



10 things every Umbraco-developer should know

Markus Johansson

- Umbracian since 2007 (Umbraco 3)
- Partner at web agency Obviuse, Stockholm
- CodeGarden-Count: 9
- Core Contributor
- Package Developer



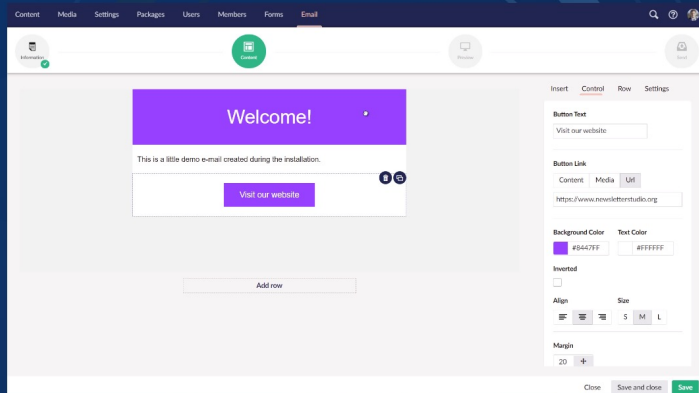
Developed for  umbraco

Reach your customers with e-mails from Umbraco

Create and send email campaigns and transactional emails from the Umbraco backoffice. It's simple, it's smart and it's integrated.

See features

Download



- Home
- Products
- People
- About Us
 - About this Starter Kit
- Blog
- Contact
- Recycle Bin

Recent activities

- Shows recent activities from all users in the back office.
- June 16, 2020 11:53 AM
Admin saved and published [About this Starter Kit](#).
 - June 10, 2020 5:48 PM
Markus Johansson saved and scheduled [Contact](#) for publishing at June 10, 2020 5:48 PM.
 - June 10, 2020 5:26 PM
Markus Johansson saved and published [Home](#).
 - June 10, 2020 5:20 PM
Markus Johansson saved and published [About Us](#).
 - June 10, 2020 3:57 PM
Markus Johansson saved and published [This will be great](#).
 - June 10, 2020 3:57 PM
Markus Johansson saved and published [Another one](#).
 - June 10, 2020 1:34 PM
Markus Johansson moved [Todo list for the Starter Kit](#) to [recycle bin](#).
 - June 8, 2020 10:04 PM
Admin saved and published [My Blog Post](#).
 - June 8, 2020 10:04 PM

0 **Unpublished content**
Shows unpublished content that has not been scheduled for publish.

- Your recent activity**
Shows your own recent activities
- June 16, 2020 11:53 AM
Saved and published [About this Starter Kit](#).
 - June 9, 2020 5:43 PM
Saved and published [Home](#).
 - June 8, 2020 10:04 PM
Saved and published [Contact](#).
 - June 8, 2020 10:04 PM
Saved and published [This will be great](#).
 - June 8, 2020 10:04 PM
Saved and published [Another one](#).
 - June 8, 2020 10:04 PM
Saved and published [My Blog Post](#).
 - June 8, 2020 10:04 PM
Saved and published [Blog](#).
 - June 8, 2020 10:04 PM
Saved and published [Todo list for the Starter Kit](#).
 - June 8, 2020 10:04 PM
Saved and published [About Us](#).

- 23 Published content nodes
- 1 Nodes in recycle bin
- 0 Members on website
- 0 New members last week



**Scratch the
surface**

**High
&
Low**

Newbie
or
Pro

@markusjoha

Let's start...



Markus Johansson | @markusjoha | obviuse.se



1. Property?

Document Types

The screenshot shows a configuration interface for a document type named "Content Page". At the top, there is a header bar with a document icon, the name "Content Page", a lock icon, and the identifier "contentPage". Below this are navigation tabs for "Design", "List view", "Permissions", and "Templates".

The main configuration area is titled "Content" and indicates it is "Inherited from: Content Base". It features two buttons: "Compositions..." and "Reorder".

Two properties are listed:

- pageTitle**: Labeled "Page Title", it is a "Textstring" and "Mandatory". The description states: "The title of the page, this is also the first text in a google search result. The ideal length is between 40 and 60 characters". The configuration shows it is "Inherited from Content Base" and has a text input field with "500 characters left" remaining.
- bodyText**: Labeled "Content", it is a "Content Base - Content - Grid layout". The description says "Enter a description...". The configuration shows it is "Inherited from Content Base" and includes an "Add a row" button.

An "Add property" button is located at the bottom of the configuration area.

Group →

Property →

Property →



Document Type Property

Choose a Beer

Please choose the beer you like the most.

Data Type

Carlsberg



Data Type

Carlsberg



Property Editor

+

Configuration



Data Type

Property Editor

Dropdown - Beer Picker

Id 1212
08ac6405-e908-4005-94a7-3ddf8716726c

Property editor Dropdown
Select a property editor

Property editor alias Umbraco.DropDown.Flexible

Configuration

Enable multiple choice

When checked, the dropdown will be a select multiple / combo box style dropdown.

Add prevalue **Add**

Add, remove or sort values for the list.

+ Carlsberg **Remove**

+ Other **Remove**



Data Type

Property Editor

Property editor	Rich Text Editor
Select a property editor	
Property editor alias	Umbraco.TinyMCE

Configuration

Toolbar	<input checked="" type="checkbox"/> <> Source code editor
	<input type="checkbox"/> Remove format
	<input type="checkbox"/> Undo
	<input type="checkbox"/> Redo
	<input type="checkbox"/> Cut
	<input type="checkbox"/> Copy
	<input type="checkbox"/> Paste
	<input checked="" type="checkbox"/> Style select
	<input checked="" type="checkbox"/> B Bold
	<input checked="" type="checkbox"/> <i>I</i> Italic
	<input type="checkbox"/> <u>U</u> Underline



A Property Editor can be used in any number of Data Types

Alter property editor behavior using
Data Type configuration



Take away

Document
Types

has

Properties

has

Data Types



Take away

Data Types

has

Property
Editor

Configuration






2. Property Value Converters

Example: Multi Node Tree Picker

Related Pages

Choose some related pages

- | | |
|--|--------|
|  About this Starter Kit
<i>/about-us/about-this-starter-kit/</i> | Remove |
|  Todo list for the Starter Kit
<i>/about-us/todo-list-for-the-starter-kit/</i> | Remove |
|  About Us
<i>/about-us/</i> | Remove |

Add



Property Value Converters

```
foreach (var page in Model.Value<IEnumerable<IPublishedContent>>("relatedPages"))  
{  
    <p>@page.Name</p>  
}
```



Property Value Converters

```
foreach (var page in Model.RelatedPages)
{
    <p>@page.Name</p>
}
```



Property Value Converters

Note:

UDI is currently not an acronym for something. There is no official definition of what it's short for. Therefore it's called *UDI*

This is what's stored in the database:

UDI

umb://document/ee82cba3a0e740639ae13026a4f72a3d,
umb://document/9c1c3a2c72b045e2a3c33c068164a018,
umb://document/3cce2545e3ac44ecbf55a52cc5965db3

```
SELECT id, versionId, propertyTypeId, textValue,intValue,decimalValue,dateValue,varcharValue FROM umbracoPropertyData ORDER BY id DESC
```

Result	Messages	Execution Plan	
id	versionId	propertyTypeId	textValue
541	67	104	umb://document/ee82cba3a0e740639ae13026a4f72a3d,umb://document/9c1c3a2c72b045e2a3c33c068164a018,umb://document/3cce2545e3ac44ecbf55a52cc5965db3
540	66	104	umb://document/ee82cba3a0e740639ae13026a4f72a3d,umb://document/9c1c3a2c72b045e2a3c33c068164a018,umb://document/3cce2545e3ac44ecbf55a52cc5965db3
539	67	54	NULL
...



Markus Johansson | @markusjoha | obviuse.se

A Property Value Converter is a class that knows how to convert the stored value into something useful for the front end



MultiNodeTreePickerValueConverter

Will take the UDIs and create a list of IPublishedContent from the cache

```
namespace Umbraco.Web.PropertyEditors.ValueConverters
{
    public class MultiNodeTreePickerValueConverter : PropertyValueConverterBase
    {
        public override bool IsConverter(IPublishedPropertyType propertyType)
        {
            return propertyType.EditorAlias.Equals(
                Constants.PropertyEditors.Aliases.MultiNodeTreePicker
            );
        }
    }
}
```



Can return different types depending on configuration

Ex: MNTP with min/max 1 item

`IEnumerable<IPublishedContent>` `@Model.RelatedContent.First().Name`

vs.

`IPublishedContent` `@Model.RelatedContent.Name`



Super Value Converters

By Callum Whyte

[https://our.umbraco.com/packages/
developer-tools/supervalueconverters/](https://our.umbraco.com/packages/developer-tools/supervalueconverters/)



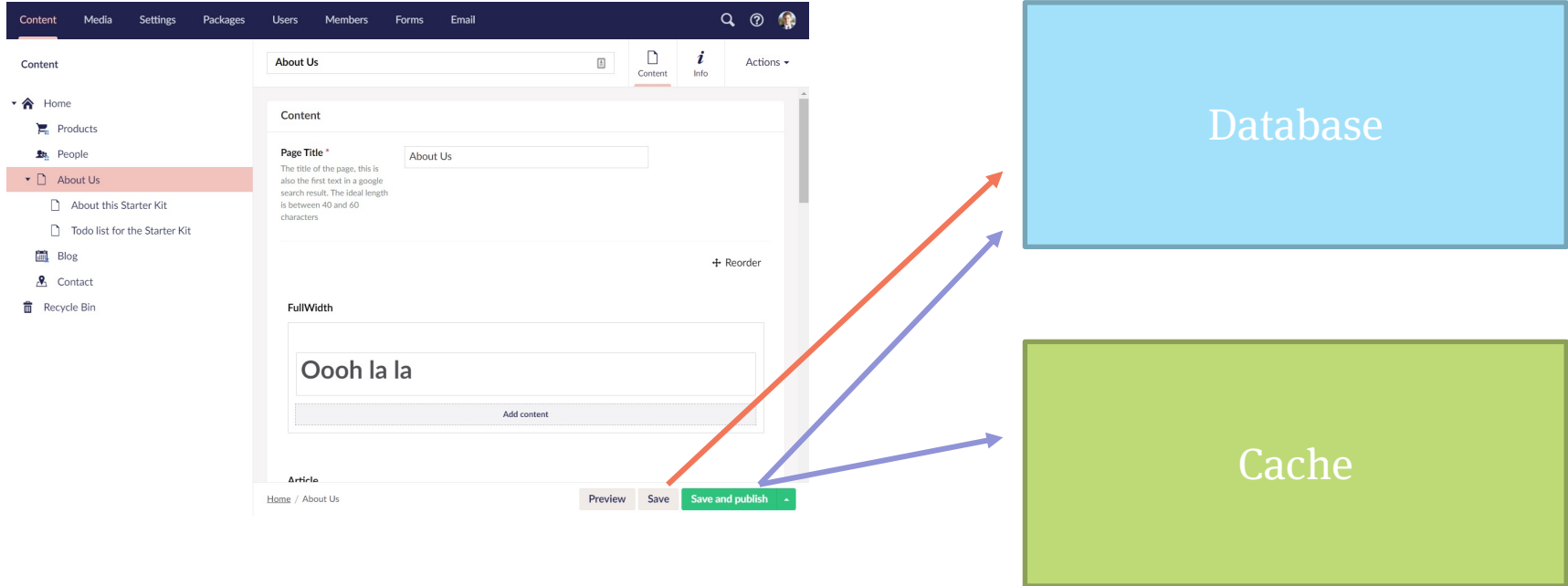
When building your own Property Editors:

Don't forget to create a Property Value Converter.

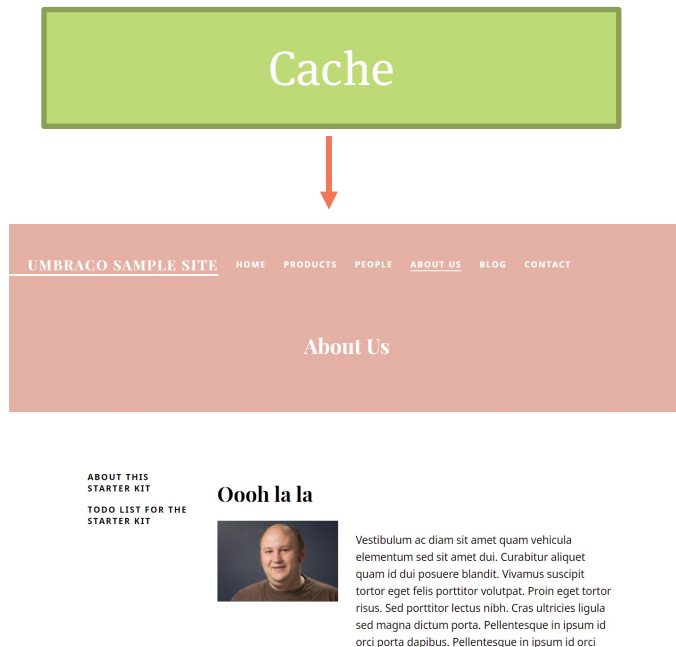


3. Database vs. Cache

Database vs. Cache



Database vs. Cache



Avoid hitting the database on the front end.



- IPublishedContent
- UmbracoHelper
- PublishedContentCache on UmbracoContext



- IContent
- ContentService
- MediaService
-



Cache vs. Database

```
public class HomeController : RenderMvcController
{
    public override ActionResult Index(ContentModel model)
    {
        var settings = this.Umbraco.Content(1050);

        return base.Index(model);
    }
}
```

UmbracoHelper



In your own services, background jobs etc: IUmbracoContextFactory and IScopeProvider



Cache vs. Database

```
public BlogService(  
    IUmbracoContextFactory contextFactory,  
    IScopeProvider scopeProvider)  
{  
    _contextFactory = contextFactory;  
    _scopeProvider = scopeProvider;  
}  
  
public List<BlogPost> GetPosts(string category)  
{  
    using (var scope = _scopeProvider.CreateScope(autoComplete:true))  
    {  
        using (var ctx = _contextFactory.EnsureUmbracoContext())  
        {  
            var blogContainer = ctx.UmbracoContext.ContentCache.GetById(1123);  
  
            var posts = blogContainer.Children.ToList();  
  
            return BlogPostMapper.Map(posts);  
        }  
    }  
}
```

Details: <https://our.umbraco.com/forum/using-umbraco-and-getting-started/102676-triggering-index-rebuild-via-hangfire-causes-objectdisposedexception-in-nucache>

- Works in background threads
 - Hangfire
 - Recurring Tasks
 - Background Tasks
 - Umbraco-Events
- UmbracoHelper does not



4. Content Versions

Content in Umbraco is versioned.

**Every time you save or publish,
a new version is created**



Content Versions

Versions

The screenshot shows a content management system interface for an 'About Us' page. At the top, there is a search bar containing 'About Us' and navigation tabs for 'Content', 'Info', and 'Actions'. Below this, the 'Links' section shows the current URL 'en-US /about-us/'. The 'History' section displays a list of seven content versions, each with a user profile picture, name, timestamp, and a 'Publish' or 'Save' button. A 'Rollback...' button is positioned to the right of the history list. An orange arrow points from the 'Rollback' text on the right towards this button. The 'General' panel on the right shows the current version's status as 'Published', its creation date as 'January 28, 2021 3:55 AM', document type as 'Content Page', and template as 'ContentPage'. The ID is '1120' with a long alphanumeric string below it.

Rollback



**Avoid storing data that changes a lot
as content in Umbraco**

Don't
store
volatile
data

**Ie. Running a import every 30 minutes
would create 48 versions per day.
or 17 520 versions in a year**



Default: Unlimited number of versions

Use UnVersion:

[https://our.umbraco.com/packages/
website-utilities/unversion/](https://our.umbraco.com/packages/website-utilities/unversion/)



5. Models Builder

- **Provides strongly typed models in views**
 - **Before Models Builder:**
`Model.GetPropertyValue("myProperty")`
 - **With ModelsBuilder**
`Model.MyProperty`
- **Wrapper around IPublishedContent**



Configure in web.config

```
<add key="Umbraco.ModelsBuilder.Enable" value="true" />  
<add key="Umbraco.ModelsBuilder.ModelsMode" value="LiveAppData" />  
<add key="Umbraco.ModelsBuilder.ModelsNamespace" value="MySite.Web.Models.Cms" />  
<add key="Umbraco.ModelsBuilder.ModelsDirectory" value="~/../MySite.Web/Models/Cms" />  
<add key="Umbraco.ModelsBuilder.AcceptUnsafeModelsDirectory" value="true" />
```



Models Builder

```
▲ 📁 CmsModels
  ▶ 📄 Blog.generated.cs
  ▶ 📄 Blogpost.generated.cs
  ▶ 📄 Contact.generated.cs
  ▶ 📄 ContentBase.generated.cs
  ▶ 📄 ContentPage.generated.cs
  ▶ 📄 Feature.generated.cs
  ▶ 📄 File.generated.cs
  ▶ 📄 Folder.generated.cs
  ▶ 📄 Hero.generated.cs
  ▶ 📄 Home.generated.cs
  ▶ 📄 Image.generated.cs
  ▶ 📄 Member.generated.cs
  ▶ 📄 NavigationBase.generated.cs
  ▶ 📄 People.generated.cs
  ▶ 📄 Person.generated.cs
  ▶ 📄 Product.generated.cs
  ▶ 📄 Products.generated.cs
```



Models Builder

```
<h1>@Model.PageTitle</h1>
```

```
///  
///  
///  
///  
///  
[ImplementPropertyType("pageTitle")]  
public string PageTitle => this.Value<string>("pageTitle");
```



Create from any IPublishedContent

```
IPublishedContent content = Umbraco.Content(1234);  
var blogPost = new BlogPost(content);  
var title = blogPost.PageTitle;
```



Models Builder makes Umbraco return actual instances of the model-classes

```
IPublishedContent content = Umbraco.Content(4211); content = {Blogpost}  
var type = content.GetType(); type = {Name = "Blogpost" FullName = "UmbracoCms.
```

```
IPublishedContent content = Umbraco.Content(4211);  
var blogPost = content as Blogpost;  
var title = blogPost.PageTitle;
```



Models Builder

Composition

The screenshot displays the 'Models Builder' interface. On the left, a 'Blogpost' model is being configured. The 'Content' section includes fields for 'Page Title' (Textstring, Mandatory), 'Categories (tags)' (Navigation Base - Keywords - Tags), 'Excerpt' (Textarea, Mandatory), and 'Body/Text' (Content Base - Content - Grid layout). An 'Add property' button is visible at the bottom of the configuration area. A red arrow points from the 'Add property' button to the 'Hero' property in the 'Compositions' panel on the right.

The 'Compositions' panel on the right shows a search bar 'Type to filter...' and a list of composition options:

- /
- Home
- Product
- /Compositions/
- Content Base
- Feature
- Hero
- Navigation Base
- Seo

At the bottom of the 'Compositions' panel, there are 'Close' and 'Submit' buttons. A red arrow points from the 'Hero' option in the list to the 'Add property' button in the 'Blogpost' configuration area.



Composition

```
public partial interface IHero : IPublishedContent
{
    /// <summary>Hero Image</summary>
    [global::System.CodeDom.Compiler.GeneratedCodeAttribute
    Image HeroImage { get; }
}
```



Check if a model uses a Composition

```
IPublishedContent content = Umbraco.Content(4211);  
string heroImageUrl = null;  
if (content is IHero hero)  
{  
    heroImageUrl = hero.HeroImage.Url;  
}
```



6. Debugging

Logs

Content Media **Settings** Packages Users Members Forms Email

Settings

- Document Types
- Media Types
- Member Types
- Data Types
- Macros
- Relation Types
- Log Viewer**
- Languages
- Content Templates

Templating

- Templates
- Partial Views
- Partial View Macro Files
- Stylesheets
- Scripts

Log Overview for Today

Saved Searches

- All Logs 🔍
- Find all logs where the Level is NOT Verbose and NOT Debug 🔍
- Find all logs that has an exception property (Warning, Error & Critical with Exceptions) 🔍
- Find all logs that have the property 'Duration' 🔍
- Find all logs that have the property 'Duration' and the duration is greater than 1000ms 🔍
- Find all logs that are from the namespace 'Umbraco.Core' 🔍
- Find all logs that use a specific log message template 🔍
- Find logs where one of the items in the SortedComponentTypes property array is equal to 🔍
- Find logs where one of the items in the SortedComponentTypes property array contains 🔍
- Find all logs that the message has localhost in it with SQL like 🔍
- Find all logs that the message that starts with 'end' in it with SQL like 🔍

Common Log Messages

Total Unique Message types: 33

Document [ContentName] (id={ContentId}) has been published.	4528
Result for {HealthCheckName}: {HealthCheckResult}, Message: '{HealthCheckMessage}'	493
{LogPrefix} Terminating {Immediate}	306

Number of Errors

155

Log Types

- Info
- Debug
- Warning
- Error
- Critical



Logs

Content Media **Settings** Packages Users Members Forms Email

Settings

- Document Types
- Media Types
- Member Types
- Data Types
- Macros
- Relation Types
- Log Viewer**
- Languages
- Content Templates

Templating

- Templates
- Partial Views
- Partial View Macro Files
- Stylesheets
- Scripts

← Log Search

Search logs... Search







Example Searches ▾

Jun 9, 2021 9:33:05 PM	Information	EMSERVER	Result for "SMTP Settings": "Error", Message: "In the Web.config file system.net/mailsettings section, the host is not configured."
Jun 9, 2021 9:33:05 PM	Warning	EMSERVER	Checks for "Macro errors" completed with errors.
Jun 9, 2021 9:33:05 PM	Information	EMSERVER	Result for "Macro errors": "Error", Message: "MacroErrors are set to 'throw' which will prevent some or all pages in your site from loading completely if there are any errors in macros. Rectifying this will set the value to 'inline'."
Jun 9, 2021 9:33:05 PM	Warning	EMSERVER	Checks for "Notification Email Settings" completed with errors.
Jun 9, 2021 9:33:05 PM	Information	EMSERVER	Result for "Notification Email Settings": "Error", Message: "Notification email is still set to the default value of your@email.here."
Jun 9, 2021 9:33:05 PM	Information	EMSERVER	Checks for "Try Skip IIS Custom Errors" all completed successfully.
Jun 9, 2021 9:33:05 PM	Information	EMSERVER	Result for "Try Skip IIS Custom Errors": "Success", Message: "Try Skip IIS Custom Errors is set to 'true' and you're using IIS version '10.0'."
Jun 9, 2021 9:33:05 PM	Information	EMSERVER	Checks for "Custom Errors" all completed successfully.
Jun 9, 2021 9:33:05 PM	Information	EMSERVER	Result for "Custom Errors": "Success", Message: "Custom errors are set to 'RemoteOnly'."
Jun 9, 2021 9:33:05 PM	Information	EMSERVER	Checks for "Trace Mode" all completed successfully.



Logs

/app_data/logs

Name	Date modified	Type
 UmbracoTraceLog.EMSERVER.20210128.json	2021-01-28 18:16	JSON File
 UmbracoTraceLog.EMSERVER.20210222.json	2021-02-22 16:12	JSON File
 UmbracoTraceLog.EMSERVER.20210411.json	2021-04-11 01:11	JSON File
 UmbracoTraceLog.EMSERVER.20210605.json	2021-06-05 22:55	JSON File
 UmbracoTraceLog.EMSERVER.20210606.json	2021-06-06 23:29	JSON File
 UmbracoTraceLog.EMSERVER.20210607.json	2021-06-07 00:36	JSON File



Compact Log Viewer

The screenshot shows the Compact Log Viewer application window. At the top, there is a search bar labeled "Search logs..." with "Clear" and "Search" buttons. Below the search bar is a dropdown menu set to "Descending". A pagination bar shows "Previous", "1" (selected), "2", "3", "4", "5", "6", "7", "8", "9", "10", "Last", and "Next". The main log area contains a table with the following data:

Timestamp	Level	Message
Jan 28, 2021 6:16:51 PM	Information	Nothing to send, checking again soon.
Jan 28, 2021 6:16:51 PM	Information	Newsletter Studio campaign scheduler checking for work.
Jan 28, 2021 6:16:46 PM	Information	Nothing to send, checking again soon.
Jan 28, 2021 6:16:46 PM	Information	Newsletter Studio campaign scheduler checking for work.
Jan 28, 2021 6:16:41 PM	Information	Nothing to send, checking again soon.
Jan 28, 2021 6:16:41 PM	Information	Newsletter Studio campaign scheduler checking for work.

On the right side, there is a red box labeled "Errors" with the number "9". Below it is a "Log Levels" section with a donut chart and a table:

Level	Count
Verbose	0

By Warren Buckley

<https://github.com/warrenbuckley/Compact-Log-Format-Viewer>



Markus Johansson | @markusjoha | obviuse.se



Use in your code!



Markus Johansson | @markusjoha | obviuse.se



Logging

```
public class MyThing : IMyThing
{
    public MyThing(ILogger logger)
    {
        _logger = logger;
    }

    public void DoSomething(string value)
    {
        _logger.Info<MyThing>("DoSomething {value}", value);
    }
}
```

Other log Levels: Debug, Warning, Error



IProfilingLogging

```
public MyThing(IProfilingLogger profLog)
{
    _profLog = profLog;
}

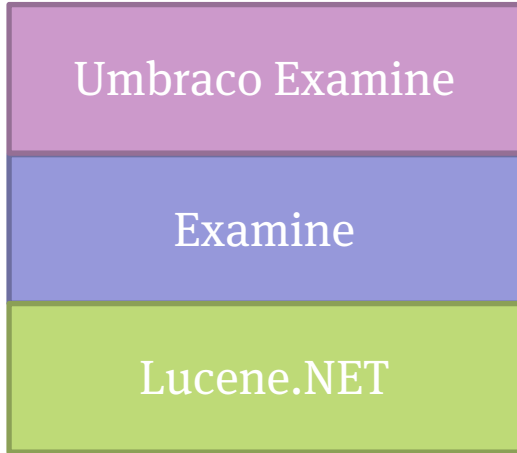
public void DoSomething(string value)
{
    using (_profLog.TraceDuration<MyThing>("Performing work", "Work done"))
    {
        Thread.Sleep(250);
    }
}
```

Timestamp ▾	Level	Machine	Message
Jun 7, 2021 12:50:41 AM	Information	EMSERVER	"Work done" (257ms) [Timing "4215d46"]
Jun 7, 2021 12:50:41 AM	Information	EMSERVER	"Performing work" [Timing "4215d46"]



7. Lucene / Examine

Umbraco Examine



- **Search Index**
- **Provides search for backoffice and front end if needed**
- **Uses Lucene.NET**
- **Examine = API on top of Lucene**
- **Good for filtering lots of data, product/article filter etc.**



Umbraco Examine

Slow

Database

- All Content
- Content Versions

Fast

Cache

- IPublishedContent
- UmbracoHelper,
UmbracoContext
.ContentCache

Use

- Listings
- Small/mid-size listings
with filters

Examine / Lucene

- Documents
- IExamineManager

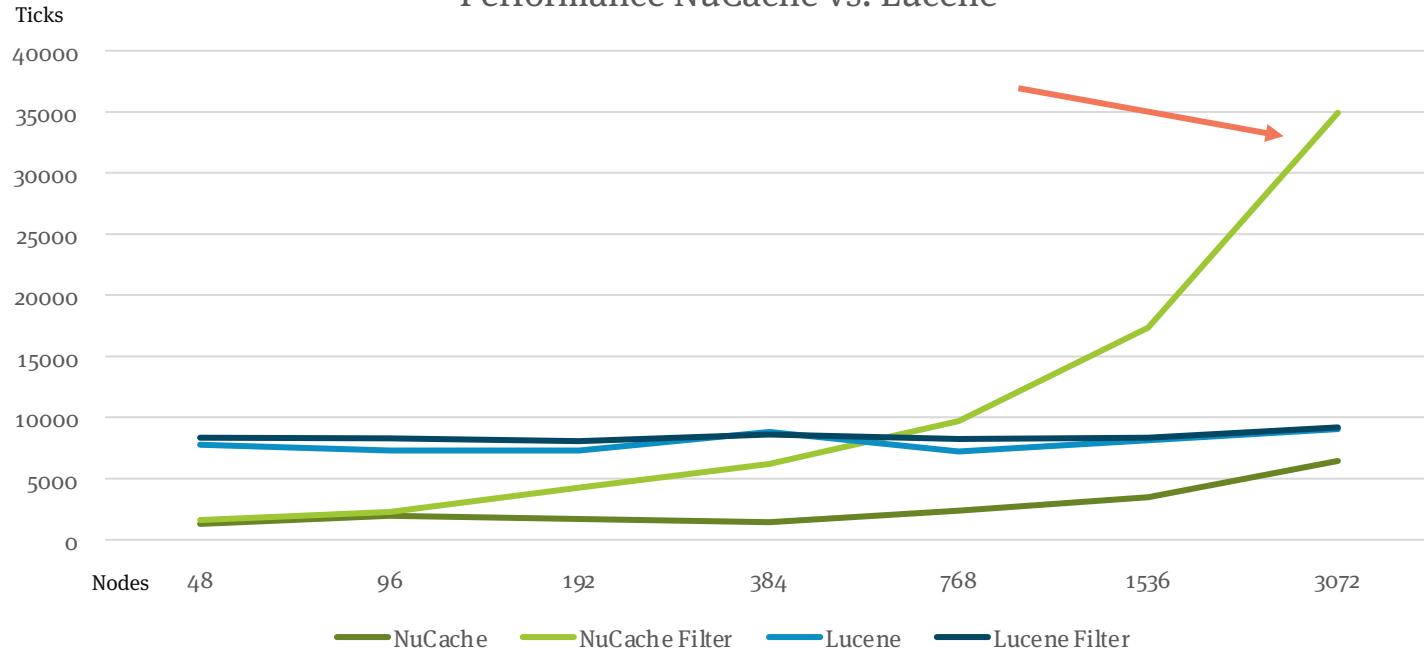
Use:

- Free text search
- Big listing with filters



Umbraco Examine

Performance NuCache vs. Lucene



- **NuCache is really fast**
- **Prefer NuCache if less than 500 nodes**
- **Use Examine to filter more than 500 nodes**
 - **Both really fast 1-3ms**
- **Examine shines with large data sets and many filters**

Another presentation with more details and code examples:
<https://umbracospark.com/media/poxgaq2u/querying-umbraco.pptx>



Lucene Examine

Doc. #: 0 43 61

Buttons: Add, Reconstruct & Edit, Delete, More like this...

Delete specified list of documents: Del List

Example: 0,12,45-90,17,123,30-32

Browse by term: FooterCtaLink

Decoded value:

Browse documents with this term (0 documents)

Document: ? of ? First Doc Next Doc

Show All Docs Delete All Docs

Term freq in this doc: ? Show Positions

Doc #: 43

Flags: I - Indexed (docs,freqs,pos) T - Tokenized; S - Stored; V - Term Vector (offsets,pos)
N - with Norms; L - Lazy; B - Binary; # - Numeric;

Field	IdfpTsvOpNLB#	Norm	Value
__icon	Idfp-S---N---	1.0	icon-calendar color-black
__indexType	IdfpTsv--N---	1.0	content
__key	IdfpTsv--N---	0.4375	9c581f42-f3fb-4e3b-86a0-a9a6cd7fa2c2
__Nodeid	Idfp-S---N---	1.0	1152
__NodeTypeAlias	IdfpTsv--N---	1.0	blogpost
__Path	Idfp-S---N---	1.0	-1,1104,1123,1152
__Published	Idfp-S---N---	1.0	y
__Raw_bodyText	-----S-----	---	{\r\n "name": "1 column layout",\r\n "sections": {\r\n {\r\n "grid": 12,\r\n "rows": {\r\n {\r\n "label": "Article",\r\n "name": "Article",\r\n n
__VariesByCulture	Idfp-S---N---	1.0	
bodyText	IdfpTsvOpN---	0.0625	Donec sollicitudin molestie malesuada. Proin eget tortor risus. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus magna justo, lacin
bodyText.Article	IdfpTsvOpN---	0.0625	Donec sollicitudin molestie malesuada. Proin eget tortor risus. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Vivamus magna justo, lacin
categories	IdfpTsvOpN---	0.625	["cg16","codegarden"]
createDate	Id--TS-----	---	637585905577970000
creatorID	Id--TS-----	---	-1

Selected field: TV Show Set norm Save

Copy text to Clipboard: Selected fields Complete document

Index name: D:\Develop...EMP\Examine\Indexes\External (R)



Markus Johansson | @markusjoha | obviuse.se

<https://code.google.com/archive/p/luke/downloads>



8. IOC / DI

Inversion of Control

Dependency Injection

- **Inject dependencies into your classes**
 - **First-class citizen in .NET Core**



IOC / DI

Do

```
public class BlogService : IBlogService
{
    private readonly IUmbracoContextFactory _contextFactory;

    public BlogService(IUmbracoContextFactory contextFactory)
    {
        _contextFactory = contextFactory;
    }
}
```

Don't

~~Current.Services.ContentService.Save(content);~~



In V8: Register dependencies with IUserComposer

```
public class SiteComposer : IUserComposer
{
    public void Compose(Composition composition)
    {
        composition.Register<IMyThing, MyThing>(Lifetime.Scope);
        composition.Register<IBlogService, BlogServiceContextFactory>();
    }
}
```



Life time's



Markus Johansson | @markusjoha | obviuse.se



Umbraco 8 - Lightinject

Lifetime	Meaning
Transient	New instance every time
Singleton	One instance during application lifetime
Scope	One instance per scope. Scope = <u>Web request</u>
Request	One instance per request to the container \approx Transient.

Umbraco 9 - .NET 5 DI

Lifetime	Meaning
AddTransient<>	New instance every time
AddSingleton<>	One instance during application lifetime
AddScoped<>	One instance per scope. Scope = Web request



9. NPoco

- **Micro ORM**
 - **Database read/update/delete**
- **A "lightweight version" of Entity Framework**
 - **Use for your custom tables**
 - **Entity Framework also works**
 - **But avoid EF for packages**



NPoco

```
public class MovieRepository : IMovieRepository
{
    private readonly IScopeProvider _scopeProvider;

    public MovieRepository(IScopeProvider scopeProvider)
    {
        _scopeProvider = scopeProvider;
    }

    public List<Movie> GetByYear(int year)
    {
        using (var scope = _scopeProvider.CreateScope(autoComplete:true))
        {
            var dtos = scope.Database.Fetch<MovieDto>(
                scope.SqlContext.Sql().SelectAll()
                    .From<MovieDto>()
                    .Where<MovieDto>(x=>x.Year == year));

            return MovieMapper.Map(dtos);
        }
    }
}
```



NPoco

```
public MovieService(
    IScopeProvider scopeProvider,
    IMovieRepository movieRepository,
    IContentService contentService)
{
    _scopeProvider = scopeProvider;
    _movieRepository = movieRepository;
    _contentService = contentService;
}

public void AddMovie(Movie movie)
{
    using (var scope = _scopeProvider.CreateScope())
    {
        if (SaveMovieAsUmbracoContent(movie)) ← Scope inside
        {
            if(_movieRepository.Save(movie))
                scope.Complete();
        }
    }
}
```



Complete your scopes!



Markus Johansson | @markusjoha | obviuse.se



10. AngularJS-hacks for the backoffice

Hack the backoffice with AngularJS:

`$httpProvider.interceptors`

Intercepts requests and responses
to/from the Backoffice APIs



Fixing TinyMCE-plugin

- **TinyMce plugin: wiris_formulaEditor**
 - **/config/tinyMce.config**
- **Problem: Umbraco will lowercase the plugin name**



AngularJS-hacks: Injectors

```
x Headers Preview Response Initiator Timing Cookies
▼ ]}]', {plugins: [{name: "paste"}, {name: "anchor"}, {name: "charmap"}, {name: "table"}, {name: "lists"},...],...}
  ▶ commands: [{name: "Source code editor", alias: "ace", mode: "Insert"},...]
  ▶ customConfig: {entity_encoding: "raw"}
    invalidElements: "font"
  ▼ plugins: [{name: "paste"}, {name: "anchor"}, {name: "charmap"}, {name: "table"}, {name: "lists"},...]
    ▶ 0: {name: "paste"}
    ▶ 1: {name: "anchor"}
    ▶ 2: {name: "charmap"}
    ▶ 3: {name: "table"}
    ▶ 4: {name: "lists"}
    ▶ 5: {name: "advlist"}
    ▶ 6: {name: "hr"}
    ▶ 7: {name: "autolink"}
    ▶ 8: {name: "directionality"}
    ▶ 9: {name: "tabfocus"}
    ▶ 10: {name: "searchreplace"}
  validElements: "+a[id|style|rel|data-id|data-udi|rev|charset|hreflang|dir|lang|tabindex|accesskey|type|name|href|target|title|c
```



Fixing TinyMCE-plugin

```
angular.module('umbraco.services').config(['$httpProvider', function ($httpProvider) {
    $httpProvider.interceptors.push(['$q', '$injector', function ($q, $injector) {
        return {
            'response': function(response) {

                var url = '/umbraco/backoffice/UmbracoApi/RichTextPreValue/GetConfiguration';

                if (response.config.url.indexOf(url) > -1) {

                    // Find item in collection and change the "e" in formulaEditor to "E"
                    var formula = response.data.commands.filter(function(x) {
                        return x.alias === 'tiny_mce_wiris_formulaeditor';
                    }).shift();

                    if (formula) {
                        formula.alias = 'tiny_mce_wiris_formulaEditor';
                    }
                }

                return response;
            }
        };
    }]);
}]);
```



Change default form in Umbraco Forms

🔒 dataConsent

* Consent for storing submitted data

Help text

Yes, I give permission to store and process my data



Markus Johansson | @markusjoha | obviuse.se



Change default form in Umbraco Forms

```
angular.module('umbraco.services').config(['$httpProvider', function ($httpProvider) {
    $httpProvider.interceptors.push(['$q','$injector', function ($q, $injector) {
        return {
            'response': function(response) {

                var scaffoldUrl = 'backoffice/UmbracoForms/Form/GetScaffoldWithWorkflows?template=';
                if (response.config.url.indexOf(scaffoldUrl) > -1) {
                    // Remove last field
                    response.data.pages[0].fieldSets[0].containers[0].fields.pop();
                }

                return response;
            }
        };
    }]);
}]);
```



80 slides in 30 minutes

Phuuu...





SuperTak!!

@markusjoha