

УДК 004.415.2
DOI: 10.15827/0236-235X.131.516-522

Дата подачи статьи: 16.03.20
2020. Т. 33. № 3. С. 516–522

Модуль автоматизированного управления системы мониторинга производственных процессов

Г.М. Соломаха¹, д.ф.-м.н., профессор, gsolomakha@yandex.ru
С.В. Хижняк¹, аспирант, stanislav.khizhnyak@gmail.com

¹ Тверской государственный университет, кафедра математической статистики и системного анализа, г. Тверь, 170100, Россия

В работе представлены архитектура, средства и технологии, используемые при реализации модуля автоматизированного управления системы мониторинга производственных процессов, предоставляющего возможность автоматической и ручной корректировки режима работы оборудования. Модуль обеспечивает удаленное ручное и автоматизированное управление в условиях географической распределенности производственной инфраструктуры. Компоненты модуля, протокол и порядок их взаимодействия, способ и формат передачи данных ориентированы на использование как на географически распределенных, так и на сосредоточенных производствах.

В статье обоснована целесообразность разработки модуля. Изложены сформированные в ходе исследования требования к нему, выполнение которых необходимо для решения поставленной перед модулем задачи, в частности, возможности дистанционного автоматизированного и ручного управления, управления с максимального количества платформ, интеграции со сторонними системами. Обоснованы все сформированные требования, приведены способы их достижения. Представлены основные подсистемы и компоненты, описаны их назначение, функции и принципы работы. Приведены формат и способ передачи данных между компонентами модуля, а также способ их хранения.

В качестве формата представления данных авторы используют JSON, данные передаются с помощью WebSocket. Обработка данных осуществляется посредством интегрированных обработчиков на языке JavaScript, интерпретируемых с использованием Rhino JavaScript Engine. Язык реализации модуля – Kotlin. В качестве модели обмена данными между компонентами модуля используется событийная модель. Приведена схема архитектуры модуля, названы ее компоненты. По итогам проведенных исследований сделаны выводы.

Спроектированная архитектура обеспечивает высокую степень мобильности и предоставляет возможность управления в условиях географической распределенности производства. При использовании в составе системы мониторинга производственных процессов обеспечивается высокая степень гибкости и масштабируемости.

Ключевые слова: информационная система, автоматизированное управление, архитектура информационной системы, системный анализ, модуль управления.

При наличии достоверной и своевременно доставленной системой мониторинга информации о текущем состоянии производства в случае необходимости становится возможным своевременное реагирование на его изменения, например, при возникновении нештатной ситуации на производстве, а также поломке или некорректной работе каких-либо устройств. Так как информация о проблеме такого типа и, возможно, ее характере уже получена, при наличии на месте квалифицированных специалистов могут быть приняты необходимые меры. Однако в силу различных причин устранение проблемы в ручном режиме может оказаться затруднительным или невозможным. Одной из мер снижения количества подобных ситуаций может быть полная или частичная автоматизация управления производственными устройствами.

Автоматизированное управление производственным оборудованием можно назвать следующим шагом в информатизации производства после организации мониторинга с помощью информационной системы. Обеспечение управления производством в автоматическом режиме имеет ряд преимуществ [1].

Возможность автоматического включения/отключения оборудования в случае опасных ситуаций. При возникновении нештатных ситуаций на производстве ущерб можно свести к минимуму, своевременно отключив оборудование или скорректировав режим его работы таким образом, чтобы соответствующие показатели пришли в норму. Однако в случае прямого ручного вмешательства человека время на выполнение нужных операций может быть существенно большим, чем при использовании автоматических средств управления, что мо-

жет привести к дополнительному ущербу. Таким образом, возможность автоматического включения, отключения или изменения режима работы оборудования в случае возникновения нештатных ситуаций является весомым преимуществом автоматизированного управления над ручным.

Возможность автоматической корректировки режима работы оборудования. Автоматическая корректировка позволит снизить влияние человеческого фактора на производственный процесс, тем самым повышая предсказуемость качества продукции.

Снижение затрат человеческого труда. Очевидно, что даже частичная автоматизация процесса управления производственным оборудованием снизит количество времени, затрачиваемого специалистами предприятия на управление, что в конечном итоге снизит расходы в целом.

Высокая скорость реагирования на изменение ситуации. Повышение скорости реагирования на изменение состояния оборудования и положение на производстве потенциально снизит риск возникновения опасных ситуаций, брака и других невыгодных обстоятельств.

Потенциальное снижение времени производственного цикла. При снижении времени производственного цикла повышается количество производимой продукции без снижения уровня качества, что ведет к целому ряду очевидных положительных последствий.

Современный рынок АСУ технологическими процессами (АСУТП) предлагает довольно широкий ряд решений по автоматизации управления производственными процессами, однако взаимодействие с ними в большинстве случаев организовано способами, не предоставляющими достаточной степени мобильности [2]. В силу топологического и сложностного многообразия производственных инфраструктур, а также широкого набора возможных комбинаций информационных систем, работающих на предприятии, внедрение АСУТП является сложным и дорогостоящим процессом в условиях недостаточной гибкости имеющихся решений и их низкой приспособленности к совместному использованию с различными системами сбора данных и мониторинга. Кроме того, существующие решения имеют низкую масштабируемость, что усложняет их расширение в соответствии с путями расширения производства, а также внедрение в условиях географической распределенности производства [3].

Таким образом, можно сделать вывод о целесообразности разработки отдельного модуля автоматизированного управления для использования в связке с *системой мониторинга производственных процессов* (СМПП), архитектура, компоненты и основные внутренние и внешние функции которой изложены в [4]. Практическую значимость внедрения подобного решения можно оценить как высокую.

Как основной источник данных для принятия логических решений по управлению предполагается использовать СМПП, например, работающую во франчайзинговой производственной системе [5], однако целесообразно обеспечить возможность использования модуля и отдельно от данной системы, например, в качестве дублирующего средства на случай экстренных ситуаций в связке с какой-либо другой информационной системой либо средства упрощения ручного управления. Кроме того, отдельные подразделения производства могут быть распределены географически и непосредственное ручное управление может быть вообще невозможным, поэтому необходима возможность удаленного ручного управления. Все это нужно учитывать при проектировании и разработке модуля.

Требования к модулю

Производственная инфраструктура может быть довольно сложной и к тому же распределенной географически, поэтому необходимо обеспечить возможность удаленного управления производственным оборудованием.

Для обеспечения возможности внедрения модуля в качестве дублирующего средства или средства упрощения ручного управления необходима реализация возможности ручного удаленного управления подконтрольным оборудованием с максимального количества платформ.

Наиболее целесообразным подходом к обмену данными с модулем и между его компонентами является событийный [6]. Совокупность событий в определенный промежуток времени может отражать некоторую ситуацию, требующую автоматической корректировки режима работы оборудования или его включения/отключения. Соответственно, необходимо предоставить пользователю системы возможность создания обработчиков событий и их совокупностей для выявления таких ситуаций и определения соответствующих автоматических действий. Во избежание необходимости вмешательства в производственный процесс

добавление и удаление обработчиков не должны нарушать работу модуля или требовать его перезапуска.

При возникновении определенных ситуаций может потребоваться корректировка режима работы или включение/отключение нескольких устройств одновременно. Модуль должен предоставлять такую возможность.

При корректировке режима работы одного устройства может быть обязательной соответствующая корректировка режима работы другого. Следовательно, необходимо предоставить возможность обработки событий, являющихся результатом работы других обработчиков.

В силу необходимости ручного управления должна быть возможность предотвращения случайных или ошибочных управляющих сигналов. Соответственно, модуль должен обеспечивать программируемую фильтрацию управляющих событий.

Таким образом, разрабатываемый модуль должен обеспечивать следующие возможности:

- 1) дистанционное управление в силу географической распределенности;
- 2) ручное управление;
- 3) управление с максимального количества платформ;
- 4) генерация управляющих событий по совокупности событий;
- 5) добавление и удаление управляющих логических компонентов без нарушения работы модуля;
- 6) генерация управляющих событий по другим управляющим событиям;
- 7) управление одним событием с нескольких устройств;
- 8) интеграция со сторонними системами.

На рисунке приведена схема архитектуры, удовлетворяющей всем предъявляемым требованиям.

Компоненты и подсистемы

Модуль включает следующие блоки: Outer, Automated Control module, Storage, DB. Они могут иметь произвольное географическое местоположение, а их взаимная видимость обеспечивается соответствующей конфигурацией сети. По аналогии с СМПП возможно одновременное использование нескольких экземпляров одного или нескольких блоков.

Форматом для передачи данных между компонентами модуля по аналогии с СМПП вы-

бран JSON [7], а в качестве транспорта – WebSocket [8, 9]. Для реализации модуля использован язык программирования Kotlin.

Event Source – внешний источник событий, по которым будут сгенерированы управляющие события. В качестве источника могут выступать как система мониторинга, так и программные терминалы ручного управления, реализованные на любых платформах, предоставляющих возможность установки WebSocket-соединения.

Таким образом, подобный выбор формата, транспорта и модели обмена данными обеспечивает выполнение требований 1, 2 и 3.

Компоненты блока Outer включают в себя программы для преобразования данных из формата системы в управляющие устройствами сигналы, драйверы устройств и сами управляемые устройства.

Блок Storage включает в себя драйверы и необходимые для преобразования данных в формат БД программные компоненты. В стандартной реализации предполагается использование MongoDB в качестве системы управления как

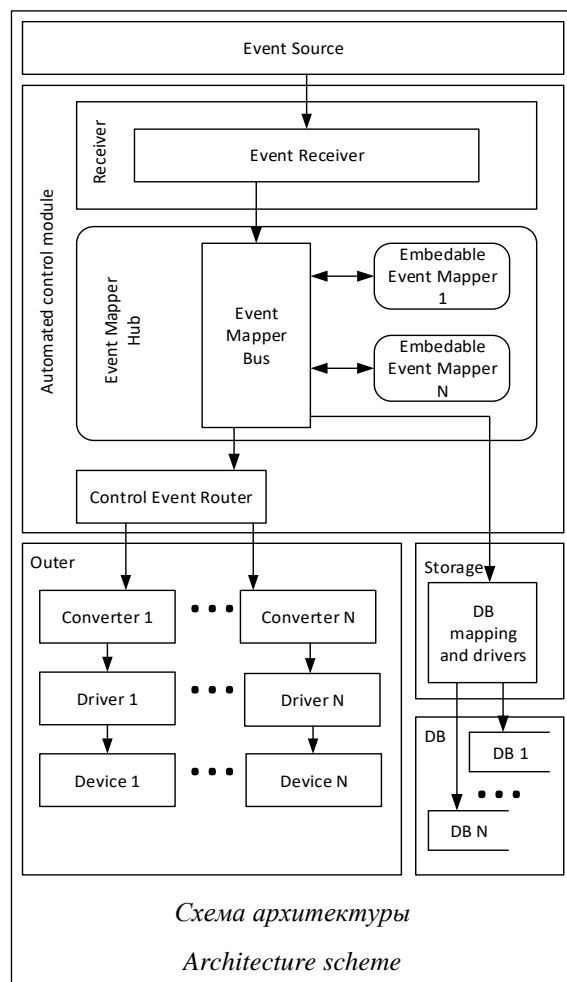


Схема архитектуры

Architecture scheme

наиболее подходящей для подобного случая [10, 11], однако жестко ограничивать пользователя модуля в выборе СУБД нецелесообразно в силу возможности наличия у него каких-либо технических ограничений в данном вопросе. Связь с блоком, как и со всеми остальными независимыми компонентами модуля, осуществляется с помощью WebSocket, а данные передаются в формате JSON, что позволяет при необходимости использовать иную реализацию блока, предполагающую использование иных систем управления БД.

Блок DB представляет внешние БД. В стандартной реализации предполагается использование MongoDB.

Блок Automated Control module представляет собой ядро модуля и включает Event Receiver, Event Mapper Hub, Control Event Router.

Event Receiver является точкой входа событий. Основные функции данного компонента – получение событий извне и их передача в Event Mapper Hub для последующей обработки. Каждое событие представляется в виде JSON-объекта, содержащего имя события и вложенный объект с дополнительной информацией, характеризующей событие. Например, событие, представляющее перегрев устройства с идентификатором «Device_1», может быть отражено в следующем виде:

```
{
  "name": "Overheat",
  "data": {
    "value": "Device_1",
    "date": 1553109651
  }
}
```

Event Mapper Hub состоит из Event Mapper Bus – шины, объединяющей логические обработчики событий – экземпляры Embeddable Event Mapper. Основной функцией самого Event Mapper Hub является автоматическая передача событий в каждый из обработчиков с последующим получением от них управляющих событий при их наличии и дальнейшей передачей их в Control Event Router и Storage. Каждый Embeddable Event Mapper представляет собой записанную в отдельном файле функцию на языке JavaScript, принимающую на вход одно или несколько событий заранее определенных типов и автоматически возвращающую в шину управляющее событие в случае соответствия входа необходимым условиям. Файл интерпретируется, после чего выполняется функция, в которую передаются обрабатываемые ею события. Таким образом обеспечивается выполнение требования 4.

В силу того, что обработчики представлены в виде отдельных файлов с JavaScript-функциями, становится возможной их подгрузка на лету без нарушения работы системы, что обуславливает выполнение требования 5. Выбор JavaScript обусловлен его широким применением в качестве встраиваемого языка и удобством для создания приложений, управляемых событиями [12, 13]. Сгенерированные обработчиками события возвращаются обратно в шину и при наличии ожидающих их обработчиков передаются в них. Это удовлетворяет требованию 6. Для обеспечения совместимости управляющие события имеют схожую с входными сигнатуру. Например, событие, содержащее команду на выключение устройств с идентификаторами «Device 2» и «Device 3», может выглядеть так:

```
{
  "name": "Shutdown",
  "type": "CONTROL"
  "data": {
    "value": ["Device_2", "Device_3"],
    "date": 1553109651
  }
}
```

Как видно из примера, сигнатуры практически идентичны, кроме поля type, необходимого для фильтрации входных событий от управляющих, а также для обеспечения возможности дальнейшей реализации функционала группирования управляющих событий. Также ясно, что в качестве значения value может быть передан массив идентификаторов устройств. Это необходимо для выполнения требования 7.

Теперь предположим, что при перегреве устройства «Device_1» необходимо выключить устройства «Device_2» и «Device_3», тогда функция, представляющая Embeddable Event Mapper для решения подобной задачи, в простейшем виде может выглядеть так:

```
function handle(event) {
  if (event.name == "Overheat" &&
    event.data.value == "Device_1") {
    return {
      "name": "Shutdown",
      "type": "CONTROL"
      "data": {
        "value": ["Device_2", "Device_3"],
        "date": 1553109651
      }
    }
  } else return null
}
```

Интерпретация JavaScript-файлов производится с помощью Rhino JavaScript Engine, являющегося Java-реализацией языка JavaScript, что в совокупности с открытостью исходного

кода делает его оптимальным выбором для использования с языком Kotlin.

Использование JSON в качестве формата данных, в котором осуществляется обмен, и WebSocket в качестве транспорта делает возможной и несложной организацию взаимодействия с любой сторонней внешней системой в качестве источника данных, что способствует выполнению требования 8.

Control Event Router отвечает за маршрутизацию управляющих событий к соответствующим устройствам. Маршрутизация осуществляется по соответствию идентификаторов и IP-адресов устройств.

Заключение

В ходе исследования обоснована целесообразность реализации модуля автоматизирован-

ного управления для СМПП, сформированы требования, предъявляемые к модулю. В соответствии со сформированными требованиями разработана архитектура модуля автоматизированного управления, выбраны средства реализации и используемые технологии. Модуль, обладающий данной архитектурой, имеет высокую степень мобильности благодаря возможности реализации управляющих терминалов на большом количестве современных платформ, что одновременно обеспечивает возможность удаленного управления, а значит, и в условиях географической распределенности производства. В силу заложенных возможностей пользователю предоставляется высокая степень гибкости и масштабируемости при использовании модуля автоматизированного управления в составе СМПП.

Литература

1. Андрюхин С.Н. Компоненты интегрированной системы управления // Междунар. студ. науч. вестн. 2019. № 6. С. 18–28.
2. Турицын Ю.А., Баранникова И.В., Пасечник И.А. Обзор современных АСУТП и АСДУ на промышленных предприятиях // Горный информационно-аналитический бюллетень. 2009. № S2. С. 355–362.
3. Меламед А.Д., Черномзав И.З., Пека Г.С. Опыт и проблемы разработки и внедрения АСУТП ПГУ на электростанциях России и Белоруссии // Докл. БГУИР. 2015. № 6. С. 94–99.
4. Соломаха Г.М., Хижняк С.В. Архитектура системы мониторинга производственных процессов в условиях географической распределенности производства // Программные продукты и системы. 2019. № 2. С. 251–257. DOI: 10.15827/0236-235X.126.251-257.
5. Соломаха А.Г., Соломаха Г.М., Язенин А.В. Нахождение параметров производственного франчайзингового договора методами теории иерархических игр // Вестн. ТвГУ. Прикладная математика. 2018. № 4. С. 64–75. DOI: 10.26456/vtprmk518.
6. Соломаха Г.М., Хижняк С.В. Событийная модель обмена данными в системах мониторинга производственных процессов // Вестн. ТвГУ. Технич. науки. 2019. № 3. С. 88–94.
7. Сергушкин В.В., Бодров О.А. Использование формата данных JSON для взаимодействий с пользователем при разработке программного обеспечения аппаратуры передачи данных // СТНО-2018: матер. Междунар. науч.-технич. форума. 2018. Т. 5. С. 217–220.
8. Хабаров С.П. Взаимодействия узлов сети по протоколу WebSocket // Информационные системы и технологии: теория и практика: сб. науч. тр. конф. 2017. С. 104–115.
9. Pimentel V., Nickerson B.G. Communicating and displaying real-time data with WebSocket. IEEE Internet Computing, 2012, no. 16, pp. 45–53. DOI: 10.1109/MIC.2012.64.
10. Федоров А.Р., Васильчук К.С., Дорофеев А.В. Создание масштабируемых средств для решения задач анализа больших объемов данных на основе системы управления базы данных MongoDB // Вестн. Поволжского гос. ун-та. Радиотехнические и инфокоммуникационные системы. 2016. № 1. С. 55–63.
11. Голиков О.И., Панкратова И.А. Исследование способов повышения эффективности обработки данных в реляционных БД на примере СУБД MySQL // Вестн. Волжского ун-та им. В.Н. Татищева. 2016. Т. 2. № 2. С. 124–130.
12. Берко В.А., Таран В.Н. Применение языка программирования JavaScript в разработке систем «Интернета вещей» // Дистанционные образовательные технологии: матер. III Всеросс. науч.-практич. конф. 2018. С. 198–201.
13. Kossakowski G., Amin N., Rompf T., Odersky M. JavaScript as an Embedded DSL. Proc. 26th ECOOP, 2012, pp. 409–434. DOI: 10.1007/978-3-642-31057-7_19.

Automated control module for a production process monitoring system

G.M. Solomakha¹, *Dr.Sc. (Physics and Mathematics), Professor, gsolomakha@yandex.ru*
S.V. Khizhnyak¹, *Postgraduate Student, stanislav.khizhnyak@gmail.com*

¹ *Tver State University, the Department of Mathematical Statistics and System Analysis, Tver, 170100, Russian Federation*

Abstract. The paper presents the architecture, tools, and technologies used in the implementation of the automated control module of the monitoring system for production processes, which provides the ability to automatically and manually adjust the operating mode of the equipment. The module provides the ability to control geographically distributed production infrastructure remotely manually or automatically. The components of the module, the protocol, and the order of their negotiation, the method, and format of data transfer are oriented to use both in geographically distributed and concentrated industries.

The paper substantiates the feasibility of developing the module. The paper describes the requirements for the module formed during the research, which are necessary to solve the problem set for the module, in particular, the possibility of remote automated and manual control, control from the maximum number of platforms, integration with third-party systems. The paper describes ways to achieve the former requirements. The authors present the main subsystems and components, describe their purpose, functions, and principles of operation, give the format and method of data transfer between the module components, as well as the method of storing them.

The authors use JSON as the data representation format. WebSocket transmits data. Integrated handlers in JavaScript perform data processing. Rhino JavaScript Engine provides the interpretation of these handlers. The module implementation language is Kotlin. The event model is used as a model for data exchange between module components. The authors provide the module architecture diagram and name its components. Based on the results of their research, they draw conclusions.

The designed architecture provides a high degree of mobility and allows management in the conditions of the geographical distribution of production. When used as part of a production process monitoring system, it provides a high degree of flexibility and scalability.

Keywords: information system, automated control, information system architecture, system analysis, control module.

References

1. Andryukhin S.N. Components of the integrated management system. *Mezhdunar. Stud. Nauch. Vestn.*, 2019, vol. 6, pp. 18–28 (in Russ.).
2. Turitsin Yu.A., Barannikova I.V., Pasechnik I.A. The overview of the modern automatic process control systems (APCS) and automated dispatch control systems (ADCS) at the industrial enterprises. *Mining Inform. and Analytic. Bull.*, 2009, vol. S2, pp. 355–362 (in Russ.).
3. Melamed A.D., Chernomzav I.Z., Peka G.S. Experience and problems in the development and implementation I&C of CCPP at power plants of Russia and Belarus. *Dokl. BGUIR*, 2015, no. 6, pp. 94–99 (in Russ.).
4. Solomakha G.M., Khizhnyak S.V. The architecture of a production processes monitoring system in terms of geographically distributed production. *Software & Systems*, 2019, no. 2, pp. 251–257 (in Russ.). DOI: 10.15827/0236-235X.126.251-257.
5. Solomakha A.G., Solomakha G.M., Yazenin A.V. Finding the parameters of a franchising agreement in the production sphere by methods of theory of hierarchical games. *Vestn. TvGU. Prikladnaya Matematika*, 2018, no. 2, pp. 64–75 (in Russ.). DOI: 10.26456/vtpmk518.
6. Solomakha G.M., Khizhnyak S.V. Event driven data exchange model in production processes monitoring system. *Vestn. TvGTU. Tekhnich. Nauki*, 2019, no. 3, pp. 88–94 (in Russ.).
7. Sergushkin V.V., Bodrov O.A. Using the JSON data format to interact with user when developing the software for data transmission equipment. *Proc. Conf. STNO-2018*, 2018, vol. 5, pp. 217–220 (in Russ.).
8. Khabarov S.P. Interaction of network nodes using the WebSocket protocol. *Proc. Conf. Information Systems and Technologies: Theory and Practice*, 2017, pp. 104–115 (in Russ.).
9. Pimentel V., Nickerson B.G. Communicating and displaying real-time data with WebSocket. *IEEE Internet Computing*, 2012, no. 16, pp. 45–53. DOI: 10.1109/MIC.2012.64.

10. Fedorov A.R., Vasilchuk K.S., Dorofeev A.V. Creation of scalable tools for the solution of the problems of the analysis of large volumes of data based on DMBS MongoDB. *Vestn. Volga Tech. Radio Eng. and Infocom. Syst.*, 2016, no. 1, pp. 55–63 (in Russ.).

11. Golikov O.I., Pankratov I.A. A study of ways to improve the efficiency of data processing in relational databases on mysql database example. *Vestn. Volzhsky Univ. after V.N. Tatishchev*, 2016, vol. 2, no. 5, pp. 124–130 (in Russ.).

12. Berko V.A., Taran V.N. JavaScript application in the development of Internet of things systems. *Proc. 3rd All-Russ. Sci. Pract. Conf. Remote Educational Technologies*, 2018, pp. 198–201 (in Russ.).

13. Kossakowski G., Amin N., Rompf T., Odersky M. JavaScript as an Embedded DSL. *Proc. 26th ECOOP*, 2012, pp. 409–434. DOI: 10.1007/978-3-642-31057-7_19.

Для цитирования

Соломаха Г.М., Хижняк С.В. Модуль автоматизированного управления системы мониторинга производственных процессов // Программные продукты и системы. 2020. Т. 33. № 3. С. 516–522. DOI: 10.15827/0236-235X.131.516-522.

For citation

Solomakha G.M., Khizhnyak S.V. Automated control module for a production process monitoring system. *Software & Systems*, 2020, vol. 33, no. 3, pp. 516–522 (in Russ.). DOI: 10.15827/0236-235X.131.516-522.