

Instituto Tecnológico de Costa Rica

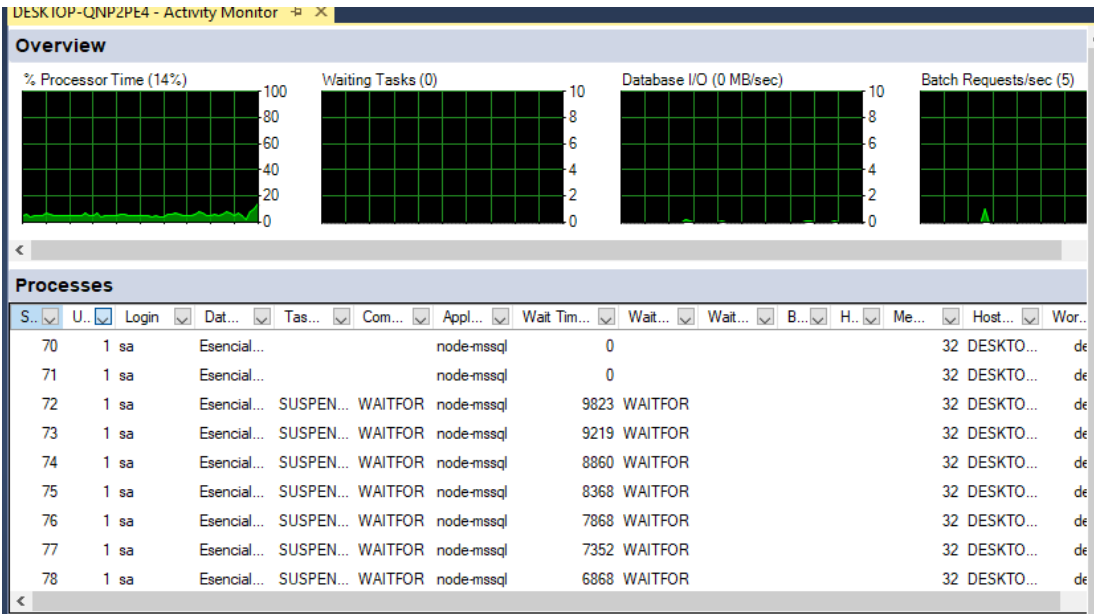
Bases de datos I

Preeliminar #4 - Caso #3

Mauricio Fernández Brizuela

Diego Mora Montes

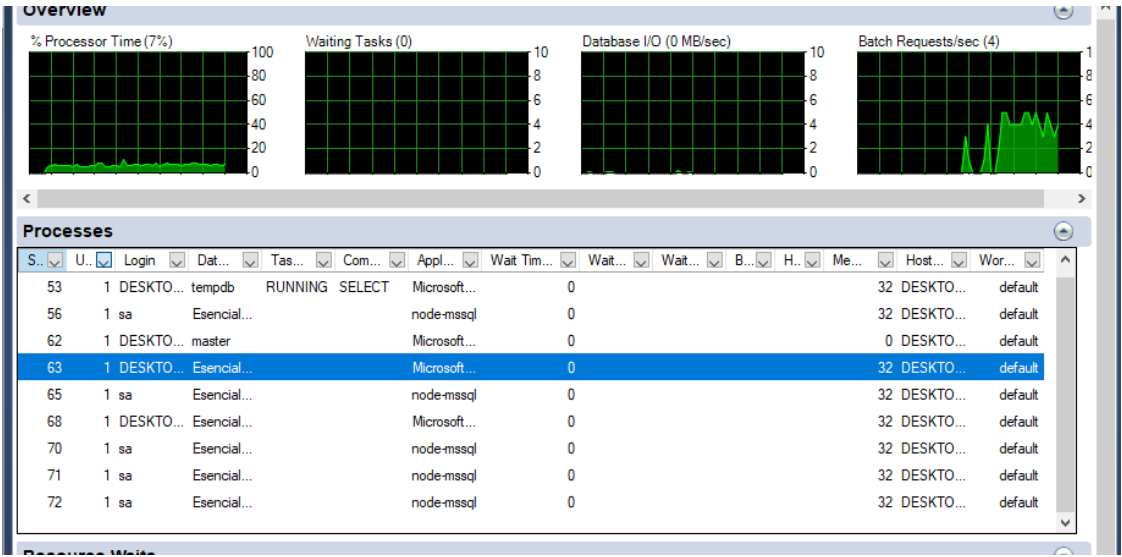
20 hilos de ejecución usando sin pooling



Sample #	Start Time	Thread Name	Label	Sample Time(...	Status	Bytes	Sent Bytes	Latency	Connect Time(...
1	16:46:00.546	Thread Group 1...	HTTP Request	828	✓	160362	224	827	5
2	16:46:01.046	Thread Group 1...	HTTP Request	343	✓	160362	224	342	1
3	16:46:01.542	Thread Group 1...	HTTP Request	174	✓	160362	224	173	1
4	16:46:02.041	Thread Group 1...	HTTP Request	240	✓	160362	224	240	1
5	16:46:02.543	Thread Group 1...	HTTP Request	181	✓	160362	224	181	2
6	16:46:03.043	Thread Group 1...	HTTP Request	160	✓	160362	224	159	1
7	16:46:03.544	Thread Group 1...	HTTP Request	165	✓	160362	224	165	2
8	16:46:04.043	Thread Group 1...	HTTP Request	138	✓	160362	224	137	1
9	16:46:04.543	Thread Group 1...	HTTP Request	130	✓	160362	224	129	1
10	16:46:05.043	Thread Group 1...	HTTP Request	297	✓	160362	224	296	1
11	16:46:05.542	Thread Group 1...	HTTP Request	151	✓	160362	224	150	1
12	16:46:06.051	Thread Group 1...	HTTP Request	165	✓	160362	224	165	1
13	16:46:06.541	Thread Group 1...	HTTP Request	156	✓	160362	224	155	1
14	16:46:07.042	Thread Group 1...	HTTP Request	113	✓	160362	224	112	1
15	16:46:07.540	Thread Group 1...	HTTP Request	171	✓	160362	224	170	1
16	16:46:08.042	Thread Group 1...	HTTP Request	108	✓	160362	224	107	1
17	16:46:08.541	Thread Group 1...	HTTP Request	114	✓	160362	224	113	1
18	16:46:09.041	Thread Group 1...	HTTP Request	113	✓	160362	224	112	1
19	16:46:09.542	Thread Group 1...	HTTP Request	296	✓	160362	224	295	1
20	16:46:10.042	Thread Group 1...	HTTP Request	163	✓	160362	224	162	3

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/...	Sent KB/sec	Avg. Bytes
Thread Group...	20	210	108	828	155.61	0.00%	2.1/sec	324.26	0.45	160362.0
TOTAL	20	210	108	828	155.61	0.00%	2.1/sec	324.26	0.45	160362.0

20 hilos de ejecución usando pooling



Sample #	Start Time	Thread Name	Label	Sample Time(...	Status	Bytes	Sent Bytes	Latency	Connect Time(...
1	16:43:35.397	Thread Group 1...	HTTP Request	2056	✓	320562	223	2038	366
2	16:43:35.690	Thread Group 1...	HTTP Request	2414	✓	320562	223	2411	73
3	16:43:36.009	Thread Group 1...	HTTP Request	2704	✓	320562	223	2702	2
4	16:43:36.508	Thread Group 1...	HTTP Request	2618	✓	320562	223	2614	2
5	16:43:37.516	Thread Group 1...	HTTP Request	2331	✓	320562	223	2329	3
6	16:43:38.584	Thread Group 1...	HTTP Request	1933	✓	320562	223	1931	12
7	16:43:37.012	Thread Group 1...	HTTP Request	4163	✓	320562	223	4160	2
8	16:43:38.050	Thread Group 1...	HTTP Request	3800	✓	320562	223	3799	15
9	16:43:39.031	Thread Group 1...	HTTP Request	3328	✓	320562	223	3324	9
10	16:43:39.529	Thread Group 1...	HTTP Request	3723	✓	320562	223	3722	6
11	16:43:40.636	Thread Group 1...	HTTP Request	3125	✓	320562	223	3124	2
12	16:43:41.025	Thread Group 1...	HTTP Request	3337	✓	320562	223	3335	5
13	16:43:40.059	Thread Group 1...	HTTP Request	4732	✓	320562	223	4730	1
14	16:43:41.527	Thread Group 1...	HTTP Request	3801	✓	320562	223	3799	12
15	16:43:42.012	Thread Group 1...	HTTP Request	3750	✓	320562	223	3748	5
16	16:43:42.517	Thread Group 1...	HTTP Request	3692	✓	320562	223	3690	4
17	16:43:43.006	Thread Group 1...	HTTP Request	3619	✓	320562	223	3617	2
18	16:43:44.010	Thread Group 1...	HTTP Request	2967	✓	320562	223	2965	2
19	16:43:43.520	Thread Group 1...	HTTP Request	3878	✓	320562	223	3876	4
20	16:43:44.508	Thread Group 1...	HTTP Request	3304	✓	320562	223	3303	2

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/...	Sent KB/sec	Avg. Bytes
Thread Group...	20	3263	1933	4732	719.21	0.00%	1.6/sec	504.31	0.35	320562.0
TOTAL	20	3263	1933	4732	719.21	0.00%	1.6/sec	504.31	0.35	320562.0

Utilizando ORM

S...	U...	Login	Dat...	Tas...	Com...	Appl...	Wait Tim...	Wait...	Wait...	B...	H...	Me...	Host...	Wor...
51	1	DESKTO...	master			Microsoft...	0						32	DESKTO...
52	1	DESKTO...	master			Microsoft...	0						0	DESKTO...
53	1	DESKTO...	tempdb	RUNNING	SELECT	Microsoft...	0						32	DESKTO...
54	1	NT SER...	master			SQLServ...	0						32	DESKTO...
57	1	DESKTO...	Esencial...			Microsoft...	0						32	DESKTO...
63	1	DESKTO...	Esencial...			Microsoft...	0						32	DESKTO...
69	1	sa	Esencial...			Python	0						32	DESKTO...

Sample #	Start Time	Thread Name	Label	Sample Time(...	Status	Bytes	Sent Bytes	Latency	Connect Time(...
1	18:12:02.481	Thread Group 1...	HTTP Request	397	✓	197736	208	395	12
2	18:12:02.983	Thread Group 1...	HTTP Request	63	✓	197736	208	61	6
3	18:12:03.481	Thread Group 1...	HTTP Request	75	✓	197736	208	72	10
4	18:12:03.979	Thread Group 1...	HTTP Request	63	✓	197736	208	61	7
5	18:12:04.478	Thread Group 1...	HTTP Request	71	✓	197736	208	69	6
6	18:12:04.980	Thread Group 1...	HTTP Request	57	✓	197736	208	55	4
7	18:12:05.481	Thread Group 1...	HTTP Request	77	✓	197736	208	75	5
8	18:12:05.978	Thread Group 1...	HTTP Request	80	✓	197736	208	79	11
9	18:12:06.481	Thread Group 1...	HTTP Request	59	✓	197736	208	58	5
10	18:12:06.981	Thread Group 1...	HTTP Request	70	✓	197736	208	69	7
11	18:12:07.480	Thread Group 1...	HTTP Request	60	✓	197736	208	58	5
12	18:12:07.980	Thread Group 1...	HTTP Request	82	✓	197736	208	79	7
13	18:12:08.483	Thread Group 1...	HTTP Request	93	✓	197736	208	92	7
14	18:12:08.980	Thread Group 1...	HTTP Request	98	✓	197736	208	97	7
15	18:12:09.479	Thread Group 1...	HTTP Request	91	✓	197736	208	90	6
16	18:12:09.979	Thread Group 1...	HTTP Request	112	✓	197736	208	111	4
17	18:12:10.485	Thread Group 1...	HTTP Request	112	✓	197736	208	111	5
18	18:12:10.981	Thread Group 1...	HTTP Request	96	✓	197736	208	95	5
19	18:12:11.482	Thread Group 1...	HTTP Request	102	✓	197736	208	101	6
20	18:12:11.983	Thread Group 1...	HTTP Request	86	✓	197736	208	85	5

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/...	Sent KB/sec	Avg. Bytes
HTTP Request	40	81	51	397	53.04	0.00%	50.6/min	162.83	0.17	197736.0
TOTAL	40	81	51	397	53.04	0.00%	50.6/min	162.83	0.17	197736.0

Al usar pool se tienen mejores prestaciones como fue mostrado anteriormente debido a que:

- Reutilizar conexiones previamente establecidas en lugar de abrir y cerrar conexiones individuales en cada solicitud, ya que se evita el tiempo de establecimiento de la conexión, que puede ser costoso en términos de rendimiento. Esto resulta en una menor latencia y tiempos de respuesta más rápidos para las operaciones que requieren acceso a recursos externos.
- Se puede reducir la carga en el servidor al evitar el proceso de establecimiento de nuevas conexiones en cada solicitud. El servidor solo necesita administrar un conjunto fijo de conexiones que se reutilizan, en lugar de administrar múltiples conexiones abiertas y cerradas continuamente. Esto puede llevar a una mejor

escalabilidad del sistema y un menor consumo de recursos del servidor, permitiendo manejar un mayor número de solicitudes simultáneas de manera más eficiente.