

2. Майника Э. Алгоритмы оптимизации на сетях и графах. – М.: «Мир», 1981. – 324 с.  
 3. Кристофидес Н. Теория графов. Алгоритмический подход. – М.: Мир, 1978. – 432 с.

4. Федосеева Л.И., Юрлов А.А. Аппаратная оптимизация графовых задач / Л.И. Федосеева А.А. Юрлов // XXI век: итоги прошлого и проблемы настоящего плюс: пер. науч. изд. – Пенза, №10 (14), 2013, с. 172-175.

**Секция «Проблемы моделирования, проектирования  
и разработки программных средств»,  
научный руководитель – Рыбанов А.А., канд. техн. наук, доцент**

**ИССЛЕДОВАНИЕ МЕТОДОВ ОЦЕНКИ КАЧЕСТВА  
ОНТОЛОГИЧЕСКИХ МОДЕЛЕЙ**

Андрич О.Ф., Макушкина Л.А.

*Волжский политехнический институт, филиал  
Волгоградского государственного технического  
университета, e-mail: andrich\_olga@mail.ru*

Вопрос оценки качества создаваемых онтологических моделей является одной из актуальных проблем современного онтологического инжиниринга. Процесс разработки онтологических моделей важен в практическом плане, и это является причиной того, что разными группами ученых разработано множество различных подходов в области оценки онтологических моделей.

В настоящее время известно более десятка методов, и задача выбора подходящей методики для решения конкретной задачи становится все более сложной.

Целью данной работы является повышение качества онтологических моделей за счет выработки рекомендаций по их построению.

Существует несколько методов оценки качества построенных онтологических моделей [1]:

- FIGO
- OntoMetric
- EvaLexon
- Natural Language Application metrics
- OntoClean
- Declarative Methods

Данные методы проводят оценку онтологических моделей по следующим критериям:

- Полнота и точность словаря предметной области.
- Адекватность структуры с точки зрения таксономии, отношений и т.п.
- Восприимчивость (с когнитивной точки зрения).
- Производительность.
- Выбор лучшей онтологии из нескольких имеющихся.

Для построения более качественных онтологических моделей необходимо проанализировать существующие методы оценки онтологических моделей, определить недостатки в данных методах, и устранить их, а также в результате анализа усовершенствовать существующие онтологические модели.

Разрабатываемая система оценки качества готовых онтологических моделей предназначена для проведения оценки онтологической модели на основе методов: FIGO, OntoMetric; EvaLexon; Natural Language Application metrics; OntoClean; Declarative Methods – для выдачи рекомендаций по повышению качества модели: по классам, связям (где устранить лишнюю связь, либо добавить новую), а также для оценки сложности модели.

В результате проведенных исследований будет представлено формализованное описание математической модели модуля оценки качества онтологических моделей, а также разработано программное средство оценивающее качество онтологических моделей.

**Список литературы**

1. Hartmann J. Methods for ontology evaluation // Knowledge Web Deliverable 2005. С. 11-29.
2. Горовой В.А. Модель классификации методов оценки онтологий // Материалы 2-й международной молодежной конференции «Искусственный интеллект: философия, методология, инновации». Санкт-Петербург, 15-17 ноября 2007 – с. 307-310.
3. Gangemi A., Catenacci C., Ciaramita M., Lehmann J. Ontology evaluation and validation // An integrated formal model for the quality diagnostic task. 2005 С. 30-36.
4. Сайт междисциплинарных исследований – <http://www.agpl.ru/forum/2-----/4---.html>.

**ШАБЛОН ПРОЕКТИРОВАНИЯ MVC КАК  
ЭФФЕКТИВНОЕ СРЕДСТВО ПОСТРОЕНИЯ  
АРХИТЕКТУРЫ ПРОГРАММНОЙ СИСТЕМЫ**

Симонова О.Н., Лясин Д.Н.

*Волжский политехнический институт, филиал  
Волгоградского государственного технического  
университета, Волжский, Россия,  
e-mail: olga\_troshchenko@mail.ru*

В настоящее время весьма часто перед начинающим программистом встает проблема структуризации кода. Наиболее действенным методом в решении данной проблемы является применение шаблона проектирования (паттерна).

Паттерн, или шаблон проектирования представляет собой модель взаимодействия классов для решения какой-либо типичной задачи.

Применение шаблонов проектирования при разработке программного продукта обеспечивает:

- 1) одинаковое понимание последовательности действий, которые необходимы для решения поставленной задачи, и, как следствие, сокращение времени выполнения поставленной задачи.
- 2) использование шаблонов проектирования грамотно структурирует программный код, что благотворно влияет на эффективность работы разрабатываемого приложения.

Таким образом, можно сделать вывод: шаблон проектирования (паттерн) является важнейшим инструментом, позволяющим облегчить работу начинающим сотрудникам и увеличить эффективность работы специалиста.

Выделяют 4 основных группы шаблонов:

1. Фундаментальные шаблоны;
2. Порождающие шаблоны;
3. Структурные шаблоны;
4. Поведенческие шаблоны.

Отдельно можно выделить аналитические, коммуникационные, организационные шаблоны, шаблон MVC (данный шаблон заслуживает более детального рассмотрения).

Model-view-controller (MVC) – шаблон проектирования, с помощью которого его компоненты (модель данных приложения, пользовательский интерфейс и взаимодействие с пользователем) разделены на три отдельных так, что модификация одного из них оказывает минимальное воздействие на остальные. Модель предоставляет данные и методы работы с ними, реагирует на запросы, изменяя своё состояние. Представление отвечает за визуализацию.

Часто в качестве представления выступает форма с графическими элементами. Контроллер обеспечивает ввод данных пользователем и использует мо-

дель и представление для реализации необходимой реакции [2].

Схема взаимодействия компонентов приведена ниже:



Главным преимуществом концепции MVC является разделение логики управления приложения, получения данных и их отображения.

В ходе исследования необходимо изучить методы построения архитектуры программной системы и создать эффективную программную систему с использованием шаблона Model-View-Controller.

Для достижения поставленной цели, необходимо решение следующих исследовательских задач:

1. Разработка формального описания функционирования программной системы на основе шаблона MVC.
2. Разработка программной системы с использованием шаблона MVC.
3. Оценка эффективности программной архитектуры с использованием шаблона MVC.

**Список литературы**

1. Кухаренко Б.Г. Принцип открытости-закрытости в программной инженерии и паттерны проектирования, Программная инженерия, №6, 2011 – 2-15 с.
2. Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. Приемы объектно-ориентированного проектирования. Паттерны проектирования– Питер, 2010 – 366 с.

**ИССЛЕДОВАНИЕ МЕТОДОВ  
АВТОМАТИЗИРОВАННОЙ КЛАССИФИКАЦИИ  
ЗАПУСКАЕМЫХ НА КОМПЬЮТЕРЕ ПРИЛОЖЕНИЙ**

Чегурихина Д.Ю., Лясин Д.Н.

*Волжский политехнический институт, филиал ФГБОУ ВПО «Волгоградский государственный технический университет», Волжский, e-mail: dianachegurihin@mail.ru*

Для того чтобы помочь пользователям разобраться с программами, установленными на их компьютере, оптимизировать работу на компьютере, освободить память занимаемую приложениями, проверить загружает ли программа что-то из интернета и т.д. , существуют приложения, которые в основном интегрированы в антивирусы, но нет программ в свобод-

ном доступе, которая показывает важные для пользователя данные о программах.

Главная задача исследования – помочь пользователю выяснить, какие программы ему нужны обязательно, а без каких он может спокойно обойтись.

Чтобы реализовать программу, осуществляющую автоматизированную классификацию запущенных на компьютере приложений, необходимо произвести многокритериальную классификацию приложений, и отнести запущенное приложение к какому либо разработанному кластеру.

Осуществление классификации приложений удобно производить с помощью Data Mining. Data Mining – не один метод, а некая совокупность различных методов обнаружения знаний. Сам выбор метода во многом будет зависеть от типа имеющихся данных и от того какую информацию необходимо получить в результате.

Выделяют пять стандартных типов закономерностей, позволяющие выявлять методы Data Mining (рисунок).

Для будущего исследования наиболее оптимальным типом закономерности является кластеризация, так как в результате её решения происходит разбиение объектов на группы.

Применение кластерного анализа можно свести к следующим этапам:

1. Выбор объектов для кластеризации.
2. Определить множество переменных, по которым будут оцениваться объекты в выборке, если есть необходимость –нужно нормализовать значения переменных.
3. Вычислить значения меры сходства между всеми объектами.
4. Применить метод кластерного анализа и создать группы схожих объектов (кластеров).
5. Представить результаты анализа.