Practical Performance - Make it go fast!

Which is Faster?

```
<cfif x IS 2>
```

OR

```
<cfif x EQ 2>
```

OR

```
<cfif compare(x,2) IS false>
```

Which is Faster?

```
<cfif x IS 2>
  <cfset doSomething() />
<cfelse>
  <cfset doSomethingElse() />
</cfif>
```

OR

```
<cfswitch expression="#x#">
  <cfcase value="2">
    <cfset doSomething() />
  </cfcase>
  <cfdefaultcase>
    <cfset doSomethingElse() />
  </cfdefaultcase>

</cfswitch>
```

Which is Faster?

```
<cfset one = doSomething() />
<cfset two = doSomethingElse() />
```

OR

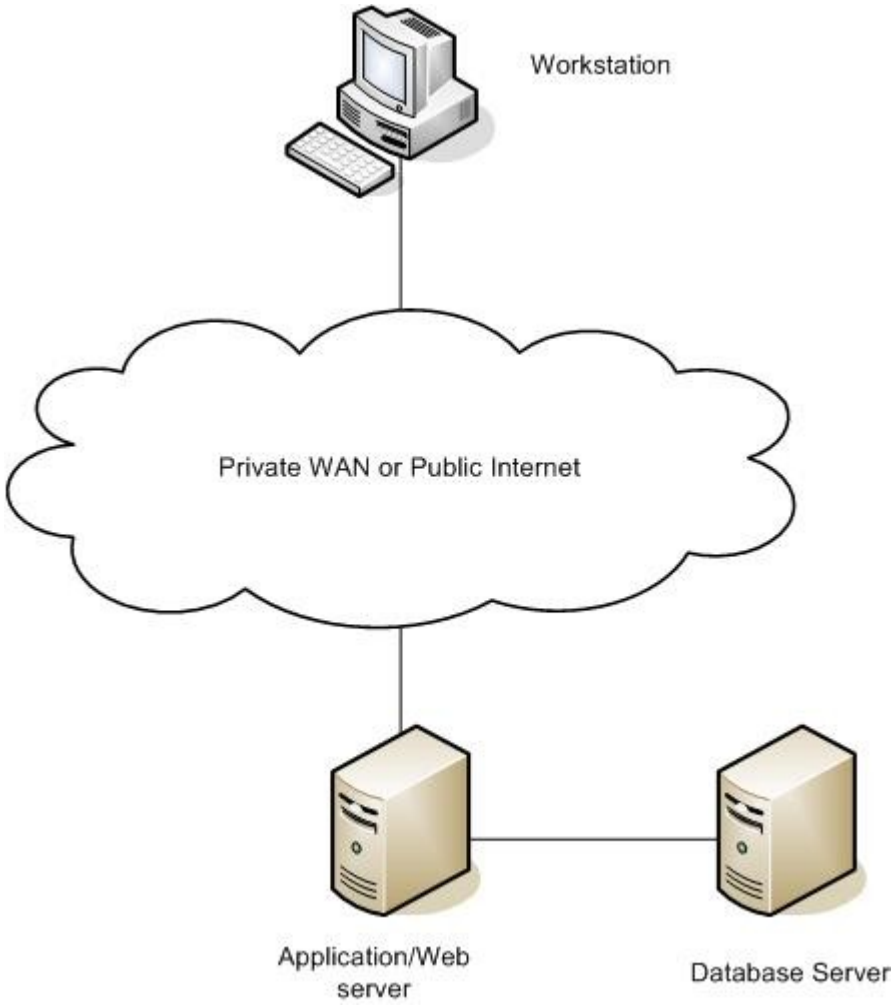
```
<cfscript>
  one = doSomething();
  two = doSomethingElse();
</cfscript>
```

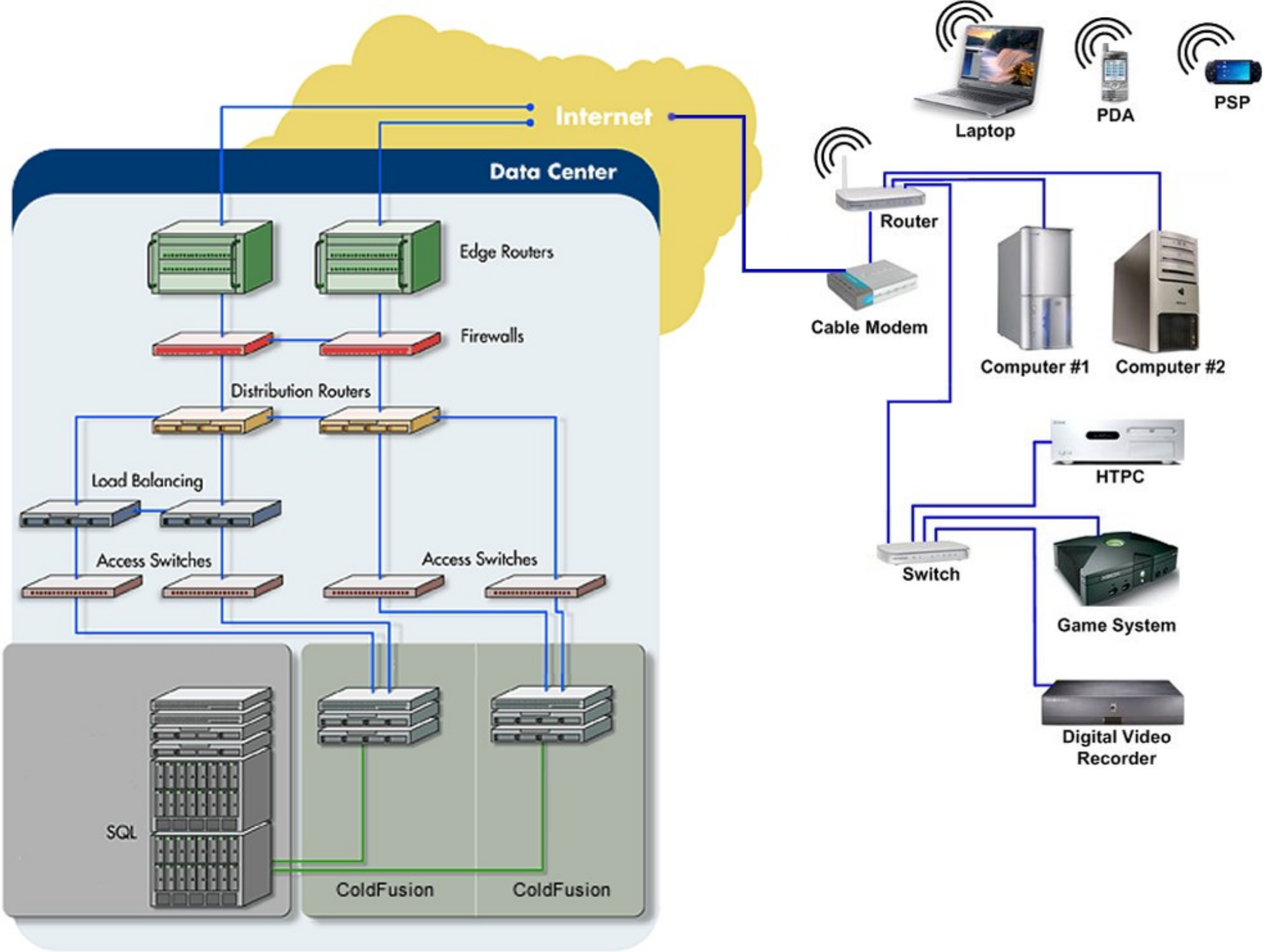
who
cares?

"We should forget about small efficiencies,
say about 97% of the time:
premature optimization is the root of all evil."
- Donald Knuth

"Knowing when optimization is premature
defines the difference between the
master engineer and the apprentice."
- Theo Schlossnagle

There's the bottleneck


Workstation
Private WAN or Public Internet
Application/Web server
Database Server


Internet
Data Center
Edge Routers
Firewalls
Distribution Routers
Load Balancing
Access Switches
Access Switches
SQL
ColdFusion
ColdFusion
Laptop
PDA
PSP
Router
Cable Modem
Computer #1
Computer #2
HTPC
Switch
Game System
Digital Video Recorder

Dividing Lines

Work Better or Work Less

Case 1 – S↑2C™ WELCOME TO STAND UP TO CANCER

Telethon for Cancer

- 18.3 million viewers
- $80 million pledged

Carried on:

Web Store Infrastructure

- 3 Windows/CF9 Std. Servers
- 1 SQL Server 2000 DB
- 1 Windows server with Squid
- 1 Load Balancer
- 1 Firewall (old)

SU2C - Web Store Traffic

Export | Email

Advanced Segments: All Visits

Dashboard

Aug 21, 2010 - Sep 20, 2010

Visits

Graph by:

30,000

15,000

Aug 23 | Aug 30 | Sep 6 | Sep 13

Site Usage

109,234 Visits

1,100,715 Pageviews

10.08 Pages/Visit

18.19% Bounce Rate

00:05:09 Avg. Time on Site

69.74% % New Visits

Visitors Overview

30,000

15,000

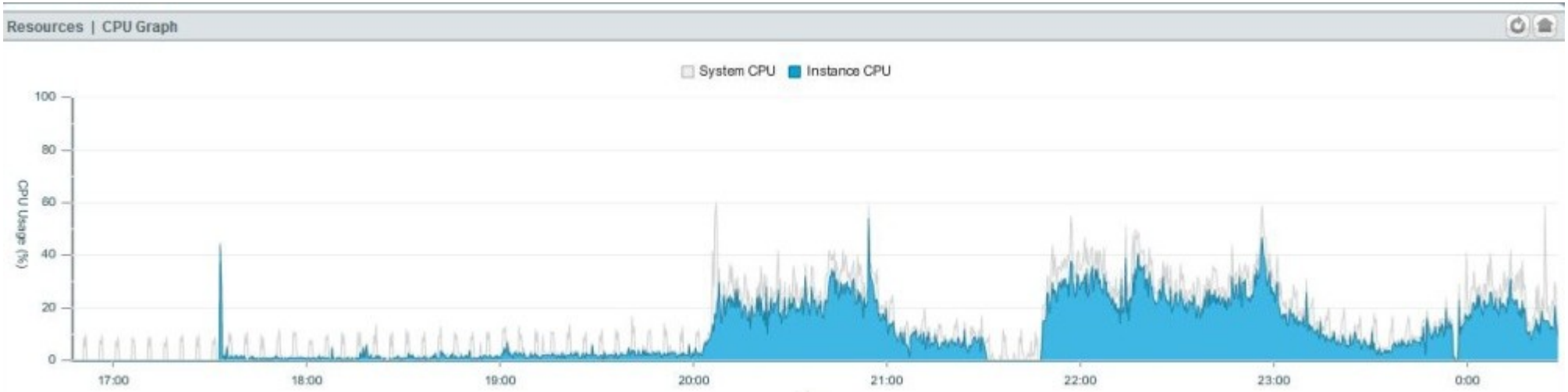
77,759 Visitors

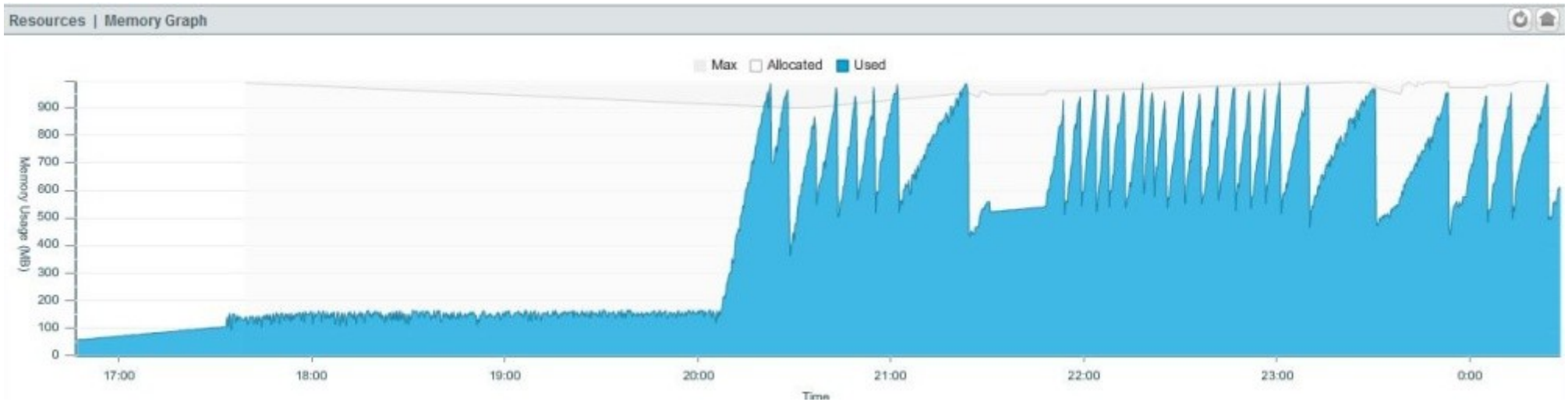
view report

Map Overlay

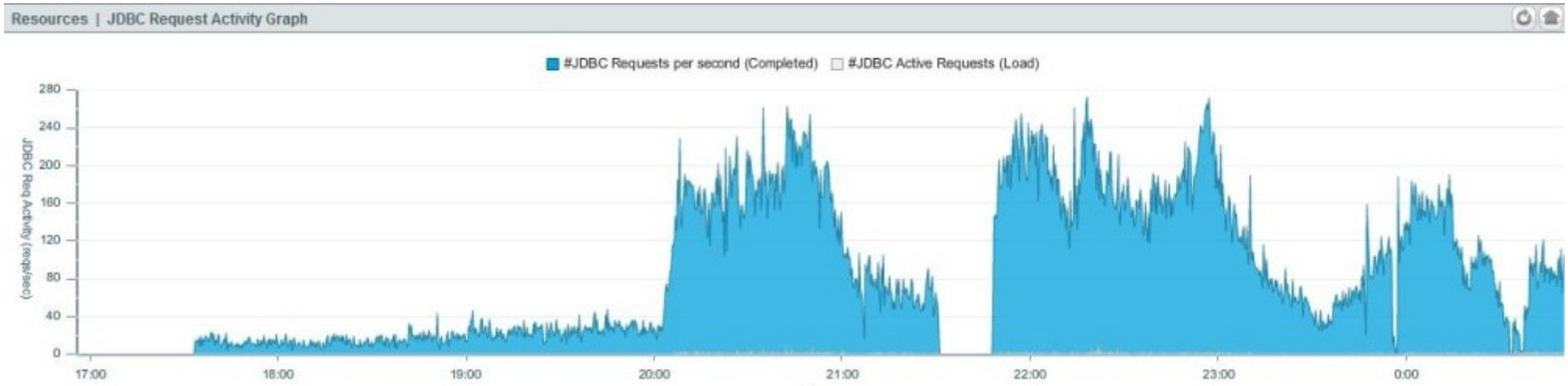
view report

SU2C Server Metrics

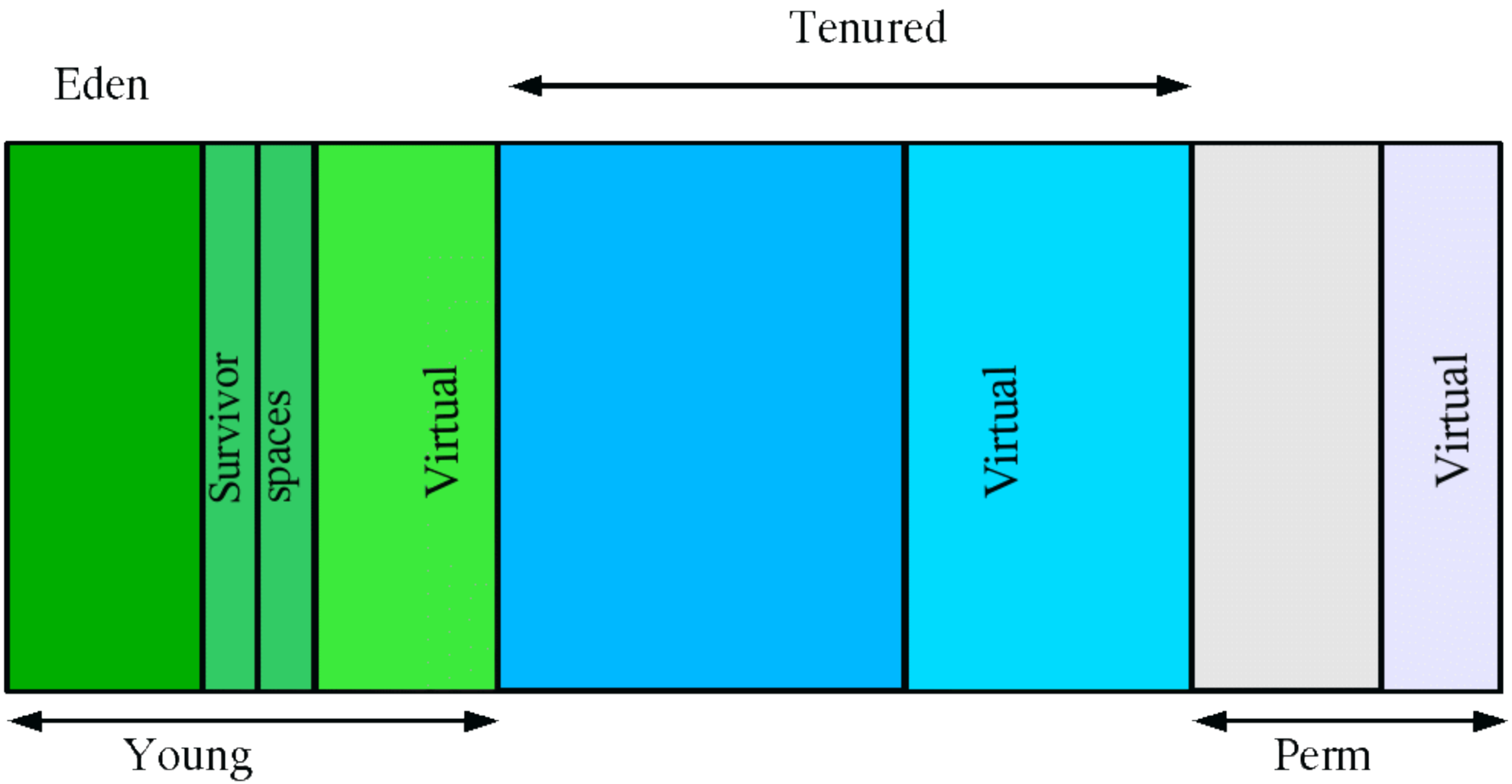




SU2C Request Metrics



Work Better- JVM Tuning



jvm.config – memory parameters

Stock CF jvm.config

```
# Arguments to VM
java.args= -server
-Xmx512m
-XX:MaxPermSize=192m
```

2GB jvm.config

```
# Arguments to VM
java.args= -server
-Xms1024m
-Xmx2048m
-XX:PermSize=128m
-XX:MaxPermSize=192m
```

JVM Garbage Collection Logging

```
java.args= -server
           -Xms1024m
           -Xmx2048m
           -Dsun.io.useCanonCaches=false
           -XX:PermSize=128m
           -XX:MaxPermSize=192m
           -XX:+UseParallelGC
           -XX:+PrintGCDetails
           -XX:+PrintGCTimeStamps
           -XX:+PrintHeapAtGC
           -XX:+PrintTenuringDistribution
           -verbose:gc
           -Xloggc:dwilsongc.log
           -Xbatch
           -Dcoldfusion.rootDir={application.home}/
```

Garbage Collection Log Sample

```
{Heap before GC invocations=81 (full 2):
 PSYoungGen      total 54592K, used 272K
  eden space 50944K, 0% used
  from space 3648K, 7% used
  to   space 3520K, 0% used
 PSOldGen        total 466048K, used 387261K
  object space 466048K, 83% used
 PSPermGen       total 131072K, used 62175K
  object space 131072K, 47% used
  3605.644: [Full GC (System)
  [PSYoungGen: 272K->0K(54592K)]
  [PSOldGen: 387261K->62994K(466048K)] 387533K->62994K(520640K)
  [PSPermGen: 62175K->62175K(131072K)], 1.2773877 secs]
  [Times: user=0.45 sys=0.03, real=1.28 secs]
  Heap after GC invocations=81 (full 2):
 PSYoungGen      total 54592K, used 0K
  eden space 50944K, 0% used
  from space 3648K, 0% used
  to   space 3520K, 0% used
 PSOldGen        total 466048K, used 62994K
  object space 466048K, 13% used
 PSPermGen       total 131072K, used 62175K
  object space 131072K, 47% used
}
```

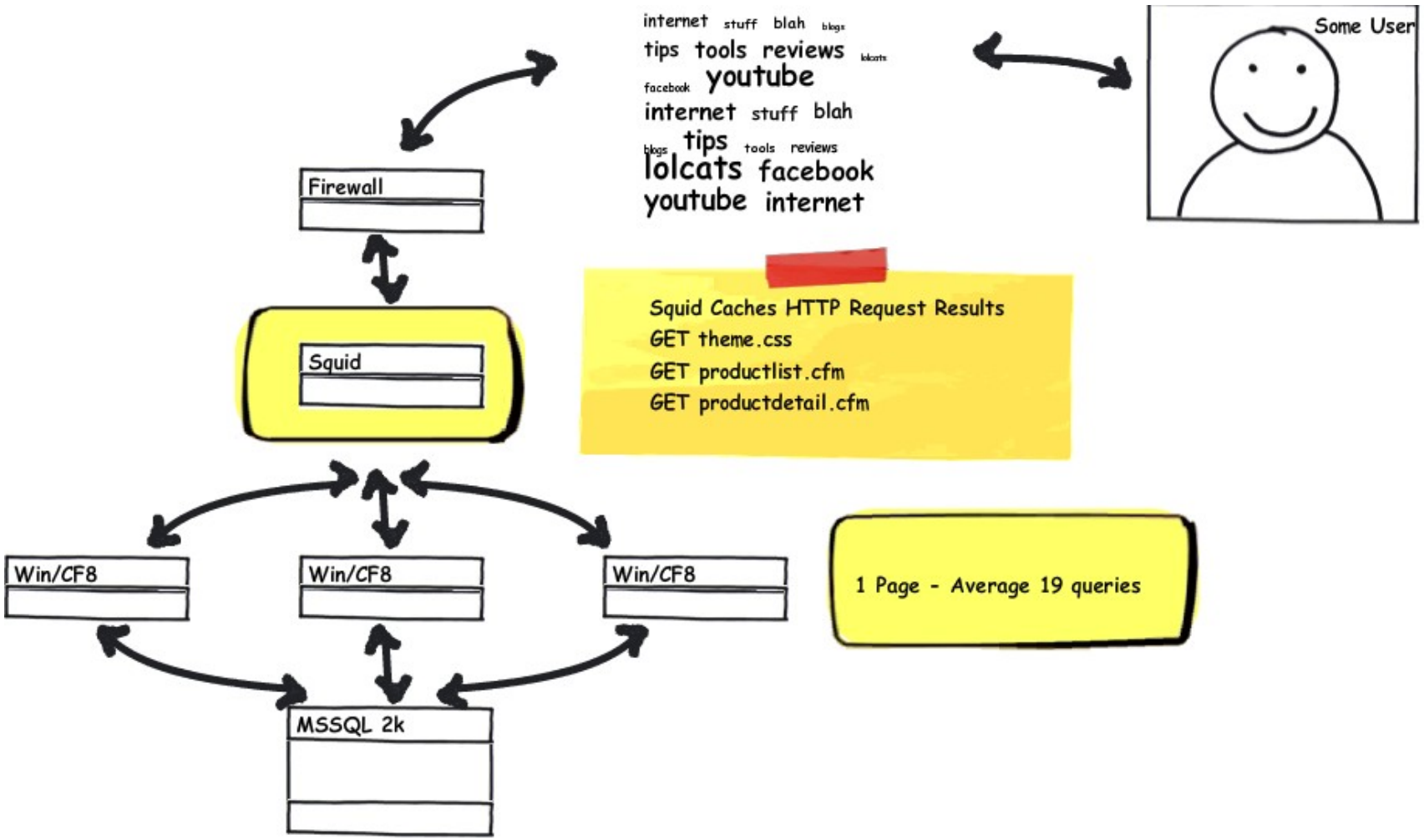
JVM Metrics Logging

```
<!-- ================================================================== -->
<!-- This Service represents JRun's logging facility                    -->
<!-- ================================================================== -->
<service class="jrunx.logger.LoggerService" name="LoggerService">
  <attribute name="format">{server.date} {log.level} {log.message}{log.exception}</attribute>
<!-- You can modify format to suit your need. -->
<!-- EXAMPLE: <attribute name="format">{date MM/dd HH:mm:ss} {log.message}{log.exception}</attribute> -->
<attribute name="errorEnabled">true</attribute>
<attribute name="warningEnabled">true</attribute>
<attribute name="infoEnabled">true</attribute>
<attribute name="debugEnabled">false</attribute>
  <!-- You may also need to uncomment MetricsService if you want metrics enabled -->
  <attribute name="metricsEnabled">true</attribute>
  <attribute name="metricsLogFrequency">60</attribute>
     <attribute name="heading">Time;Total;Listen;Idle;Running;Delayed;Time Delayed;Dropped;Completed;Time to Completion;sessions;Sessions In Memory;Total Memory;Free Memory</attribute>
  <attribute name="metricsFormat">{jrpp.totalTh};{jrpp.listenTh};{jrpp.idleTh};{jrpp.busyTh};{jrpp.delayRq};{jrpp.delayMs};{jrpp.droppedRq};{jrpp.handledRq};{jrpp.handledMs};{sessions};{sessionsInMem};{totalMemory};{freeMemory}</attribute>
  <service class="jrunx.logger.ThreadedLogEventHandler" name="ThreadedLogEventHandler">
  <service class="jrunx.logger.ConsoleLogEventHandler" name=":service=ConsoleLogEventHandler" />
  <service class="jrunx.logger.FileLogEventHandler" name="FileLogEventHandler">
    <attribute name="filename">{jrun.rootdir}/logs/{jrun.server.name}-event.log</attribute>
    <attribute name="rotationSize">200k</attribute>
    <attribute name="rotationFiles">3</attribute>
    <!-- Use heading to specify an alternate log heading for system event log. -->
    <!-- EXAMPLE: <attribute name="heading">#  Created by JRun on {date MM/dd HH:mm:ss}</attribute> -->
    <attribute name="closeDelay">5000</attribute>
    <attribute name="deleteOnExit">false</attribute>
  </service>
  <service class="jrunx.logger.FileLogEventHandler" name="MetricsFileLogEventHandler">
    <attribute name="filename">{jrun.rootdir}/logs/{jrun.server.name}-{log.level}.log</attribute>
                    <attribute name="heading">Time;Total;Listen;Idle;Running;Delayed;Time Delayed;Dropped;Completed;Time to Completion;sessions;Sessions In Memory;Total Memory; Free Memory</attribute>
    <attribute name="rotationSize">200k</attribute>
    <attribute name="rotationFiles">3</attribute>
    <!-- Use heading to specify an alternate log heading for system event log. -->
    <!-- EXAMPLE: <attribute name="heading">#  Created by JRun on {date MM/dd HH:mm:ss}</attribute> -->
    <attribute name="closeDelay">5000</attribute>
    <attribute name="deleteOnExit">false</attribute>
  </service>
</service>
</service>
```

Metrics Logging Output

```
Time;Total;Listen;Idle;Running;Delayed;Time Delayed;Dropped;Completed;Time to Completion;sessions;Sessions
In Memory;Total Memory; Free Memory
01/11 05:09:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;459769
01/11 05:10:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;459622
01/11 05:11:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;459474
01/11 05:12:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;459333
01/11 05:13:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;459193
01/11 05:14:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;459031
01/11 05:15:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;458883
01/11 05:16:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;458743
01/11 05:17:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;458597
01/11 05:18:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;458455
01/11 05:19:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;458310
01/11 05:20:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;458167
01/11 05:21:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;458022
01/11 05:22:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;457868
01/11 05:23:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;457719
01/11 05:24:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;457576
01/11 05:25:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;457432
01/11 05:26:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;457294
01/11 05:27:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;457147
01/11 05:28:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;457006
01/11 05:29:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;456851
01/11 05:30:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;456699
01/11 05:31:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;456561
01/11 05:32:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;456417
01/11 05:33:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;456271
01/11 05:34:43 metrics 1;1;0;0;0;0;0;0;0;0;0;0;518720;456126
```

Work Less - Squid



Open Source Front Caches

Squid

- Built in SSL support. Varnish needs stud, nginx or stunnel to do SSL.
- Support for antivirus-plugins
- Good support for streaming content
- CARP support
- Native on Windows, Linux
- Single Process

Varnish

- Designed from ground up as a front cache
- More powerful configuration options
- Can run on Windows via Cygwin DLL.
- CARP support
- Multithreaded

Case Study 2 - Wikimedia

Wikipedia

Wiktionary

Wikinews

Wikibooks

Wikiquote

Wikisource

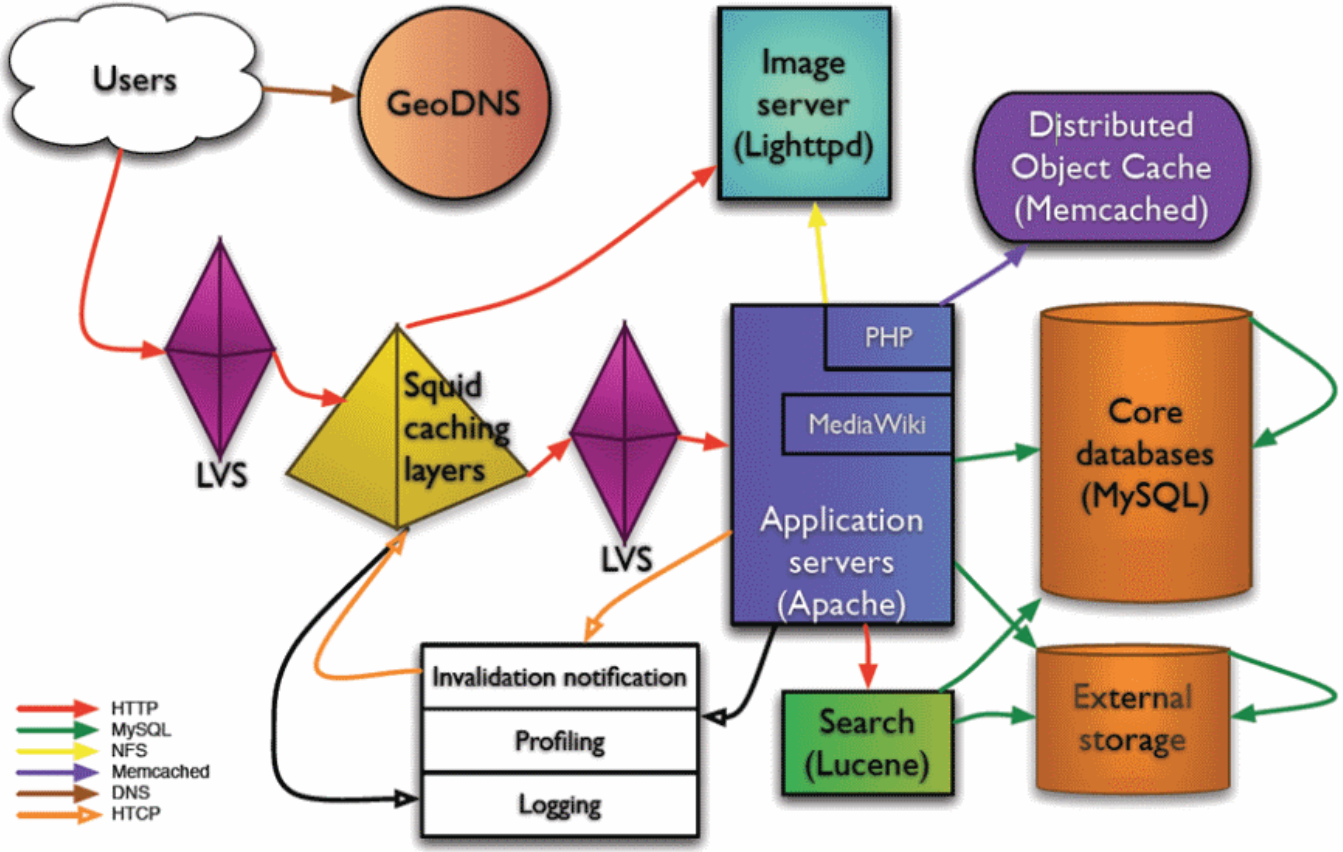
Wikiversity

Wikispecies

Wikimedia

- PHP/MySql
- 8 million articles over hundreds of language projects
- 110 million revisions
- 10th busiest site in the world (source: Alexa)
- 30,000 HTTP request/s during peak-time
- 3 Gbit/s of data traffic
- 350 servers
- managed by ~ 6 people
- Wow

Wikimedia Architecture

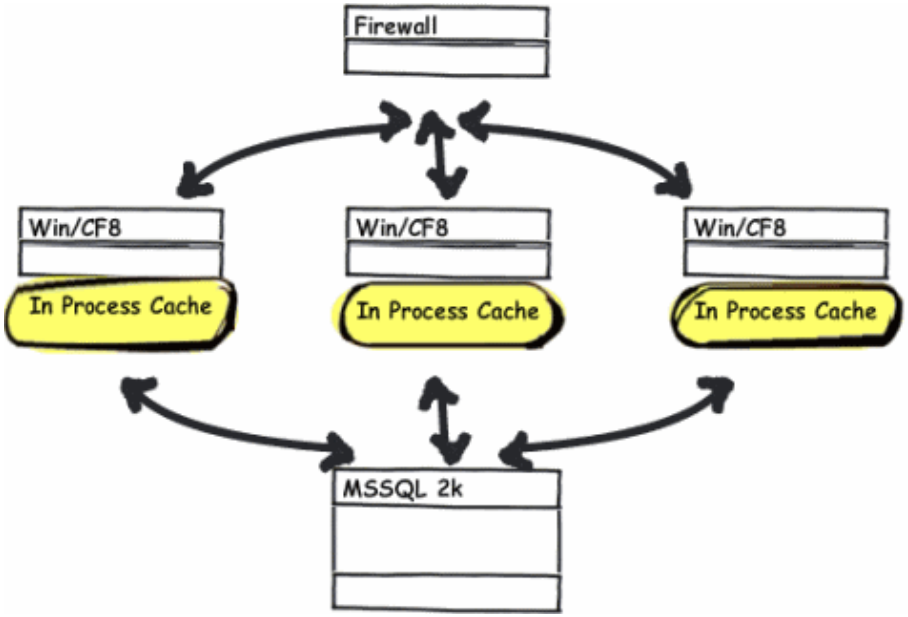


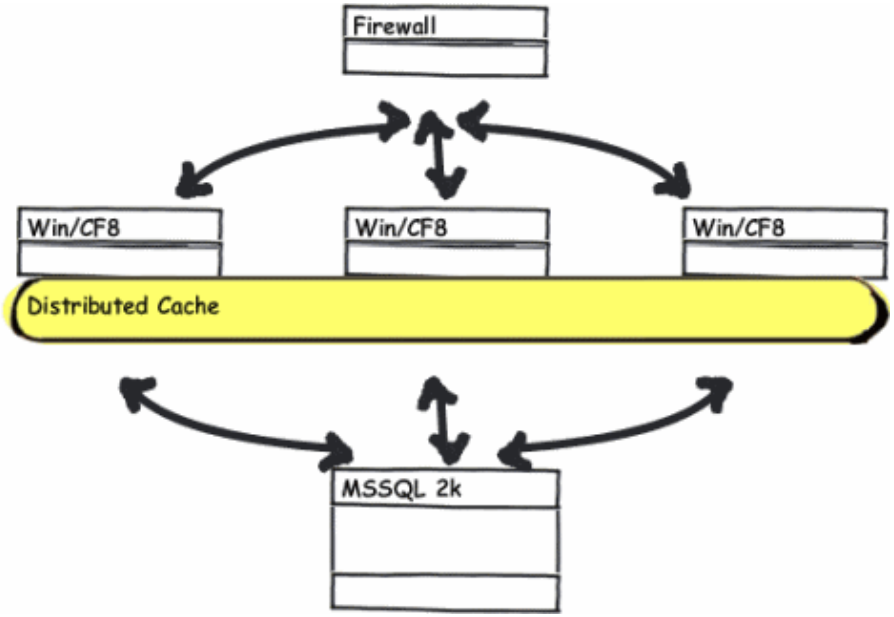
Mark Bergsma, mark@wikimedia.org, Wikimedia Foundation Inc.
Wikimedia architecture

Memory Cache

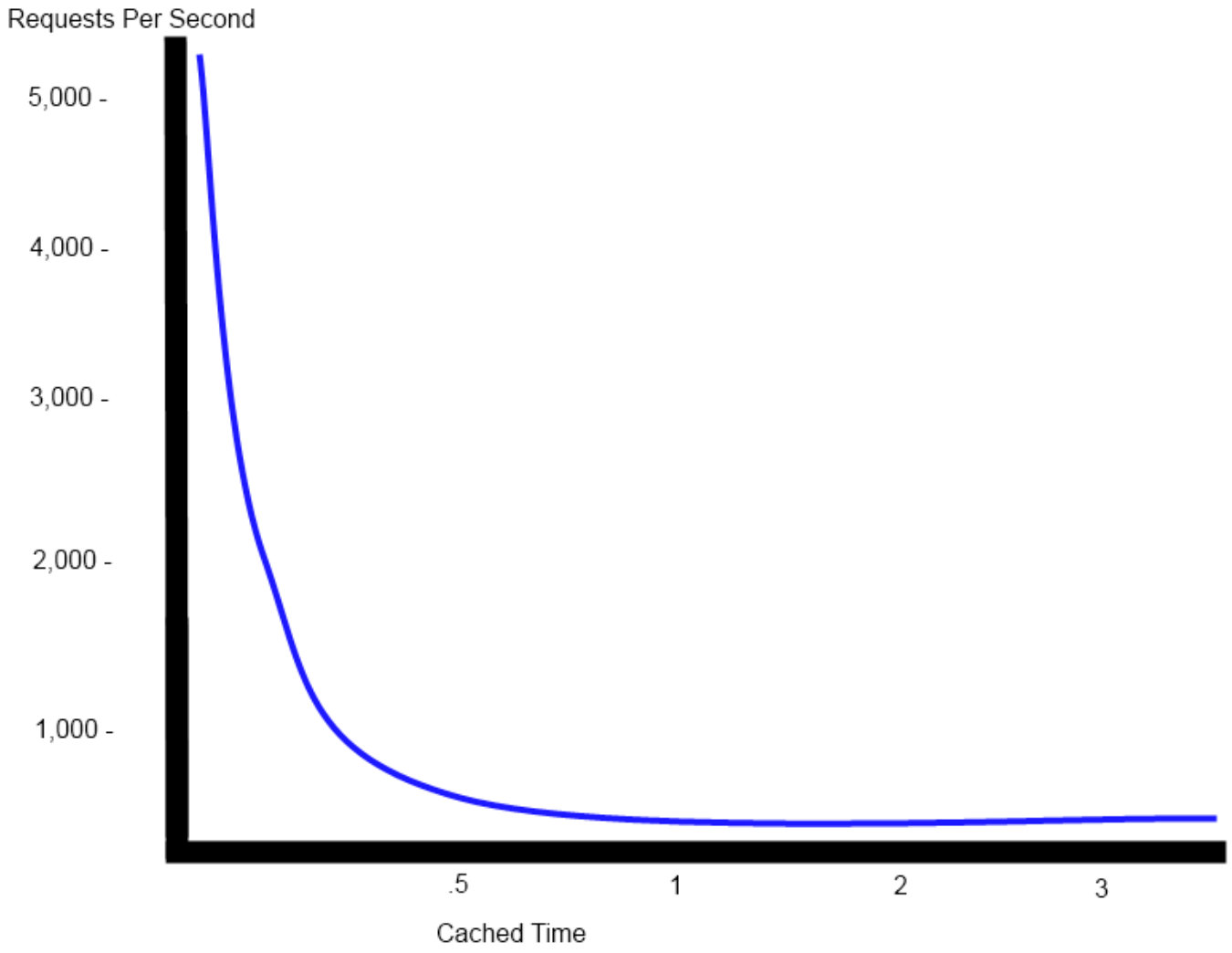
	Local Cache (Ehcache)	Memcached	Enterprise Ehcache
Easy of integration	Out-of-the-box	Simple code changes	Out-of-the-box, Snap-in
Cache API feature set	Rich: eviction, listeners, decorators, …	Simple	Rich: eviction, listeners, decorators, …
What uses (templates, ORM, other data, …) – Automatic/Manual	All	Just 'other data' (not templates/ORM) – manual)	All
Shared data between multiple machines	No*	Yes	Yes
Data capacity	Limited to JVM Heap (~1GB)	Unlimited (up to 1TB?)	Unlimited (up to 1TB)
Performance	Memory speed (~1μsec)	Network Speed (~10ms)	Memory speed (~1μsec) for hotset, network speed (~10ms) for rest
High Availability	No	No	Yes
Coherency/Drift-free consistency	No	No	Yes (configurable option)
XA Transactions	Yes (configurable option)	No	Yes (configurable option)

Memory Cache Diagram





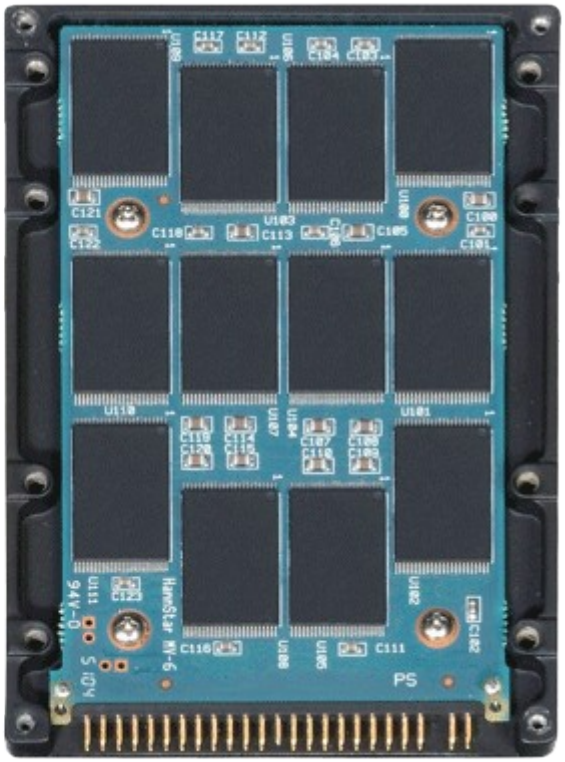
Small Interval Caching = Life Insurance



350

Avoid Work

Pick Your Disk



Shared Resource

Indexes

How Compound Indexes Work

CREATE INDEX IDX_CUSTOMER_LOCATION
on CUSTOMER (City, Country)

Finds by City or City+Country
NOT by Country or Country+City

Wilson's Rule of Database Indexes

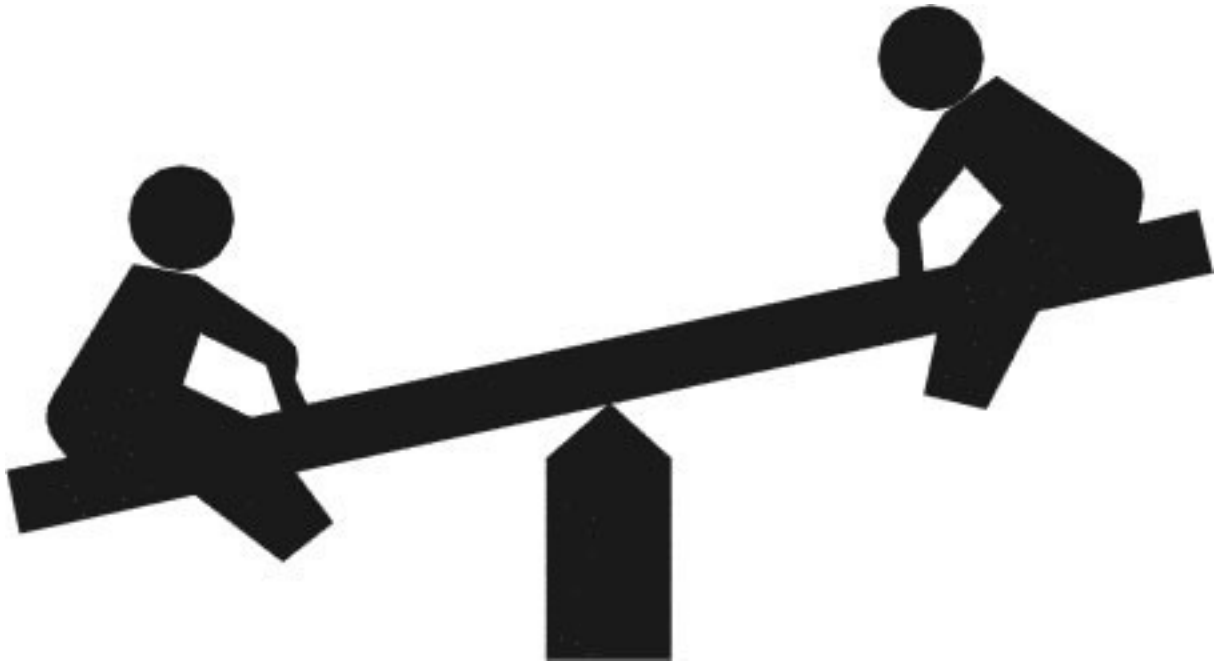
All database performance problems can be solved by creating indexes, save the problem of too many indexes

- Dan Wilson

The See Saw Tradeoff of Indexes

Index speeds up read operations at the expense of write operations.

Most web applications are read heavy. Is yours?



Punt on the hard stuff

- Real time?
- Pagination?
- Synchronous?
- Server Side?

Be Scientific! Measure Everything!

Thanks

Dan Wilson

twitter.com/DanWilson

nodans.com

datacurl.com

challengewave.com

model-glue.com