



CS544 EA

Applications

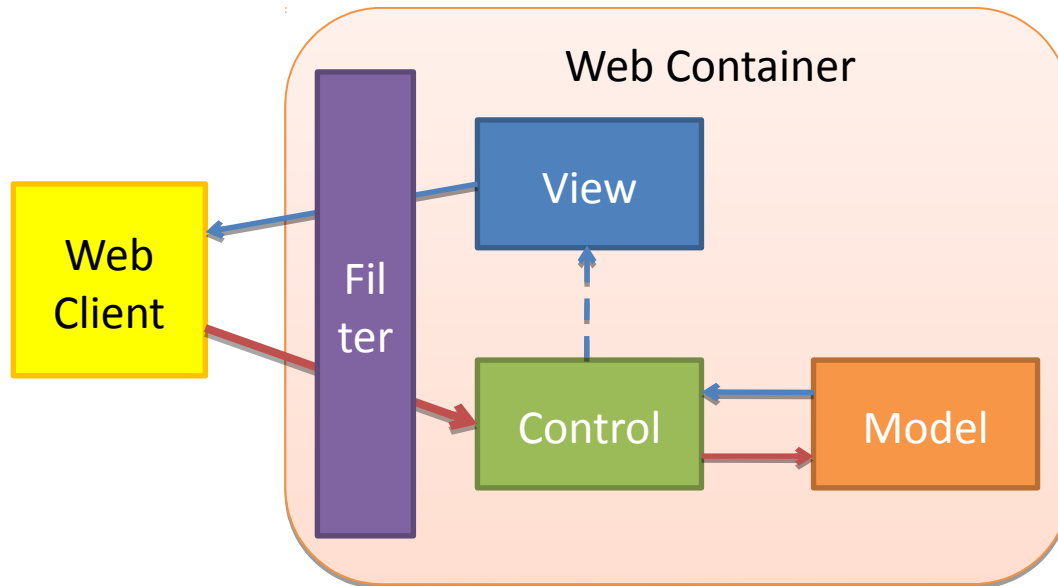
Hibernate Web: Open EntityManager in View

View

- Having all the entity objects loaded before they reach the view is a **best practice**
- Nevertheless, people find it frustrating that they cannot do lazy loading in the view
 - Solution: **OpenEntityManagerInView** pattern
 - Very popular / but some see it as an anti-pattern

OpenEntityManagerInView

- **Uses a filter** to open TX before controller
 - Closes TX after view is done



Open in View Filter

```
@WebFilter(filterName = "OpenEntityManagerInView", urlPatterns = "/*")
public class EntityManagerInterceptor implements Filter {
    @Override
    public void destroy() { }
    @Override
    public void init(FilterConfig fc) throws ServletException { }
    @Override
    public void doFilter(ServletRequest req, ServletResponse res, FilterChain chain)
        throws IOException, ServletException {

        EntityManager em = EntityManagerHelper.getCurrent();
        try {
            em.getTransaction().begin();
            chain.doFilter(req, res);
            em.getTransaction().commit();
        } catch (RuntimeException e) {

            if (em != null && em.isOpen())
                em.getTransaction().rollback();
            throw e;

        } finally {
            em.close();
        }
    }
}
```

Wraps all Servlets
(controller and view)
In one big transaction

Service Method

- **Just load first object**
 - Any related objects can be loaded by view

```
public class CustomerService {  
    ...  
    public Customer getCustomer(Long id) {  
        return customerDao.load(id);  
    }  
}
```

No longer has
Transaction code
(TX inside Filter)

No longer loads related
data (view can do so)

Criticism

Not possible to add
a console app
or JavaFX app

- **Ties the Service Layer to Web Environment**
 - Only a web environment can lazy load (with filter)
- Can create Transaction that is too big
 - But each request usually calls only one service method
- Database activity happens during view
 - Some consider it a violation of layers
 - But it's not seen in code (just happens)