

Практикум АСВК 2016

Архитектура приложения

Никитин Михаил

`mnikitin@graphics.cs.msu.ru`

МГУ имени М.В. Ломоносова

Факультет ВМК

Кафедра АСВК

19 сентября 2016 г.

Задание по практикуму

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

Резюме

Вопросы

Литература

- 1 Разделение логики и интерфейса (3 балла).
- 2 Построение компонентной архитектуры (2 балла).

Разделение логики и интерфейса

Общее описание

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

**Общее
описание**

Преимущества
Model-View-
Controller

Наблюдатель

Пример

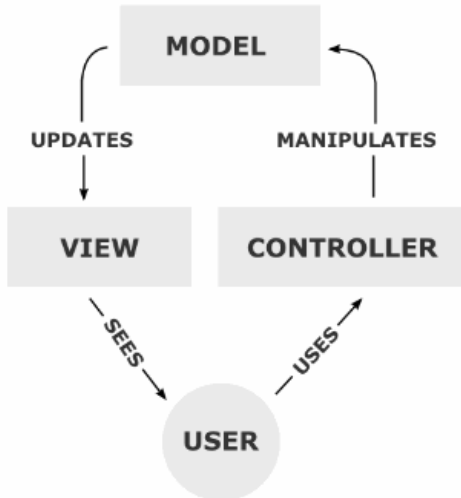
Библиотеки
GUI

Построение
компонент-
ной
архитектуры

Резюме

Вопросы

Литература



Разделение логики и интерфейса

Общее описание

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Общее
описание

Преимущества
Model-View-
Controller
Наблюдатель
Пример
Библиотеки
GUI

Построение
компонент-
ной
архитектуры

Резюме

Вопросы

Литература

3 компоненты:

- 1 Модель (Model);
- 2 Представление (View);
- 3 Контроллер (Controller);

Требование — логика программы не должна зависеть:

- от способа представления:
 - графический / консольный интерфейсы;
 - таблицы / диаграммы;
 - ...
- от способа интерпретации ввода пользователя.

Разделение логики и интерфейса

Преимущества

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Общее
описание

Преимущества

Model-View-
Controller

Наблюдатель

Пример

Библиотеки
GUI

Построение
компонент-
ной
архитектуры

Резюме

Вопросы

Литература

- повторное использование модели;
- использование нескольких представлений;
- использование нескольких контролеров;
- уменьшение количества зависимостей между компонентами;

Разделение логики и интерфейса

Model-View-Controller

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Общее
описание

Преимущества
**Model-View-
Controller**

Наблюдатель

Пример

Библиотеки
GUI

Построение
компонент-
ной
архитектуры

Резюме

Вопросы

Литература

- Пассивная модель:
 - структурное программирование;
 - модель — структура данных;
 - „толстый“ контроллер:
 - логика
 - отслеживание изменения модели;

Разделение логики и интерфейса

Model-View-Controller

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Общее
описание

Преимущества
**Model-View-
Controller**

Наблюдатель

Пример

Библиотеки
G UI

Построение
компонент-
ной
архитектуры

Резюме

Вопросы

Литература

- Пассивная модель:
 - структурное программирование;
 - модель — структура данных;
 - „толстый“ контроллер:
 - логика
 - отслеживание изменения модели;
- Активная модель:
 - Объектно-ориентированное программирование;
 - модель:
 - структура данных;
 - логика приложения;
 - „тонкий“ контроллер:
 - реакция на пользовательский ввод.

Разделение логики и интерфейса

Model-View-Controller

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Общее
описание

Преимущества
**Model-View-
Controller**

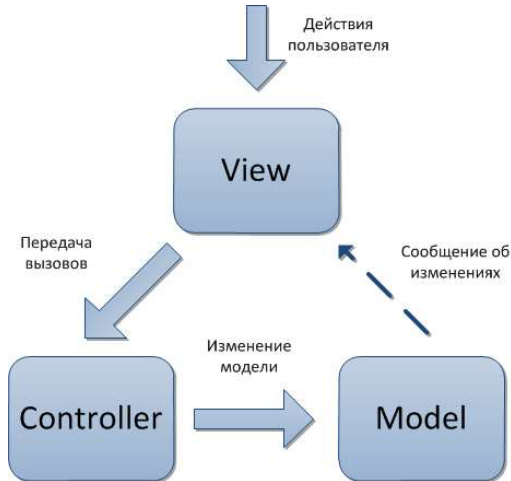
Наблюдатель
Пример
Библиотеки
G UI

Построение
компонент-
ной
архитектуры

Резюме

Вопросы

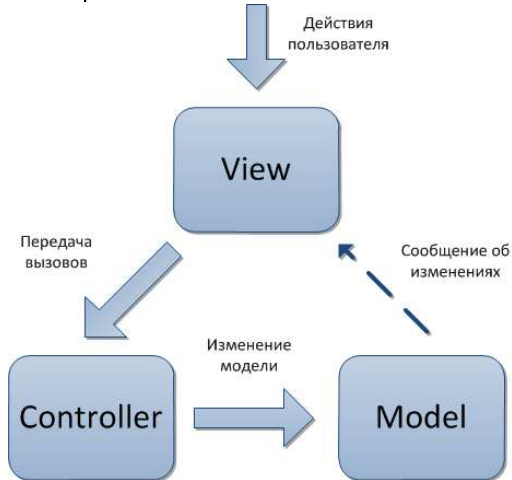
Литература



Разделение логики и интерфейса

Model-View-Controller

Как организовать связь *model* → *view*?



Разделение логики и интерфейса

Наблюдатель

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Общее
описание

Преимущества
Model-View-
Controller

Наблюдатель

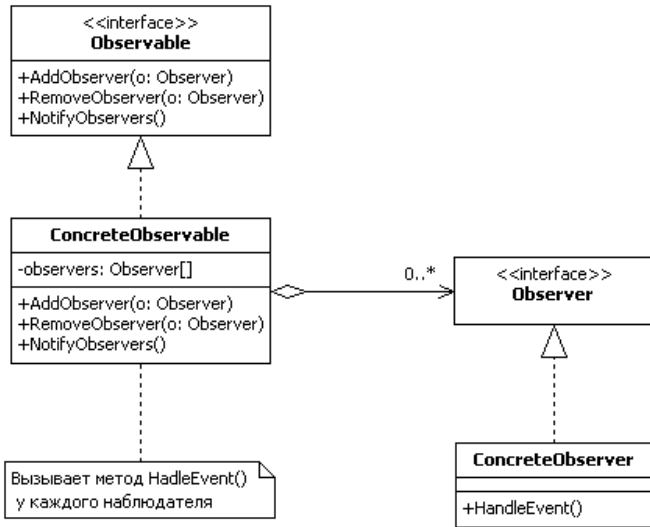
Пример
Библиотеки
GUI

Построение
компонент-
ной
архитектуры

Резюме

Вопросы

Литература



Разделение логики и интерфейса

Наблюдатель

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Общее
описание

Преимущества
Model-View-
Controller

Наблюдатель

Пример
Библиотеки
GUI

Построение
компонент-
ной
архитектуры

Резюме

Вопросы

Литература

```
class Model : public Observable{[...]};  
class ConcreteView1 : public Observer{[...]};  
  
int main (){  
    ConcreteView1 *view1 = new ConcreteView1;  
    ConcreteView2 *view2 = new ConcreteView2;  
  
    Model *model = new Model;  
  
    model->AddObserver(view1);  
    model->AddObserver(view2);  
  
    [...]  
}
```

Разделение логики и интерфейса

Пример

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Общее
описание

Преимущества
Model-View-
Controller

Наблюдатель

Пример

Библиотеки
GUI

Построение
компонент-
ной
архитектуры

Резюме

Вопросы

Литература

- Модель — то, что описано в задании по машграфу;
- Представление — способ вывода состояния программы на экран:
 - изображение;
 - журнал событий (логи);
- Контроллер — обработка пользовательского ввода:
 - разбор параметров командной строки;
 - анализ выбранных пунктов в пользовательском интерфейсе;

Разделение логики и интерфейса

Библиотеки GUI

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Общее
описание

Преимущества
Model-View-
Controller

Наблюдатель

Пример

**Библиотеки
GUI**

Построение
компонент-
ной
архитектуры

Резюме

Вопросы

Литература

- Qt
- MFC (Windows only)
- FLTK
- GTK+
- Ultimate++
- CEGUI

Построение компонентной архитектуры

Что такое компонента?

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

**Что такое
компонента?**

Примеры
применения

Преимущества

Подключение
динамиче-
ских
библиотек в
Linux

Подключение
динамиче-
ских
библиотек в
Windows

Компоненты

Простое
решение

Недостатки

Компонента (plug-in) — программный модуль:

- не зависит от используемого приложения;
- подключается на этапе выполнения;
- обычно расширяет возможности приложения.

Приложение может практически полностью состоять из сборных модулей (плагинов):

- Adobe Photoshop
- Far Manager
- Mozilla Firefox

Построение компонентной архитектуры

Примеры применения

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

Что такое
компонента?

**Примеры
применения**

Преимущества

Подключение
динамиче-
ских
библиотек в
Linux

Подключение
динамиче-
ских
библиотек в
Windows

Компоненты

Простое
решение

Недостатки

- Работа с аудио и видео контентом в разных форматах;
- Обработка изображений с помощью фильтров;
- Спам-фильтры
- Плагины для CMS
- ...

Построение компонентной архитектуры

Преимущества

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

Что такое
компонента?

Примеры
применения

Преимущества

Подключение
динамиче-
ских
библиотек в
Linux

Подключение
динамиче-
ских
библиотек в
Windows

Компоненты

Простое
решение

Недостатки

- Улучшение качества и однородности кода;
- Улучшение модульности проекта;
- Радикальное уменьшение времени сборки;
- Возможность замены компонентов;
- Возможность использования компонент с различными открытыми/закрытыми лицензиями.

Построение компонентной архитектуры

Подключение динамических библиотек в Linux

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

Что такое
компонента?

Примеры
применения

Преимущества

**Подключение
динамиче-
ских
библиотек в
Linux**

Подключение
динамиче-
ских
библиотек в
Windows

Компоненты

Простое
решение

Недостатки

libName.c (или .cpp):

```
#include <dynamic.h>
```

```
extern "C" int func_name()  
{  
    std::cout << "Hello, _World!" << std::endl  
              << "I'm _function _func_name" << std::endl;  
    return 0;  
}
```

Построение компонентной архитектуры

Подключение динамических библиотек в Linux

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

Что такое
компонента?

Примеры
применения

Преимущества

Подключение
динамиче-
ских
библиотек в
Linux

Подключение
динамиче-
ских
библиотек в
Windows

Компоненты

Простое
решение

Недостатки

main.c (или .cpp):

```
#include <dlfcn.h>

...
// load library
void *h = dlopen("libName.so", FLAGS);
// import function
int (*fun)(void) = dlsym(h, "func_name");
...
// call imported function
int x = fun();
...
// unload library
dlclose(h);
```

Построение компонентной архитектуры

Подключение динамических библиотек в Linux

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

Что такое
компонента?

Примеры
применения

Преимущества

**Подключение
динамиче-
ских
библиотек в
Linux**

Подключение
динамиче-
ских
библиотек в
Windows

Компоненты

Простое
решение

Недостатки

Построение объектного файла:

```
gcc -fPIC -c -o libName.o libName.cpp
```

Получение разделяемого модуля:

```
gcc -shared -o libName.so libName.o
```

Сборка main:

```
gcc -ldl main.c
```

Построение компонентной архитектуры

Подключение динамических библиотек в Windows

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

Что такое
компонента?

Примеры
применения

Преимущества

Подключение
динамиче-
ских
библиотек в
Linux

Подключение
динамиче-
ских
библиотек в
Windows

Компоненты

Простое
решение

Недостатки

libName.c (или .cpp)

```
#include <stdio.h>
```

```
__declspec(dllexport) int func_name()  
{  
    std::cout << "Hello, _World!" << std::endl  
              << "I'm _function _func_name" << std::endl;  
    return 0;  
}
```

Построение компонентной архитектуры

Подключение динамических библиотек в Windows

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

Что такое
компонента?

Примеры
применения

Преимущества

Подключение
динамиче-
ских
библиотек в
Linux

Подключение
динамиче-
ских
библиотек в
Windows

Компоненты

Простое
решение

Недостатки

main.c (или .cpp):

```
#include <windows.h>

...
// load library
HINSTANCE h = LoadLibrary("libName.dll");
// import function
int (*fun)(void) =
    (int (*)(void)) GetProcAddress(h, "func_name");
...
// call imported function
int x = fun();
...
// unload library
FreeLibrary(h);
```

Построение компонентной архитектуры

Компоненты

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

Что такое
компонента?

Примеры
применения

Преимущества

Подключение
динамиче-
ских
библиотек в
Linux

Подключение
динамиче-
ских
библиотек в
Windows

Компоненты

Простое
решение

Недостатки

Компоненты подгружаются во время выполнения!

- 1 Поиск компонент;
 - bin/plugins
- 2 Регистрация компонент в системе.

Построение компонентной архитектуры

Простое решение

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

Что такое
компонента?

Примеры
применения

Преимущества

Подключение
динамиче-
ских
библиотек в
Linux

Подключение
динамиче-
ских
библиотек в
Windows

Компоненты

Простое
решение

Недостатки

Компонента:

```
// factory method to construct proper plugin  
IPlugin *createInstance(const char *);
```

Приложение:

```
// method to construct proper plugin  
IPlugin *plugin =  
    createInstance("FilterPlugin");  
IFilter *filter =  
    dynamic_cast<IFilter>(plugin);
```

Построение компонентной архитектуры

Недостатки простого решения

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

Что такое
компонента?

Примеры
применения

Преимущества

Подключение
динамиче-
ских
библиотек в
Linux

Подключение
динамиче-
ских
библиотек в
Windows

Компоненты

Простое
решение

Недостатки

- Одна из трех проблем:
 - 1 ■ Нарушается безопасность типов
 - `static_cast`
 - 2 ■ Ограничивается применение плагинов
 - `dynamic_cast`
 - 3 ■ Необходима разработка сложной и ломкой система динамической идентификации типа данных (RTTI)
 - `QueryInterface`
- Как искать однотипные плагины?
 - по имени?

Построение компонентной архитектуры

Предлагаемое решение

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

Что такое
компонента?

Примеры
применения

Преимущества

Подключение
динамиче-
ских
библиотек в
Linux

Подключение
динамиче-
ских
библиотек в
Windows

Компоненты

Простое
решение

Недостатки

- Интерфейсы определяются в приложении
- Для интерфейсов применяются соглашения COM
 - C++ интерфейс = абстрактный тип данных
- Плагины сами регистрируют себя в системе.

Построение компонентной архитектуры

Фабрики для плагинов

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

Что такое
компонента?

Примеры
применения

Преимущества

Подключение
динамиче-
ских
библиотек в
Linux

Подключение
динамиче-
ских
библиотек в
Windows

Компоненты

Простое
решение

Недостатки

```
template<class Interface>
class IPluginFactory{
public:
    virtual STDMETHODCALLTYPE Interface *Create() = 0;
};

class IRenderer{
public:
    virtual void STDMETHODCALLTYPE BeginScene() = 0;
    virtual void STDMETHODCALLTYPE EndScene() = 0;
};

typedef IPluginFactory<IRenderer> IRendererFactory;
```

Построение компонентной архитектуры

Фабрики для плагинов

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

Что такое
компонента?

Примеры
применения

Преимущества

Подключение
динамиче-
ских
библиотек в
Linux

Подключение
динамиче-
ских
библиотек в
Windows

Компоненты

Простое
решение

Недостатки

Приложение:

```
class IPluginManager{
public:
    // stores the input factory
    virtual void STDMETHODCALLTYPE RegisterRenderer(
        IRendererFactory* in_factory) = 0;
    // stores the input factory
    virtual void STDMETHODCALLTYPE RegisterSceneManager(
        ISceneManagerFactory* in_factory) = 0;
};
```

Компонента:

```
// registers all available factories in
// the input manager.
extern "C" void registerPlugins(
    IPluginManager *in_pluginManager);
```

Построение компонентной архитектуры

Проблемы и размышления

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

Что такое
компонента?

Примеры
применения

Преимущества

Подключение
динамиче-
ских
библиотек в
Linux

Подключение
динамиче-
ских
библиотек в
Windows

Компоненты

Простое
решение

Недостатки

- Поддержка работы из разных сред.
- Версии интерфейсов / библиотек.
- Как искать плагины?

Построение компонентной архитектуры COM

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

Что такое
компонента?

Примеры
применения

Преимущества

Подключение
динамиче-
ских
библиотек в
Linux

Подключение
динамиче-
ских
библиотек в
Windows

Компоненты

Простое
решение

Недостатки

Component Object Model

- Спецификация Microsoft для любого ЯП
- ОО модель:
- Приложение и компонента взаимодействуют через интерфейс
- Идентификация по GUID (Globally Unique Identifier)
- Любой интерфейс — производный от IUnknown
- Минимум 3 метода:
 - 1 QueryInterface()
 - 2 AddRef()
 - 3 Release()

Резюме

Разделение логики и интерфейса

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

Резюме

**Разделение
логики и
интерфейса**

Построение
компонент-
ной
архитектуры

Вопросы

Литература

Резюме

Разделение логики и интерфейса

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

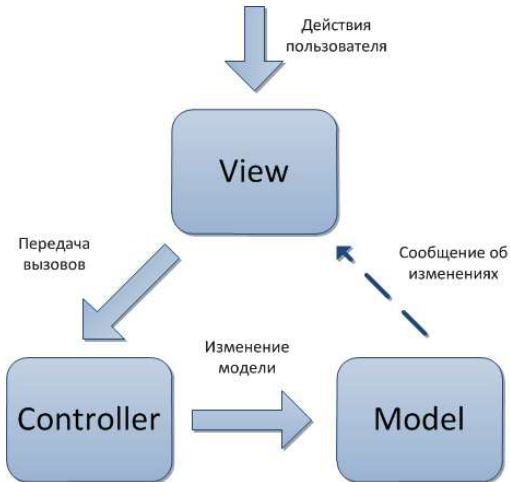
Резюме

**Разделение
логики и
интерфейса**

Построение
компонент-
ной
архитектуры

Вопросы

Литература



Резюме

Построение компонентной архитектуры

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

Резюме

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

Вопросы

Литература

Приложение:

```
class IPluginManager{
public:
    // stores the input factory
    virtual void STDMETHODCALLTYPE RegisterRenderer(
        IRendererFactory* in_factory) = 0;
    // stores the input factory
    virtual void STDMETHODCALLTYPE RegisterSceneManager(
        ISceneManagerFactory* in_factory) = 0;
};
```

Компонента:

```
// registers all available factories in
// the input manager.
extern "C" void registerPlugins(
    IPluginManager *in_pluginManager);
```


Вопросы

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

Резюме

Вопросы

Литература



Литература

Практикум
АСВК 2016

Задание по
практикуму

Разделение
логики и
интерфейса

Построение
компонент-
ной
архитектуры

Резюме

Вопросы

Литература

