Děkuji za pozornost



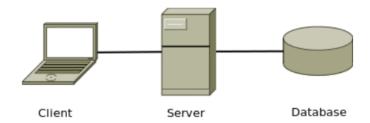




3 vrstvy architektury

- 3-vrstvá architektura je nedostačující
- Každá služba jako další vrsva
 - Vyhledávání
 - Cache
 - NoSQL
 - CDN

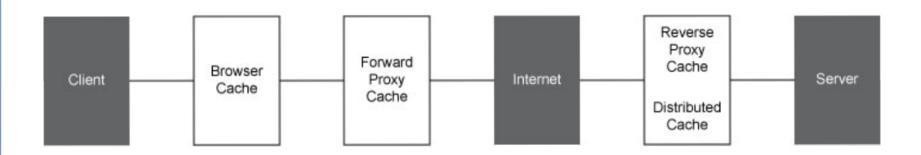
• ...





Druhy cache

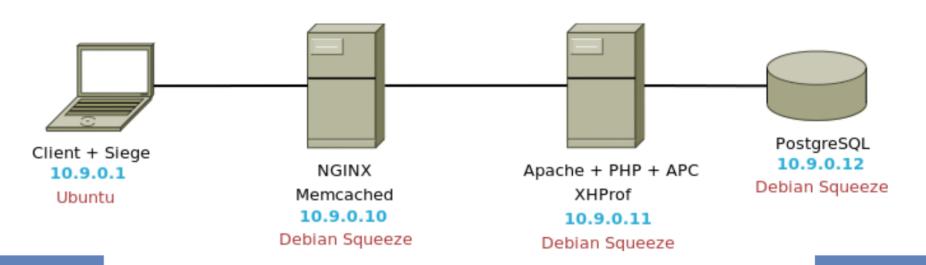
- Statický vs dynamický obsah
- HTTP hlavičky
- Velikost cache dat co nejmenší
- Jmenný prostor konkrétní a jednoznačný
- Znovuzkonstruovatelná data
- Invalidace a cache kaskády





Aplicace a nástroje

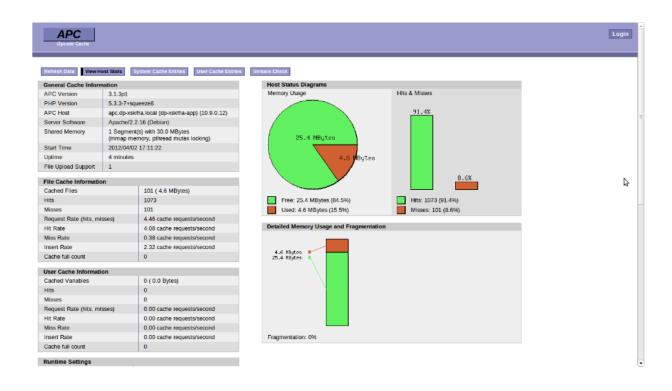
- NGINX reverzní proxy cache
- Apache + PHP + APC webový server s opcode cache
- Memcached aplikační cache
- PostgreSQL
- XHProf profilování aplikace
- Siege testování vysokého zatížení





Aplikační vrstva - PHP + APC

- PHP je interpretovaný jazyk, zkompilován do Opcode (mezikód), který se následně vykoná.
- Proces je možné urychlit pomocí APC Opcode Cache.



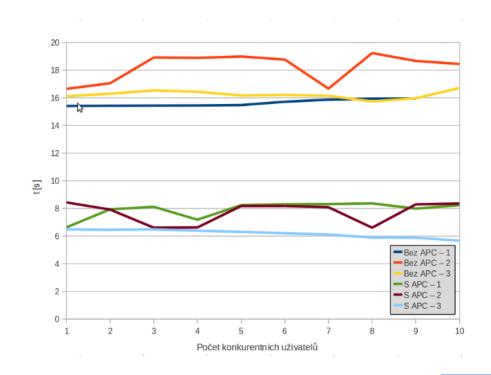


APC - testy a výsledky

- Aplikace "Hello world" v PHP Symfony 2
- Pomocí Siege simulováno vytížení serveru:
 - 10 konkurenčních uživatelů stejný okamžik 3 repetice
 - 10x Apache vlákno s PHP
 - 100% vytížení serveru aplikace

XHProf

- \cdot 1x bez APC = 1,3s
- 1x s APC = 0,407s



Databázová optimalizace

- · Problém:
 - Databázové záznamy v řádech statisíců a miliónů
 - Žádná indexace
 - Opakující se SQL dotazy
 - · Místo jednoho dotazu, několik doplňujících dotazů
- Řešení:
 - Indexace na místech pro odstranění sekvenčího skenu
 - Lepší konstrukce dotazů
- Možná další řešení:
 - Replikace databáze
 - Škálování (vertikální, horizontální)



DB - testy a výsledky 1

Indexace pro odstranění sekvenčního skenu

SELECT * FROM dp_comment WHERE article_id = 15 ORDER BY create_time DESC;

"Sort (cost=121201.07..121201.08 rows=3 width=540) (actual time=16300.440..16300.444 rows=3 loops=1)"

[&]quot;Total runtime: 0.303 ms"

SQL před o	•									
les 43 d										
	op or i	r mily me	(January 1981)	Ches.C.						
Service Service	-	can.	Ded Wall Time microsco	DAMES.	Ind.CRU microsop	Kps.	Just Months Openio	(Western	Lat. Dod Viralia Provi	Designation
Carrel Academ							99.00			
reference another		25400	25 558, 654		185005	22.7%	324,364	154%	5,5,000	10.0
Participation for the Making and the Participation of the Control Particip	9		980	15%	E	15%	40,200	1.94	4504	13.9
Sunda of January and Sunda Joseph Countries of Sunda of Sunda Sund		2.144	21.100.710	1 mm	111.000	19.65	19864	\$5.04	V1348	FA 9
Tyle Tempo polantes		27.00	2400		4.88	AD	44724	8/25	8000	1.0
Official	-			-					2.110	
Opiddae neutrickendelverlebromie gromiegs evengetomiegs, van	1	0.00	2000200	1979	4.88	ATT	14,700	204	7260	1.4
Date is the Primer Resulted Michigan No. Control of the Control of		6.8%	12,048,700	20.00	81,000	ANTEL	107,000	WITE	449,140	MAY
Opelche meterdierodel verbloure betreit wezer getand storre		5. do	5,000,012			445	I MID:	234	18/44	×4
Dp. A draft meCo In Article Order in Administration Services of Landermonth and Landermonth		6.4%	190,000			5.9%	35,000	6.7%	21,700	42
By six fact with safety address in Category Category Insure age Characters		E. Ser	EQU:			LONG	46,00	1,94	100100	7.5
Dp. sk deaf media in de Model Make in Geograph Belly Congrupting Citic	21		- 25		- (3,8%	644	01%	274	0.1
Open helt mich i delt steffens soll hielt delt delt hiele bilde steffens	3			1570		15%	841	1.14	924	11.2
Dp-A-haifmeiter dh-Acadd Naide Arabb Endy Andderge Namber than h-Arabban Arabbahan NA Ababbahan Arabbahan	10	344%	96			3.0%	2.00	01%	(A	030
Dp://dwiftee@cndbis/cddi/stolds/CargonyEntry/Circoporters/El		1100	- 14			AUTO	140	Vis	60	941
The first and a field addition to Control and American	11		- 2			18	841	019	900	01
posicias neutrobaceldos ideoradas provendos e d		2.60	50			AID	540	Visi	314	941
Deck and rectal delication and electricity and electricity	15		85			37%	640	0.1%	44	0.0
		3.60		1175		1.070	844	916	414	10.23
Dpolelas naturalis addivisõis valais arapenatar galantis aa	L									





[&]quot; Sort Key: create_time"

[&]quot; Sort Method: quicksort Memory: 20kB"

[&]quot; -> Seq Scan on dp_comment (cost=0.00..121201.05 rows=3 width=540) (actual time=13.531..16300.393 rows=3 loops=1)"

Filter: (article id = 15)"

[&]quot;Total runtime: 16300.526 ms"

[&]quot;Sort (cost=8.47..8.48 rows=3 width=540) (actual time=0.119..0.134 rows=4 loops=1)"

[&]quot; Sort Key: create_time"

[&]quot; Sort Method: quicksort Memory: 21kB"

[&]quot; -> Index Scan using dp_comment_article_id_fk_i on dp_comment (cost=0.00..8.45 rows=3 width=540) (actual time=0.043..0.068 rows=4 loops=1)"

Index Cond: (article_id = 1441)"

SQL před optimalizací

Parent/Child report for call_user_func_array@1 [View Callgraph]

Function Name	Calls	Calls%	Incl. Wall Time	IWall%	Incl. CPU	ICpu%	Incl. MemUse	IMemUse%	Incl. PeakMemUse	IPeakMemUse%
Turcuon France	Cans	Cans	(microsec)	1 VV dii 70	(microsecs)	гери л	(bytes)	IIVICIIIO SC 70	(bytes)	ir cakivicinose /o
Current Function										
call user func array@1	209	49.4%	35,558,194	97.1%	116,005	22.3%	524,384	13.4%	519,596	13.0%
Exclusive Metrics for Current Function			1,839	0.0%	0	0.0%	50,260	9.6%	4,888	0.9%
Parent functions										
Symfony\Component\DependencyInjection\ContainerBuilder::createService	5	2.4%	35,555,786	100.0%	112,005	96.6%	468,648	89.4%	511,588	98.5%
Twig Template::getAttribute	204	97.6%	2,408	0.0%	4,000	3.4%	55,736	10.6%	8,008	1.5%
Child functions										
Dpxskrha\FrontBundle\Model\Models\Comment\CommentService::getCommentsCount	1	0.5%	15,898,330	44.7%	4,000	3.4%	19,780	3.8%	7,976	1.5%
Dpxskrha\FrontBundle\Model\Models\Article\ArticleService::getArticlesNew	1	0.5%	14,046,220	39.5%	80,005	69.0%	357,188	68.1%	448,548	86.3%
Dpxskrha\FrontBundle\Model\Models\Article\ArticleService::getArticlesCount	1	0.5%	5,397,372	15.2%	4,000	3.4%	19,816	3.8%	18,912	3.6%
Dpxskrha\FrontBundle\Model\Models\Member\MemberService::getMembersCount	1	0.5%	186,680	0.5%	8,000	6.9%	35,120	6.7%	21,700	4.2%
Dpxskrha\FrontBundle\Model\Models\Category\CategoryService::getAllCategories	1	0.5%	27,077	0.1%	16,000	13.8%	34,420	6.6%	13,804	2.7%
Dpxskrha\FrontBundle\Model\Models\Category\Entity\Category::getTitle	31	14.8%	96	0.0%	0	0.0%	644	0.1%	284	0.1%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getCategory	30	14.4%	94	0.0%	0	0.0%	644	0.1%	644	0.1%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getMember	30	14.4%	91	0.0%	0	0.0%	640	0.1%	64	0.0%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getImageUrl	15	7.2%	81	0.0%	0	0.0%	2,024	0.4%	504	0.1%
Dpxskrha\FrontBundle\Model\Models\Category\Entity\Category::getUrl	23	11.0%	72	0.0%	0	0.0%	640	0.1%	640	0.1%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getContent	15	7.2%	52	0.0%	0	0.0%	644	0.1%	588	0.1%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getUrl	15	7.2%	50	0.0%	0	0.0%	640	0.1%	312	0.1%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getTitle	15	7.2%	50	0.0%	0	0.0%	640	0.1%	64	0.0%
Dpxskrha\FrontBundle\Model\Models\Member\Entity\Member::getFirstName	15	7.2%	46	0.0%	0	0.0%	644	0.1%	412	0.1%
Dpxskrha\FrontBundle\Model\Models\Member\Entity\Member::getLastName	15	7.2%	44	0.0%	4,000	3.4%	640	0.1%	256	0.0%



SQL po optimalizaci

-0

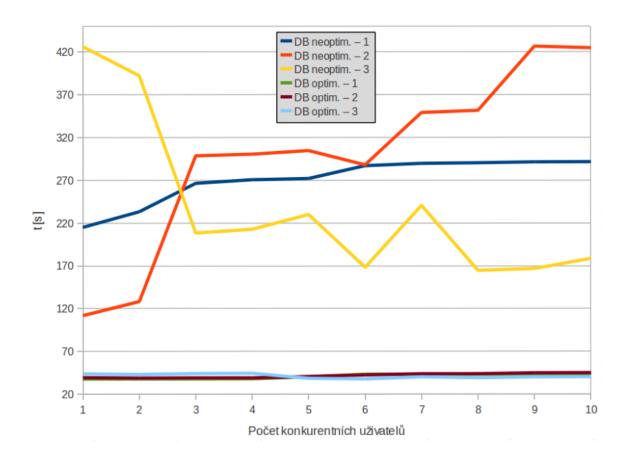
Parent/Child report for call_user_func_array@1 [View Callgraph]

Function Name	<u>Calls</u>	Calls%	Incl. Wall Time (microsec)	IWall%	Incl. CPU (microsecs)	ICpu%	Incl. MemUse (bytes)	IMemUse%
Current Function								
call user func array@1	209	49.4%	24,632,393	84.0%	160,008	7.2%	622,576	8.7%
Exclusive Metrics for Current Function			1,985	0.0%	0	0.0%	50,488	8.1%
Parent functions								
Symfony\Component\DependencyInjection\ContainerBuilder::createService	5	2.4%	24,629,750	100.0%	160,008	100.0%	566,656	91.0%
Twig Template::getAttribute	204	97.6%	2,643	0.0%	0	0.0%	55,920	9.0%
Child functions								
Dpxskrha\FrontBundle\Model\Model\Comment\CommentService::getCommentsCount	1	0.5%	18,444,038	74.9%	52,003	32.5%	99,988	16.1%
Dpxskrha\FrontBundle\Model\Models\Article\ArticleService::getArticlesCount	1	0.5%	5,708,528	23.2%	8,000	5.0%	20,416	3.3%
Dpxskrha\FrontBundle\Model\Models\Article\ArticleService::getArticlesNew	1	0.5%	262,783	1.1%	92,005	57.5%	374,012	60.1%
Dpxskrha\FrontBundle\Model\Model\Member\MemberService::getMembersCount	1	0.5%	194,205	0.8%	4,000	2.5%	36,220	5.8%
Dpxskrha\FrontBundle\Model\Models\Category\CategoryService::getAllCategories	1	0.5%	20,080	0.1%	4,000	2.5%	33,652	5.4%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getTitle	15	7.2%	116	0.0%	0	0.0%	640	0.1%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getMember	30	14.4%	113	0.0%	0	0.0%	640	0.1%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getCategory	30	14.4%	101	0.0%	0	0.0%	644	0.1%
Dpxskrha\FrontBundle\Model\Models\Category\Entity\Category::getTitle	31	14.8%	92	0.0%	0	0.0%	644	0.1%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getImageUrl	15	7.2%	85	0.0%	0	0.0%	2,024	0.3%
Dpxskrha\FrontBundle\Model\Models\Category\Entity\Category::getUrl	23	11.0%	74	0.0%	0	0.0%	640	0.1%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getUrl	15	7.2%	56	0.0%	0	0.0%	640	0.1%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getContent	15	7.2%	48	0.0%	0	0.0%	644	0.1%
Dpxskrha\FrontBundle\Model\Models\Member\Entity\Member::getFirstName	15	7.2%	47	0.0%	0	0.0%	644	0.1%
$Dpxskrha \label{lem:lember:getLastName} Dpxskrha \label{lember:getLastName} In the lember \label{lember:getName} In th$	15	7.2%	42	0.0%	0	0.0%	640	0.1%



DB - testy a výsedky 2

- · Vygenerována zátěž na hlavní stránku aplikace pomocí Siege
- 10 konkurentních uživatelů 1 okamžik 3 repetice





Aplikační cache - Memcached

- Práce s modelovou vrstvou MVC vzoru
- Ukládaní modelů (výsledků SQL) do Memcached
- Model obsahuje vrstvy Service-DAO-Mapper-Storage
- Tři druhy SQL dotazů pro ukládání do cache:
 - Skalár, Entita, Kolekce
- Invalidační kaskády
 - Stejná data v různých interpretacích
 - Řešené přes Dependency Injection kontejner
- Možnost určení vlastní akce při invalidaci



Aplikační cache - Class Diagram

// @todo

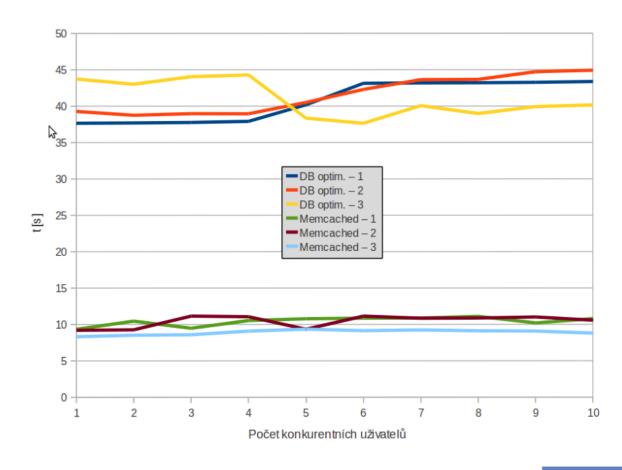


Aplikační cache - testy a výsledky 1

• Siege - Homepage - 10 uživatelů - 1 okamžik - 3 repetice

XHProf pro 1 požadavek:

- Optim. DB = 25,89s
- Memcached = 0,95s

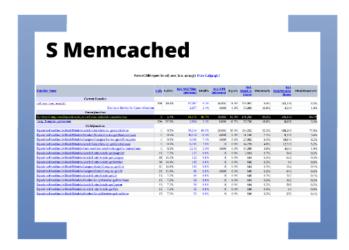




Aplikační cache - testy a výsledky 2

Srovnání výsledků Modelu pomocí XHProf







Bez Memcached

Parent/Child report for call_user_func_array@1 [View Callgraph]

Function Name	Calls	Calls%	Incl. Wall Time	IWall%	Incl. CPU	ICpu%	Incl. MemUse	IMemUse%	<u>Incl.</u> PeakMemUse	IPeakMemUse%
Tunction France	Cuis	Cuiis 70	(microsec)	11111170	(microsecs)	Торило	(bytes)	Intellio Se 70	(bytes)	II cuntivicino se 70
Current Function										
call user func array@1	209	49.4%	24,632,393	84.0%	160,008	7.2%	622,576	8.7%	431,748	5.9%
Exclusive Metrics for Current Function			1,985	0.0%	0	0.0%	50,488	8.1%	3,776	0.9%
Parent functions										
Symfony\Component\DependencyInjection\ContainerBuilder::createService	5	2.4%	24,629,750	100.0%	160,008	100.0%	566,656	91.0%	425,884	98.6%
Twig_Template::getAttribute	204	97.6%	2,643	0.0%	0	0.0%	55,920	9.0%	5,864	1.4%
Child functions										
Dpxskrha\FrontBundle\Model\Models\Comment\CommentService::getCommentsCount	1	0.5%	18,444,038	74.9%	52,003	32.5%	99,988	16.1%	88,676	20.5%
Dpxskrha\FrontBundle\Model\Models\Article\ArticleService::getArticlesCount	1	0.5%	5,708,528	23.2%	8,000	5.0%	20,416	3.3%	19,432	4.5%
Dpxskrha\FrontBundle\Model\Models\Article\ArticleService::getArticlesNew	1	0.5%	262,783	1.1%	92,005	57.5%	374,012	60.1%	270,412	62.6%
Dpxskrha\FrontBundle\Model\Models\Member\MemberService::getMembersCount	1	0.5%	194,205	0.8%	4,000	2.5%	36,220	5.8%	22,580	5.2%
Dpxskrha\FrontBundle\Model\Models\Category\CategoryService::getAllCategories	1	0.5%	20,080	0.1%	4,000	2.5%	33,652	5.4%	24,224	5.6%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getTitle	15	7.2%	116	0.0%	0	0.0%	640	0.1%	64	0.0%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getMember	30	14.4%	113	0.0%	0	0.0%	640	0.1%	0	0.0%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getCategory	30	14.4%	101	0.0%	0	0.0%	644	0.1%	0	0.0%
Dpxskrha\FrontBundle\Model\Models\Category\Entity\Category::getTitle	31	14.8%	92	0.0%	0	0.0%	644	0.1%	284	0.1%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getImageUrl	15	7.2%	85	0.0%	0	0.0%	2,024	0.3%	504	0.1%
<pre>Dpxskrha\FrontBundle\Model\Models\Category\Entity\Category::getUrl</pre>	23	11.0%	74	0.0%	0	0.0%	640	0.1%	640	0.1%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getUrl	15	7.2%	56	0.0%	0	0.0%	640	0.1%	0	0.0%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getContent	15	7.2%	48	0.0%	0	0.0%	644	0.1%	588	0.1%
Dpxskrha\FrontBundle\Model\Models\Member\Entity\Member::getFirstName	15	7.2%	47	0.0%	0	0.0%	644	0.1%	312	0.1%
Dpxskrha\FrontBundle\Model\Models\Member\Entity\Member::getLastName	15	7.2%	42	0.0%	0	0.0%	640	0.1%	256	0.1%



S Memcached

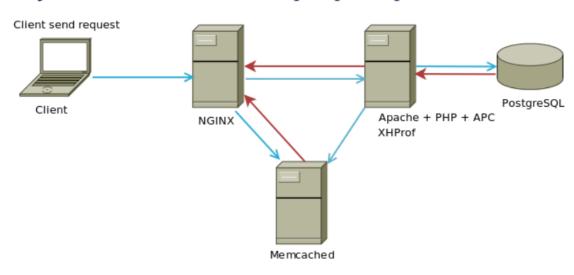
Parent/Child report for call_user_func_array@1 [View Callgraph]

Function Name	Calls	Calls%	Incl. Wall Time	IWall%	Incl. CPU	ICpu%	Incl. MemUse	IMemUse%	Incl. PeakMemUse	IPeakMemUse%
			(microsec)		(microsecs)	•	(bytes)		(bytes)	
Current Function										
call user func array@1	209	49.4%	87,007	9.2%	48,002	11.0%	296,992	8.0%	243,108	6.5%
Exclusive Metrics for Current Function			2,077	2.4%	4,000	8.3%	50,260	16.9%	4,584	1.9%
Parent functions										
$Symfony \label{local:component} \label{local:component} Symfony \label{local:component} \label{local:component} Local \label{local:component} Symfony \label{local:component} \label{local:component} Local \label{local:component} \label{local:component} Symfony \label{local:component} \label{local:component} \label{local:component} Symfony \label{local:component} \label{local:component} Symfony \label{local:component} Symfony \label{local:component} \label{local:component} Symfony \label{local:component} \label{local:component} Symfony $	5	2.4%	84,173	96.7%	40,002	83.3%	241,256	81.2%	235,100	96.7%
Twig Template::getAttribute	204	97.6%	2,834	3.3%	8,000	16.7%	55,736	18.8%	8,008	3.3%
Child functions										
Dpxskrha\FrontBundle\Model\Models\Article\ArticleService::getArticlesNew	1	0.5%	56,214	64.6%	28,002	58.3%	164,252	55.3%	189,384	77.9%
Dpxskrha\FrontBundle\Model\Models\Member\MemberService::getMembersCount	1	0.5%	10,114	11.6%	4,000	8.3%	21,168	7.1%	8,176	3.4%
Dpxskrha\FrontBundle\Model\Models\Category\CategoryService::getAllCategories	1	0.5%	6,436	7.4%	4,000	8.3%	25,092	8.4%	19,840	8.2%
Dpxskrha\FrontBundle\Model\Models\Article\ArticleService::getArticlesCount	1	0.5%	6,181	7.1%	0	0.0%	14,220	4.8%	12,720	5.2%
Dpxskrha\FrontBundle\Model\Models\Comment\CommentService::getCommentsCount	1	0.5%	5,143	5.9%	4,000	8.3%	14,200	4.8%	4,636	1.9%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getImageUrl	15	7.2%	127	0.1%	0	0.0%	2,024	0.7%	504	0.2%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getCategory	30	14.4%	122	0.1%	0	0.0%	644	0.2%	644	0.3%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getMember	30	14.4%	109	0.1%	0	0.0%	640	0.2%	64	0.0%
Dpxskrha\FrontBundle\Model\Models\Category\Entity\Category::getTitle	31	14.8%	103	0.1%	0	0.0%	644	0.2%	284	0.1%
Dpxskrha\FrontBundle\Model\Models\Category\Entity\Category::getUrl	23	11.0%	86	0.1%	4,000	8.3%	640	0.2%	640	0.3%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getUrl	15	7.2%	69	0.1%	0	0.0%	640	0.2%	312	0.1%
Dpxskrha\FrontBundle\Model\Models\Member\Entity\Member::getFirstName	15	7.2%	59	0.1%	0	0.0%	644	0.2%	412	0.2%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getContent	15	7.2%	58	0.1%	0	0.0%	644	0.2%	588	0.2%
Dpxskrha\FrontBundle\Model\Models\Article\Entity\Article::getTitle	15	7.2%	56	0.1%	0	0.0%	640	0.2%	64	0.0%
Dpxskrha\FrontBundle\Model\Models\Member\Entity\Member::getLastName	15	7.2%	53	0.1%	0	0.0%	640	0.2%	256	0.1%



Reverzní proxy cache - NGINX

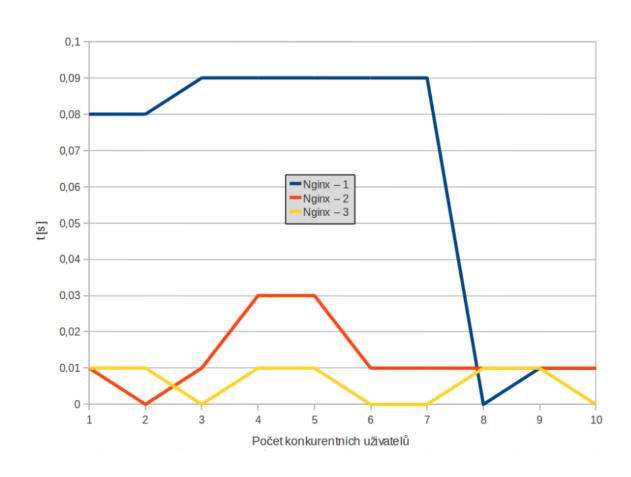
- · Cache pro statický obsah (obrázky, css, js, HTML, apod.).
- Statický obsah je ukládán do cache souborů NGINX (Mode Reverse Proxy cache).
- Ukládání je řízeno HTTP hlavičkami.
- NGINX může spolupracovat i s Memcached (Mode Memcached).
- Ukládají se výstupy aplikační vrstvy (celé HTTP požadavky).
- Řízeno aplikací.
- Pokud NGINX nalezne požadavek v Memcached, tak jej vrátí přímo z Memcached.
- · Jinak propustí požadavek na aplikační vrstvu.
- Ta jej zpracuje a uloží do memcached pro příští požadavek.





NGINX - výsledky

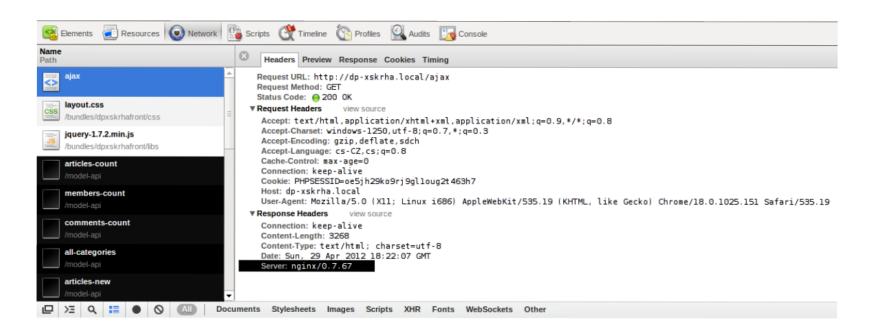
• Siege - 10 uživatelů - 1 okamžik - 3 repetice





NGINX + Ajax

- · NGINX navrátí statickou stránku z cache, která obsahuje požadavky na jiné služby.
- · Odpovědi na tyto požadavky jsou uloženy do cache pro NGINX.
- Odpovědi jsou typu JSON nebo HTML fragmenty.
- · Invalidace může být propojena s aplikační vrstvou.
- Aplikace se tak rozdělí na více podslužeb.





Cache - srovnání

- Aplikační cache:
 - Udržuje aplikaci stále dynamickou +
 - Klade vysoké nároky na aplikační vrstvu (vývoj, aplikační servery) -
 - Složitější invalidace kaskád -
- Reverzní Proxy Cache
 - Rychlá odezva +
 - Nižší náklady a režie +
 - Aplikace se stává více statickou -
 - Nastavení politiky časové invalidace -





Závěr

- Porovnat náklady na vybodování cache s investicí do HW prostředků.
 - Rozhodnutí dynamické, statické či hybridní aplikace.
 - · Určit přesnou politiku tvorby a práce s cache.
 - · Nepodceňovat invalidaci.
 - · Zvážit rozdělení aplikace na více služeb.



Děkuji za pozornost



