

удк 002.53:61

М. И. Хаткевич, Ю. И. Хаткевич

## Подсистема назначений комплексного медицинского центра. Опыт разработки и эксплуатации

**Аннотация.** Данная статья посвящена описанию основных концептуальных и технологических решений, полученных при разработке, внедрении и сопровождении подсистемы назначений в комплексном, многопрофильном лечебном учреждении на протяжении нескольких лет. Формулируются основные требования к подсистеме назначений по разделам: многоаспектность, функциональная полнота, общесистемные требования. Формулируются основные концептуальные и технологические решения по формулировке, базису, привязке, кодированию, вводу назначений. Предлагаются средства типизации назначений. Описывается механизм шаблонов назначений, который существенно повышает скорость ввода назначений в информационную систему.

*Ключевые слова и фразы:* медицинская информационная система, подсистема назначений, многоаспектность, шаблоны назначений.

### 1. Введение

Лечебный процесс — это сложный комплекс взаимосвязанных процессов, происходящих в лечебно-профилактическом учреждении (ЛПУ), особенно если это касается комплексного, многопрофильного ЛПУ, инфраструктура которого содержит стационар, поликлинику, диагностический и реабилитационный центры.

Основным и корневым для всех остальных является процесс назначения и исполнения диагностических и лечебных назначений. От того, насколько четко отлажено взаимодействие всех служб ЛПУ в рамках данного процесса, во многом зависит качество лечения, эффективность работы учреждения в целом и, в конечном счете, конкурентоспособность.

Рыночные отношения, складывающиеся в настоящее время в сфере предоставления медицинских услуг, побуждают ЛПУ развивать технологию работы с целью повышения данных характеристик.

Однако существующая в подавляющем большинстве лечебных учреждениях бумажная технология назначений является фактором, сдерживающим развитие технологии работы ЛПУ.

Процесс назначения и исполнения диагностических и лечебных назначений имеет существенную информационную составляющую, поэтому возможна его эффективная информатизация. Информатизация основного процесса лечебного учреждения позволяет сформировать более мощный базис для развития технологии работы ЛПУ и предоставляет возможности для анализа деятельности ЛПУ, контроля качества лечения и контроля обоснованности расходования ресурсов.

Описанию основных концептуальных и технологических решений, полученных на протяжении нескольких лет при разработке, внедрении и сопровождении подсистемы назначений в комплексном многопрофильном учреждении и посвящена данная статья.

## 2. Требования

Ввиду того, что процесс назначений является основным в лечебном учреждении, к подсистеме назначений предъявляются очень высокие требования [1, 2]. Перечислим основные из них.

**2.1. Многоаспектность функциональности.** Одной из базовых характеристик системы назначений является многоаспектность. Причем каждый аспект имеет свою устоявшуюся за многие десятилетия технологию работы, свои традиции и условности. По мере возможности, все это должно быть учтено при реализации системы назначений, чтобы внедрение и эффективное использование системы было возможно. Основными являются следующие аспекты и относящиеся к ним функциональные блоки:

1. Лечебный аспект. Включает в себя следующие функциональные блоки:
  - назначение,
  - диспетчеризация,
  - исполнение,
  - контроль.
2. Аспект статистической обработки:
  - статистика по назначенному (по врачу, по отделению, по ЛПУ в целом),

- статистика по исполненному (по врачу, по отделению, по ЛПУ в целом).
3. Экономический аспект:
- справочник услуг,
  - прејскуранты,
  - контроль экономической обоснованности назначения.
4. Аспект учета материальных ценностей:
- списание препаратов и расходных материалов на исполнение назначения,
  - контроль за расходом материальных ценностей.
5. Потребление ресурсов:
- ведение справочника ресурсов,
  - контроль наличия ресурса и выделение в момент заполнения.

**2.2. Функциональная полнота.** Подсистема назначений должна позволять:

- работать со всем спектром назначений;
- учитывать особенности бизнес-логики прохождения всех типов назначений;
- ввод, хранение и печать всех необходимых документов;
- подключать различное медицинское диагностическое оборудование для получения результатов исследований;
- интеграцию с диагностическими и лабораторными системами.

### **2.3. Общесистемные требования.**

1. *Высокие потребительские характеристики.* При внедрении подсистемы назначений существенным образом меняется технология работы предприятия. Информационная система начинает играть жизненно важную роль. Поэтому система не должна быть «узким местом» по всем направлениям потребительских характеристик:

- *Производительность.* В крупной клинике количество назначений в день может исчисляться десятками тысяч, они должны обрабатываться с приемлемой скоростью. Время реакции на запрос должно удовлетворять эргономическим нормам.

- *Надежность программно-аппаратного комплекса.* Фиксация назначений на приеме врача-специалиста поликлиники требует работы системы в реальном масштабе времени в течение всего рабочего дня; ночью, когда в системе работает приемное отделение и дежурные врачи, требование работы в реальном времени не обязательно. Прием врача в среднем занимает от 15 до 30 минут, поэтому сбой и восстановление после сбоя не должны превышать 5 – 10 минут. 5 минут приблизительно соответствуют перезагрузке клиентского программного обеспечения (ПО), а 10 минут — перезагрузке серверного ПО. Практика использования информационных систем показывает, что частота отказов работы системы не воспринимается персоналом болезненно, если значение этого параметра не превышает одного раза в неделю.
- *Устойчивость к фатальным ошибкам.* Ошибки, при которых портится большое количество информации без возможности восстановления, можно назвать фатальными. Интерфейс системы должен запрещать эти операции для конечного пользователя, а при необходимости совершения пользователем критических операций любая из них должна иметь возможность «отката».
- *Эффективный механизм исправления ошибок.* В случае ошибок оператора при вводе информации в систему последняя должна предоставлять эффективный механизм исправления ошибочно введенной информации. Подход, при котором пользователю в любое время разрешается редактировать введенную информацию, совершенно не годится для систем данного класса, поскольку система должна предохранять введенные данные как от случайного, так и от преднамеренного искажения семантики. С другой стороны, ошибки ввода данных неизбежны, поэтому без механизма исправления ошибок система нежизнеспособна.

2. *Настраиваемость и модифицируемость.* Каждой установке подсистемы назначений приходится сталкиваться с

особенностями технологии работы конкретного предприятия. Подсистема не должна быть ригидной, должно быть предусмотрено несколько уровней адаптивности системы:

- уровень конечного пользователя (например, настройка своих шаблонов);
- уровень ответственного специалиста (ведение справочника услуг в рамках своей компетенции, ведение прейскурантов, настройка шаблонов отделения и т.д.);
- уровень администратора системы (ведение справочника услуг целиком, классифицирование услуг, массовые операции при работе с прейскурантами, и т.д.);
- уровень прикладного разработчика со стороны исполнителя (настройка общей бизнес-логики работы подсистемы, определение системы ограничений бизнес-логики, ведение классификатора);
- уровень разработчика (модификация программного кода).

3. *Расширяемость*. Чтобы система назначений не утратила актуальность сразу после внедрения, она должна уметь развиваться без вмешательства разработчика. Однако невозможно требовать от системы одинаковой способности развиваться во всех направлениях. Это приведет к выхолащиванию из системы специфики, превратит ее в универсальную оболочку-конструктор для построения плохих систем. Необходимо на этапе проектирования определить перечень направлений и глубину потенциального развития и снабдить эти направления конструкторами.

4. *Универсальность*. Функциональность подсистемы назначений должна быть проанализирована на предмет повторяющихся фрагментов, и для реализации общих фрагментов функциональности необходимо прибегнуть к использованию уже имеющихся универсальных механизмов или построению новых. Это существенным образом снижает сложность и улучшает характеристики системы.

5. *Концептуальная целостность*. Система назначений не должна представлять собой набор разнородных средств, имеющих неоднородный интерфейс и разные дисциплины работы. Напротив, каждый фрагмент системы должен быть

проникнут концептуальным единством, чтобы большое количество навыков и знаний, полученных при знакомстве с одним фрагментом системы, пригодилось бы и для освоения других фрагментов.

6. *Отчуждаемость*. Для успешного прохождения важного этапа в жизни системы — отчуждения информационной системы от разработчика — система должна быть построена таким образом, чтобы она могла сопровождаться относительно небольшим числом людей, а требования к персоналу не должны быть слишком высокими.
7. *Простота*. Все виды деятельности в системе (освоение, использование, сопровождение, настройка, расширение) должны иметь такую степень простоты, которая бы соответствовала персоналу, уже работающему на данных позициях на предприятии. Если внедрение системы будет предполагать существенную смену персонала предприятия, такая система не будет жизнеспособна.

### 3. Концептуальные и технологические решения

**3.1. Формулировка назначения.** Основной интерфейсной парадигмой подсистемы назначения выбрана нотация листа назначений в стационаре, которая представляет собой таблицу, где строки — это кратные назначения, а колонки — это даты. В квадрате пересечения строки и колонки можно задать назначение, отметив его дату. Таким образом, назначение — это всегда пара:

- *формулировка назначения* — множество атрибутов, определяющих, «что» нужно сделать пациенту;
- *дата или диапазон дат назначения* — множество атрибутов, определяющих, «когда» это нужно сделать.

Такое решение позволяет существенно увеличить скорость работы врача в случае кратных назначений. Врач один раз вводит формулировку, а затем делает назначение, проставляя крестик в колонке с интересующей датой. Временная сетка может иметь разный масштаб, например, в отделениях реанимации используется почасовая сетка, в поликлинике может использоваться недельная, месячная и т. д.

Формулировка существует в двух видах:

- текст формулировки назначения (именно так ее видит и подписывает врач) является основным документом;

- множество структурированных атрибутов формулировки, необходимых системе для анализа.

Структурированные атрибуты делятся на:

- универсальные атрибуты, которые имеют смысл для всех назначений (например, «сколько раз», «какое количество», «периодичность» и т. д.), реализуются реляционным образом;
- специальные атрибуты, имеющие смысл только для назначений определенного типа, реализуются в присоединенной таблице в виде пар «наименование атрибута — значение атрибута».

Возможны две стратегии ввода назначений:

- врач вводит формулировку посредством множества структурированных атрибутов, а затем текст генерируется в соответствии с алгоритмом, определенным для данного типа;
- врач вводит назначение в виде простого текста, а затем текстовая формулировка автоматизированным образом кодируется, т. е. выделяется необходимый минимум структурированных атрибутов. См. ниже — кодирование назначений.

**3.2. Базис назначения.** Сложной задачей является реализация комплекса медицинских назначений. Задача структурируется на три подзадачи:

- (1) комплекс медикаментозных назначений;
- (2) назначение с медикаментами;
- (3) комплекс произвольных назначений.

Подавляющее большинство случаев использования комплекса медицинских назначений — это использование первых двух вариантов комплекса. Например: 1 — капельница, 2 — процедура с медикаментом. Первые две задачи решаются при помощи усиления базиса назначения (рис. 1):

*Процедура + {Медикамент}\**

Таким образом, внутри системы комплекс медикаментозных назначений и назначение с медикаментами комплексами не являются, а реализуются как одно назначение. Третья подзадача — реальный комплекс — поддерживается в системе при помощи шаблонов.

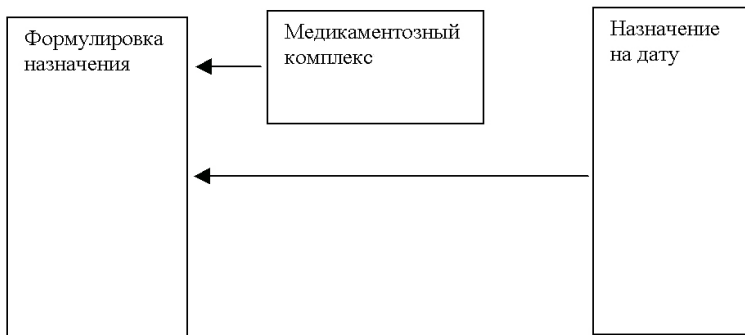


РИС. 1. Базис медицинского назначения

**3.3. Привязка назначения.** Вся информация медицинского характера в системе привязывается к медкарте. При этом медкарта может быть одной из двух сущностей: история болезни, амбулаторная карта.

- *История болезни:* основная привязка — это ссылка на медкарту из формулировки назначения, дополнительная привязка — ссылка на идентификатор сессии (подход к пациенту) в самом назначении.
- *Амбулаторная карта:* основная привязка — это ссылка на медкарту из формулировки назначения, также есть две дополнительные привязки:
  - ссылка на случай обращения из формулировки (чаще всего заболевание, оформляется талон амбулаторного пациента);
  - ссылка на прием из самого назначения.

**3.4. Кодирование назначения.** Удобство и скорость ввода назначений — важнейшие характеристики подсистемы назначений. Недостаточность одной из этих характеристик может критическим образом отразиться на возможности использования подсистемы в ЛПУ. Для улучшения данных характеристик была разработана концепция отложенного кодирования информации в системе. Поясним, в чем ее суть.

Традиционно при работе с реляционными базами данных принято структурировать информацию в момент ввода, т. е. в момент ввода



информация раскладывается по атрибутам таблиц реляционной модели данных. Это для многих случаев удобно, особенно если поток информации небольшой, а работа носит регулярный характер. Не так в случае с вводом назначений врачом. Во-первых, поток очень большой (например, в поликлинике врач за прием может делать сотни назначений), а во-вторых, формат ввода — не регулярен, так как зависит от конкретного типа назначений. Кроме того, врач — основной фигурант лечебно-профилактической деятельности, поэтому система должна предоставлять возможность освободить врача от рутинной деятельности, коей, несомненно, является структурирование информации о назначении на входе системы.

Наряду с традиционной парадигмой работы в системе предусматривается следующая парадигма. Врач вводит назначение свободным текстом; заметим, что этого уже достаточно, чтобы осуществлять несколько важных функций в системе: транспортную (когда введенную формулировку видно другим участникам процесса лечения так, как ее ввел врач) и печать в медицинских документах.

Структурирование информации можно выполнять, во-первых, в другое время, во-вторых, этим могут заниматься другие сотрудники и не для всего массива назначений, а только для интересующих типов назначений или категории пациентов или исходя из других соображений. Таким образом, можно разгрузить не только врача, но и других участников лечебно-профилактической деятельности. Например, кодирование информации о назначениях можно возложить на сотрудников экономического отдела, чтобы они выполняли его по мере надобности, для определенного контингента пациентов (договорные пациенты) и для дорогостоящих, т. е. не включенных в стоимость койкодня назначений.

Таким образом, допускается создание формулировки назначения простым текстом с последующим кодированием. Отметки о кодировании хранятся в полях таблицы формулировки:

- состояние кодирования;
- дата последнего изменения кодирования;
- автор последнего изменения кодирования.

Выделяется несколько уровней кодирования:

- *Кодирование типа.* Соотнесение назначения с типом. Поддерживается пять уровней типизации.

Пример значений трех первых уровней типов для УЗИ:

- (1) диагностическое назначение,
  - (2) лучевое исследование,
  - (3) ультразвук.
- *Кодирование количественно-временных атрибутов*: количество, единицы количества, сколько раз, период, единицы периода, дополнительная инструкция. Например: 30 мг 5 раз в день перед сном.
  - *Кодирование назначения по справочнику услуг*. Установление ссылки в назначении на определенную позицию справочника услуг.
  - *Стоимостное кодирование*. Задание финансовых атрибутов назначения.
  - *Конкретизация назначения (диспетчеризация)*. Кем (сотрудник), где (отделение, кабинет), при помощи чего (аппарат) будет выполняться данное назначение.

Пример распределения нагрузки по вводу, исполнению и кодированию назначений:

Врач:

- ввод формулировки;
- кодирование типа;
- назначение (приписывание конкретной даты).

Постовая медсестра:

- кодирование количественно-временных атрибутов;
- кодирование назначения по справочнику услуг для своих назначений;
- выполнение для своих назначений;
- диспетчеризация назначений для параклиники общая: спецификация отделения, лаборатории.

Диспетчер исполнителя:

- диспетчеризация назначений для параклиники детальная: спецификация исполнителя.

Исполнитель:

- кодирование назначения по справочнику услуг;
- исполнение.

Финансист:

- стоимостное кодирование.

**3.5. Ввод назначения.** Ввод назначения в систему разделяется на две фазы: ввод формулировки назначения и приписывание конкретного времени исполнения.

**Создание формулировки назначения.** Выделяются следующие состояния формулировки назначения:

- «Шаблон» — обезличенная формулировка, шаблоны показываются только в форме конструктора шаблонов, назначение невозможно.
- «План» — не включенная в рабочее множество формулировка, показывается серым, назначение невозможно.
- «В работе» — в этом режиме только и возможно назначение. Без ограничений можно редактировать формулировку назначения, находящуюся в состоянии «План» и «Шаблон». Формулировку в состоянии «В работе» разрешается редактировать, только если не было выполнено ни одного назначения.
- «Закончено» — исключенные из рабочего множества формулировки. В этом состоянии временная привязка невозможна. Не разрешается редактировать формулировку, находящуюся в этом состоянии.

Состояние формулировки назначения хранится в полях таблицы формулировки:

- состояние;
- дата последнего изменения состояния;
- автор последнего изменения состояния.

**Приписывание формулировке конкретного времени исполнения.** В этот момент и происходит собственно назначение. Для этого в правой части формы организована шкала дат. Сверху — управление шкалой: поле для задания даты (по умолчанию текущая дата), относительно которой разворачивается шкала дат, возможность подвинуть шкалу вправо-влево. Возможно изменение масштаба шкалы: день, час, время дня (утро, день, вечер, ночь). Выделяется цветом колонка текущей даты. Состояние назначения выделяется цветом символа, который проставлен на временной шкале:

- «Назначено» — без выделения;
- «Отменено» — красным цветом.

Состояние назначения хранится в полях таблицы назначения:

- состояние;
- дата последнего изменения состояния;
- автор последнего изменения состояния.

Состояние исполнения выделяется символом:

- '?' — дата запланирована;
- '!' (или пробел) — дата утверждена;
- '+' — выполнено;
- '+-' — выполнено частично;
- '-' — не выполнено (отказ).

Состояние исполнения хранится в полях таблицы назначения:

- дата последнего изменения состояния;
- причина изменения состояния;
- автор последнего изменения состояния.

**3.6. Типизация назначений.** Для реализации полноценной медицинской системы необходим мощный механизм поддержки классификации назначений (услуг), однако такой механизм целесообразно вынести из подсистемы назначений и реализовать отдельно как надстройку над справочником услуг. Однако для реализации подсистемы необходим внутренний классификатор, который бы удовлетворял технологическим требованиям к подсистеме. Например, ввод медикаментозного, диагностического или лабораторного назначений требует разных интерфейсных решений. Исходя из этого, в подсистеме используется внутренний трехуровневый классификатор назначений. Два уровня трехуровневой технологической классификации назначений:

1. Медикаментозный комплекс.
2. Процедура.
3. Лабораторное исследование.
  - 3.1. Клиническое.
  - 3.2. Биохимическое.
  - 3.3. Иммунологическое.
  - 3.4. Бактериологическое.
4. Диагностическое исследование.
  - 4.1. Лучевое.
  - 4.2. Функциональное.
  - 4.3. Эндоскопическое
5. Консультация.

6. Операция.
7. Трансфузия.
8. Режим (общий, платный, полупостельный, постельный, постельный строгий).
9. Наблюдение (дежурный врач, сиделка).
10. Диета.
11. Прием врача.
12. Рецепт.
13. Проживание (комната, койка, классность, персональная медицинская сестра).
14. Прочая услуга (шкаф, телефон, и т.д.).
15. Внешняя услуга.

**3.7. Шаблоны назначений.** Для ускорения работы по вводу медицинских назначений разработана система шаблонов, позволяющая врачу заранее приготовить (ввести и закодировать) множества назначений, которые ему будут регулярно нужны в работе.

#### 3.7.1. *Характеристики механизма шаблонов.*

- Шаблон назначений представляет собой множество назначений, которое врач может уточнить (пометить нужное или снять пометку с ненужного) в момент использования.
- В момент использования шаблона врач может специфицировать дату назначения, тогда все отмеченные назначения шаблона станут назначениями. Если врач не выберет дату, то назначения поместятся в лист назначений как набор формулировок (своего рода план), даты конкретных назначений врач уже определит в листе назначений путем проставления крестика в колонке дат.
- Три уровня шаблонов: общие для всего ЛПУ, общие для отделения, персональные. Определяются права доступа для конструирования шаблонов. Персональные шаблоны конструирует владелец, шаблоны отделения и ЛПУ в целом готовит ответственный специалист, а администратор системы устанавливает уровень.
- Для группировки шаблонов по тематике предусмотрен механизм папок.
- Для ускорения процесса создания шаблонов предусмотрено копирование шаблонов целиком и множества помеченных отдельных назначений.

3.7.2. *Решаемые задачи.* При использовании системы шаблонов эффективно удается решать следующие задачи:

- Создание рабочих множеств назначений для отделения, врача или специальности как по типам назначений (например, «анализы отделения хирургии») и нозологии (например, «назначения при диабете», «назначения при ИБС»), так и по конкретным регулярным событиям (например, «назначения при поступлении в отделение», «назначения после операции»).
- Создание стандартных наборов назначений для регулярных мероприятий (например, «исследования для плановой госпитализации»).
- Создание и использование комплексов назначений.
- Поддержка технологических карт лечения пациента, когда шаблон содержит все необходимые в данной карте назначения, а время назначения, очередность, кратность и т. д. определяет врач, исходя из стратегии и тактики лечения.
- Существенно облегчается решение задачи кодирования медицинских назначений, так как при создании шаблона пользователь обязывается максимально закодировать введенное назначение.

Практика показала, что грамотное использование механизма шаблонов позволяет существенно увеличить скорость работы врача в части ввода медицинских назначений даже по сравнению с бумажной технологией.

#### 4. Реализация и апробирование

Перечисленные в данной статье концептуальные и технологические решения были использованы для построения подсистемы назначений в информационной системе комплексного медицинского учреждения ИНТЕРИН. Основательность указанных решений подтвердило апробирование данной подсистемы на практике, в условиях эксплуатации в реальных крупных медицинских учреждениях. Получены положительные результаты и отклики персонала при работе подсистемы в условиях как стационара, так и поликлиники.

## Список литературы

- [1] Хаткевич М. И. *Объектно-реляционный дуализм в больших информационных системах* // Программные продукты и системы. — Т. 3, 2002, с. 22–26. ↑<sup>2</sup>
- [2] Малых В. Л., Пименов С. П., Хаткевич М. И. *Объектно-реляционный подход к созданию больших информационных систем* // Программные системы, Москва, Наука Физматлит, 1999, с. 177–184. ↑<sup>2</sup>

ИССЛЕДОВАТЕЛЬСКИЙ ЦЕНТР МЕДИЦИНСКОЙ ИНФОРМАТИКИ ИПС РАН

М. И. Khatkevich, Yu. I. Khatkevich. *Physician Order Subsystem of a Complex Medical Center. Development and Operation Experience.* (in russian.)

ABSTRACT. The paper outlines the key conceptual and technological solutions that arose from the several years experience of developing, implementing and maintaining a computerized physician order subsystem in a complex, multi-profile healthcare delivery institution. The basic demands on the subsystem are formulated, pertaining to its multiple aspects, functional complexity, general system requirements. The key conceptual and technological solutions for prescription formulation, basis, assignment, encoding and input are set out. Tools for prescription typification are offered. A prescription template mechanism is described, which considerably shortens the time to enter the orders into the information system.