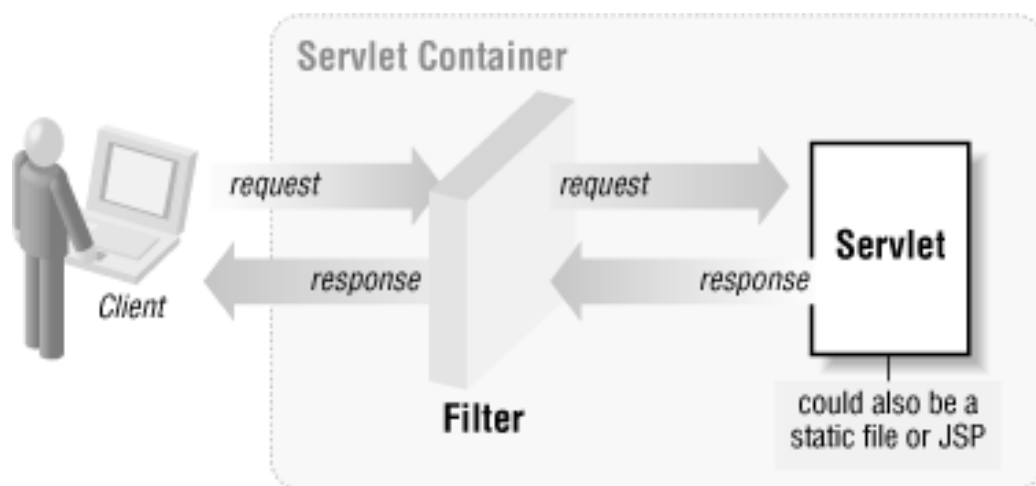


Интернет технологии

Servlet филтри. Същност и принципи на използване

Servlet филтри

- Сървлет филтрите са Java класове, които могат да бъдат използвани в Servlet програмирането за следните цели :
 - За прихващане на заявки от клиент преди да му се даде достъп до ресурсите в системата;
 - За манипулация на отговорите от сървъра преди те да бъдат изпратени обратно на клиента.



Видове филтри

- за аутентикация;
- за компресия на данни;
- за кодиране и декодиране;
- за управление достъпа до ресурси;
- за преобразуване на изображения;
- за управление на достъпа до системата;
- верига от филтри за MIME - TYPE;
- за токенайзер;
- XSL / T филтри (трансформации на XML съдържание)

Filter API

- Filter
- FilterChain
- FilterConfig

Filter interface

- `public void init(FilterConfig filterConfig)` - извиква се от уеб контейнера, за да се окаже на филтъра, че е вече активен;
- `public void doFilter (ServletRequest, ServletResponse, FilterChain)` - извиква се от контейнера всеки път при обработка на request/response, в който се изисква достъп до ресурс от клиент;
- `public void destroy()` - извиква се от уеб контейнера, за да се окаже на филтъра, че вече не е активен.

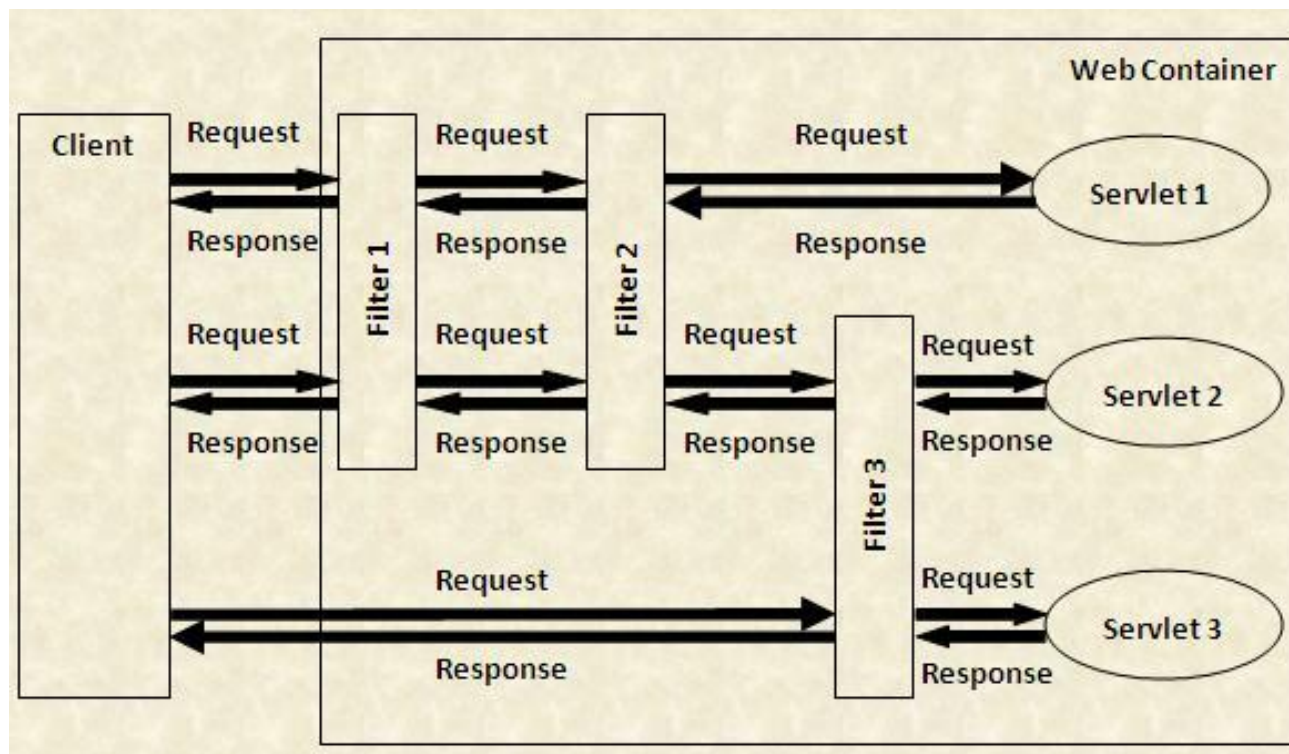
FilterChain interface

- FilterChain

FilterConfig interface

- `public void init(FilterConfig config)`
- `public String getInitParameter(String parameterName)`
- `public java.util.Enumeration getInitParameterNames()`
- `public ServletContext getServletContext()`

Верига от филтри (Filter chain)



Декларация на филтър

- web.xml
 - Верига от филтри
- @WebFilter

Филтър за IP адрес

```
@WebFilter(urlPatterns = "/CheckBox")
public class LogFilter implements Filter {

    public void init(FilterConfig fConfig) throws ServletException { }

    public void doFilter(ServletRequest request, ServletResponse response, FilterChain chain) throws
        IOException, ServletException {

        String ipAddress = request.getRemoteAddr();
        System.out.println("IP "+ ipAddress + ", Time: " + new Date().toString());
        chain.doFilter(request, response);
    }

    public void destroy() { }

}
```

Филтър за повече от един ресурс

```
@WebFilter(servletNames = {"MyOwnServlet", "UploadServlet"})  
public class MyFilter implements Filter {  
    ...  
}
```

```
@WebFilter(urlPatterns = {"MyOwnServlet", "UploadServlet"})  
public class MyFilter implements Filter {  
    ...  
}
```

Филтър с инициализиращи параметри

```
@WebFilter(  
    urlPatterns = "/uploadFilter",  
    initParams = @WebInitParam(name = "fileTypes", value =  
        "doc;xls;zip;txt;jpg;png;gif")  
)  
public class UploadFilter implements Filter {  
    ...  
}
```

Филтър с допълнителна информация

```
@WebFilter(  
    urlPatterns = "/admin/*",  
    filterName = "AdminFilter",  
    description = "Filter all admin URLs"  
)  
public class MyFilter implements Filter {  
    ...  
}
```

Филтър с оказване типа на диспечера на обработката

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

ServletFilter пример

```
import java.io.*;
import javax.servlet.*;

public class MyFilter implements Filter{
    public void init(FilterConfig arg0) throws ServletException {}

    public void doFilter(ServletRequest req, ServletResponse res,
        FilterChain chain) throws IOException, ServletException {

        PrintWriter out=res.getWriter();

        out.print("<br/>this site is underconstruction..");
        out.close();

    }
    public void destroy() {}
}
```

ServletFilter пример

```
import java.io.*;

import javax.servlet.*;

public class MyFilter implements Filter {

    static int count=0;

    public void init(FilterConfig arg0) throws ServletException {}

    public void doFilter(ServletRequest req, ServletResponse res, FilterChain chain) throws IOException,
        ServletException {

        PrintWriter out=res.getWriter();
        chain.doFilter(request,response);
        out.print("<br/>Total visitors "+ (++count));
        out.close();
    }

    public void destroy() {}
}
```


ServletFilter пример

```
import java.io.*;

import javax.servlet.*;

public class MyFilter implements Filter {

    static int count=0;

    public void init(FilterConfig arg0) throws ServletException {}

    public void doFilter(ServletRequest req, ServletResponse res,
        FilterChain chain) throws IOException, ServletException {
        PrintWriter out=res.getWriter();
        long before=System.currentTimeMillis();
        chain.doFilter(request,response);
        long after=System.currentTimeMillis();
        out.print("<br/>Total response time "+(after-before)+" milliseconds");
        out.close();
    }

    public void destroy() {}
}
```