ISSN: 0368-0797. Известия высших учебных заведений. Черная металлургия. 2017. Том 60. № 7. С. 573 – 579.
© 2017. *Гурин И.А., Лавров В.В., Спирин Н.А., Никитин А.Г.*

УДК 658.52.56.004.057.5

ВЕБ-ТЕХНОЛОГИИ ПОСТРОЕНИЯ АВТОМАТИЗИРОВАННЫХ ИНФОРМАЦИОННО-МОДЕЛИРУЮЩИХ СИСТЕМ ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ В МЕТАЛЛУРГИИ

***Гурин И.А.*[1], *аспирант кафедры «Теплофизика и информатика в металлургии»* (ivan.gurin@urfu.ru)**

***Лавров В.В.*[1], *д.т.н., профессор кафедры «Теплофизика и информатика в металлургии»* (v.v.lavrov@urfu.ru)**

***Спирин Н.А.*[1], *д.т.н., профессор кафедры «Теплофизика и информатика в металлургии»* (n.a.spirin@urfu.ru)**

***Никитин А.Г.*[2], *д.т.н., доцент, директор института машиностроения и транспорта, заведующий кафедрой механики и машиностроения* (dir.imit@mail.ru)**

[1] **Уральский федеральный университет имени первого Президента России Б.Н. Ельцина**
(620002, Россия, г. Екатеринбург, ул. Мира, 28)
[2] **Сибирский государственный индустриальный университет**
(654007, Россия, Кемеровская обл., г. Новокузнецк, ул. Кирова, д. 42)

Аннотация. Рассмотрены особенности программного обеспечения (ПО) информационных систем (ИС) для работы инженерно-технологического персонала на металлургических предприятиях. ПО включает автоматизированные рабочие места, системы поддержки принятия решений, информационно-моделирующие системы, экспертные системы и др. Они представляют собой оконные (десктопные) приложения, написанные на высокоуровневых языках программирования (Visual C#, Visual Basic и др.). Обработка технологической информации, поступающей с сервера управления базами данных (СУБД) предприятия, включает в себя необходимость решения комплекса задач математического программирования, систем дифференциальных уравнений, задач математической физики и др. Такие задачи невозможно решить с помощью стандартного набора математических средств языков программирования общего назначения, поэтому разработка информационно-моделирующих систем осуществляется при взаимодействии с внешним ПО, например Microsoft Excel, MATLAB. В основе взаимодействия с Microsoft Excel лежит технология COM Interop, предполагающая установку ПО Microsoft Office на каждый клиентский компьютер. Для взаимодействия с MATLAB требуется предварительная сборка библиотеки в среде MATLAB Compiler и ее подключение к программе. Выполнение на клиентском компьютере достигается установкой свободно распространяемого пакета MATLAB Runtime. Однако требования по функциональности, доступности и кроссплатформенной переносимости, предъявляемые к современным информационным системам предприятия, не могут быть выполнены в оконных приложениях, использующих Windows Forms. Этим обусловлен поиск новых технологий и средств создания информационных систем. Наиболее рациональной является технология построения веб-приложений на базе фреймворка ASP.NET MVC, позволяющая без модификации перенести математические библиотеки, модули взаимодействия с Microsoft Excel, MATLAB из Windows Forms. Представлена структура веб-приложения, используемая при разработке ПО, отвечающего современным требованиям к работе информационных систем, что предполагает наличие на веб-странице следующих функциональных областей: логотип и заголовок текущей страницы, меню состояния сеанса, функциональное меню, групповые операции, уведомления и рабочее пространство.

Ключевые слова: информационная система, программное обеспечение, архитектура, веб-программирование, математическая библиотека, дизайн, Windows Forms, ASP.NET MVC, MATLAB Runtime.

DOI: 10.17073/0368-0797-2017-7-573-579

Информационными системами для работы инженерно-технологического персонала на промышленном предприятии являются автоматизированные рабочие места, системы поддержки принятия решений, информационно-моделирующие системы, экспертные системы и др. Такие системы предназначены для обработки и представления информации о технологическом процессе, решения технологических задач, модельной поддержки технологии обработки материалов, что в конечном итоге направлено на улучшение технико-экономических показателей и эффективности металлургического производства. В доменном производстве информационно-моделирующие системы позволяют решать комплекс взаимосвязанных технологических задач, в частности производить: расчет материальных и тепловых балансов доменной плавки; моделирование теплового, шлакового и газодинамического режимов работы доменных печей; выбор оптимального состава доменной шихты; расчет зоны вязкопластичного состояния железорудных материалов; прогнозирование технологических ситуаций; оптимизацию распределения топливно-энергетических ресурсов в группе доменных печей [1 – 5].

Большинство созданных автоматизированных информационных систем для работы инженерно-техно-

логического персонала представляют собой оконные приложения, написанные на высокоуровневых языках программирования (Visual C#, Visual Basic или других языках разработки приложений для платформы Microsoft .NET Framework). Это клиент-серверное ПО, в котором сервером является СУБД предприятия, хранящая в базе данных (БД) первичную технологическую информацию. Клиентское приложение запрашивает данные из БД, обрабатывает их и выводит расчетные показатели на пользовательскую форму. Процесс обработки данных, как правило, включает в себя необходимость численного решения: задач математического программирования, систем алгебраических и дифференциальных уравнений, задач математической физики. Такие задачи невозможно решить с помощью стандартного набора математических средств языков программирования общего назначения, поэтому разработка информационно-моделирующих систем осуществляется при взаимодействии с внешним ПО, например, программой для работы с электронными таблицами Microsoft Excel, пакетом прикладных программ для решения задач технических вычислений MATLAB. Связь с внешними приложениями накладывает ограничения на выбор средств разработки, поэтому рассмотрим особенности интеграции разрабатываемых на языке программирования C# приложений с Microsoft Excel и MATLAB.

Интеграция C# и Microsoft Excel. В основу взаимодействия с Microsoft Excel положена технология COM Interop, позволяющая объектам COM взаимодействовать с объектами программной платформы .NET Framework. Поддержка работы технологии COM Interop обеспечивается подключением библиотеки Microsoft. Office.Interop.Excel. Библиотека позволяет выполнять практически любые операции, которые можно осуществить вручную через интерфейс пользователя. Она включает возможности чтения и изменения ячеек на листах книги, создание диаграмм и вызов макросов.

Рассмотрим схему решения математической задачи на языке программирования C# с подключением к Microsoft Excel. Для решения задачи используется заранее подготовленный файл электронной таблицы, в котором записаны структура математической модели (этапы решения технологической задачи) и нормативно-справочные данные для расчета. Этот файл встраивается в создаваемое ПО и открывается в скрытом режиме при запуске пользователем алгоритма расчета. После открытия осуществляется обновление заранее определенного диапазона ячеек, вызывается процесс расчета, а затем происходит считывание его результатов. Вызов процесса расчета реализуется макросом. Шаги такой процедуры представлены на рис. 1.

Первоначально математические модели и алгоритмы обычно проверяются и отлаживаются в файлах электронных таблиц на наборе тестовых данных, поэтому подход интеграции C# и Microsoft Excel может повысить скорость разработки ПО за счет исключения времени на разработку и отладку математической библиотеки.

Такой подход имеет ряд недостатков:

– максимальное количество ограничений в надстройке «Поиск решения» при нахождении целевой функции составляет 100; это ограничение не позволяет гибко решать сложные задачи с большим количеством ограничений;

– низкая скорость решения задачи; действия с файлом электронной таблицы осуществляются через открытие приложения Microsoft Excel в фоновом режиме, открытие занимает значительный промежуток времени, а выполнение операций чтения и записи осуществляется с задержкой;

– зависание процесса Excel на компьютере пользователя; даже корректное программное завершение работы с приложением Microsoft Excel и освобождение ресурсов на стороне приложения не приводит к реальному освобождению ресурсов на вычислительной станции;

– необходимость хранения отношений между данными приложения и соответствующими ячейками Excel-файла; другими словами, логика работы клиентского приложения связана с реализацией модели оптимизации в Excel-файле, поэтому любое изменение модели требует постоянной коррекции диапазонов ячеек и перекомпиляции приложения;

– требования программной совместимости версий; библиотека объектов поставляется вместе с приложением Microsoft Excel и может быть несовместима с версией, используемой при разработке программного обеспечения.

Интеграция C# и MATLAB. Отличительной особенностью MATLAB от других математических пакетов является наличие встроенных в него разнообразных средств решения большинства известных математических задач, которые можно использовать (встраивать) при создании автономных приложений [6]. Такая возможность реализуется с помощью продукта MATLAB

Рис. 1. Схема решения математической задачи при интеграции C# и Microsoft Excel

Fig. 1. The scheme of mathematical problem solving through integration of C # and Microsoft Excel

Compiler. Этот продукт позволяет собирать модули в виде:

– независимых программных приложений;

– C/C++ библиотек (DLL в Windows, разделяемые библиотеки в Linux);

– программных компонентов, таких как классы Java, сборки .NET или надстройки Microsoft Excel для использования в других приложениях.

Приложения и библиотеки, созданные с помощью MATLAB Compiler, используют свободно распространяемое исполняемое ядро, называемое MATLAB Runtime (MCR). Это позволяет запускать приложения, которые используют MATLAB-функции без необходимости установки дорогостоящей копии MATLAB.

Одним из инструментов для создания математических библиотек или внешних компонентов является Deployment Tool, представляющий собой графический интерфейс. Вызов данного инструмента осуществляется командой *deploytool*.

Название проекта определяет пространство имен, используемое в дальнейшем в Visual C#. Проект состоит из классов, которые в свою очередь включают M-файлы. Каждый M-файл включает одну функцию, которая вызывается как метод соответствующего класса в C#.

Анализ опыта решения математических задач с использованием пакетов Microsoft Excel и MATLAB позволяет отметить следующее:

– работа с файлом электронной таблицы осуществляется только при наличии установленной версии Microsoft Office, а вызов функций MATLAB осуществляется при наличии установленной версии свободно распространяемого исполняемого ядра MATLAB Runtime;

– использование внешних пакетов позволяет существенно сократить время разработки ПО за счет использования реализованных в них алгоритмов и методов решения математических задач.

Рассмотрим структуру информационной системы для работы инженерно-технологического персонала на металлургическом предприятии. Как уже говорилось, приложения, входящие в информационную систему, являются оконными. Таких приложений может быть несколько на одном рабочем месте, каждое рабочее место может включать различное ПО. Структура такой системы представлена на рис. 2.

Требования, предъявляемые к современным информационным системам (ИС) предприятия [7], не могут быть выполнены в оконных приложениях, в частности:

– функциональность – свойство, обусловливающие способность ПО ИС выполнять определенный перечень функций, которые удовлетворяют установленные или предполагаемые потребности в соответствии с его назначением;

– надежность функционирования – свойство ПО ИС сохранять работоспособность и преобразовывать

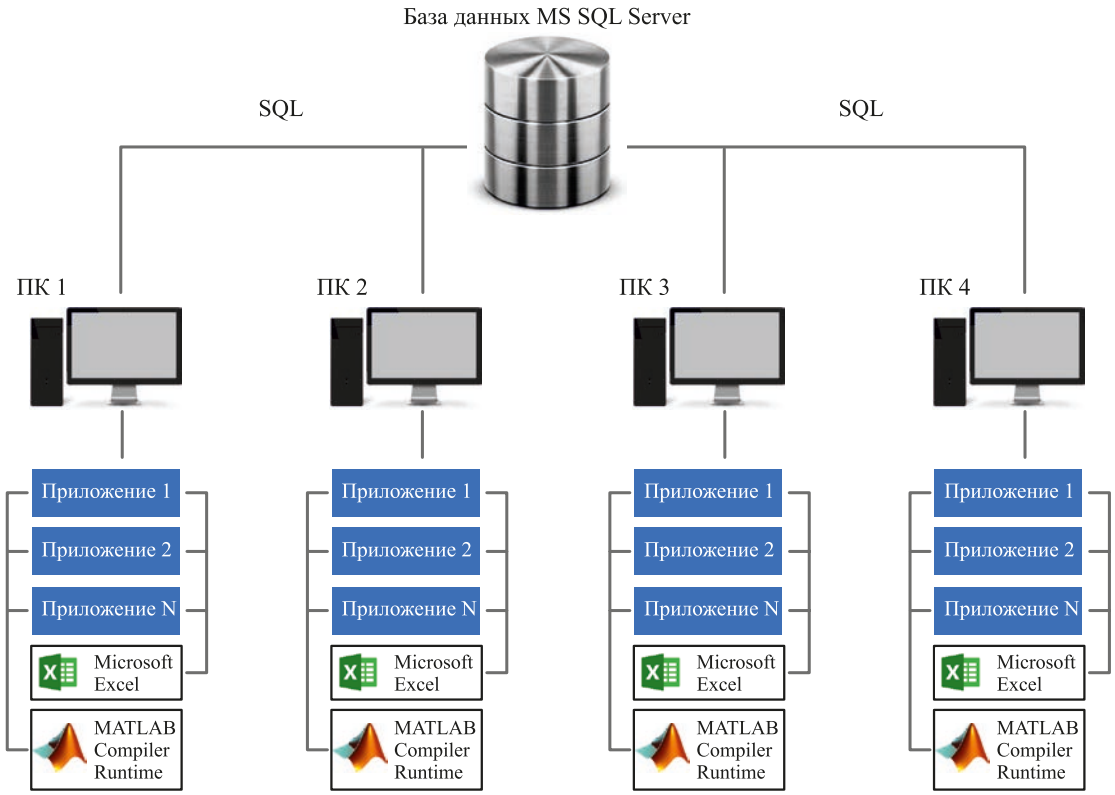

Рис. 2. Структура информационной системы для работы инженерно-технологического персонала металлургического предприятия

Fig. 2. Structure of the information system for engineering personnel of metallurgical enterprise

исходные данные в искомый результат в номинальных условиях за установленный период времени;

– удобство использования и доступность ПО – свойство, обеспечивающие пользователям комфортные условия и доступность для использования с заданного места;

– эффективность – свойство, характеризующееся степенью соответствия используемых ресурсов среды функционирования уровню качества функционирования при заданных условиях применения;

– переносимость – свойство, обусловливающие приспособленность ПО для переноса из одной среды функционирования в другую.

Созданное ранее ПО соответствует требованиям функциональности, надежности функционирования и эффективности. Однако оно разрабатывалось под конкретную аппаратно-программную платформу, поэтому не соответствует требованиям доступности и кроссплатформенной переносимости. Современные требования подразумевают оперативный доступ к информации, что невозможно обеспечить при данном подходе. Этим обусловлен поиск новой технологии и средств создания информационных систем для инженерного персонала предприятия [8 – 18].

Все приложения можно разделить на две больших категории: десктоп и веб. Десктоп-приложения при используемой клиент-серверной архитектуре делятся на «толстый» и «тонкий» клиенты в зависимости от «функционала», который реализуется на машине клиента. Архитектура «тонкого» клиента позволяет лишь увеличить скорость разработки кроссплатформенного ПО, но не снимает задачу разработки отдельных продуктов под каждую платформу.

Современные десктоп-приложения разрабатываются как универсальные приложения, которые компилируются под разные платформы. Однако и они не решают в полной мере проблему переносимости на любую платформу, будь то смартфон, планшет или ноутбук.

Веб-технологии позволяют создавать приложения с различной переносимостью. Например, Rich Internet Application (RIA), использующие технологии ActiveX, Adobe Flash, Java, Silverlight, не позволяют создавать кроссплатформенные и кроссбраузерные. Традиционные веб-приложения доступны с любого устройства, имеющего браузер, позволяют создавать адаптивные сервисы, обеспечивающие правильное отображение данных на различных устройствах.

Среди недостатков веб-приложений можно отметить их общедоступность, необходимость наличия соединения с сетью и невозможность запуска сложных (тяжеловесных) проектов. Разрабатываемые системы являются клиент-серверными, поэтому проблема соединения с сетью не является актуальной. Разрабатываемые приложения не являются тяжеловесными, а проблема общедоступности решается настройкой безопасности сети.

На основе проведенного анализа рациональной технологии для перехода от оконных приложений Windows Forms к построению веб-приложений выбрана разработка на базе фреймворка ASP.NET MVC [8, 10]. И те, и другие используют программную платформу .NET Framework. Это позволяет без модификации перенести динамически подключаемые библиотеки, модули взаимодействия с Microsoft Excel и MATLAB из Windows Forms. Переход к ASP.NET не требует понимания бизнес-процессов, технологии производства на предприятии, знания математических основ моделирования, а включает рутинную работу по созданию веб-страниц и обработчиков, аналогичных соответствующим формам в оконном приложении.

Рассмотрим структуру информационной системы для работы инженерно-технологического персонала, построенную на базе ASP.NET MVC (рис. 3).

В такой системе в качестве клиента выступает любое устройство, имеющее браузер. На клиенте теперь

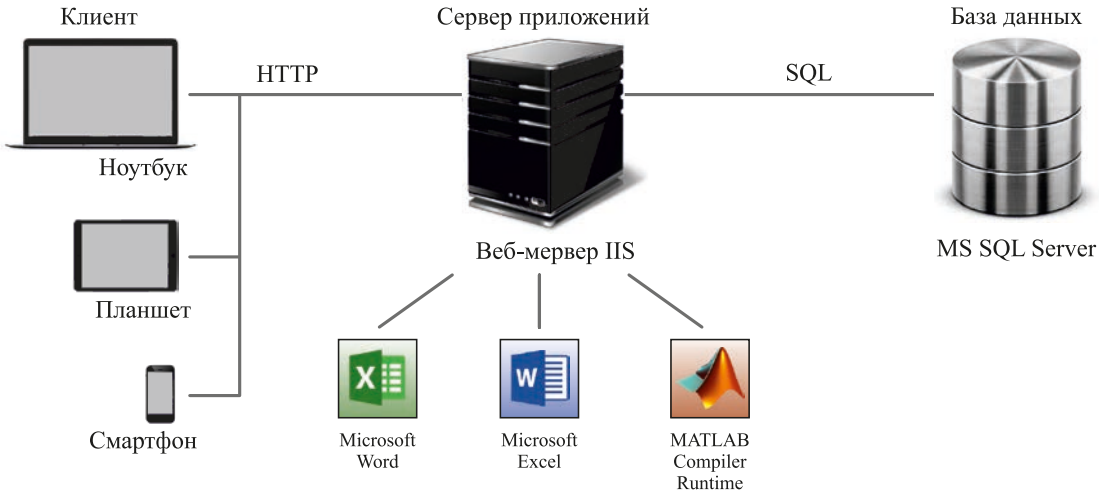

Рис. 3. Структура информационной системы для инженерно-технологического персонала, построенной на базе ASP.NET MVC

Fig. 3. Structure of the information system for engineering personnel, built on the basis of ASP.NET MVC

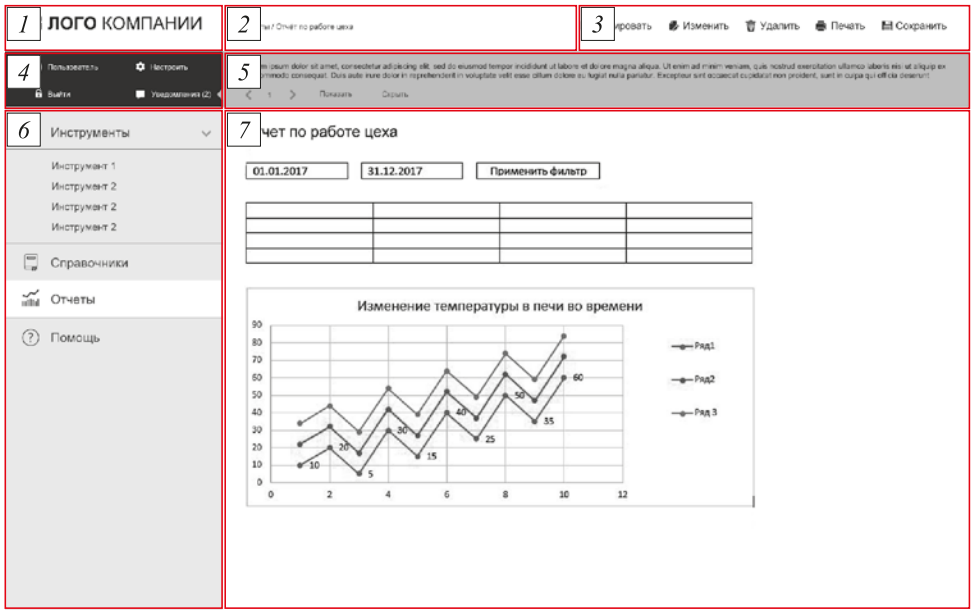

Рис. 4. Структура предлагаемого веб-приложения:
1 – логотип; *2* – заголовок текущей страницы; *3* – групповые операции; *4* – меню состояния сеанса; *5* – уведомления; *6* – функциональное меню; *7* – рабочее пространство

Structure of web application proposed:
1 – logo; *2* – title of current page; *3* – group operations; *4* – session status menu; *5* – notices; *6* – functional menu; *7* – workspace

не требуется установка специализированного программного обеспечения. Программная платформа .NET Framework, пакеты Microsoft Office и MATLAB установлены на сервере приложений. Сервер приложений выполняет роль веб-сервера, самостоятельно выполняет обращения к базе данных и взаимодействует с внешними пакетами.

На основе исследования тенденций в создании структуры и дизайна современных веб-приложений [15, 19 – 22], собственного опыта проектирования и создания информационно-моделирующих систем в аглодоменном производстве [1 – 5] разработана структура веб-приложения, представленная на рис. 4. Эта структура соответствует современным требованиям к информационным системам и предусматривает наличие на веб-странице следующих функциональных областей: логотип и заголовок текущей страницы, меню состояния сеанса и функциональное меню, групповые операции обработки данных, уведомления и рабочее пространство (формы изменения данных, табличные и графические данные).

Выводы. Комплекс требований к ПО информационно-моделирующих систем технологических процессов в металлургии по функциональности, надежности функционирования и переносимости существенно возрос, что вызывает необходимость совершенствования технологии программирования, методологии и средств разработки. Предложены варианты методов и средств разработки ПО, отвечающие современным требованиям. Установлено, что при создании информационно-моделирующих систем, в основе которых лежит клиент-серверная архитектура, этим требованиям в наибольшей степени соответствует веб-технология. Показаны преимущества разработки веб-приложений на базе «фреймворка» ASP.NET MVC при переходе на них от существующих оконных приложений. Предложен вариант структуры веб-приложения, которая предполагает сохранение функционала оконных приложений и компоновки данных в рабочем пространстве. В то же время используются удобные и привычные для веб-проектов области информационных и навигационных элементов.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Математическое моделирование металлургических процессов в АСУ ТП / Н.А. Спирин, В.В. Лавров, В.Ю. Рыболовлев и др.; под ред. Н.А. Спирина. – Екатеринбург: изд. УрФУ, 2014. – 558 с.
2. Spirin N.A., Gileva L.Y., Lavrov V.V. Information modeling system for blast furnace control. IOP Conference Series: Materials Science and Engineering [Proc. 19th International Scientific Conference on Metallurgy: Technologies, Innovation, Quality, Metallurgy 2015]. Novokuznetsk, 2016, Vol. 150, I. 1. DOI: 10.1088/1757-899X/150/1/012011.
3. Spirin N.A., Lavrov V.V., Kosachenko I.E., Onorin O.P., Istomin A.S., Burykin A.A., Shchipanov K.A. Software for the raw-materials management system in blast-furnace smelting // Metallurgist. 2015. Vol. 59. I. 1. P. 104 – 112. DOI: 10.1007/s11015-015-0068-7.
4. Lavrov V.V., Spirin N.A., Burykin A.A., Shchipanov K.A., Rybolovlev V.Y. Automatic control software for the blast-furnace shop // Steel in Translation. 2015. Vol. 45. I. 9. P. 669 – 673. DOI: 10.3103/S0967091215090090.
5. Lavrov V.V., Spirin, N.A. Automated information system for analysis and prediction of production situations in blast furnace plant. IOP Conference Series: Materials Science and Engineering [Proc. 19th

International Scientific Conference on Metallurgy: Technologies, Innovation, Quality, Metallurgy 2015]. Novokuznetsk. 2016. Vol. 150. I. l. DOI: 10.1088/1757-899X/150/1/012010.
6. Кетков Ю.Л. MATLAB 7: программирование, численные методы. – СПб.: БХВ-Петербург, 2005. – 752 с.
7. Vyatkin V. Software engineering in industrial automation: state-of-the-art review // IEEE Transactions on Industrial Informatics. 2013. Vol. 9. № 3. P. 1234 – 1249.
8. Макдональд М. WPF: Windows Presentation Foundation в .NET 4.0 с примерами на C# 2010 для профессионалов. – М.: Вильямс, 2011. – 1024 с.
9. Dimitrov B.H., Nenov H.B., Marinov A.S. Comparative analysis between methodologies and their software realizations applied to modeling and simulation of industrial thermal processes [Proc. 36th International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2013]. Opatija; Croatia; 20-24 May 2013. Article number 6596383. P. 891 – 895.
10. Amalfitano D., Fasolino A.R., Maggio V., Tramontana P., Di Mare G., Ferrara F., Scala S. Migrating legacy spreadsheets-based systems to Web MVC architecture: An industrial case study. [Proc. Software Evolution Week - IEEE Conference on Software Maintenance, Reengineering, and Reverse Engineering 2014, CSMR-WCRE 2014]. Antwerp, Belgium. 3-6 February 2014. Article number 6747201. P. 387 – 390. DOI: 10.1109/CSMR-WCRE.2014.6747201.
11. Zhang B., Bao L., Zhou R., Hu S., Chen P. A black-box strategy to migrate GUI-based legacy systems to web services. SOSE 2008 [Proc. of the 4th IEEE International Symposium on Service-Oriented System Engineering]. Jhongli, Taiwan. 18-19 December 2008. Article number 4730458. P. 25 – 31. DOI: 10.1109/SOSE.2008.8.
12. Canfora G. Software evolution in the era of software services. [Proc. 7th International Workshop on Principles of Software Evolution, IWPSE 2004]. Kyoto, Japan. 6-7 September 2004. P. 9 – 18.
13. Zdun U. Reengineering to the Web: A reference architecture [Proc. of the European Conference on Software Maintenance and Reengineering, CSMR 2002]. Budapest, Hungary. 11-13 March 2002. Article number 995801. P. 164 – 173. DOI: 10.1109/CSMR.2002.995801.
14. Tilley S., Gerdes J., Hamilton T., Huang S., Müller H., Wong K. Adoption challenges in migrating to web services. [Proc. 4th International Workshop on Web Site Evolution, WSE 2002]. Montreal, Canada. 2 October 2002. Article number 1134086. P. 21 – 29. DOI: 10.1109/WSE.2002.1134086.
15. Sreedevi E., Prasanth Y. Critical review on web services and legacy systems – And its migration // International Journal of Applied Engineering Research. 2015. Vol. 10. I. 7. P. 16321 – 16329.
16. Kastner T. Concept to migrate an access based environmental information system into a Web-based front-end solution for the environmental management of a waste disposal company [Proc. 3rd International ICSC Symposium on Information Technologies in Environmental Engineering, ITEE 2007]. Oldenburg, Germany. 29-30 March, 2007. P. 559 – 566.
17. Khalaf R., Keller A., Leymann F. Business processes for Web Services: Principles and applications // IBM Systems Journal, 2006. Vol.45. I. 2. P. 425 – 446. DOI: 10.1147/sj.452.0425.
18. Patel K., Ragha L. Survey and Analysis on Migration of Legacy System to Service Oriented Architecture // IOSR Journal of Computer Engineering, 2013. Vol. 9. I. 2. P. 32 – 35.
19. Stroulia E., El-Ramly M., Sorenson P. From legacy to web through interaction modeling [Proc. IEEE International Conference on Software Maintenance 2002]. Montreal, Canada. 3-6 October, 2002. P. 320 – 329.
20. Mijan J.L., Garrigos I., Firmenich S. Supporting personalization in legacy web sites through client-side adaptation. [Proc. 16th International Conference on Web Engineering, ICWE 2016]. Lugano, Switzerland. 6-9 June, 2016. P. 588 – 592. DOI: 10.1007/978-3-319-38791-8_54.
21. Mao C.-Y., Lu Y.-S. A method for measuring the structure complexity of Web application // Wuhan University Journal of Natural Sciences, 2006. Vol. 11. I. 1. P. 143 – 150. DOI: 10.1007/BF02831720.
22. Rossi G., Schwabe D. Object-oriented design structures in Web application models // Annals of Software Engineering, 2002. Vol. 13. I. 1-4. P. 97 – 110. DOI: 10.1023/A:1016593309733.

Поступила 14 июня 2017 г.

Izvestiya VUZov. Chernaya Metallurgiya = Izvestiya. Ferrous Metallurgy. 2017. Vol. 60. No. 7, pp. 573–579.

WEB-TECHNOLOGIES FOR CONSTRUCTION OF AUTOMATED INFORMATION-MODELING SYSTEMS OF TECHNOLOGICAL PROCESSES IN METALLURGY

***I.A. Gurin*[1], *V.V. Lavrov*[1], *N.A. Spirin*[1], *A.G. Nikitin*[2]**

[1] Ural Federal University named after the first President of Russia B.N. Yeltsin, Ekaterinburg, Russia
[2] Siberian State Industrial University, Russia, Novokuznetsk

Abstract. The features of software of information systems for the work of engineering and technological staff at metallurgical enterprises are considered. The software includes automated workstations, decision support systems, information modeling systems, expert systems, etc. It is desktop applications written in high-level programming languages (Visual C#, Visual Basic, etc.). Processing of technological information coming from the enterprise database management server (DBMS) includes the need to solve a set of mathematical programming problems, systems of differential equations, problems of mathematical physics, etc. Such tasks cannot be solved with the help of a standard set of mathematical tools for general programming languages. So the development of information modeling systems is carried out by interacting with external software – Microsoft Excel and MATLAB. The interaction with Microsoft Excel is based on COM Interop technology, which assumes the installation of Microsoft Office software on each client computer. To interact with MATLAB, you must pre-assemble the library in the MATLAB Compiler environment and connect it to the program. The execution on the client computer is achieved by installing the freely distributed MATLAB Runtime package. However, the functionality, availability, and cross-platform portability requirements for modern enterprise information systems cannot be met in desktop applications that use Windows Forms. This is due to the search for a new technology and means of creating information systems. The most rational is the technology for building web applications based on the ASP.NET MVC framework, which allows to transfer mathematical libraries, interaction modules with Microsoft Excel, MATLAB from Windows Forms without modification. The concept of the structure of a web application used in the development of software that meets modern requirements for the operation of information systems is presented. This concept suggests the availability of the following functional areas on the web page: logo and title of the current page, session status menu, functional menu, group operations, notifications and tworkspace.

Keywords: information system, software, architecture, web programming, math library, design, Windows Forms, ASP.NET MVC, MATLAB Runtime.

DOI: 10.17073/0368-0797-2017-7-573-579

REFERENCES

1. Spirin N.A., Lavrov V.V., Rybolovlev V.Yu. etc. *Matematicheskoe modelirovanie metallurgicheskikh protsessov v ASU TP* [Mathe-

matical modeling of metallurgical processes in automated process control system]. Spirin N.A. ed. Ekaterinburg: UrFU, 2014, 558 p. (In Russ.).
2. Spirin N.A., Gileva L.Y., Lavrov V.V. Information modeling system for blast furnace control. In: *IOP Conference Series: Materials Science and Engineering. Proc. 19th International Scientific Conference on Metallurgy: Technologies, Innovation, Quality, Metallurgy 2015*. Novokuznetsk, 2016, vol. 150, issue l.
3. Spirin N.A., Lavrov V.V., Kosachenko I.E., Onorin O.P., Istomin A.S., Burykin A.A., Shchipanov K.A. Software for the raw-materials management system in blast-furnace smelting. *Metallurgist*. 2015, vol. 59, Issue 1, pp. 104–112.
4. Lavrov V.V., Spirin N.A., Burykin A.A., Shchipanov K.A., Rybolovlev V.Y. Automatic control software for the blast-furnace shop. *Steel in Translation*, 2015, vol. 45, Issue 9, pp. 669–673.
5. Lavrov V.V., Spirin, N.A. Automated information system for analysis and prediction of production situations in blast furnace plant. In: *IOP Conference Series: Materials Science and Engineering. Proc. 19th International Scientific Conference on Metallurgy: Technologies, Innovation, Quality, Metallurgy 2015*. Novokuznetsk, 2016, vol. 150, Issue. l.
6. Ketkov Yu.L. *MATLAB 7: programmirovanie, chislennye metody* [MATLAB 7: programming, numerical methods]. St. Petersburg: BKhV-Peterburg, 2005, 752 p. (In Russ.).
7. Vyatkin V. Software engineering in industrial automation: state-of-the-art review. *IEEE Transactions on Industrial Informatics*. 2013, vol. 9, no. 3, pp. 1234–1249.
8. Makdonal'd M. *WPF: Windows Presentation Foundation v .NET 4.0 s primerami na C# 2010 dlya professionalov* [WPF: Windows Presentation Foundation in .NET 4.0 with examples on C # 2010 for professionals]. Moscow: Vil'yams, 2011, 1024 p. (In Russ.).
9. MacDonald Matthew. *Pro WPF in C# 2010: Windows Presentation Foundation with .NET 4.0*. Apress, 2010, 1181 p. (Russ.ed.: MacDonald M. *WPF: Windows Presentation Foundation v .NET 4.0 s primerami na C# 2010 dlya professionalov*. Moscow: Vil'yams, 2011, 1024 p.).
10. Dimitrov B.H., Nenov H.B., Marinov A.S. Comparative analysis between methodologies and their software realizations applied to modeling and simulation of industrial thermal processes. In: *Proc. 36th International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2013, Opatija; Croatia; 20-24 May 2013.* Article number 6596383, pp. 891–895.
11. Amalfitano D., Fasolino A.R., Maggio V., Tramontana P., Di Mare G., Ferrara F., Scala S. Migrating legacy spreadsheets-based systems to Web MVC architecture: An industrial case study. In: *Proc. Software Evolution Week – IEEE Conference on Software Maintenance, Reengineering, and Reverse Engineering 2014, CSMR-WCRE 2014. Antwerp, Belgium. 3-6 February 2014*. Article number 6747201, pp. 387–390.
12. Zhang B., Bao L., Zhou R., Hu S., Chen P. A black-box strategy to migrate GUI-based legacy systems to web services. SOSE 2008. In: *Proc. of the 4th IEEE International Symposium on Service-Oriented System Engineering, Jhongli, Taiwan. 18-19 December 2008*. Article number 4730458, pp. 25–31.
13. Canfora G. Software evolution in the era of software services. In: *Proc. 7th International Workshop on Principles of Software Evolution, IWPSE 2004, Kyoto, Japan. 6-7 September*, pp. 9–18.
14. Zdun U. Reengineering to the Web: A reference architecture. In: *Proc. of the European Conference on Software Maintenance and Reengineering, CSMR 2002, Budapest, Hungary, 11-13 March 2002*. Article number 995801, pp. 164–173.
15. Tilley S., Gerdes J., Hamilton T., Huang S., Müller H., Wong K. Adoption challenges in migrating to web services. In: *Proc. 4th International Workshop on Web Site Evolution, WSE 2002, Montreal, Canada, 2 October 2002*. Article number 1134086, pp. 21–29.
16. Sreedevi E., Prasanth Y. Critical review on web services and legacy systems - And its migration. *International Journal of Applied Engineering Research*. 2015, vol. 10, Issue 7, pp. 16321–16329.
17. Kastner T. Concept to migrate an access based environmental information system into a Web-based front-end solution for the environmental management of a waste disposal company. In: *Proc. 3rd International ICSC Symposium on Information Technologies in Environmental Engineering, ITEE 2007, Oldenburg, Germany, 29-30 March, 2007*, pp. 559–566.
18. Khalaf R., Keller A., Leymann F. Business processes for Web Services: Principles and applications. *IBM Systems Journal*. 2006, vol. 45, Issue 2, pp. 425–446.
19. Patel K., Ragha L. Survey and Analysis on Migration of Legacy System to Service Oriented Architecture. *IOSR Journal of Computer Engineering*. 2013, vol 9, Issue 2, pp. 32–35.
20. Stroulia E., El-Ramly M., Sorenson P. From legacy to web through interaction modeling. In: *Proc. IEEE International Conference on Software Maintenance 2002, Montreal, Canada, 3-6 October, 2002*, pp. 320–329.
21. Mijan J.L., Garrigos I., Firmenich S. Supporting personalization in legacy web sites through client-side adaptation. In: *Proc. 16th International Conference on Web Engineering, ICWE 2016, Lugano, Switzerland, 6-9 June, 2016*, pp. 588–592.
22. Mao C.-Y., Lu Y.-S. A method for measuring the structure complexity of Web application. *Wuhan University Journal of Natural Sciences*. 2006, vol. 11, Issue 1, pp. 143–150.
23. Rossi G., Schwabe D. Object-oriented design structures in Web application models. *Annals of Software Engineering*. 2002, vol. 13, Issue 1-4, pp. 97–110.

Information about the authors:

I.A. Gurin**,** *Postgraduate of the Chair "Thermophysics and Informatics in metallurgy"* (ivan.gurin@urfu.ru)
V.V. Lavrov**,** *Dr. Sci. (Eng.), Professor of the Chair "Thermophysics and Informatics in metallurgy"* (v.v.lavrov@urfu.ru)
N.A. Spirin**,** *Dr. Sci. (Eng.), Professor, Head of the Chair "Thermophysics and Informatics in metallurgy"* (n.a.spirin@urfu.ru)
A.G. Nikitin**,** *Dr. Sci. (Eng.), Assist. Professor, Director of the Institute of Mechanical Engineering and Transport, Head of the Chair of Mechanics and Machine Engineering* (dir.imit@mail.ru)

Received June 14, 2017