Client-Server Communication

Agenda

Client Server Model
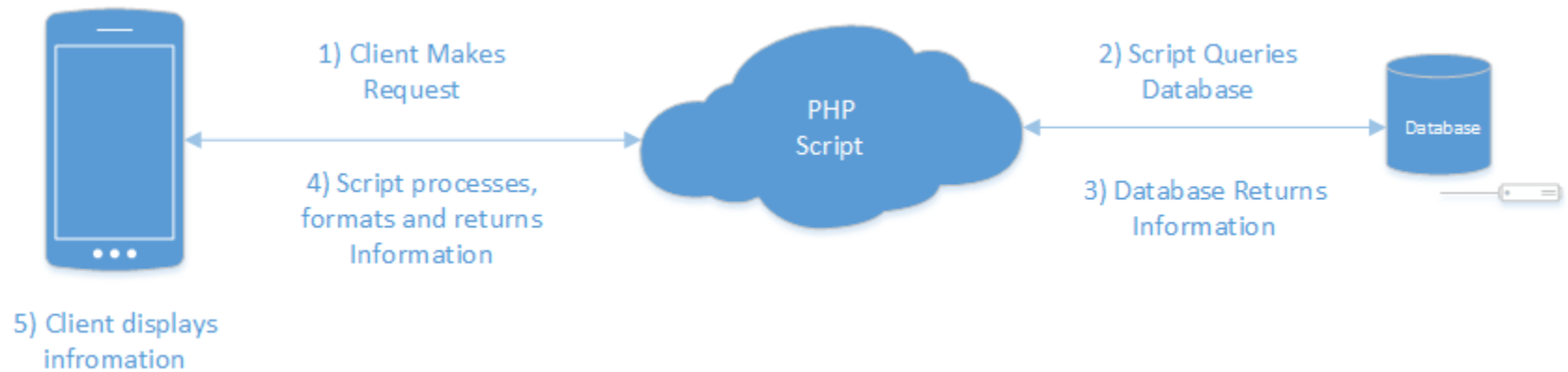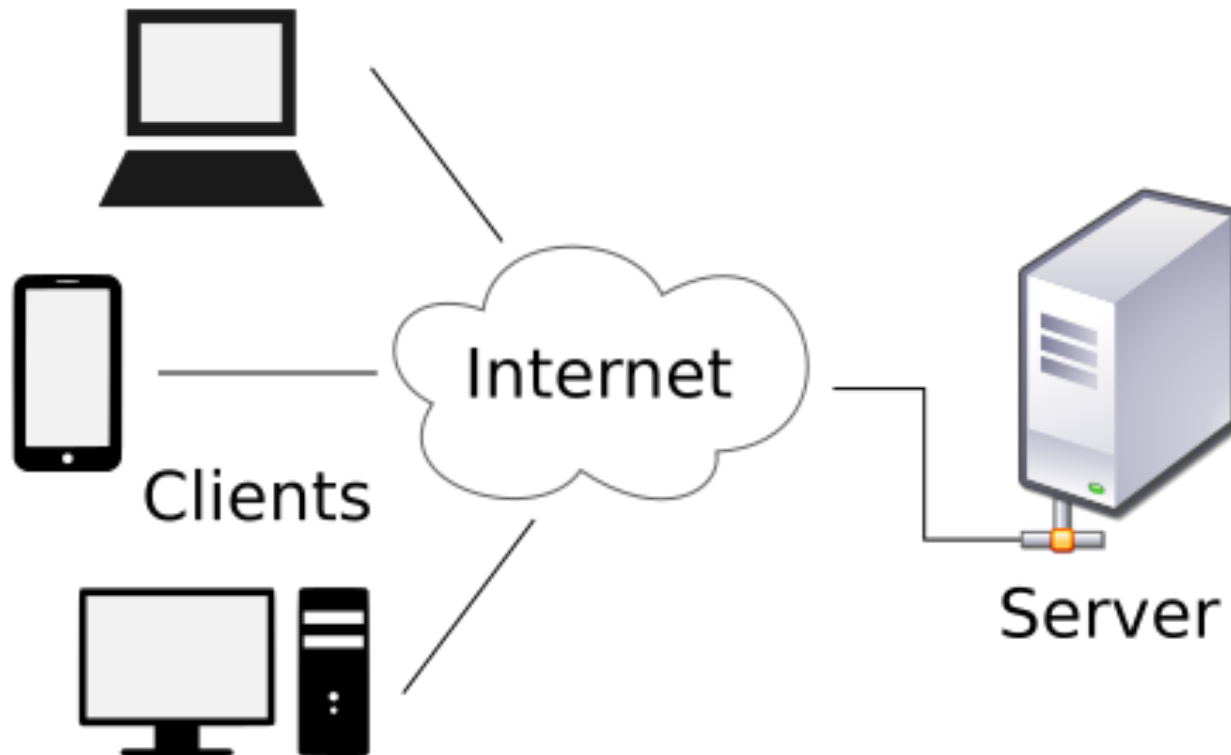Types of Access
AJAX in depth
JSON
REST

JS

Client Server Model

The **client–server model** is a distributed application structure that partitions tasks or workloads between the providers of a resource or service, called servers, and service requesters, called clients.
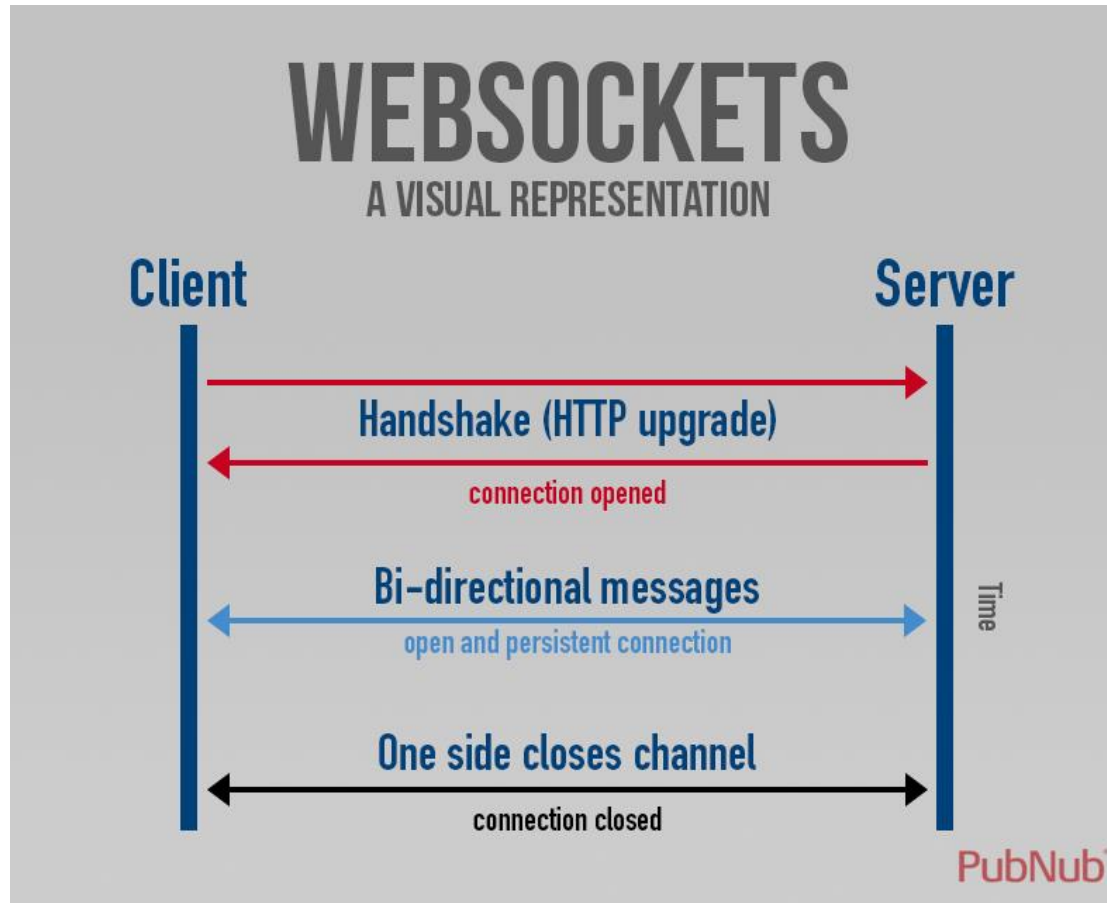
JS

Client Server Model

Client Server Model
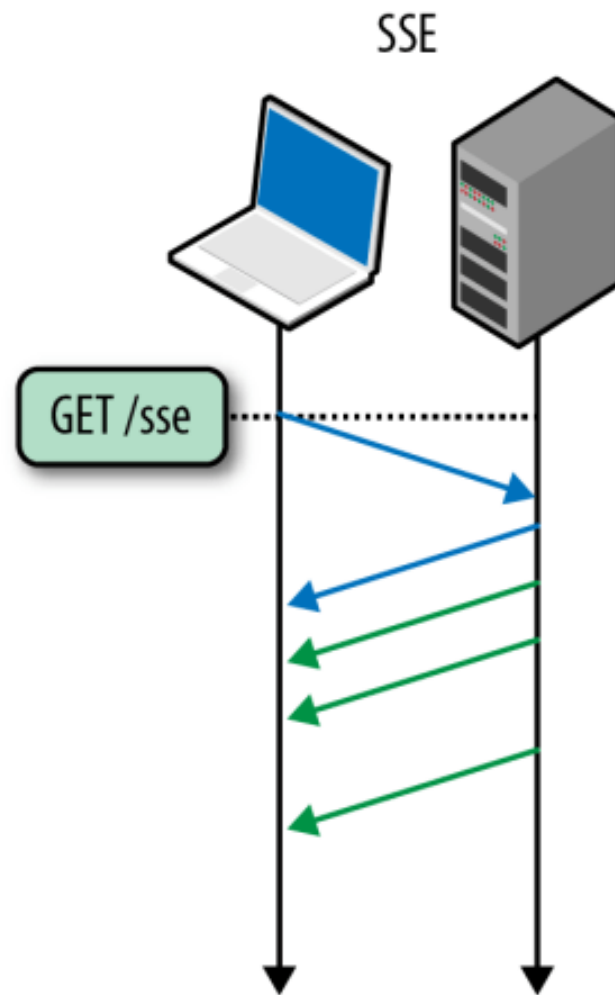
Types of Access

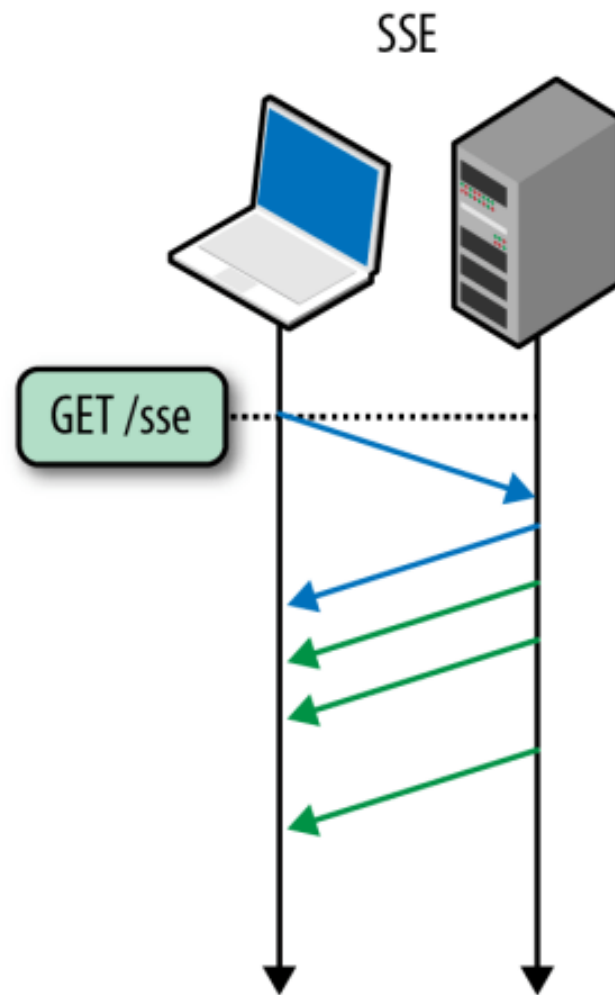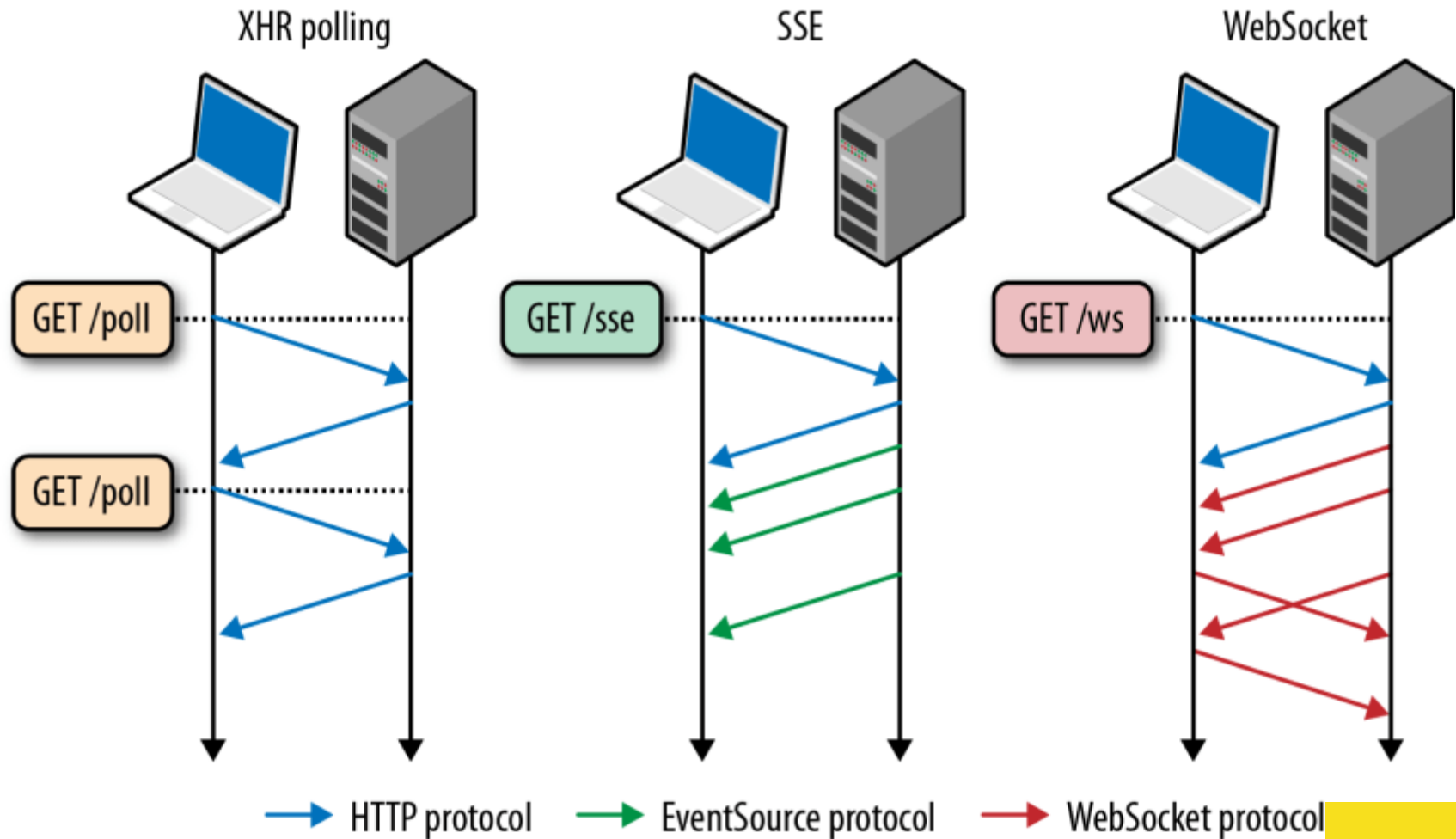Websockets
SSE
XHR

JS

Websockets

SSE

SSE

Comparison

XHR

XMLHttpRequest

XMLHTTP changed everything. It put the "D" in DHTML. It allowed us to asynchronously get data from the server and preserve document state on the client… The Outlook Web Access (OWA) team's desire to build a rich Win32 like application in a browser pushed the technology into IE that allowed AJAX to become a reality.
— Jim Van Eaton Outlook Web Access: A catalyst for web evolution

JS

XMLHttpRequest

Create the request - new XMLHttpRequest()
Initialise the parameters - xhttp.open()
Process the response - xhttp.onreadystatechange

JS

XMLHttpRequest

```javascript
function loadDocAJAX() {
    var xhttp = new XMLHttpRequest();
    xhttp.onreadystatechange = function() {
        if (this.readyState == 4 && this.status == 200) {
            document.getElementById("demo").innerHTML =
            this.responseText;
        }
    };
    xhttp.open("GET", "ajax_info.txt", true);
    xhttp.send();
}
```
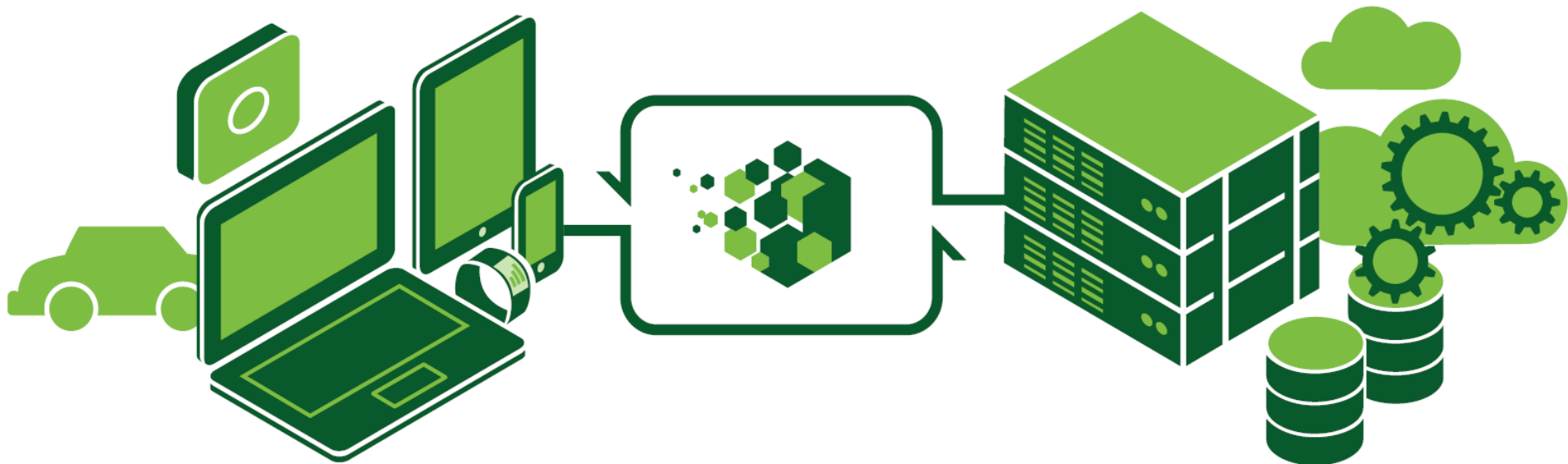
JSON

JSON

JS Object Notation
Data exchange format

```
{
    "id": "1242160000000072038",
    "description": "3",
    "website": "3",
    "numberOfEmployees": "3",
    "phone": "3",
    "name": "account3",
    "shippingAddress": {
        "country": "3",
        "stateOrProvidence": "3",
        "city": "3",
        "postalCode": "3",
        "street1": "3"
    },
    "billingAddress": {
        "country": "3",
        "stateOrProvidence": "3",
        "city": "3",
        "postalCode": "3",
        "street1": "3"
    }
}
```

REST

NAH!

Architecture

JS Object Notation
Data exchange format

RESTful API

JS Object Notation
Data exchange format

Thank You!