

Jämförelse av mjukvara för användning av gridsystem

Christian Bogel
Kandidatavhandling i datavetenskap
Handledare: Marina Waldén
Institutionen för Informationsteknologi
Åbo Akademi 2015

Innehåll

1. Inledning
2. Griddsystem
 - 2.1. Griddsystem jämfört med andra distribuerade system
 - 2.2. Historia
3. Griddsystemens arkitektur
 - 3.1. Arkitekturens struktur
 - 3.2. Fysiska nivå
 - 3.3. Anslutningsnivå
 - 3.4. Resursnivå
 - 3.5. Kollektivnivå
 - 3.6. Applikationsnivå
4. Mjukvara för användning av griddsystem
 - 4.1. Globus Toolkit
 - 4.2. BOINC
 - 4.3. Jämförelse mellan Globus Toolkit och BOINC
5. Sammanfattning
6. Källhänvisning

Referat

Ordlista

ARPANET

CVS

Distribuerade system

Ethernet

Mellanvara

RPC

TCP/IP

1. Inledning

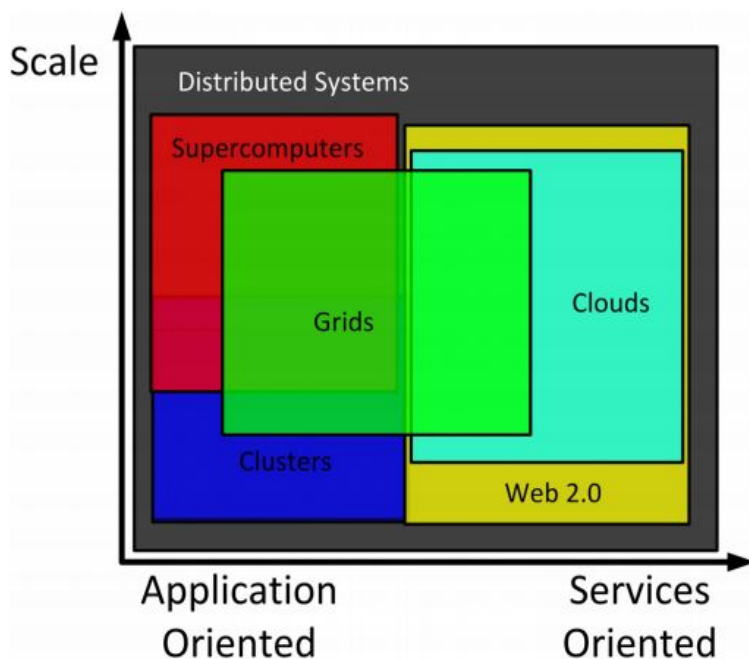
Gridsystem (eng. Grid Computing) har blivit en viktig del av våra dagliga liv. Flera av funktionerna som relaterar till datorer och Internet använder sig av detta system. Användning av sociala nätverk så som Facebook och Twitter är byggt på principerna av gridsystem. Att dela och skriva dokument på Google Docs är en annan funktion vilket är möjligt genom detta system. Websidor var man kan hantera e-post är ett annat exempel på vad man kan åstadkomma genom gridsystem. Dessa är bara några exempel på funktioner som inte kunde existera utan gridsystem. Detta system åstadkommer detta genom att använda idéerna bakom distribuerad system. Distribuerad systemets funktion är att använda flera olika datorers resurser för att åstadkomma ett större mål än vad enbart en dator kunde åstadkomma. Oftast används detta för att hantera stora beräkningar. Gridsystem bygger på detta genom att göra datorernas fysiska läge till en mindre faktor. Genom systemet kan olika universitet runt om världen dela deras datorresurser sinsemellan och åstadkomma även större prestationer än vad de kunde åstadkomma utan. Gridsystemens funktion är att koordinera och möjliggöra olika datorresurser att fungera som ett enhetligt system.

Syftet med denna avhandling är att gå igenom hur ett gridsystem i principen är uppbyggt och ge exempel på hur vissa applikationer använder sig av systemet. Gridsystemets arkitektur är vad som huvudsakligt kommer att gå igenom i denna text. För att bättre kunna förstå arkitekturen beskrivs först de grundläggande principerna bakom gridsystem, systemets utveckling och jämförelse kring andra distribuerad system. Slutligen ges en jämförelse mellan två olika system som använder sig av gridsystem och hur de skiljer sig från varandra. Avhandlingen kommer inte att gå igenom hur man bygger upp en gridsystem, utan tyngdpunkten ligger på vilka funktioner och protokoll som behövs för en detta system.

Det slutliga frågeställningen som avhandling ska svara på är hur olika applikationer använder sig av systemet. Hur de olika applikationer mål skiljer sig från varandra och om eller hur deras implementation av gridsystem skiljer sig från varandra är andra frågor som avhandling ska försöka svara på.

2. Griddsystem

Griddsystem är en form av distribuerade system vars funktion är att möjliggöra beräkningar med hög prestanda och resursdelandet mellan flera datorer. Grunden till griddsystem och andra liknande system är distribuerade system (se figur 1). Distribuerade systemens grundidé är att datorer, vilka är ihopkopplade i ett nätverk, kan koordinera och kommunicera sina handlingar sinsemellan genom att enbart skicka meddelanden [GC-DSCD:s. 20]. Detta gör det möjligt att använda resurser av flera olika datorer för att utföra stora beräkningar på några sekunder vilket skulle ta flera år för en dator att beräkna. Griddsystem bygger på detta genom att möjliggöra denna funktion i mer decentraliserad form. Detta innebär att datorer inte behöver vara lokaliserade i samma område eller nätverk. Istället kan griddsystem utföra distribuerade systemens huvudsakliga funktion genom anslutning till ett bredare nätverk det vill säga Internet.



Figur 1

2.1. Griddsystem jämfört med andra distribuerad system

Figur 1 demonstrerar hur flera olika system är uppbyggd från distribuerade system. För att bättre förstå hur griddsystem fungerar så måste man först förstå hur det skiljer sig från andra distribuerad

system. Gridsystem och klustersystem (cluster computing) har flera liknande egenskaper. Båda systemen involverar flera ihopkopplade datorer vars funktion är att använda datorernas resurser till olika ändamål. Skillnaden ligger i att klustersystem är uppbyggd av flera sammankopplade datorer som fungerar mera som en enhetlig entitet. Gridsystem syfte är att förverkliga en mer decentraliserad version av klustersystem. Principen bakom gridsystem är att flera olika distribuerade system kan dela med sina resurser sinsemellan. Vad detta innebär är att datorerna inte behöver vara belägna på en centraliserad område så som i en klustersystem, utan datorernas placering kan vara utsprid på olika områden runtom världen. Detta leder till en system var av flera olika organisationer kan samarbeta och dela med sig sina resurser sinsemellan.

[BW-GCTA:s. 10-11]

Molnbaserade datortjänster är i princip uppbyggt som en form av gridsystem. Både molnbaserade datortjänster och gridsystem använder sig av Internet för att göra resurser tillgängliga. Skillnaden blir huruvida man använder dessa resurser och till vilken ändamål. Molnbaserad datortjänst är enligt Barry Wilkinson (s.11) en kommersialisering av gridsystem. Vad detta innebär är att molnbaserad datortjänst använder tekniken bakom gridsystem för att skapa en tjänst vars huvudsakliga syfte är att skapa ekonomisk vinst för en företag. Jämfört med gridsystem vars primära syfte är att möjliggöra samarbete och delande av resurser mellan olika entiteter.

[BW-GCTA:s. 10-11]

//Webb 2.0 och supercomputing

2.2. Historia

På början av 1990-talet började gridsystem bli en genomförbart system tack vare utvecklingen av klustersystem, hög kapacitet lokala och globala nätverk. Genom utvecklingen av Internet och bättre nätverk kunde principerna bakom klustersystem möjliggöra utvecklingen och implementation av gridsystem. Det fanns också andra utvecklingar som var viktiga milstolpar vars utveckling krävdes för att implementera gridsystem. Parallellprogrammering är en av dessa milstolpe. Datorer började med endast en processor, men genom tiderna började konceptet uppstå att man kunde öka datorns prestation och hastighet genom att ha flera processorer i en dator. För att kunna effektivt använda och hantera flera processorer krävdes ett nytt sätt att hantera processorerna. Denna nya sätt heter parallellprogrammering vilket utvecklades vidare till att inkludera flera ihopkopplade datorer vars resurser är koordinerat att utföra varierande beräkningar. På 1970-talet börjades utvecklingen av en teknologi vilket möjliggjorde bättre sammankoppling av datorer. Detta kallades Ethernet. Ethernet är en samling av teknologi och standarder vars huvudsakliga funktioner är att

hantera kollisioner i nätverkskommunikation, fungera som identifikator inom nätverk och att hantera överföring av data inom en nätverk på ett pålitligt sätt.

För att möjliggöra griddsystemens funktion att dela resurser mellan olika entiteter vars geografiska placeringar är utspritt krävdes utveckling av hög kapacitet nätverk. En av de första formen av sammankopplade nätverk var i form av ARPANET. Utvecklingen av ARPANET (Advanced Research Projects Agency Network) började på 1960-talet och finansierades av DARPA (Department Advanced Research Projects Agency) vars syfte var att skapa en paketförmedlande nätverk mellan fyra universitet i USA. ARPANET var ett lyckat koncept vars framgång även var orsaken till att TCP/IP (Transmission Control Protocol/Inter Protocol) blev en standard för nätverks hantering i dagens värld. TCP protokollens syfte är att garantera pålitlig kommunikation medan IP protokollens huvudsakliga uppgift är dirigerering av kommunikation i en nätverk. TCP/IP är ett viktigt protokoll vilket används även idag och är ett av de grundläggande protokoll som Internet är uppbyggt av.

En av de viktigaste funktionerna i ett griddsystem är användning av distribuerade resurser. För att möjliggöra användning av distribuerade resurser skapades i mitten av 1980-talet ett program som heter RPC (remote procedure call). RPC fungerar genom att skicka instruktioner till den avlägsna datorn så att den tar emot instruktioner som om att de hade uppstått lokalt på datorn. Principerna bakom RPC utvecklades till objektorienterad versioner så som COBRA (Common Request Broker Architecture) och Java RMI (Java Remote Method Invocation).

[BW-GCTA:s. 1-6]

Virtualisering var en viktig utveckling för att möjliggöra griddsystemets förmåga att utnyttja flera olika datorers resurser i en mer sammanhängande form. Detta uppstod först i form av virtuella maskiner, vars förmåga var möjligheten att hantera flera operativsystem i en dator [BHCC-MDN]. Genom virtualisering av maskiner kan griddsystem göra mellanvara mellan klienten och resurserna fungera som en virtuell dator som kan sköta om klientens beräkningar. Andra utvecklingen inom virtualisering som är viktigt för griddsystem relaterar till nätverk. Virtuella privat nätverk (VPN) funktion är att åstadkomma privata nätverkens egenskaper genom öppna nätverk. Egenskaper så som säkerhet och funktionalitet är viktigt för griddsystem och genom att använda virtuella privat nätverkens kan griddsystem replikera dessa egenskaper.

3. Gridsystemens arkitektur

För att en gridssystem kan fungera effektivt så måste det finnas en grundläggande struktur som kan bemöta problem som kan uppstå. Problem så som hur rättigheterna hanteras är avgörande när ett system används av flera olika användare och institutioner. Detta kapitel huvudsikt är att gå igenom gridsystemens arkitektur, det vill säga vilka komponenter, regler och system som gridssystem behöver för att fungera.

Så att en entitet har möjligheten att effektivt hantera en gridsystems resurser krävs det att det finns regler för vem, när och hur resurserna används. En gridssystem används oftast av flera olika entitet så som universitet eller liknande institutioner. Dessa entitet har oftast varierande uppbyggnad av användare med olika rättigheter dessutom kan strukturen mellan olika organisationer variera enormt. För att göra det möjligt att flera olika organisationer kan använda sig av samma gridssystem behövs det en system som kan hantera rättigheterna mellan olika entitet. För gridssystem uppstår detta i form av Virtuella Organisation.

[FB-GC:s.174-176]

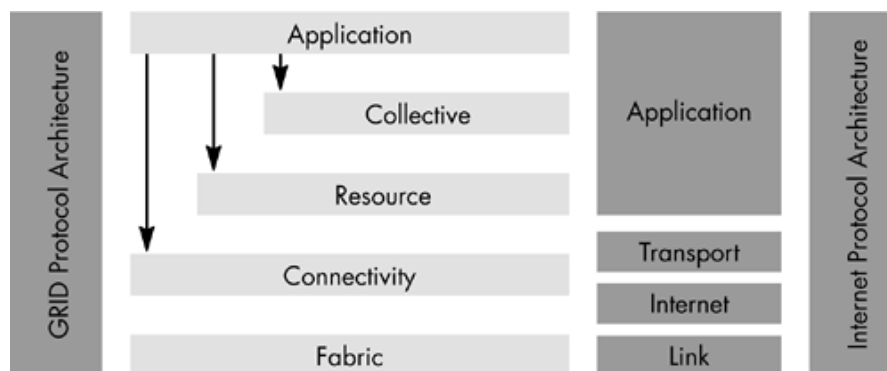
Virtuella organisationers funktion är att hantera vilka resurser en organisation vill tillåta användarna av en gridssystem att använda. Vad detta innebär är att Virtuella Organisationer är uppbyggda av en mängd regler för vem som har rättigheter att använda resurserna som existerar i gridsystemet. Resurserna som delas i en gridssystem uppstår inte endast av filer utan det är även direkt tillgång till datorer, data och programvara. Därför är det viktigt att det finns regler för vem som har rättigheter att komma åt dessa resurser.

[FB-GC:s.172]

Den huvudsakliga uppbyggnaden av gridsystemens arkitektur är i form av protokoll. Protokoll i denna sammanhang är regler hur kommunikation sker mellan datorer, datorprogram och nätverksnoder. *Dessutom definieras protokoll hur olika element i en gridssystem påverkar varandra för att åstadkomma en viss effekt. Men gridsystemens arkitektur är också uppbyggt av

komponenter och funktioner som behövs för att griddsystem kan fungera effektivt. Griddsystemens arkitektur är uppbyggt av fem olika nivåer vilket reflekterar olika komponenter, funktioner och protokoll som dessa olika nivåer hanterar. Figur 2 visar hur griddsystemens olika nivåerna är uppbyggda och det fungerar även som en abstraktion över hur dessa nivåer är bundna till varandra. De fem olika nivåerna är den fysiska nivån, anslutningsnivån, resursnivån, kollektivnivån och applikationsnivån.

[FB-GC:s.176-178]



Figur 2

3.1. Fysiska nivå

Den fysiska nivån hanterar fysiska resurserna vilka griddsystemen hanterar som dator- och nätverksresurser, lagringssystem, kataloger och sensorer. Resurserna kan även vara en logiskt entitet, det vill säga resurser vars entitet består av klustersystem, distribuerade filsystem eller förekommer i form av distribuerade datorer. Denna nivå sköter om resurs specifika operationer som uppstår genom användning av griddsystem. Den fysiska nivån sköter även om flera olika mekanismer relaterade till de fysiska resurserna. Dessa mekanismer sköter om olika aspekter av resurserna som den fysiska nivån hanterar.

För att hantera beräkningsresurserna i griddsystemet krävs det en mängd mekanismer. En av dessa är relaterad till påbörjande av program. Denna mekanisms funktion är att övervaka och hantera processerna som programmen kommer att utföra. Det är även viktigt att det existerar övervakningsmekanismer för resurserna som dessa processer kommer att använda. När det gäller lagrings resurserna är de viktigaste funktioner relaterade till placering och hämtning av filer. Exempel på funktioner som dessa mekanismer hanterar relaterar kring läsandet och skrivandet av olika delmängder av filer eller mekanismer som möjliggör kontroll över resurserna som är allokerad

till data överföring.

För nätverksresurser är mekanismer som möjliggör hantering av resurser som är allokerad till nätverksöverföring viktiga. Mekanismer som möjliggör hantering av olika versioner av kod och objekt är en viktig funktion för datakataloger. För detta ändamål brukar man använda versionskontroll systemet CVS (Concurrent Versions System).

[FB-GC:s.178-180]

3.2 Anslutningsnivå

Anslutningsnivån innehåller all grundläggande kommunikations- och autentiseringsprotokoll vilket behövs för att möjliggöra nätverkstransaktion i griddsystem. Kommunikationsprotokoll hjälper till med att möjliggöra data överföring mellan fysiska nivåns olika resurser. Autentiseringsprotokollens funktion är däremot att säkerställa resurserna genom säker kryptografisk verifiering av användarna och fysiska nivåns resurser. För att användarna skall effektivt ha möjligheten att använda sig av griddsystemens olika resurser krävs det att det existerar endast en autentiseringsprocess. Detta sköts med "Single Sign-On" protokoll. Detta innebär att användarna behöver endast skriva in sig (login) en gång i systemet för att ha tillgång till griddsystemets resurser. Detta behövs till exempel när olika institutioner använder sig av samma griddsystem.

Delegationsmöjlighet är en annan viktig del för denna nivå av griddarkitektur. Vad detta innebär för en användare är att det finns möjligheter till att auktorisera program med samma rättigheter som användaren, så att dessa program kan hantera griddsystemens resurser. Vad detta innebär är att en program som en användare har auktoriserat kan hantera resurserna i en griddsystem autonomt.

En av de viktigare funktioner som denna nivå hanterar relaterar kring säkerhetssystem. På grund av att flera olika resurser och webbsidor använder egna säkerhetssystem så är det viktigt att griddsystemets säkerhetssystem kan användas parallellt med de olika säkerhetssystem. Inom denna nivå finns det flera protokoll vars funktion är undvika konflikter som kan uppstå på grund av de varierande säkerhetssystem som finns i en griddsystem. Men i och med att griddsystem är oftast uppbyggt av flera olika entiteter, som delar med sina resurser sinsemellan, så måste det finnas öppenhet mellan dessa olika entiteter när det gäller säkerhetssystem. Det vill säga användaren av griddsystem måste enkelt ha möjligheten att använda de olika resurserna. Griddsystemets säkerhetssystem hanterar användarens säkerhetsrättigheter så att entiteterna vars resurser används inom griddsystemet inte behöver hantera individuellt alla rättighetsförfrågningar.

[FB-GC:s.180-181]

3.3 Resursnivå

Resursnivån använder sig av anslutningsnivån olika kommunikations- och autentiseringsprotokoll för att säkerställa, möjliggöra kontroll och övervakning av griddsystemens individuella resurser. Vad detta innebär är att denna nivåns huvudsakliga funktion är att enbart hantera individuella resurser i en griddsystem. Protokoll som uppstår i denna nivå har funktioner som gör det möjligt att hantera individuella resurser. Informationsprotokoll är en av dessa protokoll och det används för att ge information om griddsystemens resurser. Information om resursers kapacitet och konfiguration är sådant som dessa protokoll sköter om. Även regler kring hur dessa resurser skall användas existerar i denna nivå.

En av de viktigare protokoll för anslutningsnivån är förvaltning protokoll. Vad dessa protokoll hanterar är tillgång och specifikationskrav för griddsystemens resurser. Eftersom flesta resurser i en griddsystem är tillgängliga för flera användare så måste det finnas regler hur dessa resurser får användas. Förvaltning protokoll funktion blir att se till att dessa regler kommer att följas.

[FB-GC:s.181-182]

3.4 Kollektivnivå

Kollektivnivån innehåller protokoll och tjänster vars funktion är hantering av interaktion mellan olika samlingar av resurser. Protokoll och tjänster i denna nivå hanterar helheten av resurserna som finns tillgängligt i griddsystemet. Tjänster så som fil katalog tjänster (directory services) är ett bra exempel på detta. Denna tjänst ger användarna som är med i en virtuell organisation möjligheten att söka igenom griddsystemet för specifika resurser. Dessa resurser kan sökas genom till exempel namn eller attribut. Fil katalog tjänster är väsentliga för att göra det möjligt för effektiv användning av griddsystem.

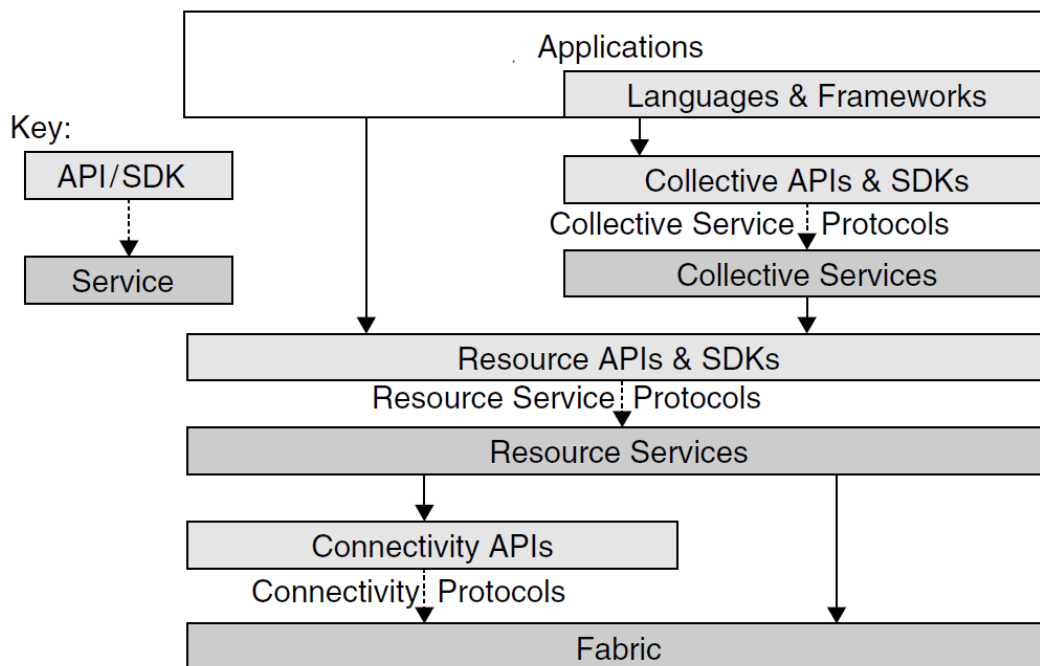
Det uppstår även en stor mängd av tjänster och protokoll i denna nivå som hanterar problem som kan uppstå i griddsystem. Protokoll och tjänster som hanterar allokering och tidsplanering av resurser är mycket viktiga för griddsystem på grund resursernas tillgänglighet i systemet. Dessutom måste det existera tjänster för att övervaka fel i resurser eller attack mot resurserna. Protokoll och tjänster i denna nivå har en mycket mer utbrett ansvarsområden än de andra nivåer.

[FB-GC:s.182-184]

3.5 Applikationsnivå

Den femte och sista nivån av griddsystemens arkitektur är applikationsnivån. Applikationsnivån är applikationer vilka är uppbyggt av andra nivåers tjänster och protokoll. Denna nivå har applikationer som användare utnyttjar för att åstadkomma griddsystemens olika funktioner. Applikationsnivåns funktion är att anropa olika tjänsterna som uppstår i andra nivåerna av en griddsystem och figur 3 visualiserar hur applikationsnivån anropar dessa olika tjänster. Dessa applikationer uppstår i olika former beroende på vilken syfte de används för.

[FB-GC:s.185]



Figur 3

4. Mjukvara för användning av griddsystem

I denna del av avhandlingen så kommer vi gå igenom två olika applikationer vars funktion är att ge användare tillgång till griddsystemens resurser. Även om de två olika applikationer, det vill säga Globus Toolkit och BOINC, använder sig av liknande principer för bygga upp sin system så skiljer dom sig genom vilka funktioner som applikationerna kräver av griddsystemet.

4.1. Globus Toolkit

Globus Toolkit är en samling av mjukvara vilket används för att skapa griddsystem och griddapplikationer. Mjukvara samlingen innehåller en bibliotek av tjänster och mjukvara vars syfte är att adressera problem som måste hanteras i griddsystem så som säkerhet och ankomst till information. Samlingen fungera som en verktyg för att enklare skapa och hantera griddsystem. Globus Toolkit är även uppbyggt i principen av öppen källkod. Vad detta innebär är att vem som helst kan inspektera, modifiera och ändra på källkoden av Globus Toolkit.

[FB-GC:s.222-224]

Grunderna av Globus Toolkit började från 1994 med I-WAY.

[GT]

4.2. BOINC

BOINC eller Berkleys Open Infrastructure for Network Computing är en öppen källkod variant av en mellanvara som används för att stöda griddsystem projekt vars resurser kommer från volontär datorer. Vad detta innebär är att resurserna som griddsystemen hanterar kommer huvudsakligen från datorer som frivilligt låter projektet använda datorns resurser för beräkningar.

4.3. Jämförelse mellan Globus Toolkit och BOINC

Den stora

5. Sammanfattning

6. Källhänvisning

Källor:

- Grid computing: Techniques and applications – Barry Wilkinson [BW-GCTA]
- Grid computing : making the global infrastructure a reality - Fran Berman, Geoffrey Fox, Tony Hey [FB-GC]
- Distributed Systems: Concepts and Design - George Coulouris, Jean Dollimore, Tim Kindberg, Gordon Blair [GC-DSCD]
- A brief history of cloud computing – [Maximiliano Destefani Neto](#), March 18, 2014 [BHCC-MDN] - <http://www.thoughtsoncloud.com/2014/03/a-brief-history-of-cloud-computing/>
- Globus Toolkit [GT]
<http://toolkit.globus.org/toolkit/about.html>

*/*Källor som inte finns med i texten*/*

- Grid 2: Blueprint for a new computing infrastructure – Ian Foster and Carl Kesselman
- Comparison of Grid Computing vs. Cluster Computing
http://www.jatit.org/research/introduction_grid_computing.htm
- User Demand Prediction-Based Resource Management Model in Grid Computing Environment
<http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=4622896&queryText=%3Dgrid+computing+resource+management>
- Study on Computing Grid Distributed Middleware and Its Application
<http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=5232155&queryText=%3Dgrid+computing>