MVC

Mozart Violin Concerto vs. Model-View-Controller

DON BURKS

Head Instructor, Lighthouse Labs
Vancouver, BC
@don_burks
https://donburks.com
don@lighthouselabs.ca

Also Don Burks...

BMA Musical Arts (Performance)

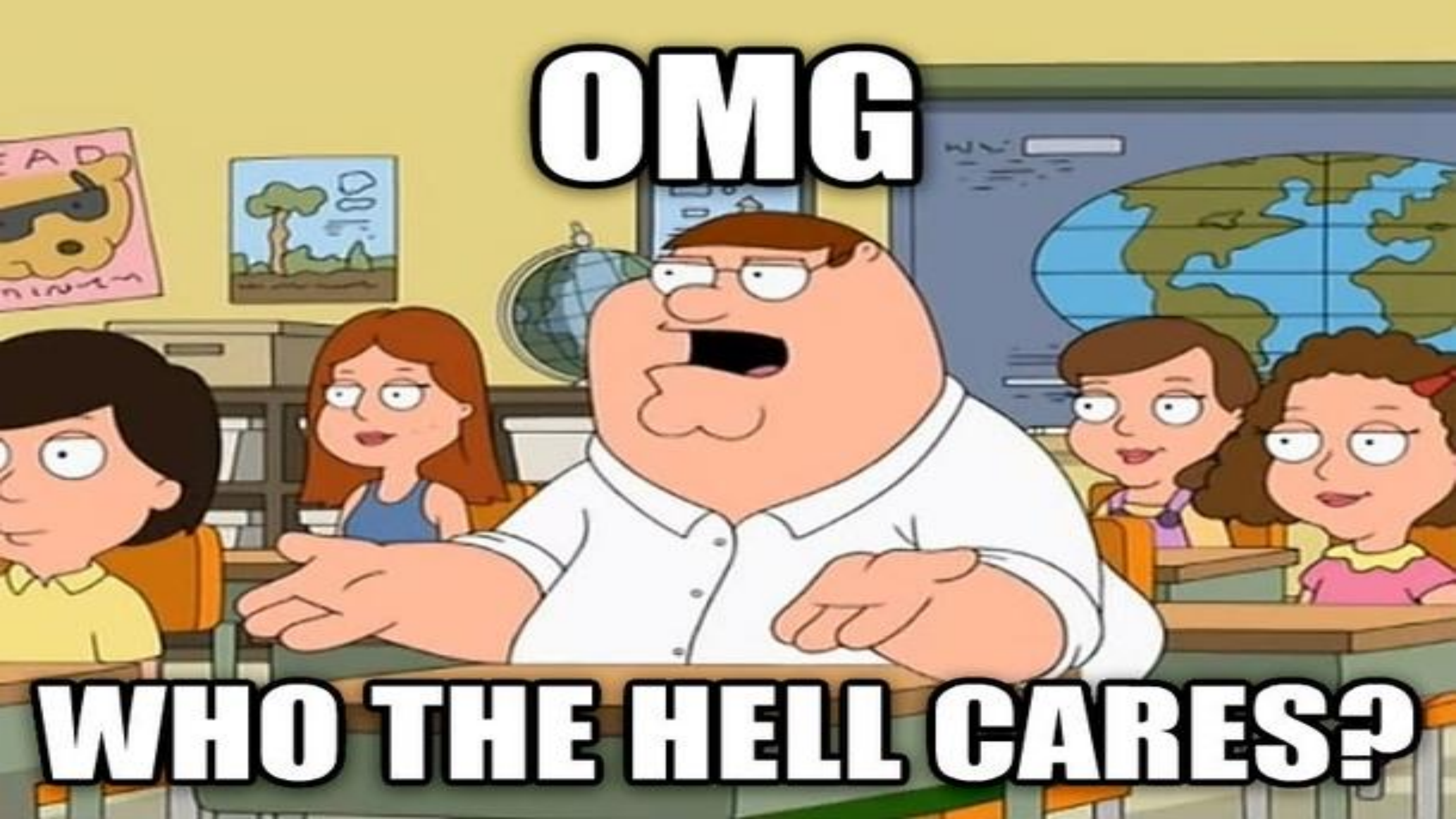
Columbus State University

1996

COLUMBUS STATE UNIVERSITY ®

Obligatory GIF

And now, for some **culture**.


OMG
WHO THE HELL CARES?

We should care.

Because musicians and developers have to think similarly.

Users care.

Because both software and music are a craft.

You should care.

Because approach is **everything**.

I should care.

Because I have to give a **talk** on it.

Wolfgang Amadeus Mozart

1756 - 1791

A few relevant numbers

By the age of **36**, Mozart’s body of work included...

41 Symphonies

27 Piano Concertos

5 Violin Concertos

27 Concert Arias

23 String Quartets

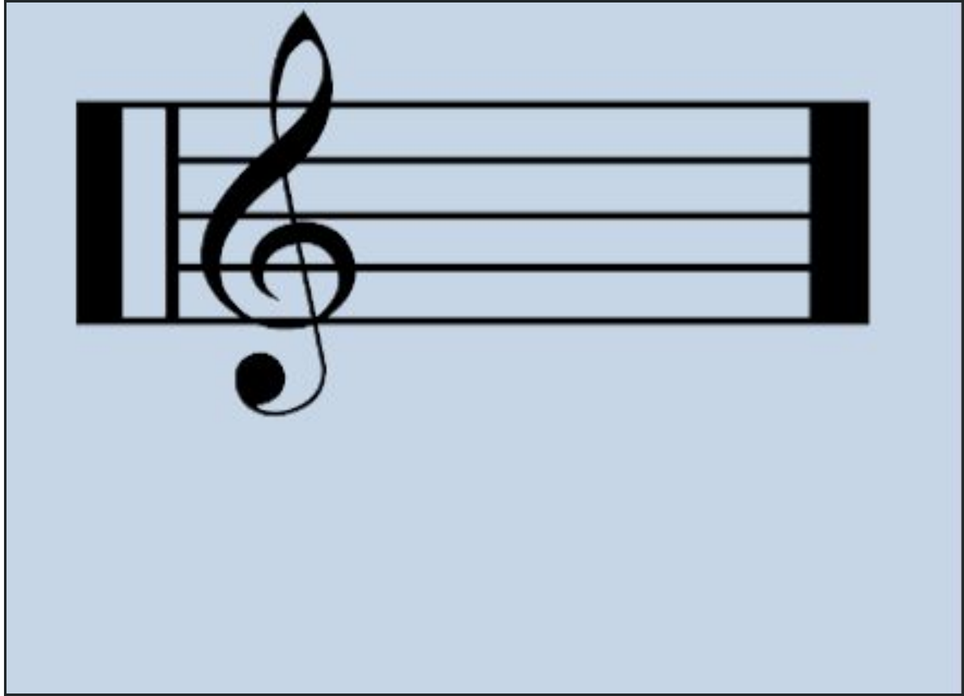
4 Horn Concertos

18 Masses

22 Operas

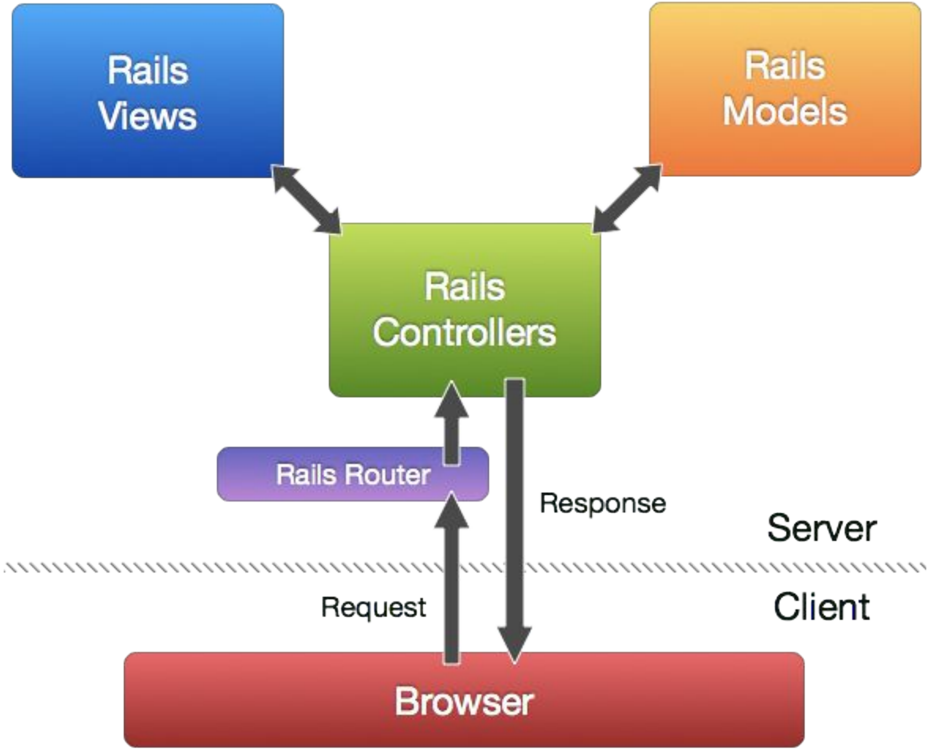
I’m 43, and I managed to put on pants.

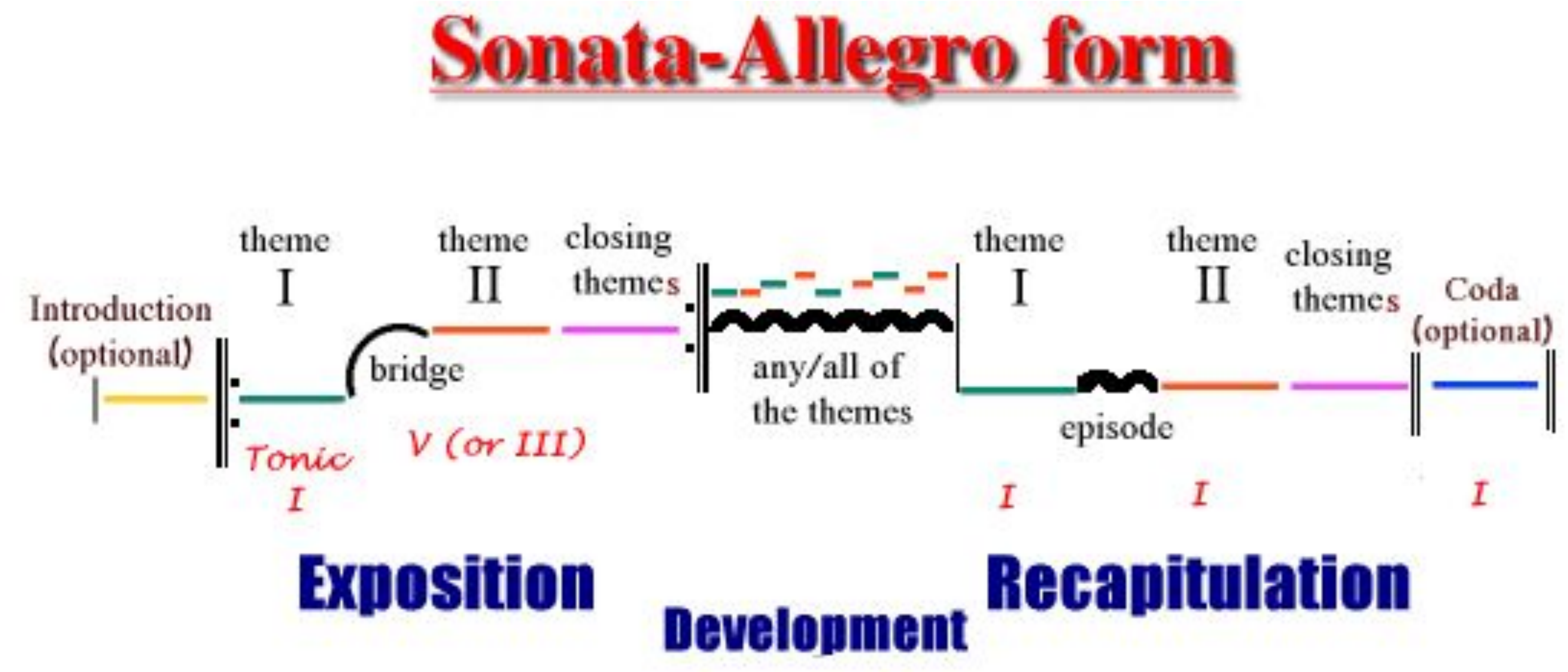
But what does Mozart have to do with code?



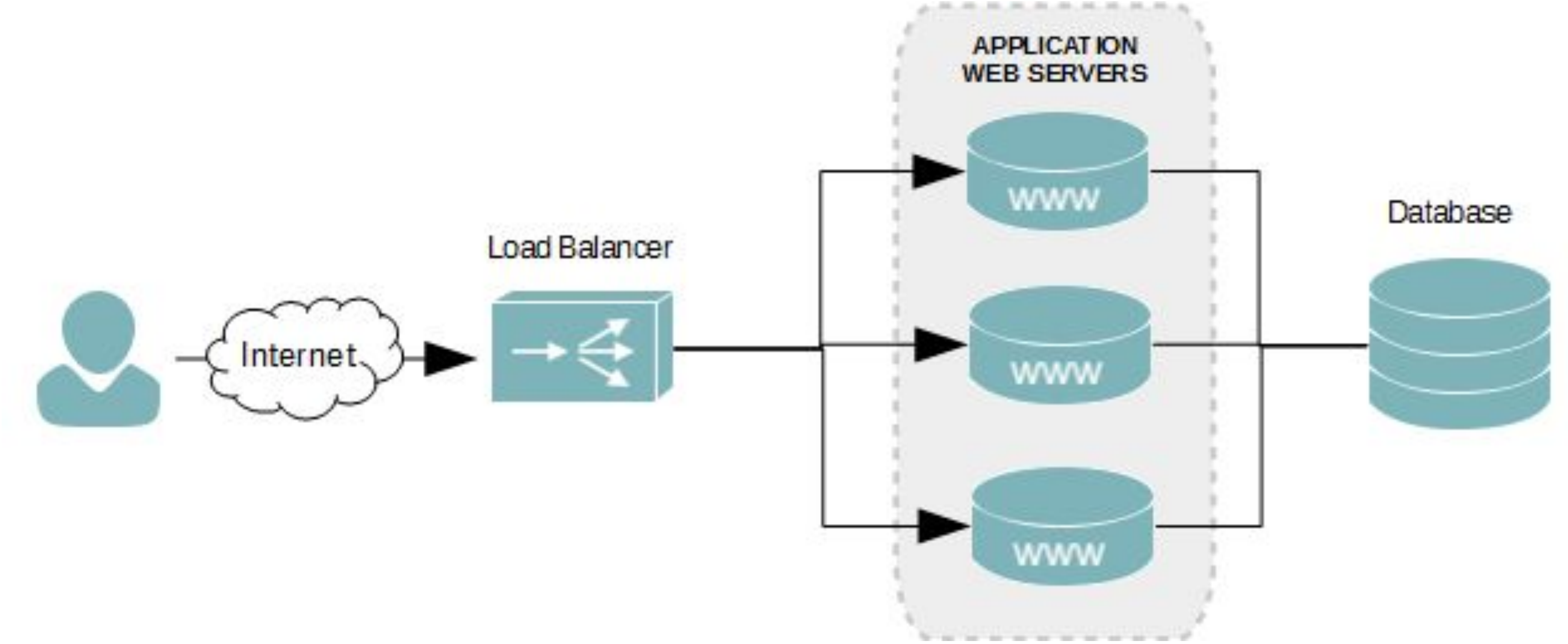
Structure

Apps have structure



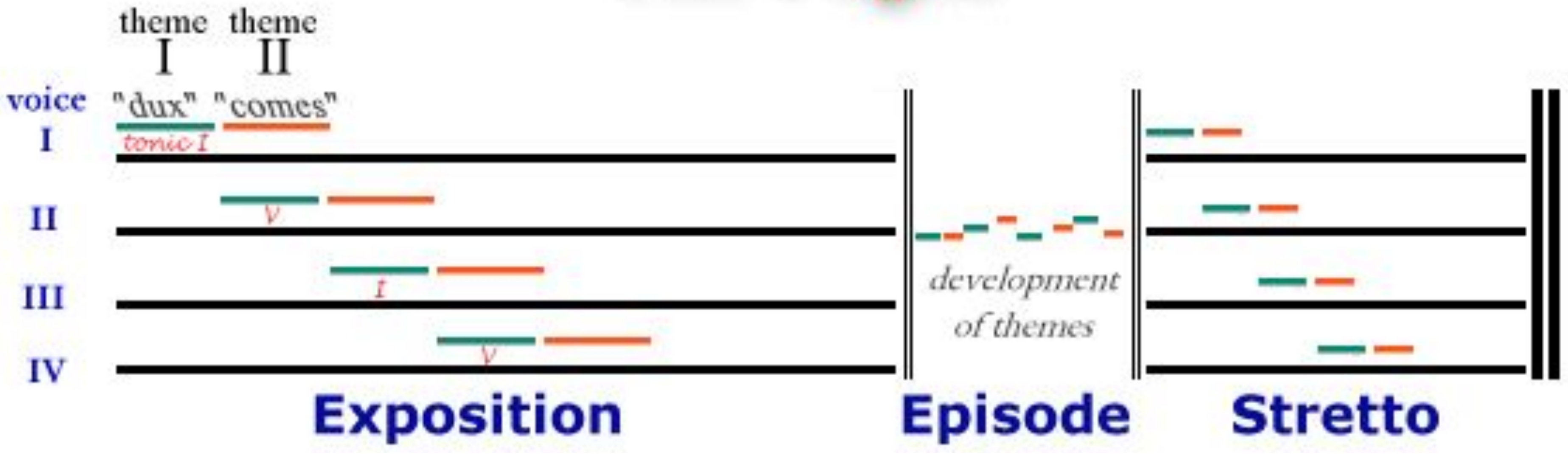


A Violin Concerto has **structure**, too.



We envision our applications as whole systems...

The Fugue



Composers have been doing this for centuries.

Approach

Horizontal

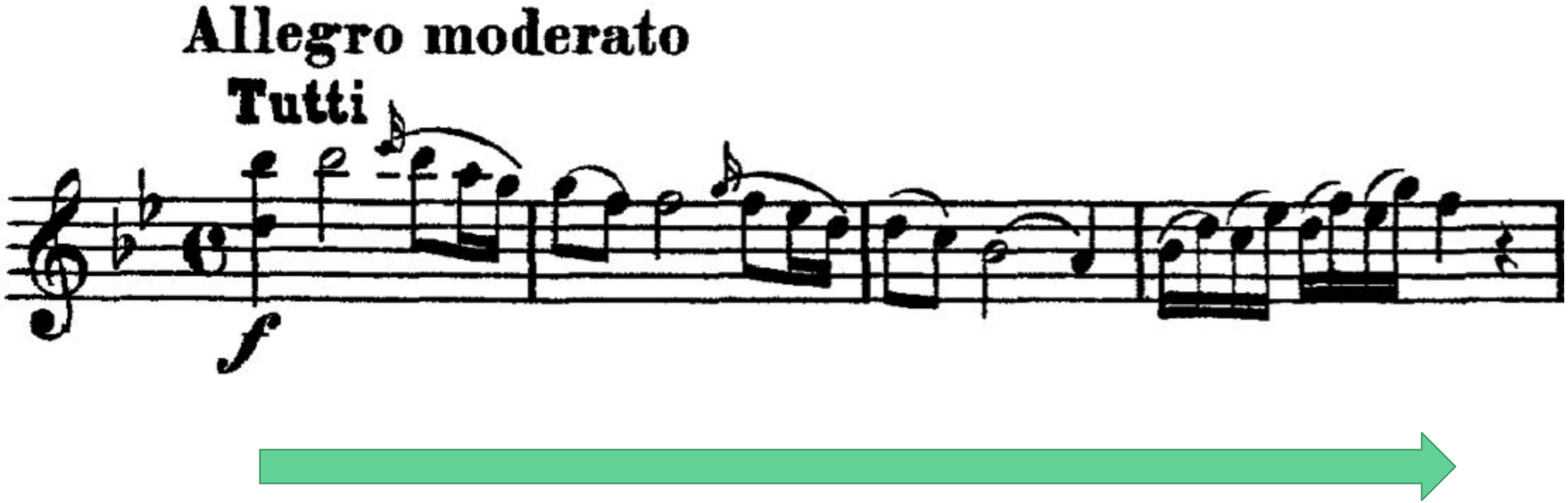
Lateral Thinking

Edward de Bono (1967)

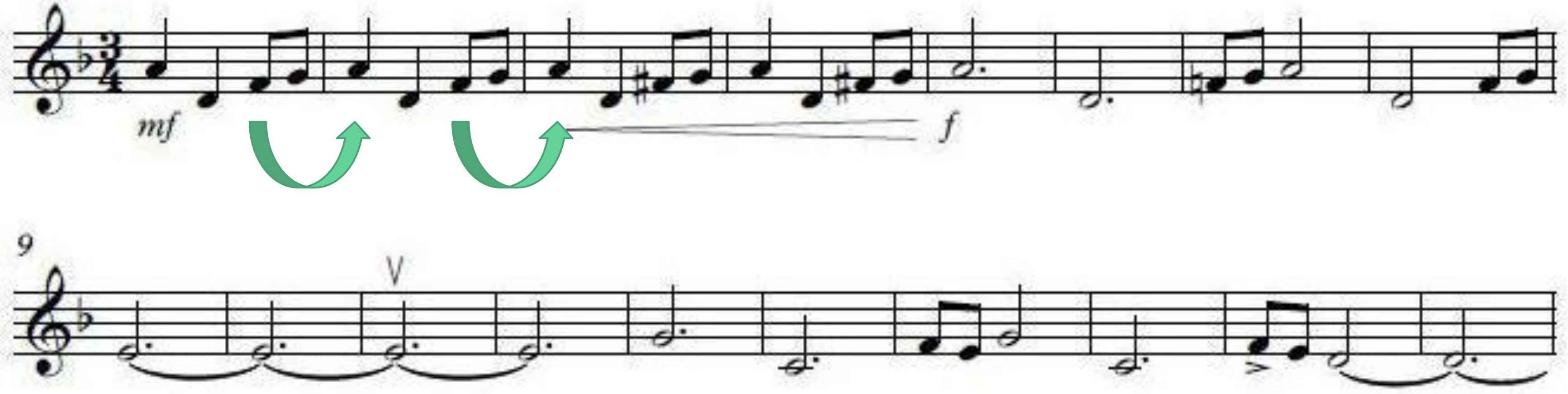
Lateral thinking is solving problems through an indirect and creative approach, using reasoning that is not immediately obvious and involving ideas that may not be obtainable by using only traditional step-by-step logic.

★ Wikipedia

Horizontal Thinking in Musicians

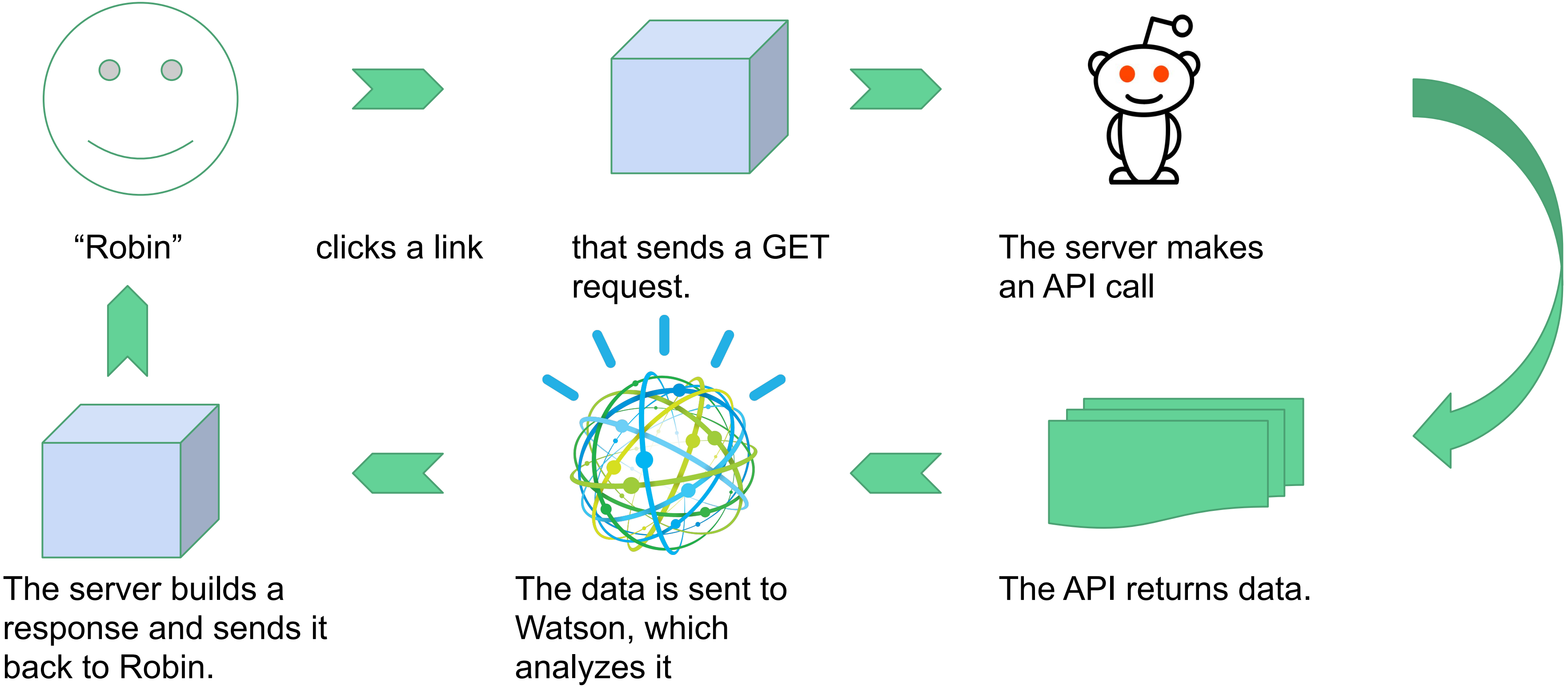


Melody moves the music forward



> “Algorithms are the melodies of an application.”
>
> - *A wise, humble man*

Horizontal Thinking in Developers



“Robin”

clicks a link

that sends a GET request.

The server makes an API call

The API returns data.

The data is sent to Watson, which analyzes it

The server builds a response and sends it back to Robin.

FizzBuzz

Great example!

Make a list of numbers from 1 - 100.

For each number:
Check to see if it is evenly divisible by 3. If yes, print “Fizz”.
Check to see if it is evenly divisible by 5. If yes, print “Buzz”.
Check to see if it is evenly divisible by both 3 **and** 5. If yes, print “FizzBuzz.”

Otherwise, just print the number.

Our code moves from one step to another. The thinking that fuels our solutions is **linear**, **lateral**, **horizontal**.

Vertical

Ember

Ruby on Rails

PostgreSQL

Docker

Linux

We always call it a “**stack**”, don’t we?

We have to code vertically as well as horizontally

Web applications, desktop applications, mobile apps, all have a particular necessity for vertical elements. Included, but not limited to:

- AJAX
- APIs
- Isomorphism
- Client-side caching
- Proxies / VPNs
- Deployments
- PWA

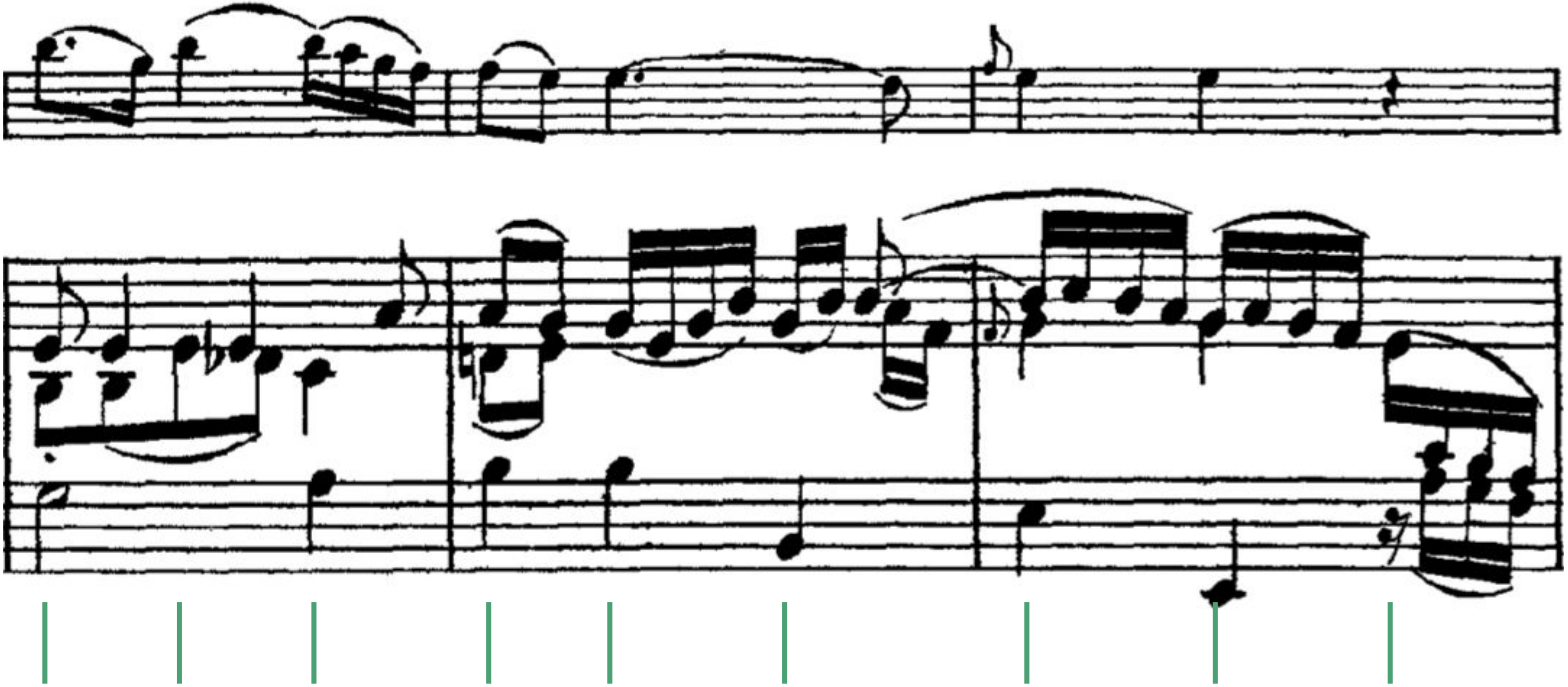
Our challenge...

...is to have our work be **ignored.**

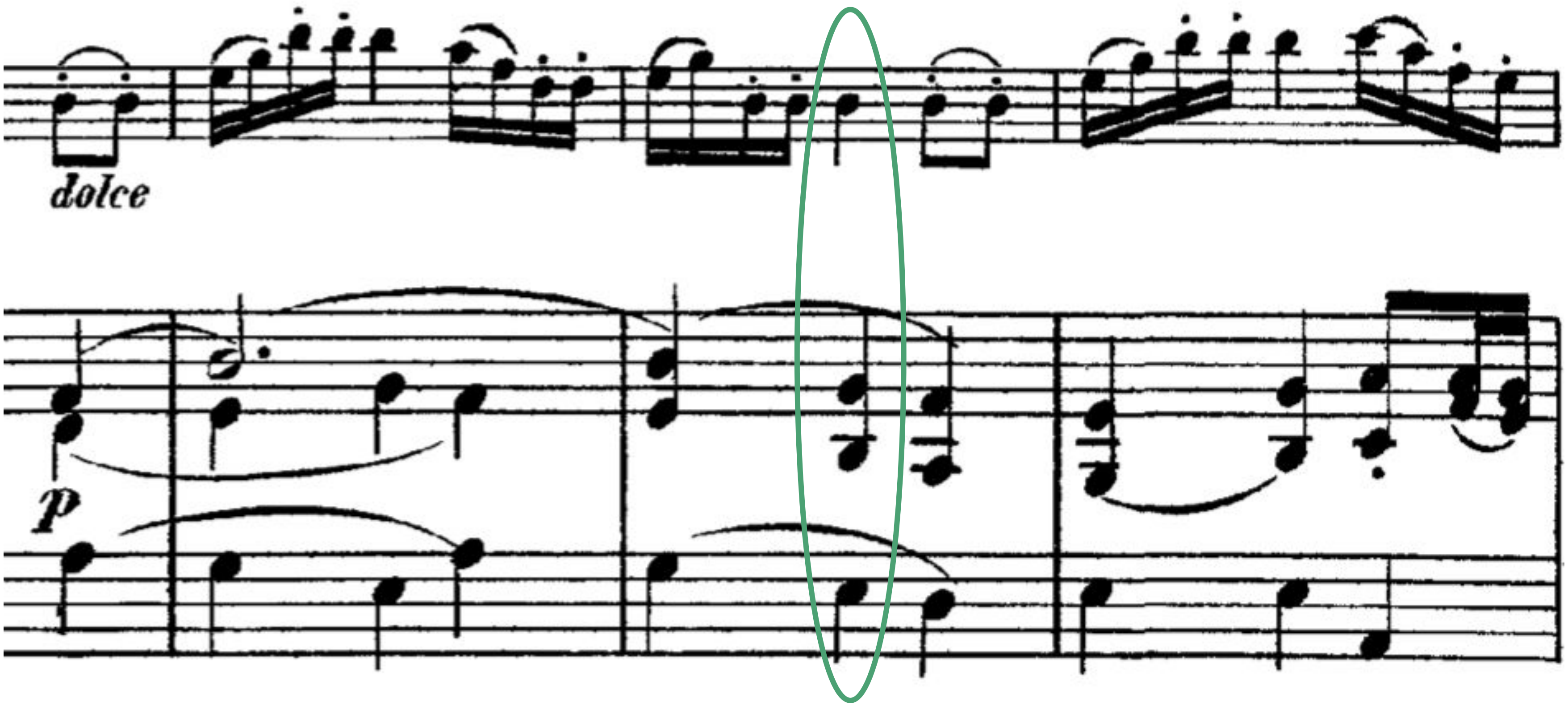
Our horizontal problem-solving and vertical architecture should be transparent to our users.

Musicians

Have it **just** as hard



Do the beats align with the accompaniment?



As a musician in an ensemble, am I in tune?

“In tune,
On time.”

But can we apply that **mentality** to development?

YES WE
CAN.

A good user experience is one where the app runs…

“In tune, on time.”

Musicians vs. Developers

Musicians tell a story.

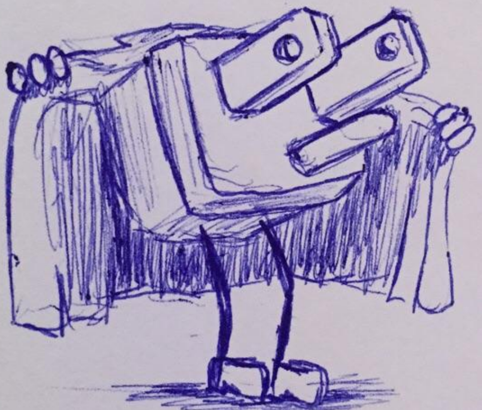
They decorate time instead of space.

A performer must string together notes into melodies, stacked on top of harmonies.

Developers offer a solution.

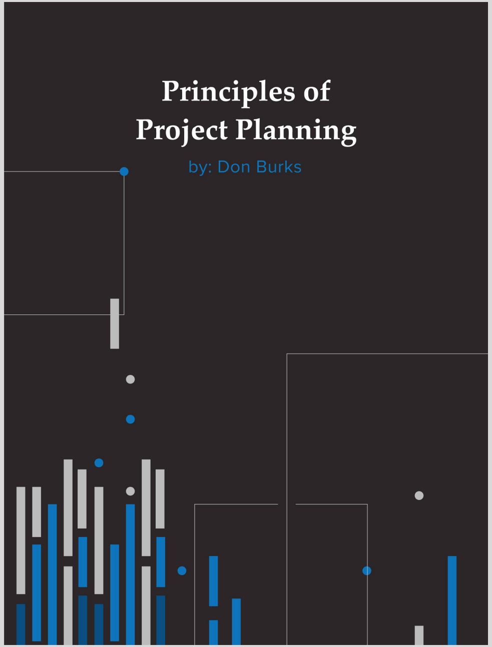
They give access to data, giving benefit through implementation.

An algorithm, implemented into an architecture, creates a product.



Shameless plug

BUY MY BOOK!



Principles of
Project Planning

by: Don Burks

gum.co/project_planning

Thank you.

@don_burks
https://donburks.com
don@lighthouselabs.ca