

## Sitecore CMS 7.0 以降

# CMS 診断ガイド

Sitecore CMS のパフォーマンス診断のための開発者用ガイド



## 目次

Chapter 1	イントロタクション	5
Chapter 2	診断の手順	6
2.1 ブラ	ラウザー ページのテスト	7
2.1.1	セットアップ	7
2.1.2	AOL Pagetest ブラウザー プラグインの使い方	7
情報の	)収集	8
第1	パス — ブラウザー キャッシュがクリアされている	9
第 2 /	パス - ブラウザー ページを最新の情報に更新	11
第 3 /	パス — 最近訪問した Web ページ	12
2.1.3	結果の分析	13
Wate	rfall タブ	13
Chec	klist タブ	14
不合材	各と注意を解消するための情報	17
•	nization Report タブ	
	Details タブ	
2.2 ^-	-ジのパフォーマンスの調査 — IIS ログの使用	18
2.2.1	必要なスキル	18
2.2.2	症状	
2.2.3	パートナー/お客様への確認	
2.2.4	長い時間がかかっている要求の IIS ログのクエリの手順	18
2.2.5	結果を理解する	19
2.2.6	Sitecore の推奨事項	
•	rt Findings	
	·ダリング パフォーマンス	
2.3.1	必要なスキル	
2.3.2	症状	
2.3.3	Sitecore の統計情報ページの使用手順	
2.3.4	結果を理解する	
2.3.5	Sitecore の推奨事項	
	D報告	
	ecore ログを使用したメモリ リークの調査	
2.4.1	必要なスキル	
2.4.2	症状	
2.4.3	Sitecore ログ アナライザーを使った Sitecore ログの解析	
	ess¥Private Bytes および # Bytes in all Heaps カウンターの値の取得 Dグラフの作成	
2.4.4	結果の理解(グラフ)	
2.4.5	注:	36



2.5 Site	core パイプライン ブロファイリング	37
2.5.1	必要なスキル	37
2.5.2	Sitecore パイプライン プロファイリング ページの統計情報ページの使用手順	37
2.5.3	概要	37
2.5.4	Usage 使用	38
2.5.5	結果の理解	38
2.6 Site	core デバッガー	41
2.6.1	Sitecore デバッガーの起動手順	41
2.6.2	使用	42
	検索診断	
3.1 Fill	DB	47
3.1.1	FillDB Page を使ったアイテムの作成	47
	長なログ	
3.3 Lind	qScratchPad	50
3.3.1	必要なスキル	50
3.3.2	使用	50



この文書に記載されている情報は、記載内容に関する発行時点におけるサイトコア コーポレーションの見解を示すものであり、いかなる時点にも予告なく変更される場合があります。この文書およびその内容はいかなる保証もなく現状有姿のまま提供されるものであり、サイトコアによる提供または確約と見なされるものではありません。サイトコアは記載されている情報の正確性に関するいかなる保証もいたしません。サイトコアはこの文書について、明示または黙示にかかわらず、いかなる保証もいたしません。

他の会社の製品についての説明が記載されている場合、その説明はお客様の利便性のみを考慮して提供されているものであり、サイトコアがその製品を推奨またはサポートしていることを示すものではありません。サイトコアはそれらの情報の正確性を保証することはできません。また、それらの製品は時間の経過に伴い変更されることがあります。それらの説明は内容を網羅したものではなく、お客様の理解に役立てていただけるよう主要部分を要約したものです。それらの製品の正式な説明については、それぞれの製造元にお問い合わせください。

すべての商標は各社の商標です。

©2013 Sitecore Corporation. All rights reserved.



## Chapter 1

## イントロダクション

このガイドは Sitecore CMS パフォーマンス ガイドと併せてご使用いただけます。パフォーマンス上の問題を識別するために役立つ、一連の診断の手順を説明します。またチューニングによってパフォーマンスを向上させる方法についても説明します。

この文書には次の章があります。

- Chapter 1 イントロダクション
- Chapter 2 診断の手順
- Chapter 3 検索診断



## Chapter 2

## 診断の手順

診断の手順には、Sitecore の実装におけるパフォーマンス上の問題を識別するための、一連のテストが含まれます。

記載されている手順は CMS のチューニングを補完するものです。Sitecore CMS のチューニングを行う前と後に、この診断の手順を実行することにより、パフォーマンスの向上を記録することができます。

この章には次のセクションがあります。

- ブラウザー ページのテスト
- ページのパフォーマンスの調査 IIS ログの使用
- レンダリング パフォーマンス
- Sitecore ログを使用したメモリ リークの調査
- Sitecore パイプライン プロファイ



## 2.1 ブラウザー ページのテスト

Web サイトの上位 500 サイトの Web ページの平均サイズは 2003 年とくらべて 5 倍以上になりました。2003 年から 2009 年の間に、Web ページの平均サイズは、93.7 K から 507 K 以上 (図 1 参照) へと、5.4 倍になりました (Domenech et al.2007、Flinn & Betcher 2008、Charzinsk 2010)。この 6 年の間に、平均的な Web ページ上のオブジェクト数は、ページあたり 25.7 個から 64.7 個と、2 倍以上になりました。さらに長期の統計によると、1995 年と比較して、平均的な Web ページのサイズは 35 倍に、ページあたりのオブジェクト数は 28 倍に増加したと報告されています。-http://www.websiteoptimization.com/speed/tweak/average-web-page/

この事実が示すことは、今日の Web ページにおいては、1 ページに読み込まれるオブジェクトの数とサイズを理解することが重要であるということです。また、Web サーバーでオブジェクトのキャッシュを設定し、実行される要求の数をなるべく少なくすることも重要です。

Web ページがブラウザーに読み込まれる際の動作を分析することにより、多くの情報を収集できます。要求の対象、 レスポンス時間、オブジェクトのサイズ、などに関する情報を収集できます。また、キャッシュ、圧縮、CDNの使用、キー プアライブ、などについてサーバーの設定に関する情報も参照できます。

このタスクでは、オープン ソースのプラグインである AOL Pagetest の使い方と、結果の分析方法について説明します。

### 2.1.1 セットアップ

AOL Pagetest ブラウザー プラグインは下記からダウンロードできます。 http://sourceforge.net/projects/pagetest/files/

ダウンロードしたら、既定の場所にインストールします。AOL Pagetest プラグインは [ツール] メニューに表示され、IE 7 以降のブラウザーで動作します。

#### メモ

インストール時に IE が開かれていた場合、AOL Pagetest が [ツール] メニューに表示されるようにするには、IE を再起動する必要があります。

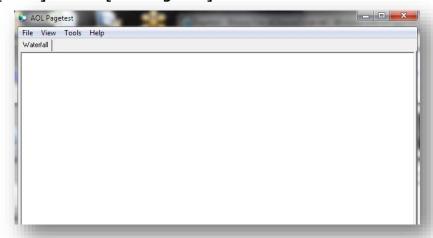
## 2.1.2 AOL Pagetest ブラウザー プラグインの使い方

AOL Pagetest を起動するには、次の手順を行います。

1. IE を起動します。



2. [Tools] メニューで [AOL Pagetest] をクリックします。



#### 情報の収集

この演習では、ページの読み込みに関する情報の収集について、3 つの役立つシナリオを説明します。ブラウザー キャッシュがクリアされている場合の Web ページの対応、Web ページの再読み込み、読み込まれたキャッシュのある Web ページの要求、です。

• ブラウザー キャッシュがクリアされている — 使用するブラウザーから初めてそのページを訪問する場合をシミュレートします。

IE 8 以降でブラウザー キャッシュをクリアする方法:

- 1) 「セーフティ] メニューで [閲覧の履歴の削除] をクリックします。
- 2) [お気に入り Web サイト データを保存する] のチェックボックスをオフにし、[インターネット一時ファイル]、「Cookies]、「履歴] のチェックボックスをオンにします。

**IE 7** でブラウザー キャッシュをクリアする方法:

- 1) 「**ツール**] メニューで [インターネット オプション] をクリックします。
- 2) [閲覧の履歴]で[削除]をクリックします。
- 3) キャッシュを削除するには [ファイルの削除] をクリックします。
- 4) 「**閉じる**]、「**OK**] をクリックします。
- Web ページの再読み込みまたは最新の情報に更新

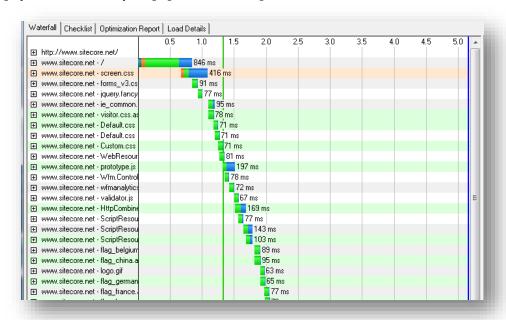
これによりキャッシュにあるオブジェクトへの要求を強制的に実行し、状態コード 304 となります。状態 304 の場合には、オブジェクトへの要求は実行されますが、ダウンロードは発生しません。

最近訪問した Web ページへのアクセス — URL を入力
こくわずかな要求チェーンとなります。要求が発生することなく、以前にブラウザー キャッシュに 読み込まれたアイテムにアクセスします。



## 第1パス - ブラウザー キャッシュがクリアされている

- 1. IE を起動します。
- 2. ブラウザー キャッシュをクリアします。
- 3. AOL Pagetest を起動します。
- 4. テストする URL を入力します。 例: http://www.sitecore.net.
- AOL Pagetest ウィンドウには 4 つのタブがあります [Waterfall]、[Checklist]、
   [Optimization Report]、[Load Details]。

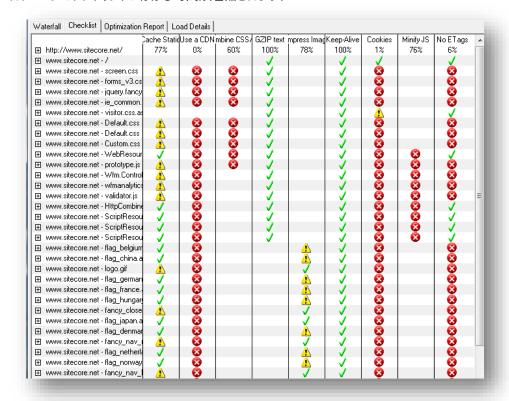


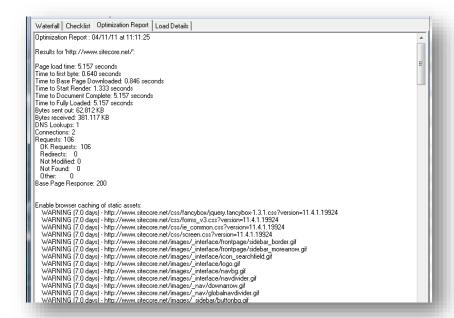
#### 考察は次のとおりです。

- [Waterfall] タブには、Web ページの各オブジェクトに対して実行されたすべての要求が表示されています。
- CSS や JS ファイルなどは独立した項目になっており、それらをダウンロードするために複数の要求が実行されています。 すべての CSS と JS ファイルをまとめて表示すると、時間軸は左に寄せられ、1 つの要求が実行さ

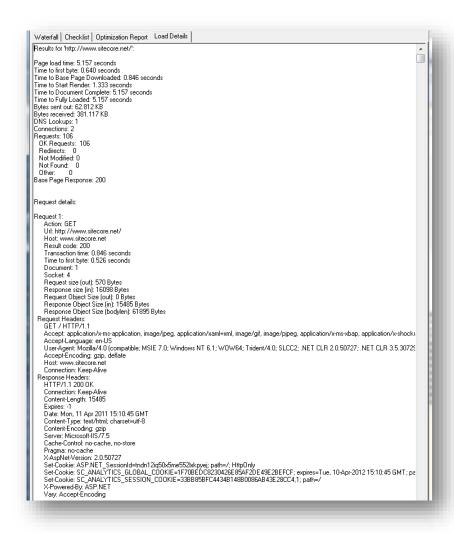


#### れ、ページのレンダリングにかかる時間が短縮されます。







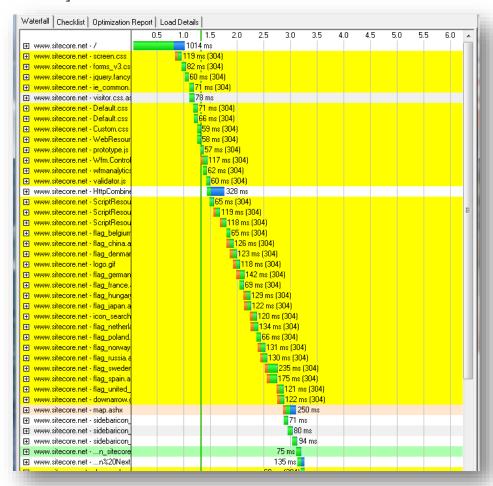


## 第2パス-ブラウザーページを最新の情報に更新

- 1. AOL Pagetest の [File] メニューで [New] をクリックします。
- 2. ブラウザーページを最新の情報に更新します [最新の情報に更新] をクリックするか、[F5] を押します。



3. [Waterfall] タブで結果を確認し、第1パスの結果と比較します。



#### 考察は次のとおりです。

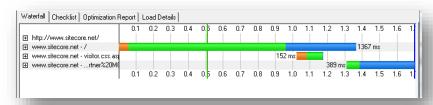
- 黄色で塗られているオブジェクトは 30x の状態コードになっています。
- この結果は、キャッシュ済みのアイテムが要求されたが、ダウンロードは行われなかったことを示しています。これは [最新の情報に更新]を行ったページにおける通常の動作です。
- 304 状態コードとなったアイテムは第 3 パス「最近訪問したページ」では要求されないはずであることも、この 図からわかります。

#### 第 3 パス — 最近訪問した Web ページ

- 1. AOL Pagetest の [File] メニューで [New] をクリックします。
- 2. ブラウザーで URL を選択して、[Enter] キーを押します ブラウザーで最新の情報に更新するボタンをクリックしないでください。



3. [Waterfall] タブで結果を確認し、第1と第2パスの結果と比較します。



考察は次のとおりです。

- 要求ツリーは大幅に減少しました。
- 最初の 2 つの要求である / と visitor.css は動的であり、キャッシュされていないことがわかります。
- 3 つ目のアイテムの要求がキャッシュから取得しなかった原因は調査が必要です。
   さらに調査を行った結果、この要求は訪問のたびに変化する、ローテーションの画像であることがわかりましたがってキャッシュとならなくても問題ありません。

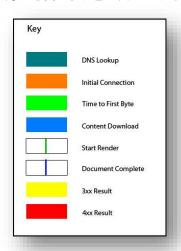
### 2.1.3 結果の分析

[Waterfall]、[Checklist]、[Optimization Report]、[Load Details] の各タブで表示されている情報 について説明します。

#### Waterfall タブ

[**Waterfall**] タブは Web ページで要求されたオブジェクトを構成する要求チェーンの情報を示します。要求されたオブジェクトは左側に表示され、上部に時間が表示されています。

要求と応答を示す色は次のとおりです。



- DNS Lookup 要求された URL の IP アドレスを検索するのにかかる時間です。
- Initial Connection クライアントからサーバーへのソケットを開くのにかかる時間です。



要求のたびに初期接続が実行されている場合には、サーバーで HTTP キープアライブが有効化されていないことを示している場合があります。

- Time to First Byte 要求からオブジェクトの最初のバイトを受信するまでにかかる時間です。If the この時間が長くかかる場合には、サーバー側に問題がある場合 (たとえば、パフォーマンス上の問題)や、ネットワークの問題である場合があります。
- Content Download オブジェクトをダウンロードするのにかかる時間です。この時間が長くかかる場合には、オブジェクトのサイズが大きい場合があります。オブジェクトのサイズを調べるとよい場合があります。たとえば、最適化されていない画像や圧縮されていない画像が Web ページのパフォーマンスの原因となる場合があります。
- **Start Render** Waterfall タブに示される緑の縦線です。これはブラウザーでユーザーに対してコンテンツ が表示され始める時間を示しています。理論的には、これは Web ページの高さと幅が 0 を超える値となった時間です。
- **Document Complete** ドキュメントの完成イベントが発生した時間です。
- **3xx Result** ―黄色で強調表示されたオブジェクトは、要求またはキャッシュ要求などの、3xx 状態コードが発生したことを示します。
- **4xx Result** ―赤で強調表示されたオブジェクトは、4xx 状態コード、またはエラー コードが発生したことを示します。これは直ちに調査を行う必要があります。

#### Checklist タブ

[Checklist] タブは、サイトにおけるパフォーマンス関連の設定や手順の活用度合いを示す成績表です。 左から右に重要な順に項目が並んでいます。

- 緑のチェックマークは「合格」を示しています。
- 黄色の三角印は「注意」を示しています。



• 赤の X 印は「不合格」を示しています。

	Cache Static	Use a CDN	Combine CSS/JS	GZIP text	Compress Images	Keep-Alive	Cookies	Minify JS	No ET ags	4
http://www.sitecore.net/	77%	0%	60%	100%	73%	100%	0%	76%	6%	
www.sitecore.net - /				✓		✓	<u> </u>		<b>✓</b>	
www.sitecore.net - screen.css	<u> </u>	€3	€3	✓		✓	<b>②</b>		€3	
www.sitecore.net - forms_v3.css	<u> </u>	000	<b>②</b>	<b>√</b>		✓	<b>②</b>		8	
www.sitecore.net - jquery.fancybox-1.3.1.css	1	€3	8	J		V	€3		8	
www.sitecore.net ⋅ ie_common.css	1	<b>3</b>	<b>3</b>	V		V	<b>②</b>		8	
www.sitecore.net - visitor.css.aspx				<b>√</b>		<b>√</b>	<u> </u>		<b>√</b>	
www.sitecore.net - Default.css	<u>^</u>	<b>3</b>	<b>3</b>	<b>√</b>		✓ /	<b>A</b>		8	
www.sitecore.net - Default.css	<u> </u>	€3		<b>√</b>		✓	€3		8	
www.sitecore.net - Custom.css	1	(3)	8	1		1	<b>②</b>		8	
www.sitecore.net - WebResource.axd	7	Ö	Ø.	j j		J	Ø	€3	J	
www.sitecore.net - prototype.is	1	(3	i Ö	J		J	Ö	Ö	8	
www.sitecore.net - Wfm.Controls.js		Č)		j j		J	€3	<b>(2)</b>	(S)	
www.sitecore.net - wfmanalytics.is	<u>^</u>	Ö		J		J	<b>©</b>	8	<u> </u>	
www.sitecore.net - validator.js		Ö		j j		j	Ö	Ö	Ö	- 5
www.sitecore.net - HttpCombiner.ashx	<b>-</b>	ā		j j		j	Ö	ā	J	
www.sitecore.net - ScriptResource.axd	l j	00000000000000000000000000000000000000		<i>)</i>		j.	Ö	Ø Ø	j j	
www.sitecore.net - ScriptResource.axd	1	ā		j j		j	Ö	Ö	1	
www.sitecore.net - ScriptResource.axd     www.sitecore.net - ScriptResource.axd	1 7	ă		<i>"</i>		7	Ö	ă	, j	
www.sitecore.net - flag_belgium.ashx	· /	ă		•	1	1	Ö	_	Ò	
www.sitecore.net - flag_china.ashx	1 2	ă			$\overline{\Lambda}$	7	Ö		ă	
www.sitecore.net - logo.gif	<b>A</b>	ă			7	,	Ö		ă	
www.sitecore.net - flag denmark.ashx	7	×			i i	y,	ä			
www.sitecore.net - fancy close.png	\ \ \ \ \ \	ă			7	y,	Ö		8	
www.sitecore.net - flag germany.ashx	1 <del>'</del>	×			<b>X</b>	٧,	ä		×	
www.sitecore.net - idg_gerinally.darix     www.sitecore.net - icon_searchfield.qif	i i	×			<del>'</del>	· ',	Ö		8	
www.sitecore.net - flag france.ashx	- <del>'</del>	×			X I	· ',	Ö		×	
www.sitecore.net - flag_flafice.asrix     www.sitecore.net - fancy_nav_right.png	i X	×			-	- V,	ä		8	
www.sitecore.net - fancy_nav_ligni.prig     www.sitecore.net - flag_hungary.ashx		×			Y Y	٧,	ä		×	
	- V,	×			4	٧,	ä		×	
www.sitecore.net - flag_japan.ashx	- Y	×			Y Y	√,			×	
www.sitecore.net - flag_netherlands.ashx	<b> </b>	N N			4	- √,	Q		N N	
www.sitecore.net - fancy_nav_left.png	4	×			<b>Y</b>	√,	Q		N W	
www.sitecore.net - flag_norway.ashx	<b>∀</b> ,	×			4	- √,	Ø		N N	
www.sitecore.net - flag_poland.ashx	<b>∀</b> ,	<u> </u>			<b>✓</b>	√,	<b>Q</b>		w w	
www.sitecore.net - flag_russia.ashx	<u> </u>	000000000000000000000000000000000000000			4	√,	<b>Q</b>		80000000000000000000000000000000000000	
www.sitecore.net - fancy_shadow_nw.png	4	₩ ₩			<b>√</b> ,	√,	<b>Q</b>		<b>S</b>	
www.sitecore.net - downarrow.gif	4	₩ ₩			<b>✓</b>	√.	<b>Q</b>		₩ W	
www.sitecore.net - flag_spain.ashx	<b>₹</b>	₩.			4	√.	<b>②</b>		₩.	
www.sitecore.net - flag_sweden.ashx	<b>✓</b>	<b>2</b>			4	√.	<b>②</b>		<b>2</b>	
www.sitecore.net - fancy_shadow_w.png	4	<b>₩</b>			✓ ✓	✓.	<b>©</b>		<b>S</b>	
www.sitecore.net - flag_united_kingdom.ashx	<b>✓</b>	<b>9</b>			₩	✓.	<b>②</b>		₩ 😂	
www.sitecore.net - map.ashx	<u> </u>	₩.			₩	√.	<b>②</b>		₩.	
www.sitecore.net - sidebaricon_facebook.ashx	<u> </u>	₩ 🛂			<b>√</b>	✓.	8		8	
www.sitecore.net - fancy_shadow_sw.png	<u> </u>	Ø			<b>✓</b>	✓	<b>②</b>		<b>②</b>	
www.sitecore.net - sidebaricon_linkedin.ashx	1	<u> </u>			/ /	<b>√</b>	€3		<b>©</b>	

次の表に、チェックリストタブの各項目の対象のオブジェクト、チェックの内容を示します。

Column	Objects	Description 説明
Cache Static	対象オブジェクト	MIME の種類が "text/*"、"*javascript*"、"image/*" であるすべての非 HTML オブジェクトで、0 または -1 の有効期限ヘッダー、または "private"、"no-store"、"no-cache" のキャッシュ コントロール ヘッダー、または "no-cache" のプラグマ ヘッダーを明示的に持たないもの
	チェックの内容	有効期限ヘッダーが存在 (0 または -1 でなく) するか、または "cache-control:max-age" ディレクティブが存在し、1 時間以上 に設定されていること。有効期限が 30 日より短く設定されている場合、注意が表示されます (現時点では max-age のみに適用されます)。
Use A CDN	対象オブジェクト	すべての静的な非 HTML コンテンツ (CSS、js、画像)



Column	Objects	Description 説明		
	チェックの内容	既知の CDN (CNAME が既知の CDN ネットワークにマッピングされている) でホストされているかどうかをチェックします。 既知の CDN とは Akamai、 Amazon CloudFront、 Coral Cache、 Edgecast、 Google、 Highwinds、 Internap、 Limelight、 Mirror Image、 Panther、 Yahoo です。		
Combine CSS/JS	対象オブジェクト	すべての css と javascript オブジェクト		
	チェックの内容	同じ種類の複数のファイルがサービスされている場合、2 つ目以降の各 css ファイルは 5% を差し引き、2 つ目以降の各 Javascript ファイルは 10% を差し引く		
GZIP Text	対象オブジェクト	Mime の種類が "text/*" または "*javascript*" であるすべての オブジェクト		
	チェックの内容	転送エンコードが gzip であることを確認します。そうでない場合、ファイルは圧縮され、圧縮率が結果となります(圧縮によって 30% のサイズを減少可能なページは、テスト結果が 70% となります)。		
Compress Images	対象オブジェクト	Mime の種類が "image/*" であるすべてのオブジェクト		
	チェックの内容	GIF — すべて合格 PNG — 8 ビット以下である必要があります — 24 ビット PNG は合格しません。 JPEG — Photoshop Quality 50 が 10% 以下である場合は合格、50% までは注意、それ以上は不合格となります。 総合スコアは画像の再圧縮により節減可能なバイト数を示すパーセントとなります。		
Keep-Alive	対象オブジェクト	ページの 2 つ以上のオブジェクトをサービスするドメインのすべてのオブジェクト、たとえば、あるドメインが 1 つのオブジェクトのみをサービスする場合には、そのドメインはチェックされません。		
	チェックの内容	応答ヘッダーが "keep-alive" ディレクティブを含むか、または、あるホストで 2 つ以上のオブジェクトで同じソケットが使用される。		
Cookies	対象オブジェクト	すべての要求		
	チェックの内容	静的オブジェクトに cookie を送るすべての要求は不合格となります。 その他の cookie を送る要求は注意を受けます。		
Minify JS	対象オブジェクト	すべての HTML、javascript、json 応答		



Column	Objects	Description 説明
	チェックの内容	Javascript は jsmin によって実行されます。 元のコンテンツが gzip でエンコードされている場合、縮小バージョンも gzip されて比較されます。 節減量が 5 KB よりも大きいか、10% よりも大きい場合、不合格と なります。 節減量が 1 KB よりも大きい場合、注意となります。それ以外は合格 となります。
No Etags	対象オブジェクト	すべての要求
	チェックの内容	応答ヘッダーに ETag ヘッダーを含まれている場合、要求は不合格となります。

#### 不合格と注意を解消するための情報

- [Cache Statics] 列で不合格が表示されている場合、HTTP Expires Web コンテンツ ヘッダーが設定されていません。IIS Expire Web コンテンツ ヘッダーを参照してください。
- [Cache Statics] 列で注意が表示されている場合、HTTP Expires Web コンテンツ ヘッダーで 30 日 よりも少ない値が設定されています。その場合には、IIS Expire Web コンテンツ ヘッダーを参照してください。
- [Combine CSS /JS] 列で不合格が表示されている場合、CSS または JS ファイルはより少数のファイル に組み合わされません。オブジェクトを取得するために複数の要求が実行されています。
- [GZIP Text] 列で不合格が表示されている場合、Web サーバーで静的圧縮が有効化されていません。 その場合には、:IIS 静的コンテンツ圧縮の有効化を参照してください。
- [Keep-Alive] 列で不合格が表示されている場合、Web サーバーで HTTP キープアライブが有効化されていないため、オブジェクトが要求されるたびに接続が起動されています。その場合には、IIS HTTP キープアライブの有効化を参照してください。

## Optimization Report タブ

[Optimization Report] タブでは、ページの読み込み時間に関する一般情報と改善を行うべき箇所 ([Checklist] タブに表示される項目に基づきます) が表示されます。圧縮設定など、どれだけのサイズの節減が可能であるかの推定値が表示される場合もあります。

[Optimization Report] タブは [Waterfall] および [Checklist] タブで表示される内容をテキスト表示したものです。

#### Load Details タブ

[Load Details] タブはすべての要求で発生している詳細な内容の説明です。この情報は特定の要求での問題を 絞り込むために役立ちます。



### 2.2 ページのパフォーマンスの調査 — IIS ログの使用

このページのパフォーマンスの調査のタスクでは、IIS ログを使用して、往復で 4000 ms (4 秒) 以上かかっているページを見つけます。結果は記録され、それを基に、どのレンダリング コンポーネントがパフォーマンス低下の原因となっているのかを特定するための、さらに詳細な調査を行います。

ページのパフォーマンスを直接測定するのでなく、IIS ログの情報を解析することの利点は、地理位置情報を参照できること (地理位置に起因するネットワーク上の問題である場合があります)、また利用のピーク時に発生している問題を知ることができること (容量上の問題である場合があります)です。

#### 2.2.1 必要なスキル

- Log Parser についての実践的な知識
- IIS ログについての実践的な知識

#### 2.2.2 症状

• ページの往復時間の遅延。

#### 2.2.3 パートナー/お客様への確認

問題の発生箇所や読み込みの遅いページの存在場所を見極める最初のステップは、パートナー、お客様、Web サイトのオーナーに、遅いページに気付いているかどうかを確認することです。URI を記録し、パフォーマンス上の問題の原因をさらに調査します。

さらに、Web サイトのピーク利用時間に関する情報を調べます。この情報を使って、Web サイトがピーク時の負荷を処理する容量を備えているかどうか調べます。

#### 2.2.4 長い時間がかかっている要求の IIS ログのクエリの手順

Log Parser のインストールに関する情報は

http://www.microsoft.com/en-us/download/details.aspx?id=24659 を参照してください。

このクエリは、要求された URL を、要求の処理に必要な時間によって降順で並べ替えて参照できます。これによって 調査を行うレンダリングを絞り込むことができます。

ここでは /sitecore/ を含む URL を除外して、Sitecore のクライアント ツールのトラフィックを除外していることに注意してください。

IIS ログのクエリを行って、時間がかかっている要求を探す方法:

- 1. Log Parser がインストールされているディレクトリでコマンド ラインを起動します。
- 2. 次のコマンドを実行します #logs location# および #output file# を変更します。

logparser -i:IISW3C -o:CSV



```
"SELECT TO_TIMESTAMP(date, time) as [Timestamp], cs-uri-stem as [URI], c-ip AS [Client IP], time-taken as [Time], sc-status as [Status]

INTO #output file#

FROM #logs location#

WHERE ((EXTRACT_EXTENSION(URI) = 'aspx' OR EXTRACT_EXTENSION(URI) = '') AND URI NOT LIKE
'%/sitecore%')

AND time-taken > 4000

AND Status = '200'

ORDER BY time-taken DESC"
```

#### 2.2.5 結果を理解する

出力ファイルを Excel で開き、次の 2 つの点で結果を確認します。まず、同じクライアント IP アドレスからの要求で長い時間がかかっている一連の要求です。これを探すためには、Excel の並べ替えまたはフィルター機能を使います。これは社内でのテストを行っている場合があります。または帯域幅の遅い IP アドレスである場合があります。次に、外部からサイトへ来ているトラフィックで同じ結果が生じているものです。

#### 結果の例 1

Timestamp	URI	Client IP	Time	Status
27.01.2011 16:24:40	/Sverige.aspx	88.131.15.18	567,640	200
30.01.2011 08:35:30	/en/Norge.aspx	195.184.101.130	468,644	200
30.01.2011 08:35:30	/Danmark.aspx	195.184.101.130	468,363	200
30.01.2011 08:35:31	/Sverige.aspx	195.184.101.130	467,082	200
30.01.2011 08:35:31	/Sverige.aspx	195.184.101.130	466,504	200
30.01.2011 08:35:32	1	195.184.101.130	451,616	200
30.01.2011 08:35:33	/en/Norge.aspx	195.184.101.130	445,305	200
30.01.2011 08:35:32	/Danmark.aspx	195.184.101.130	444,945	200
30.01.2011 08:35:34	/Sverige.aspx	195.184.101.130	443,945	200
30.01.2011 08:35:33	/Sverige.aspx	195.184.101.130	443,492	200
30.01.2011 08:35:36	/UnitedKingdom.aspx	195.184.101.130	437,650	200
30.01.2011 08:35:36	1	195.184.101.130	437,462	200
30.01.2011 08:35:35	/Danmark.aspx	195.184.101.130	436,869	200
30.01.2011 08:35:35	/UnitedKingdom.aspx	195.184.101.130	436,791	200
30.01.2011 08:35:34	/en/Norge.aspx	195.184.101.130	436,588	20
30.01.2011 08:35:36	1	195.184.101.130	422,122	20

クライアント IP アドレスから地理位置情報が判別できるかどうかを確認するには、http://ip2location.com/1.2.3.4 (1.2.3.4 をクライアント IP アドレスで置換) を行います。たとえば、http://ip2location.com/195.184.101.130 は確認可能です。





#### メモ

これは地理位置に基づくネットワーク上の問題を追跡する場合にも役立ちます。

#### 結果の例 2

Timestamp	URI	Client IP	Time	Status
31.01.2011 10:13:48	/en/Company/Contact/Japan.aspx	173.203.158.156	5,170	200
31.01.2011 11:59:07	/News/RSS/Feeds/Denmark-News.aspx	193.3.234.5	5,155	200
30.01.2011 21:54:23	/Danmark.aspx	195.184.101.130	5,139	200
30.01.2011 11:54:36	/de/Hungary.aspx	78.46.71.246	5,121	200
27.01.2011 10:36:27	/en/Products/Resources/Tours.aspx	65.61.164.180	5,092	200
31.01.2011 10:17:45	/en/Customers/Selected-Customers.aspx	173.203.158.156	5,077	200
30.01.2011 11:57:48	/en/Solutions/Best-CMS-Solutions- Education.aspx	78.46.71.246	5,076	200
28.01.2011 01:09:32	/en/Japan.aspx	166.205.138.71	5,061	200
31.01.2011 10:14:02	/en/Products.aspx	173.203.158.156	5,046	200
29.01.2011 10:47:10	/en/Partners.aspx	65.61.143.45	5,030	200
27.01.2011 07:33:24	/products/resources/whitepapers/gartner-magic- quadrant.aspx	202.155.14.116	5,030	200
31.01.2011 10:14:10	/en/Partners/Hosting-Partners.aspx	173.203.158.156	4,999	200
31.01.2011 10:17:50	/en/Customers/Selected-Customers.aspx	173.203.158.156	4,983	200
28.01.2011 10:37:16	/en/Products/Industry-Commentary.aspx	64.39.4.224	4,983	200
29.01.2011 10:50:27	/en/Customers.aspx	65.61.143.45	4,983	200
31.01.2011 10:16:31	/Japan.aspx	173.203.158.156	4,983	200
30.01.2011 11:58:36	/en/News/NewsAndEvents.aspx	78.46.71.246	4,982	200
30.01.2011 11:56:21	/en/Company/Contact.aspx	78.46.71.246	4,966	200
28.01.2011 15:37:03	/Sverige/Partners/KnowlT.aspx	67.195.37.153	4,952	200
27.01.2011 10:35:45	/en/Partners/Become-Partner.aspx	65.61.164.180	4,952	200

テスト結果によって、または地理位置によるネットワーク上の問題によって、URI を識別した後で、残りの情報を記録します。

多くの URI でしきい値 (今回の場合、4 秒) を超えている場合、それが利用のピーク時に発生している場合には、 Web サイトが負荷を処理するための十分な容量を備えていないことを示す場合があります。

## 2.2.6 Sitecore の推奨事項

Sitecore はすべての .aspx ページの往復時間が 4 秒以下となるようにすることを推奨します。



## **Report Findings**

	_	_		
ᇨᆂ	ш	т	報	ᇁ
πп	ᅏ	v	ŦΙΥ	

心木の取口					
Web サイトの利用の	のピーク時間帯: _				
往復時間が 4 秒を 使用します。	超えた URI をすべ	て記録します。これらの URL	はレンダリング コン	ポーネントの分析のタス	クで
タイムスタンプ	URI	クライアント IP	状態コード	時間	
4 秒のしきい値を超	えた要求があります	-。 はい いいえ			
利用のピーク時間帯	きで、多くの URL か	「あります。 はい いいえ			
4 秒のしきい値を超	えた要求があります	ー しいいえ			
問題ありません。往	<b>注復で 4 秒を超え</b> た	こ要求はありません。			
4 秒のしきい値を超	えた要求があります	- = (はい			
   エラーです。往復で	:4 秒を超えた要求	はがあります。さらに調査が必要	要です。レンダリング	コンポーネント パフォー	マン
スのタスクを参照し	てください。				

利用のピーク時間帯で、多くの URL があります = はい

エラーです。利用のピーク時間帯で、多くの URI で 4 秒のしきい値を超えています。これは容量上の問題である場合があり、さらに調査が必要です。



## 2.3 レンダリング パフォーマンス

Sitecore の統計情報ページを使ってレンダリングに関する情報を収集することができます。

統計情報ページでは、(統計情報ページを参照する場合に応じて) ページまたはサイト全体で使用されているさまざまなレンダリングについて、次のような情報を提供します。

- Rendering レンダリングの名前。
- Site レンダリングの情報が収集されるサイトの名前。
- Count 統計情報ページが最後にリセットされてから、レンダリングが呼び出された回数。
- From cache レンダリングがキャッシュから取得された回数。
- Avg. time (ms) レンダリングが出力されるまでにかかった平均時間。
- Avg. items レンダリングに含まれたアイテムの平均の数。
- Max. time 出力をレンダリングするためにかかった最大時間。
- Max. items レンダリングに含まれたアイテムの最大数。
- Total time 統計情報ページが最後にリセットされてから、このレンダリングのすべてのインスタンスにかかった 総時間。
- Total items 統計情報ページが最後にリセットされてから、このレンダリングのすべてのインスタンスに含まれたアイテムの総数。
- Last run 最後に統計情報が収集された時間。

#### 2.3.1 必要なスキル

• Sitecore の stats.aspx ページについての実践的な知識。

#### 2.3.2 症状

Web ページの往復時間の遅延。

#### 2.3.3 Sitecore の統計情報ページの使用手順

この手順を行うには、/sitecore/admin フォルダーの aspx ページへのアクセス許可が必要です。また、統計情報ページは別のタブまたは別のブラウザーのウィンドウで開くことを推奨します。それにより、他のタブやウィンドウを使って、調査するサイトを移動することができます。

統計情報ページを起動する方法:

- 1. 2 つの Web ブラウザーまたはタブを起動します。ここでは統計情報ウィンドウとサイトウィンドウと呼びます。
- 2. 統計情報ウィンドウで次の場所に移動します。http://<site>/sitecore/admin/stats.aspx.



3. **統計情報**ウィンドウには、統計情報ページが最後にリセットされたとき以降に要求されたレンダリングの情報が表示されます。

これをリセットする場合には、更新ボタンをクリックします — たとえば、個別の Web ページの情報を参照する場合には、統計情報ページをクリアする必要があります。

- 4. 最後にリセットされたとき以降に収集されたレンダリングの統計情報を表示する場合には、表示されている情報を使用します。
- 5. 1 つのページに関する情報を収集する場合には、統計情報ページをリセットします。**サイト**ウィンドウで、含まれているレンダリングについての統計情報を収集する Web サイトまたは Web ページに移動します。

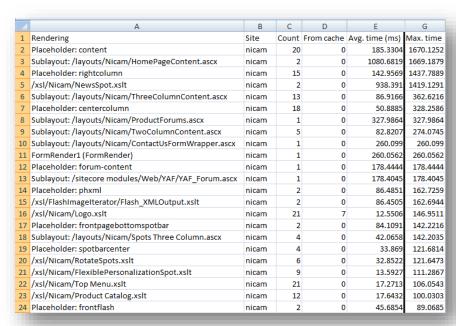
#### メモ

キャッシュと平均についての情報を収集するには、ページの呼び出しを数回行います。

6. 統計情報ページの表の情報を Excel にエクスポートして、情報を並べ替えて、データを容易に分析できるようにすることを推奨します。情報をエクスポートするには、表を右クリックして、[Export to Microsoft Excel] をクリックします。

#### 2.3.4 結果を理解する

次に示す統計情報の表は、Excel にエクスポートを行い、Max. time で並べ替え、いくつかの列を非表示にしたものです。



この表から次の考察を得ることができます。

100 ms を超える Max time となっているレンダリングをすべて調査し、推奨のコーディング手法に沿っているかどうかを確認する必要があります。



• From cache 列が 0 となっているものは、レンダリング (HTML 出力) のキャッシュが構成されていないことを示しています。

## 2.3.5 Sitecore の推奨事項

Sitecore はレンダリングの Max time を 100 ms より小さくすることを推奨します。またレンダリング (HTML 出力) のキャッシュを有効化して構成することを推奨します。



#### 結果の報告

#### **Record the Results**

レンダリングの Max times は 100 ms よりも小さい。\_\_はい \_\_いいえ From cache の値が 0 のものがいくつかある。\_\_はい \_\_いいえ

レンダリングの Max times は 100 ms よりも小さい = はい

問題ありません。 Max time が 100 ms を超えているレンダリングはありません。 これは Sitecore の推奨事項に沿っています。

レンダリングの Max times は 100 ms よりも小さい = いいえ

エラーです。Max time が 100 ms を超えているレンダリングがあります。Sitecore はレンダリングの Max times を 100 ms より小さくすることを推奨します。

There are several *From cache* values equal to 0 = NO:

問題ありません。レンダリング (HTML 出力) のキャッシュを有効化して構成することは Sitecore の推奨事項です。

From cache の値が 0 のものがいくつかある = いいえ

エラーです。レンダリング (HTML 出力) のキャッシュが有効化されていないか、または構成されていません。 Sitecore はレンダリング (HTML 出力) のキャッシュが有効化して構成し、サイトのパフォーマンスを向上させることを推奨します。 レンダリングのキャッシュの有効化と構成に関する詳細な情報は、『キャッシュ設定ガイド』および『プレゼンテーション コンポーネント リファレンス』 のマニュアルを参照してください。



## 2.4 Sitecore ログを使用したメモリ リークの調査

Sitecore システムは Sitecore のログ ファイルに 10 分間隔でログを取得するパフォーマンス カウンターを提供しています。

- Process\Private Bytes
- Process\Virtual Bytes
- Process\Page File Bytes
- .net CLR Memory\# Bytes in all Heaps
- .net CLR Memory\% Time in GC
- .net CLR Memory\Large Object Heap size
- .net CLR Loading\Bytes in Loader Heap
- .net CLR Loading\Current Assemblies

Sitecore パフォーマンス カウンターがインストールされていない場合には、下記からダウンロードすることができます。 http://sdn.sitecore.net/upload/sdn5/faq/administration/sitecorecounters.zip.

このセクションではカウンターを使った問題の識別と解決方法について説明します。
<a href="http://sdn.sitecore.net/Scrapbook/Working%20with%20Sitecore%20Performance%20Counters.aspx">http://sdn.sitecore.net/Scrapbook/Working%20with%20Sitecore%20Performance%20Counters.aspx</a>
このタスクで使用する 2 つのカウンターは Process¥Private Bytes および .net CLR Memory¥# Bytes in all Heaps です。

Process¥Private Bytes カウンターは、プロセス — w3wp.exe — に排他的に割り当てられ、システムの他のプロセスと共有を行うことができないすべてのメモリについてレポートします。また、.net CLR Memory¥# Bytes in all Heaps カウンターは、Gen0、Gen1、Gen2 および大きなオブジェクト ヒープのすべてを合わせたサイズをレポートします。

通常、Private Bytes と # Bytes in all Heaps は、同じ割合で増減します。Private Bytes が増加し、# Bytes in all Heaps が安定している場合、アンマネージ メモリがリークしています。両方が増加しており、クリアされない場合、マネージ メモリがリークしている可能性があります。

Sitecore ログ ファイル、Sitecore ログ アナライザー、および Excel を使用すると、これらの 2 つのカウンターのグラフを作成して、リークの可能性を探すことができます。

#### 2.4.1 必要なスキル

- Sitecore ログについての実践的な知識。
- Sitecore ログ アナライザーについての実践的な知識。
- Microsoft Excel を使ってグラフを作成する実践的な知識。

#### 2.4.2 症状

OutOfMemory 例外



- IIS アプリケーション プール リサイクル
- メモリ使用の増加によるパフォーマンスの低下。

## 2.4.3 Sitecore ログ アナライザーを使った Sitecore ログの解析

Sitecore ログ アナライザーのインストールと設定に関する情報は、

http://sdn.sitecore.net/Resources/Tools/Log Analyzer.aspx を参照してください。

このタスクでは、ログ ファイルから必要なヘルス モニター カウンターの値を取得し、それを CSV ファイルにエクスポートします。

エクスポートした結果を Excel で開き、グラフを作成して比較します。

#### Process¥Private Bytes および # Bytes in all Heaps カウンターの値の取得

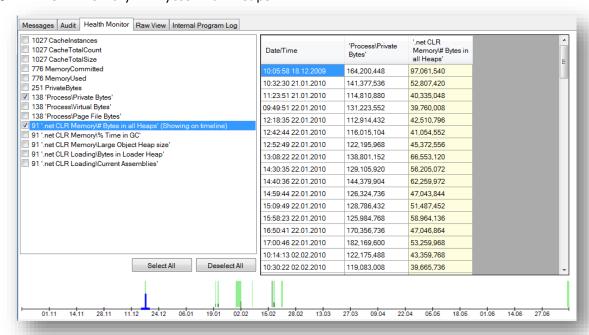
**Sitecore ログ アナライザー**の [**ヘルス モニター**] タブを使うと、解析されたログ ファイルのすべてのヘルス モニター カウンターを参照して便利なフォーマットにエクスポートすることができます。

Date/Time	'Process\Private Bytes'	'.net CLR Memory\# Bytes in all Heaps'
10:05:58 18.12.2009	164,200,448	97,061,540
10:32:30 21.01.2010	141,377,536	52,807,420
11:23:51 21.01.2010	114,810,880	40,335,048
09:49:51 22.01.2010	131,223,552	39,760,008
12:18:35 22.01.2010	112,914,432	42,510,796
12:42:44 22.01.2010	116,015,104	41,054,552
12:52:49 22.01.2010	122,195,968	45,372,556
13:08:22 22.01.2010	138,801,152	66,553,120
14:30:35 22.01.2010	129,105,920	56,205,072
14:40:36 22.01.2010	144,379,904	62,259,972
14:59:44 22.01.2010	126,324,736	47,043,844
15:09:49 22.01.2010	128,786,432	51,487,452
15:58:23 22.01.2010	125,984,768	58,964,136
16:50:41 22.01.2010	170,356,736	47,046,864
17:00:46 22.01.2010	182,169,600	53,259,968
10:14:13 02.02.2010	122,175,488	43,359,768
10:30:22 02.02.2010	119,083,008	39,665,736

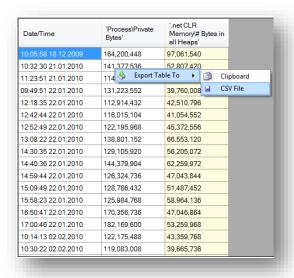
- 1. Sitecore ログ アナライザーを起動します。
- 2. 分析するログを選択します。
- 3. [分析/更新] をクリックします。



4. [ヘルス モニター] タブに移動し、必要なカウンターを選択します。— Process¥Private Bytes および .NET CLR Memory¥# Bytes in all Heaps



- 5. 表で右クリックし、[表形式にエクスポート] > [CSV ファイル] を選択します。
- 6. csv ファイルに適切な名前を付けて、Excel で開きます。

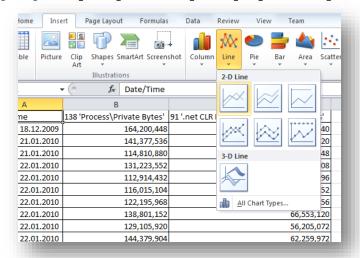


#### 結果のグラフの作成

**Excel** でグラフを作成するには、いくつかの方法があります。ここでは、折れ線グラフを作成して、Process¥Private Bytes の結果を .net CLR Memory¥# Bytes in all Heaps と比較する手順を説明します。



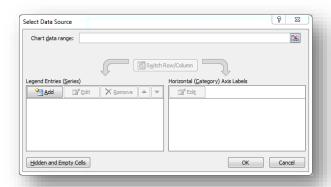
1. In Excel, on the Insert tab, in the Charts group, click Line and then click 2D Line. Excel の [挿入] タブの [グラフ] グループで、[折れ線] 、[2D 折れ線] とクリックします。



2. [データ] グループで、[データの選択] をクリックします。

シート上で選択されているものがある場合には、Excel でそのグラフが作成されます。

3. グラフをクリアするには、すべての系列を削除し、グラフ データの範囲をクリアします。[**データ ソースの選択**] ダイアログ ボックスが表示されています。

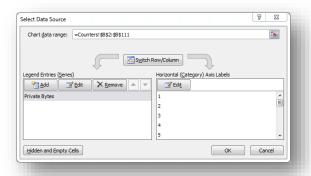


- 4. 「追加」をクリックします。
- 5. 最初の系列では、系列名を「**Private Bytes**」とし、データ範囲として Process¥Private Bytes の列を 選択します。 — 列 B





6. 横軸 (x 軸) のラベルを **Timestamp** 列 (列 A) に設定し、[**編集**] をクリックして Sheet 1 から列 A を選択します。





7. **[OK**] をクリックします。

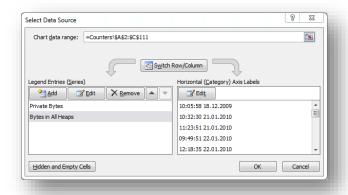


8. [**追加**] をクリックして、同様に 2 つ目の系列を設定します — Bytes in all Heaps。データ範囲として .net CLR Memory¥# Bytes in all Heaps 列を選択します —列 C。

横軸 (x 軸) のラベルを再度設定する必要はありません。

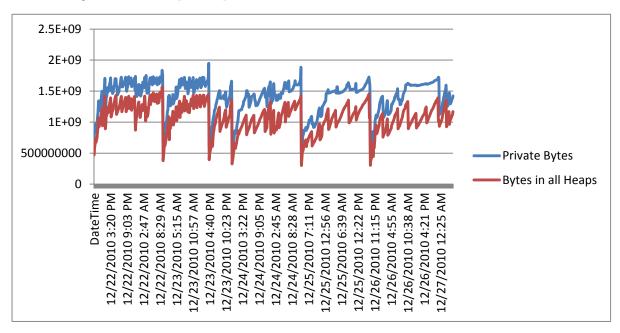






9. **[OK]** をクリックします。

## 2.4.4 結果の理解(グラフ)



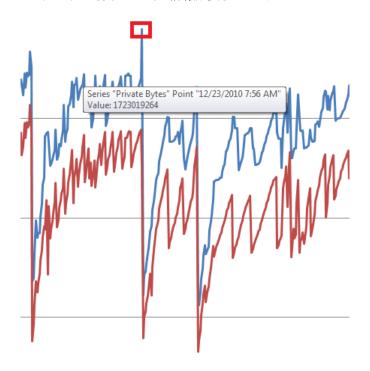
#### Graph 1

- このグラフはマネージ メモリにおけるメモリリークの例を示しています。5 つの大きな落下は、莫大な Sitecore キャッシュのクリアまたはアプリケーション プールの再起動を示しています。これはアプリケーション プールのプライベート メモリが制限に達したか、または OOM 例外によるものです。れぞれの落下の間では、Private bytes および Heap bytes が並行して増加しています。これはメモリリークによる症状ですが、メモリ使用量の増加とキャッシュのクリアを要求するサーバー上において、ロードが増加する現象に対する一般的なリアクションです。
- この例では、Private Bytes と Bytes in all Heaps の両方が同様の割合で上下動しています。メモリがある程度まで上昇して、その状態を保っている場合には問題ありません。



この例では、グラフ上でメモリのリセットが 5 回発生していることを示しています。さらに調査を行う方法については、この後で説明します。

- もし、Private Bytes が上昇し、Bytes in all Heaps に変化がない場合には、アンマネージ メモリでリークが発生している可能性があります。 さらに調査が必要です。
- それぞれのピークは、メモリの割り当てを示しています。マウスをピーク上に移動すると、割り当てられたメモリの 量とイベントの時間についての情報が表示されます。

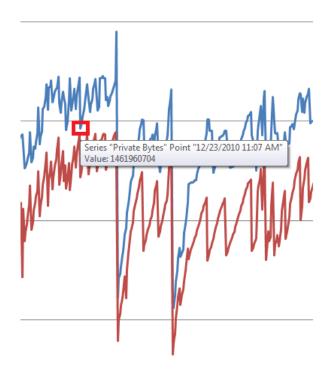


#### グラフ 2

Private Bytes の急上昇は、アプリケーション プールのリサイクルまたは OOM 例外の前の、メモリ使用のピークを示しています。



それぞれの谷間は、ガベージ コレクション イベントが発生したことを示しています。マウスを谷間の上に移動すると、割り当てられているメモリの量とイベントが発生した時間についての情報が表示されます。



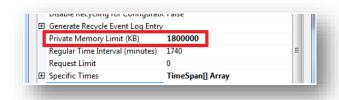
### グラフ 3

- 谷間、つまりガベージ コレクション イベントの詳細を示しています。これはアプリケーション プールのリサイクルまたは OOM 例外ではなく、ガベージ コレクターの正常な活動です。
- ピークの最高値と谷間の最低値を参照し、この情報を Windows のイベント ログから取得できる情報と併せて分析します。
- メモリがリセットされる直前のピークを、アプリケーション プールに設定されているプライベート メモリ制限と比較 します — IIS マネージャーで適切なアプリケーション プールを選択し、[詳細設定] を選択します。

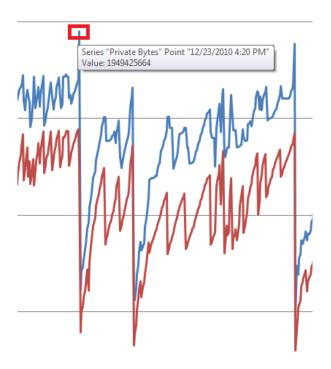
#### УŦ

アプリケーション プールのプライベート メモリ制限に値が設定されていない場合、Sitecore ログをチェックして、この時間帯に OutOfMemory 例外が発生していないかどうかを確認します。

下記の画像では、メモリが 1800000 KB に達したときに、アプリケーション プールがリセットされていることが わかります。





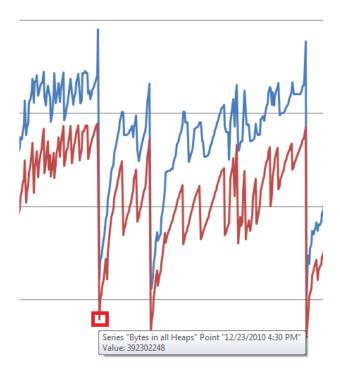


#### グラフ 4

- アプリケーション プールのリサイクルまたは OOM 例外の直前のメモリの急上昇を示しています。この情報を使って、使用メモリがアプリケーション プールが使用できるメモリを超過したかどうかを判別することができます。
- ピークの後の底を Windows イベント ログの情報と比較することにより、アプリケーション プールのリサイクルが 実際に発生していることがわかります。 Private Bytes と Bytes in all Heaps の両方が一定に増加し、 そして設定されたメモリ制限を超過し、アプリケーション プールのリサイクルが発生していることは、マネージ メモ

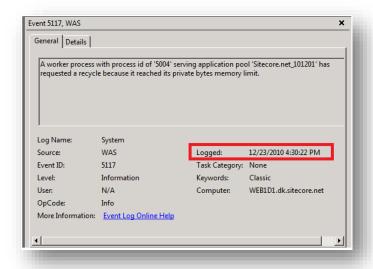


リでのリークが発生している可能性を示しています。さらに調査を行う必要があります。



#### グラフ 5

• メモリ リセット イベントの詳細を示します。この日時を使って、Sitecore ログおよび Windows イベント ログ から取得できる情報と併せて分析し、OOM 例外またはアプリケーション プールのリサイクル イベントが発生しているかどうかを調べます。



タイムスタンプが一致している場合には、アプリケーションプールがリサイクルされたことを示します。



#### 2.4.5 注:

.net のメモリ使用とリークに関する調査を行うための詳細な情報は、下記を参照してください。

- http://msdn.microsoft.com/en-us/magazine/cc163491.aspx
- <a href="http://msdn.microsoft.com/en-us/library/Ee817660(pandp.10).aspx">http://msdn.microsoft.com/en-us/library/Ee817660(pandp.10).aspx</a>



# 2.5 Sitecore パイプライン プロファイリング

Sitecore パイプライン プロファイリング ページを使うことによって、主な Sitecore パイプライン操作に関連するメトリック を監視し収集することができます。 パイプライン プロファイリングが有効な場合は、

Sitecore.Pipelines.CorePipeline.Run()メソッドで起動されたパイプラインのみがプロファイリングの対象となります。

# 2.5.1 必要なスキル

• Sitecore パイプラインとその使用方法についての実践的な知識。

# 2.5.2 Sitecore パイプライン プロファイリング ページの統計情報ページの使用手順

この手順を行うには http://<site>/sitecore/admin ディレクトリの aspx ページへのアクセス許可が必要です。

パイプライン プロファイリングを有効にするには、web.config ファイルで Pipelines.Profiling.Enabled 設定を true に設定するか、または App\_Config\Include\Sitecore.PipelineProfiling.config インクルード ファイルを使用します。

<setting name="Pipelines.Profiling.Enabled" value="true" />

以下で Sitecore パイプライン プロファイリング ページを閲覧します:

http://<site>/sitecore/admin/pipelines.aspx

### 2.5.3 概要

パイプライン プロファイリングを無効化にすると、ページにエラーメッセージとプロファイリングを有効にする方法が表示されます。

### **Pipeline Profiles**

A snapshot from the pipeline profiler.

Pipeline profiling is disabled. No data is currently available.

To enable pipeline profiling, in the 'web.config' file, set the value of the 'Pipelines.EnableProfiling' setting to 'True'.



パイプラインプロファイリングが有効になると、ページにパイプラインと使用統計による最新の使用プロッセサのリストが表示されます。



# 2.5.4 Usage 使用

パイプライン プロファイリングには、[Refresh] と [Reset] の 2 つのボタンがあります。

最新のプロファイルカウンター値のスナップショットを復元し、ページをレンダリングするには、[Refresh] をクリックします。

パイプライン プロファイリングをリセットするには、[Reset] をクリックします。[Reset] をクリックした後は、ページを更新して更新済みのスナップショットを取得する必要があります。

Pipeline Profiles A snapshot from the pipeline profiler.	
Refresh	

# 2.5.5 結果の理解

### メモ

パイプライン プロファイリング ページで [Refresh] をクリックをするたびに、特定のパイプラインが実行されます。従って、これらのパイプライン (たとえば、httpRequestBegin、 preprocessRequest、 renderLayout、getTranslation) の結果に影響を及ぼします。

パイプラインプロファイリングページには、Sitecore パイプラインについての以下の情報が含まれます。

- # Executions 列には、プロファイラーのライフタイム中にパイプラインまたはプロセッサーが実行された回数が表示されています。
- % Wall Time 列はプロセッサーのみに該当します。プロセッサーの実行時間と、プロセッサーを起動したパイプラインの実行時間との比率が表示されています。列の値を参照して、パイプラインの実行時に最も時間を費やすプロセッサーを測定できます。



- Wall Time 列には、プロファイラーのライフタイム中におけるパイプラインまたはプロセッサーのそれぞれの総実 行時間が表示されています。列の値は累積時間で、この値を参照して主に平均実行時間を計算できます。
- Max Wall Time 列には、パイプラインまたはプロセッサー実行時の最大経過時間が表示されています。列の値は、プロファイラーの寿命中における最大実行時間です。この値を参照して、平均実行継続時間と最高実行継続時間を比較することができます。
- **% CPU** 列はプロセッサーのみに該当します。プロセッサーと、プロセッサーを起動したパイプラインによる CPU 使用時間の比率が表示されています。

#### УŦ

% **CPU** メトリックは、Pipelines.Profiling.MeasureCpuTime 設定がweb.config ファイル (またはコンフィグを含む)で true に設定されている場合にのみ計測され表示されます。
<setting name="Pipelines.Profiling.MeasureCpuTime" value="true" />

Time / Execution 列には、プロファイラーのライフタイム中における パイプラインまたはプロセッサーのそれぞれの平均実行時間が表示されています。列の値を参照して、特定の期間(たとえばロードまたはパフォーマンステスト中) におけるパイプラインとプロセッサーを監視することができます。

(execution/wall time 列の) パフォーマンの低いプロセスの上位3つは、数字のアイコンが示されます。

preprocessRequest	1137		308.20	16.30	0.27		0.01<
Site core. Pipelines. Preprocess Request. Suppress Form Validation. Process	1137	1.49	4.59	0.05	0.01<	2.24	0.01<
2 Sitecore.Pipelines.PreprocessRequest.NormalizeRawUrl.Process	1137	6.40	19.73	7.91	0.02	5.86	0.01<
Sitecore.Pipelines.PreprocessRequest.IIS404Handler.Process	1137	0.85	2.60	0.01	0.01<	1.20	0.01<
Isitecore.Pipelines.PreprocessRequest.WebDAVCustomHandler.Process	1137	86.40	266.29	7.96	0.23	85.35	0.01<
Sitecore.Pipelines.PreprocessRequest.FilterUrlExtensions.Process	1137	3.93	12.12	0.31	0.01	4.25	0.01<
Sitecore.Pipelines.PreprocessRequest.StripLanguage.Process	468	0.93	2.87	0.05	0.01<	1.10	0.01<
I w laws :	4407		2.00	0.00			

プロセッサーの名前には 70 文字以上を含むものもあります。結果の表でプロセッサーの名前の長さを制限するために、 プロセッサー名の上にマウスを移動させるとプロセッサーのフルネームがツールチップに表示されます。

互換性と検索のニーズのために、プロセッサーの情報タグには以下の属性が含まれます。

- Title (互換性) と pname (検索) 属性 にプロセッサーのフルネームが含まれます。
- alt にはツールチップ(html) 用に追加のプロセッサー情報が含まれます。



Sitecore.Automation.MarketingAutomation.Pipelines.LoadVisitor.LoadA	Site core. Automation. Marketing Automation. Pipelines. Load Visitor. Load Automations. Processing a contract of the processing and the processi
excludeRobots	% Wall Time: 1.20 Wall Time: 2.447758
2 Sitecore.Analytics.Pipelines.ExcludeRobots.TryObtainCachedResult.Pr	Max Time: 0.071993  Avg Time: 0.071993
Site core. Analytics. Pipelines. Exclude Robots. Check User Agent. Process	% CPU cycles: 3.09
I Sitecore.Analytics.Pipelines.ExcludeRobots.CheckIpAddress.Process	Time / call: 0.071993 CPU cycles: 6.32 G
Sitecore.Analytics.Pipelines.ExcludeRobots.AddResultToCache.Process	CPU cycles / call: 185.87 M



# 2.6 Sitecore デバッガー

Sitecore にはブラウザー ベースのデバッガーがあり、これによって不完全または速度の遅いプレゼンテーション コンポーネントに伴う問題を見つけるのに役立ちます。開発者はデバッガーを使うことによって、ページ全体と、ひとつのページ上にある個々のプレゼンテーション コンポーネントの両方における出力を分析しやすくなります。

# 2.6.1 Sitecore デバッガーの起動手順

この手順を行うには次のことが必要です。

Sitecore ユーザーは、Core データベースの次のアイテムへの読み取りアクセス権が必要です:

/sitecore/content/Documents 及び settings/All users/Start menu/Right/Debug

- 現行の <site> の allowDebug 属性を web.config ファイル(またはコンフィグを含む) で true に設定する必要があります。
- 現行の <site> の enableDebugger 属性を、web.config ファイル(またはコンフィグを含む) で true に設定する必要があります。

Sitecore デバッガーを起動する方法:

- 1. Sitecore デスクトップにログインします。
- 2. Sitecore をクリックしてから [デバッグ] をクリックします。

新しいブラウザー ウィンドウまたはタブが開き、上部にデバッグのリボンを持つ現行の Web サイトのホームページが表示されます。リボンの右側上のボタンを使って、デバッグのリボンを表示または非表示にできます。





# 2.6.2 使用

Sitecore デバッグのリボンの [ブラウザ表示] グループで [編集] をクリックすると、インライン編集を有効または無効にできます。また、[プレビュー] と [デバッグ] モードの切り替えも可能です。



[プロファイル] グループで、現行ページの Sitecore プロファイリングを有効または無効にできます。 Sitecore プロファイリングが起動されている場合は、現行ページのプロファイル レポートの保存またはダウンロードも可能です。

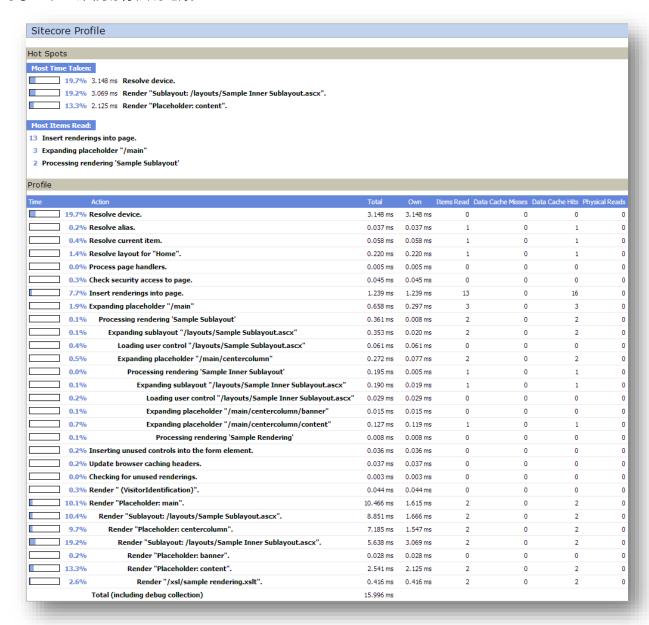


プロファイル レポートはページ操作についての高度な要約です。プロファイル レポートを使ってページ内のパフォーマン性の低いコンポーネントと、それらのコンポーネント内でパフォーマンス性の低い処置を特定することができます。

Sitecore プロファイリングが起動されると、ページの最後にプロファイル レポートが表示されます。プロファイル レポートは、最もパフォーマンス性の低いコンポーネントや多くのアイテムにアクセスしたコンポーネントなど、重要箇所を特定しようとします。 たとえば、プレゼンテーション コンポーネントを存在しないプレースホルダーに追加するという指示がレイアウト詳細にある場合は、プロファイル コンポーネントはエラーメッセージを表示します。 そうでなければ、プロファイル レポートに



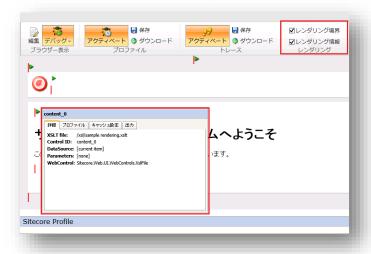
### よるページへの出力は行われません。



[レンダリング] グループで、[レンダリング境界] と [レンダリング情報] を有効または無効にすることができます。[レンダリング境界] オプションを有効にすると、ページに各プレゼンテーション コンポーネントをアウトライン化する罫線が表示されます。[レンダリング情報] オプションを有効にすると、ページ上の各プレゼンテーション コンポーネントに情報アイコン (緑色の三角形) が表示されます。



マウスを情報アイコン上に移動させると、ページにプロファイル、キャッシュの設定と出力を含む、個々のプレゼンテーション コンポーネントのインラインに関する情報が表示されます。



### УŦ

[レンダリング情報] チェックボックスをオンにすると、Sitecore はページのリクエストに応じて、各プレゼンテーション コンポーネントの出力をキャッシュを使用せずに行います。プレゼンテーションコンポーネントのキャッシュをデバッグする場合は、「レンダリング情報] チェックボックスをオフにします。

[トレース] グループで、ページの Sitecore トレースを有効または無効にできます。 Sitecore トレースが有効な場合は、ページのトレース レポートの保存またはダウンロードが可能です。 トレースは、ページの生成プロセスに関する手順を簡単に説明したものです。



Sitecore トレースが起動している場合、ページの最後にトレース レポートが表示されます。トレース レポートを使って、各プレゼンテーション コンポーネントの出力がキャッシュから行われたか否かを測定できます。また、コンポーネントがペー



ジに追加された順番と、各コンポーネントをレンダリングするのに要した時間も同様に測定できます。

cecore Trace	Elapsed since last entry	Elapsed since
Starting trace.	0.00 ms	0.0
Current site is "website".	0.01 ms	0.0
Current domain is "extranet".	0.01 ms	0.0
Current language is "en".	0.02 ms	0.0
Profiling is active.	0.01 ms	0.0
Device set to "Default".	2.31 ms	2.3
Current item is "/sitecore/content/home/".	4.30 ms	6.6
Current layout is "Sample Layout".	0.64 ms	7.3
Using physical layout "/layouts/Sample Layout.aspx".	0.03 ms	7.3
Checking security for current user "sitecore\admin".	2.29 ms	9.6
Access granted as the current user "sitecore\admin" has read access to current item.	0.03 ms	9.6
Inserting rendering "Sample Sublayout" into page.	1.25 ms	10.9
Inserting rendering "Sample Inner Sublayout" into page.	0.02 ms	10.9
Inserting rendering "Sample Rendering" into page.	0.01 ms	10.9
Adding Http headers to disable caching.	2.71 ms	13.6
Adding Http header to indicate last modification. Date: 6/18/2009 12:53:23 PM.	0.02 ms	13.6
Starting rendering " (VisitorIdentification)".	1.80 ms	15.4
Elapsed time since start render start: 0.042 ms.	9.65 ms	25.1
Collected debug information	3.52 ms	28.6
Finished rendering " (VisitorIdentification)" .	0.61 ms	29.2
Starting rendering " (WebEditRibbon)".	0.11 ms	29.3
Elapsed time since start render start: 150.586 ms.	150.60 ms	179.9
Finished rendering " (WebEditRibbon)" .	0.04 ms	179.9
Starting rendering "Placeholder: main".	0.01 ms	179.9
Starting rendering "Sublayout: /layouts/Sample Sublayout.ascx".	1.83 ms	181.8
Starting rendering "Placeholder: centercolumn".	1.55 ms	183.3
Starting rendering "Sublayout: /layouts/Sample Inner Sublayout.ascx".	1.61 ms	184.9
Starting rendering "Placeholder: banner".	1.48 ms	186.4
Elapsed time since start render start: 0.015 ms.	1.49 ms	187.9
Collected debug information	0.07 ms	188.0



# **Chapter 3**

# 検索診断

この章で説明するツールと対応は、Sitecore 検索に関連する問題を特定しトラブルシューティングに役立たせることを目的としています。テクニックを使うことで、検索操作の向上に関する実践的な情報を活用することもできます。

この章には次のセクションが含まれます。

- FillDB
- 冗長なログ
- LinqScratchPad



### 3.1 FIIIDB

FillDB は、Sitecore 開発者がサンプルデータを使って簡単にデータベースを作成することが可能なツールです。

このセクションは、CMS tuning and diagnostics の補足説明です。情報の生成前後に Sitecore CMS のチューニングを行い、情報の生成前後に診断手順を行うことによって、サイズの異なるデータセット全体においてパフォーマンスのメトリックを比較することができます。 開発者とシステム管理者は、これらのメトリックを使って、ソリューションまたはシステムアーキテクチャの潜在的なボトルネックを特定することができます。

# 3.1.1 FillDB Page を使ったアイテムの作成

この処置を行うには、http://<site>/sitecore/admin フォルダーの aspx を閲覧する許可が必要です。
Sitecore では、FillDB ページを使って、実働データベースにサンプル アイテムを作成することはお勧めしていません。サンプル アイテムには、テスト データベースまたは一時的なデータベースを使用する必要があります。

FillDB ページで生成されたサンプル アイテムは、Sample Item テンプレートに基づきます。このテンプレートはスタンダードな Sitecore インストレーションにあり、テンプレート内で Title と Text を使用します。デフォルトの Folder テンプレートは、サンプル アイテムが保存されている階層を作成するために使用されます。これらのテンプレートは、FillDB プロセスの使用前から Sitecore インスタンスに存在しています。

### FillDB を使用する方法:

- 1. ItemGenerator.sql ファイルを、Web サイトの /sitecore/admin/SqlScripts ディレクトリに配置します。
- 2. SQL スクリプトを、サンプル アイテムを挿入する Sitecore Master データベースに対して実行します。
  Web サイトのルート ディレクトリで data という名前のディレクトリを作成し、次にその data ディレクトリ内に words というディレクトリを作成します— c:\sitename\Website\data\words。
- 3. data\words ディレクトリに、大きなテキスト ボディを持つ.txt ファイルをいくつか配置します。

### 果

プロジェクトグーテンベルクは、ブックのフリーテキストベースのバージョン (http://www.gutenberg.org/)をダウンロードする際に有用なリソースです。

- 4. FillDBページを閲覧します: http://<site>/sitecore/admin/FillDB.aspx
- 5. [Parent Guid] フィールドに、サンプル アイテムを作成するコンテンツツリーのアイテム ID を入力します。そのフィールドに提供されているデフォルト値は Sitecore ホーム アイテム (/sitecore/content/Home) に付属しています。

#### ΥE

FillDB ページで生成されるサンプル アイテムは、ブランチに 100 未満のアイテムがある、自動的に生成されるフォルダーとアイテムの階層に配置されます。



- 6. [Database Name] フィールドに、サンプル アイテムを挿入する Sitecore データベースの名前を入力 します。この名前は、web.config ファイルの [データベース] セクションで定義されたデータベースの id 属性とマッチングする必要があります。このフォルダーのデフォルト値は、master です。
- 7. [Number of items] フィールドに作成するサンプル アイテムの数を指定します。
- 8. [Go!] をクリックし、アイテムを作成します。

サンプルアイテムが作成された後、そのアイテムを作成するのにかかった時間を示すメッセージが表示されます。

9. アイテムが作成されると、Web サイトのキャッシュをクリアにします。

Web サイトのキャッシュをクリアにする方法:

o Sitecore 管理者のキャッシュページを閲覧します:

http://<site>/sitecore/admin/cache.aspx:

このページ上で、[Clear all] をクリックして Sitecore キャッシュのすべてをクリアにします。

または

○ IIS をリセットします。

または

○ この Web サイトのアプリケーション プールをリサイクルします。



# 3.2 冗長なログ

冗長な検索ログは、検索のインデックス構成を容易にし、検索のトラブルシューティング シナリオを見識するのに役立ちます。たとえば、インデックスされていないアイテムがある場合、VerboseLogger はさらにコンテキストを提供し、その問題の解決を促進します。

### 重要

VerboseLogger コンポーネントは特別な事情がある場合にのみ有効にします。実働環境において長期間使用することはできません。そうしなければ、莫大なログファイルが生成され、パフォーマンに影響を及ぼす場合があります。

### 冗長な検索ログを有効にする方法:

- 1. App Config/Include/Sitecore.ContentSearch.config ファイルを編集します。
- 2. ContentSearch.EnableSearchDebug 設定を true に設定します。

```
<setting name="ContentSearch.EnableSearchDebug" value="true" />
```

3. App\_Config/Include/Sitecore.ContentSearch.VerboseLogging.config.examp ファイルの名前をSitecore.ContentSearch.VerboseLogging.config に変更します。

ファイル拡張子.example を削除します。

または、web.config ファイルで、ContentSearch.VerboseLogging 設定を追加またはパッチします。

```
<setting name="ContentSearch.VerboseLogging" value="true" />
```

4. web.config ファイルで、Sitecore 検索口グの log4net ログ レベルの値を DEBUG に設定します。



# 3.3 LingScratchPad

Sitecore 7.0 には LINQ 検索 API があり、これによって一般的な LINQ クエリとシンタックスを使った便利な検索機能を活用できます。 LinqScratchPad を使用することで、 Sitecore 開発者は検証と推定を行うときに、 LINQ 検索クエリを簡単に実行することができます。

### 3.3.1 必要なスキル

Sitecre 検索クエリと LINQ シンタックスについての実務知識

# 3.3.2 使用

**LinqScratchPad ツールを使用する**場合は、http://<site>/sitecore/admin ディレクトリの aspx ページ を閲覧する権限が必要です。

LingScratchPadページを閲覧します:

http://<site>/sitecore/admin/LinqScratchPad.aspx

次のようなインターフェースが表示されます:

```
using System;
using System.Collections.Generic;
               using System.Diagnostics;
using System.Ling;
using System.Web;
using Statecore.ContentSearch.SearchTypes;
               using Sitecore.Contentsearch.Search using Sitecore.Buckets.Extensions; using Sitecore.Buckets.Interfaces; using Sitecore.Buckets.Search; using Sitecore.Buckets.Search.Tags; using Sitecore.Configuration;
               using Sitecore.ContentSearch;
using Sitecore.ContentSearch.Utilities;
using Sitecore.Data;
               using Sitecore.Data.Fields;
               using Sitecore.Data Items;
using Sitecore.Globalization;
using Sitecore.SecurityModel;
using Sitecore.Sites;
18
19
20
21
               using Sitecore.Web;
               using Sitecore;
               using Constants = Sitecore.Buckets.Util.Constants;
               using SearchManager = Sitecore.ContentSearch.SearchManager;
               namespace Test {
                     class Program {
                                                private static Stopwatch stopWatch = new Stopwatch();
public static IEnumerable<SearchResultItem> Main(string str)
29
30
31
32
33
34
35
                                                         using (var context = SearchManager.GetIndex("sitecore_master_index").CreateSearchCor
                                                                 return context.GetQueryable<SearchResultItem>().Take(10).ToList();
                                                  public static string RunTimer(string str) {
return stopWatch.ElapsedMilliseconds.ToString();
36
37
41
         4
                 Reset
                                                                                                                                                            Clear Run
```



このインターフェースを使うと C# コードを記入することができます。 C# コードはコンパイルされ、実行されます。 しかし、ページに正しくコードを実行することを確実にするには、次の規約に従う必要があります。

- Test.Program タイプを使用します— ネームスペース Test、クラス Program。
- クラスには2つのメソッドも含まれます: Main と Run Timer。
  - O Test.Program.Main メソッドは、テスト中のクエリ/コードを実行するために使用されます。
    Test.Program.Main メソッドは、string タイプのひとつの引数を受け入れ、
    IEnumerable<SearchResultItem> タイプの値を返す必要があります。
  - O Test.Program.RunTimer メソッドは、Test.Program.Main メソッドからコードを実行するためにかかった時間をレポートするために使用されます。Test.Program.RunTimer メソッドは string タイプのひとつの引数を受け入れ、string タイプの値を返す必要があります。

また、stopWatch という名前の、非公開の Stopwatch フィールドもあります。このフィールドは、
Test.Program.Main メソッドを使った際のクエリとコードの実行時間を測定し、Test.Program.RunTimer
メソッドを使った実行時間をレポートするために使用することができます。このフィールドは必須ではありませんが、正しく
実行を行うために使用します。

### 重要

Test.Program.Main メソッドは、LinqScratchPad で結果のグリッドを入力するために使用する
IEnumerable<SearchResultItem> タイプの値を返します。たとえば 1000 個の結果など、大規模な結果
の一連を返すことによって、ページの結果のレンダリング速度を減速し、クエリの実行時間を想定する際に好ましくない
ことに注意してください。返される結果の数を制限し、不必要な遅延を防ぐことをお勧めします。

Sitecore 検索 API における LINQ の使用についての追加情報は、SDN の <u>Developer's Guide to Item</u> Buckets and Search を参照してください。