

PHP 7.x highlights en wijzigingen

Met de introductie van PHP 7.x is de eerste vraag uiteraard waarom er geen PHP 6.x is geweest. De reden komt vanuit de ontwikkelaars van PHP. In 2005 is begonnen aan de ontwikkeling van PHP 5.6. met als grootste wijziging de ondersteuning van Unicode. Als deze versie ooit het levenslicht zou hebben gezien dan zou het nu mogelijk zijn om ook andere tekens dan het latijns te gebruiken in programmatuur.

De complexiteit was echter dermate groot dat slechts enkele ontwikkelaars van PHP dit ook daadwerkelijk volledig begrepen en ontwikkelden. Omdat het ontwikkelproces zo lang duurde is uiteindelijk besloten om functionaliteit die wel goed werkte en een mooie toevoeging was voor PHP. De implementatie van deze features zat in PHP 5.3 en 5.4. Om verder geen verwarring te veroorzaken is er voor gekozen om de nieuwste release daarom PHP 7 te noemen.

Wat is het voordeel van een upgrade naar PHP 7?

Bekende content management systemen zoals Wordpress en Joomla krijgen een behoorlijke performance boost. Het geheugen gebruik neemt met circa 50% af en de processor load verlaagd met 25%. **Bij Joomla kun je rekenen op 50% en bij Wordpress op 200-300% performance winst!** PHP 7 is de meest veilige versie van PHP en ook nog eens de aller snelste!

Waarom zou ik nu upgraden?

Vanaf 1 januari 2017 worden er geen nieuwe updates meer uitgebracht voor PHP 5.6. en zijn we afhankelijk van security updates om uw server veilig te houden.

Een volledige roadmap van PHP vind je [hier](#). Deze roadmap geeft duidelijkheid over welke versie van PHP niet langer ondersteund wordt en tot wanneer de security updates lopen. Tot en met PHP 5.5. worden op dit moment geen security updates meer gemaakt, waardoor een server mogelijk onveilig is.

Welke versie van mijn bekende CMS moet ik hebben om PHP 7 te ondersteunen?

- Wordpress (versie 4.4 of hoger).
- Joomla (officieel vanaf [versie 3.5](#) of hoger)
- Drupal ([versie 8](#) en hoger)

Wat zijn de belangrijkste wijzigingen bij een upgrade naar PHP 7?

- [Deprecated features](#) (welke een deprecated foutmelding gaan geven na een upgrade)
- [Gewijzigde functies](#)
- [Veranderingen in SAPI modules](#)
- [Verwijderde extensies en SAPIs](#)
- [Backwards incompatible changes](#)

Nieuwe mogelijkheden

Deze mogelijkheden hebben geen invloed hebben op uw applicatie, maar maken PHP wel een stuk uitgebreider voor nieuwe programmatuur.

- [New features](#)
- [New functions](#)
- [New Classes and interfaces](#)

- [New Global Constants](#)
- [Andere aanpassingen](#)