

Ytelsestesting

**Erfaringer og utfordringer i forbindelse med
Ytelsestesting mot Komplekse og Integreerte Løsninger**

Rune Lervåg Skår
Know IT Quality Management

Rune L Skår

- Partner i Know IT Quality Management
- Har i ca 15 år jobbet som konsulent, primært mot Test og QA de siste 10 år
- Mange ulike roller som løsningsarkitekt, prosjektleder og testansvarlig
- Hovedsakelig i større utviklings/migrerings/infrastruktur prosjekter
- Jobbet mye i grensesnittet mellom utvikling/prosjekter og drift/produksjon
- Ytelses-/tekniske tester av komplekse web, e-business og nettbankløsninger
- Jobbet med utfordringer relatert til teknisk testing med verktøy siden "Y2K"

Hva er ytelsestesting...? og hvordan og hvorfor skal vi teste...?



Vi forventer i økende grad at alt skal fungere 100% - hele tiden...

Konsekvensene ved at ting ikke virker som de skal blir stadig større

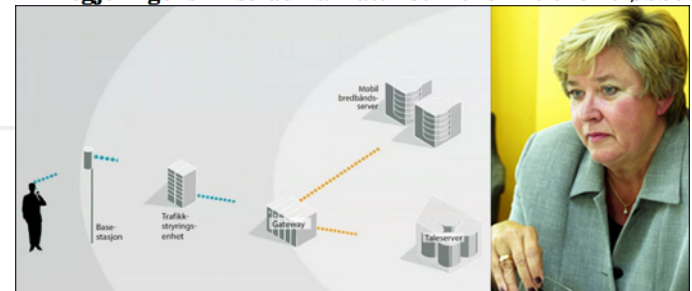


Mobiler når ikke nødnummer

Telenor har store problemer med mobilnettet. Folk i akutt nød bør ringe nødnumrene via fasttelefon.

Telenor oppdaget ny feil: Regjeringsråd slo krisealarm

**** Regjeringens kriseråd har fått hotline for Telenor-trøbbel**



Årsaken til Telenors kollaps funnet

Av [Marius Jørgenrud](#), tirsdag 19. juli 2011 kl 08:17

Fant endelig kilden til deres verste mobilhavari noensinne.

I pinsehelgen skjedde det utenkelige: Telenors mobilnett knakk sammen i Norge, selve utstillingsvinduet for den internasjonale telegiganten.

Det skulle vise seg å bli den verste svikten i mobilnettet siden det ble oppført i 1993.

Telenor redegjorde etter hvert for noen av feilene som oppsto. (Les: [Svitsjen låste seg fullstendig](#)). Samtidig famlet Telenor og kunne ikke peke på den egentlige kilden til problemene, noe som bekymret Post- og teletilsynet (PT).

Nå, nesten halvannen måned senere, melder Telenor at de endelig har identifisert det de kaller «rotårsaken».

Store samfunnsmessige konsekvenser, mange brukere rammes



Aktuelt og presse | Kontakt og hjelp |

altinn
enkler dialog med det offentlige

Forsiden Min meldingsboks

Stor pågang Altinn

Foto: SKJERMBILDE

Innloggingsproblemer for Altinn - igjen

Påtrykket hos Altinn var mandag større enn hva tjenesten kan håndtere. Regnskapsførere som har leveringsfrist for sine kunder nettopp denne dagen fortviler.

altinn problemer

Omtrent 200 000 resultater (0,08 sekunder)

[Ny Altinn-feil på oppløpssiden - IT - E24](#) 🔍

[e24.no/it/ny-altinn-feil-paa-opploepssiden/20051800](#) - Bufret

29. apr 2011 – Fossum kunne ikke gi noen garanti for at det ikke oppstår **problemer** med **Altinn** i det fristen for å levere inn selvangivelsen nærmer seg. ...

▶ [Altinn sliter fortsatt](#) 🔍

[www.aftenposten.no/okonomi/innland/article4069333.ece](#) - Bufret

22. mar 2011 – **Altinn** lovet bedring på nettstedet i løpet av tirsdag kveld, men onsdag morgen er det fortsatt store **problemer**.

[Retting i gang etter feil i Altinn - Nyheter - Innenriks - Aftenposten.no](#) 🔍

[www.aftenposten.no/nyheter/iriks/article4153209.ece](#) - Bufret

21. jun 2011 – Feilen i internettportalen **Altinn** er funnet, og vil bli ...

✚ [Vis flere resultater fra aftenposten.no](#)

Lederskifte i Basefarm og Accenture

Av [Marius Jørgenrud](#), fredag 6. mai 2011 kl 11:22

Har selvfølgelig intet med Altinn-skandalen å gjøre.

Driftsleverandør Basefarm og konsultentselskapet Accenture har begge måttet tåle et ekstra kritisk blikk etter at Altinn krasjet ved fremleggelsen av årets skatteoppgjør.

Rett etter en omfattende innspurt før leveringsfristen for selvangivelsen gikk ut, der frykten for nye avbrudd var påtagende, kommer flere endringer i selskapenes ledelse.

Innrømmer dårlig Altinn-testing

Altinn-sjef Erik Fossum advarer mot nye kømareritt i det fristen for innlevering av selvangivelsen nærmer seg.



Konsernsjef Terje Mjøs i EDB ErgoGroup (foto: Scanpix).

Taper 50 millioner på minibank-kaos

Finanstilsynet kritiserer EDB ErgoGroup

Kortskandalen: - Uakseptabelt av EDB

Sentrale deler av Norges banksystem kjøres uten fungerende sikkerhetsnett, og det forårsaket kortkaoset i påsken. **Finanstilsynet er i harnisk.**

Sparebank1 dumper EDB Ergogroup



Ikke alt brukertestes godt nok

– Billettsystemet er håpløst



Forbrukerrådet får stadig klager over frustrerte bybanepassasjerer.



Forbrukerrådet har bedt om et møte med Skysst angående billettsystemet. (Illustrasjonsfoto) Foto: MAGNE TURBY

Billettkaos på Bybanen

Forbrukerrådet klager på billettsystemet



LANGT DER FRAMME: Køen foran billettautomaten var lang etter kampen mellom Brann og Leeds.

Foto: Ragnhild Osnes Olsen/NRK

Det er ikke mulig å kjøpe tur-retur billett til Bybanen. Dermed blir det kaos etter store arrangement.



Foto: Anders Haakonsen

Åpningen av Knappetunnelen utsatt

Åpningen er utsatt

Åpning av Ringveg vest, den nye firefeltsvegen mellom Dolvik og Sandeide i Bergen, må utsettes. Ved testing av styringssystemene i forbindelse med sikkerhetsgodkjenning av Knappetunnelen ble det registrert feil. Feilsøkingen i styringssystemene vil ikke bli ferdig til den fastsatte åpningen 11. august.
[29.07.2010]



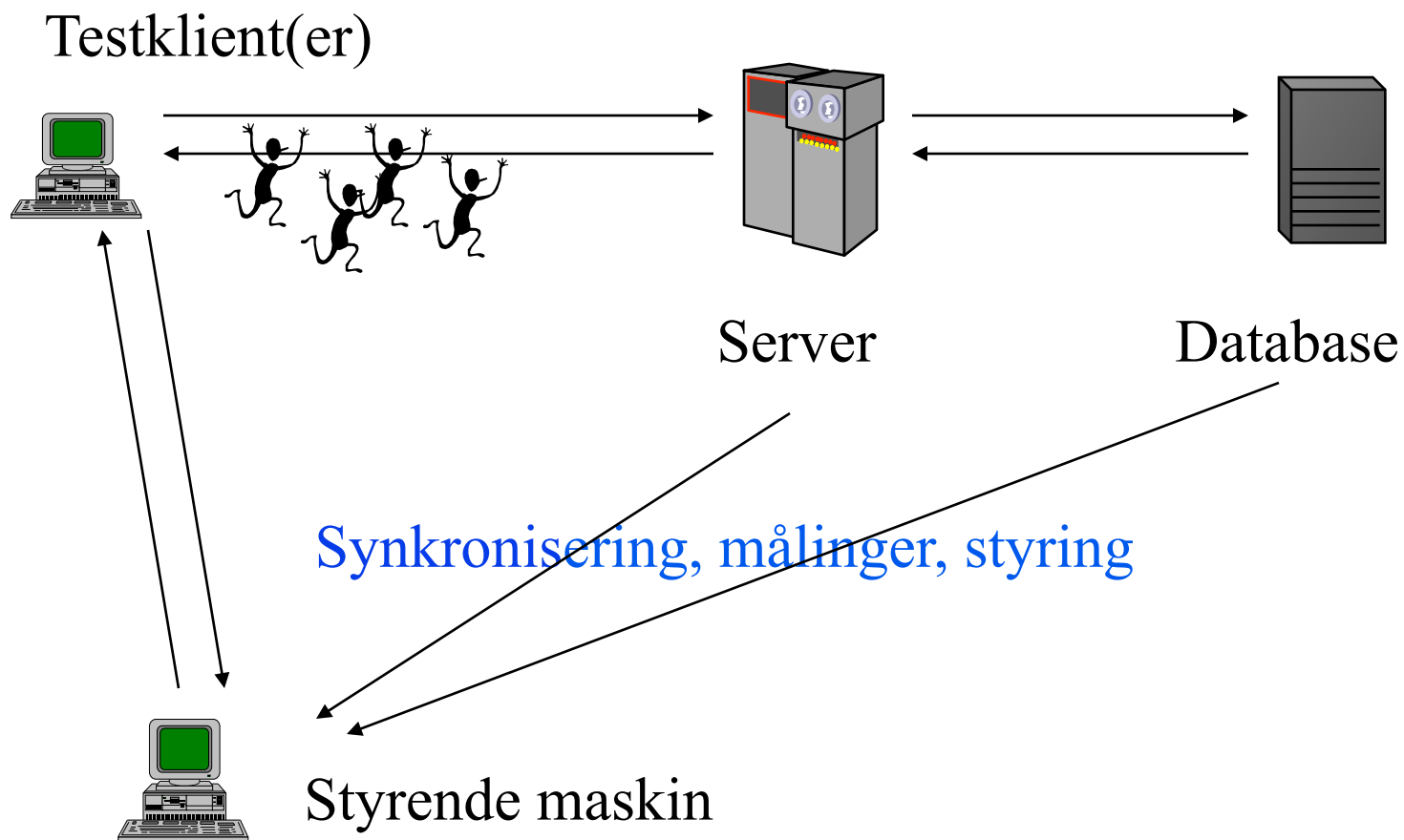
a data conversion from a 64-bit floating point to 16-bit signed integer value caused a hardware exception (more specifically, an arithmetic overflow, as the floating point number had a value too large to be represented by a 16-bit signed integer). Efficiency

No matter how well applications are functionally designed, no matter how well they meet business requirements, they are virtually useless to end-users if performance is slow or unreliable.

ISTQB Glossary – hva er ytelse og robusthet...

Engelsk	Norsk
performance: The degree to which a system or component accomplishes its designated functions within given constraints regarding processing time and throughput rate. [IEEE 610] See also efficiency.	ytelse: Graden til hvilken et system eller en komponent oppfyller sine bestemte funksjoner innen gitte restriksjoner når det gjelder behandlingstid og kapasitet. [IEEE 610] Se også effektivitet.
performance testing: The process of testing to determine the performance of a software product. See also efficiency testing.	ytelsestesting: Testing for å måle ytelsen til et programvareprodukt. Måling av systemets svartider eller testing av systemets kapasitet under gitte krav til svartider.
performance testing tool: A tool to support performance testing that usually has two main facilities: load generation and test transaction measurement. Load generation can simulate either multiple users or high volumes of input data. During execution, response time measurements are taken from selected transactions and these are logged. Performance testing tools normally provide reports based on test logs and graphs of load against response times.	ytelsestestverktøy: Et verktøy som understøtter ytelsestesting som vanligvis har to hovedkomponenter: Lastgenerering og måling av testtransaksjoner. Lastgenerering kan enten simulere flere brukere eller store mengder inputdata. Under utførelsen blir responstiden målt for utvalgte transaksjoner og disse blir logget. Slike verktøy gir vanligvis rapporter og grafer som viser responstid og kapasitet mot belastning og som er basert på loggene.
robustness: The degree to which a component or system can function correctly in the presence of invalid inputs or stressful environmental conditions. [IEEE 610] See also error tolerance, fault-tolerance.	robusthet: Graden som en komponent eller et system kan fungere korrekt når ugyldige inputs eller vanskelige betingelser i omgivelsen er tilstede. [IEEE 610], Se også feiltoleranse.
robustness testing: Testing to determine the robustness of the software product.	test for robusthet: Testing for å bestemme et produkts robusthet.

Ytelsestesting - hvordan...



- *Hensikten med ytelsestest er å fastslå respons, dataflyt, pålitelighet og/eller skalerbarhet av et system ved en gitt last.*

Ytelsestest er normalt sett gjennomført for å oppnå følgende:

- *Vurdere hvor klar applikasjonen er for produksjon*
- *Evaluering mot ytelseskriterium*
- *Sammenligne ytelsesfaktorer av flere systemer og system konfigurasjoner*
- *Finne kilder til ytelsesproblemer*
- *Støtte system tuning/stabilisering*
- *Finne grensesverdier for hva infrastruktur, OS og applikasjoner kan håndtere før det går utover ytelse og stabilitet*

Ytelsestesting: Testing for å måle ytelsen til et programvareprodukt. Måling av systemets svartider eller testing av systemets kapasitet under gitte krav til svartider.
(ISTQB)

I tillegg gjennomføres "ikke funksjonelle tester" i forhold til

- Robusthet i løsningen
- Negative tester / Avvikshåndtering
- Terskelverdier / Metningsgrenser
- Kapasitetsplanlegging
- Sikkerhetstester
- Endring i infrastruktur / systemsoftware
- Endringer i omkringliggende/tilgrensende løsninger

Ytelsestesting – Ulike typer tester og formål

Testmålsetning	Web Ytelse Test
Fastslå grenser	Last Test
Måle bruker opplevelse	Browser Ytelse Test
Teste spesifikke siloer	Målrettet Infrastruktur Test
Verifisere redundans	Failover Test
Gå utover grenseverdier	Stress Test
Planlegge for fremtiden	Kapasitets Test
Pålitelighet over tid	Soak Test

Ytelsestesting – Ulike typer tester og formål

Test Objective

Web Performance Test

Determine Limits

Load Test

Measure User Experience

Browser Performance Test

Test Specific Silos

Targeting Infrastructure Test

Verify Redundancy

Failover Test

Exceed Break Points

Stress Test

Plan For The Future

Capacity Test

Reliability Over Time

Soak Test

Utfordringer – FØR oppstart av Ytelsestesting

Vanlige spørsmål (fra oss som skal teste...)

- Er Løsningen og Verdikjedene som skal testes tilstrekkelig Dokumentert?
- Er Testmiljøer komplette og skalerte?
- Er Akseptansekriterier klart definerte?
- Er det en klar plan for Hva og Hvordan det skal testes?
- Er Testmiljø og Testdata konfigurert og tilrettelagt?
- Er det tilrettelagt for Verktøy og Målepunkter for å Monitorere løsningen?
- Er Nøkkelpersoner (utviklere, arkitekter, drift) avklart/prioritert?
- Er det tilstrekkelig Tid før løsningen skal i produksjon til rette det som eventuelt avdekkes?

Svaret er veldig sjelden ubetinget JA...

Planlegging – ha klar Strategi ved oppstart av Ytelsestesting

Core Performance Testing Activities

1. Identify Test Environment

2. Identify Performance Acceptance Criteria

3. Plan and Design Tests

4. Configure Test Environment

5. Implement Test Design

6. Execute Tests

7. Analyze, Report, and Retest

Vanlig nivå på ”*formelle krav*” til prosjektene i forhold til ytelsestesting:

	Ja	Nei
Er ytelsestest gjennomført	<input type="checkbox"/>	<input type="checkbox"/>

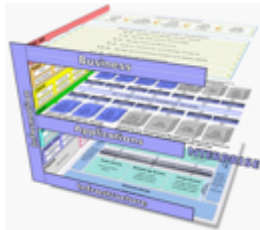
Eksempler på svar fra Prosjektene

(til oss som skal teste...)

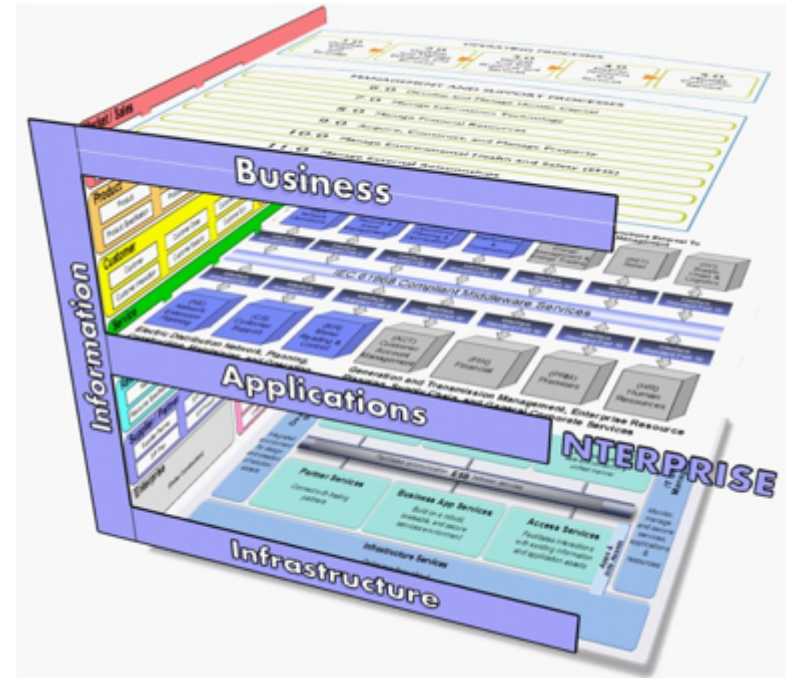
- *Hva som skal ytelsestestes? – ALT!*
- *Skal teste med trafikk som det vil bli i Produksjon, men vi vet ikke hva det vil bli...*
- *Vi har rettet alle feil og er ferdig med funksjonelle tester og skal i produksjon neste uke, skal bare ha gjennomført en ytelsestest først...*
- *Vi kan ikke prioritere å avgi ressurser til planlegging, monitorering eller feilsøking, men kom dersom dere er sikker på at feilen ligger hos oss...*

Overordnet arkitektur og lagdeling ligner mye på det som fantes for 10 år siden...

2001



2011

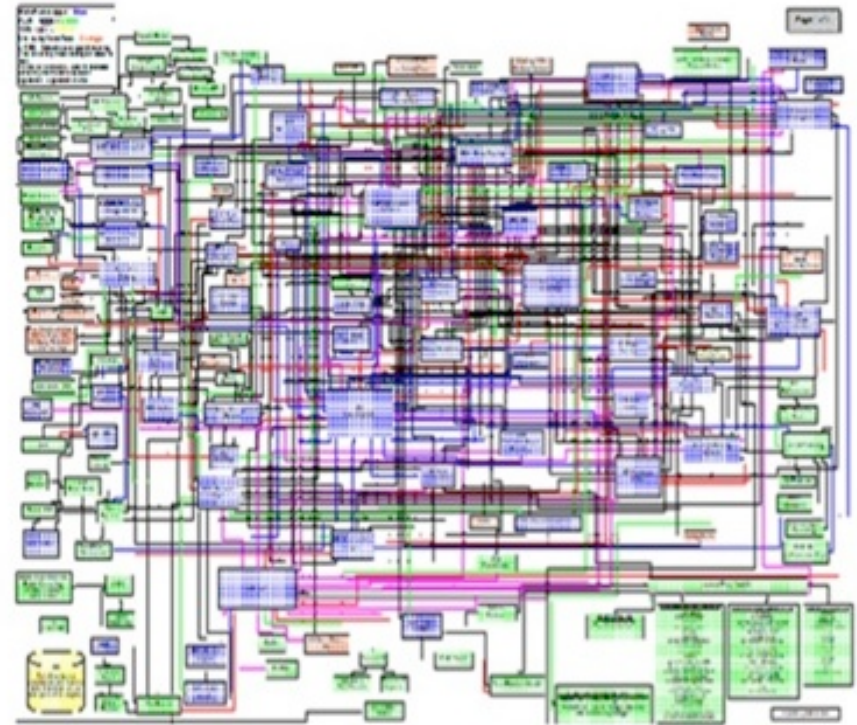


... men underliggende funksjonalitet og kompleksitet er mye mer integrert og komplisert...

2001



2011

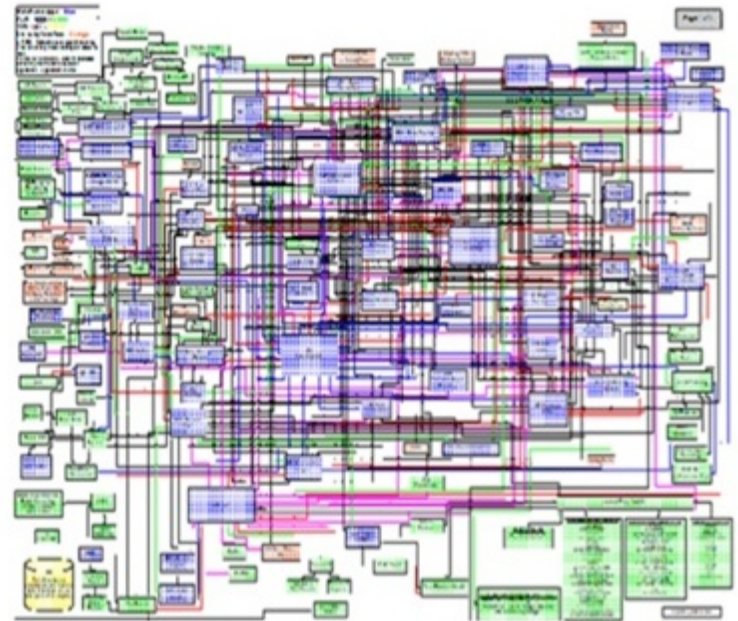


... dermed øker utfordringene og kravene til de som skal gjennomføre ytelsestest

- **Totalløsningene**
så komplekse at få (eller ingen)
som har total oversikten ift

- Arkitektur
- Infrastruktur
- Grensesnitt
- Avhengigheter
- Testdata
- VERDIKJEDER...

2011



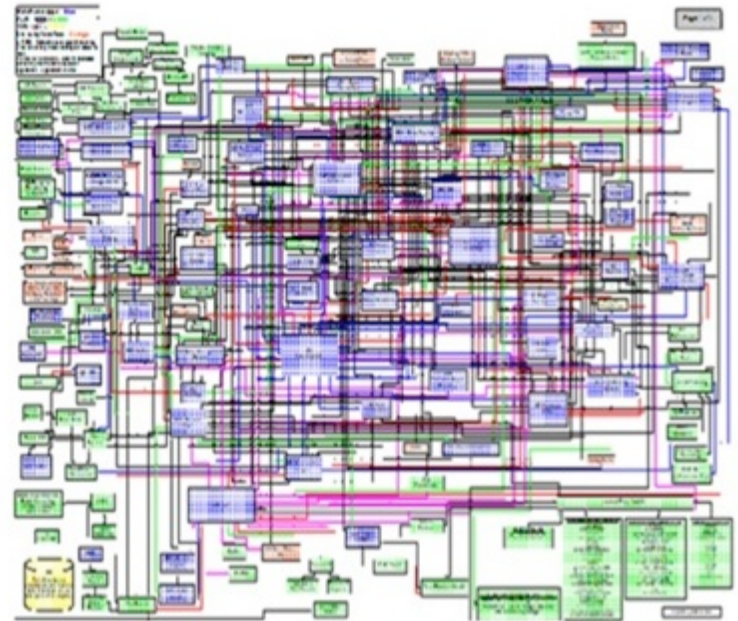
... dermed øker utfordringene og kravene til de som skal gjennomføre ytelsestest

2011



• Testmiljøer

- Ikke "produksjonslik" (kapasitet/konfig)
- Ikke "dedikert"
 - andre testaktiviteter
- Ikke "stabilt"
 - mange endringer
 - leveranser som ikke er kvalitetssikret.
- Feilsøking ikke "prioritert"
 - vanskelig og ressurskrevende
 - innsats fra utvikling og drift



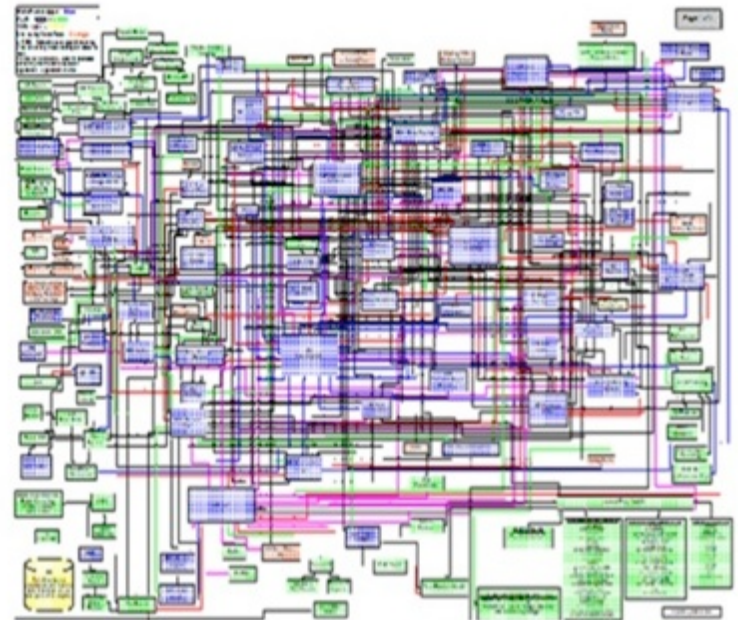
... dermed øker utfordringene og kravene til de som skal gjennomføre ytelsestest

2011



- **Nøkkelressurser**

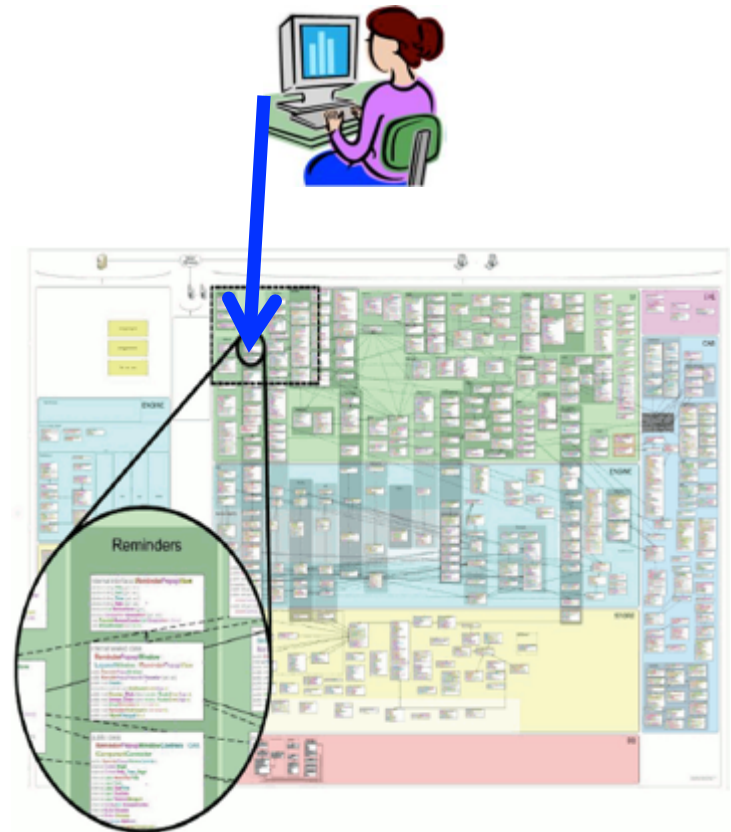
- Teknisk Testleder
- Teknisk Tester/
Verktøy Ekspert
- Utviklere/Arkitekter
- Driftsressurser/
Infrastruktur kompetanse



Ytelsestester blir ofte mer omfattende og komplisert en antatt

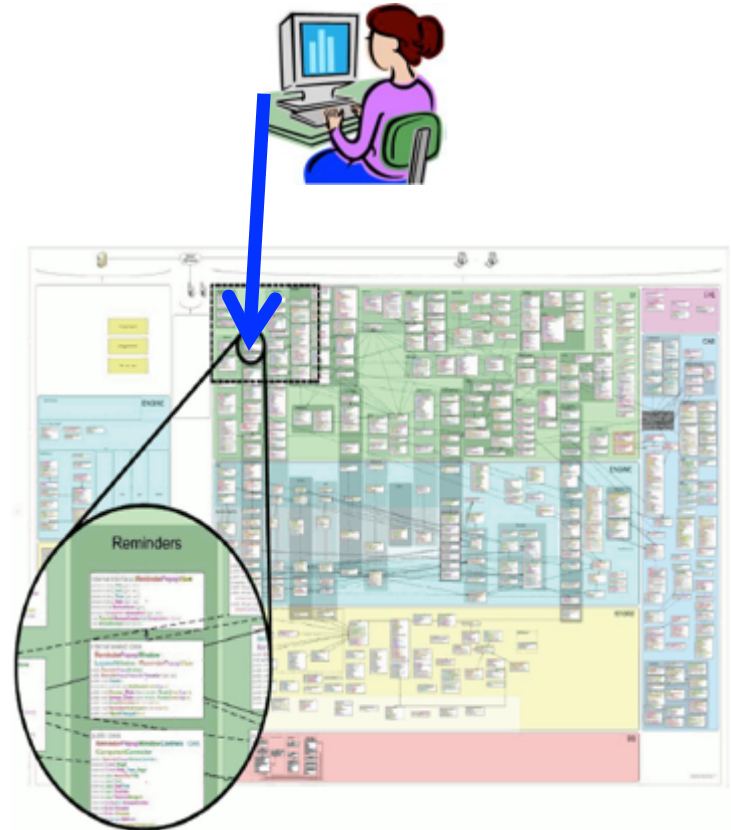
Testing av enkeltkomponenter/ mindre funksjonelle endringer

- Kreve komplett testmiljø for å få testet verdikjeder/sluttbruker konsekvens
- Føre til behov for omfattende regresjonstesting av andre løsninger
- Påvirke omkringliggende systemer



Feilsituasjoner

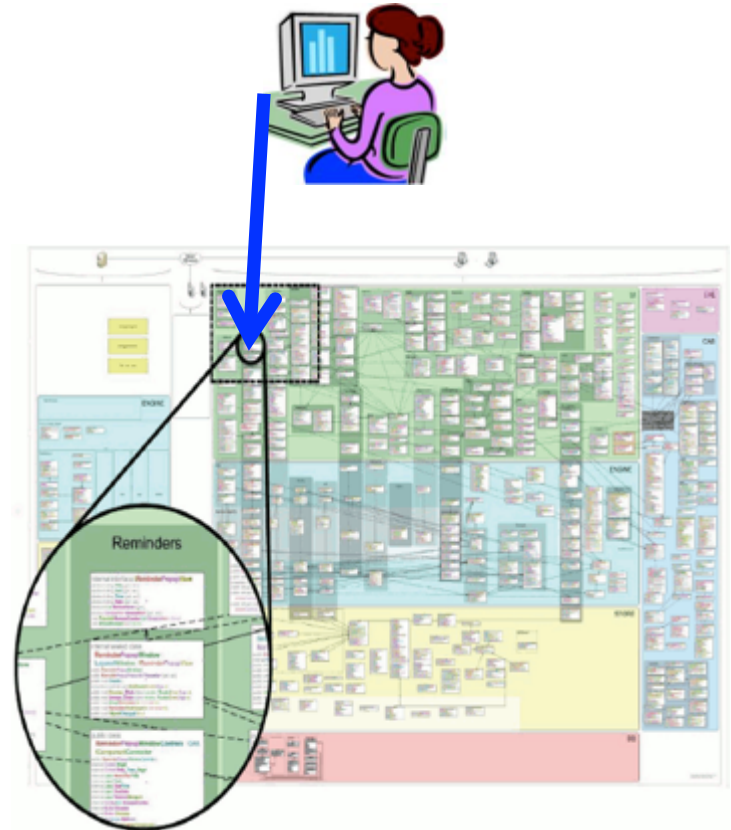
- Må være reproduerbare
- Må avklare om skyldes ustabile miljøer/ feil Ikke "produksjonslike" (kapasitet/konfig)
- Har en tilstrekkelig loger/ monitorering/kilder til å finne årsak



Ytelsestester blir ofte mer omfattende og komplisert en antatt

Rotårsak til Feil

- Ofte dukker feilene opp andre steder enn der en "leier" – tester applikasjon og finner feil relatert til OS konfig
- Veldig tidkrevende og vanskelig å feilsøke i komplekse løsninger – skyldes feil
 - Applikasjon
 - System Software
 - Driftsmiljø
 - Konfigurasjon/Deploy
 - "annet" ...?

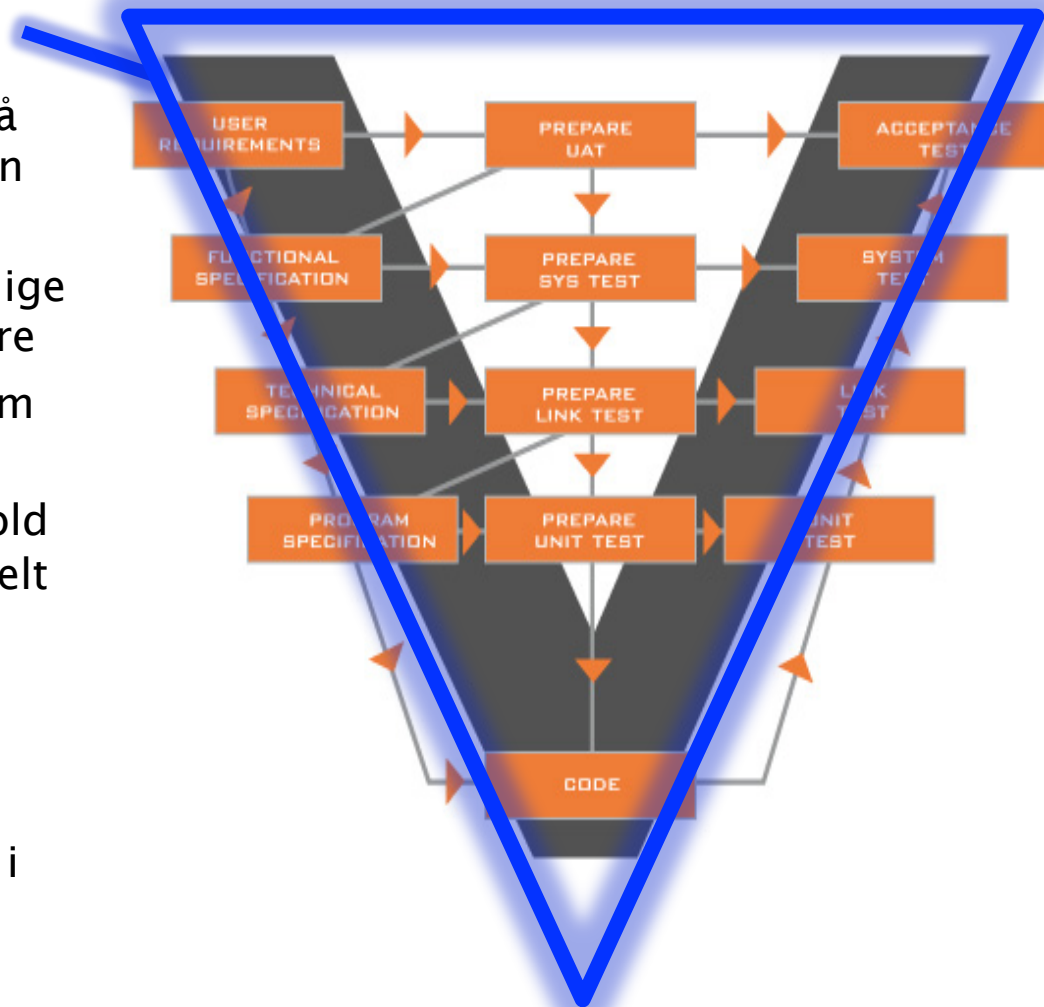


Når Bør Ytelsestesting Begynnes

Ytelsest bør være del av HELE utviklings og test prosessen

- KRAV til Ytelse og Robusthet må på plass i Kravspek Design fasen (ikke bare funksjonalitet...)
- Analysere og estimere hvor mulige flaskehalser i løsningen kan være
- Testmiljøer må konfigureres som produksjon (gjerne skalert ned)
- Utviklere må enhetsteste i forhold til ytelse/feilhåndtering på enkelt komponenter/grensesnitt
- Tilrettelegge for monitorering/ logging av gjennom hele verdikjeden
- Ytelsestesting kan være aktuelt i ALLE testfaser

THE V MODEL



Strategi i praksis - for gjennomføring av ytelsestest mot funksjonalitet i en løsning

- Prioritere viktigste funksjonalitet og analysere mest sannsynlige flaskehalser / endring i trafikkmønster
- Identifisere og kjøre test på minste/korteste "verdikjede" for en test. (oftest begrenset av testmiljø)
- Finne baseline/få stabile resultater før testene utvides og verdikjedene gjøres mer omfattende og komplekse
- Kjøre oppramping med stadig økende trafikk/antall virtuelle brukere for å finne terskelverdier/knekkpunkter i forhold til hvor mye trafikk vi får gjennom.
- Gjennomføre stabile kjøring med ca 80% belastning (før "knekkpunkt")
- Sammenligne resultater med trafikk tall/statistikker/estimer for produksjon (der dette er tilgjengelig)

Fokus på Ytelsetesting og Robusthet

- De siste årene har det vært veldig fokus og økt satsning på FUNKSJONELL testing.
- Må gjøres tilsvarende satsning og løft i forhold til Ikke funksjonelle testing (Ytelse, Robusthet, Feilhåndtering, Sikkerhet, Skalerbarhet, med mer)
- Økt krav til driftstabilitet og oppetid kombinert med stadig mer integrerte og kompliserte løsninger fører til at det må få på plass en økt fokus på ytelse og robusthet gjennom hele utviklingsprosessen
- Ytelsestesting og planlegging av dette må i gang mye tidligere og allerede i "krav fasen" (tilsvarende som har skjedd for funksjonell testing)

Dette vil sette større krav til kvalitet og satsning på

- Testmiljøer
 - Konfigurasjonsstyring og Drift,
 - Dedikerte Testmiljø for Tekniske tester
- Forvaltning av Testdata
- Applikasjons Deploy og Kvalitet i Leveranser
- Verktøy for Feilsøking/Logging/Monitorering
- **Ytelsetest og Robustifisering er ikke avsluttet når en løsning er produksjonsatt**

Hva kan vi som holder på med ”Teknisk Testing” bidra med?

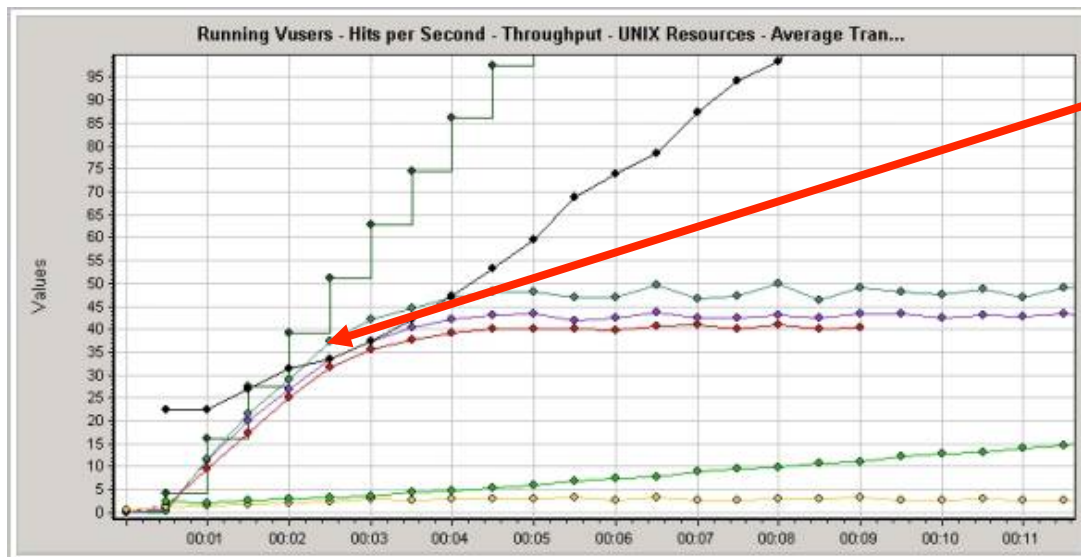
- Rådgivning til Forretningssiden/kunde til å få på plass KRAV til Ytelse og Driftstabilitet allerede i kravspesifikasjonsfasen
- Bistå i Design fasen til å analysere løsningene og estimere endring i trafikk mønster og identifisere mulige flaskehalser
- Kjøre ”teknisk POC” i forhold til ytelse/robusthet på nye løsninger før endelig valg av løsning/arkitektur/plattform gjøres
- Støtte Utviklerene for å tilrettelegge og gjennomføre ytelsetester på enkeltkomponenter/grensesnitt, få på plass bedre logging/monitorering, øke robusthet i løsningene
- Bistå Drift til å kjøre regresjonstester, teste tekniske endringer, analysere bruks-/trafikk mønster endringer, bedre overvåkning/monitorering, feilsøke/gjenskape avvikssituasjoner.

Ytelsestest som fag... er det bare problemer..?

- Kjennskap og overblikk over de store linjene og kunne dykke ned i de konkrete tekniske problemene/feilsituasjonene
- Analysere og forstå bruksmønster og kundeopplevelse og samtidig forstå løsningens arkitektur, teknologi og plattform
- "Oversette" brukerkrav for respons/ytelse til tekniske tester og script
- Kjennskap og kunnskap ulike teknologier, rammeverk, software, arkitekturer, infrastruktur – på tvers i organisasjon
- Jobbe med spennende Verktøy (ytelsetest, analyse, monitorering, feilsøking,..)
- Kommer i kontakt med "alle" ressurser i et prosjekt, fra forretningside/kunde, arkitekter, utviklere og driftsressurser
- Og fremfor alt... vi finner ALLTID noe... som ellers ville ha truffet som avvik i produksjon (med ofte store konsekvenser... og avis overskrifter...)

⇒ **Utfordrende, Allsidig og Spennende Fagområde**

Flaskehals - i forhold til samtidig trafikk

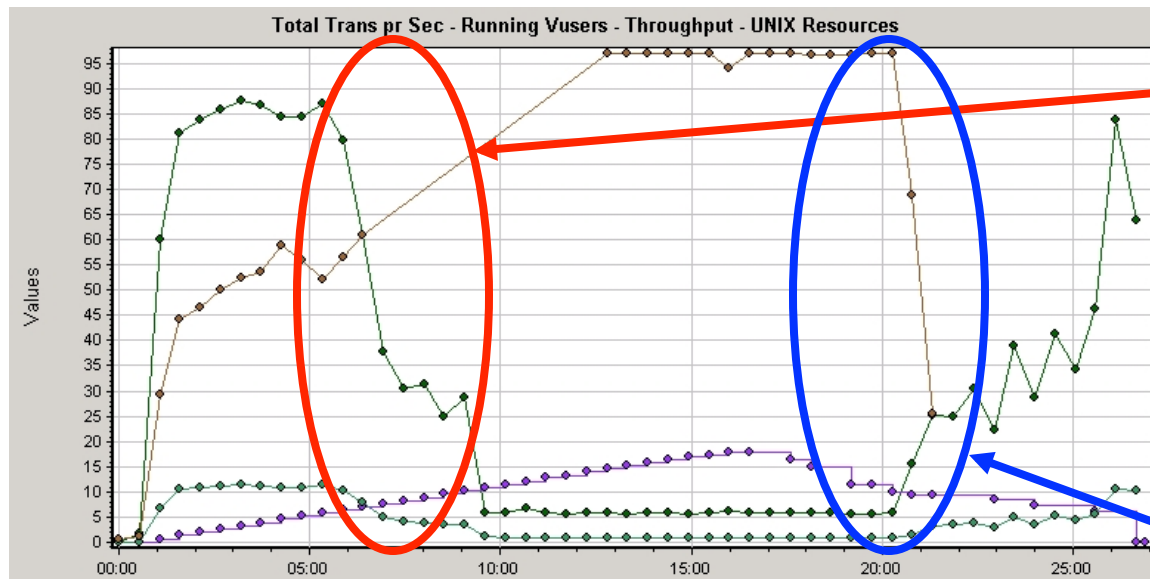


Result summary (example)

- Reach a bottleneck in traffic trough *srv1*
- With 4 *srv1* instances we get max 550Hits pr sec (220 on one instance) even though *srv1* is configured with 1536 threads per instance
- Response Time increase from 0,2 sec to 2,5 sek
- Only able to utilize
 - Ca 15% of configured threads
 - Ca 15% CPU pr *srv1* instance
 - Ca 3% CPU *srv2* *srv*
 - Ca 10% network bandwidth

Color	Graph	Scale	Measurement	Graph's Min.	Graph's Ave.	Graph's Max.	Graph's Median	Graph's SD	Machine Name	Monitor Type
	Running Vusers	0.2	Run	0.0	1174.723	2000	1315	523.782	N/A	N/A
	Hits per Second	0.05	Hits	0.0	780.241	878.033	825.867	140.671	N/A	N/A
	Throughput	4E-05	Throughput	0.0	1086866.063	1262814.7	1150174.3	200216.938	N/A	N/A
	UNIX Resources	1	CPU Utilization (Unix Kernel Statistics)	0.384	31.644	40.964	39.908	13.541		
	UNIX Resources	1		0.437	2.598	3.245	2.788	0.679		
	Average Transaction Response Time	10	_logo_jpg	0.221	1.789	2.528	1.878	0.494	N/A	N/A
	Average Transaction Response Time	100	_logo_jpg	0.224	1.754	32.023	1.574	2.486	N/A	N/A

"Metning" - ved høy trafikk belastning, køing av trafikk (100% cpu, ingen trafikk slipper gjennom)



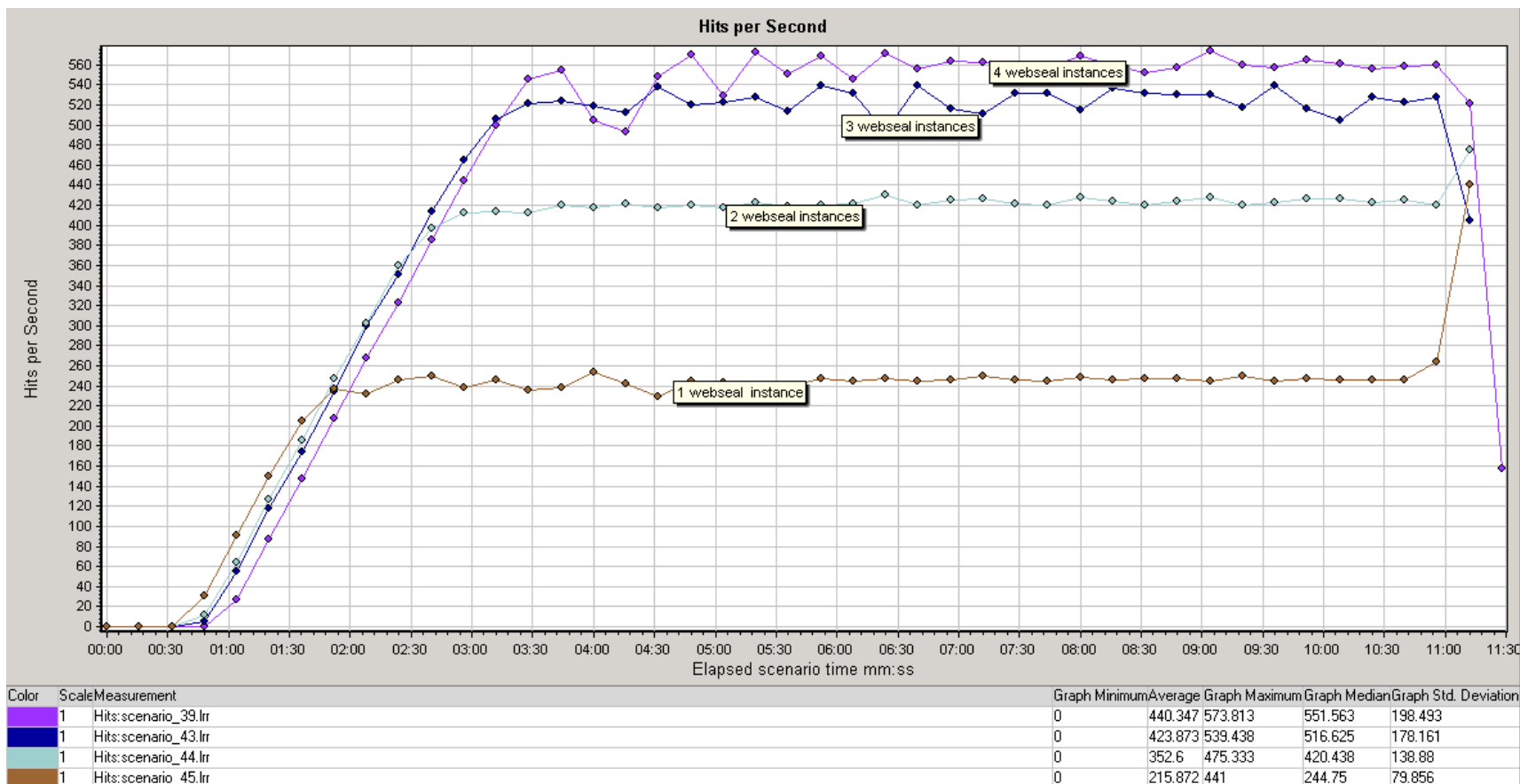
Resultat fra kjøring i test

- Når trafikken mot en SRV2 instans kommer opp i et visst nivå, så går CPU'en på SRV3 plutselig i taket (hopper fra 15% til +95% belastning)
- samtidig stuper ytelsen/respons stuper og en får vesentlig mindre trafikk gjennom (reduert til ca 10%)
- Responstidene øker til de mangedobbelte og de fleste sesjonene blir stående å henge til timeoutverdier er nådd

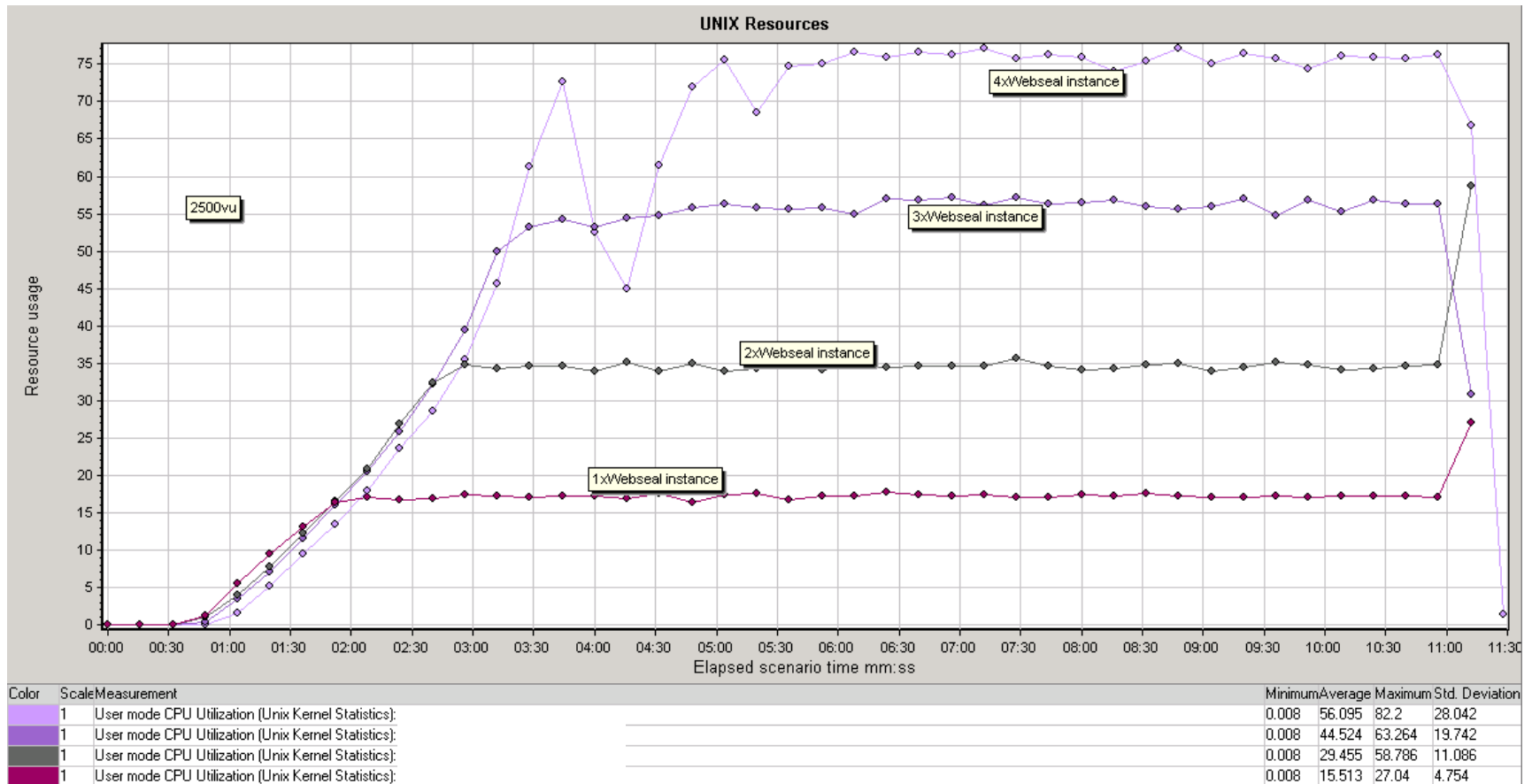
- Når trafikken mot SRV2 instansen kommer under terskelverdien (kundene/sesjonene har gått i timeout eller trafikken avtar) tar SRV2 instansen seg inn igjen
- CPU forbruk synker til "normal" belastning, hits pr sek og throughput øker og responstidene synker.

Color	Graph	Scale	Measurement	Graph's Min.	Graph's Ave.	Graph's Max.	Graph's Median	Graph's SD	Machine Name	Monitor Type
■	Total Transactions per Second	0.01	Pass	0.0	3161.055	8746.237	2477.15	3047.334	N/A	N/A
■	Running Vusers	0.01	Run	0.0	906.773	1786	950	545.656	N/A	N/A
■	Throughput	1E-06	Throughput	0.0	4095638.468	11304831.931	3331933.769	3970581.947	N/A	N/A
■	UNIX Resources	1	CPU Utilization (Unix Kernel Statistics)	0.459	70.192	96.905	93.989	29.957		

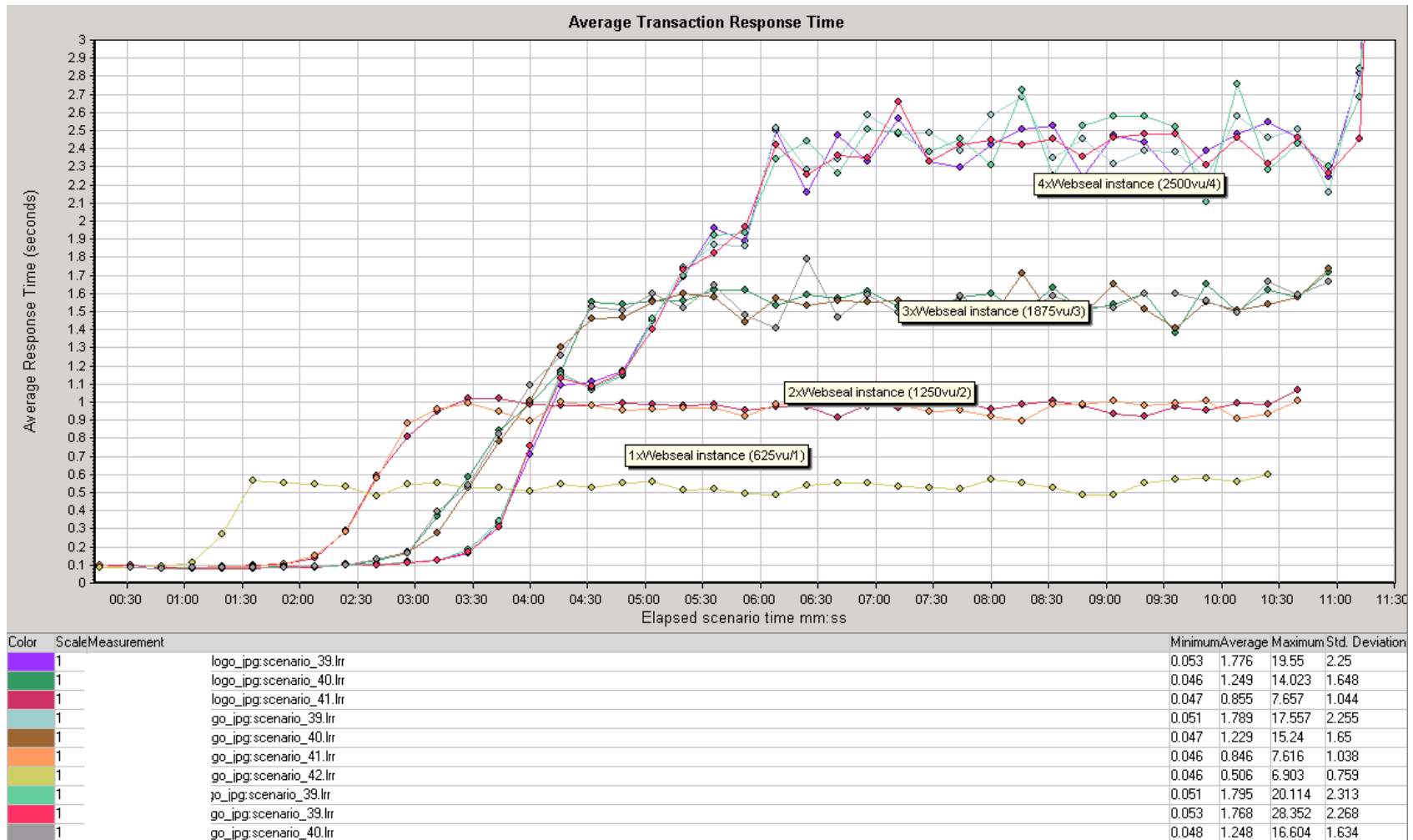
Trafikk - hits pr sec/throughput flater ut ved flere instanser på samme server



Ressursbruk - CPU forbruk øker lineært (ca 20% pr instans) selv om trafikk gjennom flater ut



Responstider - transaksjonstider øker markant ved økt trafikk over flere instanser



Spørsmål???

Takk for meg!

Rune L Skår - rune.skar@knowit.no

<http://www.linkedin.com/in/runels>