

## HowTo: Konfigurieren von Caching in SharePoint 2007

Eines haben alle SharePoint Seiten gemeinsam, die meisten ihrer Daten sind in einer SQL Datenbank gespeichert. Das Wenige was nicht in der einer SQL Datenbank gespeichert ist, sind die JavaScript Files, CSS, Bilder und ASPX Dateien welche für alle Site Templates gleich sind. Bei einer Publishing Site hingegen wird der gesamte Benutzerinhalt, JavaScripts, style sheets, Bilder und ASPX Dateien in der SQL Datenbank gespeichert. Jedesmal wenn also eine Seite angefordert wird, wird der gesamte Inhalt aus der SQL Datenbank ausgelesen. Dies hat einen offensichtlichen Einfluss auf die Performance der anzuzeigenden Seiten. Dies könnte man doch deutlich schneller gestalten wenn man den Inhalt von viel aufgerufen Seiten von einer lokalen Disk oder Memory zur Verfügung stellen könnte. SharePoint stellt für die Beseitigung dieses „bottleneck“ Caching Mechanismen zur Verfügung.

SharePoint stellt Administratoren von Publishing Sites drei Typen von Caching Mechanismen zur Verfügung:

**Page Output Caching:** Ist ein Mechanismus um die kompilierte Seite im persistenten Speicher des SharePoint Server zu behalten. Dies eliminiert den Datenbank Zugriff für jeden weiteren Zugriff auf Inhalte dieser Seite.

**Disc Caching:** Statischer Inhalt, wie etwa Bilder, CSS und JavaScript Files werden persistent auf die Harddisk des SharePoint Webservers gespeichert. Bei einem Aufruf werden diese von der Disk gelesen, statt erneut von der SQL Datenbank ausgelesen zu werden.

**Object Cache:** Der Object Cache speichert, nachdem alle Caches abgelaufen sind und sich eine Seite im „draft“ Modus befindet, die am meisten verwendeten Ressourcen. Zum Beispiel die Site Navigation Struktur, oder den Inhalt von Feldern einer Dokumenten Bibliothek.

### Page Output und Disk Caching

Um Output Caching für eine SharePoint Seite zu aktivieren, muss das „MOSS Publishing Infrastructure“ Feature für die Site Collection (Web Seiten Sammlung) aktiviert sein. Wenn die Seite mit dem Template „Publishing Portal“ oder „ Collaboration Portal“ Site Template erstellt wurde, ist dieses Feature bereits aktiviert. Um zu Überprüfen ob das Feature aktiviert ist:

1. Site Actions -> Site Settings > Modify All Site Settings
2. Site Collection Features

### Erster Schritt: Das Verhalten des Output Caching anpassen

MOSS 2007 bietet Cache Konfigurationen, basierend auf ASP.NET, in Form von „output cache profiles“ an. Ein Cache Profil ist ein Liste in welcher das Verhalten des Output Caches spezifiziert wird. Jede Site Collection enthält, eine oder mehrere, dieser Profile. Diese können unterschiedlichen Teilen der Webseite zugewiesen werden, so dass die Performance des Systems optimiert werden kann.

Unten (Abbildung 1) ist ein Beispiel wie eine Cache Profile Konfiguration aussehen kann. Einige Einstellungen können vorgenommen werden. Uns interessieren im Moment aber nur „Checking for ACL, Vary by User Rights, Duration und Check for Changes.“

   <span style="float: right;">* indicates a required field</span>	
<b>Title *</b>	Extranet (Published Site)
<b>Display Name</b>	<input type="text"/> Display name is used to populate the list of available cache profiles for site owners and page layout owners.
<b>Display Description</b>	<input type="text"/> Optimized for public extranet site with the following characteristics: Display description is used to populate the list of available cache profiles for site owners and page layout owners.
<b>Perform ACL Check</b>	<input checked="" type="checkbox"/> Selecting this check box ensures that all items in the cache are appropriately security trimmed. Clearing this check box allows for better performance but should only be applied to sites or page layouts that do not have information that needs security trimming.
<b>Enabled</b>	<input checked="" type="checkbox"/> Selecting this check box turns caching on. If you clear this check box, caching will not take place anywhere this profile is selected. Clearing this check box can be useful for troubleshooting the rendering of all sites and page layouts associated with this cache profile. Remember to select this check box when troubleshooting is complete.
<b>Duration</b>	<input type="text" value="180"/> Duration in seconds to keep the cached version available.
<b>Check for Changes</b>	<input type="checkbox"/> Selecting this check box validates on each page request that the site has not changed and flushes the cache on changes to the site. Clearing this check box can improve performance but will not check for updates to the site for the number of seconds specified in duration.
<b>Vary by Custom Parameter</b>	<input type="text"/> As specified by <code>HttpCachePolicy.SetVaryByCustom</code> in ASP.Net 2.0.
<b>Vary by HTTP Header</b>	<input type="text" value="Browser"/> As specified by <code>HttpCachePolicy.VaryByHeaders</code> in ASP.Net 2.0.
<b>Vary by Query String Parameters</b>	<input type="text"/> As specified by <code>HttpCachePolicy.VaryByParams</code> in ASP.Net 2.0.
<b>Vary by User Rights</b>	<input checked="" type="checkbox"/> Selecting this check box ensures that users must have identical effective permissions on all securable objects to see the same cached page as any other user.
<b>Cacheability</b>	<input type="text" value="ServerAndPrivate"/> As specified by <code>HttpCacheability</code> in ASP.Net 2.0.
<b>Safe for Authenticated Use</b>	<input checked="" type="checkbox"/> This check box should be selected for only those policies that you want to allow to be applied to authenticated scenarios by administrators and page layout designers.
<b>Allow writers to view cached content</b>	<input type="checkbox"/> Selecting this check box bypasses the normal behavior of not allowing people with edit permissions to have their pages cached. This check box should be selected only in scenarios in which you know that the page will be published, but will not have any content that might be checked out or in draft.
Created at 12/24/2006 1:34 PM by System Account Last modified at 12/24/2006 1:34 PM by System Account <span style="float: right;"> <input type="button" value="OK"/> <input type="button" value="Cancel"/> </span>	

Abbildung 1

Das Beispiel aus Abbildung 1 ist für das Extranet gedacht. Deswegen wird ein ACL Check durchgeführt. ACL steht für „Access Control List“ und entspricht den erweiterten Berechtigungen von NTFS. Für eine Site welche einen anonymen Zugriff erlaubt, etwa eine Internet Seite, kann diese Einstellung deaktiviert werden, um noch einmal ein bisschen Performance zu gewinnen. Ebenso wird für den Extranet Zugriff „Vary by User Rights“ aktiviert. Zusammen mit den ACL's wird sicher gestellt, dass nur authentifizierte Benutzer basierend auf den ACL auf den gecacheten Inhalt Zugriff bekommen. Das System lässt so keinen Zugriff auf Inhalt auf dem Cache zu welcher vom Benutzer nicht auch auf der eigentlichen Webseite eingesehen werden könnte.

Wenn die Seite gecached ist, werden alle Requests auf diese Seite vom Cache her beantwortet. Der Cache verfällt nach der spezifizierten Dauer welcher bei „Duration“, in unserem Beispiel 180 Sekunden, beträgt. Nach dieser Zeit wird der aktualisierte Inhalt für die neuen Benutzer gecached.

Bei einer Grundinstallation werden schon einige Cache Profile mitgeliefert, man kann jedoch auch eigene erstellen. Der Name der Liste ist „Cache Profile“. Man kommt zu dieser Liste über:

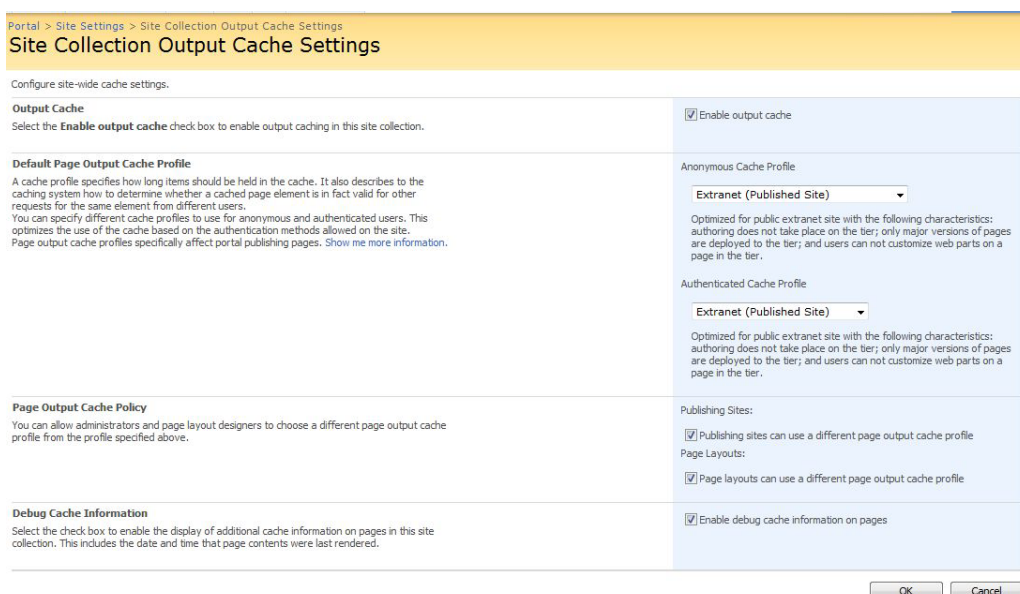
Site Actions -> Modify all site settings

Auf der Root Webseite. Dort klickt man „Site collection cache profiles“ unter der „Site collection Administration“ Spalte auf der „site setting“ Seite. Dort sind alle verfügbaren Profile aufgeführt. Wenn man ein Profil auswählt, kommt man über „Edit item“ in die Konfiguration des Profils. Ja, wie immer bei SharePoint, viele Klicks führen nach Rom.

## Zweiter Schritt: Output Caching aktivieren und einer Sektion der Seite zuweisen

Wenn das Profil den Bedürfnissen angepasst ist, oder ein neues erstellt wurde, ist man bereit diese seiner Seite, oder Sektion einer Seite zuzuweisen. Für das Beispiel gehe ich davon aus, dass wir ein Extranet konfigurieren.

1. In den Site Settings geht man zu „Site Collection Output Cache Settings“. Diese ist in den Site Settings allerdings nur als „Site collection output cache“ beschriftet (Abbildung 2)



Portal > Site Settings > Site Collection Output Cache Settings  
**Site Collection Output Cache Settings**

Configure site-wide cache settings.

<b>Output Cache</b> Select the <b>Enable output cache</b> check box to enable output caching in this site collection.	<input checked="" type="checkbox"/> Enable output cache
<b>Default Page Output Cache Profile</b> A cache profile specifies how long items should be held in the cache. It also describes to the caching system how to determine whether a cached page element is in fact valid for other requests for the same element from different users. You can specify different cache profiles to use for anonymous and authenticated users. This optimizes the use of the cache based on the authentication methods allowed on the site. Page output cache profiles specifically affect portal publishing pages. <a href="#">Show me more information.</a>	Anonymous Cache Profile Extranet (Published Site) <p>Optimized for public extranet site with the following characteristics: authoring does not take place on the tier; only major versions of pages are deployed to the tier; and users can not customize web parts on a page in the tier.</p> Authenticated Cache Profile Extranet (Published Site) <p>Optimized for public extranet site with the following characteristics: authoring does not take place on the tier; only major versions of pages are deployed to the tier; and users can not customize web parts on a page in the tier.</p>
<b>Page Output Cache Policy</b> You can allow administrators and page layout designers to choose a different page output cache profile from the profile specified above.	Publishing Sites: <input checked="" type="checkbox"/> Publishing sites can use a different page output cache profile Page Layouts: <input checked="" type="checkbox"/> Page layouts can use a different page output cache profile
<b>Debug Cache Information</b> Select the check box to enable the display of additional cache information on pages in this site collection. This includes the date and time that page contents were last rendered.	<input checked="" type="checkbox"/> Enable debug cache information on pages

Abbildung 2

2. Output Cache aktivieren (enable)
3. Anonymous kann deaktiviert werden oder auch nicht, da vermutlich bei einer Extranet Seite keine anonymen Benutzer zugreifen werden. Hingegen bei den authentifizierten Benutzern das gewünschte Profil angegeben.
4. Bei den Cache Policys definiert man nun wie die Vererbung der hier gesetzten Cache Profile angewendet werden soll. Wird man untergeordneten Seiten und Listen andere Profile zuweisen, müssen diese beiden Optionen aktiviert werden.

Wenn man diese vier Schritte durchgeführt hat, ist Output Cache für die gesamte Site Collection, mit dem gewünschten Profil, aktiviert. Die selektive Kontrolle des Output Cache Verhalten kann auf zwei weiteren Levels eingestellt werden: auf Stufe „sub-site“ und „page layout“.

Wenn man den „Debug Cache“ aktiviert, wird am Ende des HTML Markups eine Information mitgegeben, welche anzeigt, dass der Cache aktiviert ist. Diese Option ist nur für das Debugging gedacht. Sie zeigt an, welches Cache Profil angewendet wurde sowie den Zeitstempel der gecachten Version.

```
</BODY>
</HTML>
<!-- 8 Output cache not used. Reason: User can view an unpublished version
of the current page. -->
```

```
</BODY>
</HTML>
<!-- Rendered using cache profile:Public Internet at: 2007-08-13T22:38:59 -
-->
```

### Dritter Schritt: Aktivieren von disk-based Cache

Eine typische SharePoint Webseite enthält ein oder mehrere Referenzen auf Bilder oder CSS Dateien. Diese Ressourcen könnten in einer Library auf einer SharePoint Seite gespeichert sein. SharePoint bietet einen disk-based Cache für solche Objekte. Dieser Cache wird auch „blob“ Cache genannt. Dabei steht „blob“ für „Individual Binary Large Object“. Der Zweck dieses Cache ist es, zu verhindern dass zusätzliche Datenbank abfragen für immer dieselben Objekte gemacht werden müssen. Dies wird gemacht, in dem diese Dateien auf dem File System des Front-End Server gespeichert werden. Standardmässig ist der disk-based Cache deaktiviert. Jeder Front-End Server in einer Farm hat seinen eigenen disk-based Cache.

Um den disk-based Cache auf einem Front-End Server zu aktivieren muss man lediglich den „blob-Cache“ in der „web.config“ Datei in der gewünschten Web Applikation auf „enable=true“ setzen.

```
<BlobCache location="C:\blobCache" path="\. (gif|jpg|png|css|js)$"
maxSize="10" max-age="86400" enabled="true" />
```

Eine interessante Funktion ist das „max-age“ Attribut. „max-age“ ist die Zeit (in Sekunden) in der, der Browser die Objekte auf dem Client Computer lokal gecached hält. Wenn hier ein Wert grösser als 0 steht, wird der Browser die URL des Objektes nicht abrufen, so lange die lokale Kopie noch nicht verfallen ist. Wenn viele Objekte auf einer Seite nicht oft wechseln, ist das eine Option, von der man

auf jeden Fall Gebrauch machen sollte. Wenn man disk-based Cache nicht aktiviert hat, behält der Browser in den meisten Fällen eine lokale Cache Kopie. SharePoint 2007 sendet dem Client einen 304 Code um ihm anzuzeigen, dass seine Kopie aktuell ist. Trotzdem gewinnt man hier Performance da unnötige HTTP-Requests und eventuelle Datenbank-Anfragen wegen Berechtigungen auf Objekte, wegfallen.

Wenn man zusätzliche Dateien für seine Seite verwendet welche man auch noch auf dem Front-End Server gecached haben möchte, kann man diese auch hier eintragen. Hier ein Beispiel für Bitmaps.

```
<BlobCache    location="C:\blobCache"    path="\.(gif|jpg|png|css|js|bmp)$"
maxSize="10" max-age="86400" enabled="true" />
```

Der disk-based Cache kann sowohl für anonymen wie auch für authentifizierten Zugriff verwendet werden. Der disk-based Cache kann aber nur Ressourcen verarbeiten welche in SharePoint Bibliotheken gespeichert und publiziert sind. Objekte welche den Status „Draft“ oder „Checked out“ haben werden nicht gecached.

Damit nun alle geänderten Modifikationen aktiviert werden, muss die Seite wieder eingchecked und publiziert werden.

## Object Cache

Anders als Output Cache oder disk-based Cache, ist der Object Cache standardmässig aktiviert. Er kann im Gegensatz zu den anderen Cache Methoden auch für „draft“ Seiten verwendet werden.

SharePoint 2007 speichert seine Inhalte in einer SQL Datenbank. Das beziehen benötigter Daten aus der Datenbank ist langsamer als der Zugriff auf ein lokales Filesystem. Da Output Cache und disk-based Cache während der Zeit da eine Seite editiert wird nicht greift, wird hier der Object Cache verwendet. Der Object Cache speichert, nachdem alle Caches abgelaufen sind und sich eine Seite im „draft“ Modus befindet, die am meisten verwendeten Ressourcen. Zum Beispiel die Site Navigation Struktur, oder den Inhalt von Feldern einer Dokumenten Bibliothek. Dieses Caching passiert eher MOSS intern, so dass man hier nicht viel ändern muss, ausser der Grösse des Object Cache.

Als SharePoint Administrator kann man die maximale Anzahl von Memory, die für den Object Cache verwendet, wird festlegen. Es gibt für den Performance Monitor Indikatoren welche man hinzufügen kann um diesen Wert zu messen und den aktuellen Bedürfnissen anzupassen (Abbildung 3). Das Objekt nennt sich „SharePoint Publishing Cache“. Um hier eine Aussage machen zu können vergleicht man den „Publishing hit ratio“ und den „Total Object discards“ Indikator. Bei all diesen Messungen und Anpassungen sollte man allerdings nicht vergessen, dass die verschiedenen Caches den gleichen virtuellen Adressraum teilen. Wenn der Object Cache zu gross gesetzt ist, hat man eventuell nicht genug Platz für andere Caches, wie etwa „output cache“. Dies ist vor allem für 32-Bit System wichtig. Jeder IIS Prozess (W3WP) hat ein Maximum von 2 GB RAM für die Applikation welcher er betreibt. Egal wie viel RAM im Server eingebaut sind. Wenn „output cache“ im grossen Stil gebraucht wird ist es auf jeden Fall angezeigt einen solchen Front-End Server auf einer 64-Bit Plattform zu betreiben. Um hier zu testen, kann man mit einem Object Cache Grösse von 200 MB beginnen und dann kleiner werden, sobald man sieht, dass die „Publishing hit ratio“ klein ist, sprich kleiner als 85%. Ebenfalls ein

guter Test ob die Performance ausbalanciert ist, die Webseite editieren und zu schauen wie „schnell“ die „User Experience“ ist.

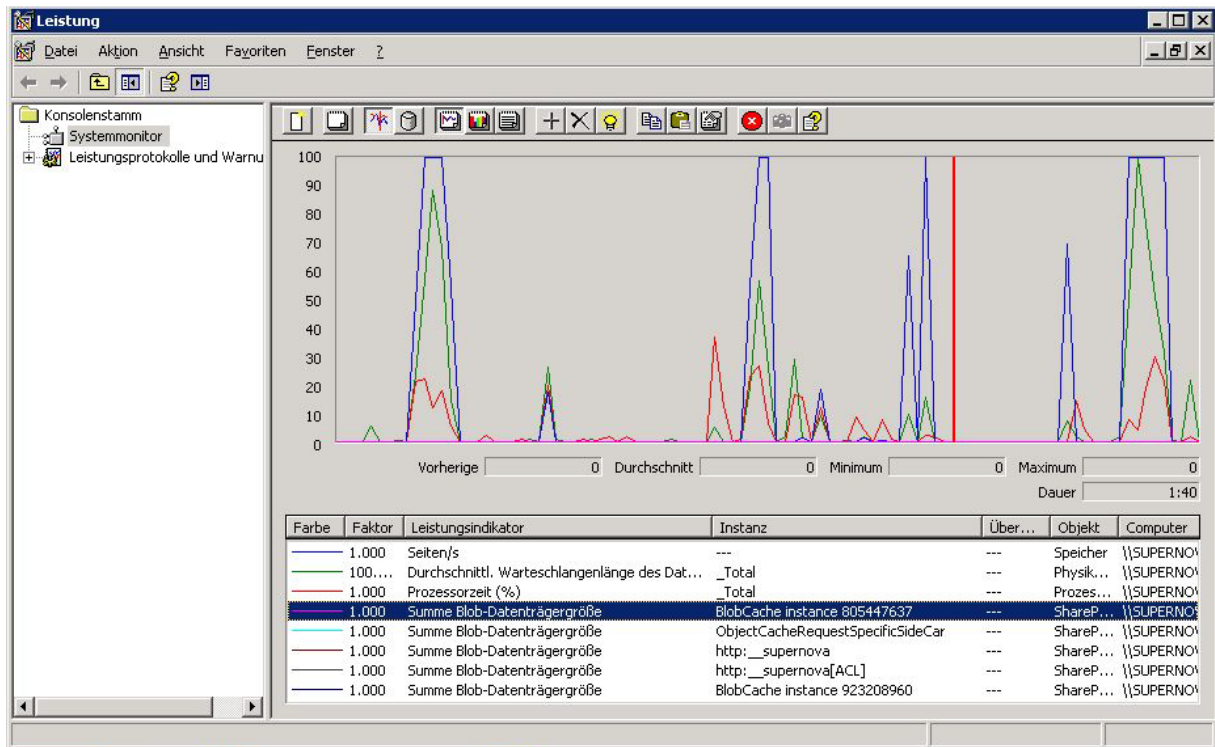


Abbildung 3

## Zusammenfassung

Cache Art	Wo	Bemerkung
Output Caching und Cache Profile	Individuelle Seiten	Ideal für stark frequentierte Seiten die nur alle paar Minuten neue Inhalte haben.
Object Caching	Web Part, Felder und Inhalt	Optimal für Seiten auf denen viel editiert wird.
Disk-based Cache	Individuelle Seiten	Ideal für Bilder, Ton, Filme und Code Dateien.

## Mehr Informationen:

Disk-based Cache: <http://msdn2.microsoft.com/en-us/library/aa604896.aspx>

Output Caching und Cache Profile: <http://msdn2.microsoft.com/en-us/library/aa661294.aspx>

8.2007 – Christoph Müller, scolab.ch