Here’s the structured HTML text based on the input provided: ```html

HTTP, RST, and HTTP Architecture

Introduction

HTTP stands for Hypertext Transfer Protocol. It is the dominant protocol for transmitting data—such as HTML pages, images, and videos—between clients and servers on the internet.

Response Model

Request Phase

The response model operates on a request-response model in which the client sends a request to the server, and the server responds with the requested data or an error message.

Server-Side Response Model

Statelessness

The response model is stateless, meaning that the server handles each request independently without any knowledge of previous requests.

HTTP Methods

HTTP Methods

In fact, it is not possible to send a request to a REST API without an HTTP method. These methods enable clients to specify the action they'd like to perform on a given resource.

HTTP Architecture

Stateless Interactions

The architecture of HTTP is stateless, meaning that each request is handled independently by the server. This eliminates the need for tracking previous requests, which simplifies communication and ensures consistency.

REST APIs

Overview

REST stands for Representational State Transfer. It is the most commonly used architectural style for building web services and APIs, and it emphasizes standardized, stateless interactions between clients and servers.

REST APIs as Stateless Interactions

Unique API Endpoints

These are designed around resources that are accessible via unique API endpoints. These characteristics make HTTP the ideal choice for implementing RESTful principles.

Resources and Resources Endpoints

Resources

Which Endpoints are Unique

The resources that use unique endpoints are typically those that can't be retrieved using standard HTTP methods. For example, file transfers, form submissions, and certain REST operations may require unique endpoint identifiers.

Conclusion

In summary, HTTP is a stateless protocol that operates on a request-response model to communicate with servers. It plays a crucial role in the architecture of REST APIs. Understanding these principles helps in designing robust and scalable web applications.