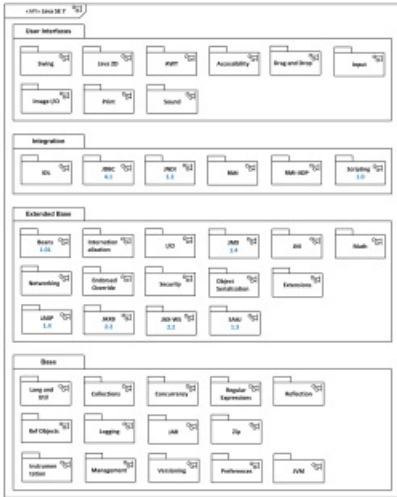


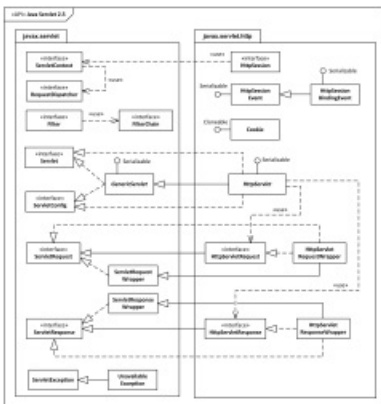
UML Package Diagram Examples



Purpose: An example of UML **package diagram** representing **Java™ Platform Standard Edition (SE) 7 API**.



Purpose: An example of UML **package diagram** representing most important interfaces and classes of **Java™ Servlet 2.5 API**.



Purpose: An example of UML **package diagram** representing most important interfaces and classes of **Java™ Servlet 3.0 API**.

The **javax.servlet.http** is package containing API interfaces and classes specialized for the servlets supporting **HTTP** protocol and corresponding runtime environment. The **javax.servlet.annotation** and **javax.servlet.descriptor** packages allow to declare servlets, filters and listeners by using annotations and to have access to a web application's configuration information.



Summary: Application model shows several layers - presentation layer, services layer, business, data, and cross-cutting layers. All layers are represented as UML **models**.



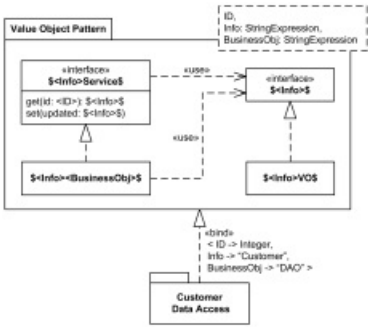
Summary: ***Dependencies** between packages are created in such a way as to avoid circular dependencies between packages. Higher level packages depend on lower level packages. Packages belonging to the same level could depend on each other. Data transfer objects and common exceptions are used by packages at higher levels.*



Purpose: A UML package template example for the design pattern known as Transfer Object in Core J2EE patterns

[CJ2EEP 02] or **Data Transfer Object (DTO)** in **Microsoft .Net Application Architecture Guide [MAAG 09]**. It was also known as **Value Object (VO)** in older version of J2EE design patterns.

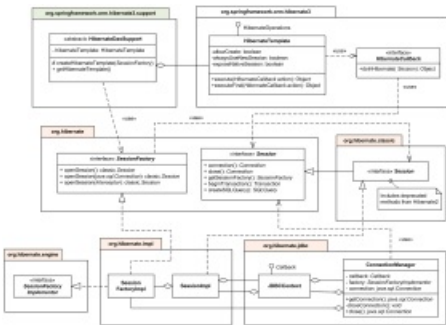
Summary: (Data) Transfer Object is some "coarse-grained" data structure intended to be moved across different boundary layers or transported over the network, usually serializable, so that a single method call could be used to get or to send the whole Transfer Object to reduce the number of remote calls. Examples of DTOs are Customer, Purchase Order, Account, etc.



Spring and Hibernate ORM data access packages and classes

Purpose: An example of UML **package diagram** representing some **Spring** and **Hibernate** data access packages and classes.

Summary: *The Spring Framework provides integration with several Object Relational Mapping (ORM) data access technologies including Hibernate, JDO, Oracle TopLink, iBATIS SQL Maps and JPA. Support packages for object relational mappers comply with Spring's generic transaction and DAO exception hierarchies.*



Noticed a spelling error? Select the text using the mouse and press Ctrl + Enter.



This document describes UML versions up to **UML 2.5** and is based on the corresponding **OMG™ Unified Modeling Language™ (OMG UML®)** specifications. UML diagrams were created in **Microsoft® Visio®** 2007-2016 using **UML 2.x Visio Stencils**. **Lucidchart** is a nice, free UML tool that I recommend for students.

You can send your comments and suggestions to [webmaster](mailto:webmaster@uml-diagrams.org) at **webmaster@uml-diagrams.org**.

Copyright © 2009-2018 uml-diagrams.org. All rights reserved.