ZFS

Solaris og ZFS som ny hjemmekatalogløsning
for ansatte og studenter ved UiB

UNIVERSITETET I BERGEN

Hva er ZFS?

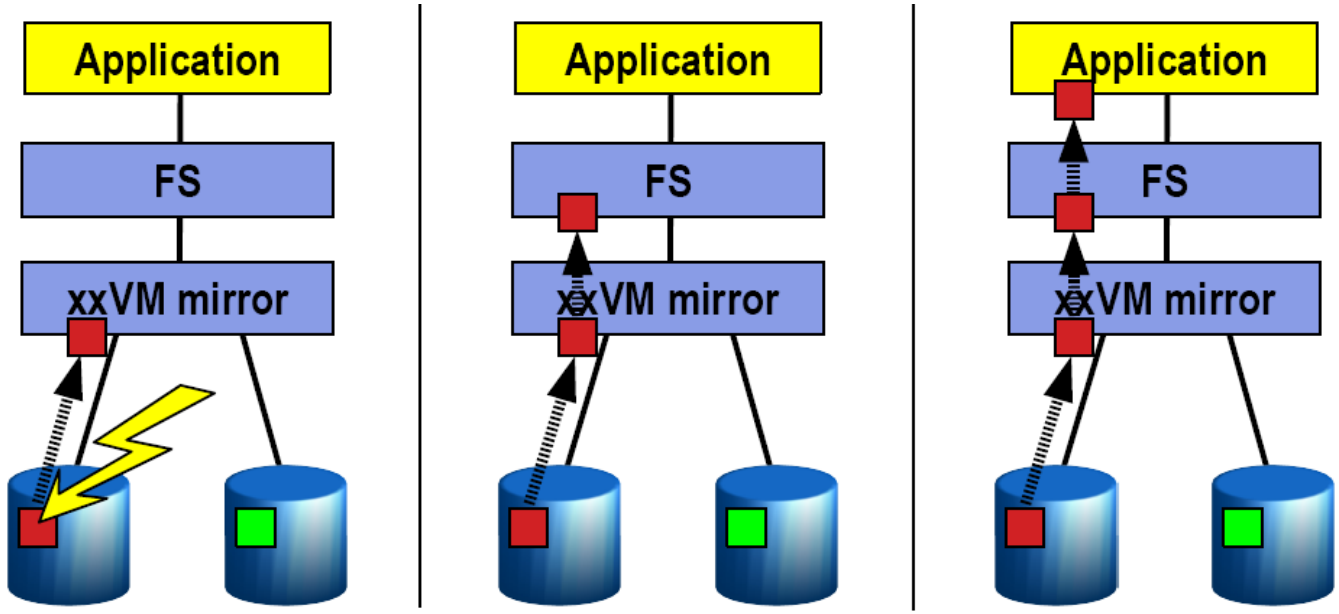
- ZFS (Zettabyte filesystem) er utviklet fra grunnen med en del helt radikale endringer i forhold til eksisterende filsystemer
- Highlights:
  - ZFS er vanvittig skalerbart. Det er et 128-bit filsystem med 16 milliarder milliarder større kapasitet enn kapasiteten til 32/64-bits systemer.
  - Alle data er beskyttet av 64bits checksum'er og data consistency er opprettholdt til enhver tid

- Highlights forts.
  - Hastigheten er ekstremt bra i forhold til dagens løsninger. Dette er noe vi også har sett bevis på her ved UiB, vi har brukt ZFS til /var/mail lenge og det fungerer veldig bra
  - Ekstremt forenklet fra et systemansvarlig-ståsted i forhold til tidligere volume managere, gui'er, soft-partisjoner osv osv
  - POSIX compliant, noe som betyr at alle applikasjoner virker med det uten endringer i applikasjonene
  - ZFS er "pooled storage" noe som betyr at flere filsystemer deler på diskplassen som er lagt til i pool'en. Legger man til mer disk i en pool, får alle filsystemene i pool'en automatisk tilgang på det.

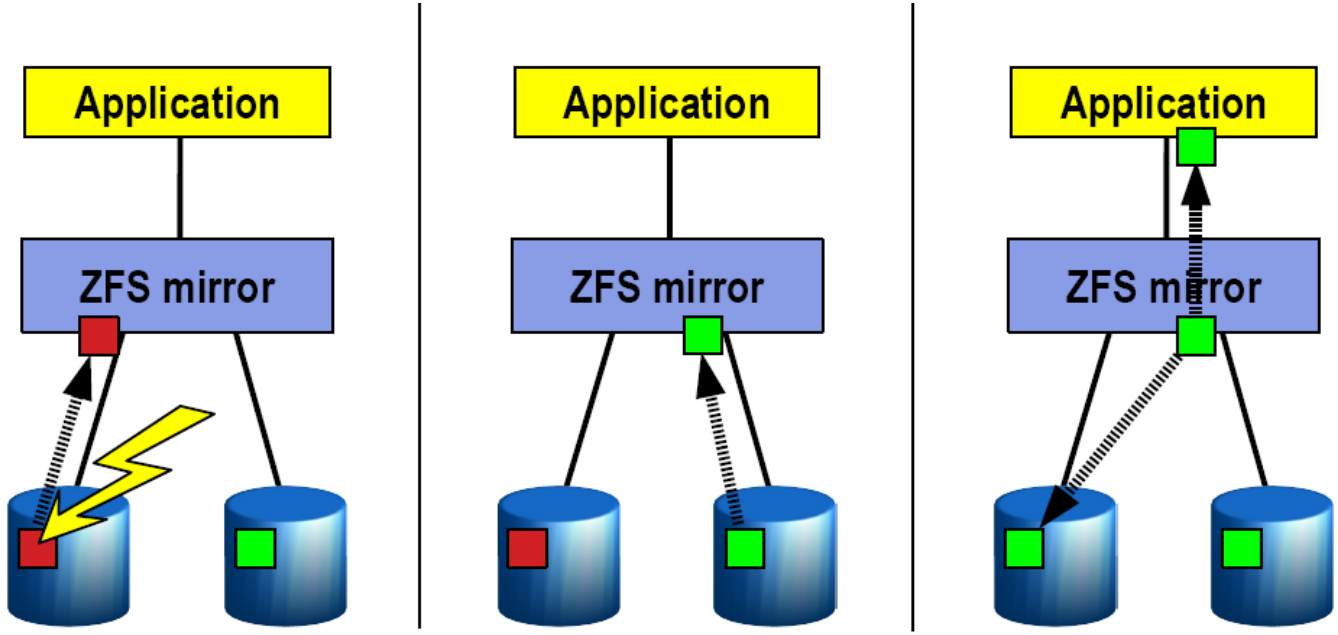
- Highlights forts.
  - Man kan styre størrelsen på filsystemene i pool'en med quota's og reservations.
  - ZFS bruker copy-on-write noe som gjør snapshots og kloner veldig kjapt og bra. Uendret data tar ikke plass i et snapshot, det er først når filer endres at originalfilene blir værende på disk, og dermed tar opp ekstra plass.
  - ZFS er et transaksjonsbasert filsystem, data vil alltid være konsistent på disk. Man trenger aldri kjøre fsck igjen...
  - ZFS bruker dynamisk blokkstørrelse opp til 128kb, som gjør at man mister mindre plass enn man gjør i dag.

- Highlights forts.
  - ZFS har end-to-end checksum'ing, noe som gjør at om man bruker mirror eller raidz så vil ZFS automatisk oppdage feil og fikse de "live".
  - Filsystemer og hardware-raid har generelt ikke mulighet til å finne "silent errors" i dag. CERN har kjørt en test der de skrev 1mb, sov 1sec, leste 1mb osv. På 3000 rack-servere med HW raid var det etter 3 uker 152 tilfeller av "silent data corruption" som bare skyldes "fluke error", ikke noen hardwarefeil. Dette er noe man ikke oppdager i dag, men som ZFS ville oppdaget og fikset live.

Tradisjonell raid-1 (mirror) ved feil



Self-Healing Data i ZFS



- Highlights forts.
  - Metadata blir automatisk replikert over alle devicer som er i en pool for å unngå å miste metadata i fall en disk feiler.
  - ZFS tar seg av all mounting og sharing (både NFS og CIFS). Man slipper vfstab/dfstab osv.
  - ZFS støtter ACL'er som er veldig lik Windows sine rettigheter. Dette er fullt ut støttet også i NFS4 og Samba/CIFS.
  - ZFS støtter disk scrubbing, der den går gjennom alle data og sjekker for feil, og har man mirror eller raidz så fikser den feilene automatisk.

- Highlights forts.
  - ZFS støtter "True Windows SIDs" noe som vil gjøre kompatibiliteten med Windows mye bedre.
  - OpenSolaris har nå innebygget støtte/overbygging for CIFS (/Samba)i kjernen. Sun står for patching/vedlikehold, og det har vist seg mye mer kompatibelt med Windows enn tradisjonell Samba. Det er også en veldig økt ytelse med å ha det direkte i kjernen contra som et program.

ZFS og ytelse

Task (tested with 3 million files)	**ZFS [min.]**	**UFS [min.]**	**ZFS:UFS performance ratio**
Create	3,75	93,75	25 x faster
Rebuild HELIOS Desktop	25,30	116,75	4,6 x faster
Remove	16,75	217,30	13 x faster

[min.] = minutes

ZFS vs Veritas File System (brukt i dag)

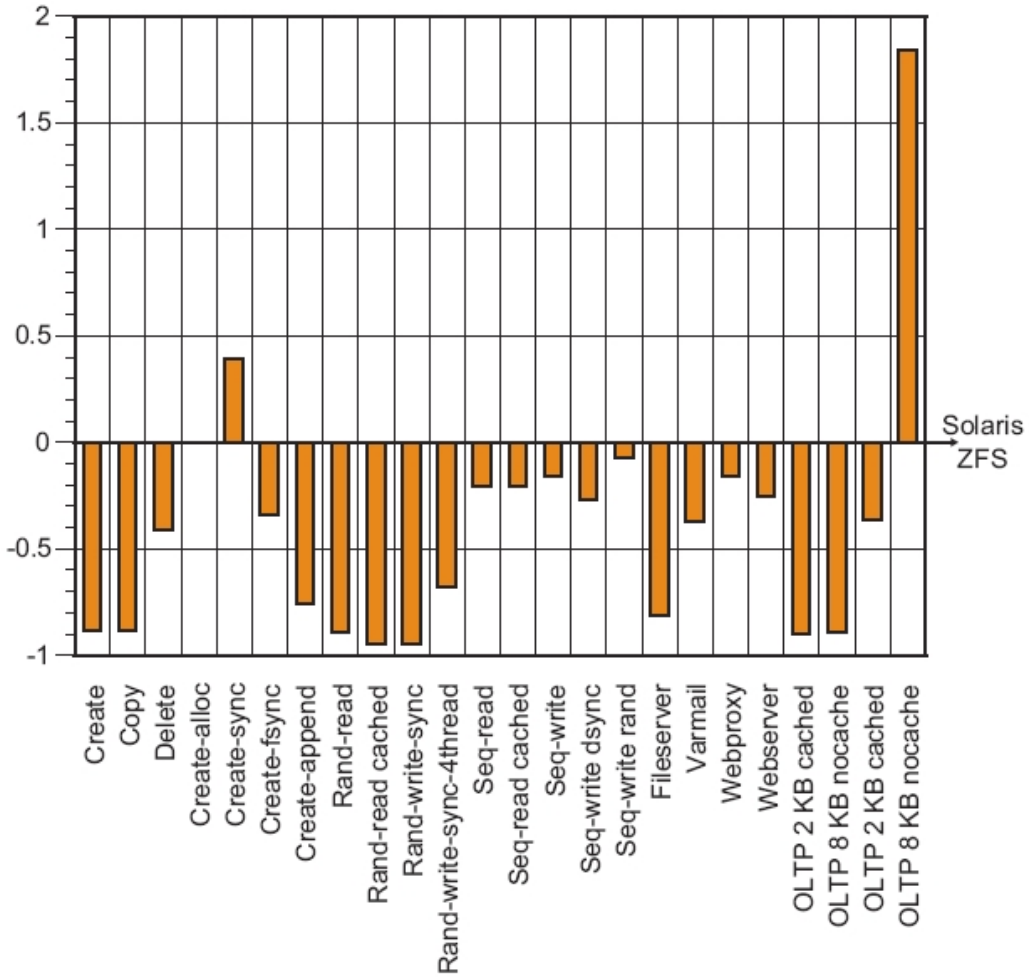


Figure 1-1. Filebench testing summary for the Veritas Storage Foundation versus Solaris ZFS

ZFS vs ext3 (linux)

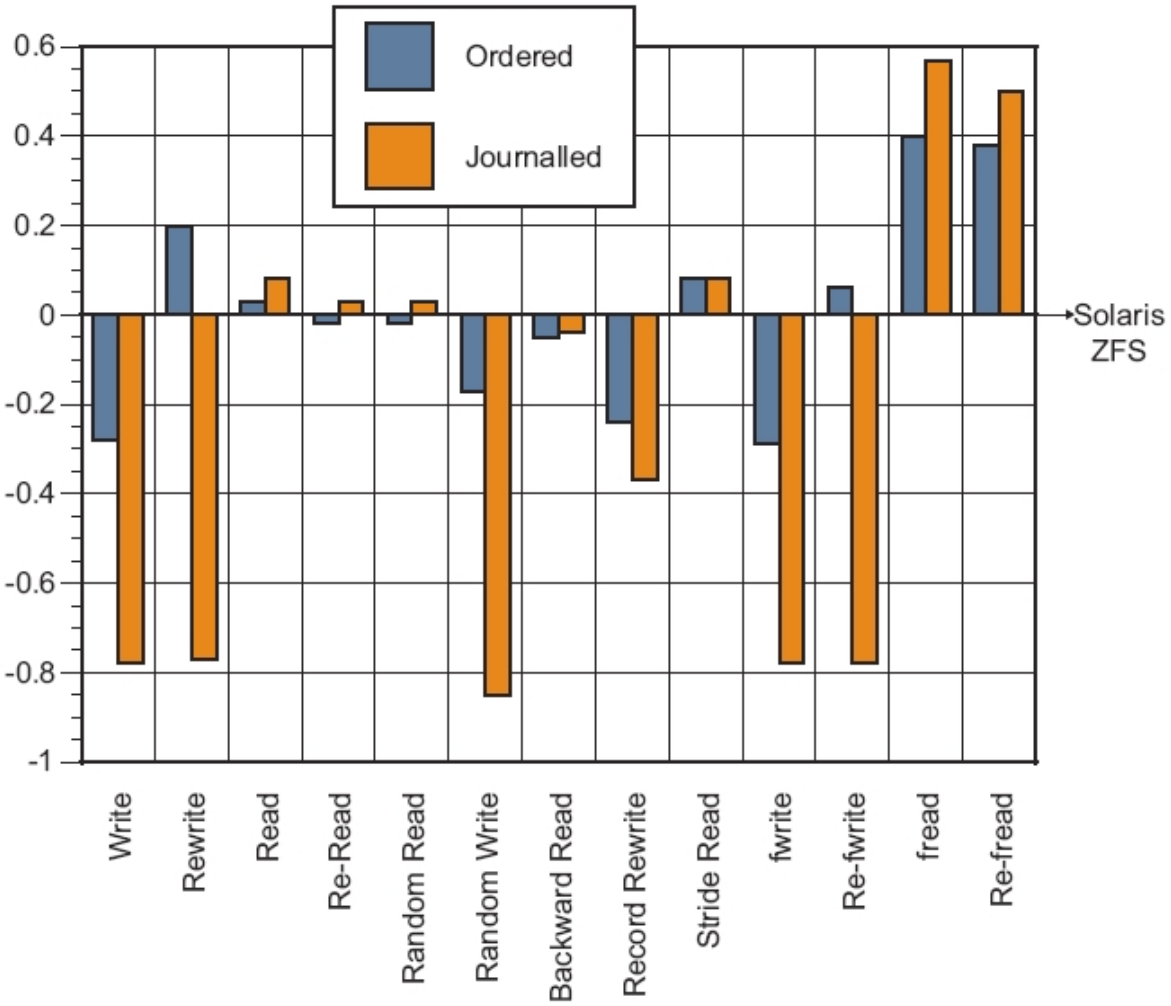


Figure 1-1. IOzone testing summary for the Red Hat Enterprise LInux ext3 file system versus Solaris ZFS

ZFS vs NTFS

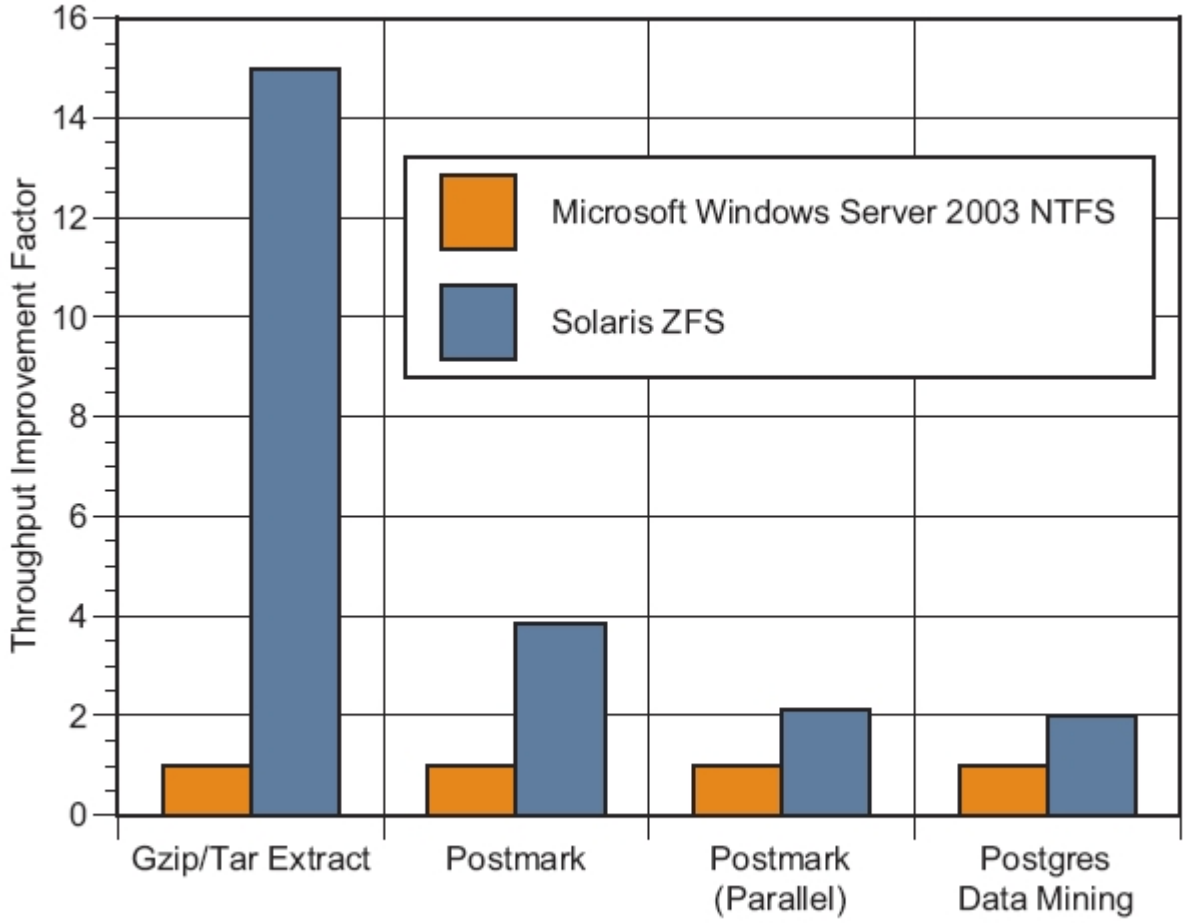
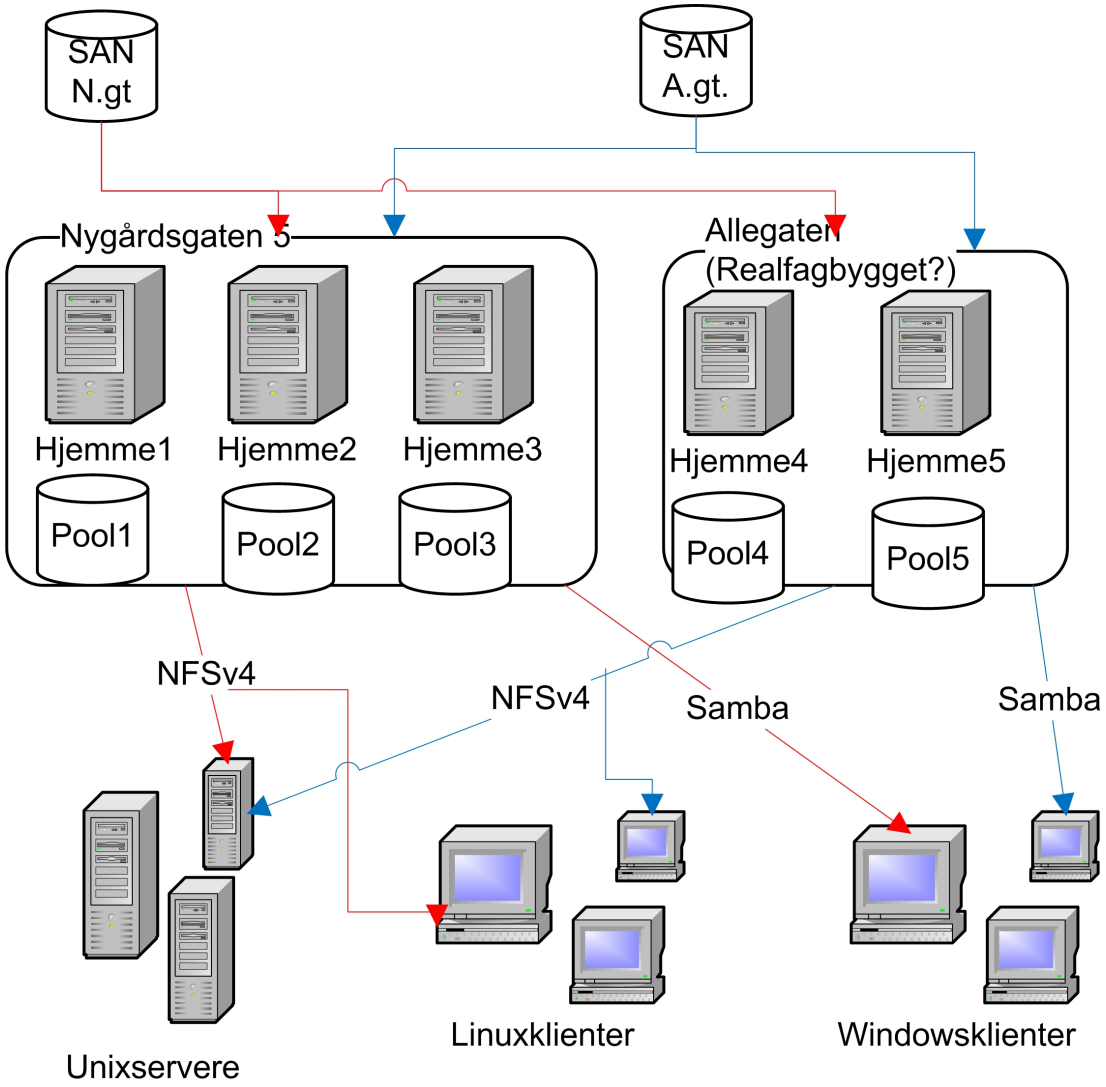


Figure 1-1. Testing summary for the Microsoft Windows Server 2003 NTFS file system and Solaris ZFS

Tanker rundt hjemmeområder fremover



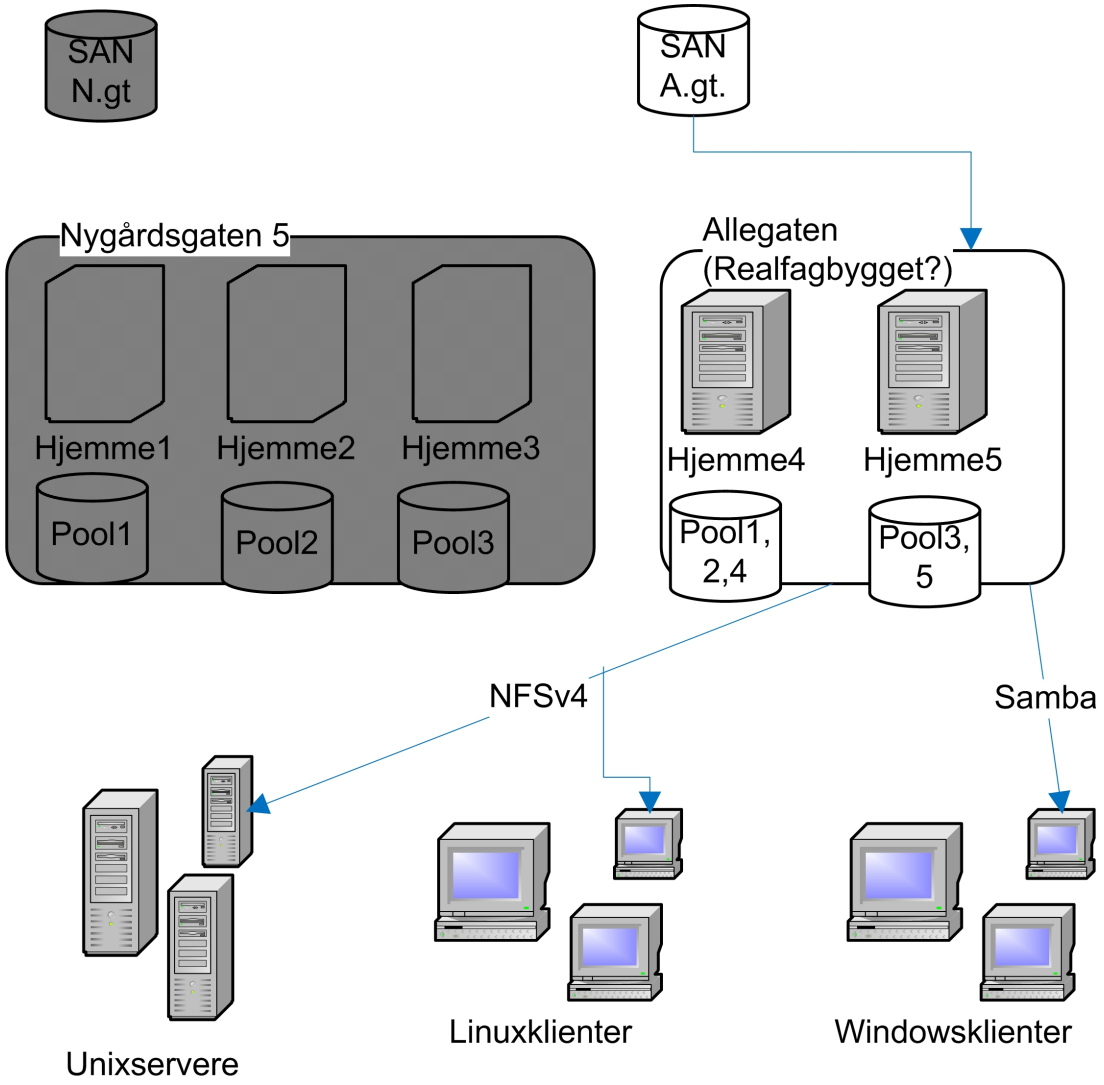
Hjemmeområder med ZFS

- Vi setter opp et antall servere (anslått 5) – noen av de står i nygårdsgaten og noen i allegaten.
- Hver server får hver sin pool med ca. 5-6000 hjemmeområder, alle med hvert sitt ZFS filsystem i pool'en.
- Hver pool får disk fra begge san, og bruker ZFS til å mirror'e data.
- Hver pool har et navn/ip som er det som brukes i sebra m.m. for hjemmekataloger.
- Vi kan kanskje vurdere mer enn en pool per server for å lettere kunne utvide med flere servere om det er nødvendig.

Hjemmeområder med ZFS

- Det er enkelt å tildele mer disk til en pool, vi bare får en ny disk fra hvert san og legger den til i pool'en, som automatisk vil ha den nye disken tilgjengelig for alle filsystemene i pool'en.
- Vi bruker quota i ZFS på hver bruker sitt filsystem, veldig enkelt grensesnitt for å endre på quota – kan med fordel integreres i Sebra.
- Evt. feil på disk vil oppdages av ZFS og fikses live. I tillegg kan vi kjøre scrub på alle disker f.eks. en gang i uken.
- Hver server deler ut med NFSv4 og Samba til alle klienter og andre servere.

Hvis Nygårdsgaten går ned



Hvis Nygårdsgaten går ned

- Vi lager script som kan ta over en annen maskin sin pool ved å mounte den fra san'et, og tar samtidig over navn/ip og deler den ut med NFSv4 og Samba.
- Dette vil være en manuell jobb. Vi ønsker ikke clustring og dermed må vi sikre oss at maskinen som er nede er helt nede før vi kjører scriptet for å ta pool'en opp på en annen server.

Alternativer

- Sun har nylig lansert et produkt som er bortimot akkurat det vi har kommet frem til på egenhånd, men som en ferdig løsning. Den er såpass økonomisk gunstig, pluss at man får alt ferdig, at vi har bestemt oss for å undersøke det nærmere.
- I tillegg er det kommet til meninger og alternativer som bør utredes grundig nok til at vi er trygg på at vi går riktig vei.
- Vi arbeider nå på spreng med å få frem alle alternativer for å ha et så bredt grunnlag som mulig for å ta en avgjørelse.
- Vi jobber med kravspec på unix og windows, som vi igjen vil bruke til å gå ut og spørre leverandører om potensielle løsninger.

Tidsplan

- Jobbes med:
  - Utrede og teste boot-tider osv. nærmere
  - Få avklart resterende uklarheter.
  - Teste OpenSolaris vs Solaris, samt teste ytelsen på NFSv4 og CIFS på begge OS.
  - Sjekke ut Sun sin 7000-løsning nærmere, samt se om det er flere alternativer som bør sjekkes ut nærmere.
- Snarest: Bestemme oss og bestille
- Sommeren 2009: Sette opp og teste løsningen.
- Høsten 2009: Flytte over brukere, ansatte først så studentene.