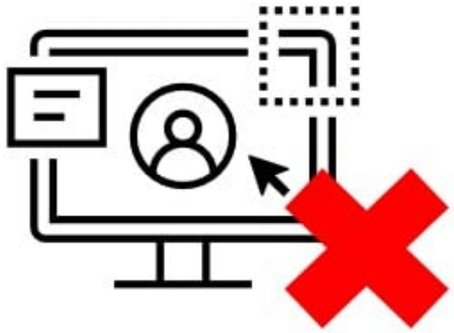
Serving modes

MLOPS DEPLOYMENT AND LIFE CYCLING

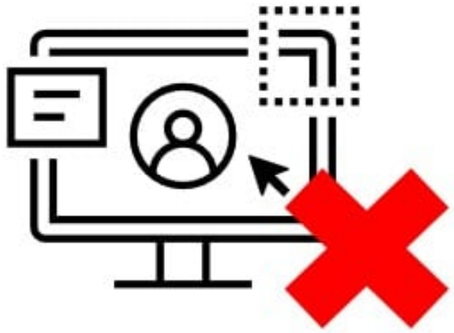
Nemanja Radojkovic
Senior Machine Learning Engineer

Model as a service

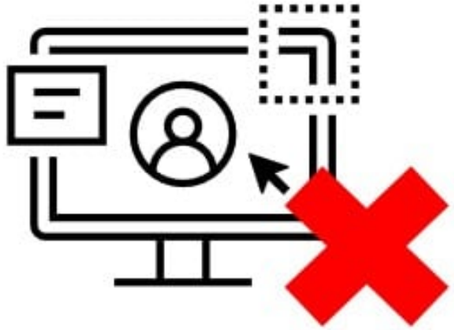
Model as a service

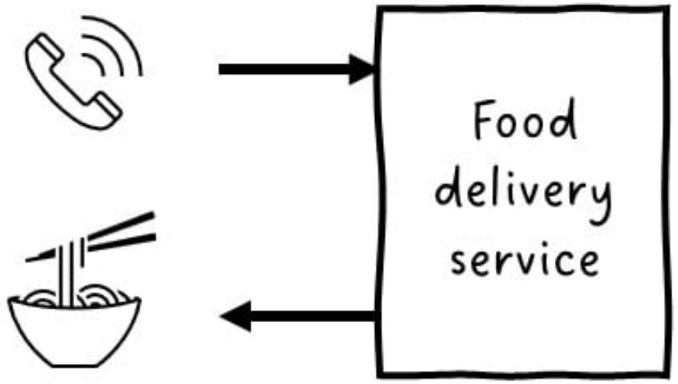


Model as a service

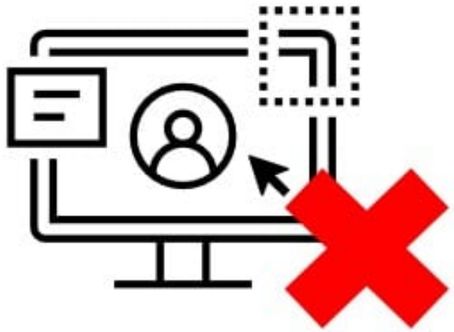


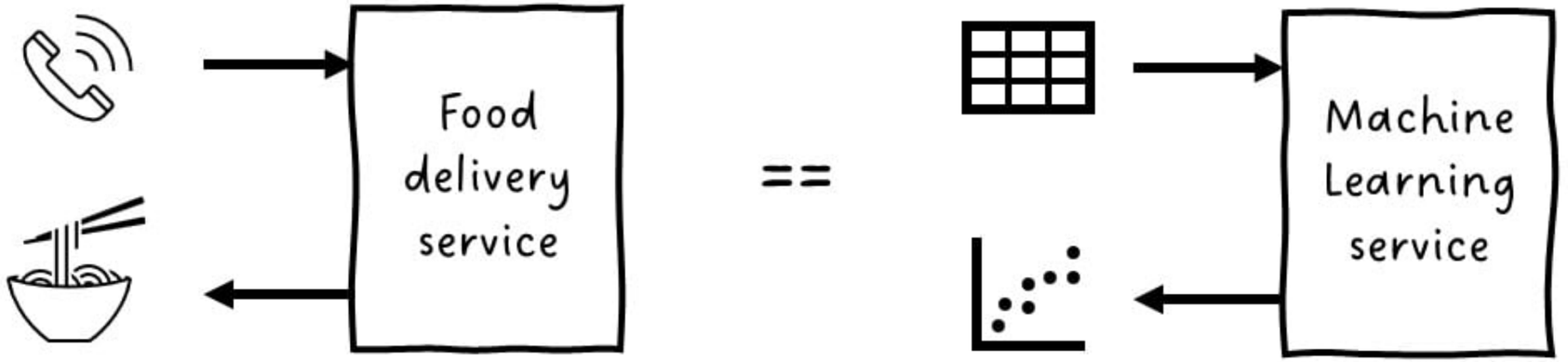
Model as a service





Model as a service

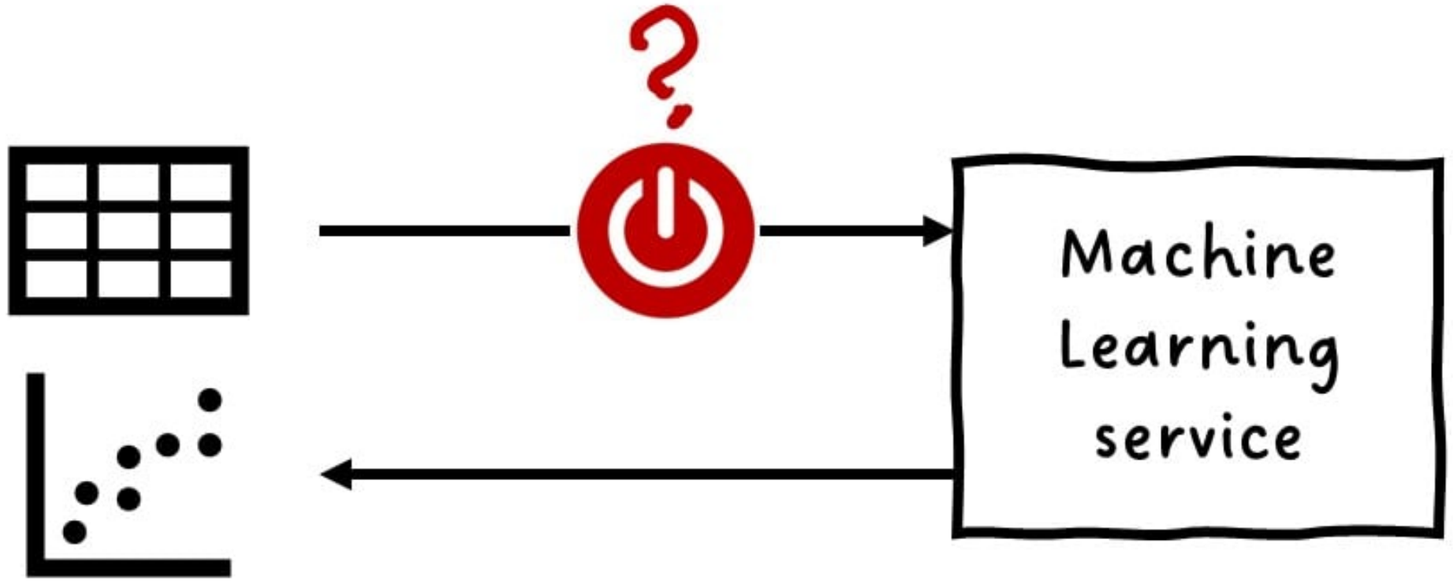


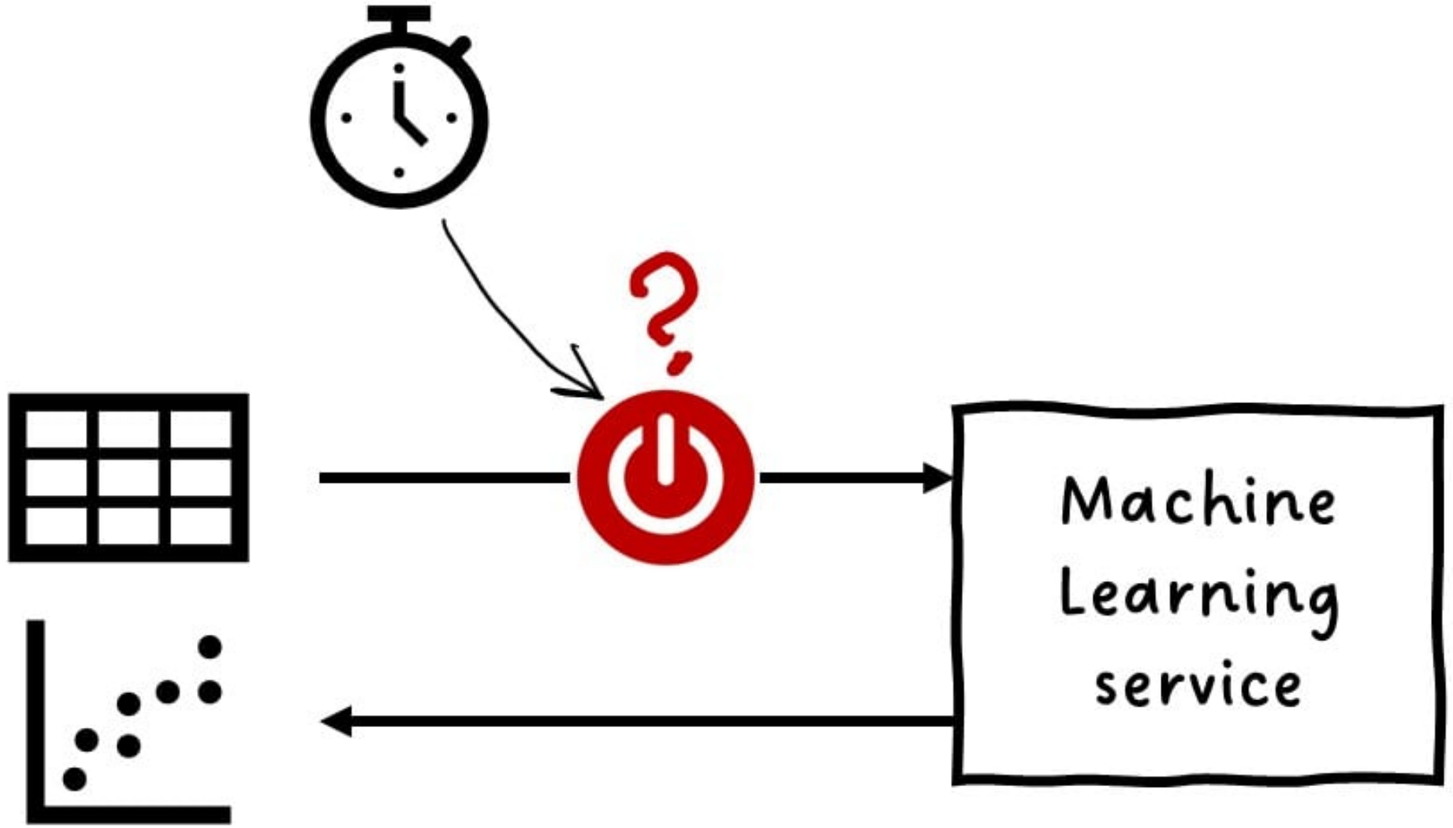


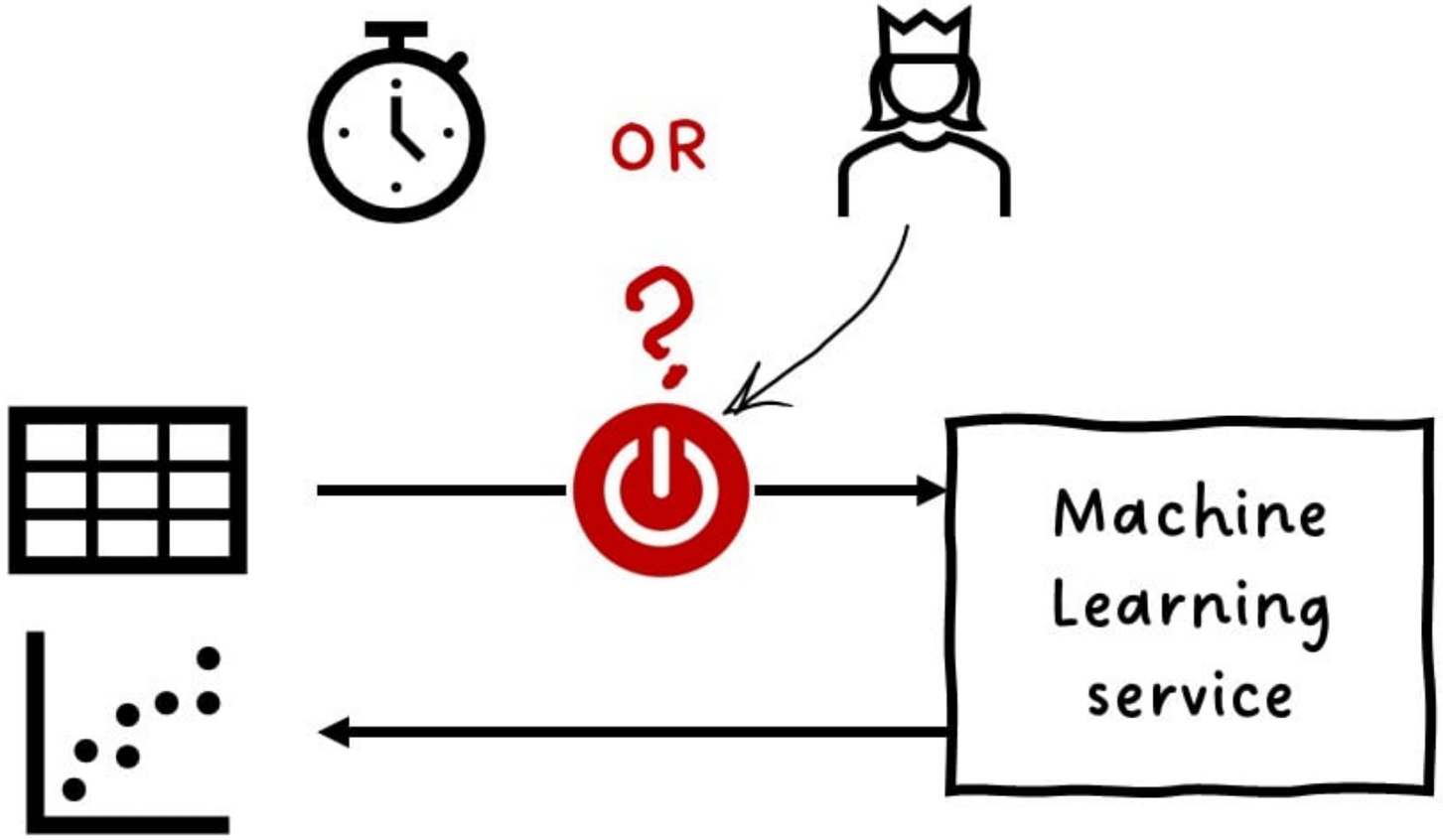
Serving and serving mode

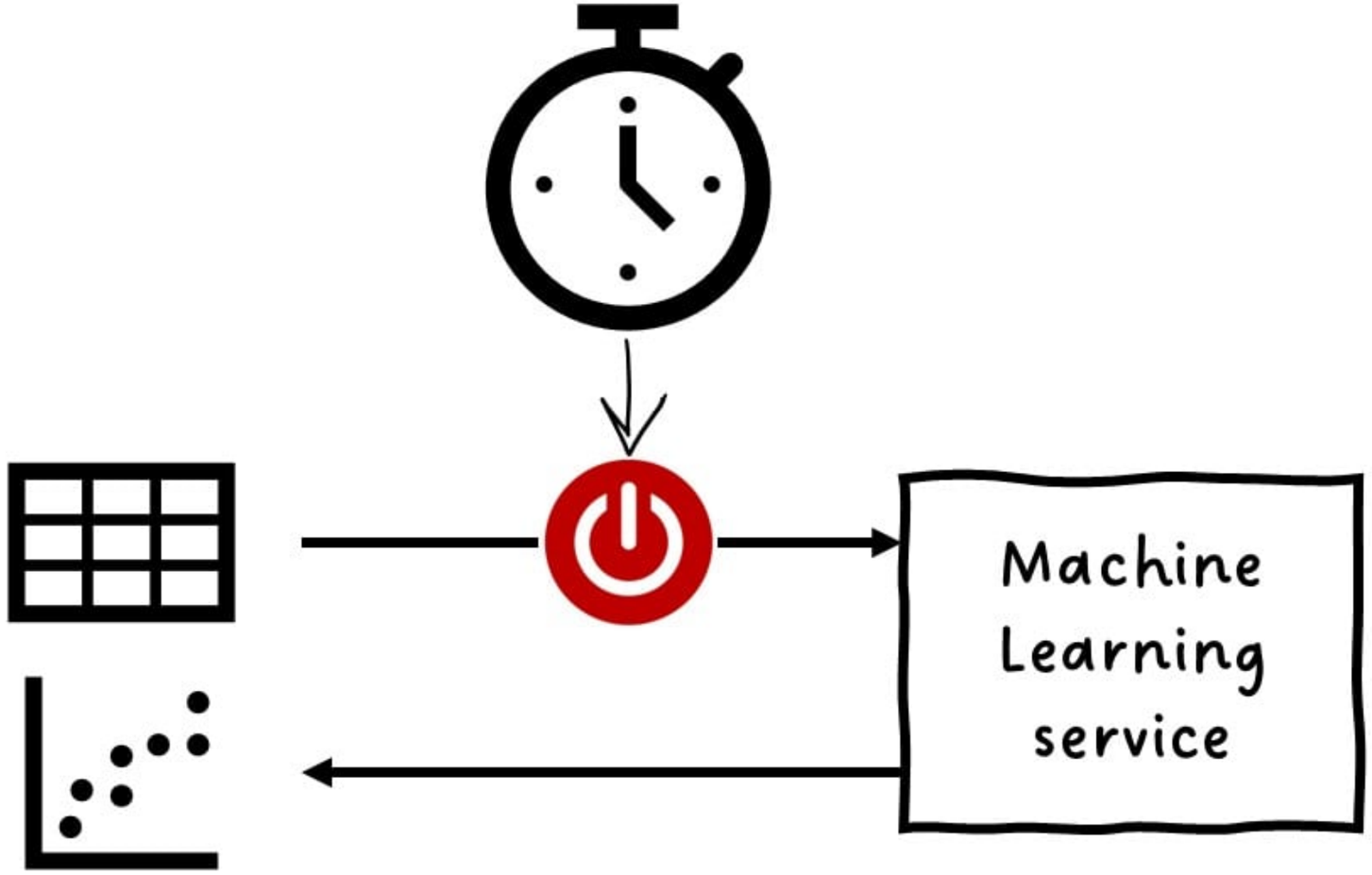
- Providing prediction service == Model serving
- Implementation of a specific type of serving == Serving mode

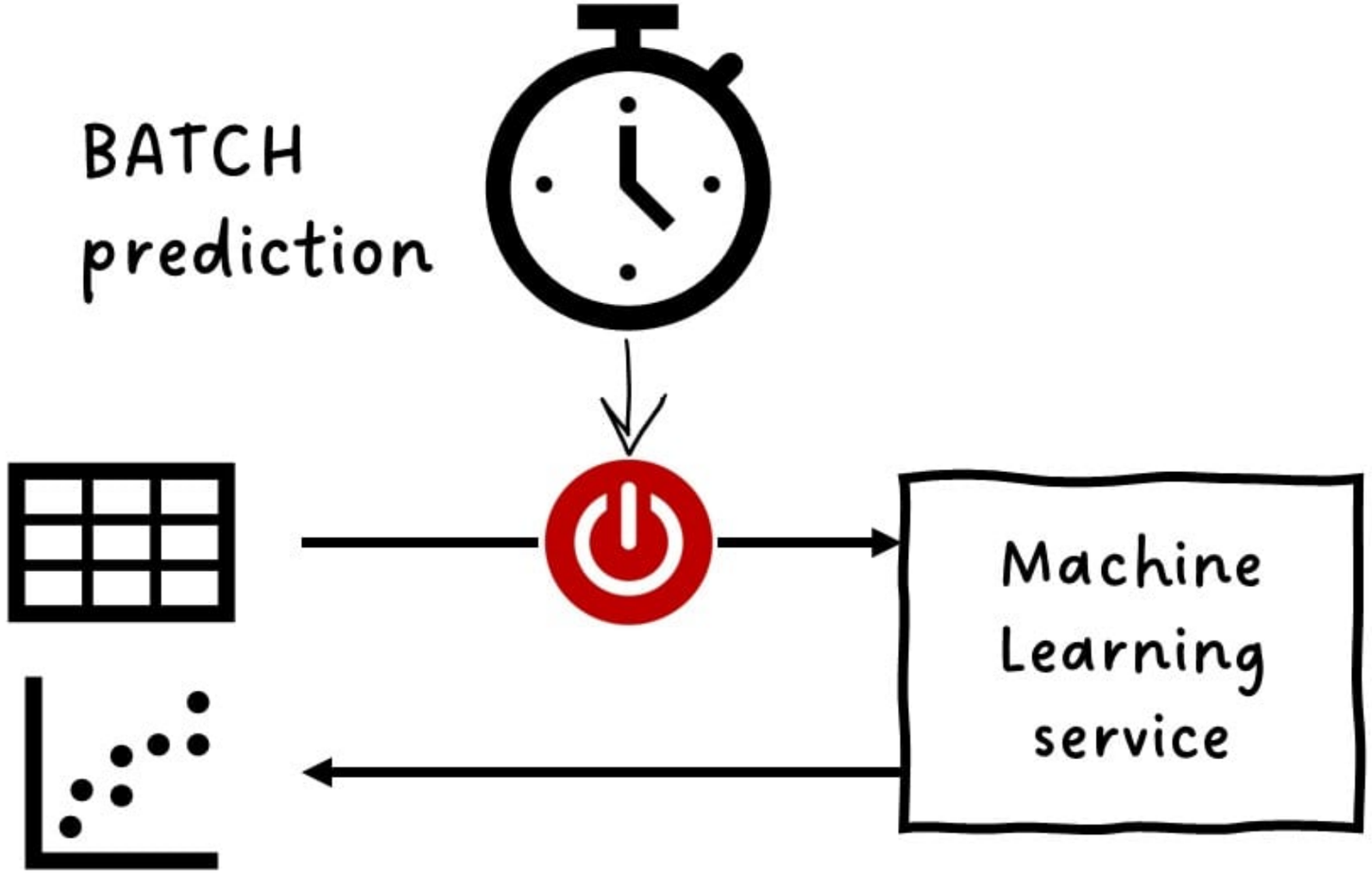
Choose carefully!

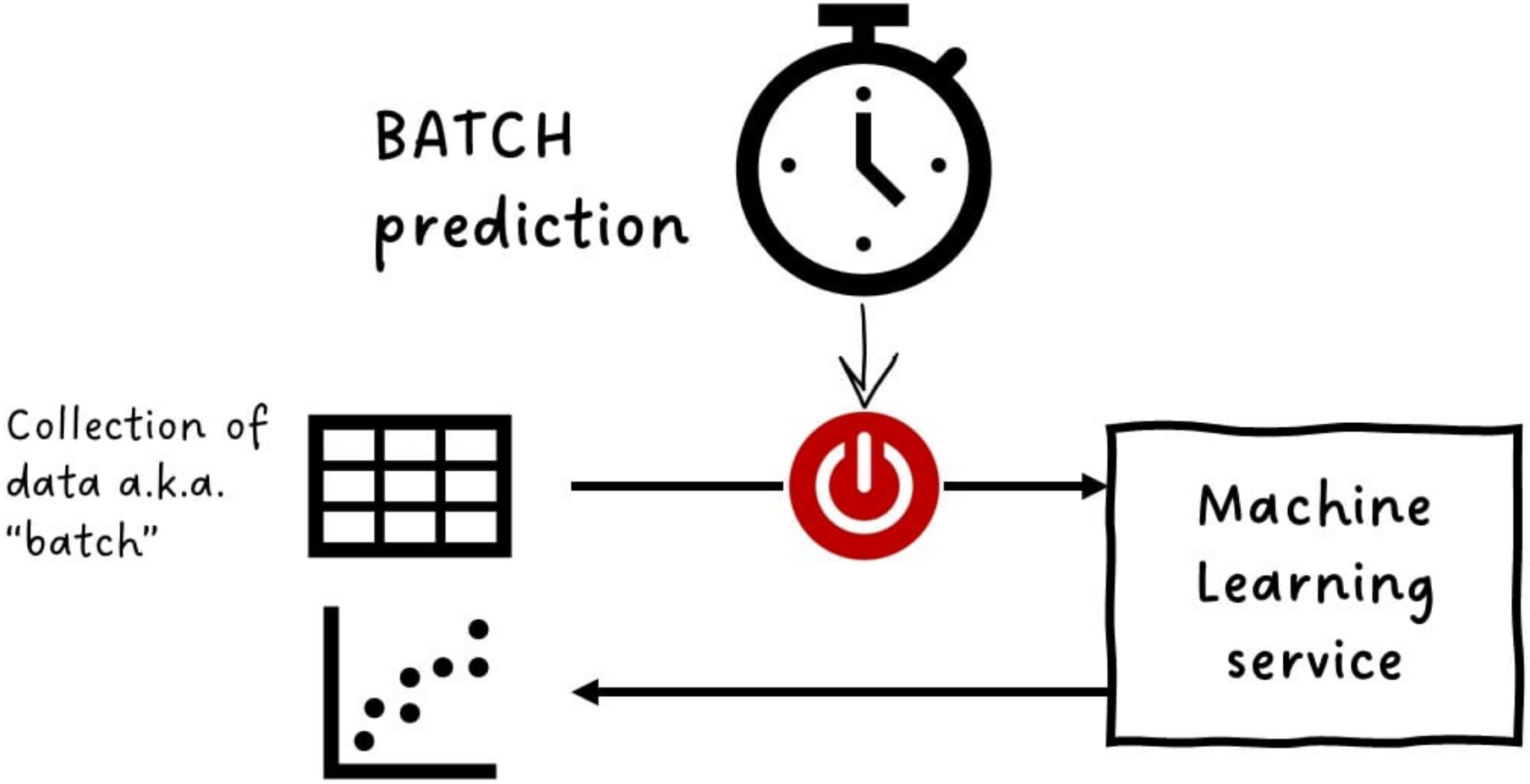

Machine
Learning
service

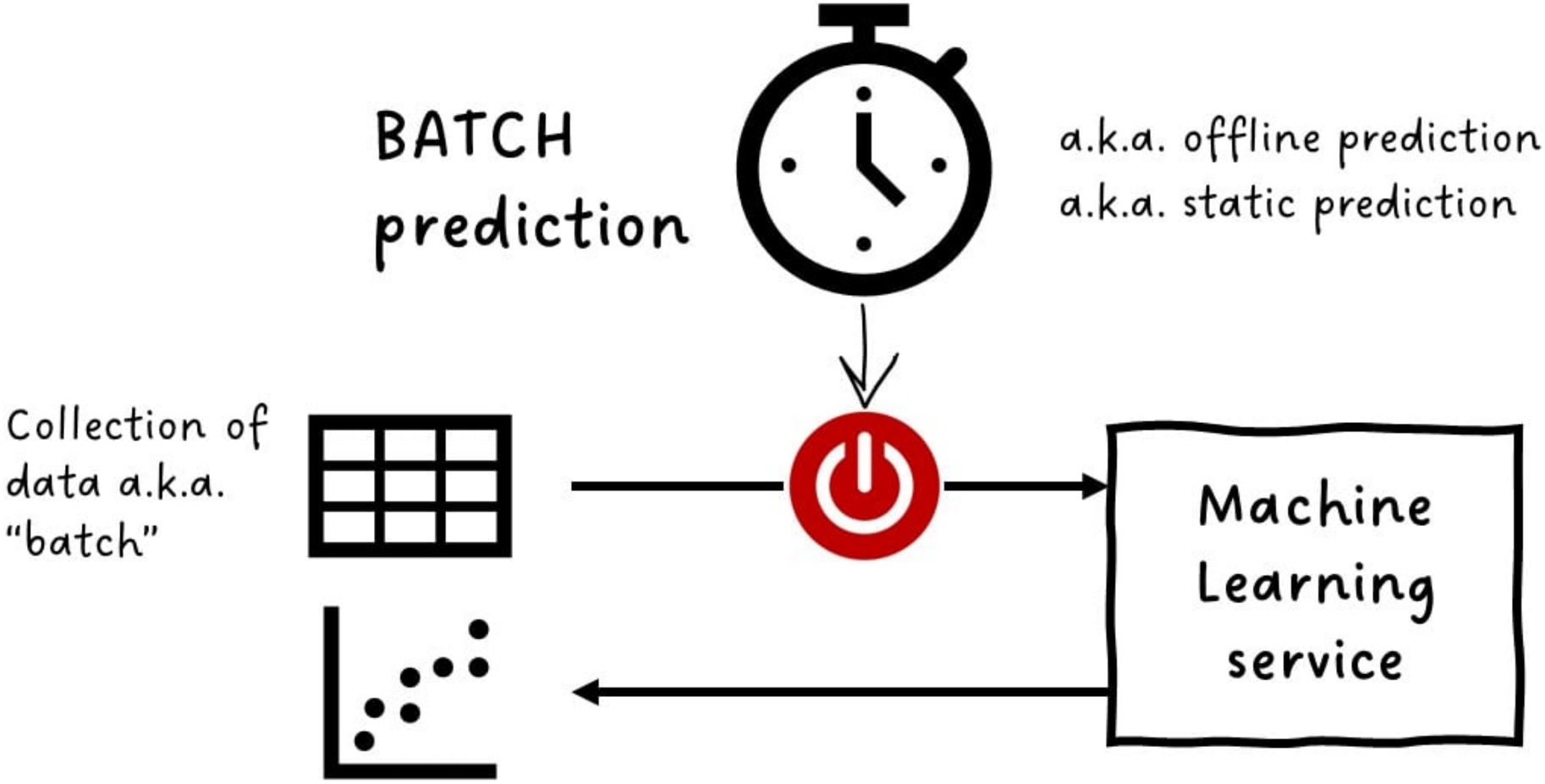

Machine
Learning
service


OR
?
Machine
Learning
service


Machine
Learning
service

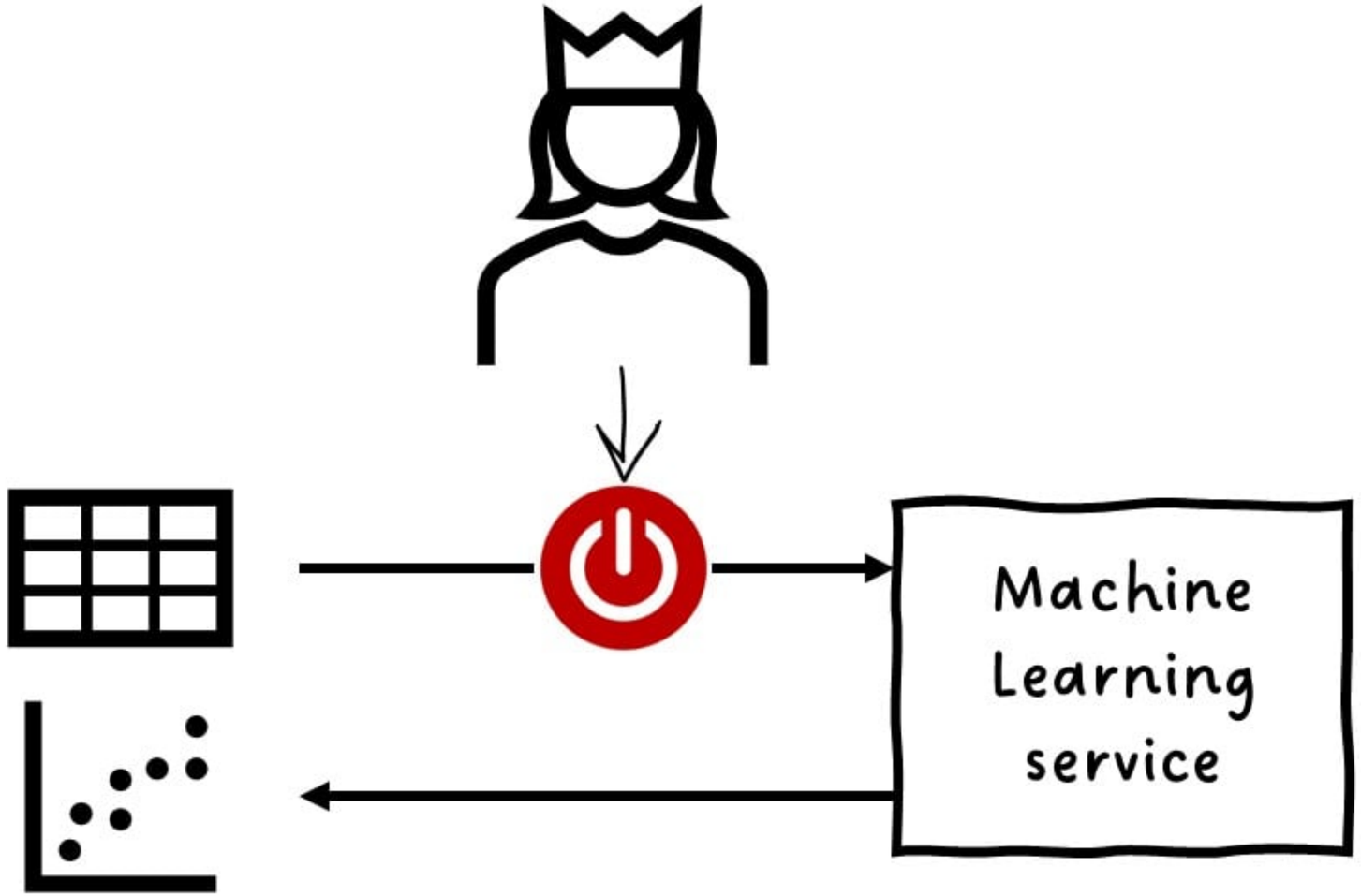

BATCH
prediction
Machine
Learning
service

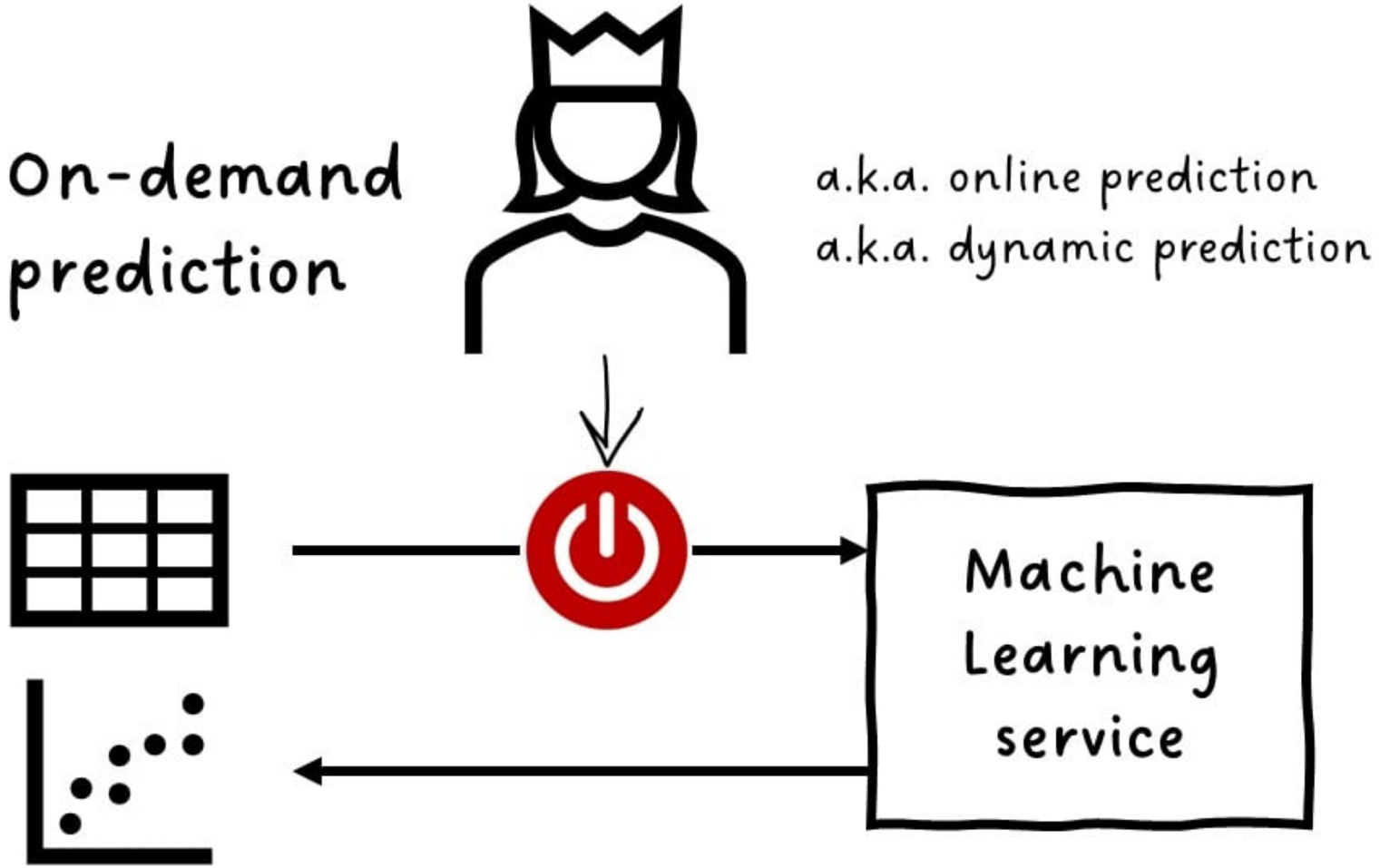

BATCH
prediction
Collection of
data a.k.a.
"batch"
Machine
Learning
service

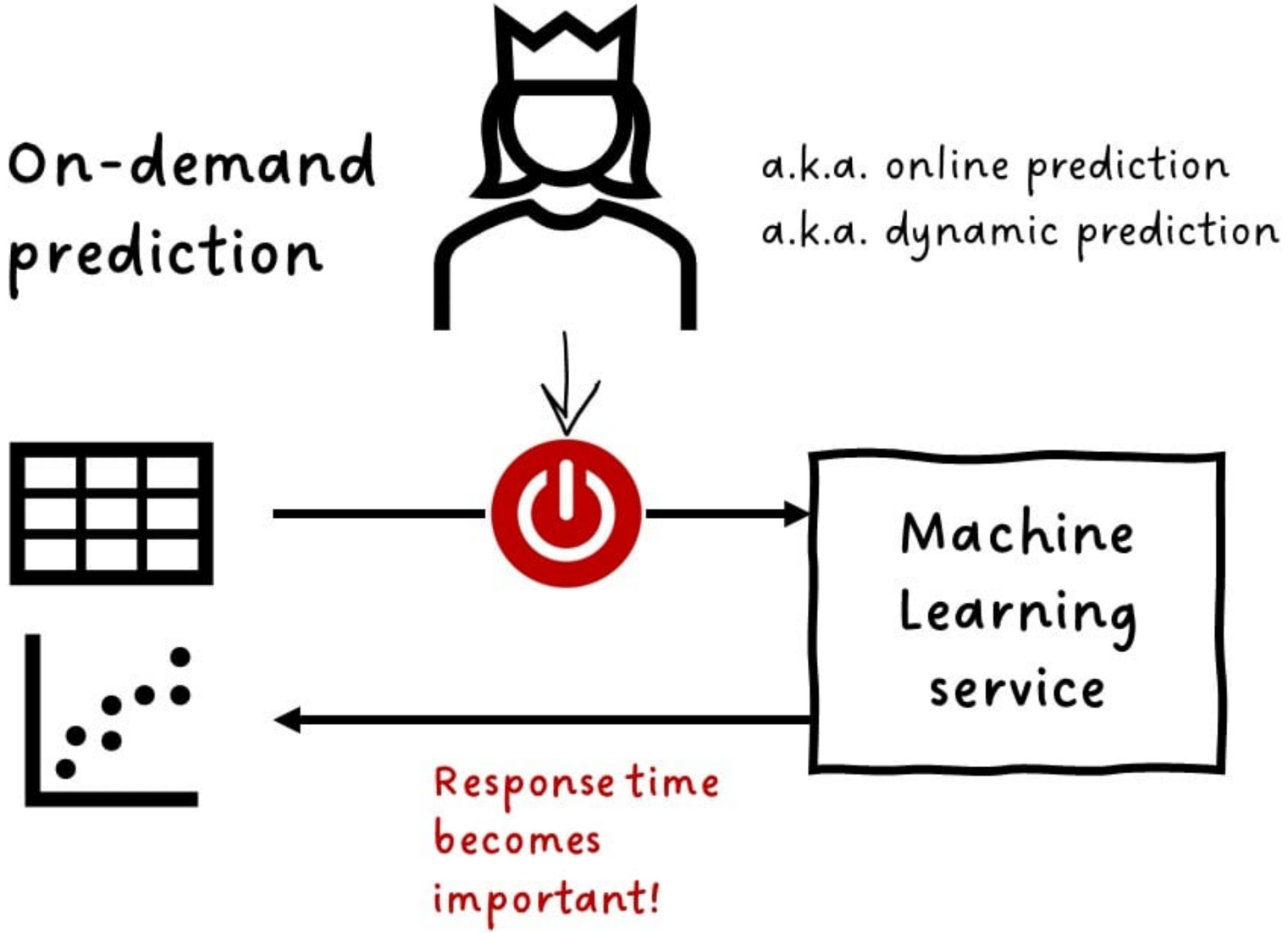

BATCH prediction
a.k.a. offline prediction
a.k.a. static prediction
Collection of data a.k.a. "batch"
Machine Learning service

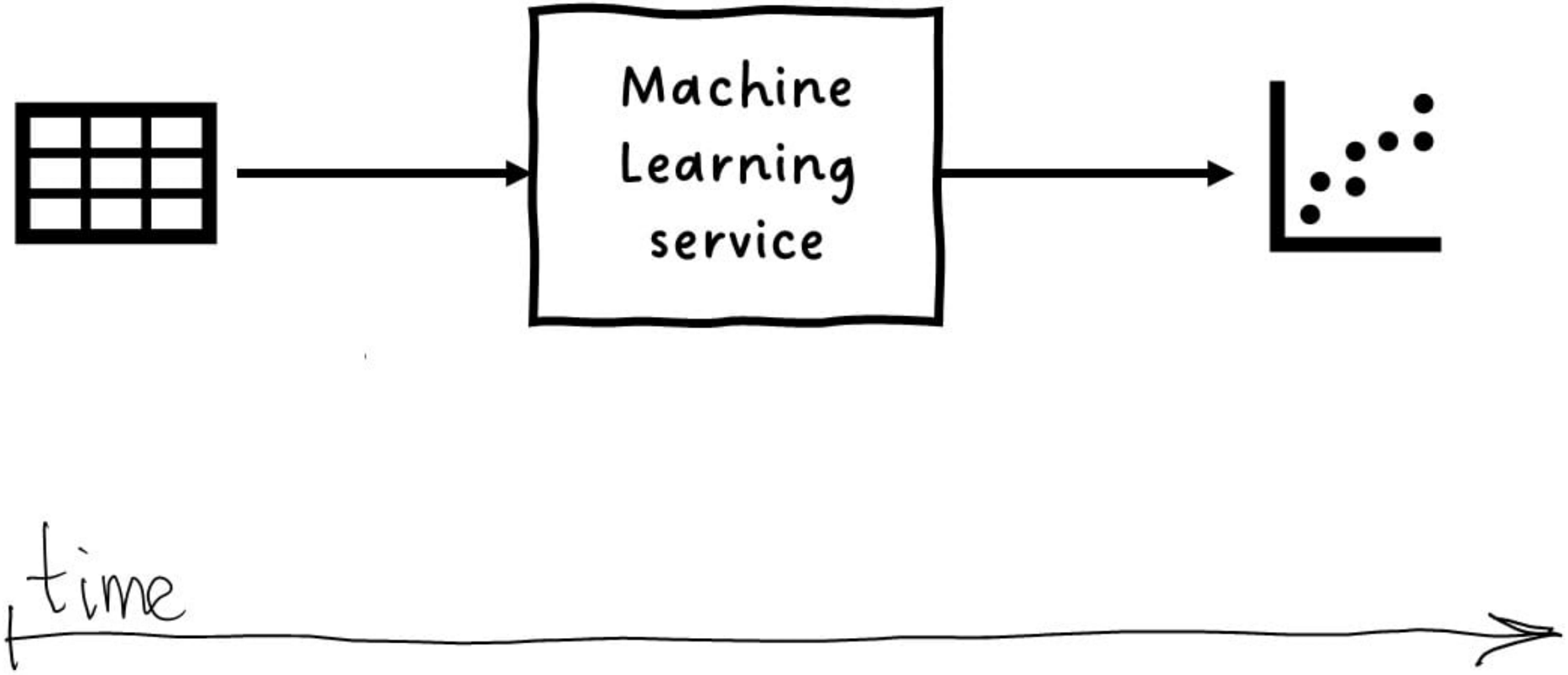
Batch prediction: Keep it simple

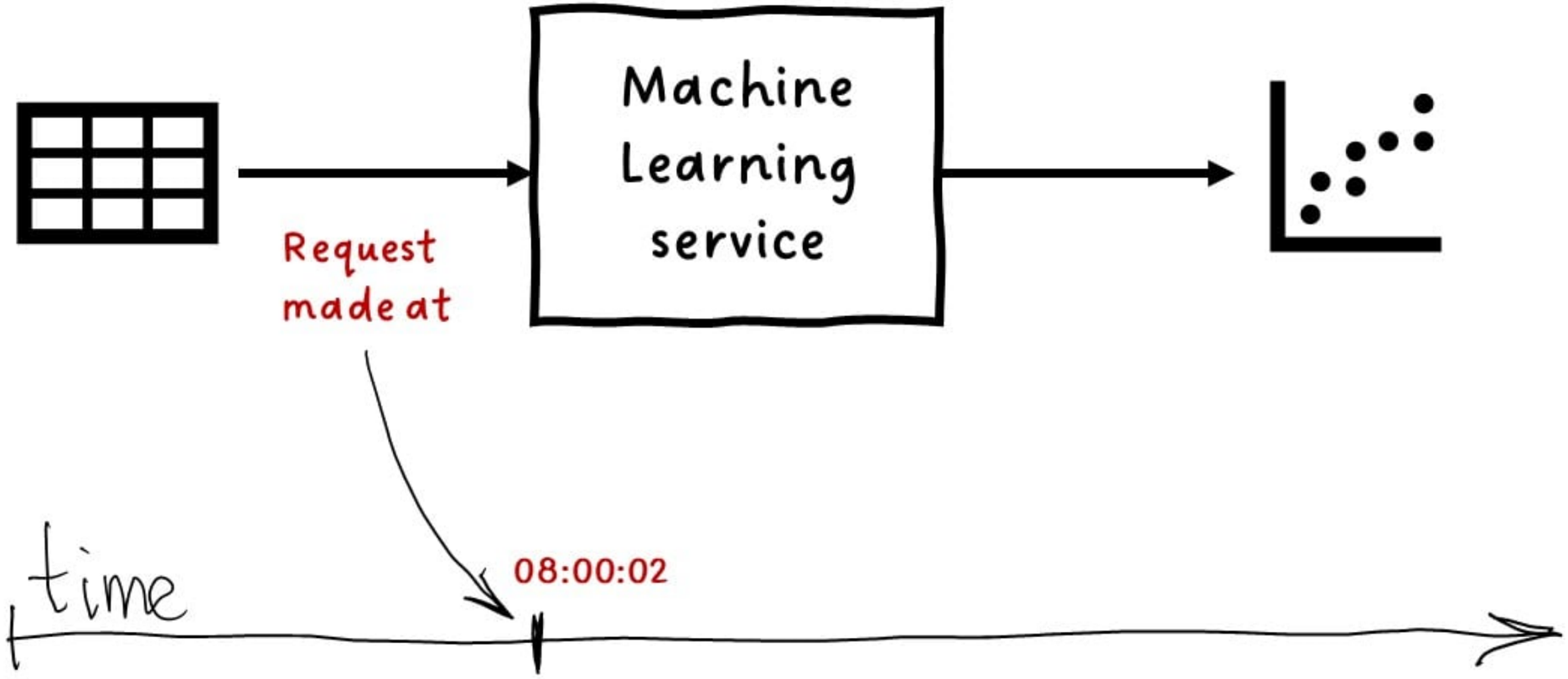
- Batch prediction is the simplest
- If use case allows it, go for it
- Good fit: monthly generation of sales forecasts

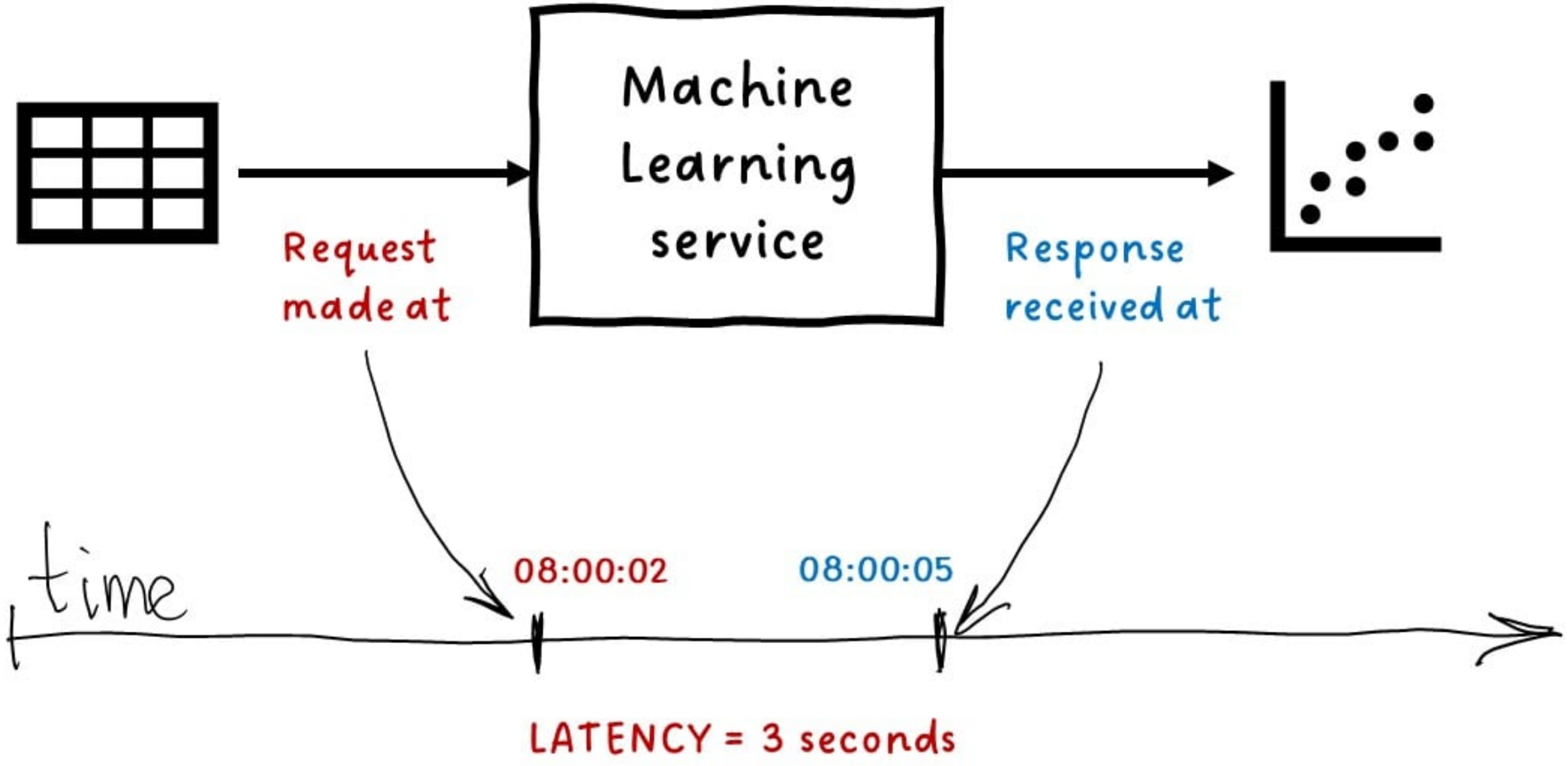

Machine
Learning
service


On-demand prediction
a.k.a. online prediction
a.k.a. dynamic prediction
Machine Learning service


On-demand prediction
a.k.a. online prediction
a.k.a. dynamic prediction
Machine Learning service
Response time becomes important!


Machine
Learning
service
time


Machine
Learning
service
Request
made at
time
08:00:02


Machine Learning service
Request made at
Response received at
time
08:00:02
08:00:05
LATENCY = 3 seconds

Acceptable latency

What is acceptable?

- < 1 hour?
- < 1 minute?
- < 1 second?
- < 1 millisecond?

Near-real time prediction a.k.a. Stream processing

Acceptable latency ~= X minutes

Also known as ***stream processing*** (requests and responses form "data streams")

Real-time prediction

Acceptable latency < 1 sec

Example:

- Credit card fraud detection
- Late prediction as good as useless

When latency is a priority

- Weaker, but faster model more valuable than a stronger, but slower one
- Models deployed to end user devices to reduce latency => "edge deployment"
  - ML-infused smartphone apps:
    - navigation apps
    - unlocking via facial recognition
    - image filters

Let's practice!

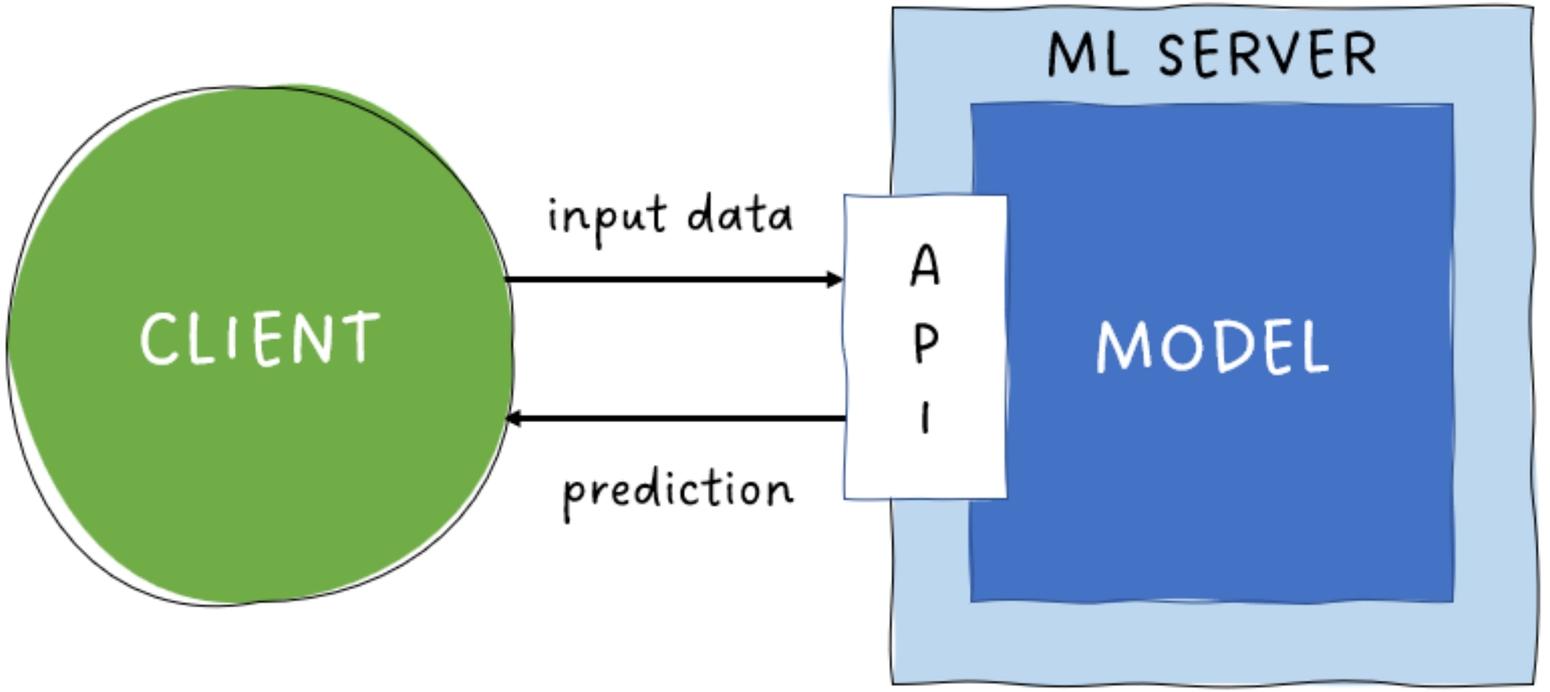
MLOPS DEPLOYMENT AND LIFE CYCLING

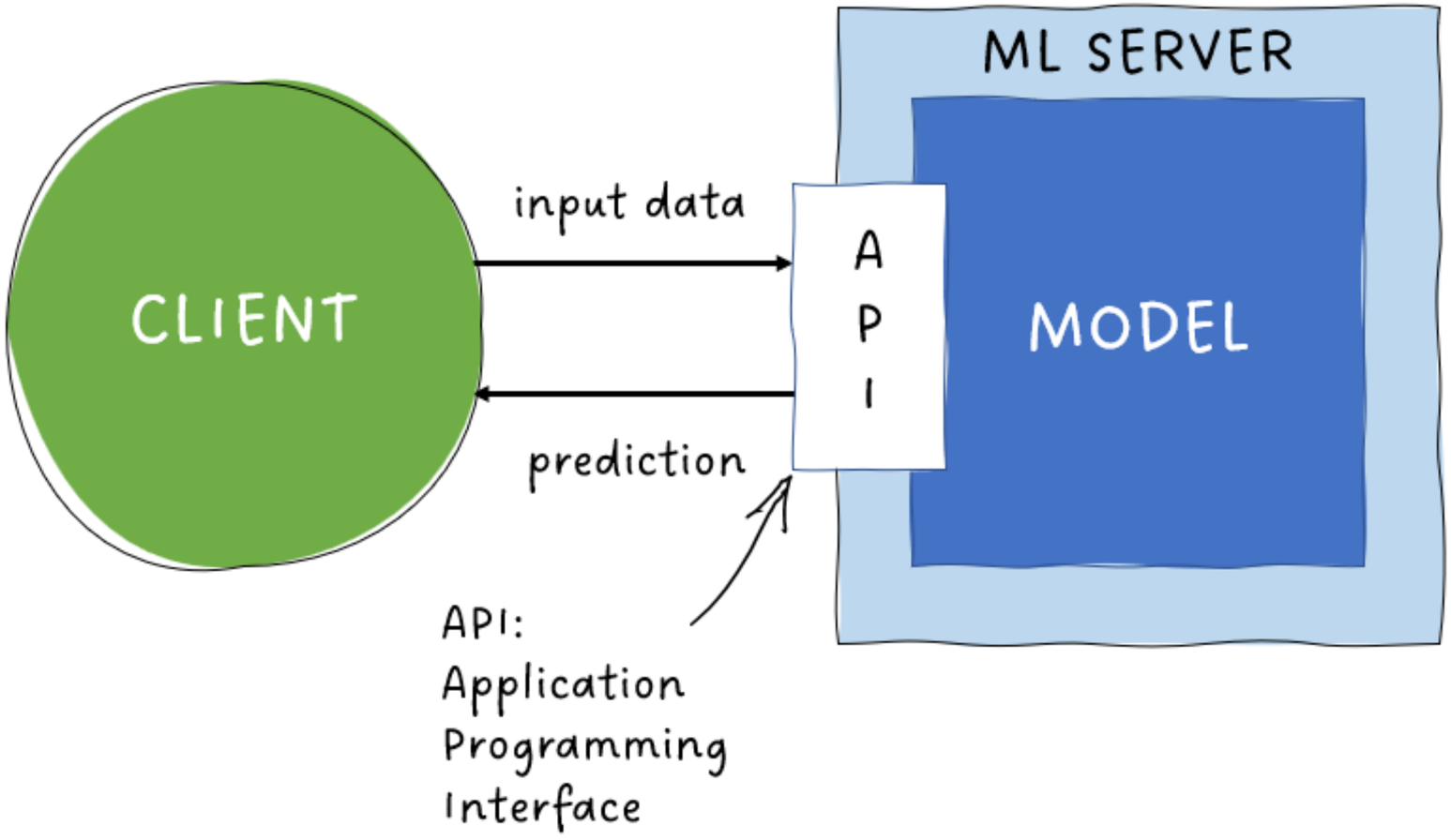
Building the API

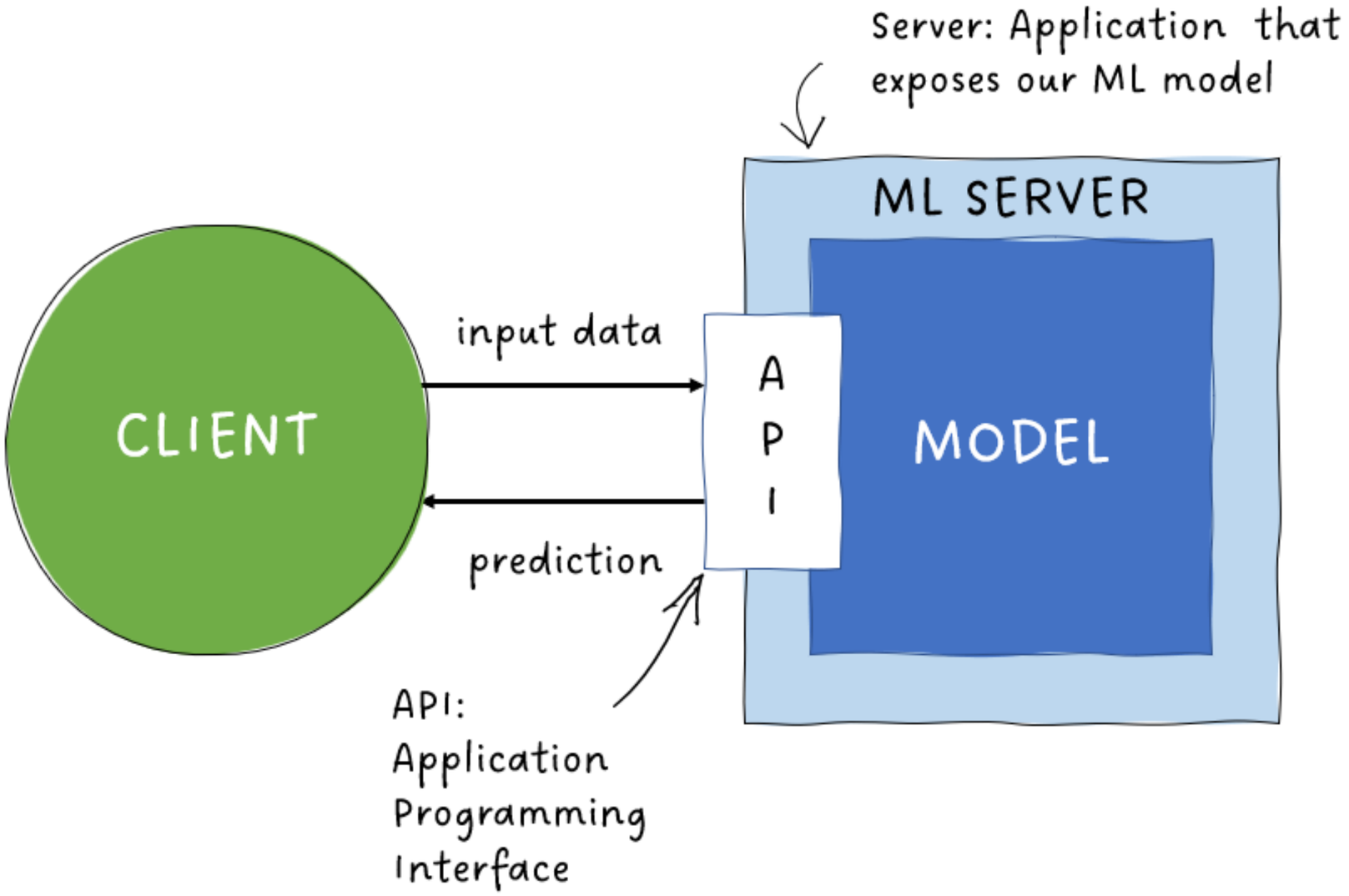
MLOPS DEPLOYMENT AND LIFE CYCLING

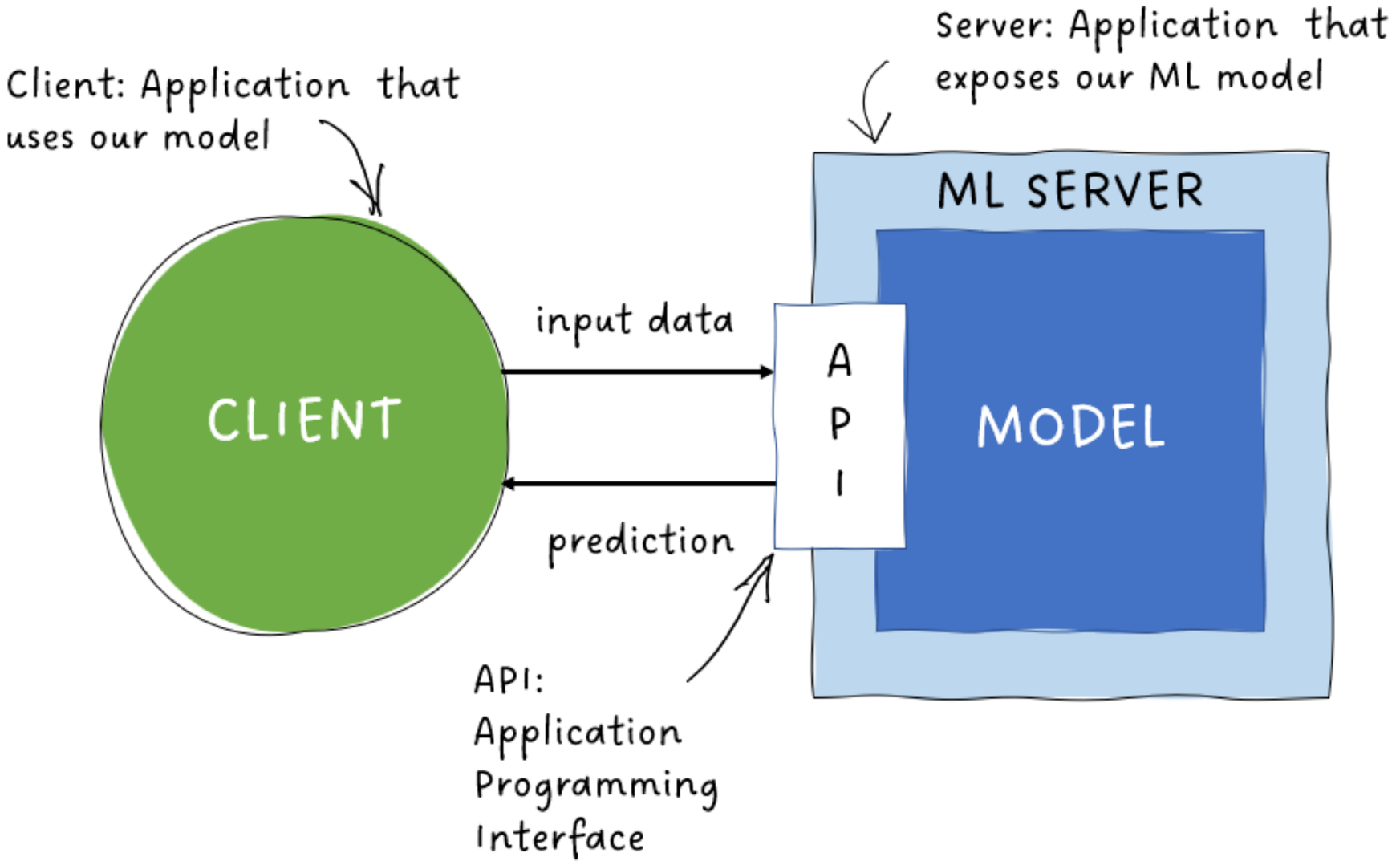
Nemanja Radojkovic
Senior Machine Learning Engineer

datacamp


ML SERVER
CLIENT
input data
prediction
A
P
I
MODEL


ML SERVER
CLIENT
input data
prediction
A
P
I
MODEL
API:
Application
Programming
Interface


Server: Application that exposes our ML model
ML SERVER
CLIENT
input data
prediction
A
P
I
MODEL
API: Application Programming Interface


Client: Application that uses our model
Server: Application that exposes our ML model
ML SERVER
CLIENT
input data
prediction
A
P
I
MODEL
API: Application Programming Interface

Service to the client

Running procedures on our server

Example: Sentiment analysis API for Amazon Comprehend

Access remote databases

Example: Twitter API, New York Times API

API architectures

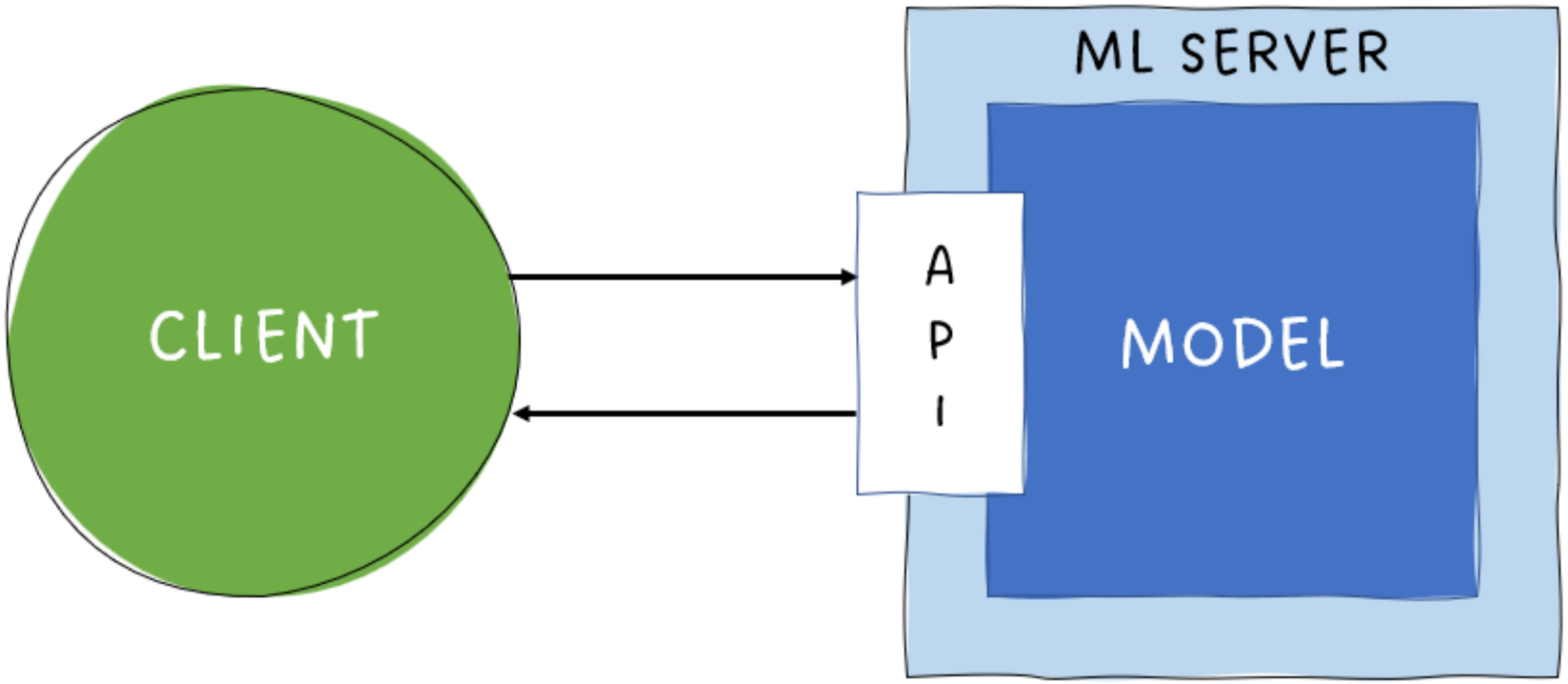
1. **REST**: **RE**presentational **S**tate **T**ransfer
2. **RPC**: **R**emote **P**rocedure **C**all
3. **SOAP**: **S**imple **O**bject **A**ccess **P**rotocol

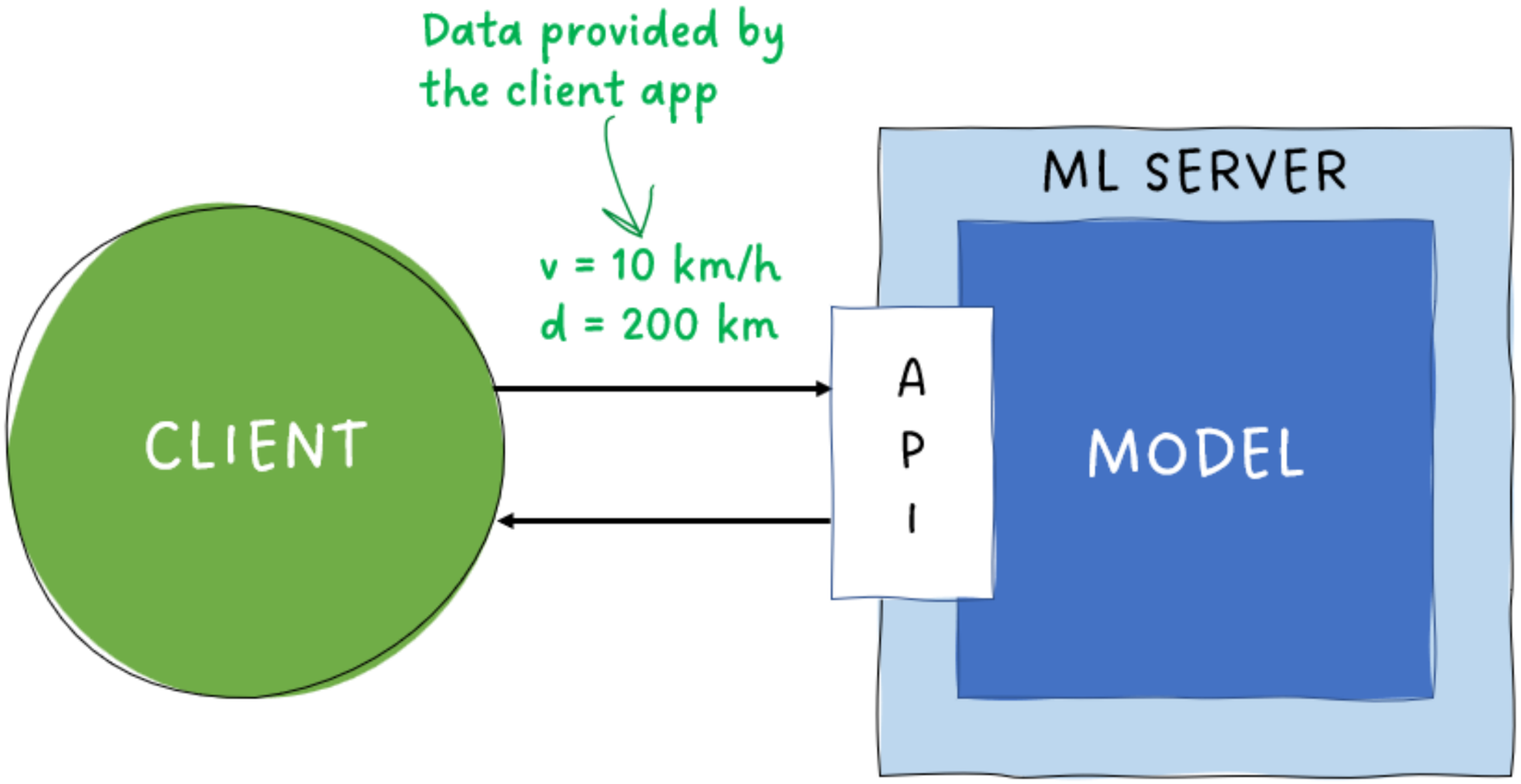
In this course: Architecture-agnostic view

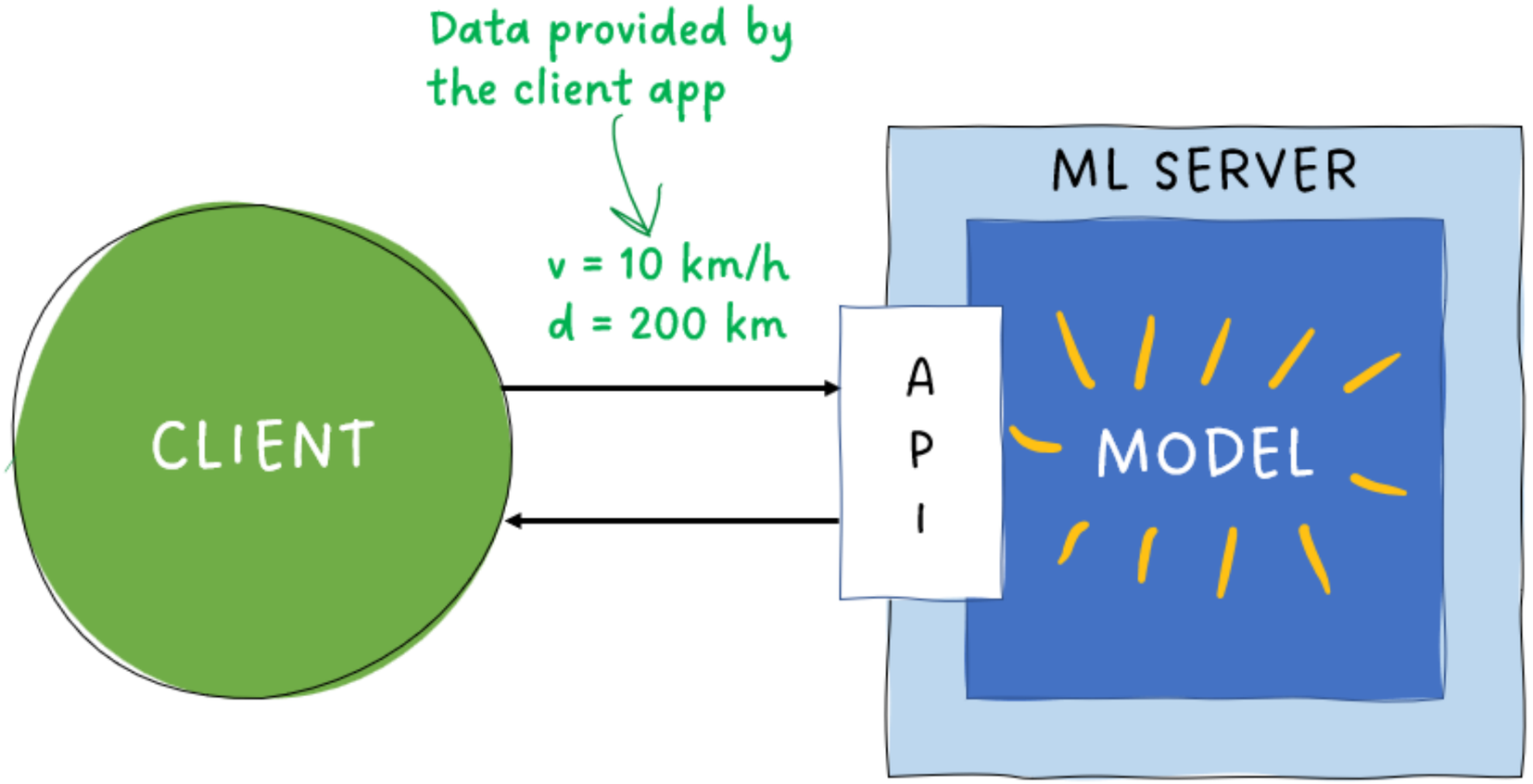
Example: Navigation app

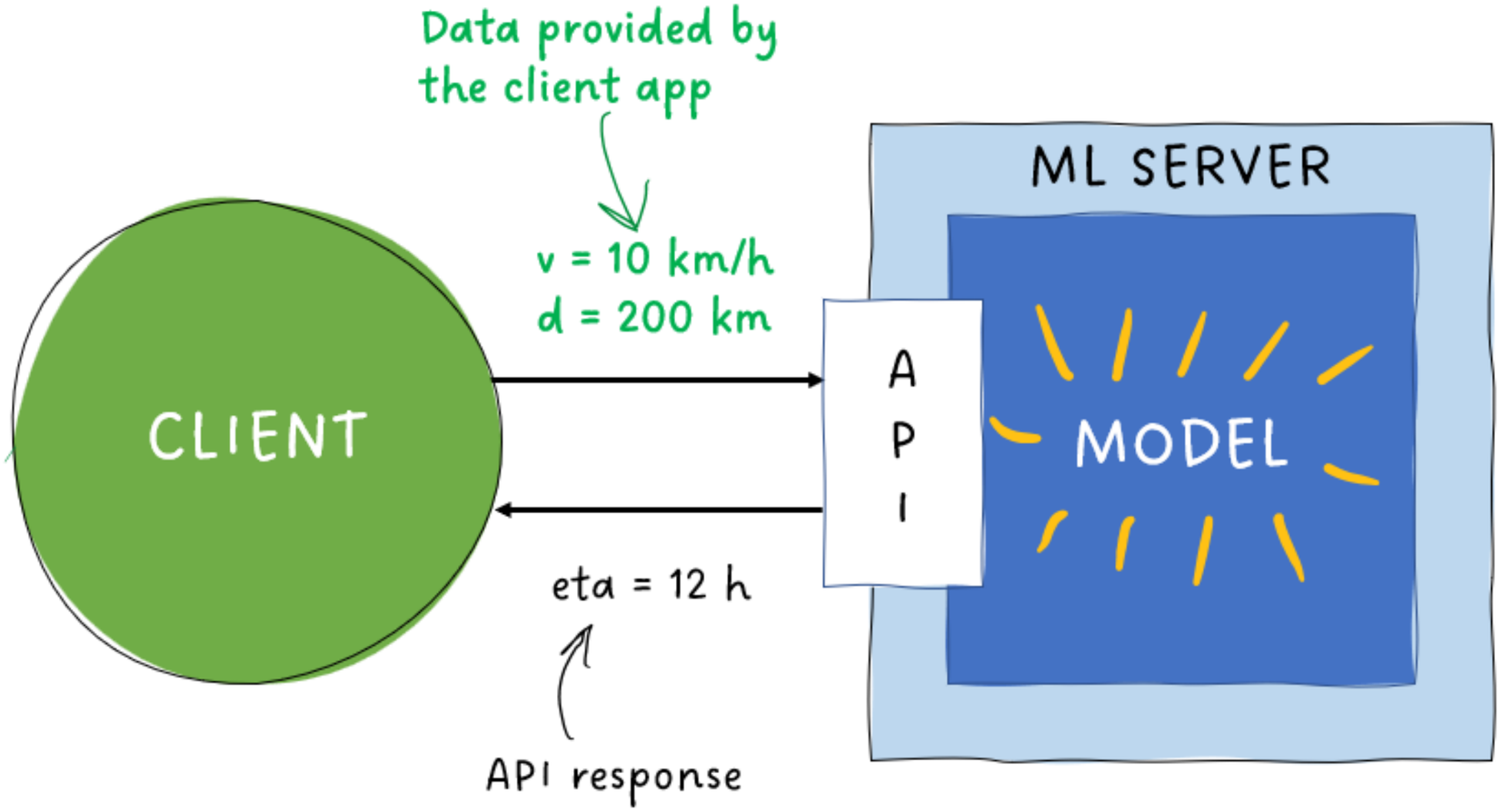
API purpose: Enable client apps to calculate the estimated time of arrival

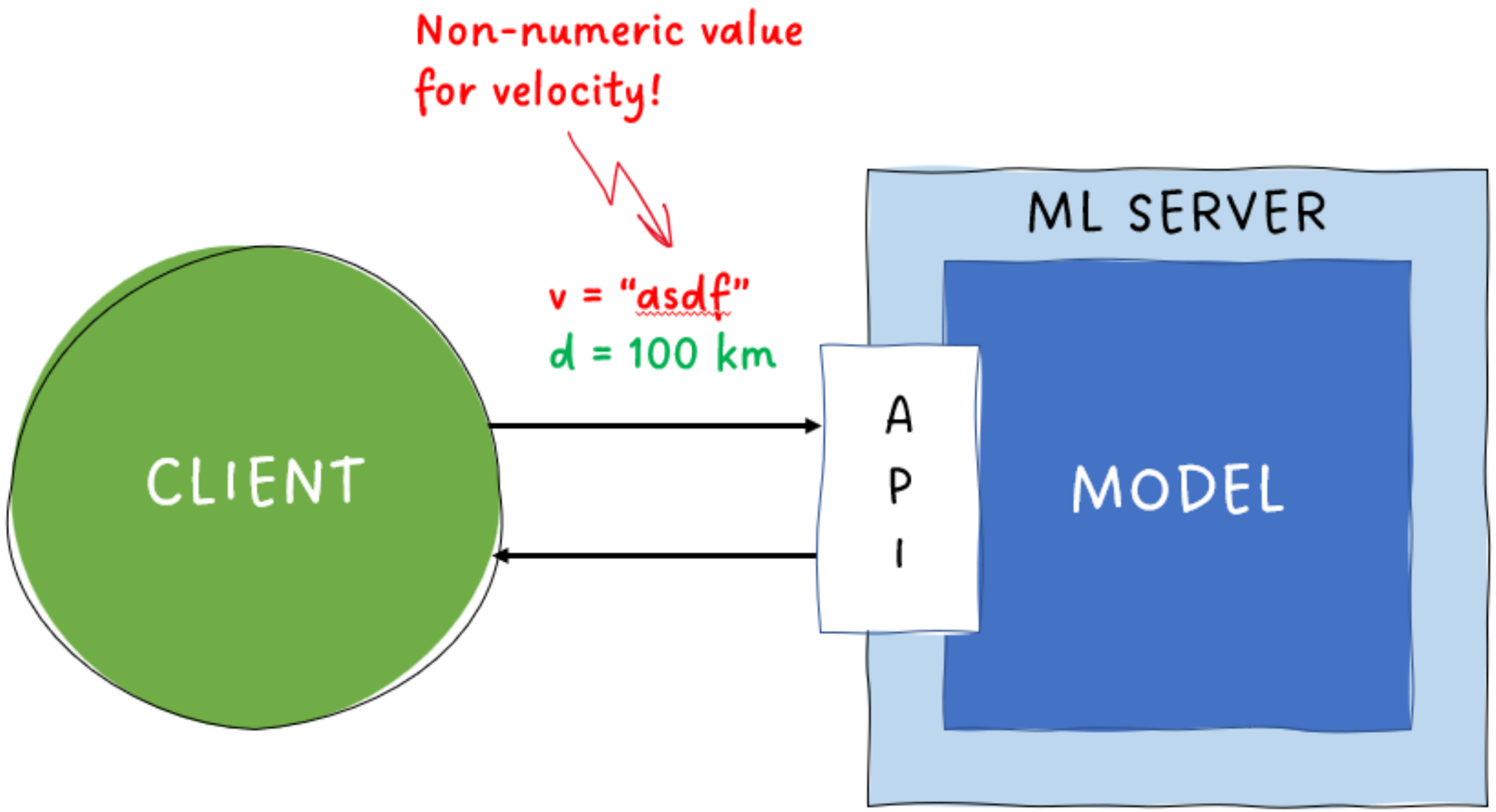
[1] Image: Pixabay, https://www.stockvault.net/user/profile/161904

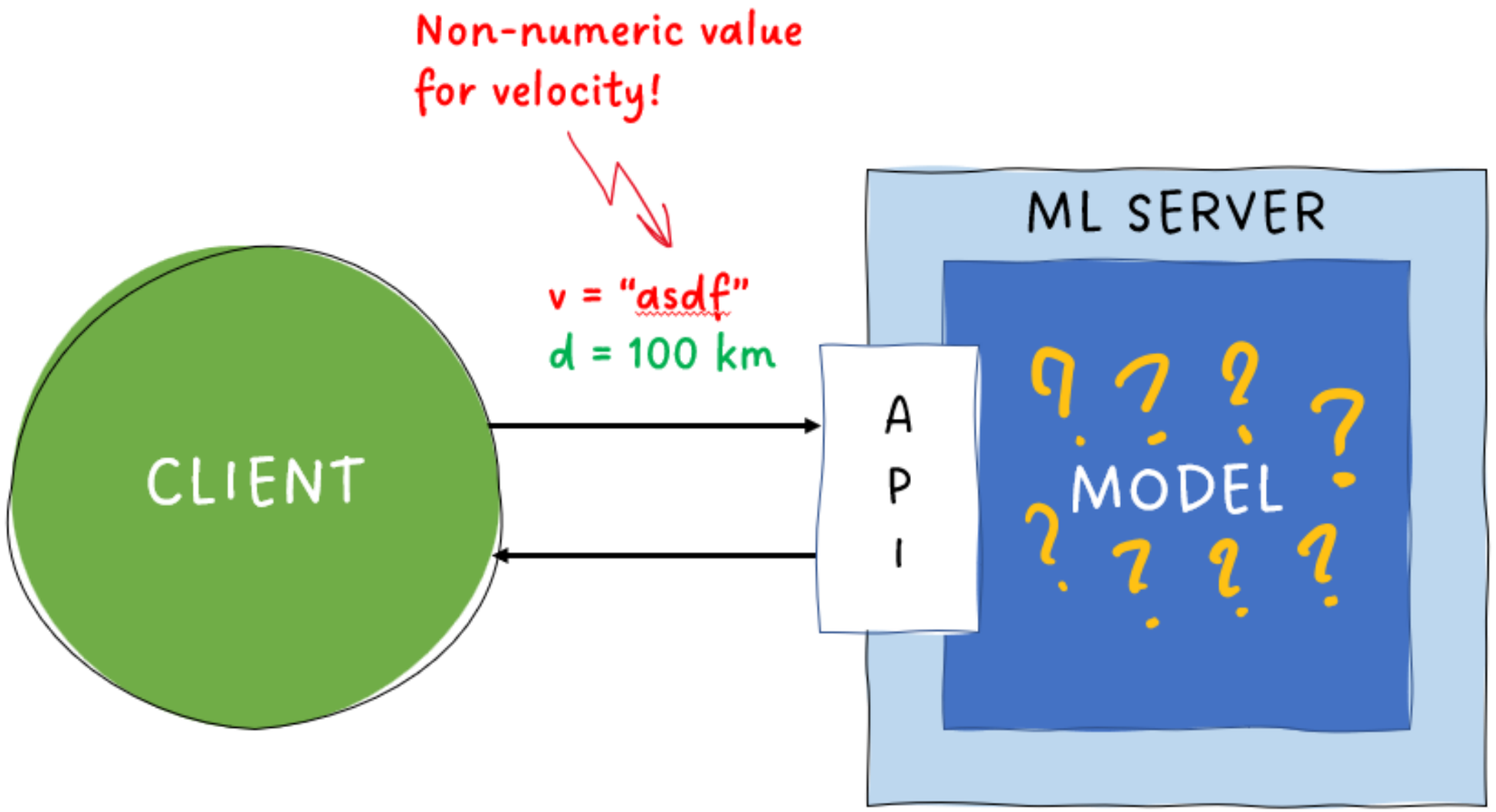

ML SERVER
CLIENT
A
P
I
MODEL

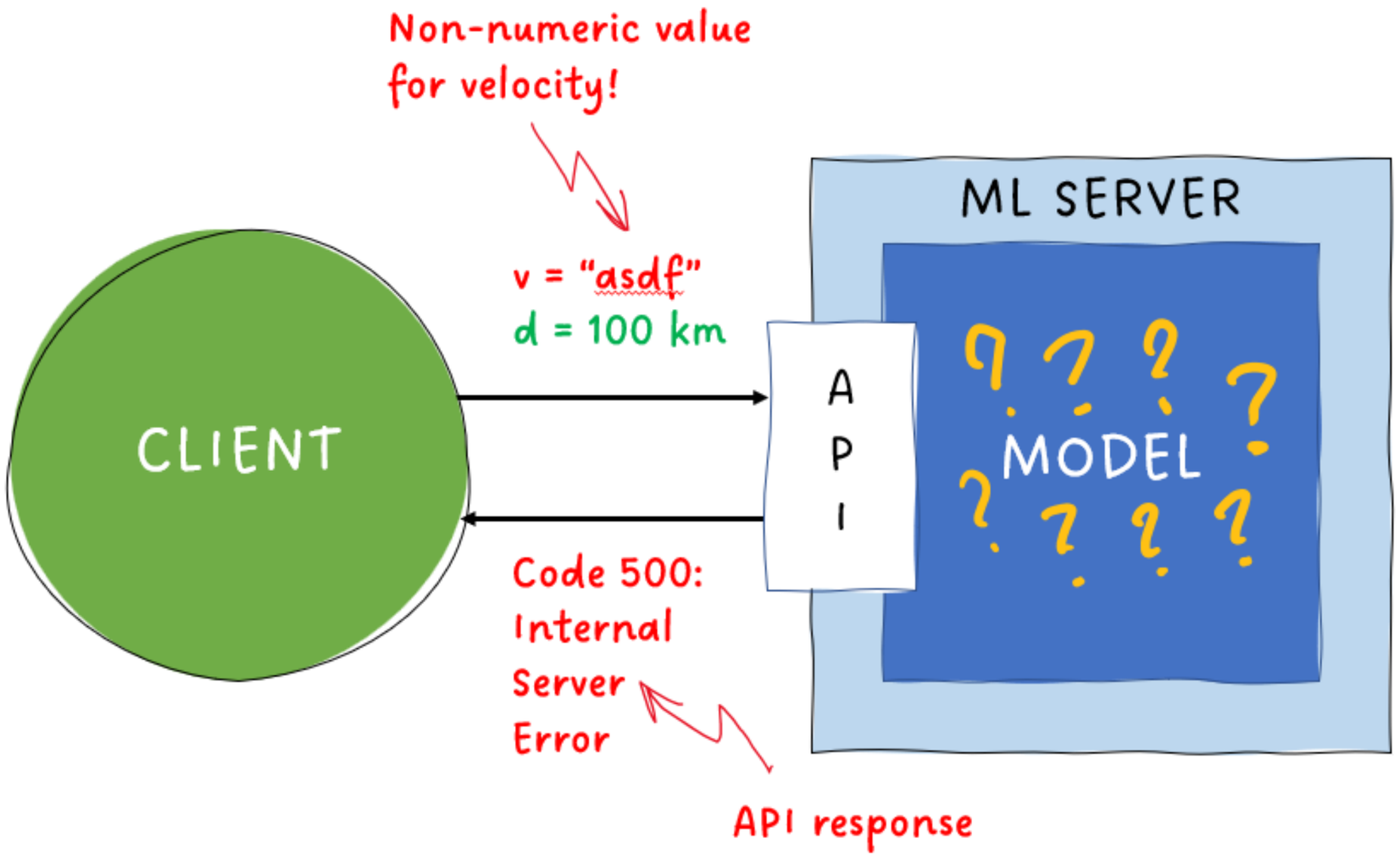

Data provided by
the client app
v = 10 km/h
d = 200 km
CLIENT
ML SERVER
A
P
I
MODEL

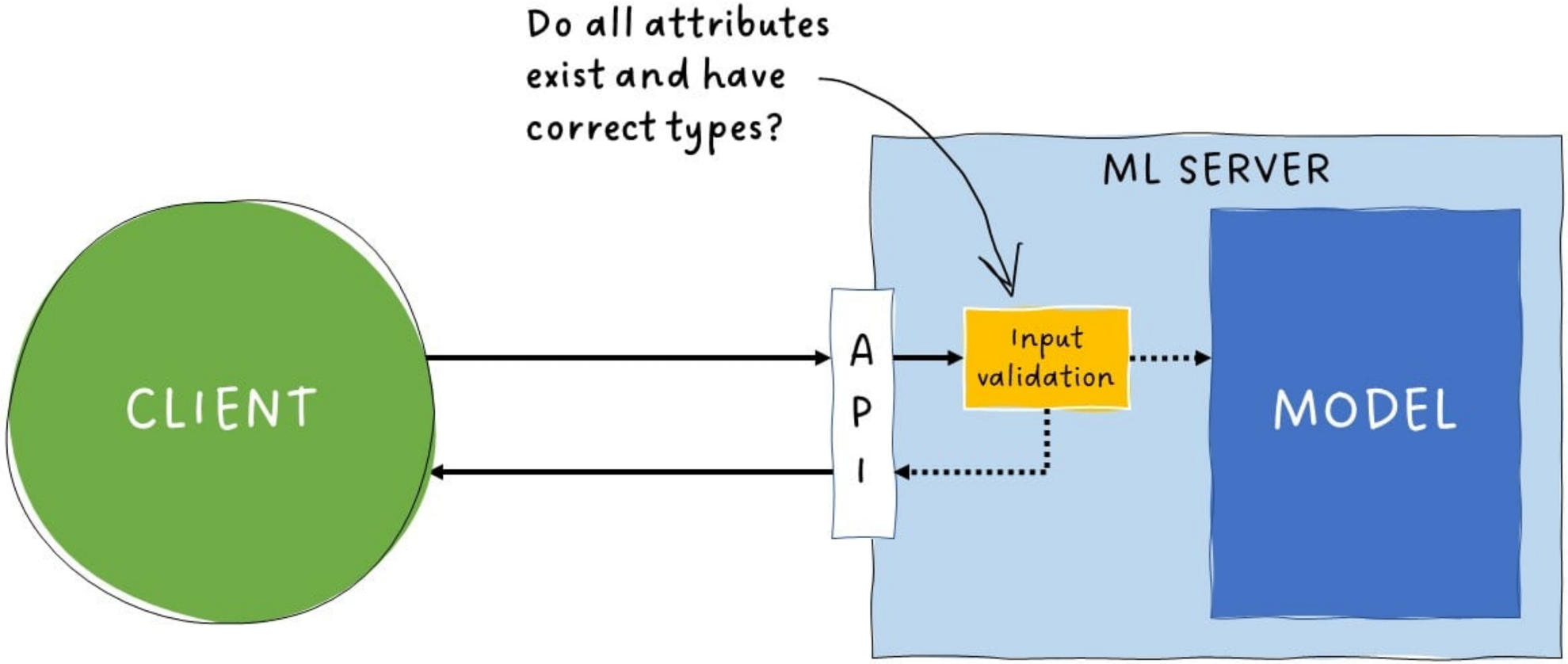

Data provided by
the client app
v = 10 km/h
d = 200 km
CLIENT
ML SERVER
A
P
I
MODEL

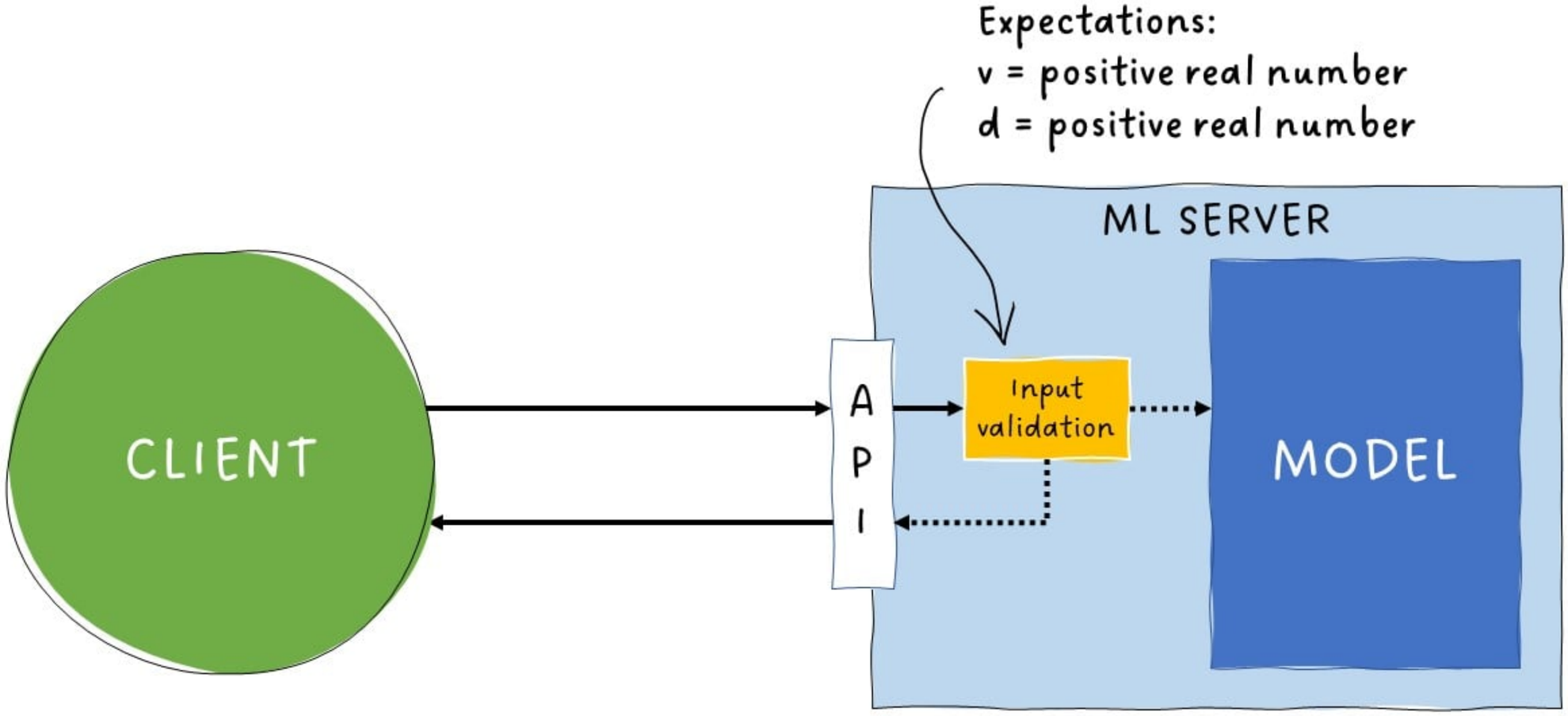

Data provided by
the client app
v = 10 km/h
d = 200 km
CLIENT
ML SERVER
A
P
I
MODEL
eta = 12 h
API response

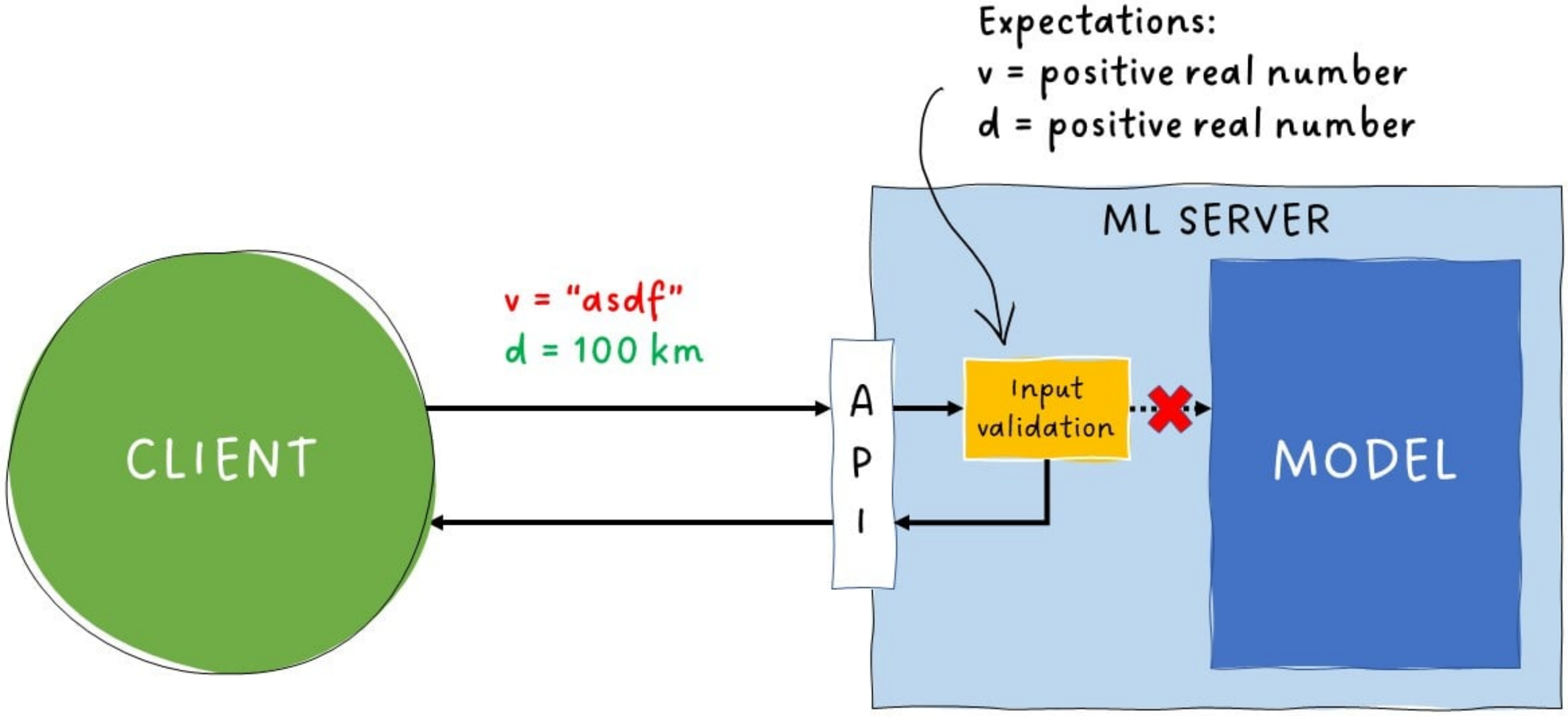

Non-numeric value
for velocity!
v = "asdf"
d = 100 km
CLIENT
ML SERVER
A
P
I
MODEL

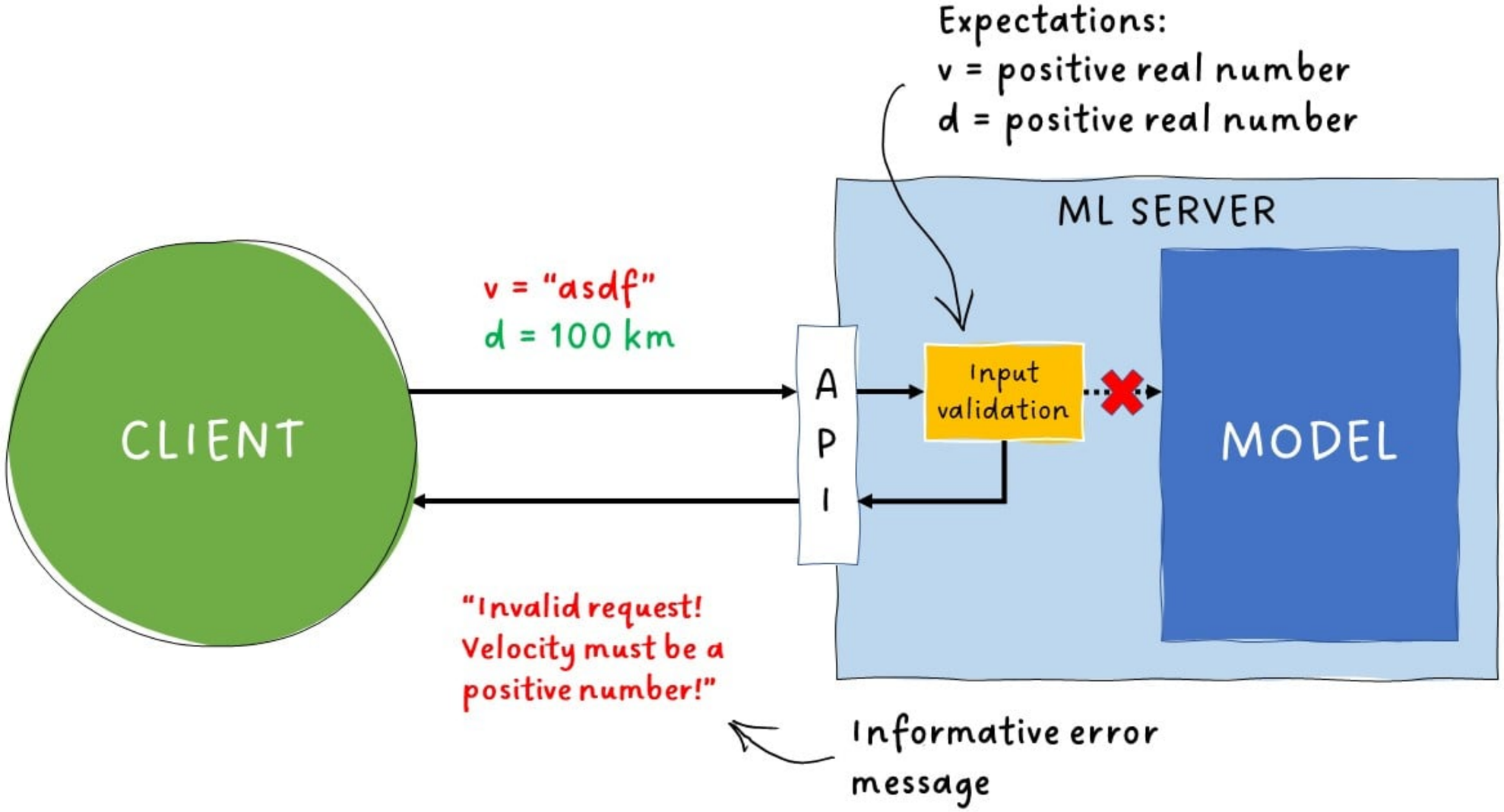

Non-numeric value
for velocity!
v = "asdf"
d = 100 km
CLIENT
A
P
I
ML SERVER
MODEL

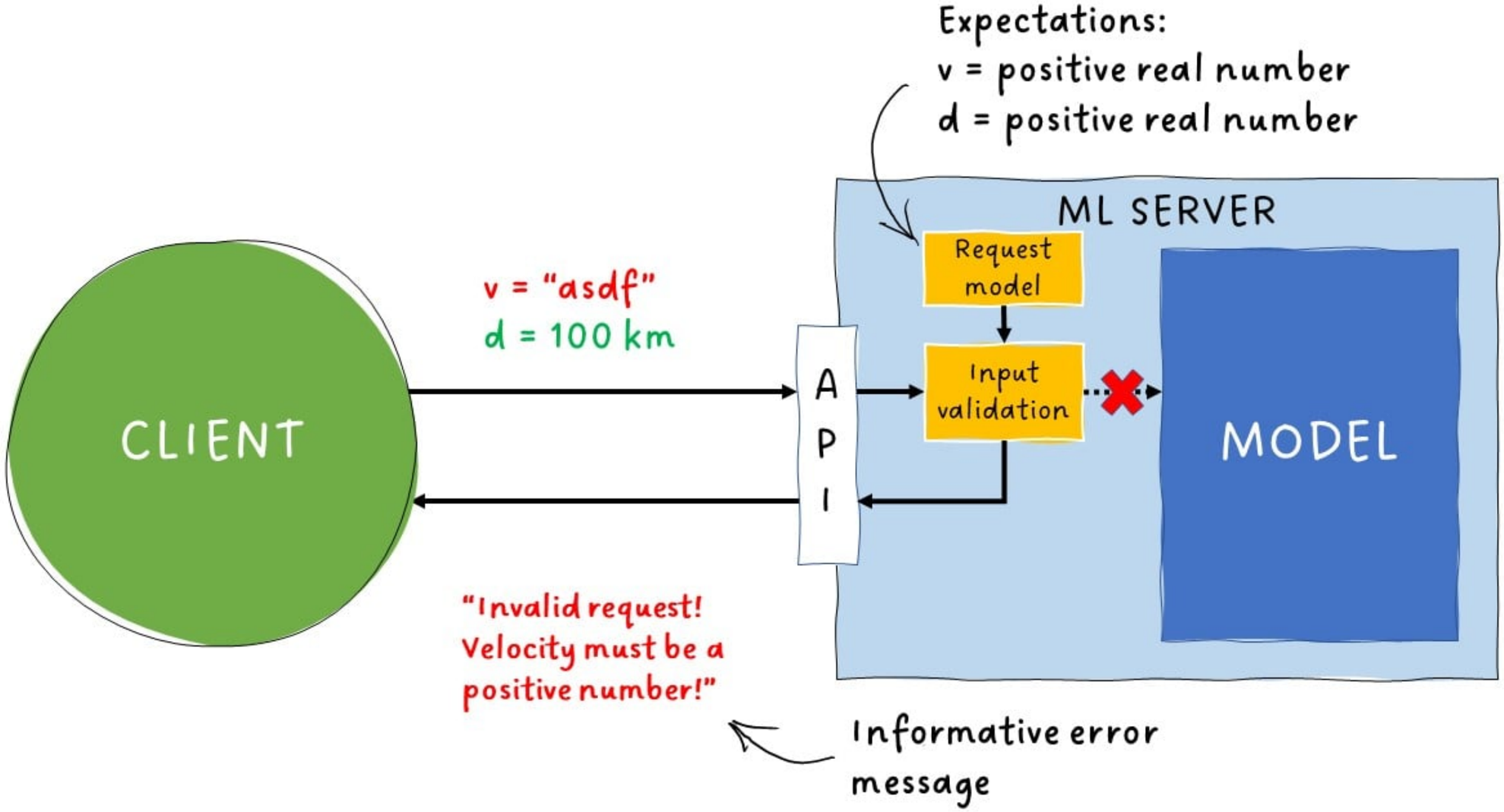

Non-numeric value for velocity!
v = "asdf"
d = 100 km
CLIENT
A P I
ML SERVER
MODEL
Code 500: Internal Server Error
API response

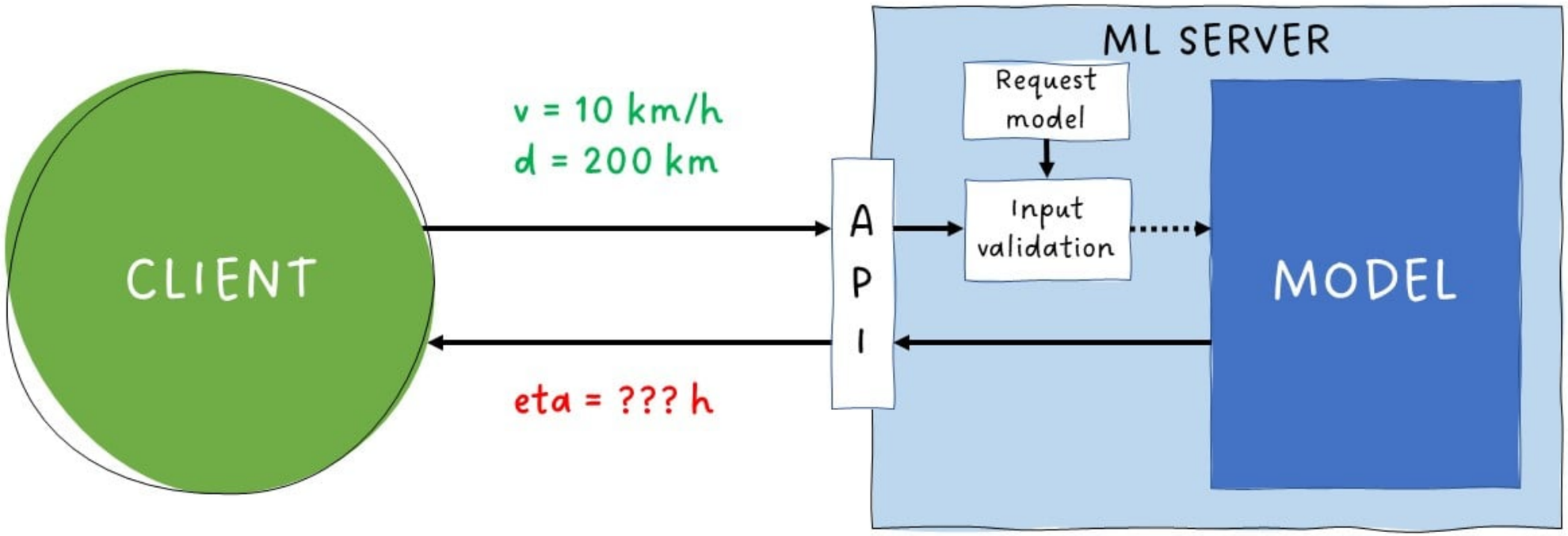

Do all attributes exist and have correct types?
ML SERVER
CLIENT
A P I
Input validation
MODEL

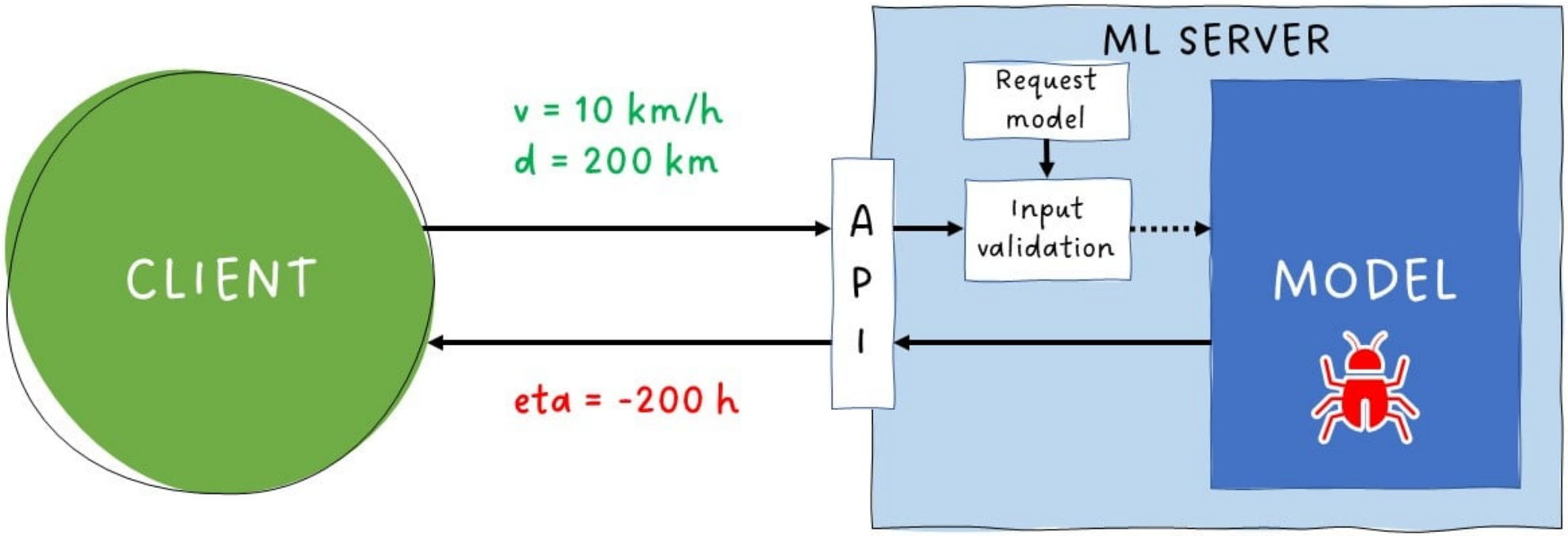

Expectations:
v = positive real number
d = positive real number
ML SERVER
CLIENT
A
P
I
Input validation
MODEL

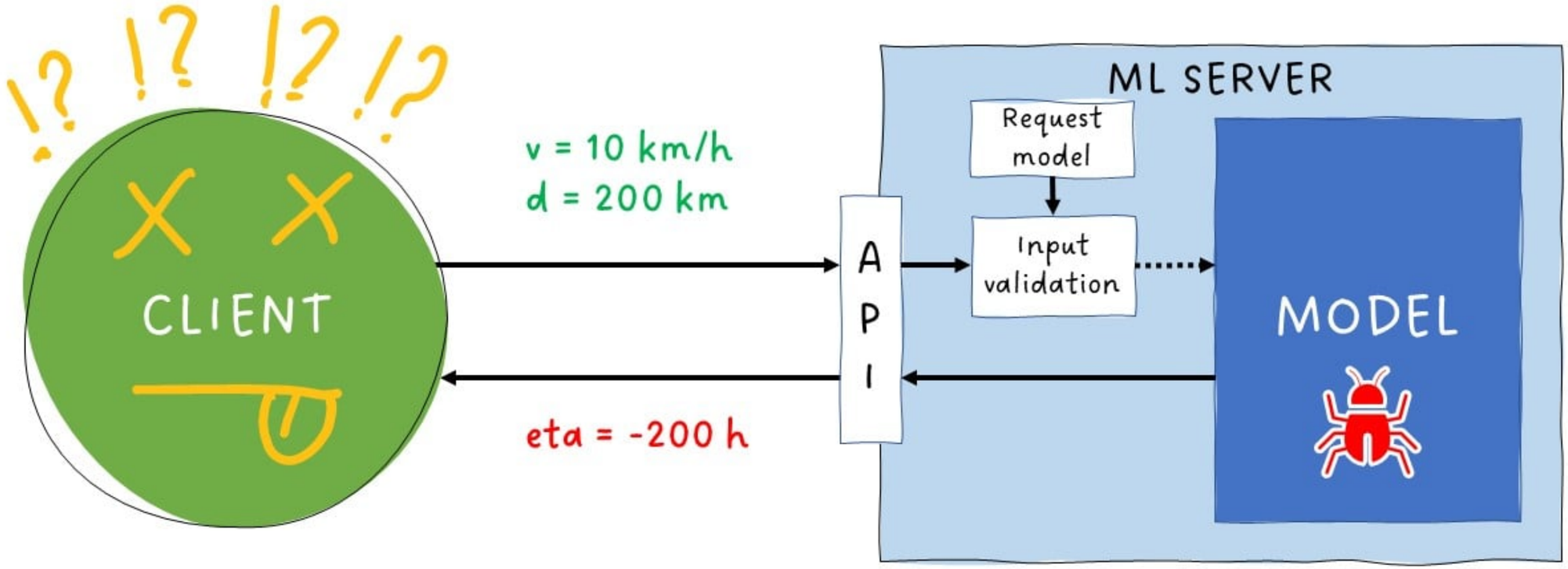

Expectations:
v = positive real number
d = positive real number
ML SERVER
v = "asdf"
d = 100 km
CLIENT
A
P
I
Input validation
MODEL

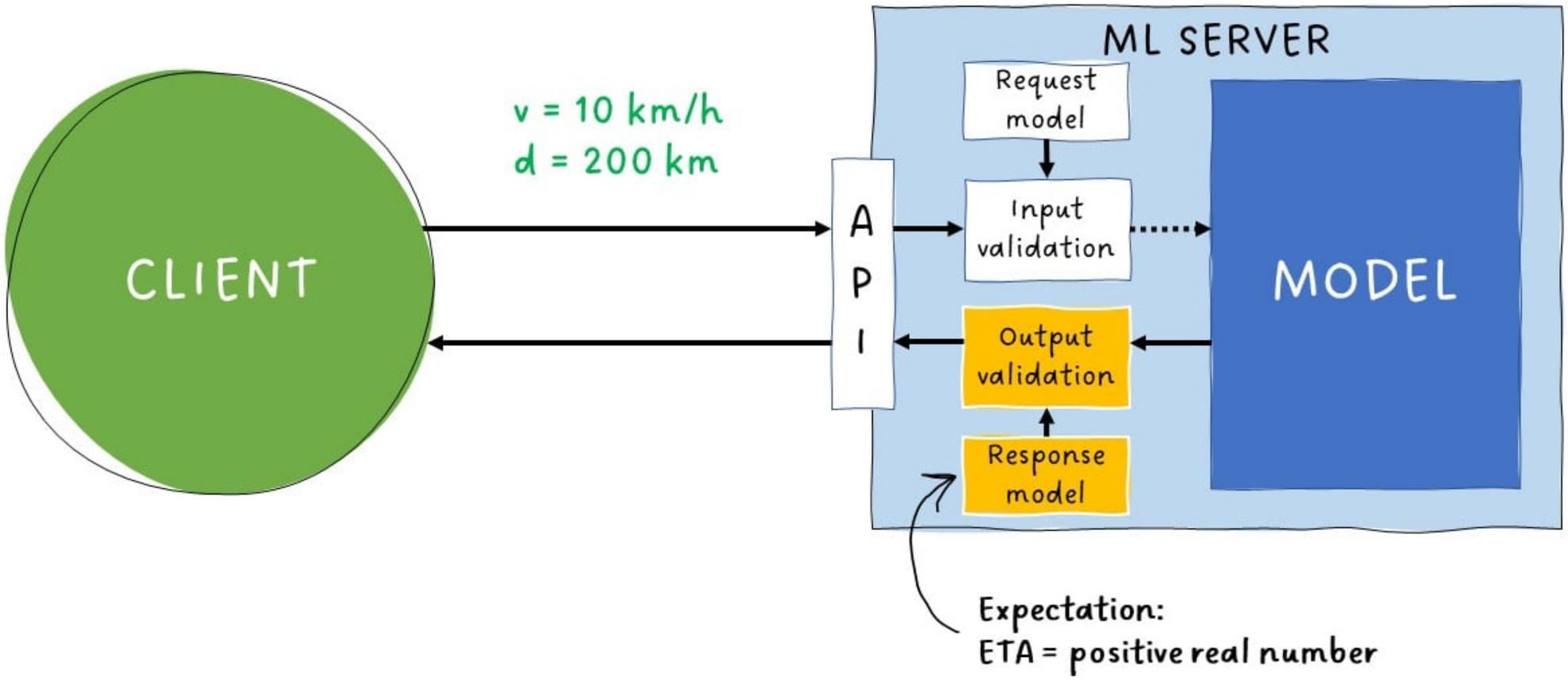

Expectations:
v = positive real number
d = positive real number
ML SERVER
CLIENT
v = "asdf"
d = 100 km
API
Input validation
MODEL
"Invalid request!
Velocity must be a
positive number!"
Informative error
message

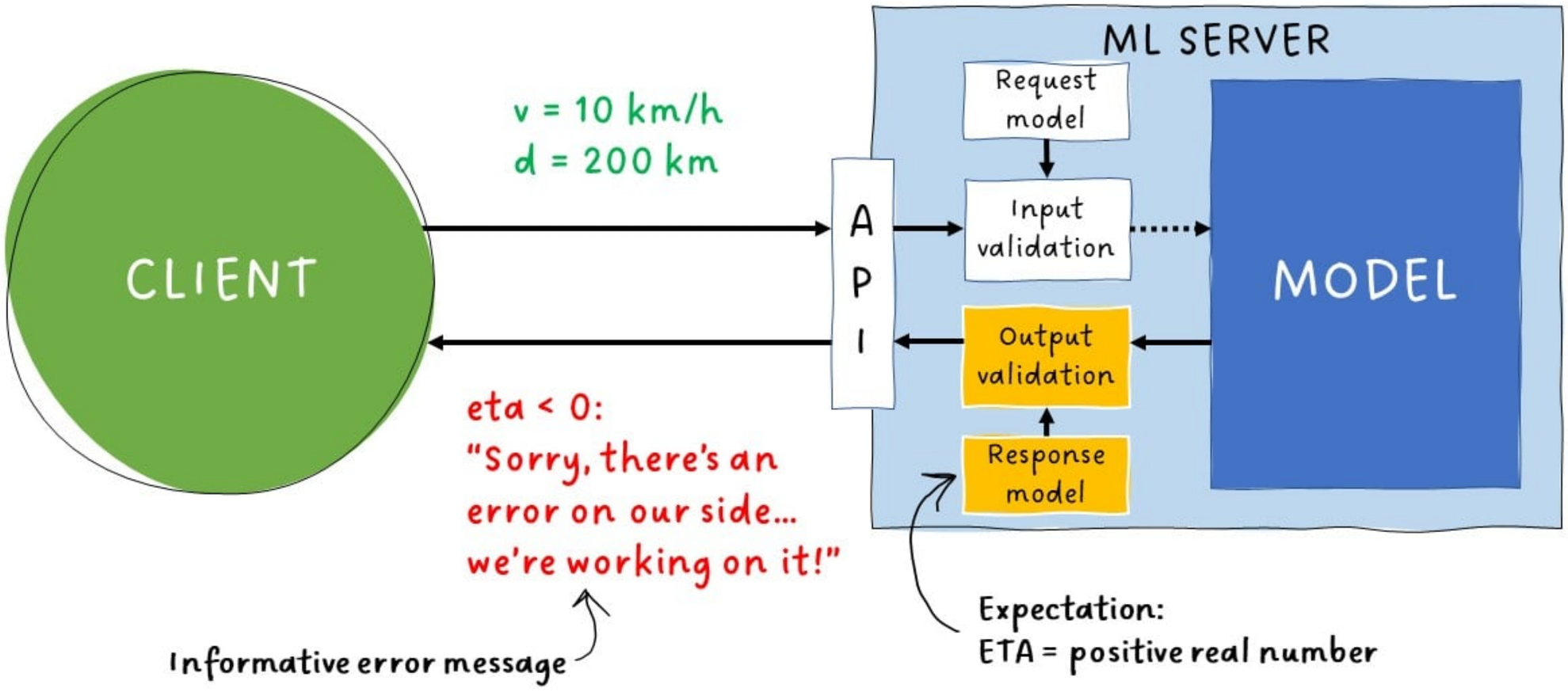

Expectations:
v = positive real number
d = positive real number
ML SERVER
Request model
Input validation
MODEL
CLIENT
A
P
I
v = "asdf"
d = 100 km
"Invalid request! Velocity must be a positive number!"
Informative error message

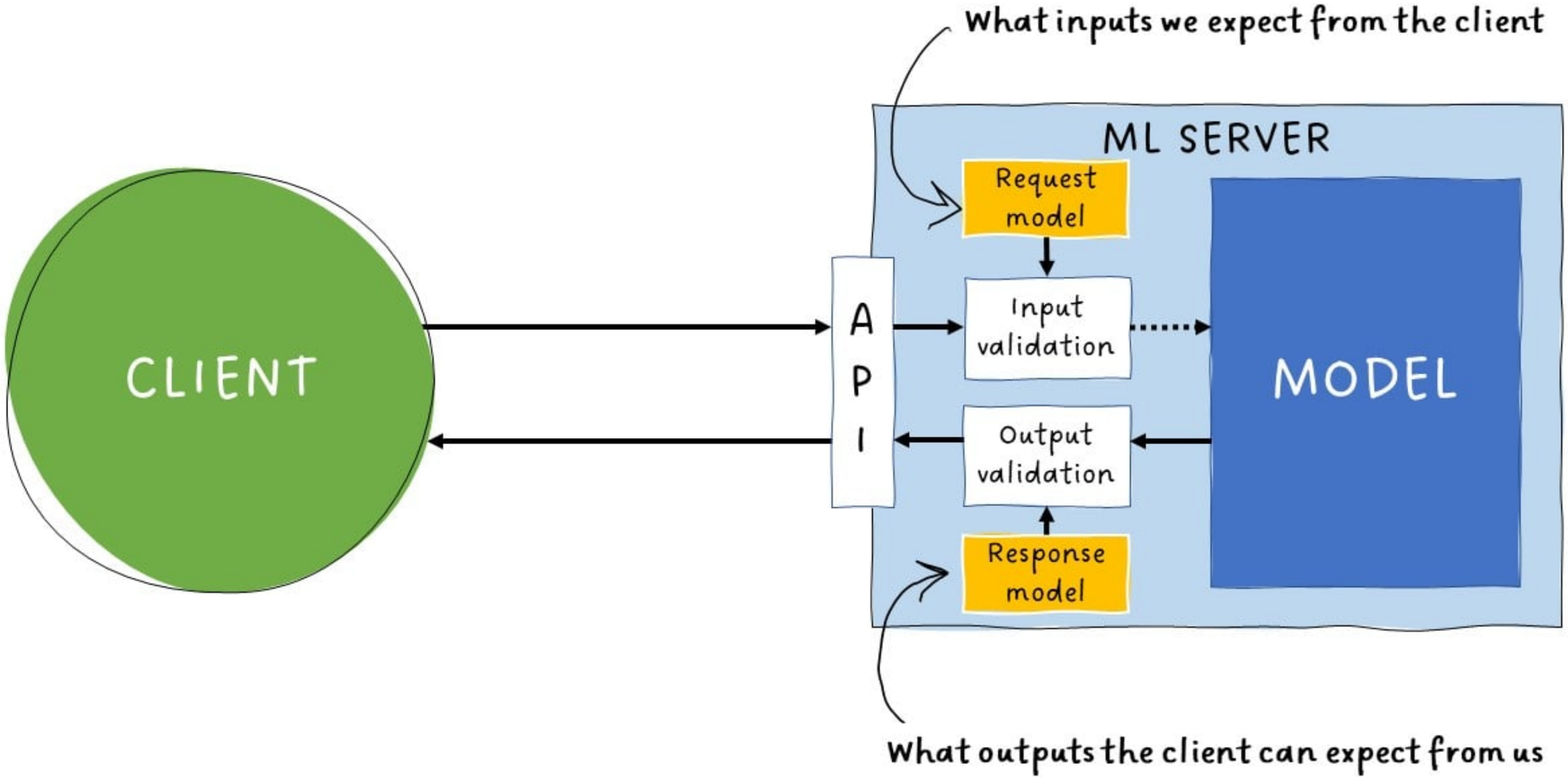

ML SERVER
Request model
Input validation
MODEL
CLIENT
v = 10 km/h
d = 200 km
A
P
I
eta = ??? h

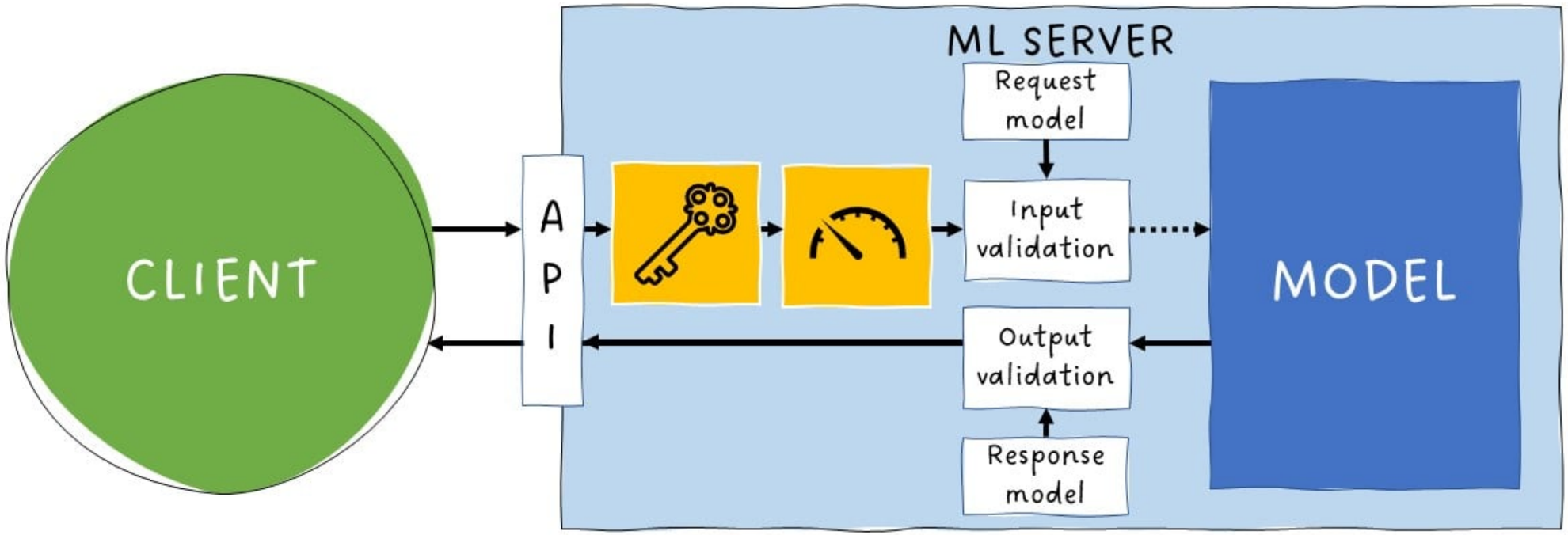

ML SERVER
Request model
Input validation
MODEL
CLIENT
v = 10 km/h
d = 200 km
eta = -200 h
A
P
I

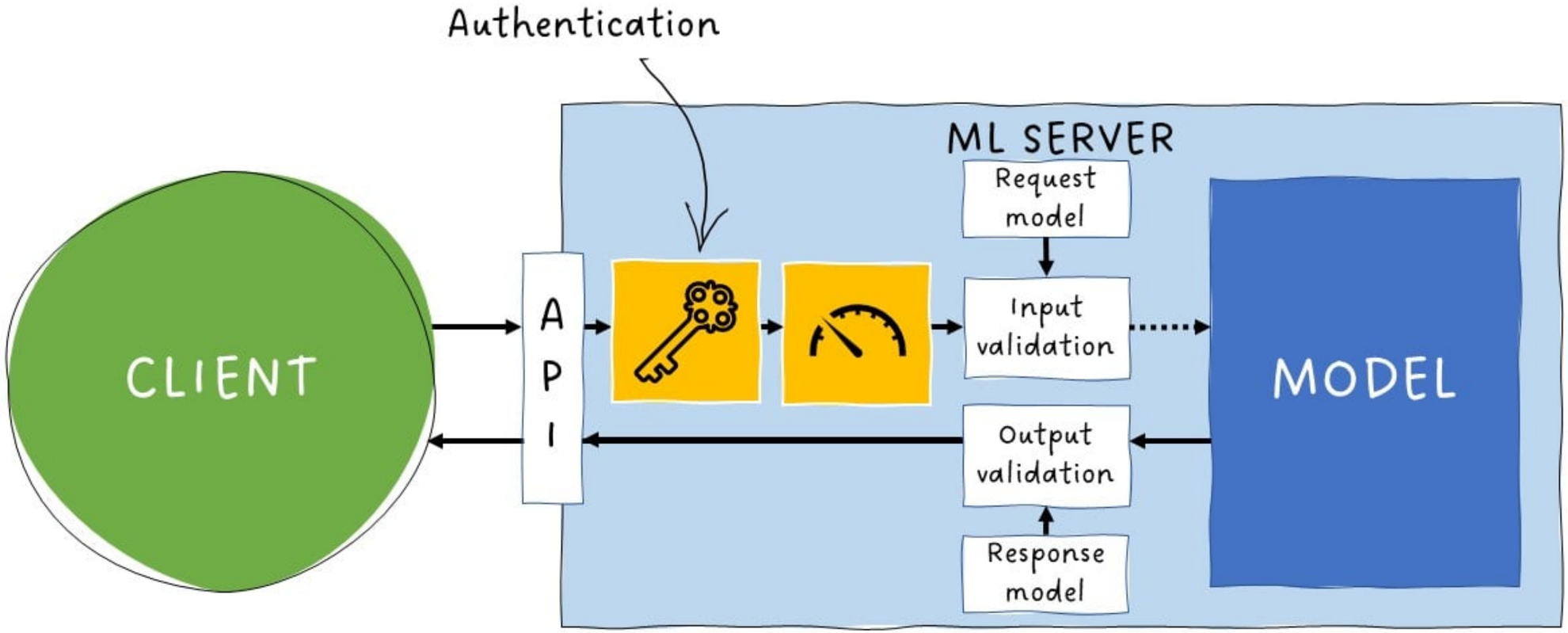

!? !? !? !?
CLIENT
v = 10 km/h
d = 200 km
eta = -200 h
API
ML SERVER
Request model
Input validation
MODEL

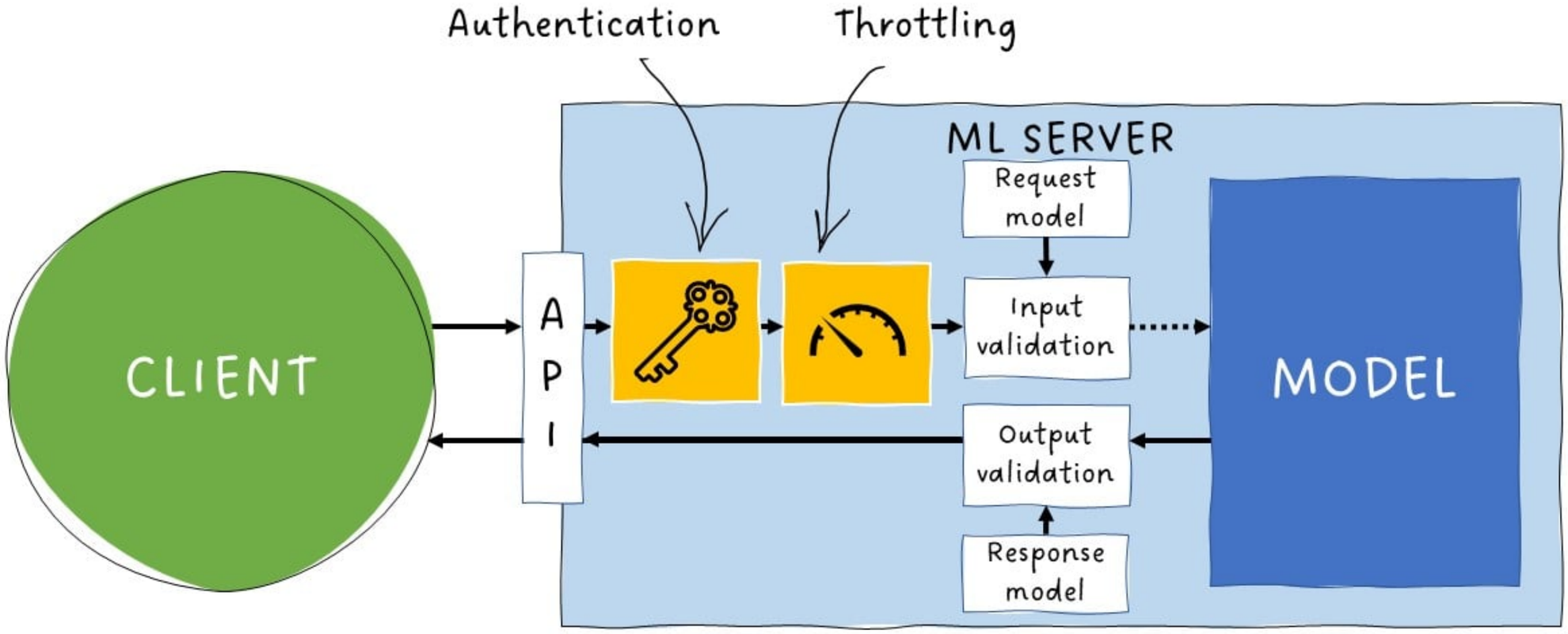

ML SERVER
Request model
Input validation
Output validation
Response model
MODEL
A
P
I
CLIENT
v = 10 km/h
d = 200 km
Expectation:
ETA = positive real number


ML SERVER
Request model
Input validation
Output validation
Response model
MODEL
A
P
I
CLIENT
v = 10 km/h
d = 200 km
eta < 0:
"Sorry, there's an error on our side... we're working on it!"
Informative error message
Expectation:
ETA = positive real number


What inputs we expect from the client
ML SERVER
Request model
Input validation
Output validation
Response model
MODEL
API
CLIENT
What outputs the client can expect from us


ML SERVER
Request model
CLIENT
API
Input validation
MODEL
Output validation
Response model


Authentication
ML SERVER
Request model
CLIENT
A
P
I
Input validation
MODEL
Output validation
Response model


Authentication
Throttling
ML SERVER
Request model
CLIENT
API
Input validation
MODEL
Output validation
Response model

- open-source API framework for Python
- all essential features out of the box
- build and launch REST APIs in no time

[1] https://fastapi.tiangolo.com

Let's practice!

MLOPS DEPLOYMENT AND LIFE CYCLING

Deployment progression and testing

MLOPS DEPLOYMENT AND LIFE CYCLING

Nemanja Radojkovic
Senior Machine Learning Engineer

datacamp

Not so fast!

- Don't rush to deploy without thorough testing!

Agenda

TEST TYPES

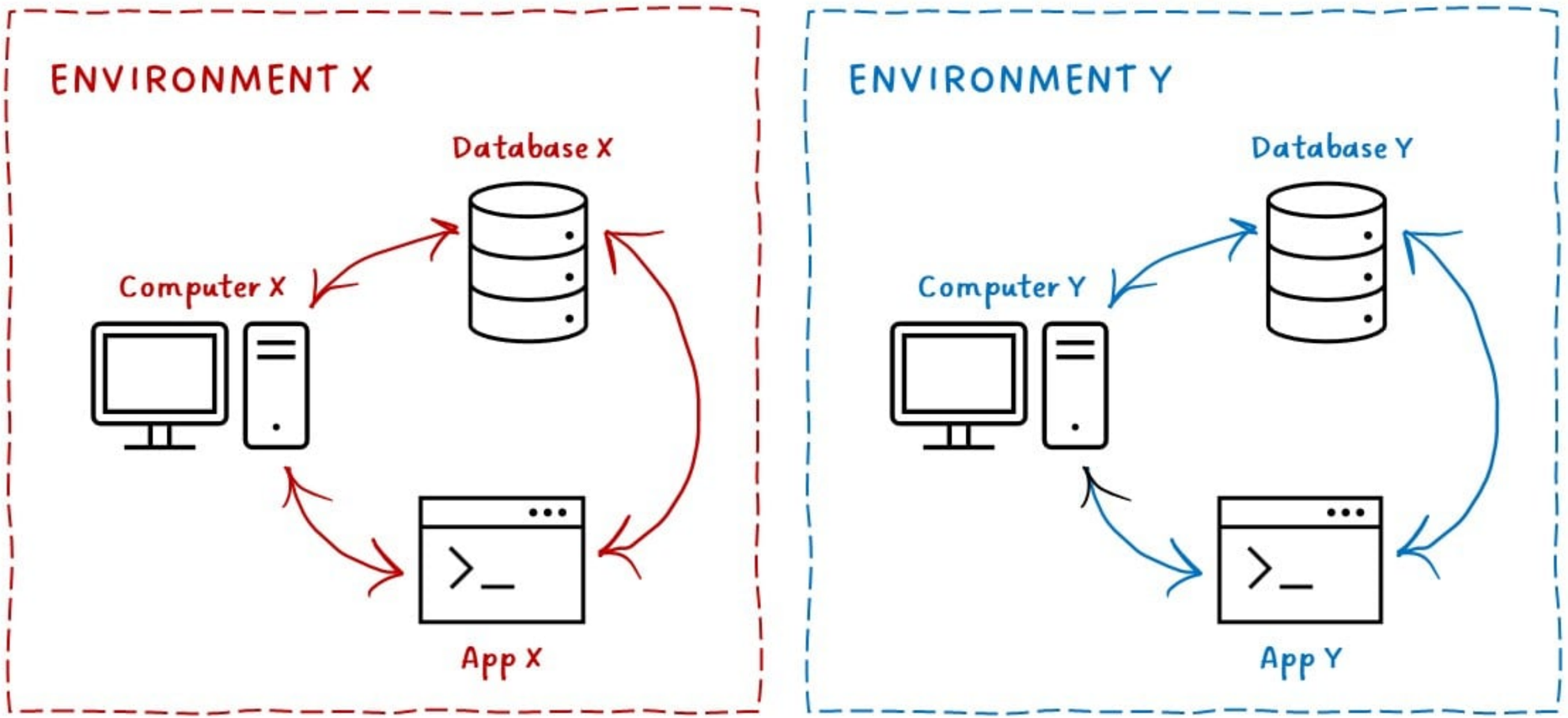
- Unit tests
- Integration tests
- Smoke tests
- Load tests
- Acceptance tests

DEV, DEPLOY, TEST ENVIRONMENTS

- Development
- Test
- Staging
- Production

OUT OF SCOPE: Predictive model performance

IN SCOPE: Testing the overall ML app


ENVIRONMENT X
Database X
Computer X
App X
ENVIRONMENT Y
Database Y
Computer Y
App Y

Unit testing

Checking if independent code units behave as expected

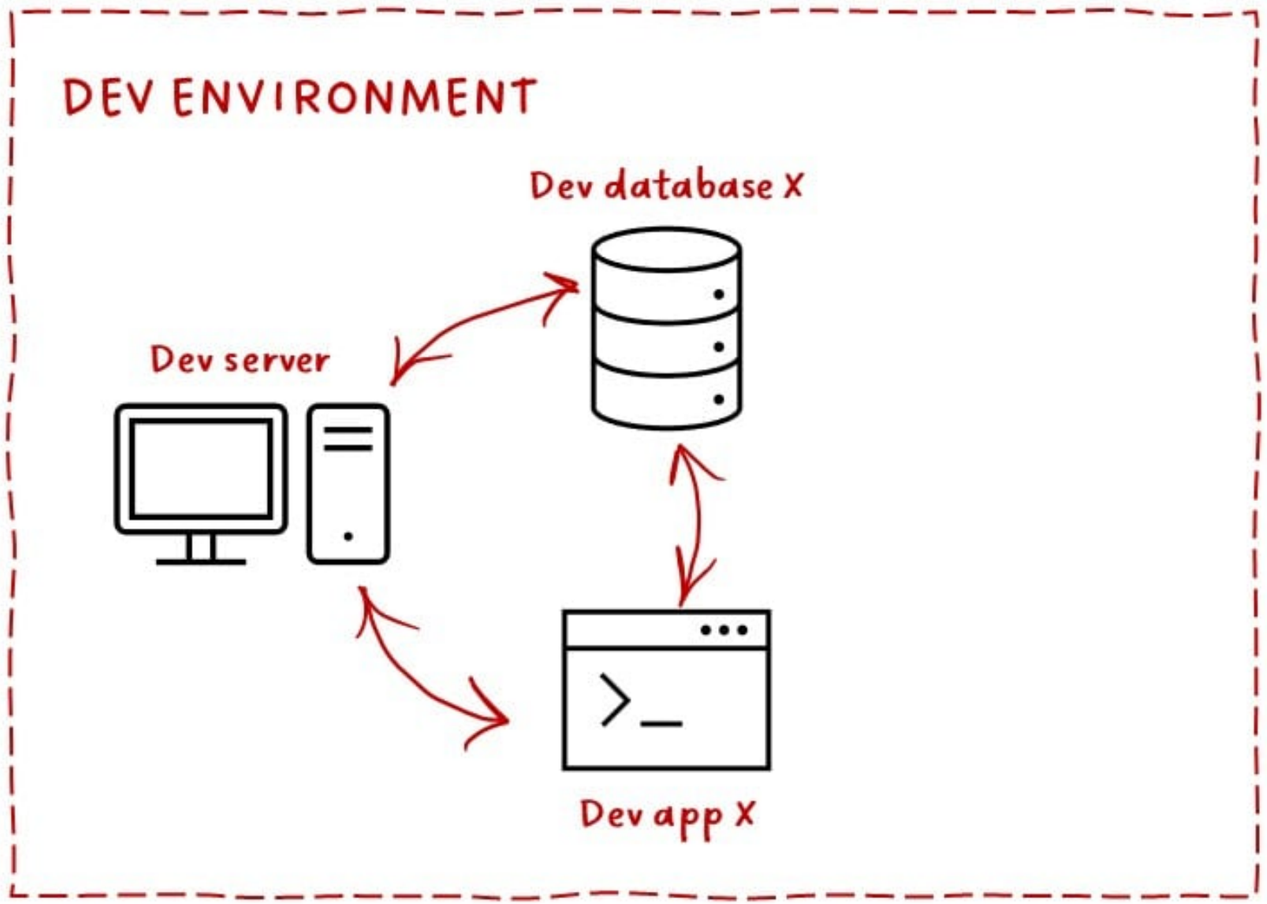
EXAMPLE

Function to test:

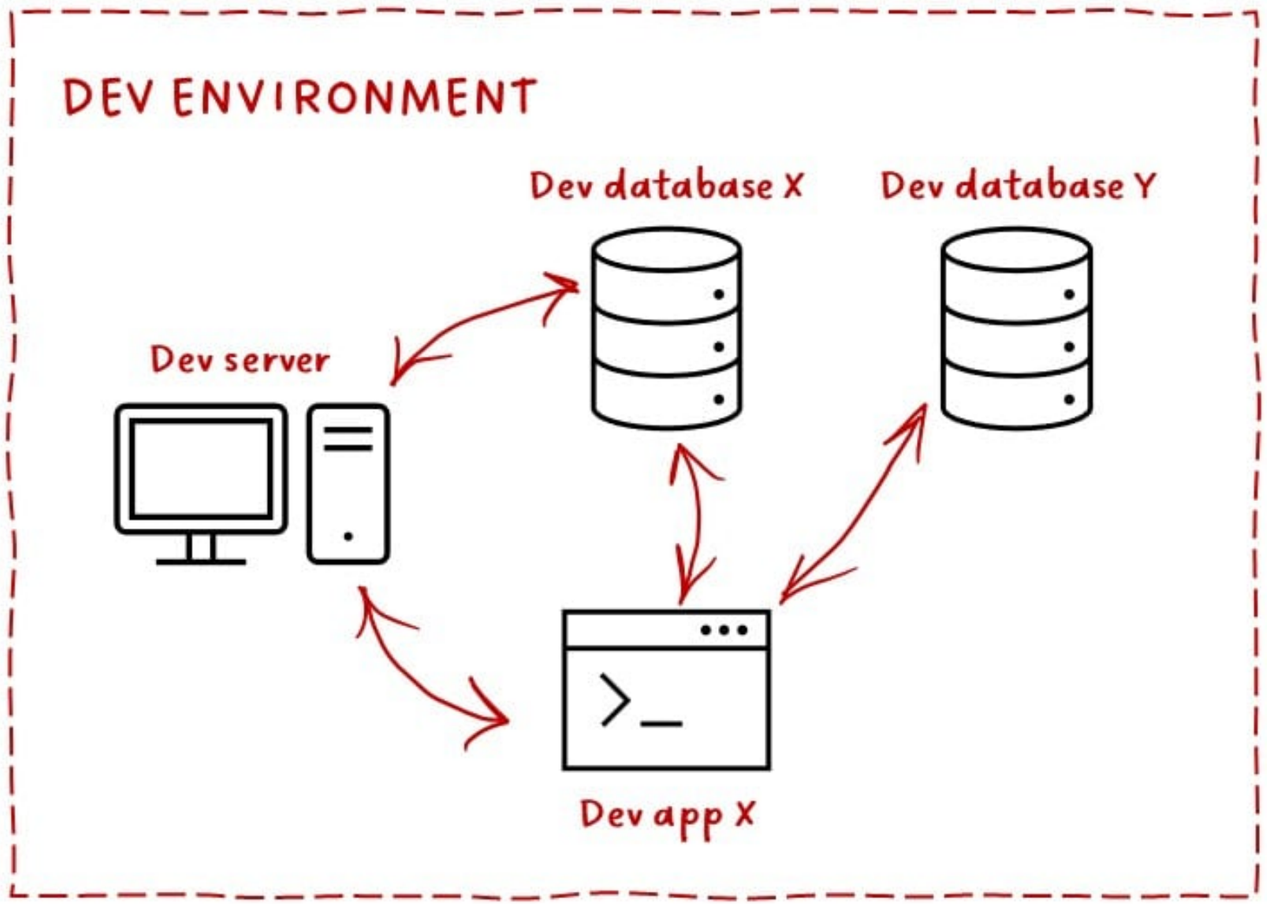
```
def add_two_numbers(x, y):
    return x + y
```

Unit test:

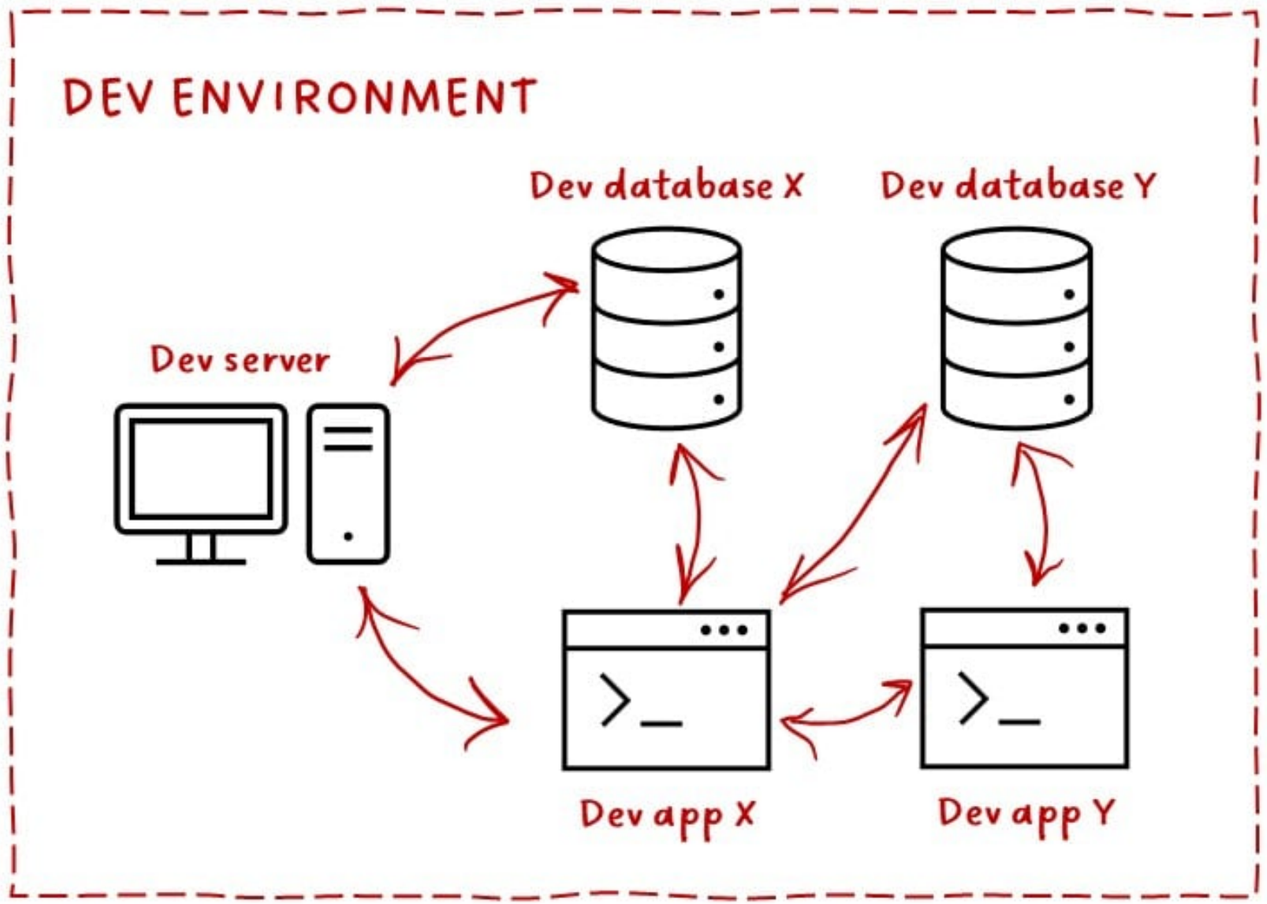
```
add_two_numbers(1, 1) == 2
add_two_numbers(2, 3) == 5
```



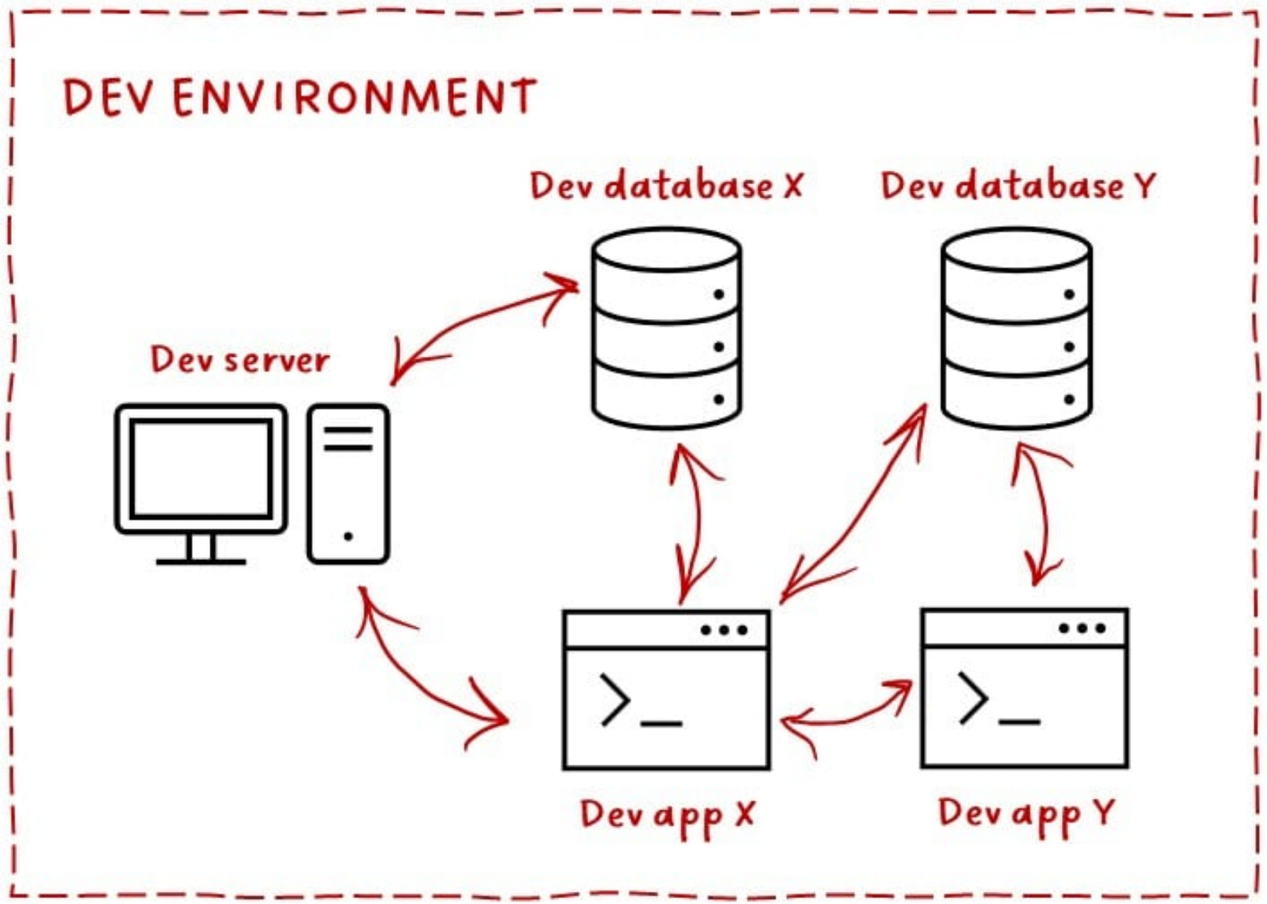
- Experimentation
- Constantly changing



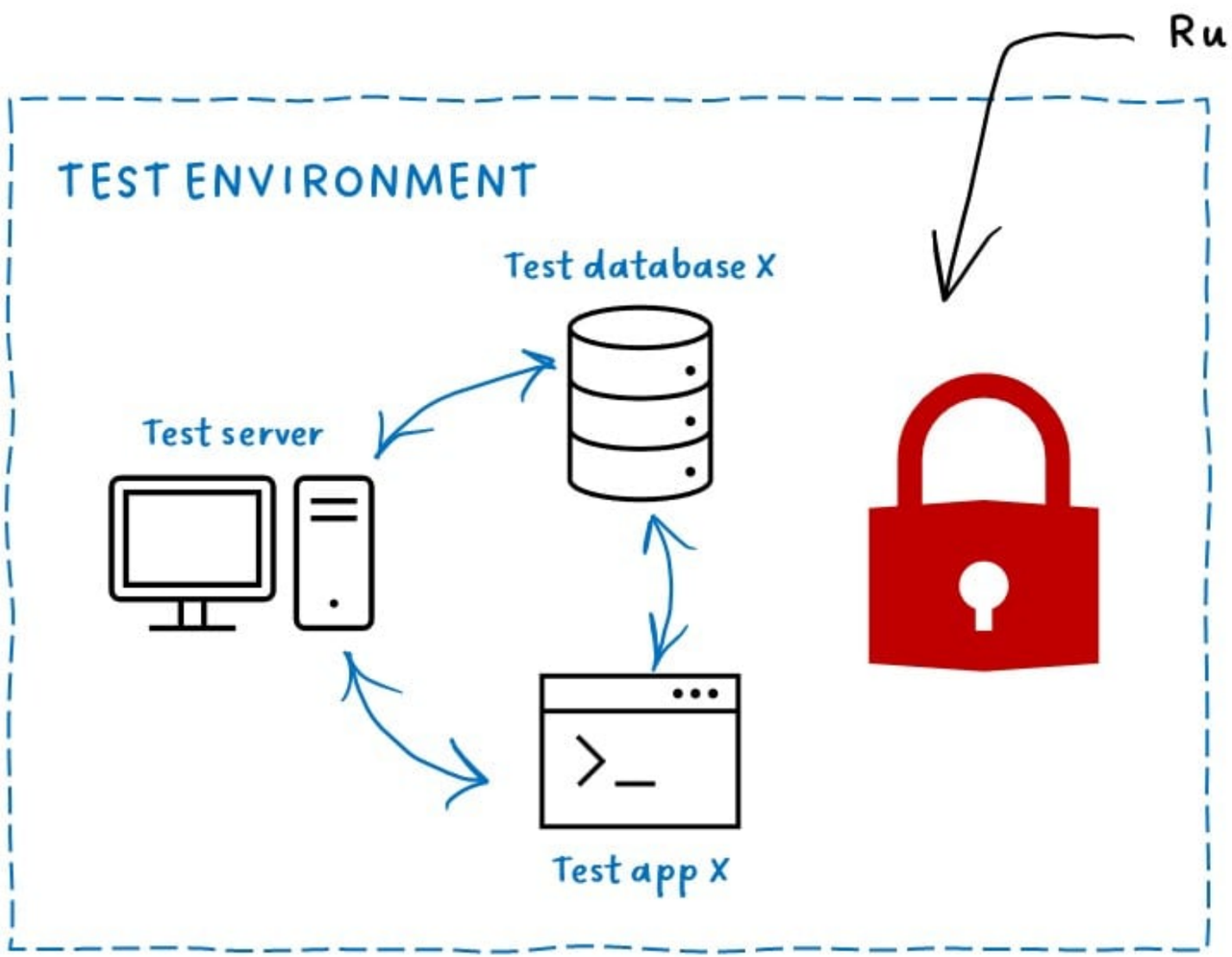
- Experimentation
- Constantly changing



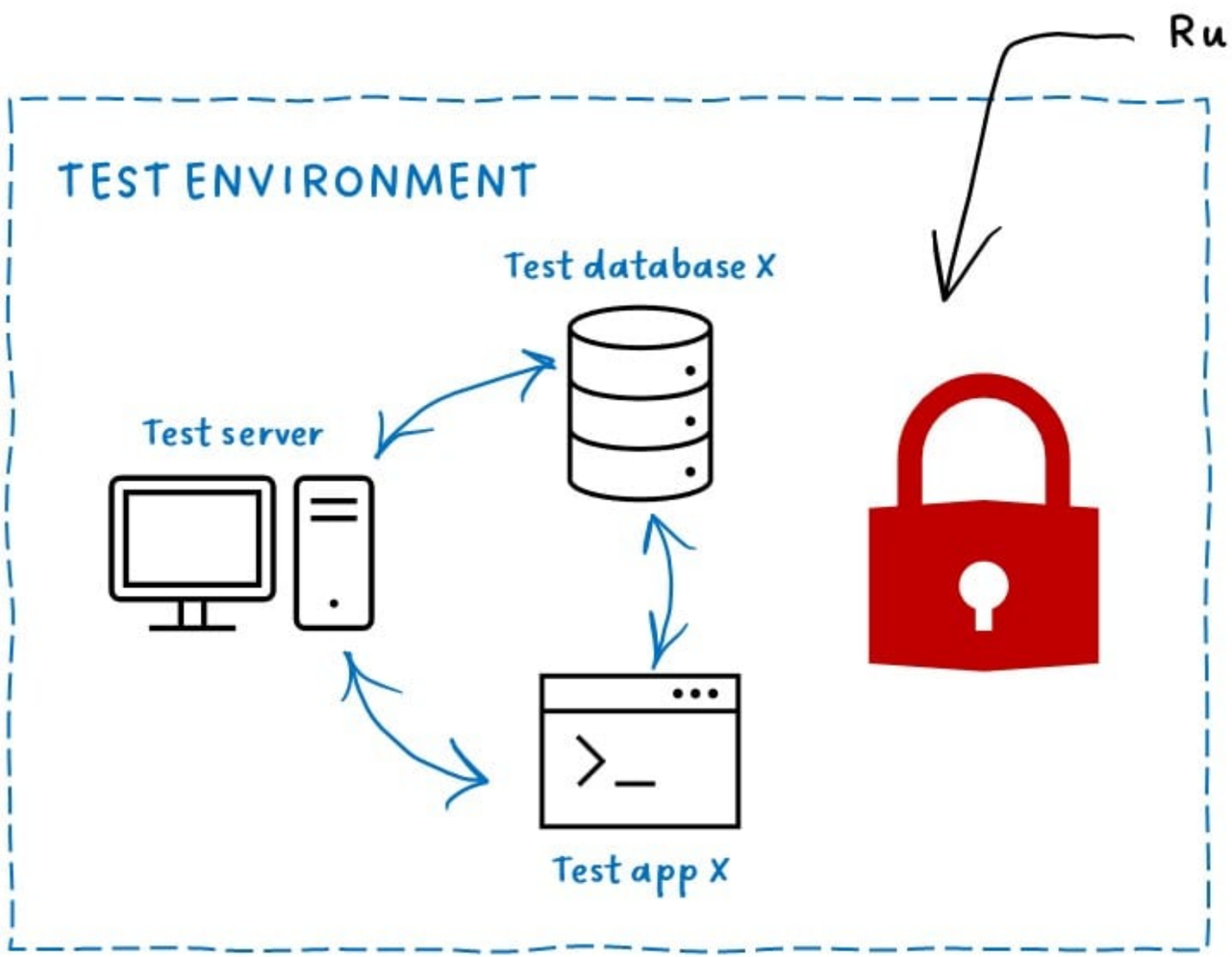
- Experimentation
- Constantly changing



- Experimentation
- Constantly changing
- Tests require stability

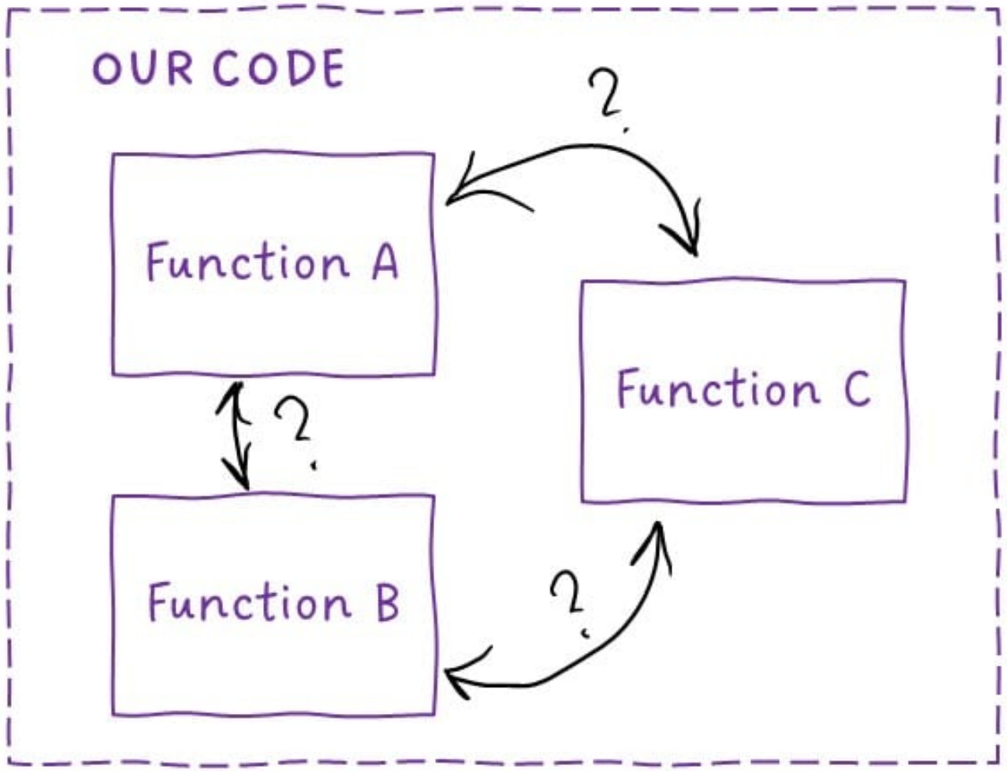


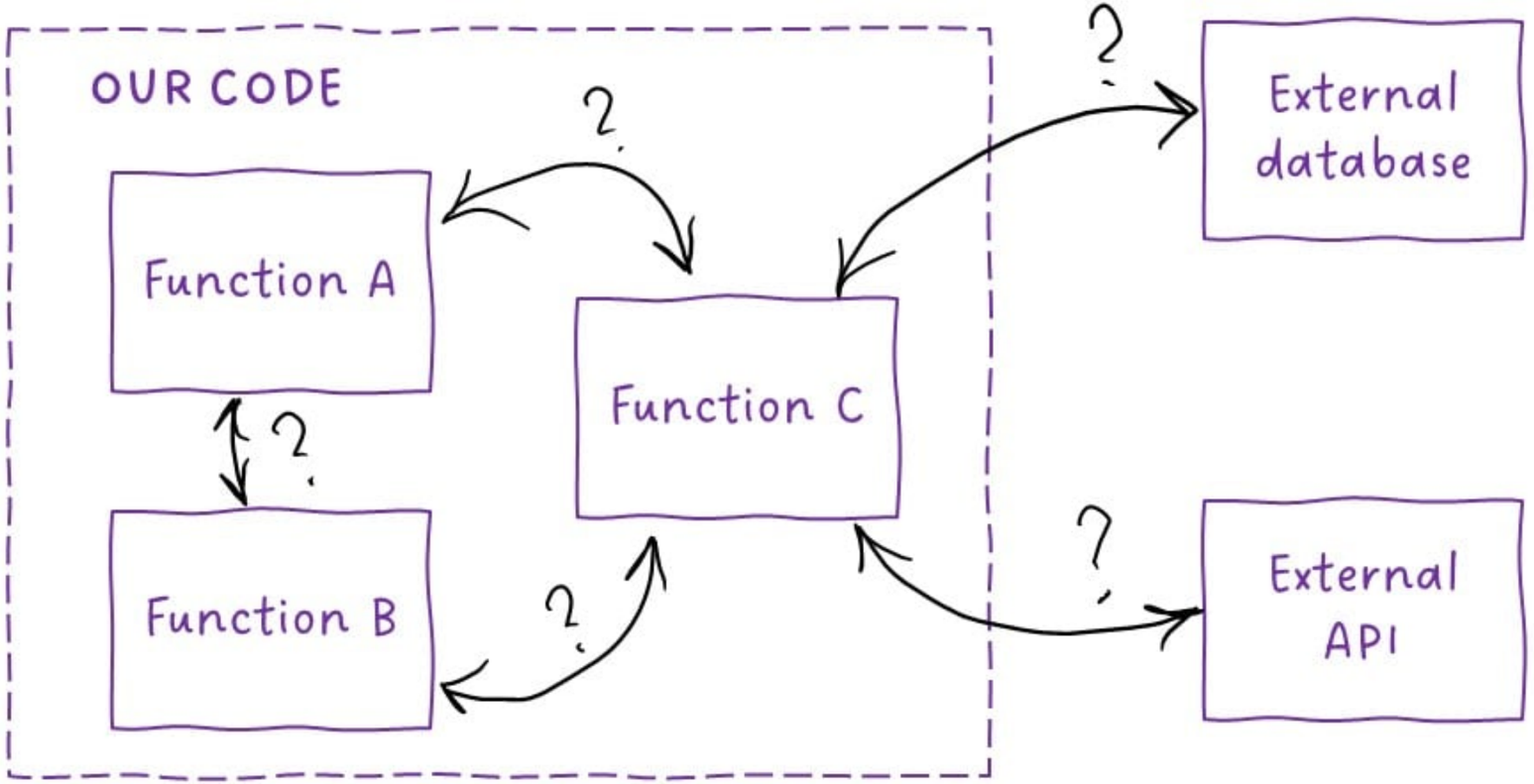
Dedicated, stable TEST environment



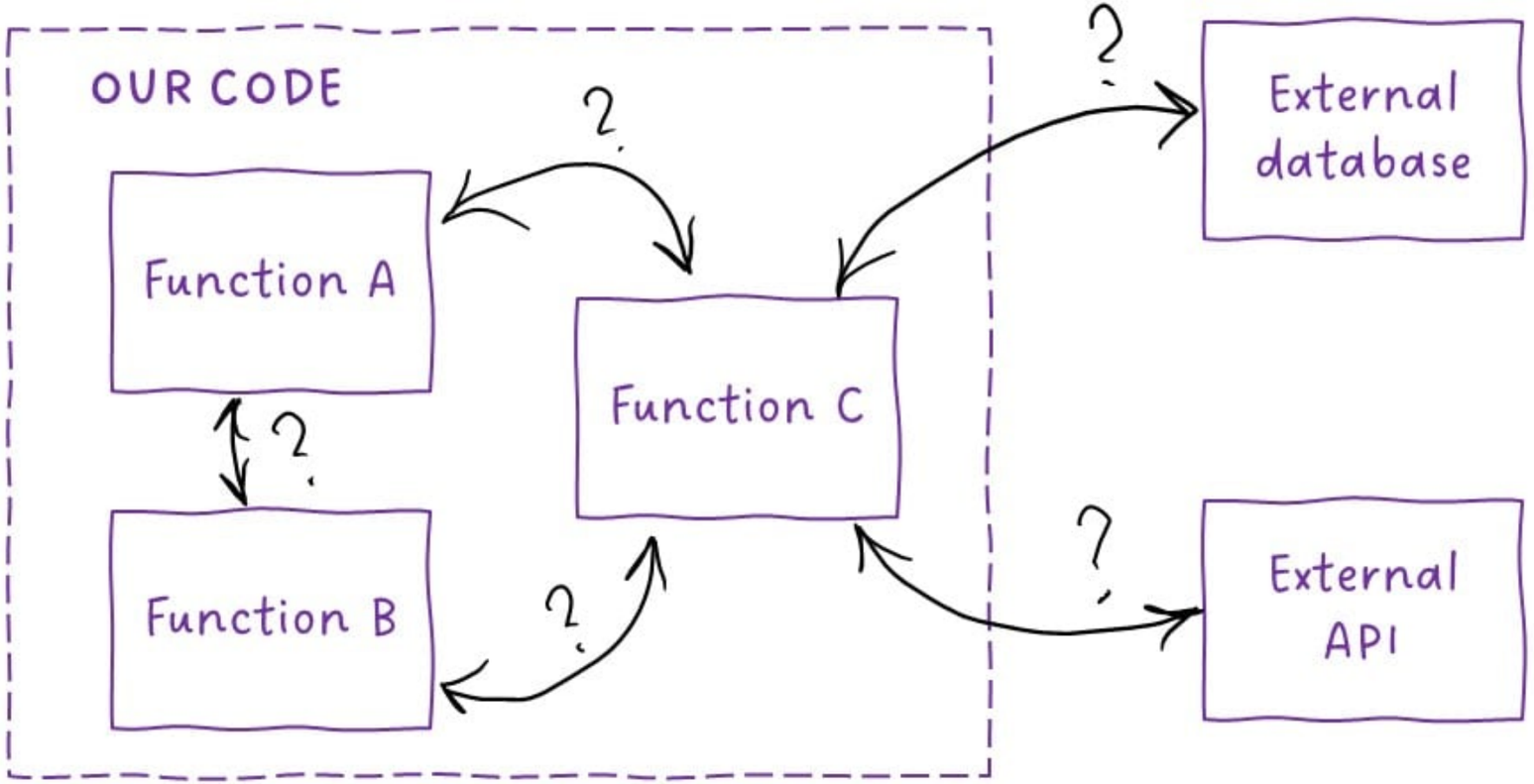
Dedicated, stable TEST environment

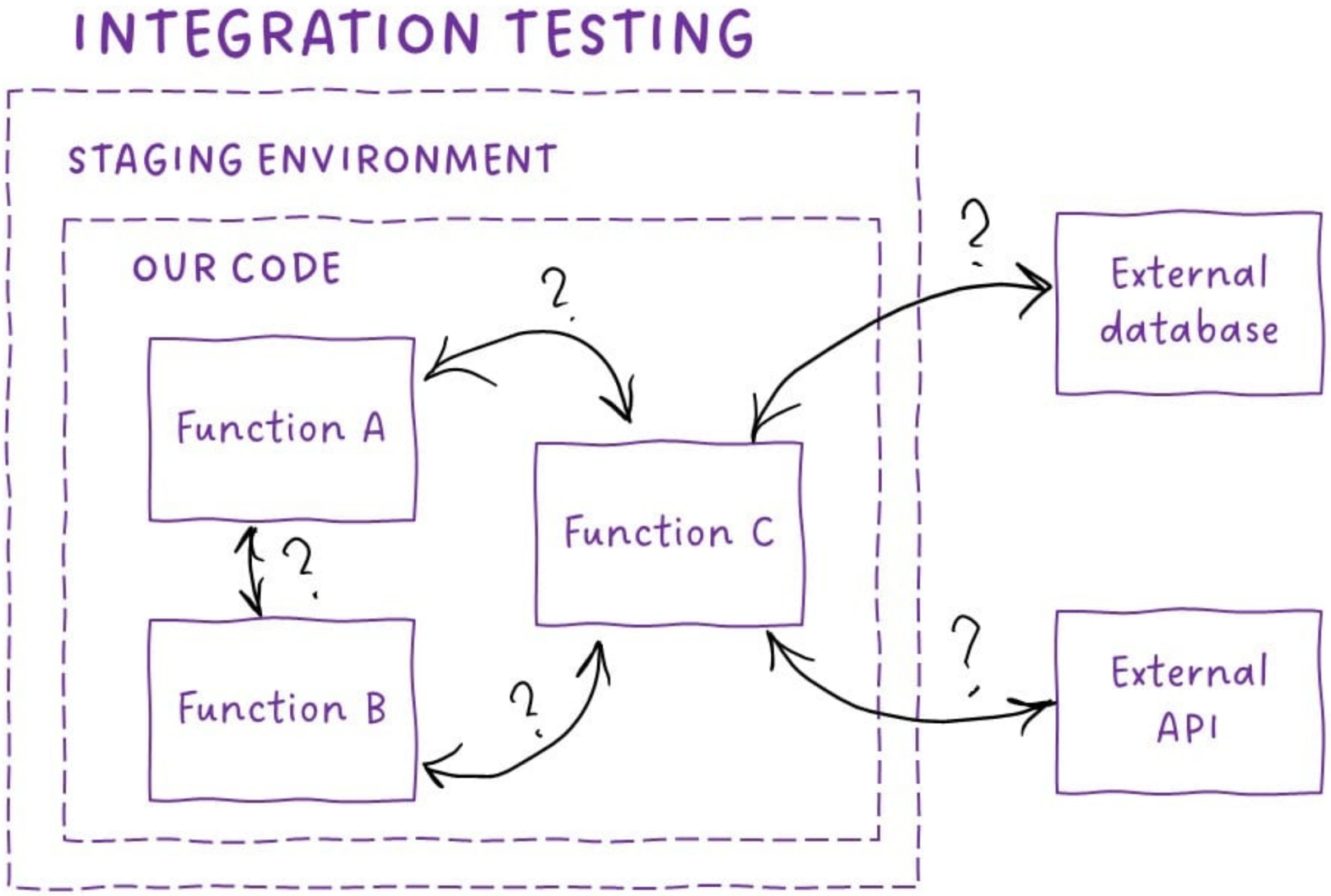
* This also frees-up dev environment resources

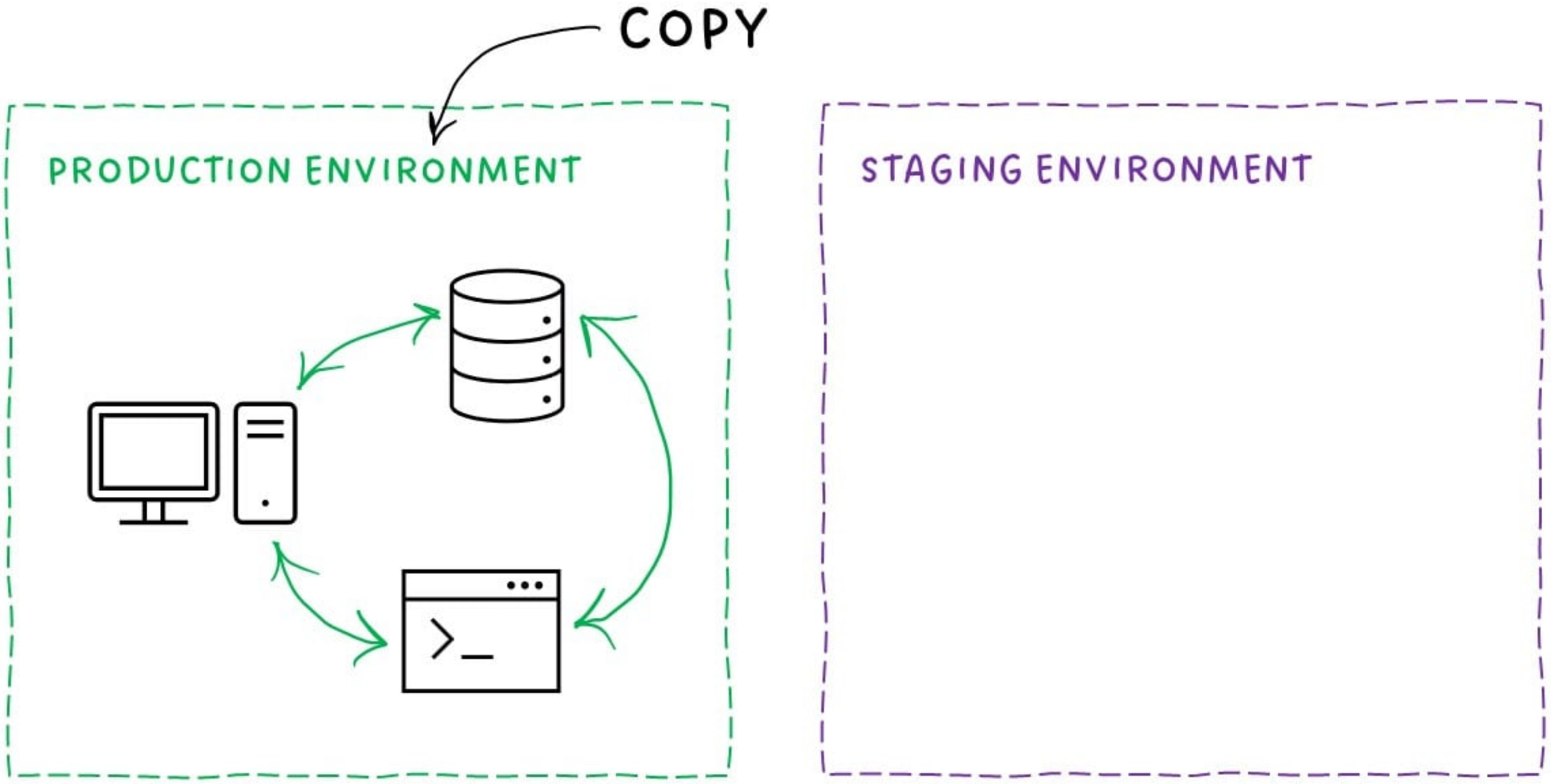

OUR CODE
Function A
Function B
Function C
2.
2.
2.
2.

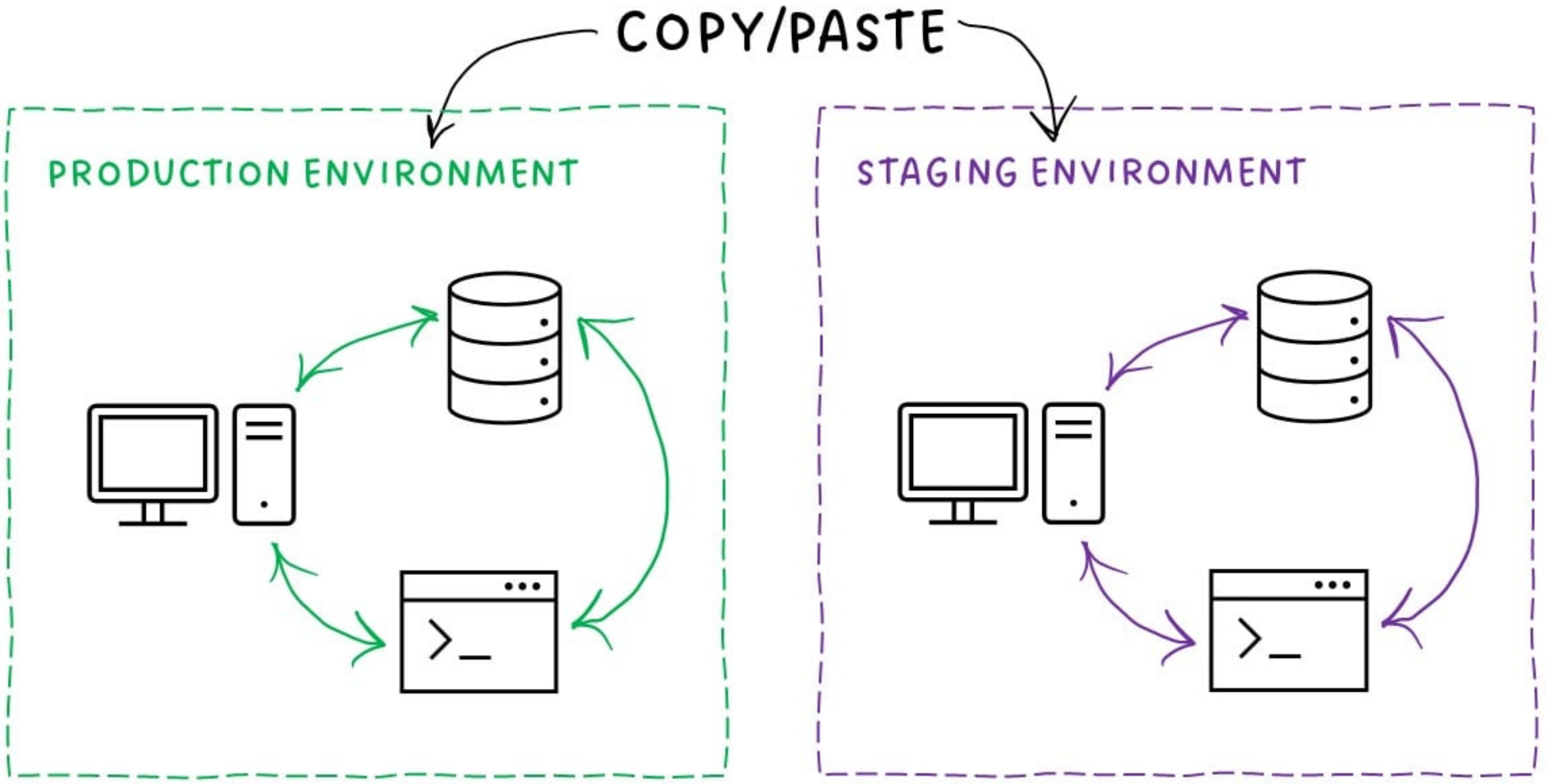

OUR CODE
Function A
Function B
Function C
External database
External API
?

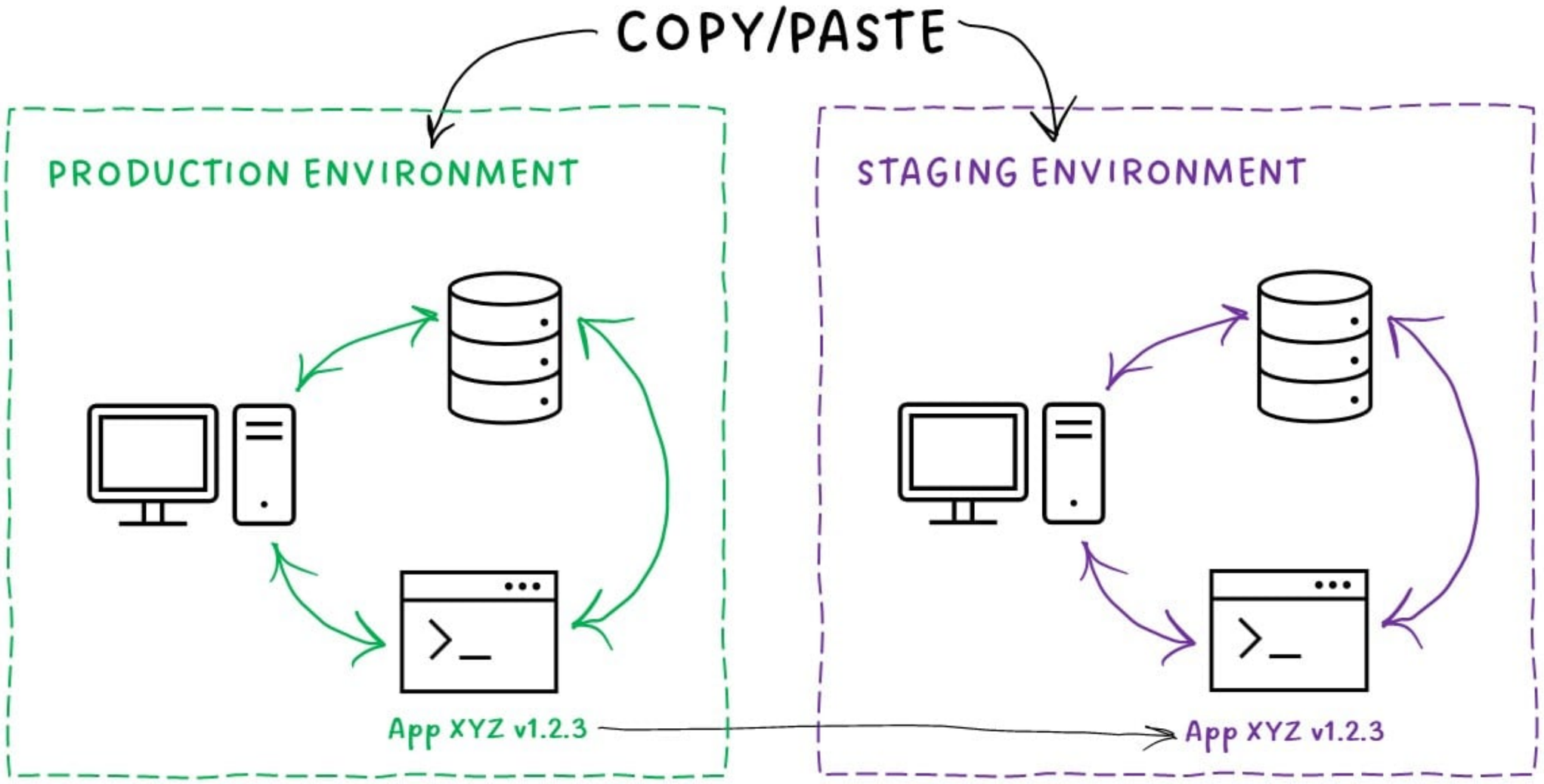
INTEGRATION TESTING

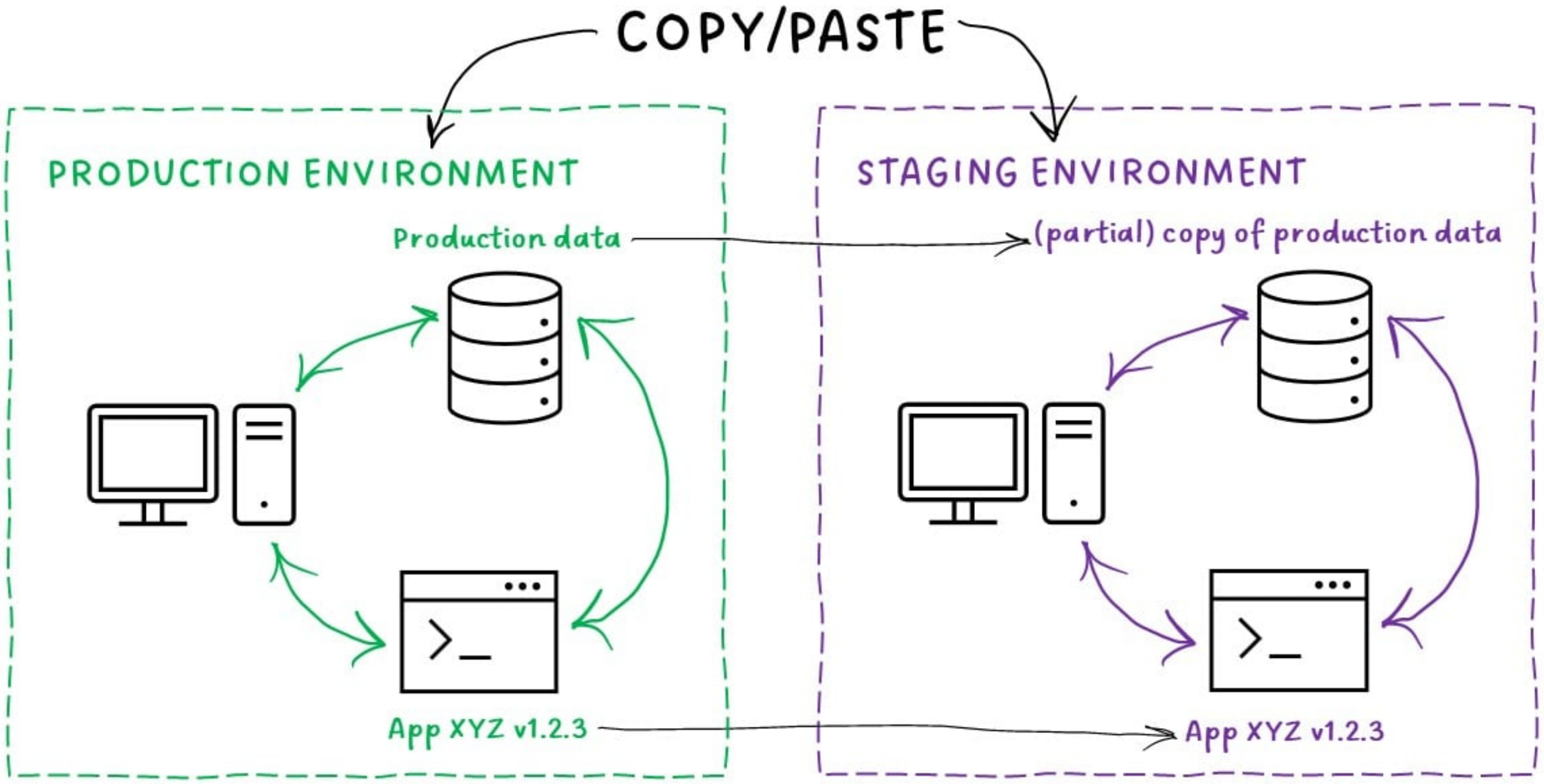


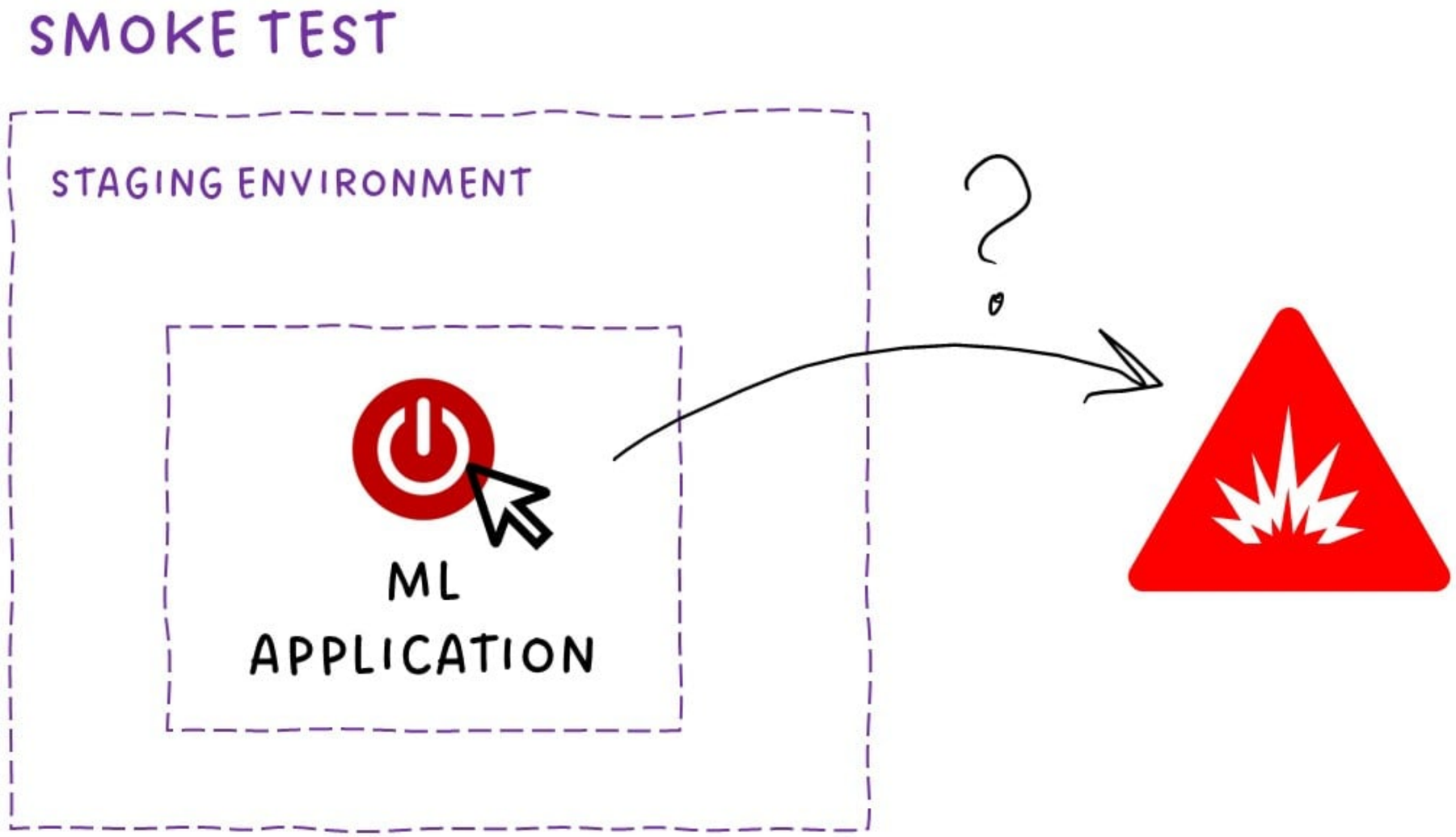

INTEGRATION TESTING
STAGING ENVIRONMENT
OUR CODE
Function A
Function B
Function C
External database
External API
?


COPY
PRODUCTION ENVIRONMENT
STAGING ENVIRONMENT

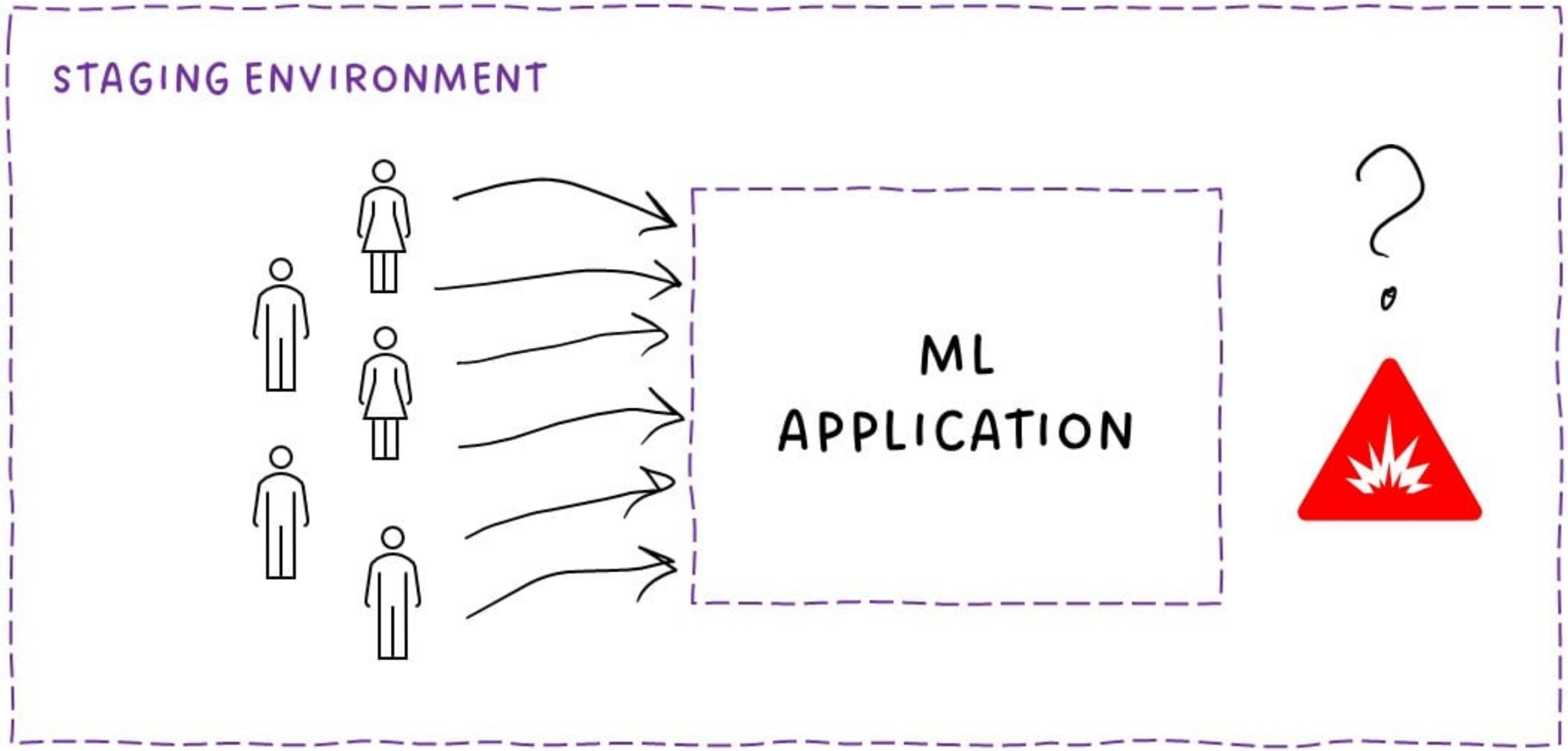

COPY/PASTE
PRODUCTION ENVIRONMENT
STAGING ENVIRONMENT

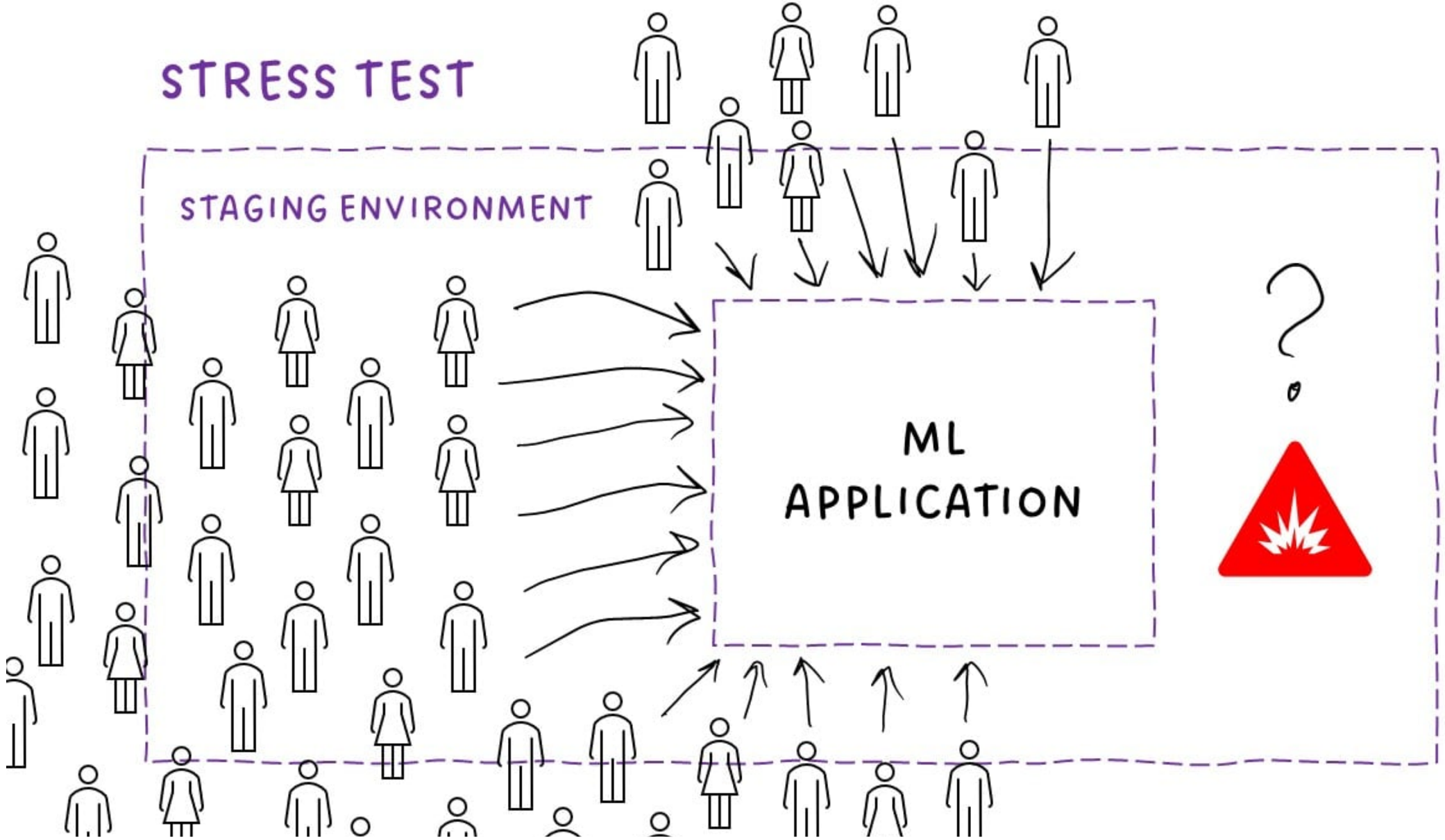

COPY/PASTE
PRODUCTION ENVIRONMENT
STAGING ENVIRONMENT
App XYZ v1.2.3
App XYZ v1.2.3


COPY/PASTE
PRODUCTION ENVIRONMENT
Production data
App XYZ v1.2.3
STAGING ENVIRONMENT
(partial) copy of production data
App XYZ v1.2.3


SMOKE TEST
STAGING ENVIRONMENT
ML
APPLICATION

LOAD TEST



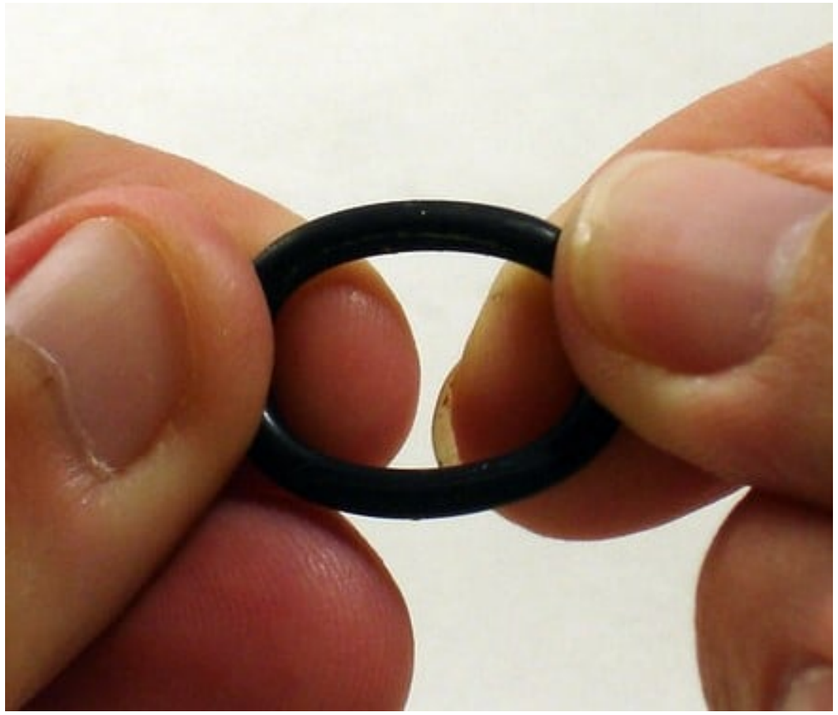

STRESS TEST
STAGING ENVIRONMENT
ML
APPLICATION

USER ACCEPTANCE TESTING (UAT)

Final "seal of approval" before deploying to production

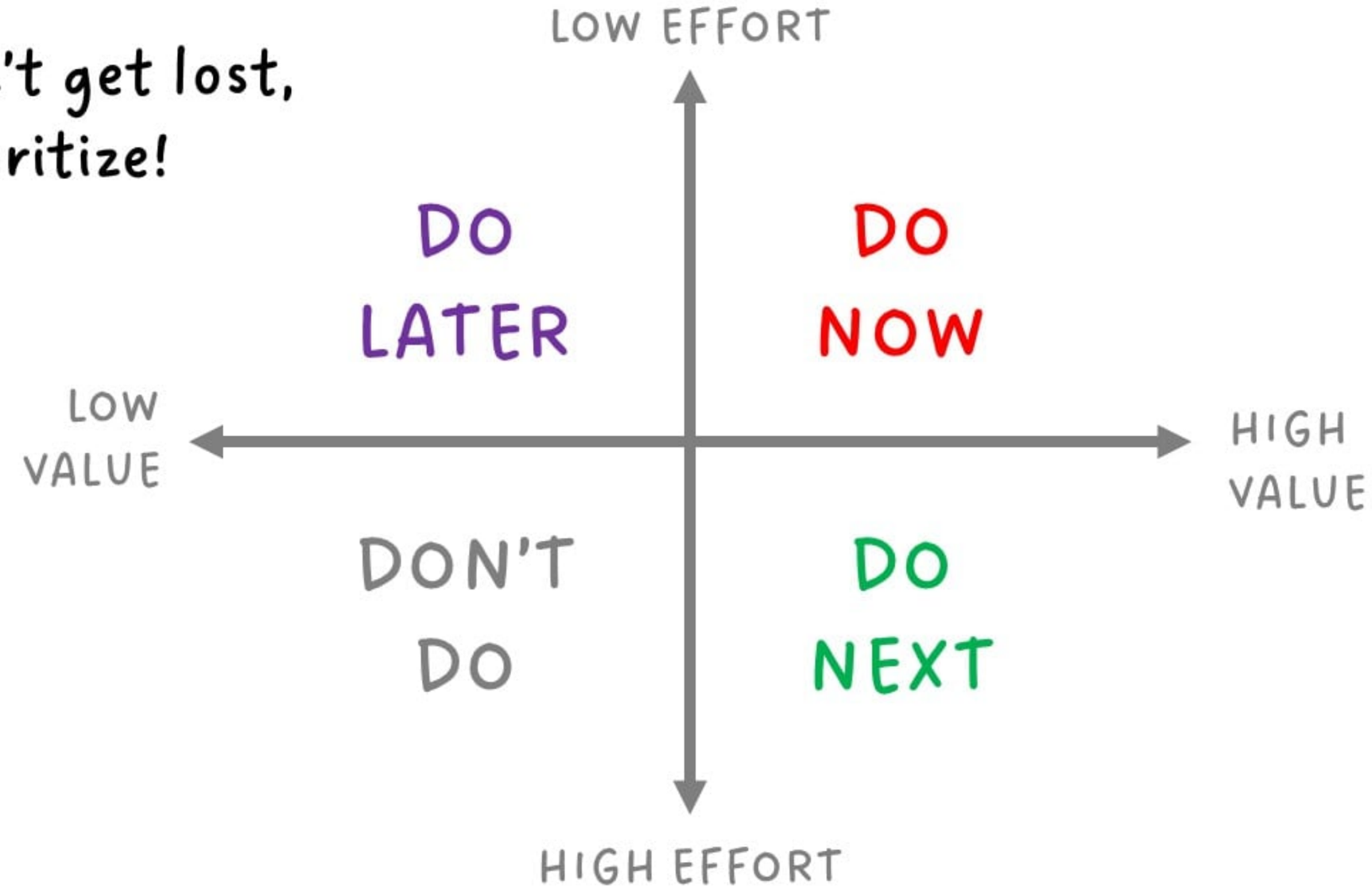
A single weak link breaks the system

We must be careful what we introduce in production.



The Challenger shuttle disaster in 1986 was caused by a failed rubber sealing ring

Don't get lost, prioritize!



Let's practice!

MLOPS DEPLOYMENT AND LIFE CYCLING

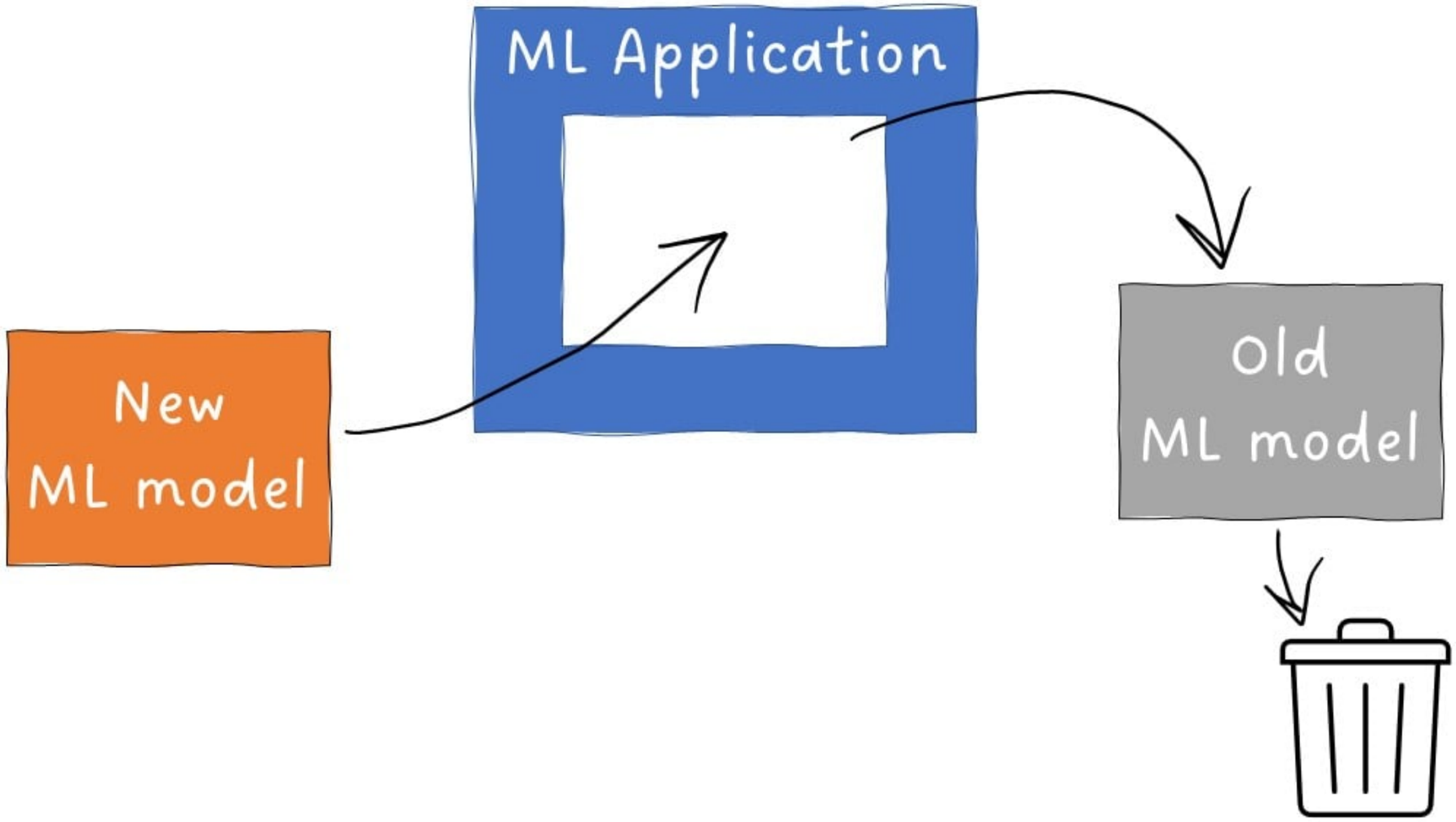
Model deployment strategies

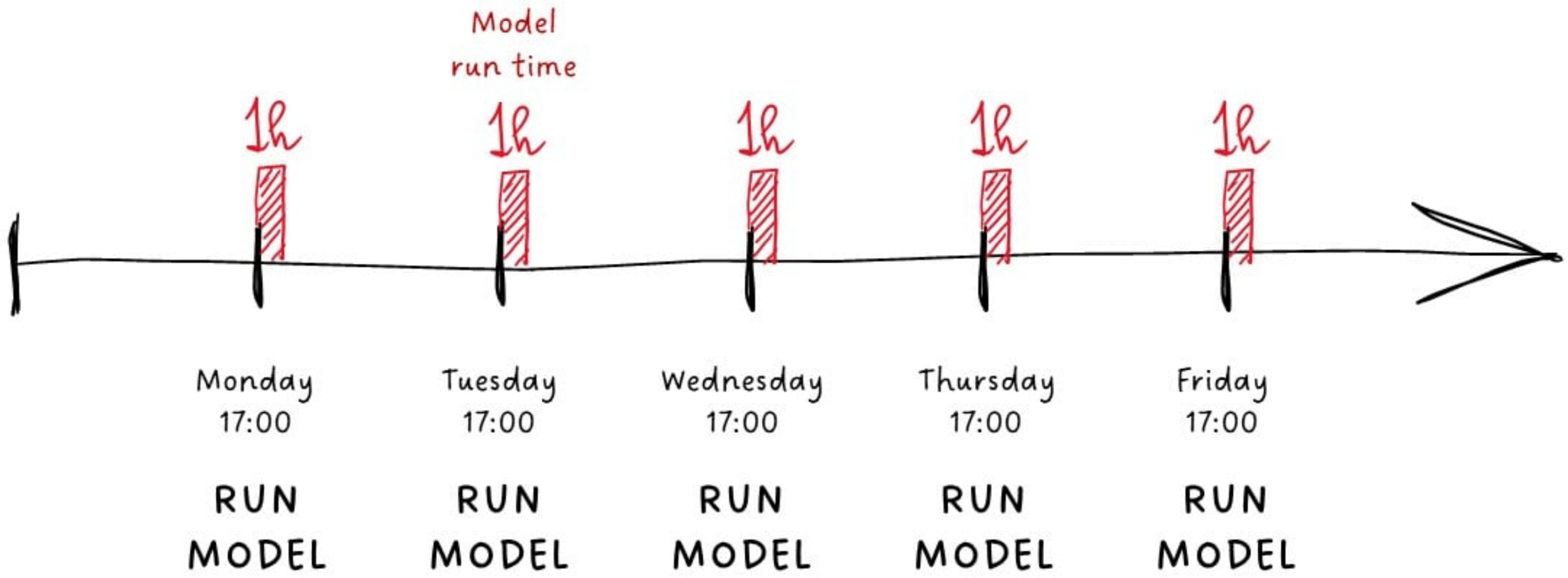
MLOPS DEPLOYMENT AND LIFE CYCLING

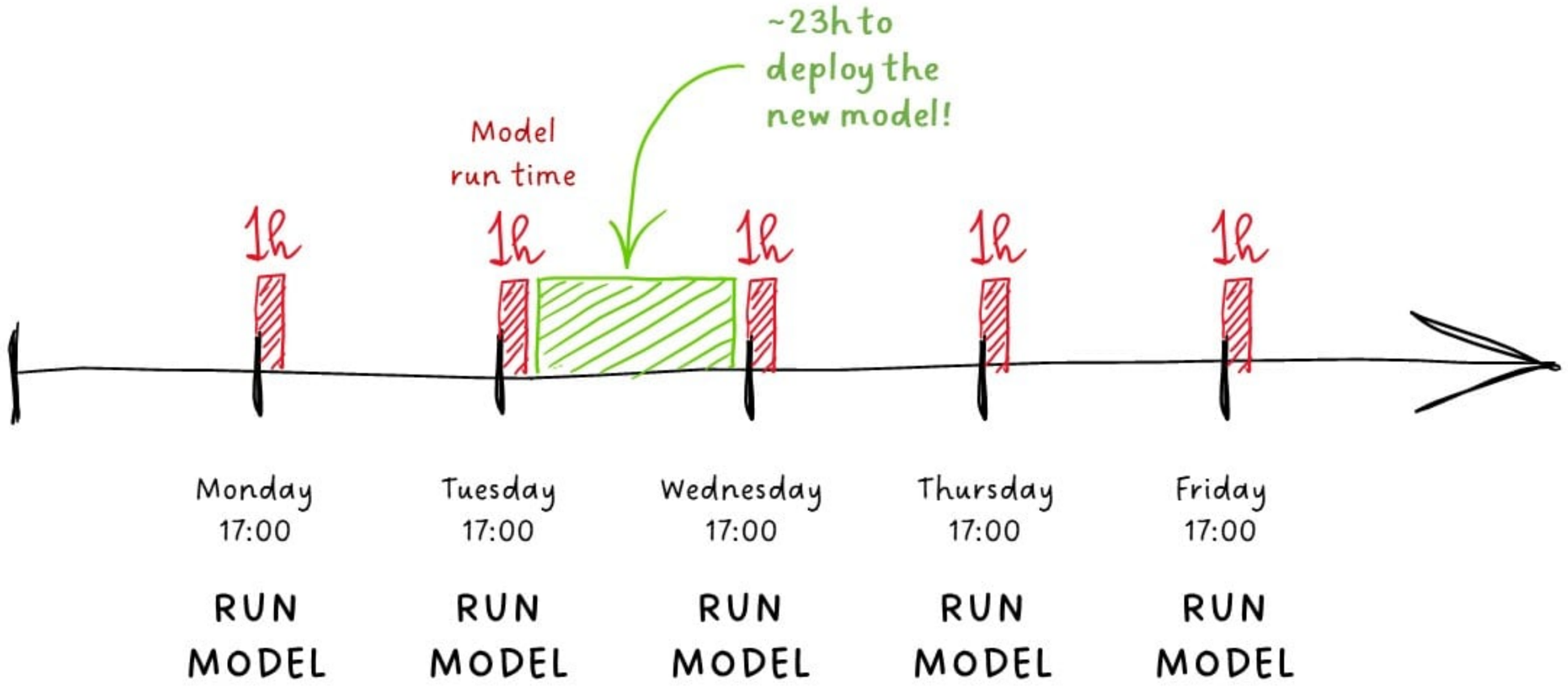
Nemanja Radojkovic
Senior Machine Learning Engineer

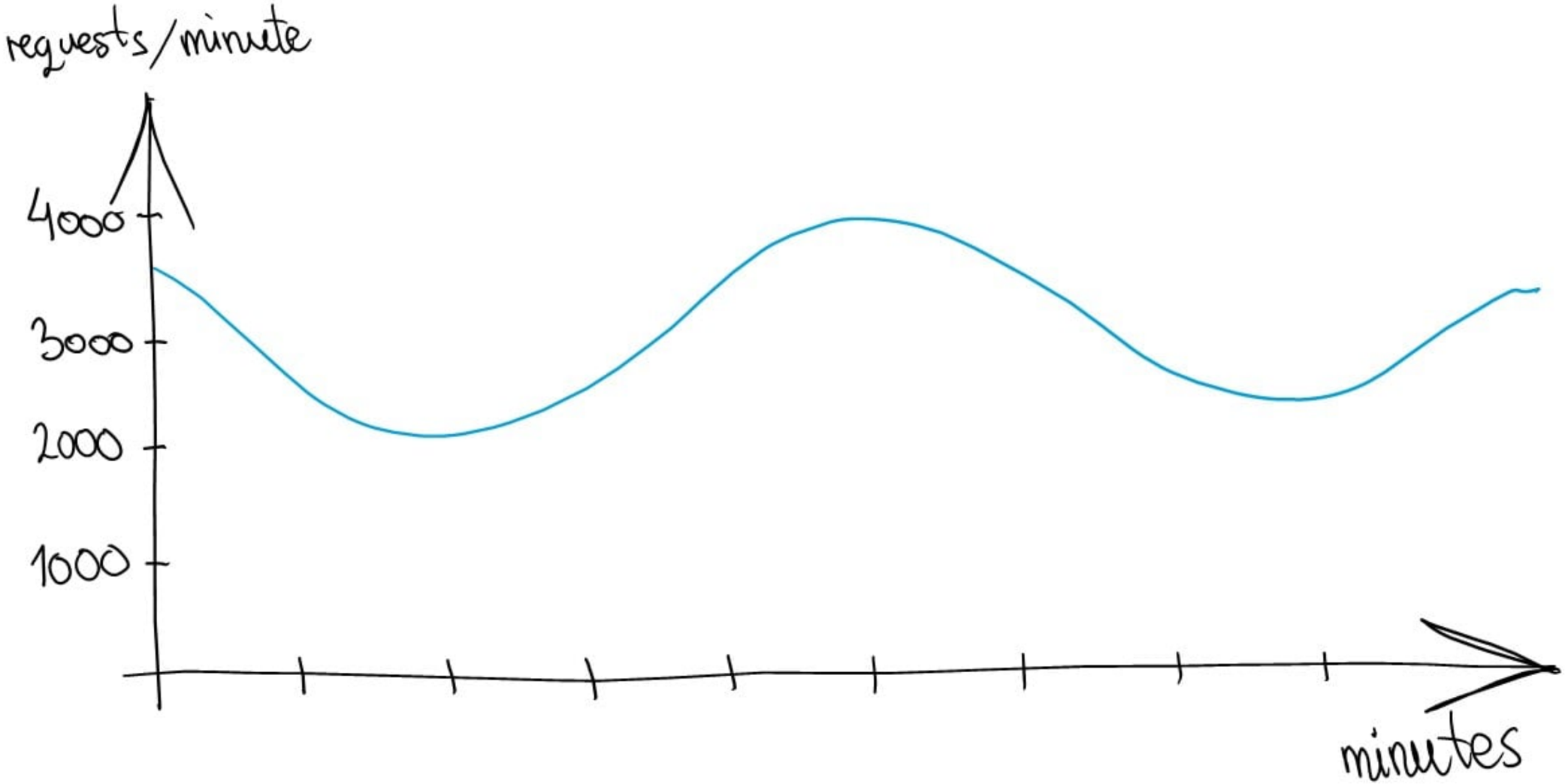
Deployment successful!

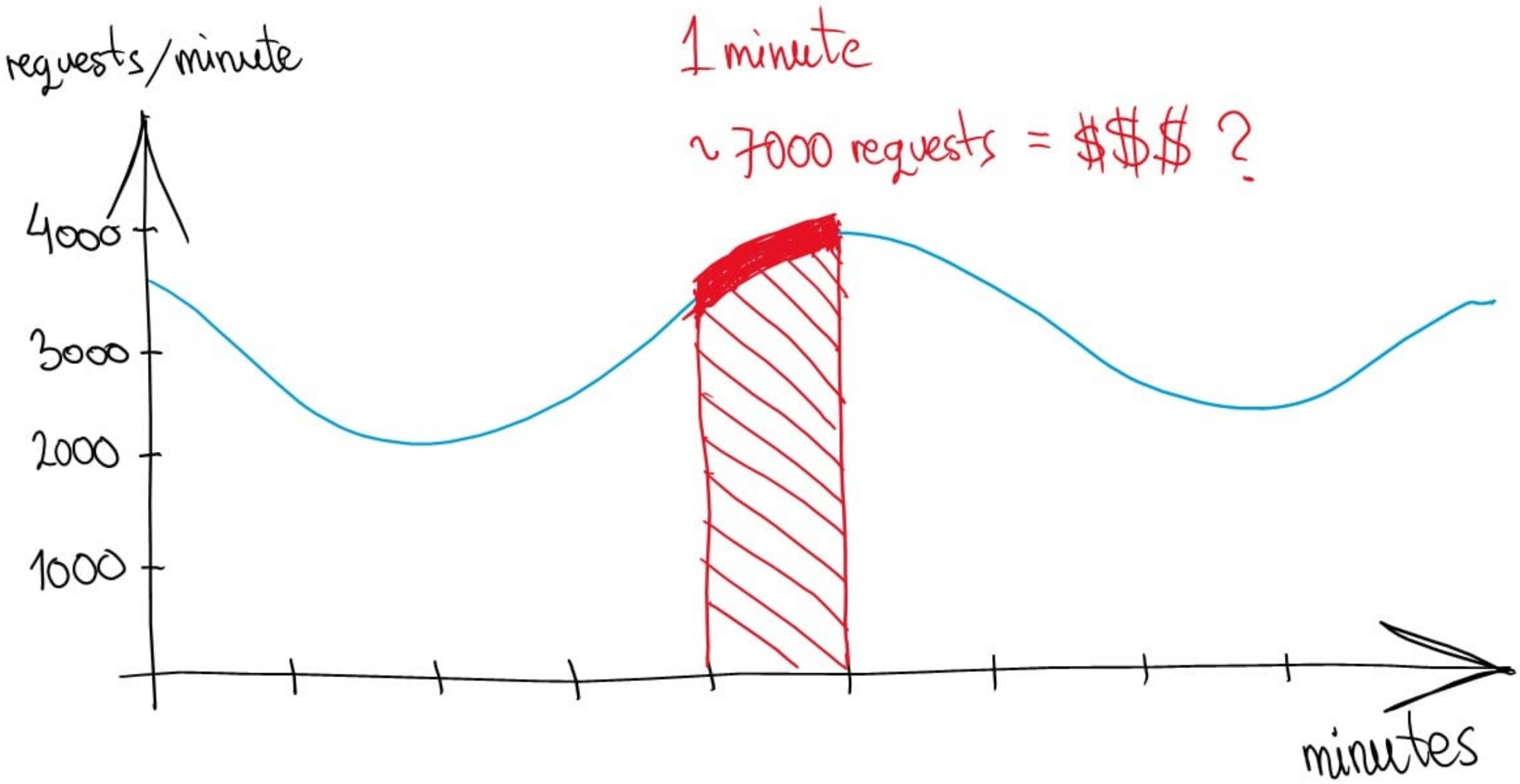
- ML app is running
- API handles 1000s of requests/hour
- One month later:
  - New, better features found
  - Collected new batch of training data
  - Ran the model build pipeline
  - New model package created

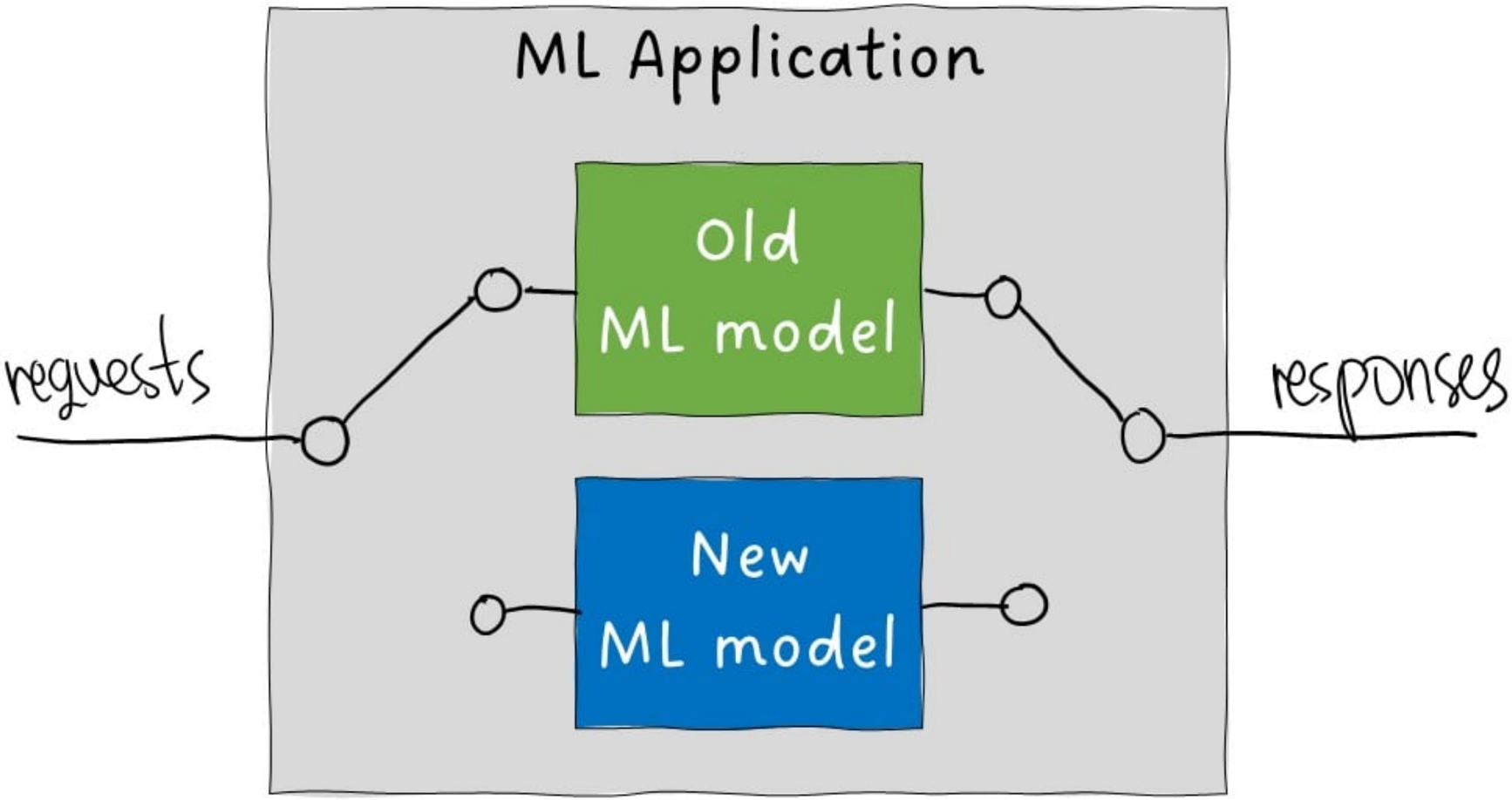

ML Application
New
ML model
Old
ML model

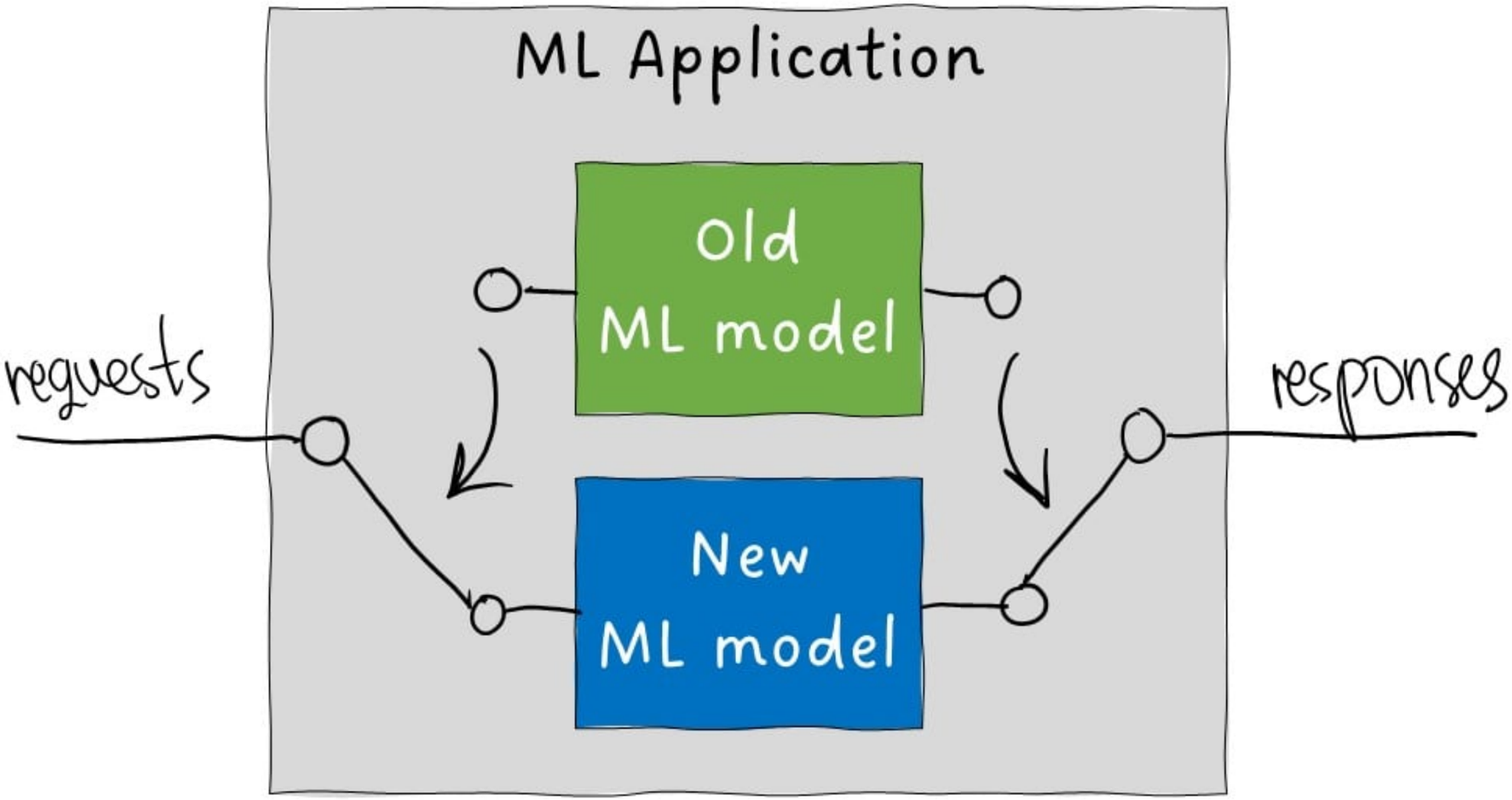

Model
run time
1h
1h
1h
1h
1h
Monday
17:00
Tuesday
17:00
Wednesday
17:00
Thursday
17:00
Friday
17:00
RUN
MODEL
RUN
MODEL
RUN
MODEL
RUN
MODEL
RUN
MODEL


~23h to deploy the new model!
Model run time
1h
1h
1h
1h
1h
Monday 17:00
Tuesday 17:00
Wednesday 17:00
Thursday 17:00
Friday 17:00
RUN MODEL
RUN MODEL
RUN MODEL
RUN MODEL
RUN MODEL

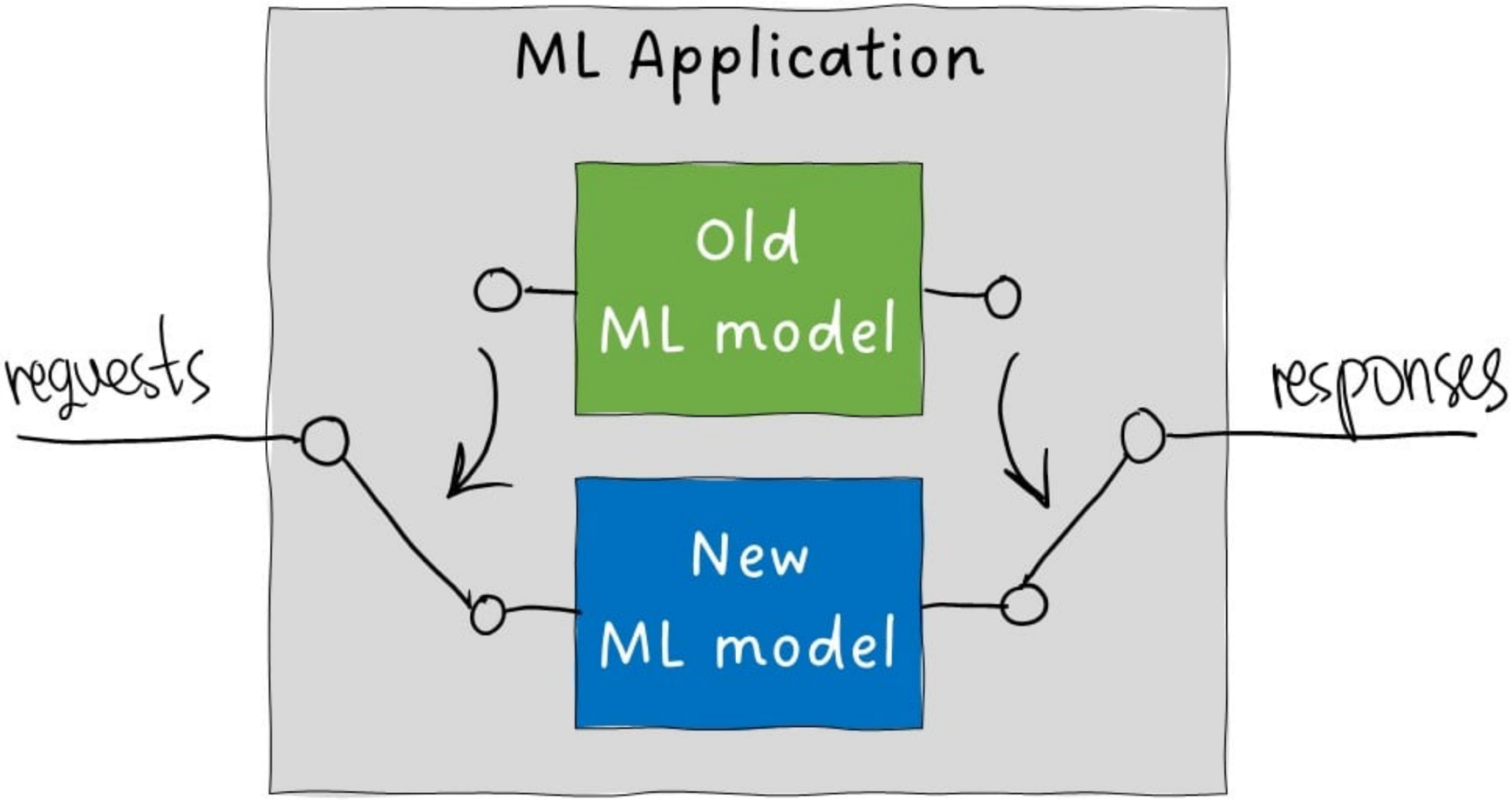

requests/minute
4000
3000
2000
1000
minutes


requests/minute
4000
3000
2000
1000
minutes
1 minute
~ 7000 requests = $$$?

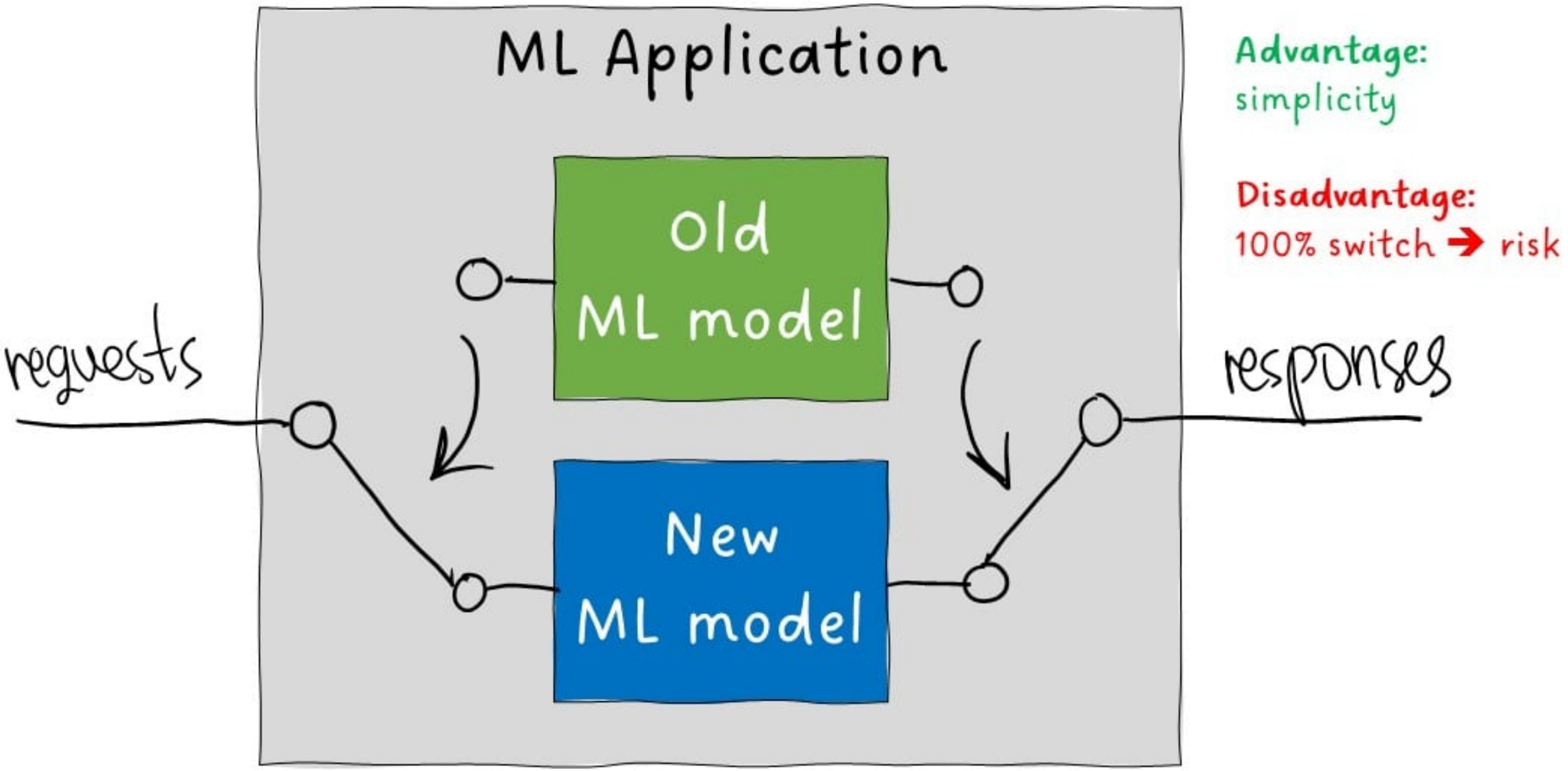

ML Application
Old
ML model
New
ML model
requests
responses


ML Application
Old
ML model
New
ML model
requests
responses

Blue/green deployment



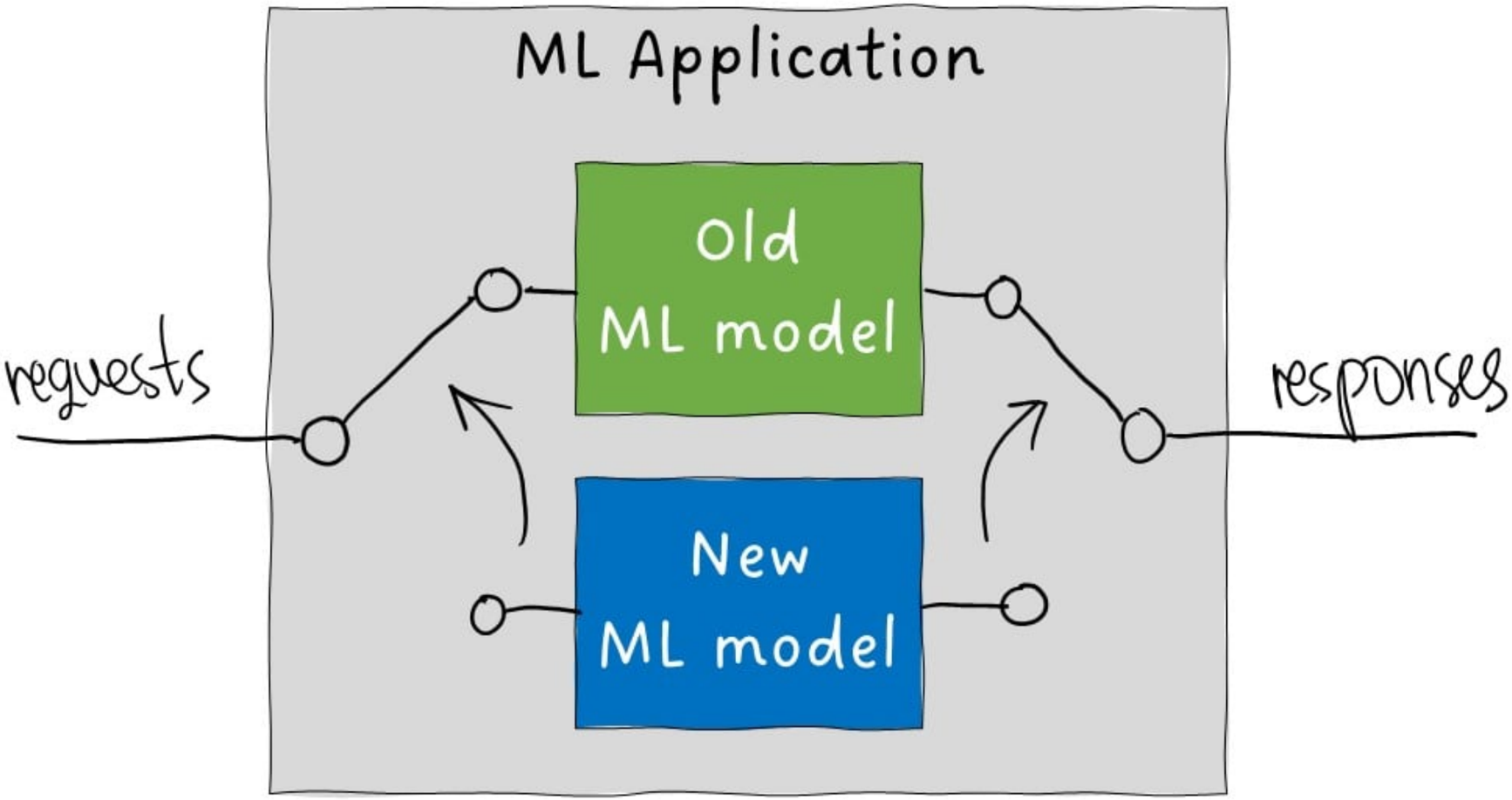
Blue/green deployment



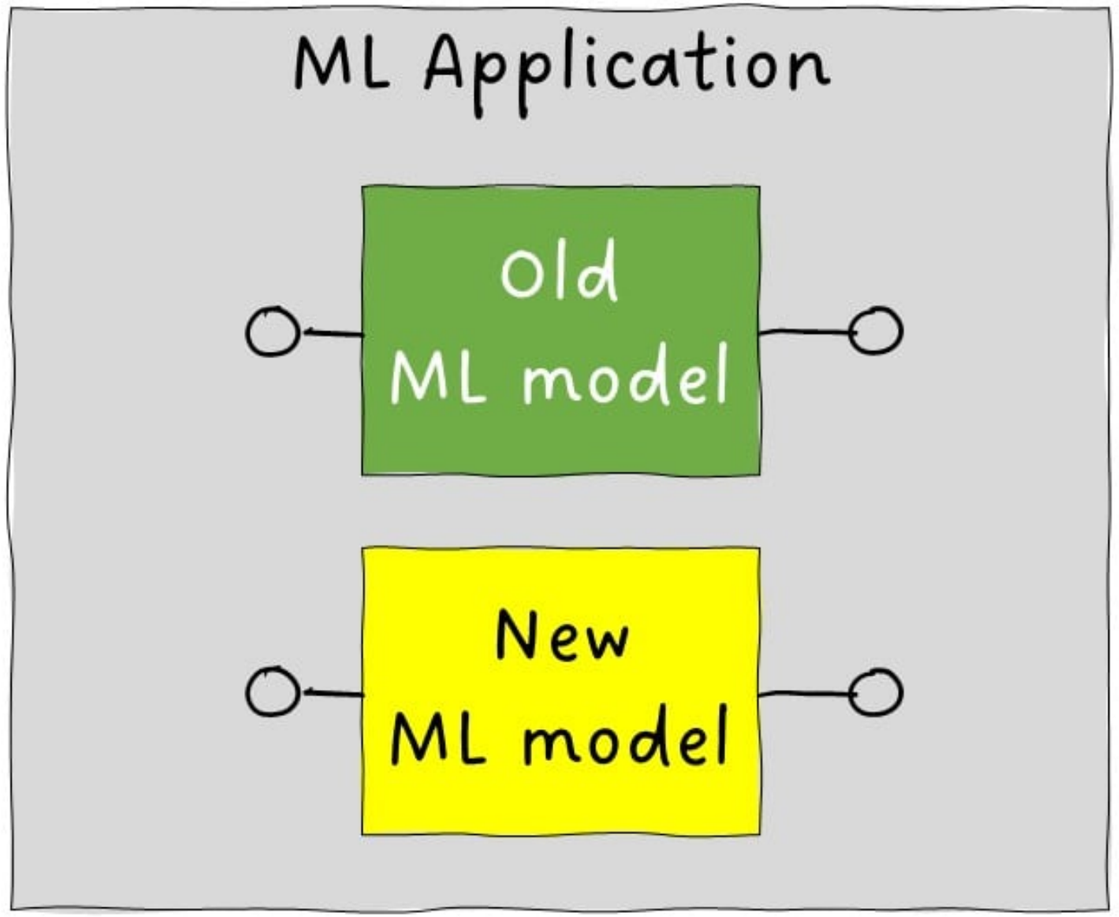
Advantage:
simplicity

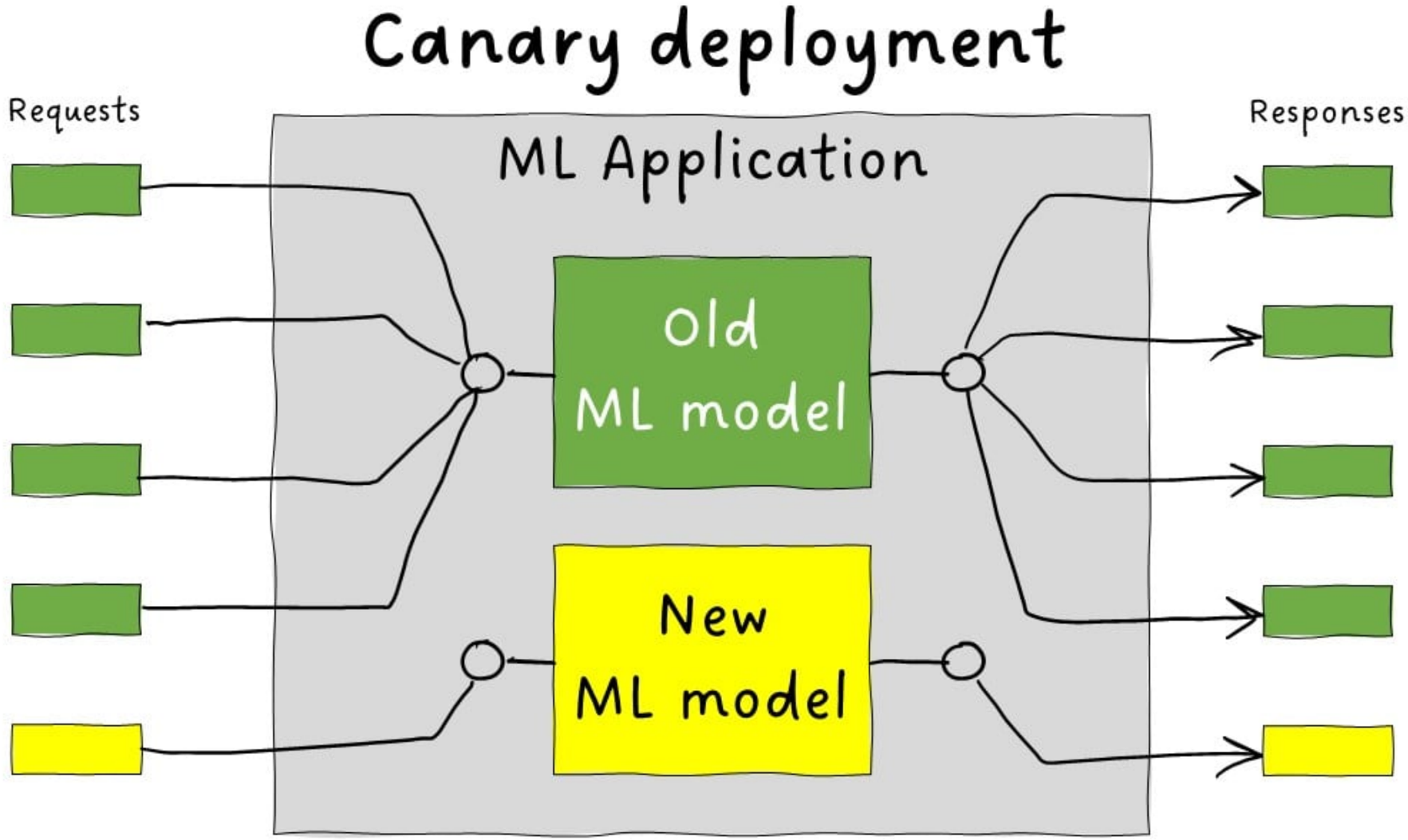
Disadvantage:
100% switch → risk

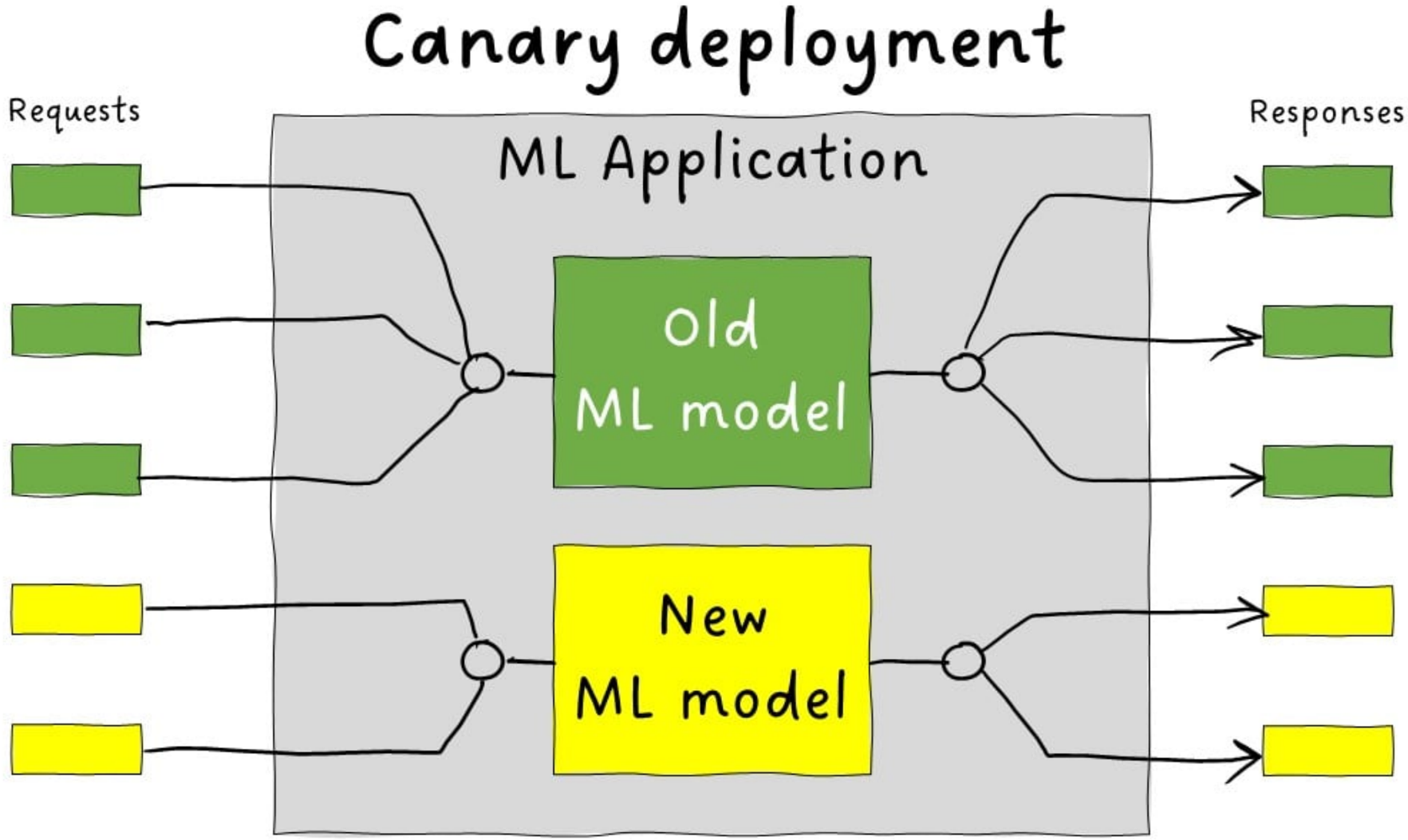
Rollback

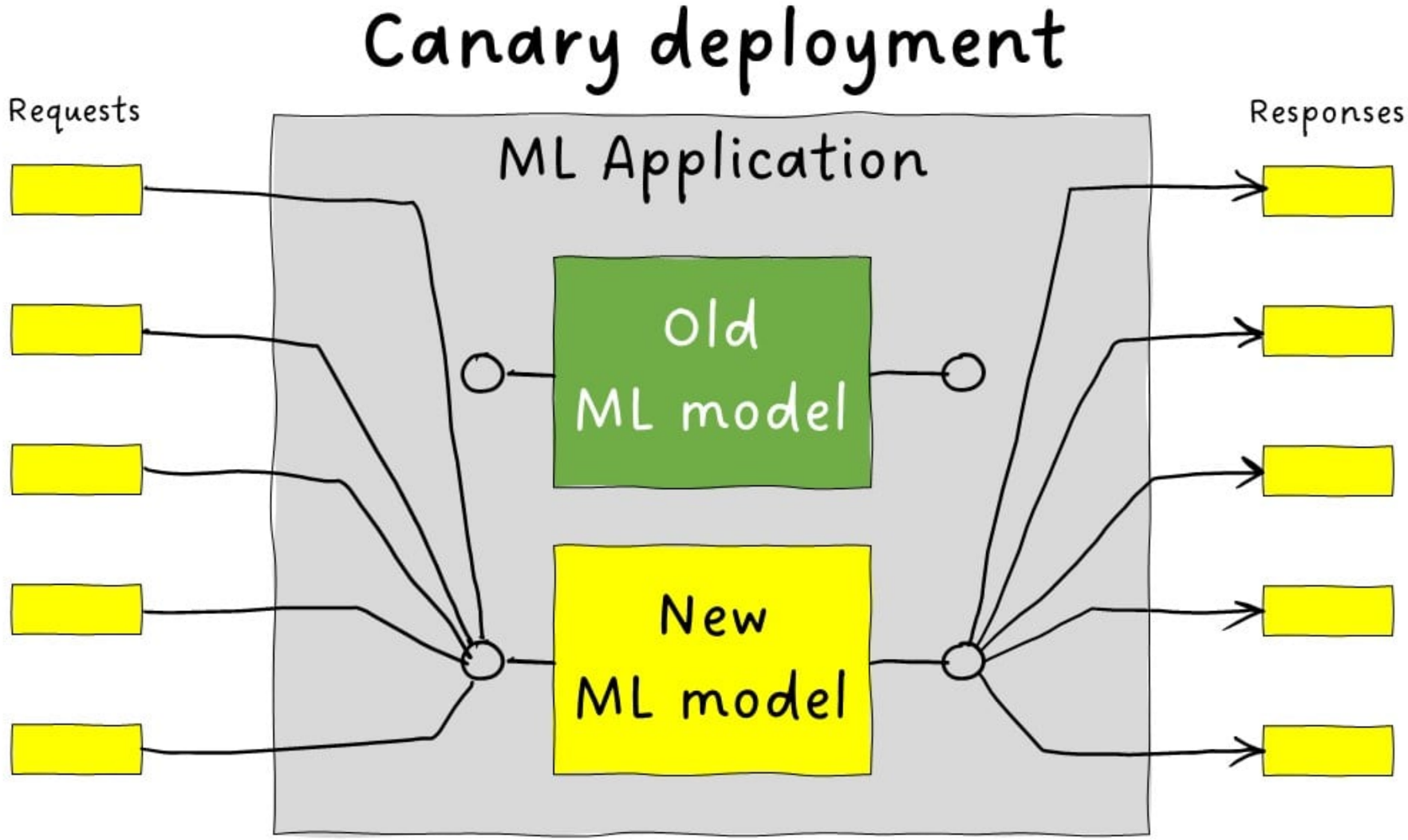

ML Application
Old
ML model
New
ML model
requests
responses

Canary deployment

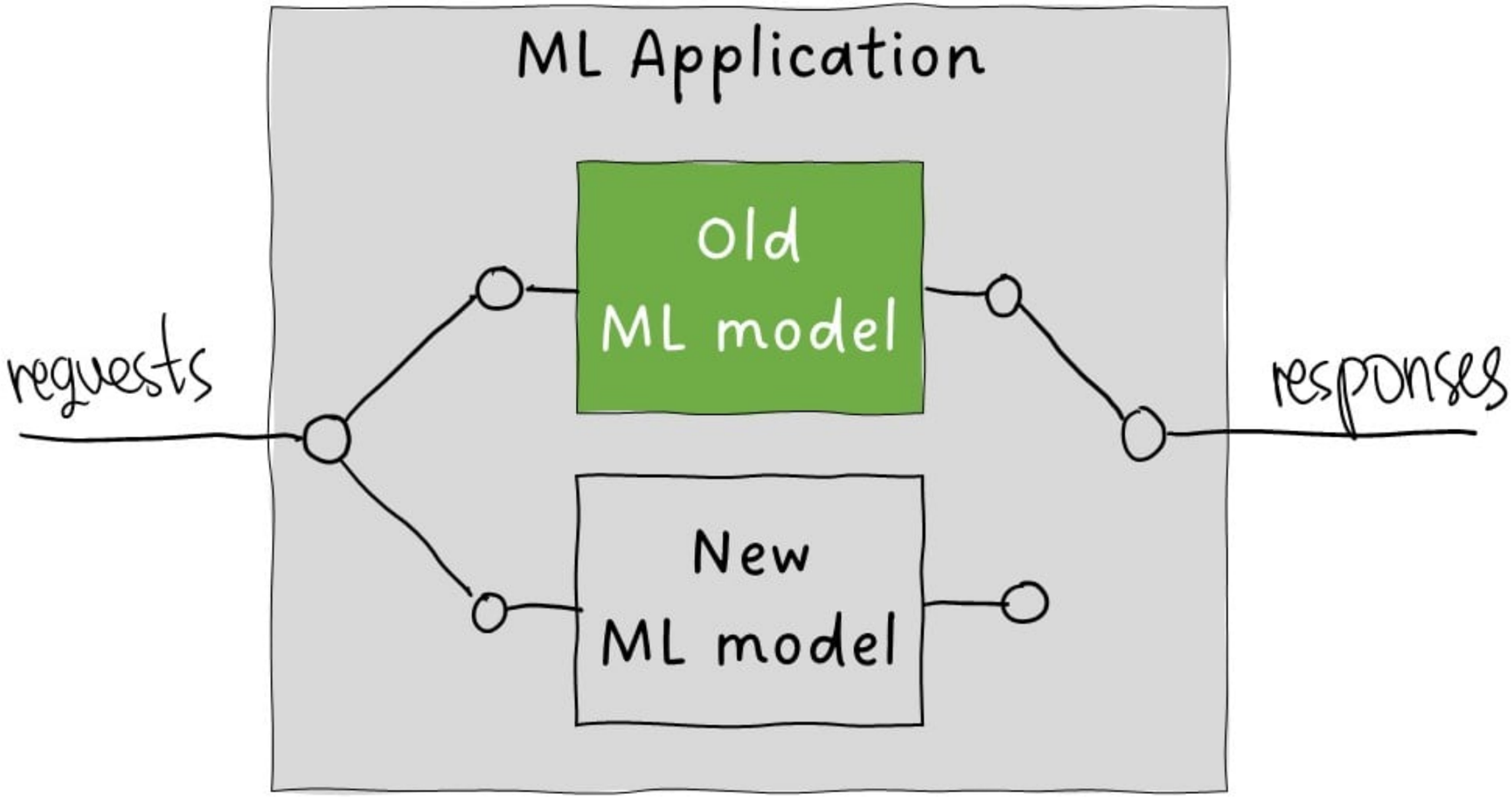



Canary deployment
Requests
ML Application
Old
ML model
New
ML model
Responses

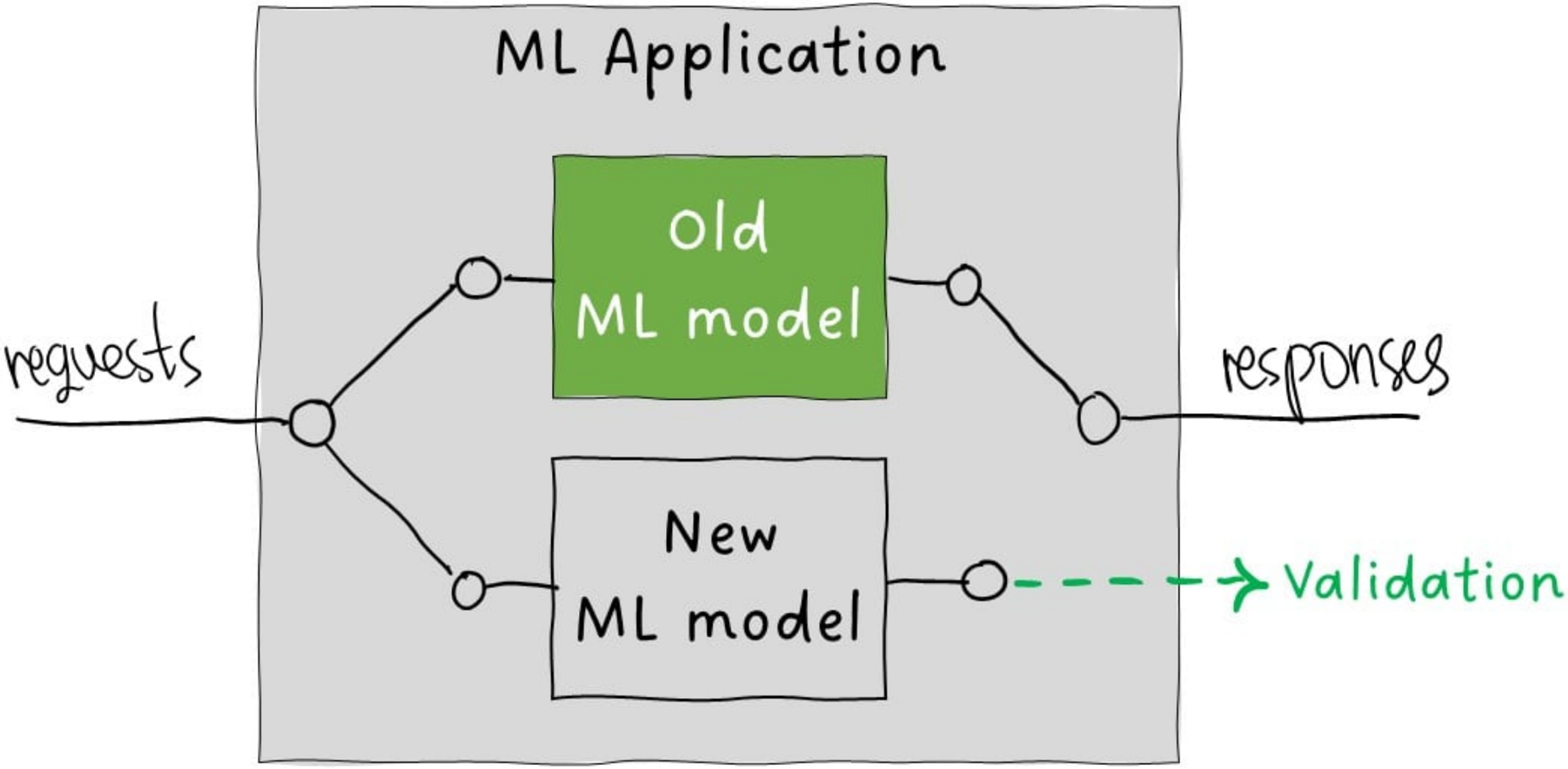

Canary deployment
Requests
ML Application
Responses
Old
ML model
New
ML model


Canary deployment
Requests
ML Application
Old
ML model
New
ML model
Responses

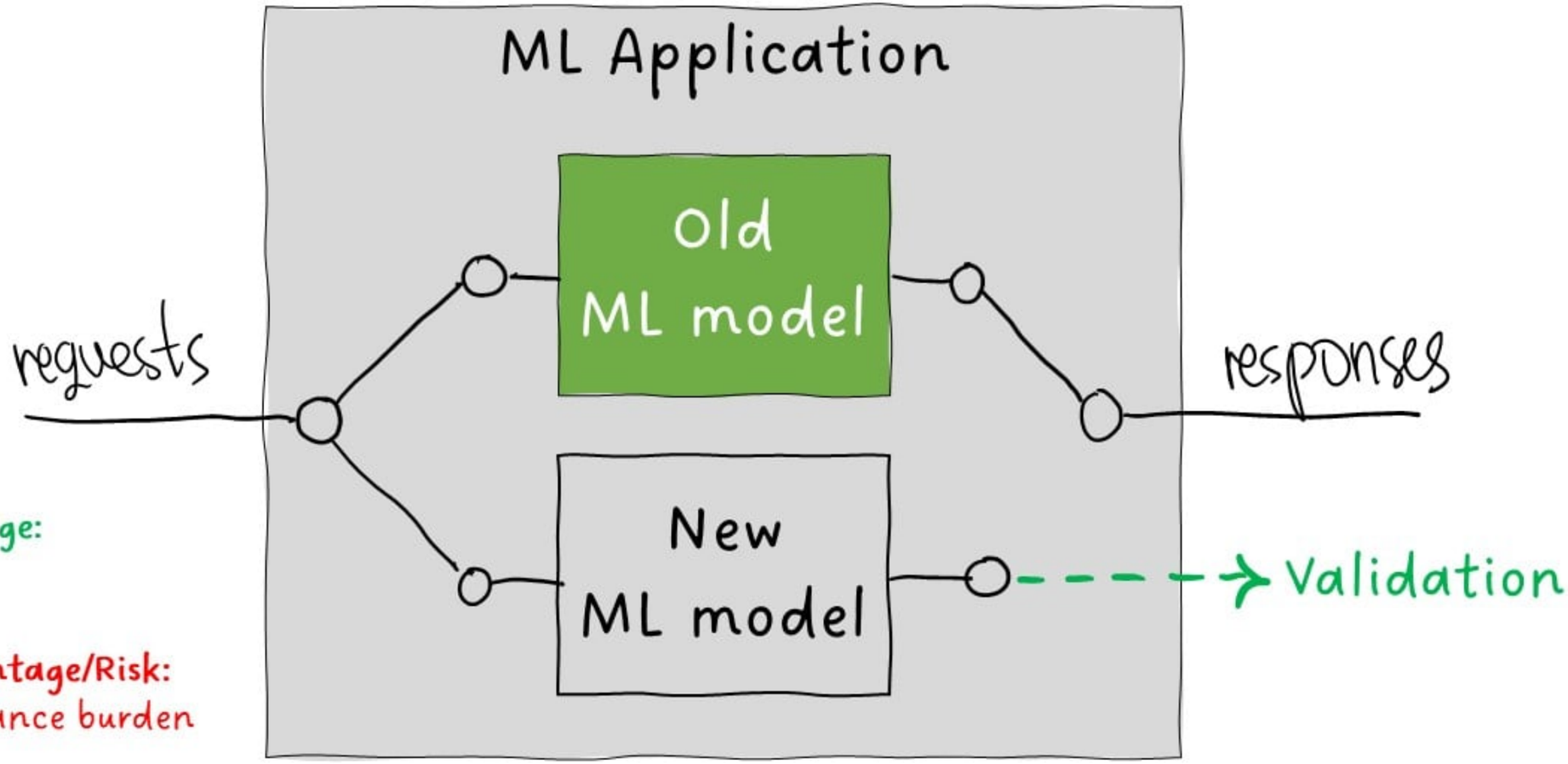
Shadow deployment



Shadow deployment



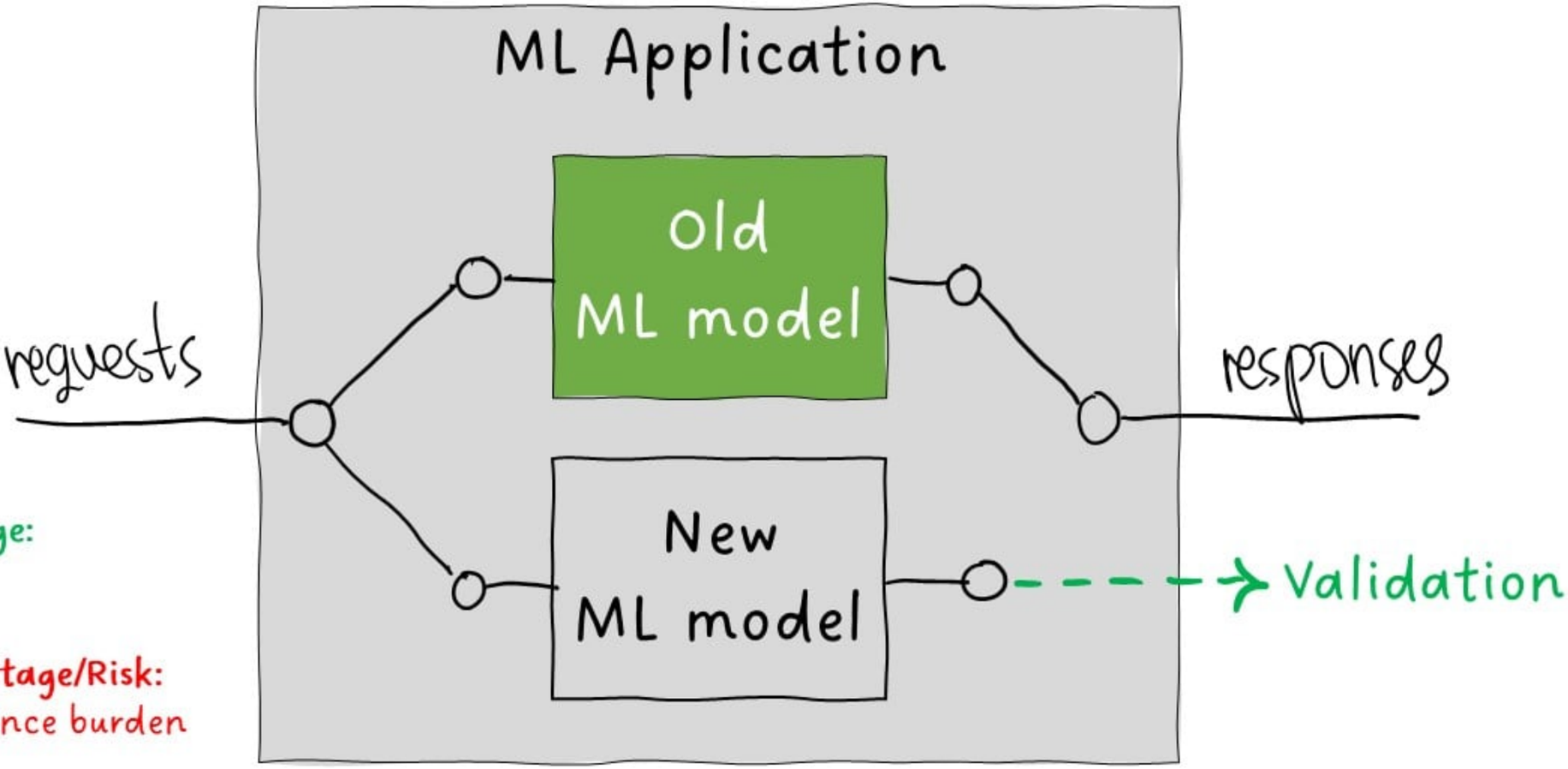
Shadow deployment



Advantage:
Safest

Disadvantage/Risk:
Performance burden

Shadow deployment



Advantage:
Safest

Disadvantage/Risk:
Performance burden

- Use shadow model on a fraction of requests ("canary shadow")
- Run shadow model offline, outside of peak hours.

Let's practice!

MLOPS DEPLOYMENT AND LIFE CYCLING