Zend Framework Einführung

Ich möchte hier eine theoretische Einführung und einen kleinen Vorgeschmack in das Zend Framework geben um das "Was", "Wozu" und "Warum" zu klären.
Bei dem Wort "Zend Framework" sollte jedem, der mit PHP arbeitet, das Wort "Zend" schon über den Weg gelaufen sein.

== Was ist ein Framework? ==
Was genau ist ein Framework? Die Definition der Wikipedia lautet:

> Ein Framework (engl. für „Entwicklungsrahmen“) ist ein Programmiergerüst, das in der Softwaretechnik, insbesondere im Rahmen der objektorientierten Softwareentwicklung sowie bei komponentenbasierten Entwicklungsansätzen, verwendet wird.

> Ein Framework ist selbst noch kein fertiges Programm, sondern stellt den Rahmen, innerhalb dessen der Programmierer eine Anwendung erstellt, zur Verfügung ...

Anhand dieser beiden Sätze, kann man schon erahnen was auf einen zukommt.

Kurz gesagt ist das Zend Framework ein Gerüst, dass die Arbeit in Bezug auf Strukturierung und Entwicklung erleichtern soll.

== Zend Framework ==
Wenn man die Dokumentation überfliegt, dann verfliegt auch bei den meisten die Euphorie.
Die "Warum"-Frage sollte sich bei jedem von selbst klären, wenn man sich das ganze "Ausmaß" anschaut:
framework.zend.com/manual/de/

=== Wozu das Zend Framework? ===
Die Frage "Wozu brauch ich das alles, bis jetzt hat auch immer geklappt" ist berechtigt.

Man darf das Zend Framework nicht als Hilfestellung für seine jetzigen Projekte sehen (Natürlich hat man auch die Möglichkeit einzelne Klassen und Funktionen zu nutzen), sondern muss es als ganzes, fast wie eine neue Programmiersprache, sehen.

Alle auf der rechten Seite in der Dokumentation vorgestellten Klassen sind miteinander verbunden und verzweigt und lassen kaum Wünsche offen, was nicht heißt, dass man alles nutzen muss.
Über eine Datenbankklasse, eine Mailklasse, Templateklasse etc. ist alles in einem Paket enthalten.
Diese Klassen habe ich bewusst aufgezählt, da dies die gängigsten Klassen sind, welche man bei seinen Projekten benutzt.

Für den kurzen Überblick gibt es noch eine zusammengefasste Komponentenliste:
framework.zend.com/about/components

Es scheint so, als dass man ein gewisses Know-How braucht um hier überhaupt voran zu kommen.
Das stimmt nur teilweise, denn ohne PHP Kenntnisse kommt man logischerweise nicht weit, das ist richtig, aber auch wenn an diesem Punkt sich nur immer mehr Fragen aufwerfen ist das Zend Framework die beste Möglichkeit "richtig" programmieren zu lernen bzw. bietet das Framework eine Möglichkeit die volle objektorientierte Leistungsfähigkeit der Zend Engine zu nutzen und gibt auch erfahrenen Programmieren einen sehr guten Anreiz PHP als vollwertige Programmiersprache anzusehen.

Womit auch gleich die Frage geklärt ist, "Wozu man das alles braucht".

Insgesamt ist das Framework ein zukunftssicheres und sehr gelungenes Konzept mit einer perfekten Umsetzung und sehr

guten Dokumentation.

== Persönliche Erfahrungen ==

Wenn man etwas neues erlernen möchte, ist es zwingend notwendig den ersten Schritt zu tun und sich darauf einlassen. Ohne Ehrgeiz kommt man nirgendwo weiter.
Für die Arbeit mit dem Zend Framework ist es sehr wichtig, von seinen jetzigen "geliebten" Klassen (sei es AdoDB, Smarty, phpMailer o.ä) loszulassen und sich auf was neues einzulassen.
Man hat später die Möglichkeit z.b Smarty als Templateklasse in das Zend Framework zu integrieren, aber diesen Schritt sollte man bei Bedarf erst später wählen und Pferd nicht hinten aufziehen.