

Использование параллельных и фасетных заданий в системе дистанционного обучения Moodle

В статье рассматриваются особенности и достоинства параллельных и фасетных тестов и заданий, их реализация в популярных системах дистанционного обучения. Описываются принципы работы и достоинства разработанного модуля для СДО Moodle, реализующего фасетный подход к заданиям и позволяющим дальнейшее расширение.

Ключевые слова: фасетные задания, параллельные задания, СДО, Moodle, poasassignment, лабораторные работы по вариантам.

USE OF PARALLEL AND FACET ASSIGNMENTS IN THE SYSTEM OF DISTANT LEARNING "MOODLE"

The paper describes features and advantages of parallel and facet tests and tasks, their implementation in various popular learning management systems. Principles of functioning and advantages of developed module for LMS Moodle are described. The developed module implements facet approach and allows further development.

Keywords: facet assignments, parallel assignments, LMS, Moodle, poasassignment, labs by variants.

1. Постановка задачи

В образовательном процессе преподаватели и учителя постоянно встречаются с необходимостью контроля знаний обучающихся. Обычно подобная задача решается с помощью самостоятельных и контрольных работ. Распространенным методом проверки знаний студентов является проведение таких работ с применением параллельных заданий. Параллельными называются задания, имеющие сходное содержание из одной и той общей дидактической единицы, примерно совпадающие меры трудности и вариации результатов испытуемых [1].

Многие виды параллельных заданий – фасетные задания, или фасеты. Фасетами называются задания, которые можно представить в следующем виде [2]:

Функция $\{y = x, y = x^2, y = -x^2, y = x^3\}$ является:

- а) четной;
- б) нечетной;
- в) ни четной, ни нечетной.

В данном примере в фигурных скобках записано фасетное мно-

жество. Выбирая по одному из значений множества можно составить 4 параллельных задания. В фасетном задании может быть больше одного фасетного множества, тогда количество конструируемых параллельных заданий с помощью этого фасета равно произведению мощности всех множеств.

Так выглядит фасетный тест. Фасетные задания используются реже [3], так как требуют больше времени на подготовку и проверку. Однако в высшей школе фасетные задания, как правило, используются при проведении курсовых проектов, семестровых и курсовых работ. Рассмотрим несколько типичных примеров фасетных заданий из практики Волгоградского государственного технического университета (ВГТУ). Семестровая работа по электротехнике проводится следующим образом: студентам выдают несколько общих заданий – расчет параметров электроцепей. У всех одинаковые цепи, но конкретные значения проводимостей их элементов определяются по номеру варианта.

В большинстве гуманитарных дисциплин, преподаваемых в ВГТУ, в качестве семестрового задания студенты должны написать реферат на одну из предложенных тем, что также можно считать фасетом. Количество тем при этом обычно больше числа студентов в одной группе.

При использовании фасетных заданий возникает задача распределения заданий между учащимися. Можно выделить следующие методы распределения вариантов:

Вар	Операторные проводимости				
	Y ₁	Y ₂	Y ₃	Y ₄	Y ₅
1	G ₁	G ₂	pC ₁	G ₃	pC ₂
2	pC ₁	pC ₂	G ₁	pC ₃	G ₂
3	pC ₁	G ₁	G ₂	G ₃	pC ₂
4	pC ₁	G ₁	pC ₂	G ₂	pC ₃
5	G ₁	pC ₁	pC ₂	pC ₃	G ₂
6	G ₁	pC ₁	G ₂	pC ₂	pC ₃
7	G ₁	pC ₁	G ₂	pC ₂	G ₃
8	pC ₁	pC ₂	G ₁	G ₂	pC ₃
9	G ₁	pC ₁	G ₂	G ₃	pC ₂
10	pC ₁	pC ₂	G ₁	pC ₃	G ₂

Рис. 1. Таблица значений для задания по Электротехнике



Анатолий Андреевич Ерофеев,
магистрант кафедры
Программное обеспечение
автоматизированных систем
Тел.: 8 (917) 843-7911
Эл. почта: yerofeev.aa@gmail.com
Волгоградский государственный
технический университет
www.vstu.ru

Anatoly A. Yerofeev,
graduate student,
the Automated Systems Software
Department
Tel.: 8 (917) 843-7911
E-mail: yerofeev.aa@gmail.com
Volgograd State Technical University
www.vstu.ru/en



Олег Александрович Сычев,
к.т.н., доцент кафедры
Программное обеспечение
автоматизированных систем
Тел.: 8 (905) 396-45-60
Эл. почта: oasychev@gmail.com
Волгоградский государственный
технический университет
www.vstu.ru

Oleg A. Sychev,
Candidate of Engineering Science,
Associate Professor,
the Automated Systems Software
Department
Tel.: 8 (905) 396-45-60
E-mail: oasychev@gmail.com
Volgograd State Technical University
www.vstu.ru/en

- случайное распределение заданий (вытягивание экзаменационного билета);
- самостоятельный выбор задания каждым студентом;
- выбор задания по определенной формуле (основываясь на порядковом номере обучающегося в группе/классе или на цифрах его зачетной книжки);
- назначение преподавателем или старостой (в таком случае может поддерживаться обратная связь – пожелания обучающихся могут учитываться);
- распределение заданий с учетом сложности (задания повышенной сложности повышают рейтинговую оценку).

В ВГТУ студенты первого курса факультета электроники и вычислительной техники в рамках дисциплины «Основы программирования» в течение первого семестра обучения выполняют шесть лабораторных работ, связанных с самостоятельной разработкой программ (в каждой в среднем по 30 вариантов заданий, что можно считать своеобразным фасетом, причем к каждому из которых составлено не менее 3-х отчетных задания). Ставится требование уникальности выполняемого студентом задания в пределах группы. Во втором семестре четыре лабораторных работы, выполняемые по схожей технологии. Однако кроме индивидуальных вариантов заданий в них используются также фасетные требования, например способ ввода/вывода перечней (радиокнопки, список или комбинированный перечень), вид используемых программой контейнеров (ассоциативные или последовательные) и итераторов (в стиле Java или STL) и т.д. Такая практика позволяет значительно затруднить списывание работ в соседних группах и у старших курсов, так как при совпадении вариантов задания обычно отличаются фасетные требования. Дисциплина потоковая, обучаются четыре группы до тридцати человек в каждой.

Упомянутая выше проблема с распределением заданий между студентами встречалась и в этой дисциплине. Изначально старосте группы выдавался список заданий с

графой для фамилии студента и поручалось давать выбирать задания с контролем уникальности в пределах группы. Такой способ имел ряд недостатков. Часто по недосмотру старосты (а иногда и по умыслу, надеясь сдать чужую работу) одно и то же задание могли получить несколько человек в одной группе. Разрешение таких конфликтов происходит нелегко, так как каждый студент настаивает на своей правоте и не хочет переделывать задачу, а способа объективной проверки самостоятельного выбора задания не существовало. Нередки были и случаи выбора варианта под списывание: студенты разных групп по сговору брали одинаковые варианты или же студент предварительно выяснял, какие задания можно списать у старшекурсников.

2. Применение систем дистанционного обучения в образовании

Системы дистанционного обучения (СДО) за годы своего существования не только открыли совершенно новый формат образования – дистанционный, но и позволили повысить эффективность традиционного образования. В России данный формат получил не столь широкое распространение, как за рубежом, в основном по техническим причинам [4]. Тем не менее наблюдается тенденция к улучшению этой ситуации, и СДО с каждым годом набирают все большую популярность.

Внедрение СДО в систему очного образования оказывает положительное влияние на систему образования. Повышение эффективности достигается за счет автоматизации ряда задач, которые до этого преподаватель выполнял вручную – ведение журналов, подготовка и проведение контрольных, самостоятельных работ или экзамена; а также возможности доступа студентами ко многим функциям СДО через сеть интернет в удобное для них время [5].

На сегодняшний день разработаны десятки СДО, как коммерческих (Blackboard, eCollege, Glow, Ning), так и бесплатных (aTutor, Dokeos, LAMS, Moodle, OpenACS, Sakai) [6].

Результат сравнения решений для управления заданиями

Модуль	Задания	Способ организации фасета	Методики распределения заданий	Контроль уникальности задания в пределах группы	СДО
Poasassignment (разработанный модуль)	Общее или фасетные	Настраиваемые поля различных типов	4 настраиваемых методики	Есть	Moodle
Assignment	Общее	—	—	—	Moodle
Random assignment	Фасетные	Файлы	Случайное распределение	—	Moodle
Assignments	Общее	—	—	—	Sakai
Assignment	Общее	—	—	—	Blackboard

Одной из популярнейших является бесплатная система Moodle. Под ее управлением работают более семидесяти тысяч сайтов по всему миру, больше тысячи в России [7, 8]. Система постоянно обновляется и совершенствуется.

СДО Moodle применяется и в ВГТУ. Для упомянутой ранее дисциплины «Основы программирования» в системе создан раздел, в котором студенты могут скачать лекции по предмету, дополнительные материалы, проверить свою подготовленность к занятиям с помощью тестов (контрольное тестирование на занятиях также проводится на сайте СДО). До весеннего семестра 2012 г. на лабораторных занятиях студенты использовали Moodle только для выполнения тестов входного контроля, а также загрузки на сайт исходных кодов своих программ для проверки, после того как лабораторная работа была выполнена и отчитана.

В связи с описанными ранее проблемами выдачи заданий возникла потребность в автоматизации процесса выдачи и контроля уникальности в пределах группы индивидуальных заданий с помощью СДО Moodle. Для системы не существует модуля, который позволял бы выдавать различным студентам разные задания. Стандартный модуль заданий Assignment позволяет выдавать лишь общее задание всем студентам, что не подходит при автоматизации параллельных и фасетных заданий.

3. Существующие решения для управления заданиями студентов

Перед разработкой нового модуля был произведен обзор аналогичных решений. Предпочтение

отдавалось модулям, которые могли бы:

- работать с параллельными и фасетными заданиями;
- распределять задания между студентами по разным методикам (с учетом настроек уникальности);
- следить за ходом выполнения заданий студентами.

Среди всех модулей для СДО, как стандартных, так и написанных сторонними разработчиками, были выделены следующие:

- Assignments для СДО Sakai [9];
- Assignment, стандартный модуль Moodle;
- Random assignment, разработка L. Cirka и M. Fikar (Словацкий технический университет в Братиславе) для СДО Moodle [10];
- Assignment для СДО Blackboard.

Результат сравнения найденных решений по ключевым критериям представлен в таблице.

Как видно из таблицы, ни одно из существующих решений не удовлетворяет всех выдвинутых требований.

Решение начать разработку нового модуля было принято после анализа архитектуры стандартного Assignment. Данный модуль допускает расширение (Random assignment является одним из таких расширений), но эти расширения невозможно комбинировать друг с другом — нельзя совместить случайную выдачу из одного дополнения и технологию черновиков из другого. Разработанный много лет назад модуль Assignment страдал большим количеством архитектурных недостатков. Поэтому была спроектирована более гибкая архитектура для нового модуля.

Разработанный модуль состоит из нескольких взаимосвязанных частей:

- ядро;
- подсистема плагинов — ответов студентов;
- подсистема плагинов — методик распределения заданий.

Для понимания работы модуля будут необходимы следующие понятия.

Экземпляр модуля Poasassignment — совокупность заданий, одно из которых должен выполнить студент, чтобы данный вид работы считался выполненным. Каждая лабораторная работа в нашем примере с дисциплиной «Основы программирования» — это экземпляр модуля.

Задание — формулировка задачи, которую необходимо решить студенту. Состоит из названия, описания и дополнительных полей.

Поле задания — дополнительный тип информации о решаемой задаче в определенном экземпляре модуля Poasassignment.

Значение поля задания — дополнительная информация о конкретном задании.

Задания и поля заданий задаются на отдельных страницах модуля (рис. 2 и 3).

Возможность создавать параллельные задания реализована в ядре модуля. При создании экземпляра модуля производится настройка дополнительных полей заданий данного экземпляра. С точки зрения пользователя, создание параллельных заданий представляет собой заполнение таблицы заданий, в которой у каждого задания есть название, описание и дополнительные поля.

В приведенном на рис. 2 примере создано поле типа «Список с множественным выбором». Количество

Разработка дистанционных курсов в системе Moodle

Вы вошли под именем Ерофеев Анатолий (Выход)

В начало ► Мои курсы ► Moodle_dev ► Тема 5 ► Лабораторная работа №3 ► Поля таблицы заданий Обновить POAS assignment

Навигация

- В начало
- Моя домашняя страница
- Страницы сайта
- Мой профиль
- Мои курсы
 - Moodle_dev
 - Участники
 - Отчеты
 - Общее
 - Тема 1

Лабораторная работа №3 : Поля таблицы заданий

Добавить новое поле

Название поля	Тип поля	Диапазон	Случайное поле	Скрытое поле	Показывать в таблице
Уровень сложности	Список со множественным выбором	[Простая, Средняя, Сложная]	✗	✓	✓

Рис. 2. Таблица полей заданий

Разработка дистанционных курсов в системе Moodle

Вы вошли под именем Ерофеев Анатолий (Выход)

В начало ► Мои курсы ► Moodle_dev ► Тема 5 ► Лабораторная работа №3 ► Задания Обновить POAS assignment

Навигация

- В начало
- Моя домашняя страница
- Страницы сайта
- Мой профиль
- Мои курсы
 - Moodle_dev
 - Участники
 - Отчеты
 - Общее
 - Тема 1
 - Тема 2
 - Тема 3
 - Тема 4
 - Тема 5
 - Лабораторная работа №3

Лабораторная работа №3 : Задания

Название задания	Описание задания	Уровень сложности
Работа с массивами1	Выполнить сортировку по ...	Простая Средняя
Работа с массивами2	Найти повторяющиеся ...	Средняя Сложная
Работа с массивами3	Найти максимальный элемент ...	Сложная
Работа с массивами4	Найти моду массива	Простая Сложная
Работа с массивами5	Посчитать количество ...	Средняя

Добавить задание

Рис. 3. Таблица заданий

полей в экземпляре модуля не ограничено, каждое поле может быть:

- целым или дробным числом (с указанием допустимых диапазонов);
- строкой;
- текстом;
- файлом;
- датой;
- списком элементов (этот тип поля позволяет создать фасетные задания в их классическом виде, приведенном в начале статьи);
- списком с множественным выбором.

Числовые поля и поля-списки можно сделать случайными, так что поле будет получать значение само, сразу после того, как студент получит задание, что дает большие возможности реализации фасетных заданий: достаточно описать множество возможных

значений полей, и каждый студент получит набор их случайных значений вместе с заданием, а преподаватель сможет его просмотреть. Преподаватель может настроить, когда студент сможет увидеть значение поля: заранее или только после получения задания.

Поля задания помогают сделать каждое задание индивидуальным, даже если исходных формулировок задач меньше, чем количество студентов на потоке. Числовые поля могут использоваться как коэффициенты в уравнениях. Поля-даты упрощают заполнение задания за счет удобной навигации по датам при создании фасетного задания. Поля-списки могут использоваться, когда список альтернатив заранее известен. Например, в курсе «Основы программирования» поле типа список с множественным выбором используется для обозначения уров-

ня сложности текущего задания – «простая», «средняя», «сложная».

После настройки необходимых полей заданий можно перейти к созданию самих заданий (рис. 3). Для каждого задания необходимо задать его название и описание, а также все созданные в экземпляре модуля поля (кроме случайных). Количество заданий, как и полей, не ограничено и для каждого экземпляра Poasassignment задается отдельно.

В модуле используются 4 методики распределения заданий:

- случайное распределение;
- самостоятельный выбор;
- параметрический выбор;
- «из другого экземпляра» – студент получает тот же вариант, что был у него в другом экземпляре модуля, применяется в случае наличия серии взаимосвязанных заданий.

При параметрическом выборе студенты могут указать желаемые значения полей, описывающие, какое задание они хотели бы получить, а система случайным образом выбирает одно из заданий с этими параметрами. Это дает возможность совместить случайное распределение заданий с возможностью студента выбрать уровень сложности или примерную тематику работы. Чтобы методика параметрического выбора работала, преподаватель должен указать, по каким полям студент может указывать свои пожелания к заданиям.

При любой выбранной методике можно настроить уникальность распределения – в пределах группы, в пределах потока, в пределах курса, без контроля уникальности. Для случая, когда студентов больше, чем заданий, предусмотрена опция, которая умеет выдавать задания «по кругу» – когда все задания разобраны хотя бы по одному разу, она начинает снова их выдавать, стремясь обеспечить максимально возможное разнообразие.

К примеру, реализация семестровой работы по электротехнике (пример в начале статьи, рис. 1) можно реализовать за 3 действия:

- создать новый экземпляр модуля, загрузив в него изображение цепи и общую формулировку задания;

- создать 5 случайных полей (Y1, Y2, Y3, Y4, Y5) типа «Список» с возможными значениями {G1, G2, G3, G4, G5, G6, pC1, pC2, pC3, pC4, pC5, pC6};
- создать одно задание для всех студентов.

После этих действий любой студент, посетивший страничку модуля, увидит сформированное для него фасетное задание. При этом задание каждого студента будет, скорее всего, уникальным из-за малой вероятности генерации двух одинаковых заданий. Но если требуется точно перенести задания из методички, необходимо создать поля без случайной генерации и вручную ввести все 10 вариантов заданий с различными значениями полей.

В дисциплине «Основы программирования» (2 семестр изучения дисциплины) структура заданий схожа с семестровыми заданиями по электротехнике. Студенты должны разработать приложение – хранилище данных, но у каждого задания есть особенности.

- логика добавления данных (например, добавляется пустая запись, а затем заполняется или запись добавляется только после заполнения, всего 4 варианта);
- способ редактирования свойств-списков (группа радиокнопок, список или выпадающий список);

- способ хранения перечня в памяти (строкой, порядковым номером значения или перечислением);
- вид записей на экране (данные представлены списком или таблицей);
- тип контейнера для хранения данных (QList, QMap, QLinkedList, QHash);
- тип итератора для обхода контейнера (STL, Java).

4. Результаты внедрения

Разработанный модуль Roasassignment был внедрен в эксплуатацию в ВГТУ в феврале 2012 г., и с тех пор используется преподавателями кафедры программного обеспечения автоматизированных систем. Система продолжает развиваться, в нее регулярно добавляются новые возможности и улучшаются старые функции. Этому способствует модульная организация системы, позволяющая легко добавлять новые страницы, виды ответов, стратегии выдачи заданий и т.д.

По результатам внедрения модуля в дисциплину был произведен опрос студентов, чтобы узнать их мнение о системе. В опросе участвовали 53 студента, основными пунктами анкеты была общая оценка и оценка удобства выбора задания из дома (от -5 до 5 баллов). По результатам анализа анкет можно

сказать, что наибольшее недовольство студентов вызвал не сам модуль, а варианты заданий, которые им попадались случайным образом.

Большинство студентов оценили систему положительно (средняя общая оценка 3,2; средняя оценка удобства 4,4). Самая низкая оценка системы – -3, поставлена из-за ошибки модуля у данного студента. Замечания об ошибках модуля, высказанные в ходе опроса, к тому моменту уже были исправлены. Отдельная оценка была от старост – старосты всех групп отметили удобство раздачи заданий, отсутствие преимуществ в выборе у тех, кто получает задание первым.

Ознакомиться с документацией и скачать модуль можно по ссылке (<http://code.google.com/r/oggfire-moodle-plugins/>).

В ближайших планах (весна 2013 г.) – реализация автоматического тестирования программ студентов на выделенном защищенном сервере. Реализация данной функции позволит преподавателю дисциплин, связанных с программированием, тратить больше времени на объяснение ошибок студентов, чем на проверку программ, а студентам – предварительно проверять работоспособность своих программ на ограниченном количестве тестовых примеров.

Литература

1. Аванесов В.С. Применение тестовых форм в e-Learning [Электронный ресурс]. – Режим доступа: <http://testolog.narod.ru/Theory30.html>
2. Ящепкова Л.Я. Конструирование тестовых заданий и обработка результатов тестирования [Электронный ресурс]. – Режим доступа: <http://kpmi.wl.dvgu.ru/library/aschepkova/tests.php>
3. Шихова О.Ф., Шаламова А.В., Шкляева А.А. Системы заданий в тестовой форме // Вестник Ижевского государственного технического университета. – 2011. – № 1. – С. 153–155.
4. Анисимов А.М. Работа в системе дистанционного обучения Moodle: учеб. пособие. – Харьков: ХНАГХ, 2009. – 292 с.
5. Хахмович А.И. Теория и практика эффективного применения систем дистанционного обучения // Управление человеческим потенциалом. – 2012. – № 1. – С. 33–58.
6. Выбор системы дистанционного обучения / И.Б. Готская и др. [Электронный ресурс]. – Режим доступа: <http://ra-kurs.spb.ru/2/0/2/1/?id=13>
7. Фандей В.А. Использование модульной объектно-ориентированной динамической учебной среды moodle в преподавании межкультурной коммуникации // Вестник Московского университета. Серия 19: Лингвистика и межкультурная коммуникация. – 2011. – № 4. – С. 125–130.
8. Алдияров К.Т. Обучение техническим дисциплинам в колледже с использованием электронных ресурсов программной среды Moodle // Вестник Российского университета дружбы народов. Серия: Информатизация образования. – 2011. – № 4. – С. 89–96.
9. Sakai Assignments [Электронный ресурс]. – Режим доступа: <https://sakai.rutgers.edu/helpdocs/assignments.html>
10. LMS MOODLE – RANDOM ASSIGNMENT [Электронный ресурс]. – Режим доступа: <http://virtuni.eas.sk/rocnik/2007/pdf/fid000452.pdf>