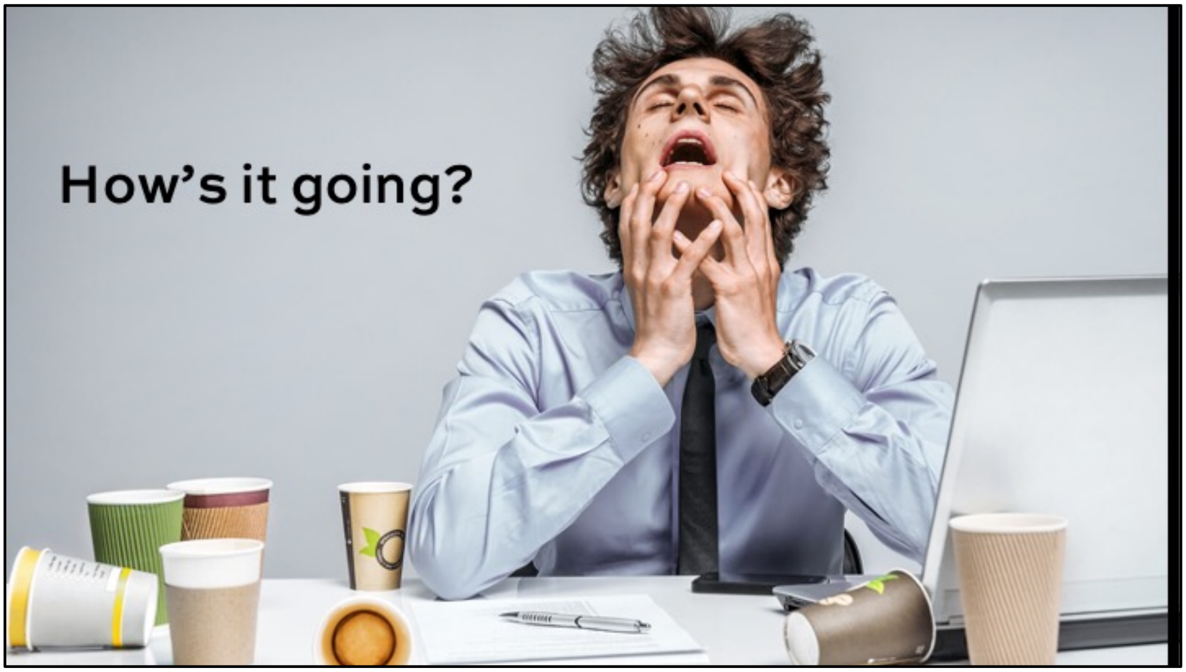
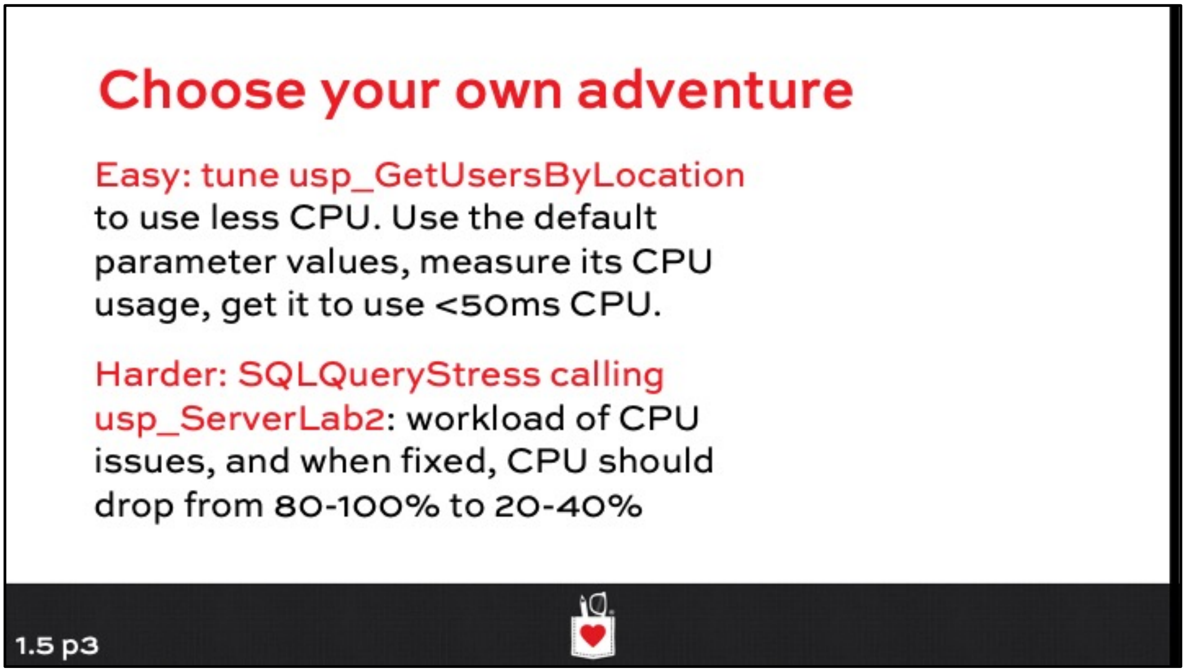
BRENT OZAR UNLIMITED®

Lab 2: CPU-Intensive Workload

1.5 p1



How’s it going?

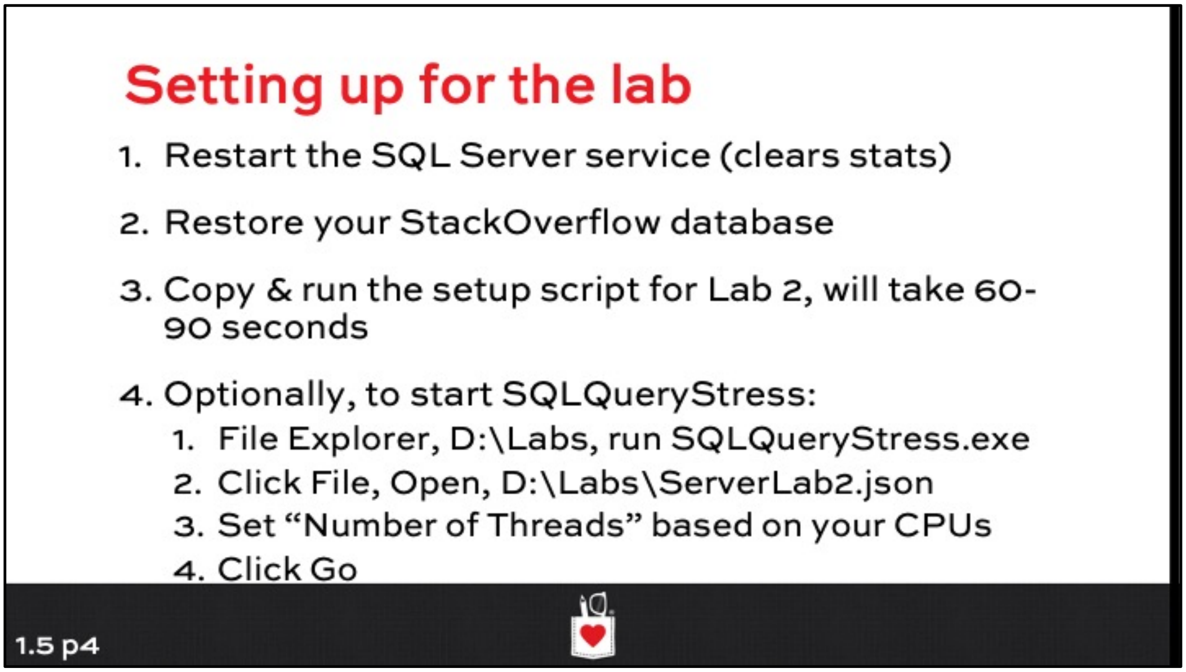


Choose your own adventure

Easy: tune usp_GetUsersByLocation to use less CPU. Use the default parameter values, measure its CPU usage, get it to use <50ms CPU.

Harder: SQLQueryStress calling usp_ServerLab2: workload of CPU issues, and when fixed, CPU should drop from 80-100% to 20-40%

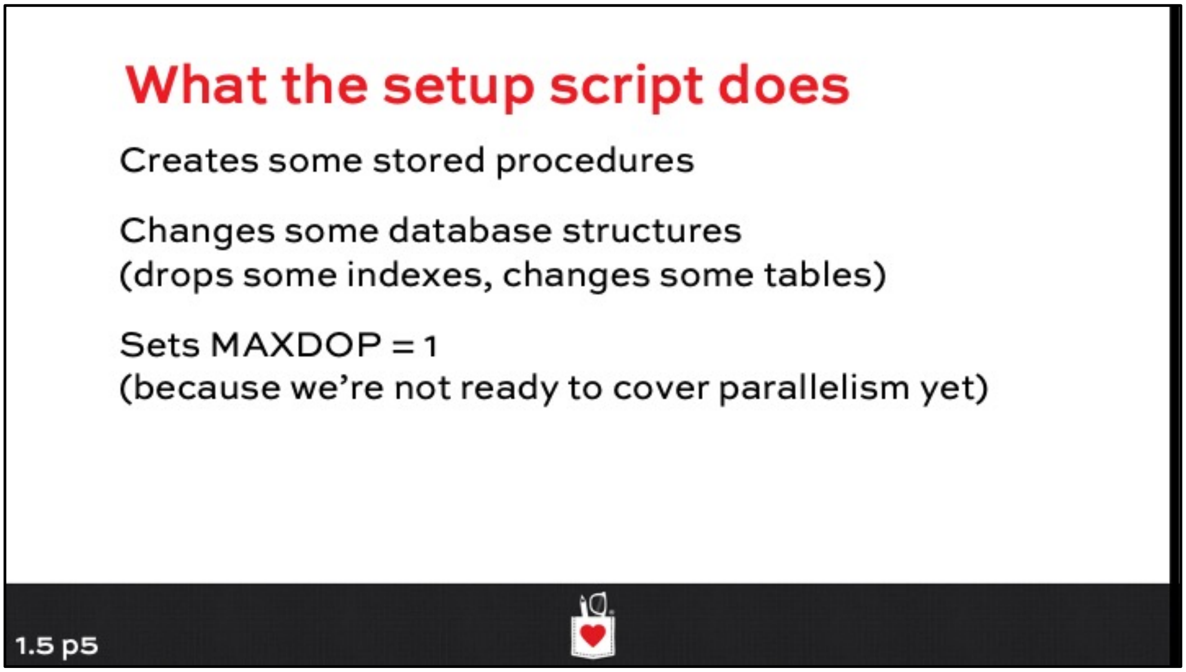
1.5 p3



Setting up for the lab

1. Restart the SQL Server service (clears stats)
2. Restore your StackOverflow database
3. Copy & run the setup script for Lab 2, will take 60-90 seconds
4. Optionally, to start SQLQueryStress:
   1. File Explorer, D:\Labs, run SQLQueryStress.exe
   2. Click File, Open, D:\Labs\ServerLab2.json
   3. Set "Number of Threads" based on your CPUs
   4. Click Go

1.5 p4



What the setup script does

Creates some stored procedures

Changes some database structures
(drops some indexes, changes some tables)

Sets MAXDOP = 1
(because we're not ready to cover parallelism yet)

1.5 p5

Doing the stress test? I'm your manager.

Just like real life…

Our end users are yelling at us to make it go faster.

I’m doing the best I can to fend them off.

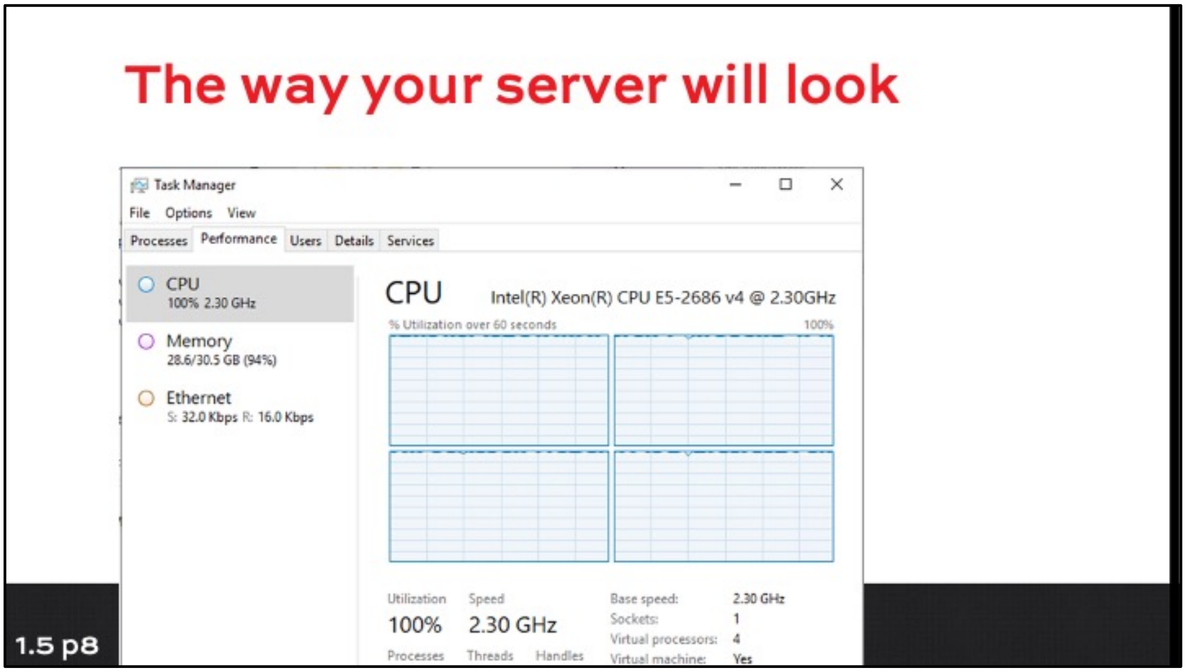
They don’t wanna make changes to production.

When you wanna change something:

- Explain what you’re going to change, and why
- Explain the difference you expect to see

You don’t have to wait for my approval, though.

1.5 p7



The way your server will look

Task Manager
File Options View
Processes Performance Users Details Services
% Utilization over 60 seconds
100%

CPU
100% 2.30 GHz

Memory
28.6/30.5 GB (94%)

Ethernet
S: 32.0 Kbps R: 16.0 Kbps

CPU Intel(R) Xeon(R) CPU E5-2686 v4 @ 2.30GHz

Utilization 100%

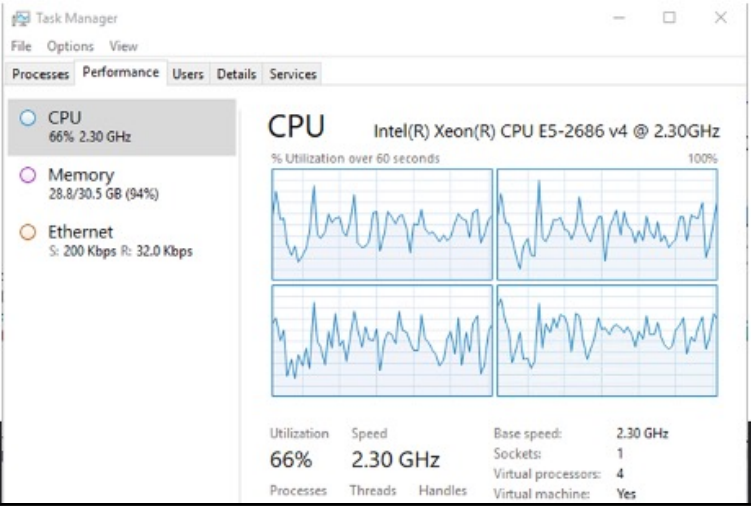
Speed 2.30 GHz

Processes Threads Handles

Base speed: 2.30 GHz
Sockets: 1
Virtual processors: 4
Virtual machine: Yes

1.5 p8

But if you do an awesome job…


Task Manager
File Options View
Processes Performance Users Details Services
% Utilization over 60 seconds
100%

CPU
66% 2.30 GHz

Memory
28.8/30.5 GB (94%)

Ethernet
S: 200 Kbps R: 32.0 Kbps

CPU Intel(R) Xeon(R) CPU E5-2686 v4 @ 2.30GHz

Utilization 66%

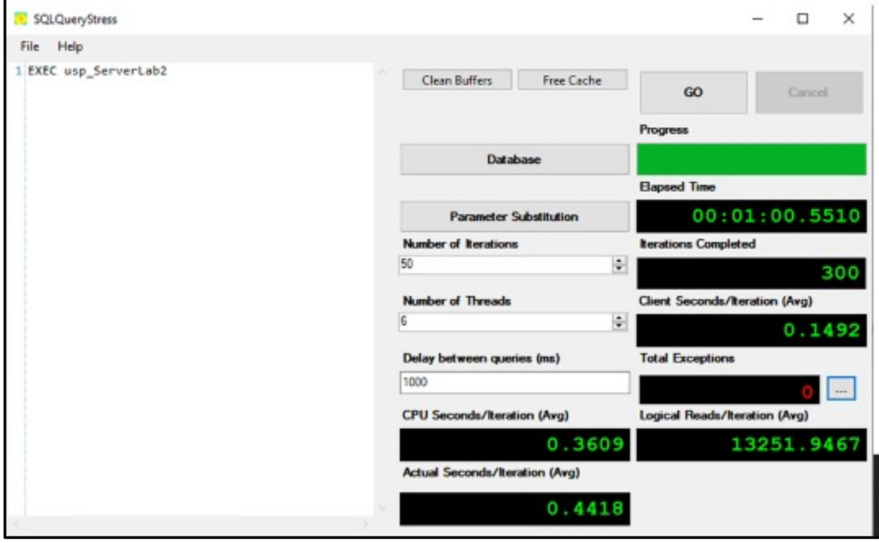
Speed 2.30 GHz

Processes Threads Handles

Base speed: 2.30 GHz
Sockets: 1
Virtual processors: 4
Virtual machine: Yes

1.5 p9

The load will finish in ~60 secs.



```
EXEC usp_ServerLab2
```

Number of Iterations: 50

Number of Threads: 6

Delay between queries (ms): 1000

CPU Seconds/Iteration (Avg): 0.3609

Actual Seconds/Iteration (Avg): 0.4418

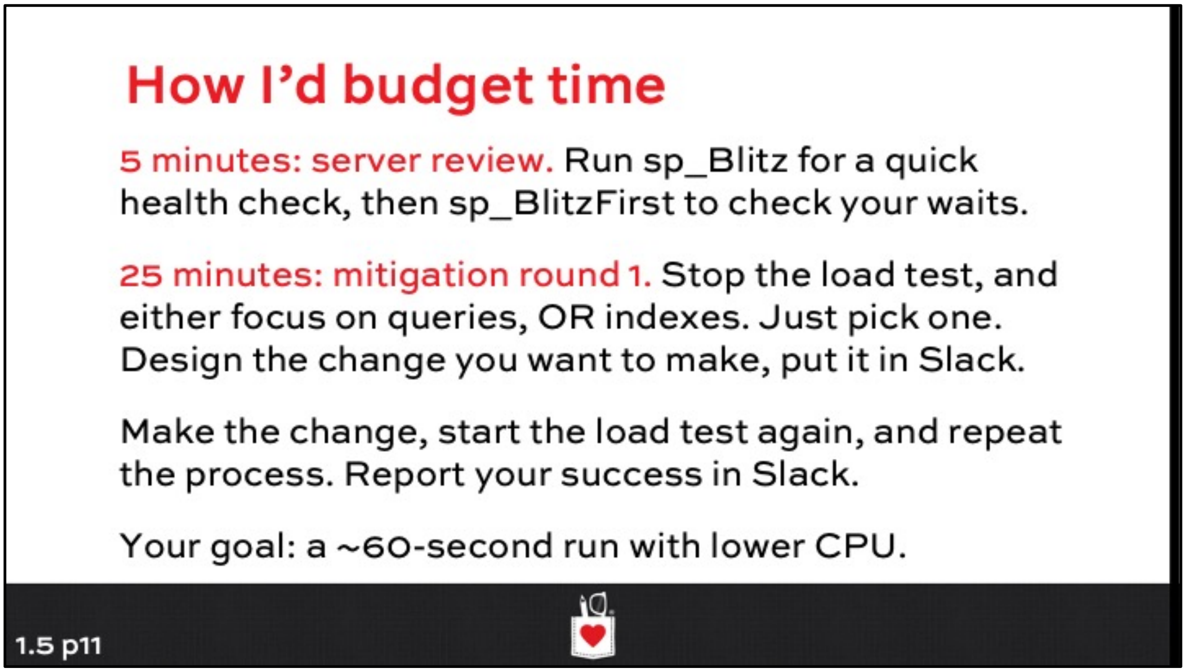
Elapsed Time: 00:01:00.5510

Iterations Completed: 300

Client Seconds/Iteration (Avg): 0.1492

Total Exceptions: 0

Logical Reads/Iteration (Avg): 13251.9467



How I'd budget time

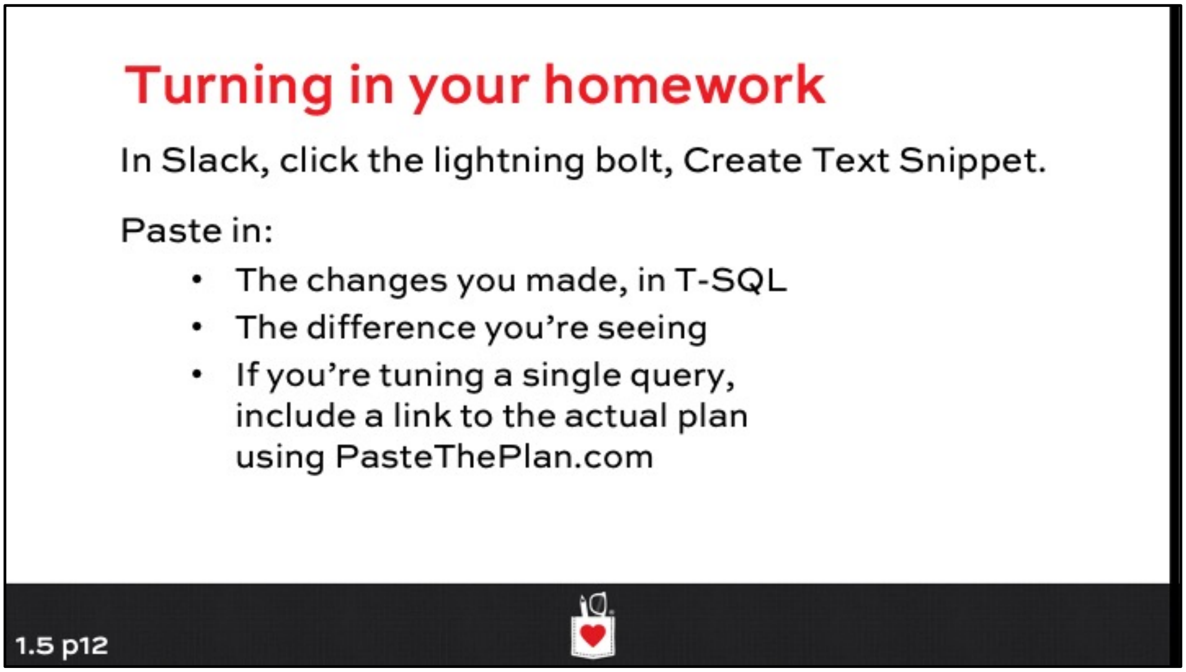
5 minutes: server review. Run sp_Blitz for a quick health check, then sp_BlitzFirst to check your waits.

25 minutes: mitigation round 1. Stop the load test, and either focus on queries, OR indexes. Just pick one. Design the change you want to make, put it in Slack.

Make the change, start the load test again, and repeat the process. Report your success in Slack.

Your goal: a ~60-second run with lower CPU.

1.5 p11



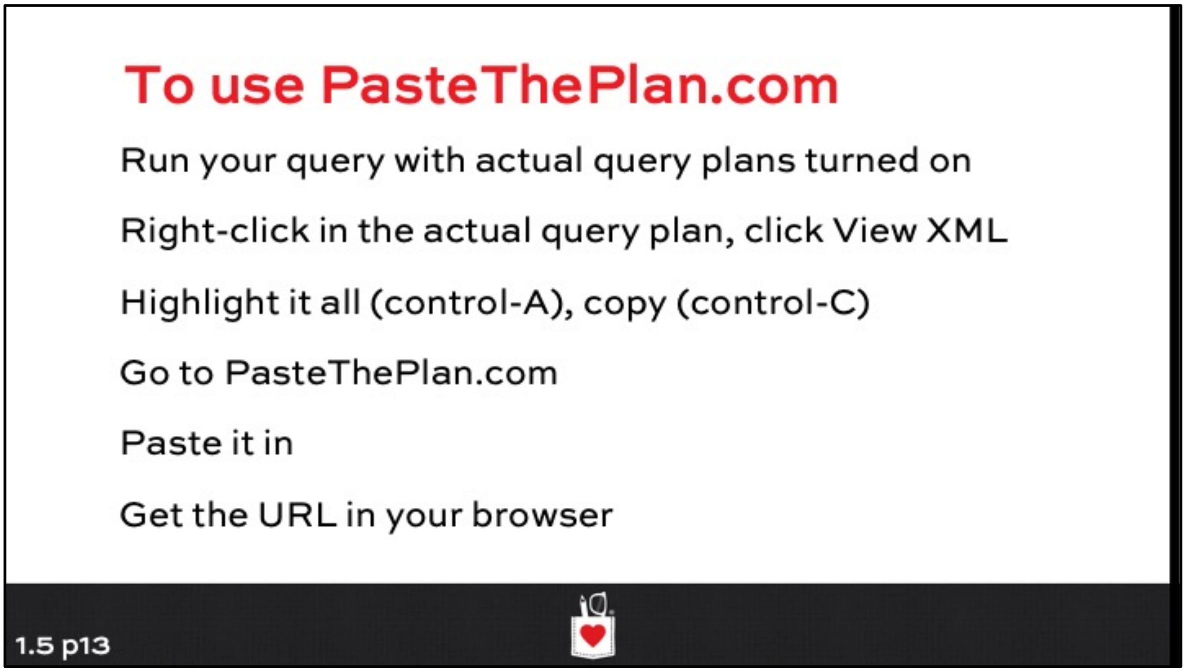
Turning in your homework

In Slack, click the lightning bolt, Create Text Snippet.

Paste in:

- The changes you made, in T-SQL
- The difference you're seeing
- If you're tuning a single query, include a link to the actual plan using PasteThePlan.com

1.5 p12



To use PasteThePlan.com

Run your query with actual query plans turned on

Right-click in the actual query plan, click View XML

Highlight it all (control-A), copy (control-C)

Go to PasteThePlan.com

Paste it in

Get the URL in your browser

1.5 p13