

# Профессиональный практикум программной инженерии

Лекция №6.1: «Документация программных  
продуктов. Использование автоматических  
генераторов документации»

Грищенко Виктор Игоревич  
<[victor.grischenko@gmail.com](mailto:victor.grischenko@gmail.com)>

# Виды программной документации

- Описание
- Справочник программного интерфейса (API Reference)
- Руководство разработчика

# Справочник программного интерфейса

- Классы
- Методы
- Свойства
- Библиотеки

# Сложность поддержки

- Много рутинной работы
- Продукт развивается
- Рефакторинг

# Автоматические генераторы документации

- Файл конфигурации настраивается один раз
- Быстрая генерация
- Поддержка большого числа выходных форматов
- Поддержка в IDE

# Поддержка в IDE

```
2
3
4 $vehicle = new Vehicle;
5 $vehicle->|
6
7
8 /**
9  * Т
10  * К
11  * с
12  * Функция
13  * @
14  */
15
16 class
17
18
19
20 * @var string $brand
21 * @see brand()
22 */
23 protected $brand;
24
25 /**
```

\_\_construct  
brand  
mixed brand( null  
brand = NULL )  
Контейнер: Vehicle  
Доступ: public Тип:  
Функция  
(definition)  
model  
weight

...ное транспортное  
...актеристиками.  
nko <victor.grischenko@gmail.com>

...производитель транспортного средства

# Автоматические генераторы документации

- DocBlox
- Doxygen
- JsDoc3

# Doxygen

- Сайт: [www.doxygen.org](http://www.doxygen.org)
- Поддерживаемые языки:
  - C++, Си, Objective-C, Python, Java, IDL, PHP, C#, Фортран, VHDL
- Форматы вывода:
  - HTML, LaTeX, man, RTF, XML, CHM, PostScript, PDF (через LaTeX)



# Doxygen: документирование класса

```
/**  
 * Транспортное средство  
 *  
 * Класс описывает абстрактное транспортное  
 * средство с базовыми характеристиками.  
 *  
 * @author    John Doe  
 * @version   1.0.1  
 * @copyright GNU Public License  
 * @todo      Реализовать все методы  
 */  
class Vehicle {}
```

# Doxygen: документирование класса

```
/// Транспортное средство
///
/// Класс описывает абстрактное транспортное
/// средство с базовыми характеристиками.
///
/// @author    John Doe
/// @version   1.0.1
/// @copyright GNU Public License
/// @todo      Реализовать все методы
class Vehicle {}
```

# Документ: документирование свойств

```
/**
```

```
* Производитель транспортного средства
```

```
*
```

```
* Используем только простое символьное
```

```
* имя производителя. Если будет
```

```
* необходима детализация, создадим
```

```
* класс Brand
```

```
*
```

```
* @var      string $brand
```

```
*/
```

```
public $brand;
```

# Doxygen: документирование методов

```
/**
 * Работа со свойством {@link $brand}
 *
 * Если аргумент определен, то устанавливается новое
 * значение свойства и возвращается указатель на
 * объект, иначе возвращается текущее значение
 * свойства.
 *
 * @param string $brand Производитель
 * @return mixed Возвращает текущее значение
 *                свойства или указатель на
 *                объект
 */
public function brand($brand=NULL);
```

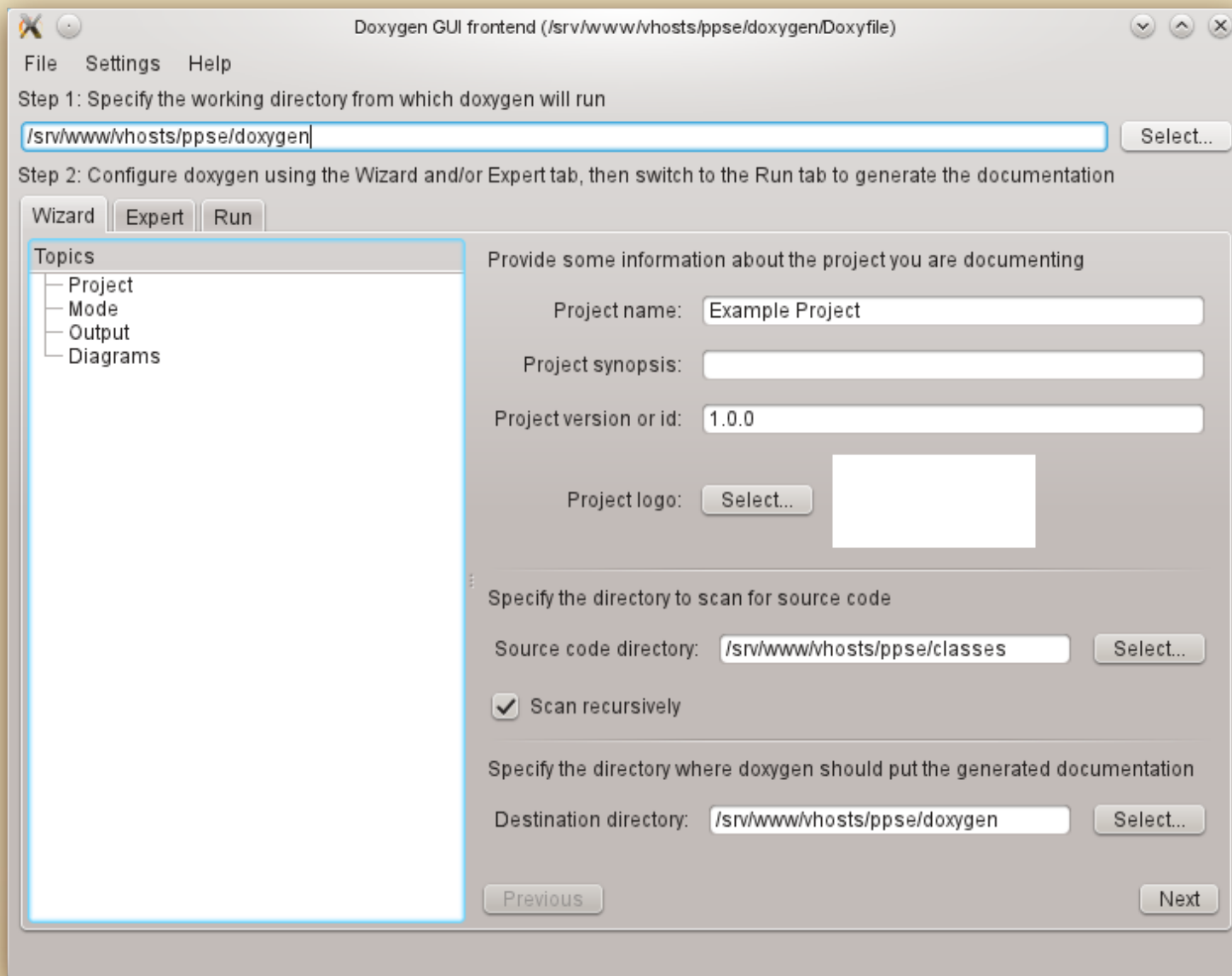
## Doxygen: установка

- Обычное приложение
- Стандартная установка в Windows
- Есть в основных репозиториях OpenSUSE, Ubuntu, Debian, etc.

# Doxygen: настройка

- Файлы конфигурации
  - ini-подобный синтаксис
- Мастер настройки

# Доxygen: настройка проекта



The screenshot shows the Doxygen GUI frontend window titled "Doxygen GUI frontend (/srv/www/vhosts/ppse/doxygen/Doxyfile)". The window has a menu bar with "File", "Settings", and "Help".

**Step 1: Specify the working directory from which doxygen will run**

The working directory is set to `/srv/www/vhosts/ppse/doxygen` in a text field, with a "Select..." button to its right.

**Step 2: Configure doxygen using the Wizard and/or Expert tab, then switch to the Run tab to generate the documentation**

The "Wizard" tab is selected. On the left, a "Topics" list contains "Project", "Mode", "Output", and "Diagrams".

**Provide some information about the project you are documenting**

- Project name: `Example Project`
- Project synopsis: (empty text field)
- Project version or id: `1.0.0`
- Project logo: "Select..." button and an empty image placeholder.

**Specify the directory to scan for source code**

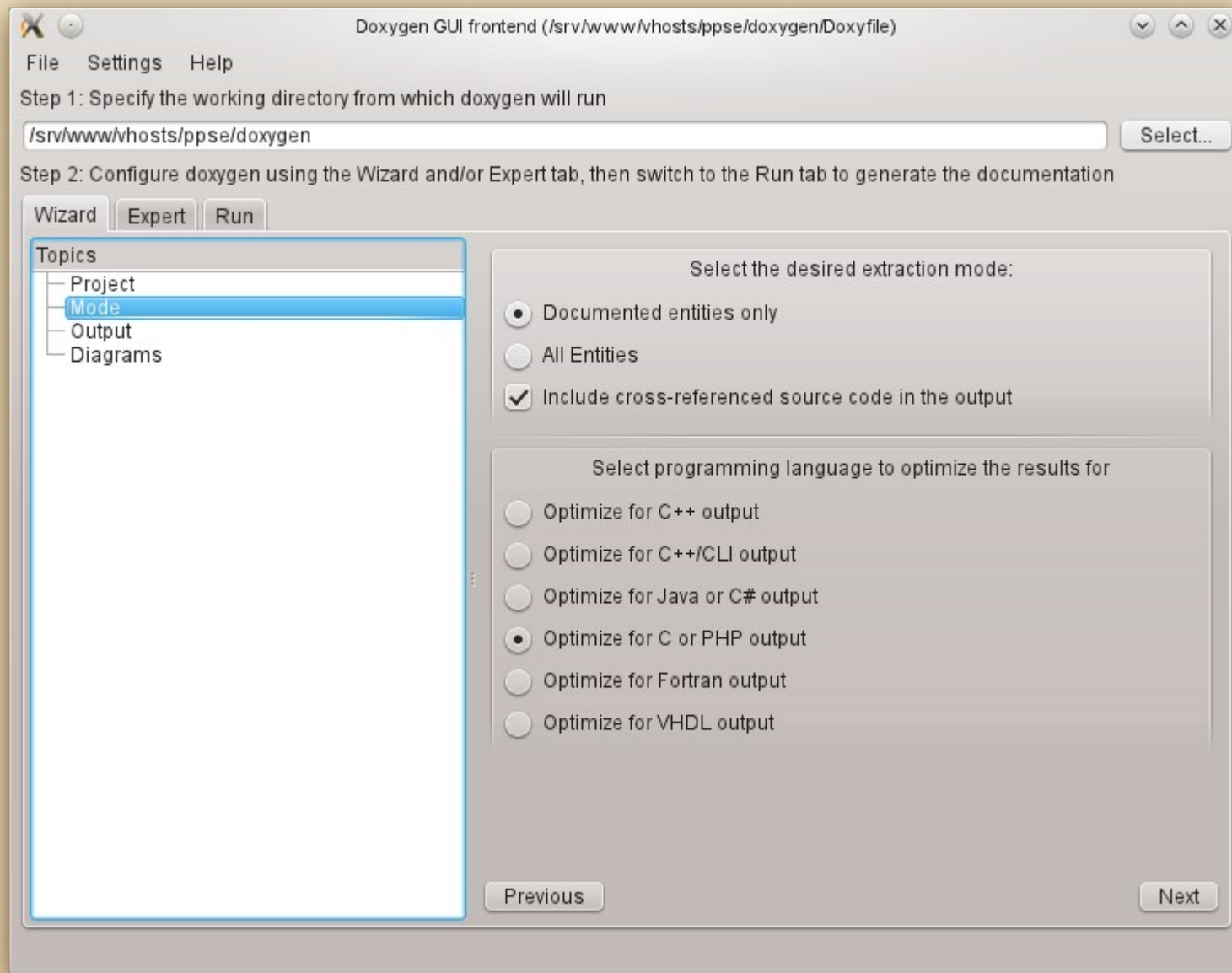
- Source code directory: `/srv/www/vhosts/ppse/classes` with a "Select..." button.
- ☒ Scan recursively

**Specify the directory where doxygen should put the generated documentation**

- Destination directory: `/srv/www/vhosts/ppse/doxygen` with a "Select..." button.

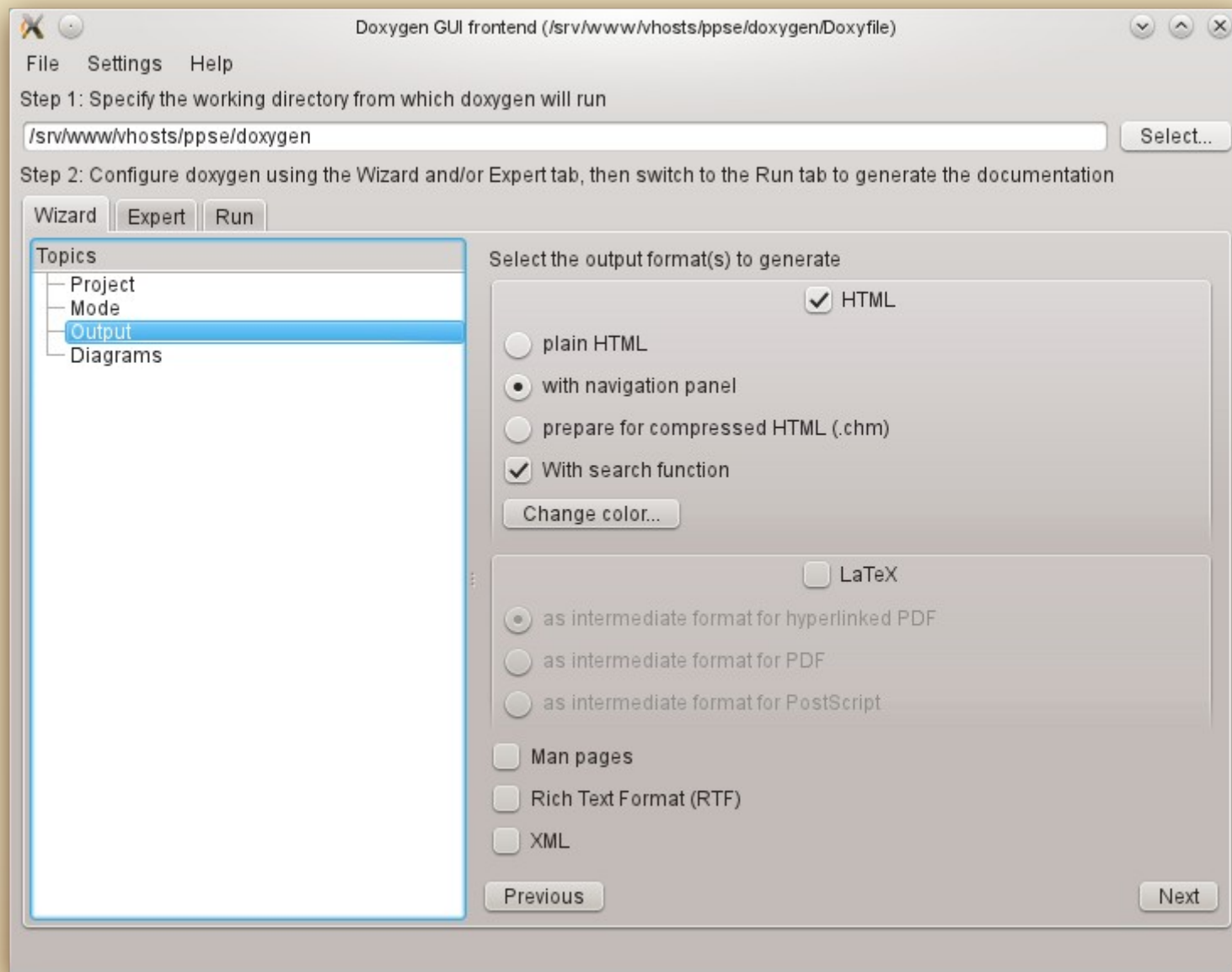
Navigation buttons at the bottom: "Previous" and "Next".

# Доxygen: настройки языка

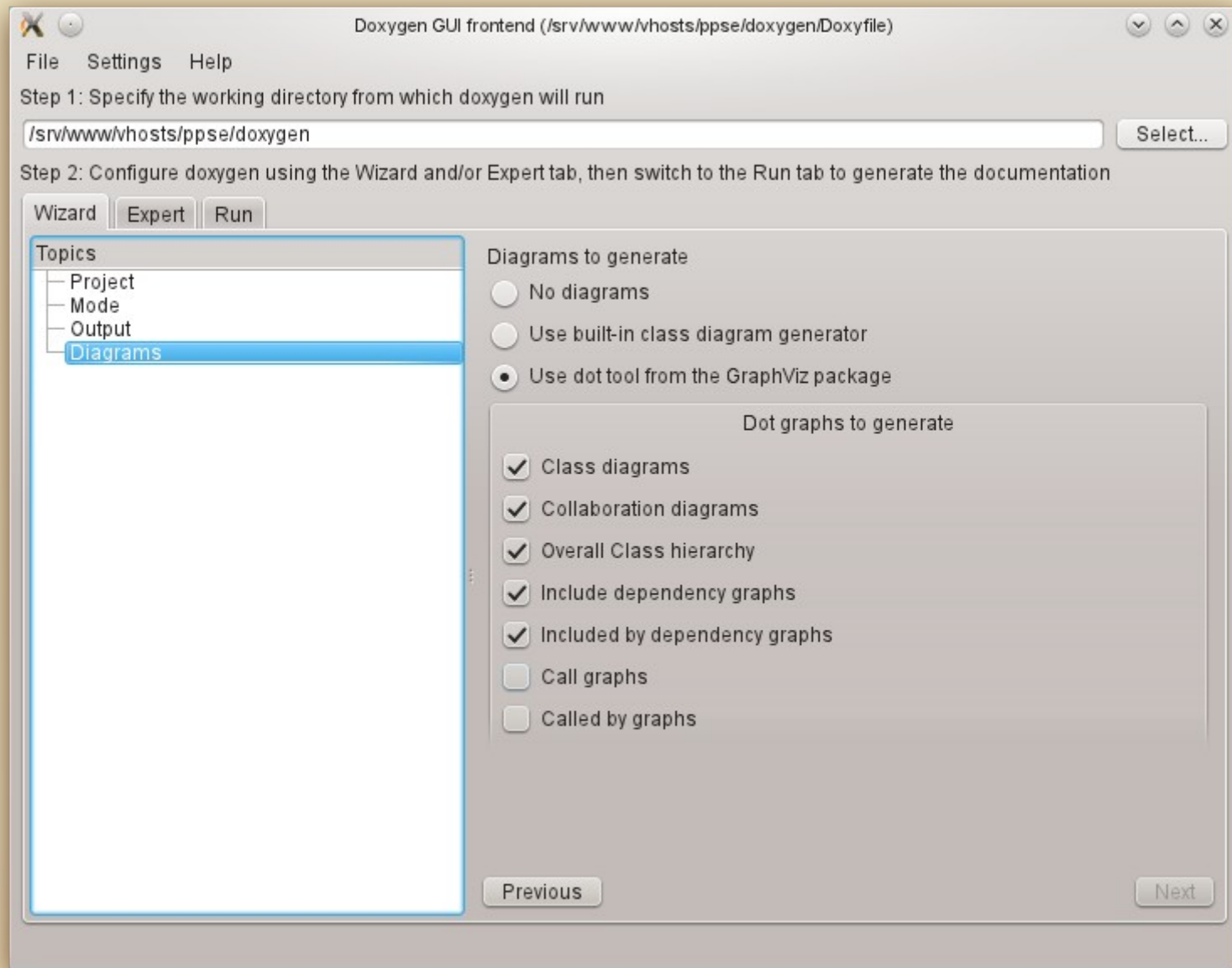




# Doxygen: настройки формата вывода

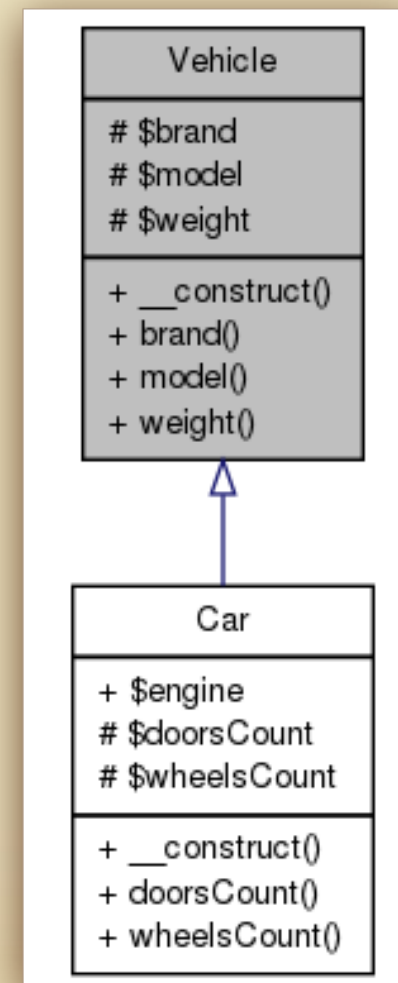
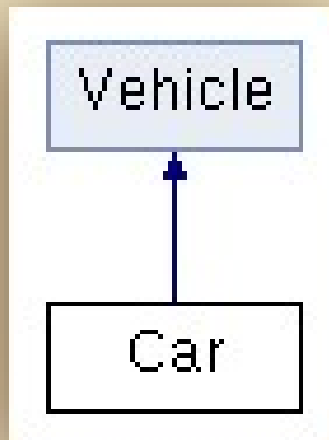


# Доxygen: настройки диаграмм

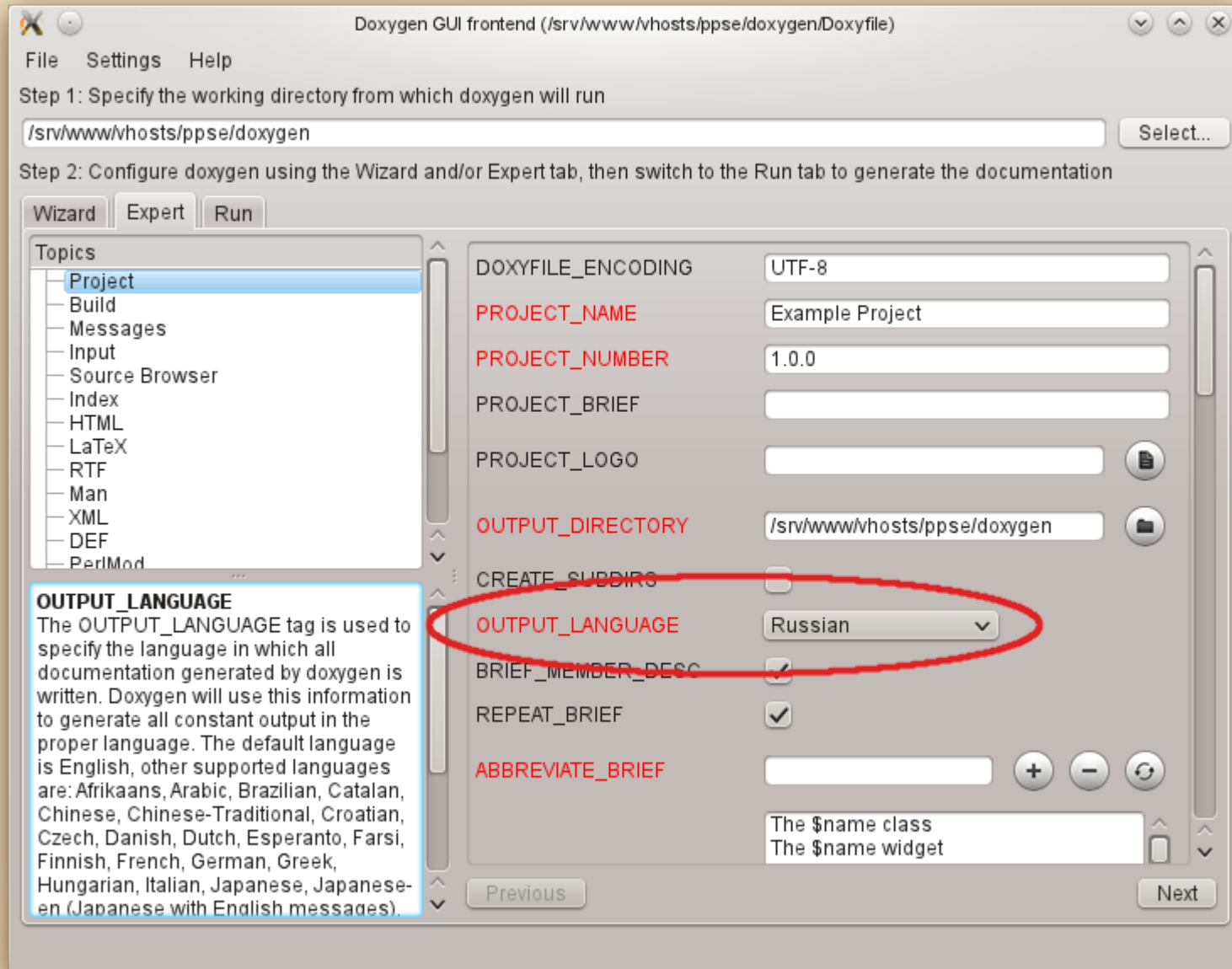


# Генераторы диаграмм

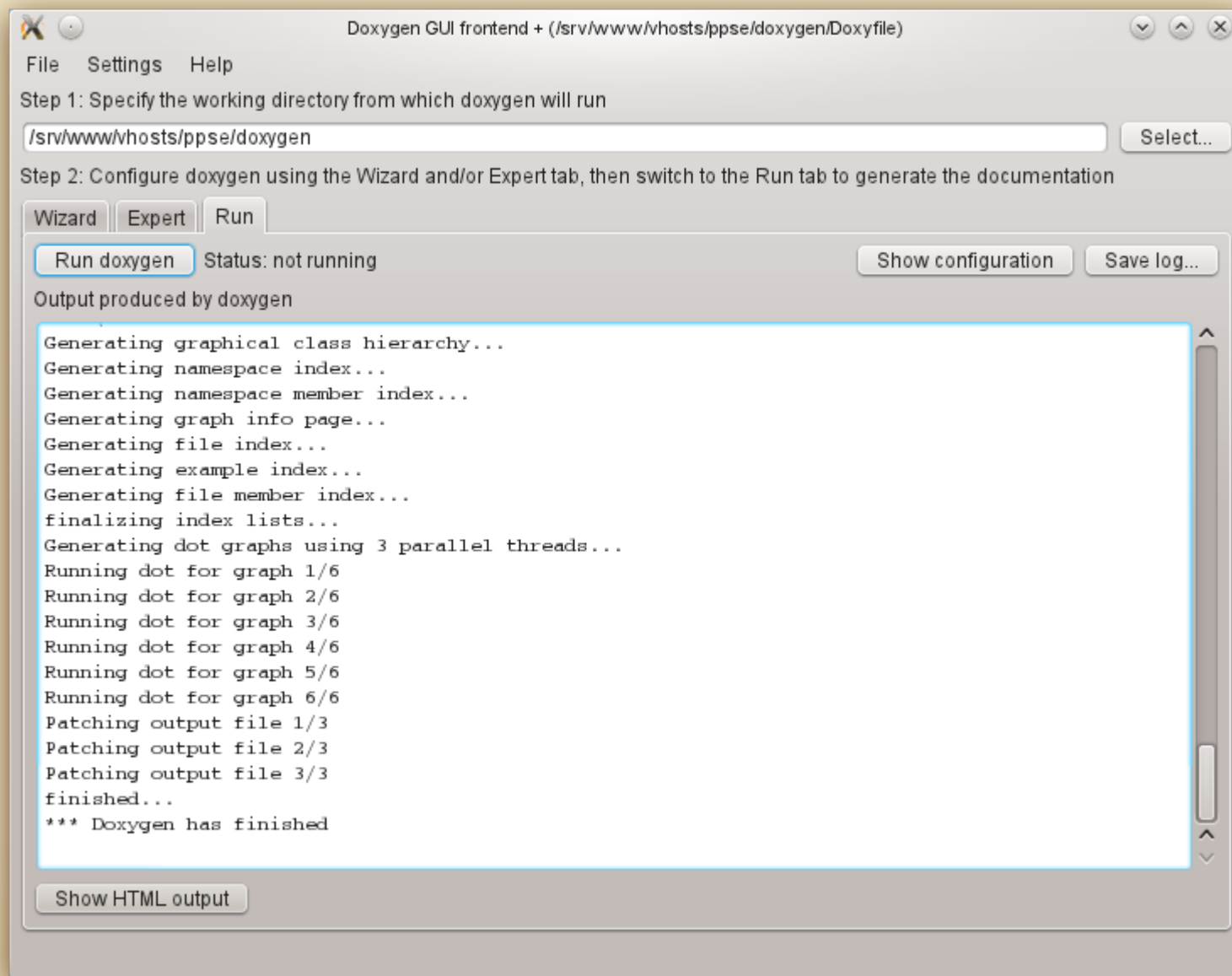
- Встроенный генератор
- Генератор GraphViz



# Доxygen: настройки языка



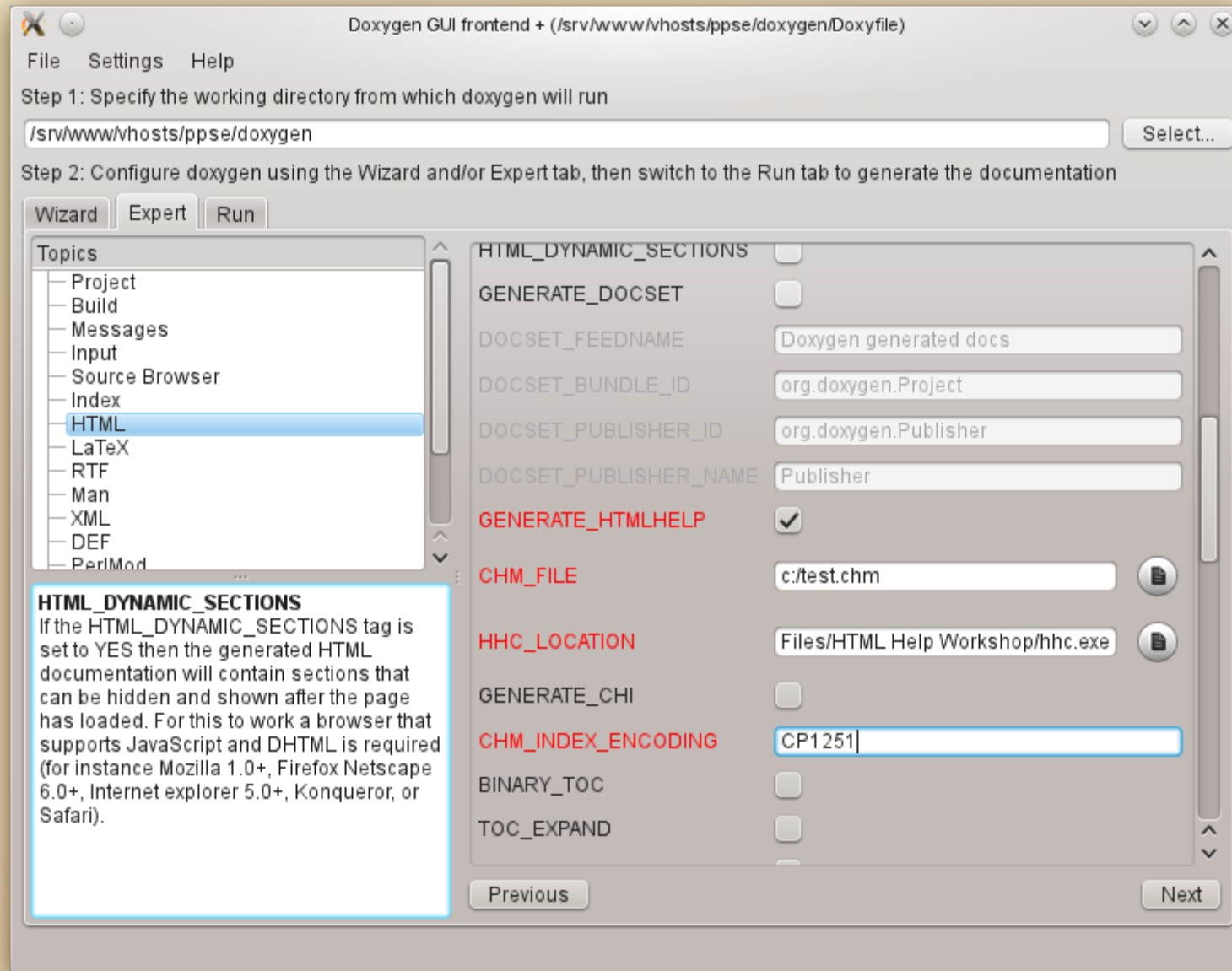
# Доxygen: генерация документации



# Генерация CHM-файлов

- Генерация множества html и файлов содержания
- Сборка CHM с использованием утилиты HTML Help Workshop
  - <http://www.microsoft.com/download/en/details.aspx?displaylang=en&id=21138>

# Дохуген: генерация СНМ



# Doxygen: особенности

- Преимущества
  - Активно развивается
  - Реализован в виде обычного приложения
  - Есть мастер настройки
  - Поддержка большого числа языков
- Недостатки
  - Не полная поддержка PHP
  - Для работы с JavaScript требуются дополнительные преобразования



# JsDoc3

- Сайт: [github.com/jsdoc3](https://github.com/jsdoc3)
- Поддерживаемые языки:
  - JavaScript
- Форматы вывода:
  - HTML