

[About Mathias and this blog](#)[Webbshop](#)

## Hiding and showing edit mode properties on same content type in Optimizely CMS depending on context

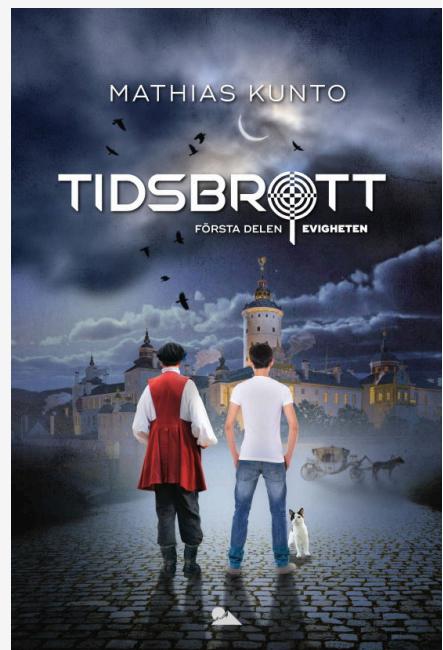
👤 Mathias Kunto ⏰ October 11, 2025 🛍 EPiServer,

Optimizely 💬 No Comments

At my current client we have the need to hide or show properties for editors in [Optimizely](#) edit mode, on the same page type, depending on where/how the page type is used. Our structure consists of global pages, and local instances of the same pages. There are times when global editors should be able to edit properties, while local editors should not. Since we do not want to clutter the interface for the local editors, we simply hide the properties for them.

```
1 [AttributeUsage(AttributeTargets.Property)]  
2 public class  
3     HidePropertyOnLocalPagesInEditModeAttribute  
4         : Attribute, IMetadataAware  
5     {  
6         public void  
7             OnMetadataCreated(ModelMetadata  
8                 metadata)  
9             {  
10                 metadata.ShowForDisplay = true;  
11                 metadata.ShowForEdit = true;
```

Evigheten del 1:  
Tidsbrott



Första delen i serien [Evigheten](#), en historisk sci-fi för ungdomar. Provläs på [Smakprov.se](#) och köp ditt exemplar hos [Adlibris](#), [Bokus](#) eller [Akademibokhandeln](#). Du kan också låna den på ditt lokala bibliotek.

Tidsbrott finns som ljudbok hos bland andra [Storytel](#), [Nextory](#) och [Bookbeat](#) samt på många bibliotek.

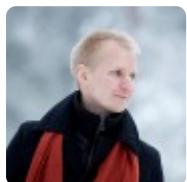
```

9     if (metadata is not
10    ExtendedMetadata extendedMetadata)
11    {
12      return;
13    }
14
15    IContent ownerContent =
16    extendedMetadata.FindOwnerContent();
17    if (ownerContent is not
18    PageDataBase pageDataBase)
19    {
20      return;
21    }
22
23    if (pageDataBase.GlobalPage != null) // Or some other condition
24    {
25      metadata.ShowForDisplay = false;
26      metadata.ShowForEdit = false;
27    }
28  }

```

The method we use is having our attribute implement the *IMetadataAware* interface, and then set the two properties *ShowForDisplay* and *ShowForEdit* as needed. In our case, we have a pointer to the GlobalPage version defined as a property on the page types, and if it is set, we know we're on a local page and can set the metadata accordingly.

## About The Author



### Mathias Kunto

Mathias is a developer from the northern parts of Sweden who currently works at Neoleaf AB in Stockholm.

"Mathias Kuntos debutroman 'Tidsbrott' är spännande hela vägen genom. Det finns inte ett kapitel, inte en sida då jag inte vill veta vad som händer härnäst." — Linnea Lindblom,

Recension hos NSD, 2020-01-08

## Recent posts

Updated Exception enrichment functionality for added safety

Allow only certain number of items in ContentAreas in Optimizely

Hiding and showing edit mode properties on same content type in Optimizely CMS depending on context

Changing Optimizely property return type yields

MissingMethodException on getter

Object reference not set to an instance of an object in EPiServer.Shell.Web.Mvc.Http method

## Recent comments

[Mathias Kunto](#) on React  
and EPiServer: Aggregated  
output with  
[EPiServer.ContentDeliveryAp](#)

[wałdis iljuczonok](#)  
(@tech\_fellow) on React  
and EPiServer: Aggregated  
output with  
[EPiServer.ContentDeliveryAp](#)

[Josef Ottosson](#) on React  
and EPiServer – Custom  
rendering of EPiServer  
block in XhtmlString field  
without partial block  
controller

[Mathias Kunto](#) on React  
and EPiServer – Custom  
rendering of EPiServer  
block in XhtmlString field  
without partial block  
controller

[Josef Ottosson](#) on React  
and EPiServer – Custom  
rendering of EPiServer  
block in XhtmlString field  
without partial block  
controller