

# Appendiks A: CMS3 Navngivnings konvention

*Dette dokument definerer den navngivnings konvention, der gælder for CMS3 og alle udvidelser rette mod denne platform.*

## Forord

Dette dokument beskriver navngivningen i CMS3. Det skal for god ordenskyld siges at denne navngivnings konvention er blevet til før udviklingen af CMS3 system. Der er dog kommet enkelte rettelser, f.eks. er afsnittet om i18n (internationalisering) forsvundet, fordi dervar for meget arbejde i at implementere dette; hvilket ikke var særligt interessant når ingen af de nuværende plugins ville være lette at porte til den udtænkte i18n implementering. Derfor blev dette udsat på ubestemt tid.

## Indledning

Da PHP ikke definerer nogen bestemt navngivnings konvention eller vejledninger af dette, definere dette projekt selv en navngivnings konvention. Formålet med denne navngivnings konvention er naturligvis at undgå koalitioner. Dette er normalt ikke et problem, men da PHP ikke understøtter navnerum<sup>1</sup>, er vi nødt til at lave nogle regler for at undgå koalitioner.

Normalt er en navngivnings konvention ikke noget alle respekterer, men jeg vil dog appellere til udvidelses udviklerne om bare en smule respekt. Sagen er nemlig den, at hvis koalitioner forekommer mellem 2 forskellige udvidelser, vil det være en fejl der vil være meget svært at rette. Desuden risikere man at fejlen kun vil forekomme på systemer hvor lige nøjagtig de 2 udvidelser er installeret. Derfor er det ikke usandsynligt at ingen af udviklerne på de 2 udvidelser vil blive opmærksom på problemet. Derfor bedes alle så vidt muligt at overholde navngivnings konventionen.

Ved inkludering af tredjedes parts biblioteker<sup>2</sup>, kunne man evt. inkludere bibliotekerne under `/lib/<PluginID>/`, som specificeret i appendiks B: Filsystems konvention. Det forventes at man konvertere alle navne i disse tredjedes parts biblioteker til at være kompatible med denne navngivnings konvention. Hvis man ikke ønsker at gøre dette, det kan f.eks. være lettere at opdatere til nye versioner fra upstream<sup>3</sup>, hvis man bibeholder upstream-udviklerens navngivnings konvention, skal man sikre sig at den ikke er i konflikt med andre udvidelser. Hvis dette er tilfældet kan man skal en specifik udvidelse der leverer dette tredjeparts bibliotek. Så andre udvidelses udviklere, kan benytte det samme tredjeparts bibliotek gennem den specifikke udvidelse du skabte. Denne metode anbefales ikke, ofte vil en konvertering til denne navngivnings konvention kunne gøres med et shellscript. Du kan finde mere information om udvidelser der leverer biblioteker i appendiks E: Pakke vejledning.

## Udvidelser

Alle udvidelser, eller på engelsk plugins, skal have deres eget PluginID. Et PluginID skal være helt unikt, der må ikke findes andre udvidelser med samme PluginID. Et PluginID skal være en streng, den skal starte med et bogstav, gerne stort, og må indeholde understreg, tal og bogstaver [a-Z].

En officiel liste over registrerede PluginIDs bliver forhåbentlig tilgængelig på et tidspunkt. Dit PluginID bør være forståeligt for mennesker, eller i hvertfald system administratorer. Et ikke unikt PluginID, vil kunne skabe store problemer i et system, du bedes derfor vælge PluginID med omhu. Mange andre steder i dokumentationen vil der blive refereret til PluginID'et som `<PluginID>`, f.eks. `/lib/<PluginID>/`. De steder hvor CMS3 systemet har behov for et PluginID benytter den "System", derfor er blandt andet denne reserveret.

---

<sup>1</sup> Navnerum: På engelsk "namespace", som de kendes fra andre programmeringsprog som f.eks.: C++, Java, C# osv.

<sup>2</sup> Biblioteker: I normal (engelsk) terminologi: "libraries", en samling logik der benyttes til at udvikle software.

<sup>3</sup> Upstream: Udvikleren af et stykke tredjeparts software.

## **Klasser**

Alle klasser bedes præfikses med: "CMS3\_<PluginID>\_", hvor <PluginID> naturligvis er dit PluginID, som specificeret i ovenstående afsnit. F.eks. vil en klasse, fra udvidelsen: MyExamplePlugin, til en LaTeX engine kunne kaldes CMS3\_MyExamplePlugin\_LaTeXEngine. Der er ingen navngivnings konvention for navngivning af metoder og felter på disse klasser. En udvidelses klasse skal altid hedder CMS3\_<PluginID> og ligge i filen /lib/<PluginID>/CMS3\_<PluginID>.php for mere information se appendiks C: Plugin udvikling.

## **Interfaces**

Alle interfaces bedes præfikses med "CMS3\_<PluginID>\_I". F.eks. vil et interface, fra udvidelsen: MyExamplePlugin, til definition af en LaTeX Engine kunne kaldes CMS3\_MyExamplePlugin\_ILaTeXEngine. System interfaces er som er meget benyttede vil kun være præfikset med I, f.eks. IOutputHandler og IProvidesContent. Se appendiks A: Filsystems konvention, for information om hvor delte interfaces skal placeres.

## **Funktioner**

Alle funktioner bedes præfikses med: "CMS3\_<PluginID>\_". Gerneralt anbefales brugen af funktioner ikke, benyt i stedet gerne statistiske metoder.

## **Konstanter**

Alle konstanter bedes præfikses med "CMS3\_<PluginID>\_". Gerneralt anbefales brugen af funktioner ikke, benyt i stedet gerne statistiske felter på dine klasse(r).

## **Database tabeller**

Alle tabeller som din udvidelse skaber skal være præfikset: "CMS3\_<PluginID>\_" eller bare "CMS3\_<PluginID>" hvis du kun har en tabel. Hvis ikke din(e) tabeller ikke er præfikset sådan, bedes du oprette en separat database, det kan du kun gøre ved at bede brugeren om database forbindelses oplysninger. CMS3 systemets databaser hedder CMS3\_System og CMS3\_Userspace (Disse kan dog have præfikses ). Databaser der ejes af et enkelt plugin bedes navngivet "CMS3\_<PluginID>", eller ved flere databaser præfikset: "CMS3\_PluginID\_".