How Skin Caching Works in Handshake

# Introduction

Handshake is capable of caching the HTML rendered by a skin. Caching can be set globally but be overridden in individual skins. Caches are defined by three things:

1. Content to cache
2. Expiration time
3. Key that determines if an item is already present in the cache

In Handshake skin rendering the content to cache is everything generated by the skin. The expiration time is managed using the settings described below.

All input properties are used to determine the cache key. This means that the same skin rendered for two users created different entries in the cache. The same skin rendered for the same user, but for a different MPInfo (client, matter, etc.) results in two different cache entries.

# Global Control of Caching

The global setting for caching are controlled in the appsettings in the HandshakeWebServices web.config file. For any caching to take place the “UseRenderingCache” must be “1”.

The default cache expiration is determined by the “DefaultCacheDuration” setting in web.config. This value (in minutes) is added to the current time to determine when that cached item expires.

# Controlling Caching in Individual Skins

The <page> element in a skin is where a skin can request different caching than the default. Use a “cacheduration” attribute to override the global default. For example:

<page cacheduration=”3”>

Requests a 3 minute cache time for this skin. To request no caching when global caching is enabled use a value of “none” or “0”.

Skins may also request a cache that is based on a specific time of day. This can be useful is a skin is reporting from a warehouse that is updated once a day. To use this type of caching put a “cachetime” attribute on the <page> element as shown here:

<page cachetime=”06:00”>

This example requests that the cache be maintained until 6AM today (or tomorrow if 6AM has already passed today).

# Preventing Caching for an Individual Request

Even when caching would normally be used for a rendering request you can prevent the cached version from being used by passing \_HS\_NoCache with any value as one of the input properties.

# Detecting Cache Usage in the Handshake Log

The RenderControl log can tell you if a skin is cached when it is rendered. It can also tell you if a cached copy of skin rendering is used to execute a request.

When a rendering is cached (if the log level is at least “Info”) the RenderControl log has an entry saying “Adding skin rendering to cache.” When a cached rendering is returned (if the log level is at least “Debug”) the RenderControl log has an entry saying “Returning results from cache.”

# RenderControl.asmx Methods that Affect the Cache

The RenderControl web service has several methods that remove cached items:

## RemoveFromCacheBySkinName(skinName)

This method removes all cached entries that relate to rendering of the specified skin.

## RemoveFromCacheByUser(username)

This method removes all cached entries for the user with the specified user name.

## ClearCache()

This method removes all entries from the rendering cache.