|  |
| --- |
| **Dokumentation «borovcnik.ch»** |
| Diese Dokumentation beschreibt den Aufbau der PHP/JS-Umgebung für borovcnik.ch |

Inhaltsverzeichnis

**Es wurden keine Einträge für das Inhaltsverzeichnis gefunden.**

# Javascript

## app.js

Definiert das Ausgangsobjekt.

init()

Setzt die Anwendung an den Ausgangspunkt.

Core::init

## lib

### lib/query.js

Erzeugt asynchrone Ajax-Anfragen. Die Anfragen werden an **server.php** gerichtet.

Direkter Aufruf

# PHP

## Konstanten

| **Konstante** | **Beschreibung** | **Beispiel** |
| --- | --- | --- |
| DB\_HOST | config.php – definiert den Datenbank-Host | 127.0.0.1 |
| DB\_NAME | config.php – definiert den Datenbank-Namen | boro |
| DB\_USER | config.php – definiert den Datenbank-Benutzer | root |
| DB\_PASSWORD | config.php – definiert das Datenbank-Passwort im Klartext |  |
| SYS\_DEBUG | config.php – aktiviert den Debugger | false |
| SYS\_MAX\_UPLOAD | config.php – definiert die maximale Dateigrösse für Uploads | 50M |
| SYS\_MAX\_MEMORY | config.php – definiert die Memory-Limite für PHP | 1024M |
| APP\_PATH | io.php – definiert das Webroot-Verzeichnis | /var/www/boro/ |

## Tools

### Log

Eine Protokollklasse zur Ausgabe von Meldungen in die Protokolldatei **debug.log**.

Log::out($msg='', $trace=false)

Erzeugt immer eine Kurzmeldung in die Protokolldatei.  
$trace=true erzeugt den Backtrace

### Util

Util::dump($msg=null, $indent=0, newline="\n")

Erzeugt eine Zeichenkette, die $msg zur Analyse aufbereitet.  
Arrays werden arrangiert und Objekte erklärt

## Controller

### server.php

Es gibt einen zentralen Einstiegspunkt für die Verarbeitung aller Server-Anfragen.

Diese müssen immer über server.php laufen.

Hier wird eine Kommandosequenz verarbeitet und allenfalls PHP-Objekte ins Leben gerufen.

Konsolen-Aufruf

Der Aufruf ab Konsole geschieht über den PHP-Script-Interpreter,wobei dieser maximal 2 Parameter erhält:  
Der 1. ist das Script selbst (io.php) und der 2 umfasst in einer Zeichenkette alle Script-Parameter.

php server.php "class::method::p1=v1&p2=v2&p3=v3"

Browser-Aufruf

Der Aufruf ab Browser geschieht über eine http-Anfrage an io.php.  
Der Browser-Aufruf unterstützt ebefalls die Short-Notation

http://server.php?class::method&p1=v1&p2=v2&p3=v3

### Core

IO\_Core::init()