

# **Programmeringsspråket Java och webbprogrammering**

Kandidatavhandling  
Våren 2012  
Datavetenskap vid Åbo Akademi  
Skribent: John Tran  
Handledare: Annamari Soini

# Referat

Webbprogrammering har en mer komplicerad model jämfört med den klassiska programmeringsmodel. Istället för den vanliga ensamstående program med all dess input och output på samma maskin, består webprogrammerings modellen av tre huvudkomponenter browser, webserver och databas. Webbläsare sköter själva användar interaktion med programmet och tillskillnad från den klassiska modellen behöver inte browsern köras på samma maskin som programmet. Webserver har som uppgift att köra själva programmet och kan försörja olika entitet av samma program till många olika användare samtidigt. Databas fungerar som en data behållare för programmet.

Den här avhandling kommer att utforska huvudsakligen hur webbprogrammering för java språket fungerar och dess uppbyggnad. En mer detaljerat undersökning på vilka extra funktionalitet som har bilvit tillagd till java språket för att bemöta krav för omvandlig till ett webprogrammeringsspråk kräver. Avhandlingen kommer att gå igenom hur java hanterar server, databas och browser modellen.

## Table of Contents

|  |    |
|--|----|
| Referat.....                                     | 2  |
| Inledning.....                                   | 4  |
| Kapitel 1.....                                   | 5  |
| 1.1 Webbprogrammering.....                       | 5  |
| Kapitel 2.....                                   | 6  |
| 2.1 Java Enterprise Edition (J2EE).....          | 6  |
| 2.2 J2EE komponent teknologi.....                | 7  |
| 2.2.1 J2EE skikt.....                            | 8  |
| 2.2.1.1 J2EE klient skikt.....                   | 8  |
| 2.2.2.1 Web komponent i server skiktet.....      | 9  |
| 2.2.2.1 Servlet.....                             | 9  |
| 2.2.2.2 JavaServer Pages (JSP).....              | 9  |
| 2.2.2.3 Enterprise JavaBean (EJB) komponent..... | 10 |
| 2.3 J2EE arkitektur.....                         | 10 |
| 2.4 Sammanfattning.....                          | 11 |
| Kapitel 3.....                                   | 11 |
| 3.1 Slutsats.....                                | 11 |
| Källor.....                                      | 12 |

# Inledning

Industri-applikationer idag övergår allt mer från enkel skikt program arkitektur till många skikt arkitektur oftast i så kallade server-baserad arkitektur. Server-baserad arkitektur är av typen tre skikt arkitektur med en klient-, server- och datadel. Fördelar med en tre eller många skikt arkitektur är separation av presentation, affärslogik och database. Många skikt arkitektur innebär också att det är enklare att byta ut komponenter i ett skikt utan att det skall påverka en annan skikt. Många skikt arkitektur innebär också att det olika skikt måste inte vara fysik på samma maskin utan varje del kan separat köras på sin egen maskin. Kommunikation mellan det olika maskiner vara via nätverk såsom World Wide Web.

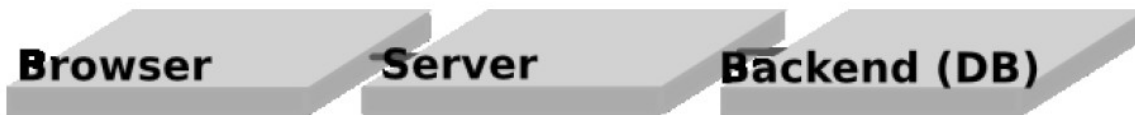
Den här avhandling kommer att utforska huvudsakligen hur webbprogrammering för java språket fungerar och dess uppbyggnad. En mer detaljerat undersökning på vilka extra funktionalitet som har blivit tillagd till java språket för att bemöta krav för omvandlig till ett webbprogrammeringsspråk kräver. Avhandlingen kommer att gå igenom hur java hanterar server, databas och browser modellen.

# Kapitel 1.

Den här kappitel går igenom webbprogrammering dess uppbyggnad och drag.

## 1.1 Webbprogrammering

Webbprogrammering har en mer komplicerad model jämfört den klassiska programmering där ett program med input och output som körs på samma maskin. Den typiska modellen för ett webbsystem består oftast av många skikt, det vanligaste är med tre skikt enligt Figur 1. Varje skikt har ett specialiserat roll. Webbbrowser skiktet hanterar användar interaktion med programmet. Server skiktet har sköter om logiska beräkningar och har direkt kommunikation med webbrowser skiktet och databas skiktet. Databas skiktet hanterar som huvuduppgift att spara och hämta data på ett tillförlitlig sätt.



Figur 1 Tre skikt arkitektur

En stor nackdel med webprogrammodell är att programmerare måste behärska all språk till varje skikt. I Webbbrowser skiktet där användare interagera med själva programmet är programmeringspråket oftast en blandning av HTML och Javascript. Server skiktet är programmerings språket oftast av en generel programmeringsspråk tex Java, perl python eller PHP. Databasskiktet krävs kunskap i relationsdatabas så som SQL. Interaction mellan det tre olika skiktterna kan också leda till problem.

# Kapitel 2

Den kapitel kommer vi att gå igenom Java Enterprise Edition och dess viktigaste komponenter.

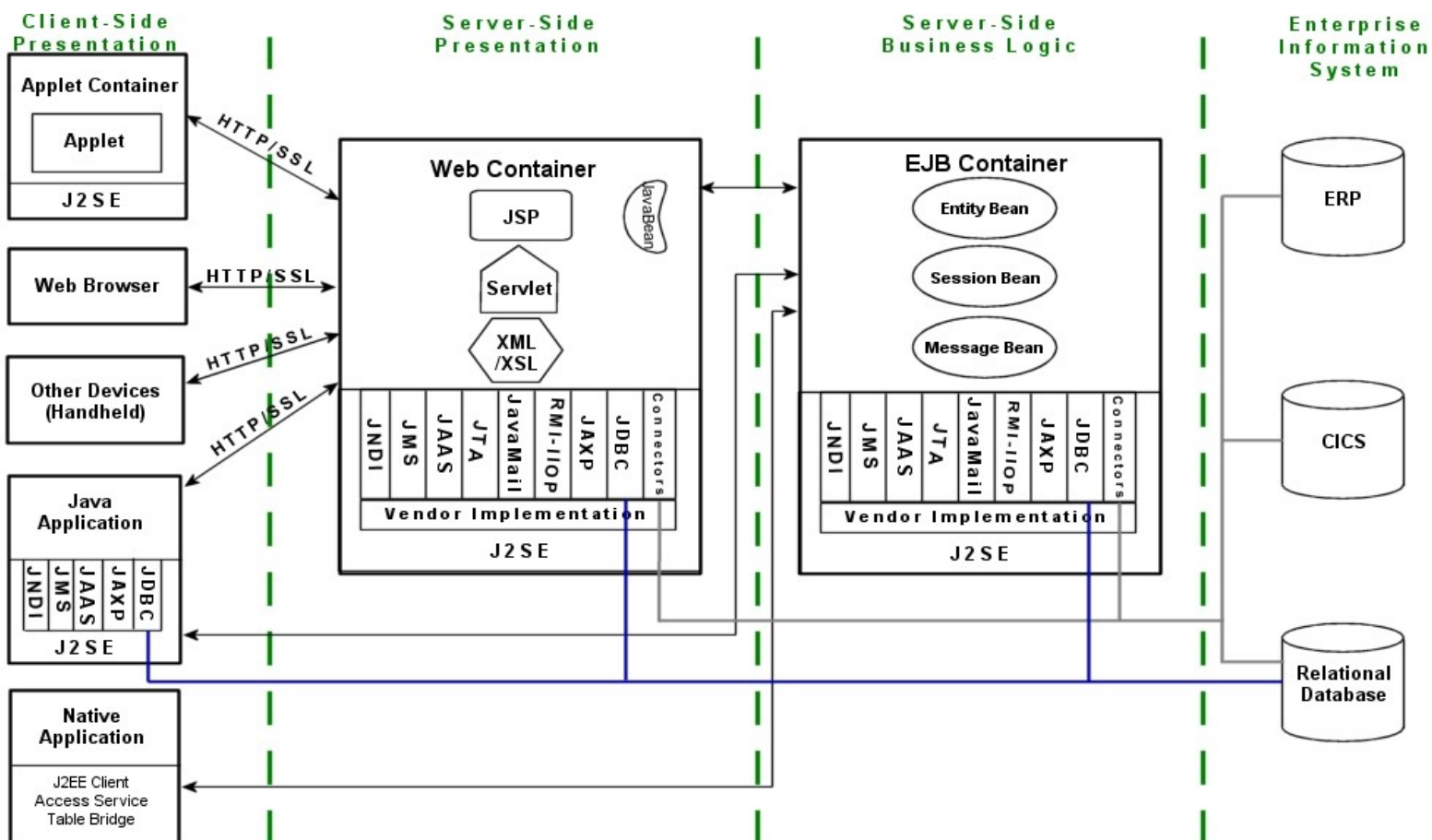
## 2.1 Java Enterprise Edition (J2EE)

Java 2 Enterprise Edition är en Java version från Sun Microsystems som används för utveckling av serverbaserad industri-applikationer. J2EE är en utvidgad version av Java 2 Standard Edition, J2EE tillägger nödvändiga funktioner för utveckling av stabila, säkra och snabba industri-applikationer. De tre viktigaste teknologier som J2EE inför är: komponent, service och kommunikation.

Komponent teknologi används för att skapa de viktigaste delarna i en industri-applikation, nämligen användargränssnitt och business-logik. Komponent teknologi möjliggör utveckling av moduler som kan återanvändas i andra industri-applikationer. J2EE ramverk stöder API för databas åtkomst och industri-informationssystem så som SAP och CICS. J2EE inför också teknologi för kommunikation mellan klient och server.

## 2.2 J2EE komponent teknologi

J2EE definerar en standard för design, utveckling och distribution av många skikt serverbaserad industr-applikation. Många skikt applikation är en applikation som är uppdelat i många komponenter, varje komponent har en specifik funktion. Figur 2 visar arkitektur diagram för J2EE platform och det komponenter som stöds av J2EE platform. De viktigaste komponenter för utveckling av industri-applikation I J2EE är Servlet, JSP och EJB. Alla J2EE komponenter måste exekveras i en system nivå container. Container förser komponenter med service såsom säkerhet, threading och driftsättning.



Figur 2 J2EE arkitektur

## 2.2.1 J2EE skikt

En J2EE applikation är oftast uppbyggd av fyra skikt Klient skikt, Web skikt, affärs skikt och Enterprise Information System (EIS) skikt.

-Klient skikt består av komponenter för presentation av grafisk användargränssnitt och hantering av användarinteraktion. Klient skikt komponenter skall köras på klient maskin.

-Web skikt består av web komponenter som hanterar presentationslogik och service som användarauthentikering. Web skikt komponenter skall köras på server maskin som stöder J2EE standard.

-Affärs skikt består av komponenter som implementerar affärslogiken och affärsdata. Affärs skiktet måste också köras på en server maskin med stöd för J2EE standard. Web skikt och affärs skikt kan köras på server maskin.

-Enterprise Information System skikt (EIS) består av komponenter för hantering av databaser. Det här skiktet skall köras på en eller flera server men separat från web och business skikt.

### 2.2.1.1 J2EE klient skikt

En J2EE-applikation har stöd olika typer av klienter och de kan exekveras på borddator, laptop eller telefoner. J2EE klienter att kommunicera med server komponenter via World Wide Web, trådnätverk eller trådlösnätverk. Klienttyperna är följande:

-Applet är Java-baserad klient komponent som vanligen exkveras i en webläsare och har tillgång till alla funtioner som Java språket erbjuder.

-Applikations klient exekveras i sin egen container. Applikations klient är ett användargränssnitt som tillåter direkt kommunikation med EJB skiktet.

-Trådlös klient är baserad på Mobile Information Device Profile teknologi och används för utveckling av applikationer på trådlösa apparater.



## **2.2.2.1 Web komponent i server skiktet**

J2EE web komponent finns i server skiktet och kan endast exekveras i en J2EE web-container. Web-container förser web komponenter med nätverks service a typen request-response mellan klient och server med HTTP(S) protokollet. Web-container administrerar också livscykel för varje komponent den innehåller. Den förser utöver det standard gränssnitt också service såsom sessions hantering, säkerhets service som autentikering och rättigheter. Web komponent har som huvuduppgift att generera dynamiska användargränssnitt.och websidor åt klient sidan. Det finns två typer av web komponent i J2EE ramverket: servlet och "JavaServer Pages" (JSP).

### **2.2.2.1 Servlet**

Java servlet är den viktigaste komponent i en java web applikation. Den kan endast exekveras i en servlet-container. Servlets huvuduppgift är att generera dynamiska innehåller åt klient komponenter. Web server mappar specifika URL till varje servlet så att när en HTTP-request sker till en URL, körs den servlet som är mappad till den sökta URL. När en servlet tar mot en HTTP-request från klient komponent extraherar den information från HTTP-requesten och med denna information bygger dynamisk upp en lämplig HTTP-response. Vid uppbyggnad av HTTP-response kan servlet också använda sig av affärlogik från EJB container och data från databas skiktet.HTTP-response kan sedan skickas till klient sidan ifrom av HTML eller XML dokument.

### **2.2.2.2 JavaServer Pages (JSP)**

JavaServer Pages JSP används av servlet för genering av dynamiska innehåll åt web klienten. JSP är ett textbaserad dokument som beskriver uppbyggnad av HTTP-response. En JSP sida innehåller

-Mall för konstruktion av web dokument. Den typiska JSP mall innehåller HTML och XML elementer, som kan editeras utan att påverka utseende det dynamisk genererade innehåll.

-JSP elementer och script för generering av dynamisk innehåll för ett web dokument . För det mesta används JavaBeans komponenter för mer komplexa applikationer.

### 2.2.2.3 Enterprise JavaBean (EJB) komponent

EJB är komponent på server sida för implementation av affärslogik och presentation av data objekt. EJB komponenter måste exekveras i en EJB-container som förser EJB komponenter med service såsom överförings kontroll, concurrency kontroll, säkerhet och data persistence. Det finns tre typer av EJB

-Session bean som har som uppgift att utföra affärslogiken eller access databaser. Session bean kan ha två tillstånd: tillståndslös eller med tillstånd. Tillståndslös session bean används vid korta operationer och session bean med tillstånd används vid operationer som kväver bibehållning av tillstånd.

-Entity bean som används för presentation av en eller flera object med persistent data.

-Message-Driven bean är konceptuellt samma som entity bean men kan endast exekveras via meddelandeköer. Web klient kan inte direkt kalla på message-driven bean.

## 2.3 J2EE arkitektur

J2EE applikationer kan ha olika arkitekturtyp beroende på dess design och funktionalitet. två av det vanligaste applikations arkitekturtyp är web-centrerad, EJB-centerad arkitektur. I en web-centrerad applikation har web-container ansvar för alla affärslogik och datahantering utöver det standard innehålls generering och presentation. I en EJB-centerad applikation sköter EJB-container om affärslogik och datahantering. Affärslogik implementeras med EJB session bean för persistent data. J2EE applikation kan också ha en sådan arkitektur där både web-container och EJB-container används. Den här arkitektur används oftast i samband med web applikationer där klient sidan körs på en browser.

## 2.4 Sammanfattning

J2EE ramverk bygger på komponent baserad teknologi (EJB, Servlet och JSP )som är tänkt för att förenkla utveckling av industri-applikationer. J2EE ramverket förse en systemnivå service i så kallade container som tillåter implementation av specialanpassad komponenter för många applikations typer.

## Kapitel 3

Den här kapitel går vi genom för och nackdelar med J2EE ramverket

### 3.1 För och nackdelar med J2EE ramverk

J2EE ramverk har en många skikt arkitektur vilket tillåter separation av prestation, affärslogik och databas. Många skikt arkitektur innebär också att det är enklare att byta ut komponenter i ett skikt utan att det skall påverka en annan skikt. I ett många skikt arkitektur måste inte det olika skikt vara fysik på samma maskin utan varje del kan separat köras på sin egen maskin. Kommunikation mellan det olika maskiner vara via nätverk såsom World Wide Web.

J2EE många skikt arkitektur har också nackdelar, en programmerare som vill utvecklat en applikation med J2EE måste behärska all språk till varje skikt. I Webbrowser skiktet där användare interagera med själva programmet är programmeringspråket oftast en blandning av HTML och Javascript. I Server måste programmeran behärska Java. Databas skiktet krävs kunskap i relationsdatabas så som SQL. Interaction mellan det tre olika skiktterna kan också leda till problem. J2EE passar för utveckling av industri-applikationer men är inte bra för utveckling av små webapplikationer.

## Slutsatser

I detta arbete har jag undersökt vilka ändringar och tillägg som har gjorts till J2SE för att Java programmeringsspråket kan bli ett webbaserat programmeringsspråk. De tillägg som har gjorts till Java språket skall stöda webbaserad utveckling av applikationer är övergång till komponent baserad och många skikt arkitektur.

J2EE är ett bra ramverk för utveckling av industri-applikationer. Ramverket är av typen öppen standard och är gratis till alla användare. J2EE ramverkets är har en många skikt arkitektur vilket tillåter separation av presentation, affärslogik och databas.

J2EE många skikt arkitektur har också nackdelar, en programmerare som vill utvecklat en applikation med J2EE måste behärska all språk till varje skikt. J2EE passar för utveckling av industri-applikationer men är inte bra för utveckling av små webapplikationer. EJB

## Källor

Java EE at a Glance

<http://www.oracle.com/technetwork/java/javaee/overview/index.html> 16 mars 2012

Flanagan, D. (1999). Java™ Foundation Classes in a Nutshell: A Desktop Quick Reference. Sebastopol, Kalifornien, USA: O'Reilly & Associates, Inc.

Wikipedia. Java Platform, Enterprise Edition Hämtat från Wikipedia:

[http://en.wikipedia.org/wiki/Java\\_Platform,\\_Enterprise\\_Edition](http://en.wikipedia.org/wiki/Java_Platform,_Enterprise_Edition) den 12 Februari 2012

Wikipedia. Java Platform, Standard Edition Hämtat från Wikipedia:

[http://en.wikipedia.org/wiki/Java\\_Platform,\\_Standard\\_Edition](http://en.wikipedia.org/wiki/Java_Platform,_Standard_Edition) den 12 Februari 2012

J2EE for managers

<http://www.juddsolutions.com/j2ee.html> 5. mars 2012

Koskenniemi, Saastamoinen, Eerola: Kvalitetskriterier för webbtjänster (2007)

Finansministeriet.fi 29 mars 2012

[http://people.cis.ksu.edu/~hankley/d764/j2ee/Nambiar\\_PetStore.doc](http://people.cis.ksu.edu/~hankley/d764/j2ee/Nambiar_PetStore.doc)

Java Petstore: a case study by Rohit Nambiar 2002

J2EE Development without EJB 2004

Rod Johnson, Wiley Publishing Inc, Indianapolis Indiana