NGINX

NGINX: Comparison with Apache

NGINX : Web-Server & Load Balancer

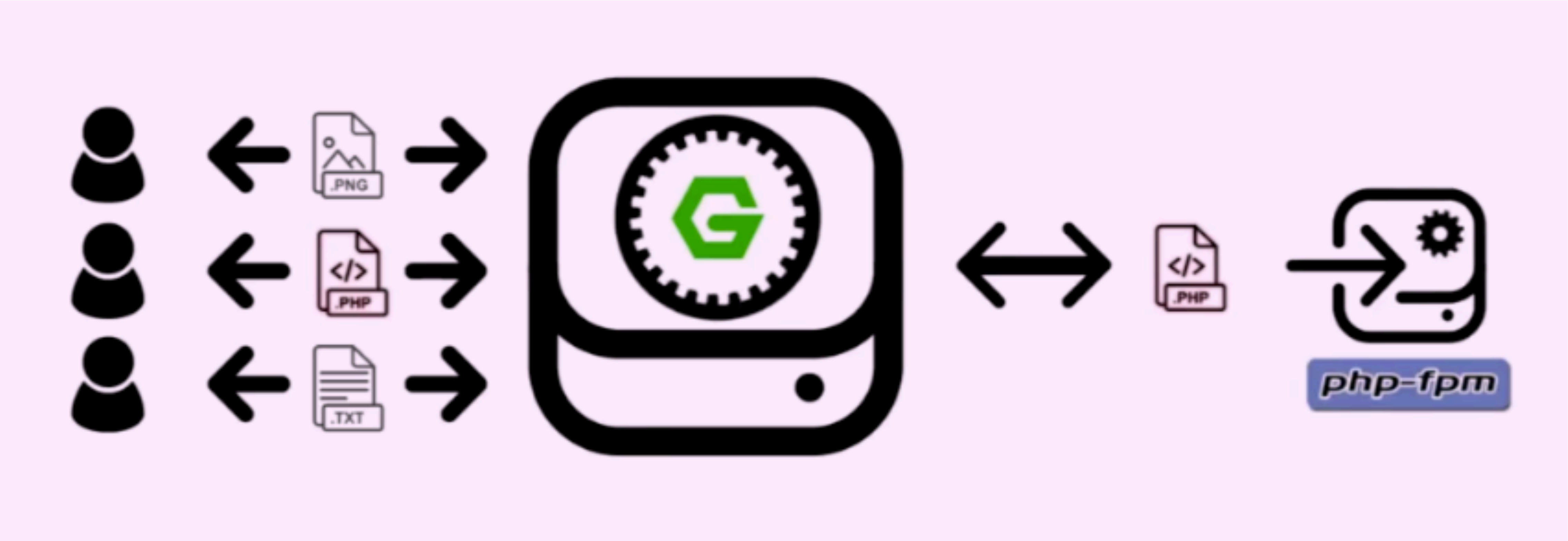
- **NGINX** and Apache both are web-servers and have their own characteristics.
- The main difference between Apache and NGINX lies in their design architecture. Apache uses a process-driven approach and creates a new thread for each request. Whereas NGINX uses an event-driven architecture to handle multiple requests within one thread.
- **Simplicity** -
  - Developing and innovating applications on Apache is easy. one-connection-per-process model makes it very easy to insert modules at any point in its web serving logic.
  - NGINX module developers need to be very careful to create efficient and accurate code, without any failures, and to interact appropriately with the complex event-driven kernel to avoid blocking operations.

NGINX : Web-Server & Load Balancer

- **Performance-**
- Apache performs better when hosting sites that experience relatively low levels of traffic – perhaps 1000 requests or fewer per hour. NGINX performs better for sites that experience lots of requests simultaneously.
- Reason of poor performance in Apache is .htaccess file I/O Operations. Also Caching Model of NGINX.
- Static Content - NGINX performs 2.5 times faster than Apache according to a benchmark test performed by running up to 1,000 simultaneous connections.
- Dynamic Content - Apache process dynamic content by embedding a processor of a language like PHP into each of its worker instances. This allows it to execute dynamic content within the web server itself without having to rely on external components.
- NGINX does not have any ability to process dynamic content natively. To handle PHP and other requests for dynamic content, NGINX must pass to an external processor for execution and wait for the rendered content to be sent back.

NGINX : Web-Server & Load Balancer

- Performance-



- **Static:** As far as Static content is concerned, **Nginx** overpasses Apache.
- **Dynamic:** Both are great at processing Dynamic content.

NGINX : Web-Server & Load Balancer

- **OS Support-**
- Apache runs on all operating systems such as UNIX, Linux or BSD and has full support for Microsoft Windows.
- NGINX also runs on several modern Unix-like systems and has support for Windows, but its performance on Windows is not as stable as that on UNIX platforms.
- **Distributed / Centralized Configuration**
- Apache allows additional configuration on a per-directory basis via .htaccess files.
- Nginx doesn’t allow additional configuration

NGINX : Web-Server & Load Balancer

- **Request Interpretation-**
- Apache pass file System location.
- Nginx Passes URI to interpret requests.
- This very design of interpreting requests as URI locations allows Nginx to easily function as not only a web server but also as a proxy server, load balancer, and HTTP cache.
- **Nginx** has won because of its quicker interpretation and response.

NGINX : Web-Server & Load Balancer

- **Feature Modules-**
- Apache have 60 official dynamically loadable modules that can be turned On/Off.
- Nginx have 3rd Party core modules (not dynamically loadable).
- NGINX provides all of the core features of a web server, without sacrificing the lightweight and high–performance qualities that have made it successful.

NGINX : Web-Server & Load Balancer

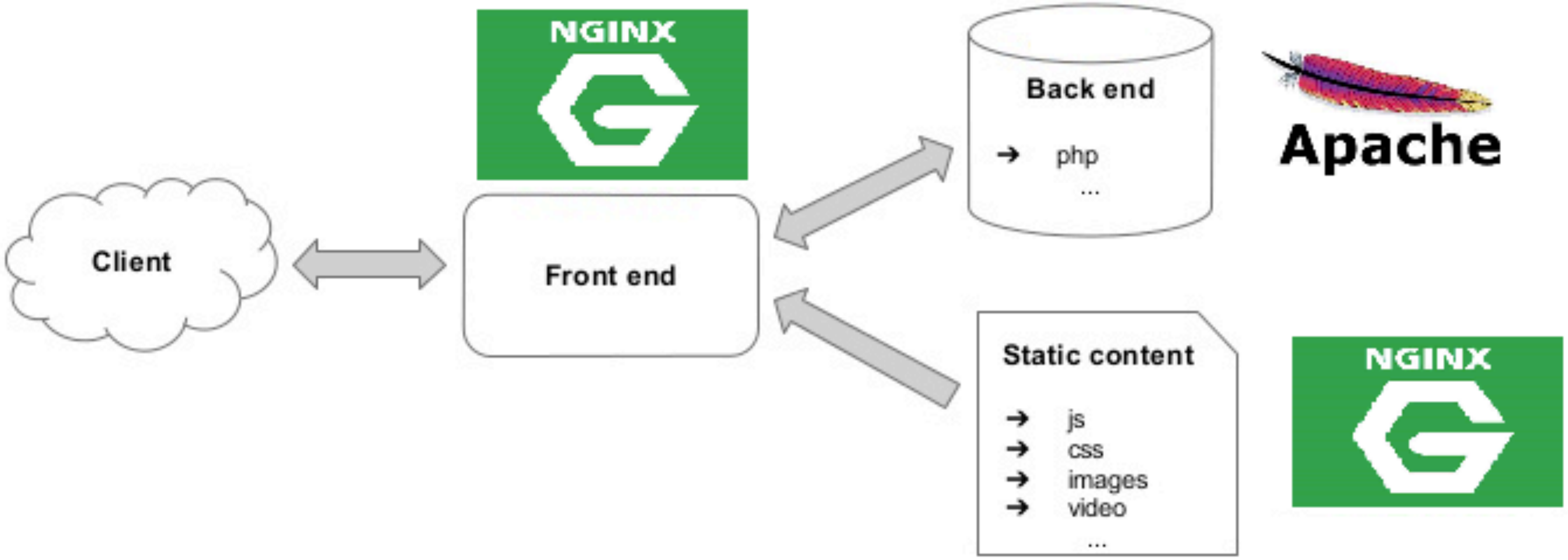
- **Flexibility-**
- Apache Supports customization of web server through dynamic modules.
- Nginx is not flexible enough to support dynamic modules and loading.
- **Security -**
- Apache makes sure that all the website that runs on its server are safe from any harm and hackers.
- Apache offers configuration tips for DDoS attack handling, as well as the mod_evasive module for responding to HTTP DoS, DDoS, or brute force attacks.
- Nginx - Better security with the smaller codebase.

NGINX : Web-Server & Load Balancer

- **When Choose Apache over NGINX?**
- When needs .htaccess files, you can override system-wide settings on a per-directory basis.
- In a shared hosting environment, Apache works better because of its .htaccess configuration.
- In case of functionality limitations – use Apache
- **When Choose NGINX over Apache?**
- Fast Static Content Processing
- Great for High Traffic Websites

NGINX : *Web-Server & Load Balancer*

- **Use Both of them -Together**
- User can use Nginx in front of Apache as a server proxy.



- For static connections, Nginx will serve the files quickly to the clients. For dynamic content, for example, Php files, Nginx reverse proxy server will proxy their request to Apache which can then process their results and return their rendered page.

Will see you in Next Lecture…

See you in next lecture …