



Instituto Tecnológico de Costa Rica
Campus Tecnológico Central Cartago
Escuela De Ingeniería En Computación

Bases de datos I
Caso#3 - Preliminar#4
I-Semestre 2023
Lunes 15 mayo

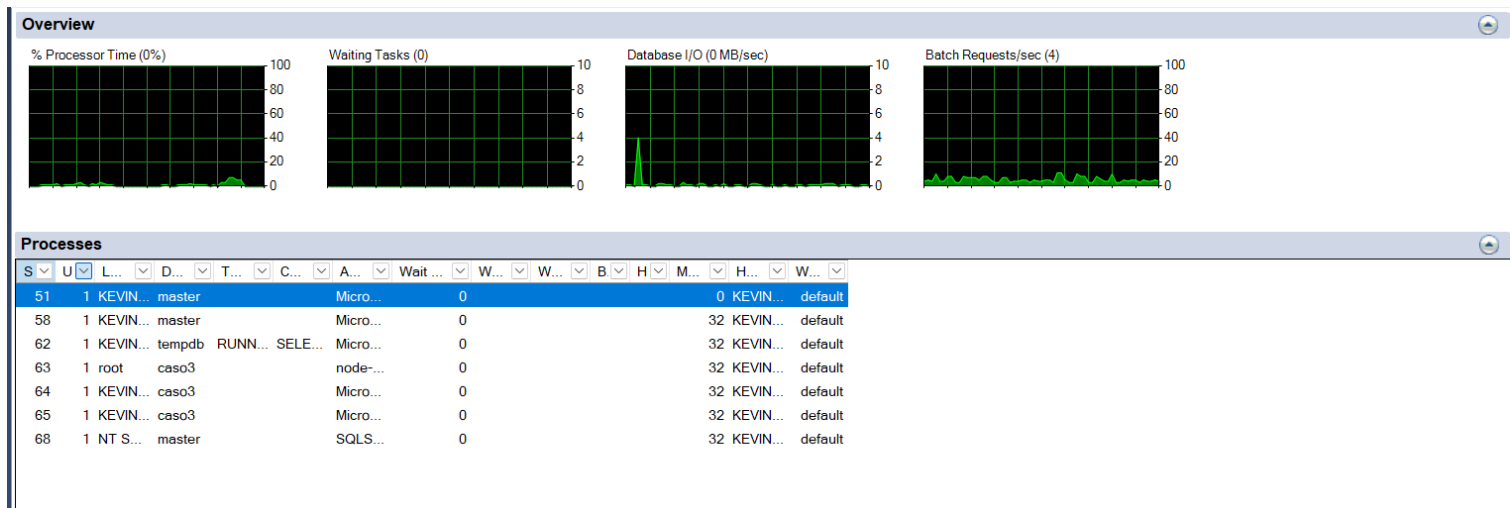
Integrantes:
Kevin Chang - 2022039050
Erika Cerdas - 2022138199

Documentación:

A continuación pondrán a prueba los servicios REST. Existen dos endpoints que llaman al stored procedure 'showContractData' uno que utiliza un pool de conexión y otro que no lo utiliza. Se van a hacer pruebas con 20 threads y un thread delay de 100ms. 2 pruebas por tipo.

Primero las pruebas con pool:

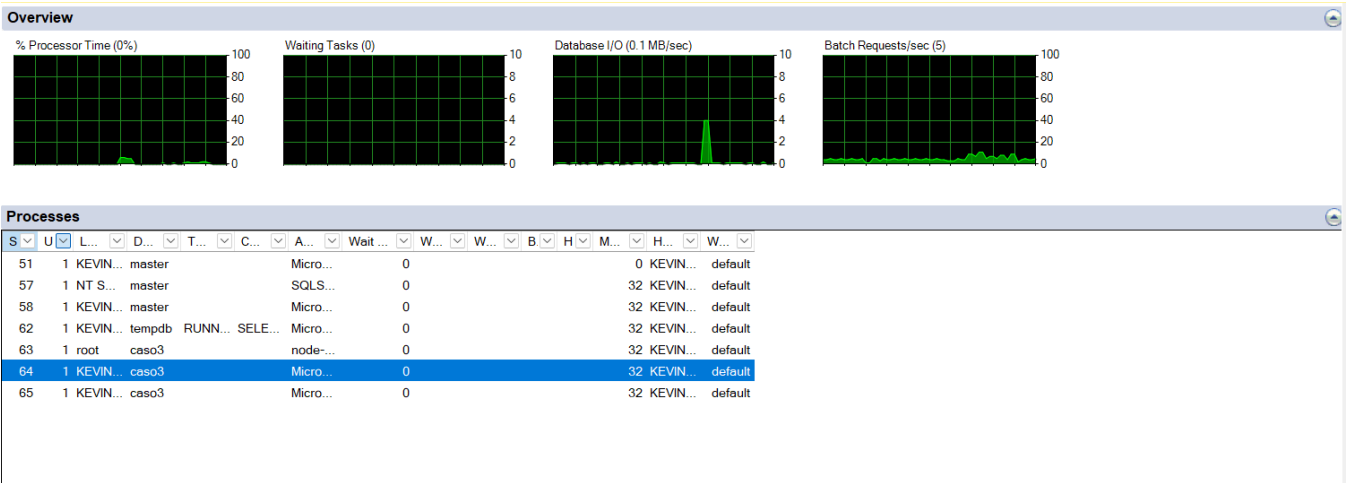
Entrada = 10. Activity Monitor, Summary Report y Results table



Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/...	Sent KB/sec	Avg. Bytes
Pool/No Pool	20	3817	540	8004	2170.35	0.00%	1.1/sec	2786.28	0.24	2499213.0
TOTAL	20	3817	540	8004	2170.35	0.00%	1.1/sec	2786.28	0.24	2499213.0

Sample #	Start Time	Thread Name	Label	Sample Time(...)	Status	Bytes	Sent Bytes	Latency	Connect Time(...)
1	15:41:28.778	Thread 1-1	Pool/No Pool	540	✓	2499213	219	534	2
2	15:41:29.292	Thread 1-2	Pool/No Pool	686	✓	2499213	219	677	2
3	15:41:29.789	Thread 1-3	Pool/No Pool	1018	✓	2499213	219	1011	3
4	15:41:30.280	Thread 1-4	Pool/No Pool	1417	✓	2499213	219	1410	2
5	15:41:30.795	Thread 1-5	Pool/No Pool	1787	✓	2499213	219	1774	2
6	15:41:31.278	Thread 1-6	Pool/No Pool	2227	✓	2499213	219	2218	2
7	15:41:31.781	Thread 1-7	Pool/No Pool	2509	✓	2499213	219	2496	1
8	15:41:32.280	Thread 1-8	Pool/No Pool	2895	✓	2499213	219	2885	2
9	15:41:32.787	Thread 1-9	Pool/No Pool	3147	✓	2499213	219	3136	3
10	15:41:33.279	Thread 1-10	Pool/No Pool	3514	✓	2499213	219	3504	3
11	15:41:33.782	Thread 1-11	Pool/No Pool	3851	✓	2499213	219	3841	3
12	15:41:34.280	Thread 1-12	Pool/No Pool	4295	✓	2499213	219	4282	3
13	15:41:34.792	Thread 1-13	Pool/No Pool	4702	✓	2499213	219	4690	2
14	15:41:35.295	Thread 1-14	Pool/No Pool	4919	✓	2499213	219	4909	2
15	15:41:35.791	Thread 1-15	Pool/No Pool	5317	✓	2499213	219	5306	3
16	15:41:36.280	Thread 1-16	Pool/No Pool	5586	✓	2499213	219	5575	2
17	15:41:36.783	Thread 1-17	Pool/No Pool	5951	✓	2499213	219	5940	2
18	15:41:37.279	Thread 1-18	Pool/No Pool	6508	✓	2499213	219	6492	2
19	15:41:37.789	Thread 1-19	Pool/No Pool	7476	✓	2499213	219	7459	1
20	15:41:38.293	Thread 1-20	Pool/No Pool	8004	✓	2499213	219	7991	2

Entrada = 12. Activity Monitor, Summary Report y Results Table



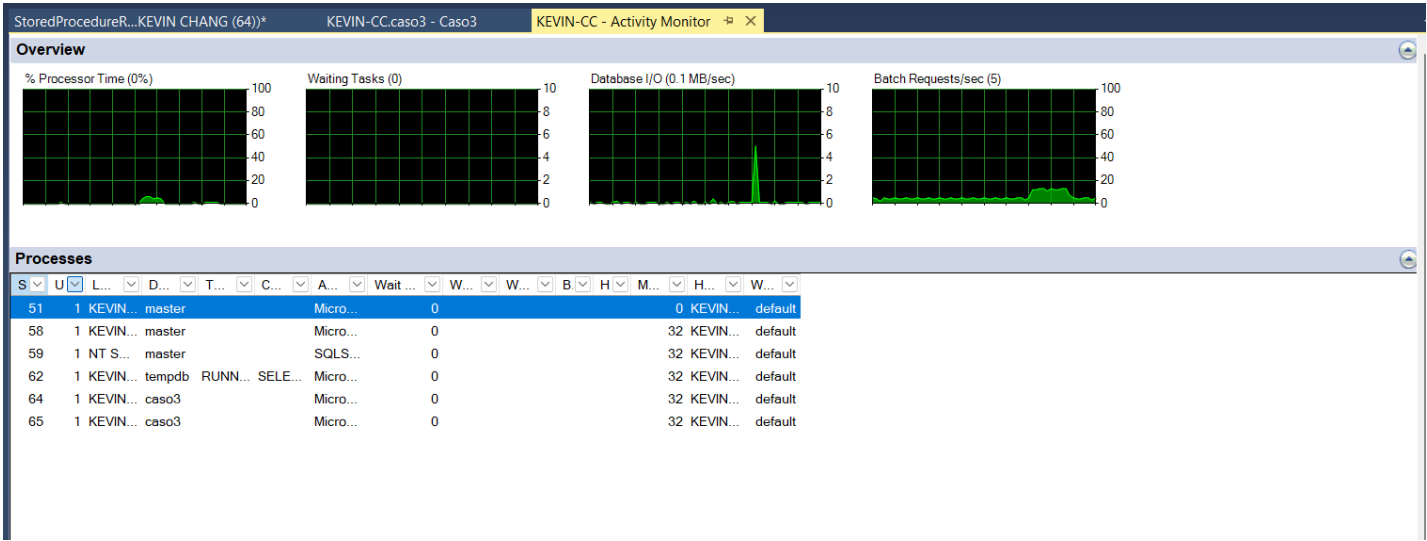
Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/...	Sent KB/sec	Avg. Bytes
Pool/No Pool	20	3289	485	6277	1856.82	0.00%	1.3/sec	3283.29	0.27	2652691.0
TOTAL	20	3289	485	6277	1856.82	0.00%	1.3/sec	3283.29	0.27	2652691.0

Sample #	Start Time	Thread Name	Label	Sample Time(...)	Status	Bytes	Sent Bytes	Latency	Connect Time(...)
1	15:45:37.011	Thread 1-1	Pool/No Pool	496	✓	2652691	219	490	3
2	15:45:37.511	Thread 1-2	Pool/No Pool	485	✓	2652691	219	480	2
3	15:45:38.017	Thread 1-3	Pool/No Pool	792	✓	2652691	219	785	2
4	15:45:38.512	Thread 1-4	Pool/No Pool	1130	✓	2652691	219	1121	3
5	15:45:39.025	Thread 1-5	Pool/No Pool	1464	✓	2652691	219	1457	3
6	15:45:39.513	Thread 1-6	Pool/No Pool	1793	✓	2652691	219	1785	2
7	15:45:40.022	Thread 1-7	Pool/No Pool	2121	✓	2652691	219	2109	2
8	15:45:40.513	Thread 1-8	Pool/No Pool	2482	✓	2652691	219	2473	2
9	15:45:41.018	Thread 1-9	Pool/No Pool	2812	✓	2652691	219	2803	2
10	15:45:41.514	Thread 1-10	Pool/No Pool	3092	✓	2652691	219	3083	2
11	15:45:42.012	Thread 1-11	Pool/No Pool	3443	✓	2652691	219	3432	2
12	15:45:42.513	Thread 1-12	Pool/No Pool	3823	✓	2652691	219	3814	2
13	15:45:43.024	Thread 1-13	Pool/No Pool	4171	✓	2652691	219	4163	2
14	15:45:43.512	Thread 1-14	Pool/No Pool	4425	✓	2652691	219	4415	2
15	15:45:44.022	Thread 1-15	Pool/No Pool	4785	✓	2652691	219	4777	2
16	15:45:44.511	Thread 1-16	Pool/No Pool	5035	✓	2652691	219	5027	3
17	15:45:45.015	Thread 1-17	Pool/No Pool	5433	✓	2652691	219	5425	4
18	15:45:45.534	Thread 1-18	Pool/No Pool	5695	✓	2652691	219	5686	2
19	15:45:46.024	Thread 1-19	Pool/No Pool	6040	✓	2652691	219	6011	2
20	15:45:46.514	Thread 1-20	Pool/No Pool	6277	✓	2652691	219	6268	2

☐ Scroll automatically?☐ Child samples?No of Samples 20Latest Sample 6277Average 3289Deviation 1856

Segundo las pruebas sin Pool:

Entrada = 10. Activity Monitor, Summary Report y Results Table

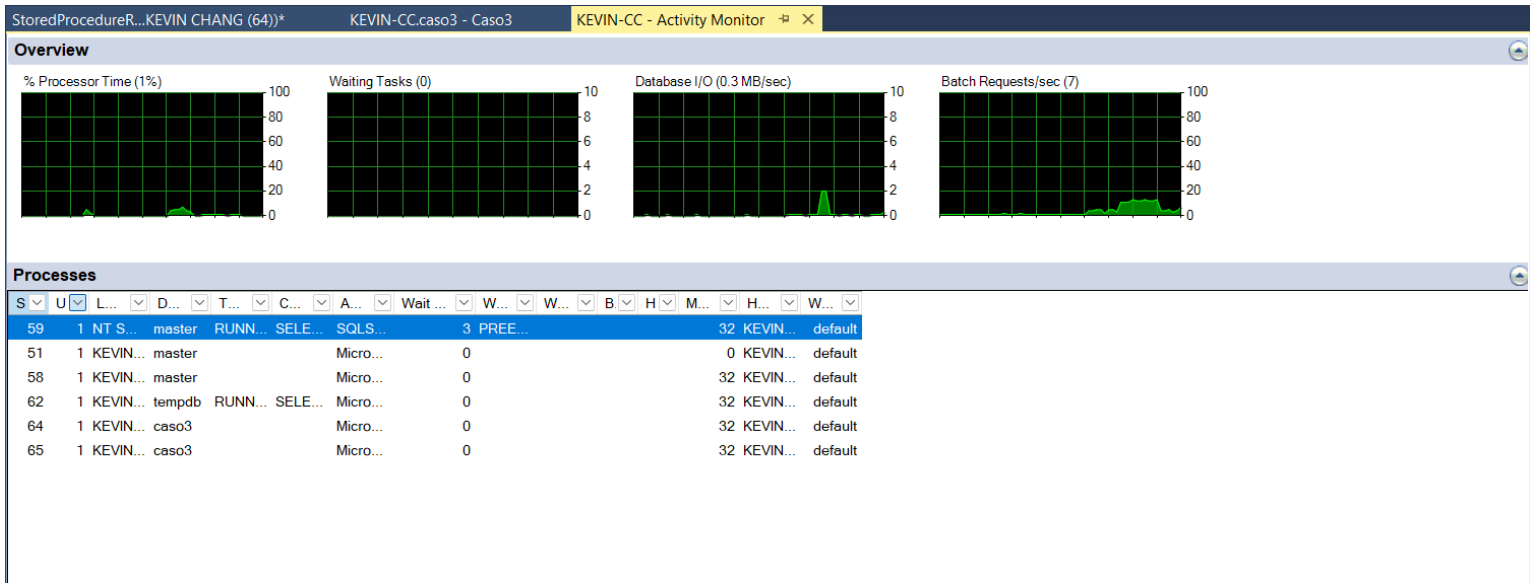


Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/...	Sent KB/sec	Avg. Bytes
Pool/No Pool	20	256	221	306	22.40	0.00%	2.1/sec	5007.46	0.44	2499213.0
TOTAL	20	256	221	306	22.40	0.00%	2.1/sec	5007.46	0.44	2499213.0

Sample #	Start Time	Thread Name	Label	Sample Time(...	Status	Bytes	Sent Bytes	Latency	Connect Time(...
1	15:51:08.584	Thread 1-1	Pool/No Pool	306	✔	2499213	221	259	2
2	15:51:09.097	Thread 1-2	Pool/No Pool	231	✔	2499213	221	227	1
3	15:51:09.588	Thread 1-3	Pool/No Pool	230	✔	2499213	221	207	1
4	15:51:10.083	Thread 1-4	Pool/No Pool	247	✔	2499213	221	243	4
5	15:51:10.594	Thread 1-5	Pool/No Pool	268	✔	2499213	221	251	1
6	15:51:11.081	Thread 1-6	Pool/No Pool	282	✔	2499213	221	277	2
7	15:51:11.592	Thread 1-7	Pool/No Pool	274	✔	2499213	221	269	1
8	15:51:12.080	Thread 1-8	Pool/No Pool	280	✔	2499213	221	251	1
9	15:51:12.591	Thread 1-9	Pool/No Pool	272	✔	2499213	221	268	3
10	15:51:13.080	Thread 1-10	Pool/No Pool	277	✔	2499213	221	255	3
11	15:51:13.590	Thread 1-11	Pool/No Pool	254	✔	2499213	221	226	4
12	15:51:14.082	Thread 1-12	Pool/No Pool	276	✔	2499213	221	250	3
13	15:51:14.587	Thread 1-13	Pool/No Pool	268	✔	2499213	221	241	3
14	15:51:15.095	Thread 1-14	Pool/No Pool	221	✔	2499213	221	217	4
15	15:51:15.578	Thread 1-15	Pool/No Pool	234	✔	2499213	221	231	2
16	15:51:16.079	Thread 1-16	Pool/No Pool	265	✔	2499213	221	241	1
17	15:51:16.590	Thread 1-17	Pool/No Pool	242	✔	2499213	221	221	1
18	15:51:17.083	Thread 1-18	Pool/No Pool	244	✔	2499213	221	224	1
19	15:51:17.590	Thread 1-19	Pool/No Pool	228	✔	2499213	221	224	2
20	15:51:18.093	Thread 1-20	Pool/No Pool	239	✔	2499213	221	235	4

☐ Scroll automatically?☐ Child samples?No of Samples 20Latest Sample 239Average 256Deviation 22

Entrada = 12. Activity Monitor, Summary Report y Results Table



Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/...	Sent KB/sec	Avg. Bytes
Pool/No Pool	20	309	230	551	89.84	0.00%	2.1/sec	5319.89	0.44	2652691.0
TOTAL	20	309	230	551	89.84	0.00%	2.1/sec	5319.89	0.44	2652691.0

Sample #	Start Time	Thread Name	Label	Sample Time(...)	Status	Bytes	Sent Bytes	Latency	Connect Time(...)
1	15:48:53.826	Thread 1-1	Pool/No Pool	373	✓	2652691	221	303	2
2	15:48:54.328	Thread 1-2	Pool/No Pool	551	✓	2652691	221	543	1
3	15:48:54.826	Thread 1-3	Pool/No Pool	528	✓	2652691	221	508	2
4	15:48:55.325	Thread 1-4	Pool/No Pool	387	✓	2652691	221	370	1
5	15:48:55.833	Thread 1-5	Pool/No Pool	380	✓	2652691	221	360	2
6	15:48:56.323	Thread 1-6	Pool/No Pool	315	✓	2652691	221	310	3
7	15:48:56.836	Thread 1-7	Pool/No Pool	318	✓	2652691	221	297	3
8	15:48:57.325	Thread 1-8	Pool/No Pool	280	✓	2652691	221	258	3
9	15:48:57.837	Thread 1-9	Pool/No Pool	275	✓	2652691	221	271	1
10	15:48:58.327	Thread 1-10	Pool/No Pool	236	✓	2652691	221	230	3
11	15:48:58.826	Thread 1-11	Pool/No Pool	265	✓	2652691	221	261	2
12	15:48:59.327	Thread 1-12	Pool/No Pool	268	✓	2652691	221	241	3
13	15:48:59.827	Thread 1-13	Pool/No Pool	265	✓	2652691	221	241	4
14	15:49:00.339	Thread 1-14	Pool/No Pool	238	✓	2652691	221	218	3
15	15:49:00.826	Thread 1-15	Pool/No Pool	250	✓	2652691	221	246	1
16	15:49:01.345	Thread 1-16	Pool/No Pool	238	✓	2652691	221	226	2
17	15:49:01.837	Thread 1-17	Pool/No Pool	281	✓	2652691	221	276	1
18	15:49:02.328	Thread 1-18	Pool/No Pool	258	✓	2652691	221	237	3
19	15:49:02.835	Thread 1-19	Pool/No Pool	250	✓	2652691	221	246	3
20	15:49:03.335	Thread 1-20	Pool/No Pool	230	✓	2652691	221	225	2

Una conexión con pool reutiliza las conexiones existentes en lugar de crear una nueva para cada solicitud a diferencia de una que no utiliza el pool. Reutilizar conexiones puede mejorar significativamente el rendimiento y reduce la carga en la base de datos. Esto es lo que normalmente ocurre en entornos con alta concurrencia. [1]

En este caso específico, la conexión que no utiliza pool tiene un menor sample time y mayor throughput. Así como mayor kb/s received y sent. Viendo el ejemplo con entrada 10 con y sin pool, podemos formar la siguiente tabla.

Las muestras del summary report y el report table favorecen significativamente al no uso de pool.

	Average Sample time (ms)	Throughput	Received kb/s	Sent kb/s
No Pool	256	2.1/sec	5007.46	0.44
With Pool	3817	1.1/sec	2786.28	0.24

Los activity monitors de Sql Server muestra poca actividad donde lo más notable es el pico en el database I/O y el pequeño incremento de Batch Request/sec cuando se corren los hilos. Pero en general la diferencia es pequeña entre el pool y el no pool favoreciendo al no pool.

El endpoint que no utiliza pool parece tener mejor presentación, esto se puede dar por varios factores como lo son los siguientes:

- Ya que tiene un bajo volumen de conexiones y no realiza muchas consultas a la base de datos, un pool de conexiones no es tan necesario.
- El pool de conexiones puede que aumente la complejidad de la consulta ya que sola, esta consulta es muy simple. En este caso se hace más sencillo abrir y cerrar conexiones cada vez que se necesite

Lo más posible es que el no tener un pool de conexiones es mejor en este caso, ya que el programa y la consulta es muy sencilla.

[1] R. Donovan, "Improve database performance with connection pooling," Stack Overflow Blog, <https://stackoverflow.blog/2020/10/14/improve-database-performance-with-connection-pooling/>. Encontrado: (Mayo 14, 2023).

Incrementar el max y el min del pool de conexión incrementa el average sample time de la que utiliza pool de conexiones.

La siguiente imagen muestra el summary report de esto. (con max=5 y min=5).

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
Pool/No Pool	20	7204	1616	11803	3004.39	0.00%	56.4/min	2434.81	0.20	2652691.0
TOTAL	20	7204	1616	11803	3004.39	0.00%	56.4/min	2434.81	0.20	2652691.0

Lo más posible es que esto se deba a que la aplicación es muy pequeña y tener un pool de conexiones no es mejor que crear nuevas conexiones cada vez.