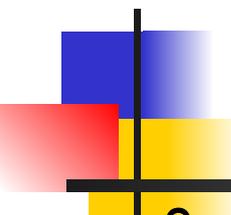


PHP #1.1.: ВВЕДЕНИЕ



- Задача динамического построения Веб страниц
- Понятие о CGI
- Скрипты серверной стороны
- Основные особенности и история развития PHP
- Схема работы PHP и "Привет, Мир!"
- Литература



Примеры задач, требующих реализации алгоритмов серверной стороны

- Организация доступа к информационным системам и базам данных в Интернет
 - Получение от пользователя параметров запроса
 - Модификация ресурса
 - Представление информации через страницы, генерируемые «на лету»
- Таргетинг
 - Установка ключиков
 - Сбор и хранение информации о пользователе
 - Подбор и размещение рекламных объявлений в соответствии с интересами пользователя



Common Gateway Interface (CGI): общий интерфейс шлюза

- Отсутствие в первых Веб серверах функциональности для динамического создания Web-страниц
- Идея: запускать на стороне сервера внешние программы, передавать им данные и получать от них сформированные страницы
- Задача: разработать спецификацию обмена данными между прикладной программой выполняемой по запросу пользователя, и HTTP-сервером, который данную программу запускает
- Создание CGI - 1993, NCSA, Роб МакКул в рамках разработки NCSA HTTPd на основе CERN HTTPd;
- Стандарт CGI 1.1(RFC3875) – IETF, 2004;



Common Gateway Interface (CGI): ОСНОВНЫЕ ДАННЫЕ

- CGI - стандарт интерфейса, используемого для связи с веб-сервера и внешней программы. Такую программу, работающую совместно с веб-сервером, принято называть «CGI-программа» или «CGI-скрипт» (сценарий).
- CGI позволяет использовать любой язык программирования, работающий со стандартными устройствами ввода и вывода.
- CGI является исторически первым и одним из наиболее распространённых средств создания динамических веб-страниц



Принцип работы CGI

- При HTTP запросе, Веб сервер определяет, что требуемый ресурс не статичный документ, а CGI программа
- Веб сервер формирует пул параметров в виде переменных окружения, запускает программу скрипта и передает ей в потоке ввода данные пользователя;
- При работе программы поток вывода направляется серверу для формирования тела HTTP ответа;
- При успешном завершении программы Веб сервер передает подготовленный HTTP ответ пользователю



Скрипты (сценарии) серверной стороны

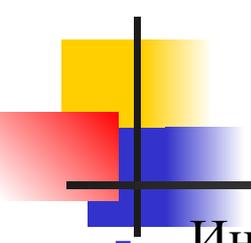
- Неудобство создания всего документа в CGI программе
- Идея «доработки» страниц
- Скрипты и их обработка
- Языки написания скриптов серверной стороны



Что такое PHP?

- PHP – язык программирования общего назначения с открытым исходным кодом
- Основная цель PHP – предоставление веб-разработчикам средств быстрого создания динамически генерируемых веб-страниц
- PHP создавался специально для ведения веб-разработок и код на нем может внедряться непосредственно в HTML код
- PHP – от англ. Personal Home Page tools, далее Hypertext Preprocessor
- PHP – интерпретатор (компилирующего типа)
- Основные достоинства PHP:
 - простота
 - скорость выполнения
 - низкая ресурсоемкость
 - богатая функциональность
 - кроссплатформенность





PHP и Web

- Интеграция с веб-серверами:
 - автоматическое извлечение POST и GET параметров и переменных окружения
 - автоматическая отправка HTTP
 - работа с HTTP авторизацией
 - работа с cookies и сессиями
 - работа с файлами
 - обработка файлов, загружаемых на сервер
- Взаимодействие с большим количеством различных СУБД, в том числе MySQL

История

- 1994г. – Расмус Лердорф создал набор скриптов для ведения статистики посещения веб-страниц PHP/FI (Personal Home Page / For Interpretator)
- 1996 г. – оформление языка программирования;
- 1998 г. – версия PHP 3.0
 - Энди Гутманс и Зеев Сураски (Израиль)
 - существенно переработан весь язык
 - возможность расширения ядра дополнительными модулями
 - изменение расшифровки PHP = PHP: Hypertext Preprocessor
- 2000 г. – PHP 4.0
- 2004 г. – PHP 5.0



Как работает PHP?

1. HTTP запрос к Веб серверу с указанием PHP сценария
2. Запуск PHP процессора (первоначальный вариант, возможны также реализация в виде специального модуля Веб сервера и подключение по быстрым API интерфейсам) :
3. Передача PHP процессору сценария, параметров, переданных пользователем, и деталей HTTP запроса
4. Выполнение PHP процессором сценария и подготовка заготовки для HTTP ответа, передача ответа Веб серверу
5. Формирование Веб сервером HTTP ответа и отправление ответа клиенту



Hello, World!

```
<HTML>
  <HEAD>
    <TITLE> Hello, World! </TITLE>
  </HEAD>
  <BODY color="blue">
    <P>
      <?php echo 'Hello, World!' ?>
    </P>
  </BODY>
</HTML>
```



Литература

- *Официальный сайт PHP Group.* PHP: Hypertext Preprocessor.
<http://php.net>
- *Нина Савельева.* Язык программирования PHP. - Национальный открытый университет ИНТУИТ. <http://www.intuit.ru>
- *Энди Харрис.* PHP/MySQL для начинающих. - Издательство: КУДИЦ-Образ. 2007 г.