

УПРАВЛЕНИЕ ИНТЕГРАЦИЕЙ, СИНХРОНИЗАЦИЕЙ И ОБРАБОТКОЙ ДАННЫХ В АДМИНИСТРАТИВНЫХ УЧЕТНО-АНАЛИТИЧЕСКИХ СИСТЕМАХ

Е. В. Богомолова, Д. А. Ваганов, Я. В. Коновалова, Ю. В. Ланских, А. В. Малышева, Н. О. Мокрецова, Н. А. Черепанов, Н. А. Шмакова

Вятский государственный университет

Поступила в редакцию 27.05.2016 г.

Аннотация. Рассматривается подход к решению задачи управления информационными процессами и потоками в учетно-аналитических информационных системах. Указывается на штатность практики построения информационных систем на основе унаследованных информационных систем. Формулируются общие принципы, на которых может быть основана интеграция гетерогенных информационных систем. Рассматривается структурная основа реализации интегрированных информационных систем. Приводятся примеры полученных авторами алгоритмов и структур информационных систем и рассматриваются в свете теории многоагентных систем.

Ключевые слова: слова: информационно-аналитическая система, многоагентные системы, объект управления, система электронного документооборота, микропрограммное устройство управления, UML.

Annotation. The approach to solving the problem of information processes and flows management in accounting and analytical information systems. Indicated on commonness of practice of creation of information systems based on legacy information systems. The general principles on which can be based integration of heterogeneous information systems is formulated. The structural basis for the implementation of integrated information systems is considered. The authors obtained examples of algorithms and information systems structures are considered in the light of the multi-agent systems theory.

Keywords: information-analytical system, multi-agent systems, controlling object, electronic document management system, the microprogram control unit, UML.

ВВЕДЕНИЕ

Существенный вклад в автоматизацию учетно-аналитической составляющей административной деятельности предприятия вносит решение задач интеграции и диспетчеризации процессов обработки и передачи информации. Независимо от того, используются для автоматизации участков учетной или аналитической деятельности приобретаемые программные компоненты или унаследованные системы собственной разработки, с максимальной точностью отражающие специфику бизнес-процессов предприятия, а так-

же от того, насколько задачи автоматизации управления передачей и обработкой данных поддерживаются стандартными средствами, принципиальное значение имеет теоретическое осмысление принципов построения платформенного программного обеспечения, на котором основывается интеграция предметно-ориентированных программных компонент и управление ими.

Управление передачей и обработкой данных может быть организовано самыми различными способами и средствами и во многих случаях является уникальной разработкой ИТ-специалиста предприятия. Для построения системного подхода к решению указанных задач необходимо разработать:

– формальную основу описания интеграционных компонент информационной системы;

© Богомолова Е. В., Ваганов Д. А., Коновалова Я. В., Ланских Ю. В., Малышева А. В., Мокрецова Н. О., Черепанов Н. А., Шмакова Н. А., 2016

– общую структуру платформенного программного обеспечения управляющего и интеграционного назначения, в частности, необходимые структуры данных для описания управляемых процессов и основные функциональные модули программного обеспечения.

Дальнейшему рассмотрению подлежит формализация выбора варианта решения задачи системной интеграции из множества возможных приобретаемых и разрабатываемых решений.

ПОСТАНОВКА ЗАДАЧИ

В качестве основы для построения главных производственных систем предприятий могут быть использованы системы электронного документооборота (СЭД) [1, 2]. Набор средств интеграции, реализуемый в результате в рамках СЭД, обозначают как «СЭД-шина».

Как показывает практика использования СЭД для решения интеграционных задач, СЭД способны обеспечить полноценную интеграцию различных информационных систем предприятия между собой и с внешними системами, построение сложных маршрутов прохождения информационных и управляющих потоков, автоматизацию обработки информации.

Авторами уже рассматривалась формализация задачи синхронизации справочной информации на примере задачи автоматизации учета транспорта на предприятии связи [3].

Множество описаний бизнес-процессов будем считать информационной моделью предметной области:

$$OF = \{BP_i\}.$$

AS-IS-модель (модель информационной системы на момент постановки задачи автоматизации) может быть представлена в виде совокупности реальных бизнес-процессов, поддерживаемых системой и отличных от идеализированных, использованных выше:

$$IS' = \{BP_i'\}.$$

TO-BE-модель (модель требуемой системы) также следует представить как совокупность бизнес-процессов:

$$IS'' = \{BP_i''\}.$$

Каждый бизнес-процесс BP_i'' является тождественным или нетождественным исходному BP_i' в зависимости от того, подвергается ли он изменению в ходе решения задачи автоматизации.

Каждая подсистема информационной системы автоматизирует некоторое подмножество множества автоматизируемых бизнес-процессов:

$$ISS_j'' = \{BP_i^j''\}.$$

В оптимальном случае функции подсистем не пересекаются, позволяя при этом полностью покрыть все функциональные требования к внедряемой системе:

$$\bigcap_j BP_i^j'' = \emptyset$$

$$\bigcup_j BP_i^j'' \equiv \{BP_i\}$$

В реальности формирование информационной системы путем последовательного подбора компонентов приводит к тому, что два последних выражения могут не выполняться.

Возможны ситуации и взаимной избыточности внедряемых компонентов и частичной избыточности и недостаточности их. Наиболее общим (и наиболее часто встречающимся) случаем является следующий:

$$\left\{ \begin{array}{l} \bigcup_j BP_i^j'' \setminus \{BP_i\} \neq \emptyset \\ \{BP_i\} \setminus \bigcup_j BP_i^j'' \neq \emptyset \end{array} \right\}'$$

когда внедряемая совокупность подсистем не автоматизирует полностью все бизнес-процессы предприятия, но имеются функции внедряемых подсистем, не являющиеся необходимыми с точки зрения информационных потребностей предприятия.

Итогом деятельности по автоматизации должно стать формирование информационной системы, для которой выполнялось бы условие:

$$\bigcup_j BP_i^j'' \cup \bigcup_k BP_i^k'' \supseteq \{BP_i\}.$$

Задачи выбора приобретаемых компонентов и проектирования разрабатываемых компонентов сводятся фактически к задаче многокритериальной оптимизации, в которой в качестве переменных необходимо учитывать,

в том числе, и такие сложноформализуемые величины, как степень риска внедрения непопулярных программных систем, удобство работы с той или иной подсистемой и т. п.

Для автоматизации управления процессами синхронизации, обмена данными и обработки данных может использоваться технологическая платформа, сформированная в соответствии с предложенной авторами методикой синтеза модуля управления (МУ) информационно-аналитической системой (ИАС) [4, 5]. В основу методики легли методология синтеза микропрограммных устройств управления процессорами и декомпозиция выполняемых системой функций и, как следствие, реализующих их модулей на функции хранения, обработки, передачи данных и управления [6].

МУ формируется в соответствии с известной структурой микропрограммного устройства управления процессора [7] (рис. 1). Не останавливаясь на терминологических аналогиях, отметим: команда, поступающая в МУ, выполняется путем отработки метакоманды, представляющей собой совокупность метакоманд, состоящих из субопераций. В команде, поступающей в МУ, содержится код операции КОП. Контроллер последовательности метакоманд КПКМ выбирает метакоманду из памяти метакоманд ПМК и помещает ее в регистр метакоманд РМК. В метакоманду входят: адрес следующей метакоманды АСМК; код субоперации КСО, из которого формируется управляющее слово УС, подаваемое на модуль обработки; код признака КНР.

ПМК представляет собой базу метаданных. ПМК обладает свойствами ассоциативности (то есть возможности обращения к памяти по значениям различных реквизитов) и иерархичности структуры. Формат записи ПМК:

$$\text{MetaCmd} = \langle \text{MCCode}, \text{NextMcCode}, \text{SOCode}, \{ \langle \text{ParName}, \text{ParDef} \rangle \} \rangle,$$

где *MCCode* – код метакоманды; *NextMCCode* – код следующей метакоманды; *SOCode* – код субоперации; *ParName* – наименование параметра субоперации; *ParDef* – описание параметра субоперации.

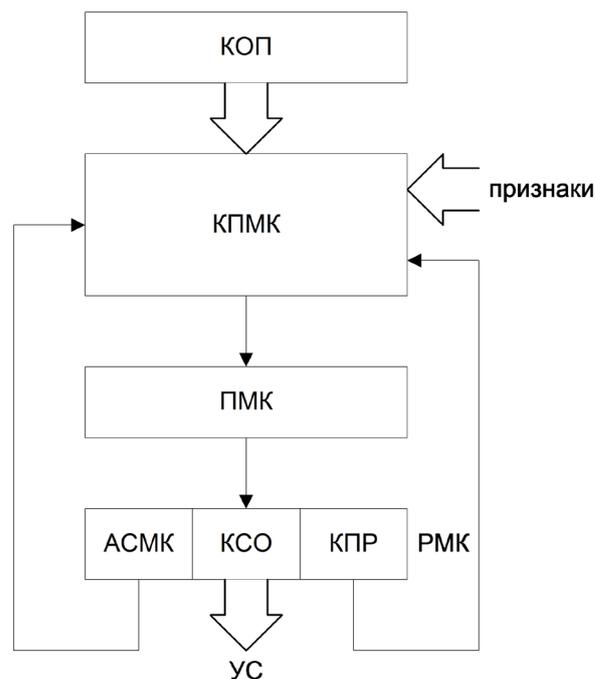


Рис. 1. Структура МУ ИАС

Код субоперации в существующей реализации включает имена исполнимых модулей обработки и визуализации результатов. В описание параметров субоперации включаются тип параметра, минимальное и максимальное его значения, шаг изменения и т. п.

В памяти инструкций (ПИ), которая обрабатывает, как и ПМК, свойствами ассоциативности и иерархичности структуры, хранятся код и параметры операций. Формат записи ПИ:

$$\text{Op} = \langle \text{OpCode}, \{ \langle \text{MCCode}, \{ \langle \text{ParName}, \text{ParValue} \rangle \} \rangle \rangle \rangle,$$

где *OpCode* – код операции; *MCCode* – код метакоманды; *ParName* – наименование параметра субоперации; *ParValue* – значение параметра субоперации.

Работа контроллеров памяти инструкций и метакоманд в процессе выборки инструкций из памяти инструкций и их обработки показана на рис. 2.

Для автоматизации обработки данных необходимо автоматическое формирование памяти инструкций модулем планирования обработок (планировщиком). Работа планировщика основывается на:

- имеющемуся наборе операций;
- доступных диапазонах параметров для каждой субоперации;

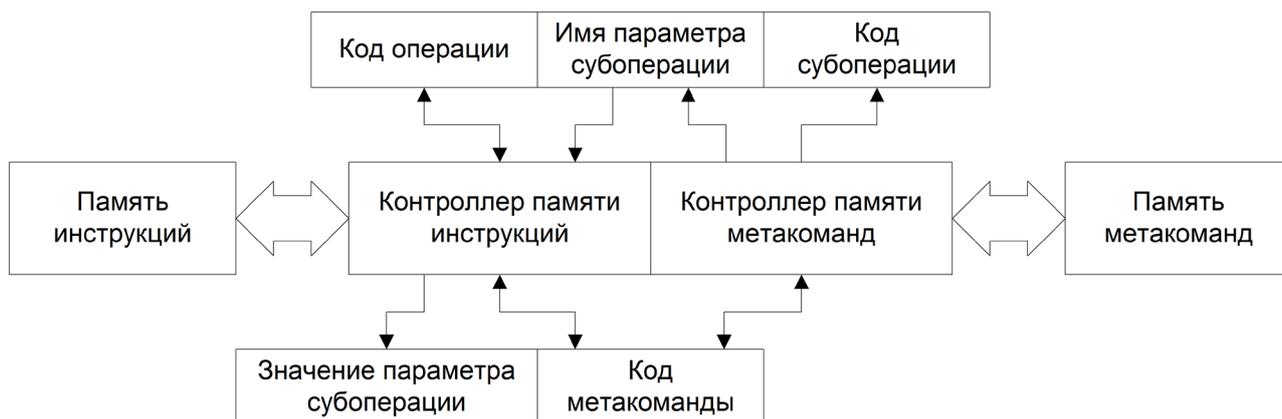


Рис. 2. Функционирование контроллеров памяти инструкций и метакоманд

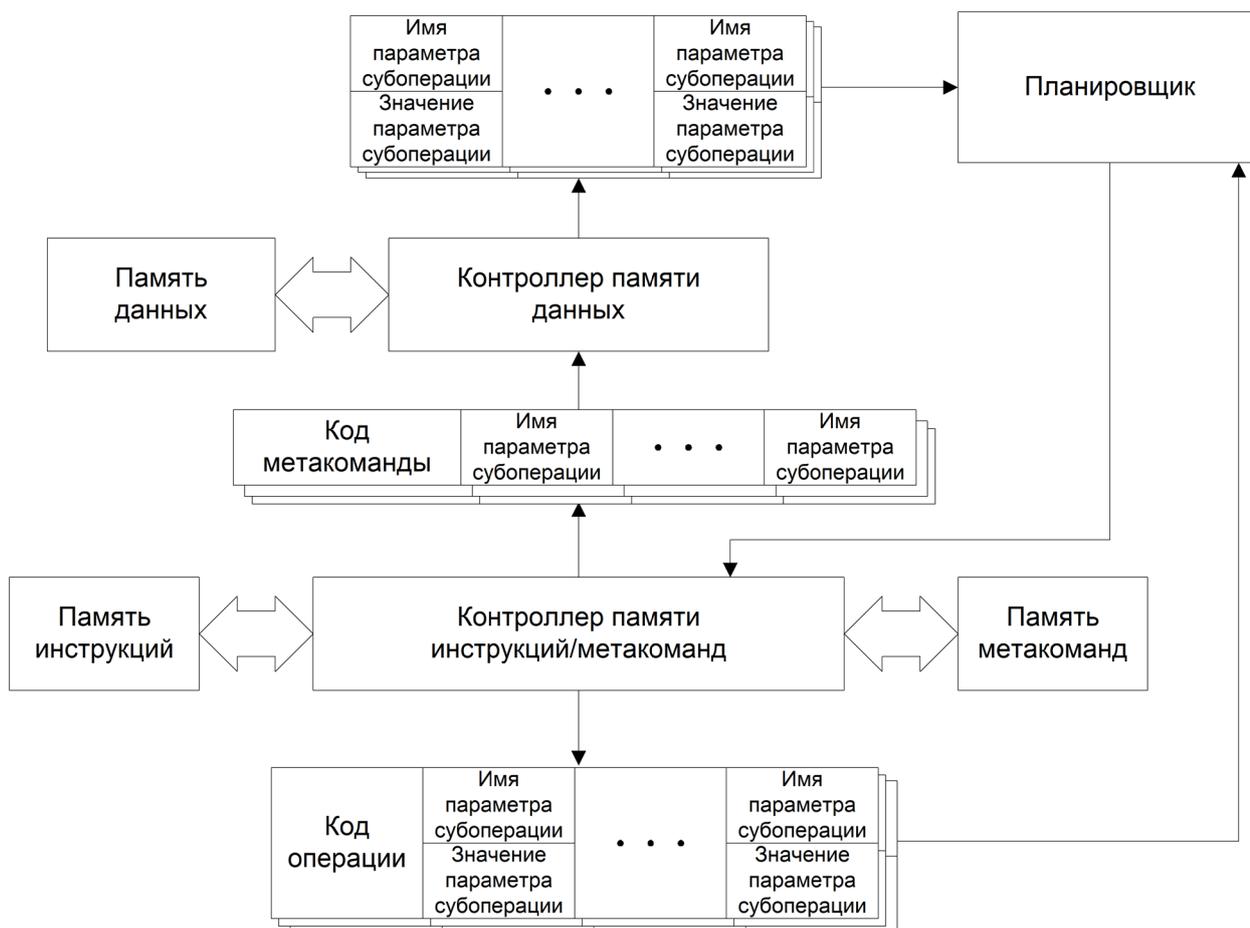


Рис. 3. Функционирование контроллера памяти инструкций/метакоманд, контроллера памяти данных и планировщика

- обработанных данных, в частности метакомандах и параметрах субопераций, в результате выполнения которых были получены имеющиеся информационные единицы;
- запланированных (находящихся в памяти инструкций) обработках.

Отсюда видно, что память данных (ПД), хранящая результаты анализа данных, также должна обладать свойствами ассоциативно-

сти и иерархичности. Формат записи памяти данных:

$$Data = \langle MCCode, \{ \langle \langle ParName, ParValue \rangle \}, DataBlock \rangle \rangle,$$

где *MCCode* – код метакоманды; *ParName* – наименование параметра субоперации; *ParValue* – значение параметра субоперации; *DataBlock* – блок данных.

Функционирование совмещенного контроллера памяти инструкций/метакоманд, контроллера памяти данных и планировщика в процессе планирования инструкций и занесения их в память инструкций представлено на рис. 3.

Кроме того, в исполнительную часть модуля управления входят диспетчер обработок, выполняющий запуск обрабатываемых процедур по расписанию, и менеджер визуализации, выполняющий запуск процедур визуализации в соответствии с правилами их запуска и в зависимости от имеющихся данных.

Авторами рассматривались ряд практических аспектов организации запуска обработчиков, таких, например, как поддержка взаимных зависимостей обработчиков, позволяющих выстраивать последовательности их запуска. Подобные зависимости и ограничения поддерживаются наличием соответствующих структурных элементов в описании инструкций. Очевидно, однако, что расширение логики правил планирования и запуска обработчиков путем расширения их описаний неограниченно. Комплексный характер обработки данных, являющийся следствием сложности предметных областей, приводит к необходимости интеллектуального диагностирования их состояния, в частности, возможности планирования и запуска, ожидаемого времени выполнения.

Теоретическим базисом интеллектуализации обеспечения работы с обработчиками может стать теория самоорганизации, в рамках которой объект управления приобретает «субъективность», т. е. способность самостоятельно оценивать эффективность своего поведения и частично его формировать.

Самоорганизация – это вид действий элементов искусственных систем, выражающийся в стихийном упорядочении действий, направленных на обеспечение собственных интересов, согласуемых в определенных пределах с интересами субъекта управления, зависящих от обстановки и образа действий субъекта управления [8].

Реализация элементов самоорганизации приводит к концепции построения систем, называемых многоагентными (мультиагент-

ными). Прообразом таких систем являются коллективы специалистов, решающих многоаспектные интеллектуальные проблемы. Работой над отдельными аспектами интеллектуальной задачи в мультиагентной системе (МАС) занимаются программные компоненты, называемые агентами, каждый из которых является не пассивным исполнителем запросов, а активным мотивированным субъектом, способным расставлять приоритеты при планировании решения задач, руководствуясь собственными предыдущими обещаниями, собственными намерениями и желаниями в отношении множества имеющихся задач, а также способностью к социальной мотивации и компромиссам. В соответствии с теорией МАС агент способен не только принять задачу или установить приоритет ее выполнения, но и обоснованно отказаться от задачи [9, 10]. При этом агент может быть реализован как СОМ-сервис, Java-класс, CORBA-объект и т. п.

Авторами уже рассматривалось построение многопоточковых обработчиков, позволяющих повысить суммарную производительность однотипной обработки крупных массивов данных на многоядерных архитектурах [4]. Рассмотренный подход подразумевал создание потоков до тех пор, пока создание очередного потока позволяет повысить суммарную производительность. При этом обработчик доходил до максимально возможного уровня использования ресурсов и позволял получить максимально возможную производительность.

В качестве альтернативы стратегии индивидуального соперничества между агентами, реализованной в описанном подходе, следует рассматривать подходы к формированию кооперативного поведения агентов, поскольку построенные указанным образом обработчики могут оказаться неспособны к коллективному решению задач, являющихся суперпозициями индивидуальных заданий, в условиях ограниченных ресурсов, в частности, временных.

Как ранее указывалось коллегами [11], для реализации кооперативного поведения в алгоритме должны быть реализованы:

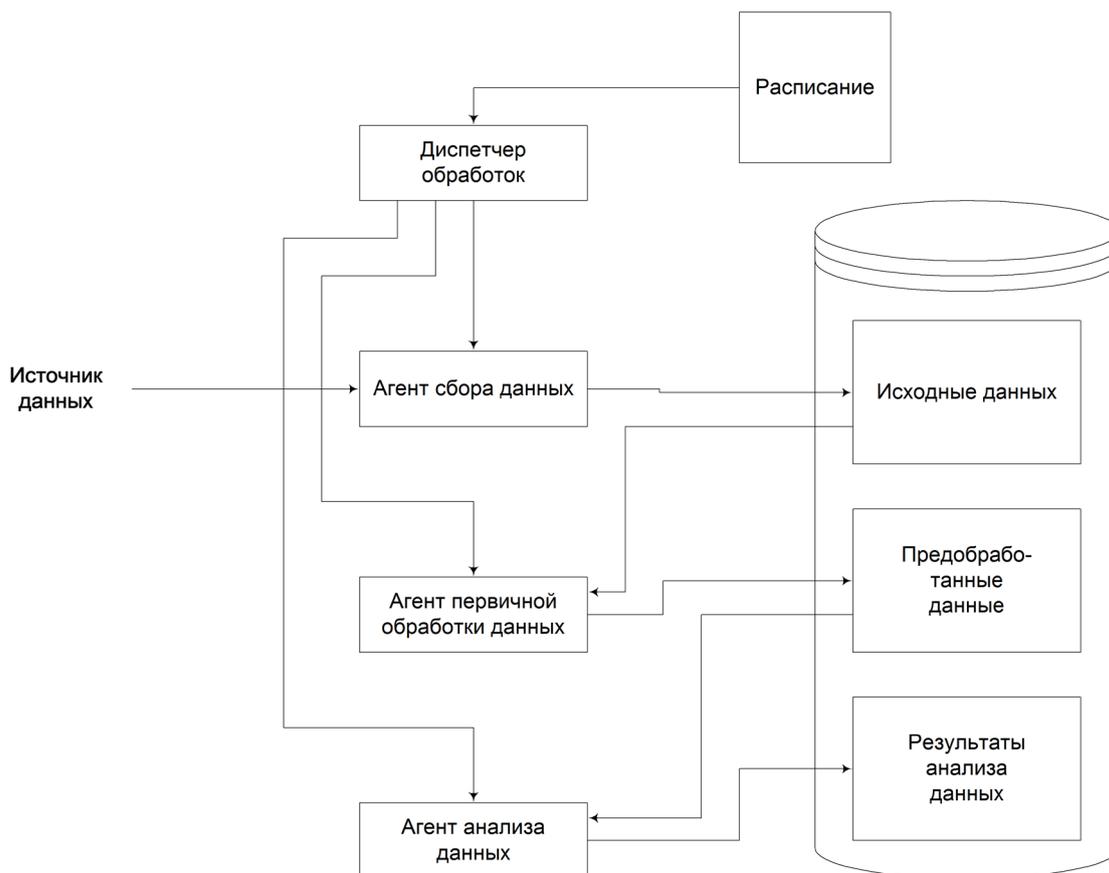


Рис. 4. Пример структуры информационно-аналитической системы с многоэтапной обработкой данных

- анализ соответствия актуального статуса решаемой задачи и обязательств агента;
- выдача сигнала диспетчеру обработок (называемому также «местом встречи агентов» – AMP – Agent Meeting Place) об опасности невыполнения обязательств;
- обработка сигнала от AMP о необходимости освобождения ресурсов в интересах других агентов.

В результате авторы [11] приходят к адаптивному принципу кооперативно-конкурентного использования ресурсов, где диспетчер обработок руководствуется общим планом запуска обработок и некоторыми правилами запуска, а распределение ресурсов производится оперативно в зависимости от запросов обработчиков к диспетчеру.

На рис. 4 приведен пример системы, в которой реализован рассматриваемый подход, где решение некоторой предметной задачи осуществляется в три этапа: занесение данных из источника в базу данных, первичная обработка данных с формированием исход-

ных данных для решения задачи анализа данных, и, наконец, основная аналитическая задача. Можно обозначить классы задач, где данные каждой из упомянутых разновидностей представляют собой наборы однотипных данных, которые могут обрабатываться последовательно. Поэтому представляется неоптимальным строго последовательный запуск агентов сбора данных, первичной обработки и анализа данных. А при их параллельной работе необходимо обеспечивать наличие данных, являющихся исходными для каждого последующего этапа обработки. Это означает, что в отсутствие инструментов адаптивного управления скоростью управления агентов возможно возникновение явлений недообеспеченности того или иного агента исходными данными или ресурсами для обработки.

Авторами была рассмотрена также задача автоматизации обработки данных в системах мониторинга систем связи [12]. Решение этой задачи также представляет собой совокуп-



Рис. 5. Структурная схема программной платформы обработки данных

ность двух подходов: использование метаописаний данных и технологии многоагентной обработки. Структура рассмотренной системы приведена на рис. 5.

ЗАКЛЮЧЕНИЕ

Таким образом, сформулирован тезис о возможности использования систем управления документооборотом в качестве основы функциональности синхронизации и интеграции модулей и подсистем информационной системы предприятия, рассмотрена формально-теоретическая основа построения интеграционных платформ систем административно-аналитического назначения.

В качестве теоретического базиса интеллектуализации управления обработкой и передачей данных в информационной системе рассматривается теория самоорганизации и концепция многоагентных систем.

Для реализации кооперативного поведения программных агентов предложен адаптивный принцип кооперативно-конкурентного использования ресурсов, заключающийся в распределении ресурсов под управлением диспетчера по запросам программных агентов. В условиях использования технического обеспечения и системного программного обеспечения информационной системы для решения множества взаимоисключающих задач получение результатов может

сопровождаться превышением ресурсных ограничений, поэтому именно адаптивный подход позволит реализовать наиболее гибкую стратегию управления, позволяющую оперативно реагировать на актуальную ситуацию ресурсной обеспеченности и оптимизировать процесс получения решения задач.

СПИСОК ЛИТЕРАТУРЫ

1. Мамышева Е. Мы уже давно вышли за рамки традиционных СЭД / Е. Мамышева // PC WEEK Russian edition. – 2012. – № 12. – Режим доступа: <http://www.pcweek.ru>.

2. Богомолова Е. В., Ланских Ю. В. Практические решения по использованию системы электронного документооборота в информационной системе предприятия // Общество, наука, инновации. (НПК-2014). Всерос. ежегод. науч.-практ. конф. : сб. материалов, 15–26 апреля 2014 г. / Вят. гос. ун-т. – Киров, 2014. (Факультет автоматики и вычислительной техники (ФАВТ). Секция «Вопросы проектирования систем обработки и представления данных». Статья № 8).

3. Ланских Ю. В., Богомолова Е. В., Татарина Е. О. Решение задач системной интеграции при автоматизации транспортного учета в региональном предприятии связи // Регионы в условиях неустойчивого развития : материалы международной научн. конф. «Регионы в условиях неустойчивого развития» (Шарья – Кострома, 1–3 ноября 2012 г.) : в 2 т. – Т. 2 / сост. А. М. Базанков, И. Г. Криницын, А. П. Липаев. – Кострома : КГУ им. Н. А. Некрасова, 2012. – 226 с. – с. 188-193

4. Ланских Ю. В. Принципы организации технологической платформы и процедур обработки данных в информационно-аналитической системе // Вестник ИТАРК – Сыктывкар : ИТ-Ассоциация Республики Коми, 2012. – 68 с. (С. 24–28).

5. Ланских Ю. В., Нечаев Д. С. Организация управления в системах интеллектуального анализа данных // Вестник Вятского научно-го центра Верхне-Волжского отделения АТН

РФ. Серия: Проблемы обработки информации. Выпуск 1(5)/2004. – Киров : Вятский научный центр, 2004. – 149 с. (С. 144–149).

6. Балашов Е. П., Пузанков Д. В. Проектирование информационно-управляющих систем. – М. : Радио и связь, 1987. – 256 с.

7. Балашов Е. П. и др. Микро- и мини-ЭВМ: Учебное пособие для вузов. – Л. : Энергоатомиздат. Ленингр. отд-ние, 1984. – 376 с.

8. Лавреш И. И. Самоорганизация объектов управления и меры согласования интересов субъекта и объекта управления // Автоматизация и современные технологии: межотраслевой научно-технический журнал / Министерство образования Российской Федерации. – М. : Машиностроение, 2011. – № 3.– 48 с. (С. 36–41)

9. Тарасов В. Б. Агенты, многоагентные системы, виртуальные сообщества: стратегическое направление в информатике и искусственном интеллекте // Новости искусственного интеллекта : Сб. – 1998. – № 2. – С. 5–63.

10. Shoham Yoav. Agent-oriented programming (англ.) // Artificial Intelligence. – 1993. – № 60(1). – С. 51–92.

11. Бармин И. Н. Принципы интеллектуализации объекта управления в информационно-аналитической системе / И. Н. Бармин, В. В. Куклин, С. А. Куклин, В. Г. Ланских, Ю. В. Ланских, В. Н. Поздин, В. И. Семеновых // Вестник ВГУ, Серия: Системный анализ и информационные технологии. – 2015 – № 3. – С. 156–161.

12. Ваганов Д. А., Коновалова Я. В. Разработка универсальной программной платформы автоматизации мониторинга систем связи // Всероссийская ежегодная научно-практическая конференция «Общество, наука, инновации» (НПК-2013). Общеуниверситетская секция, БФ, ГФ, ФЭМ, ФАВТ, ФАМ, ФПМТ, ФСА, ХФ, ЭТФ: 15-26 апр. 2013 г. : сб. материалов / Вят. гос. ун-т; отв. ред. С. Г. Литвиненц. – Киров, 2013. – (Факультет автоматики и вычислительной техники. Секция «Вопросы проектирования систем», ст. 20).

*Е. В. Богомолова, Д. А. Ваганов, Я. В. Коновалова, Ю. В. Ланских, А. В. Малышева,
Н. О. Мокрецова, Н. А. Черепанов, Н. А. Шмакова*

Богомолова Е. В. – аспирант кафедры автоматике и телемеханики, факультет автоматике и вычислительной техники, Вятский государственный университет.
E-mail: elenaoops@mail.ru

Bogomolova E. V. – postgraduate, Department of Automation and Telemechanics, Faculty of Automation and Computing Machines, Vyatka State University.
E-mail: elenaoops@mail.ru

Ваганов Д. А. – магистрант кафедры автоматике и телемеханики, факультет автоматике и вычислительной техники, Вятский государственный университет.
E-mail: shi-ni-gami@hotmail.com

Vaganov D. A. – undergraduate, Department of Automation and Telemechanics, Faculty of Automation and Computing Machines, Vyatka State University.
E-mail: shi-ni-gami@hotmail.com

Коновалова Я. В. – магистрант кафедры автоматике и телемеханики, факультет автоматике и вычислительной техники, Вятский государственный университет.
E-mail: jana-yana92@mail.ru

Konovalova Ya. V. – undergraduate, Department of Automation and Telemechanics, Faculty of Automation and Computing Machines, Vyatka State University.
E-mail: jana-yana92@mail.ru

Ланских Ю. В. – канд. техн. наук, доцент кафедры автоматике и телемеханики, факультет автоматике и вычислительной техники, Вятский государственный университет.
E-mail: lyuv@inbox.ru

Lanskikh Yu. V. – PhD in Technique, Associate Professor, Department of Automation and Telemechanics, Faculty of Automation and Computing Machines, Vyatka State University.
E-mail: lyuv@inbox.ru

Малышева А. В. – программист, ООО «Акселот-М».
E-mail: annyml@mail.ru

Malysheva A. V. – programmer, Axelot-M Ltd.
E-mail: annyml@mail.ru

Мокрецова Н. О. – магистрант кафедры автоматике и телемеханики, факультет автоматике и вычислительной техники, Вятский государственный университет.
E-mail: natycenbka@yandex.ru

Mokretsova N. O. – undergraduate, Department of Automation and Telemechanics, Faculty of Automation and Computing Machines, Vyatka State University.
E-mail: natycenbka@yandex.ru

Черепанов Н. А. – магистрант кафедры автоматике и телемеханики, факультет автоматике и вычислительной техники, Вятский государственный университет.
E-mail: demond600@mail.ru

Cherepanov N. A. – undergraduate, Department of Automation and Telemechanics, Faculty of Automation and Computing Machines, Vyatka State University.
E-mail: demond600@mail.ru

Шмакова Н. А. – старший преподаватель кафедры автоматике и телемеханики, факультет автоматике и вычислительной техники, Вятский государственный университет.
E-mail: shmakova.natalya@mail.ru

Shmakova N. A. – Senior Lecturer, Department of Automation and Telemechanics, Faculty of Automation and Computing Machines, Vyatka State University.
E-mail: shmakova.natalya@mail.ru