

АЛГОРИТМИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ИНТЕГРАЦИИ ИНФОРМАЦИОННЫХ СИСТЕМ

Манвелидзе А.Б.¹, Ковшов Е.Е.¹

¹ ФГБОУ ВПО Московский государственный технологический университет «СТАНКИН», Москва, Россия (127055, г. Москва, Вадковский пер., д. 3А), e-mail: e.kovshov@stankin.ru

Рассматриваются особенности, модели и алгоритмы интеграции разнородных совместимых (интероперабельных) информационных сред в составе информационных систем. Освещаются основные проблемы, связанные с интеграцией существующих программных продуктов в рамках действующих информационных систем. Применение распределенной архитектуры для решения задачи интеграции и разработки информационных систем, как жесткой, так и гибкой конфигурации позволяет добиться повышения скорости интеграции, а также снизить накладные расходы и совокупную стоимость владения информационным продуктом. Предложенный подход обеспечивает совместимость информационных сред за счет унификации интерфейсов взаимодействия модулей, опытным путем получить показатели эффективности разработки и интеграции систем, найти оптимальный путь интеграции существующих информационных систем, а также разработки новых. Кроме того, предложенный подход позволяет упростить процесс интеграции и разработки информационных сред, а также повысить надежность системы в целом за счет инкапсуляции возможной ошибки внутри подсистем низших уровней и обеспечить быстрый и не требующий затрат способ перевода полученных программных решений на различные программно-аппаратные платформы.

Ключевые слова: информационные системы и среды, программный модуль, интеграция, функциональная совместимость

ALGORITHMIC SUPPORT OF THE INFORMATION SYSTEMS INTEGRATION

Manvelidze A.B.¹, Kovshov E.E.¹

¹ Moscow State University of Technology "STANKIN", Moscow, Russia (127055, Moscow, Vadkovsky ln., 3A), e-mail: e.kovshov@stankin.ru

It is discussed the features, models and algorithms for the integration of diverse compatible (interoperable) information environments into the information systems. Highlighting the main problems associated with the integration of existing software in the framework of the existing information systems. The use of a distributed architecture for solving the problem of integration and development of information systems, rigid and flexible configuration allows to boost the speed of integration, as well as to reduce overhead and total cost of ownership information product. The proposed approach provides compatibility information environments due to the unification of interfaces for interaction of the modules empirically get performance indicators development and system integration, to find the optimal path for the integration of existing information systems and develop new ones. Besides, the offered approach allows to simplify the process of integration and development of information media, as well as increased reliability by encapsulating the possible error inside the subsystems of the lowest levels and provide a way of moving the received software solutions fast and not demanding expenses for various software and hardware platforms.

Keywords: information systems and environments, program module, integration, functional compatibility

Основой современных информационных технологий являются информационные системы, которые представляют собой организованную и упорядоченную совокупность информационных продуктов и информационной инфраструктуры. Разнообразие существующих информационных систем и сред в различных отраслях промышленности, а также методов их реализации предопределяет разнородность программных продуктов и комплексов. Зачастую различные программные реализации задач автоматизации и обработки информации не имеют унифицированных интерфейсов взаимодействия, что образует проблему их совместного использования для решения комплексных задач. Под

информационной средой понимается совокупность информационных систем, которые позволяют решить комплексные задачи в рамках предприятия. Такие комплексные задачи возникают при расширении потенциала предприятия и охвата новых сфер деятельности.

В настоящее время наблюдаются попытки разработки предприятиями собственных программных решений, направленных на автоматизацию определенных задач, в том числе автоматизацию обучения и повышения квалификации персонала без отрыва от производства. Такие комплексные мультимедийные системы обучения позволяют подготовить квалифицированных специалистов, обладающих не только теоретическими знаниями, но и практическими навыками работы со сложными технологическими объектами [3]. Таких программных решений, как правило, много, и зачастую они не отвечают понятию интероперабельности или функциональной совместимости, которая заключается в способности программного продукта или информационной системы, интерфейсы которых полностью открыты, взаимодействовать и функционировать с другими программными решениями или информационными системами без каких-либо ограничений доступа и реализации.

При организации человеко-машинного взаимодействия с программными сервисами среди прочих мер, помимо тестирования надежности и функциональной совместимости информационных сред, необходимо оценивать качество приложений. Характеристики качества выявляются в ходе исследования человеко-машинного взаимодействия с продуктом, которое может проводиться по одной из существующих методик или включать в себя их комбинацию. К сожалению, такие тестирования либо дорогостоящи, либо сложны и трудоёмки, поэтому в качестве альтернативы предлагается применение инновационной парадигмы исследований, позволяющей автоматизировать этапы сбора, обработки и предварительного анализа экспериментальных данных человеко-машинного взаимодействия [4].

На сегодняшний день разработано множество информационных систем функционирующих на основе распределенных информационных технологий. Одним из наиболее популярных решений для их разработки является применение программных сервисов. Перспективным направлением является использование сервисно-ориентированной архитектуры (COA), в которой основополагающим способом обмена данными между сервисами является использование протокола SOAP (Simple Object Access Protocol) [7].

С точки зрения реализации прикладных информационных систем, COA определяет программное обеспечение в виде сервисов, имеющих слабые связи и определенных с целью совместимости, повышения гибкости и возможности многократного использования, в том числе, при интеграции систем различного уровня посредством языка XML (Extensible

Markup Language), являющимся на сегодняшний день наиболее популярным языком разметки [5]. Модель сервисно-ориентированной архитектуры сосредотачивает внимание на таких аспектах архитектуры, как сервисы и действия. Основная цель этой модели заключается в объяснении отношений между агентами, предоставляющими сервисы и запрашивающими сервисы.

В качестве решения описанных выше научно-практических задач представим прикладную информационную систему как систему, состоящую из совокупности взаимодействующих подсистем, а именно – программных модулей, выполняющих определенный набор задач и обладающих функциональной совместимостью между собой. Каждый программный модуль может рассматриваться как средство борьбы с дублированием, то есть как средство накопления и многократного использования знаний [2]. В свою очередь, каждый модуль обладает унифицированными интерфейсами взаимодействия, с помощью которых без лишних затрат можно эффективно построить новую информационную систему, исходя из ее функционального назначения. Примером является инкапсуляция, где выбираются некоторые подмножества свойств модуля в качестве информации о модуле, доступной разработчикам системы или другим модулям. Иначе говоря, каждый модуль известен всем остальным через соответствующие дескрипторы. Общедоступные свойства модуля являются его интерфейсом, при этом, чем меньше общедоступная часть, тем больше вероятность того, что изменения в модуле будут содержаться в его скрытой части. Данное положение касается изменения скрытых элементов модуля, не затрагивающих его общедоступных свойств [10].

К преимуществам, информационных сред, построенных по такому принципу, относятся:

1. Функциональная полнота – интеграция всех имеющихся автоматизированных процессов в ИС и их функциональная совместимость.
2. Масштабируемость - способность ИС или информационного процесса адекватно справляться с увеличением рабочей нагрузки, увеличивая свою производительность при добавлении соответствующих ресурсов.
3. Гибкость и открытость – тонкая настройка ИС под нужды конечного потребителя путем интеграции программных модулей.

Модель такой модульной архитектуры, позволяющей обеспечить интеграцию разнородных, функционально совместимых информационных сред, представлена на рис. 1.

В такой системе совокупность информационных модулей может быть организована в систему с контейнерами сообщений и хранилищем данных. Контейнер сообщений

представляет собой подсистему, позволяющую хранить сообщения или данные и выдавать их другой компоненте в процессе функционирования системы.

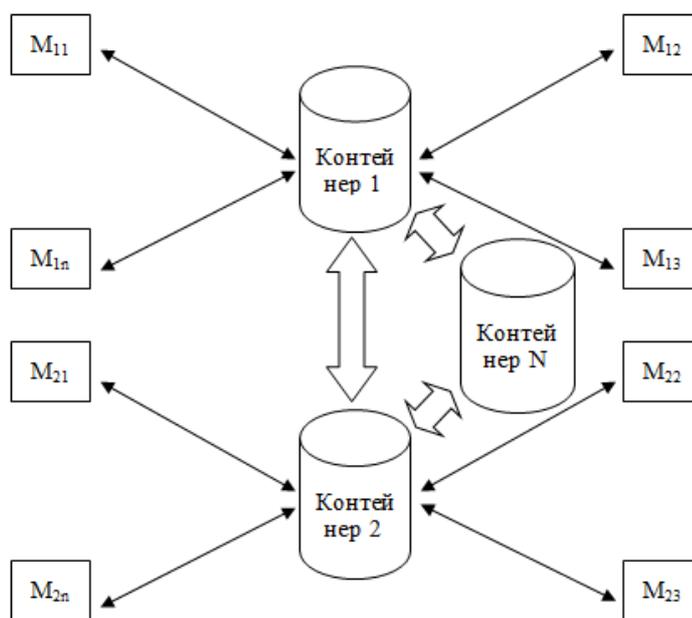


Рис. 1. Модель модульной архитектуры интеграции ИС

Такое сообщение или данные, переданные контейнеру, не будут использовать ресурсы модульной компоненты системы, что позволяет приступить к выполнению очередной программной задачи. Таким образом, данная модульная архитектура отвечает принципу распределенной информационной системы.

Представленная модель архитектуры интеграции позволяет производить распараллеливание процесса обработки данных [8], поскольку модули являются независимыми от архитектуры, программной реализации и аппаратной платформы.

В ряде случаев для обеспечения взаимодействия между модулями может потребоваться создание специализированных программных компонент, называемых адаптерами [9] и обеспечивающих интеграцию данных модулей, иначе говоря, осуществляющих архитектурную функцию системы.

Модульный метод разработки интерактивных программных систем должен удовлетворять четырем основным положениям: 1) декомпозиции; 2) композиции; 3) непрерывности; 4) защищенности.

Из этого можно выделить пять правил обеспечения модульности ИС: 1) прямое отображение; 2) минимизация интерфейсов; 3) слабая связность интерфейсов; 4) наличие явных интерфейсов; 5) инкапсуляция.

Задачу разработки функционально совместимой информационной системы можно определить как задачу реализации искомой системы A на основе совокупностей унифицированных модулей E_1, \dots, E_k . Реализацию A на основе указанной совокупности обозначим как $R(A, E_1, \dots, E_k)$. Как известно, искомую задачу удобнее решать, разбивая ее на

более мелкие подзадачи. Технология разработки программных систем на основе модульного подхода позволяет вести такое разбиение от задачи A к совокупности модулей E_1, \dots, E_k . В соответствии с указанным подходом вначале выделяется и реализуется общая идея системы. При этом каждый раз, делая очередной шаг вниз, формулируются уточнения, реализуя тем самым итеративный процесс создания программного продукта или модуля.

Схематически такой процесс можно представить в виде дерева, что отражает рис. 2.

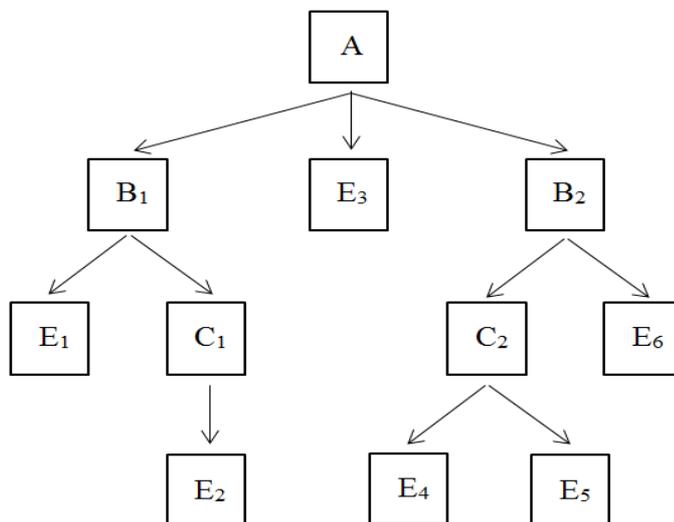


Рис. 2. Модель декомпозиции на основе итеративного процесса

Опираясь на вышеизложенное, необходимо отметить, что модульная структура информационной системы имеет вид графа, в узлах которого размещаются сами модули, а направленные дуги указывают их статическую подчиненность.

Технологическая цепочка последовательности разработки информационной системы приведена на рис. 3. Центральной задачей является декомпозиция или шаг декомпозиции.

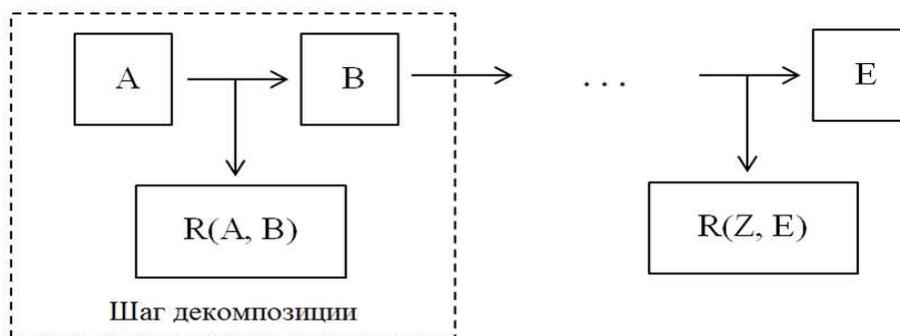


Рис. 3. Цепочка разработки информационной системы на основе модели процесса декомпозиции

Шаг декомпозиции состоит в том, чтобы по имеющимся исходным данным задачи реализации реализовать систему A на базе готового или нового решения B .

В результате очередного шага возникают внешние описания, то есть интерфейсы и реализация $R(A, E_1, \dots, E_k)$. Некоторые из реализованных модулей для системы A могут

являться результатом объединения нижестоящих по уровню иерархии модулей, то есть $B_i \in \{E_1, \dots, E_k\}$.

В ходе реализации новых модулей информационных сред важным фактором является время реализации функционала [6], а также время разработки модулей информационных сред в целом. Во время моделирования и последующей интеграции программной системы в составе информационной среды необходимо оценивать время разработки i -ого программного модуля, трудоемкость разработки системы и других неотъемлемых параметров, при помощи которых можно оценить состояние разработки всей информационной среды и отдельных ее составляющих [9, 10]. Помимо этого, следует осуществлять контроль и перераспределение ресурсов, необходимых для реализации общего функционала информационной системы, между группами разработчиков, тем самым ускорять разработку всей системы в целом.

Существует три способа организации разработки системы [1]:

1. Последовательный способ ($m=1$).
2. Параллельный способ ($m \geq n$);
3. Последовательно-параллельный способ ($n > m > 1$).

Основываясь на совокупности применения данных способов, можно определить общее время разработки системы:

$$T = \begin{cases} \sum_{i=1}^n k_i \cdot t(i), & m = 1 \\ \sum_{i=1}^n k_i \cdot t(i) - \sum_{i=1}^{n-1} t_k(i), & n > m > 1 \\ \max_n \max_{i=1; R_n} \sum_{i=1}^{R_n} (k_i \cdot w_{i_n}), & m \geq n \end{cases}, \quad (1)$$

где m – количество модулей системы, n – количество разработчиков или групп разработчиков, $t(i)$ – функция времени разработки модуля i , $W = \|w_i\|$ – матрица времени разработки i -ого модуля, k_i – коэффициент нормирования, учитывающий сложность разработки i -ого модуля системы, t_k – наименьшая норма времени между каждой i -ой парой смежных операций, R_n – количество модулей n -ого разработчика.

Резюмируя изложенное, следует отметить, что применение распределенной архитектуры для решения задачи интеграции и разработки корпоративных информационных систем, как жесткой, так и гибкой конфигурации позволяет добиться повышения скорости интеграции, а также снизить накладные расходы и совокупную стоимость владения информационным продуктом.

Кроме того, предложенный способ интеграции позволяет обеспечить функциональную совместимость информационных сред за счет унификации интерфейсов

взаимодействия программных модулей, опытным путем получить показатели эффективности разработки и интеграции систем, найти оптимальный путь интеграции существующих информационных систем, а также – эффективные алгоритмы разработки новых.

Процесс декомпозиции в совокупности с кроссплатформенным подходом позволяют упростить процесс интеграции и разработки информационных сред, а также повысить надежность системы в целом за счет инкапсуляции возможной ошибки внутри подсистем низших уровней и обеспечить быстрый и не требующий затрат способ перевода полученных программных решений на различные программно-аппаратные платформы.

Список литературы

1. Андон Ф.И., Коваль Г.И., Коротун Т.М., Сулов В.Ю. Основы инженерии качества программных систем. – К.: Академперіодика, 2002. – 502 с.
2. Батова М.М., Ковшов Е.Е., Смирнов О.С. Разработка информационных систем инновационного промышленного предприятия на основе унифицированного модульного подхода. // Журнал об инновационной деятельности «Инновации». – 2011. – № 05 (151). – С. 102-106.
3. Борисенко Е.В., Ковшов Е.Е. Применение инструментальных средств обработки корпоративной информации на основе программно-аппаратных технологий // Вестник МГТУ «Станкин». – М.: МГТУ «Станкин», 2010. – №3 – С. 123-129.
4. Ковшов Е.Е., Мартынов П.Н., Хуэ Н.Н., Киен Ф.Ч. Автоматизация оценки эффективности взаимодействия конечного пользователя с обучающей информационной системой // Научно-практический журнал «Открытое образование». – 2010. – №1. – С. 37-43.
5. Ковшов Е.Е., Мутин Д.И. Управление гетерогенными данными в корпоративной информационной системе медицинского учреждения путем применения XML-технологий // Мехатроника, автоматизация, управление. – 2008. – №3. – С. 51-54.
6. Ковшов Е.Е., Смирнов О.С. Методы оценки и повышения надёжности информационных сред промышленного предприятия // Динамика сложных систем – XXI век. – 2012. – №2. – С. 112-116.
7. Костров И.А., Ковшов Е.Е. Сервисно-ориентированная архитектура приложений как средство организации распределенных систем в среде слабоструктурированных данных // Вестник МГТУ «Станкин». Научный рецензируемый журнал. – М.: МГТУ «СТАНКИН», №3(22), 2012. – С. 140-145: ил.

8. Митропольский Н.Н., Ковшов Е.Е. Применение обобщённых функций высших порядков для параллельной обработки растровых изображений // Вестник МГТУ «Станкин». Научный рецензируемый журнал. – М.: МГТУ «СТАНКИН», №1, том 1 (18), 2012. – С. 94-98: ил.
9. Смирнов О.С. Конвергенция неоднородных информационных сред на основе кроссплатформенных программных компонент: Автореф. дис. канд. техн. наук. – Москва, 2013. – 23 с.
10. Смирнов О.С. Конвергенция систем поддержки образовательного контента на основе унифицированного модульного принципа построения // Современные проблемы науки и образования. – 2012. – № 6; URL: www.science-education.ru/106-7977 (дата обращения: 10.11.2013).

Рецензенты:

Волков А.Э., д.т.н., профессор, декан факультета МТО ФГБОУ ВПО МГТУ «СТАНКИН», г.Москва.

Климанов В.П., д.т.н., профессор, профессор кафедры информационных систем ФГБОУ ВПО МГТУ «СТАНКИН», г. Москва.