Scalable LAMP Development for Growing Web Apps

Matthew Ogle, <u>matt@last.fm</u> FoWA 2007



The Social Music Revolution © Last.fm 2007. For internal use only.

Workshop Overview

- 1. Introductions (who am I? who are you? why are you here?)
- 2. A few definitions...
- 3. Scalable Development Practices
- 4. Hardware / Software Solutions (that won't break the bank)
- 5. Social Software Growth (get open, get viral)
- 6. Open Mic / Q&A



Introductions

- About me: I joined Last.fm in early 2005
- First big project: merging Audioscrobbler.com and Last.fm (mid-2005)
- Spent nearly two years 'in the trenches' as the site and team grew rapidly...

1. Introductions



000		oscrobbler :: Home"					
★ ► X +	As http://web.archive.org/web/20041114035859/http://w	ww.audioscrobbler.com/	^ Q- Google				
	CROBBLER "Nobody Does It Like Audioscrobbler."			Î			
Home Help Get a Plugin Development Donate	Welcome to Audioscrobbler. Audioscrobbler builds a profile of your musical etc). Plugins send the name of every song you with the new song. Every person with a plugin system automatically matches you to people with	play to the Audioscrobbler server has their own page on this site whi	, which updates your musical profile ch shows their listening statistics. The				
Downloads	Find out more Sign up Download a plugin		nusic networl	nusic network – Personalised online radio station"			
Charts Groups	New Artist Similarity	Poste	84849/http://	/www.last.fm/	 Q_▼ Google 		
Forums	Artist similarity data (see 'Similar' link on artist p mins to run on my PC using the most recent data		Streetprint v	Streetprint ▼ Weather ▼ Design Blogs ▼ Journey Planner News ▼			
Log In U:	Soon as I have the artist moderation merge script at the moment.	: finished i'll re-run it, which will el	iminate all the duplicate artists you see	Labels	About 🏾 🍟 Donate	Artist	
P: Log In Lost password?	More submissions news Okay, finished the clean-up. Dragons vanquished. If you had strange tracks appearing in your Top T Submissions are open, backlog to clear. Normal s	racks list, they should vanish when		Personalised	Online Radio Station	LET'S ROLL	
Sign Up Easy/fast/free/No email required ;)	Sign Up Easy/fast/free/No Ok. seems like we eliminated the problems in the submission process, and it now runs as it should. wn online radio station that you can fill						
Find Artist	 but it might take a while due to the sheer volume * Submissions will either be rejected, or accepted will not update in the mean-time. Status report in a couple of hours 	of data that arrived when we re-o	pened submissions with the new code.	iy you new an	y you new artists and songs you might		
	Update: Will be back on the case in the morning. RJ	This is the last hurdle (i reckon) : Sign up for tree an		1	abels / Artists	✓ Start music ✓ Open popup p	
		E Listen to your favourite music no		en Fields Of	Get your music heard by the right	Forgotten your pa	
		with your personalise last.fm radio station.			people (we are serious about this!)	Recent News	
		Discover new a and songs recomme based on your music	ended ? A ? ? taste. Esqueleto Win	No C	Sell your music with links back to your online shop or as downloads via Last.fm.	 <u>Happy New Year</u> <u>Everybody!</u> <u>Radio's full stream</u> <u>power restored!</u> <u>Maximum numbe</u> <u>Listeners - Happy</u> <u>Christmas!</u> <u>Top List Update</u> <u>Rebuffering</u> <u>Submissions now</u> 	
1 Intro	ductions	Loading "http://web.archive.org/	/web/20050103084849/http://www.last.f	m/", completed 40 of	f 81 items (27 errors)	* Sunnissions now	

I. Introductions

ost.fm

The Social Music Revolution © Last.fm 2007. For internal use only.

LET'S ROLL!

> Radio's full streaming power restored! > Maximum number of Listeners - Happy Christmas! Top List Update Rebuffering

Submissions now running

Ŧ

Open popup player

(Login) Forgotten your password?

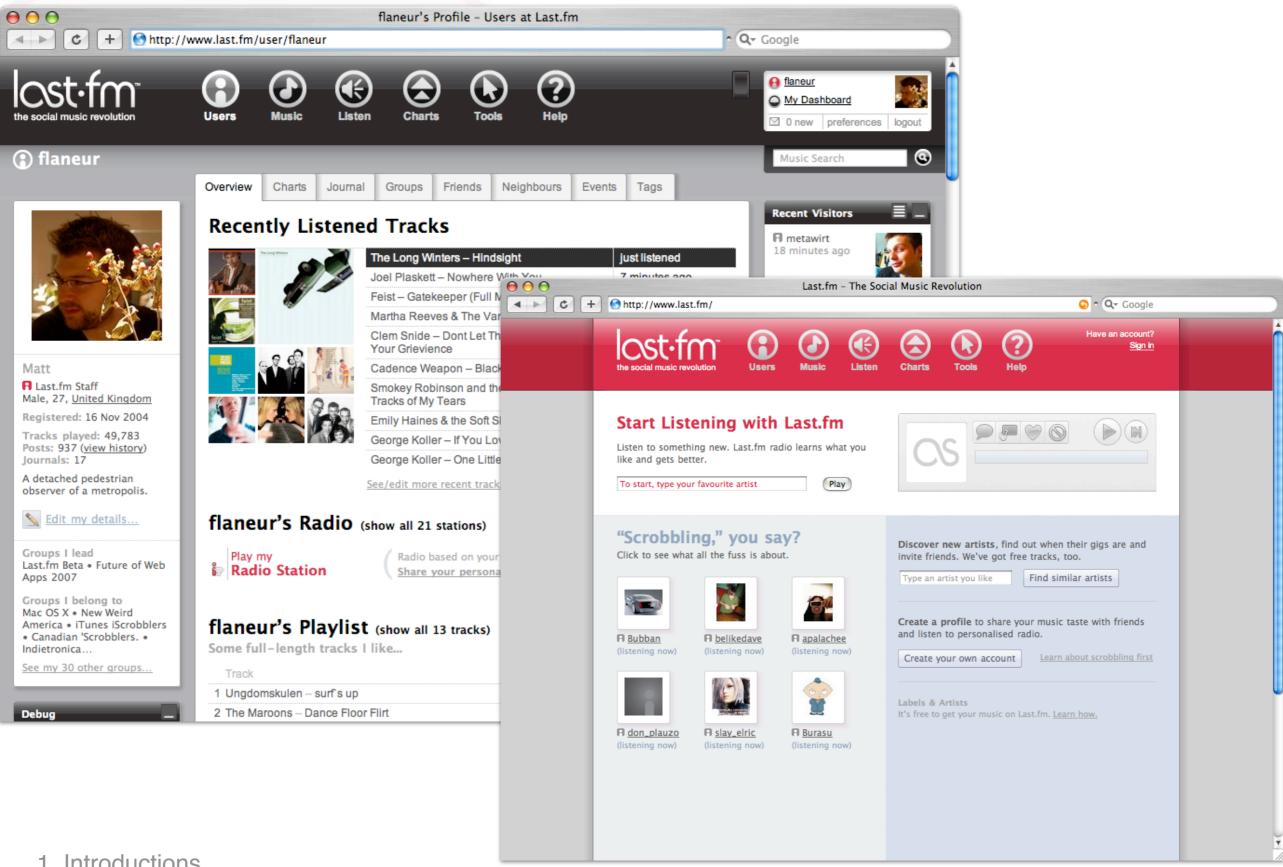
Create a new profile:

find

FREE







1. Introductions









The Social Music Revolution © Last.fm 2007. For internal use only.

Introductions

We've got a room full of web apps and expertise. Let's hear yours!

1. Introductions



Characteria Control Co

So, today is about...

Scalable LAMP Development for Growing Web Apps

2. A few definitions



Characteria Control Co

"Scalable"

- Scalability myths:
 - 1. Scalability is about performance
 - 2. Scalability requires "enterprise technology" or specific protocols/platforms
 - 3. Scalability is an architectural problem



"Scalable"

- What is a scalable system?
 - 1. It can accommodate increased usage.
 - 2. It can accommodate an increased dataset.
 - **3.** It's maintainable.

Cal Henderson, Building Scalable Web Sites

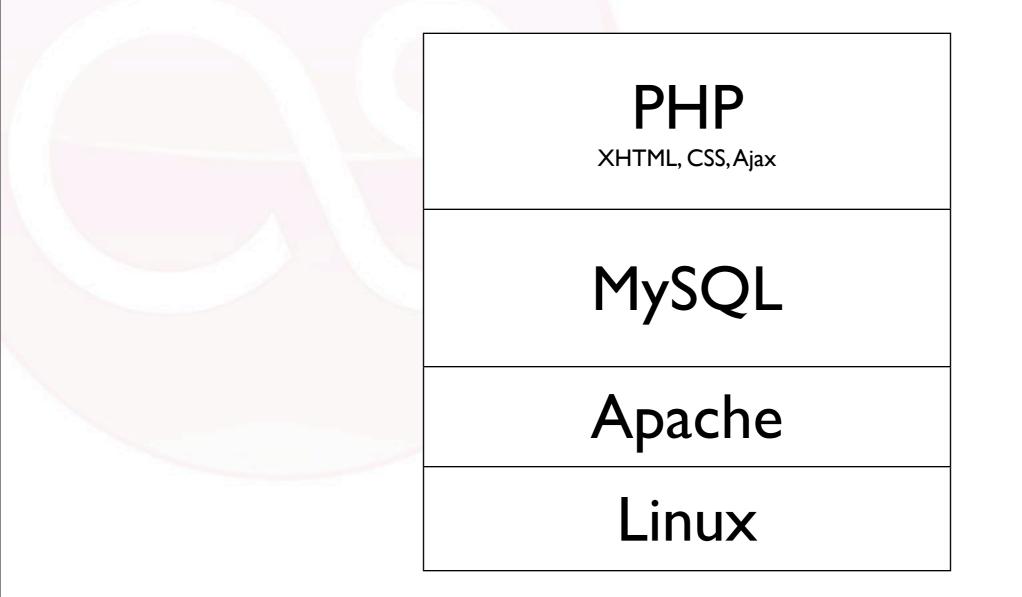


"LAMP"

- Coined in the late 1990s to describe a viable free software alternative to commercial web stacks
- Linux Apache MySQL PHP / Perl / Python ...
- LAMR, LAMAR, AMP, BAPP, MARS, FAMP ...

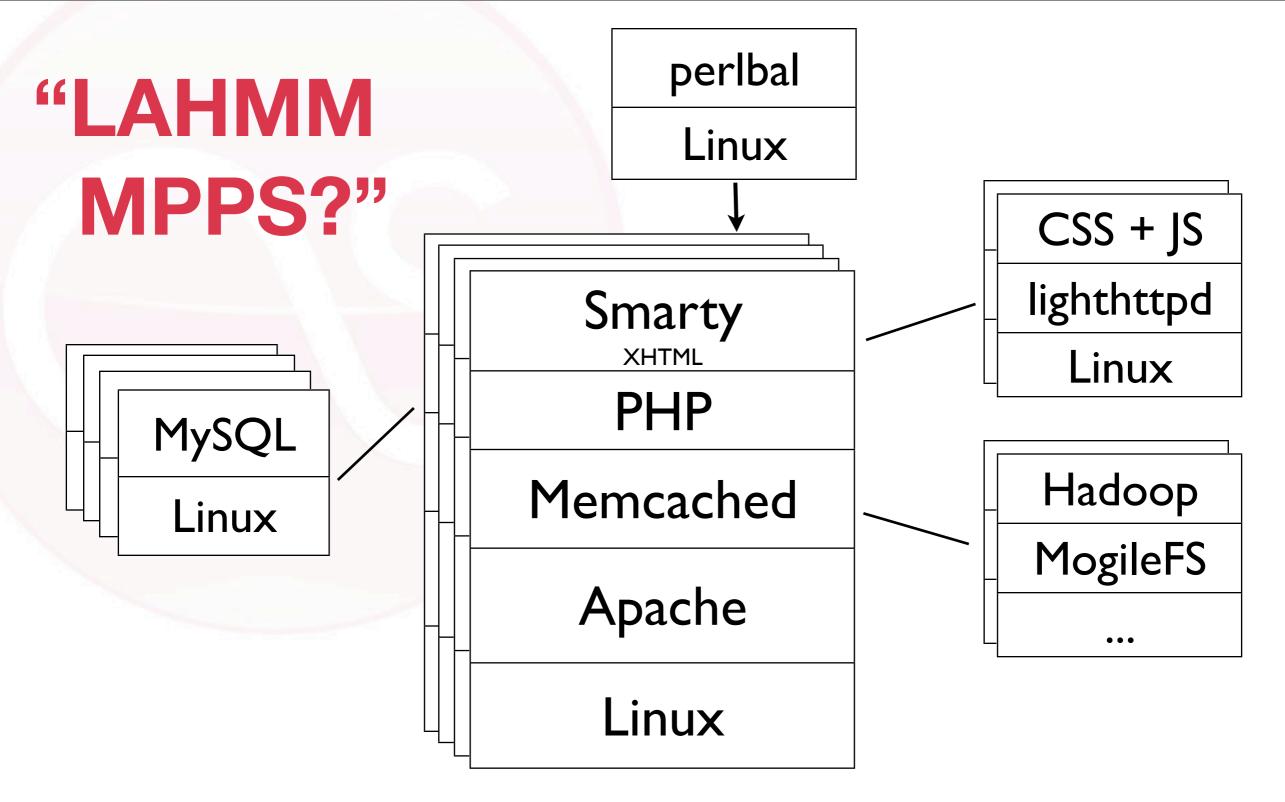






Where it all starts: single-server LAMP stack





Where we can go: horizontal LAMP scaling example



"Development"

- Development myths:
 - 1. Development teams ought to scale just like site/server growth
 - 2. Hiring more developers speeds projects up (Fred Brooks, *The Mythical Man-Month*)
 - **3.** Choosing the perfect platform (eg. Rails) means your app will practically write itself!



"Growing Web Apps"

- Growing: adjective or verb?
- When should you plan for growth?

...premature optimization is the root of all evil (or at least most of it) in programming.

– Don Knuth

- How does growth happen? Who drives it?
- What strategies can stimulate it?



The Basics

- In the beginning... files are edited directly on the server.
- Problems quickly emerge with this model:
 - 1. Hard to work collaboratively
 - 2. Hard to track what's been done and needs doing
 - 3. Site can appear broken while you work on it
- No modern web app should be developed without a source control system and a bug tracking system.



Source Control

- "The ability to undo your mistakes."
- Wide range of uses, from simple (single-developer revision history) to complex (managing large projects across multiple apps and releases)
- Last.fm strongly recommends Subversion (svn)
 <u>http://subversion.tigris.org</u>
- Learn it, use it, love it.
 FREE svn book: <u>http://svnbook.red-bean.com/</u>



Subversion Use at Last.fm

- Last.fm develops most new features in the svn trunk
- As releases approach, we branch from trunk for each major release (eg. a new public beta at http://beta.last.fm)
- We maintain a branch for each live version of the website (ie. beta.last.fm, <u>www.last.fm</u>, <u>www.lastfm.de</u>, etc)
- Bugs are fixed in the "highest" (oldest) branch in which they occur, and then changesets are merged downwards to trunk
- Any major refactoring takes place in a refactoring branch which is merged to trunk once complete
- Not the only model... what's yours?
- 3. Scalable Development Practices



Issue / Bug Tracking

- After source control, the most useful tool for a growing web app
- Helps you track and prioritize bugs and new feature development
- **Trac** is free and can be integrated with Subversion <u>http://trac.edgewall.org</u>/



trac	Management						Search			
	F					Settings Help/Guide				
	r Wik	i Timeline	Roadmap	Browse Source	View Tickets	New Ticket	Search			
Create New Ticket										
Short summary:										
Makes slides for that FoWA work	shop									
Type: enhancement 🛟										
Full description (you may u B I A 💿 🖻 — ¶		natting here):								
Maybe take some screenshots to										
Ticket Properties										
Priority: maj	or ≑		Milesto	one: General (on-	going) Support	•				
Component: www	v.last.fm	•	Keywo	rds:						
Assign to: matt				Cc:						
I have files to attach to	this ticket									
Preview Submit ticket										



	trac Integrated SCM & Project Management
--	---

	Integrated SCM & Project Managemen	nt							Search
	integrated sem of roject managemen					logged in as ma	tt Logout	Settings Help/Gu	ilde About Trac
	9	Wiki	Timeline	Roadmap	Brow	vse Source V	iew Tickets	New Ticket	Search
								Available Reports	Custom Query
Custo	Custom Query (256 matches)								
Filters									
			_						
Statu	Status 📄 new 📄 assigned 📄 reopened 🗹 closed 🔄								
Mileston	e is 🗧 Embdeddable	Media Relea	se	÷					-
								Add filter	•
Group res	sults by		Show	full descript	ion under (each result			Update
			_						<u>.</u>
Ticket	Summary				Owner	Туре	Priority A	Component	Resolution
1253						enhancement	critical	www.last.fm	fixed
1264					steve	defect	critical	www.last.fm	fixed
	it is								
1267	http://www.last.fm/liste	en/ has a	404		matt	defect	critical	www.last.fm	fixed
1276	Radio stations: new disp	olayable c	lasses, PHP		steve	enhancement	critical	www.last.fm	fixed
1277	Radio stations: new disp	olayable c	lasses, Sma	rty	julian	enhancement	critical	www.last.fm	fixed
1282	getAvatar returns no im	age in so	me case		matt	defect	critical	www.last.fm	fixed
1294	Migrate all RadioButtons	to new s	tations mod	el	matt	enhancement	critical	www.last.fm	fixed
1321	i18n footer				julian	enhancement	critical	www.last.fm	fixed
1331	Add track to playlist				julian	enhancement	critical	www.last.fm	fixed
1333	Track Pages: Playlists w	ith this tr	ack		julian	enhancement	critical	www.last.fm	fixed
1352	Make {radioStation} tag for Julian to fling about	that crea	ates station of	objects	matt	enhancement	critical	www.last.fm	fixed
1545	Implement back-end for	Friend Fi	nder		jonty	enhancement	critical	www.last.fm	fixed
1564	merge down + make su	re dev ar	tist + user i	mages	steve	defect	critical	www.last.fm	fixed

3. Scalable Development Practices



trac Integrated SCM & Project Management	Search
	logged in as matt Logout Settings Help/Guide About Trac
Wiki Timeline Roadmap	Browse Source View Tickets New Ticket Search
Roadmap Milestone: Embdeddable Media Release 1 week late (09/02/07)	Show already completed milestones Update
	92%
Closed tickets: 256 Active tickets: 22	
Milestone for the release of our embeddable media stuff, currently the I	ive release at www.last.fm.

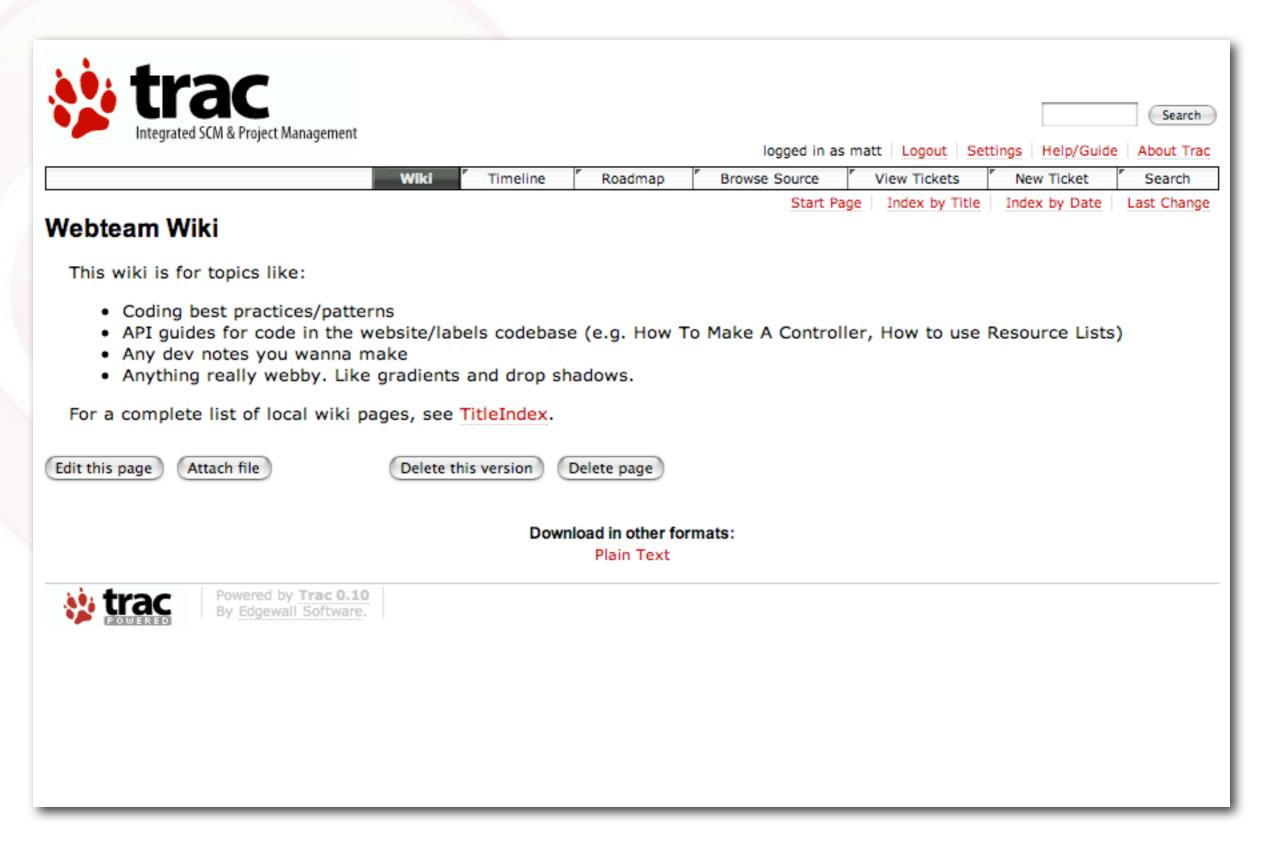


🔌 trac							
Integrated SCM & Project Managemen	nt						Search
, , ,	-		-		s matt Logout Set	-	About Trac
	Wiki	Timeline	Roadmap	Browse Source	View Tickets	New Ticket	Search
Timeline					View changes from 18	/02/07 and 3	days back.
18/02/07:					Milestones	,02,07	
14:29 Changeset [25249] by n	nokele				 Ticket changes Ticket details Repository checkin Wiki changes 	s	
03:50 Ticket #1924 (defect) c fixed: I gave it a shot. It's not			ctually, at				Update
03:49 Ticket #1931 (enhancer I did some stuff, but it would b							
03:46 Changeset [25248] by junctications so one can all the modifications so one can all the mod		nt kinds of charts	with css				
03:25 Changeset [25247] by junction of the second secon		one column for	track charts to				
17/02/07:							
20:18 Ticket #2028 (defect) u My settings are set to: Play ev			on a				
20:17 Ticket #2028 (defect) u cc changed	pdated by	matt					
20:17 Ticket #2028 (defect) u owner changed Muz, I'm offline for the rest of			on details				
20:11 Ticket #2028 (defect) c	reated by	muz					



Integrated SCM & Project Mar		-		logged in as matt Logout Settings Help/Guide About Trac			
	Wiki	Timeli	ne Roa	dmap Browse Source View Tickets New Ticket Search Last Change Revision Log			
root							
				View revision:			
Name 🔺	Size	Rev	Age	Last Change			
🛄 core		25237	2 days	edouard: fix tix #1879			
🛄 display		25248	14 hours	julian: little modifications so one can target different kinds of charts with css			
🛄 emails		10741	7 months	root: I can't commit this as my userRuss			
🗋 external		19086	3 months	mokele: removing docs - consult online docs			
🛄 flash-ads		23948	2 weeks	robl: DataLoader? now sets the number of thumbs on stage on the mpu to 12			
🛄 flash-charts		22165	1 month	h anil: flash chart widget project - initial import			
🛄 flash-playlist		24373	1 week	robl: Changed player intro sequence in fla Playlist shows trackitems			
📄 flash-tasteometer		24347	1 week	robl: Updated charts fla/swf to add widget=1 to all web service calls			
flashclient		25238	2 days	anil: flash:warner skip restrictions work			
🔲 labels.last.fm		24732	6 days	jonty: Dev branch			
🛄 languages		25192	2 days	vincro: lang update 1 of the day			
📜 lastfm		8522	9 months	s russ: Add loveban database support			
🔲 mogstreamer		17388	4 months	s rj: no throttling on previews			
🔲 new.last.fm		25249	3 hours	mokele:			
🛄 quilt		24580	1 week	robl: added noimage checking in WebServiceLoader?.as			
scripts		22335	1 month	mokele:			
🗀 sql		22306	1 month	rj: latest sql schema modifications			







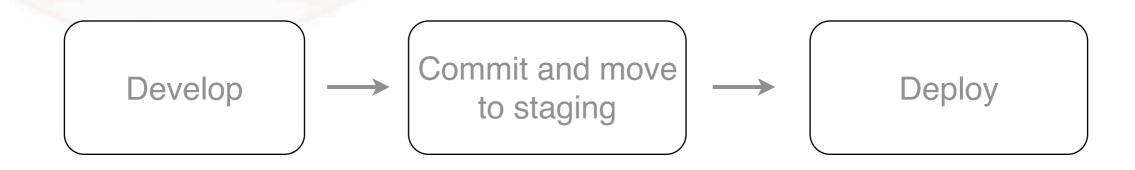
Work Environment

- Once your app starts to grow, you'll need to split your development environment into two or three parts
- Development eg. <u>http://www.dev.last.fm</u>
- Usually a dedicated server running a reduced data version of the live database
- Staging eg <u>http://www.staging.last.fm</u>
- Used to test release branches on production hardware/data



Work Environment

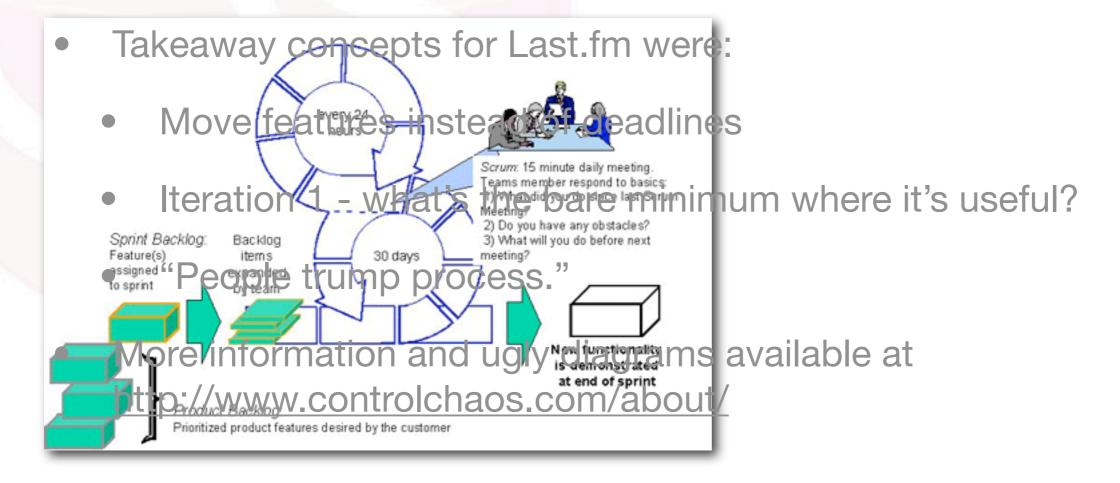
- Production (live) eg. <u>http://www.last.fm</u>
- Branches are deployed to production once they've been tested on staging.
- The whole process:





Agile Development

- Last.fm follows a modified version of the "Scrum" model
- Releases are developed iteratively over 2-4 week "sprints"





Keeping it together

- More than 5 developers... time to split into teams
- Systems guys, back end devs, front end devs, designers...
- ...directors, marketing people, interns...
- Need a way to radiate information across the company
- Endless meetings suck and prevent people from "just getting on with it"



Osmotic Communication

- Find ways to keep everyone "in the loop" as your team grows
- Example from Last.fm's IRC channel (with hooks into svn + trac)

irccat: SVN commit by norman (23872) 'randomSplitter: splits data into train and test sets randomly' (changeset: <u>https://admindev.last.fm/</u> <u>trac/changeset/23872</u>)

3:40 PM

irccat: *** jonty is refreshing webnodes now: 'Fix for group owners'

mischa: jonty: memcache key should be set to: java-playlist-10093 where 10093 ==userid.

irccat: Trac: ticket #1779 (http://support.last.fm/trac/ticket/1779) changed by julian, Comment: Fixed for the next release.

irccat: number of anon flash streams is 1440, number of registered flash streams is 764

felix: hey abc when did you put the adsense leaderboard on bottom cat pages live?

abc: friday

3:45 PM

mokele: ? lookup track 11082618

irccat: track.id(11082618) Zetan Spore ??? Subspace Distortion <u>http://www.last.fm/music/Zetan+Spore/_/Subspace+Distortion</u> (lastfm=t) pete_bug: jonty, can you please suspend PP campaign 3670?

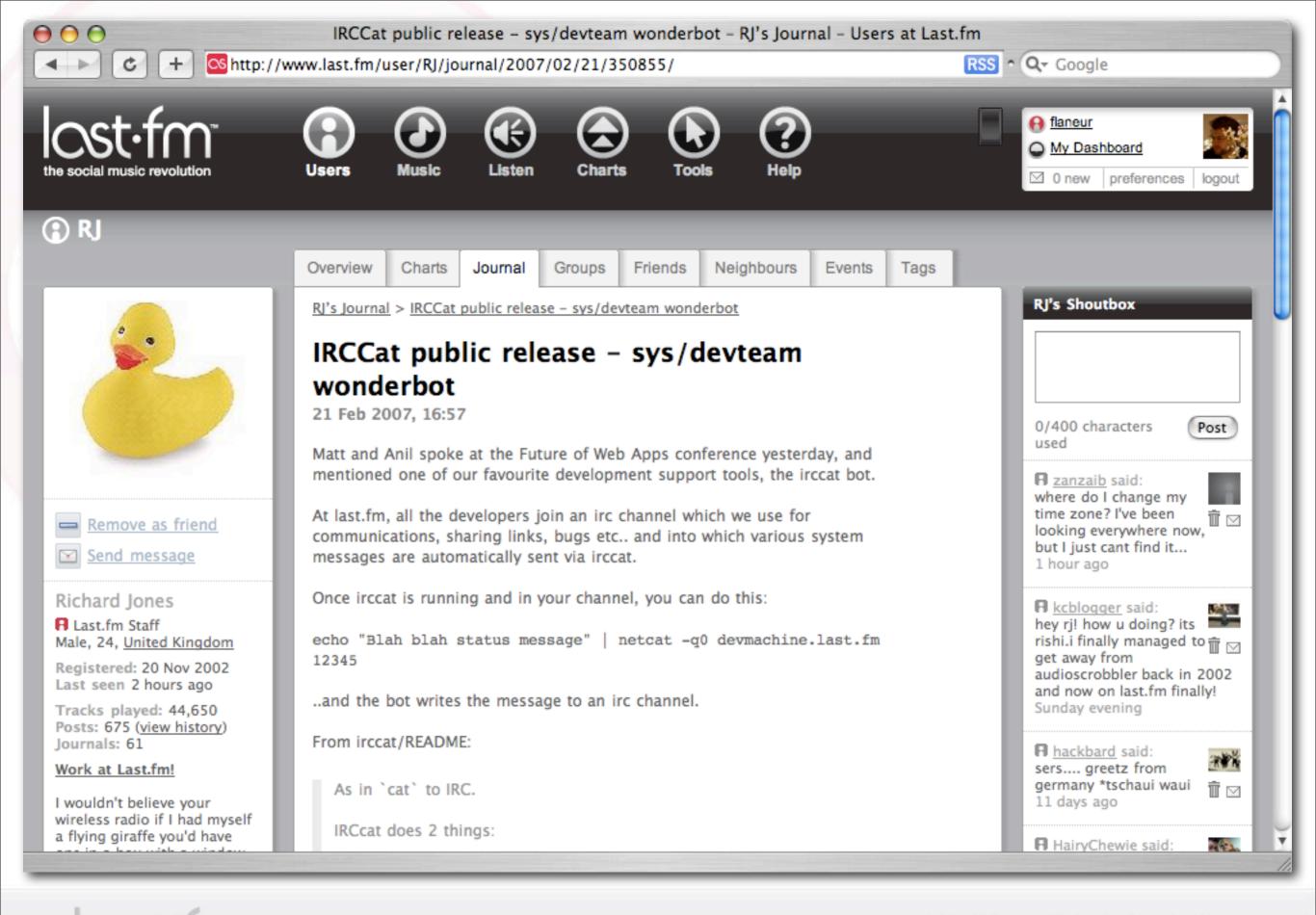
jonty: pete_bug, sure one sec.

irccat: Trac: ticket #1526 (http://support.last.fm/trac/ticket/1526) "group recommendations are slow" created by muz.

3. Scalable Development Practices



Characteria Control Co



The Social Music Revolution @ Last.fm 2007. For internal use only.

IRCCat Goes Open Source

- Grab it at: <u>http://static.last.fm/rj/irccat.tar.bz2</u>
- From the README:

IRCcat does 2 things:

- Listens on a specific ip:port and writes incoming data to an IRC channel. This is useful for sending various announcements and log messages to irc from shell scripts, Nagios and other services.
- 2) Hands off commands issued on irc to a handler program (eg: shell script) and responds to irc with the output of the handler script. This only happens for commands addressed to irccat: or prefixed with ?. (easily extend irccat functionality with your own scripts)



IRCCat SVN commit notifications

• This is what we have in our SVN repo/hooks/post-commit file:

REPOS="\$1" REV="\$2" LOG=`/usr/bin/svnlook log -r \$REV \$REPOS` AUTHOR=`/usr/bin/svnlook author -r \$REV \$REPOS` echo "SVN commit by \$AUTHOR (r\$REV) '\$LOG' <u>http://web-svn-interface.last.fm/.../?</u> rev=\$REV" | netcat -q0 machinename 12345

3. Scalable Development Practices



Characteria Control Co

Trac ticketing notifications

• Same sort of thing, but in Python this time:

```
import socket
from trac.core import *
from trac.ticket.api import ITicketChangeListener
class IrcCatListener(Component):
    implements(ITicketChangeListener)
    def sendText(self, ticketid, text):
        try:
            s = socket.socket(socket.AF INET, socket.SOCK STREAM)
            s.connect(("1.2.3.4",12345))
            s.send("#last.fm Trac: ticket #%i (<u>http://www.example.com/trac/ticket/%i</u>) %s" %
(ticketid, ticketid, text))
            s.close()
        except:
            return
    def ticket created(self, ticket):
        self. sendText(ticket.id, "\"%s\" created by %s." % (ticket.values['summary']
[0:100], ticket.values['reporter']))
    def ticket changed(self, ticket, comment, author, old values):
        self. sendText(ticket.id, "changed by %s, Comment: %s." % (author, comment[0:100]))
```

IRCCat Feedback

- Email: rj@last.fm
- Web: <u>http://www.last.fm/user/RJ</u>
- Irc: irc.audioscrobbler.com/audioscrobbler

3. Scalable Development Practices



Summing Up

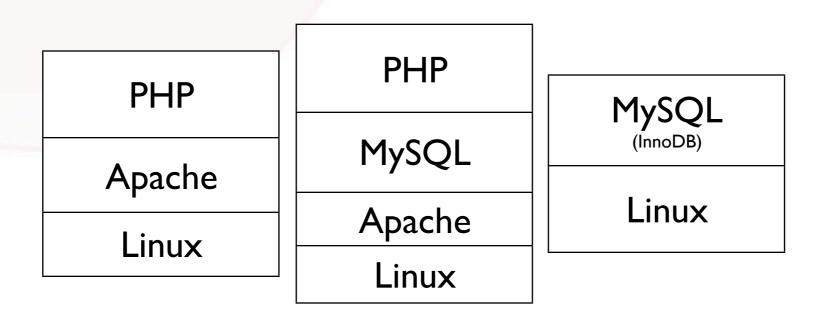
- The best tools are the simplest tools
- It something works for your team, hack it into something even better (plus you might even accidentally create Flickr or something)
- Beware expensive seminars and books titled "Exxxtreme Coding To The Max Lightweight Iterative Agilicious Productotron"
- Your process is working when it doesn't feel like a process
- People trump process

3. Scalable Development Practices



Growing Hardware + Software

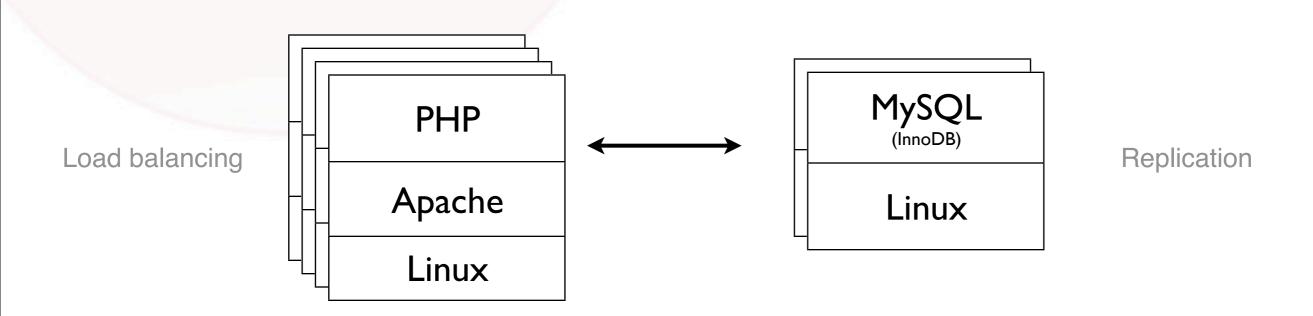
- As traffic to your app grows, a single server will quickly become overwhelmed.
- With some clever use of free software, you can help keep costs down as you begin to expand your hardware capacity.





Growing Hardware + Software

- As traffic to your app grows, a single server will quickly become overwhelmed.
- With some clever use of free software, you can help keep costs down as you begin to expand your hardware capacity.



4. Hardware / software solutions

lost.fm

Characteria Control Co

Load Balancing with perlbal

- Courtesy Brad Fitzpatrick (Livejournal), Perl-based reverse proxy / load balancer
- http://www.danga.com/perlbal/
- Sits in front of your webservers, farming incoming requests to the servers best able to handle them
- On-the-fly (no restart) configuration
- Last.fm uses perlbal with diskless netboot webservers
- Stats / reporting
- Can do interesting things with queues + priorities, URL mapping, ...



Mining perlbal data...



4. Hardware / software solutions

lost.fm

The Social Music Revolution @ Last.fm 2007. For internal use only.

Attack of the replicants

- Aside from increased load capacity / scaling abilities, there are other good reasons to replicate your databases.
- Hot spares (fail-over)
- Another backup of your data



MySQL Replication

- Replication features built in to MySQL
- Master / slave replication
- Master DB records all queries to a log, which slaves read and run locally
- Master can have many slaves, slaves can only have one master
- Can only **write** to master, can **read** from master or slave (but: slave replication is asynchronous, replication lag)
- If your app requires equal numbers of reads + writes, consider circular replication ('quasi master-master')
- Great article on advanced MySQL replication tips: <u>http://www.onlamp.com/pub/a/onlamp/2006/04/20/advanced-</u> <u>mysql-replication.html</u>



Replication-aware Apps

- You're using a database layer, right? Whew.
- Before replication, you might have code like this:

```
function getBlah() {
    global $dbmanager;
    $db = $dbmanager->getGlobalDB();
    if ($db) {
        $blah = $db->getOne("select foo from blah where id={$this->id}");
        return $blah;
    }
    else return false;
}
function saveBlah() {
    global $dbmanager;
    $db = $dbmanager->getGlobalDB();
    if ($db) {
        $blah = $db->query("update blah set foo={$this->blah} where id={$this->id}");
        return $blah;
    }
}
```



```
// meanwhile, in the DBManager class...
function getGlobalDB() {
   global $DB_DSN, $DB_OPTIONS;
   $connection = DBManager::connect($DB_DSN, $DB_OPTIONS, 'Global');
   if (!DB::isError($connection))
      return $connection;
   else return false;
```

• If functions that read and write are separated in your app logic, supporting replication can be as easy as adding a parameter.

```
function getGlobalDB($forWrite = false) {
    global $DB_DSN, $DB_OPTIONS, $DBSLAVE_DSNS, $DBSLAVE_OPTIONS;
    if ($forWrite) {
        $connection = DBManager::connect($DB_DSN, $DB_OPTIONS, 'Global');
        // handle errors and return connection
        }
    else {
        shuffle($DBSLAVE_DSNS); // let's pick a random slave to hit
        $connection = DBManager::connect($DBSLAVE_DSNS[0], $DBSLAVE_OPTIONS, 'Global Slave');
        // handle errors and return
    }
}
// then, back in saveBlah(), a one-line change...
$db->getGlobalDB(true);
```

lost fm

The Secret Weapon - Memcached

- Once again, Brad Fitzpatrick / Danga to the rescue: <u>http://www.danga.com/memcached/</u>
- "memcached is a high-performance, distributed memory object caching system, generic in nature, but intended for use in speeding up dynamic web applications by alleviating database load."
- Every app has pieces of data like user account settings which need to be read for every page load but seldom written to
- Why bother the database at all?
- Especially when you can easily run the memcached daemon across any number of machines with spare RAM... (like your webservers)
- 4. Hardware / software solutions



Memcache report

Host Vers	sion Hit Ratio	Hits:Misses	Connectio	ns Requests per second	Capacity
10.0.1.1:11211 1.2.	0 91.58%	5271815276:484522846	1243	621.70	831.73 MB
10.0.1.2:11211 1.2.	0 90.85%	3695591337:372101825	1185	619.12	828.22 MB
10.0.1.3:11211 1.2.	0 93.34%	6573610987:468812869	1322	760.61	830.62 MB
10.0.1.4:11211 1.2.	0 91.35%	5120652105:485095858	1244	605.45	830.25 MB
10.0.1.5:11211 1.2.	0 92.94%	6272542683:476791792	1222	728.99	825.86 MB
10.0.1.6:11211 1.2.	0 92.53%	5916995911:477718187	1204	690.70	828.4 MB
10.0.1.7:11211 1.2.	0 91.08%	4992092980:488648381	1143	592.13	825.46 MB
10.0.1.9:11211 1.2.	0 91.61%	5163094274:472550202	1240	608.89	828.85 MB
10.0.1.9:11212 1.2.	0 90.78%	4928402188:500718849	1234	586.58	826.67 MB
10.0.1.10:11211 1.2.	0 92.88%	878049226:67355542	1241	904.09	838.47 MB
10.0.1.10:11212 1.2.	0 91.50%	700028156:65031759	1200	731.63	838.26 MB
10.0.1.11:11211 1.2.	0 92.85%	6275341621:482969873	1259	730.28	827.35 MB
10.0.1.11:11212 1.2.	0 91.39%	5106896842:481278556	1214	603.84	826.3 MB
10.0.1.12:11211 1.2.	0 92.58%	5986400577:479666367	1238	698.76	827.99 MB
10.0.1.13:11211 1.2.	0 91.48%	5081904017:473328958	1177	600.39	830.41 MB
10.0.1.14:11211 1.2.	0 91.74%	5294506729:476452199	1191	623.74	835.14 MB
10.0.1.15:11211 1.2.	0 91.26%	4995005267:478368699	1163	591.61	829.35 MB
10.0.1.15:11212 1.2.	0 91.29%	5048759926:481579967	1172	597.76	830.74 MB
10.0.1.16:11211 1.2.	0 91.34%	5090904134:482456886	1160	601.83	828.14 MB
10.0.1.16:11212 1.2.	0 92.22%	5657426905:477079238	1165	662.43	827.59 MB
10.0.1.17:11211 1.2.	0 92.40%	5815506666:478117815	1159	680.29	830.88 MB
10.0.1.17:11212 1.2.	0 92.43%	5887950607:482051120	1192	688.55	829.65 MB
10.0.1.19:11211 1.2.	0 92.30%	5776806525:482147105	1206	676.58	831.89 MB
10.0.1.19:11212 1.2.	0 91.59%	5303936929:487231640	1172	626.01	831.61 MB
10.0.1.20:11211 1.2.	0 93.49%	6968114403:484964245	1202	805.68	830.29 MB
10.0.1.20:11212 1.2.	0 91.70%	5339037827:483222456	1170	629.39	833.06 MB
10.0.1.21:11211 1.2.	0 91.29%	1349009458:128744084	1157	750.52	836.22 MB
10.0.1.22:11211 1.2.	0 91.42%	188243951:17668303	1137	804.38	838.12 MB
10.0.1.23:11211 1.2.	0 91.64%	196797544:17942376	1145	836.34	838.18 MB
10.0.1.24:11211 1.2.	0 92.12%	202551281:17317128	1153	861.01	837.09 MB

Average hit ratio: 91.9%

Average reqs/sec: 683.976

Total Capacity: 24932773356 bytes (24.93 GB)

Using memcached in your app

- Mature memcached libraries exist for PHP, Perl, Python, Ruby, Java, C#, and C.
- Like most caching systems, very simple interface: you get or set values based on keys
- Interesting approach to distributed caching: when you set a value, the API hashes your key to a unique server (by hashing to an integer modulo # of memcache servers you have)



Basic usage pattern

```
function getBlah($id=1){
  global $dbmanager, $memcache;
  $key = "blah:$id";

  // see if blah exists in memcache:
  $blah = $memcache->get($key);

  // if blah doesnt exist..
  if(!$blah){
    // connect to db:
    $global =& $dbmanager->getGlobalDB();
    // get blah from database:
    $blah =& $global->getOne("select blah from table where id=?", array($id));
    // save blah to memcache for next time
    $memcache->set($key, $blah, 86400); // expires after 24hrs
}
return $blah;
```



Sample pattern for classes

```
class Blah {
    function Blah($db row) {
        // constructor populates instance vars from db row
        $this-> load($db row);
    }
    function cache() {
        // save the current ver of this object into memcache
        global $memcache;
        $memcache->set(Blah::getCacheKey(), $this, 86400); // 24 hrs
    }
    function uncache() {
        // clear the cached ver of this object (after writing to db, etc)
        global $memcache;
        $memcache->delete($this->getCacheKey());
    }
    function getCacheKey($id = false) {
        if (!$id)
            $id = $this->id;
        return "blah:$id";
    }
    // static function - Factory pattern stylee
    function getById($id) {
        $key = Blah::getCacheKey($id);
        ... // check memcache for key, return if found
        ... // otherwise query db and construct with Blah($db row)
    }
```

volution

Memcached gotchas, pt 1

- Need to be careful not to re-cache objects
- Imagine a database : Artist id 1 = Radiohead Album id 1 = OK Computer
- Let's assume that your classes for Artist and Album have 'getByld' methods which can be called to return objects which (ideally) have been stored in memcache:

\$album = Album::getById(1); // retrieved from memcache with key 'album:1'
\$artist = Artist::getById(1); // retrieved from memcache with key 'artist:1'

• If we further assume that an album always has an associated artist object, it's tempting to make Album::getById function such that

\$artist = \$album->artist; // same \$artist object, stored as a var in \$album



Memcached gotchas, pt 1

- Problem: what happens when you call \$album->cache()?
- Keep your classes atomic and your caching infrastructure will have more room to store all your objects in memory

\$artist = \$album->getArtist(); // getArtist() can make a call to Artist::getById(),

4. Hardware / software solutions



Characteria Control Co

Memcached gotchas, continued

- You can only invalidate single keys, not wildcard ranges (can't say: delete 'blah:*')
- What happens when you add more servers?
- What happens when you need the same object multiple times on a page?
- What about get_multi?

"We use it a lot. We divide the data for a given page into "stuff we need immediately for the business logic that will change what other data we need to fetch," "stuff we need for the business logic that we can evaluate in isolation," and "stuff we're going to display." The first gets fetched as needed during the execution of the page. The second and third, we queue up internally and request all in one big "get" just before rendering the page at the end of the request; for the second class of data, we have a callback mechanism wrapped around the memcached client so that we can run our business logic using some of the returned data. There are some additional wrinkles but that's the rough idea."

– Steven Grimm, Facebook



The Social Music Revolution © Last.fm 2007. For internal use only.

Serving Static Content

- Apache and PHP (or Python, or...) work well together, but aren't optimized for serving static files (straight HTML, CSS, images, etc)
- Last.fm serves javascript and CSS from a dedicated server running lighthttpd (<u>http://www.lighttpd.net</u>/)
- User avatars, artist images, etc are served seamlessly from multiple machines thanks to **MogileFS** (<u>http://www.danga.com/</u><u>mogilefs/</u>), an open source distributed file system



Versioning CSS + Javascript

• Problem 1:

Some browsers cache CSS and Javascript forever, and you need a painless way to push out improvements and bugfixes.

• Problem 2:

Browsers load CSS + JS more quickly if they aren't contained in many separate files, but it's hard to develop collaboratively when everything's in one big file



Compiled CSS + Javascript

- At Last.fm we split our CSS and Javascript into multiple small files, broken down by site area (CSS) or functionalities (JS)
- On our static server, /js/ and /css/ contain numbered directories and a 'source' directory
- The source directory contains all our small files and a Makefile.
- Here's what happens when you 'make install' in our CSS source:

```
Main.css: Main_orig.css
cat Main_orig.css | sed -e 's/{$$static}/http:\/\/static.last.fm/g' > ./Main.css
Main_orig.css: clean
cat *.css > Main_orig.css
install: Main.css
cp Main.css "../"`find ../ -type d -iregex "\.\.\/[0-9]+" | sed -e 's/\.\.\///' | sort -
rn | head -n1`
clean:
cp Main.css Main.bak
rm -f Main_orig.css
rm -f Main_orig.css
```

Compiled CSS + Javascript

 Our javascript make file uses Rhino to include error checking and to compression:

```
main.js: main_orig.js
      java -jar /usr/share/java/custom_rhino.jar -c ./main_orig.js > main.js
main_orig.js: clean
      cat *.js > main_orig.js
install: main.js
      cp main.js ../`find ../ -type d -iregex "\.\.\/[0-9]+" | sed -e 's/\.\.\///' | sort -
rn | head -n1`
```

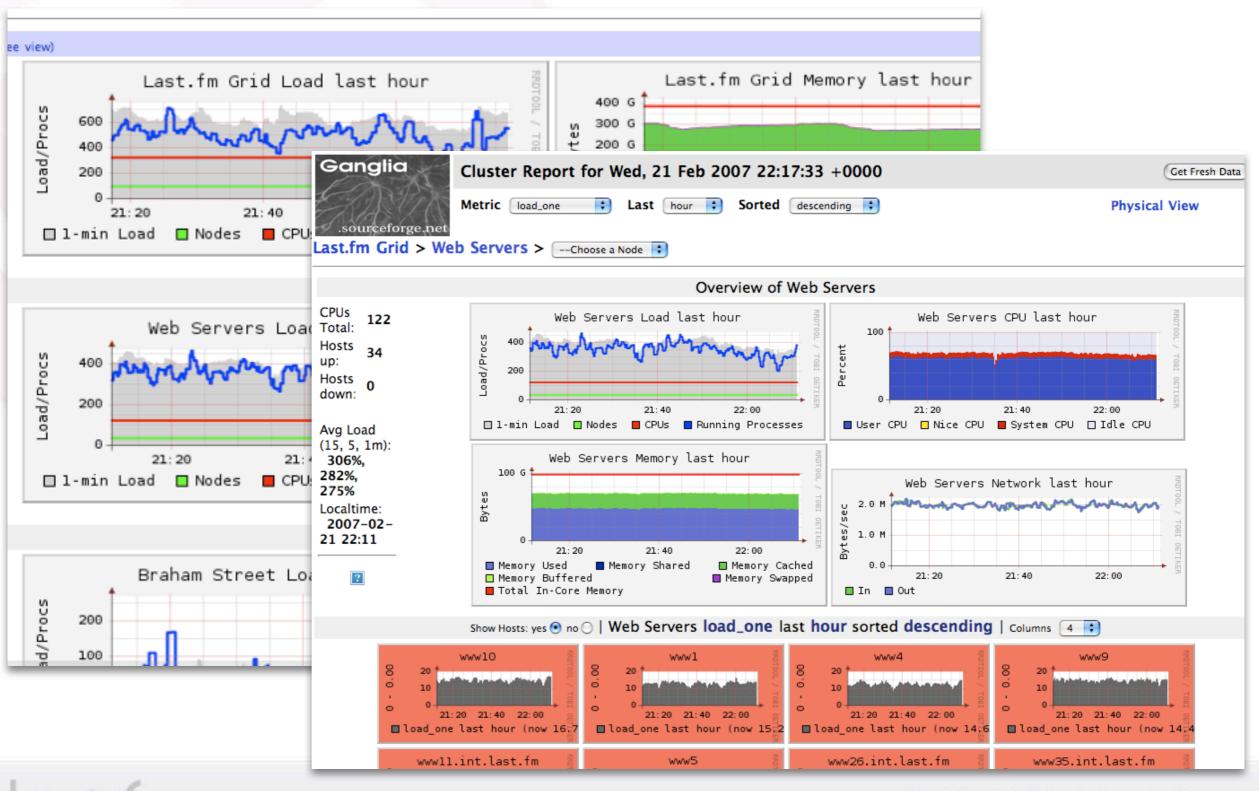


Putting it all together - profiling

- Profile your database and memcache requests as you develop it's usually easier to optimize as you work on new feature
- Last.fm's development sites include a profiling footer, and use *_Profiling classes which extend our DBManager and Memcache frameworks.
- Some things are harder to profile: internal HTTP requests, Javascript. What's the best profiling you've seen?



Monitoring - Ganglia



Characteria Control Co

Monitoring - Nagios

Nagios

General

Home
Documentation

Monitoring

- Tactical Overview
 Service Detail
 Host Detail
 Hostgroup Overview
 Hostgroup Summary
 Hostgroup Grid
 Servicegroup Overview
 Servicegroup Grid
 Servicegroup Grid
 Status Map
- 3-D Status Map

Service Problems
 Host Problems
 Network Outages

Show Host:

Comments
Downtime

Process Info
 Performance Info
 Scheduling Queue

lost.fm

Reporting

Current Network Status

Last Updated: Wed Feb 21 22:20:24 UTC 2007

Updated every 90 seconds Nagios® - <u>www.nagios.org</u> Logged in as *matt*

View Service Status Detail For All Host Groups

View Status Overview For All Host Groups View Status Summary For All Host Groups View Status Grid For All Host Groups



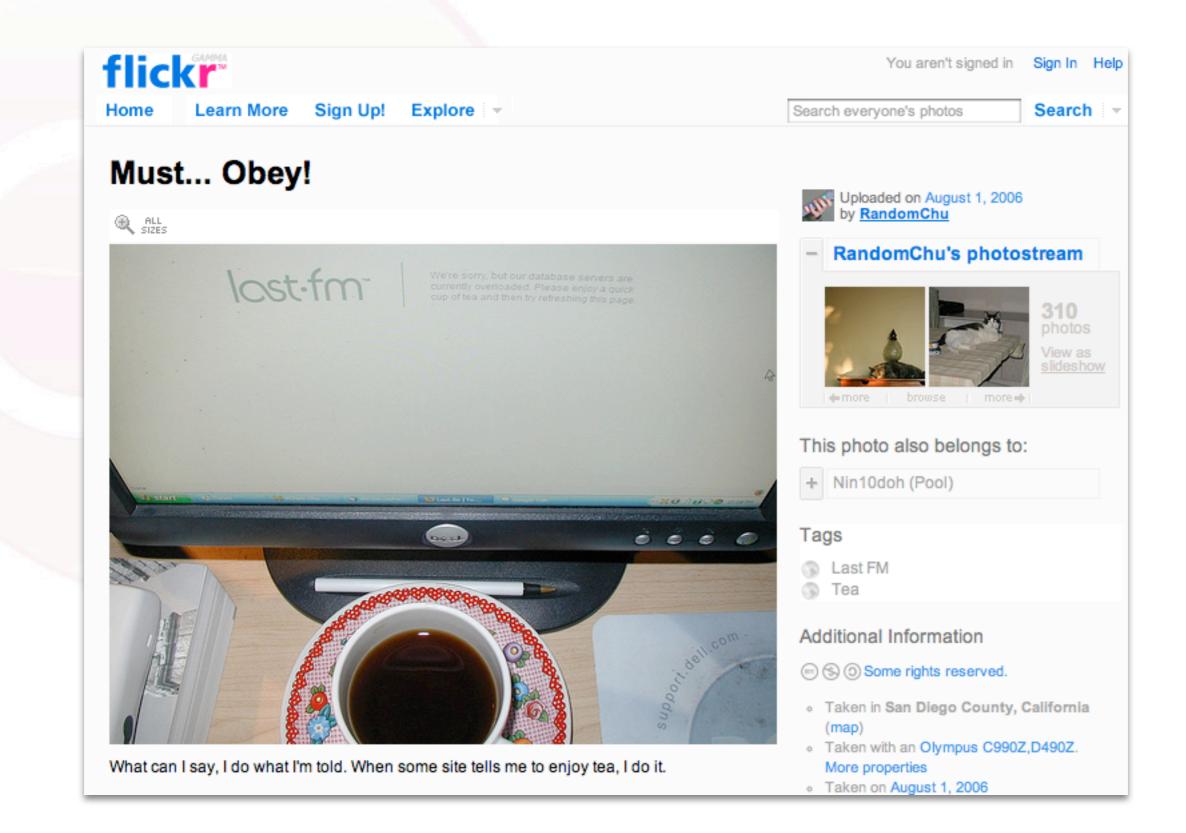
Host Status Details For All Host Groups

y	Host ∱∲	Status ↑√	Last Check 🐴	Duration ↑∲	Status Information
	arwen 🚯	UP	2007-02-15 21:36:42	49d 3h 26m 46s	PING OK - Packet loss = 0%, RTA = 0.17 ms
	<u>charts1</u>	UP	2007-02-15 21:36:42	49d 3h 26m 46s	PING OK - Packet loss = 0%, RTA = 0.17 ms
	<u>charts2</u>	UP	2007-02-15 21:36:42	21d 11h 5m 10s	PING OK - Packet loss = 0%, RTA = 0.51 ms
	compute1 🚯	UP	2007-02-15 21:36:42	23d 9h 34m 45s	PING OK - Packet loss = 0%, RTA = 0.39 ms
	compute2	UP	2007-02-21 22:18:27	49d 3h 24m 46s	PING OK - Packet loss = 0%, RTA = 0.39 ms
	compute3	UP	2007-02-15 21:36:42	49d 3h 25m 46s	PING OK - Packet loss = 0%, RTA = 0.38 ms
	compute4 🛛 🚯	UP	2007-02-15 21:36:42	49d 3h 24m 46s	PING OK - Packet loss = 0%, RTA = 0.39 ms
	core1 🚯	DOWN	2007-02-21 22:18:33	6d 5h 24m 11s	CRITICAL - Host Unreachable (10.0.0.63)
	dote 💽 🖓 🚱	DOWN	2007-02-21 22:18:33	78d 10h 7m 6s	CRITICAL - Host Unreachable (10.0.2.3)
	<u>db1b</u>	UP	2007-02-15 21:36:42	49d 3h 26m 46s	PING OK - Packet loss = 0%, RTA = 0.17 ms
	db2a 🚯	UP	2007-02-15 21:36:42	49d 3h 26m 46s	PING OK - Packet loss = 0%, RTA = 0.19 ms
	db2b 💼 🕰 🦳 🕵	DOWN	2007-02-21 22:18:32	78d 10h 3m 35s	CRITICAL - Host Unreachable (10.0.3.4)

Growing Social Software

- To recap a few points from yesterday...
- Involve your users in your application's growth story: it helps insulate against growing pains, and it's just kinda nice too
- Don't fear open forums
 - Best way to find out when stuff is broken
 - Best source of new feature ideas
- Find ways of rewarding community leaders
- Make your site's growth a selfish aim for existing users
- Have fun with tone
- Go global...





5. Growing Social Software





🕙 www.last.fm

+4

We're sorry, but our database servers are currently overloaded. Please enjoy a quick cup of tea and then try refreshing this page.

Deutsch Es tut uns sehr Leid, aber unsere Datenbankserver sind gerade überlastet. Habe bitte ein wenig Geduld, trinke eine Tasse Tee mit uns und versuche dann, die Seite erneut zu laden.

Español Perdón, lamentamos que nuestros servidores de base de datos esten sobre cargados en estos momentos. Por favor, vuelve a intentarlo dentro de unos minutillos.

Français Nous sommes désolés, les serveurs qui hébergent notre base de données sont actuellement en surcharge. Servez-vous une petite tasse de thé puis essayez de rafraîchir cette page.

Italiano Siamo spiacenti, ma i nostri server sono attualmente intasati. Riprova fra qualche minuto, giusto il tempo di gustarti una tazzina di caffè!

Polski Przepraszamy, ale nasza baza danych jest w tej chwili przeciążona. Proponujemy poświęcić ten czas na zaparzenie herbaty, po czym usiąść wygodnie przed monitorem i odświeżyć stronę. Português Desculpe-nos, mas os nosso bancos de dados estão sobrecarregados no momento. Por que você não toma um cafezinho e tenta atualizar a página depois?

^ Q-

Русский Извини, сейчас сервер перегружен. Пропусти пока стаканчик чаю, после чего попробуй обновить страницу.

日本語 現在、トラフィック急増のためサーバーが混み合っています。 ご 迷惑をおかけして申し訳ありませんが、復旧まで今しばらくお待ちくだ さい。 ご理解ご協力の程、何卒よろしくお願いいたします。

한국어 죄송합니다. 서버 접속량이 많아 정비중입니다. 잠깐 차 한잔 즐기 신 후 다시 시도해 주세요.

中文(简体)对不起,我们的数据库服务器超载。请稍等片刻再刷新页面。

5. Growing Social Software

lost.fm

The Social Music Revolution @ Last.fm 2007. For internal use only.

Going Viral

- Take your web application's most compelling content or feature, and make it exportable.
- Or, as Fred Wilson suggests...
 - 1 Microchunk it Reduce the content to its simplest form.
 - 2 Free it Put it out there without walls around it or strings on it.
 - 3 Syndicate it Let anyone take it and run with it.
 - 4 Monetize it Put the monetization and tracking systems into the microchunk.

5. Growing Social Software



Last.fm Viral Experiments

• Stop, demo-time.

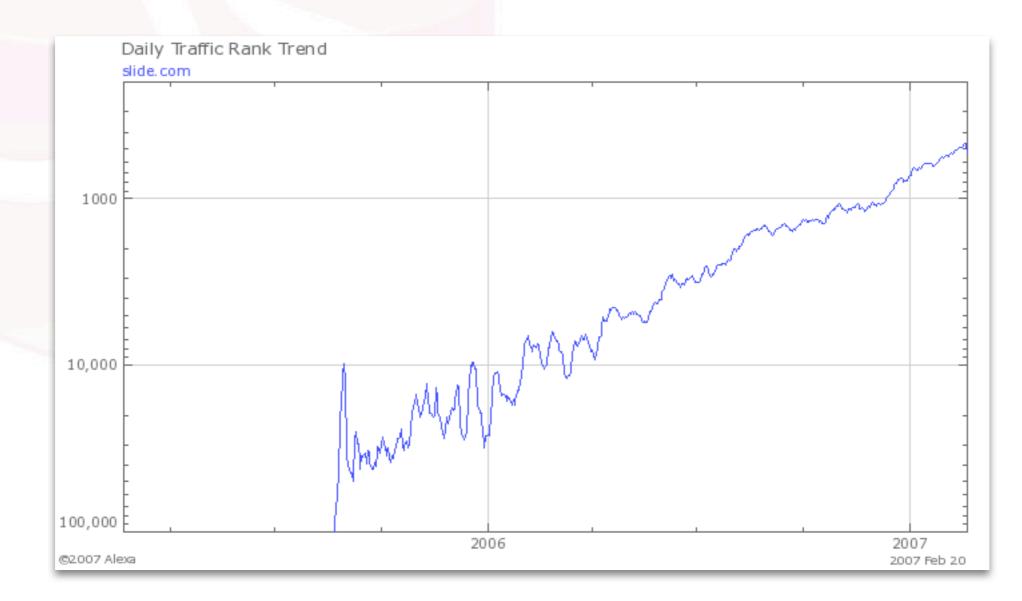
5. Growing Social Software



The Social Music Revolution © Last.fm 2007. For internal use only.

The Reigning Champs

 Slide.com Demo (via <u>http://profile.myspace.com/index.cfm?</u> <u>fuseaction=user.viewprofile&friendid=131137589</u>)



5. Growing Social Software



Characteria Control Co

Growing Userbases

- As your site grows, your userbase will change
- Khoi Vinh yesterday:
 - "Most users are intermediates, but most features are designed for experts."
 - "It's better to piss of the experts than the beginners."
- Hard core scrobblers vs "Sellout Last.fm"
- This challenge can inspire design + interface innovation

5. Growing Social Software



Open mic!





The Social Music Revolution © Last.fm 2007. For internal use only.

This has been fun.

- Stay in touch drop me a line at: <u>matt@last.fm</u>
- By Friday evening a PDF version of this presentation will be available online at: <u>http://static.last.fm/matt/fowa/workshop.pdf</u>
- Come work with us!
 <u>http://www.last.fm/about/jobs/</u>

