

# CDI



CONTEXTS AND DEPENDENCY INJECTION



CDI är ett standard framework sen JEE6 och sköter om livscykeln på komponenterna med hjälp av domän-specifika contexter och injicerade komponenter.

För att ens kunna använda CDI måste man ha en *beans.xml* fil i sin META-INF katalog.

*beans.xml* tvingar applikationen att scanna alla klasser i *bean archive* vid initieringen av filen. Därefter sker mycket automatiskt i form av olika event som skall triggas i bakgrunden.



Det som är speciellt viktigt att veta är att nästan varje klass har en *managed bean* som registreras.

Sen finns det ett par annoteringar man kan använda sig av samt registrera speciella *bean id*.

De vanligaste annoteringarna är `@PostConstruct` och `@PreDestroy`.

`@PostConstruct` är initiering av en bean, medans `@PreDestroy` då är uppstädning.



Mellan *Service/Dao* lagret och *Klient/Rest* klassen annoterar man med *@Inject* som är arbetshästen för CDI. Detta skapar en länk mellan lagrena, likt en vanlig klass initiering, men istället injekteras en böna som håller reda på statusen i objektet på ett typsäkert sätt.

Skillnaden mellan en fabrik och CDI är att CDI kan betraktas som en extremt stor fabrik, även om man kan skriva denna fabrik själv, en så kallad DYDI.



# Fördelar med CDI

- Looser Coupling
- Easier Testing
- Better Layering
- Interface-Based Design
- Dynamic Proxies



För att CDI skall veta var injektionen skall hamna, måste det finnas minst en `@Default` annotering på en klass som *implementerar* alla metoder som Injekteras.

Man kan om man vill kunna välja även annotera med `@Alternative`, vilket är vanligt vid testkörningar (om man inte använder sig av mockning).

En mindre vanlig annotering är `@Named` och används om man vill sätta eget namn på injektions punkten.



Skulle man hamna i problemet med *Ambiguous Injection Point* så kan man skriva *Custom Qualifiers*.

Dessa annoteras med:

*@Qualifier*

*@Retention(RetentionPolicy.RUNTIME) (CLASS, SOURCE)*

*@Target({ElementType.X})*

*public interface test {}*

Inte nog med att man har ett semantiskt namn till sin service, de binder även injektions metadatan.