



Predictive Modelling of SAP ERP Applications: Challenges and Solutions

Jerry Rolia¹, Giuliano Casale², **Diwakar Krishnamurthy**³, Stephen Dawson², Stephan Kraft²

¹: Automated Infrastructure Lab, HP Labs, Bristol, UK, e-mail: jerry.rolia@hp.com

²: SAP Research, CEC Belfast, UK, e-mail: firstname.lastname@sap.com

³: University of Calgary, Calgary, AB, Canada, e-mail: dkrishna@ucalgary.ca

© 2009 Hewlett-Packard Development Company, L.P. The information contained herein is subject to change without notice

LABS^{hp}

SAP RESEARCH



Outline

- Overview of SAP Architecture
- System under study
- Measurements used
- Models
 - QNM
 - LQM
- Summary and conclusions



SAP RESEARCH

SAP Architecture

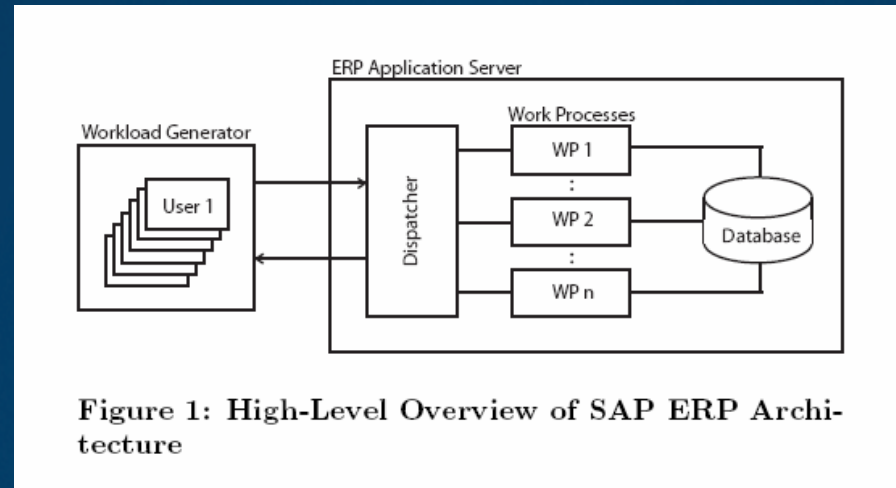


Figure 1: High-Level Overview of SAP ERP Architecture

- A dispatcher receives and routes requests to work processes
- All work processes access the DB synchronously
- There are separate fixed size pools of work processes for
 - Dialog workers: execute transactions for sessions of requests
 - Update workers: cache and commit changes to data by Dialog
 - Update2 workers: lower priority database updates
- Messages by Dialog workers to Update and Update2 workers are asynchronous

System under Study

- Sales and distribution workload running in a virtual machine
- Requests submitted to system by emulated clients with exponentially distributed think times with mean of 10 seconds
- SAP ERP application server and DB running in a virtual machine
- All running on a single physical host with 4 cores @ 2.2 GHz, 32GB of RAM and 230 GB of storage
- Emulated client vm has one virtual CPU and is pinned to one CPU
- Application server and DB server vm has 2 virtual CPUs and is pinned to two CPU
- Remaining physical CPU not used
- Main factors
 - Number of clients
 - Number of dialog work processes
- Each measurement run took ~ an hour, with 20 min of startup/warm down transients



SAP RESEARCH

Measurements

Metric	Source
dialog step mean response time [ms]	load generator
server CPU utilization	OS monitor SAPoscol
DB process CPU utilization	OS monitor SAPoscol
dialog step CPU service demand [ms]	STAD monitor
update CPU service demand [ms]	STAD monitor
update2 CPU service demand [ms]	STAD monitor
estimated DB CPU service demand [ms]	-
processed dialog transactions	STAD monitor
processed update transactions	STAD monitor
processed update2 transactions	STAD monitor
experiment duration [s]	load generator



SAP RESEARCH



The Measurements

Table 1: Measurements for ERP System with 6 Dialog Work Processes

Metric	Source	10	50	75	100	150	175
dialog step mean response time [ms]	load generator	142.12	151.77	157.15	165.36	209.62	293.11
server CPU utilization	OS monitor SAPoscol	0.08	0.36	0.49	0.60	0.76	0.84
DB process CPU utilization	OS monitor SAPoscol	0.01	0.04	0.05	0.06	0.08	0.08
dialog step CPU service demand [ms]	STAD monitor	119.08	108.72	99.98	91.04	78.07	74.08
update CPU service demand [ms]	STAD monitor	44.11	39.21	35.80	32.62	29.61	28.76
update2 CPU service demand [ms]	STAD monitor	30.06	25.52	22.71	20.51	18.56	18.34
estimated DB CPU service demand [ms]	-	7.48	5.62	5.00	4.40	3.81	3.45
processed dialog transactions	STAD monitor	2522	12433	18610	24812	35933	42537
processed update transactions	STAD monitor	669	3299	4943	6588	9544	11251
processed update2 transactions	STAD monitor	169	822	1235	1646	2378	2806
experiment duration [s]	load generator	2516	2400	2400	2400	2400	2401

Table 2: Measurements for ERP System with 3 Dialog Work Processes

Metric	Source	10	50	75	100	150	175
dialog step mean response time [ms]	load generator	142.08	154.55	164.23	190.56	320.32	473.23
server CPU utilization	OS monitor SAPoscol	0.08	0.36	0.49	0.60	0.76	0.84
DB process CPU utilization	OS monitor SAPoscol	0.01	0.03	0.04	0.05	0.06	0.07
dialog step CPU service demand [ms]	STAD monitor	119.82	109.72	100.94	92.57	79.43	74.90
update CPU service demand [ms]	STAD monitor	47.92	41.94	37.82	35.21	31.02	29.74
update2 CPU service demand [ms]	STAD monitor	32.98	26.81	23.11	21.11	19.18	18.34
estimated DB CPU service demand [ms]	-	6.05	4.71	4.30	3.64	3.17	2.81
processed dialog transactions	STAD monitor	2719	12214	18193	24696	36054	42549
processed update transactions	STAD monitor	721	3248	4840	6566	9513	11268
processed update2 transactions	STAD monitor	181	811	1211	1639	2371	2816
experiment duration [s]	load generator	2637	2401	2401	2401	2401	2401

Demands change with the number of clients

Demands don't change much with the number of dialog work processes



SAP RESEARCH

Predicting demands using cubic splines

- Six dialog work process case
- Data for 10, 100, 175 clients
- Predict demands for 50, 75 and 150 clients

Metric	10	*50	*75	100	*150	175
dialog step CPU service demand [ms]	119.08	105.87	98.11	91.04	79.30	74.08
update CPU service demand [ms]	44.11	39.21	35.80	32.62	29.61	28.76
update2 CPU service demand [ms]	30.06	25.14	22.51	20.51	18.68	18.34
estimated DB CPU service demand [ms]	7.48	5.92	5.08	4.40	3.66	3.45

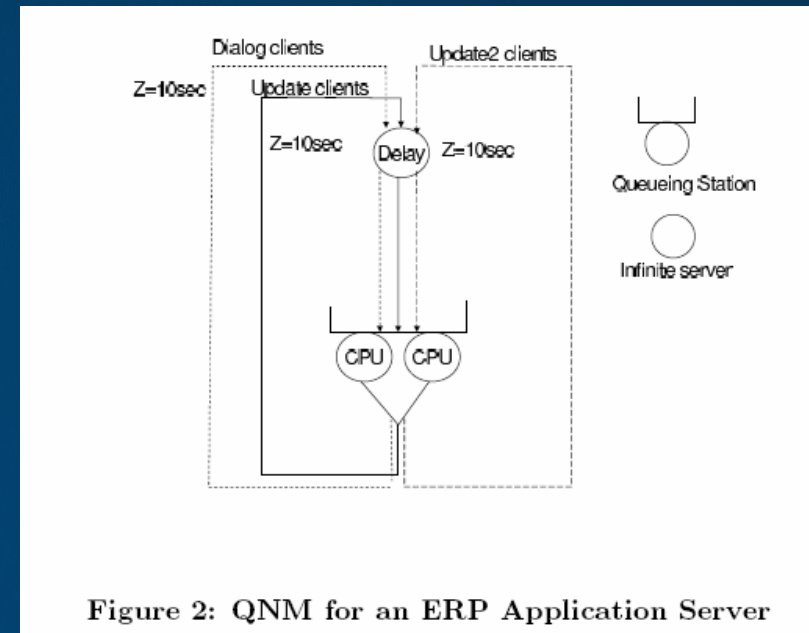
Demands estimated using this approach were within 3.3% of measured values (encouraging)



SAP RESEARCH

QNM

- QNMs being applied to multi-tier systems in the recent literature
 - Try them here...
- Dialog work processes, Update processes and Update2 processes are all customer classes
 - $Z = 10 \text{ sec}$
 - Demands are as measured
- QNM has different service times for 3 classes which is not product-form
- Results for QNM found using simulation



N_{dialog} = number of clients
 N_{update} and N_{update2} are chosen to match measured throughputs

QNM Simulation Results

- Response times are low at low customer populations
 - Some queuing behaviour ignored?
- Response times do not increase with the client population

Table 5: QNM Mean Dialog Response Times and Utilization for 6 Dialog Work Processes Case

Pop	Measured CPU U	Measured DialogResp	QNM (MVA)				
			Dialog Pop.	Update Pop.	Update2 Pop.	CPU Util	DialogResp
10	0.08	142.12	10	3	1	0.07	127.03
50	0.36	151.77	50	13	3	0.32	125.20
75	0.49	157.15	75	20	5	0.44	126.65
100	0.60	165.36	100	27	7	0.53	128.30
150	0.76	209.62	150	40	10	0.68	144.61
175	0.85	293.11	175	46	12	0.75	165.00

QNM with mainly product form features
doesn't predict the behaviour of the system



SAP RESEARCH

LQMs

- LQMs are extended QNMs that take into account software interactions
- LQMs are decomposed into a series of QNMs that are solved iteratively using Approximate MVA (AMVA) based on Linearizer
 - Method of Layers
- Can use residence time expressions for AMVA – e.g., priorities
- Residence time expressions also developed to handle other hardware and software interactions
 - E.g., Synchronous/Asynchronous messaging, multi-processor servers, multi-threaded application servers, 2nd-phase of service

Will LQMs do better than QNMS?
What modelling features matter most?



SAP RESEARCH

Do numbers of Dialog work processes matter?

6 Work process case

Pop	Measured CPU U	Measured DialogResp
10	0.08	142.12
50	0.36	151.77
75	0.49	157.15
100	0.60	165.36
150	0.76	209.62
175	0.85	293.11

3 Work process case

Pop	Measured CPU U	Measured DialogResp
10	0.08	142.08
50	0.36	154.55
75	0.49	164.23
100	0.60	190.56
150	0.76	320.32
175	0.84	473.23

- Does not affect CPU utilization
- Has a big impact on client response times for the higher customer populations

Multi-threading for the ERP application server matters
- not a feature of QNM but is a feature of LQMs



SAP RESEARCH

Are work processes affected differently by load levels?

- Processing time is an elapsed time measure
- Expansion factor = Processing time/CPU time

Client Pop	Processing Time	CPU Time	Expansion Factor
Update transactions			
10	64.393	44.05	1.46
50	69.476	39.206	1.77
75	73.741	35.802	2.06
100	78.113	32.617	2.39
150	99.263	29.64	3.35
Update2 transactions			
10	58.337	30.059	1.94
50	63.72	25.523	2.50
75	67.358	22.713	2.97
100	75.264	20.51	3.67
150	103.58	18.532	5.59
Dialog transactions			
10	111.76	119.1	0.94
50	113.38	108.72	1.04
75	113.16	99.983	1.13
100	112.33	91.042	1.23
150	115.07	78.068	1.47

Impacted

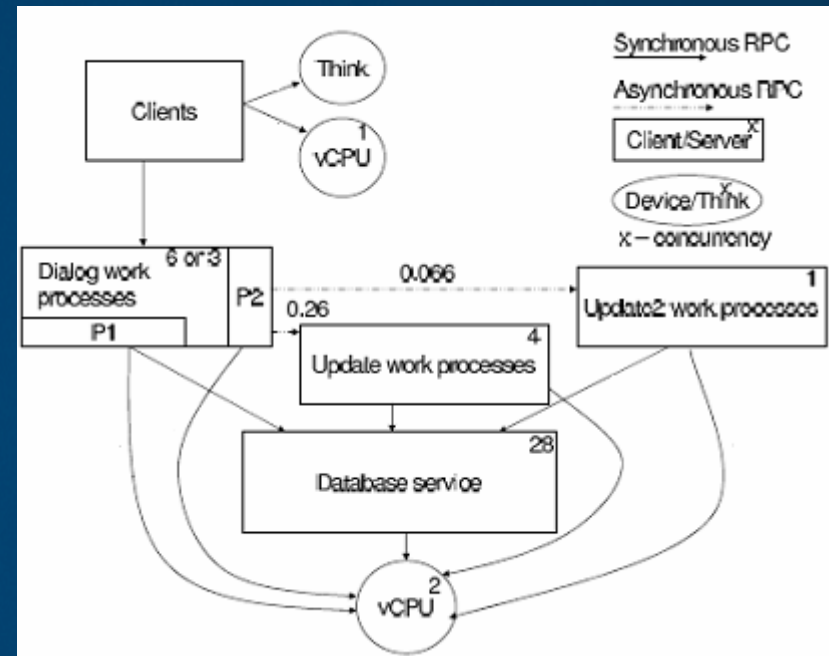
Impacted most

Impacted least

SAP documents suggest that Update2 messages are treated with a lower priority by the dispatcher, which is consistent with these measures

LQM for ERP Application

- Clients access Dialog work processes synchronously
- Dialog work processes have two phases
 - Phase 1 uses vCPU and visits DB synchronously, releases client
 - Phase 2 sends messages to Update and Update2 workers asynchronously
- Update and Update2 processes use vCPU and visit the DB synchronously
- Update2 work processes access the vCPU with a low priority



Population levels were as configured in the system under study
Demands as measured



SAP RESEARCH



LQM Results for various feature sets

6 worker process case

Pop	Measured CPU U	Measured DialogResp	All Features DialogResp	UpdateLowPri DialogResp	No Priority DialogResp	No Async DialogResp	No Threading DialogResp	No P2 DialogResp
10	0.08	142.12	139.67	139.15	139.69	140.04	138.56	139.67
50	0.36	151.77	149.14	145.79	150.98	151.33	146.73	149.23
75	0.49	157.15	158.42	152.20	164.12	162.09	158.48	158.42
100	0.60	165.36	170.63	160.25	183.73	176.27	177.27	171.24
150	0.76	209.62	241.19	205.70	292.89	259.82	286.02	243.40
175	0.85	293.11	329.66	252.45	375.87	378.74	345.78	326.88

3 worker process case

Pop	Measured CPU U	Measured DialogResp	All Features DialogResp	UpdateLowPri DialogResp	No Priority DialogResp	No Async DialogResp	No Thread Dialog Resp	No P2 DialogResp
10	0.08	142.08	135.62	135.09	135.66	136.34	133.99	135.63
50	0.36	154.55	154.96	151.02	158.21	159.85	149.74	154.93
75	0.49	164.23	167.94	161.25	180.36	179.16	164.17	169.09
100	0.60	190.56	180.59	168.23	209.79	201.45	179.87	182.35
150	0.76	320.32	322.84	252.27	696.12	494.59	288.25	349.02
175	0.84	473.23	586.71	375.88	1068.62	824.17	339.32	631.40

- Priorities/async/threading most important
- Update at regular priority, Update2 at low priority worked best (as per documentation)
- Results show that key features together provided best results



SAP RESEARCH



Summary and conclusions

- ERP application servers can have complex behaviour not well captured by QNMs
- Need to consider different load and threading levels when validating models
- Demands can change with load level
- LQMs are extended QNMs for modelling software systems
- We were able to validate a LQM for the complex system under study
- What about using such models at runtime?
 - Accurate predictions were sensitive to 40 min of steady state behaviour
- Next steps
 - Consider more workload mixes & burstiness behaviour



SAP RESEARCH



Related papers

- J. Rolia, D. Krishnamurthy, G. Casale, and S. Dawson "BAP: A Benchmark-driven Algebraic Method for the Performance Engineering of Customized Services", WOSP/SIPEW International Conference on Performance Engineering, January 2010
- D. Krishnamurthy, J. Rolia, and M. Xu. "WAM - The Weighted Average Method for Predicting the Performance of Systems with Bursts of Customer Sessions," IEEE Transactions on Software Engineering.
- J. Rolia, A. Kalbasi, D. Krishnamurthy, and S. Dawson "Resource Demand Modeling for Multi-Tier Services", WOSP/SIPEW International Conference on Performance Engineering, January 2010
- G. Casale, A. Kalbasi, D. Krishnamurthy, and J. Rolia "Automatic Stress Testing of Multi-Tier Systems by Dynamic Bottleneck Switch Generation", Middleware 2009, November/December 2009
- G. Casale, A. Kalbasi, D. Krishnamurthy, and J. Rolia. "Automatically Generating Bursty Benchmarks for Multi-Tier Systems," HotMetrics 2009, June 2009
- D. Krishnamurthy, J. Rolia, and S. Majumdar. "A Synthetic Workload Generation Technique for Stress Testing Session-Based Systems," IEEE Transactions on Software Engineering, Vol. 32, No. 11, pp. 868-882, November 2006.



SAP RESEARCH



SAP RESEARCH



UNIVERSITY OF
CALGARY