

11. Semjonov S., Glaesemann G. High-speed tensile testing of optical fibers – New understanding for reliability prediction. *Micro- and Opto-Electronic Materials and Structures: Physics, Mechanics, Design, Reliability, Packaging*. Boston: Springer, 2007, pp. 595–625.
12. Dvajt G.B. *Integral Tables and Other Mathematical Formulas*. Moscow: Nauka, 1977, 228 p. (In Russ.)
13. Abramovits M., Stigan I. *Handbook of Special Functions*. Moscow: Nauka, 1979, 830 p. (In Russ.)
14. Gradshtejn I., Ryzhik I. *Tables of Integrals*. Moscow: Fizmatgiz, 1962, 1100 p. (In Russ.)
15. Burdin V.A., Nizhgorodov A.O. Forecast of the service life of the building length of an optical cable taking into account the loads on the fiber in the process of its production. *Foton-Ekspress*, 2019, vol. 158 (6), pp. 124–125. (In Russ.)
16. Nizhgorodov A.O., Burdin V.A. Model for predicting the service life of an optical cable of a communication line put into operation. *Opticheskie tehnologii v telekommunikatsijah OTT-2019: mat. XVII mezhd. nauch.-prakt. konf*, 2019, vol. 2, pp. 216–217. (In Russ.)

Received 25.12.2020

ТЕХНОЛОГИИ КОМПЬЮТЕРНЫХ СИСТЕМ И СЕТЕЙ

УДК 004.738.5

КОНТЕНТ КОМПЬЮТЕРНЫХ СЕТЕЙ: ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО КОНТРОЛЬНОГО БЛОКА

Василенко К.А.¹, Золкин А.Л.², Абрамов Н.В.³, Курганов Д.О.³

¹ *Владивостокский государственный институт экономики и сервиса, Владивосток, РФ,*

² *Волжский государственный университет водного транспорта (Самарский филиал), Самара, РФ,*

³ *Дальневосточный федеральный университет, Владивосток, РФ*

E-mail: k2857@mail.ru, alzolkin@list.ru, nikolay.abramov1990@mail.ru, kurganov_vl@mail.ru

Рассматривается процесс разработки программного контрольного блока контента (движка) как собственной системы его управления; указываются уязвимые места, набор необходимых функций, присущих указанному движку. Каждый новый разработанный программный движок может быть подвергнут несанкционированному доступу и с хищением содержимого, в связи с этим иногда стоит учитывать процент уникальности методики разработки и кода. Программный контрольный блок (движок) должен быть написан с применением своих уникальных переменных и названий файлов, что поможет обезопасить сайт от стороннего вмешательства и не потребует покупки более дорогого хостинга для своего размещения. Существует несколько этапов в процессе освоения технологии разработки собственного движка. На первом этапе необходимо освоить среду HTML/CSS, на втором – PHP, как среду разработки динамических сайтов, на третьем – разобраться с вопросами создания и управления сайтом через админ-панель, изучив язык MS SQL. На предпоследнем этапе желательно пополнить свои знания, изучив язык интерактивных технологий JavaScript, который в дальнейшем, несомненно, пригодится. На завершающем этапе уже без особого труда написать свою собственную систему управления контентом сайта (CMS) и создавать на ее основе свои сайты. В настоящее время CMS сайта является наиболее популярной системой или методом создания сайта, учитывая факт существования некоторых движков сайтов с дополнениями на бесплатной основе.

Ключевые слова: компьютерная сеть, сайт, программирование, движок, файл, CMS, контент, интерфейс, HTML/CSS, система управления, хостинг, PHP, LAMP

Введение

На сегодняшний день абсолютно любая компьютерная сеть содержит такой неотъемлемый компонент, как сайт. Сформированная база в сфере IT-технологий и программирования образует устойчивую тенденцию к уяснению процесса создания движка сайта. В первую очередь в указанном процессе может стать необходимым име-

ющаяся разработка конкретного шаблона сайта. В этой ситуации разработка начнется с изучения особенностей HTML/CSS. Данная необходимость возникает в процессе разработки собственного проекта с одновременным использованием бесплатных блоговых ресурсов, предоставляющих требуемые условия для отображения текстовых и графических сегментов web-страницы [1].

Особенности разработки программного контрольного блока с применением уникальных переменных и названий файлов

Однако это весьма ограниченные возможности, а это означает, что в перспективе у разработчика формируется необходимость в создании собственного сайта на платном хостинге, что в итоге обычно представляет собой стандартный статистический сайт из нескольких страниц, созданный на простом HTML, с небольшими возможностями использования динамических элементов.

В случае успешности авторского проекта неизбежно постепенное разрастание сайта до нескольких десятков страниц. При этом для многостраничного сайта самые стандартные приемы по поддержке сайта (например, потребность в изменении дизайна или вставка нового блока, счетчика или баннера одновременно на всех страницах сайта) приводят к перелопачиванию всех страниц, и нужно в каждой отдельно взятой изменять код, что крайне неэффективно и связано со значительными временными затратами [2]. В этот момент и созревает идея по упрощению его обслуживания.

Конечно же, ситуация не является безвыходной. Для этого в первую очередь дополнительно следует обратиться к теории, а именно к языку программирования PHP, что даст возможность, в свою очередь, перейти на следующий уровень – создание динамических сайтов на PHP. Примечательно, что разработанный в среде PHP специальный шаблон существенно упростит не только сам процесс создания, но и поддержки многостраничных сайтов. Также появится возможность легко изменять дизайн, добавлять новые страницы, вставлять дополнительные блоки и т. д.

Однако в этом случае возникает одно небольшое неудобство, а именно – это редактирование всех страниц в «родной» среде с последующей их загрузкой по ftp на сервер. Но данный нюанс легко компенсируется теми преимуществами, которыми теперь обладает разработчик.

Помимо этого, среда PHP предоставляет возможность использования базы данных MySQL, создания административной панели с возможностью редактирования контента и управления разнообразными дополнениями и приложениями, созданными ранее web-страницами, несмотря на наличие исключительно строки вывода наполнения самой страницы [3].

Процесс разработки программного контрольного блока контента как собственной системы его управления

Далее в процессе автоматизации процедур создания сайтов следует этап разработки и использования так называемого «движка», представляющего собой программный комплекс, управляющий контентом (программный контрольный блок), с помощью которого в рамках контента реализуются разработка, поддержка и управление абсолютно всеми процессами.

Необходимо подчеркнуть, что система управления контентом сайта – СУКС (CMS, движок сайта) – это специальное скриптовое приложение либо программное обеспечение для полуавтоматического управления содержимым сайта. В него включены также выполнение задач обработки запросов, отправка в браузер сформированных страниц сайта. Применение информационной базы рассматривается как самый успешный вариант при использовании движка, в другом случае можно использовать отдельно взятые файлы, но при этом нет гарантии повышенной нагрузки на хранилища серверов [3; 4]. На сегодня СУКС (CMS) – самая популярная система или метод разработки сайта, принимая во внимание наличие некоторых движков сайтов с учетом бесплатных дополнений. Отдельные сравнения рейтингов бесплатных CMS приведены ниже на рисунке.

Помимо этого, с целью облегчения процесса обучения работе с какой-либо средой CMS имеются разнообразные аудио- и видеоматериалы.

Наиболее популярен и распространен на сегодня движок WordPress. С его помощью собрано большинство блогов и сайтов Рунета. В результате это привело к тому, множество сайтов имеют коды с одинаковыми файлами и переменными и целыми сегментами кодовых блоков, что, в свою очередь, представляет собой широкое поле деятельности для грамотного взломщика [5].

Создание собственного движка, блока управления, программного контроллера объясняется стремлением собственника будущего сайта к его уникальности и простоте в использовании, к наличию оптимального интерфейса, к акценту на информационную безопасность, оригинальные эстетику и дизайн, что поспособствует приданию сайту определенной исключительности.

Сам ход изучения технологии разработки собственного движка можно разделить на несколько этапов. На первом этапе осваивается среда HTML/CSS, второй этап – PHP, он посвящен освоению разработки динамических сайтов, на третьем этапе необходимо разобраться с задачами

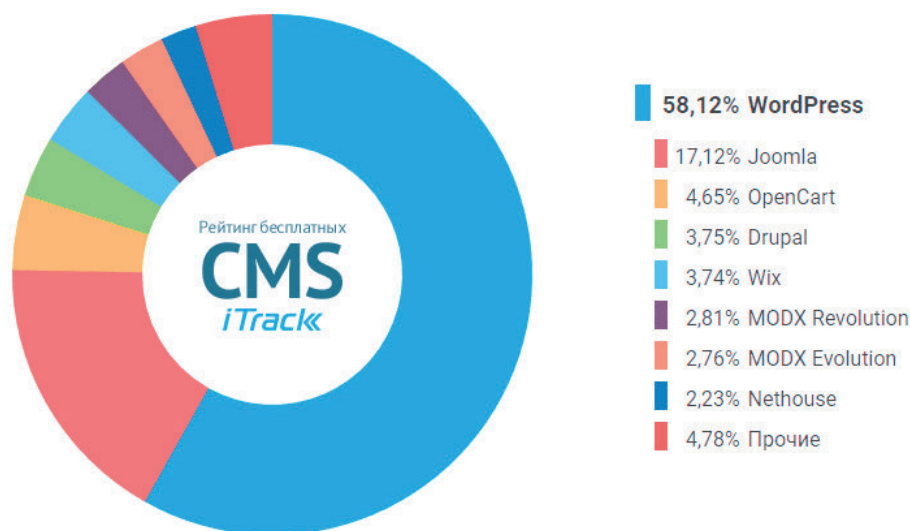


Рисунок. Рейтинг бесплатных CMS

создания и управления сайтом посредством админ-панели и изучения языка MS SQL. На последнем этапе не лишним будет пополнить свои знания посредством изучения языка интерактивных технологий JavaScript, который, несомненно, пригодится в дальнейшем. И наконец, на завершающем этапе, уже не прилагая особых усилий, можно написать свою собственную CMS и на ее основе разрабатывать собственные сайты.

Более детально следует разобраться в структуре движка, поскольку при отсутствии знаний о его структуре практически невозможно уяснить принцип его работы.

База данных: на всех платных хостингах доступны услуги использования баз данных MS SQL, в которых, как правило, содержатся все страницы сайта и вся дополнительная информация.

Панель управления: данный раздел или модуль непосредственно решает задачи, влекущие за собой максимальное упрощение обслуживания всего ресурса в целом. Есть возможность менять дизайн с панели администратора, а также добавлять или удалять страницы, редактировать, то есть делать все необходимое, если такая функция запрограммирована. А осуществить это несложно, достаточно предварительно изучить нужную информацию [6]. Редактором для панели администратора может послужить абсолютно любой свободный проект, например, можно воспользоваться редактором SKEditor.

Сам движок сайта: CMS, или система управления, – здесь называют по желанию. Этот модуль осуществляет управление показом страниц, всей навигацией и реализует все дополнительные функции, которые на него возложены. Суть его работы сводится к следующему: с помощью файла .htaccess* все без исключения запросы к

сайту переадресуются на основной php-скрипт, непосредственно формирующий страницу по запросу, учитывая тип запроса: GET или POST. htaccess (hypertextaccess) – файл дополнительной конфигурации веб-сервера Apache, а также подобных ему серверов. В этой ситуации становится возможным использование разнообразных дополнительных конфигураций и обновлений для эффективной работы веб-сервера в соответствующих папках. При этом данные каталоги могут не изменять ключевых конфигураций, но будет осуществляться контроль доступа к каталогам и замены типологии файлов.

Аналогом htaccess является httpd.conf, разница заключается лишь в действии исключительно на тот каталог, в котором располагается, и на его дочерние каталоги. Файл htaccess можно разместить в любом каталоге. Директивы этого файла действуют на все файлы в текущем каталоге и во всех его подкаталогах (если эти директивы не переопределены директивами нижележащих файлов .htaccess).

С целью возможности использования файлов htaccess необходимо произвести соответствующие настройки главного конфигурационного файла. Как правило, абсолютное большинство хостеров позволяют использование собственных файлов htaccess.

Пути к файлам и каталогам следует указывать от корня сервера, например /var/www/domain.com/htdocs/. Он делает это просто, разбирая URL на составляющие, выводя необходимую информацию из базы данных. Если запрошенная страница отсутствует, скрипт выводит текст ошибки.

Шаблонизатор, получая данные от движка, выводит их на экран согласно дизайну, созданному для HTML-шаблона, всю информацию, на-

ходящуюся в базе данных, касающуюся адреса, набранного в браузере для каждой страницы конкретно, и число раз, соответствующее количеству запросов одновременно либо последовательно [7].

Непосредственно процесс работы не слишком сложный и включает в себя двухсторонние обмены между базой данных и панелью управления, где происходят манипуляции с наполнением сайта. Далее движок извлекает информацию из базы данных и перенаправляет ее шаблонизатору, переадресующему уже обработанный код в браузер, который и демонстрирует пользователю то, что он видит на экране.

Пользователь имеет возможность напрямую общаться с движком, применяя навигации по сайту, а также при помощи добавления комментария. Здесь процесс общения идет немного по-другому. Движок извлекает из базы данных информацию о системе комментирования и принуждает шаблонизатор отобразить пользователю HTML-форму, где тот может оставить свой комментарий. Данная информация сперва уходит в движок, который, в свою очередь, ее обрабатывает, следуя заданному алгоритму. Далее движок чистит и редактирует код комментария, а затем отправляет его на хранение в базу данных и шаблонизатору, который и загружает его в браузер пользователя.

CMS включает в себя множество дополнительных модулей, обеспечивающих выполнение необходимых функций управления, задействованных пользователем. Последнему для работы с CMS вовсе не обязательно иметь навыки программирования, но при этом ему необходимо вникать в саму среду управления для полноценного авторского создания своего сайта.

Заключение

Поскольку применение общедоступного движка чревато тем, что кто-нибудь может воспользоваться такой общедоступностью и найдет очередное слабое звено, с помощью чего возможен несанкционированный доступ с последующим управлением содержимым сайта, это необходимо иметь в виду и осознавать, что выходом из данной ситуации может быть исключительно разработка собственной системы управления контентом.

Такая система сможет выполнять только необходимый набор функций, будет написана с применением своих уникальных переменных и названий файлов, что, в свою очередь, хотя и не полностью, но в большей степени сможет обезопасить сайт от постороннего вмешательства и не

затребует приобретения более дорогого хостинга для своего размещения.

Языком программирования выбран PHP – интерпретируемый скриптовый язык программирования, созданный специально для генерации HTML-страниц на веб-сервере и работы с базами данных. В сфере веб-программирования на сегодняшний день PHP наиболее распространен (наряду с Perl, ASP/.NET и Python) благодаря простоте, скорости выполнения и богатой функциональности. PHP распространяется свободно и поддерживается абсолютным большинством представителей сетевого хостинга [7].

Название PHP расшифровывается как PHP: Hypertext Preprocessor, или Personal Home Page. Он был создан как надстройка над Perl с целью облегчить разработку веб-страниц. Одна из самых сильных его сторон – возможность расширения ядра [8]. Интерфейс написания расширений вызвал интерес к PHP со стороны множества разработчиков своих модулей. Это позволило PHP работать с огромным количеством баз данных, протоколов, поддерживать большое число API.

PHP поддерживает ООП (деструкторы, открытые, закрытые и защищенные члены и методы, final-члены и методы, интерфейсы и клонирование объектов) и XML [9]. Выбирая хостинг, можно остановиться на LAMP (LinuxApacheMySQLPhp), преимущества этой связки очевидны. LAMP – акроним, обозначающий набор (комплекс) серверного программного обеспечения, широко используемый во Всемирной паутине [10]. LAMP назван по первым буквам входящих в его состав компонентов: Linux – операционная система Linux; Apache – веб-сервер; MySQL – СУБД; PHP – язык программирования, используемый для создания веб-приложений (помимо PHP могут подразумеваться другие языки, такие как Perl и Python). Акроним LAMP можно использовать при обозначении: инфраструктуры веб-сервера; парадигмы программирования; пакета программ [11].

Несмотря на то что изначально эти программные продукты не разрабатывались непосредственно для работы друг с другом, такая взаимосвязь получила широкую популярность из-за своей гибкости, производительности и низкой стоимости (все ее составляющие являются открытыми и могут быть бесплатно загружены из Internet). Набор LAMP входит в состав большинства дистрибутивов Linux и предоставляется большинством хостинговых компаний.

Литература

1. Бунин О. Системы управления контентом. URL: <http://www.osp.ru/nets/2003/04-05/150045> (дата обращения: 02.02.2020).
2. Интернет-агентство АЛБТЕРА «Система управления сайтом». URL: <http://web.altermedia.com/cms> (дата обращения: 02.02.2020).
3. Золкин А.Л., Просвиров Ю.Е. Программные средства контроля и прогнозирования параметров износа коллекторов тяговых электродвигателей // Вестник СамГУПС. 2009. № 1. С. 60–63.
4. Берёза Н.В. Современные тенденции развития мирового и российского рынка информационных услуг // Инженерный вестник Дона. 2012. № 2. URL: ivdon.ru/ru/magazine/archive/n2y2012/758 (дата обращения: 02.02.2020).
5. Тормозов В.С., Золкин А.Л., Василенко К.А. Настройка, обучение и тестирование нейронной сети долгой краткосрочной памяти для задачи распознавания образов // Промышленные АСУ и контроллеры. 2020. № 3. С. 52–57.
6. Бондарев В.В. Введение в информационную безопасность автоматизированных систем: учебное пособие. М.: Изд-во МГТУ им. Н.Э. Баумана, 2017. 255 с.
7. Панкратов С.А. Использование графической информации для защиты программного и информационного обеспечения // Инженерный вестник Дона. 2012. № 2. URL: ivdon.ru/ru/magazine/archive/n2y2012/792 (дата обращения: 02.02.2020).
8. Уилсон Э. Мониторинг и анализ сетей. Методы выявления неисправностей. М.: Лори, 2016. 480 с.
9. Аладышев О.С., Овсянников А.П., Шабанов Б.М. Развитие корпоративной сети Межведомственного суперкомпьютерного центра. URL: <http://vbakanov.ru/methods/1441/corpor.pdf> (дата обращения: 02.02.2020).
10. Perlman R. Interconnections: Bridges & Routers. Boston: Addison-Wesley, 2016. 245 p.
11. Khalil H.K. Nonlinear Systems; 3rd ed. New Jersey: Pearson, 2016. 766 p. URL: <https://en.bookfi.net/book/1417228> (дата обращения: 02.02.2020).

Получено 05.05.2020

Василенко Константин Александрович, преподаватель Колледжа сервиса и дизайна Владивостокского государственного университета экономики и сервиса. 690092, Российская Федерация, Приморский край, г. Владивосток, ул. Добровольского, 20. Тел. +7 964 453-06-36. E-mail: k2857@mail.ru

Золкин Александр Леонидович, к.т.н., преподаватель кафедры естественнонаучных и общепрофессиональных дисциплин Самарского филиала Волжского государственного университета водного транспорта. 443099, Российская Федерация, г. Самара, ул. Молодогвардейская, 62-64. Тел. +7 960 825-68-49. E-mail: alzolkin@list.ru

Абрамов Николай Викторович, студент Дальневосточного федерального университета (ДФУ). 690920, Российская Федерация, Приморский край, г. Владивосток, ул. Суханова, 8. Тел. +7 914 373-91-16. E-mail: nikolay.abramov1990@mail.ru

Курганов Даниил Олегович, студент ДВФУ. 690920, Российская Федерация, Приморский край, г. Владивосток, ул. Суханова, 8. Тел. +7 999 059-37-74. E-mail: kurganov_vl@mail.ru

COMPUTER NETWORK CONTENT: DESIGNING OF A SOFTWARE CONTROL UNIT

Vasilenko K.A.¹, Zolkin A.L.², Abramov N.V.³, Kurganov D.O.³

¹ Vladivostok State University of Economics and Service, Vladivostok, Russian Federation,

² Volga State University of Water Transport (Samara branch), Samara, Russian Federation,

³ Far Eastern Federal University, Vladivostok, Russian Federation

E-mail: k2857@mail.ru, alzolkin@list.ru, nikolay.abramov1990@mail.ru, kurganov_vl@mail.ru

The process of development of the program control unit of a content (engine) is reviewed in the article as an in-house system of its control; its vulnerabilities as well as set of necessary functions that are inherent to the specified engine are indicated in the article. Each new developed software engine can be subjected to unauthorized access and theft of content. Due to this fact, sometimes it is worth to

consider the percentage of uniqueness of the development methodology and code. The program control unit (engine) shall be written using its unique variables and file names. It will help to protect the site from third-party interference and will not require the purchase of the more expensive hosting for its placement. There are several stages in the process of application of the technology of development of the in-house engine. At the first stage, the HTML/CSS environment shall be assimilated. At the second stage, the PHP environment shall be assimilated as an environment for development of dynamic sites. At the third stage the issues of creating and managing a site through the admin panel shall be solved by studying the MSQL language. At the penultimate stage, it is advisable to replenish the knowledge by studying the language of interactive technologies JavaScript. It would be undoubtedly useful in the future. And at the final stage, it will be relatively easy to write your own CMS and create your own websites on its basis. Currently, the CMS is the most popular system or method of creation of the site, even despite the fact that there are some free-issued site engines with add-ons.

Keywords: *computer network, site, programming, engine, file, CMS, content, interface, HTML/CSS, control system, hosting, PHP, LAMP*

DOI: 10.18469/ikt.2021.19.1.04

Vasilenko Konstantin Alexandrovich, Service and Design College of the Vladivostok State University of Economics and Service, 20, Dobrovolskogo Street, Vladivostok, Primorsky krai, 690092, Russian Federation; Highest category teacher. Tel. +7 964 453-06-36. E-mail: k2857@mail.ru

Zolkin Alexander Leonidovich, Samara Branch of the Volga State University of Water Transport, 62-64, Molodogvardeyskaya Street, Samara, 443099, Russian Federation; Teacher of Natural-science and General Professional Disciplines Department, PhD in Technical Sciences. Tel. +7 960 825-68-49. E-mail: alzolkin@list.ru

Abramov Nikolai Viktorovich, Far Eastern Federal University, 8, Sukhanova Street, Vladivostok, Primorsky krai, 690092, Russian Federation; Student (bachelor). Tel. +7 914 373-91-16. E-mail: nikolay.abramov1990@mail.ru

Kurganov Daniil Olegovich, Far Eastern Federal University, 8, Sukhanova Street, Vladivostok, Primorsky krai, 690092, Russian Federation; Student (bachelor). Tel. +7 999 059-37-74. E-mail: kurganov_vl@mail.ru

References

1. Bunin O. Content management systems. URL: <http://www.osp.ru/nets/2003/04-05/150045> (accessed: 02.02.2020).
2. Internet agency ALTERA «Content management system». URL: <http://web.altera-media.com/cms> (accessed: 02.02.2020).
3. Zolkin A.L., Prosvirov Yu.E. Software tools for monitoring and predicting wear parameters of traction electric motor collectors. *Vestnik SamGUPS*, 2009, no. 1, pp. 60–63. (In Russ.)
4. Bereza N.V. Modern trends in the development of the world and Russian market of information services. *Inzhenernyj vestnik Dona*, 2012, no. 2. URL: ivdon.ru/ru/magazine/archive/n2y2012/758 (accessed: 02.02.2020). (In Russ.)
5. Tormozov V.S., Zolkin A.L., Vasilenko K.A. Setting up, training and testing a neural network of long short-term memory for the problem of pattern recognition. *Promyshlennyye ASU i kontroly*, 2020, no. 3, pp. 52–57. (In Russ.)
6. Bondarev V.V. *Introduction to Information Security of Automated Systems: A Tutorial*. Moscow: Izd-vo MGTU im. N.E. Bauman, 2017, 255 p. (In Russ.)
7. Pankratov S.A. The use of graphic information for the protection of software and information support. *Inzhenernyj vestnik Dona*, 2012, no. 2. URL: ivdon.ru/ru/magazine/archive/n2y2012/792 (accessed: 02.02.2020). (In Russ.)

8. Uilson E. *Monitoring and Analysis of Networks. Troubleshooting Methods*. Moscow: Lori, 2016, 480 p. (In Russ.)
9. Aladyshev O.S., Ovsjannikov A.P., Shabanov B.M. Development of the corporate network of the Interdepartmental Supercomputer Center. URL: <http://vbakanov.ru/metods/1441/corpor.pdf> (accessed: 02.02.2020).
10. Perlman R. *Interconnections: Bridges & Routers*. Boston: Addison-Wesley, 2016, 245 p.
11. Khalil H.K. *Nonlinear Systems*; 3rd ed. New Jersey: Pearson, 2016, 766 p. URL: <https://en.bookfi.net/book/1417228> (accessed: 02.02.2020).

Received 05.05.2020

НОВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

УДК 004.738

МОНИТОРИНГ СОСТОЯНИЯ ОБЪЕКТОВ ИНФРАСТРУКТУРЫ ЖЕЛЕЗНОДОРОЖНОГО ТРАНСПОРТА НА ОСНОВЕ ТЕХНОЛОГИИ FUSN

Белоногов А.С., Шорохов Н.С.

Самарский государственный университет путей сообщения, Самара, РФ

E-mail: abelonogov@rambler.ru

В рамках концепций «Цифровая железная дорога» и «Интеллектуальные транспортные сети» предлагается новый способ организации системы мониторинга состояния объектов инфраструктуры железнодорожного транспорта на примере опор контактной сети и устройств линейных кабельных сооружений. За основу взята технологическая платформа всепроникающих сенсорных сетей Интернета вещей, в частности класс «летающие сенсорные сети». Кратко представлены основные характеристики и особенности построения летающих сенсорных сетей. Проведено аналитическое моделирование летающих сенсорных сетей, которое позволило оценить различные показатели сети на физическом уровне, установить зависимости скорости передачи данных от мощности сигнала и расстояния между приемными и передающими антеннами. Предложено архитектурное решение по системе мониторинга состояния объектов инфраструктуры железнодорожного транспорта.

Ключевые слова: *система мониторинга, опоры контактной сети, устройства линейных кабельных сооружений, всепроникающие сенсорные сети, летающие сенсорные сети, Интернет вещей, наземный и летающий сегменты, шлюз, беспилотный летательный аппарат*

Введение

Бесперебойность перевозочного процесса ставит жесткие требования к объектам и устройствам инфраструктуры железнодорожного транспорта. В условиях, когда с ростом скорости последствия отказов устройств могут быть особенно опасными, весьма актуальным становится вопрос удаленного мониторинга неисправностей и предотказных состояний устройств инфраструктуры железнодорожного транспорта.

Создание системы мониторинга рассмотрим на примере контроля состояния опор контактной сети и элементов линейных кабельных сооружений связи. Контроль состояния опор контактной сети железной дороги в режиме реального времени позволит предотвратить аварии и значительно увеличить рентабельность эксплуатации дороги, так как будет осуществляться ремонт или замена лишь тех опор, которые действительно нуждаются в этом. В линейных кабельных сооружениях

связи актуальным является вопрос контроля герметичности кабельных муфт, контроля целостности оболочек кабелей, находящихся под постоянным давлением, а также мониторинга рабочей температуры кабелей.

Особенности беспроводных летающих всепроникающих сенсорных сетей

Рассмотрим особенности применения беспроводных летающих всепроникающих сенсорных сетей (Flying Ubiquitous Sensor Networks – далее FUSN) для сбора данных о состоянии устройств и объектов железнодорожной инфраструктуры. Сети FUSN представляют собой один из классов всепроникающих сенсорных сетей USN. Преимуществами таких сетей являются [1]:

- отсутствие необходимости в прокладке кабелей для передачи данных;
- низкая стоимость оборудования, пусконаладочных работ и технического обслуживания;