

Т. Г. Пенькова, канд. техн. наук, доц., ст. науч. сотр., e-mail: penkova_t@icm.krasn.ru,
С. Н. Кочетков, программист I кат., e-mail: serega@icm.krasn.ru,
Институт вычислительного моделирования СО РАН, г. Красноярск

Технологический подход к организации унифицированного обмена данными между гетерогенными системами

Предложен технологический подход к организации обмена данными между гетерогенными системами, основанный на формировании унифицированного представления структур данных. Представлена формальная и концептуальная модели унифицированного обмена данными. Предложена архитектура программных средств, обеспечивающая организацию унифицированного обмена данными. Элементы технологического подхода рассматриваются на примере размещения заказа для государственных и муниципальных нужд.

Ключевые слова: унифицированный обмен данными, информационное взаимодействие гетерогенных систем, технологический подход, концептуальная модель, архитектура программных средств

Введение

Современный уровень и масштаб автоматизации различных видов деятельности создают необходимость обеспечения информационного взаимодействия разнородных систем. Системы, функционирующие в организациях, должны иметь возможность обмениваться данными между собой и с внешними ресурсами [1, 2]. Одним из ярких примеров, где информационное взаимодействие систем и обмен данными между гетерогенными ресурсами наиболее актуальны и востребованы, является сфера размещения заказа для государственных и муниципальных нужд. Государственная политика, направленная на открытость данных, а также нормативно-правовая база в сфере закупок обязывают органы власти, реализующие процедуры закупок, и учреждения, осуществляющие закупки за счет бюджетных средств, организовать интеграцию систем поддержки размещения заказа с Официальным общероссийским сайтом закупок и двусторонний информационный обмен с Электронными торговыми площадками [3, 4]. При этом задача организации взаимодействия систем существенно усложняется тем, что объект информационного обмена меняется достаточно часто, и каждая система имеет свой формат хранения данных.

Существуют несколько основных подходов к организации межсистемного взаимодействия. Одним из наиболее простых способов является передача файлов (*File Transfer*), при этом файл выступает как универсальный механизм хранения данных [5]. Однако несмотря на наличие общего формата обмена в процессе создания, передачи или обработки файлов высока вероятность возникновения рассинхронизации данных. Другой подход — использование общей базы данных (*Shared Database*) [6], где все системы работают с актуальной и синхронизированной информацией, однако общая база данных может стать "слабым звеном" интеграционного решения и причиной отказа работы при увеличении числа взаимодействующих систем. Следующий подход — обмен сообщениями (*Messaging*), позволяющий избежать большинства из указанных проблем, но реализация данного решения требует более сложного процесса разработки, тестирования и отладки [7]. Также в последнее время широко применяется сервис-ориентированная архитектура (*SOA, Service-oriented Architecture*), позволяющая использовать веб-сервисы для взаимодействия разнородных информационных систем [8, 9]. Кроме этого, часто задача обмена данными решается путем разработки дополнительных программных модулей, преобразующих данные из одного формата в другой для конкретного объекта информационного обмена [10—13].

Такой подход сопровождается необходимостью создания нового модуля для каждого нового информационного объекта или системы, а также перепрограммирования модуля в случае изменения схемы хранения данных у одной из взаимодействующих систем. Таким образом, возможности современных технологий, преимущества подходов, а также ограничения существующих решений определяют актуальность разработки технологического подхода, обеспечивающего унифицированный обмен данными между гетерогенными системами с учетом вариативности форматов хранения данных и условий функционирования взаимодействующих систем.

В данной работе предложен технологический подход к организации обмена данными между гетерогенными системами, основанный на формировании унифицированного представления структур данных. Представлена формальная и концептуальная модели унифицированного обмена данными, основанные на развитии ранее предложенной модели межсистемного взаимодействия [13, 14]. Предложена архитектура программных средств, обеспечивающая организацию унифицированного обмена данными. Элементы технологического подхода рассматриваются на примере задачи размещения заказа для государственных и муниципальных нужд.

Модель унифицированного обмена данными

Обмен данными между системами можно представить в виде пары:

$$M = (\langle O_{In}, I_{In} \rangle, \langle O_{Out}, I_{Out} \rangle),$$

где I — системы информационного обмена; O — объекты информационного обмена. Каждая система может участвовать в информационном обмене в качестве "отправителя" (I_{in}) или "получателя" (I_{out}). При этом одна и та же система может выступать одновременно в качестве "отправителя" и "получателя", например, когда необходимо перевести данные из одного формата в другой в рамках одной информационной системы. Система информационного обмена характеризуется рядом параметров, описывающих ее особенности: способ хранения данных, параметры доступа к данным, канал связи и др. Объект информационного обмена можно представить как тройку: $O = \langle D, F, S \rangle$, где D — передаваемые или получаемые системой данные, F — формат хранения данных, S — структура, в которой представлены данные. Для системы "отправителя" и системы "получателя" объект информационного

обмена определяется как $O_{In} = \langle D_{In}, F_{In}, S_{In} \rangle$ и $O_{Out} = \langle D_{Out}, F_{Out}, S_{Out} \rangle$ соответственно. Процесс унифицированного обмена данными заключается в преобразовании информационного объекта "отправителя" в информационный объект "получателя" на основе формирования унифицированного представления структур данных и может быть представлен следующей цепочкой:

$$\begin{aligned} O_{in} \langle D_{In}, F_{In}, S_{In} \rangle &\xrightarrow{Q \langle D_{In}, S_{In} \rangle} U_{In} \langle D_{In}, S_{In} \rangle \xrightarrow{G \langle U_{In}, S_{Out} \rangle} \\ &\xrightarrow{G \langle U_{In}, S_{Out} \rangle} U_{Out} \langle D_{Out}, S_{Out} \rangle \xrightarrow{\bar{Q} \langle D_{Out}, S_{Out}, F_{Out} \rangle} \\ &\xrightarrow{\bar{Q} \langle D_{Out}, S_{Out}, F_{Out} \rangle} O_{Out} \langle D_{Out}, F_{Out}, S_{Out} \rangle, \end{aligned}$$

где Q — оператор унификации, выполняющий преобразование данных "отправителя" в унифицированный вид: $Q: O_{In} \rightarrow U_{In}$, U_{In} — данные "отправителя" в терминах унифицированной модели; \bar{Q} — оператор деунификации, выполняющий преобразование данных в терминах унифицированной модели в формат данных "получателя": $\bar{Q}: U_{Out}, F_{Out} \rightarrow O_{Out}$, где U_{Out} — данные "получателя" в терминах унифицированной модели, F_{Out} — это формат хранения данных "получателя". При этом унифицированная модель U представляет собой некоторую универсальную иерархическую структуру, не зависящую от формата данных, в терминах которой может быть представлена любая структура данных. G — оператор, выполняющий преобразование унифицированной структуры данных "отправителя" в унифицированную структуру данных "получателя": $G: U_{In}, S_{Out} \rightarrow U_{Out}$. Оператор G работает с определенной парой структур (структурой данных "отправителя" и структурой данных "получателя") и не зависит от содержания данных, что позволяет использовать оператор многократно для обмена данными заданных структур. На рис. 1 представлена концептуальная модель унифицированного обмена данными в виде диаграммы классов в нотации *UML (Unified Modeling Language)*, описывающая логику процесса, основные сущности и отношения между ними [15].

Согласно формальному определению процесса унифицированного обмена концептуальная модель содержит следующие базовые классы: "Система информационного обмена", "Коммуникационный канал", "Объект информационного обмена", "Формат хранения", "Структура объекта", "Унифицированная модель", "Оператор унификации / деунификации" и "Оператор преобразования". Связь между сущностями определяется соответствующим типом отношений. Отношение агрегации уста-

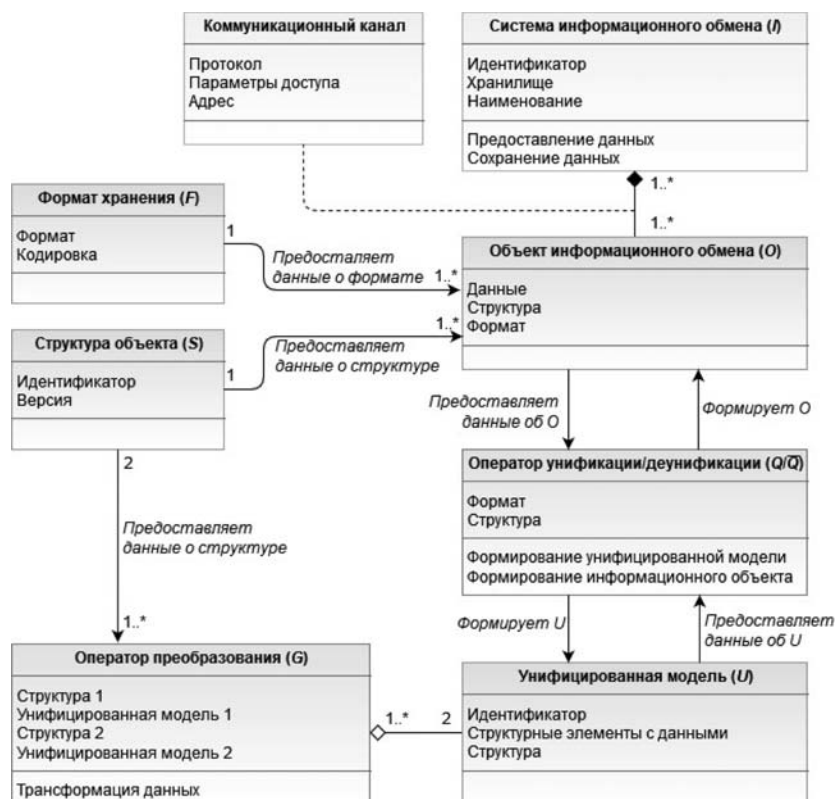


Рис. 1. Концептуальная модель унифицированного обмена данными

навливается между сущностями, когда одна сущность включает множество экземпляров другой сущности и описывает иерархию объектов. На диаграмме отношение агрегации изображается линией с незакрашенным ромбом со стороны сущности-агрегата и с обозначением кратности полюсов, характеризующим число экземпляров сущностей, участвующих в отношении. Отношение композиции задает жесткую зависимость времени существования сущности-части от сущности-агрегата. Графически отношение композиции изображается так же, как и отношение агрегации, но с закрашенным ромбом. Отношение ассоциации отражает наличие произвольной взаимосвязи между сущностями и изображается линией с указанием имени и числа экземпляров сущностей, участвующих в отношении. Отношение зависимости отражает влияние класса, на диаграмме обозначается штриховой линией. Рассмотрим основные атрибуты и характер отношений элементов модели, описывающей информационный обмен на основе унифицированного представления структур данных.

Система информационного обмена (*I*) на диаграмме представлена классом "Система информационного обмена". К данному классу относятся системы, подсистемы или модули системы, осуществляющие отправку или получение данных.

Так, например, в сфере управления закупками в качестве систем информационного обмена выступают модули Автоматизированной системы поддержки размещения муниципального заказа (АСП МЗ), внешние системы размещения заказа (ВСРЗ), Официальный общероссийский сайт закупок (ООС) и электронные торговые площадки (ЭТП). Основными атрибутами данного класса являются: "Идентификатор", "Наименование" и "Хранилище". "Идентификатор" и "Наименование" определяют "отправителя" и "получателя" в процессе информационного обмена, "Хранилище" задает способ и формат хранения данных у "отправителя" и "получателя". Любая информационная система может осуществлять отправку и прием множества объектов информационного обмена.

Объект информационного обмена (*O*) представлен классом "Объект информационного обмена" и связан с классом "Система информационного обмена" *n*-арным отношением композиции. К объекту информационного обмена относятся данные, передаваемые "отправителем" (или направляемые "получателю") в определенном формате.

Активное развитие веб-сервисов создало стандарт обмена данными в виде XML-документов. Так, при организации закупок, из АСП МЗ на ООС и ЭТП передаются сведения о закупках и заключенных контрактах в виде XML-документов, такие как: "Извещение о проведении закупки", "Протокол рассмотрения заявок", "Протокол подведения итогов" и др. В свою очередь, АСП МЗ получает с ООС XML-документы: "План-график закупок", "Сведения о заключенных контрактах", а также необходимые справочники, с ЭТП — "Заявки", "Сведения о поставщике" и др. Кроме того, большинство корпоративных систем в качестве хранилища данных используют реляционные базы данных. Например, между подсистемами АСП МЗ осуществляется передача сведений о планировании закупки и заключении контрактов. Также одной из распространенных форм передачи данных остаются документы, подготовленные средствами Microsoft Word в форматах doc и docx. Например, подсистемами АСП МЗ автоматически формируются и передаются такие документы, как: "Извещение о проведении закупки", "Протокол рассмотрения заявок", "Протокол подведения итогов", "Протокол аукциона" и др.

Класс "*Коммуникационный канал*" обеспечивает передачу объекта информационного обмена между системами. Атрибутами данного класса являются "*Протокол*", "*Параметры доступа*", "*Адрес*", "*Порт*". Например, ВСПЗ обмениваются по протоколу AS2, АСП МЗ принимает данные по протоколу HTTPS с использованием криптографического протокола TLS. XML-документы АСП МЗ формируются в кодировке UTF-8. Информация в АСП МЗ передается с использованием метода POST (Content-Type: multipart/form-data), используя параметры: имя пользователя, пароль пользователя, тип клиента, электронная подпись XML-документа. После получения POST-запроса ООС осуществляет контроль полученной информации по заданным параметрам доступа. Запрос протокола, содержащего сведения о результатах загрузки, ошибках, выявленных нарушениях, выполняется с использованием метода GET, используя такие параметры, как имя пользователя, пароль пользователя, номер загрузки.

Основными атрибутами класса "*Объект информационного обмена*" являются: "*Данные*" (*D*), "*Формат*" (*F*) и "*Структура*" (*S*). Атрибут "*Данные*" определяет содержание объекта информационного обмена. В сфере организации закупок передаваемые данные, как правило, регламентируются законодательными актами и соглашениями информационного взаимодействия.

Атрибут "*Формат*" определяет формат хранения данных в системе информационного обмена и формируется классом "*Формат хранения*" с атрибутами "*Формат*" и "*Кодировка*", от которых зависит способ чтения и записи данных. В настоящее время существует множество форматов хранения, наиболее популярными из них являются XML (eXtensible Markup Language), JSON (Java Script Object Notation), RDB (Relational Data Base), значительно реже в межсистемном взаимодействии используются CSV (Comma-Separated Values) и DOCX (Microsoft Word Open XML Document).

Атрибут "*Структура*" определяет состав и порядок расположения информационных элементов объекта и формируется классом "*Структура объекта*" с атрибутами "*Идентификатор*" и "*Версия*", которые однозначно определяют структуру данных. Обмен одними и теми же данными между системами может осуществляться в одном формате, но в разных структурах. Например, в задаче управления закупками системы ВСПЗ предоставляют одинаковые данные в ЕИС и ЭТП в формате XML, используя при этом разные структуры данных, что связано с независимой разработкой протоколов обмена систем. Формат хранения и структура данных

являются ключевыми метаданными о способе обработки информационного объекта.

Для любого объекта информационного обмена характерна иерархическая структура данных. Например, структура XML-документов описывается с помощью схем данных XSD (XML-schema) и представляет собой дерево, каждый узел которого является XML-тегом и имеет набор свойств, позволяющих определять наличие и число узлов в XML-документах. Структура реляционной базы данных также может быть представлена в виде двухуровневого списка — набора таблиц и их полей. При этом современные СУБД предоставляют доступ к метаданным, из которых можно получить описание таблиц и полей. Структура документов Microsoft Word может быть представлена в виде иерархии элементов, заданных с помощью специализированного языка разметки с набором свойств, позволяющих управлять структурой и содержанием документов. Таким образом, структуры данных объекта информационного обмена в разных форматах за счет единых принципов построения могут быть сопоставлены друг с другом посредством некоторой иерархической унифицированной модели.

Унифицированная модель (*U*) представлена классом "*Унифицированная модель*" и представляет собой набор универсальных взаимосвязанных структурных элементов, организованных по иерархическому принципу, к которому может быть преобразована любая структура данных. В отличие от объекта информационного обмена унифицированная модель не обладает определенным форматом хранения. Основными атрибутами данного класса являются "*Идентификатор*" и "*Структурные элементы*". Представление разнородных структур данных в виде унифицированной модели позволяет сопоставлять и автоматически преобразовывать данные разных систем.

Операции унификации (*Q*) и деунификации (\bar{Q}) представлены на диаграмме классом "*Оператор унификации / деунификации*", связывающим класс "*Объект информационного обмена*" и "*Унифицированная модель*" посредством ассоциативных отношений: предоставление и формирование. Класс "*Оператор унификации / деунификации*" имеет два метода. Метод "*Формирование унифицированной модели*" осуществляет чтение данных из объекта информационного обмена, построение и наполнение структуры в терминах унифицированной модели, освобождаясь от формата хранения. Метод "*Формирование информационного объекта*" обеспечивает обратный процесс — извлечение данных из унифицированной модели в объект информационного обмена с конкретным форматом хранения.

Операция преобразования (G) представлена классом "Оператор преобразования". Основными атрибутами данного класса являются две пары — "Структура 1" и "Унифицированная модель 1" и "Структура 2" и "Унифицированная модель 2" — по одной на "отправителя" и "получателя". Класс "Оператор преобразования" имеет метод "Трансформация данных", который осуществляет перевод данных из одной унифицированной модели в другую путем сопоставления структурных элементов унифицированных моделей и выполнения преобразования данных по установленным взаимным отношениям. Класс "Оператор преобразования" связан с классом "Унифицированная модель" отношением агрегации и с классом "Структура объекта" ассоциативной связью: предоставление данных о структуре.

Унифицированная модель вместе с операциями унификации (деунификации) и преобразования предоставляют возможность автоматизировать процесс обмена данными между гетерогенными системами на основе преобразования структур данных в унифицированное представление, исключив необходимость перепрограммирования при добавлении новых объектов информационного обмена или изменении форматов хранения данных.

Архитектура программных средств унифицированного обмена данными

В соответствии с предложенной моделью разработана архитектура программных средств, обеспечивающая организацию унифицированного обмена данными. На рис. 2 представлена диаграмма, описывающая основные модули архитектуры и их взаимодействие.

На диаграмме "отправитель" и "получатель" данных представлены в виде UML-элементов "актер" — "Data supplier" и "Data recipient" соответственно. Весь процесс унифицированного обмена данными между гетерогенными системами можно условно разделить на два этапа: подготовка и применение.

Этап подготовки (верхняя часть диаграммы) включает такие процессы, как описание структуры данных и описание взаимного соответствия элементов структур данных, которые выполняются "отправителем" и "получателем" данных с помощью визуальных компонентов: *Structure Builder* и *Link Builder*.

Structure Builder — конструктор, предназначенный для описания структуры отправляемых (входных) или получаемых (выходных) данных (S_{In} и S_{Out}). Конструктор предоставляет пользователям информационных систем "отправителя" и "получателя" инструмент для моделирования структуры данных в терминах унифицированной модели. *Link Builder* — конструктор, предназначенный для установления связей между элементами двух разных унифицированных структур, а также для описания операций над отдельными элементами, выполнение которых необходимо при преобразовании одной структуры в другую (*Links Set*). Доступ к визуальным конструкторам обеспечивается через web-интерфейс. *Cross Operand Builder* — конструктор, который на основании описаний структур данных и взаимосвязей между элементами формирует сценарий преобразования данных $A(S_{In}, S_{Out})$ в виде последовательности команд выполнения операций над элементами унифицированных структур данных (перестановка, удаление, агрегация, перерасчет и др.). Поскольку необходимо учитывать особенности структур входных и выходных данных, а также возможные изменения самих данных, сценарий преобразования данных имеет строго односторонний характер, обратное преобразование выполняется по отдельному сценарию. По-

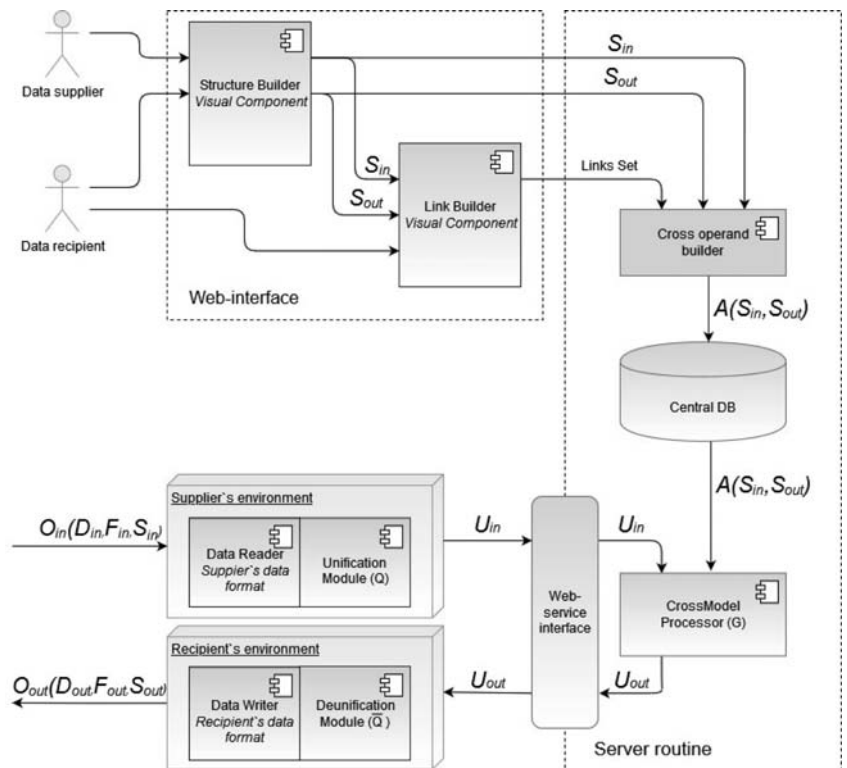


Рис. 2. Архитектура программных средств организации унифицированного обмена данными

строенный сценарий преобразования данных сохраняется в базу данных *Central DB* для дальнейшего применения.

Этап применения (нижняя часть диаграммы) направлен непосредственно на преобразование данных. На вход цепочки преобразования "отправителем" передается объект информационного обмена $O_{In} = \langle D_{In}, F_{In}, S_{In} \rangle$. Формирование унифицированного представления выполняется с помощью двух модулей: *Data Reader* и *Unification Module*. *Data Reader* — модуль, выполняющий считывание данных из источника в формате "отправителя" (F_{In}). *Unification Module* — модуль унификации (Q), выполняющий перевод считанных данных в термины унифицированной модели (U_{In}) и освобождающий данные от их физической природы — формата хранения (F_{In}). В целях устранения возможных конфликтов политик безопасности модули работают на стороне "отправителя" (*Supplier's environment*). На сервер обработки передаются готовые к дальнейшему преобразованию данные в унифицированном виде (U_{In}). *Cross Model Processor* (G) — модуль преобразования, выполняющий преобразование данных отправителя в унифицированном виде (U_{In}) в требуемые получателем данные в унифицированном виде (U_{Out}) на основе заданного сценария $A(S_{In}, S_{Out})$. Сценарий преобразования выбирается из базы данных *Central DB* по паре структур (S_{In}, S_{Out}), где S_{In} — структура, в которой представлены унифицированные данные "отправителя" (U_{In}), S_{Out} — структура, в которой представлены унифицированные данные "получателя" (U_{Out}). В результате обработки — выполнения команд сценария — модулем преобразования *Cross Model Processor* формируются данные "получателя" в терминах унифицированной модели (U_{Out}).

Формирование объекта информационного обмена "получателя" выполняется на стороне "получателя" (*Recipient's environment*) с помощью двух модулей: *Deunification Module* и *Data Writer*. *Deunification Module* — модуль деунификации (\bar{Q}), выполняющий перевод данных в терминах унифицированной модели (U_{Out}) в данные в формате хранения "получателя" (F_{Out}). *Data Writer* — модуль, выполняющий запись объекта информационного объекта "получателя" $O_{Out} = \langle D_{Out}, F_{Out}, S_{Out} \rangle$ в хранилище. Обмен данными между сервером обработки и клиентским окружением "отправителя" и "получателя" осуществляется посредством *Web-service* интерфейса. Возможность хранения сценария преобразования $A(S_{In}, S_{Out})$ позволяет при однократной подготовке структур данных многократно выполнять преобразование данных из структуры S_{In} в структуру S_{Out} . Представленная архитек-

тура программных средств обеспечивает унифицированный обмен данными между гетерогенными системами с учетом вариативности форматов хранения данных и условий функционирования взаимодействующих систем.

Заключение

Обмен данными между гетерогенными системами сегодня является необходимым условием автоматизации деятельности различных организаций. В данной работе предложен технологический подход к унифицированному обмену данными, учитывающий особенности функционирования взаимодействующих систем и вариативность форматов данных. Подход заключается в применении иерархической унифицированной модели, в терминах которой может быть представлена любая структура данных, и последующем выполнении преобразований унифицированных структур. Основные элементы и логика процесса унифицированного обмена описаны с помощью формальной и концептуальной моделей. Ключевыми элементами унифицированного обмена являются унифицированная модель, оператор унификации (деунификации), выполняющий перевод исходных данных в унифицированный вид (и наоборот), и оператор, выполняющий преобразование одной унифицированной структуры данных в другую. Унифицированная модель и операторы позволяют автоматизировать процесс обмена данными, исключив необходимость перепрограммирования при добавлении новых объектов информационного обмена или изменении форматов хранения данных. Также в работе предложена архитектура программных средств, позволяющая реализовать стандартизированный обмен данными на основе формирования унифицированного представления структур данных.

Последующие этапы развития предложенного подхода связаны с разработкой структуры унифицированной модели, алгоритмов формирования и преобразования унифицированных представлений структур данных для различных форматов.

Список литературы

1. Hohpe G., Wolf B. Enterprise integration patterns: design, building, and deploying messaging solutions. Addison-Wesley Professional: Computers, 2004. 683 p.
2. Hasselbring W. Information system integration // Communications of the ACM. 2000. Vol. 3, N. 6. P. 33–38. DOI: 10.1145/336460.336472.
3. Закон РФ "О контрактной системе в сфере закупок товаров, работ, услуг для обеспечения государственных и муниципальных нужд" от 05.04.2013 № 44-ФЗ (в ред. от 24.04.2020).

4. **Morozov R., Zhuchkov D.** Heterogeneous Information Systems Interaction // Proceedings of 3rd Russian-Pacific Conference on Computer Technology and Applications (RPC). 2018. P. 1–4. DOI: 10.1109/RPC.2018.8482146.

5. **Fedak G., He H., Cappello F.** BitDew: a programmable environment for large-scale data management and distribution // Proceedings of International conference for high performance computing, networking, storage and analysis. 2008. P. 1–12.

6. **Haas L. M., Lin E. T., Roth M. A.** Data integration through database federation // IBM Systems Journal. 2002. Vol. 41, N. 4. P. 578–596.

7. **Damodaran S.** B2B integration over the internet with XML — RosettaNet successes and challenges // Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters. 2004. P. 188–195. DOI: 10.1145/1013367.1013398.

8. **Хохгуртль Б.** С# и Java: межплатформенные Web-сервисы. М.: Связь, 2004. 213 с.

9. **Ужаринский А. Ю.** Модель интеграции разнородных источников данных при объединении разнородных приложений на основе web-сервисов // Информационные системы и технологии. 2013. № 6 (80). С. 46–53.

10. **Fernandez M. F., Morishima A., Suci D., Tan W.-Ch.** Method for converting relational data into XML. Patent US 6785673 B1, 2004.

11. **Vrdoljak B., Banek M., Rizzi S.** Designing Web Warehouses from XML Schemas // LNCS. 2003. Vol. 2737. P. 89–98.

12. **Пенькова Т. Г., Коробко А. В., Жучков Д. В., Белорусов А. И., Морозов Р. В.** Программный модуль межсистемного информационного обмена в сфере организации муниципальных закупок // Свидетельство о регистрации программы для ЭВМ RU 2017663163, 2017.

13. **Пенькова Т. Г., Коробко А. В.** Концептуальная модель межсистемного информационного взаимодействия на основе унифицированного представления структуры обмена данными // Информатизация и связь. 2017. № 3. С.17–22.

14. **Penkova T., Kochetkov S.** Unified Hierarchical Model of the Data Exchange // Advances in Intelligent Systems Research. 2017. Vol. 153 P. 143–146. DOI:10.2991/amms-17.2017.33.

15. **Фаулер М.** UML. Основы. Пер. с англ. СПб: Символ Плюс, 2004. 192 с.

T. G. Penkova, PhD, Senior Research Associate, e-mail: penkova_t@icm.krasn.ru,

S. N. Kochetkov, Programmer, e-mail: serega@icm.krasn.ru,

Institute of Computational Modelling SB RAS, Krasnoyarsk, Russian Federation

Technological Approach to the Organization of a Unified Data Exchange between Heterogeneous Systems

This paper presents a technological approach to the organization of a unified data exchange between heterogeneous systems based on the formation of unified representation of data structures. The formal and conceptual models of unified data exchange are presented here. There is proposed the software architecture that provides unified data exchange. Elements of the technological approach are considered in the case of orders for state and municipal needs.

Keywords: unified data exchange, informational interaction of heterogeneous systems, technological approach, conceptual model, software architecture

DOI: 10.17587/it.27.25-31

References

1. **Hohpe G., Wolf B.** Enterprise integration patterns: designing, building, and deploying messaging solutions, Addison-Wesley Professional: Computers, 2004, 683 p.

2. **Hasselbring W.** Information system integration, *Communications of the ACM*, 2000, vol. 3, no. 6, pp. 33–38, DOI: 10.1145/336460.336472.

3. **The law** of the Russian Federation "On the contract system in the field of procurement of goods, work and services for state and municipal needs" dated 04.04.2013 No. 44-FZ (as amended on 04.24.2020) (in Russian).

4. **Morozov R., Zhuchkov D.** Heterogeneous Information Systems Interaction, *Proceedings of 3rd Russian-Pacific Conference on Computer Technology and Applications*, 2018, pp. 1–4, DOI: 10.1109/RPC.2018.8482146.

5. **Fedak G., He H., Cappello F.** BitDew: a programmable environment for large-scale data management and distribution, *Proceedings of International conference for high performance computing, networking, storage and analysis*, 2008, pp. 1–12.

6. **Haas L. M., Lin E. T., Roth M. A.** Data integration through database federation, *IBM Systems Journal*, 2002, vol. 41, no. 4, pp. 578–596.

7. **Damodaran S.** B2B integration over the internet with XML — RosettaNet successes and challenges, *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, 2004, pp. 188–195, DOI: 10.1145/1013367.1013398.

8. **Hochgurtl B.** С # and Java: cross-platform Web services, Moscow, Svyaz, 2004, 213 p. (in Russian).

9. **Uzharinsky A. Yu.** Model for integrating heterogeneous data sources when combining heterogeneous applications based on web services, *Informatsionnyye Sistemy i Tekhnologii*, 2013, no. 6(80), pp. 46–53.

10. **Fernandez M. F., Morishima A., Suci D., Tan W.-Ch.** Method for converting relational data into XML, Patent US 6785673 B1, 2004.

11. **Vrdoljak B., Banek M., Rizzi S.** Designing Web Warehouses from XML Schemas, *LNCS*, 2003, vol. 2737, pp. 89–98.

12. **Penkova T. G., Korobko A. V., Zhuchkov D. V., Belorusov A. I., Morozov R. V.** Program module for intersystem information exchange in the field of municipal procurement, Svidetel'stvo o registratsii programmy dlya EVM RU 2017663163, 2017 (in Russian).

13. **Penkova T. G., Korobko A. V.** Conceptual model of cross-system interaction based on unified representation of data structure, *Informatizatsiya i svyaz*, 2017, no. 3, pp. 17–22 (in Russian).

14. **Penkova T., Kochetkov S.** Unified Hierarchical Model of the Data Exchange, *Advances in Intelligent Systems Research*, 2017, vol.153, pp. 143–146, DOI:10.2991/amms-17.2017.33.

15. **Fowler M.** UML. The basics. Per. from English St. Petersburg, Symbol Plus, 2004,192 p. (in Russian).