Badanie wydajności struktur klastrowych

Tomasz RAK

Department of Computer and Control Engineering
Rzeszow University of Technology

Tuesday 5th June, 2018

We need to calculate the web cluster performance.

RZESZOW UNIVERSITY OF TECHNOLOGY

Table of contents

Table of contents

Previous works (1)

- Model evolution: QN, TCPN and QPN
- System architecture - DayTrader (e-trading system as a benchmark)
- Performance analysis - QPN model

Previous works (2)

QPN formalism [a]

[a] Rak T.: **Performance Modeling Using Queueing Petri Nets**, Communications in Computer and Information Science, Springer, pp. 321-335 (**2017**) doi:10.1007/978-3-319-59767-6

Rak T.: **Response Time Analysis of Distributed Web Systems Using QPNs**, Mathematical Problems in Engineering, Hindawi Limited, Volume 2015, Article ID 4 (**2015**) doi:10.1155/2015/490835

Rak T.: **Performance Analysis of Cluster-Based Web System Using the QPN Models**, Information Sciences and Systems, Springer, pp. 239-247 (**2014**) doi:10.1007/978-3-319-09465-6_25

Rak T.: **Performance Analysis of Distributed Internet System Models Using QPN Simulation**, Computer Science and Information Systems (FedCSIS) (**2014**) doi:10.15439/2014F366

Related works:
- Chen X., Kounev S., Koziolek H., Meier P., Rathfelder C., Spinner S., Zatwarnicki K.
- Requeno, J. I., Merseguer, J., Bernardi, S., Zhou, J., Reniers, G., Su, J., Huang, G., Mironescu, I., Vintan, L., Li, Z., Jiao, L., Hu, X., Chongyuan, H., Hanhong, J., Yuan, Y., Wanzhi, R., Lianwu, H.

The question:

What is the system performance?

The main aim of my work was to develop models of cluster-based distributed Web system. The related works can be divided into publications based on analysis of QN and PN models.

Table of contents

Cluster n-tier architecture

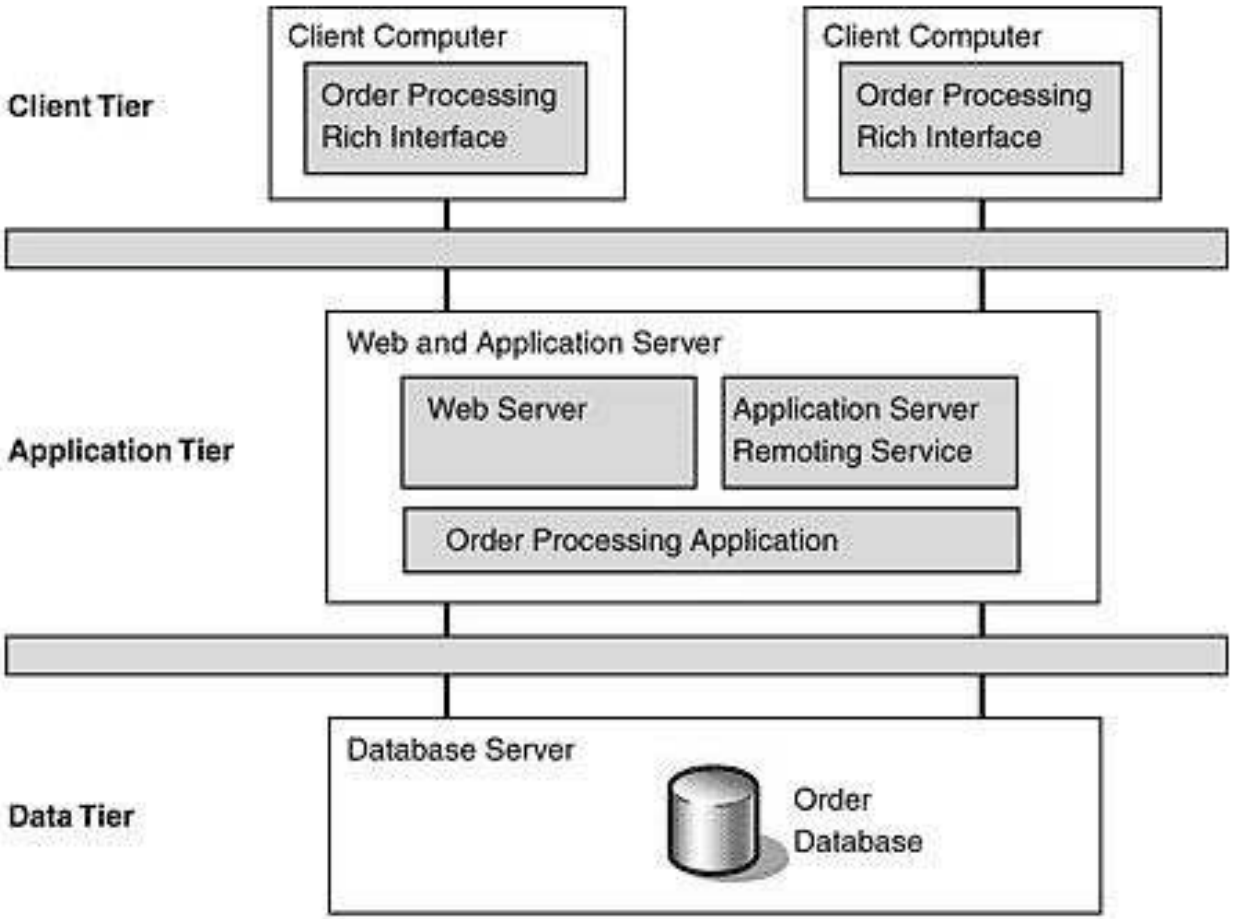


Table of contents

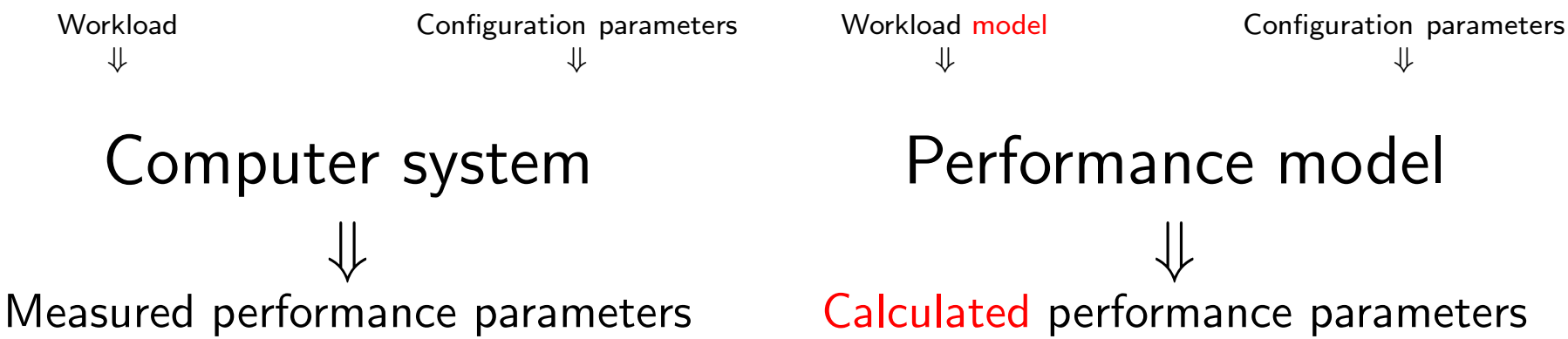
My approach

- Educated Guess
- Load Testing
- Performance Engineering models:
  - Performance model
  - Availability model
  - Reliability model
  - Cost model

Categories

- Performance Prediction
- Performance Verification
- Capacity Analysis
- Scalability Analysis
- Bottleneck Analysis
- Performance Tuning
- Performance Optimalization
- Cost/Performance Analysis
- Sizing and Capacity Planning

Computer system (experiments) and performance model (simulations) – response time parameter



Queueing Nets and Petri Nets (combination)

QNs – quantitative analysis

Queueing Nets have a queue, scheduling discipline.

PNs – qualitative analysis

Petri Nets have tokens representing the tasks.

QPNs add queueing and time aspects to the net

Queueing Petri Nets have the advantages of Queueing Nets and Petri Nets

Parameters that determine the response time

- Workload intensity, hardware and software parameters
- Queueing time, service demand was determined experimentally: $d_{FE\_CPU} = 1/RPS_{FE\_CPU} = 0,714$ [ms], $d_{BE\_I/O} = 1/RPS_{BE\_I/O} = 0,133$ [ms]

Table of contents

Description

- Servers of the FE layer are modeled using -/M/1/PS/∞ queueing systems (*FE_CPU* queueing places). The BE servers are modeled by -/M/1/FIFO/∞ queueing systems (*BE_I/O* queueing places).
- *FE* and *BE* places are used to stop incoming requests when they await application server threads and database server connections respectively.
- Clients think time is modeled by IS scheduling strategy (*CLIENTS* queueing place).
- Application server threads pool and database server connections pool are modeled respectively by *THREADS* and *CONNECTIONS* places.

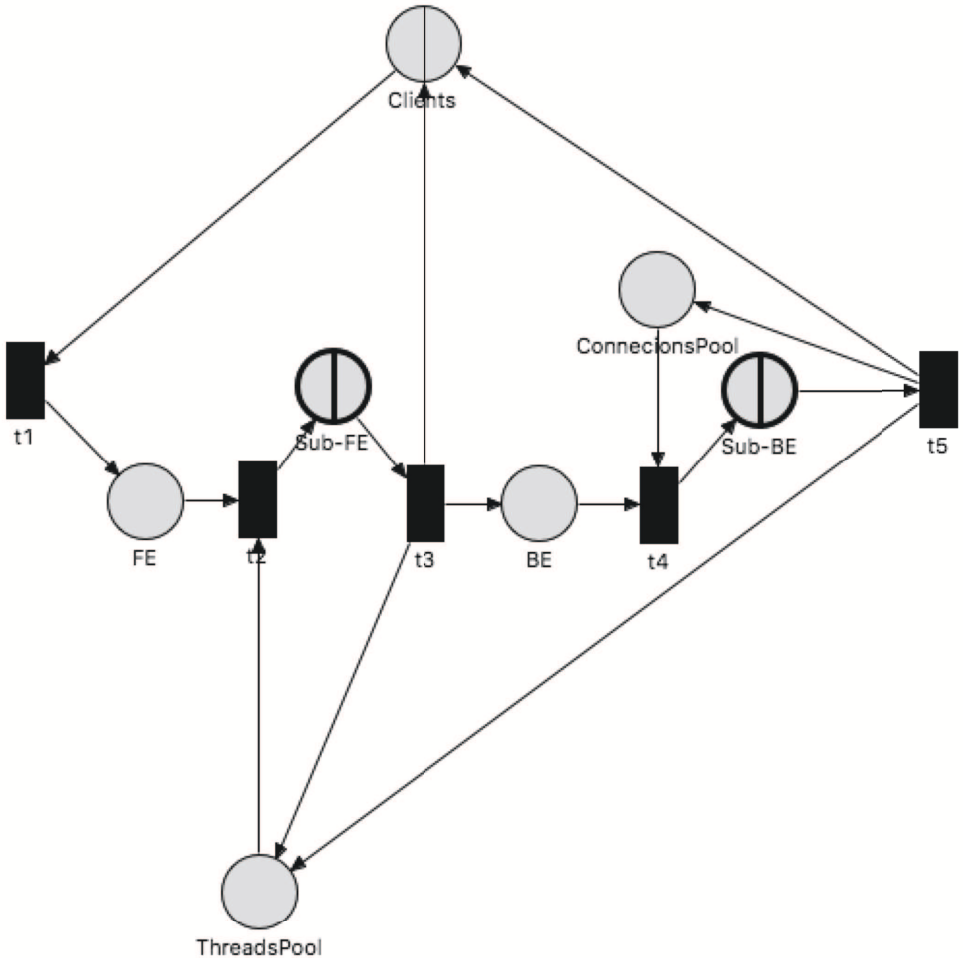
-/M/1/PS/∞ – randomly arriving requests, exponential service rate, one server, scheduling strategy, infinite queue.

Input and workload parameters

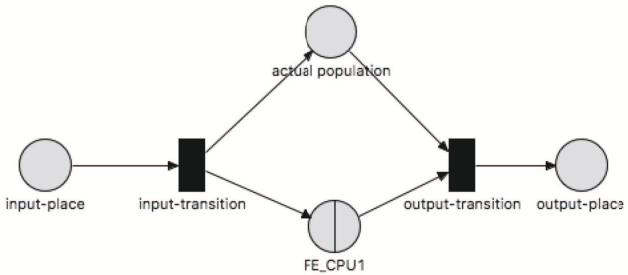
Parameter	Value for one class (x1)	Values for two classes (x1 and x2)
d_{FE_CPUn} [ms]	0.714 (1400 [RPS])	0.714 (1400 [RPS]) or 0.0714 (14000 [RPS])
$d_{BE\_I/O}$ [ms] [a]	0.133 (7500 [RPS])	0.133 (7500 [RPS])
THREADS place	30[b]	3000[c]
CONNECTIONS place	40[d]	4000[e]
CLIENTS queueing place	500 or 250+250	250+250 or 2500+250
$d_{CLIENTS}$ [ms]	15 [RPS]; 30 [RPS]; 45 [RPS]; 60 [RPS]	7.5 and 7.5 [RPS]; 15 and 15 [RPS]; 22.5 and 22.5 [RPS]; 30 and 30 [RPS]
Simulation time [s]	300	300

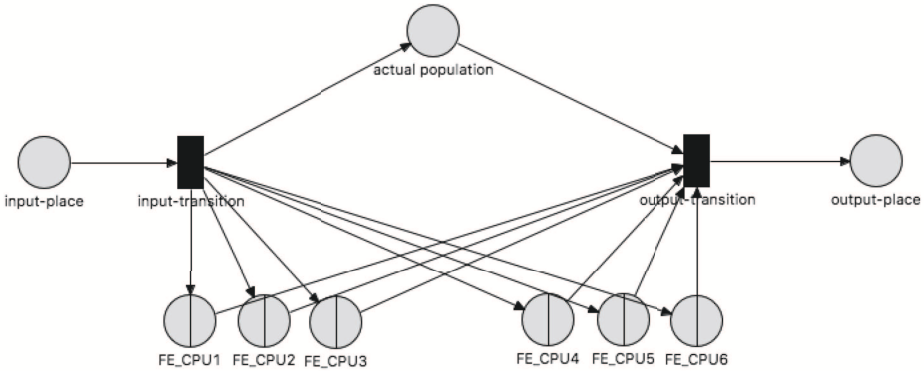
[a] n – number of FE and BE nodes (1, 3, 6, 9, 12)
[b] 30, 60 and 90 threads for one, two and three FE nodes respectively – Initial marking per node
[c] 3000, 9000 and 18000 threads for one, three and six FE nodes respectively – Initial marking per node
[d] 40, 80 and 120 connections for one, two and three BE nodes respectively – Initial marking per node
[e] 4000, 12000 and 24000 connections for one, three and six BE nodes respectively – Initial marking per node

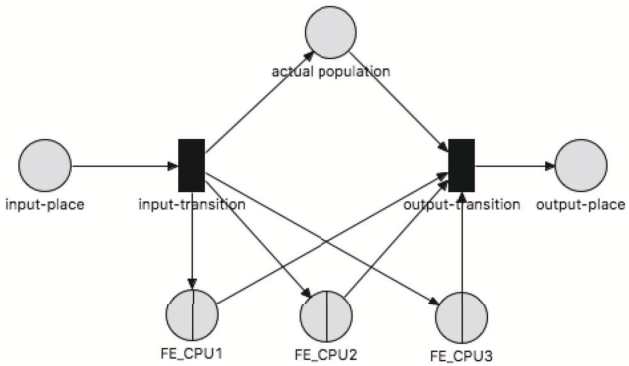
Main page

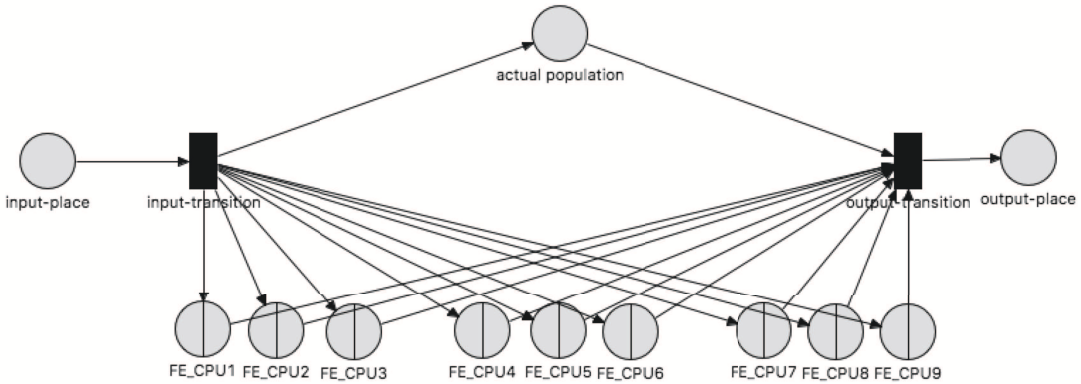

Clients
ConnectionsPool
t1
Sub-FE
Sub-BE
t5
FE
t2
t3
BE
t4
ThreadsPool

Sub-FE page

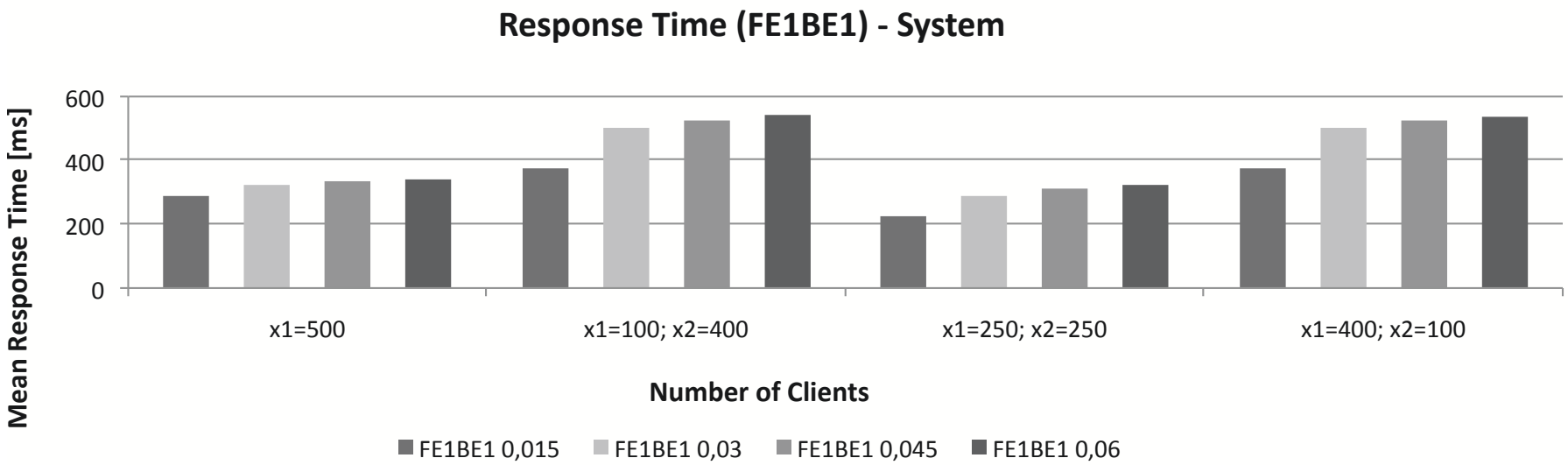

actual population
input-place
input-transition
output-transition
output-place
FE_CPU1


actual population
input-place
input-transition
output-transition
output-place
FE_CPU1
FE_CPU2
FE_CPU3
FE_CPU4
FE_CPU5
FE_CPU6

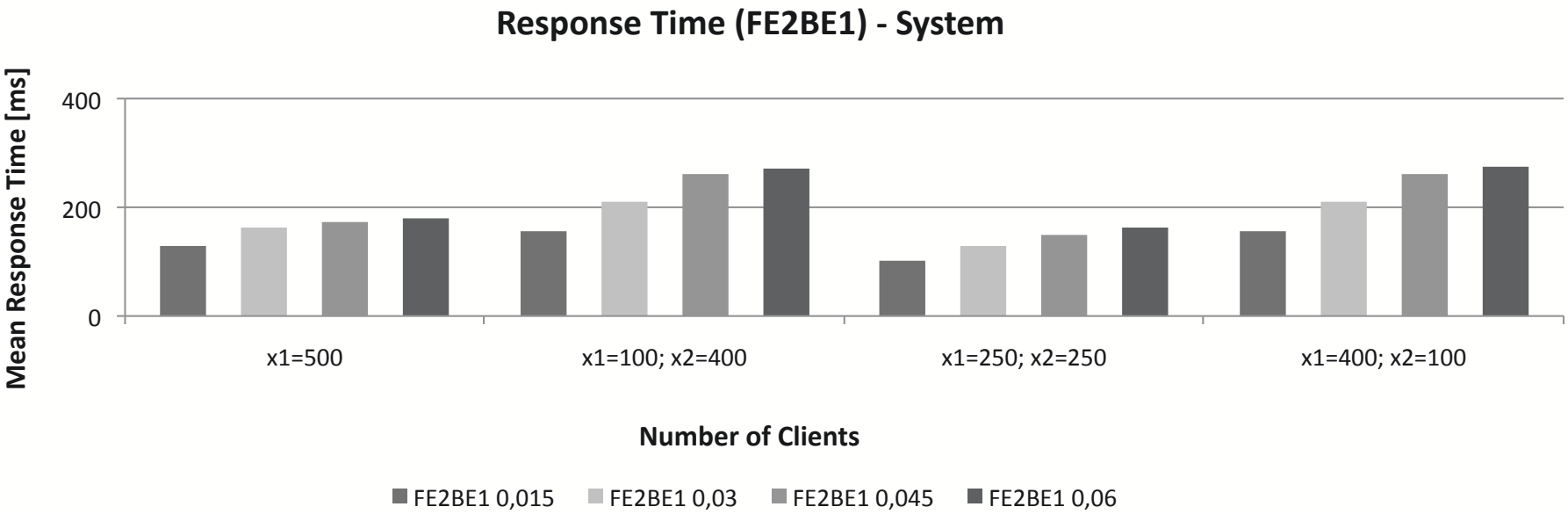

actual population
input-place
input-transition
output-transition
output-place
FE_CPU1
FE_CPU2
FE_CPU3


actual population
input-place
input-transition
output-transition
output-place
FE_CPU1
FE_CPU2
FE_CPU3
FE_CPU4
FE_CPU5
FE_CPU6
FE_CPU7
FE_CPU8
FE_CPU9

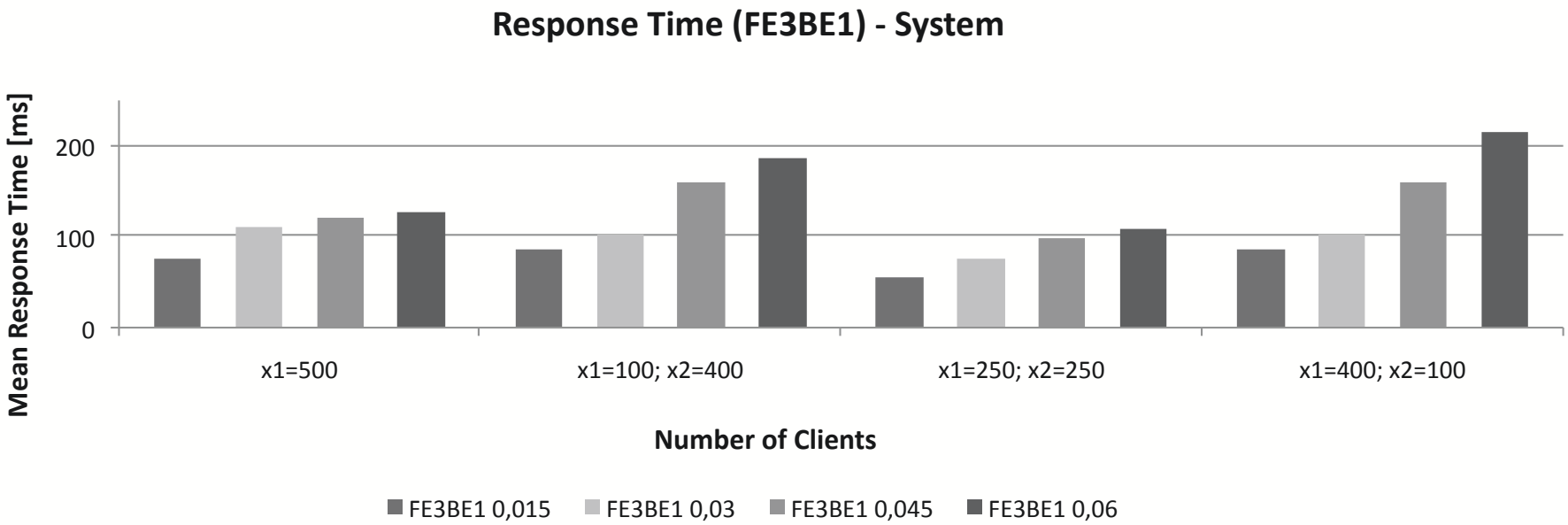
Response time for one and two client classes - FE1 (500 clients, different RPS workload)



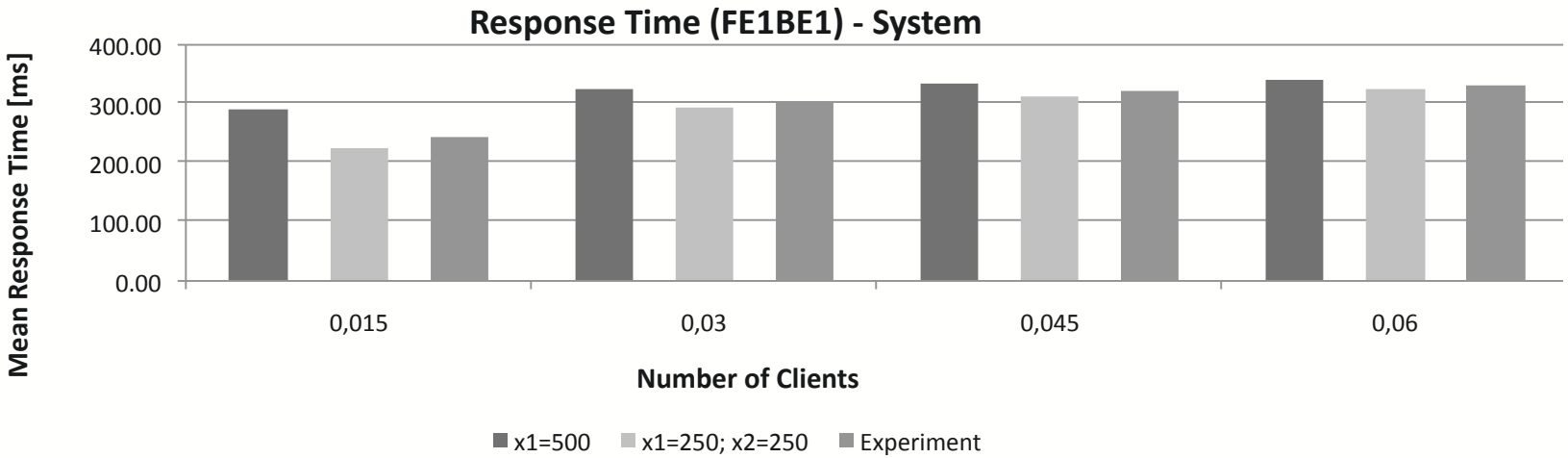
Response time for one and two client classes - FE2 (500 clients, different RPS workload)



Response time for one and two client classes - FE3 (500 clients, different RPS workload)

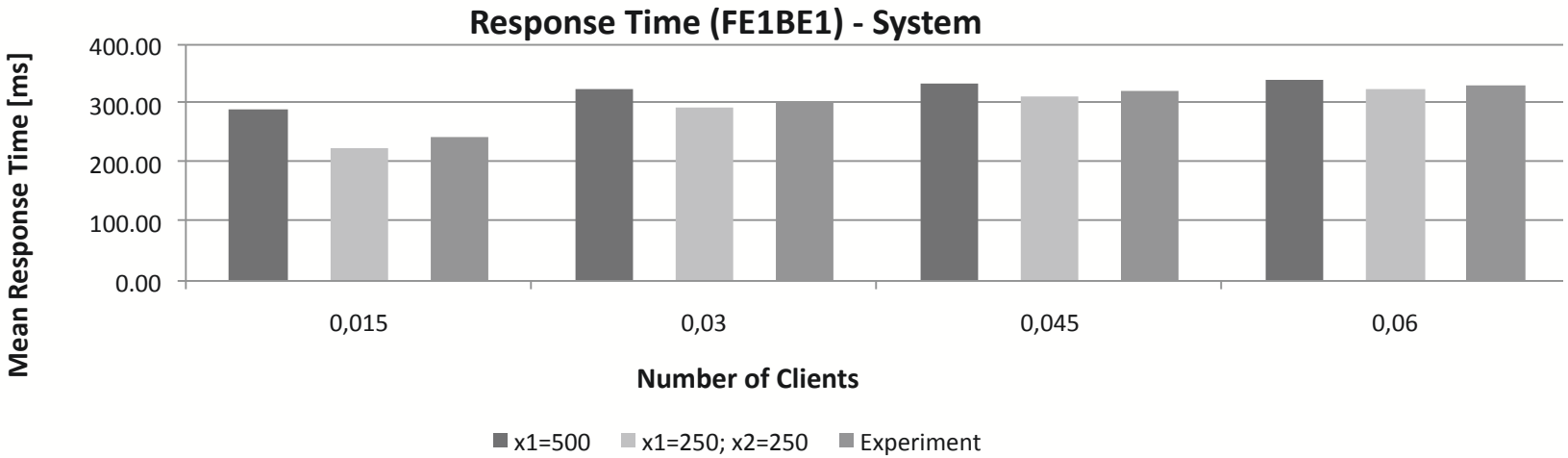


Response time error FE1 (500 clients, different RPS workload)



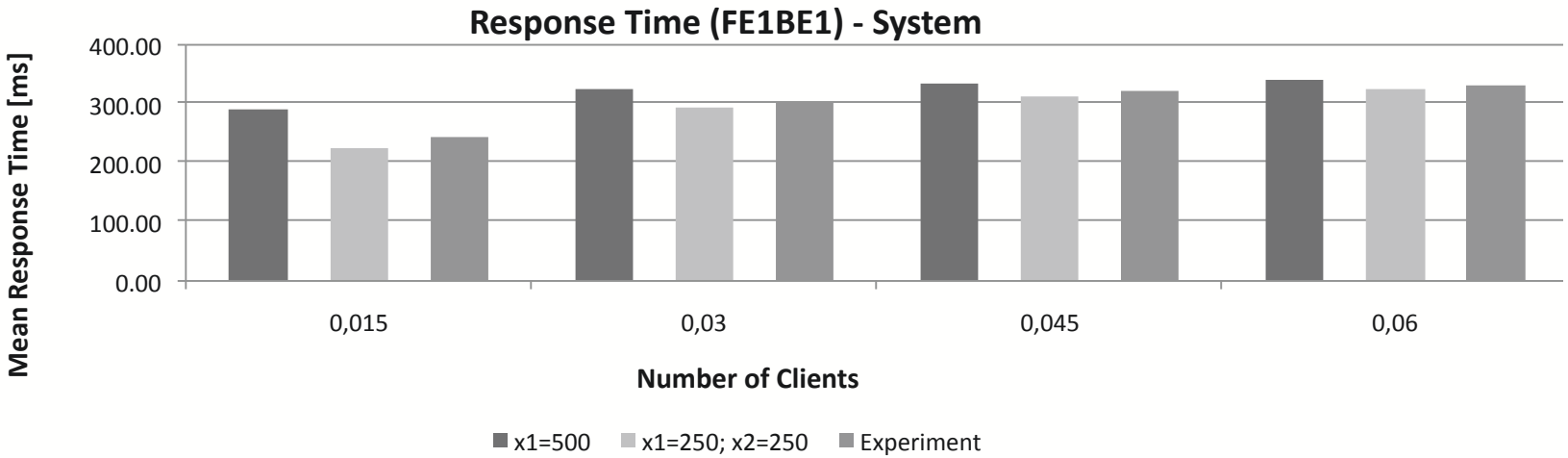
Client think time [ms]	Model with one client class [ms]	Model with two client classes [ms]	Measured [ms]	Error for one client class [%]	Error for two client classes [%]
133.33 and 133,33	291,14	224,58	241,0	20,63	6,94
66,67 and 66,67	323,46	291,87	303,0	6,55	3,85
44,44 and 44,44	334,45	312,62	321,0	4,16	2,64
33,33 and 33,33	340,31	324,71	330,0	2,98	1,74

Response time error FE2 (500 clients, different RPS workload)



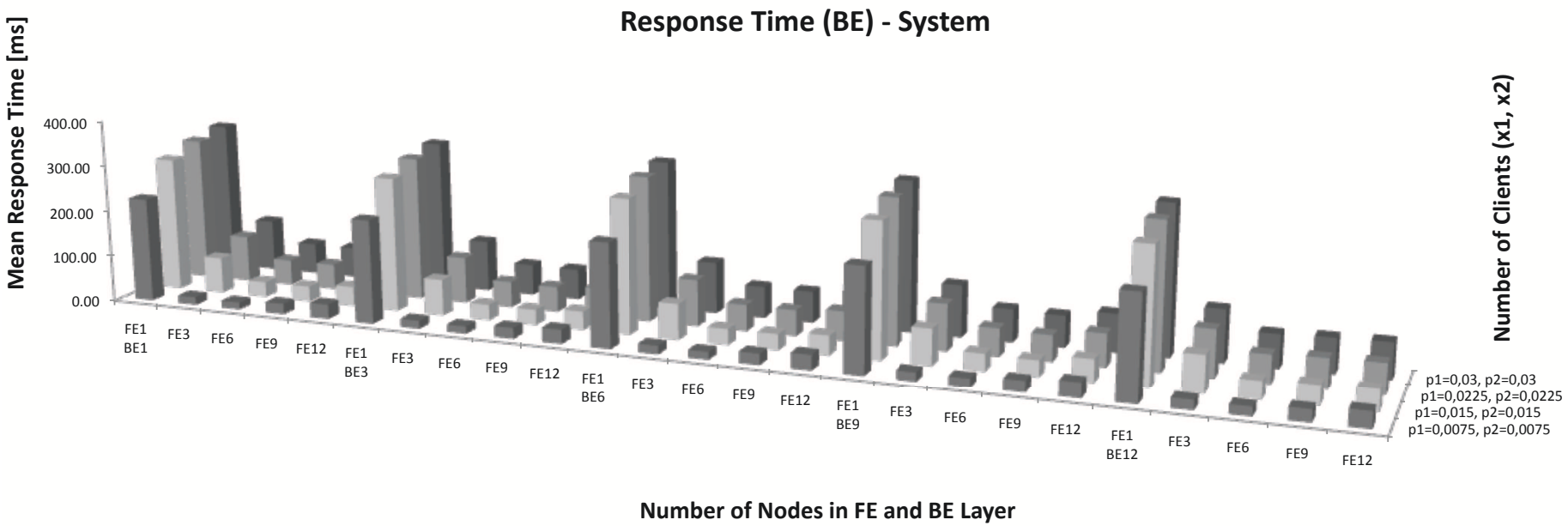
Client think time [ms]	Model with one client class [ms]	Model with two client classes [ms]	Measured [ms]	Error for one client class [%]	Error for two client classes [%]
133.33 and 133,33	128,58	102,11	106,66	20,55	4,27
66,67 and 66,67	162,36	128,44	135,01	20,26	4,87
44,44 and 44,44	173,49	150,63	159,42	8,83	5,51
33,33 and 33,33	178,48	162,33	175,43	1,74	7,47

Response time error FE3 (500 clients, different RPS workload)

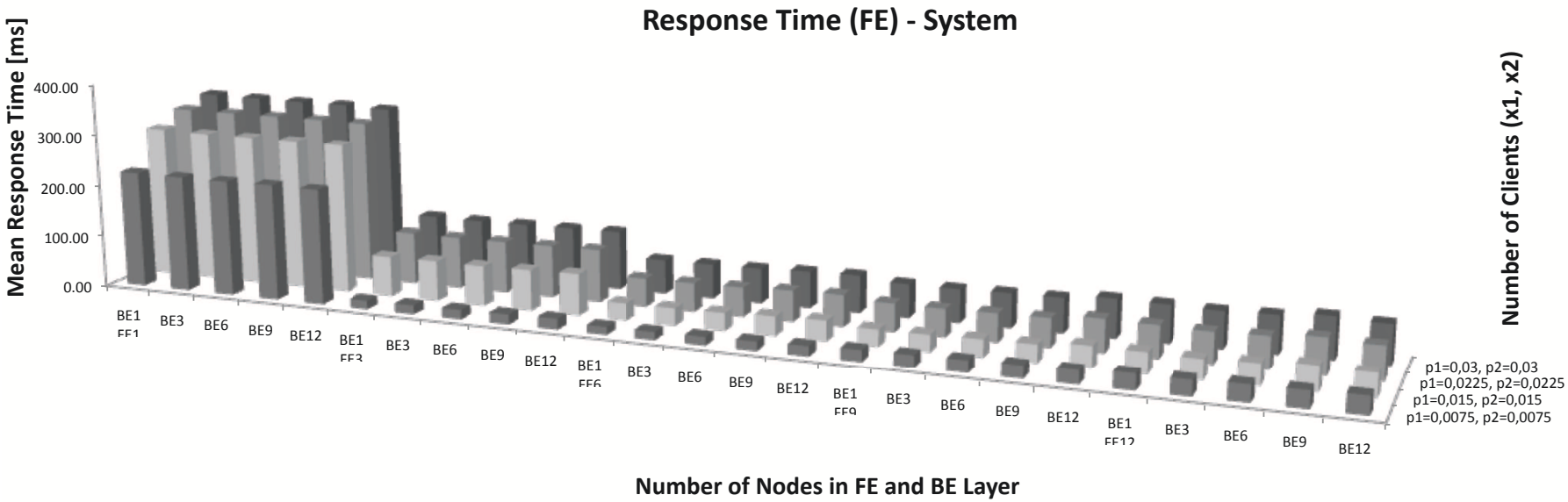


Client think time [ms]	Model with one client class [ms]	Model with two client classes [ms]	Measured [ms]	Error for one client class [%]	Error for two client classes [%]
133.33 and 133,33	76,46	56,23	65,12	17,41	13,65
66,67 and 66,67	110,32	76,78	85,28	29,36	9,96
44,44 and 44,44	121,23	99,38	110,83	9,38	10,34
33,33 and 33,33	126,59	109,76	120,94	4,68	9,25

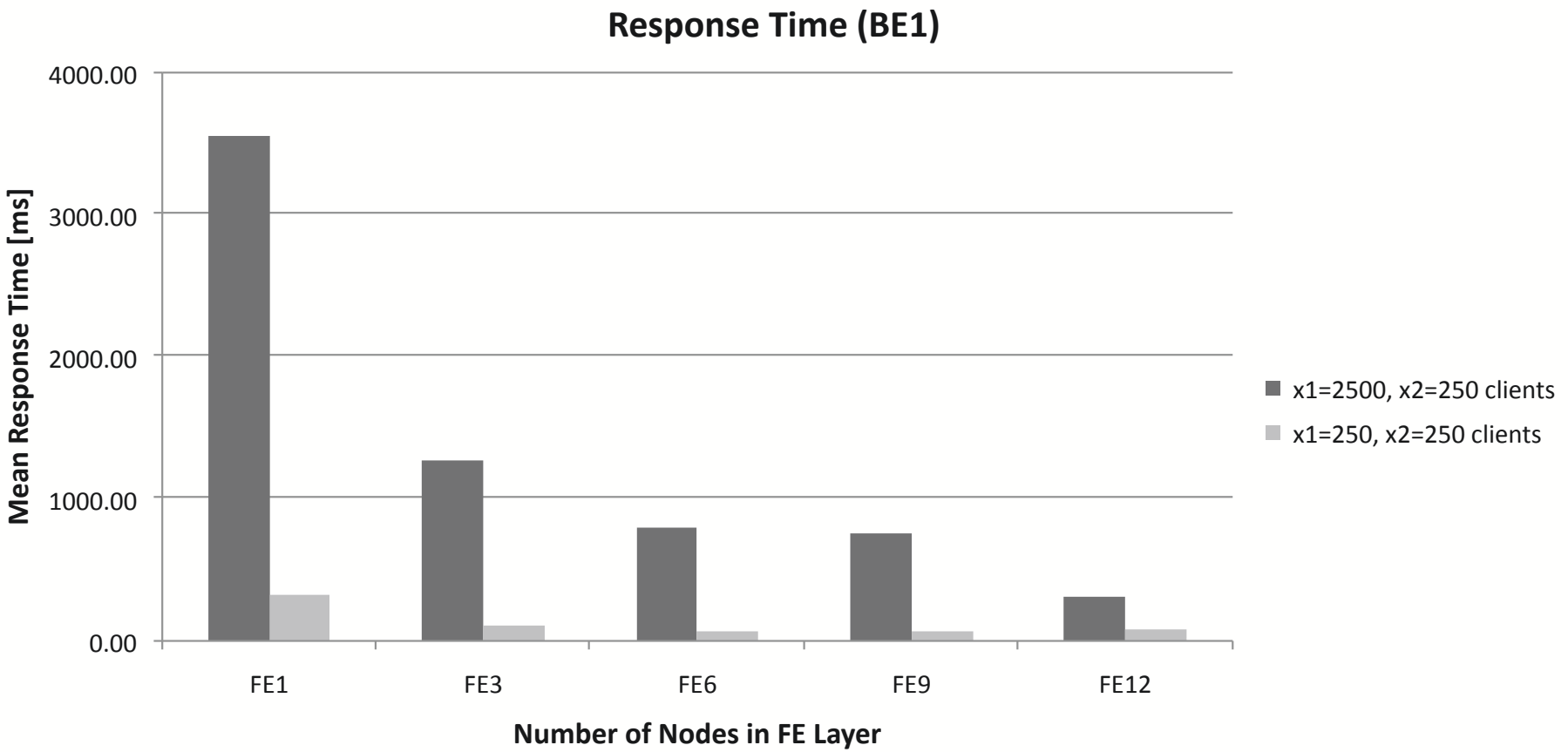
Response time for different number of nodes in BE and FE Layer



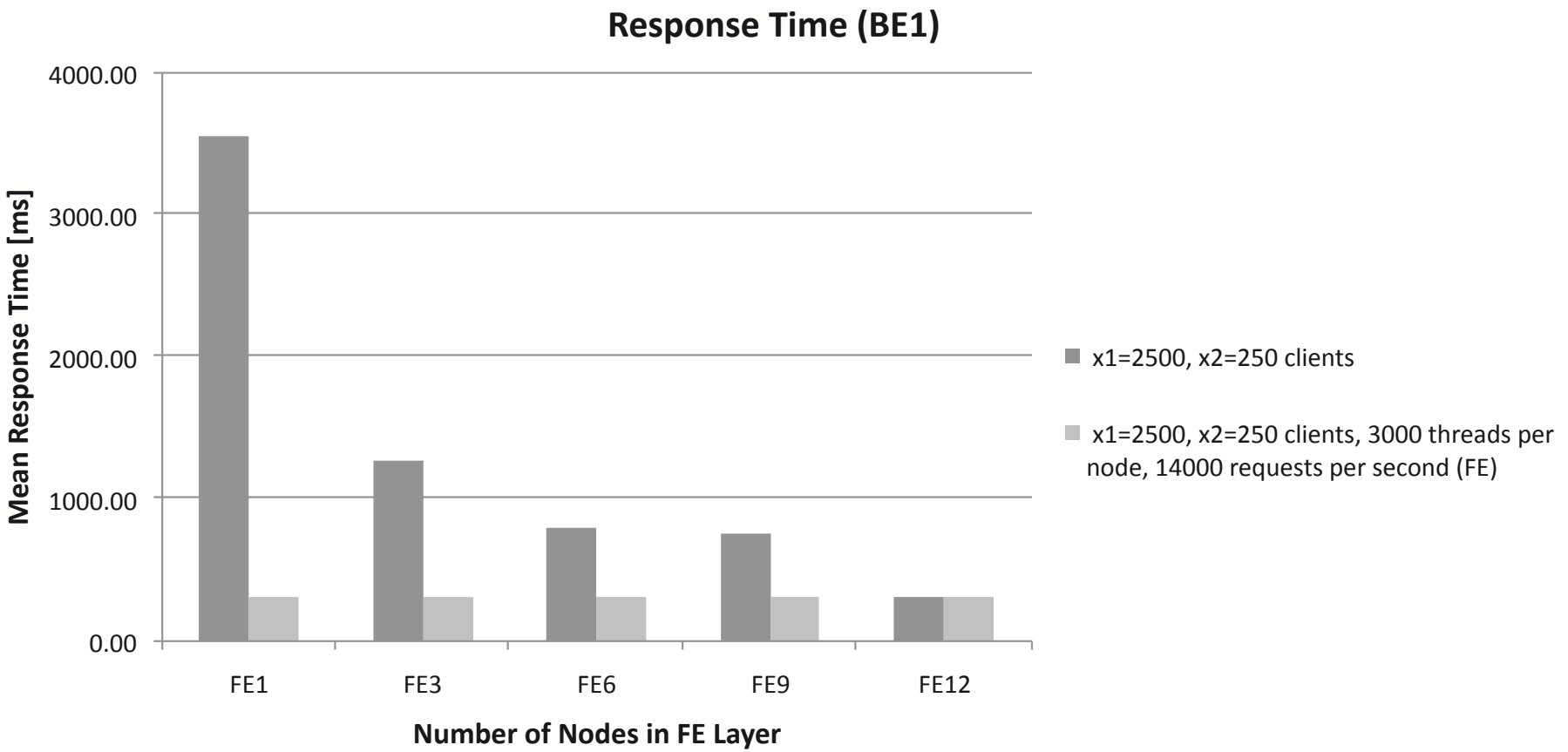
Response time for different number of nodes in FE and BE Layer



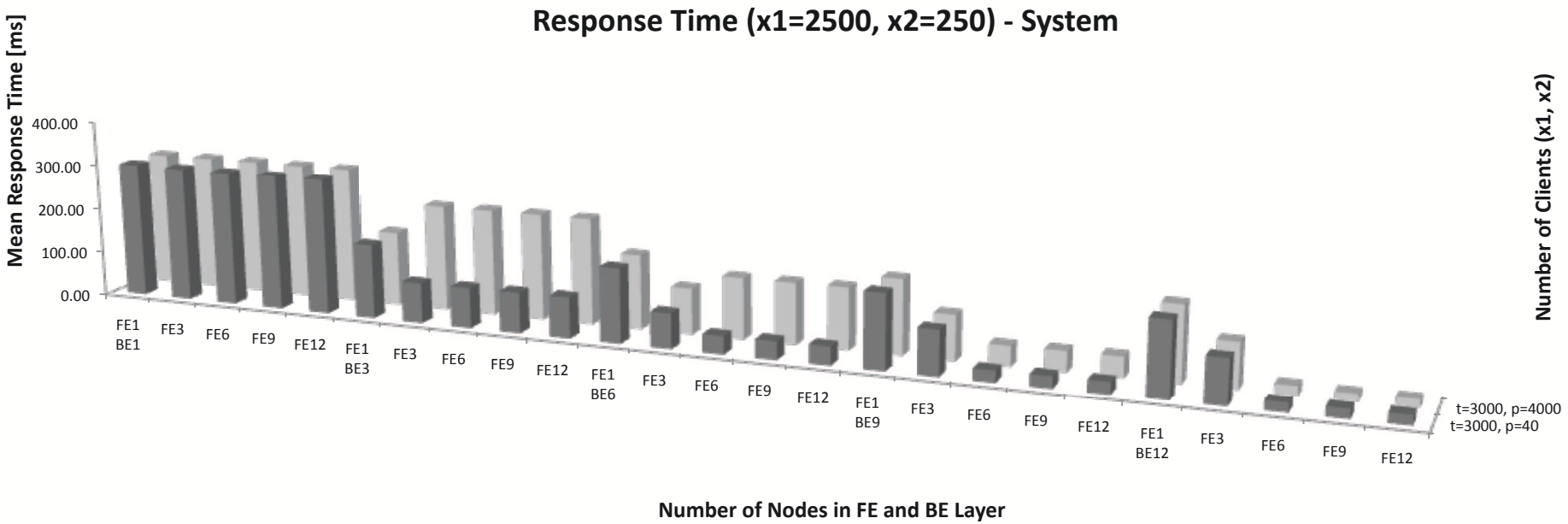
Response time for x1=2500 clients (BE1)



Response time for x1=2500 clients (BE1), 3000 threads per node in FE layer



Response time for x1=2500 clients, 3000 threads per FE node, 0,071 FE service demand, 4000 process per BE node



t=3000 per node, p=40 per node (12_2)

		Sub-FE.FE_CPU (qplace:queue):	Sub-FE.FE_CPU (qplace:depository):	BE (place):	Sub-BE.BE_IO (qplace:queue):	Sub-BE.BE_IO (qplace:depository):
BE1	FE1	5,12	0,00	290,58	5,35	0,00
BE1	FE3	0,28	0,23	294,46	5,34	0,00
BE1	FE6	0,47	0,69	293,76	5,33	0,00
BE1	FE9	0,68	1,26	294,25	5,34	0,00
BE1	FE12	0,89	1,90	294,06	5,34	0,00
BE3	FE1	162,69	0,00	0,00	0,67	0,35
BE3	FE3	0,49	0,39	82,27	3,47	2,05
BE3	FE6	0,50	0,74	82,14	3,48	2,03
BE3	FE9	0,69	1,14	82,25	3,47	2,05
BE3	FE12	0,90	1,91	82,20	3,48	2,04
BE6	FE1	162,41	0,00	0,00	0,86	1,25
BE6	FE3	36,04	22,19	13,47	2,78	3,03
BE6	FE6	0,63	0,90	32,58	2,79	2,99
BE6	FE9	0,74	1,36	32,59	2,40	3,04
BE6	FE12	0,92	1,96	32,37	2,79	3,02
BE9	FE1	162,45	0,00	0,00	1,22	2,23
BE9	FE3	59,90	34,71	0,06	1,97	3,26
BE9	FE6	0,83	18,36	17,19	2,72	3,99
BE9	FE9	0,81	1,48	17,03	2,72	3,99
BE9	FE12	0,96	2,03	17,11	2,72	3,97
BE12	FE1	162,28	0,00	0,00	1,61	3,38
BE12	FE3	60,35	34,44	0,03	2,04	4,08
BE12	FE6	1,10	1,51	10,27	2,79	4,95
BE12	FE9	1,98	1,59	10,45	2,79	4,96
BE12	FE12	1,00	2,09	10,39	2,79	4,95

t=3000 per node, p=4000 per node (12_3)

		Sub-FE.FE_CPU (qplace:queue):	Sub-FE.FE_CPU (qplace:depository):	Sub-BE.BE_IO (qplace:queue):	Sub-BE.BE_IO (qplace:depository):
BE1	FE1	5,52	0,00	294,17	0,00
BE1	FE3	0,28	0,23	299,92	0,00
BE1	FE6	0,47	0,69	300,20	0,00
BE1	FE9	0,68	1,26	299,06	0,00
BE1	FE12	0,89	1,90	299,02	0,00
BE3	FE1	162,76	0,00	0,67	0,36
BE3	FE3	0,53	0,42	150,47	82,24
BE3	FE6	0,51	0,74	150,26	82,56
BE3	FE9	0,69	1,14	148,54	84,06
BE3	FE12	0,90	1,91	150,24	82,03
BE6	FE1	162,22	0,00	0,86	1,25
BE6	FE3	59,30	33,20	3,93	4,81
BE6	FE6	0,75	1,06	65,32	68,05
BE6	FE9	0,78	1,43	65,78	67,71
BE6	FE12	0,95	2,00	65,25	68,32
BE9	FE1	162,27	0,00	1,22	2,24
BE9	FE3	59,87	34,84	2,02	3,43
BE9	FE6	1,83	2,50	16,24	24,62
BE9	FE9	0,99	1,78	16,96	26,47
BE9	FE12	1,05	2,20	16,93	26,20
BE12	FE1	162,45	0,00	1,61	3,37
BE12	FE3	60,20	34,66	2,05	4,16
BE12	FE6	2,98	4,01	5,01	9,11
BE12	FE9	1,10	1,96	5,06	9,32
BE12	FE12	1,10	2,30	4,95	9,30

t=3000 per node, p=40 per node (12_2)
t=3000 per node, p=4000 per node (12_3)

		FEq	FEq	FEd	FEd	BE	BE	BEq	BEq	BEd	BEd
BE1	FE1	5,12	5,52	0,00	0,00	290,58	0,00	5,35	294,17	0,00	0,00
BE1	FE3	0,28	0,28	0,23	0,23	294,46	0,00	5,34	299,92	0,00	0,00
BE1	FE6	0,47	0,47	0,69	0,69	293,76	0,00	5,33	300,20	0,00	0,00
BE1	FE9	0,68	0,68	1,26	1,26	294,25	0,00	5,34	299,06	0,00	0,00
BE1	FE12	0,89	0,89	1,90	1,90	294,06	0,00	5,34	299,02	0,00	0,00
BE3	FE1	162,69	162,76	0,00	0,00	0,00	0,00	0,67	0,67	0,35	0,36
BE3	FE3	0,49	0,53	0,39	0,42	82,27	0,00	3,47	150,47	2,05	82,24
BE3	FE6	0,50	0,51	0,74	0,74	82,14	0,00	3,48	150,26	2,03	82,56
BE3	FE9	0,69	0,69	1,14	1,14	82,25	0,00	3,47	148,54	2,05	84,06
BE3	FE12	0,90	0,90	1,91	1,91	82,20	0,00	3,48	150,24	2,04	82,03
BE6	FE1	162,41	162,22	0,00	0,00	0,00	0,00	0,86	0,86	1,25	1,25
BE6	FE3	36,04	59,30	22,19	33,20	13,47	0,00	2,78	3,93	3,03	4,81
BE6	FE6	0,63	0,75	0,90	1,06	32,58	0,00	2,79	65,32	2,99	68,05
BE6	FE9	0,74	0,78	1,36	1,43	32,59	0,00	2,78	65,78	3,04	67,71
BE6	FE12	0,92	0,95	1,96	2,00	32,37	0,00	2,79	65,25	3,02	68,32
BE9	FE1	162,45	162,27	0,00	0,00	0,00	0,00	1,22	1,22	2,23	2,24
BE9	FE3	59,90	59,87	34,71	34,84	0,06	0,00	1,97	2,02	3,26	3,43
BE9	FE6	0,83	1,83	18,36	2,50	17,19	0,00	2,72	16,24	3,99	24,62
BE9	FE9	0,81	0,99	1,48	1,78	17,26	0,00	2,72	16,96	3,99	26,47
BE9	FE12	0,96	1,05	2,03	2,20	17,11	0,00	2,72	16,93	3,97	26,20
BE12	FE1	162,28	162,45	0,00	0,00	0,00	0,00	1,61	1,61	3,38	3,37
BE12	FE3	60,35	60,20	34,44	34,66	0,03	0,00	2,04	2,05	4,08	4,16
BE12	FE6	1,10	2,98	1,51	4,01	10,27	0,00	2,79	5,01	4,95	9,11
BE12	FE9	1,98	1,10	1,59	1,96	10,45	0,00	2,79	5,06	4,96	9,32
BE12	FE12	1,00	1,10	2,09	2,30	10,39	0,00	2,79	4,95	4,95	9,30

Conclusions

- We can use this analysis to apply the systems modification without interfering in the computer construction or in software. (main achievement)
- For the service response times, the relative prediction error was around 15%.
- The modeling approach presented in this paper differs from my previous works that they were based on two classes of requests (requests classes were not used earlier in simulations).

Future Works

- Using model parameters used by other researchers
- Experimental designation of service demand for different request classes

Performance Analysis and Prediction of Cluster Structures (FedCSIS'2018[a])

[a] WSC'18 - 10th Workshop on Scalable Computing

Thank you for your attention!

> Suppose one of you wants to build a tower. Won't you first sit down and estimate the cost to see if you have enough money to complete it? – The Bible, Luke 14:28