



Performance tuning Stack Overflow tags

Marco Cecconi



@sklivvz

<http://sklivvz.com>

Scale







- **1-100**

You & your friends

- **100-10,000**

Early Product

- **10,000-100,000,000**

<A TON OF PROBLEMS>

- **100,000,000-Infinity**

World domination? I can't even

How to Even

FOR
DUMMIES®

Learn to:

- Even
- Literally Even



An underwater scene with blue water, light rays filtering down from the surface, and bubbles rising. The text "DEEP DIVE" is centered in the middle of the image.

DEEP DIVE

Adding a detail text labels to an existing array of strings, UITableView



I have this array in my UITableViewController.m

0



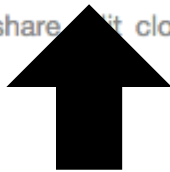
```
self.movies = [NSArray arrayWithObjects:  
    @"Zardoz",  
    @"Dark City",  
    @"The Empire",  
    @"The Godfather Part II",  
    @"The Godfather Part I",  
    @"The Godfather",  
    @"The Godfather: Part III",  
    @"The Godfather: The Cursed",  
    @"The Godfather: The Godfather",  
    nil];
```

I would like to initWithStyle:UITableViewCellStyleValue2. I am trying to add a detailTextLabel to each corresponding to each movies IDBM rating(double). Do I create another array for the values? Thanks.

objective-c

uitableview

mod share edit close delete flag



edited Jun 2 at 22:24



rmaddy

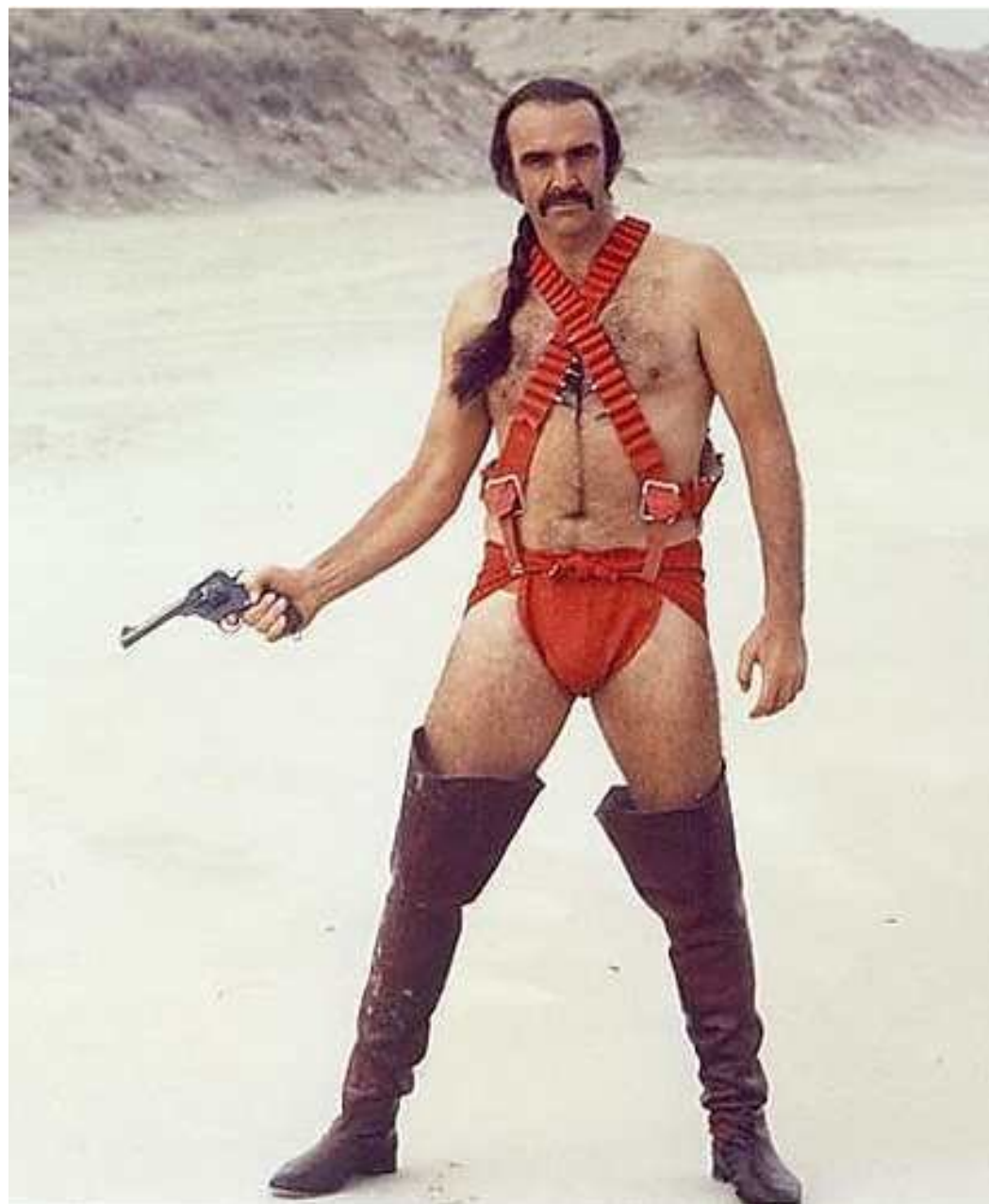
116k ● 9 ● 132 ● 182

asked Jun 2 at 22:20



Michael Castro

27 ● 4



sql-server 

javascript  and not jquery 

windows*

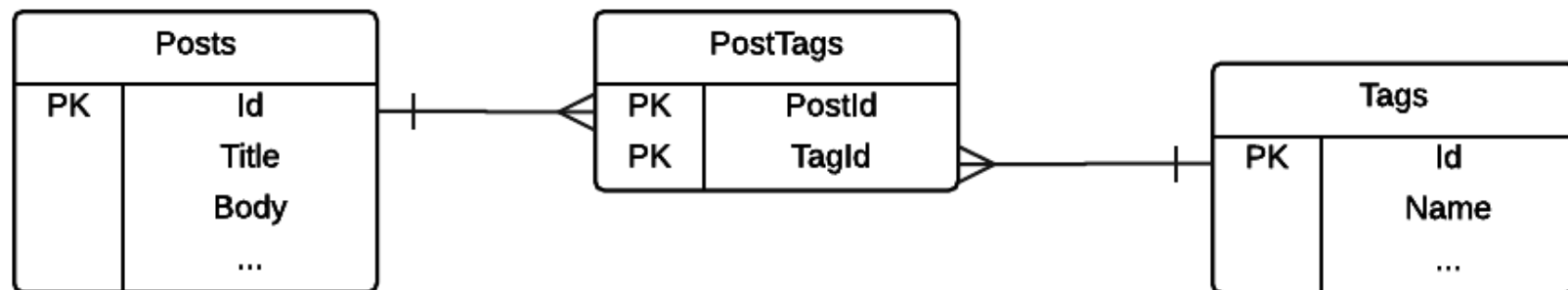
windows or windows-phone-7 or windows-phone-8 or windows-8 or windows-7 or windows-runtime or
windows-phone-8.1 or windows-services or windows-phone or windows-store-apps or windows-installer or
windows-10 or windows-8.1 or windows-xp or windows-mobile or windows-azure-storage or windows-server-2008 or
windows-ce or windows-authentication or windows-phone-7.1

Our Mission

- Custom sorts
- Get page N of matches
- Count the matches
- Find related tags







Find all questions tagged **"javascript"** but not
"jquery"

```

SET STATISTICS IO ON
SET STATISTICS TIME ON
GO

SELECT      TOP 50 Tag1.PostId, LastActivityDate
FROM        (
    SELECT p.Id PostId, LastActivityDate
    FROM    Posts p
    JOIN    PostTags pt ON p.Id = pt.PostId
    WHERE   pt.TagId = 3
) Tag1
LEFT JOIN (
    SELECT p.Id PostId
    FROM    Posts p
    JOIN    PostTags pt ON p.Id = pt.PostId
    WHERE   pt.TagId = 820
) Tag2 ON Tag1.PostId = Tag2.PostId
WHERE      Tag2.PostId IS NULL
ORDER BY   LastActivityDate DESC

SET STATISTICS TIME OFF
SET STATISTICS IO OFF
GO

```

<http://data.stackexchange.com/stackoverflow/query/589988>

Table 'PostTags'. Scan count 5, logical reads 6344, [...]

Table 'Posts'. Scan count 5, logical reads 302062, [...]

SQL Server Execution Times: CPU time = 5173 ms, elapsed time = 2194 ms.

Related Tags:

Find the top 10 tags in those questions

```

SET STATISTICS IO ON
SET STATISTICS TIME ON
GO

SELECT      TOP 10 PostTags.TagId, COUNT(1) as Number
FROM        (
    SELECT p.Id PostId, LastActivityDate
    FROM Posts p
    JOIN PostTags pt ON p.Id = pt.PostId
    WHERE pt.TagId = 3
) Tag1
LEFT JOIN (
    SELECT p.Id PostId
    FROM Posts p
    JOIN PostTags pt ON p.Id = pt.PostId
    WHERE pt.TagId = 820
) Tag2 ON Tag1.PostId = Tag2.PostId
JOIN        PostTags ON PostTags.PostId = Tag1.PostId
WHERE       Tag2.PostId IS NULL
AND         PostTags.TagId NOT IN (3, 820)
GROUP BY    PostTags.TagId
ORDER BY    2 DESC

SET STATISTICS TIME OFF
SET STATISTICS IO OFF
GO

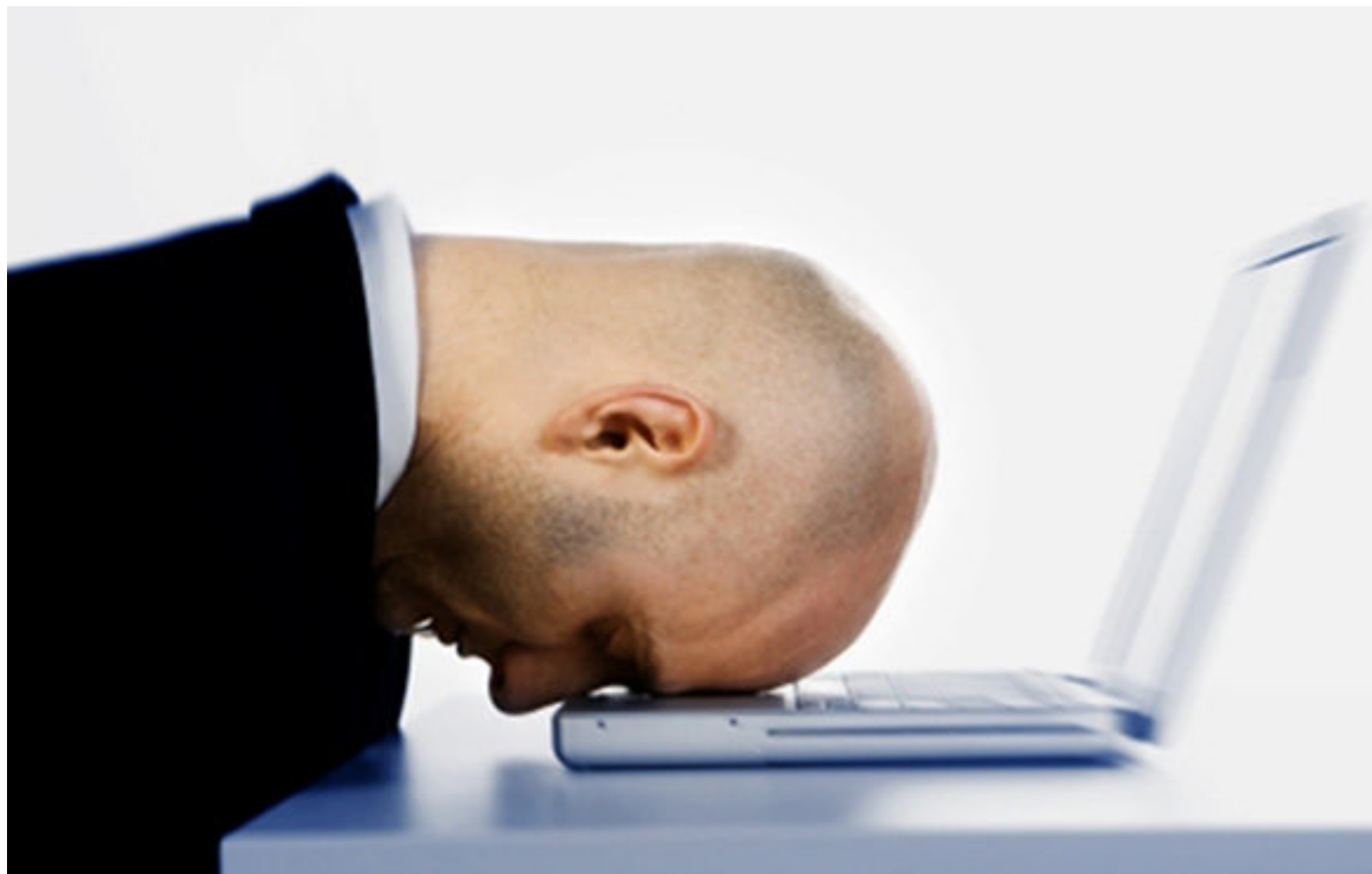
```

<http://data.stackexchange.com/stackoverflow/query/589992>

Table 'PostTags'. Scan count 15, logical reads 169532, [...]
read-ahead reads 122731 [...]
Table 'Posts'. Scan count 10, logical reads 329038, [...]

SQL Server Execution Times:

CPU time = 18875 ms, elapsed time = 5399 ms.





from facebook

THE EXPERT'S VOICE IN SQL SERVER

Pro Full-Text Search in SQL Server 2008

Take advantage of the newly new and improved
full-text search features in SQL Server 2008



Michael Coles
with Hilary Carter

Apress®

vbcaaz

ISBN 978-1-449-30551-9
9 781449305519

```
SET STATISTICS IO ON
SET STATISTICS TIME ON
GO
```

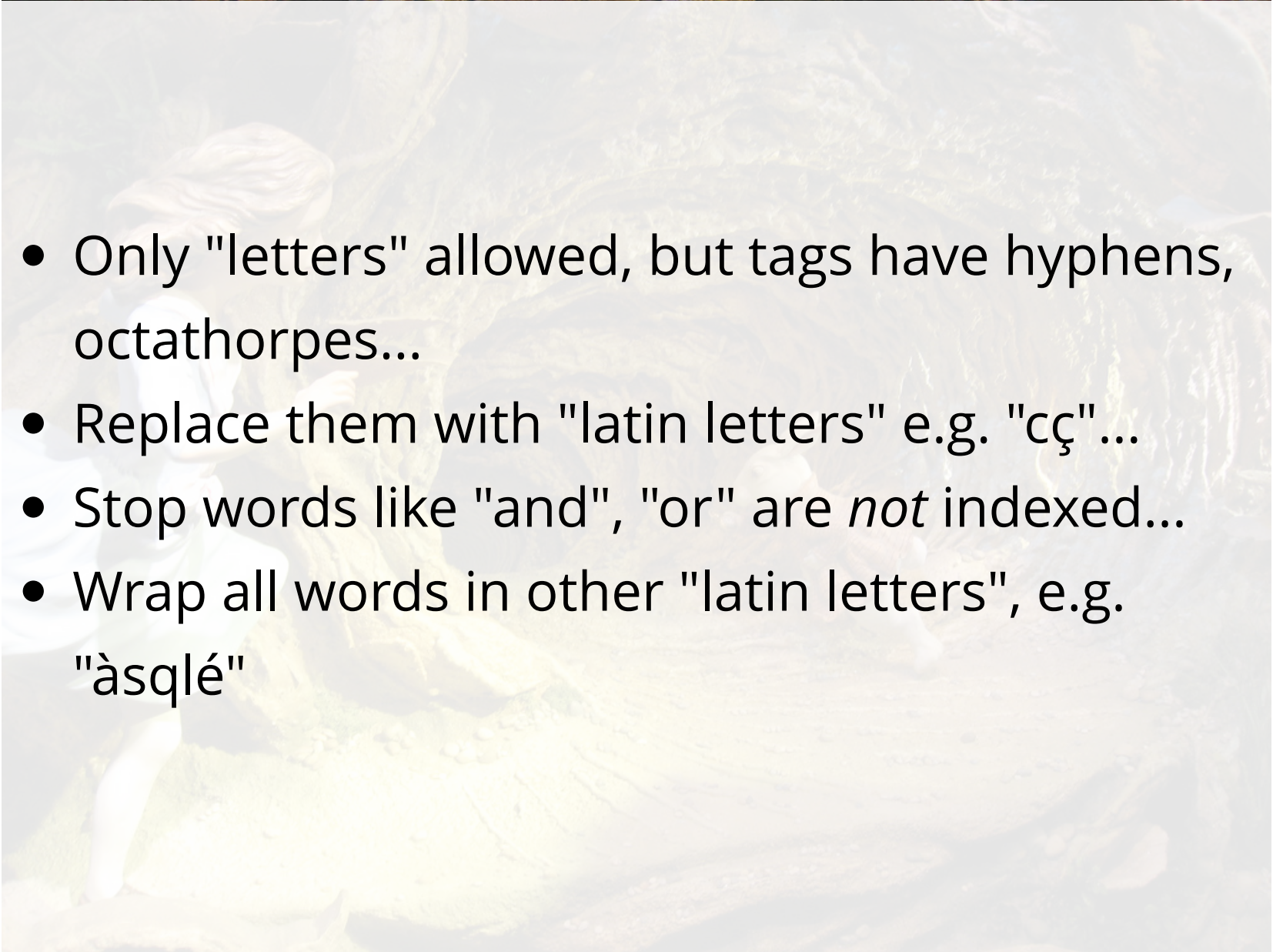
```
SELECT TOP 100 Id, LastActivityDate
FROM Posts
WHERE PostTypeId = 1
AND CONTAINS(Tags, 'javascript' AND NOT 'jquery')
ORDER BY 2 DESC
```

```
SET STATISTICS IO OFF
SET STATISTICS TIME OFF
GO
```

Table 'Posts'. Scan count 5, logical reads 179483, [...]

SQL Server Execution Times:

CPU time = 5202 ms, elapsed time = 2044 ms.

- 
- Only "letters" allowed, but tags have hyphens, octathorpes...
 - Replace them with "latin letters" e.g. "cç"...
 - Stop words like "and", "or" are *not* indexed...
 - Wrap all words in other "latin letters", e.g. "às qlé"

CONTAINS (tags, 'éûnetà AND écñà')

INDEX



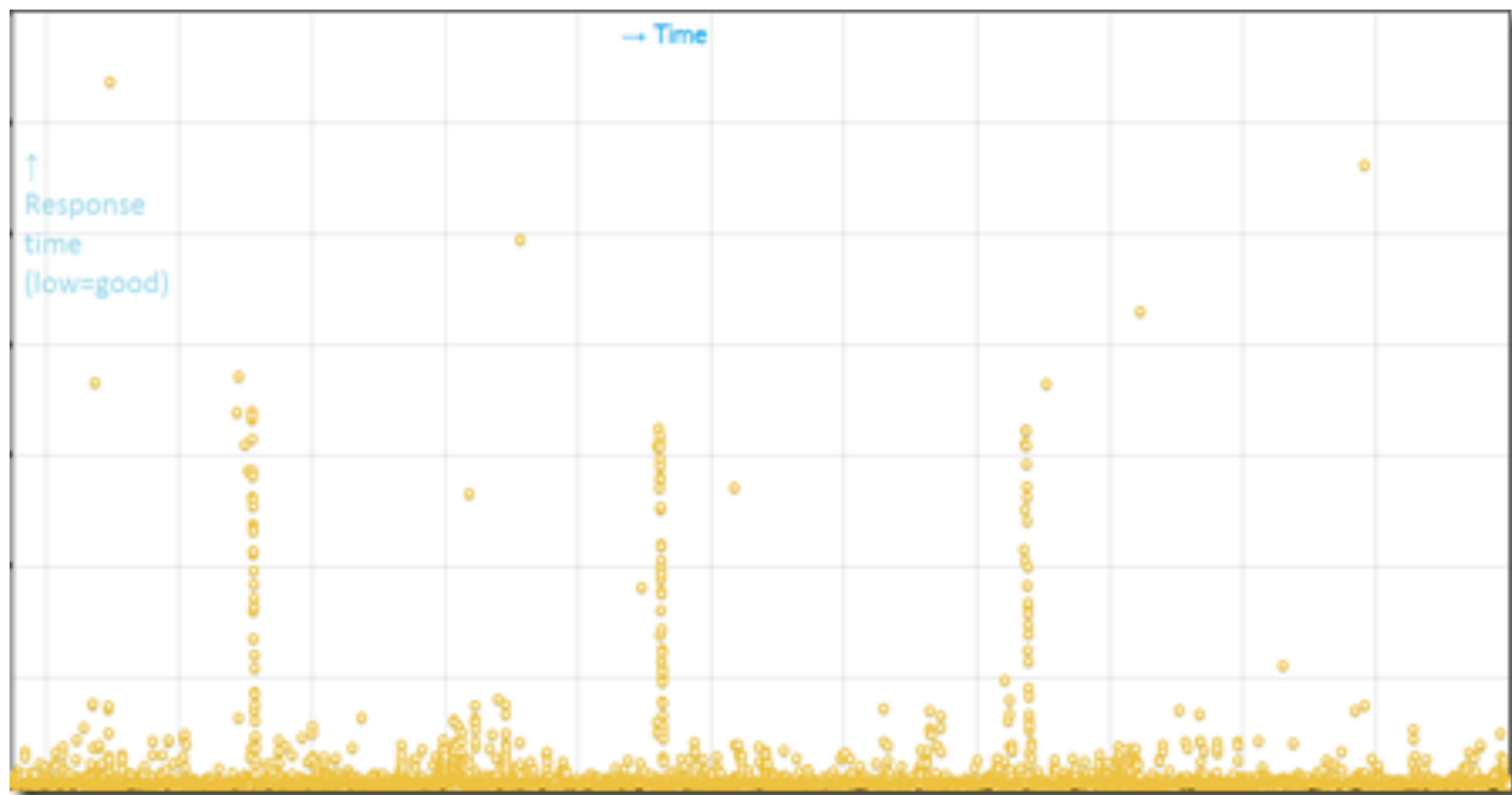
- Keep a list of all the questions (Ids and data used for sorting only)
- Use optimized sort algorithms or sub-indices

*// The tag engine typically serves out queries at about 1ms per query.
Our questions list page runs at an average of about 35ms per page
render, over hundreds of thousands of runs a day.*

-- Sam Saffron, "In Managed Code We Trust"

→ Time

↑
Response
time
(low=good)



Monday, 24 October 2011

Assault by GC

TL;DR;

We had performance spikes, which we eased with some insane use of `structs`.

History

For a while now, we had been seeing a problem in the Stack Exchange engine where we would see regular and predictable stalls in performance. So much so that our sysadmins blogged about it [back here](#). Annoyingly, we could only see this problem *from the outside* (i.e. haproxy logging,

Game programming tricks

- **List<Posts>**
 - Dynamically allocates space
 - Keeps a list of pointers only
- **Post []**
 - Space fixed once and for all at load time
- **struct Post**
 - No pointer, direct embedding

Game programming tricks

- **Post.List<Tags>**
 - Becomes a pointer
 - Size of "Post" is variable
- **Post.Tags[5]**
 - Space fixed once and for all at load time

Game programming tricks

- **Index = int[]**
 - C-style "pointers"
- **fixed**
 - Prevents GC from moving memory

*" Just use
Linq2Objects, right?"*



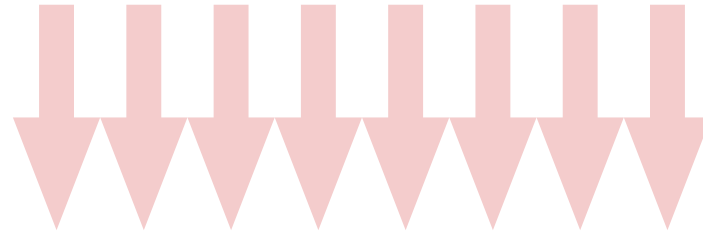
```
var results = new int[50];
var count = 0;
var relatedTags = new Dictionary<string, int>();

for (var i = 0; i < Index.length /* 10,000,000 */; i++)
{
    if (matcher.Match(Posts[Index[i]]) {
        count++;
        if (count<51) {
            results[count-1] = Posts[Index[i]];
        }
    }

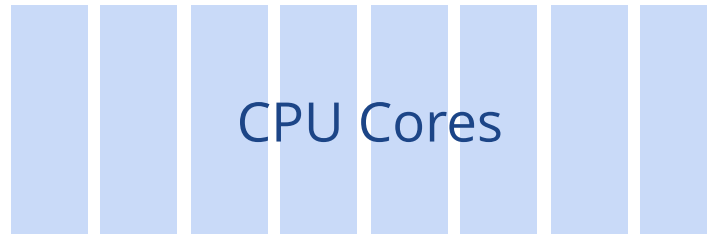
    for(int t = 0; t<5; t++) {
        relatedTags[Posts[Index[i]].Tags[t]] =
            relatedTags[Posts[Index[i]].Tags[t]] + 1;
    }
}
```

*“ Just go parallel,
right?”*



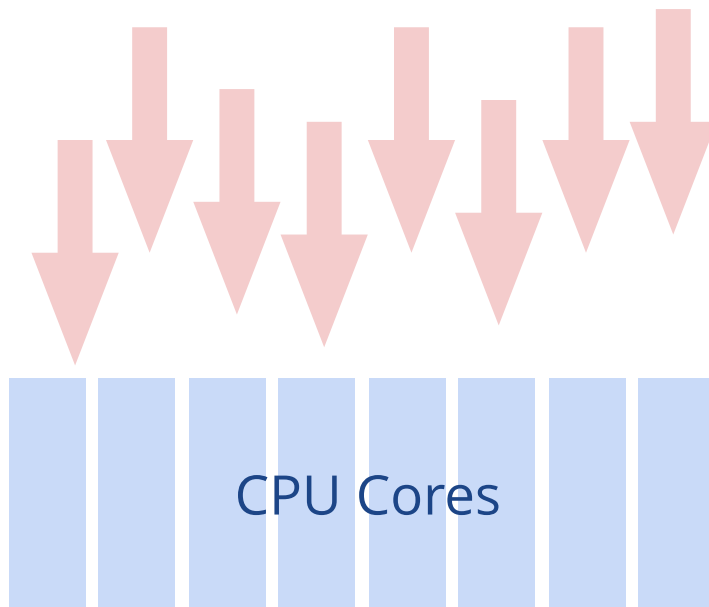


tag engine

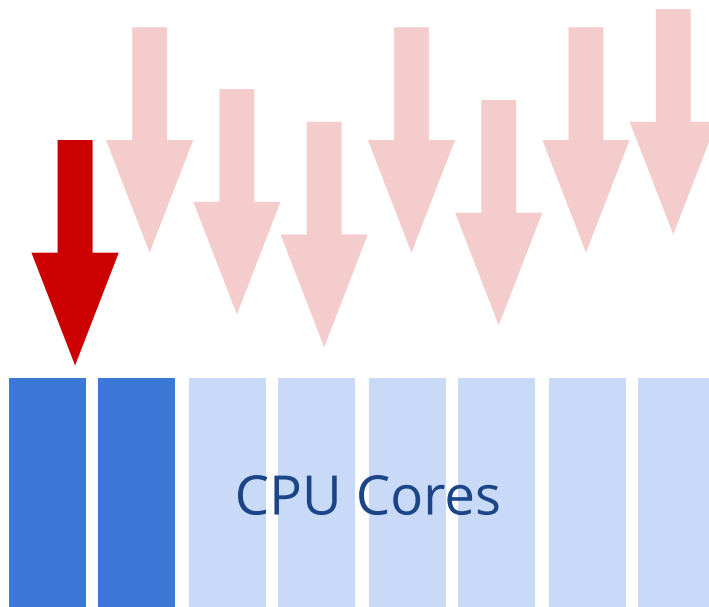


CPU Cores

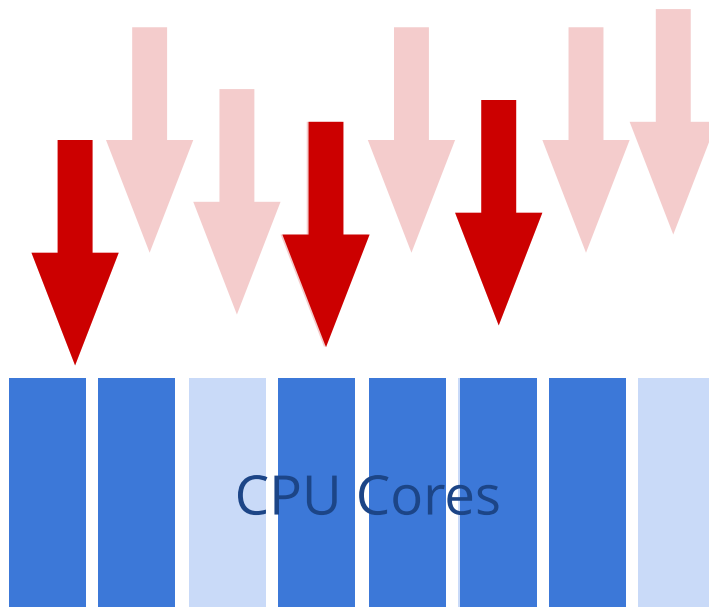
tag engine



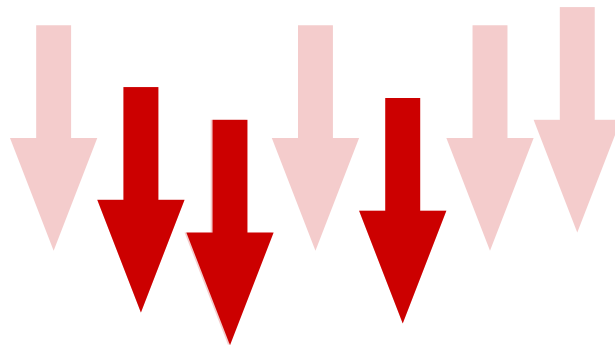
tag engine



tag engine



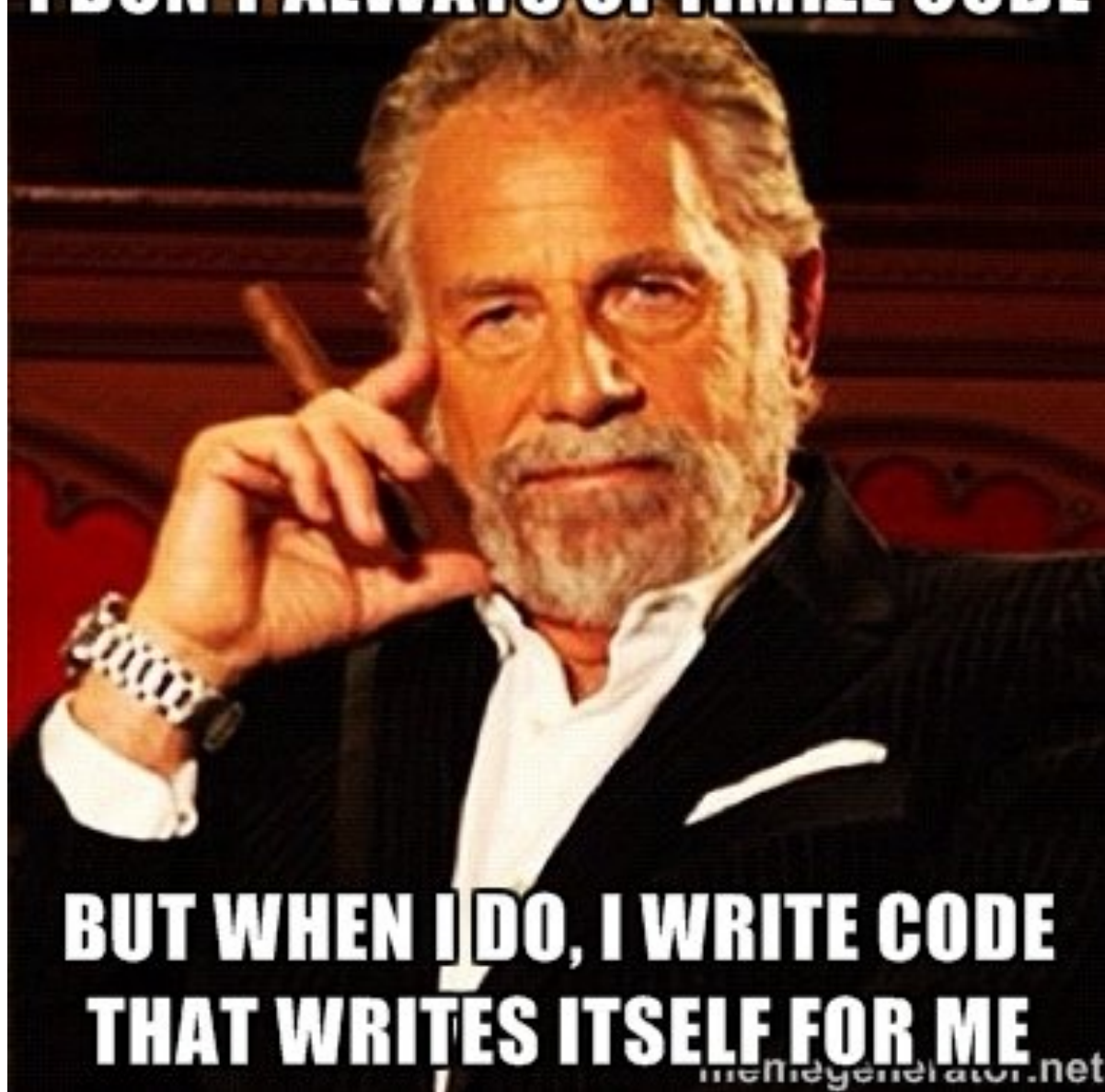
tag engine



CPU Cores

- Many concurrent queries, few cores
- Measure all the things!
- Optimal parallelism: 2 threads
- Better, but same order of magnitude

I DON'T ALWAYS OPTIMIZE CODE



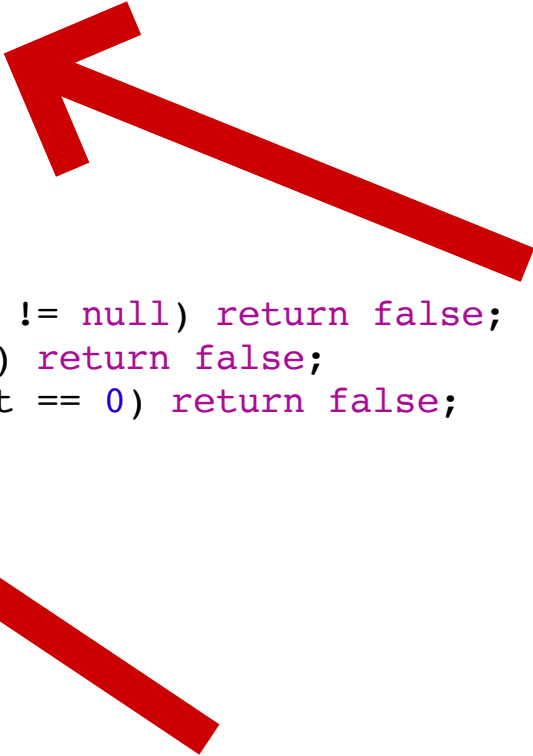
**BUT WHEN I DO, I WRITE CODE
THAT WRITES ITSELF FOR ME**

```
var results = new int[50];
var count = 0;
var relatedTags = new Dictionary<string, int>();

for (var i = 0; i < Index.length /* 10,000,000 */; i++)
{
    if (matcher.Match(Posts[Index[i]]) {
        count++;
        if (count<51) {
            results[count-1] = Posts[Index[i]];
        }
    }

    for(int t = 0; t<5; t++) {
        relatedTags[Posts[Index[i]].Tags[t]] =
            relatedTags[Posts[Index[i]].Tags[t]] + 1;
    }
}
```

```
public class Matcher {  
  
    public bool NonClosedOnly { get; set; }  
    public bool NoAnswerOnly { get; set; }  
    public bool BountiesOnly { get; set; }  
  
    /* a few of these */  
  
    public static bool Match(Post) {  
        if (NonClosedOnly && Post.DateClosed != null) return false;  
        if (NoAnswerOnly && Post.Answers > 0) return false;  
        if (BountiesOnly && Post.BountyAmount == 0) return false;  
  
        /** a few of these **/  
  
        return true;  
    }  
}
```



```

public interface IMatcher { public bool Match(Post post) }

public class MatcherGenerator {
    public bool NonClosedOnly { get; set; }
    public bool NoAnswerOnly { get; set; }
    public bool BountiesOnly { get; set; }
    /* a few of these */

    public Func<Post, bool> GenerateMatch() {
        var il = IL.Implement(typeof(IMatcher).GetMethod(nameof(IMatcher.Match)));
        Label exclude = il.DefineLabel();

        if (NonClosedOnly) {
            EmitMemberGet(il, nameof(QuestionValue.IsClosed));
            il.Emit(OpCodes.Brtrue, exclude);
        }
        if (NoAnswersOnly) {
            EmitMemberGet(il, nameof(QuestionValue.AnswerCount));
            il.Emit(OpCodes.Brtrue, exclude);
        }
        if (BountiesOnly) {
            EmitMemberComparison(il, nameof(QuestionValue.BountySize), null);
            il.Emit(OpCodes.Brtrue, exclude);
        }
        /* a few of these */

        il.EmitLabel(exclude); /* etc */
        return il.CreateDelegate();
    }
}

```

*"Just approximate,
right?"*



Many known cases where the top N results are *not representative* of the rest of the dataset

e.g. "Questions with bounties sorted by newest"



**MongoDB is web
scale, right?**





**CHANGE
AHEAD**

Not fast enough




[Save](#)

Git is an open-source distributed version control system (DVCS).

[active](#)
[votes](#)
[unanswered](#)
[most linked](#)
[learn more...](#)
[improve tag info](#)
[top users](#)
[synonyms](#)
[git jobs](#)
[merge](#)

0

votes

0

answers

2

views

How to view git log in Emacs?

[git](#)
[emacs](#)

asked 1 min ago false 10.1k

0

votes

0

answers

3

views

VXL Installation on Ubuntu

[git](#)
[ubuntu](#)
[cmake](#)

asked 2 mins ago MrD 1,019

0

votes

0

answers

3

views

word diff from two versions of a HTML document in Git

[html](#)
[git](#)
[diff](#)

asked 7 mins ago Pavel Šimerda 1,538

0

votes

0

answers

4

views

How to pull files from AWS ec2 instance?

62,027

questions



Featured on Meta



New Navigation Release Candidate

Hot Meta Posts

16 Custom tabs don't roam across clients

17 Recent spike in flag count for SO main?

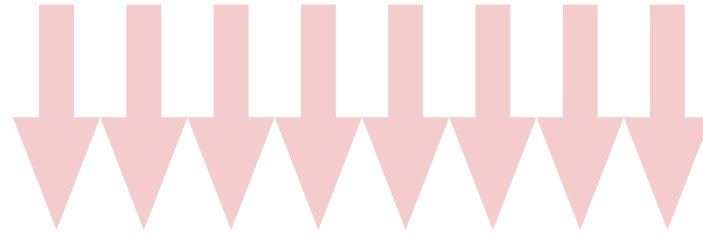
6 What's the point of downvotes when rep changes are reversed by deleting?

Favorite Tags [edit](#)

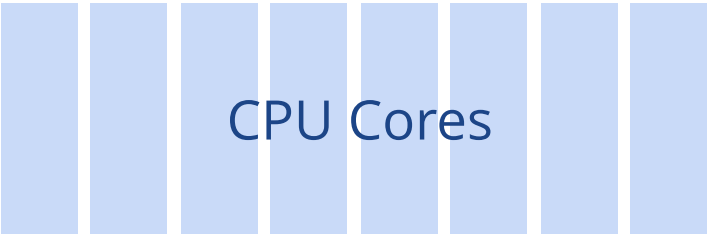
[c#](#)
[sql](#)
[sql-server](#)

Preliminary results

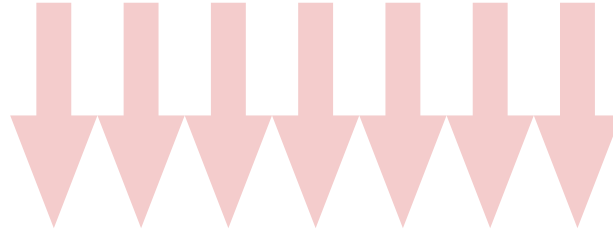
Time from 1500ms down to 700 (worst cases),
but many hits <100ms



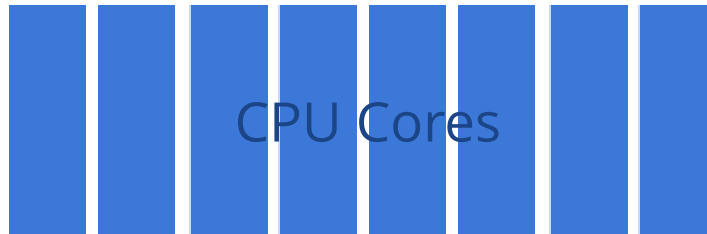
tag engine v2



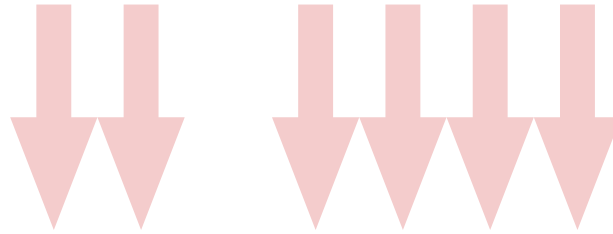
CPU Cores



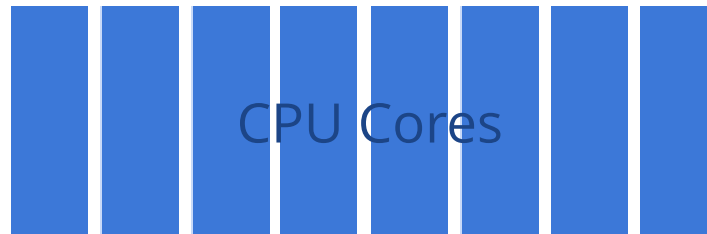
tag engine v2



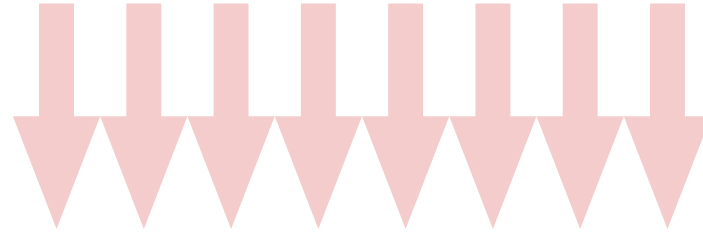
CPU Cores



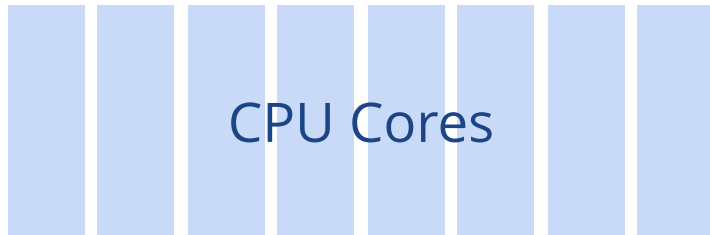
tag engine v2



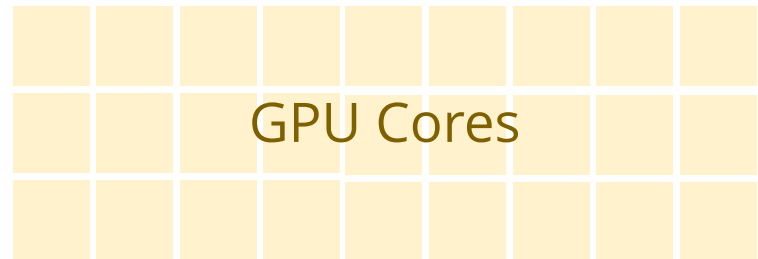
CPU Cores



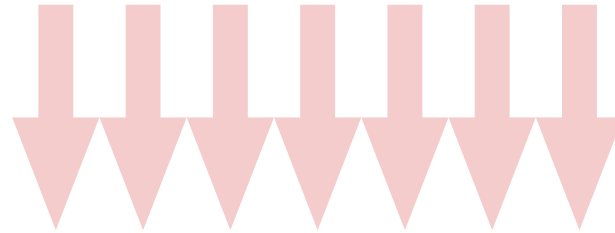
tag engine v2 GPU



CPU Cores

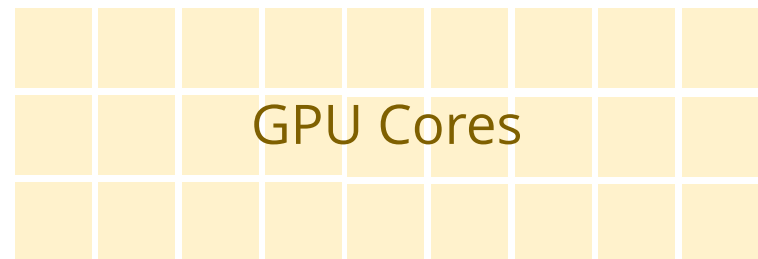
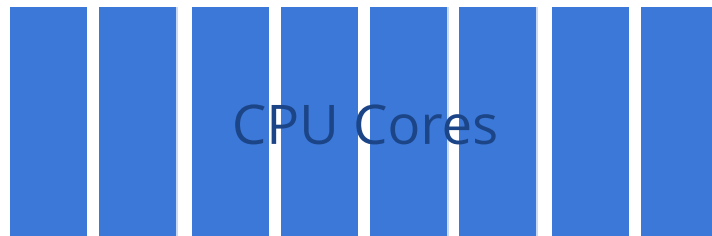


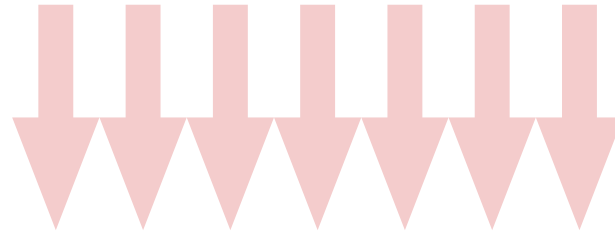
GPU Cores



tag engine v2

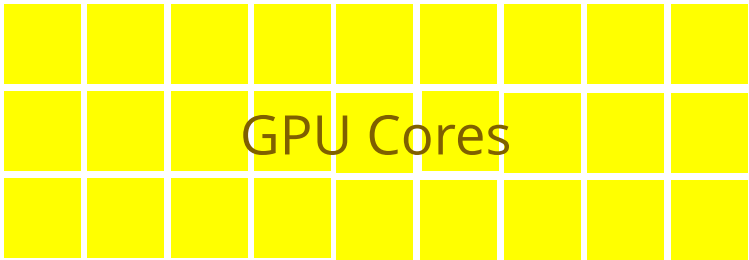
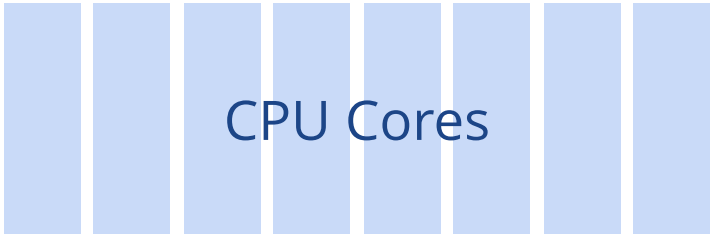
Simple query

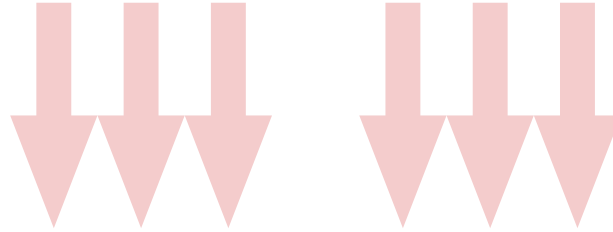




tag engine v2

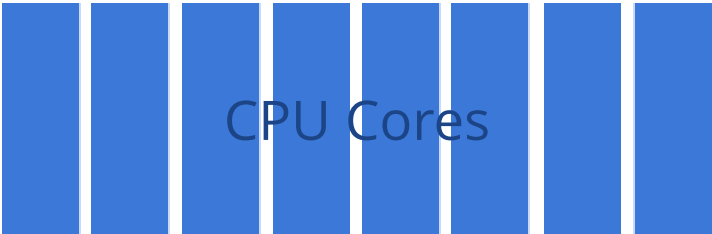
Complex query





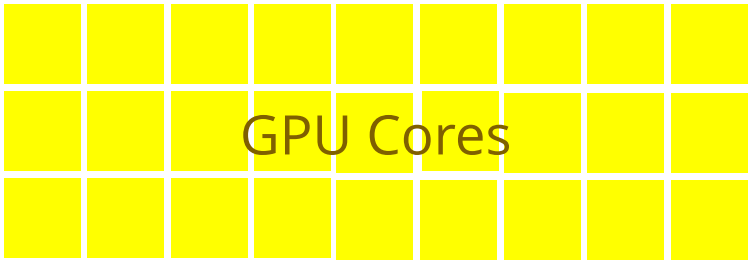
tag engine v2

Simple query



CPU Cores

Complex query



GPU Cores

Benchmarks

Rate (requests/s) - higher=better			Duration (ms) - lower=better		
GPUs	Average	Per GPU	Per PCIe	Average	Max
0 (v1)	9.6			52.59	999.7
0 (v2)	282.5			3.47	58.5
1	686.3	686.3	686.3	1.38	24.8
2	1,382.1	691.1	691.1	1.38	24.9
3	2,070.5	690.2	690.2	1.38	24.9
4	2,748.1	687.0	687.0	1.38	24.9
2	1,137.9	569.0	1,137.9	1.69	25.6
4	2,266.6	566.7	1,133.3	1.69	25.6
6	3,393.3	565.6	1,131.1	1.70	25.8
8	4,547.8	568.5	1,137.0	1.69	25.7

...the moral of the story

- Measure everything
- Don't assume anything
- Every canned is likely wrong
- Keep on evolving



- Not a Robot





```
perl -wlne  
'BEGIN{$b=rand$=}$a=qw/Tolzkh_kypam_yr_qryaimtcpdjmu_bmr_amk_enq_ghqhmfcNvm/[(($_<=>int$b)+1)];  
$a=~y/a-zA-Z/b-zA-B-ZA/;  
print $a'
```