

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN  
KHOA CÔNG NGHỆ THÔNG TIN

# **FILTER**

**Nguyễn Hoàng Anh**  
**[nhanh@fit.hcmus.edu.vn](mailto:nhanh@fit.hcmus.edu.vn)**

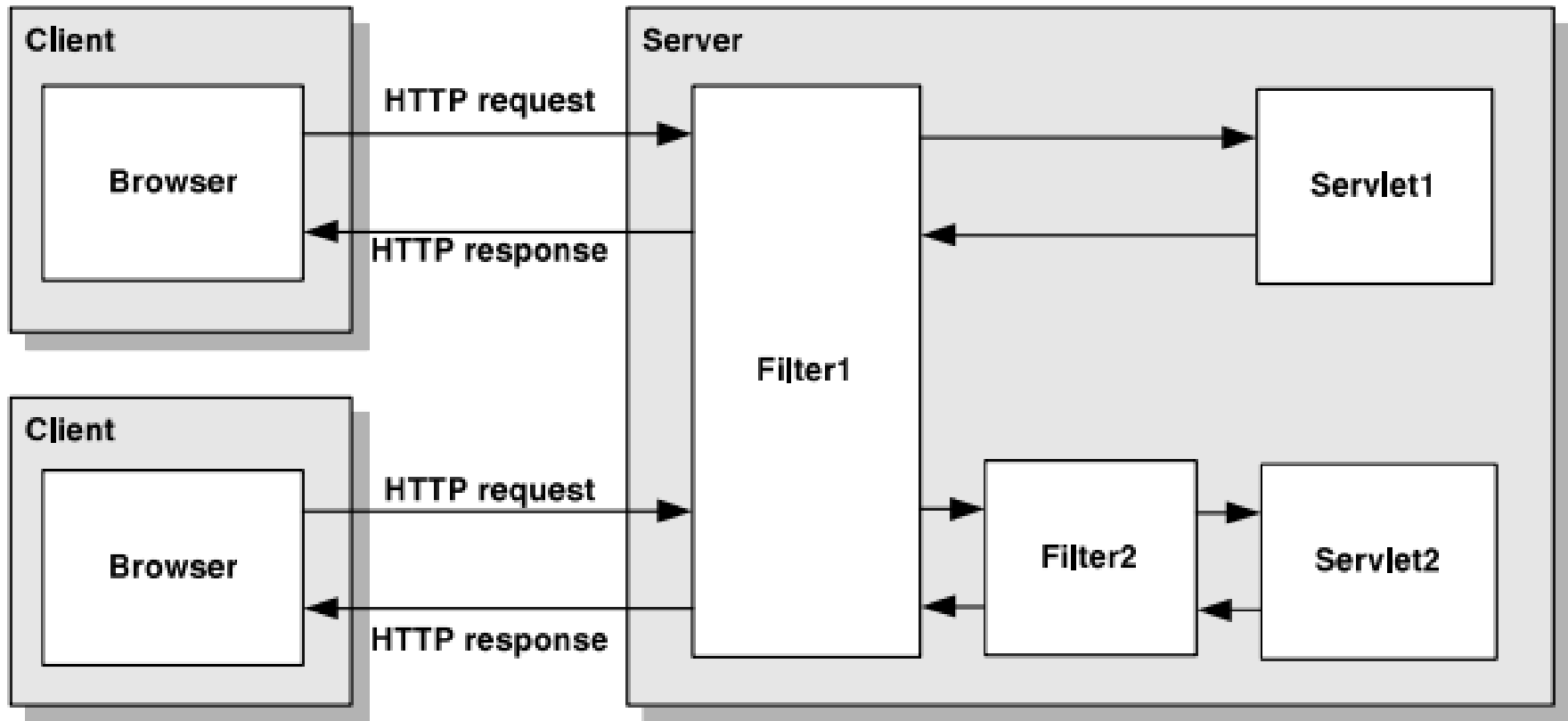
# Nội dung

- Giới thiệu Filter.
- Cách tạo Filter.
- Cách cấu hình Filter.
- Thứ tự thực thi các Filter.

# Filter

- **Filter** được sử dụng để **chặn** một **HTTP Request** và **xử lý trước** khi **HTTP Request** được truyền đến **Servlet/JSP**.
- **Filter** cũng được sử dụng để **chặn** một **HTTP Response** và **xử lý trước** khi **HTTP Response** được gửi về **client**.
- Có thể tạo một **chuỗi các Filter**.

# Filter



# Lợi ích khi sử dụng Filter

- **Modular code**: **Filter** cho phép **lưu trữ code** tại **một nơi duy nhất** nhưng có **thể áp dụng** cho **nhiều thành phần** trong **web application**.
- **Flexible code**: có thể sử dụng **web.xml** để **cấu hình** các **Filter** cho **các thành phần khác nhau** của **web application** và **bật/tắt các filter**.

# Những chức năng thông dụng của Filter

- Logging
- Authentication
- Compression
- Localization
- Image conversion
- Block IP
- ...

# Tạo Filter

```
1 public class Filter1 implements Filter {
2     public void init(FilterConfig fConfig)
3         throws ServletException {
4         . . .
5     }
6     public void doFilter(ServletRequest request,
7                         ServletResponse response,
8                         FilterChain chain)
9         throws IOException, ServletException {
10        // place your code here
11        . . .
12        chain.doFilter(request, response);
13        // place your code here
14        . . .
15    }
16    public void destroy() { . . . }
}
```

# Cấu hình Filter với web.xml

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <web-app>
3    <filter>
4      <filter-name>Filter1</filter-name>
5      <filter-class>org.nhanh.filter.Filter1</filter-class>
6    </filter>
7    <filter>
8      <filter-name>Filter2</filter-name>
9      <filter-class>org.nhanh.filter.Filter2</filter-class>
10   </filter>
11   <filter>
12     <filter-name>Filter3</filter-name>
13     <filter-class>org.nhanh.filter.Filter3</filter-class>
14   </filter>
15   <filter-mapping>
16     <filter-name>Filter1</filter-name>
17     <url-pattern>/*</url-pattern>
18   </filter-mapping>
19   <filter-mapping>
20     <filter-name>Filter2</filter-name>
21     <url-pattern>/*</url-pattern>
22     <dispatcher>REQUEST</dispatcher>
23     <dispatcher>FORWARD</dispatcher>
24   </filter-mapping>
25   <filter-mapping>
26     <filter-name>Filter3</filter-name>
27     <servlet-name>/template.do</servlet-name>
28   </filter-mapping>
29 </web-app>
```



# Cấu hình Filter với annotation

```
@WebFilter(  
    filterName="/Filter1",  
    urlPatterns="/*"  
)  
public class Filter1 implements Filter {...}
```

```
@WebFilter(  
    filterName="/Filter1",  
    urlPatterns={"/admin/*"},  
    dispatcherTypes={DispatcherType.REQUEST, DispatcherType.FORWARD}  
)  
public class Filter2 implements Filter{...}
```

```
@WebFilter(  
    filterName="/Filter3",  
    urlPatterns={"/template.do"},  
)  
public class Filter3 implements Filter{...}
```

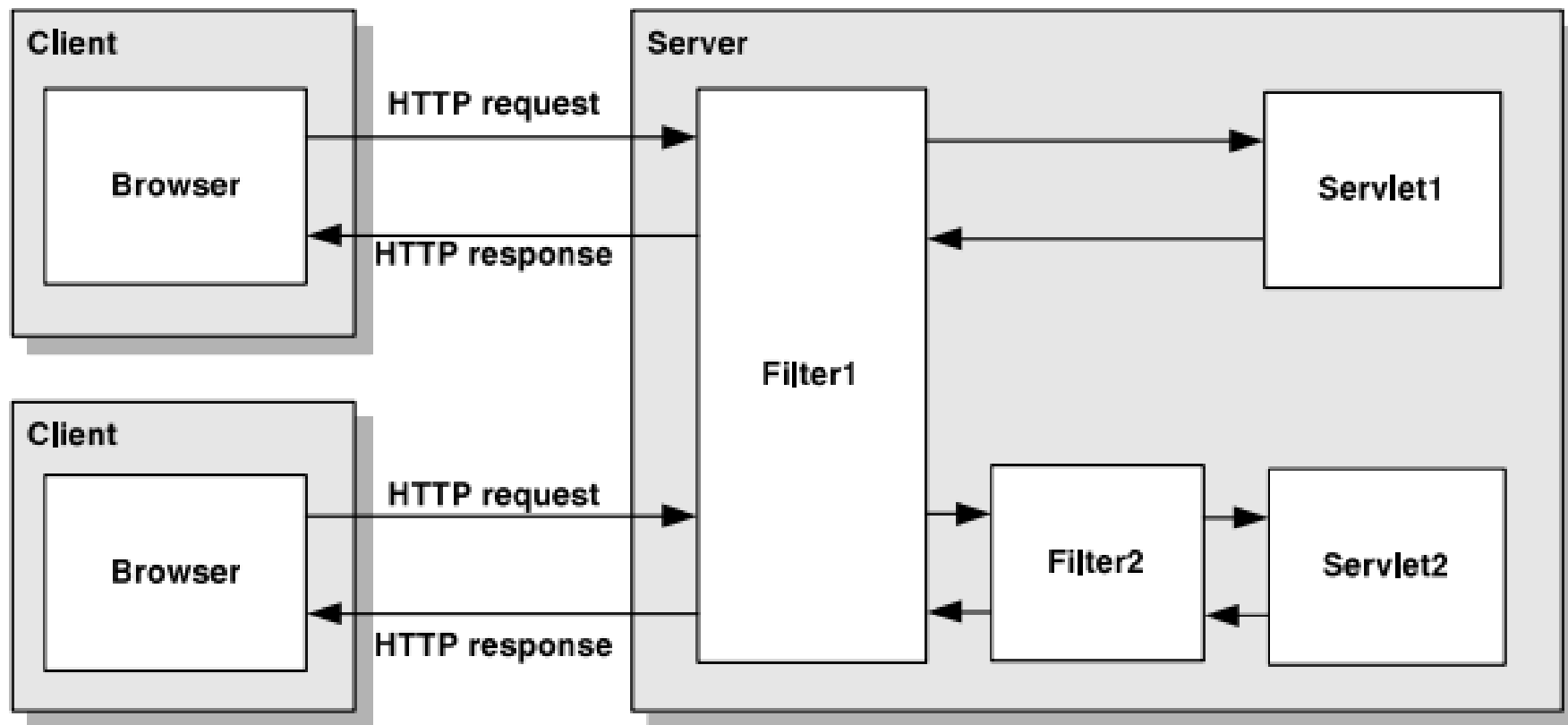
# Thứ tự thực thi các Filter

- Các **request filter** được **thực hiện** theo **thứ tự** được khai báo trong **web.xml**
- Các **response filter** được thực hiện theo thứ tự ngược lại.

# Tạo Filter

```
1 public class Filter1 implements Filter {
2     public void init(FilterConfig fConfig)
3         throws ServletException {
4         . . .
5     }
6     public void doFilter(ServletRequest request,
7                         ServletResponse response,
8                         FilterChain chain)
9         throws IOException, ServletException {
10        //request-side processing
11        . . .
12        chain.doFilter(request, response);
13        //response-side processing
14        . . .
15    }
16    public void destroy() { . . . }
17 }
```

# Thứ tự thực thi các Filter



# Demo





# HỎI VÀ ĐÁP

# Reference

- [1] Joel Murach; Michael Urban. (2014). Murach's Java Servlets and JSP, 3<sup>rd</sup> Edition. Mike Murach & Associate.