

GENERALITES

Michael
X  NATIS

L'architecture, c'est quoi ?



Compétence demandée :
Comprendre l'architecture des échanges +
architecture des applications

1. Architecture de l'infrastructure
2. Architecture du transport
3. Architecture de l'application

1. Architecture de l'infrastructure
2. Architecture du transport
3. Architecture de l'application

Architecture de l'insfratructure

<http://www.super-site.com>

X NATIS



4 super héros
sont là pour vous !
(très important)



Chrome



DNS



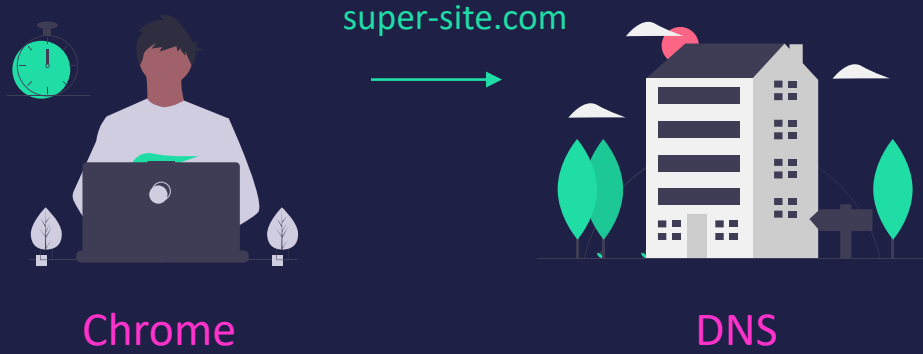
Apache



MySQL



Chrome





Chrome

Je cherche la page
pour 'vélo'

super-site.com



DNS



Chrome

Je cherche la page
pour 'vélo'

HTML + JavaScript + CSS

super-site.com



DNS



Chrome

Je cherche la page
pour 'vélo'

HTML + JavaScript + CSS

super-site.com



DNS

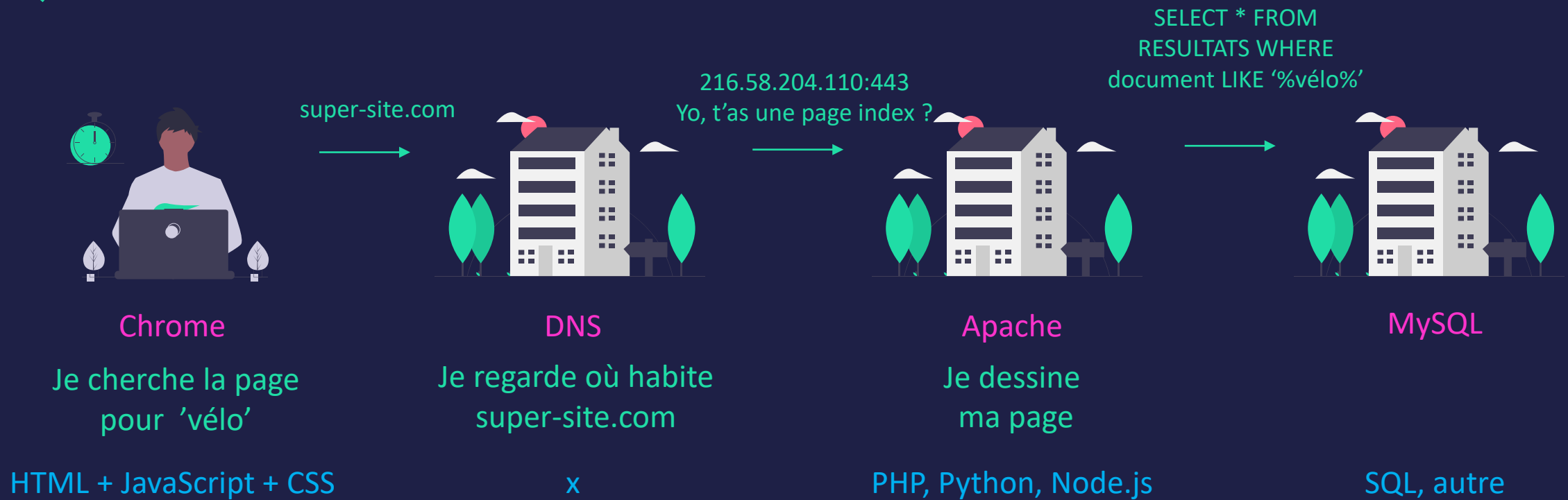
Je regarde où habite
super-site.com

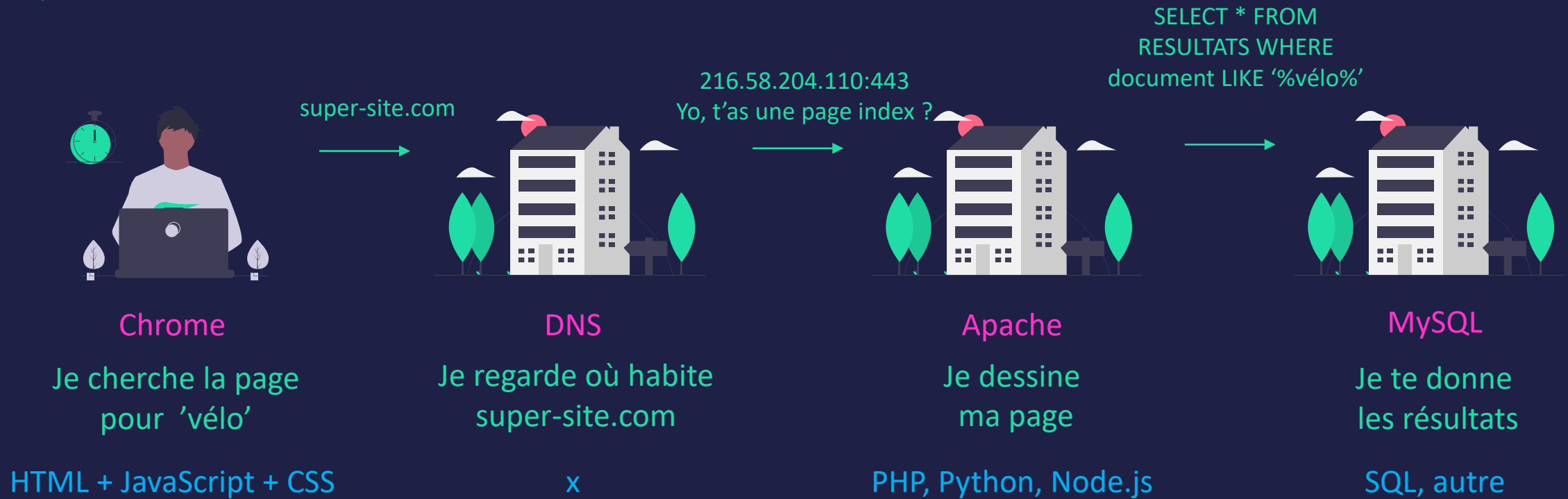
x

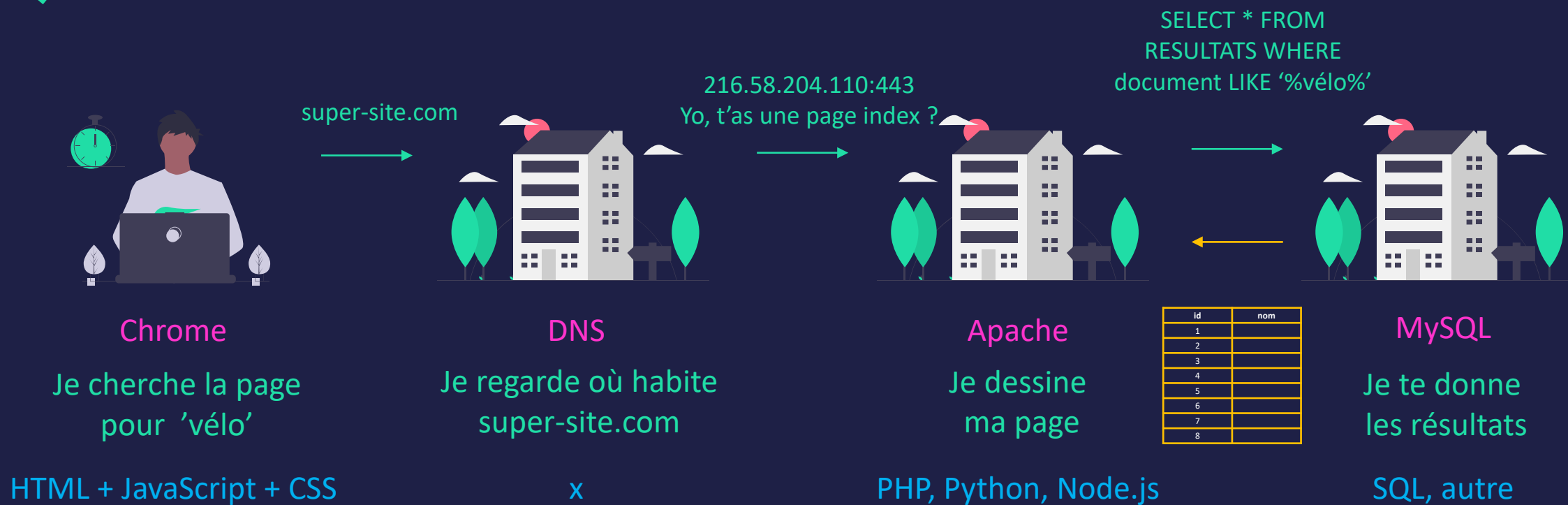


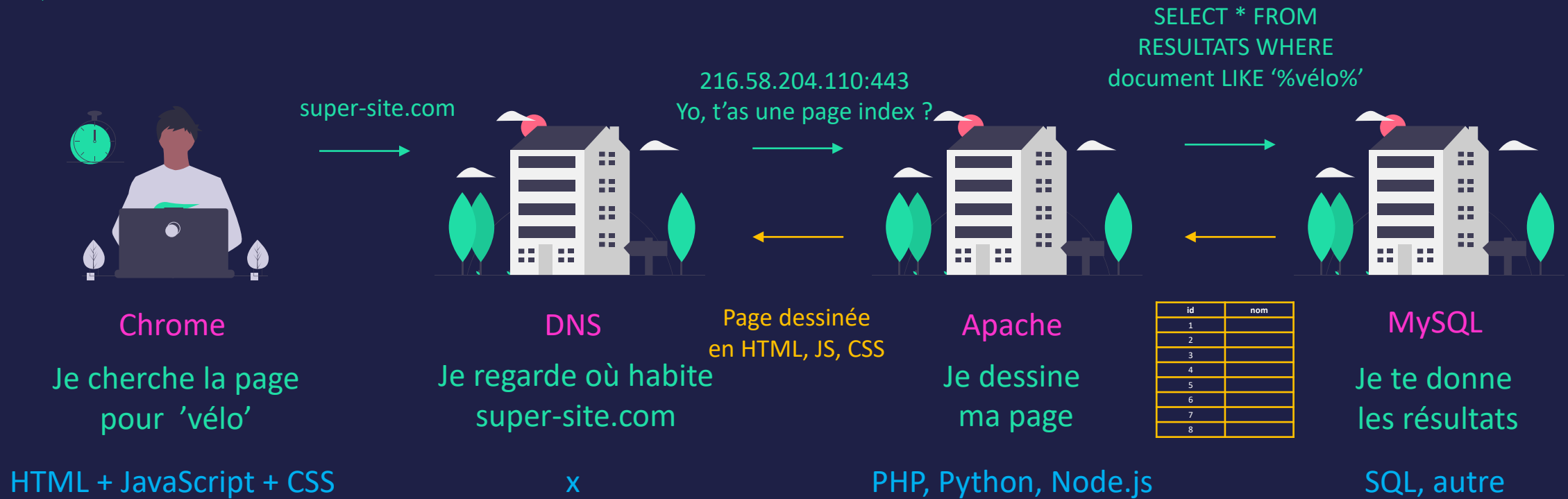






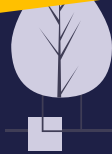


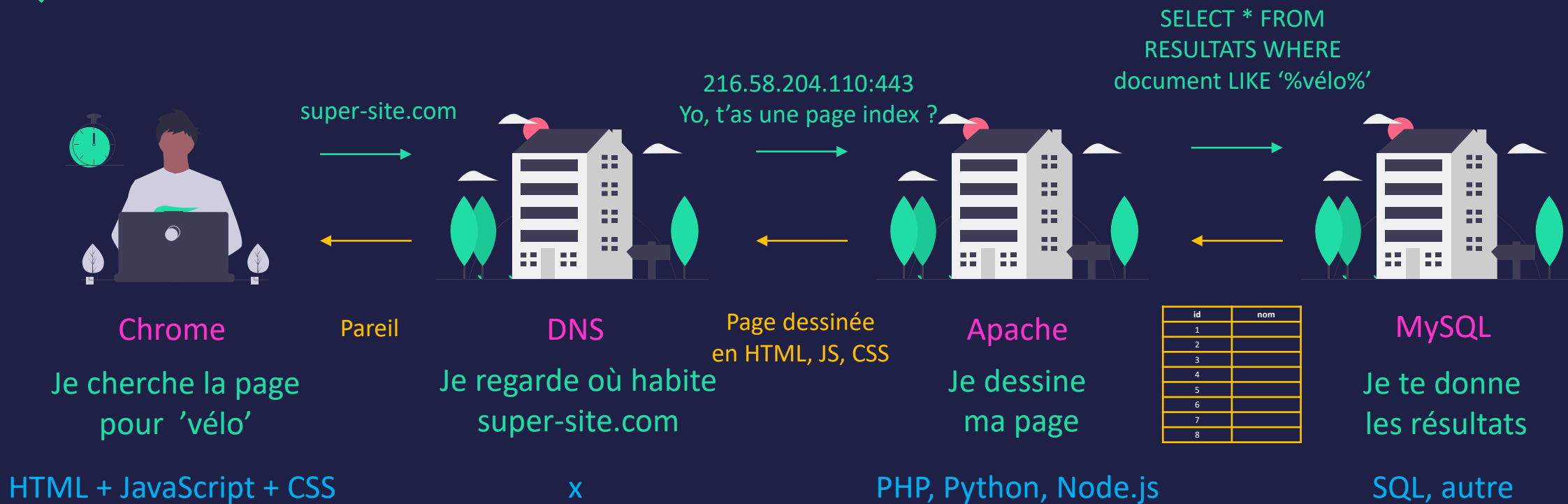






YES !









DNS



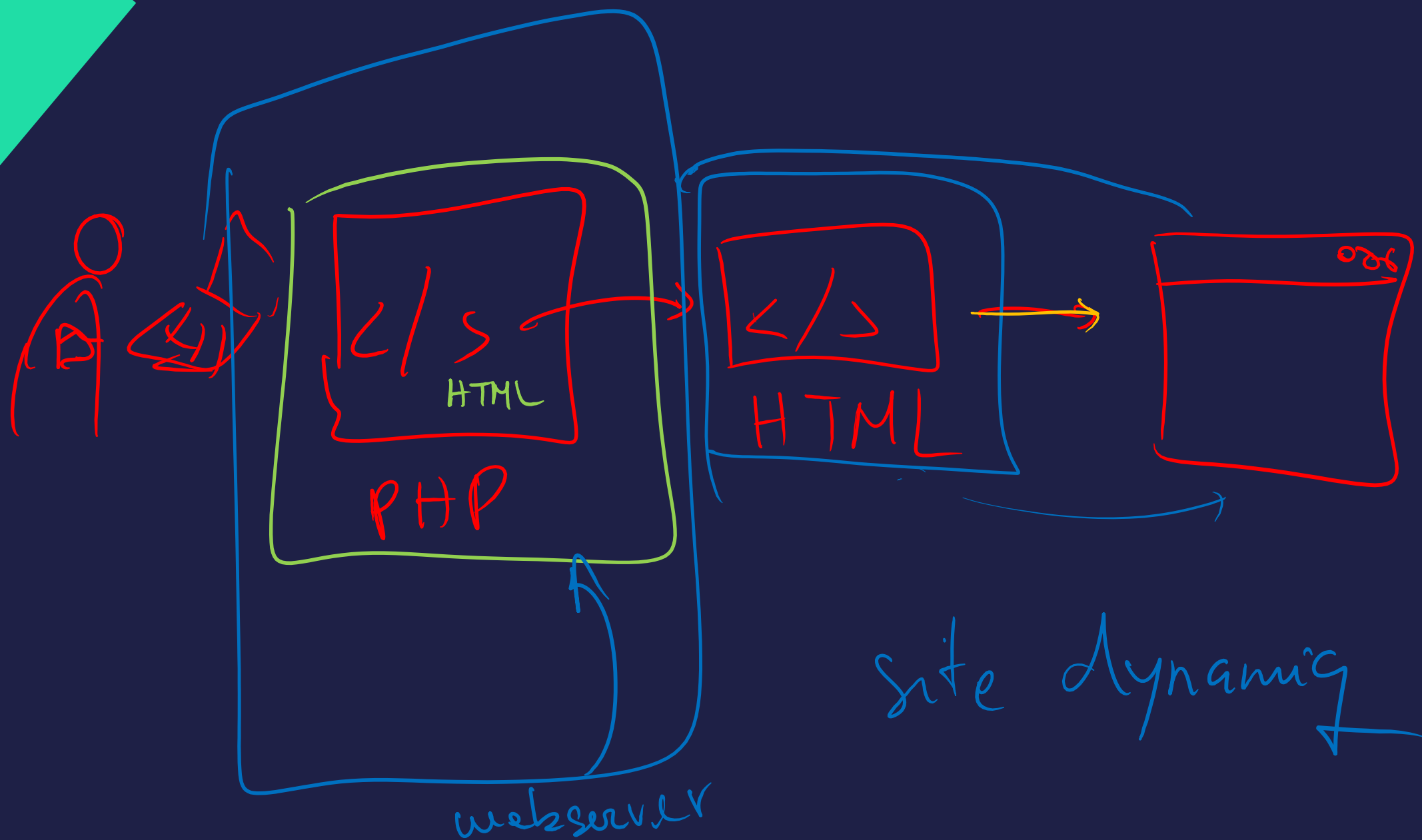
Chrome



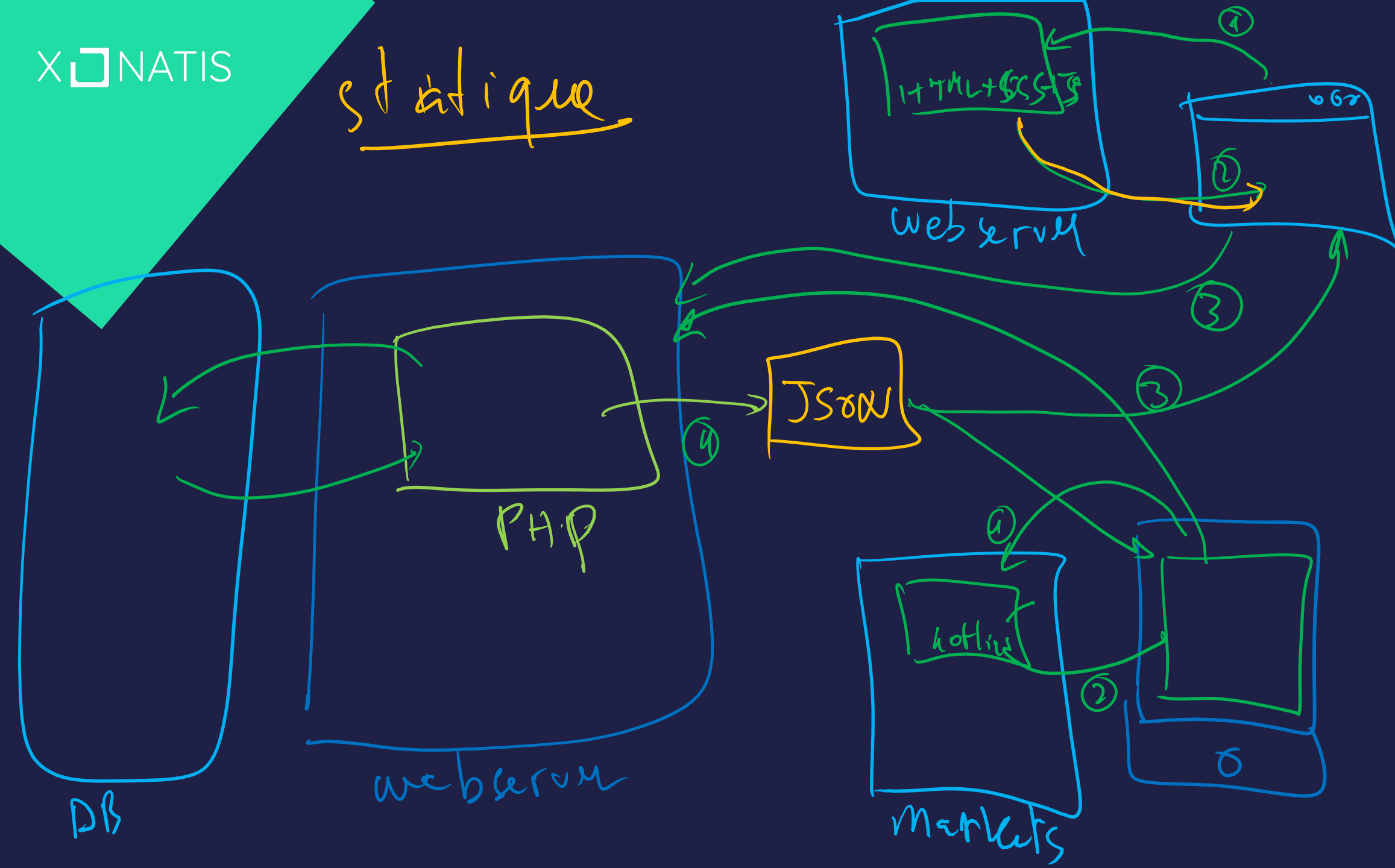
Apache



MySQL



stratégie

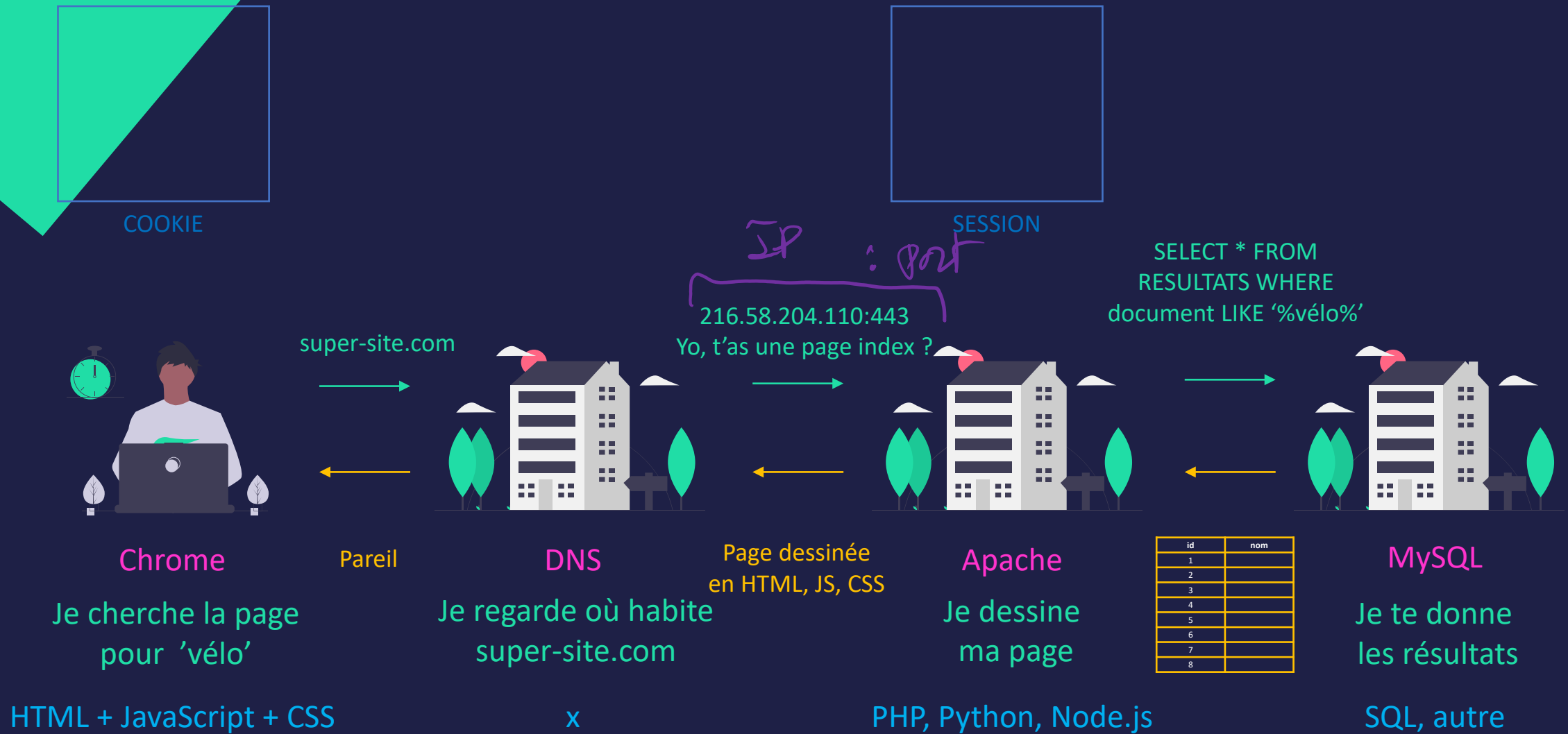


Une page statique : une page qui est renvoyée au client telle qu'elle a été stockée sur le serveur

Une page **dynamique**: une page qui est **générée à la volée par le serveur**

Une page statique : une page qui est renvoyée au client telle qu'elle a été stockée sur le serveur





Au niveau du stockage, tous les niveaux peuvent stocker de l'information. Cela dit, il est primordial de garder en tête que les cookies et la session sont temporaires

Au niveau du stockage, tous les niveaux peuvent stocker de l'information. Cela dit, il est primordial de garder en tête que les cookies et la session sont temporaires



1. Architecture de l'infrastructure
2. Architecture du transport
3. Architecture de l'application

Architecture du transport



DNS

Je regarde où habite
google



Chrome

Je cherche la page
de google pour 'vélo'

HTML + JavaScript + CSS

216.58.204.110:443
Yo, t'as une page index ?

Page dessinée
en HTML, JS, CSS



Apache

Je dessine
ma page

PHP, Python, Node.js

SELECT * FROM
RESULTATS WHERE
document LIKE '%vélo%'

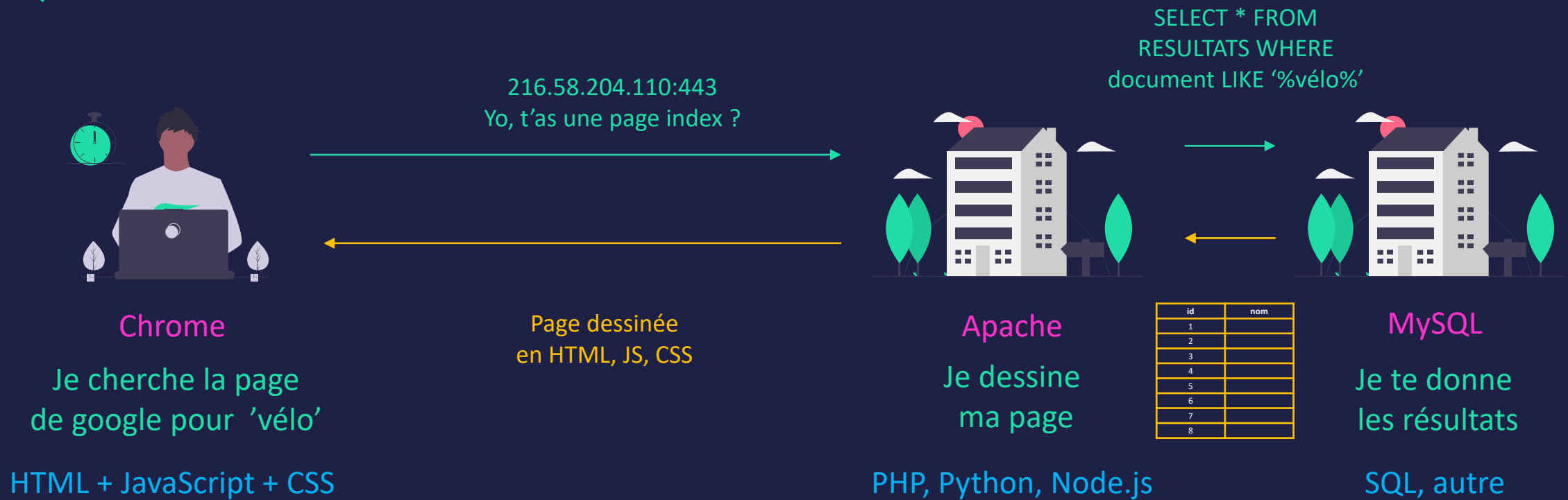


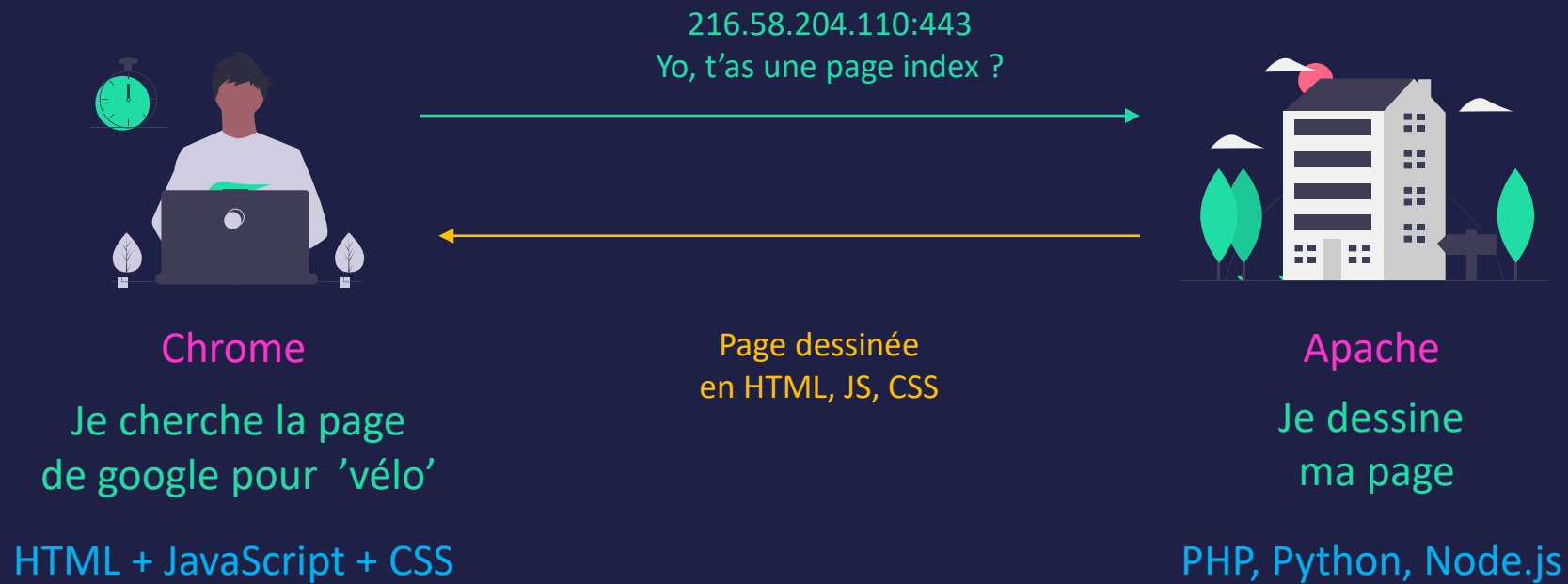
MySQL

Je te donne
les résultats

SQL, autre

| id | nom |
|----|-----|
| 1 | |
| 2 | |
| 3 | |
| 4 | |
| 5 | |
| 6 | |
| 7 | |
| 8 | |







Chrome

Je cherche la page
de google pour 'vélo'

HTML + JavaScript + CSS
EXECUTION DE LA REPONSE

testo.com

REPONSE HTML, CSS, JS + CODE

ex reu
2xx
3xx
4xx
5xx

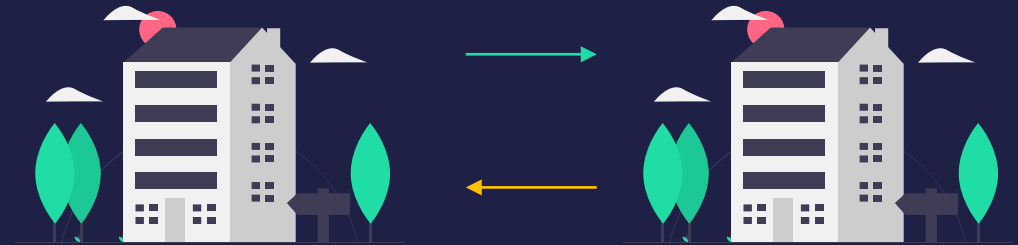


Apache

Je dessine
ma page

PHP, Python, Node.js
EXUCUTION DU PGM

SELECT * FROM
RESULTATS WHERE
document LIKE '%vélo%'



Apache

Je dessine
ma page

PHP, Python, Node.js

| id | nom |
|----|-----|
| 1 | |
| 2 | |
| 3 | |
| 4 | |
| 5 | |
| 6 | |
| 7 | |
| 8 | |

MySQL

Je te donne
les résultats

SQL, autre


```
SELECT * FROM  
RESULTATS WHERE  
document LIKE '%vélo%'
```



Apache

Je dessine
ma page

PHP, Python, Node.js



MySQL

Je te donne
les résultats

SQL, autre

| id | nom |
|----|-----|
| 1 | |
| 2 | |
| 3 | |
| 4 | |
| 5 | |
| 6 | |
| 7 | |
| 8 | |

2 logiciels peuvent-ils
communiquer entre eux ?



Un serveur

Un serveur

En informatique, un **serveur** est un **élément matériel** ou **logiciel informatique** (programme informatique) qui **fournit des fonctionnalités** à d'autres programmes ou appareils, appelés « **clients** ».

Un serveur

Cette architecture est appelée l'**architecture client – serveur**. Les serveurs peuvent fournir diverses fonctionnalités, souvent appelées « **services** », telles que le partage de données ou de ressources entre plusieurs clients, ou l'exécution de calculs **pour un client**.

Un serveur

Une machine ou un logiciel fournissant
des services à d'autres machines ou
logiciels, par exemple 24h/24

Un serveur

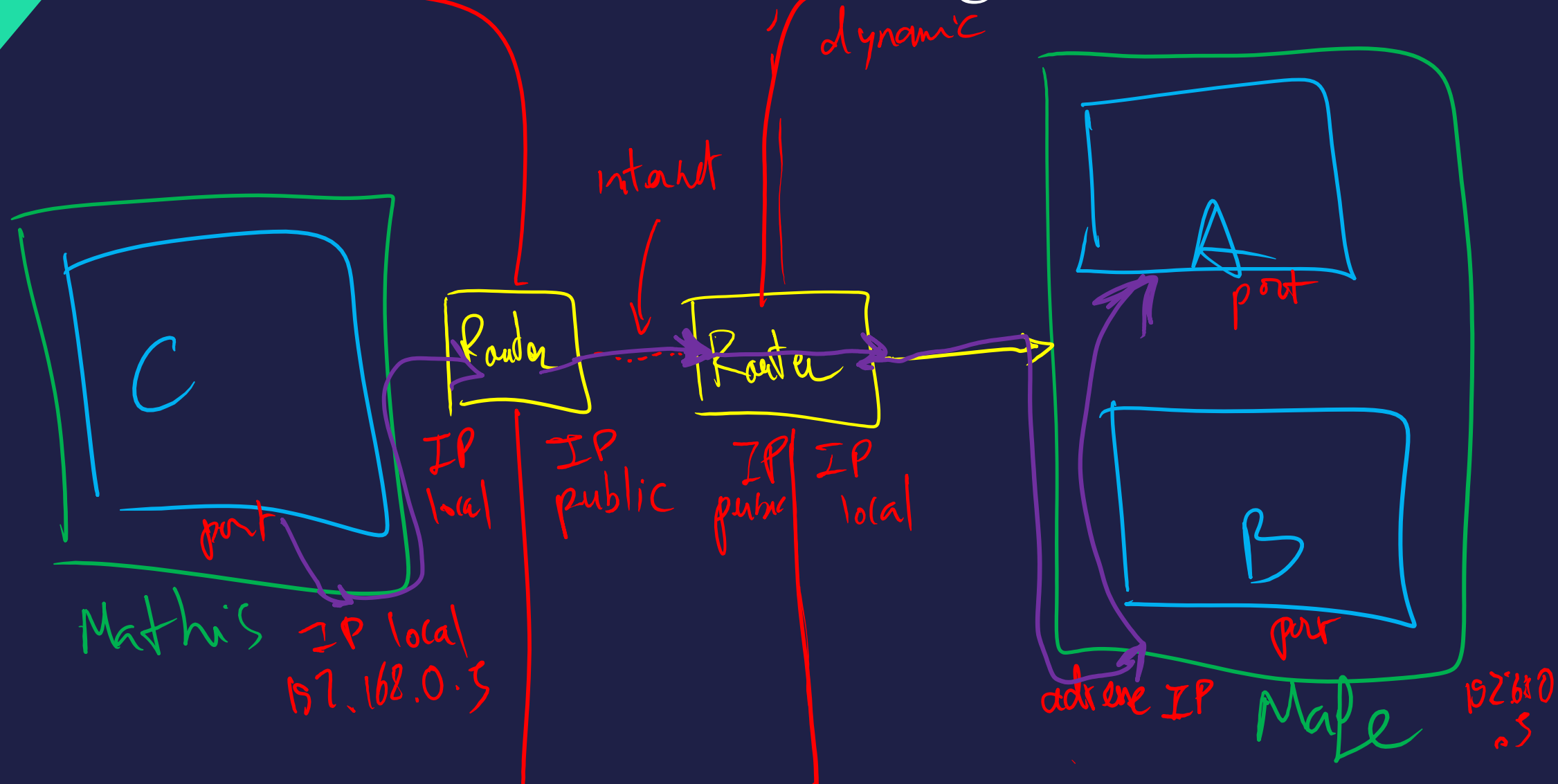
Une machine ou un logiciel fournissant
des services à d'autres machines ou
logiciels, par exemple 24h/24



Architecture client-serveur

1. Un client fait une requête à un serveur
2. Le serveur donne une réponse au client

Communication inter-logicielle



Une **adresse IP** (avec IP pour **Internet Protocol**) est un numéro d'identification qui est attribué de façon **permanente ou provisoire à chaque périphérique** relié à un réseau informatique qui utilise l'Internet Protocol.

L'adresse IP est à la base du système d'acheminement (le routage) des paquets de données sur Internet.

Dans la suite des protocoles Internet, la notion de **port** logiciel permet, sur un ordinateur donné, de **distinguer différents interlocuteurs logicielles**.

Ces interlocuteurs sont des programmes informatiques qui, selon les cas, écoutent ou émettent des informations sur ces ports.

Un port est distingué par son numéro.

Qu'est-ce qu'un réseau ?



Un réseau informatique est un **ensemble d'ordinateurs** partageant des ressources situées sur ou fournies par des nœuds de réseau. Les ordinateurs utilisent des **protocoles de communication** communs sur des interconnexions numériques pour communiquer entre eux.

Ces interconnexions sont constituées de technologies de réseau de télécommunication, basées sur des méthodes radioélectriques **physiquement câblées**, **optiques** et **sans fil** qui peuvent être agencées dans une variété de topologies de réseau.

Le DHCP est un protocole réseau chargé de la configuration automatique des adresses IP d'un réseau informatique.

Il évite ainsi à l'utilisateur qui se connecte pour la première fois à un réseau, d'avoir à configurer l'adresse IP de son équipement.

Dans le domaine des télécommunications et des communications informatiques, le terme **Interface réseau** peut signifier :

- La partie qui assure la connexion entre un terminal utilisateur et un réseau public ou privé
- La **carte réseau** d'un ordinateur
- Le point de connexion entre deux réseaux

Institute of Electrical and Electronics Engineers (IEEE)
est une organisation basée à New York qui est
connue dans l'édition des standards réseaux (local,
personnel ou même métropolitain)

Qu'est-ce qu'un paquet ?



Dans les télécommunications et les réseaux informatiques, un **paquet réseau** est une unité **formatée de données transportée** par un réseau.

Un paquet se compose d'informations de **contrôle** et de **charge utile**.

Les informations de contrôle fournissent des données pour fournir la charge utile (par exemple, les adresses de réseau source et de destination, les codes de détection d'erreur ou les informations de séquençage).

En règle générale, les informations de contrôle se trouvent dans les en-têtes et les queues de paquet.

odd-http.pcap

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter ... <Ctrl-/> Expression...

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|----------|---------------|---------------|----------|--------|---|
| 4 | 0.025749 | 172.16.0.122 | 200.121.1.131 | TCP | 54 | [TCP Window Update] [TCP ACKed unseen segment] 80 → 10554 [ACK] Seq=... |
| 5 | 0.076967 | 200.121.1.131 | 172.16.0.122 | TCP | 1454 | [TCP Previous segment not captured] [TCP Spurious Retransmission] 10... |
| 6 | 0.076978 | 172.16.0.122 | 200.121.1.131 | TCP | 54 | [TCP Dup ACK 2#1] [TCP ACKed unseen segment] 80 → 10554 [ACK] Seq=1 ... |
| 7 | 0.102939 | 200.121.1.131 | 172.16.0.122 | TCP | 1454 | [TCP Spurious Retransmission] 10554 → 80 [ACK] Seq=5601 Ack=1 Win=65... |
| 8 | 0.102946 | 172.16.0.122 | 200.121.1.131 | TCP | 54 | [TCP Dup ACK 2#2] [TCP ACKed unseen segment] 80 → 10554 [ACK] Seq=1 ... |
| 9 | 0.128285 | 200.121.1.131 | 172.16.0.122 | TCP | 1454 | [TCP Spurious Retransmission] 10554 → 80 [ACK] Seq=7001 Ack=1 Win=65... |
| 10 | 0.128319 | 172.16.0.122 | 200.121.1.131 | TCP | 54 | [TCP Dup ACK 2#3] [TCP ACKed unseen segment] 80 → 10554 [ACK] Seq=1 ... |
| 11 | 0.154162 | 200.121.1.131 | 172.16.0.122 | TCP | 1454 | [TCP Spurious Retransmission] 10554 → 80 [ACK] Seq=8401 Ack=1 Win=65... |
| 12 | 0.154169 | 172.16.0.122 | 200.121.1.131 | TCP | 54 | [TCP Dup ACK 2#4] [TCP ACKed unseen segment] 80 → 10554 [ACK] Seq=1 ... |
| 13 | 0.179906 | 200.121.1.131 | 172.16.0.122 | TCP | 1454 | [TCP Spurious Retransmission] 10554 → 80 [ACK] Seq=9801 Ack=1 Win=65... |
| 14 | 0.179915 | 172.16.0.122 | 200.121.1.131 | TCP | 54 | [TCP Dup ACK 2#5] 80 → 10554 [ACK] Seq=1 Ack=11201 Win=63000 Len=0 |
| 15 | 0.207145 | 200.121.1.131 | 172.16.0.122 | TCP | 1454 | 10554 → 80 [ACK] Seq=11201 Ack=1 Win=65535 Len=1400 [TCP segment of ... |
| 16 | 0.207156 | 172.16.0.122 | 200.121.1.131 | TCP | 54 | 80 → 10554 [ACK] Seq=1 Ack=12601 Win=63000 Len=0 |
| 17 | 0.232621 | 200.121.1.131 | 172.16.0.122 | TCP | 1454 | 10554 → 80 [ACK] Seq=12601 Ack=1 Win=65535 Len=1400 [TCP segment of ... |
| 18 | 0.232629 | 172.16.0.122 | 200.121.1.131 | TCP | 54 | 80 → 10554 [ACK] Seq=1 Ack=14001 Win=63000 Len=0 |
| 19 | 0.258365 | 200.121.1.131 | 172.16.0.122 | TCP | 1454 | 10554 → 80 [ACK] Seq=14001 Ack=1 Win=65535 Len=1400 [TCP segment of ... |
| 20 | 0.258373 | 172.16.0.122 | 200.121.1.131 | TCP | 54 | 80 → 10554 [ACK] Seq=1 Ack=15401 Win=63000 Len=0 |

> Frame 15: 1454 bytes on wire (11632 bits), 1454 bytes captured (11632 bits)

> Ethernet II, Src: Vmware_c0:00:01 (00:50:56:c0:00:01), Dst: Vmware_42:12:13 (00:0c:29:42:12:13)

> Internet Protocol Version 4, Src: 200.121.1.131, Dst: 172.16.0.122

▼ Transmission Control Protocol, Src Port: 10554, Dst Port: 80, Seq: 11201, Ack: 1, Len: 1400

Source Port: 10554

Destination Port: 80

[Stream index: 0]

[TCP Segment Len: 1400]

Sequence number: 11201 (relative sequence number)

[Next sequence number: 12601 (relative sequence number)]

Acknowledgment number: 1 (relative ack number)

0101 = Header Length: 20 bytes (5)

0020 00 7a 29 3a 00 50 a7 5c 30 08 e2 e2 ee bf 50 10 .z.:P.\0....P.

0030 ff ff bc 5e 00 00 42 4f 78 42 56 35 6a 45 52 52 ...^..BOxBV5jERR

0040 71 5a 69 63 39 34 54 77 48 4c 71 46 51 34 78 35 qZic94TwHLqFQ4x5

0050 61 62 46 30 77 55 6e 59 73 46 2b 67 6c 44 47 4c abF0wUnYsF+g1DGL

0060 33 56 75 35 65 61 33 4d 44 59 77 49 70 63 32 44 3Vu5ea3MDYwIpc2D

0070 78 4c 44 4d 74 38 6b 2f 75 42 68 38 6a 48 6d 30 xLDMt8k/ uBh8jHm0

0080 63 66 54 63 69 35 6a 77 77 4c 2f 56 4c 6f 6c 41 cFTci5jw wL/VLo1A

0090 57 4c 6c 35 63 43 79 4e 6d 63 36 52 70 58 57 7a WL15cCyN mc6RpXWz

Acknowledgment number (tcp.ack), 4 bytes

Packets: 3083 · Displayed: 3083 (100.0%) Profile: Default

Qu'est-ce qu'un routeur ?



Un **routeur** est un **périphérique réseau** qui transfère des paquets de données **entre des réseaux** informatiques. Les routeurs exécutent les fonctions de direction du trafic sur Internet.

Les données envoyées via Internet, telles qu'une page Web ou un courrier électronique, se présentent sous la forme de paquets de données.

Un paquet est généralement transféré d'un routeur à un autre via les réseaux qui constituent un interréseau (par exemple Internet) jusqu'à ce qu'il atteigne son nœud de destination.

Dans les réseaux informatiques, une **table de routage**, ou RIB, est une table de données stockée dans un routeur ou un hôte réseau qui **répertorie les routes vers des destinations réseau particulières** et, dans certains cas, les **métriques (distances)** associées à ces routes.

La table de routage contient des informations sur la topologie du réseau qui l'entoure immédiatement.

On s'en rappelle ?

| | |
|------------------|----------------------------|
| IP | Réseau |
| Adresse IP | Web server |
| DHCP | Serveur de base de données |
| Port | DNS |
| Serveur | Client |
| Interface | Paquet |
| IEEE | Routeur |
| Table de routage | Correction d'erreur |



Qu'est-ce que WEB ?

Une manière de partager de
l'information sur internet

Une manière de partager de
l'information sur internet



1. Architecture de l'infrastructure
2. Architecture du transport
3. Architecture de l'application

Architecture applicative

En génie logiciel, l'architecture en couche (souvent appelée architecture n-tier) ou architecture multicouche est une architecture client-serveur dans laquelle les fonctions de présentation, de traitement des applications et de gestion des données sont physiquement séparées.

L'utilisation la plus répandue de l'architecture multiniveau est l'architecture à trois niveaux.

Couper son code en 3 parties
M, V et C

MVC :
Model
View
Controller

MVC :
Model = Entités

MVC :

Model = Entités

View = Code pour le visuel

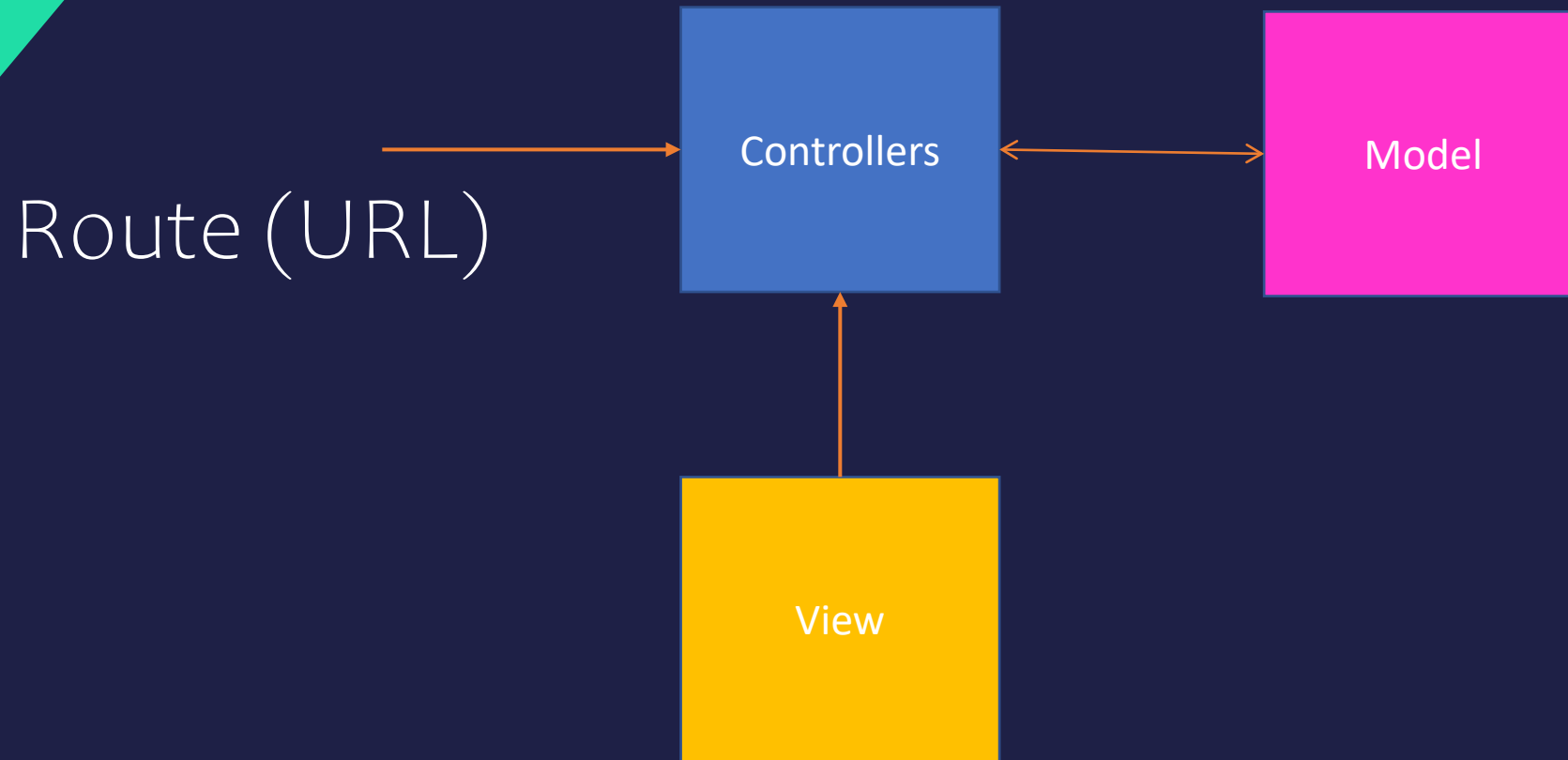
MVC :

Model = Entités

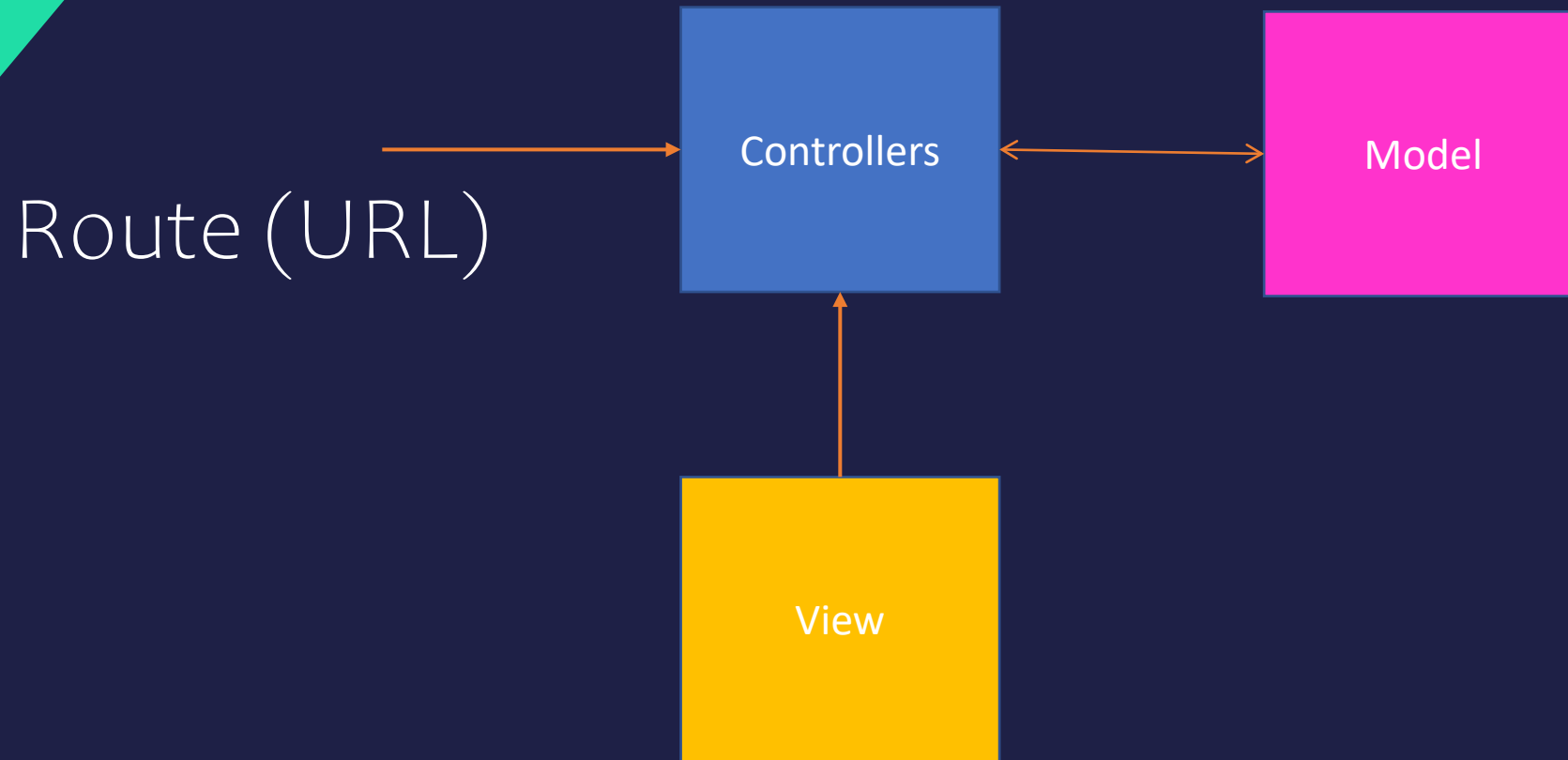
View = Code pour le visuel

Controller = Traitement

Architecture MVC



Architecture MVC



On s'en rappelle ?

| | |
|------------------------|--------------|
| Architecture en couche | MVC |
| Controller | Modèle |
| Vue | Présentation |

