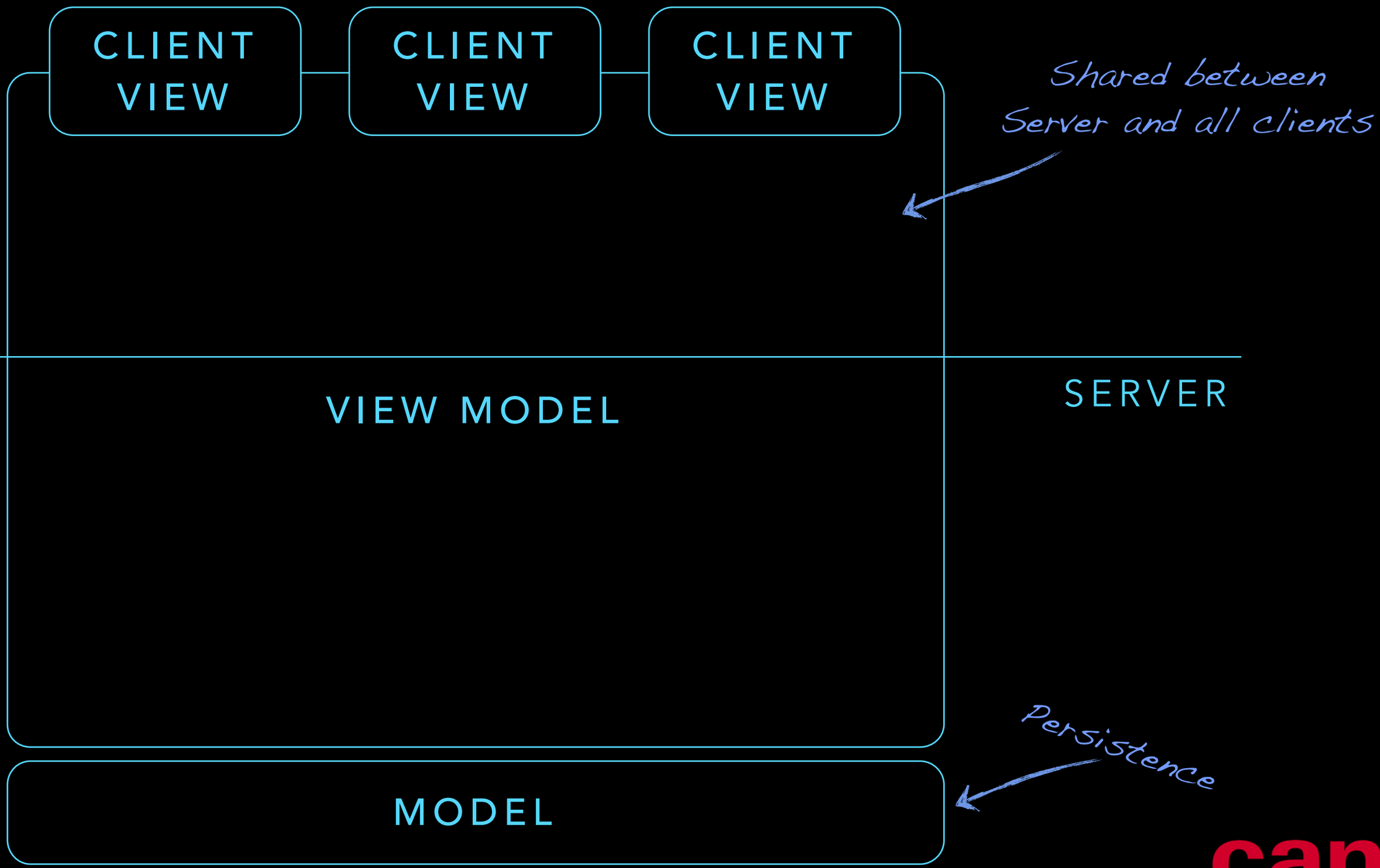
CLIENT ARCHITECTURE



canoo

ALL THE COOL PEOPLE TALK ABOUT

MVVM ARCHITECTURE



canoo

USE IT TODAY WITH THE HELP OF

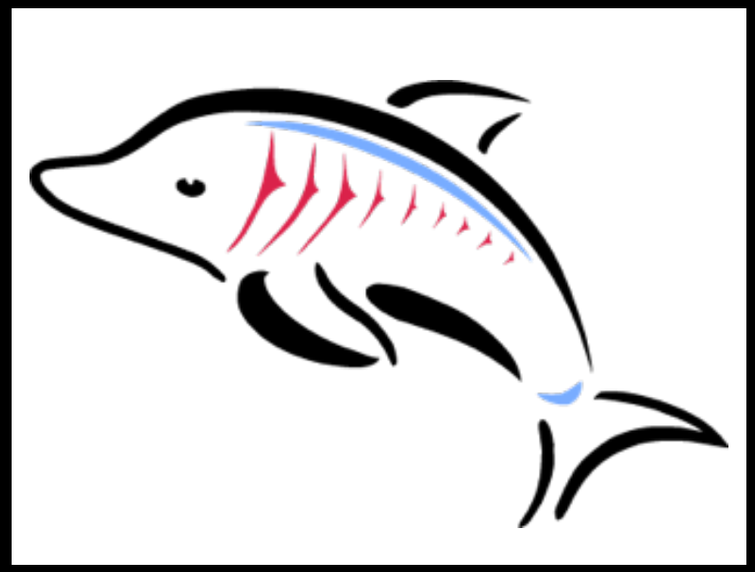
MVVM FRAMEWORKS

ANKOR.IO

MVVMFX

ANOTHER OPTION IS AN

PMVC LIBRARY



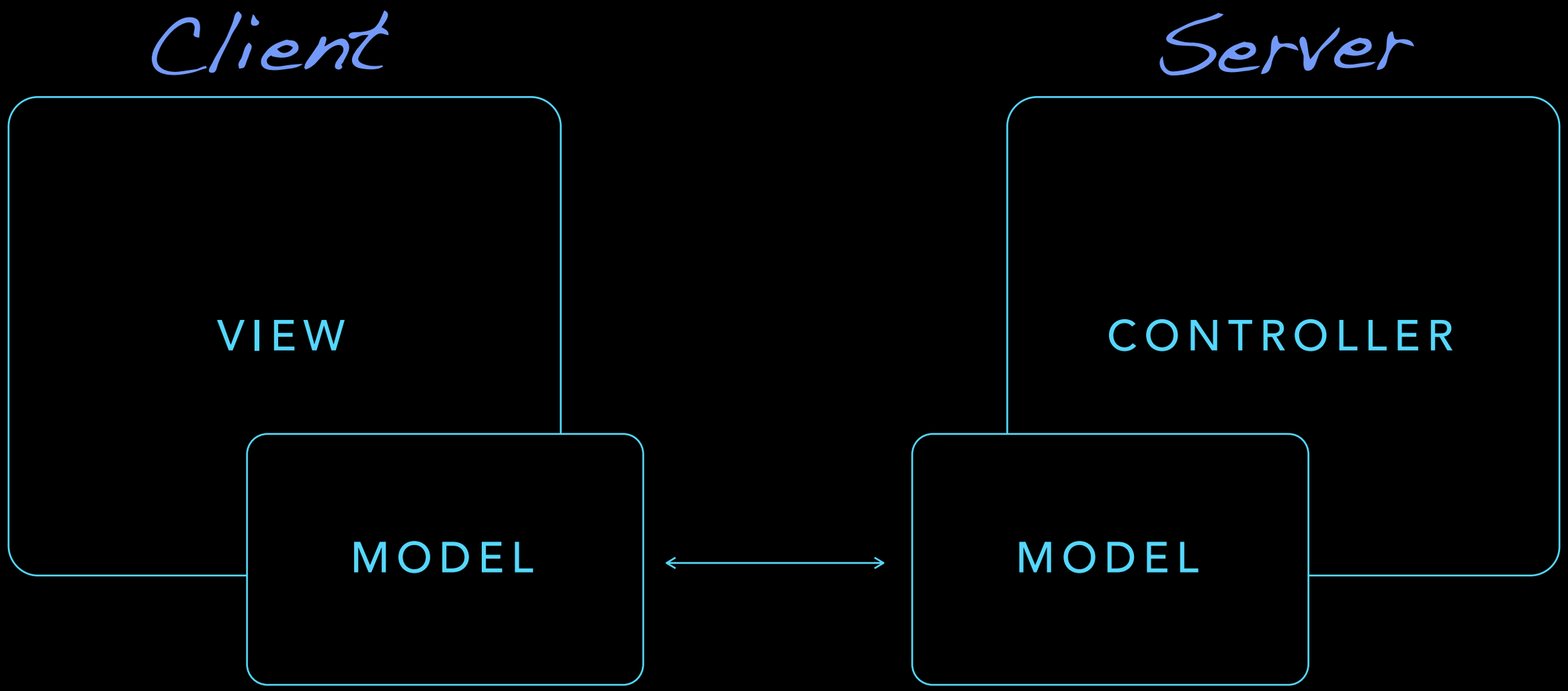
OPENDOLPHIN

an architecture for the communication between view and controller in an async [remote] fashion with presentation models.

canoo

SHARED

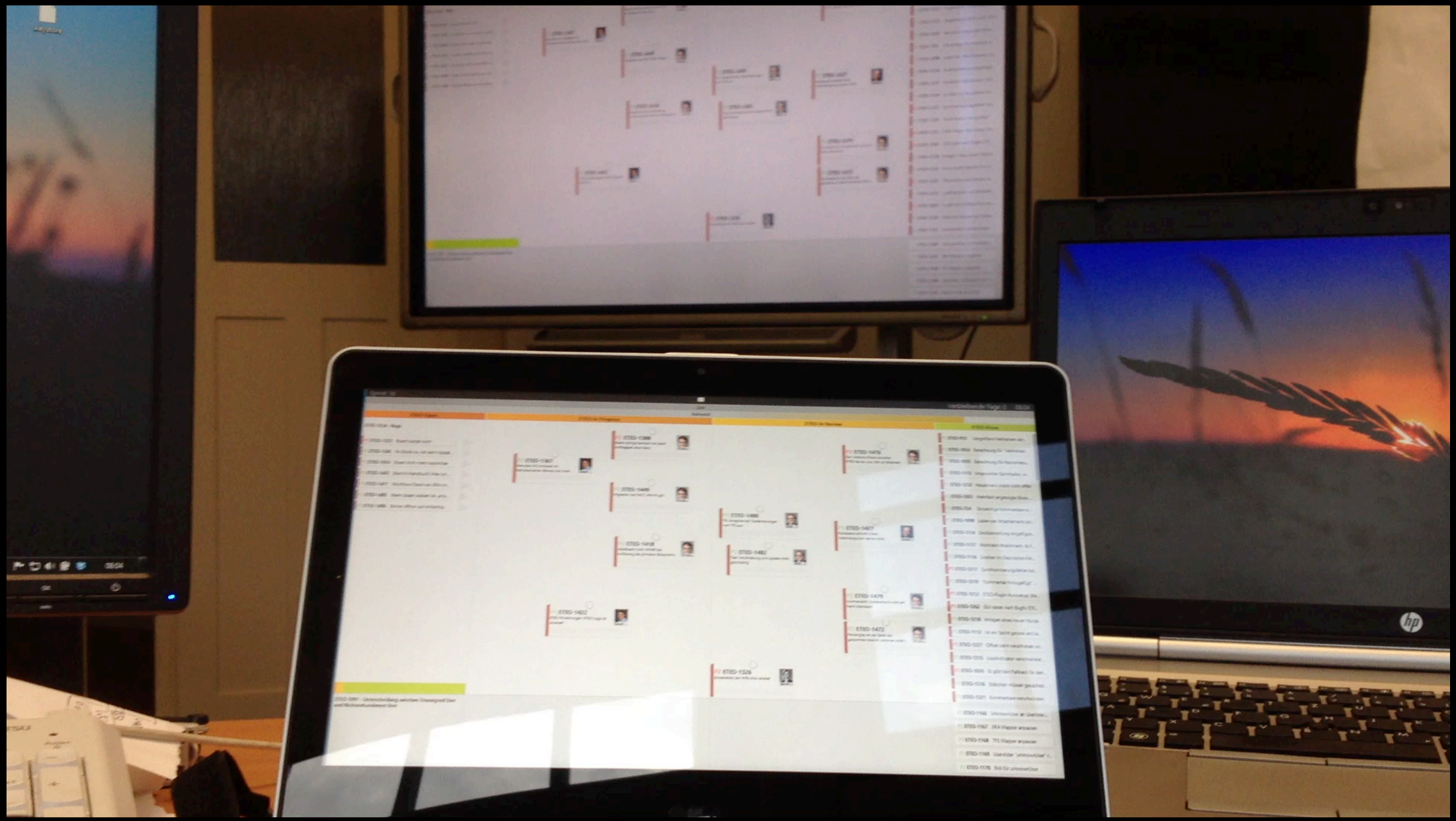
PRESENTATION MODEL



canoo

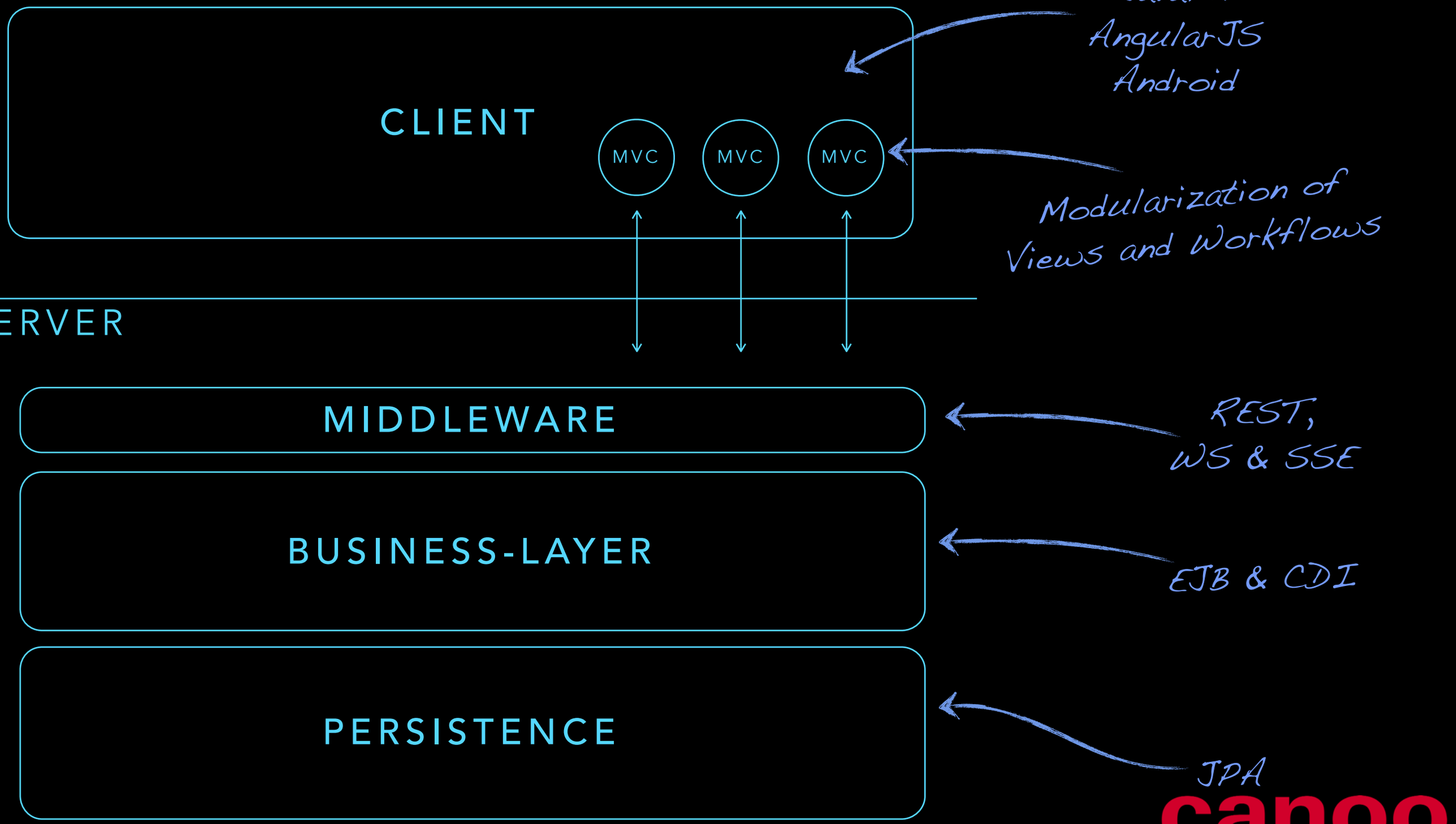
SHARE IT ON

MULTIPLE CLIENTS



BUT I HAVE A JAVA ENTERPRISE BACKEND AND NEED A

CLASSIC ARCHITECTURE



canoo

USE ONE OF THESE

FRAMEWORKS

DATAFX

afterburner.fx

AFTERBURNER.FX

JACPFX

JACPFX

canoo

OVERVIEW

AFTERBURNER.FX

- apache licensed
- as lean as possible: 3 classes, no external dependencies
- combines: FXML, Convention over Configuration and JSR-330 / @Inject
- integrated with maven 3

canoo

OVERVIEW

DATAFX

- Core API with concurrency tools
- DataReader API for fast & easy data access
  - REST, SSE, WebSocket, …
- Flow API to create workflows
- Injection API

canoo

MOST OF THEM PROVIDE

COOL FEATURES

- Application Framework for JavaFX
- Supports JEE Middleware standards

REST
WebSocket
RemoteEJB

- MVC Concept
- Implement Workflows by Flows

Like in JSF 2.2 or
Spring Flow

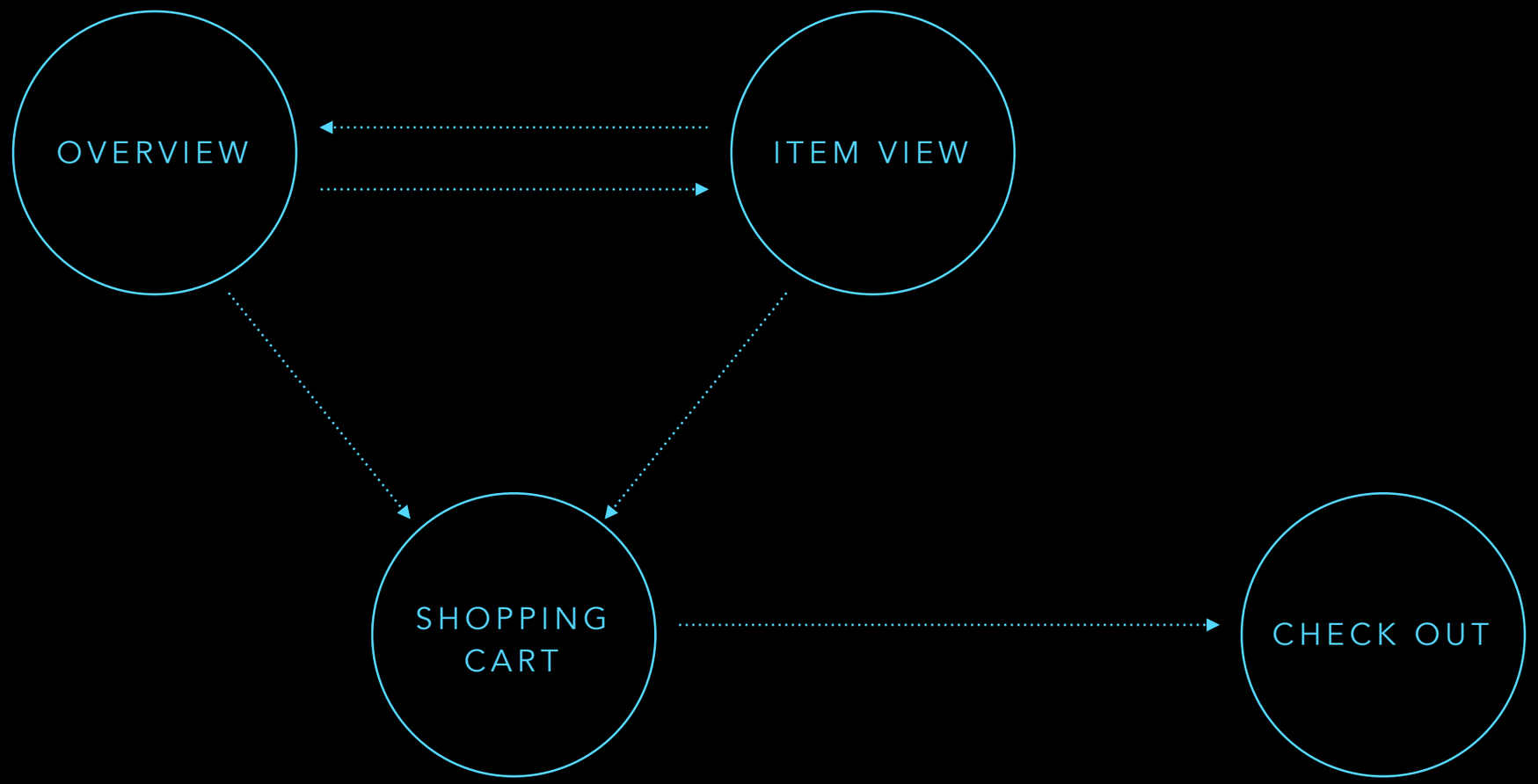
- (C)DI Support

Inject the data model
in the view controller

canoo

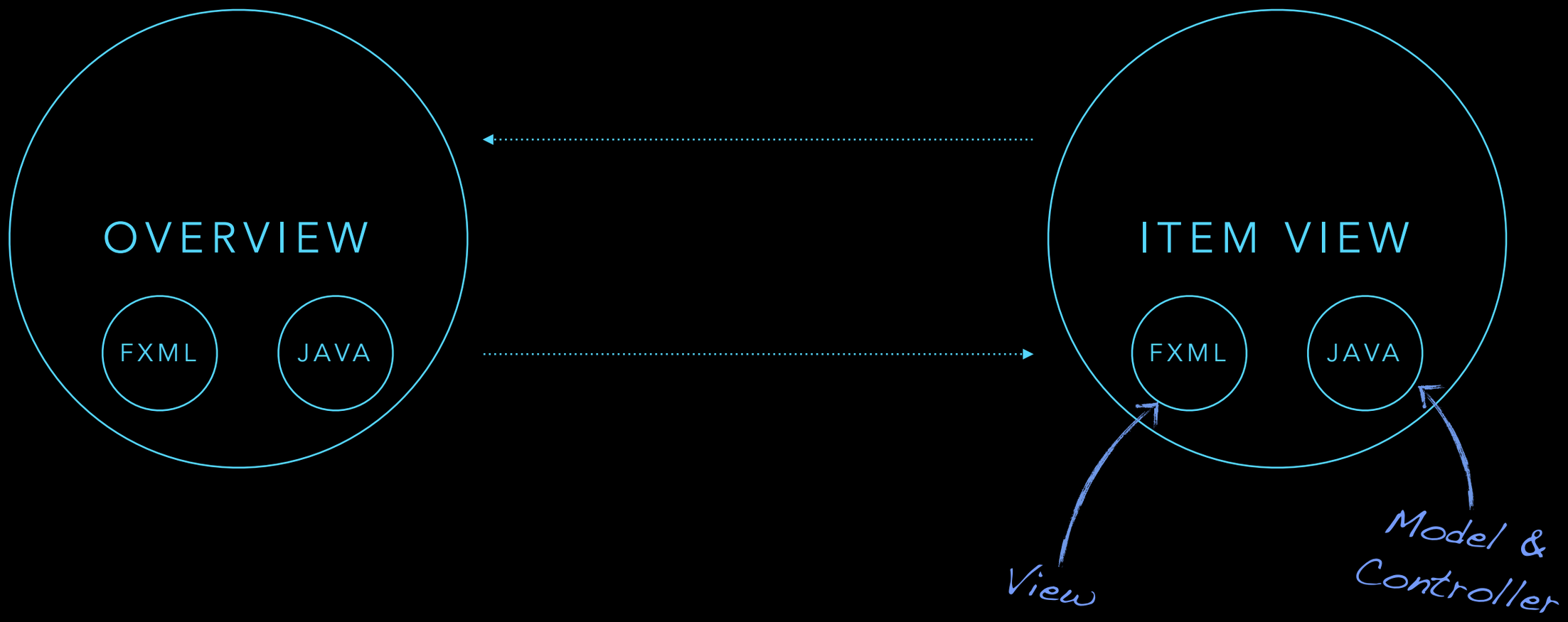
START CREATING AN

APPLICATION FLOW



WE CAN USE THE

MVC PATTERN



DEMO

CURRENT STATE OF THE

(C)DI IMPLEMENTATIONS

- Afterburner.fx and DataFX supports injection
- Afterburner.fx supports the singleton scope
- Afterburner.fx provides injection of properties
- DataFX supports different scopes
  (dependent, view, flow, application)
- DataFX provides a plugin mechanism for custom scopes

canoo

THE DARK SIDE OF

CDI IMPLEMENTATIONS

- Qualifier, etc. are currently not supported
- No default CDI implementation is used

Hackergarten?

- CDI-Spec is currently Java EE specific

We don't need a RequestScope

- Weld and OpenWebBeans core modules are Java EE specific

lots of http apis inside...

canoo

THERE ARE COOL

BUSINESS CONTROLS

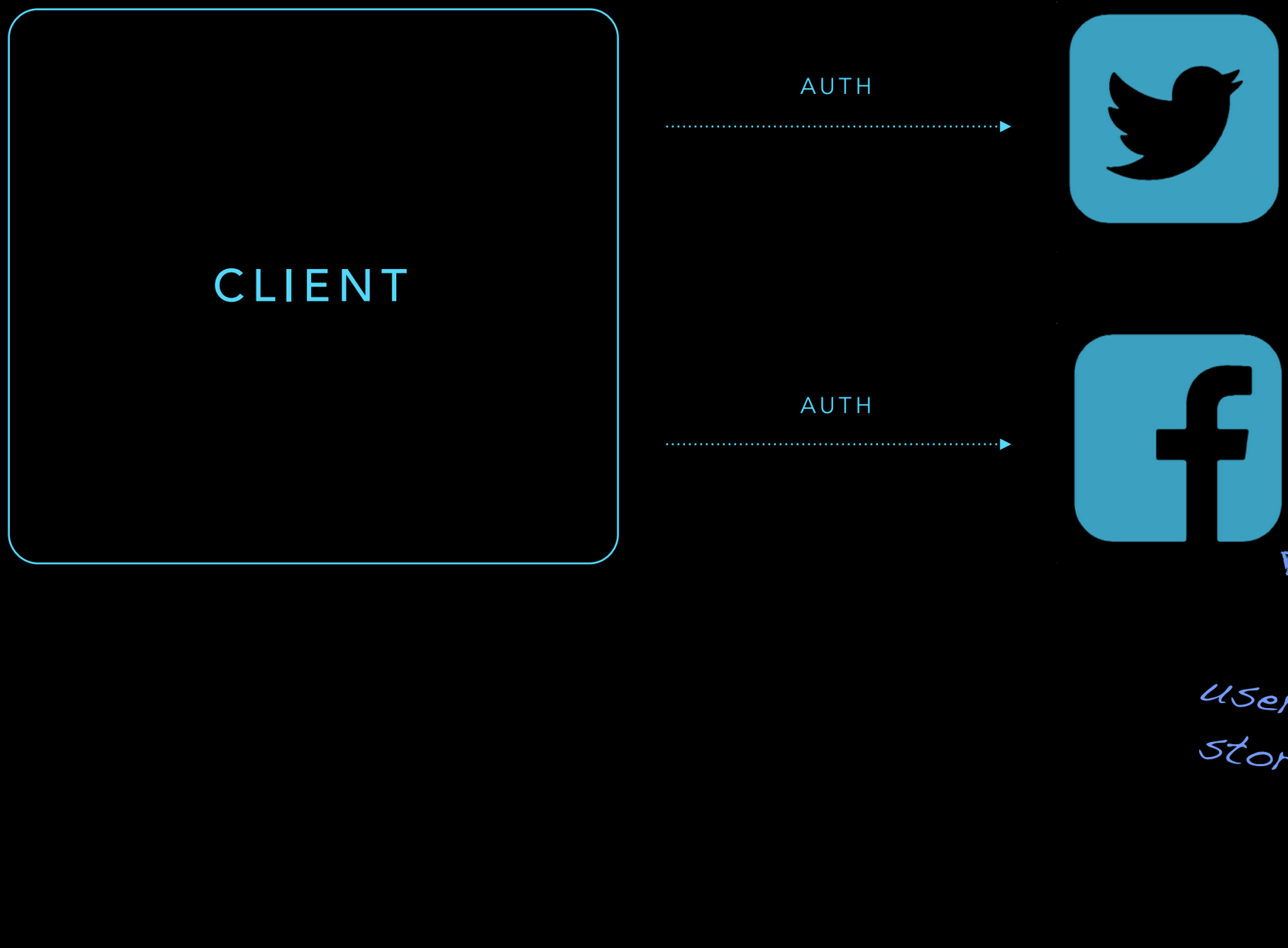
- The community created a lot of cool controls
- ControlsFX
- FlexGanttFX

canoo

DEMO

SHIT! MY HIP CUSTOMER WANTS TO

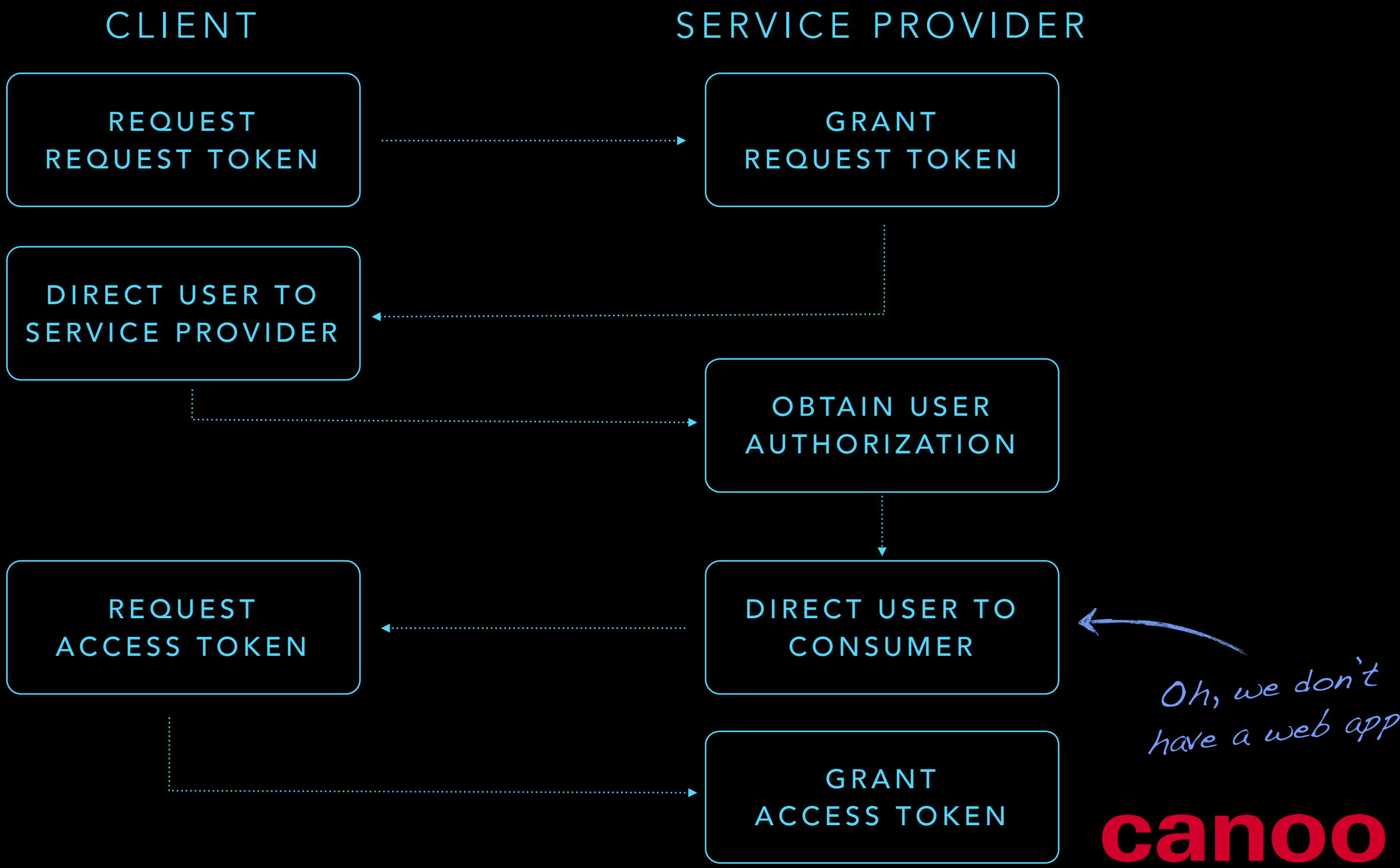
LOGIN WITH TWITTER



canoo

WEB APPS SIMPLY IMPLEMENT THE

OAUTH WORKFLOW

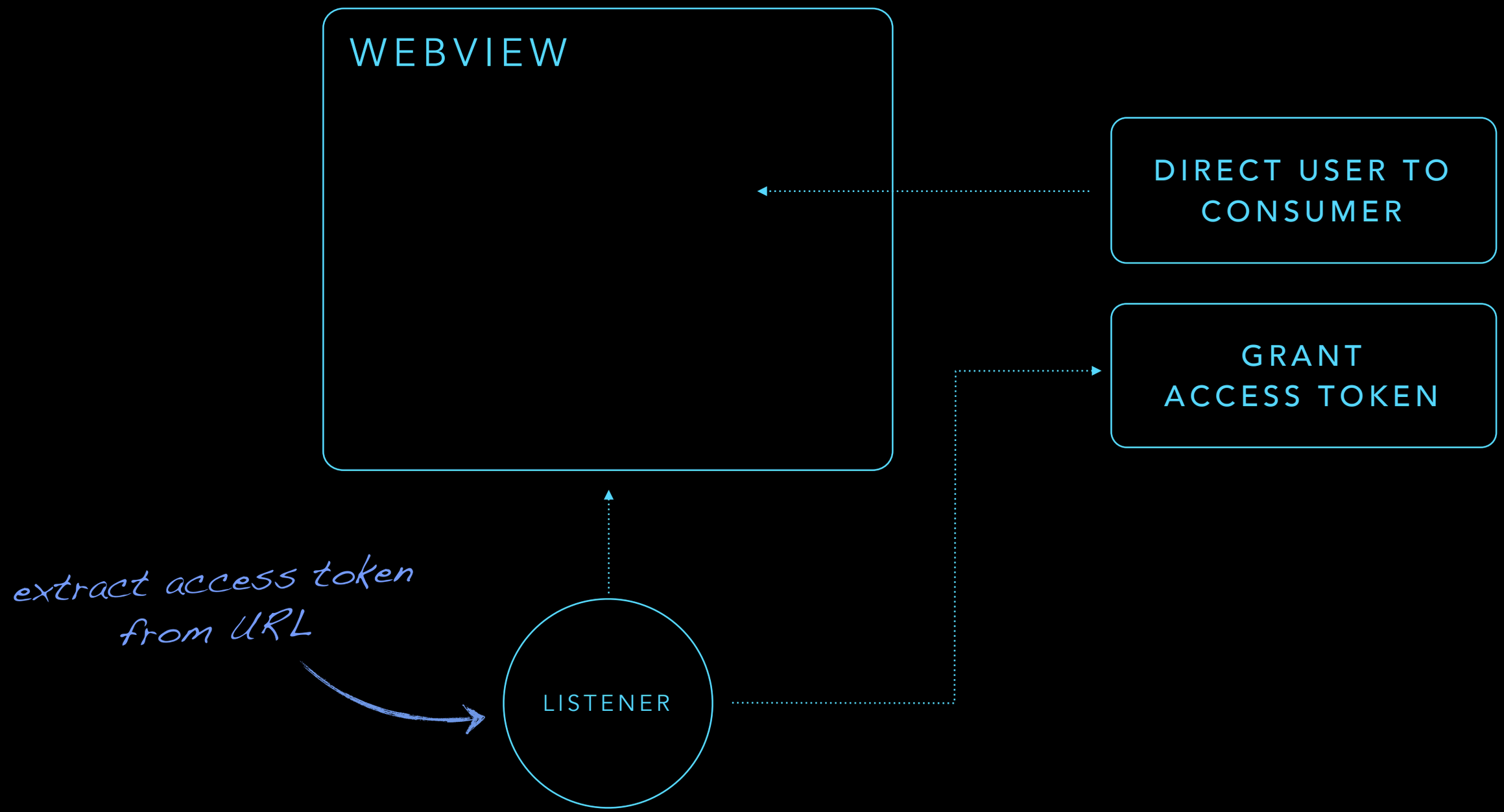


canoo

DEMO

BUT JAVAFX CAN

CONTROL THE WEBVIEW



canoo

INTRODUCING A

GUIGARAGE MODULE

```
<dependency>
  <groupId>com.guigarage</groupId>
    <artifactId>login-with-fx</artifactId>
    <version>X.Y</version>
</dependency>
```

canoo

THERE ARE MORE NEW

GUIGARAGE MODULES

Extreme Gui Makeover

WINDOW STYLER

ANIMATIONS

Extreme Gui Makeover

Smart UIs for Mobile and Embedded

FLATTER

MATERIAL DESIGN

Extreme Gui Makeover

RESPONSIVE-FX

GUI Garage

OPEN SOURCE UI STUFF

canoo

THX FOR WATCHING

QUESTIONS?

canoo