

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Dejan Ambruš

Visoko zmogljivo računanje v oblaku
Windows Azure

DIPLOMSKO DELO

VISOKOŠOLSKI STROKOVNI ŠTUDIJSKI PROGRAM
PRVE STOPNJE
RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: izr. prof. dr. Uroš Lotrič

Ljubljana, 2013

Rezultati diplomskega dela so intelektualna lastnina Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavljane ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje Fakultete za računalništvo in informatiko ter mentorja.

Besedilo je oblikovano z urejevalnikom besedil \LaTeX .



Št. naloge: 00323/2012

Datum: 03.09.2012

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Kandidat: **DEJAN AMBRUŠ**


Naslov: **VISOKO ZMOGLJIVO RAČUNANJE V OBLAKU WINDOWS AZURE
HIGH PERFORMANCE COMPUTING IN WINDOWS AZURE CLOUD**

Vrsta naloge: Diplomsko delo visokošolskega strokovnega študija prve stopnje

Tematika naloge:

V zadnjem času se mnoge računalniške storitve selijo v računalniške oblake. Ogromne kapacitete računalnikov v oblaku je mogoče izkoristiti tudi za izvajanje kompleksnih matematičnih izračunov. Z nadgradnjo programskega okolja za upravljanje z gruči Windows HPC Server 2008 se je v področje visoko zmogljivega računanja v oblaku vključilo tudi podjetje Microsoft. V nalogi opišite postopek vzpostavitve okolja Windows HPC Server 2008 v oblaku Windows Azure in na primerih preizkusite delovanje sistema pri reševanju računsko zahtevnih problemov.

Mentor:


izr. prof. dr. Uroš Lotrič



Dekan:


prof. dr. Nikolaj Zimic

IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisani Dejan Ambruš, z vpisno številko **63050001**, sem avtor diplomskega dela z naslovom:

Visoko zmogljivo računanje v oblaku Windows Azure

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvomizr. prof. dr. Uroša Lotriča,
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela
- soglašam z javno objavo elektronske oblike diplomskega dela v zbirki "Dela FRI".

V Ljubljani, dne 10. januarja 2013

Podpis avtorja:

Za pomoč pri diplomskem delu bi se rad zahvalil vsem, ki so mi pomagali pri izdelavi dela, še posebej mentorju izr. prof. dr. Urošu Lotriču in materi Alenki Rozman.

Družini

Kazalo

Povzetek

Abstract

1	Uvod	1
2	Računalništvo v oblaku	3
2.1	Delovanje in uporaba storitev v oblaku	3
2.1.1	Storitve IaaS	3
2.1.2	Storitve PaaS	4
2.1.3	Storitve SaaS	4
2.2	Windows Azure	5
2.2.1	Pregled	5
2.2.2	Storitve	6
2.3	Visoko zmogljivo računanje	7
2.3.1	Windows HPC Server in Windows Azure	9
2.3.2	Kombinacija računanja v oblaku z računanjem v lastniških prostorih	14
2.3.3	Računanje zgolj v oblaku	15
3	Namestitev storitve v oblaku Windows Azure	17
3.1	Priprava sistema za virtualizacijo	17
3.2	Nastavitve glavnega računalnika	18
3.2.1	Namestitev operacijskega sistema	18
3.2.2	Nastavitev omrežja	18
3.2.3	Nastavitev poverilnic	19
3.2.4	Nastavitev poimenovanja novih računalnikov	19
3.3	Nastavitev računa v oblaku Windows Azure	20

3.3.1	Namestitev certifikata	20
3.3.2	Kreiranje gostujoče storitve in diskovnega računa	20
3.4	Nastavitve delovne vloge	22
3.4.1	Izdelava slike za delovno vlogo	22
3.4.2	Dodajanje slike	23
3.4.3	Nalaganje slike	25
3.5	Kreiranje predloge	28
4	Izvajanje aplikacij	33
4.1	Zagon računalnikov v gruči	33
4.2	Ureditev okolja za test aplikacije	37
4.3	Test sistema	40
4.4	Rudarjenje besedil	47
4.4.1	Analiza meritev	50
5	Zaključek	53

Povzetek

Visoka zmogljivost, varnost, zanesljivost, prilagodljivost, združljivost, predvsem pa ekonomičnost, so razlogi, da postaja računanje v oblaku v zadnjem času vedno bolj priljubljeno in se vse bolj uporablja tudi v poslovnem svetu. V resnici nam računalništvo v oblaku ponuja še mnogo več. Z uporabo virtualnih računalniških gruč nam omogoča visoko-zmogljivo računanje, s tem pa odpira nove možnosti pri zagotavljanju računalniške podpore za poslovni in javni svet.

V diplomskem delu opisujemo namen uporabe oblakov, njihovo strukturo, prednosti, pa tudi nekatere slabosti. Osredotočili smo se na Windows Azure, ki ga ponuja podjetje Microsoft in bolj podrobno predstavili njegovo konfiguracijo. V drugem delu diplome predstavljamo praktični primer uporabe tega računalniškega okolja. Opisujemo način pristopa z odpiranjem računa v oblaku Windows Azure in postopek namestitve ter ureditve nastavitve sistema, v katerem smo naredili test delovanja aplikacij v načinu visoko-zmogljivega računanja. Možnost dinamičnega dodajanja in odstranjevanja računalnikov in enostaven način upravljanja nas je prepričal, da je uporaba takšnega sistema smiselna tudi z vidika vzdrževanja in zagotovo prinaša številne prednosti v primerjavi s stacionarnimi računalniškimi sistemi.

Abstract

High performance, security, availability, scalability, flexibility and lower costs of maintenance have essentially contributed to the growing popularity of cloud computing in all spheres of life, especially in business. In fact cloud computing offers even more than this. With usage of virtual computing clusters a runtime environment for high performance computing can be efficiently implemented also in a cloud.

There are many advantages but also some disadvantages of cloud computing, some of them are addressed in our thesis, mainly regarding to the Microsoft's public cloud computing platform Windows Azure. We describe the underlying infrastructure and software configuration as well as its functions and server usage charges.

In the second part we describe how to set up a Windows Azure account and configure it for high-performance computing and also how to run many virtual machines simultaneously using HPC Cluster Manager. Dynamic resource allocation, improved flexibility, broad network access, performance enhancements as well as business benefits convinced us there are many benefits of cloud computing comparing to the on-site computing resources.

Poglavje 1

Uvod

Storitve v oblaku so primerne tako za manjša, kot za velika podjetja. Majhna in nova podjetja si z njimi lahko privoščijo storitve, ki bi jim bile sicer zaradi visokih stroškov uvedbe in dragega vzdrževanja nedostopne, velika podjetja pa si z njimi zagotovijo večjo varnost, integracijo in skladnost informacijskih sistemov. Z uvedbo rešitev v oblaku podjetju ni potrebno nameščati visoko-razpoložljivih strežniških gruč, ki poleg drage uvedbe in vzdrževanja zahtevajo dodatne stroške prostora, energije in hlajenja. Bistvenega pomena za vse vrste podjetij je ekonomičnost, saj se procesorske in pomnilniške kapacitete lahko dodajajo glede na dejanske potrebe in ni nevarnosti, da jih zmanjka, niti jih ni potrebno kupovati na zalogo. Visoka razpoložljivost in zanesljivost sta med pomembnejšimi lastnostmi storitev v oblaku.

Storitve v oblaku pa postajajo zanimive še iz enega razloga – omogočajo namreč visokozmogljivo računanje, prav to pa je danes še kako potrebno. Informacijski sistemi in posamezne aplikacije obdelujejo vse večje količine podatkov, na osnovi česar se sprejemajo poslovne odločitve ali napovedujejo vremenski pojavi, ki lahko obvarujejo ljudi pred katastrofami. Skratka, potrebe po visokozmogljivem računanju so iz dneva v dan večje in ljudje si izmišljajo nove in nove naloge na področjih tehnološkega razvoja in znanosti, pa tudi pri povečevanju kvaliteti življenja in zagotavljanju zdravja in varnosti. Še pred nekaj leti je bilo visokozmogljivo računanje predrago za navaden poslovni svet, z razvojem računalniških tehnologij in arhitektur pa je postalo dosegljivo tudi v teh sferah življenja in s tem postalo tudi konkurenčna prednost pred manj zmogljivimi sistemi.

V prvem delu diplomske naloge bomo opisali storitve v oblaku, njihovo delovanje in uporabo. Predstavili bomo tri najbolj tipične oblike storitev, ki so danes v uporabi, to so infrastruktura kot storitev, računalniški sistem kot storitev in programi kot storitev. Osre-

dotočili se bomo na storitve Windows Azure, saj so sistemi Windows med najbolj razširjenimi sistemi tako v Sloveniji, kot tudi v svetu. Na kratko bomo povzeli pregled njihovih storitev in funkcionalnosti. Osrednji del našega diplomskega dela je visoko-zmogljivo računanje v oblaku, ki ga bomo preko sistema HPC predstavili bolj natančno. Razložili bomo, kje so prednosti računanja v oblaku in kakšne so možnost kombinacije s programi na stacionarnih lokacijah. Mnoga podjetja imajo namreč vrsto programov, ki jih uporabljajo v podporo svojim poslovnim procesom in jih ne morejo čez noč zamenjati, nekateri med njimi pa se iz določenih razlogov celo morajo izvajati lokalno. V nadaljevanju bomo predstavili še konkretni postopek priprave sistema za uporabo storitev Windows Azure. Opisali bomo, kako je potrebno pripraviti sistem za virtualizacijo, urediti nastavitve računalniške gruče z glavnim računalnikom in računom v oblaku, nato pa še kako pripraviti predloge, s pomočjo katerih se lahko popolnoma samodejno aktivira potrebno število računalnikov z želenimi operacijskimi sistemi in programi, ki se zaganjajo in ustavljajo ob vnaprej pripravljenih terminih.

Poglavje 2

Računalništvo v oblaku

Računalništvo v oblaku je storitev, ki uporabnikom preko omrežja zagotavlja dostop do računalniških virov, kot so omrežja, strežniki, pomnilniški viri, aplikacije in različne storitve, pri čemer je dodajanje in sproščanje teh virov hitro in enostavno.

2.1 Delovanje in uporaba storitev v oblaku

Oblaki torej skupini končnih uporabnikov v obliki storitve nudijo procesorsko moč in prostor za shranjevanje podatkov. Oskrbujejo jih s storitvami, programsko opremo in zagotavljajo računanje preko omrežja. V osnovi se tako imenovane storitve v oblaku delijo na tri modele, ki jih bomo v nadaljevanju na kratko predstavili in sicer, infrastruktura kot storitev (ang. Infrastructure as a Service, IaaS), računalniški sistem kot storitev (ang. Platform as a Service, PaaS), in programi kot storitev (ang. Software as a Service, SaaS).

2.1.1 Storitve IaaS

Infrastruktura kot storitev (ang. Infrastructure as a Service, IaaS) je najbolj osnovni model storitve v oblaku. Zagotavlja računalnike kot fizične ali še bolj pogosto, virtualne računalnike, s pomnilniškimi in procesorskimi kapacitetami ter osnovnimi omrežnimi storitvami. Virtualni računalniki tečejo preko modulov za upravljanje virtualnih računalnikov (ang. hypervisor) kot gostujoči računalniki in prav združevanje in upravljanja večjega števila takšnih modulov s pomočjo operativne podpore v oblaku zagotavlja razširljivost in podporo velikemu številu virtualnih računalnikov. Zagotovljeni so še drugi viri, knjižnice slik, diskovne kapacitete v datotečni ali blokovni obliki, požarni zidovi, izenačevanje porabe, naslovi IP, omrežja VLAN in svežnji programske opreme. Zani-

miva je storitev izenačevanja porabe, ki omogoča enakomerno razporeditev obremenitve med računalniki. Strojna oprema se fizično nahaja v tako imenovanih podatkovnih oziroma računalniških centrih. Ti centri so opremljeni tako, da zagotavljajo visoko varnost s podvojenimi ključnimi komponentami, z varnostnimi kopijami podatkov, dodatnimi viri elektrike in nadzorno-kontrolnimi sistemi, ki izvajajo zaščito pred izpadom sistema.

Storitev se uporablja tako, da si uporabnik namesti sliko operacijskega sistema in svoje aplikacije. V tem primeru je uporabnik zadolžen za vzdrževanje operacijskega sistema in aplikacij, stroški storitve pa se zaračunavajo na podlagi sredstev in moči procesiranja, ki se koristi.

2.1.2 Storitve PaaS

Računalniški sistem kot storitev (ang. Platform as a Service) predstavlja delujoč računalniški sistem z operacijskimi sistemi in okoljem za izvajanje programov. Razvijalci programske opreme lahko razvijajo in izvajajo svoje programske rešitve v oblaku, ne da bi se pri tem morali ukvarjati z nakupovanjem in vzdrževanjem drage strojne in systemske opreme. V določenih primerih je na voljo celo avtomatsko prilagajanje strojnih virov zahtevam aplikacije.

Na trgu obstaja več različnih sistemov, ki zagotavljajo storitve PaaS, med katerimi so najbolj znani Amazon EC2, Google App Engine, Engine Yard, od leta 2010 dalje pa tudi Windows Azure. Ti sistemi ponujajo upravljanje in vzdrževanje izvršljivih programov, podatkovnih baz, operacijskih sistemov, virtualizacijo, namestitve in upravljanje strežnikov, spominskih kapacitet in omrežij. Vse to lahko organizaciji prinese velike ekonomske prednosti, saj po nekaterih analizah kar 90% celotnih stroškov informacijskih sistemov predstavljajo prav operativni stroški naštetih storitev.

2.1.3 Storitve SaaS

V primeru modula programi kot storitev (ang. Software as a Service, SaaS) ponudnik storitve namesti v oblak delujočo programsko rešitev, do katere uporabniki dostopajo preko odjemalcev v oblaku. Uporabniki ne upravljajo računalniškega sistema, na katerem teče aplikacija, zato namestitev programa na lokalne uporabniške računalnike ni potrebna, kar močno poenostavi samo podporo in vzdrževanje. Aplikacija v oblaku se od standardne aplikacije razlikuje v svoji razširljivosti, ki se doseže s kloniranjem zahtevanih opravil na več virtualnih računalnikov med delovanjem. Izenačevalci obremenitve razporejajo delo

preko več računalnikov in obenem omogočajo uporabniku enotno prijavno točko.

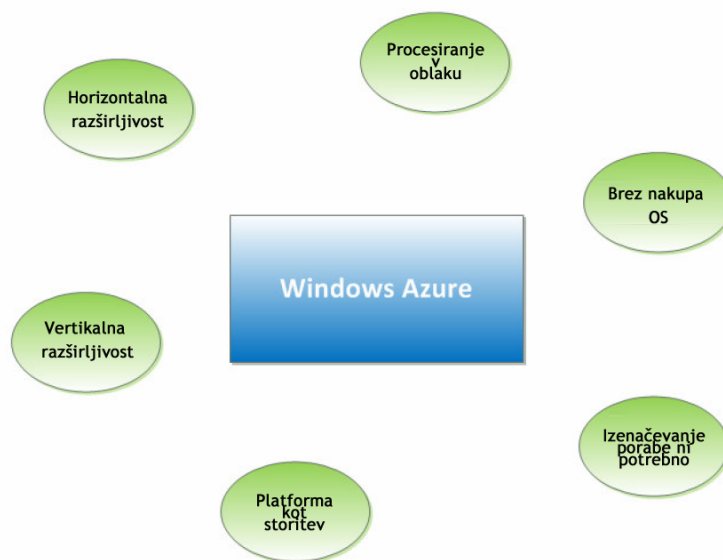
2.2 Windows Azure

Microsoftova platforma za računalništvo v oblaku Windows Azure, ponuja hitro izgradnjo, uvajanje in upravljanje aplikacij preko svetovnega spleta in podatkovnih centrov, ki jih upravlja Microsoft. To je odprta in fleksibilna platforma, na kateri lahko tečejo aplikacije, napisane v katerem koli programskem jeziku. Omogoča izgradnjo spletnih aplikacij z orodji ASP.NET, PHP ali Node.js. Takšne aplikacije je mogoče uporabljati tako v storitvah v oblaku, kot lokalno, lahko pa se tudi med seboj povezujejo. Preko Pogodbe o zagotavljanju razpoložljivosti (ang. Service Level Agreement, SLA) je zagotovljena 99,95% razpoložljivost, kar pomeni, da je skupni čas prekinitve v roku enega meseca dobrih 21 minut. Tako visoka stopnja razpoložljivosti zadosti potrebam najzahtevnejših aplikacij.

Vse te lastnosti omogočajo nemoteno izgradnjo in delovanje programov, ne da bi se organizacija pri tem morala posebej ukvarjati z infrastrukturo. Windows Azure ponuja samodejno posodabljanje in nadgrajevanje operacijskih sistemov, vgrajene sisteme za enakomerno obremenitev omrežnega prometa in povečano odpornost proti napakam v strojni opremi. Posebni uvajalni modul zagotavlja nadgrajevanje programov brez prekinitve delovanja. Prilagodi se lahko različno velikim aplikativnim potrebam po strojnih virih. Gre za nekakšno samopostrežno platformo, ki omogoča spreminjanje velikosti strojnih virov v nekaj minutah. Procesorji in pomnilnik se dinamično prilagajajo posameznim potrebam in se obračunavajo po dejanski porabi. Na voljo je v številnih podatkovnih centrih po svetu, zato se namestitev lahko izvede v bližini organizacije. Platforma lahko organizaciji prihrani visoke stroške licenc, kar zmanjšuje operativne stroške in poveča možnost investicije v aplikativne izboljšave.

2.2.1 Pregled

Pregled funkcionalnosti, ki jih prinaša Azure, predstavlja slika 2.1. Njegova primarna funkcionalnost je seveda računanje ali procesiranje, obenem s tema pa prinaša še druge prednosti, s čimer se celotna platforma uporablja zgolj kot storitev. Arhitektura Windows Azure zagotavlja razpoložljivost aplikacije med nadgradnjami in celo v primeru okvare strojne opreme. V arhitekturo so vključeni sistemi za izenačevanje obremenitve, kar zagotavlja razširljivost aplikacij. Ta je lahko horizontalna ali vertikalna. Pri verti-



Slika 2.1: Funkcionalnosti Windows Azure.

kalni razširljivosti gre za povečevanje procesorskih in pomnilniških virov, pri horizontalni razširljivosti pa za povečevanje števila aplikativnih instanc. Prilagodljiv je tudi v primeru časovno ne-enakomernih potreb po strojnih virih, saj se ti lahko povečajo, po potrebi pa tudi zmanjšajo.

2.2.2 Storitve

Med aplikativnimi storitvami je na razpolago preverjanje pristnosti, shranjevanje podatkov v predpomnilniku, elektronska pošta, medijske storitve, obveščanje, integracija, spremljanje delovanja, upravljanje in nadzor aplikacij [4].

Kontrola dostopa do aplikacij se lahko integrira s programi WIF (ang. Windows Identity Foundation) in priljubljenimi metodami identifikacije spletnih ponudnikov, vključno z Windows Live ID, Google, Yahoo, Facebook in podporo za ADFS 2.0. (ang. Active Directory Federation Services). Lahko se uporabi tudi enkratna spletna prijava (ang. single sign-on), ki je zasnovana na mehanizmu varnostnega žetona STS (ang. Security Token Service). Začasno shranjevanje podatkov iz drugih virov v predpomnilnik povečuje učinkovitost in lahko zmanjša stroške baznih transakcij v oblaku. Aplikacije v Azurjevem oblaku lahko uporabljajo funkcionalnost spletne pošte SendGrid, ki zagotavlja zanesljivo dostavo, analize v realnem času in prilagodljive knjižnice, ki uporabnikom omogočajo enostavno vključitev te storitve v aplikacije. Uporabljajo lahko tudi SMS in zvokovno storitev

Twilio, preko katere sprejemajo telefonske klice, sprejemajo in pošiljajo kratka sporočila in omogočajo zvokovno komunikacijo preko obstoječih internetnih povezav, vključno z mobilnimi povezavami. Medijske storitve zagotavljajo razširljivo platformo in se lahko uporabljajo za izpolnjevanje različnih nalog, vključno s prenosom, shranjevanjem, kodiranjem in pretakanjem različnih vsebin. Lahko se izkoristi sistem end-to-end, ki integrira posamezne komponente z obstoječimi orodji in procesi. Storitve sporočilnih vrst ponujajo preprosto zaporedno dostavo sporočil, pri čemer podpirajo standardne protokole, kot sta Representational state transfer (REST) in Advanced Message Queuing Protocol (AMQP), ter knjižnice, preko katerih se sporočila pošiljajo in pobirajo iz sporočilnih vrst.

2.3 Visoko zmogljivo računanje

Visoko zmogljivo računanje (ang. High Performance Computing, HPC) je področje računalništva, ki se osredotoča na razvoj programske opreme in algoritmov, ki omogočajo delitev programov na manjše enote na takšen način, da se posamezne enote programa izvajajo istočasno na ločenih procesorjih [1]. Takšno paralelno procesiranje zagotavlja hitro izvedbo programov z zahtevnimi računskimi nalogami. Zmogljivost procesiranja sega v območje tera flopov ali 10^{12} operacij s plavajočo vejico na sekundo. Izraz HPC se uporablja tudi kot sinonim za računanje superračunalnikov, ki pa v resnici dosegajo najvišjo stopnjo zmogljivosti, to je območje peta flopov, kar pomeni 10^{15} operacij s plavajočo vejico v sekundi.

Najpogostejši uporabniki sistemov visoko zmogljivega računanja so znanstveni raziskovalci, inženirji in akademske ustanove. Uporablja se na različnih področjih znanosti, kot so kvantna in jedrska fizika ter molekularno modeliranje, kjer se izračunavajo strukture in lastnosti kemičnih spojin, bioloških makromolekul, polimerov in kristalov. Uporablja se tudi pri podnebnih raziskavah in napovedovanju vremena, pri simulacijah letal v vetrovnikih, pri simulacijah eksplozij jedrskega orožja in v vojaški raziskavah nasploh, v zadnjem času pa vse bolj tudi na področju podatkovnega rudarjenja. Ker se zahteve po procesorski moči nenehno povečujejo, je takšno računanje vse bolj aktualno tudi v poslovnem svetu, predvsem v transakcijskih sistemih in podatkovnih skladiščih.

Microsoft je prav z operacijskim sistemom Windows HPC Server 2008 R2 naredil ključni korak pri doseganju dolgoročnega cilja, da se tovrstno procesiranje približa širšemu krogu uporabnikov. Upabniki sedaj lahko uporabljajo HPC Server kot osnovo za izgradnjo in

izvajanje simulacij, ki modelirajo svet okrog nas, pospešujejo raziskovanja in pomagajo pri sprejemanju kvalitetnejših odločitev. Windows HPC server 2008 R2 tako predstavlja izjemno priložnost za pridobitev kvalitete ogromnih količin podatkov, obenem pa omogoča izkoriščanje prednosti novih tehnologij, kot so strežniške gruče, grafično-procesne enote, računanje v oblaku in več-jedrno procesiranje. Podobne funkcionalnosti od leta 2012 ponuja odprto-kodno programsko okolje Apache Hadoop, ki z Google-ovim programskim modelom MapReduce omogoča porazdeljeno procesiranje programov na računalniških gručah in s tem računske obdelave velikih naborov strukturiranih in tudi ne-strukturiranih podatkov [13].

Bistvo visoko zmogljivega računanja je moč računanja. Visoko zmogljivo računanje je v šestdesetih letih prejšnjega stoletja pripadalo domeni super računalnikov, ki so si to vlogo zadržali skoraj do konca prejšnjega stoletja. Danes se za visoko zmogljivo računanje uporabljajo predvsem gruče računalnikov. Seveda morajo imeti računalniki za povezovanje v gruče ustrezno programsko opremo. Windows HPC Server 2008 R2, ki vsebuje pripomočke za zaganjanje programov v gruči, nam takšno povezovanje omogoča, s tem pa tudi visoko zmogljivo računanje. Številne organizacije iz ekonomskih razlogov za reševanje težkih problemov, ki zahtevajo veliko moč računanja, uporabljajo strežniške gruče Windows. Z vse bolj pogosto uporabo storitve v oblaku nastajajo spremembe tudi v svetu visoko zmogljivega računanja. Oblak Windows Azure ima veliko prednosti. Med drugim:

- ponuja mnogo virtualnih računalnikov in s tem tudi zelo veliko cenovno ugodnega prostora za shranjevanje podatkov;
- omogoča uporabo toliko sredstev, kolikor jih potrebujemo, kar je največja prednost oblaka Windows Azure;
- je enostaven za uporabo, saj gručo računalnikov dobimo z nekaj kliki;
- sredstva lahko dinamično prilagajamo in v primeru, da ena od naših aplikacij začasno potrebuje večjo procesorsko moč, jo naročimo in ko je ne potrebujemo več, to odstranimo.

Kljub številnim prednostim pa se uporaba oblaka v določenih primerih ne izkaže kot dobra rešitev, na primer:

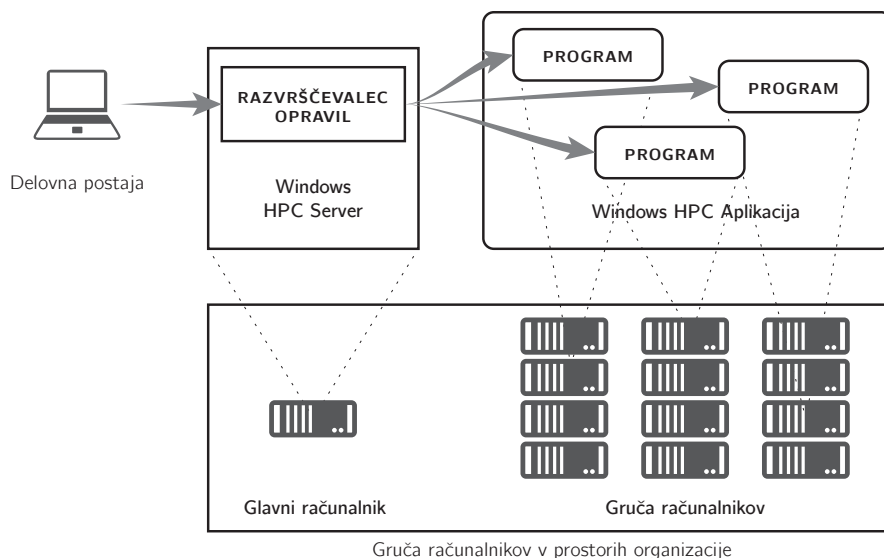
- občasno delamo s podatki, ki se ne smejo hraniti ali procesirati v oblaku zaradi pravnih, regulativnih ali drugih omejitev;

- nekatere naloge uporabljajo programsko opremo, ki jo platforma v oblaku ne podpira;
- naloge, ki se nanašajo na ogromno maso podatkov, večkrat niso primerne za računanje v oblaku, saj je potrebno prenesti podatke v oblak, kar pa lahko vzame preveč časa.

Dejstvo je, da računalniške gruče v lastniških sistemih še ne bodo izginile, zato je smiselno, da se fizične gruče v lastniških prostorih kombinirajo z virtualnimi gručami v oblaku. Ravno to je implementiral Windows Server.

2.3.1 Windows HPC Server in Windows Azure

Tehnologija HPC deluje tako, da se na vseh računalnikih izvajajo različni programi ali, da se na vseh računalnikih izvajajo isti programi, vendar se pri tem uporabljajo različni podatki. Za določevanje računalniških programov, računalnikom Windows HPC ponuja razvrščevalc opravil (ang. Scheduler), ki ga prikazuje slika 2.2.



Slika 2.2: Gruča računalnikov v lastniških prostorih.

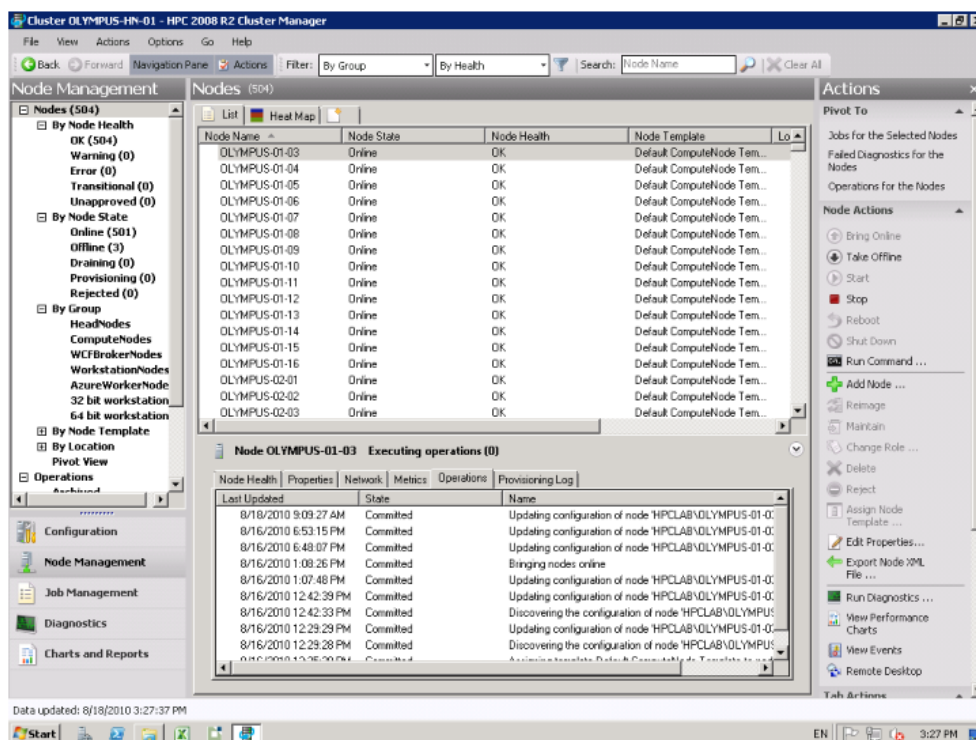
Razvrščevalc opravil se zaganja na posebnem računalniku, ki je v večini primerov tudi

glavni računalnik (ang. Head node). Naloge se prenesejo iz delovne postaje (ang. Work station) na glavni računalnik, kjer razvrščevalec opravil razdeli programe na več računalnikov. Vloga razvrščevalca opravil ni zgolj dodeljevanje nalog drugim računalnikom, temveč tudi določanje njihovih priorit. Tako se naloge z višjo prioriteto izvajajo pred nalogami z nižjo prioriteto. Hkrati pa razvrščevalec opravil omogoča izbiro računalnikov ali skupin računalnikov, na katerih želimo nalogo izvajati. Obstajajo rešitve, ki v primeru okvare glavnega računalnika, njegove naloge preselijo na drug računalnik (ang. failover). Izvajanje programov lahko poteka na računalnikih simultano, zato Windows HPC Server podpira MPI (ang. Message Passing Interface). To je standardni sistem za posredovanje sporočil, zasnovan za delovanje na različnih, vzporedno delujočih računalnikih, kar je vzpodbudilo razvoj prenosljivih in prilagodljivih vzporedno delujočih programskih rešitev. HPC za izvajane nalog pogosto izkorišča več jeder, zato mora zagotavljati knjižnice kot na primer OpenMP (ang. Open Multiprocessing), Parallel LINQ (mehanizem za hkratno oziroma paralelno izvajanje poizvedb) in TPL (ang. Task Parallel Library), ki je komponenta za razširitev paralelizma v okolje .NET. Okolje visoko zmogljivega računanja hkrati omogoča razhroščevanje paralelnih aplikacij.

HPC Server se večinoma osredotoča na kreiranje in urejanje aplikacij, kar ni ravno trivialna naloga. Administrator mora ustvariti gručo in določiti enote, ki bodo sodelovale pri nalogi, pri čemer lahko aplikacijam določi točno določen niz računalnikov za izvajanje aplikacije, hkrati pa lahko aplikacija sama za izvajanje nalog izbere dodatne računalnike. V okolju Windows se za upravljanje računalniških gruč uporablja programsko orodje HPC Cluster Manager, ki ga predstavlja slika 2.3. Orodje je namenjeno nastavljanju in upravljanju virtualnih računalnikov in računalniških gruč, upravljanju nalog, diagnostiki in različnim oblikam poročil.

Slika prikazuje, kako administrator ureja gručo računalnikov in naloge, ki se izvajajo na gruči, izvaja diagnostične teste na gruči ter izdeluje grafe in poročila o računalnikih. Urejevalnik hkrati omogoča grafični prikaz izkoriščenosti virov posameznih računalnikov, pa tudi njihove sistemske nastavitve.

Tako kot Windows HPC Server, se tudi Windows Azure za izvajanje nalog zanaša na veliko število računalnikov. Windows Azure deluje kot gruča računalnikov, ki so fizično nameščeni v računalniških centrih, te pa nadzoruje Microsoft. Prostori računalniških centrov so dostopni preko internetnega omrežja, in omogočajo uporabo toliko računalnikov, kot jih organizacija potrebuje. Stroški se obračunajo tako, da uporabnik plača le toliko enot procesorskih in pomnilniških virov, kolikor jih koristi. Pri čemer je eno enoto pro-



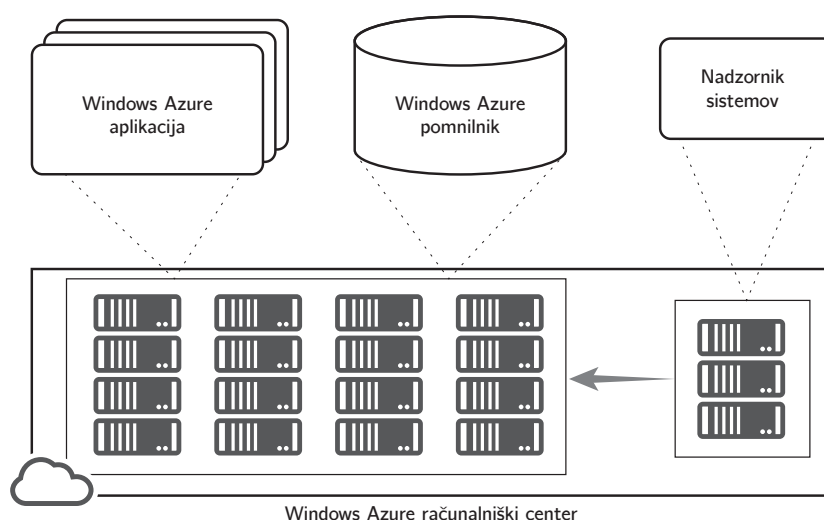
Slika 2.3: Programsko okolje HPC Cluster Manager.

cesorskih in pomnilniških virov sestavlja en CPE s frekvenco 1,6 GHz, 1,75 GB RAM-a, 225 GB diskovnega prostora in 100 Mbps pasovne širine. Enote se povečujejo vse do 8 x 1,6 GHz, 14 GB RAM-a, 2.040 GB diskovnega prostora in 800 Mbps pasovne širine.

Slika 2.4 prikazuje potek aplikacij na računalnikih Windows Azure. Tako aplikacije, ki tečejo na računalnikih Windows Azure, kot tiste, ki se izvajajo v lastniških prostorih, lahko koristijo prostor na Windows Azure.

Windows Azure omogoča dva tipa shranjevanja podatkov. Prvi je shranjevanje v nestrukturirane binarne zbirke, drug način so tabele, ki omogočajo strukturiran način shranjevanja podatkov, pri čemer tabele niso relacijske. Vse to ureja aplikacija, ki se imenuje nadzornik sistemov (ang. fabric controller). Aplikacija teče na posebej rezerviranih računalnikih v prostorih Windows Azure in ureja vse posodobitve in sistemske popravke.

Da bi razumeli, kako Windows Azure in HPC Server sodelujeta med sabo, je potrebno razumeti, kako delujejo programi v oblaku Windows Azure. Spletna vloga ali »Web role« se tipično uporablja za aplikacije, ki potekajo preko protokola HTTP in storitve IIS (ang. Internet Information Service). Delovna vloga (ang. Worker role) se običajno uporablja za različne vrste procesiranja. Ne glede na to, katera vloga se uporablja, Windows Azure zahteva, da se vloga izvaja vsaj na dveh primerkih, to pomeni, na dveh virtualnih računal-

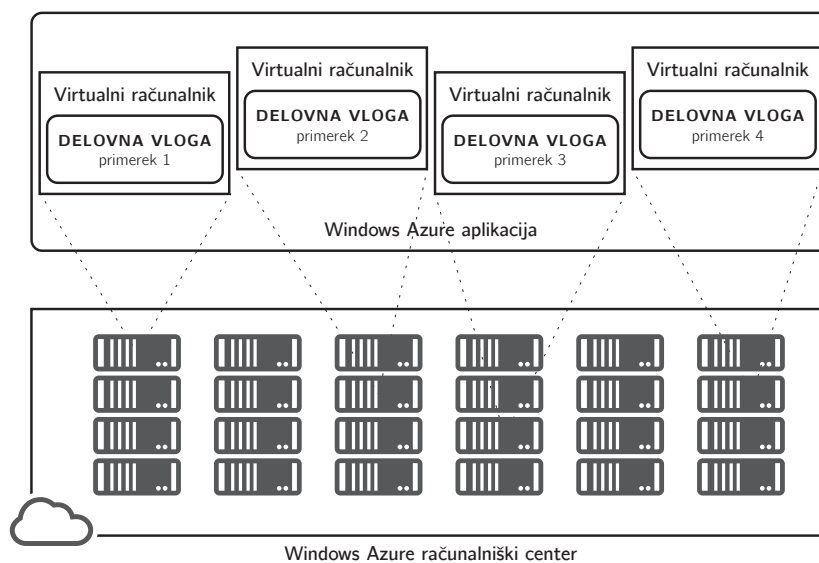


Slika 2.4: Shema računalniškega centra, ki ga nadzoruje Microsoft.

nikih. Ta dva primerka sta zamenljiva, kar zagotavlja delovanje aplikacije tudi v primeru, ko en primerek iz kakršnegakoli razloga preneha delovati ali postane nedostopen. Takšen način nam zagotavlja visoko razpoložljivost aplikacije.

V primeru na sliki 2.5 je prikazana aplikacija, ki vsebuje eno spletno vlogo in eno delovno vlogo. Od tu naprej je spreminjanje primerkov enostavno, bodisi s strani administratorja, bodisi aplikacije. Na sliki je razvidno, kako program nadzornik sistemov vsak primerek naslavlja na drug računalnik, zaradi česar je sistem bolj odporen proti napakam. V primeru, da se računalnik pokvari, nadzornik sistemov po določenem času prižge nov primerek iste vloge, med tem pa se sklene pogodba SLA (ang. Service Level Agreement). Ker ima vsak primerek enega ali več jeder rezerviranih za aplikacijo, lahko predvidevamo napredek aplikacije, pri čemer se jedra, ki skrbijo za sistem, uporabnikom ne prikazujejo. Windows Azure v osnovi podpira Windows aplikacije, zato lahko na njem poganjamo aplikacije, narejene v .NET okolju ali napisane v programskem jeziku C++. Omogoča pa tudi druge tehnologije, kot sta Java in PHP.

Ne glede na to, s katero tehnologijo je aplikacija narejena, lahko uporabljamo prostor na različnih mestih. Uporabljamo lahko Azure-jeve tabele, blob-e (ang. Binary large object) in tudi SQL Azure, storitev oblaka za relacijske podatke. Aplikacije imajo prosto pot in lahko dostopajo tako do prostora, ki pripada stacionarnim računalnikom v lastniških



Slika 2.5: Aplikacija, ki teče na računalniškem centru.

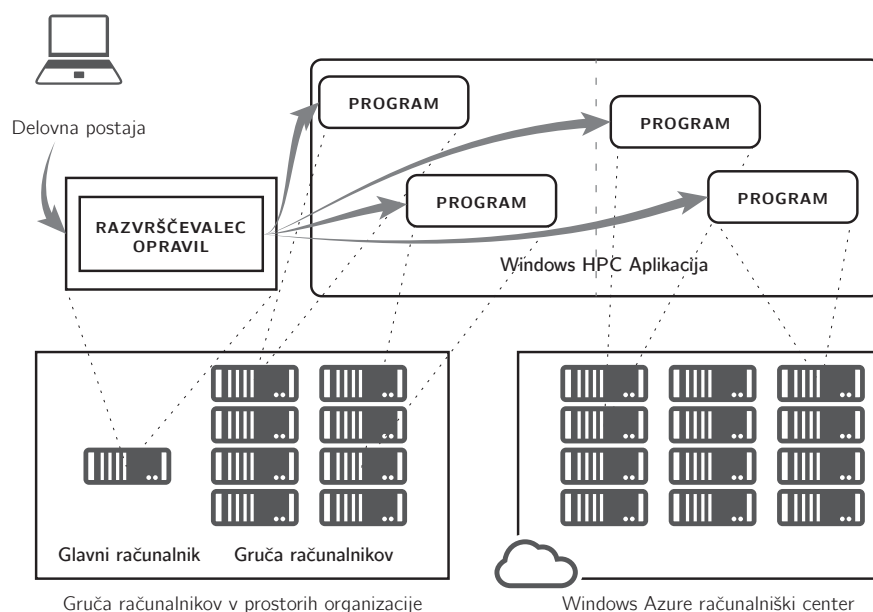
prostorih, kot do prostora v oblaku. To pomeni, da nismo omejeni le na prostor v oblaku. Da lahko bolje ocenimo Windows Azure, moramo poznati način obračunavanja storitev.

- Za procesiranje moramo plačati od \$0,05 do \$0,96 na uro za vsako enoto. Cena je odvisna od velikosti enote. Izbiramo lahko med enim, dvema, štirimi ali osmimi jedri. Cena po drugi strani ni odvisna od obremenitve, kar pomeni, da plačamo isto ceno za enoto, ki je le prižgan in enoto, ki izvaja zahtevne računske operacije.
- Prostor na disku nas stane od \$0,1 do \$0,15 za GB na mesec. Obenem moramo plačati \$0,01 na 10.000 operacij, kot so branje ali pisanje.
- Pasovna širina 1 GB prenosa z odjemalca (ang. upload) nas stane \$0,1, pasovna širina 1 GB prenosa na odjemalca (ang. download) pa \$0,15, kar velja zgolj za podatke aplikacije, ne pa aplikacije same.

Platforma v oblaku predstavlja nov pristop za nalaganje, uporabljanje in obračunavanje računalniških virov, Microsoft pa nam obenem daje nov način programiranja aplikacij v Windows okolju.

2.3.2 Kombinacija računanja v oblaku z računanjem v lastniških prostorih

Visoko zmogljivo računanje je odvisno od velikosti procesorske moči, Windows Azure pa omogoča veliko procesorske moči na zahtevo. Očitno je, da sta narejeni drug za drugega. Ena od možnosti uporabe je, da se glavni računalnik in nekaj računalnikov namesti v lastniških prostorih, ostali pa se namestijo v oblaku. Druga možnost je, da se v lastniških prostorih namesti le glavni računalnik, drugi računalniki pa so v oblaku. Iz arhitekturnega vidika je izvajanje aplikacije, ki teče delno na gruči, ki se nahaja v lastniških prostorih in delno v oblaku, zelo enostavno.



Slika 2.6: Kombinacija oblaka in gruče v lastniških prostorih.

Slika 2.6 prikazuje arhitekturo z glavnim računalnikom v lastniških prostorih. Uporabnik zaganja aplikacijo s svoje delovne postaje na običajen način. Razvrščevalec opravi nato porazdeli logiko na dostopne vire računalnikov. V tem primeru so dostopni viri v lastniških prostorih in viri v oblaku. Uporabnik ne opazi razlike pri izvajanju aplikacije. Razlika je le v tem, da mora nekaj dodatnega dela narediti administrator. Ta mora odpreti račun na Windows Azure in nastavite delovne postaje, hkrati mora nastaviti, katere aplikacije uporabljajo posamezne vire.

Delovne postaje na Windows Azure se ne nastavijo samodejno, ko se naloga začne. Na-

stavi jih administrator po potrebi. Ko jih ne potrebuje več, pa jih ugasne. Priporočljivo je, da se delovne postaje na Windows Azure v primeru, ko jih ne potrebujemo, ugašajo, saj nam sicer teče čas, ki se nam v skladu s pogodbo zaračuna.

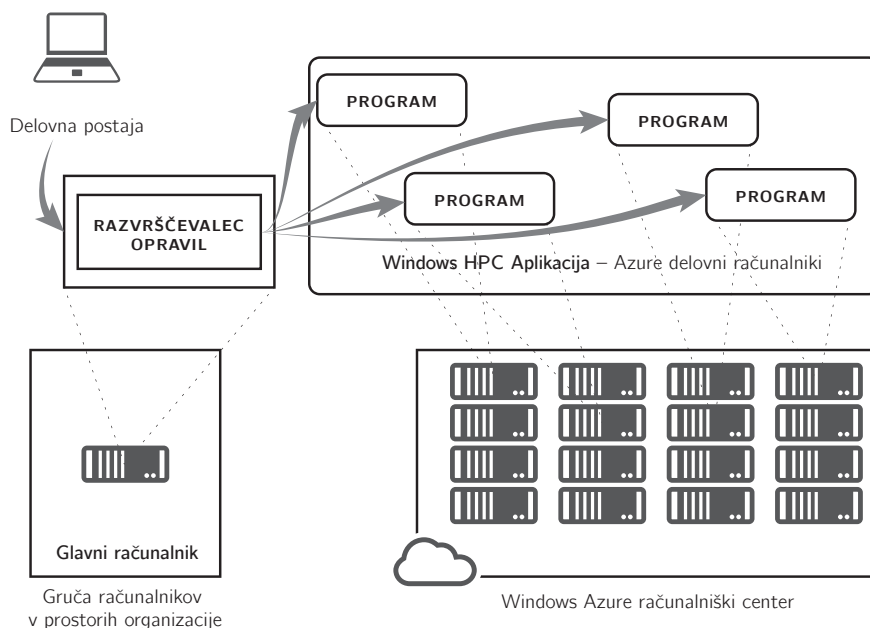
Navajmo nekaj primerov, ko je priporočljivo uporabljati kombinirano metodo, to je računalniška gruča v lastniških prostorih in gruča v oblaku.

- V primeru, da aplikacija občasno potrebuje več procesorske moči, lahko pri Windows Azure začasno naročimo dodatne računalnike. Ko jih ne potrebujemo več, jih enostavno odstranimo. To je ceneje, kot če bi kupili nove stacionarne računalnike, ki bi jih namestili v lastniške prostore in bi bili občasno neizkoriščeni.
- Podobno je v primeru, da občasno potrebujemo več procesorske moči, na primer za tedenske ali mesečne obdelave podatkov. Na Windows Azure enkrat tedensko ali mesečno dokupimo dodatne računalnike, ki jih potem, ko jih ne potrebujemo več, preprosto ugasnemo.
- V primeru, da nam v lastniških prostorih primanjkuje fizičnega prostora, električne moči ali imamo težave s hlajenjem, je Windows Azure spet ugodna rešitev, saj bi si v nasprotnem primeru nakopali veliko dodatnih stroškov z dokupom ali preureditvijo ustreznega računalniškega prostora.

Prednost imajo tudi programerji, ki programirajo visoko zmogljive aplikacije v okolju visoko zmogljivega računanja, saj lahko uporabljajo vsa orodja, ki so jih uporabljali doslej, obenem pa imajo dostop do veliko večje procesorske moči.

2.3.3 Računanje zgolj v oblaku

Če je smiselno naročanje računalnikov v Windows Azure, zakaj potem ne bi postavili oblak vseh programov in podatkov? In dejansko je to smiselno storiti, če nimamo kakšnega od že navedenih zadržkov, ko se iz določenih razlogov podatki ne smejo procesirati izven organizacije, če platforma v oblaku ne podpira naših programov ali programi ne zahtevajo prenosa zelo velikih količin podatkov [2]. Slika 2.7 prikazuje arhitekturo sistema, ko je na fizični lokaciji organizacije samo glavni računalnik, vsi ostali pa so v oblaku. Zaganjanje nalog je podobno, kot v prejšnjem primeru, le da upravitelj razdeli logiko aplikacije le na računalnike v oblaku. Ta način delovanja ima prednosti.



Slika 2.7: Aplikacija, ki teče samo na oblaku.

- Ni stroškov nakupa, nameščanja in vzdrževanja novih računalnikov. Cenovno gledano je cena novih fizičnih računalnikov primerljiva s ceno računalnikov, ki jih uporabljamo v Windows Azure.
- Podobno, kot v prejšnjem primeru, je prednost oblaka v tem, da uporabljamo in plačamo le toliko računalniških virov, kot jih potrebujemo.
- Za razliko od prejšnjega primera, so v drugem primeru vsi podatki v oblaku, kar posledično privede do hitrejšega dostopa, saj so vsi podatki na enem mestu. Zato je arhitektura z vsemi računalniki v oblaku boljša.
- Windows Azure uporablja nadzornik sistemov, ki nam pomaga pri posodabljanju sistema in odpravljanju napak. Ta dela se v oblaku izvajajo samodejno, kar pomeni prihranek pri stroških vzdrževanja.

Poglavje 3

Namestitev storitve v oblaku Windows Azure

Sama namestitev sistema HPC v oblaku Azure je preprosta, seveda pa je potrebno pred tem dobro poznati zmožnosti in omejitve, predvsem pa želene učinke sistema. Pripravili smo postopek, po katerem je preko orodja HPC Cluster Manager možno zaganjati programe, napisane za paralelno procesiranje, in izbirati vrsto in število računalnikov za izvajanje nalog. Po opisanem postopku se na glavni računalnik (ang. head node) namesti programski paket HPC, odpre se račun in vzpostavi povezava med lokalnim sistemom in oblakom. Za tem se namesti delovna vloga (ang. worker node), pripravijo se še predloge, ki jih želimo v nadaljevanju uporabljati, in osnovna nastavitve je s tem zaključena.

3.1 Priprava sistema za virtualizacijo

Računalnik, na katerega smo nameravali namestiti virtualne računalnike, je bilo najprej potrebno ustrezno pripraviti [3]. V prvem koraku smo preverili, ali računalnik podpira virtualizacijo preko programa Hyper-V. Hyper-V je Microsoftov sistemski program, ki omogoča zagon in upravljanje virtualnih računalnikov na 32 in 64 bitnih računalniških sistemih. Beta različica je bila že vključena v nekatere operacijske sisteme Windows 2008, z oznako Windows 2008 R2 pa je Hyper-V postal del standardne izdaje tega operacijskega sistema.

Kontrolo ustreznosti smo naredili tako, da smo v programu BIOS (ang. Basic Input/Output Systems, Osnovno vhodno-izhodni sistem) preverili opciji Virtualizacijske tehnologije (ang. Virtualization technology) in Onemogoči izvršitev (ang. Execute Disable), ki ju je

potrebno vključiti. Opcija Execute Disable je varnostna funkcija v Intel-ovih procesorjih, ki omogoča zaščito pred virusi tako, da aplikacijam onemogoči dostop do dela pomnilnika, ki je rezerviran procesorju. Vključili smo obe opciji in pričeli s postopkom namestitve.

3.2 Nastavitve glavnega računalnika

3.2.1 Namestitev operacijskega sistema

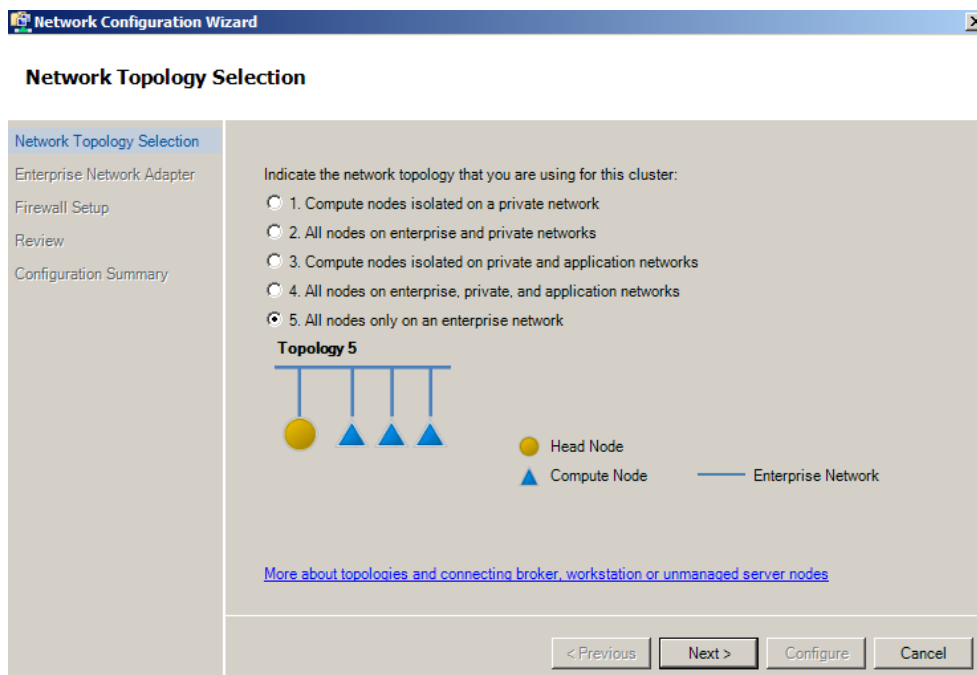
S pomočjo programa Hyper-V smo se lotili namestitve novega virtualnega računalnika z operacijskim sistemom Windows Server 2008 HPC Edition, ki bo imel vlogo glavnega računalnika (ang. Head Node). Glavni računalnik krmili celoten virtualni sistem v oblaku. V primeru, ko je med glavnim računalnikom in ostalimi virtualnimi računalniki potrebne veliko komunikacije, je bolje, da se ta namesti v oblaku, v našem primeru pa smo za test izbrali namestitev na lokalni sistem.

Glavni računalnik je v našem primeru namenjen zagonu programov v načinu visokozmogljivega računanja, za kar ga je bilo potrebno ustrezno pripraviti. Minimalna zahtevana različica popravkov za delovanje računalnika v oblaku Azure je SP2, vendar je iz varnostnih razlogov vedno bolje uporabiti najnovejšo preverjeno različico, zato smo na računalnik namestili posodobitve s paketom SP3 . Hkrati smo namestili še paket programov HPC (ang. High Performance Computing Pack), ki med drugim vsebuje tudi orodje za upravljanje gruč HPC (ang. Cluster Manager). Upravitelj gruč HPC smo kasneje uporabljali kot glavni program za zaganjanje nalog na gruči. Glavni računalnik je bilo potrebno dodati še v domeno, pri čemer smo lahko izbirali med obstoječo in novo domeno. Po priporočilu smo se odločili, da kreiramo novo domeno, ki smo jo poimenovali DejanAzure.net.

3.2.2 Nastavitev omrežja

V programu HPC Cluster Manager smo najprej nastavili omrežje in sicer smo med naborom različnih topologij izbrali topologijo vodila, po kateri so vsi računalniki povezani v eno omrežje, kar je prikazano na sliki 3.1. Z rumeno je označen glavni računalnik, z modro pa računalniki, ki se v gručo na Windows Azure priključujejo naknadno.

V primeru, da je v omrežju vključen požarni zid, je potrebno omogočiti pretok določene vrste omrežnega prometa preko ustreznih vrat, kar nam je sicer samodejno uredila na-



Slika 3.1: Nastavitev omrežja.

mestitev paketa HPC Pack SP3. Samo nastavitev in delovanje smo dodatno preverili s programom HPC Cluster Manager in sicer s funkcijo Windows Azure Firewall Ports Test. Za delovanje mora biti omogočen promet TCP preko vrat 443 in 3389.

3.2.3 Nastavitev poverilnic

Ob nastavitvi poverilnic, ki so potrebne za upravljanje virtualnih računalnikov v oblaku preko našega upravljalnika gruč, smo določili uporabniško ime in geslo, ki se mora ujemati z uporabniškim imenom in geslom na glavnem računalniku.

3.2.4 Nastavitev poimenovanja novih računalnikov

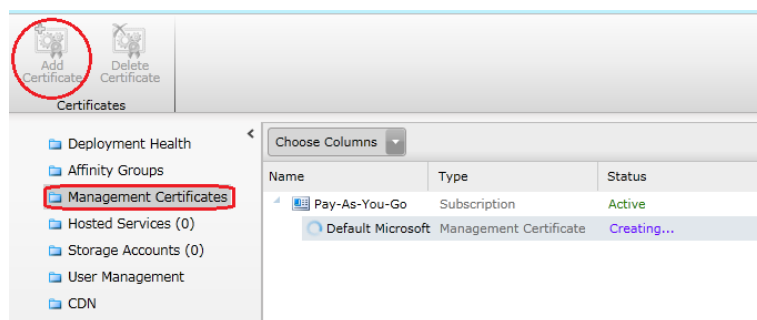
Ob nastavitvi poimenovanje določimo, kako naj bo sestavljeno ime novih virtualnih računalnikov, pri čemer se določi dolžina imena in maksimalno število novih računalnikov [5]. Takšno poimenovanje se uporabi v primeru, ko se na sistemu samodejno, preko časovnega razporejevalnika, ali ročno, glede na potrebe, dodajajo novi virtualni računalniki. Novi računalniki se lahko avtomatsko dodajo tudi na zahtevo aplikacije.

3.3 Nastavitev računa v oblaku Windows Azure

3.3.1 Namestitev certifikata

Za dostop do oddaljenih storitev v oblaku Windows Azure je potrebno pridobiti uporabniški račun in ga ustrezno urediti [6]. Ker je delo v oblaku že v svoji osnovi pogojeno s komunikacijo po omrežjih LAN in WAN, je zagotavljanje varnosti ključnega pomena. Zato nas že postopek prijave prisili v namestitev certifikata, ki preko izmenjave javnih ključev zagotavlja varno komunikacijo med našim glavnim računalnikom in ostalimi računalniki v oblaku. Pri tem se lahko odločimo za samostojno izdelavo certifikata, lahko pa uporabimo privzeti certifikat, ki ga vsebuje paket HPC Pack. Sami smo se odločili za drugo možnost in izbrali izdelan certifikat. Našli smo ga v mapi paketa HPC pod imenom hpcert.cer.

Prijavili smo se preko vstopne točke <https://manage.windowsazure.com> in uporabili uporabniško ime in geslo, ki ga je za testne namene pridobila fakulteta. S prijavo smo izbrali ustrezno namizje, kjer smo nadaljevali z nameščanjem certifikata. V meniju smo izbrali polje Hosted Services, Storage Accounts & CDN, in Management Certificates in nato dodali certifikat.

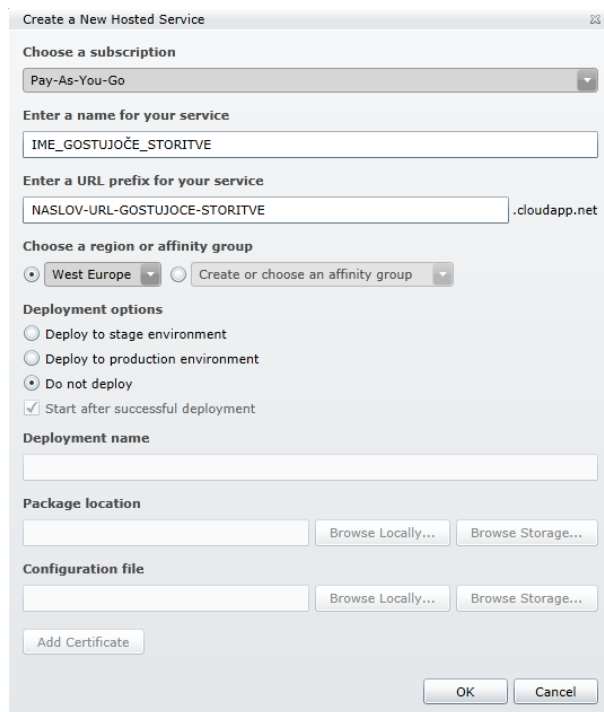


Slika 3.2: Nalaganje certifikata.

3.3.2 Kreiranje gostujoče storitve in diskovnega računa

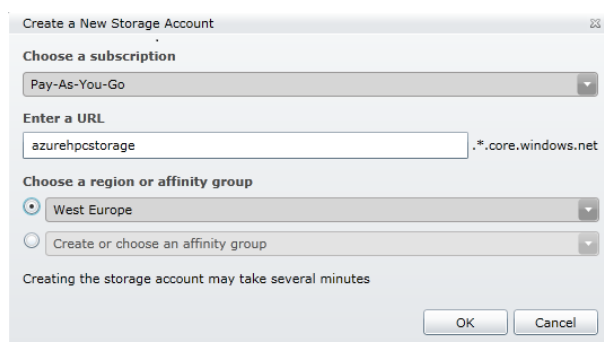
Pred namestitvijo predloge je potrebno urediti gostujočo storitev, pri čemer imamo dve možnosti. Lahko uredimo ločeno gostujočo storitev, ki velja za posamezno predlogo, ali uredimo pomnilniški račun, s katerim je zagotovljen dostop več predlogam. Z izbiro Hosted Services in New Hosted Service smo preko vnosnega obrazca vpisali ime gostujoče storitve in poljubni naslov URL. Med regijami smo izbrali Zahodno Evropo, ki za področje

Slovenije zagotavlja najhitrejšo povezavo. Med opcijami uvajanja (ang. Deployment options) smo izbrali Do not deploy, saj bo to kasneje za nas naredil Program HPC Cluster Manager.



Slika 3.3: Kreiranje gostujoče storitve.

Z izbiro Storage Accounts in New Storage Account smo uredili še nastavitve za diskovni račun. V obrazec smo vpisali željen naslov URL, preko katerega dostopamo in izbrali zahodno-evropsko regijo [7].



Slika 3.4: Kreiranje pomnilniškega računa.

3.4 Nastavitve delovne vloge

Potem, ko so urejene nastavitve računov, je potrebno urediti nastavitve sistemskega okolja, na katerem se bodo izvajale delovne vloge. Delovne vloge se namreč izvajajo na enem ali več virtualnih računalnikov v oblaku, ki jih je za to potrebno posebej pripraviti. Na virtualnih računalnikih mora biti nameščen operacijski sistem, ki mora biti ustrezno nastavljen, nameščeni pa morajo biti tudi programi, ki bodo izvajali izbrane naloge.

3.4.1 Izdelava slike za delovno vlogo

Ker nas v našem primeru konkretno zanima visoko-zmogljivo računanje v oblaku in smo si v ta namen že pripravili testne programe, smo si lahko vnaprej pripravili sliko s potrebnimi programi in nastavitvami, ki jo bomo uporabili za delovno vlogo. Za izdelavo slike smo uporabili program Hyper-V [5]. Vstavili smo DVD z operacijskim sistemom Windows Server 2008 HPC Edition, Hyper-V pa nam je izdelal sliko virtualnega diska (VHD ali Virtual Hard Disk), ki smo jo kasneje naložili na Windows Azure. Datoteka VHD predstavlja virtualni trdi disk in vsebuje particije in datotečne sisteme, ti pa direktorije in datoteke in se uporablja kot trdi disk na virtualnih računalnikih. Uporaben je predvsem zato, ker omogoča uporabo različnih operacijskih sistemov na enem gostujočem računalniku in je zato primeren za razvijalce programske kode, ki svoje programe lahko testirajo v različnih okoljih.

S pomočjo čarovnika smo namestili operacijski sistem, pri čemer smo pazili, da smo izbrali angleško verzijo operacijskega sistema, saj Windows Azure, na žalost, ne pozna slovenščine. Posodobili smo operacijski sistem z zadnjimi različicami popravkov in po posodobitvi izključili samodejno posodabljanje. Samodejno posodabljanje bi namreč kasneje lahko povzročilo zaustavljanje in ponovni zagon računalnikov, kar je potrebno preprečiti, saj morajo biti zaustavitve načrtovane in nadzorovane, sicer lahko povzročijo okvaro podatkov.

Integracija računalnika v virtualno okolje se izvede preko posebnih integracijskih komponent za virtualni računalnik v oblaku Windows Azure. Komponente smo našli v Windows Azure SDK. Pri nameščanju komponent je pomembno, da je izbrana prava različica, ki se mora ujemati z različico paketa visoko zmogljivega računanja, v našem primeru je to je SP3. Pred nameščanjem komponent smo namestili še dodatne funkcionalnosti .NET Framework 3.5.1 Features, ki takšno integracijo sploh omogočijo. Med integracijskimi komponentami smo izbrali HPC Pack 2008 R2 for Windows Azure in HPC Pack 2008 R2

MS-MPI Redistributable Package.

Nazadnje smo na našo sliko namestili še programe, narejene v programskem jeziku Matlab, ki smo jih izbrali za izvedbo testa visoko-zmogljivega računanja v oblaku. V tem delu bi lahko izbrali poljubne programe, ki izkoriščajo možnost vzporednega procesiranja. Za dokončno pripravo slike VHD za nalaganje na Azure smo uporabili program `sysprep.exe`, in opcijo `Generalize`, ki sliko pripravi tako, da se lahko uporabi večkrat. Pri uporabi programa je potrebno vedeti, da odstrani vse odvečne podatke, zato pred tem pripravimo zgolj potrebne programe. Iz varnostnih razlogov ga je možno na eni namestitvi uporabiti največ trikrat, sicer pa je uporaba trivialna.

3.4.2 Dodajanje slike

Windows Azure omogoča izdelavo računalnikov na osnovi različnih tipov slik [8]. V postopku izbire tipa slike operacijskega sistema lahko izdelamo novo sliko, ki bo vsebovala zgolj operacijski sistem, lahko pa uvozimo vnaprej pripravljeno sliko VHD, kar opisujemo v našem postopku. Postopek izdelave slike z operacijskim sistemom in programi za izvajanje visoko zmogljivega računanja, ki smo je izdelali s programom `Hyper-V`, smo že opisali v točki 3.4.1. Sliko bi sicer lahko pripravili tudi šele v tem koraku.

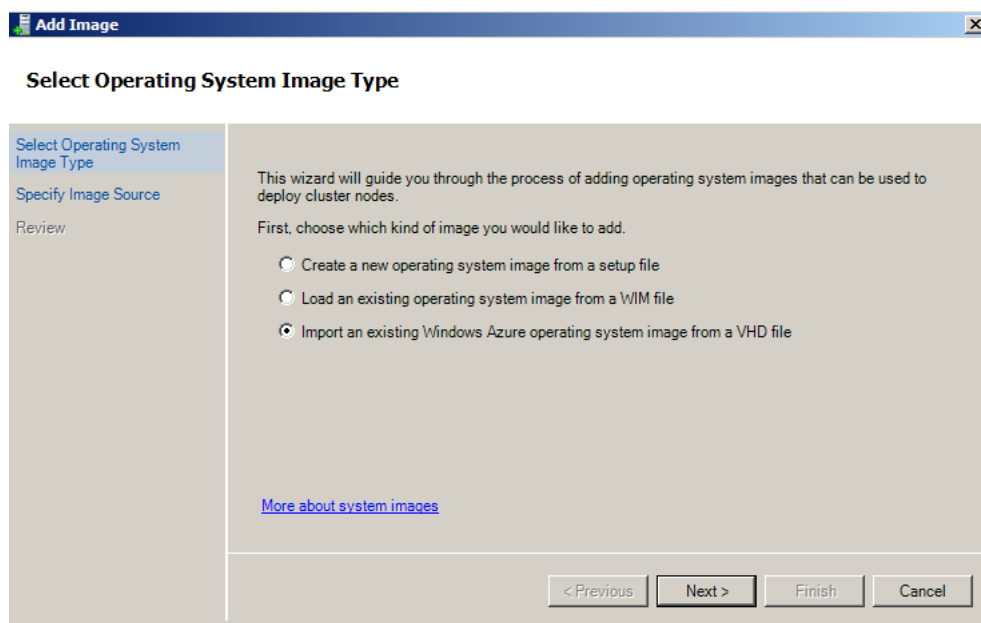
Nalaganje slike smo izvedli preko glavnega računalnika in programa `HPC Cluster Manager`. Preselili smo se torej na glavni računalnik, odprli okno `Images` in izbrali `Add Image`. Izbira te menijske postavke je sprožila zagon pomožnega orodja, tako imenovana čarovnika, ki nas je vodil skozi celoten postopek dodajanja slike in je poskrbel, da smo izvedli vse potrebne akcije in nastavitve.

V prvem koraku se je odprl obrazec za izbiro tipa slike operacijskega sistema. Glede na to, da smo imeli že pripravljeno sliko, smo izbrali opcijo `Import an existing Windows Azure operating system image from VHD file` in po pritisku na tipko `Next` določili pot do naše slike.

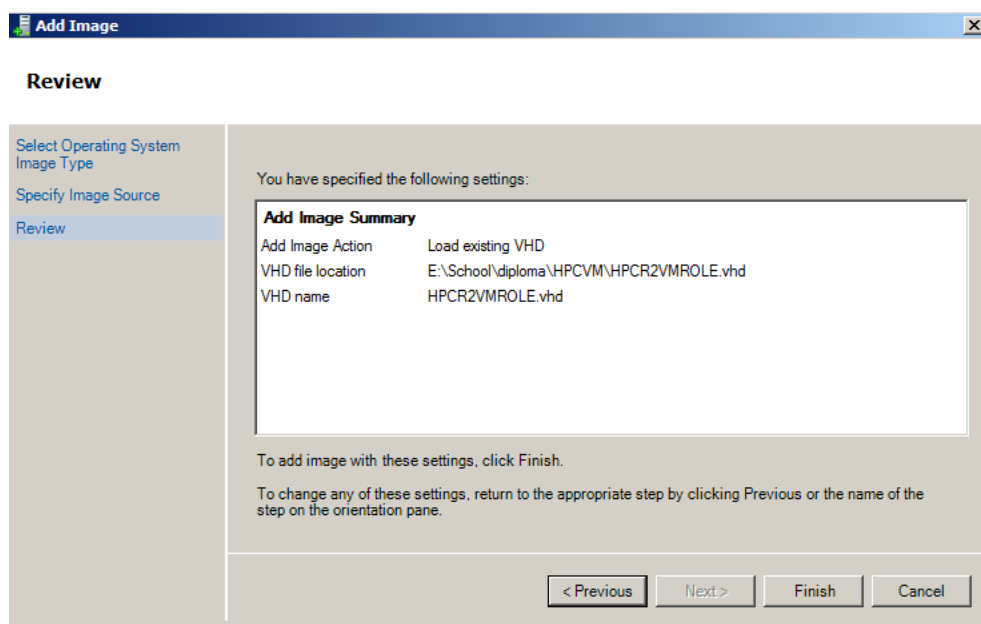
Pred dejanskim uvozom slike nam je čarovnik v pregled prikazal izbrane nastavitve. Izbrali smo uvoz obstoječe slike VHD, ki smo jo pripravili na lokalnem računalniku v mapi in ime `IMESLIKE.vhd`.

V primeru, da z nastavitvami nismo zadovoljni, na primer da smo izbrali napačno sliko ali mapo, se lahko vračamo nazaj z izbiro gumba `Previous` in napako popravimo. Ker smo bili z nastavitvami zadovoljni, smo dodali izbrane slike. Koraka prikazujeta sliki 3.5 in 3.6

V zgornjem okviru smo lahko preverili, da je naša slika prikazana v seznamu dodanih slik,



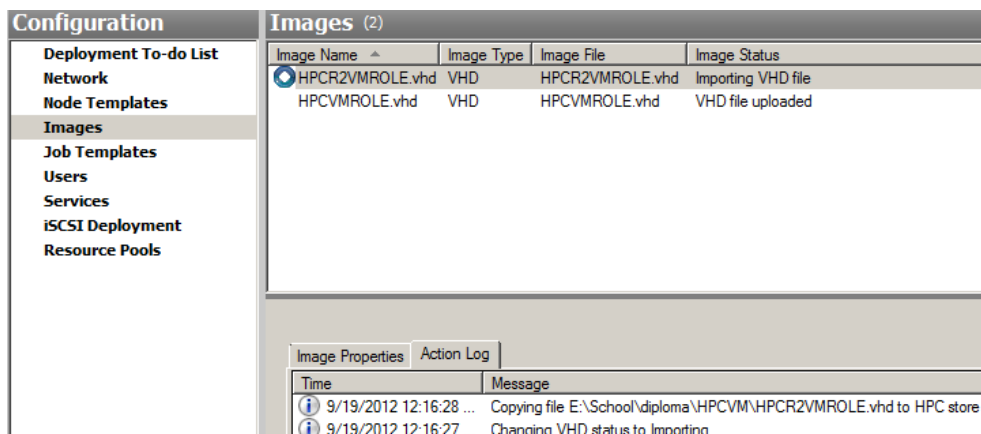
Slika 3.5: Dodajanje slike v HPC Cluster manager.



Slika 3.6: Povzetek dodajanja slike.

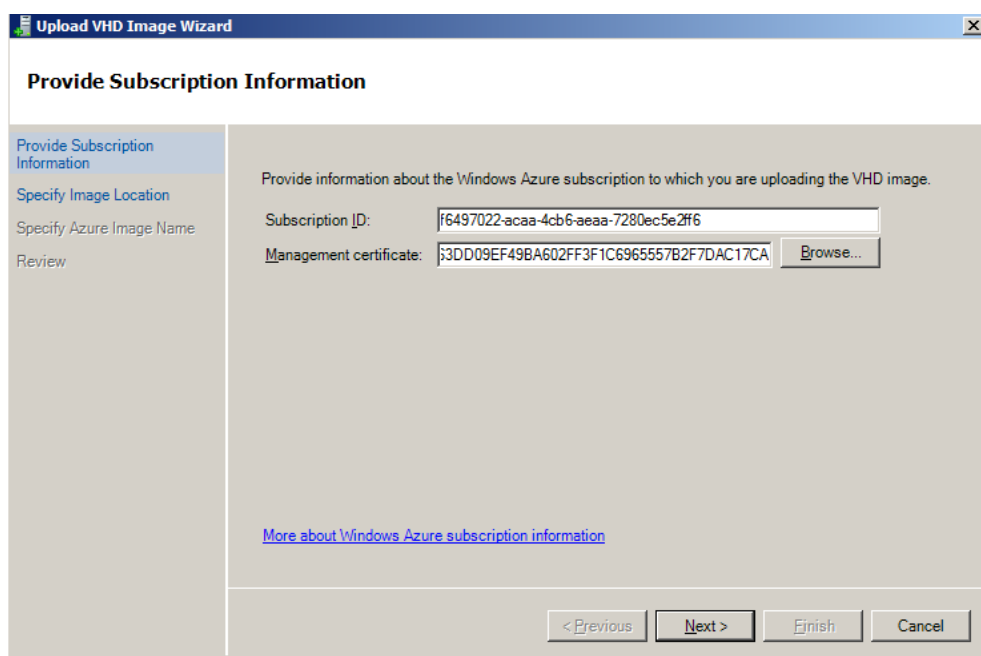
v spodnjem okviru pa smo preverili potek aktivnosti in se prepričali, da pri dodajanju ni prišlo do napak. Na zavihku Image Properties bi lahko spremenili nastavitve in lastnosti izbrane slike.

Z opcijo Upload image smo s pomočjo čarovnika vpisali podatke Subscription ID in Management Certificate. Te smo prekopirali z računa Windows Azure, pri čemer smo za



Slika 3.7: Končano nalaganje slike.

Management Certificate uporabili prstni odtis Thumbprint.



Slika 3.8: Vnos podatkov za povezavo z Windows Azure.

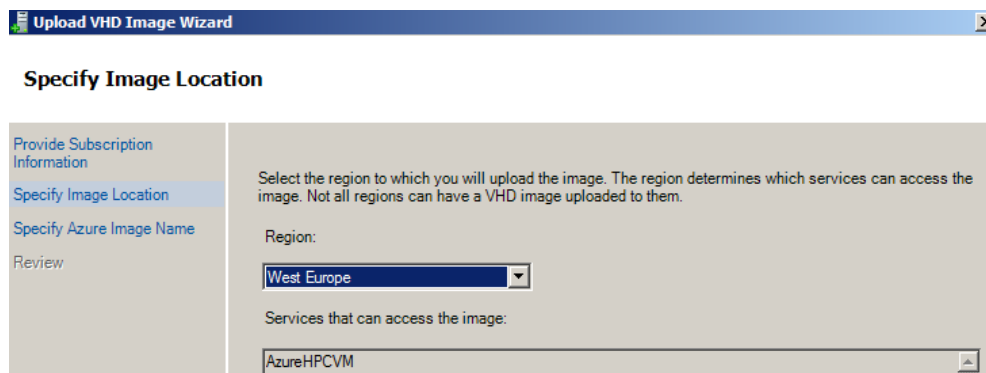
3.4.3 Nalaganje slike

Nadaljevali smo z nalaganjem izbrane slike in sicer tako, da smo v meniju Actions na desni strani zaslona izbrali menijsko postavko Upload image [8]. Tudi za ta del je na razpolago čarovnik, ki nas je vodil po obrazcih za potrebne nastavitve.

Najprej smo morali vpisati svoje naročniške podatke. V polje Subscription ID smo vpisali

identifikacijsko kodo, v polje Management Certificates pa prstni odtis našega certifikata. Oba podatka smo dobili na računu Windows Azure, kjer smo predhodno uredili nastavitve našega računa in pripravili certifikat. Na levi strani Windows Azure namizja smo v meniju izbrali Management Certificates in prekopirali naročniško kodo, ki smo jo prilepili v naš obrazec v programu HPC Cluster Manager. Enako smo naredili s prstnim odtisom.

V nadaljevanju je program preveril naše podatke in nato odprl obrazec, v kateri smo določili lokacijo, kamor smo v naslednjem koraku naložili našo sliko. Regija med drugim določa, katere storitve bo slika imela na razpolago, pri čemer vse regije ne podpirajo slike VHD. Ob izboru regije se v spodnjem okviru izpišejo storitve, ki jih lahko uporabimo. Ponovno smo izbrali zahodno Evropo in dobili na razpolago storitev Azure HPCVM.

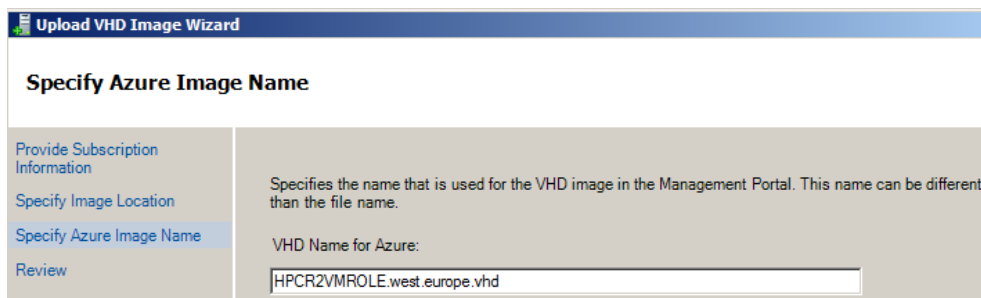


Slika 3.9: Izbira lokacije.

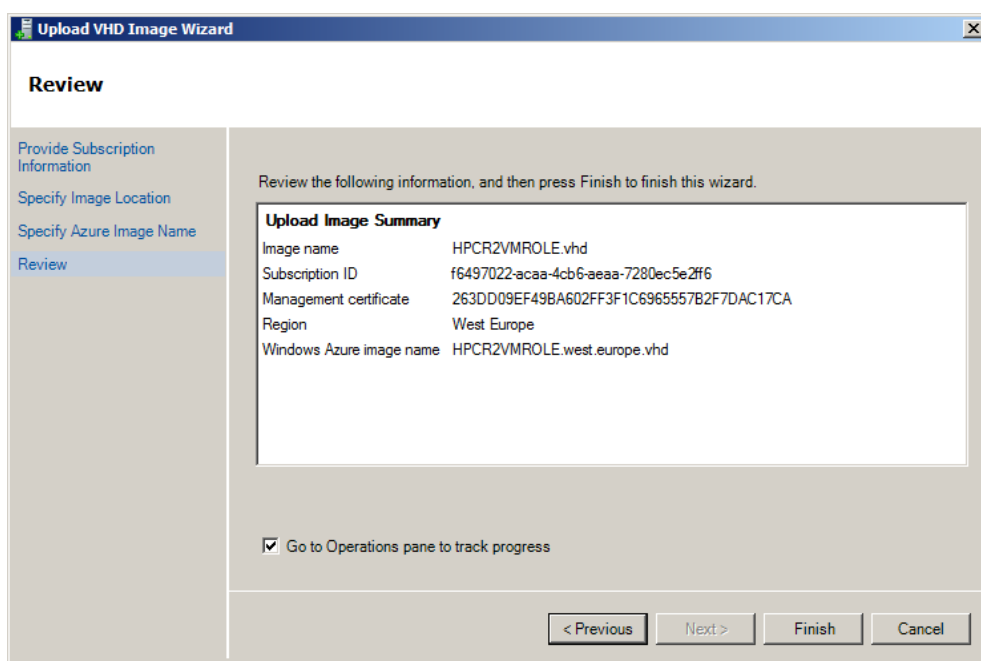
Nato smo določili ime slike. To ime se kasneje uporablja v upravljalnem namizju in se lahko razlikuje od dejanskega imena slike. Odločili smo se, da obdržimo predlagano ime, to je HPCR2VMROLE.west.europe.vhd, ki je smiselno sestavljeno iz obstoječega imena slike in izbrane regije. Pred zaključkom nam je čarovnik prikazal povzetek izbranih nastavitvev, to je imena slike, naročniške identifikacije, prstnega odtisa certifikata, izbrane regije in imena slike v Windows Azure.

Podatke smo pregledali in sprožili nalaganje slike. Za popravljanje nastavitvev bi se lahko s pomočjo gumba Previous preselili v določen obrazec in ponovili postopke do zaključnega dela. V obrazcu smo pustili izbrano možnost Go to Operations pane to track progress, da smo lahko sledili poteku nalaganja slike.

Preden se sproži nalaganje slike je potrebno preveriti omrežno povezavo v smislu pošiljanja (ang. upload), saj so te slike velike več GB in nam slabe povezave in morebitne prekinitve



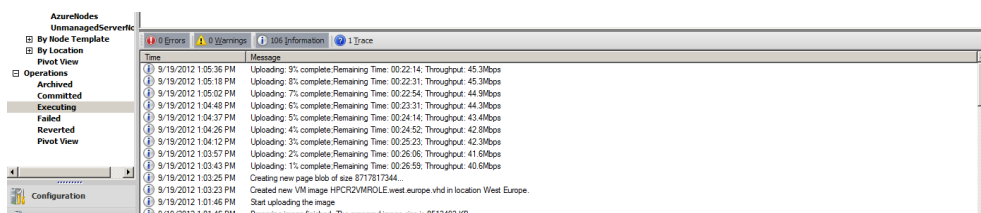
Slika 3.10: Izbira imena slike.



Slika 3.11: Povzetek pošiljanja slike.

močno podaljšajo postopek. Naša slika je bila velika približno 50 GB.

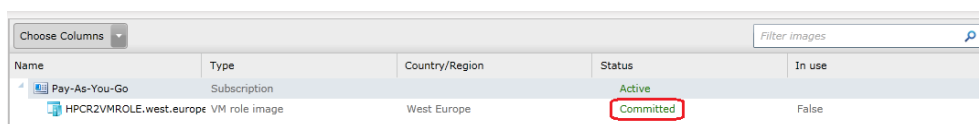
Nalaganje slike smo spremljali v spodnjem okviru zaslona, kjer lahko preko filtrov ločeno opazujemo napake, opozorila in informativna sporočila ob prenosu.



Slika 3.12: Prenos slike.

Ko se je prenos v HPC Clustre Managerju končal, smo na našem Windows Azure računu

pod menijsko izbiro VM Images lahko preverili uspešnost prenosa. Na zaslonu smo videli našo sliko s statusom Committed, kar pomeni, da se je slika uspešno naložila.



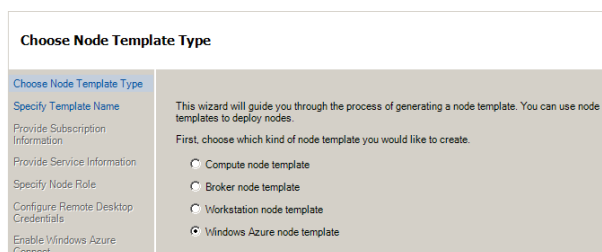
Name	Type	Country/Region	Status	In use
Pay-As-You-Go	Subscription		Active	
HPCR2VMROLE.west.europe VM role image	VM role image	West Europe	Committed	False

Slika 3.13: Pogled naložene slike na Windows Azure platformi.

3.5 Kreiranje predloge

Zadnji korak naše namestitve je bil kreiranje predloge za delovno vlogo [9]. Predloga vsebuje podatke o povezavi našega računa, naročenih storitvah, razpoložljivi politiki in o drugih nastavitvah delovne vloge. Ko so delovne vloge dodane v računalniško gručo, se lahko zaženejo, s čimer se zagotovi vloga Windows Azure instance, ta pa jih nato aktivira za izvajanje nalog v gruči visoko zmogljivega računanja.

Tudi predlogo za delovno vlogo smo izdelali v programu HPC Cluster Manager. V meniju za nastavitve smo izbrali Node Templates in nato opcijo Windows Azure node template in spet nas je čarovnik vodil preko obrazcev za potrebne nastavitve.



Choose Node Template Type

Choose Node Template Type

Specify Template Name

Provide Subscription Information

Provide Service Information

Specify Node Role

Configure Remote Desktop Credentials

Enable Windows Azure Connect

This wizard will guide you through the process of generating a node template. You can use node templates to deploy nodes.

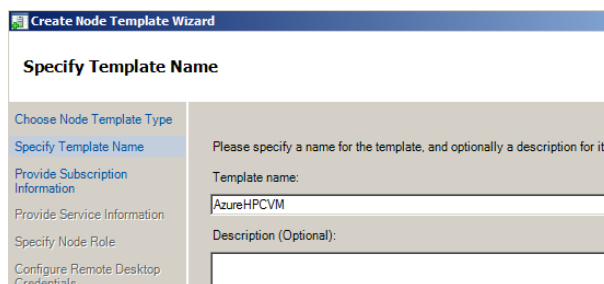
First, choose which kind of node template you would like to create.

Compute node template
 Broker node template
 Workstation node template
 Windows Azure node template

Slika 3.14: Izbira med različnimi predlogami.

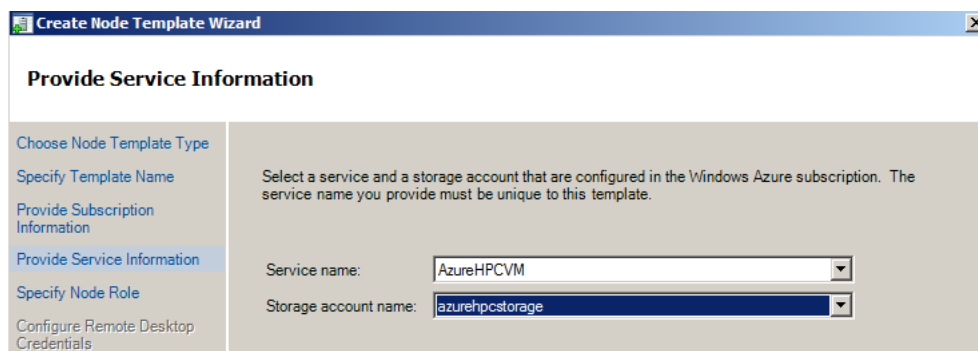
V polje Template name smo vpisali ime predloge, v našem primeru AzureHPCVM, opcijsko bi lahko vpisali tudi opis predloge.

Podobno, kot pri nalaganju slike, smo vpisali svoje naročniške podatke. V polje Subscription ID identifikacijsko kodo, v polje Management Certificates pa prstni odtis našega certifikata.



Slika 3.15: Izbira imena predloge.

Program je preveril naše podatke in odprl obrazec za vnos informacij o storitvah. V polju Service name smo izbrali ime storitve AzureHPCVM, v polju Storage account name pa ime prostora za shranjevanje podatkov azurehpcstorage.

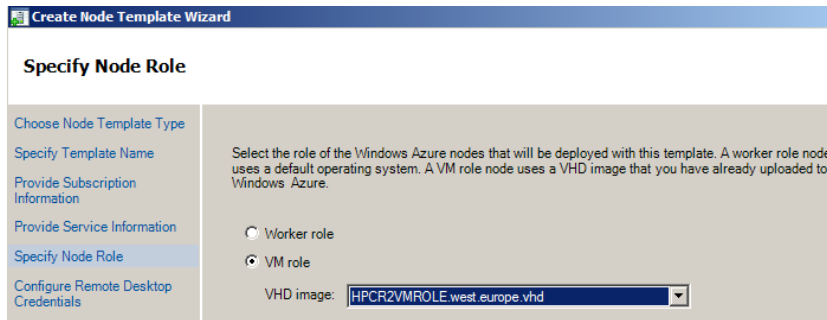


Slika 3.16: Vnos gostujoče storitve in pomnilniškega prostora.

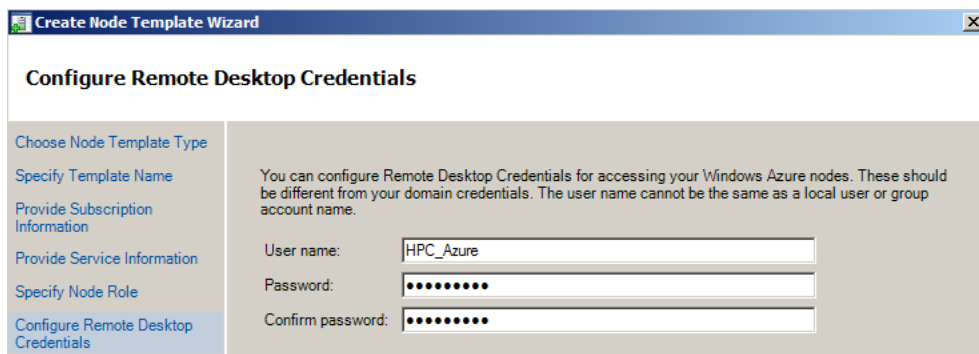
Nato smo določili ime delovne vloge. V obrazcu se izbere tista vloga, ki se nato implementira s to predlogo. Možno je izbrati med delovno vlogo, ki uporablja privzeti operacijski sistem, in vlogo VM, ki uporabi vnaprej pripravljeno sliko virtualnega računalnika. Glede na to, da smo želeli uporabiti že pripravljeno sliko virtualnega računalnika, smo izbrali drugo opcijo. V polju VHD image smo izbrali HPCR2VMROLE.

Sledil je obrazec za nastavitvev dostopa oddaljenega namizja. V tem delu se lahko nastavijo podatki za dostop do Windows Azure sistema preko oddaljenega namizja. Podatki morajo biti drugačni, kot so podatki za dostop do domene. Prav tako mora biti uporabniško ime drugačno, kot je uporabniško ime lokalnega uporabnika ali skupine.

Sledi obrazec za omogočanje povezave z Windows Azure. V tem obrazcu lahko omogočimo Windows Azure povezavi, da poveže računalnike lokalnega omrežja z računalniki,

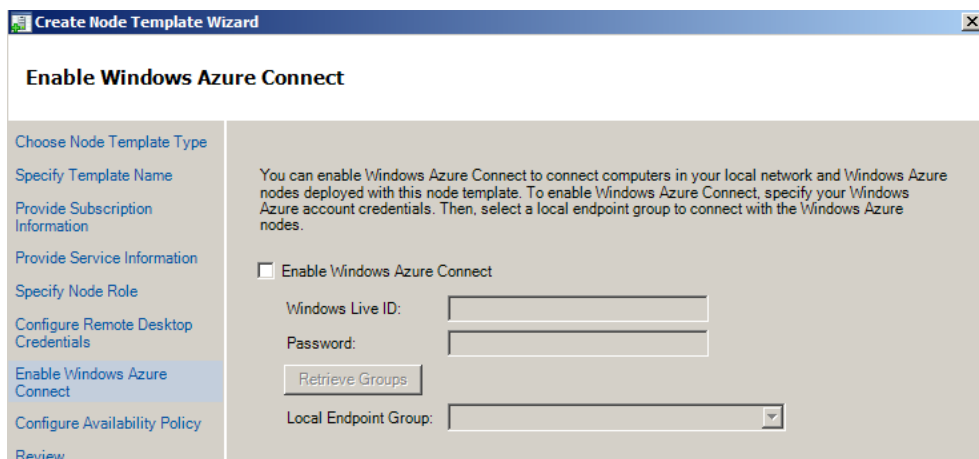


Slika 3.17: Izbira tipa virtualnega računalnika.



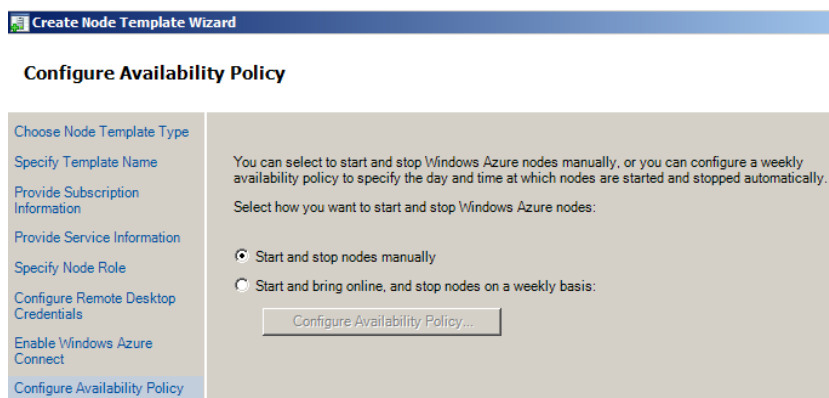
Slika 3.18: Kreiranje uporabniškega imena in gesla za delo na daljavo.

ki bodo naloženi s to predlogo. Če bi želeli to narediti, bi morali na tem mestu vpisati poverilnice za račun Windows Azure in nato izbrati lokalno skupino, ki se bi povezala z Azurjevimi vlogami. V našem primeru tega ne potrebujemo, zato smo ta obrazec pustili prazen. Nadaljevali smo z obrazcem za nastavitvev pravil razpoložljivosti.



Slika 3.19: Omogočanje Windows Azure Connect.

Obrazec za nastavitve pravil razpoložljivosti omogoča nastavitve pravil po vnaprej določenem urniku na tedenski osnovi, pri čemer se določi dan in ura, ko se izbrane vloge avtomatsko zaženejo in zaustavijo. V našem primeru tega ne potrebujemo, zato smo izbrali opcijo Start and stop nodes manually, kar pomeni, da bomo vloge zaganjali ročno.

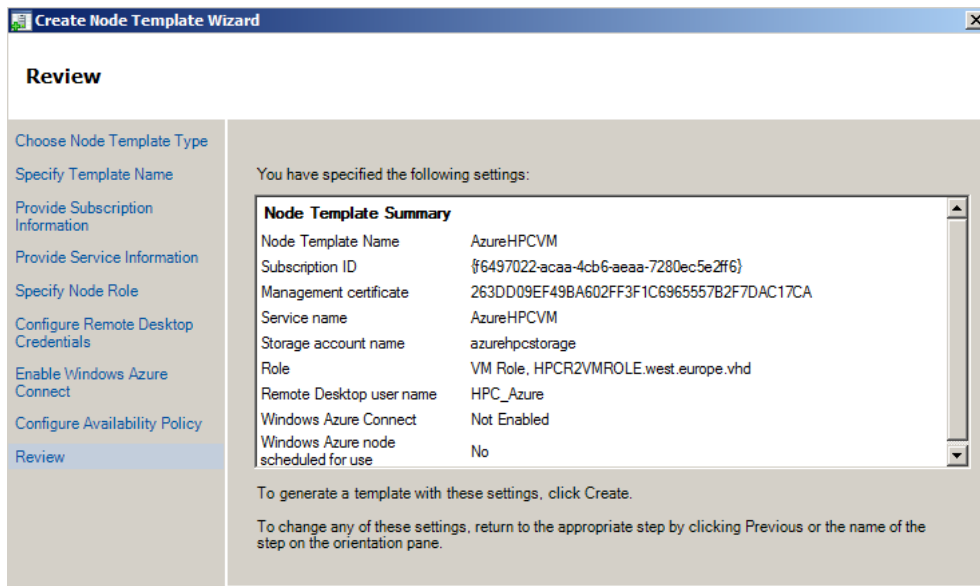


Slika 3.20: Izbira načina zagona.

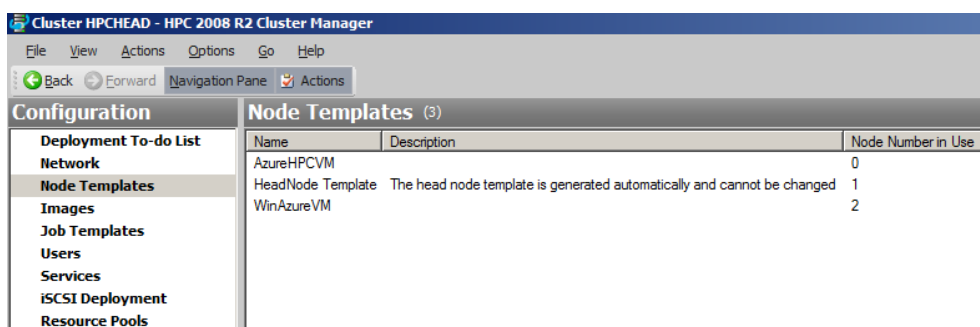
Pred zaključno potrditvijo nam je čarovnik spet prikazal izbrane nastavitve, tako, da smo imeli možnost narediti eventualne popravke. Pregledali smo nastavitve imena vloge, prijavnega identifikacije, imena storitve, imena diskovnega prostora, vloge, prijavnega imena za dostop preko oddaljenega namizja, povezave z lokalnimi računalniki in nastavitve urnika. Nastavitve smo potrdili z izbiro gumba Create.

S tem smo zaključili z nastavitvami. V orodju HPC Cluster Manager smo med predlogami vlog preverili še število posameznih predlog. V desnem okviru smo videli eno predlogo za glavno vlogo, ki je kreirana avtomatsko in je ni mogoče spreminjati in dve predlogi WinAzure VM. Okolje je bilo pripravljeno za izvajanje programov.

Ko smo imeli vse pripravljeno, smo se lotili zaganjanja računalnikov v oblaku. V orodju HPC Cluster Manager smo odprli okno upravitelj vozlišč (ang. Node Management) in izbrali dodajanje računalnikov v gručo Add Windows Azure nodes.



Slika 3.21: Povzetek predloge.



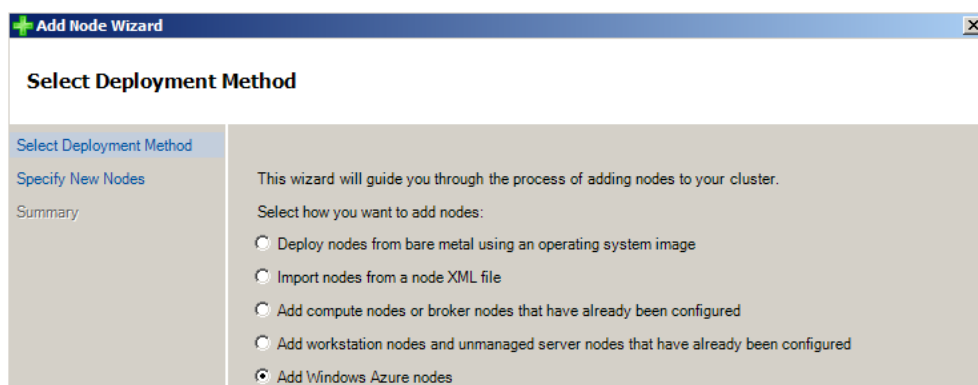
Slika 3.22: Prikaz ustvarjenih predlog.

Poglavje 4

Izvajanje aplikacij

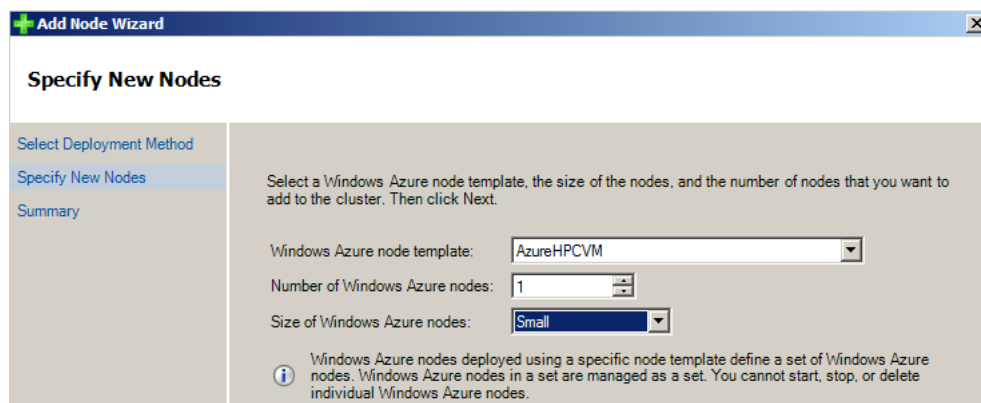
4.1 Zagon računalnikov v gruči

Za zagon računalnika smo v orodju HPC Cluster Manager izbrali akcijo Add nodes in nato v obrazcu izbrali Add Windows Azure nodes [10].



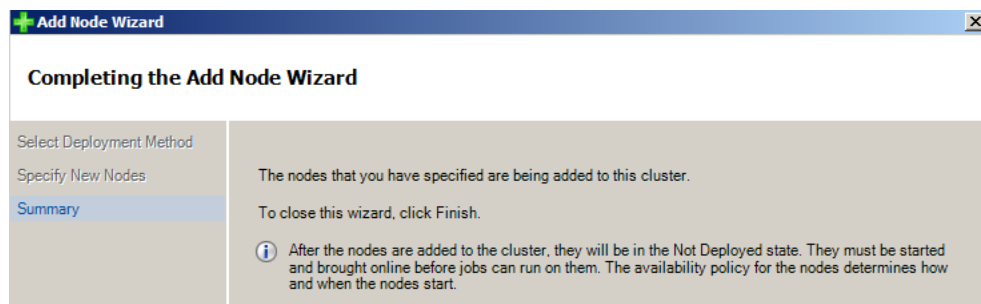
Slika 4.1: Izbira metode delovanja naših računalnikov.

Čarovnik nam je odprl obrazec za specifikacijo novih računalnikov, kjer smo izbrali našo predlogo, ki smo jo definirali v zadnjem koraku nastavitvev. V polju Windows Azure template smo tako izbrali AzureHPCVM, v polju Number of Windows Azure nodes smo izbrali en računalnik in v polju Size of Windows Azure nodes velikost Small, kar pomeni, da imajo računalniki eno jedro in 2 GB RAM-a. Obrazec nas opozarja, da je skupina računalnikov, ki jih zaženemo preko predloge, obravnavana kot ena enota, zato v primeru, da definiramo več računalnikov, ne moremo posebej zaganjati ali ustavljati samo enega med njimi.



Slika 4.2: Izbira velikosti naših računalnikov.

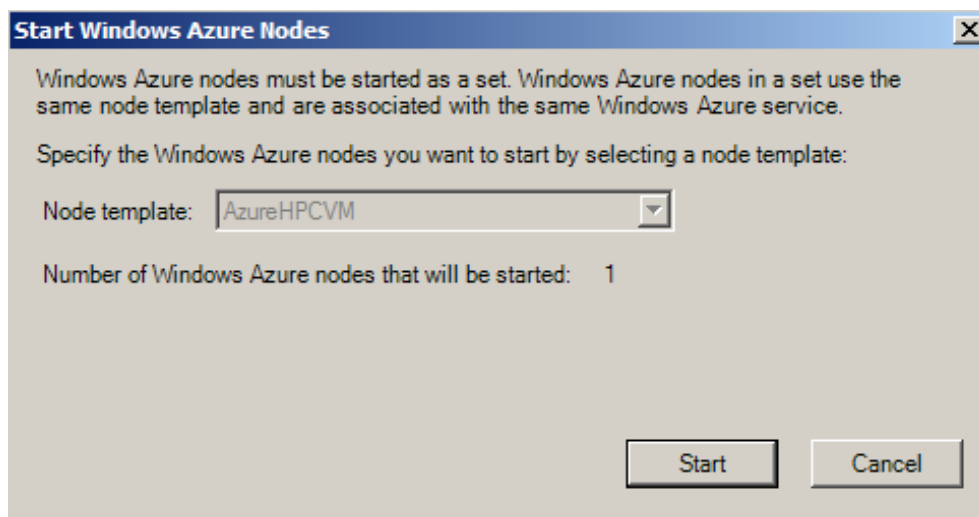
Pred dokončno potrditvijo nas obrazec obvesti, da bomo v gručo dodali določene računalnike in da bodo po vključitvi ostali v statusu, ko še ne bodo aktivni, kar pomeni, da se tudi programi še ne bodo izvajali. Pred-nastavljena politika razpoložljivosti za računalnike določa, kako in kdaj se posamezni računalniki aktivirajo.



Slika 4.3: Potrditev naše izbire.

Zagon računalnika je preprost – s klikom na desni gumb miške se odpre obrazec za zagon, na katerem izberemo gumb Start. Obrazec opozarja, da morajo biti računalniki zagnani kot skupina in da računalniki v skupini koristijo isto predlogo in so povezani z isto storitvijo Windows Azure. V polju Node template je že vpisano ime naše predloge, pod njim pa število računalnikov, ki se bo zagnalo.

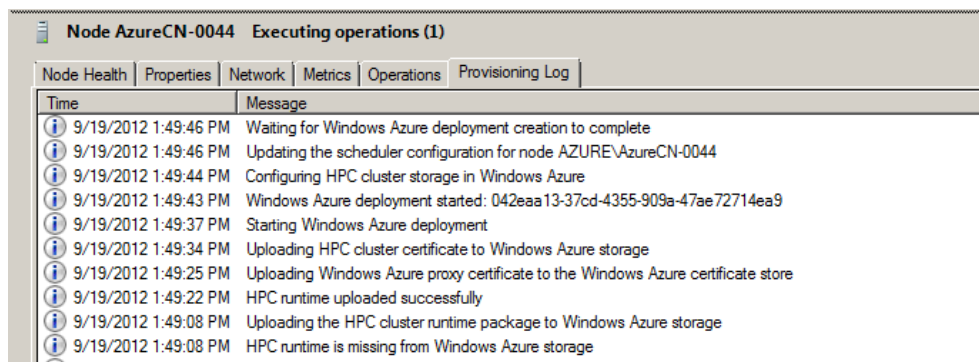
V statusni vrstici vidimo, da so se spremenili statusi za zagnani računalnik. Pod Node State je oznaka Provisioning, kar pomeni, da se računalnik aktivira, pod Node Health je status Transitional in pod Progress je izpisan tekst Moving node AZURECN0044 from



Slika 4.4: Zagon računalnika.

state NotDeployed .. , kar pomeni, da se naš računalnik zaganja.

Potek zagona lahko spremljamo v spodnjem delu zaslona na zavihku Provisioning Log, kjer imamo sicer a razpolago še več zavihkov za kontrolo. V logu se izpisuje čas, tip informacije, ki je v primeru, ko je potek normalen, informativen in nosi oznako i, zraven pa se izpisujejo tudi obvestila. Tako lahko natančno sledimo posameznim aktivnostim.



Slika 4.5: Dnevnik s podatki o računalniku med zaganjanjem.

Status smo lahko preverili tudi na našem Windows Azure računu. V menijskem delu smo izbrali postavko Hosted Services in v desnem delu zaslona dobili pregled gostujočih storitev. Seznam je vseboval ime, tip in status storitve. Videli smo, da ima storitev Deployment for Microsoft Windows statusu kreiranja Creating, po uspešnem zagonu pa

dobi storitev status Ready, kar je prikazano na naslednjih štirih posnetkih zaslona.

Name	Type	Status	Environment
Pay-As-You-Go	Subscription	Active	
AzureHPCVM	Hosted Service	Created	
Certificates			
Microsoft HPC Azure Service	Service Certificate	Created	
Microsoft HPC Azure Client	Service Certificate	Created	
Deployment for Microsoft Windows	Deployment	Creating...	Production

Slika 4.6: Pogled na zaganjanje računalnika v platformi Windows Azure.

Lahko smo spremljali dodajanje novih storitev, to je zagon strežnika HpcSmallVM, ki je prehajal od statusa Initializing in Creating host do Ready. Po pridobljenem statusu Ready je naš računalnik pripravljen za delo. Pomembno je vedeti, da se ob posameznem zagonu virtualnih računalnikov dodatno aktivirata dva posredovalna strežnika (ang. Proxy), v našem primeru HpcProxy_IN_0 in HpcProxy_IN_1.

Deployment for Microsoft Windows	Deployment	Initializing...	Production
HpcProxy	Role	Initializing...	Production
HpcProxy_IN_0	Instance	Creating host...	Production
HpcProxy_IN_1	Instance	Creating host...	Production
HpcSmallVM	VM Role	Initializing...	Production
HpcSmallVM_IN_0	VM instance	Creating host...	Production

Slika 4.7: Zaganjanje računalnika v platformi Windows Azure.

Deployment for Microsoft Windows	Deployment	Transitioning...	Production
HpcProxy	Role	Busy	Production
HpcProxy_IN_0	Instance	Bringing role online... System is initializing.	Production
HpcProxy_IN_1	Instance	Bringing role online... System is initializing.	Production
HpcSmallVM	VM Role	Initializing...	Production
HpcSmallVM_IN_0	VM instance	Starting host...	Production

Slika 4.8: Inicializacija sistema in zaganjanje gostitelja.

Deployment for Microsoft Windows	Deployment	Transitioning...	Production
HpcProxy	Role	Ready	Production
HpcProxy_IN_0	Instance	Ready	Production
HpcProxy_IN_1	Instance	Ready	Production
HpcSmallVM	VM Role	Busy	Production
HpcSmallVM_IN_0	VM instance	Setting up Windows for first use	Production

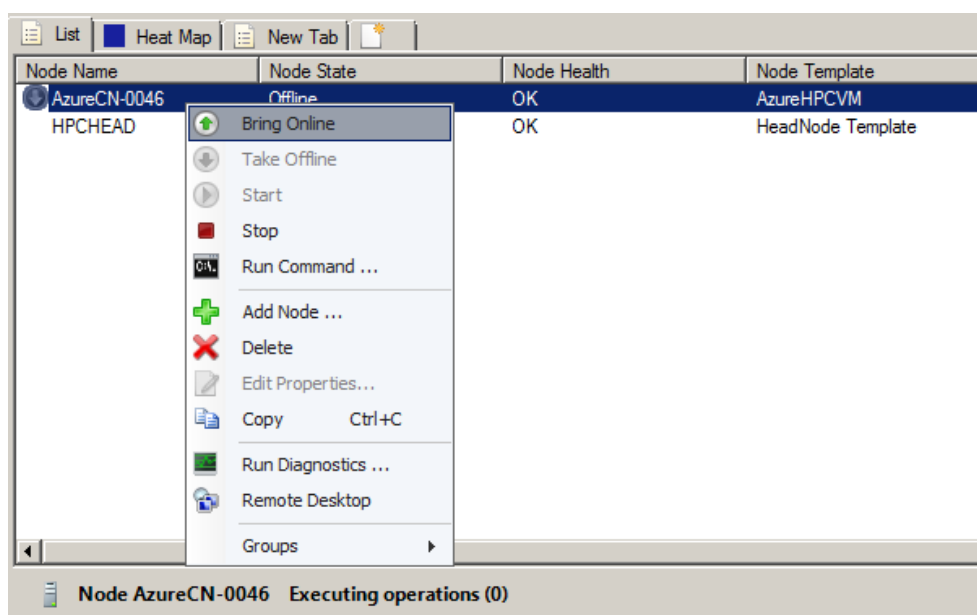
Slika 4.9: Priprava računalnika za prvo uporabo.

Potem, ko smo zagnali virtualne računalnike v našem oblaku, lahko končno zaženemo še pripravljeno vlogo. To naredimo tako, da v orodju HPC Cluster Manager izberemo naš

Deployment for Microsoft Windows	Deployment	Ready	Production
HpcProxy	Role	Ready	Production
HpcProxy_IN_0	Instance	Ready	Production
HpcProxy_IN_1	Instance	Ready	Production
HpcSmallVM	VM Role	Ready	Production
HpcSmallVM_IN_0	VM instance	Ready	Production

Slika 4.10: Prižgani računalniki.

računalnik z izbrano vlogo, ki je trenutno v statusu Offline in jo s pomočjo menija in izbiro opcije Bring Online aktiviramo.

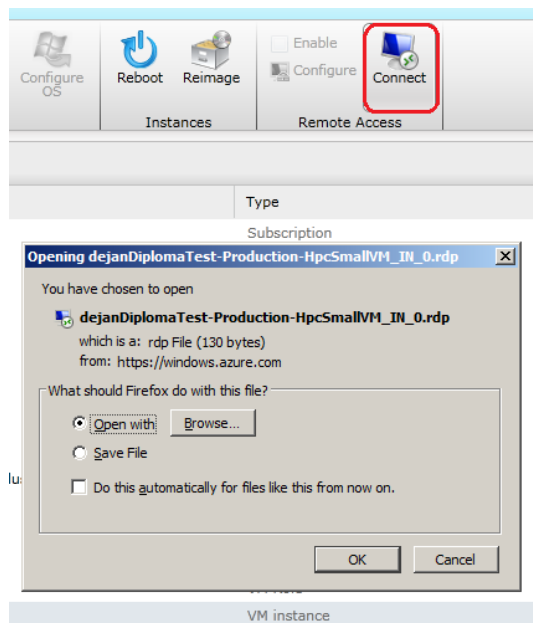


Slika 4.11: Omogočanje delovnih vlog za računanje.

4.2 Ureditev okolja za test aplikacije

Za kontrolo delovanja storitve visoko-zmogljivega računanja smo vnaprej pripravili aplikacijo, ki takšen način računanja podpira v svoji zasnovi. Za izvedbo naloge smo izbrali štiri delovne vloge. Ker naša aplikacija deluje tako, da razdeli računanje na več računalnikov, nato pa rezultate združi, potrebujemo skupno področje, ki bo dostopno vsem računalnikom, ki sodelujejo pri računanju. Odločili smo se, da na eni od delovnih vlog oziroma na enem od virtualnih računalnikov, pripravimo mapo v skupni rabi, v katero bodo posamezni računalniki shranjevali svoje rezultate.

Najprej smo torej v orodju HPC Cluster Manager izbrali delovno vlogo in jo zagnali na štirih virtualnih računalnikih in sicer po postopku, ki smo ga že opisali v točki 4.1. Nato smo nadaljevali na Windows Azure platformi. Povezali smo se na enega od virtualnih računalnikov, tako, da smo v seznamu gostujočih storitev izbrali eno od delovnih vlog in nato gumb »Connect« na desni strani zgornje menijske vrstice.

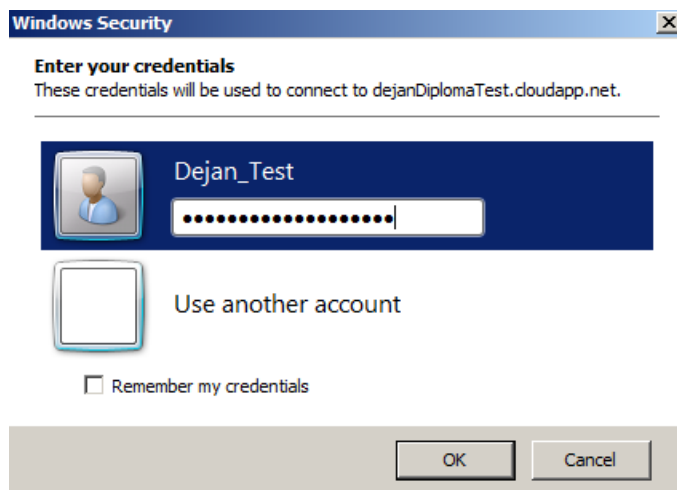


Slika 4.12: Povezovanje na enega od računalnikov v oblaku.

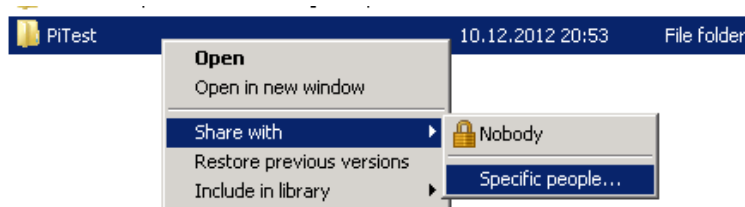
Sistem nas sicer opozori na neznanega izdajatelja, kar je del varnostne zaščite sistema, ki smo jo nastavili v fazi priprave slike delovne vloge. Ker poznamo povezavo, jo potrdimo z izbiro gumba Connect.

Naslednja varnostna zaščita zahteva od nas poverilnice preko uporabniškega imena in gesla. Vpisali smo uporabniško ime in geslo, ki smo ga določili ob ustvarjanju predloge. Prijava poteka podobno, kot z oddaljenim dostopom do fizičnega računalnika, oziroma prijava na fizični računalnik.

Po izvedenih zagonskih programih in nastavitvah smo lahko pričeli za nastavitvami za našo aplikacijo. Naredili smo novo mapo, poimenovali smo jo PiTest – naša testna aplikacija namreč računa število Pi. Mapo smo uredili tako, da smo jo dali v skupno rabo in s tem omogočili dostop iz drugih računalnikov. Z desnim klikom na miški se nam je odprl meni, kjer smo izbrali opcijo »share with« in nato »specific people«.



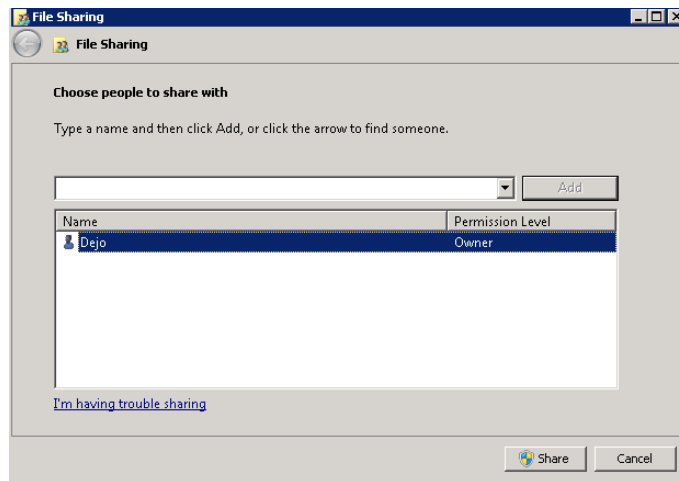
Slika 4.13: Vnos uporabniškega imena in gesla za povezavo na oddaljen računalnik.



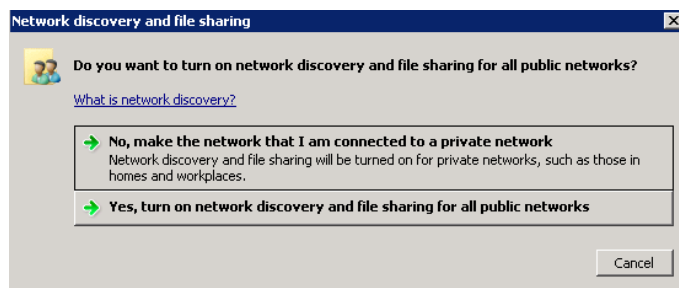
Slika 4.14: Kreiranje mape za skupno rabo.

Dodelili smo pravico skupne rabe za skupino, ki smo jo poimenovali enako, kot naše uporabniško ime. Nastavitev smo potrdili s pritiskom na gumb »Share«.

Omogočili smo skupno rabo za vsa omrežja. S tem je postala mapa PiTest v skupni rabi. Dodatno smo preverili, da je mapa PiTest res dosegljiva tudi na ostalih računalnikih. To smo naredili tako, da smo se prijavili naslednji virtualni računalnik in od tam poiskali mapo. Mapo v skupni rabi lahko uporabljamo preko imena gostujočega računalnika oziroma njegovega naslova IP in imena, ki smo ga določili pri urejanju nastavitev skupne rabe. Uporabili smo IP naslov, ki smo ga prebrali na gostujočem računalniku v komandnem načinu dela, tako, da smo vpisali ukaz `ipconfig`. Mapo smo lahko odprli v brskalniku. V mapo, ki je v skupni rabi, smo sedaj naložili še našo aplikacijo Pi.exe.



Slika 4.15: Dodeljevanje pravic za dostop.



Slika 4.16: Vklop mrežnega raziskovanja.

4.3 Test sistema

Po izvedenih aktivnostih, opisanih v prejšnjih točkah, imamo pripravljeno okolje za test visoko-zmogljivega računanja v oblaku. Štirje računalniki v gruči, ki smo jih naredili na osnovi slike delovne vloge, so prižgani in aktivni v oblaku Windows Azure. Na enem od računalnikov je pripravljena mapa PiTest v skupni rabi, kjer je naložena aplikacija Pi.exe. Pi.exe je aplikacija, ki z numeričnem integriranjem funkcije

$$f(x) = \frac{4}{1+x^2}$$

na intervalu $[0, 1]$ računa število π . Pripravljena je tako, da se integracija lahko izvaja vzporedno na več porazdeljenih računalnikih.

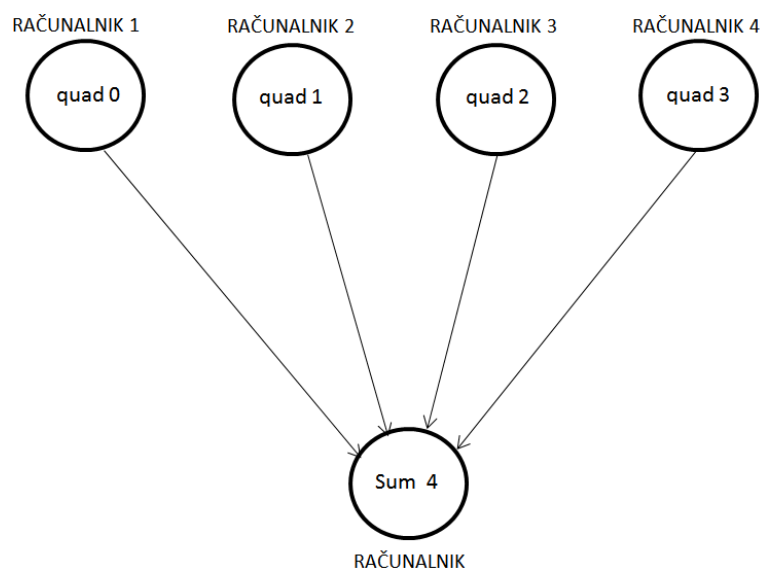
Kot prikazuje slika 4.17 program Pi.exe v prvi fazi izračuna delne integrale in v drugi fazi združi delne vsote v končni rezultat. Za izračun delnih integralov se v argumentih opredeli ime opravila quad, zaporedna številka indeksa, število procesorjev in aproksimacija

```
pi.exe quad <index> <procNo> <accuracy> .
```

Program rezultat zapiše v datoteko `output_<index>.txt`. Za končni izračun se v argumentih opredeli ime opravila `sum` in število procesorjev

```
pi.exe sum <procNo> .
```

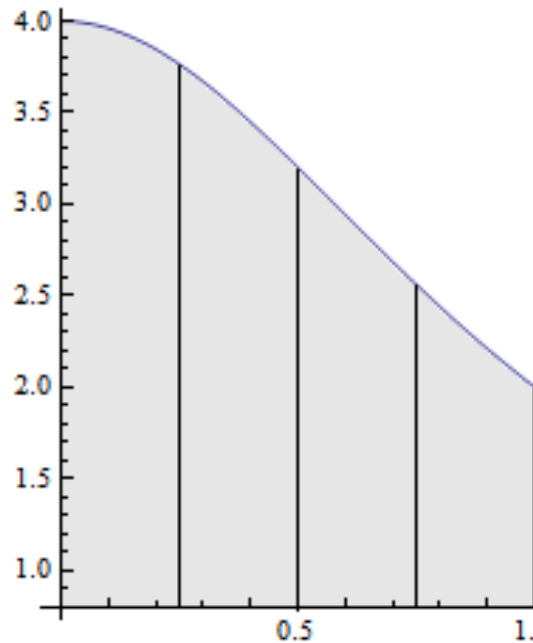
Program prebere rezultate delne integracije iz datotek `output_0.txt` do `output_<procNo>.txt` in jih sešteje. Rezultat shrani v datoteko `output_PI.txt`.



Slika 4.17: Usmerjeni neciklični graf za računanje števila π .

Za zagon aplikacij oziroma nalog v gruči HPC je potrebno imeti nameščene pomožne programe za upravljanje klientov, ki jih vsebuje HPC Pack 2008. Med programi je tudi integrirana aplikativna platforma Job Manager, ki omogoča zagon, upravljanje in razvoj paralelno delujočih aplikacij in predstavlja primarni vmesnik za izvajanje nalog v gruči. Z njim je mogoče z enega mesta potrjevati, nadzirati in upravljati naloge računanja. Vse te naloge sicer lahko izvajamo tudi z ukazi v školjki HPC Power Shell ali preko spletnega vmesnika HPC Basic Profile.

HPC Job Manager smo odprli tako, da smo med programi izbrali Microsoft HPC Pack in nato HPC Job Manager. Ob prvem zagonu programa smo izbrali glavni računalnik naše gruče in nato pričeli z nastavitvami naloge. Izbrali smo opcijo Job Management. S pritiskom na akcijo New Job se nam je odprl obrazec, v katerega smo vpisali nastavitve naloge.

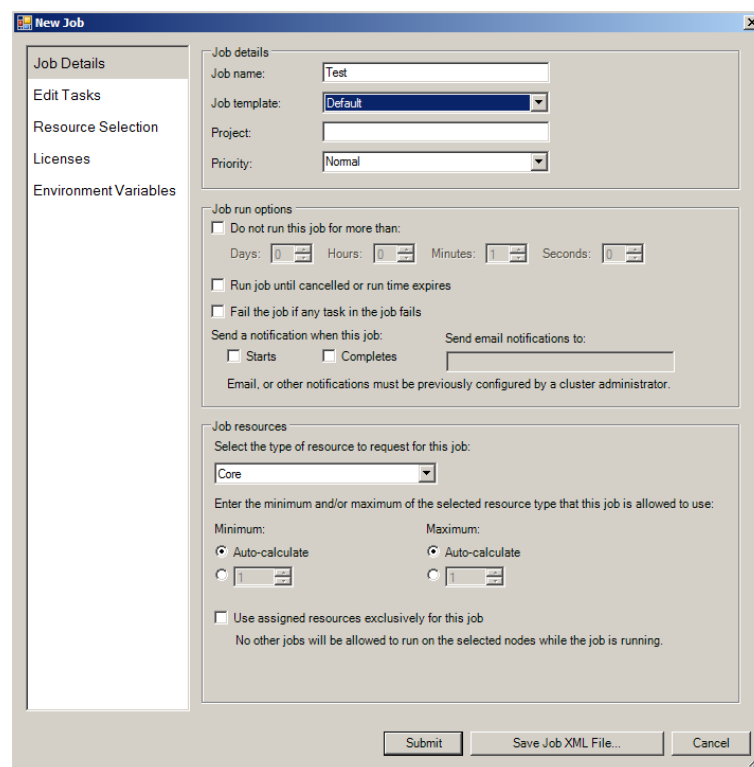


Slika 4.18: Graf funkcije $f(x) = \frac{4}{1+x^2}$ in njen integral $\int_0^1 f(x)dx = \pi$.

V nastavitvah Job Details smo vpisali samo ime naloge, in pustili privzete nastavitve predloge in prioritete. Job run options bi nastavili v primeru, da bi aplikacijo zaganjali po določenem urniku – ker bomo v našem primeru nalogo zagnali ročno, te nastavitve sedaj niso potrebne. Med viri Job resources smo pustili privzeto nastavitvev Core, prav tako smo pustili privzete nastavitve za minimalno in maksimalno število tipov virov, ki jih naloga uporabi, to je avtomatski izračun Auto-calculate. Nismo vključili možnosti Use assigned resources exclusively for this job, saj na računalnikih ne bomo izvajali drugih nalog. Sicer bi z izbiro te možnosti lahko določili, da se v času izvajanje te naloge ne smejo izvajati druge naloge.

V nastavitvah Edit Task smo uredili nastavitve opravil za izvedbo aplikacije v načinu, pri katerem različne procese zaženemo vzporedno z različnimi argumenti. Opravilo je sestavljeno iz več iteracij istega ukaza, ki ima določene vhodne parametre in izhodne datoteke. Opravilo zahteva začetno vrednost, vrednost povečevanja in končno vrednost argumenta.

Naša naloga je sestavljena iz dveh opravil. V prvem opravilu vsak računalnik oziroma delovna vloga izračuna rezultat, v drugem opravilu, pa ena delovna vloga sešteje vse rezultate, kar je končni rezultat.

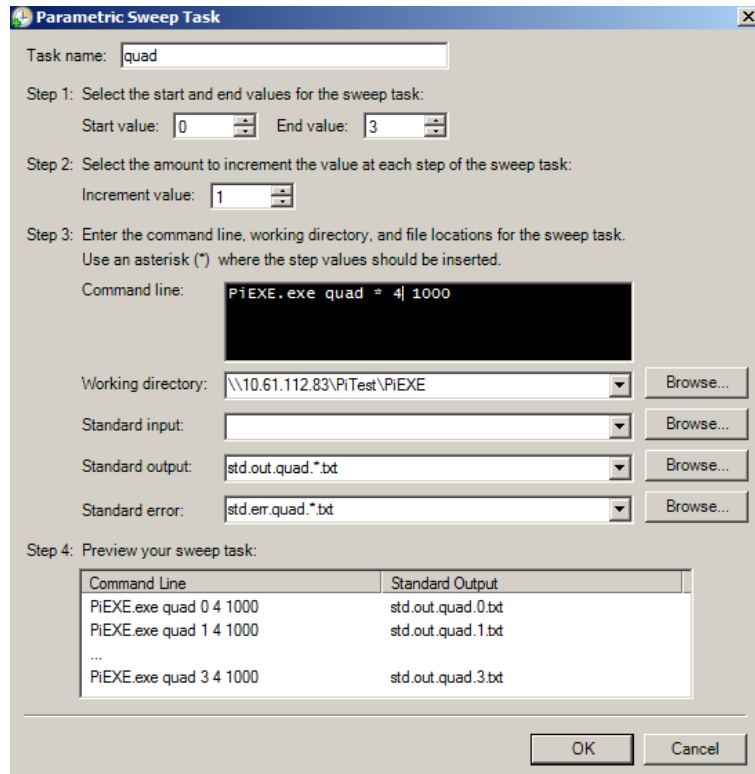


Slika 4.19: Vpis imena aplikacije.

Prvo opravilo smo poimenovali »quad«. Opravilo ima štiri korake [11]. V prvem koraku smo izbrali začetno in končno vrednost iteracij. V polje Start Value smo vpisali številko 0, v polje End value pa številko 3. V drugem koraku smo določili število za povečevanje, to je 1. S tem smo določili, da se bo ukaz izvedel v štirih iteracijah, od 0 do 3. V tretjem koraku smo določili ukazno vrstico, delovno mapo, standardni izhod in standardno napako. Ukazna vrstica vsebuje ime programa, to je Pi.exe in ustrezne argumente, ki jih zahteva program. Argument »quad« določa, da gre za računanje delnega integrala. Zvezdica se ob zagonu interpretira v zaporedno številko iteracije oziroma indeksa. Četrty korak je pregled posameznih komand in standardnih izhodov.

Določili smo še drugo opravilo za seštevek vseh rezultatov, kjer smo prav tako uporabili program Pi.exe, tokrat z argumentom »sum«. Opravilu smo določili ukazno vrstico, delovno mapo, standardni izhod in standardno napako. To opravilo se izvede samo enkrat, zato smo ga definirali kot navadno opravilo.

V nastavitvah Task Dependency for Job smo določili še odvisnost med obema opraviloma. To smo naredili tako, da smo postavili takšno pravilo, da se drugo opravilo ne sme zagnati, dokler se uspešno ne zaključi prvo opravilo. S tem smo preprečili morebitne

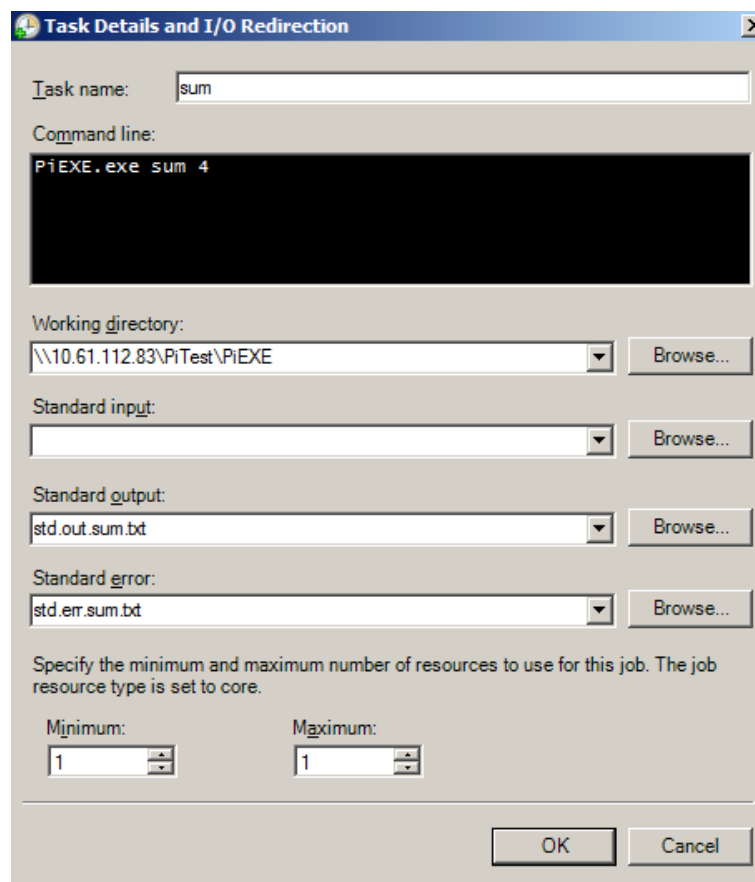


Slika 4.20: Konfiguracija prvega opravila naloge.

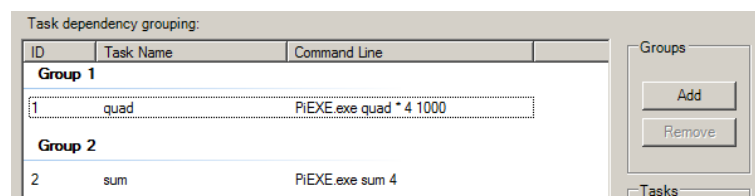
napačne rezultate, do katerih bi prišlo, če bi se drugo opravilo izvedlo preden bi se končalo računanje iz prvega opravila.

Za izvedbo naloge smo izbrali štiri delovne vlog, ki smo jih pred tem zagnali. Naloga se je pričela izvajati s pritiskom na gumb Submit.

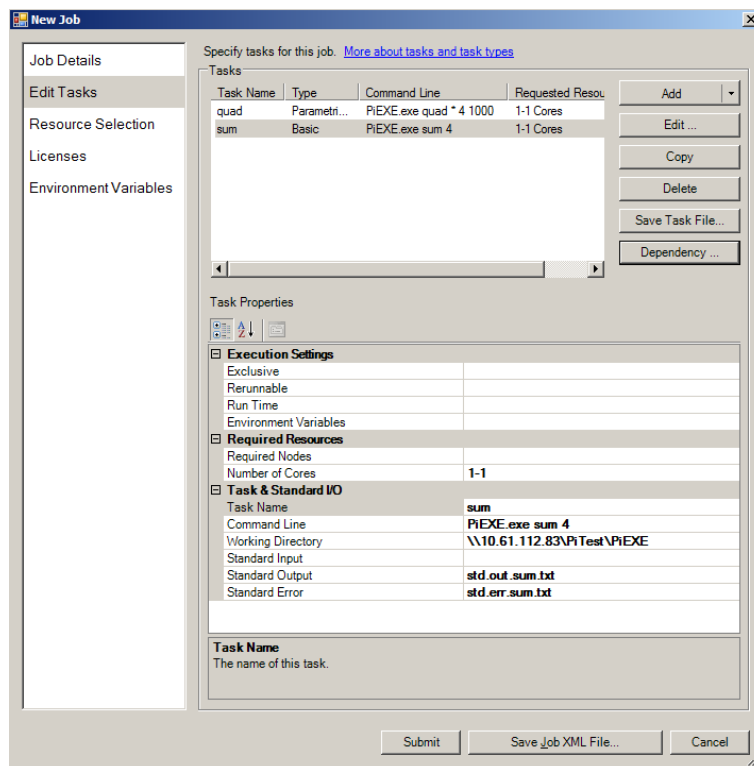
Na monitorju nalog smo lahko opazovali morebitne napake, ki se lahko pojavijo, če slučajno, katera od vloga ni dosegljiva. Naš program z ID-jem številko 116 se je pravilno izvedel, zato smo se lahko ponovno prijavi na eno od delovnih vlog, pogledali mapo, ki je v skupni rabi in prebrali rezultat, ki ga je program zapisal v tekstovno datoteko.



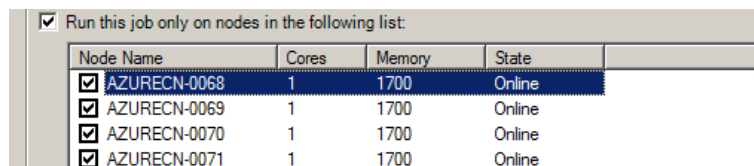
Slika 4.21: Konfiguracija drugega opravila naloge.



Slika 4.22: Določanje odvisnosti med opravili.



Slika 4.23: Pogled na nastavitve naloge za izvedbo naše aplikacije.



Slika 4.24: Izbira delovnih vlog za računanje.

4.4 Rudarjenje besedil

Visoko-zmogljivo računanje se v praksi vse bolj uporablja tudi na področju strojnega učenja in umetne inteligence [12]. Gre za tako imenovano nadzorovano učenje, ki pomeni iskanje pravil v določeni skupini podatkov in uporabo teh pravil pri klasifikaciji novih podatkov. Za iskanje pravil oziroma lastnosti podatkov in ugotavljanje korelacije med njimi se uporabljajo različne tehnike, med bolj uveljavljenimi je metoda podpornih vektorjev (ang. Support Vector Machine, SVM). Ta metoda je v uporabi tako pri razpoznavanju visoko-dimenzionalnih podatkov, kot so na primer slike objektov, kot tudi na področju rudarjenja besedil.

Danes je kar 80 % vseh razpoložljivih informacij shranjenih v obliki besedil, količina informacij pa z vse večjo uporabo interneta in informacijsko-komunikacijskih tehnologij narašča eksponentno. Težava izvajanja nalog pri metodah, kot je SVM je, da za velike količine podatkov zahtevajo izredno veliko računanja, torej veliko procesorsko moč in veliko časa. Zato uporaba porazdeljenega načina računanja v računalniških gručah, kot ga omogoča v prejšnjem poglavju opisani Windows HPC Server in Windows Azure, pomeni bistveno pohitritev izvajanja takšnih nalog.

Za potrditev te predpostavke smo na naši gruči za visoko zmogljivo računanje izvedli test klasifikacije dokumentov v besedilnem korpusu (ang. The Reuters Corpus Volume 1, RCV1). To je prosto dostopna zbirka 806.791 dokumentov novic v angleškem jeziku, namenjena raziskovalnim skupinam. Zbirka RCV1 je zasnovana na principu dokumentov XML, kar zagotavlja večjo konsistenco. Kategorije so urejene hierarhično. Štiri glavne skupine pokrivajo področje industrije, ekonomije, vlade in trgovine in so naprej razdeljene v 103 kategorije. Z vsebinskega vidika se klasifikacija besedil izvede preko posameznih nalog, ki jih predstavlja slika 4.25.

V zaporedni implementaciji je vsaka naloga izvedena s samostojnim izvršljivim programom, ki je sposoben procesirati vnaprej definirane vhodne datoteke in shranjevati rezultate v ustrezne izhodne datoteke. Pri izvajanju programov v porazdeljenem okolju, kjer izkoriščamo paralelno računanje v računalniški gruči, se celoten postopek razdeli v zaporedje posameznih programov, ti pa se še naprej delijo v posamezna opravila, ki izvajajo računanje na določenih segmentih podatkov in rezultate smiselno združujejo. Pri tem je potrebno dodatno identificirati in upoštevati medsebojno odvisnosti med opravili. Odvisnost med opravili prikazuje slika 4.26, kjer točke s črkovnimi oznakami predstavljajo posamezna opravila iz sklopa nalog, opisanih na sliki 4.25, usmerjene puščice pa nakazujejo povezave in odvisnosti med posameznimi opravili.

Oznaka	Opis
A	branje besedilnih datotek, zbiranje podatkov
B	določanje korenov besed, odstranjevanje pogostih besed
C	gradnja slovarja
D	odstranjevanje besed, ki se v slovarju redko pojavljajo
E	določanje uteži besedam v slovarju
F G	gradnja učne množice in določanje parametrov izbranega modela
F H	gradnja testne množice in preverjanje pravilnosti delovanja modela

Slika 4.25: Pregled nalog pri klasifikaciji besedil.

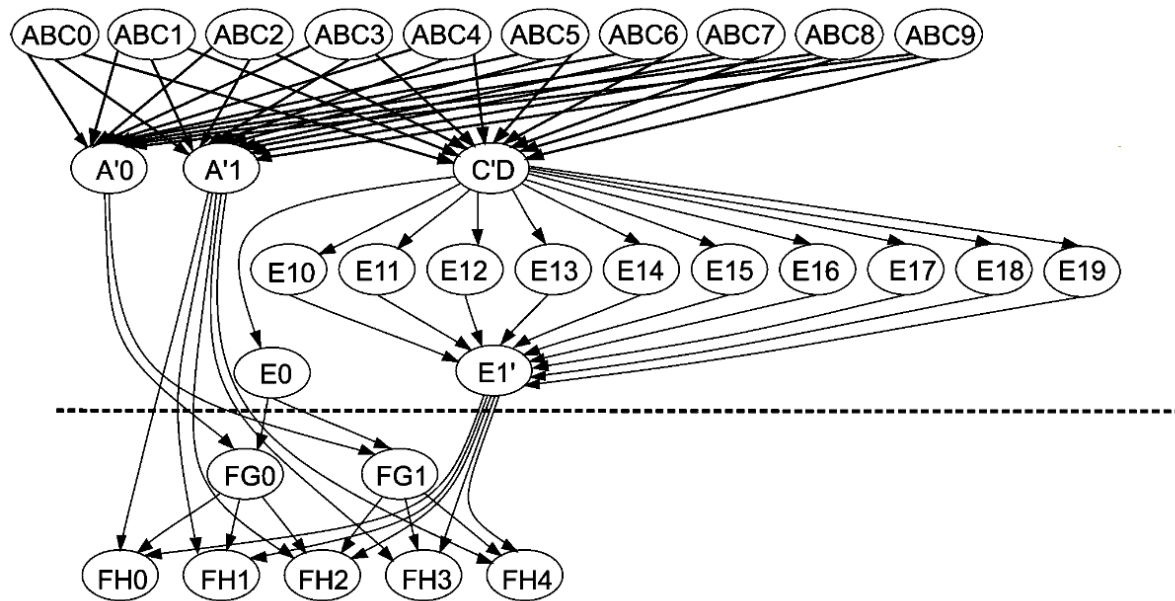
Za predstavitev dokumentov je uporabljen nabor besed iz zbirke RCV1, kjer vsaka posamezna beseda predstavlja lastnost ali ključni izraz, njena vrednost pa predstavlja moč te lastnosti. Določanje uteži besedam se lahko predstavi v obliki enačbe:

$$u(w_i) = tf(w_i) \times \log\left(\frac{|D|}{df(w_i)}\right) \quad ,$$

- w_i predstavlja besedo v slovarju,
- $tf(w_i)$ pomeni število pojavitev besede w_i v dokumentu,
- $|D|$ je število dokumentov,
- $df(w_i)$ je število dokumentov, v katerih se beseda w_i pojavi.

Metoda podpornih vektorjev je matematični postopek, ki se uporablja v fazi učenja in sicer tako, da analizira podatke, prepoznava vzorce in korelacije med podatki in na osnovi pridobljenih rezultatov določi klasifikatorje. Vsak dokument je opisan z vektorjem uteži besed.

Ob analizi toka klasifikacije se določijo splošna opravila klasifikacije besedila od A do H, predstavljenih na sliki 4.25, ki se nato implementirajo v porazdeljeno okolje. Število paralelnih opravil za posamezne naloge je izbrano tako, da je doseženo ravnotežje med obremenitvijo računanja in komunikacijskega prenosa pri delitvi.



Slika 4.26: Odvisnost med stanji pri rudarjenju besedil z modelom SVM na podatkovnem korpusu RCV1.

- Naloga zajema podatkov A, določanja korenov besed B in gradnje slovarja C so z optimizacijo algoritmov združene v eno nalogo ABC, ta pa je z namenom izkoriščanja gruče za visoko zmogljivo računanje razdeljena v deset paralelnih opravil, od ABC0 do ABC9.
- Rezultati posameznih paralelnih opravil naloge ABC se združijo v opravilih A'0 in A'1. Nalogi ABC sledi naloga C'D, ki je sestavljena iz opravila združevanja podatkov izgradnje slovarja C' in naloge D, to je odstranjevanja besed, ki se v slovarju redko pojavijo.
- Nalogi C'D sledita dve nalogi določanja uteži. Naloga E0 je namenjena izgradnji podatkov za učenje. Naloga E1 pa izgradnji testnih podatkov. Naloga določanja uteži E1 je z namenom pohitritve na račun izkoriščanja HPC gruče razdeljena v deset paralelnih opravil, od E10 do E19, ki jim sledi opravilo združevanja podatkov E'1'.
- Podobno, kot v primeru nalog ABC in C'D, je tudi naloga izgradnje učnih in testnih podatkov F združena z nalogo učenja G v nalogo FG in z nalogo testiranja H v nalogo FH. Naloga učenja FG se izvaja po zaključenih nalogah zbiranja podatkov A'0 in določanja uteži E0 v dveh paralelnih opravilih FG0 in FG1, od katerih vsaka

obdeluje pet kategorij.

- Po zaključenih nalogah zbiranja podatkov A'1, določanja uteži E1' in učenja FG, se izvede še zadnja naloga FH, to je testiranje klasifikatorjev na testnih podatkih. Naloga FH je razdeljena v pet paralelnih opravil, od FH0 do FH4, pri čemer vsako opravilo obdeluje dve kategoriji.

4.4.1 Analiza meritev

Testno smo izmerili čase delovanja programov v sekvenčni implementaciji, ki je tekla na virtualnem računalniku v oblaku in jih primerjali s časi v paralelnem načinu delovanja. Za sekvenčno implementacijo smo uporabili računalnik z 1,6 GHz procesorjem in 1,75 GB RAM pomnilnika, za paralelno delovanje smo uporabili 10 računalnikov, vsak z 1,6 GHz procesorjem in 1,75 GB RAM pomnilnika. Naredili smo pet meritev na vsakem sistemu in analizirali rezultate, ki so predstavljeni na sliki 4.27

Analiza meritev	Čas sekvenčnega izvajanja [s]	Čas paralelnega izvajanja [s]	Pohitritev	Učinkovitost
Merjenje				
1. merjenje	3.255	805	4,0	0,40
2. merjenje	3.209	823	3,9	0,39
3. merjenje	3.218	825	3,9	0,39
4. merjenje	3.207	832	3,9	0,39
5. merjenje	3.096	922	3,4	0,34
Povprečje	3.197	841	3,8	0,38
Standardni odklon	53	41		

Slika 4.27: Primerjava sekvenčnega in paralelnega sistema.

Iz analize je razvidno, da je povprečni čas sekvenčnega izvajanja programov 3.197 sekund. Standardni odklon, s katerim merimo statistično razpršenost, je majhen, 53 sekund ali

1,67%, kar pomeni, da so meritve koncentrirane okoli aritmetične sredine in niso močno razpršene. Povprečni čas izvajanja programov v paralelnem načinu je 841 sekund. Standardni odklon je 41 sekund ali približno 5 %, kar pomeni nekoliko večjo razpršenost, kot pri meritvah sekvenčnega izvajanja programov in gre predvsem na račun zadnjega merjenja. Pohitritev smo izračunali kot kvocient med povprečnim časom sekvenčnega izvajanja in povprečnim časom paralelnega izvajanja in znaša 3,8. Učinkovitost 0,38 smo izračunali glede na maksimalno število računalnikov, ki smo jih uporabili za paralelno računanje, tako, da smo pohitritev delili z 10. Podatek učinkovitosti je nekoliko slabši, kar je posledica arhitekture programskih sklopov, pri katerih je paralelizem z desetimi strežniki vključen samo pri izvajanju dveh nalog, eno nalogo smo izvedli brez paralelizma, eno smo izvedli paralelno na dveh strežnikih in eno paralelno na petih strežnikih.

Poglavje 5

Zaključek

Do pred nedavnim je bila organizacija prisiljena vlagati veliko energije in sredstev v optimalno načrtovanje in uvajanje informacijskih sistemov, še posebej na področju omrežne in strežniške infrastrukture, kot tudi systemskega in razvojnega programskega okolja. Z naglim naraščanjem količine podatkov za obdelavo in vse večjo informacijsko podporo poslovnim in drugim procesom je bilo potrebno zagotavljati tudi vse večjo razpoložljivost in varnost informacijskih sistemov. Poslovni procesi so v veliki meri postali odvisni od kvalitete delovanja programskih rešitev, ki pa so med drugim odvisni tudi od optimalno urejenega systemskega dela. Razvijalci in skrbniki programskih rešitev morajo poskrbeti za dinamične spremembe pri zagotavljanju programske podpore in pričakujejo, da bodo v vsakem hipu imeli na razpolago zadostno količino procesorskih in spominskih virov za izvajanje svojih programov. Zato morajo systemski skrbniki pri uvajanju strežniških sistemov upoštevati maksimalne možne potrebe po virih, kar pa pomeni, da se kupujejo sistemi, ki niso ves čas v celoti izkoriščeni. V velikih organizacijah je uvedba virtualizacije, združevanje strežnikov v gruče in razvoj omrežja pomnilniških naprav pripomogla k bolj racionalni izrabi teh virov, za manjše firme pa je to lahko resen problem, saj so ti sistemi dragi, neracionalni stroški pa jim znižujejo konkurenčnost na trgu.

Dinamika razvoja in sprememb v programih za podporo poslovnim in drugim procesom je iz dneva v dan večja. Prav tako smo priča vse večjim zahtevam po kvalitetnih informacijah, pridobljenih s pomočjo obdelave velikih količin podatkov, pridobljenih iz različnih virov. Organizacije si na ta način iščejo poslovne priložnosti in konkurenčno prednost na tržišču, zato je naloga računalničarjev, da poskrbimo za hitro prilagoditev novim zahtevam. To pa pomeni, da moramo imeti na razpolago kvalitetne računalniške sisteme, ki zagotavljajo ustrezne systemske nastavitve in strojne vire.

Vzdrževanje in upravljanje strežniških in omrežnih sistemov zahteva dokaj visoka sredstva, pri čemer je potrebno zagotoviti ustrezno razpoložljivost, varnost in učinkovitost, kot tudi primeren prostor, električno moč in hlajenje. Uvajanje sprememb v sistemskem delu je dražje, kot uvajanje programskih sprememb. V primeru, da se pojavi potreba po dodatnih strojnih virih, je takšno povečanje lahko dolgotrajno. Najprej moramo kupiti ustrezno opremo, jo umestiti k obstoječi opremi, pri tem zagotoviti dodatno hlajenje in električno moč, nato moramo namestiti sistemsko programsko opremo in eventualno dodatno opremo, pri čemer moramo poskrbeti, da je sistem umeščen in urejen v skladu z varnostno politiko firme.

Izkušnja s storitvami v oblakih nas je privedla do novih spoznanj o širokih možnostih izkoriščanja naprednih informacijskih tehnologij in optimizaciji izrabe računalniških virov. S testom izvajanja programov za rudarjenje besedil, kjer se uporablja računsko zahtevna metoda podpornih vektorjev, smo dokazali, da je Microsoftova storitev Windows Azure s svojim strežnikom za visoko zmogljivo računanje dobra rešitev tudi na tem področju, saj prinaša znatno pohitritev kategoriziranja besedil. Vse to kaže na velike potenciale storitev v oblaku, ki so lahko tudi konkurenčna prednost, saj poleg pohitritev prinašajo prihranke pri zagotavljanju systemske in programske podpore. Obenem omogočajo razvijalcem programske opreme, da bolj kvalitetno testirajo nove programe in programske spremembe. In še več. Omogočajo tudi dinamično prilagajanje systemskih virov dejanskim potrebam programov, kar pomeni, da uporabnik lahko hitreje pride do potrebnih podatkov in informacij, kot v primeru klasičnih sistemov. Pri tem pa se jim niti ni potrebno ukvarjati s prostorskim problemom, če je potrebno sistem razširiti.

Seveda je pri tem potrebno vložiti tudi nekaj energije. Programske rešitve morajo omogočati visoko-zmogljivo računanje, kar pomeni, da je potrebno programe predelati ali napisati nove. Ker ne moremo pričakovati, da bi se organizacije lahko odpovedale številnim, že uvedenim programom, tudi hiter prehod v takšno okolje morda ni mogoč. Z vidika obrabe sistemov morda tudi neracionalen. Pa vendar je tudi v tem delu uporaba storitev v oblaku lahko smiselna, saj ponuja kombinacijo sistemov v lokalnem okolju s sistemi v oblaku. Tako se organizacija lahko odloči, da bo uporabljala nove storitve v oblakih, stare pa integrirala in jih izvajala v svojem lokalnem delu, ali pa se odloči za postopen prehod.

Namestitev vseh komponent in vlog v Windows Azure se morda zdi na prvi pogled zapleteno. Zahteva vrsto predlog in nastavitve, ki pa so podprte s čarovniki, ki nas vodijo skozi posamezne nastavitve in nam sproti podajajo potrebne informacije. Prav te predloge nam

prihranijo čas pri nadaljnem upravljanju sistema.

Na vprašanje, ali naj organizacija prične uporabljati storitve za visoko-zmogljivo računanje v oblaku, bi bil naš odgovor vsekakor pritrdilen. Seveda pa je potrebno tudi v tem primeru, tako kot pri vsaki večji spremembi informacijskega sistema, narediti natančno analizo obstoječega stanja in zelenih ciljev.

Literatura

- [1] (2010) Windows HPC Server 2008 R2 released . Dostopno na: <http://www.onwindows.com/Articles/Windows-HPC-Server-2008-R2-released/5294/Default.aspx>
- [2] (2011) Windows PC Server and WIndows Azure . Dostopno na: http://www.davidchappell.com/writing/white_papers/Windows_HPC_Server_and_Windows_Azure,_v1.2--Chappell.pdf
- [3] (2010)Prepare Your Windows Server 2008 R2 for Hyper-V Role. Dostopno na: <http://www.petri.co.il/prepare-your-windows-server-2008-for-hyper-v-role.htm>
- [4] (2012)App Services. Dostopno na: <https://www.windowsazure.com/en-us/develop/net/app-services/>
- [5] (2012)Create a VHD for VM Nodes. Dostopno na: <http://technet.microsoft.com/en-us/library/hh184311%28v=ws.10%29.aspx>
- [6] (2012)Configure the Management Certificate for Windows Azure. Dostopno na: <http://technet.microsoft.com/en-us/library/gg481759%28v=ws.10%29.aspx>
- [7] (2012) Create a Hosted Service and a Storage Account in Windows Azure. Dostopno na: <http://technet.microsoft.com/en-us/library/hh314624%28v=ws.10%29.aspx>
- [8] (2012) Upload a VHD to Windows Azure. Dostopno na: <http://technet.microsoft.com/en-us/library/hh184312%28v=ws.10%29.aspx>
- [9] (2012) Create a Windows Azure VM Node Template. Dostopno na: <http://technet.microsoft.com/en-us/library/hh184319%28v=ws.10%29.aspx>
- [10] (2012) Start the Windows Azure VM Nodes. Dostopno na: <http://technet.microsoft.com/en-us/library/hh184309%28v=ws.10%29.aspx>

- [11] Andrej Dobnikar, Uroš Lotrič. "Porazdeljeni sistemi za modeliranje, paralelno programiranje in procesiranje", 2008.
- [12] Catarina Silva, Uroš Lotrič, Bernardete Ribeiro, Andrej Dobnikar. "Distributed Text Classification With an Ensemble Kernel-Based Learning Approach", 2010.
- [13] (2008) Hadoop Map-Reduce Tutorial. Dostopno na: http://hadoop.apache.org/docs/r0.17.0/mapred_tutorial.html