



XOOPS  
PROSEMINAR

The image shows the XOOPS logo in a stylized, outlined font. The letter 'X' is white with a yellow diagonal stroke. The letters 'OOPS' are white with a yellow outline. A large yellow arc is positioned above the 'OOPS' part of the logo. Below the main logo, the word 'PROSEMINAR' is written in a smaller, white, outlined font. The background is a dark blue gradient with a blurred screenshot of the XOOPS administration interface, showing text like 'XOOPS News', 'Blockadministration', and 'Blöcke sichtbar'.

# INHALTSVERZEICHNIS

<b>INHALTSVERZEICHNIS</b> .....	<b>2</b>
<b>EINFÜHRUNG</b> .....	<b>3</b>
Was ist XOOPS? .....	3
XOOPS und PHPNuke .....	3
Die Entstehung von XOOPS .....	3
Unterschiede zu PHPNuke .....	4
Systemvoraussetzungen .....	4
<b>GRUNDLAGEN</b> .....	<b>4</b>
Grundlegende Begriffe .....	4
Was sind Blöcke? .....	4
Was sind Module? .....	4
Was sind Templates? .....	5
Was sind Themes? .....	5
Was ist Smarty? .....	5
Wie Arbeitet Smarty in XOOPS? .....	5
Content, Design, Funktionalität .....	7
<b>MODULE</b> .....	<b>7</b>
<b>FAZIT</b> .....	<b>7</b>
Kriterien „Grundlagen Content-Management“ .....	8
Workflow .....	8
Freigabeverfahren .....	8
Versionskontrolle .....	8
Betriebssystem .....	8
Suchmaschine .....	8
Persönliche Bewertung von XOOPS .....	8
<b>QUELLEN</b> .....	<b>9</b>

# EINFÜHRUNG

## Was ist XOOPS?

**extensible,  
objectoriented,  
Portal Script**

XOOPS ist ein durch Module in seiner Funktionalität erweiterbares (eXtensible), auf einem objektorientierten Programmierkonzept basierendes (ObjectOriented), in PHP implementiertes System zur Verwaltung von Web-Portalen und Websites (Portal System).

**CMS**

Als, nach der innerhalb dieser Vortragsreihe erarbeiteten und festgehaltenen Definition im Vortrag „Grundlagen Content-Management“, weitestgehend vollständiges Content-Management-System unterstützt XOOPS den Content-Life-Cycle mit dem im Installationspaket enthaltenen Redaktionsmodul „Artikel / News“, sowie die Trennung von Design, Content und Funktionalität.

**datenbankbasiert**

XOOPS basiert auf relationalen Datenbank-Management-Systemen, momentan allerdings nur MySQL, die den Content des Systems verwalten.

**voll modularisiert**

Das XOOPSystem ist voll modularisiert, was bedeutet, dass alle Funktionalitäten, abgesehen von den im Kern realisierten in Modulen an das System angebunden werden. Dies ermöglicht eine dynamische und nahezu uneingeschränkte Erweiterung des Systems und eine große Funktionalitätensvielfalt.

**multisprachenfähig  
weltweit Unterstützt**

Das System sowie der Support, der durch Supportcommunities gewährt wird, ist in multiplen Sprachen, unter anderem auch Japanisch, Chinesisch und Koreanisch, erhältlich und auf einen Weltweiten und benutzerfreundlichen Einsatz ausgelegt.

**themebasiert**

Durch eine auf Themes basierende Oberfläche ist es den Benutzern möglich, die Seite ihren Layoutansprüchen anzupassen, indem sie aus einer Liste von vorgegebenen Themes ein für sie passendes auswählen können.

**GPL-Lizenzierung**

XOOPS ist unter den Bedingungen der GPL (GNU General Public Licence) erhältlich. Das System ist demnach frei und muss inklusive Quellcode angeboten werden.

## XOOPS und PHPNuke

### Die Entstehung von XOOPS

**PHP**

Die Ursprünge von XOOPS gehen auf eine frühe Version von PHPNuke zurück, das sich zu dieser Zeit noch stark in der Entwicklung befand, somit noch fehlerbehaftet war und nur wenige integrierte Funktionalitäten aufwies. Durch diesen Missstand wurde eine Gruppe von Personen dazu veranlasst die ihnen fehlenden integrierten Funktionen und die vorliegenden Fehler durch PHP-Scripte zu beheben und diese als unter dem Namen Nukeaddons zu publizieren.

**Nukeaddons**

Zur Entwicklung der Nukeaddons konnte allerdings nur auf schon erschienene PHPNuke-Versionen zurückgegriffen werden, so dass bei Veröffentlichung des Add-ons oft bereits eine neuere Variante von PHPNuke erhältlich war.

**MyPHPNuke**

Diesen Umstand machte sich die MyPHPNuke-Community zunutze, kombinierte die PHPNuke 4.4.1 Basis mit dem zugehörigen Nukeaddon und veröffentlichte eine PHPNuke Distribution unter dem Namen MyPHPNuke. Nach einer konstanten Weiterentwicklung dieser Basis wurde eine weitere Idee verfolgt, die zunächst unter dem Namen

**MyPHPukeSE  
XOOPS**

MyPHPNukeSE (Second Edition) firmierte: Ein Content-Management-System mit einem komplett objektorientierten Programmieransatz. Dies wurde später in XOOPS umbenannt.

## Unterschiede zu PHPNuke

<b>erweiterte Gruppenverwaltung</b>	Die Unterschiede zwischen den Systemen sind heute weiter angewachsen. XOOPS bietet gegenüber PHPNuke eine wesentlich erweiterte Gruppen- und Benutzerverwaltung. So ist es möglich, neben den drei bereits bestehenden Benutzergruppen „anonyme User“, „Registrierter Benutzer“, „Webmaster“ weitere beliebige Benutzergruppen anzulegen und ihnen individuelle Rechte zuzuteilen. Die Vergabemöglichkeit der Rechte wird durch den Kern und die installierten Module definiert. Sie setzt sich aus allgemeinen Administrationsrechten für einzelne Kernkomponenten und Module, sowie Gruppenrechten für Module und sogar einzelne Blöcke zusammen.
<b>Rechtevergabe</b>	
<b>User-Management, -Kommunikation</b>	Das User-Management wurde durch erweiterte Abfrageoptionen verbessert und eine interne Benutzer-Kommunikation integriert, die es ermöglicht systeminterne Nachrichten an Mitglieder zu schreiben.
<b>Smarty</b>	Die Integrierung der Smarty-Template-Engine seit XOOPS 2.0 unterscheidet die Systeme nun auch wesentlich im Template-System.

## Systemvoraussetzungen

<b>Websserver</b>	Zum Betreiben einer von XOOPS verwalteten Webseite sind Systeme mit einem Apache-, Roxen-Webserver oder IIS (Microsoft Internet Information Server) nötig. Zudem sollte als Datenbank-Management-System vorerst nur MySQL eingesetzt werden, da erst die nächsten XOOPS-Versionen offiziell andere Datenbank-Management-Systeme unterstützt.
<b>DBMS</b>	
<b>Betriebssystem</b>	Das Betriebssystem kann nahezu beliebig gewählt werden, getestet wurden allerdings bisher nur Linux, Mac OS X und nahezu alle Windows-Versionen (98, 98SE, ME, NT, 2000, XP, 2003, 2003 Enterprise).

# GRUNDLAGEN

## Grundlegende Begriffe

Zum besseren Verständnis der weiteren Ausarbeitung sind zunächst diverse Grundbegriffe zu erläutern.

### Was sind Blöcke?

<b>Seitenteil</b>	Ein Block stellt sich dem Betrachter als Teil der XOOPS-Seite dar, der eine bestimmte Funktionalität oder Information zeigt.
<b>Information, Funktionalität</b>	Tatsächlich sind Blöcke vielmehr eigenständige Seiten, die vom Benutzer erstellte, aus Modulen oder dem Kern stammende, Funktionalitäten und Informationen anbietet. Die Positionierung dieser Seiten auf der XOOPS-Seite wird durch Blockbereiche bestimmt, die in der Template-Datei, theme.html, festgelegt werden und auf die die einzelnen Blöcke verteilt werden können. Es können maximal fünf dieser Bereiche genutzt werden.
<b>Administrator</b>	Die Verteilung wird dabei im Administrator von XOOPS vorgenommen in dem neben der Platzierung auch die Rangfolge innerhalb der Blockbereiche und die Sichtbarkeit definiert werden können. Hier ist es auch möglich bestehende Blöcke zu ändern und neu anzulegen.

### Was sind Module?

<b>integrieren Funktionalität</b>	Module erweitern, wie schon in der Einleitung erwähnt, das System um zusätzliche, im Kern, nicht oder nicht ausreichend enthaltene Funktionalitäten, die in Form von PHP-Dateien vorliegen.
-----------------------------------	---

Um die Funktionalitäten voll in das System integrieren zu können, sind aber zusätzlich Templates (HTML-Dateien) für die durch das Modul bereitgestellten Blöcke (entspr. Block-Templates) und die Moduleseiten (entspr. Modul-Templates) sowie eventuell Mediendateien, wie Bilder, notwendig.

**Datei-Ansammlung** Diese Ansammlung von Dateien, die real in einem Verzeichnis zusammengefasst sind, nennt man allgemein Modul.

## Was sind Templates?

**Schablone** Ein Template erfüllt wie der Name erahnen lässt die Funktion einer Schablone.  
**Grundstruktur** Mit Hilfe von Templates, die als HTML-Dateien realisiert sind, werden Blöcke, Module und Seiten (Theme-Templates) in ihrer Grundstruktur definiert. Dies geschieht durch Tabellen, deren Felder die Positionierung festlegen, und Variablen der Smarty-Template-Engine, die später durch Content oder Funktionalitäten ersetzt werden.  
**Hierarchie** Zwischen den einzelnen Templates bestehen eine hierarchische Beziehung. Das oberste Template des Standard-Themes, theme.html, integriert bspw. die hierarchisch direkt darunter liegenden Modul-Templates, sowie theme\_block.html-Dateien. Letztere integrieren wiederum Block-Templates.

## Was sind Themes?

**Gesamtbild** Themes bestehen im wesentlichen aus Theme-Templates, also den Template-Dateien, die die Grundstruktur einer Seite festlegen, CSS-Dateien (Cascading-Style-Sheet-Dateien), die die Formatierung von Schrift und Hintergrund für das ganze System, also auch für Module und Blöcke, definieren, und häufig auch Mediendateien, wie Bilder und Flashanimationen. Themes bestimmen somit das Gesamtbild einer XOOPS-Webseite.

## Was ist Smarty?

**Template-Engine PHP-Klasse** S Smarty ist eine Template-Engine, die in PHP als Klasse implementiert wurde. Dies hat den Vorteil, dass sie direkt in ein objektorientiertes in PHP realisiertes System, also auch XOOPS, eingebunden werden kann. Template-Engines werden generell dazu genutzt Programmier-Logik und Design zu trennen, so dass Änderungen am Einen das jeweilig Andere unbeeinflusst lassen.  
**Template-Script-Sprache** Um das Design von Templates durch Designer zudem noch zu vereinfachen, wurde eine Scriptsprache für Smarty entwickelt, die einfach zu erlernen ist und von Smarty später direkt in komplexeren PHP-Code übersetzt wird, der die gewünschten Aktionen implementiert. Die erstellten Template-Dateien mit Smarty-Tags werden zusammen mit CSS-Dateien an die Smarty-Engine übergeben, die daraus HTML-Dateien mit ausführbaren PHP-Skripten kompiliert. Diese Dateien, im template\_c-Verzeichnis abgespeichert, können fortan verwendet werden, ohne dass sie bei jedem Aufruf neu kompiliert werden müssten.  
**kompiliert Templates**

## Wie Arbeitet Smarty in XOOPS?

**Seitenanfrage** Smarty wird nach einer Seiten-Anfrage durch die vom PHP-Parser aufgerufene header.php initialisiert. Genauer, es wird eine Instanz der Klasse template.php erstellt, die die original  
**Initialisierung** Smarty-Klasse smarty.class.php erweitert.

```

template.php
//Constructor
function XoopsTpl(){...}
header.php
$smarty = new XoopsTpl();

```

## Themeübergabe

Dem Smarty-Objekt werden nun u.a. das Theme, also die einzelnen Template-Dateien, die style.css-Datei und der Pfad zu den für das Theme relevanten Mediendateien übergeben.

```
header.php
    $xoopsTpl->assign(array('xoops_theme'..., 'xoops_imageurl'...,
        'xoops_themecss'...));
```

## Blocklisten erstellen

Ein Objekt der Klasse xoopsblock.php liefert zudem eine Liste aller Blöcke inklusive deren Positionierung im Theme-Template. Dazu wird erst anhand der Gruppenzugehörigkeit geprüft, welche Blöcke für den anfordernden Benutzer sichtbar sind.

```
header.php
    $xoopsblock = new XoopsBlock();
    ...
    $block_arr =& $xoopsblock->getAllByGroupModule(
        $xoopsUser->getGroups(), ..., XOOPS_BLOCK_VISIBLE);
    ...
```

## Blockübergabe

Danach werden die Blöcke, je nachdem, welchen Wert die Positionsvariablen der Blöcke haben, an Smarty in einem Array, hier z.B. xoops\_lBlöcke, mit den Verweisen auf den Titel des Blocks und den Content, übergeben. Das Array ist im Theme-Template, also der theme.html, wieder zu finden.

```
header.php
    foreach (array_keys($block_arr) as $i) {
    ...
        switch ($block_arr[$i]->getVar('side')) {
            case XOOPS_SIDEBLOCK_LEFT:
                $xoopsTpl->append('xoops_lBlöcke', array('title'...,
                    'content'...));
            ...
        }
```

## Theme-Template Theme-Block- Templates

Smarty übersetzt nun die Theme-Template-Datei, theme.html, in der für jeden Block des übergebenen Arrays eine entsprechende Theme-Block-Template-Datei, hier theme\_blockleft.html, eingefügt wird.

```
header.php
    $xoopsTpl->display($xoopsConfig['theme_set'].' /theme.html');
```

```
theme.html
    <{foreachitem=block from=$xoops_lBlöcke}>
        <{include file="default/theme_blockleft.html"}>
    </foreach>
```

## Block-Templates

Die jeweilige Theme-Block-Template-Datei bindet nun die spezifischen Blöcke und deren Block-Templates über Variablen ein. Die spezifischen Blöcke werden kompiliert und im template\_c-Verzeichnis gespeichert.

```
theme_blockleft.html
    <div class="blockTitle"><{$block.title}></div>
    <div class="blockContent"><{$block.content}></div>
```

## erneute Anfrage

Abschließend werden die kompilierten Theme-Block-Template-Dateien sowie die theme.html-Datei in template\_c gespeichert.

Sollte eine erneute Anfrage eine oder alle bereits kompilierten Komponenten erfordern und hat/haben sich diese Komponente/n nicht geändert, kann, ohne nochmaliges Kompilieren, die angeforderte/n Komponente/n direkt aus dem template\_c-Verzeichnis weitergeleitet werden.

## Content, Design, Funktionalität

<b>Contentdefinition</b>	Als Content wird in XOOPS generell alles bezeichnet, was in einer SQL-Datenbank und einem Dateisystem gespeichert werden kann, da für nahezu jegliche Art von Inhalt ein entsprechendes Modul geschrieben und an das System angegliedert werden kann.
<b>Content</b>	Content wird also entweder aus einer SQL-Datenbank oder aus dem Dateisystem der lokalen oder einer über ein Netzwerk erreichbaren Plattform über PHP-Befehle innerhalb der Seiten, Blöcke und Module eingepflegt.
<b>Design</b>	Dies geschieht dann, wenn das Design durch das Zusammenwirken von Template- und CSS-Dateien in Smarty eine HTML-Datei erzeugt und der darin integrierte PHP-Code vom Interpreter übersetzt und ausgeführt wird.
<b>Funktionalität</b>	Die Funktionalitäten, die in PHP-Dateien, im Kern, oder in Modulen liegen, werden dann ebenfalls durch include-Befehle oder Variablen eingebunden, interpretiert und ausgeführt.

## MODULE

In diesem Kapitel werden kurz einige der Module vorgestellt, die im Installationspaket enthalten sind, um einen Einblick in den anfänglichen Umfang des Systems zu gewähren.

### **System Admin / Systemeinstellungen**

Über die Systemeinstellungen werden die im Kern integrierten Funktionalitäten, wie die Blöckeverwaltung, Bannermanagement usw. und die XOOPS-Seite verwaltet.

### **News / Artikel**

Ein Redaktionsmodul für das XOOPS-System. Es ermöglicht den Benutzern, Artikel zu vordefinierten Themen zu verfassen und diese zur Veröffentlichung an den/die Administratoren zu senden. Letztere können über die Publikation des Artikels entscheiden.

### **Downloads und Links**

Über das Links-Modul kann der Administrator den Nutzern eine Links-Sektion mit nach Kategorien geordneten Links anbieten. Die User können die angebotenen Links bewerten und neue Links vorschlagen. Das Downloads-Modul hat analoge Funktionalitäten.

### **Polls / Umfragen**

Ein Umfragemodul, mit dem sich Umfragen zu beliebigen Themen erzeugen lassen.

### **Außerdem sind im Installationspaket noch enthalten:**

Contact Us / Webmasterkontakt  
 Forum  
 FAQ  
 Headlines / Schlagzeilen  
 Partners / Partner  
 Sections / Bereiche

## FAZIT

Die Bewertung zu dem System wird zunächst anhand der Kriterien aus dem Vortrag „Grundlagen Content Management“ vollzogen und schließt mit einer persönlichen Bewertung, die auf meinen Erfahrungen mit dem System basiert.



## Kriterien „Grundlagen Content- Management“

### Workflow

#### Webformular

#### keine speziellen Kenntnisse

Der Workflow im Bezug auf das Redaktionsmodul vollzieht sich bei XOOPS ähnlich wie bei PHPNuke. Der Autor loggt sich in das System ein und kann dann im Modul Artikel über ein Webformular, den gewünschten Artikel in einer definierten Kategorie erstellen und den gewünschten Text eingeben. Durch Buttons ist es dem Autor möglich, ohne Kenntnisse von HTML, Links und Bilder einzufügen sowie den Text zu formatieren. Der Administrator kann zudem ein Datum zur Veröffentlichung, sowie zum Ablauf der Publizierung des Dokuments anlegen.

### Freigabeverfahren

#### ohne Kontrolle mit Kontrolle

#### kein direktes Ablehnen

Die Freigabe eines Artikels erfolgt, je nach vorher definierter Einstellung direkt, ohne zusätzliche Kontrolle durch den Administrator oder erst nach Kontrolle durch die Person mit Administrationsrechten. Das direkte Freigabeverfahren ist dann zu empfehlen, wenn nur wenige Autoren an den Artikeln arbeiten und im direkten Kontakt zum Administrator stehen. Dem Freigabeverfahren in dem im Installationspaket enthaltenen Modul fehlt es allerdings an einer arbeitsflussfördernden Möglichkeit der Ablehnung eines Artikels. So wird der Artikel bei Ablehnung nicht direkt wieder an den Autor zurückgesandt. Verbesserungsvorschläge und Änderungswünsche des Administrators am Text müssen also separat in den Text eingefügt werden und per E-Mail oder über eine persönliche Nachricht an den Autor zurückgesandt werden.

### Versionskontrolle

#### keine Versionsk.

Eine Versionskontrolle ist im System nicht vorgesehen.

### Betriebssystem

#### unabhängig

Wie in der Einleitung erwähnt ist XOOPS prinzipiell betriebssystemunabhängig.

### Suchmaschine

#### nur META-Tags

Direkte Suchmaschineneintragen sind aus dem XOOPS-Administrator heraus nicht möglich. Es wird lediglich die Möglichkeit angeboten META-Tags anzulegen.

### Persönliche Bewertung von XOOPS

#### keine Dokumentation

#### benutzerfreundlich dokumentierter Quellcode

#### gute Supportcommunity

#### kleine, große Websites

Ein Einstieg in das XOOPSystem wird zu Anfang dadurch erschwert, dass eine vernünftige Dokumentation praktisch nicht zu finden ist. Trotz dieses Misstandes findet man sich allerdings aufgrund einer anfangs sehr übersichtlichen Benutzeroberfläche und dokumentiertem Quellcode schnell im System zurecht. Kommen allerdings zu den im Installationspaket enthaltenen Modulen weitere hinzu, kann die Navigation innerhalb des Administrators auch schnell unübersichtlich werden.

Hat man trotz allem noch Fragen kann, man sich an die Supportcommunities wenden, mit denen ich recht positive Erfahrungen gemacht habe. Die Antwortzeiten auf Fragen sind, zumindest in der weit größeren englischen Community, recht schnell und meist auch kompetent.

XOOPS sticht zudem, gerade gegenüber PHPNuke, durch eine gute Benutzergruppen- und Rechteverwaltung hervor, die es ermöglicht, neben kleinen privaten Webseiten mit wenigen Benutzern, auch größere Unternehmensseiten zu verwalten.

Der Einsatz von XOOPS als Content-Management-System empfiehlt sich allerdings momentan nur, in ausgewählten Bereichen, da noch eine zu große Fokussierung auf das Verwalten von Webcommunities besteht.



## QUELLEN

<http://www.myxoops.de.org> .info

<http://www.xoops.org>

<http://smarty.php.net>