

Le développement Web côté serveur avec Java EE

Module 7 – Les filtres et les événements

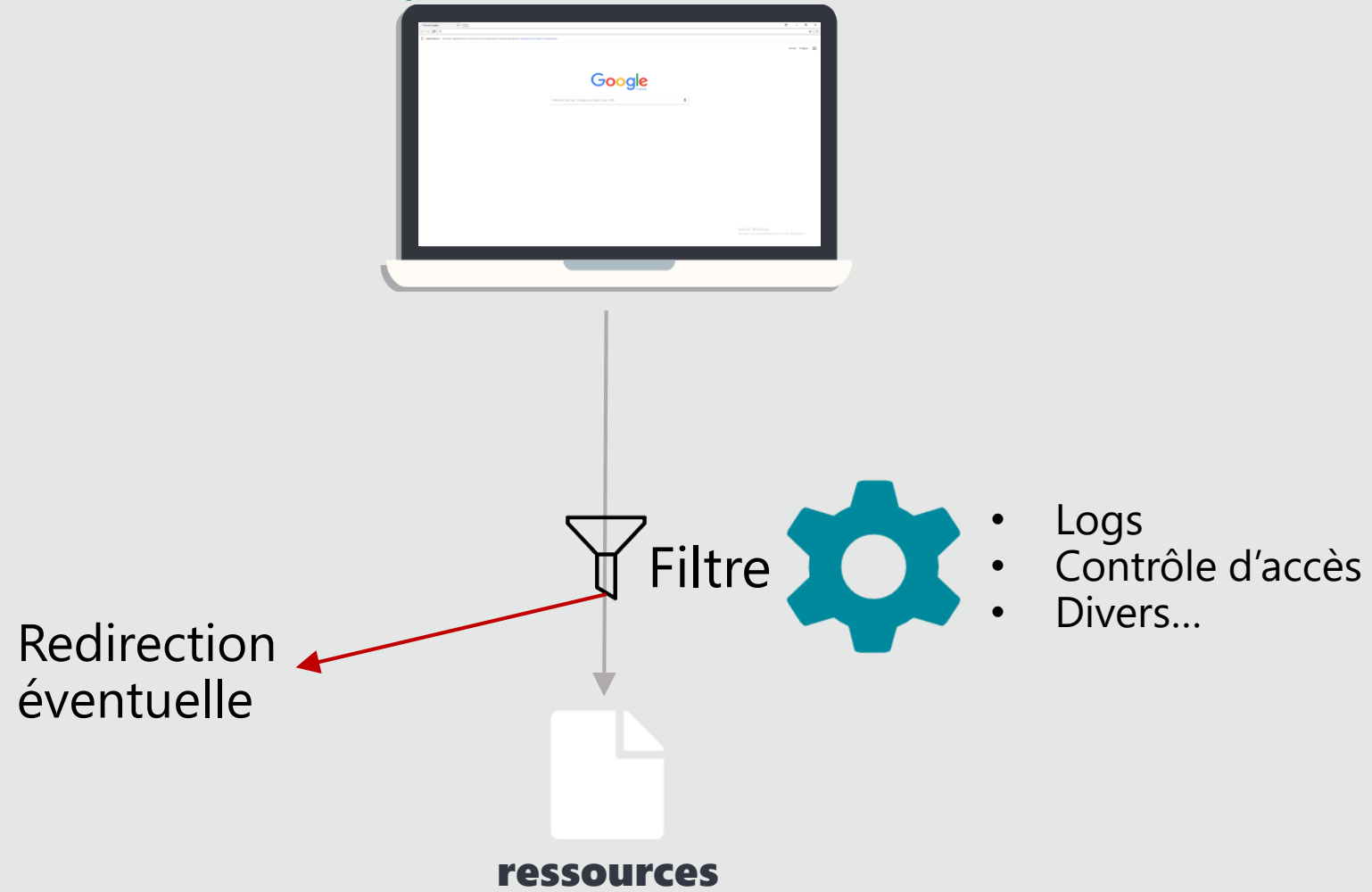


Objectifs

- Comprendre le mécanisme des filtres
- Comprendre le mécanisme des événements

Les filtres et les événements

Les filtres – le concept



Les filtres – la configuration



web.xml

```
<filter>
    <filter-name>NomDuFiltre</filter-name>
    <filter-class>nom.complet.de.la.Classe</filter-class>
</filter>
<filter-mapping>
    <filter-name>NomDuFiltre</filter-name>
    <dispatcher>REQUEST</dispatcher>
    <dispatcher>FORWARD</dispatcher>
    <dispatcher>INCLUDE</dispatcher>
    <dispatcher>ERROR</dispatcher>
    <url-pattern>/modele/url/*</url-pattern>
</filter-mapping>
```

Les filtres – la configuration



```
@WebFilter(dispatcherTypes = {  
    DispatcherType.REQUEST,  
    DispatcherType.FORWARD,  
    DispatcherType.INCLUDE,  
    DispatcherType.ERROR  
},  
urlPatterns = { "/modele/url/*" })
```

Les filtres – le code

```
@WebFilter(...)
public class FiltreRessourcesInterdites implements Filter
{
    public void doFilter(ServletRequest request,
                        ServletResponse response,
                        FilterChain chain)
                        throws IOException, ServletException
    {
        // votre code ici
        // transmission de votre requête sur la chaîne des filtres si
        // aucune redirection définie
        chain.doFilter(request, response);
    }
}
```

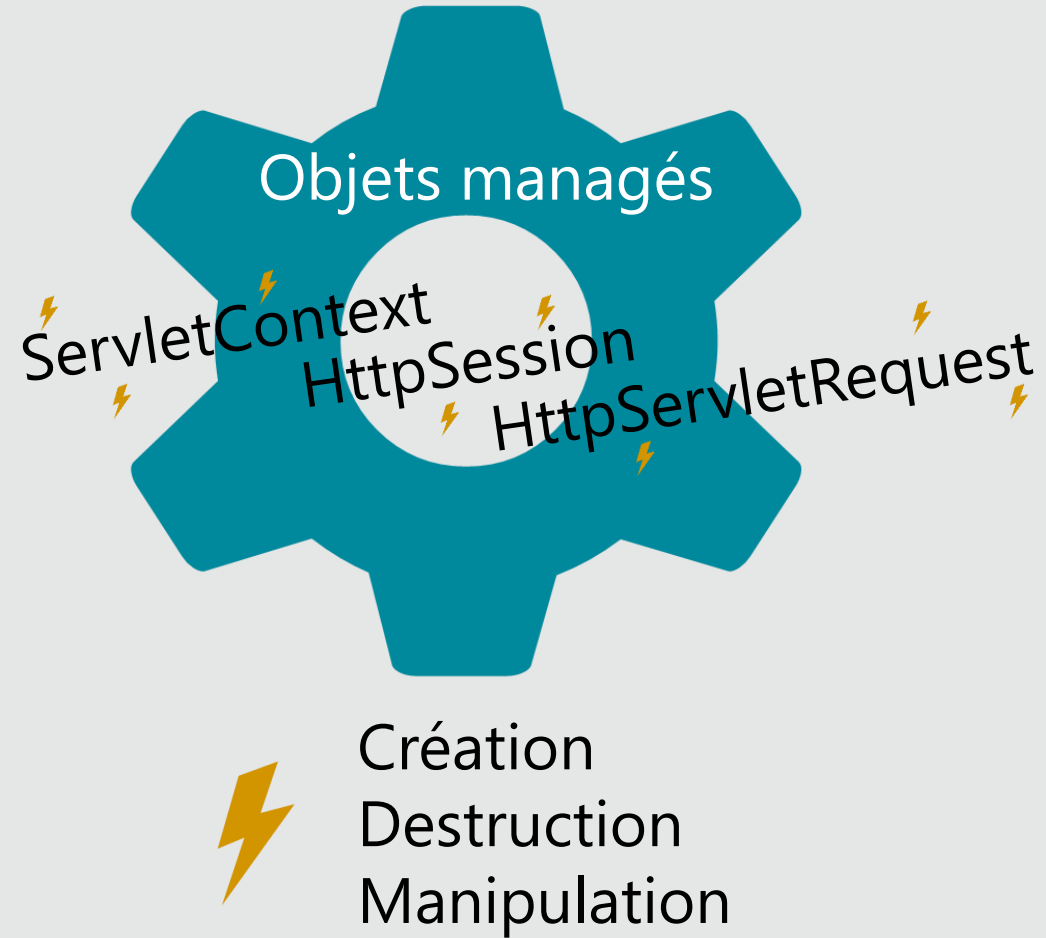
Les filtres et les événements

Les filtres

Démonstration



Les événements – le concept



Les filtres et les événements

Les événements – les listeners disponibles

Create Listener

Select the application lifecycle events to listen to.

Servlet context events

<input checked="" type="checkbox"/> Lifecycle	javax.servlet.ServletContextListener
<input type="checkbox"/> Changes to attributes	javax.servlet.ServletContextAttributeListener

HTTP session events

<input type="checkbox"/> Lifecycle	javax.servlet.http.HttpSessionListener
<input type="checkbox"/> Changes to attributes	javax.servlet.http.HttpSessionAttributeListener
<input type="checkbox"/> Session migration	javax.servlet.http.HttpSessionActivationListener
<input type="checkbox"/> Object binding	javax.servlet.http.HttpSessionBindingListener
<input type="checkbox"/> Changes to id	javax.servlet.http.HttpSessionIdListener

Servlet request events

<input type="checkbox"/> Lifecycle	javax.servlet.ServletRequestListener
<input type="checkbox"/> Changes to attributes	javax.servlet.ServletRequestAttributeListener
<input type="checkbox"/> Async events	javax.servlet.AsyncListener

Select All Deselect All

? < Back Next > Finish Cancel

Les filtres et les événements

Les événements – la configuration



web.xml

```
<listener>  
    <listener-class>nom.complet.de.la.Classe</listener-class>  
</listener>
```

Les filtres et les événements

Les événements – la configuration



@WebListener

Les événements – le code

```
@WebListener  
public class ListenerPersonnalise implements XxxListener {  
    ...  
}
```

Les filtres et les événements

Les événements

Démonstration



Les filtres et les événements

Gérer vos listes de courses (évolution)

TP



Conclusion

- Vous maîtrisez la manipulation des filtres et des événements

