

# *"Caching In" on CF Performance*

Charlie Arehart  
([charlie@carehart.org](mailto:charlie@carehart.org))

# Topics

---

- Why worry about caching?
- The top 3 caching features
- Several more caching options
- Caching outside of CF
- Resources for learning more

# Introduction: Caching in CF

---

- Will see more than a dozen ways
  - Can you name more than 3?
  - Some set/controlled in code, others in Admin
- More caching outside of CF
  - Database, web server, browser, etc.
- Do you really understand caching you've used?
  - Often conflicting recommendations
  - Easy to make a mistake, make things worse
- We'll review over a dozen approaches

# Introduction

---

- Presentation, code will be posted
  - I collected and organized much more info than we have time to cover
    - Will skip some details for you to read about later
    - Will try to balance educating beginning/adv devs
  - The details cover each kind of caching
    - Introduce, help you understand purpose
    - Provide info on making choices
    - Discuss gotchas and surprises
    - Help solve problems when things go wrong
    - Show how to manage/flush each cache
    - If available, show how to monitor each cache
- Most info applies to CF5, 6, 7, and 8 (& BD)

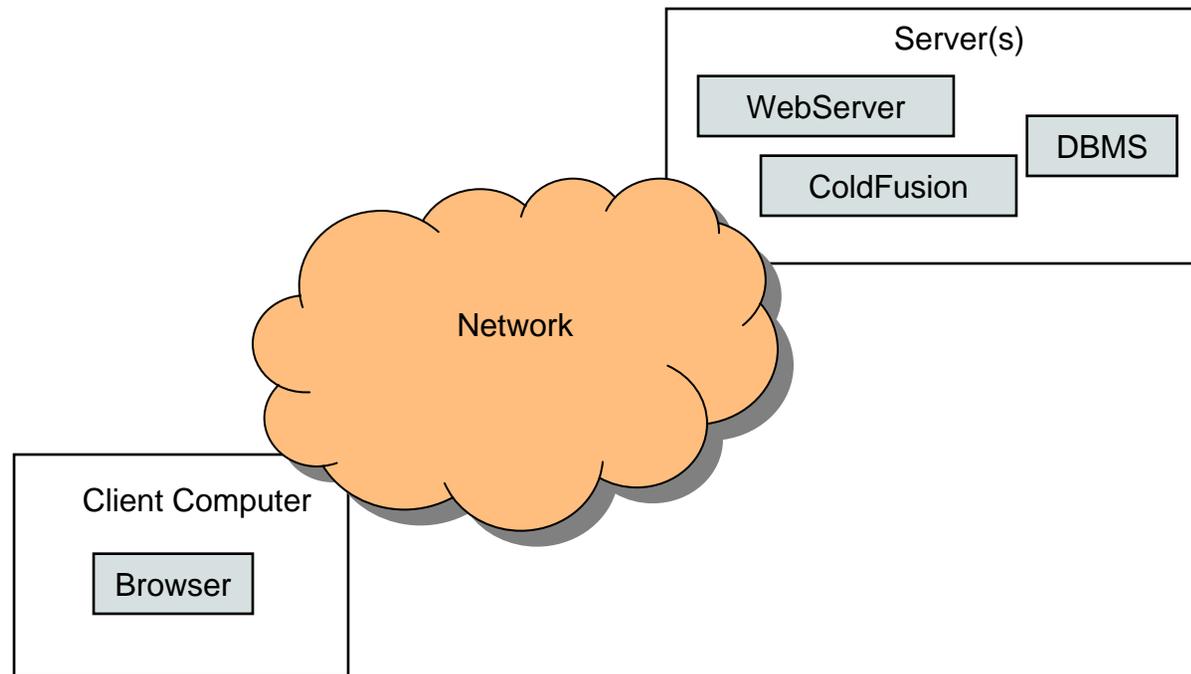
# About Charlie Arehart

---

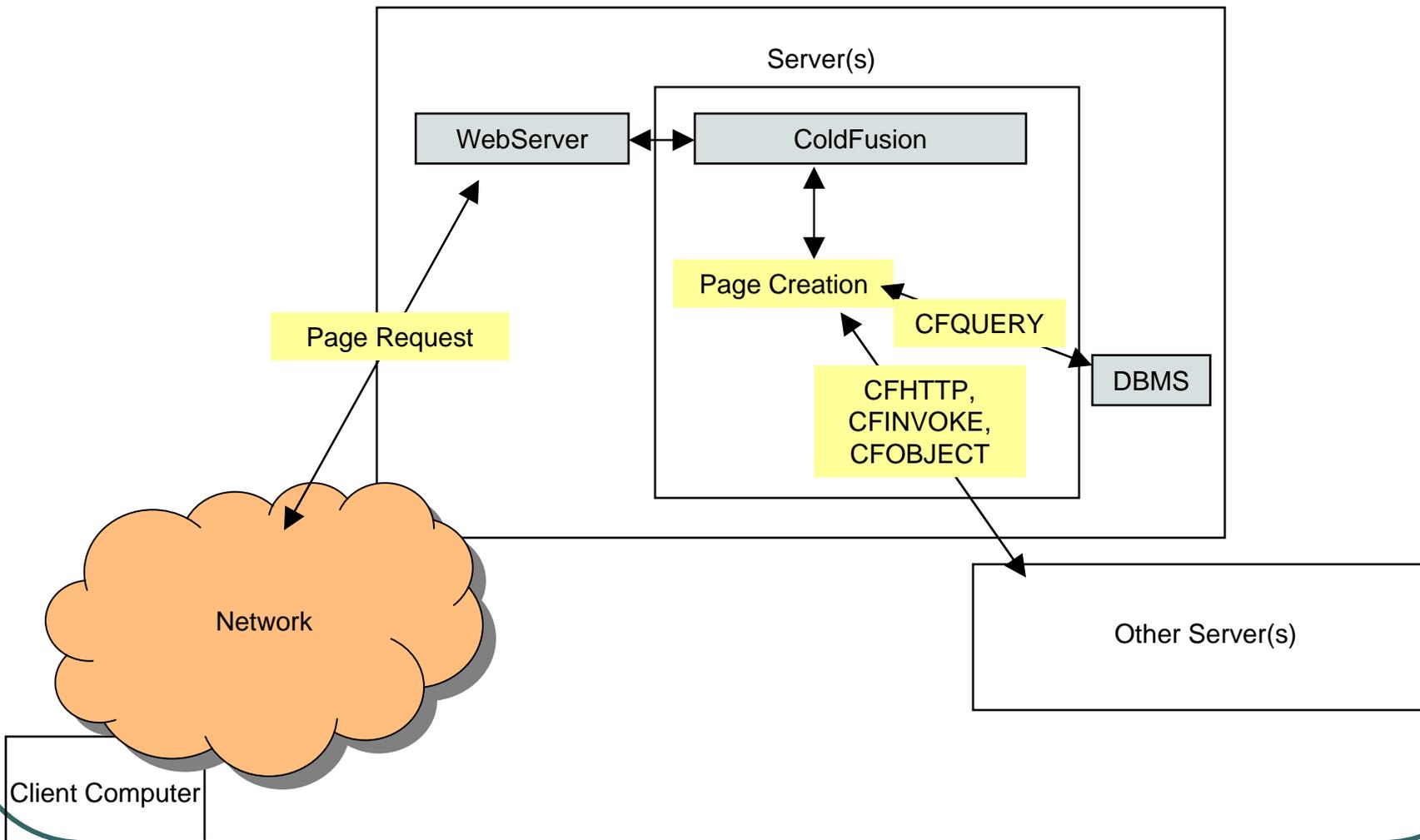
- Independent consultant since April 2006
- 10 yrs CF experience (25 in Enterprise IT)
  - Member, Adobe Community Experts
  - Certified Adv CF Developer (4 - 7), Cert. Adobe Instructor
  - Frequent speaker to user groups, conferences worldwide
  - Contributor to Ben Forta's CF8 books, past *ColdFusion MX Bible*
  - Run the Online ColdFusion Meetup (coldfusionmeetup.com)
  - Living in Alpharetta, Georgia (north of Atlanta)
- Web home at [www.carehart.org](http://www.carehart.org)
  - Hosts 175+ blog entries, 50+ articles, 70+ presentations, more
  - UGTV: recordings of presentations by over 100 CFUG speakers
  - AskCharlie: per-minute telephone & web-based CF support

# Anatomy of a Typical Page Request

---



# Anatomy of a Typical Page Request (cont)



# Why Worry About Caching?

---

- Many operations are expensive
  - Database I/O, page creation, CFC invocation, CFHTTP, web svc calls
  - During development, may not notice a problem
  - Once done hundreds, thousands, or millions of times a day, the cost can become significant
    - Worse, as database size grows, problem can get worse
- Caching can help solve performance problems
  - Caching simply means saving the result of some expensive operation, for later reuse
  - Some CFML operations do it automatically, some must be indicated
    - Others must be implemented manually
  - Yet can't use it blindly, need to understand impact
    - Never using caching is likely also bad
- OO-style vs. traditional development
  - Different caching solutions will appeal to each camp
    - Be patient if you see something you wouldn't do

## Top 3 Caching Features

---

- Query Caching
  - Template Caching
  - Page Caching
- Unlike some talks, these are not the sole focus of this talk

# Top 3 Caching: Query Caching

---

- Basics
  - Database operations are often very expensive
    - Yet frequently the returned data does not change
      - Sometimes never at all, like list of states
      - Or perhaps very infrequently, like list of employees, products
        - Might data be updated only overnight?
        - Or might user be willing to briefly live without absolute latest data?
  - CFQuery's **CachedWithin** & **CachedAfter** attributes solve this
    - Data requested once and saved for subsequent reuse
    - **CachedWithin**: cached if query re-executed within given timeframe
      - Use `#CreateTimeSpan(days, hours, minutes, seconds)#` to indicate duration
    - **CachedAfter**: cached if query executed after a given date/time
      - `#createdatetime(year(now()), month(now()), day(now()), hour, min, sec)#`
      - Don't use *\*just\** the time, or CF assumes date of 12/30/99, so caches immediately
- See resources later for docs, examples

# Top 3 Caching: Query Caching (cont)

---

- Surprises
  - Caching is saved **per unique SQL statement** (demo)
    - So can have use with dynamic queries driven by form, url variables
  - Shared across entire server (demo)
    - Any page using same QueryName, SQL, DSN, uname, pw, etc. benefits from same cached result
    - Again, though, if SQL is different, creates different cached result
  - CachedWithin and CachedAfter share same cache
    - Query cached with one can benefit a query cached with other
  - Consider benefits of micro-caching (very short cache times, like minutes or even seconds)
  - Setting admin “max number of cached queries” to 0 permits just 1
    - Was a time when did not disable, but meant unlimited!
      - [http://www.adobe.com/go/tn\\_19406](http://www.adobe.com/go/tn_19406)
      - Note true in either CFMX 7 or 6.1
        - (at least as of 6,1,0,83762, from my own testing)

# Top 3 Caching: Query Caching (cont)

---

- Managing/Flushing
  - Admin console option to limit max number of cached query results
  - How to flush
    - Set createtimespan to 0 in cache attributes (demo)
      - Consider setting cachetime in variable, to affect all queries of interest
      - But are you flushing all cached SQL variations (like URL-driven)?
      - Good place to consider a CFC holding queries, so you could manage queries
      - Will see alternative approaches that help solve above problem
    - **CFObjectCache** (demo)
      - Clears all cached queries (good for problem above)
      - But clear across entire CF server, all apps, so think twice
      - Does same as servicefactory's purgeQueryCache method in DataSourceService
  - Could also have insert/update/delete code invalidate the cache
    - Good job for a CFC

# Top 3 Caching: Query Caching (cont)

---

- Monitoring
  - Ability to tracking query cache usage is coming in CF8
  - In CF7, can use admin api to access max number of cached queries possible (demo)
    - CFIDE.adminapi.runtime.cfc's `getCacheProperty("MaxCachedQuery")` method
- Gotchas
  - "per unique SQL" aspect even means spaces, line breaks
  - Can't use CFQueryParam in a cached query
  - Can't use with CFSTOREDPROC
    - but SPs can be called with CFQUERY
  - Managing cached results is challenging, as discussed above
  - Be careful about using too much memory
  - Conversely, max setting in admin may be too low for you
  - If you find that expected cached result doesn't "stay cached", beware that someone (outside your app) may be using CFOBJECTCACHE to clear the query cache

# Top 3 Caching: Query Caching (cont)

---

- Alternative: caching queries in variables, discussed later
- BlueDragon adds new features for better cache management
  - CacheName attribute permits giving cached results a name
  - Action attribute permits flushing named or all cached queries, with optional
  - CFObjectCache CacheDomain attribute to flush all on a server
  - BlueDragon 7 adds new cache invalidation feature
  - BD 6 and 7 support CFQUERYPARAM within cached queries
    - Many would look forward to CF adding that
- Resources
  - <http://livedocs.adobe.com/coldfusion/7/htmldocs/00000316.htm#1102316>
  - See Resources later: on Admin settings, articles covering caching options

# Candidates for Query Caching

---

- Any one or more of these would make a query a good candidate for caching
  - Called very frequently
    - Whether in a page, across many pages, or even many users
  - Or returns data that takes long to gather
  - Or returns data that does not change frequently
  - Or has no or a small number of input arguments causing variation in SQL
    - Given the limiter for cached queries
- CachedAfter better suited when data being cached is updated at a set time (like 2am)

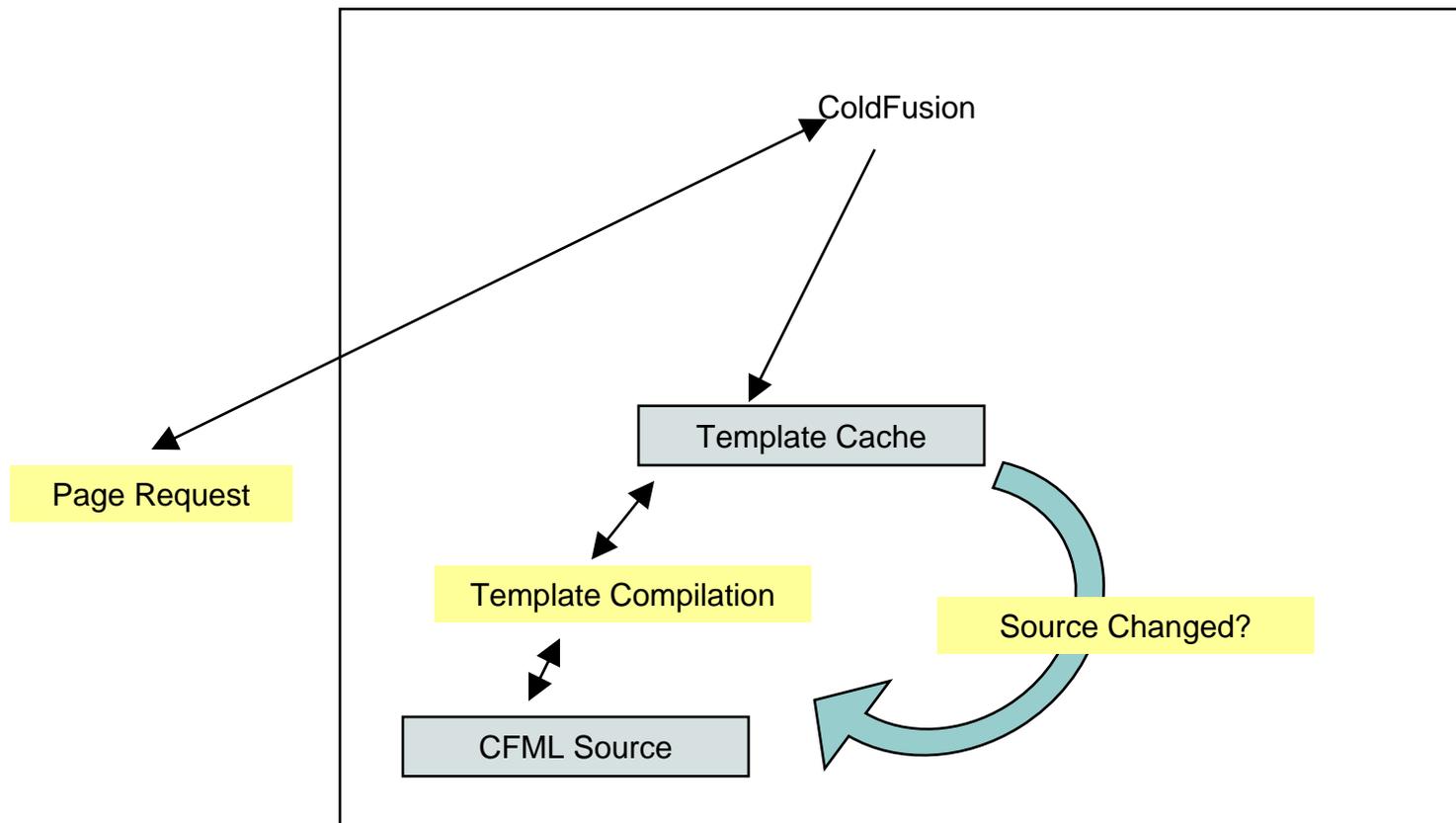
## Top 3 Caching: Template Caching

---

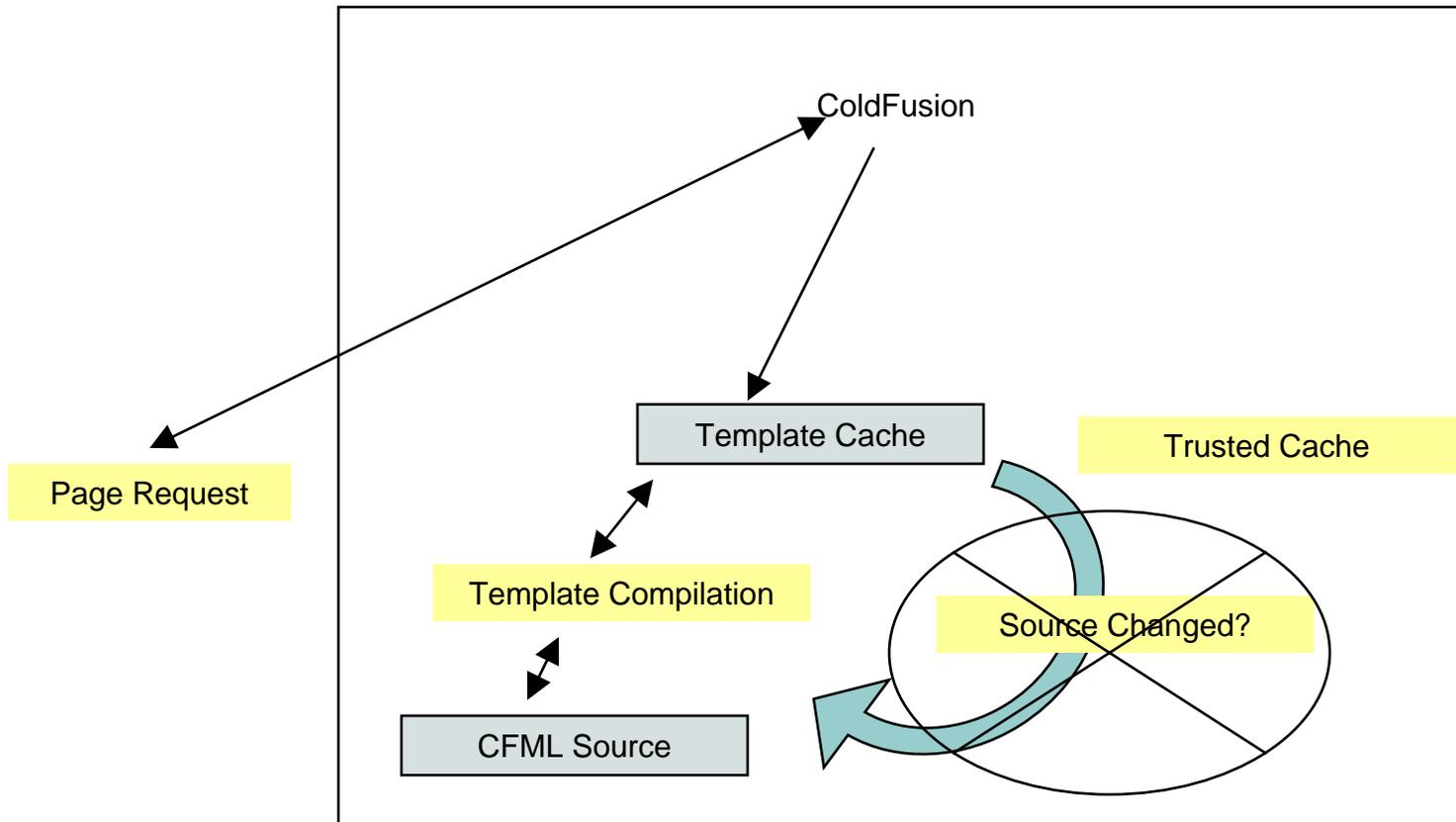
- Let's dig deeper into request process
  - Page request process
  - Trusted cache setting



# Anatomy of a Typical Page Request (cont)



# Anatomy of a Typical Page Request (cont)



# Top 3 Caching: Template Caching

---

- Basics
  - Template size and trusted cache set in Admin
- Surprises
  - Ashwin on admin sizing (blog entry URL's offered later)
- Managing/Flushing
  - Challenge with trusted cache, when updating source code
    - To reload a single page, must run/compile it while trusted cache disabled
    - Cfcompile.bat command line tool in C:\CFusionMX7\bin
  - CF7 Admin new button: flushes entire cache
    - Also cfide.adminapi.runtime.cfc clearTrustedCache() method
    - Both clear the entire cache (all templates)
  - CF7 gateway-based tool to track changes to a directory
    - <http://www.adobe.com/devnet/coldfusion/articles/cacheclear.html>

## Top 3 Caching: Template Caching (cont)

---

- Gotchas
  - Not about caching output of templates!
  - Problem of old files being ftp'ed or extracted from version control (or zip) – if change not detected
    - [http://cfmxplus.blogspot.com/2002\\_11\\_17\\_cfmplus\\_archive.html](http://cfmxplus.blogspot.com/2002_11_17_cfmplus_archive.html)
- Monitoring
  - In CF5, had available “cache pops” stat (cp/sec) in cfstat, perfmon, getmetricdata()
    - CFMX no longer provides that data (there, but -1)
  - Available admin CFC methods, in runtime.cfc
    - Get/setCacheProperty() for TrustedCache, TemplateCacheSize
  - CF8 will add monitoring
    - how many templates in cache, est. size, cache hit ratio
- Resources
  - See Resources later for more on Admin settings, and articles covering multiple caching options

# Top 3 Caching: Page Caching

---

- Basics: CFCache
  - CF saves generated HTML content of a page into a temp file
    - Under covers, that is served to users on later requests
  - Useful when entire page's execution (not just CFQUERY) takes time but results don't change much between frequent repeated requests
  - Demo
- Managing
  - Can control duration of cached result
    - using TimeSpan attribute and CreateTimeSpan() function
    - If none, cached result is "permanent" (until flushed manually)
  - Can choose location of cached result, server or client
    - Now defaults to caching on server and client
    - Storing as ClientCache is key for personalized pages
  - For server cache, can specify Directory attribute to name where to store cached (tmp) files
    - Defaults to cf\_root/cache
  - Supports both HTTP and HTTPS, as well as USERNAME/Password for WS-protected files

# Top 3 Caching: Page Caching (cont)

---

- Surprises
  - Many changes as of CFMX 6
    - Used to cache only on server. 4.5 added cache on client
      - CFMX defaults to doing both (was "optimal" option in 4.5+)
    - Can vary based on user input (URL variables)
      - `http://server/view.cfm?id=1` and `http://server/view.cfm?id=2` are cached separately
    - Can vary based on sessions!
    - No longer uses old `cache.map` file to track cached results stored on server
- Flushing
  - Available `action="flush"`
    - Can point to directory where cache results are stored
  - Available `expireURL` attribute to flush entries cached based on URL
    - Supports use of wildcards
  - The above actually delete the tmp files from the named dir.
  - Cache is refreshed if you change the CFML file itself
    - First load of page after change will really run the page, cache it

# Top 3 Caching: Page Caching (cont)

---

- Gotchas
  - Caches entire page: no support for partial page caching
    - Will discuss alternate solution later
  - While it recognizes URL vars, does **not** recognize different path\_info values
    - Search-engine-safe URLs using this will serve wrong cached result (in other words, as if path\_info ignored)
  - CF puts a CFML comment at top of generated output
    - Makes page non-standard, forces IE to "quirks" mode
  - I couldn't get ClientCache feature to work at all
- Monitoring
  - Since page caching is not "in memory", no need to track that concern
    - But the filesystem used for caching may be important
    - Would do well to monitor directory space (default or as named)
      - Not aware of any auto-cleanup process, though one may well exist
  - No real means to monitor if a page is being cached
    - other than to view that embedded comment mentioned above
- Resources
  - <http://livedocs.adobe.com/coldfusion/7/html/docs/00001126.htm#1172150>

## Several More Caching Options

---

- Caching queries in scopes
- Caching queries to disk
- Partial page caching
- Caching CFCs
- Caching CFHTTP, Web Service calls
- Still more CF caching features
- Perhaps unexpected caching

# More: Caching Queries in Scopes

---

- Basics
  - Did you know you can put query results into a scope other than local/variables?
    - Either in CFQUERY NAME or by assignment
    - Demo
  - If placed in shared scope, remains in memory
    - Can use session, application, or server scopes
  - Before CachedWithin and CachedAfter were added, this was the technique
    - Because they don't permit CFQueryParam, many still use the approach
- Resources
  - Ample coverage in resources at end of presentation
- Surprise
  - Consider this benefit: if DB goes down, app could keep running since it's not talking to it
    - if need nothing but cached data

# More: Caching Queries in Scopes (cont)

---

- Managing
  - If in shared scope, will timeout when scope does
  - Or you can manually manage purge/refresh
    - when time passes or data changes
  - Must also manage “singleton” creation
    - Ensure scope only loaded with data once for desired cache lifetime
  - All the above must be done with code
    - In application.cfm, with code to test for existence
    - Or in Application.cfc onSessionstart, onApplicationstart
    - Or in a CFC that does the work and returns results
- Gotchas
  - You also must manually manage max use of memory
    - No easy way to track query size or total cache use
  - Some coders duplicate shared scopes on each request
    - Application.cfm/OnRequestEnd.cfm
    - Beware that cached queries could balloon that process
  - Also doesn't work well when app spread across cluster
  - Might be better to use disk rather than memory...—→

## More: Caching Queries to Disk

---

- Previous page discusses caching to memory. Could instead cache to disk
- Available CF\_ExtremeQueryCache custom tag does this
- Resource
  - "Caching Queries to Disk or to Memory with ColdFusion"
    - [http://www.adobe.com/devnet/coldfusion/articles/query\\_cache.html](http://www.adobe.com/devnet/coldfusion/articles/query_cache.html)
- Uses wddx serialization
  - Which is something anyone could use for custom caching solution

# More: Partial Page Caching

---

- CFSAVEContent
  - Can save any portion of output generated in a page by wrapping in cfsavecontent
    - Can save to a local or shared scope variable
    - Can then manage that as with query caching in scopes
- Resources
  - <http://livedocs.adobe.com/coldfusion/7/htmldocs/00001127.htm#1173434>
  - See Resources later for articles covering multiple caching options
  - [http://www.adobe.com/devnet/coldfusion/articles/partialcache\\_macrochat.html](http://www.adobe.com/devnet/coldfusion/articles/partialcache_macrochat.html)
  - [http://weblogs.macromedia.com/cantrell/archives/2005/02/partial\\_page\\_ca.cfm](http://weblogs.macromedia.com/cantrell/archives/2005/02/partial_page_ca.cfm)

# More: Partial Page Caching (cont)

---

- Various custom alternatives
  - “Caching options in ColdFusion”
    - <http://ray.camdenfamily.com/index.cfm/2006/7/19/Caching-options-in-ColdFusion>
  - CF\_CacheOMatic
    - <http://www.devx.com/webdev/Article/27618>
  - Brandon Purcell’s CF\_Accelerate
    - <http://www.bpurcell.org/blog/index.cfm?mode=entry&entry=963>
    - Inspired by older CF\_SuperCache
  - Extension to that to support file-based caching
    - [http://www.throwingbeans.org/filebased\\_caching\\_in\\_coldfusion.html](http://www.throwingbeans.org/filebased_caching_in_coldfusion.html)
  - Ray Camden’s ScopeCache
    - <http://ray.camdenfamily.com/downloads/scopecache.zip>
  - Andy Powell’s JohnnyCache
    - <http://johnnycache.riaforge.org/>
  - CF\_TurboCache
    - <http://www.hotfusion.co.uk/TurboCache/index.htm>
  - CF\_HyperCache
    - [http://www.pixl8.co.uk/index.cfm/pcms/site.products.CF\\_Hypercache/](http://www.pixl8.co.uk/index.cfm/pcms/site.products.CF_Hypercache/)
  - CFCacheContent (in BD)
    - Adds option to cache to disk, as well, and more management options

# More: Caching CFCs

---

- Invocation of CFCs can be expensive
- With CFOBJECT, createObject you could save instance in shared scope
  - Again, either using shared scope to hold result, or by copying to it
  - `<cfoject component="cfcname" name="application.myinstance">`
- Gotchas
  - Has same management challenges as scope-cached queries
    - Singleton creation, memory max use, refresh challenges
  - Bigger problem regarding changes to CFC code once cached
    - Will no longer see code changes reflected immediately. Must reload CFC into scope
    - Might code for a cachereset URL var (processed in in app.cfm)
    - Or create a single page that you call as a developer to do reload the cache
  - If CFINVOKE used, no instance name created, so no means to cache instance
- Resources
  - "Caching ColdFusion Components (CFCs) in Shared Memory"
    - <http://coldfusion.sys-con.com/read/45975.htm>
  - Ashwin Matthew's SoftCache CFC
    - <http://blogs.sanmathi.org/ashwin/2006/07/01/memory-sensitive-caching-for-cf/>

## More: Caching CFHTTP, Web Service calls

---

- When you use CFHTTP or invoke a web service, that too can be an expensive, slow operation
  - Yet many forget to consider caching the result
  - Again, can save result in a shared scope
  - Same benefits and challenges as before
- Resource
  - "Cache complex Web service objects with ColdFusion"
    - [http://builder.com.com/5100-6389\\_14-5087907.html](http://builder.com.com/5100-6389_14-5087907.html)

# More: Still More CF Caching Features

---

- CF7 charting (disk and memory)
  - There are built-in, configurable caching options
  - See CF Admin
  - Available monitoring
    - Admin API Runtime.cfc  
getChartProperty("CacheSize") as well as cachetype and cachepath
    - Can also set with setchartproperty
- Verity K2 Collection caching
  - C:\CFusionMX7\verity\k2\common\K2ServerConfigurationMetaData.xml
- DB driver connection pooling
  - "Maintain connections", and more in CF Admin DSN

## More: Perhaps Unexpected Caching

---

- Multiple virtual hosts sharing same code
  - Cacherealpath/"Cache web server paths" option in CF Admin
  - Available get/setCacheProperty() in Runtime Admin CFC, for CacheRealPath
- CF caches hostname, affects some tags (like CFHTTP)
  - <http://www.adobe.com/cfusion/knowledgebase/index.cfm?id=13115e61>
  - <http://www.talkingtree.com/blog/index.cfm/2006/5/23/Configuring-the-Caching-of-Hostname-Resolution-for-ColdFusion-MX>
  - <http://tjordahl.blogspot.com/2004/10/cfm-x-and-dns-caching.html>
- WebService Proxy stub is cached
  - Can clear that programmatically in CFMX 6, 7
  - [http://carehart.org/blog/client/index.cfm/2006/12/12/refreshing\\_web\\_services\\_programmatically](http://carehart.org/blog/client/index.cfm/2006/12/12/refreshing_web_services_programmatically)

## More: Perhaps Unexpected Caching (cont)

---

- CF compilations seem cached
  - [http://www.help.ablecommerce.com/faqs/ablecommerce\\_55\\_aspnet/install\\_and\\_setup/how\\_do\\_i\\_clear\\_the\\_coldfusion\\_cache\\_.htm](http://www.help.ablecommerce.com/faqs/ablecommerce_55_aspnet/install_and_setup/how_do_i_clear_the_coldfusion_cache_.htm)
- CF caching 404 error pages
  - <http://www.talkingtree.com/blog/index.cfm?mode=entry&entry=88BDF4E4-50DA-0559-A023C957D70CD2EC>
- Browser caching authentication state
  - [http://builder.com.com/5100-6389\\_14-5087907.html](http://builder.com.com/5100-6389_14-5087907.html)
- Impact of using Select \* in CFQUERYPARAM
  - <http://www.1pixelout.net/index.php?p=15>
  - [http://steve.coldfusionjournal.com/avoid\\_select\\_.htm](http://steve.coldfusionjournal.com/avoid_select_.htm)

## Caching Outside of CF

---

- Caching content on server
- Caching content on the network
- Content caching in browser
- Caching on the database
- JVM caching/garbage collection

# Non-CF: Caching Content on Server

---

- Various tools, means to cache content not in CF, but in the web server
  - Options built into web servers
    - <http://httpd.apache.org/docs/2.2/caching.html>
  - Tools like Squid, xcache, CacheRight
    - <http://www.squid-cache.org/>
    - <https://secure.xcache.com/Page.aspx?c=45&p=416>
    - <http://www.port80software.com/products/cacheright/>
- Servlet filter-based caching
  - Since CFMX is built atop J2EE, can leverage servlet filters
    - And there are filters which implement page content caching for you
      - as alternative to CFCACHE, for instance
  - See my article for more info, and resources
    - <http://coldfusion.sys-con.com/read/41574.htm>
- Another alternative:
  - Proxy caching
    - [http://en.wikipedia.org/wiki/Proxy\\_server](http://en.wikipedia.org/wiki/Proxy_server)

## Non-CF: Caching Content on the Network (Lan/Wan)

---

- Cache content on the router/network
- Alternatives such as
  - F5 Web Accelerator
    - <http://www.f5.com/products/WebAccelerator/>
  - Akamai Web Application Accelerator
    - <http://www.akamai.com/html/technology/products/waa.html>
  - Radware Cache Server Director
    - <http://www.radware.com/content/products/csd/default.asp>

# Non-CF: Caching Content in Browser

---

- Basics
  - Happens automatically, browser caches pages in memory, on disk
  - Can be controlled in browser setup
  - Can also be controlled in server-side directives to control caching a page
    - `<cfheader name="Expires" value="#GetHttpTimeString(Now())#">`
  - Can help or hurt you
- Surprises
  - Caching can happen with scripts/stylesheets in src files, images, errors
  - Can often use Ctrl-click on refresh button, or ctrl-f5, to force browser refresh
  - Any new URL variables will force browser to refresh
    - Can use any ?xxx to force refresh
  - Can also be controlled on web server, for static filetypes at least
    - In IIS:  
<http://www.microsoft.com/technet/community/columns/insider/ii0603.msp#ELF>

# Non-CF: Caching Content in Browser (cont)

---

- Gotchas
  - Different browsers honor different cache expiration directives
  - Tempting to change your browser to "never cache". Be careful, now not like your users
  - Problem when outputting alternative content types
    - May never see result of changes to your code
    - May need to open new browser to force browser to open the page correctly
  - Browser caching can also affect data sent to flex/ajax/flash apps
    - Use techniques above to force refresh on demand, or in coding
  - Be aware of Google Web Accelerator
    - <http://webaccelerator.google.com/webmasterhelp.html>
- Resources
  - <http://www.cfcdeveloper.com/index.cfm/2007/1/11/Prevent-Cache-in-Coldfusion>
  - Caching Tutorial for Web Authors and Webmasters
    - [http://www.mnot.net/cache\\_docs/](http://www.mnot.net/cache_docs/)
  - "Preventing Caching in Flash Player", <http://www.communitymx.com/abstract.cfm?cid=827EA>
  - Can also configure in web server
    - <http://www.websiteoptimization.com/speed/tweak/cache/>

# Non-CF: Caching on the Database

---

- Database connection pooling
  - Connecting from CF to DB can be expensive, especially as traffic grows
  - Option in CF Admin datasource setup to “maintain connections”
    - Also “advanced settings” and connection string options to control pooling further
- Buffer caching
  - Database engine will likely most do some caching of retrieved content, in memory of the DB server
    - Traditionally, this has been just internal “pages” or “blocks”, held in “buffers”
      - DB engine controls entirely what’s cached, flushed
      - Though configurable in DBMS, perhaps per database
- Query plan caching
  - Your SQL statement causes creation of a “query plan” in most DBs
  - DB engine will typically cache that, internally
  - You can affect it with CFQUERYPARAM, or hints (depending on DB)
  - Maxpooledstatements in SQL Server driver
    - Its impact on prepared statements in SQL Server

# Non-CF: Caching on the Database (cont)

---

- SQL statement result caching
  - More recently, some DBMSs have started supporting caching of results per SQL Statement
    - MySQL offers query caching
      - "The MySQL Query Cache"
        - <http://www.petefreitag.com/item/390.cfm>
        - <http://www.mysqlperformanceblog.com/2006/07/27/mysql-query-cache/>
      - Oracle 11G adding SQL caching
- Resources
  - See resources at end for general Admin page configuration docs
  - [http://en.wikipedia.org/wiki/Main\\_Memory\\_database](http://en.wikipedia.org/wiki/Main_Memory_database)
  - Analyzing SQL Server 2000 Data Caching
    - [http://www.sql-server-performance.com/tp\\_analyzing\\_sql\\_server\\_2000\\_data\\_caching.asp](http://www.sql-server-performance.com/tp_analyzing_sql_server_2000_data_caching.asp)

# Non-CF: JVM Caching/Garbage Collection

---

- Underlying JVM does its own caching of objects in memory
  - All your CF Code creates some object, perhaps many
    - Some may end with conclusion of request
    - Some may live on much longer time
  - When objects no longer needed, they're "garbage collected"
- See Peter Freitag's articles
  - <http://www.petefreitag.com/item/139.cfm>
  - <http://www.petefreitag.com/articles/gctuning/>
  - <http://www.petefreitag.com/item/141.cfm>
- Many others, such as:
  - <http://www.sumoc.com/blog/index.cfm?mode=entry&entry=CDCDBF8B-5004-2066-B7460CDEAB79328F>

# Flushing Strategies Summary

---

- Rolling interval since last use
- Fixed interval at recurring time
- All at once
- By storing subsets in a struct
- When data changes
- Based on some other event
- On server restart
- On application, session restart
  
- Each of the options mentioned in this talk use one or more
  - Just keep them in mind for your own solutions
  - And that's it for the caching options we'll discuss

# Resources for Learning More

---

- This presentation (and its details)
  - Available at [carehart.org/presentations.cfm](http://carehart.org/presentations.cfm)
- CFMX documentation
  - See CFML Reference pointers, as well as
  - CFMX Developers Guide, "Optimizing ColdFusion applications"
    - <http://livedocs.adobe.com/coldfusion/7/htmldocs/00001125.htm>
  - Cache-related Admin settings
    - [http://livedocs.adobe.com/coldfusion/7/htmldocs/wwhelp/wwhimpl/common/html/wwhelp.htm?context=ColdFusion\\_Documentation&file=00001706.htm](http://livedocs.adobe.com/coldfusion/7/htmldocs/wwhelp/wwhimpl/common/html/wwhelp.htm?context=ColdFusion_Documentation&file=00001706.htm)
  - Admin API
    - [http://livedocs.adobe.com/coldfusion/7/htmldocs/wwhelp/wwhimpl/common/html/wwhelp.htm?context=ColdFusion\\_Documentation&file=00001734.htm](http://livedocs.adobe.com/coldfusion/7/htmldocs/wwhelp/wwhimpl/common/html/wwhelp.htm?context=ColdFusion_Documentation&file=00001734.htm)

# Resources for Learning More (cont)

---

- "Classic" Articles
  - Caching in ColdFusion (CF 6 era, covers the "big three")
    - <http://www.adobe.com/devnet/coldfusion/articles/cfcaching.html>
  - Programmatic Caching in ColdFusion (from CF5 era)
    - [http://www.adobe.com/devnet/server\\_archive/articles/programmatic\\_caching\\_cf.html](http://www.adobe.com/devnet/server_archive/articles/programmatic_caching_cf.html)
  - Query Caching in ColdFusion (also CF5-era)
    - [http://www.adobe.com/devnet/server\\_archive/articles/query\\_caching\\_cf.html](http://www.adobe.com/devnet/server_archive/articles/query_caching_cf.html)
  - [http://www.adobe.com/devnet/coldfusion/articles/adv\\_queries.html](http://www.adobe.com/devnet/coldfusion/articles/adv_queries.html)
- Recent blog entries from Ashwin Matthews of Adobe
  - <http://blogs.sanmathi.org/ashwin/2006/06/27/an-introduction-to-caching/>
  - <http://blogs.sanmathi.org/ashwin/2006/07/01/memory-sensitive-caching-for-cf/>
  - <http://blogs.sanmathi.org/ashwin/2006/07/12/tangling-with-the-template-cache/>
  - <http://blogs.sanmathi.org/ashwin/2006/07/13/tangled-in-the-template-cache/>

## Resources for Learning More (cont)

---

- Community MX Articles (subscribers)
  - Beyond ColdFusion Query Basics
    - <http://www.communitymx.com/content/article.cfm?cid=C6428>
  - A Quick Approach to Home-Brewed Content Caching in ColdFusion - Part 1: The Theory
    - <http://www.communitymx.com/abstract.cfm?cid=68826>
  - A Quick Approach to Home-Brewed Content Caching in ColdFusion - Part 2: The CFC
    - <http://www.communitymx.com/abstract.cfm?cid=8CEFC>
  - A Quick Approach to Home-Brewed Content Caching in ColdFusion - Part 3: In-Memory Caching
    - <http://www.communitymx.com/abstract.cfm?cid=CDBE4>

# Other Resources

---

- CF-specific caching resources
  - "Strategies For Scaling ColdFusion: Caching"
    - [http://www.cfconf.org/cf\\_scale/talks/StrategiesForScalingColdFusion\\_caching.ppt](http://www.cfconf.org/cf_scale/talks/StrategiesForScalingColdFusion_caching.ppt)
  - Various techniques
    - <http://www.coldfusioncookbook.com/category/15/Caching>
  - "Good Developers Practice Safe Query Caching"
    - <http://mkruger.cfwebtools.com/index.cfm?mode=entry&entry=EAA0D1CA-01F6-F3EC-5520AAD6EEC68061>
  - "Custom Caching Solutions", Article
    - <http://www.webmonkey.com/webmonkey/02/26/index4a.html?tw=programming>
- Generic caching resources
  - [http://en.wikipedia.org/wiki/Web\\_cache](http://en.wikipedia.org/wiki/Web_cache)
  - "Web Caching and Content Delivery Resources"
    - <http://www.web-caching.com/>
  - "Web Caching: Making the Most of Your Internet Connection"
    - <http://www.web-cache.com/>
  - "Caching and Cache Control"
    - <http://www.port80software.com/products/cacheright/cachingandcachecontrol>

# Summary

---

- Introduced caching concepts
- Covered over a dozen forms of caching
  - Query caching: tag- and scope-based
  - Page caching: Full and partial
    - CFCACHE, CFSAVECONTENT, custom tags
  - Template caching and trusted cache setting
  - Several more caching options
    - Caching queries in scopes, or to disk
    - Caching CFCs
    - Caching CFHTTP, Web Service calls
    - Still more CF caching features, and perhaps unexpected caching
  - Caching outside of CF
    - Caching content on server
    - Caching content on the network
    - Content caching in browser
    - Caching on the database
    - JVM caching/garbage collection
- Introduced various gotchas and surprises
- Showed how to manage/flush cache and monitor cache effectiveness

# Questions on presentation

---

- Charlie Arehart
  - [charlie@carehart.org](mailto:charlie@carehart.org)
- I'd really appreciate your feedback
  - <http://carehart.org/feedback/>
- Also available for setup and implementation consulting
  - Also other developer productivity coaching, system admin and tuning support, and more
  - Remote or on-site
- New Per-minute Phone/Web support
  - <http://carehart.org/askcharlie/>