

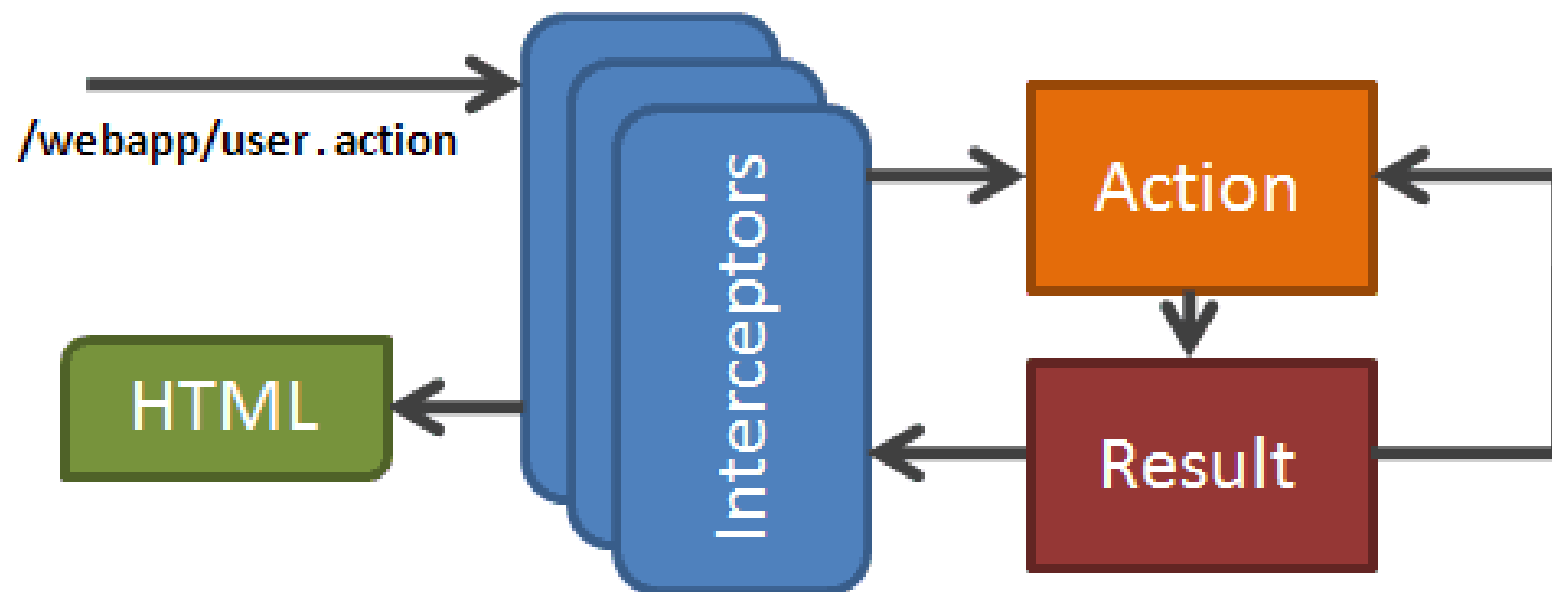
Framework



Qu'est-ce qu'un framework ?

- Structure imposée par le framework
- Utilisation de « briques logicielles » génériques
- Le développeur complète le framework avec les traitements non-génériques
- Maintenabilité améliorée
- Fichier de configuration

Framework web - Struts



Struts - Configuration

```
<action
  name="helloCustomerAction"
  class="helloCustomerAction"
  method="helloCustomer">
  <result name="success">
    pages/customer.jsp
  </result>
</action>
```

```
@Controller
public class HelloCustomerAction
    extends ActionSupport{

    public String helloCustomer() {
        //Do something
        return "success" ;
    }

}
```



customer.jsp

Struts

```
@Controller
public class ListCustomerAction
    extends ActionSupport{

    private String nom;

    public String getCustomer(){
        //Do something
        Nom = "test";
        return "success" ;
    }

    public String getNom(){
        return nom ;
    }

    public void setNom(String nom){
        this.nom = nom ;
    }

}
```

customer.jsp

```
<%@ taglib prefix="s" uri="/struts-
tags"%>
<html>
<body>

    Hello ${nom}

</body>
</html>
```

Struts - taglib

```
<%@ taglib prefix="s" uri="/struts-tags"%>
```

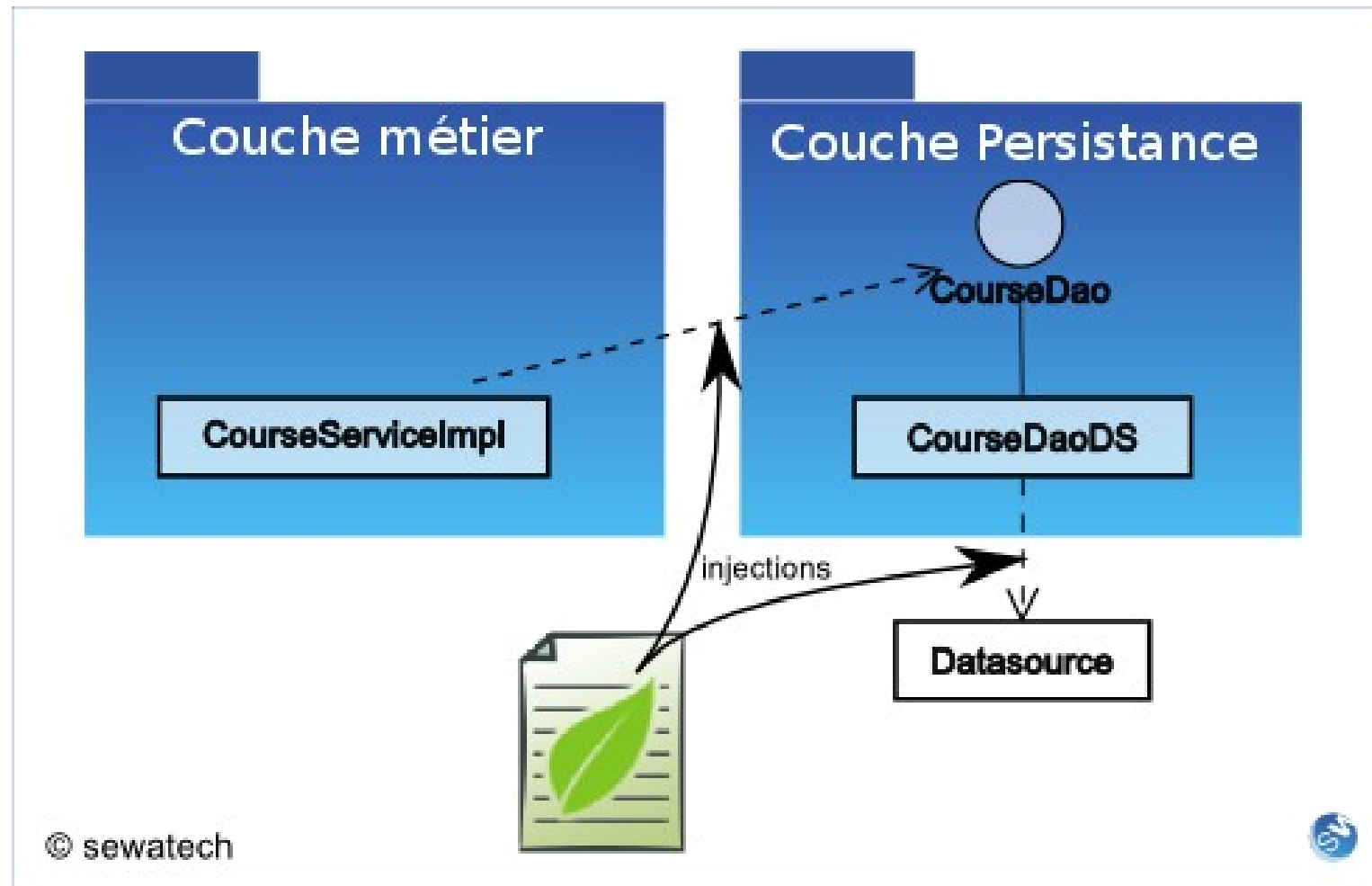
```
s:if  
s:else  
s:elseif  
s:iterator
```

```
s:a  
s:action  
s:param  
s:url
```



- Spring IOC
- Spring MVC
- Spring DATA
- Spring Cloud
- Spring Batch
- Spring Security
- etc.

Spring IOC



Spring IOC

```
Public class CustomerService{  
  
    @Autowired  
    Private CustomerDAO  
        customerDAO ;  
  
    public void printCustomers(){  
        customerDAO.getCustomers();  
    }  
  
    //getter et setter customerDAO  
  
}
```

```
public interface CustomerDAO{  
  
    List<Customer>  
    getCustomers() ;  
  
    Customer getCustomer(int id) ;  
  
}  
  
@Repository  
public class CustomerDAOImpl  
    implements CustomerDAO{  
  
    Public List<Customer>  
        GetCustomers(){  
            //Do something  
        }  
  
    Public Customer  
        getCustomer(int id){  
            //Do something  
        }  
  
}
```

Spring IOC

Et les autres ?

Framework Web



tapestry



JavaServer™ Faces

JSF



RichFaces

Framework IOC



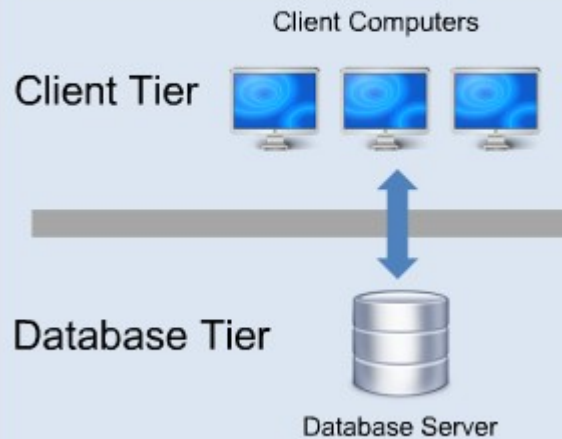
Guice

Architecture

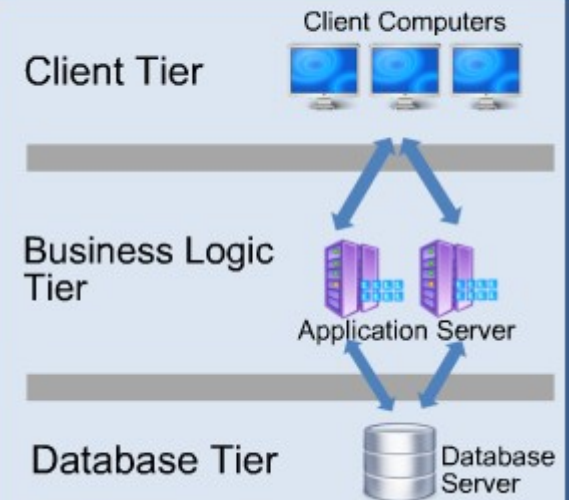
1-Tier Architecture



2-Tier Architecture



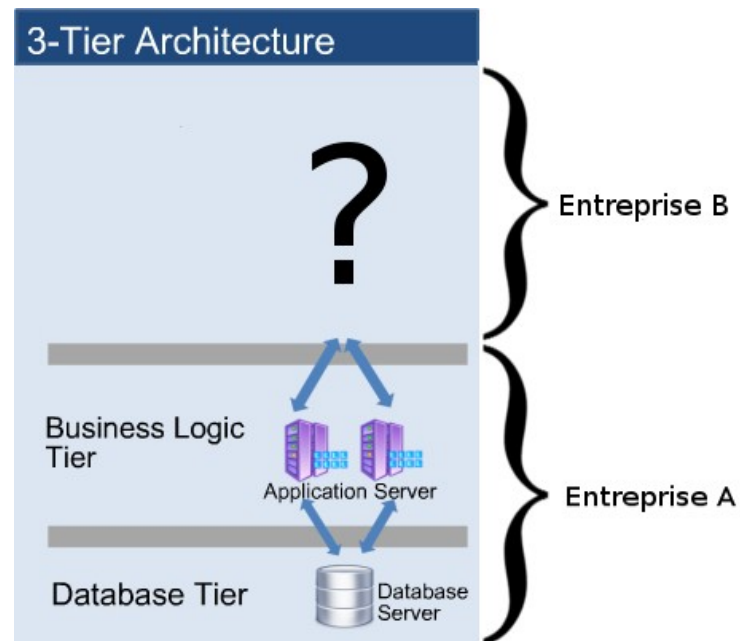
3-Tier Architecture



POJO vs EJB

```
public class CustomerPOJO{  
    private String nom ;  
  
    private List<CommandePOJO>  
        commandes ;  
  
    //getters & setters  
}
```

```
@Entity  
public class Customer{  
    private String nom ;  
  
    private List<Commande>  
        commandes ;  
  
    //getters & setters  
}
```



Conclusion

Source :

- Cours / exemples : <http://www.jmdoudoux.fr/>
- Exemples : <http://www.mkyong.com/>
- Architecture 3-tier : <http://blog.simcrest.com/what-is-3-tier-architecture-and-why-do-you-need-it/>
- Framework web :
<http://static.raibledesigns.com/repository/presentations/ComparingJavaWebFrameworks-ApacheConUS2007.pdf>
- Framework IOC :
<http://stackoverflow.com/questions/2026016/google-guice-vs-picocontainer-for-dependency-injection>