



TDDD86 – Programmeringsparadigm

# Mobila användare och Molnet

## Innehåll

Ståndpunkt .....	3
Bakgrund .....	3
Syfte och frågeställningar .....	3
Metod och material .....	3
Resultat.....	4
Olika typer av moln.....	4
SaaS - Software as a Service .....	4
PaaS - Platform as a Service .....	5
IaaS - Infrastructure as a Service .....	5
GaaS - Gaming as a Service .....	6
Molnlagring.....	7
Påverkan på programmeringsparadigm .....	8
Mathematica.....	8
Diskussion/Analys .....	9
Slutsats .....	9
Sammanfattning .....	10

## Ståndpunkt

*Inom de närmsta fem åren kommer alla användare att ha skaffat lätta mobila enheter och informations- och beräkningsresurser finns i molnet. Detta kommer att kräva en ny approach till programmering, där man ser en tillbakagång till att betona applikationer som använder begränsade mängder beräkningskraft och minne samt återanvändandet av komponenter.*

## Bakgrund

Mer och mer beräkningskraft och lagringsutrymme flyttas ut i molnet. Vi blir alltmer beroende av företagens molntjänster för att få vardagen att gå ihop. Om du inte vet om du molnlagar, så gör du antagligen det. Så fort du tar ett foto med din iPhone eller sparar ett dokument i Word så lagras det nämligen en kopia av filen i molnet. Även våra mobila enheter kräver mer och mer att det finns en server någonstans som kan underlätta dess arbete genom att bidra med beräkningskraft. Allt detta påverkar inte bara hur vi använder våra enheter utan även hur vi utvecklar dem.

## Syfte och frågeställningar

Syftet med rapporten är att undersöka hur molnet har växt fram de senaste åren och att diskutera hur det kan komma att se ut i framtiden. Finns det olika typer av molnbaserad datorkraft och vad är dess för- och nackdelar? Kommer nya programmeringsparadigm dyka upp som utnyttjar den massiva prestandan bättre och får oss att se på molnet på ett nytt sätt? Håller jag med ståndpunkten?

Jag kommer även att försöka utreda varför så många stora företag satsar så mycket pengar på molnet samt hur de använder sig av det.

## Metod och material

För att uppnå syftet och besvara dess frågeställningar så har jag valt att leta information på internet. Detta valde jag eftersom molnet är ett så pass nytt ämne att det inte finns så mycket lättillgänglig utskriven information att få tag på som det finns på webben.

Jag har använt mig av de länkarna som gavs i uppgiftsbeskrivningen, tidigare kunskap som jag bifogat källor på samt vad jag hittat när jag sökt på ämnet på nätet. Jag har även försökt se till att informationen jag hittat är sanningsenlig och i många fall så färsk som möjligt. Källor är markerade som fotnoter av formen "titel, datum, länk", om datum inte hittades så är dessa markerade med dagens datum.

## Resultat

Konceptet med molnbaserad datorkraft är inget nytt. Redan på 50-talet användes stordatorer (eng. mainframes) som användare kunde komma åt via tunna klienter (även kallade terminaler). Terminalerna hade inte själva någon processorkraft utan delade på stordatorns processorkraft med hjälp av tidsdelning. Tidsdelning<sup>1</sup> innebär att operativsystemet ger de olika terminalerna en liten del av processortiden i taget, därav namnet tidsdelning.

På 70-talet kom operativsystemet UNIX som utnyttjade detta system mycket effektivt.

Alla anställda på ett företag behövde därmed inte ha varsin dator, utan väldigt många kunde dela på en stordator, vilket var mer prisvärt och gjorde att man kunde utnyttja prestandan effektivare. Ett företag behövde inte ens äga en egen maskin utan kunde hyra in sig hos ett annat företag.

Systemet med tidsdelning mellan tunna klienter dog dock mer eller mindre ut i och med att datorer blev både kraftfullare och billigare. UNIX lever dock fortfarande kvar till viss mån och är kärnan i Mac OS X samt har inspirerat skapandet av Linux<sup>2</sup>.

Den sortens molnbaserade datorkraft som vi ser nuförtiden har dock en hel del gemensamt med systemen som utvecklades på den tiden.

Det handlar även nu om att "outsourca" hårdvaran till en annan enhet på en annan plats, som ofta ägs av ett annat företag.

## Olika typer av moln

De system som används mest idag för molnbaserad datorkraft är SaaS (Software as a Service), PaaS (Platform as a Service) och IaaS (Infrastructure as a Service). Skillnaden mellan de olika systemen är hur mycket kontroll man har själv<sup>3</sup>. Det finns även andra typer av moln, bl.a. GaaS (Gaming as a Service) och molnlagring.

### SaaS - Software as a Service

Innebär att användaren har tillgång att utnyttja plattformen men kan inte ändra den. SaaS används ofta för att ersätta lokal mjukvara för att istället kunna komma åt den via t.ex. en webbläsare, utan att behöva installera något på användarens dator.

SaaS bidrar även till att mindre piratkopiering då användarna kan tänka sig att betala för tjänsterna eftersom det blir bekvämligare samtidigt som det blir svårare att piratkopiera. Exempel på detta är Spotify, Netflix men även Office 365 som ger användaren tillgång till bl.a. ordbehandlare som inte går att piratkopiera. Många SaaS-tjänster, däribland dessa, ger tillgång via en prenumeration som betalas månadsvis.

Därför går nu företag som Microsoft och Adobe över till molnbaserade tjänster för att minska piratkopieringen och öka intäkterna<sup>5</sup>.

Exempel på SaaS är:

- Gmail
- Office 365

<sup>1</sup> "Time-sharing", den 5 november 2014, <http://en.wikipedia.org/wiki/Time-sharing>

<sup>2</sup> "Apple's Operating System Guru Goes Back to His Roots", den 8 augusti 2013, <http://www.wired.com/2013/08/jordan-hubbard/>

<sup>3</sup> "Cloud computing", den 14 november 2014 [http://en.wikipedia.org/wiki/Cloud\\_computing#Infrastructure\\_as\\_a\\_service\\_.28IaaS.29](http://en.wikipedia.org/wiki/Cloud_computing#Infrastructure_as_a_service_.28IaaS.29)

<sup>4</sup> "SaaS, PaaS, IaaS (Explained and Compared)", den 15 november 2014 <http://apprenda.com/library/paas/iaas-paas-saas-explained-compared/>

<sup>5</sup> "Can Cloud Computing Stop Software Piracy?", den 7 september 2012 <http://cloudtimes.org/2012/09/07/can-cloud-stop-software-piracy/>

### PaaS - Platform as a Service

Innebär att kunden har ansvar för applikationen som ska köras på servern och leverantören sköter bl.a. hårdvaran, operativsystemet och säkerheten. På så sätt ökar utvecklarens produktivitet samtidigt som tjänstens tid till marknaden förkortas.

Exempel på PaaS är:

- Oracle Cloud
- Google App Engine

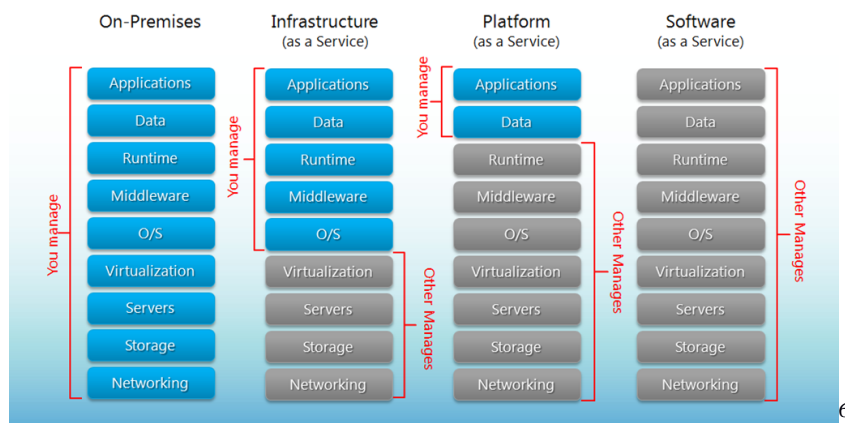
### IaaS - Infrastructure as a Service

Innebär att kunden har ansvar för i princip allt utom hårdvaran. Man får en egen virtuell maskin att installera operativsystem och all övrig mjukvara på. Kunden får även se till att hålla systemet uppdaterat och säkert.

Exempel på IaaS är:

- Amazon EC2

## Separation of Responsibilities



En av fördelarna med dessa tre för företag är dess skalbarhet.

På varje server finns en eller flera virtuella maskiner med egen mjukvara. På så sätt kan flera företag dela på samma fysiska maskin och utnyttja dess prestanda maximalt. Om en tjänst behöver mer kraft så skalas systemet upp genom att starta fler virtuella maskiner på en annan server. På så sätt kan tjänster som t.ex. har mer trafik dagtid än nattetid skala upp och ner efter behov.

Företaget betalar per maskintimme, så det kostar alltså lika mycket att hyra 1000 virtuella maskiner i en timme som att hyra en i 1000 timmar. Detta gör att företaget slipper vänta onödig tid när det ändå är samma pris.

<sup>6</sup> " SaaS, PaaS, and IaaS.. Oh my! ("Cloudy April" - Part 3)", Kevin Remde, den 3 april 2011  
<http://blogs.technet.com/b/kevinremde/archive/2011/04/03/saas-paas-and-iaas-oh-my-quot-cloudy-april-quot-part-3.aspx>

### GaaS - Gaming as a Service

Molnet är inte bara för företag. På CES-mässan 2013 visade Nvidias Jen-Hsun Huang upp en ny storsatsning inom spelbranschen: Nvidia Grid<sup>78</sup>.

Anledningen till att spelindustrin hittills hamnat utanför molnet är för att spel kräver mycket bättre prestanda än vad t.ex. strömmande av film gör, säger Jen-Hsun. När man kollar på en film på t.ex. Netflix så kan flera bildrutor skickas från servern på en gång, då det är förbestämt i filen på servern hur varje bildruta ska se ut.

Om man däremot ska strömma spel så måste servern räkna ut hur varje bildruta ska se ut beroende på användarens interaktion. Så fort användaren trycker på en tangent eller rör på musen så måste denna information skickas till servern så att den kan räkna ut hur bilden ska se ut, för att sedan skicka denna till användaren.

För att få detta att fungera så måste GPU:n inte bara kunna rendera en bild på en bildskärm som är ansluten med en vanlig HDMI-kabel, utan även på en bildskärm som kan sitta flera hundra mil bort utan märkbar fördröjning. Förutom de extremt höga kraven som detta ställer på hårdvara och uppkoppling så ställer det även krav på mjukvaran. Om man ska kunna skicka de renderade bildrutorna utan fördröjning så pass långt så krävs en bra komprimeringsteknik för att göra det möjligt. Nvidia Grid fungerar i övrigt rätt likt stordatorerna med tunna klienter som jag nämnt tidigare.

Nvidias VD påstår att de, efter fem års utveckling, har löst problemen. Detta skulle innebära att användare slipper köpa dyra spelkonsoller och behöver istället bara en tunn klient som har stöd för Nvidia Grid. De behöver inte heller lägga tid på att installera spelen då de finns förinstallerade och blir åtkomliga via alla användarens kompatibla enheter, såsom mobil, bärbar dator eller smart-tv.

Nu nästan två år senare så har man dock hört ytterst lite om Nvidia Grid. Om det beror på bristande intresse eller om tekniken inte är redo för världen ännu är oklart.

<sup>7</sup> "THE POWER OF CLOUD GAMING", den 15 november 2014, <http://www.nvidia.com/object/cloud-gaming.html>

<sup>8</sup> "NVIDIA Press Conference - NVIDIA GRID - at CES 2013 (Part 2)", den 7 januari 2013, <https://www.youtube.com/watch?v=J0Rm0OgbSIM>

## Molnlagring

Molnlagring skiljer sig från de tidigare nämnda tjänsterna. Istället för att erbjuda molnbaserad datorkraft i form av processorkraft, så handlar molnlagring om att erbjuda molnbaserat lagringsutrymme. Här får användaren ladda upp sina filer på företagets servrar, vilket kan vara fördelaktigt bl.a. dessa anledningar<sup>9</sup>:

- **Säkerhetskopiering** av filer innebär att användaren kan spara sina värdefulla filer på flera ställen, vilket är otroligt viktigt om ens egen hårddisk skulle krascha.
- **Extra lagringsutrymme** kan behövas om ens egen hårddisk är full, även om en extern hårddisk generellt är snabbare.
- **Synkronisering och fjärråtkomst** är något som många molnlagringstjänster erbjuder. Detta innebär att man installerar en mjukvara på sina enheter för att synkronisera filerna mellan dem och därmed göra dem åtkomliga från samtliga enheter.

Det kan även finnas möjliga problem med att lämna ut sina filer, däribland säkerheten och pålitligheten med en molnlagringstjänst.

Exempel på molnlagringstjänster:

- Dropbox
- Google Drive
- Onedrive
- iCloud

<sup>9</sup> "Top ten advantages of using online storage services", den 15 november 2014 <http://online-storage-service-review.toptenreviews.com/top-ten-advantages-of-using-online-storage-services.html>

## Påverkan på programmeringsparadigm

När det gäller att programmera molntjänster så kan man oftast använda de vanligaste språken. Om man t.ex. kollar på Amazons EC2 så stödjer deras API: Java, PHP, Python, .NET och Ruby<sup>10</sup>. För att kunna se något i webbläsaren så måste det alltid på något sätt genereras HTML-kod, dock finns det även en del språk som på något sätt är skapade specifikt för molnet, ett av dem är Mathematica.

## Mathematica

Stephen Wolfram och hans företag Wolfram Research, presenterade 2013 deras programmeringsspråk Wolfram Language, ofta kallat Mathematica som bl.a. använts för att skapa Wolfram Alpha. Mathematica är ett kunskapsbaserat programmeringsspråk. Detta innebär att det till skillnad från icke kunskapsbaserade språk har en databas av data och algoritmer. Språket kan t.ex. säga vad Spanien har för BNP eller hur många neutroner ett speciellt grundämne har. Detta är möjligt genom molnbaserad datorkraft. Språket använder sig alltså av externa källor, däribland statliga filer och vetenskapslabb som bidrar med information och algoritmer som räknar ut resultaten åt dig<sup>1112</sup>.

<sup>10</sup> "AWS Support for Programming Languages", den 15 november 2014, <http://docs.aws.amazon.com/AmazonCloudFront/latest/APIReference/AWSLibraries.html>

<sup>11</sup> "Where does Wolfram Alpha get its information from?", den 21 maj 2009, <http://www.theguardian.com/technology/2009/may/21/1>

<sup>12</sup> "Wolfram Language", den 15 november 2014 <http://www.wolfram.com/language/>



## Diskussion/Analys

Det råder ingen tvekan om att de flesta stora mjukvaruföretag vill utnyttja molnbaserad datorkraft. Varför det har kommit först nu kan jag inte svara på helt säkert, men jag antar att ett snabbare internet är en av huvudfaktorerna som man väntat på.

Eftersom företag går över till molnbaserade tjänster så behöver de inte lägga energi på att göra sin mjukvara svårare att piratkopiera. De får även väldigt stor kontroll över hur användarna använder deras tjänster, något som jag tror företagen uppskattar.

Man kan helt enkelt säga att de flesta mjukvaruföretag kan tjäna på att göra sina tjänster molnbaserade. Ett system där användaren har en mobil enhet som fungerar som en tunn klient och där företaget sedan kör beräkningarna på sina servrar i molnet, ligger nog i deras intresse.

Frågan är sedan vad kunden har att vinna på att köra molnbaserade tjänster.

Fördelarna för kunden är ju generellt att de kommer åt tjänsterna från alla sina enheter, de slipper installera mjukvara på sina egna datorer och de behöver inte heller lika kraftfulla datorer för att kunna köra tjänsterna.

Sedan finns det även några nackdelar. Först och främst så måste användaren vara uppkopplad för att huvud taget kunna komma åt tjänsten. Sen är det säkerheten, vågar ett företag skriva känsliga uppgifter i ett dokument i t.ex. Office 365? Hur mycket granskar Microsoft filerna som ligger på deras servrar? Säljer de informationen vidare till andra företag?

Personligen så föredrar jag fortfarande att ha min mjukvara installerad på min egen dator. Kanske för att jag är lite av ett kontrollfreak och vill sköta saker själv, kanske för att jag inte vill lita blint på företagen som erbjuder molntjänsterna. Den enda molntjänsten som jag faktiskt använder är Dropbox för säkerhetskopiering av mina allra viktigaste filer. Jag har även en NAS där jag lagrar dem, men en kopia på en annan plats skadar inte. När det kommer till ordbehandlare, som kanske är den mest använda typen av molntjänst, så föredrar jag Microsoft Word installerat på min egen dator. Därmed kan jag installera uppdateringar när och om jag vill, oavsett vad Microsoft tycker.

När det gäller programmeringsbiten så har jag innerst inne en liten fobi för att skriva program i språk som är skapade innan internets uppkomst. Visst, ett plus ett blir alltid två i alla programmeringsspråk, men med tanke på hur stor del internet är av den digitala världen så känns det lite utdaterat att använda så gamla språk i bl.a. molntjänster. Då ryker tyvärr både C och C++ som är två av de mest använda språken just nu, även om båda är uppdaterade efter internets uppkomst.

## Slutsats

Håller jag då med ståndpunkten? Jag tror att vi går mot mer och mer mobila enheter, men jag tror inte att de kommer bli svagare prestandamässigt bara för att det finns beräkningskraft i molnet.

Tvärtom tror jag att de mobila enheterna kommer att bli bättre och bättre och därmed kunna utföra mycket av arbetet själva, därmed håller jag inte med ståndpunkten.

Molnlagringen tror och hoppas jag dock är kvar för att stanna. Även om våra mobila enheter kommer vara kraftfulla nog att utföra all beräkningskraft själva så tror jag att alla kan tjäna på att ha säkerhetskopior lite här och där. Dock skulle jag vilja se en fokus på säkerhet där bekvämlighet kommer på andraplats. Kanske t.o.m. att man krypterar sina filer själv innan man laddar upp dem i molnet.

## Sammanfattning

Som du som läsare nu kanske har förstått så är det mycket som händer inom området. Man kan säga att vi befinner oss i en övergångsfas där företag håller på med att byta till molnbaserade lösningar av deras tjänster. Även då tekniken har kommit långt så finns det nog mycket att se fram emot under kommande år. Frågan är bara: är molnet här för att stanna, eller kommer det blåsa bort efter några år?

Om du vill veta mer så rekommenderar jag dig att följa de länkar som finns bifogade i dokumentet, där finns många timmars intressant läsning.