

А. Н. Базаркин, Ю. И. Хаткевич, М. И. Хаткевич

Подсистема управления кадрами в интегрированных медицинских информационных системах

Аннотация. Традиционно в сфере здравоохранения к процессу управления кадрами предъявляются гораздо более серьезные требования, чем в других отраслях деятельности. В данной статье обобщен опыт реализации подсистемы управления кадрами в рамках интегрированной медицинской информационной системы, сформулированы основные требования к подсистеме, описана архитектура и методологические принципы ее построения.

Ключевые слова и фразы: подсистема управления кадрами, объектно-реляционный дуализм, механизм генерации сценариев, механизм динамических отчетов, универсальный навигатор, механизм отбора объектов, универсальный конструктор документов.

1. Введение

Интегрированная медицинская информационная система ¹ представляет собой информационную и функциональную среду, объединяющую элементы различных классов медицинских информационных систем. ИМИС обеспечивает как информационную поддержку, так и комплексную автоматизацию всех служб медицинского учреждения [3]. Важное место в ИМИС занимает подсистема управления кадрами, ее эффективная работа является необходимым условием нормального функционирования учреждения.

Традиционно в медицине к процессу управления кадрами предъявляются гораздо более серьезные требования, чем в других отраслях деятельности, что обусловлено основной спецификой деятельности лечебно-профилактических учреждений (ЛПУ). Это влечет за собой увеличение объема, сложности и специфичности функциональности подсистемы управления кадрами. С другой стороны, класс интегрированных информационных систем комплексной информатизации деятельности ЛПУ накладывает свои достаточно жесткие требования общесистемного характера. К этому добавляется требование

¹Везде далее — ИМИС.

возможности работы подсистемы ИМИС в одной из трех конфигураций, а именно:

- с оригинальной подсистемой управления кадрами;
- с подсистемой управления кадрами другого производителя;
- без подсистемы управления кадрами;

Задача построения такой подсистемы является весьма сложной, содержащей в себе ряд системотехнических сложностей, для решения которых требуется принятие научно обоснованных решений архитектурного и методологического характера.

В данной статье обобщен опыт реализации подсистемы управления кадрами в рамках интегрированной медицинской информационной системы, сформулированы основные требования к подсистеме, описана архитектура и методологические принципы ее построения.

2. Требования

Анализ свойств ИМИС позволяет сформулировать требования, которым должна удовлетворять подсистема управления кадрами:

- (1) *Высокие потребительские характеристики* [3]. Внедрение ИМИС существенным образом меняет технологию работы предприятия. Информационная система (ИС) в некоторых случаях играет жизненно важную роль, поэтому она должна обладать рядом следующих потребительских характеристик:
 - (1.1) *производительность*. Большие объемы данных должны обрабатываться с приемлемой скоростью. Время реакции на запрос должно удовлетворять эргономическим нормам;
 - (1.2) *надежность*. Надежностные характеристики системы должны быть таковы, чтобы не раздражать пользователей и обслуживающий персонал и не создавать у них ощущение «неуверенной» работы системы;
 - (1.3) *устойчивость к фатальным ошибкам*. Система должна «оберегать» пользователя от фатальных ошибок при работе с данными;
 - (1.4) *эффективный механизм исправления ошибок*. Система должна предоставлять эффективный механизм устранения ошибок, которые, как правило, неизбежны в ходе ее реализации.

- (2) *Настраиваемость и модифицируемость.* В каждой установке системы приходится сталкиваться с особенностями технологии работы конкретного предприятия. Подсистема не должна быть ригидной. Особенно важно это в отношении ИМИС, т. к. заранее неизвестно, какой фрагмент системы потребуется локализовать и в какой степени. Поэтому к ИМИС предъявляются значительно большие требования к настраиваемости и модифицируемости.
- (3) *Модульная организация.* Из всей функциональности, которой обладает ИМИС, может быть востребована лишь некоторая ограниченная ее часть. Кроме того, процесс внедрения ИМИС может занимать годы, поэтому даже в случае внедрения системы целиком на протяжении довольно длительного периода работает лишь часть системы. Какие части системы будут внедряться, и в какой последовательности, зависит от стратегии внедрения в конкретном случае. Таким образом, система должна иметь модульную организацию, позволяющую совершать «маневры» на очень важной фазе жизненного цикла системы — этапе внедрения.
- (4) *Расширяемость.* Для того, чтобы система не утратила актуальность по завершении этапа внедрения, она должна быть в состоянии развиваться без вмешательства разработчика, однако невозможно требовать от системы одинаковой способности к развитию во всех направлениях. Это приведет к снижению качественных и специфичных характеристик системы, превратит систему в универсальную оболочку — конструктор для построения неэффективных систем. На этапе проектирования необходимо определить перечень направлений и глубину потенциального развития и снабдить эти направления конструкторами.
- (5) *Универсальность.* Функциональность системы должна быть проанализирована на предмет повторяющихся фрагментов, и для реализации общих функциональных фрагментов необходимо прибегнуть к построению универсальных механизмов. Это существенным образом снижает сложность и повышает качественные характеристики системы.



Рис. 1. Модификация трехуровневой архитектуры

- (6) *Концептуальная целостность.* Система не должна представлять собой набор разнородных средств, имеющих разнородный интерфейс и различные дисциплины работы. Напротив, каждый фрагмент системы должен быть проникнут концептуальным единством, чтобы, ознакомившись с одним фрагментом системы, большое количество навыков и знаний можно было применить и для освоения фрагментов других системы.
- (7) *Отчуждаемость.* Для прохождения важного этапа жизни системы — отчуждения ИС от разработчика — система должна быть построена таким образом, чтобы ее сопровождение могло бы осуществляться относительно небольшим числом людей более низкой квалификации, чем персонал разработчика.
- (8) *Простота.* Все виды деятельности в отношении системы: освоение, использование, сопровождение, настройка, расширение — должны иметь такую степень сложности, которая бы соответствовала возможностям персонала, уже работающего на данных позициях в ЛПУ. Если внедрение системы будет предполагать существенную смену персонала, такая система не будет жизнеспособна.

Анализ требований к ИМИС показывает, что построение подсистемы управления кадрами в рамках ИМИС — задача весьма сложная. При реализации подсистемы используется подход объектно-реляционного дуализма для построения больших информационных систем [1], который с успехом применялся для построения ИМИС Интерин, Котем 2001, ИСУ ЦКБ РАН, PROMIS и др.

3. Архитектура системы

Для реализации изложенного в работе подхода предлагается использование трехуровневой архитектуры [2]. В трехуровневой архитектуре в дополнение к известным уровням двухуровневой архитектуры (клиент и сервер) добавляется промежуточный уровень (средний), основное назначение которого собрать в одном месте всю функциональность ИС по работе с данными, подобно тому, как собраны и упорядочены в реляционную модель данные в двухуровневой архитектуре. Трехуровневая архитектура имеет следующие преимущества:

- (1) более точное распределение функциональной нагрузки:
 - нижний уровень, организация и хранение данных;
 - средний уровень, организация функциональности системы по работе с данными;
 - уровень приложения, организация функциональности системы отвечающей за пользовательский интерфейс;
- (2) функциональность по работе с данными вынесена из клиентских приложений и собрана в одном месте, что позволяет оперировать с функциональностью системы как с единым целым;
- (3) клиентские приложения отвечают только за функциональность по организации интерфейса с пользователем, что существенно упрощает приложение;
- (4) современные реляционные системы управления базами данных позволяют физически располагать средний уровень на серверном, что позволяет технологически объединить две важнейшие составляющие ИС: данные и функциональность.

Однако, в «чистом» виде трехуровневая архитектура имеет важный недостаток. Проблема в том, что в данной архитектуре реляционная модель данных является единственным источником понятий и связей между ними. А вследствие того, что реляционной модели не

обладает средствами конструирования понятий более высокого уровня, процедуры бизнес-логики и клиентские приложения вынуждены работать с понятиями, которые предоставляет реляционная модель, насколько бы неадекватными они ни были. Для устранения указанного недостатка предлагается модификация трехуровневой архитектуры, в которой функциональность среднего уровня упорядочивается в виде объектной модели, построенной над реляционной моделью данных (Рис. 1). По сравнению с традиционными средствами реализации среднего уровня (процедурные программные единицы, представления) использование объектной модели имеет ряд преимуществ:

- (1) объектные средства позволяют конструировать понятия и связи между ними произвольного уровня абстракции;
- (2) использование средств объектного программирования, в особенности таких, как инкапсуляция данных, динамический полиморфизм, наследование, повторное использование кода, позволяет реализовать уровень бизнес-логики ИС на качественно новом уровне.

Предлагаемая архитектура содержит в себе два независимых представления одних и тех же данных: реляционную и объектную модели, что придает системе качественно новое сочетание характеристик:

- (1) Две модели данных обеспечивают возможность структурирования функциональной нагрузки между реляционными и объектными средствами в соответствии с их свойствами. Реляционные средства:
 - организация и хранение данных;
 - критичные по времени и объему выборки данных;
 - формирование аккумулирующих отчетов;
 - реализация специализированных рабочих мест.Объектные средства:
 - организация бизнес-логики системы;
 - создание универсальных механизмов;
 - миграция информации (экспорт, импорт);
 - формирование объектного интерфейса системы.
- (2) Реляционная и объектная модели независимы. Первая занимается данными, вторая — функциональностью и не содержит собственных данных, а является своеобразным каналом доступа к ним. Это позволяет нескольким классам объектов, различающимся как уровнем, так и аспектом абстракции,

проецироваться на одну сущность реляционной модели. Соответственно, одному фрагменту реляционной модели данных может соответствовать несколько зависимых или независимых фрагментов объектной модели данных. Такая возможность позволяет организовать многоаспектный доступ к данным.

(3) Наличие объектной модели уже на серверном уровне позволяет использовать преимущества объектных технологий на всех уровнях. Важнейшим преимуществом является возможность структурирования функциональности системы на две составляющие: универсальную и специальную. Это чрезвычайно важно, поскольку:

- универсальная составляющая реализуется один раз и потом многократно используется как в рамках одной системы, так и от системы к системе;
- отделение универсальной составляющей функциональности позволяет более качественно реализовать специальную часть;
- универсальная составляющая представляет собой постоянно пополняющийся набор универсальных механизмов, который используется как палитра для реализации подсистем ИМИС;
- механизмы универсальной составляющей развиваются независимо от конкретных ИМИС.

4. Универсальные механизмы

Многолетний опыт использования изложенной выше архитектуры дает возможность сформировать ряд универсальных механизмов, использование которых существенно упрощает процесс разработки и реализации подсистемы управления кадрами [1].

- *Универсальный навигатор.* Универсальный навигатор служит для визуализации объектной модели, навигации по ней и выполнения действий над объектами модели. Реализованная в полной мере возможность многоаспектного доступа к данным позволяет каждому пользователю иметь свой взгляд

на данные системы. Наилучшим средством реализации данной возможности является предоставление каждому пользователю фрагмента объектной модели. В этом случае универсальный навигатор становится универсальным рабочим местом.

- *Механизм отбора объектов.* Для реализации ряда универсальных механизмов требуется средство, которое позволяет осуществлять отбор информации в реляционной модели данных и представлять результат отбора в виде множества объектных ссылок. В качестве такого средства предлагается универсальный механизм отбора объектов, который реализован на основе класса «Универсальный контейнер» и имеет специализированную форму задания условий отбора для каждого типа отбираемых объектов.
- *Механизм динамических отчетов.* Часто в работе с ИМИС требуется получение однородной информации в разных видах, например, требуется распечатать произвольное сочетание атрибутов следующих сущностей: сотрудников, пациентов, заболеваний, диагнозов, назначений, услуг и т. д. Причем крайне важно, чтобы эти действия могли быть осуществлены конечным пользователем системы, а не администратором и тем более не разработчиком. Для реализации такой возможности создан универсальный механизм динамических отчетов, который имеет следующие особенности:
 - (1) разделение между собой фазы выборки и отображения информации;
 - (2) выборка при помощи универсального механизма отбора объектов. Применение одного из описанных шаблонов отображения к папке, содержащей отображенные объектные ссылки, с помощью механизма;
 - (3) отображение результата работы данного механизма в окне универсального навигатора.
- *Механизм сценариев.* Механизм сценариев разработан для того, чтобы иметь возможность сначала подготовить, а потом применить к модели данных сразу большое количество изменений. Например, вся деятельность подсистемы управления кадрами в части персонала управляется приказами

руководителя учреждения. Приказ проходит последовательно несколько стадий: составление, согласование, утверждение, печать бумажной копии, применение к модели данных. Логика прохождения приказа хорошо реализуется при помощи механизма сценариев. Универсальный механизм сценариев имеет следующие особенности:

- (1) сценарий это составной объект, который состоит из произвольного количества пунктов. Над сценарием определены действия «Печать» и «Применить». Первое печатает сценарий, а второе применяет его к модели данных;
 - (2) пункт сценария — это объект определенного типа. Для каждого типа определен шаблон внешнего вида, который используется для печати, форма запроса параметров и алгоритм применения пункта к модели данных;
 - (3) результат действия «Печать» сценария и пункта отображается в правом окне универсального навигатора.
- *Механизм генерации сценариев.* Механизм генерации сценариев разработан для того, чтобы иметь возможность автоматического формирования сценариев с однородным содержанием. Например, в ведомственных учреждениях время от времени происходит индексация окладов, при которой оклад изменяется у каждого работника. Формирование приказов вручную для каждого сотрудника задача очень трудоемкая, но с использованием механизма генерации сценариев существенно упрощается.

Универсальный механизм генерации сценариев обладает следующими особенностями:

- (1) является примером автоматизации второго порядка по причине того, что надстроен над механизмом сценариев, который сам является средством автоматизации рутинного труда;
- (2) использует универсальные механизмы «Отбора объектов», «Сценариев». Данный механизм генерирует сценарий, в котором для каждого отобранного объекта генерируется пункт сценария заданного типа;
- (3) данный механизм является примером высокой степени автоматизации рутинного труда оператора ИС по вводу информации. Использование его позволяет в некоторых случаях уменьшить затраченное время в 50–100 раз.

- *Универсальный конструктор документов.* Иногда необходимо сделать некоторый фрагмент системы очень гибким. Например, в медицине внешний вид и содержание врачебных медицинских документов (первичный осмотр, дневник, эпикриз и т. д.) меняются не только от системы к системе, но и в рамках одной системы в зависимости от отделения, специализации врача и т. д. Для решения указанной проблемы в системе реализуется универсальный механизм конструирования документов.

5. Особенности реализации

Подсистема управления кадрами содержит анкетные и профессиональные данные сотрудников, информацию о структуре организации, штатном расписании, о расстановке штатов [4]. Значительная часть этой информации может использоваться для реализации сложных аккумулирующих отчетов, поэтому административная подсистема обязана иметь хорошо спроектированную реляционную модель данных. На крупном медицинском предприятии поток административной информации достаточно интенсивен, поэтому подсистема должна иметь высокую степень автоматизации ручного труда, чтобы силами 1–2 человек поддерживать актуальное состояние данных. Функциональность подсистемы может быть структурирована на две части: универсальную и специальную. Универсальная часть функциональности реализуется при помощи общесистемных универсальных механизмов:

- (1) механизм «Универсальный навигатор» используется для реализации универсального рабочего места пользователя;
- (2) механизм «Рабочий стол» используется для формирования информационного аспекта конкретного пользователя. Каждый пользователь имеет свой объект «Стол», который является универсальным контейнером объектов. Совокупность «лежащих на столе» объектов и определяет фрагмент объектной модели доступной пользователю;
- (3) механизм отбора объектов используется в качестве поставщика объектов для механизмов «Динамических отчетов» и «Генерации сценариев»;
- (4) механизм динамических отчетов используется для формирования различных представлений выборок объектов «Сотрудник» и «Приказ»;

- (5) механизм сценариев используется для реализации механизма приказов, который является основным способом работы с административной информацией;
- (6) механизм генерации сценариев используется для генерации приказов с большим количеством однородных пунктов. На вход механизма поступает выборка объектов «Сотрудник», на выходе — приказ по личному составу заданного типа.

Функциональность специальной части структурируется на три составляющие:

- (1) Реляционная модель данных представляет собой реляционную модель предметной области без сущностей и связей, которые реализуют сервисную функциональность.
- (2) Базовая реляционная функциональность представляет собой стандартный набор средств реализации функциональности: пакеты, формы, отчеты. Функциональность пакетов можно разбить на несколько частей: одна часть сосредоточена вокруг таблиц и предоставляет собой процедурный интерфейс к ним, вторая реализует бизнес-логику форм. Функциональность форм существенно упрощается за счет выноса из них бизнес-логики, сервисной функциональности и таких сложных для реализации компонентов как дерево, реализация которого обеспечивается механизмом «Универсальный навигатор». Мощность большинства форм такова, что они реализуют одно действие объекта, что примерно соответствует редактированию одной записи таблицы. Благодаря наличию универсального механизма динамических отчетов с помощью реляционных отчетов реализуются лишь сложные аккумулирующие отчеты.
- (3) Базовая объектная функциональность представляет собой описание всех необходимых объектов средствами механизма «Информационных объектов» (см. главу 4) и реализацию методов классов.

Поскольку большинство механизмов универсальной части являются конструкторами, система приобретает большую степень гибкости, что повышает такие ее характеристики как настраиваемость, модифицируемость, расширяемость. Наличие в системе ярко выраженного ядра в виде совокупности универсальных механизмов повышают такие ее характеристики как концептуальная целостность,

универсальность, простота. Перечисленные характеристики соответствуют требованиям к ИМИС, сформулированным в данной работе.

Список литературы

- [1] Хаткевич М. И. *Объектно-реляционный дуализм в информационных системах* // Программные продукты и системы. — № 3, 2002, с. 22–26. ↑
- [2] Хаткевич М. И., Матвеев Г. Н. Принцип объектно-реляционного дуализма для построения медицинских информационных систем. Современные информационные технологии в диагностической, лечебной и образовательной деятельности. — Москва, 2002, с. 72. ↑
- [3] Назаренко Г. И., Гулиев Я. И., Ермаков Д. Е. Медицинские информационные системы: теория и практика. — Москва: Физматлит, 2005, с. 320. ↑
- [4] Соловьев А. А. Кадровый отдел организации здравоохранения. — Москва: ПРИОР, 2006, с. 128. ↑

ИССЛЕДОВАТЕЛЬСКИЙ ЦЕНТР МЕДИЦИНСКОЙ ИНФОРМАТИКИ ИПС РАН

A. N. Bazarkin, Yu. I. Khatkevich, M. I. Khatkevich. *Personnel administration subsystem in the integrated medical information systems.* (in russian.)

АБСТРАКТ. Традиционно в сфере публичных служб здравоохранения к управленческому процессу со стороны персонала предъявляются гораздо более серьезные требования, чем в других областях деятельности. В данной статье описан опыт реализации подсистемы управления персоналом в рамках интегрированной медицинской информационной системы. Обобщены основные требования к подсистеме, сформулированы архитектурные и методологические принципы ее построения.