

УДК 504.064:004

РАЗРАБОТКА ПРОГРАММНОГО КОМПЛЕКСА ЭКОЛОГИЧЕСКОГО МОНИТОРИНГА С ПРИМЕНЕНИЕМ КОНЦЕПЦИИ ИНТЕРНЕТА ВЕЩЕЙ

Великий В.А., студент гр. ПИ-181, IV курс

Федоров С.О. студент гр. ПИ-181, IV курс

Научный руководитель: Гиниятуллина О.Л., к.т.н., доцент

Кузбасский государственный технический университет

имени Т.Ф. Горбачева

г. Кемерово

Научный консультант: Замараев Р.Ю., к.т.н., с.н.с

Федеральный исследовательский центр информационных и вычислительных технологий

Актуальность

За последние несколько десятилетий проблемы экологии не перестают занимать одну из лидирующих позиций в списке глобальных проблем человечества. Постоянное наращивание мировых производственных мощностей не позволяет оперативно устраниить пагубное воздействие предприятий на природу.

Ещё в 1972 году на Стокгольмской конференции ООН по окружающей среде были сформированы и одобрены принципы экологического мониторинга, направленные на построение системы регулярных длительных наблюдений за состоянием окружающей среды. Целью такого наблюдения является предупреждение негативных воздействий человека на природу.[1]

Экологический мониторинг позволяет как оценить фактическое состояние природной среды, так и выявить тенденции изменения биосфера и прогнозировать её будущее под влиянием антропогенных факторов.

Для охраны природы разрабатывается всё больше новых технологий, в том числе и в сфере ИТ. Так, в 1999 году была сформирована концепция “интернета вещей” (англ. “Internet of Things”, IoT), предполагающая наличие сети передачи данных между физическими объектами (“вещами”) или внешней средой. Применение данной концепции возможно практически в любой сфере человеческой деятельности. Это может быть сеть интеллектуальных устройств и приборов, регулирующих домашнее освещение, отопление и системы безопасности.

Развивается IoT и в сфере экологического мониторинга. Согласно разработанной в 2017 году на женевской “Неделе интернета вещей” международной декларации "Интернет вещей для устойчивого развития", одним из ключевых направлений деятельности в области IoT является “применение и использование IoT для сохранения биоразнообразия и экологического мониторинга с целью защиты живой природы и ее разнообразия на суше, в воздухе и под водой”.[2]

Для Кемеровской области мониторинг имеет особое значение. Из-за высокой техногенной нагрузки наблюдается высокий уровень загрязнения сточных вод (34–35 % на 2018 г.). [3] В большей степени поверхностные водные объекты загрязняются сбросами предприятий угледобывающей отрасли (40 %) и химическими веществами и элементами (18,9 %). Следует отметить, что и без того серьезная экологическая ситуация усугубляется сбросом промпредприятиями неочищенных сточных вод, содержащих тяжелые металлы, фенолы, красители, СПАВы и другие опасные загрязнения в централизованные системы водоотведения, увеличивая на них гидравлическую нагрузку и приводя к выходу их из эксплуатации. Таким образом, разработка системы экологического мониторинга для рек Кузбасса на данный момент является актуальным направлением. [4]

Таким образом, на базе Федерального исследовательского центра информационных и вычислительных технологий в настоящее время разрабатывается прототип информационной системы экологического мониторинга на основе IoT.

В ходе планирования проекта была определена структура разрабатываемой системы, требования к элементам системы, а также средства и технологии для разработки.

Структура системы и данные

Для функционирования информационной системы экологического мониторинга требуется интеграция **подсистемы сбора, подсистемы выгрузки и подсистемы анализа** данных.

Подсистема сбора данных представляет собой сеть умных датчиков, которые распределяются по интересующему водному объекту. Датчики в реальном времени измеряют концентрацию химических загрязнений воды. На основе данных о распространении основных типов загрязняющих веществ, а также о фоновом состоянии воды, набор датчиков формирует цифрового двойника реки. Таким образом, подсистема сбора данных реализует описанную выше концепцию IoT.

Подсистема выгрузки данных предназначена для сбора измерений, проведённых датчиками, и объединение их в целевую базу данных. Такой процесс, называемый ETL (“Extract” - извлечение, “Transform” - преобразование, “Load” - загрузка), позволяет привести все данные датчиков к единой системе значений и детализации, попутно обеспечив их качество и надежность.

Подсистема анализа данных представляет собой набор средств для исследования данных, загруженных подсистемой ETL в базу данных. Здесь оценивается текущее состояние искомого географического объекта и проводится оценка его будущего состояния с помощью искусственного интеллекта или привлечения экспертов.

Следует отметить, что установка датчиков на местности является дорогостоящим и длительным процессом. Поэтому, чтобы избежать неопределенности при возникновении ошибок и отказов на переходе между подсистемами сбора и выгрузки и отработать всю схему передачи данных от измерения до визуализации, подсистема IoT заменяется на конфигурируемые программные **эмуляторы** – источники данных, повторяющие информационные характеристики **датчиков**. Соответственно, при реализации ETL и подсистемы анализа данных становится возможным отработать внутренние механизмы доступа к данным и автоматической адаптации к изменениям в IoT до её непосредственной реализации.

Эмулятор датчиков включает в себе несколько сущностей. Верхняя сущность - «станция» - является агрегатором буферов и имеет индивидуальные характеристики: сетевой адрес, имя, географическая привязка, список доступных «датчиков». «Датчик» представляет собой индивидуальный поток числовых данных в виде обновляемого буфера с установленными характеристиками: имя, измеряемый параметр, частота дискретизации, статистические характеристики. Исходя из соображений, что химические показатели водоёма являются случайными величинами, распределёнными по определённому закону, то есть возможность, зная о параметрах этих случайных величин, применить методы статистического моделирования для генерации необходимой выборки.[5]

Основной принцип работы эмулятора следующий. В определённый момент времени производится обход каналов эмулятора, представляющих собой «станции». Каждый из таких каналов генерирует значения случайных величин его параметров. Полученные значения выгружаются в буфер канала вместе с датой генерации и характеристиками канала в виде JSON-строки. Буфера каналов затем просматриваются подсистемой выгрузки данных.

Итак, эмулятор обладает следующими настройками:

- 1) состояние эмулятора (работает или выключен);

- 2) размер буфера эмулятора, то есть максимальное количество записей измерений всех каналов в общем файле буфера;
- 3) режим работы эмулятора (синхронный - все каналы имеют одинаковую частоту дискретизации и генерируют значения параметров в одно и то же время, асинхронный - каждый из каналов имеет собственную частоту дискретизации);
- 4) закон и параметры распределения частоты дискретизации каналов для синхронного режима работы;
- 5) набор каналов.

В свою очередь, каждый из каналов эмулятора имеет следующие настройки:

- 1) состояние канала (работает или выключен);
- 2) сетевой адрес канала, являющийся для него уникальным идентификатором;
- 3) размер буфера канала, то есть максимальное количество записей измерений конкретного канала в файле буфера этого канала;
- 4) название канала для обозначения его географического положения (например, «Томь-1»);
- 5) географические координаты, обозначающие координаты станции, в виде набора чисел – широты и долготы;
- 6) закон и параметры распределения частоты дискретизации канала для асинхронного режима работы эмулятора;
- 7) набор генерируемых каналом параметров, являющихся случайными величинами.

Каждый параметр канала имеет такие настройки, как уникальное в пределах канала название и параметры распределения случайной величины, значение которой будет генерироваться эмулятором.

По умолчанию параметрам каналов установлено нормальное распределение. Для генерации значений используется метод обратной функции.[5]

В реальных данных не исключены выбросы – значения, выбивающиеся из сложившегося распределения некоторой величины. Они могут быть вызваны различными факторами, в том числе неисправностью оборудования. В случае экологического мониторинга, выбросы в данных могут быть вызваны выбросами в окружающую среду отходов предприятиями. Следовательно, возможность порождения выбросов также важна для эмулятора.

Для нормального распределения существует множество методов идентификации выброса. Один из них – метод стандартных, который сводится к правилу 3-сигма. Необходимо лишь сгенерировать число, выходящее за пределы

трёх стандартных отклонений от среднего значения.

Для программной реализации эмулятора используется язык программирования *Python* и микрофреймворк *Flask*, предлагающий большую свободу действий для разработчика и хорошо подходящий для небольших проектов, работающих в основном со статическим контентом. Таким образом, эмулятор представляет собой веб-приложение, основная страница которого – страница конфигурации эмулятора. Пример такой страницы продемонстрирован на рисунках 1 и 2.

В конечном итоге функциональность пользовательского интерфейса следующая:

- 1) возможность включения и выключения эмулятора;
- 2) возможность изменения всех описанных выше настроек эмулятора, его каналов и параметров каналов;
- 3) возможность настроить вероятность появления выбросов в данных, распределённых по нормальному закону, а также возможность вручную инициировать выброс в данных (рисунок 2);
- 4) возможность сохранения описанных выше настроек эмулятора в файл конфигурации для последующей загрузки и возобновлении работы после повторного запуска сервера.

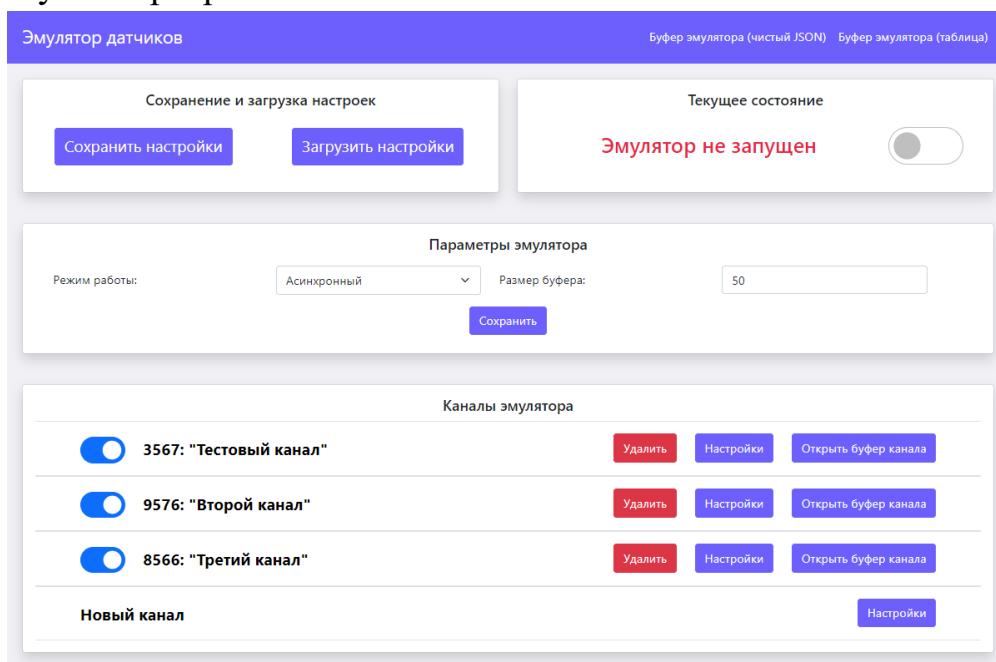


Рисунок 1. Внешний вид страницы конфигурации эмулятора

Рисунок 2. Настройки канала и его параметров

Подсистема выгрузки данных реализуется как обособленное от эмулятора приложение, реализованное на языке *Python* с использованием конвейера данных (пакета) *Bonobo*. Его преимущество состоит в облегчённой инфраструктуре, доступной для всех, кто имеет базовый уровень владения *Python*.

Конвейер данных состоит из *процедуры извлечения данных*, которая обращается по сетевым адресам к каналам эмулятора и извлекает данные из буферов, *процедуры преобразования данных*, проверяющей корректность полученных данных и подводящая их к заданному стандарту записей, и *процедуры загрузки данных*, которая загружает набор записей в базу данных.

В качестве базы данных используется *PostgreSQL*, являющейся бесплатной и неограниченной, взаимодействие с которой удобно реализовано с помощью *Python*-библиотеки *Psycopg2*.

Структуру данных в базе данных составляют три таблицы:

- 1) таблица *станций*, содержащая информацию о станциях системы, их сетевых адресах, названии и географических координатах;
- 2) таблица *датчиков*, содержащая информацию о датчиках станций, их названиях;
- 3) таблица *проб*, содержащая информацию о пробах воды, дате их забора и

концентраций химических веществ, полученную с помощью датчиков станций.

Подсистема анализа данных, аналогично эмулятору, представляет собой веб-приложение, реализованное с помощью пакета *Flask*. Основная задача данного приложения – наглядное, информативное представление пользователю состояний станций и датчиков, а также проводимых датчиками измерений на определённый момент времени.

Требования к функциональности подсистемы анализа данных следующие:

- 1) наличие визуального интерфейса для проверки состояния станций и характеристик датчиков (станции располагаются на интерактивной карте);
- 2) визуализация измерений датчиков (представление данных в виде графиков);
- 3) возможность отслеживать аварийные состояния датчиков, в том числе на графиках и на карте;
- 4) возможность регистрации в базе данных новых станций путём передачи системе ETL сетевых адресов станций.

Все данные для представления берутся из базы данных, описанной в подсистеме выгрузки данных.

Заключение

Таким образом, в настоящее время описаны подсистемы программного комплекса экологического мониторинга с учётом замены сети умных датчиков на программный эмулятор. Также учтено, что все три подсистемы реализуются в качестве независимых сервисов, взаимодействующих друг с другом по сети.

Новизна предлагаемого решения заключается в том, что еще до стадии физического размещения измерительного оборудования имеется возможность подготовить к горячему старту всю информационно-аналитическую инфраструктуру системы мониторинга, промоделировать экстраординарные ситуации и отстроить реакцию системы на них.

Стоит подчеркнуть, что, несмотря на направленность данной работы на экологический мониторинг водных объектов Кузбасса, описанный программный комплекс может применяться также и для контроля на любой местности за состоянием воздуха, земли и т.д.

Внедрение системы экологического мониторинга позволит оперативно выявлять источники загрязнения экологических объектов и предотвратить их дальнейшее распространение, что очень важно для крупнейших речных бассейнов нашей страны, испытывающих каждый год колоссальную техногенную нагрузку.

Список литературы

- 1) Хаустов, А. П. Экологический мониторинг : учебник для академического бакалавриата / А. П. Хаустов, М. М. Редина. — 2-е изд., испр. и доп. — М. : Издательство Юрайт, 2019. — 543 с. — (Серия : Бакалавр. Академический курс).
- 2) Международная декларация "Интернет вещей для устойчивого развития" [Электронный ресурс]: Принята путем аккламации участниками Недели интернета вещей 2017 года 9 июня 2017 года – Режим доступа: https://iotforum.org/wp-content/uploads/2019/10/IoT4SDG-Declaration_Russian.pdf (дата обращения: 10.03.2022)
- 3) Брель О.А., Задорожная Г.В., Сасаев Н.И., Егорова А.И. Стратегирование водных ресурсов Кузбасса [Электронный ресурс] // Экономика промышленности. 2020. №13. С. 357-365. Режим доступа: <https://www.elibrary.ru/item.asp?id=43985376&> (дата обращения: 10.03.2022)
- 4) Потапова Ю. Ускат обретет двойника. В Кузбассе создают уникальную модель управления водными ресурсами [Электронный ресурс] // Российская газета. 2012. Режим доступа: <https://rg.ru/2020/06/05/reg-sibfo/v-kuzbasse-sozdaiut-unikalnuiu-model-upravleniiia-vodnymi-resursami.html> (дата обращения: 10.03.2022).
- 5) Тынкевич, М. А. Исследование операций и имитационное моделирование : учеб. пособие / М. А. Тынкевич, А. Г. Пимонов, С. А. Веревкин ; КузГТУ им. Т. Ф. Горбачева. – Кемерово, 2015. – 248 с.