

матрицей. Затем, к полученным данным, по координатам применим сначала Φ , получив тем самым выборку на основе копулы, а затем F_i^{-1} — обратную функцию для данной координаты.

Список использованных источников:

1. Благовещенский Ю.Н. Основные элементы теории копул. Прикладная эконометрика, 2012.
2. Фантаццини Д. Моделирование многомерных распределений с использованием копула-функций. Прикладная эконометрика 2011.

ПРОГРАММНАЯ СИСТЕМА СБОРА НОВОСТЕЙ ДЛЯ ГОРОДСКИХ ИНФОРМАЦИОННЫХ ПОРТАЛОВ

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Королёва П.Б.

Жвакина А.В. – к.т. тех. наук, доцент

Информационный портал представляет собой универсальный сайт, основной контент которого – информационные тематические статьи. Существует множество классификаций таких сайтов. Блоги, словари и справочники, объявления и вакансии, социальные сети – это лишь некоторые из них. Данная работа связана с ещё одним типом – с информационными сайтами городов и районов.

Можно с уверенностью сказать, что почти каждый город имеет собственный сайт. Обычно он содержит не только историю города, основную информацию о существующих организациях и различные справочные данные, но и новости. Новости группируются по их содержанию, и почти всегда на сайте выделяются такие разделы, как «В Беларуси» или «В мире».

Проработав пару месяцев с контентом одного городского информационно-новостного сайта, я узнала, что содержание вышеупомянутых разделов формируется с помощью более крупных новостных сайтов-источников. Это происходит вручную путём изучения контента стороннего сайта, отбора новостей и стандартной их публикации с ссылками на оригинальные источники.

Данный процесс требует достаточно много усилий от контент-менеджера. Это приводит к необходимости решения задачи оптимизации рабочего времени и упрощения агрегации информации. Ручной «копи-паст» новостей в настоящее время, когда появляется всё больше и больше различных агрегаторов и грабберов является проявлением нерационального подхода. В таких случаях целесообразно использовать автоматизированную настраиваемую систему сбора новостей.

Разработанная мною программа представляет собой именно такую систему. В перспективе её можно интегрировать на любой сайт для агрегации любых видов данных, но в данный момент акцент ставится именно на сбор новостей. Принцип работы указан на рисунке 1:

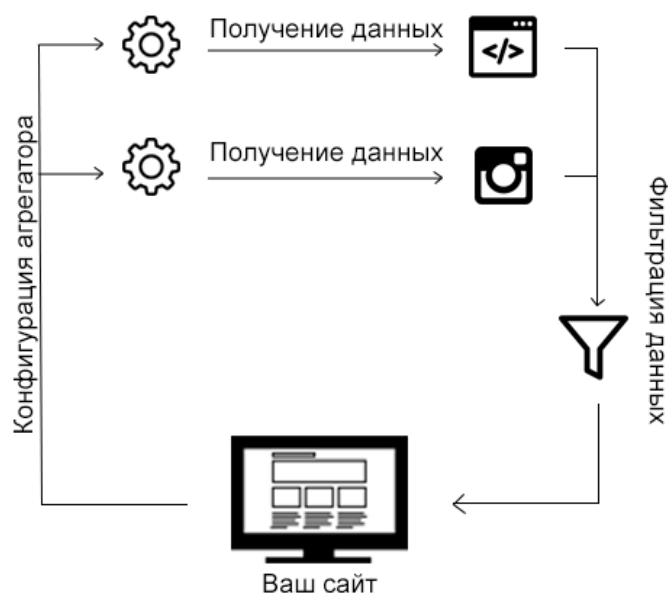


Рис. 1– Схема работы системы сбора новостей

Данный программный продукт обеспечивает возможность выбора имеющихся источников информации или создания и настройки новых.

При агрегировании исключаются повторяющиеся новости, полученные из разных источников, в зависимости от их приоритета.

Фильтр новостей по категориям, разделам, темам или тегам позволяет ограничить сбор данных по установленным критериям, подгрузив только необходимую информацию.

Последним шагом настройки данной системы является установление связи между структурой полученных данных и структурой новостей на сайте – так называемый маппинг.

Полученная система – довольна гибкая и легко дополняемая. В целом можно выделить следующие преимущества:

- значительная экономия времени в процессе сбора новостей;
- уменьшение загруженности команды контент-менеджеров путём автоматизации части их рабочего процесса или исключение потребности в наёме отдельных работников, чьей обязанностью был бы сбор данных из внешних новостных источников;
- расширяемость, возможность доработки функционала, добавления сбора информации из другого типа источников или внедрение анализа получаемого контента.

Таким образом, разработанная система позволяет значительно облегчить работу с контентом информационно-новостных порталов городов. Она может использоваться для сбора информации другого рода с целью публикации полученных данных на любом портале. Также актуально внедрение данной программы в виде отдельного модуля для систем управления содержимым сайта.

Список использованных источников:

1. Информационный онлайн-портал: цели и особенности – [Электронный ресурс] – <http://jnetwork.kz/blog/informacionnyy-onlayn-portal-celi-i-osobennosti>
2. Новостные агрегаторы как инструмент оптимизации – [Электронный ресурс] – <http://xn--h1aieep.xn--p1ai/novostnye-agregatory-v-pomoshh-optimizatoru>

АРХИТЕКТУРА СОБЫТИЙНОГО ПРЕДСТАВЛЕНИЯ ДАННЫХ

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Кравцов Д.В.

Волорова Н.А. – к.т.н., доцент

В настоящее время большинство приложений работают с данными, и типичный подход заключается в том, чтобы приложение хранило текущее состояние данных, обновляя его тогда, когда пользователи взаимодействуют с ним. Например, в традиционной модели создания, чтения, обновления и удаления (CRUD – Create, Read, Update, Delete) типичным процессом обработки данных является чтение данных из хранилища, внесение некоторых изменений и обновление текущего состояния данных новыми значениями – часто используя транзакции, которые блокируют данные. Такой подход имеет ряд ограничений:

- системы CRUD выполняют операции обновления непосредственно в хранилище данных, что может замедлить производительность и оперативность, а также ограничить масштабируемость из-за требуемых затрат на обработку;
- при совместной работе со многими параллельными пользователями конфликты обновления данных более вероятны, поскольку операции обновления выполняются на одном элементе данных;
- если нет дополнительного механизма аудита, который записывает детали каждой операции в отдельный журнал, история теряется.

Архитектура Event Sourcing определяет подход к обработке операций над данными, которые являются результатом последовательности событий, каждое из которых записывается в хранилище. Приложение отправляет ряд событий, которые однозначно описывают каждое действие, которое произошло в системе, в хранилище событий, где они сохраняются [1, 2].

Типичное использование событий заключается в том, чтобы генерировать представления сущностей одновременно с изменением их в системе, а также для интеграции с внешними системами.

Архитектура Event Sourcing предоставляет следующие преимущества:

- события неизменяемы и могут только добавляться в хранилище, что позволяет значительно повысить производительность и масштабируемость приложений, особенно для уровня представления или пользовательского интерфейса;
- события – это простые объекты вместе с любыми связанными данными, необходимыми для описания действия, представленного событием. События не обновляются непосредственно в хранилище данных, а просто записываются для обработки в соответствующее время, что может упростить внедрение и управление;
- события обычно имеют смысл для эксперта домена, тогда как несоответствие объектно-реляционного представления может затруднить понимание сложных таблиц базы данных;
- хранилище событий обеспечивает журнал, который может использоваться для мониторинга действий, выполненных в хранилище данных, восстановления текущего состояния в виде