

# ОГЛЯДИ ТА ДИСКУСІЇ

*Actori incumbit onus probandi!*

УДК 519+61:681.3

## Онтология клинической медицины: проблематика, задачи, решения

Ю.А. Прокопчук

*Украинский государственный химико-технологический университет, Днепропетровский областной диагностический центр, Днепропетровск, Украина*

### РЕЗЮМЕ, ABSTRACT

Рассмотрен базовый уровень онтологии клинической медицины. Предложен способ кодирования элементов конфигураторов тестов, участвующих в описании клинических ситуаций, и простой способ реализации вычислительных схем конфигураторов с использованием электронных таблиц. Выделены системно-когнитивные операции, которые обеспечивают реализацию метода предельных обобщений на основе онтологии. Все когнитивные операции используют предложенную схему для получения необходимого результата. Практика подтвердила доступность предлагаемой методики для широкого круга пользователей (Укр.журнал телемедицины и мед.телематики.-2010.-Т.8,№2.-С.214-220).

**Ключевые слова:** онтология, конфигураторы тестов, метод предельных обобщений, когнитивные операции

*Ю.О. Прокопчук*

### ОНТОЛОГІЯ КЛІНІЧНОЇ МЕДИЦИНИ: ПРОБЛЕМАТИКА, ЗАВДАННЯ, РІШЕННЯ

*Український державний хіміко-технологічний університет, Дніпропетровський обласний діагностичний центр, Дніпропетровськ, Україна*

Розглянуто базовий рівень онтології клінічної медицини. Запропоновано спосіб кодування елементів конфігураторів тестів, що беруть участь в описі клінічних ситуацій, і простий спосіб реалізації обчислювальних схем конфігураторів з використанням електронних таблиць. Виділено системно-когнітивні операції, які забезпечують реалізацію методу граничних узагальнень на основі онтології. Всі когнітивні операції використовують запропоновану схему для одержання необхідного результату. Практика підтвердила доступність запропонованої методики для широкого кола користувачів (Укр.журнал телемедицини та мед.телематики.-2010.-Т.8,№2.-С.214-220).

**Ключові слова:** онтологія, конфігуратори тестів, метод граничних узагальнень, когнітивні операції

*Iu. O.Prokopchuk*

### ONTOLOGY OF CLINICAL MEDICINE: PROBLEMS, TASKS, DECISIONS

*Ukrainian State University of Chemical Technology, Regional Diagnostic Center, Dnepropetrovsk, Ukraine*

The base level of the ontology of clinical medicine is considered. The structure and computing schemes of ontology information objects is proposed. Some operations of data processing are formalized (Ukr.z.telemed.med.telemat.-2010.-Vol.8,№2.-P.214-220).

**Key words:** ontology, tests configurators, method of limiting generalizations, cognitive operations

Прежде всего, необходимо дать ответы на три вопроса: что такое онтология, зачем нужна онтология для клинической медицины и кто должен (может) разрабатывать онтологию? Существуют разные определения онтологии [1], но в общем и целом суть этого

понятия таково. Нетривиальная онтология предметной области содержит: 1) систему понятий знаний; 2) систему понятий действительности; 3) отношение между этими двумя системами понятий; 4) инструменты для работы с информационными объектами

онтологии (имеется в виду компьютерная реализация онтологии). Система понятий знаний содержит определение терминов для описания знаний, определение объемов понятий, обозначенными этими терминами, а также определение связей между этими терминами – ограничения целостности знаний. Система понятий действительности содержит определение терминов для описания ситуаций (клинических ситуаций), определение объемов понятий, обозначенных этими терминами, а также определение связей между терминами – ограничения целостности ситуаций. Отношение между системой понятий знаний и системой понятий действительности задает связь между знаниями и действительностью. Онтологические соглашения содержат три группы соглашений: ограничения целостности знаний, ограничения целостности ситуаций и связи между знаниями и ситуациями. Программные инструменты должны отслеживать прежде всего непротиворечивость онтологии, должны обеспечить пользователей всем арсеналом диалоговых средств для всестороннего изучения объектов онтологии, анализа связей между объектами, а также средствами модификации онтологии.

Онтология клинической медицины позволит унифицировать способ представления медицинских знаний, установить содержательные связи и уменьшить объем материала, повысив при этом, его информативность, что должно способствовать повышению эффективности научных исследований и клинической практики, а также повышению квалификации специалистов. Структурирование информации в рамках онтологии и комплексный анализ формальными (математическими) методами позволяют выдвинуть принципиально новые гипотезы и получить новые результаты при решении медицинских задач там, где другие подходы оказываются неэффективными, несмотря на большой интерес исследователей [11]. Наконец, онтология клинической медицины будет служить фундаментом создания нового поколения госпитальных, телемедицинских, образовательных информационных систем - интеллектуальных медицинских систем и многоцелевых банков знаний [6 – 8, 10].

Для практического создания онтологии клинической медицины прежде всего должны быть разработаны общая концепция онтологии, архитектура онтологии и главное – базовый формализм описания основных информационных объектов медицины. В

этом плане определенная работа уже проделана [1,4,5,9,10], Следует отметить большой вклад в разработку общих вопросов построения онтологии клинической медицины группы специалистов Института автоматики и проблем управления ДВО РАН во главе с Клещевым А.С. Они не только много сделали и продолжают делать, но, что не менее важно, обеспечивают полный и открытый доступ к материалам своих работ через сеть Интернет. Наши исследования идут параллельно с исследованиями ИАПУ ДВО и подходы во многом отличаются. Мы двигались от практики, создавая длительное время госпитальные системы [6], и это наложило свой отпечаток на концепцию и процедурный уровень реализации онтологии [10].

Важной частью онтологии клинической медицины является формализованное описание лечебно-диагностических процессов как потоков работ (операций). В разработку этой части онтологии существенный вклад внесли работы московских ученых Назаренко Г.И. и Осипова Г.С. [2].

Мощным фундаментом для разработки онтологии клинической медицины являются работы большой когорты киевских ученых [3].

Процесс разработки онтологии (любого его фрагмента) – это захватывающий процесс. Для разработки онтологии нужна кооперация самых разных специалистов: медиков, математиков, биологов, химиков, физиков, лингвистов, философов и т.д. Только работая над фрагментом онтологии можно максимально четко выявить границы знания и незнания. Данная работа является приглашением всем медикам подключиться к разработке тех разделов онтологии, которые в наибольшей степени отвечают их интересам.

Настоящую работу следует рассматривать как продолжение публикаций [4, 5].

Целями настоящего исследования являются: 1) определение базового уровня онтологии клинической медицины; 2) разработка индуктивной модели обобщения результатов произвольного элементарного теста (конфигуратора теста); 3) разработка простой вычислительной схемы для реализации множества конфигураторов; 4) выделение и реализация фундаментальных системно-когнитивных операций, составляющих основу программного инструментария онтологии.

Ключевым положением предлагаемого формализма, лежащего в основе онтологии клинической медицины, является положение

о том, что любую ситуацию действительности, включая клиническую ситуацию, можно описать с помощью набора элементарных тестов [7]. Тест будем считать элементарным, если его результат можно представить в виде «Тест = значение» или «Тест? значение», например, «Возраст = 67», «АДс = 150», «Фамилия = Петров», «Головная боль? отсутствует» и т.д. Все дело, однако, в том, что значения одного и того же теста могут выбираться из разных множеств. Назовем эти множества *доменами* (термин заимствован из теории баз данных). Для фиксации того, что в качестве множества результатов теста  $\tau$  используется домен  $T$ , будем использовать нотацию  $\tau/T$ . Теперь результаты измерения, например, роста человека, могут быть представлены так: «Рост/ $T_1$  = 185», «Рост/ $T_2$  = [160; 190]», «Рост/ $T_3$  = высокий». Следует отметить, что у одного теста разных множеств - доменов может быть бесконечно много. В некотором смысле это хорошо, так как предоставляет полную свободу при формировании состава доменов в той или иной задаче.

Таким образом, любая ситуация действительности  $\alpha$  (например, клиническая ситуация) описывается набором элементарных тестов [4]:  $\alpha = \alpha\{\langle J\tau \tau/T, Jt t/\Lambda \rangle\}$ , где  $\tau/T$  – значение элементарного теста  $\tau$ , выбранное из домена  $T$ ;  $t/\Lambda$  - значение времени  $t$ , выбранное из домена  $\Lambda$ ;  $J$  - оператор оценки истинности значения теста. Конструкции  $\langle \tau/T, t/\Lambda \rangle$  и  $\langle \neg \tau/T, t/\Lambda \rangle$  определяют *элементарные события*. Конструкция  $\langle \tau/T, t/\Lambda \rangle$  определяет *составное событие*, а конструкция  $\langle \tau/T, \delta/\Lambda \rangle$  - *протяженное событие*, где  $\delta$  - временной интервал. Множество ситуаций действительности обозначим через  $\Omega = \{\alpha_1, \alpha_2, \dots, \alpha_n\}$ .

Оказалось, что множество доменов одного теста можно подобрать таким образом, что они строго упорядочиваются по уровню возрастания общности результата. Более того, в подобной иерархии можно задать однозначные правила пересчета значений из одного домена в другой - вышестоящий по уровню общности. Подобная иерархия доменов для любого элементарного теста получила название «конфигуратор теста». Конфигураторы тестов являются наименьшими квантами знаний, формирующими основу онтологии предметной области.

С прикладной точки зрения центральной проблемой является поиск наиболее эффективно и одновременно простого способа

представления произвольного конфигууратора. После длительного поиска методом проб и ошибок мы остановились на варианте, который описан ниже.

*Конфигураторы тестов.* *Базовым доменом* в конфигуураторе теста называется домен, с помощью которого описываются максимально точные значения теста. Дискретный домен максимальной общности называется *N-арным конструктором теста*, где  $N$  – число элементов домена. Примеры бинарных конструкторов: {Истина; Ложь}, {Норма; Отклонение}, {Эффективно; Неэффективно}, {Жалобы есть; Жалоб нет}, {Благоприятный; Неблагоприятный} и т.д. Для одного и того же теста может быть задано сколь угодно много  $N$ -арных конструкторов. Между базовым доменом теста и  $N$ -арным конструктором также может быть задано сколь угодно много промежуточных доменов. Внутри иерархии доменов задаются правила пересчета элементов одного домена в элементы домена более высокого уровня общности. Правила пересчета могут быть самыми разными: детерминированными, нечеткими, нейросетевыми и т.д. У любого теста может быть множество конфигуураторов, каждый из которых оптимизирован для решения конкретной задачи.

Доменная структура в рамках конфигууратора представляет собой ориентированный граф (в простейшем случае дерево). Общую схему конфигуураторов с использованием синтаксиса лексических деревьев [6, 7] можно представить следующим образом:

```
Тест [^ Тест...] [# ТестХ...] {
  Dom_1 [^Dom_1...] [#DomX...] { ; ; }
  Dom_2 [^Dom_2...] [#DomY...] { ; ; }
  ...
  Dom_N [^Dom_N...] { ; ; };
```

где 'Тест' – название теста; '^ Тест...' – список условных обозначений теста; '# ТестХ...' – список ссылок на более общие тесты; 'Dom\_K' – название K-го домена; '^Dom\_K...' – список условных обозначений K-го домена; '#DomX...' – ссылка на домены предки; { ; ; } – список элементов домена. Каждый элемент домена может иметь собственный список обозначений, которые также играют роль символов групп обобщения. Элементы доменов могут содержать параметры, которые обеспечивают однозначность вычислительных схем в зависимости от тех или иных факторов, например, пола.

Порядок размещения доменов в конфигуураторе – сверху вниз и слева направо – означает рост точности значений теста за

счет большей детализации (увеличения числа элементов). В упорядоченной последовательности доменов метки элементов домена явно задают однозначные правила перерасчета значений из текущего домена в другой,

размещенный выше или слева. Примеры конфигуракторов (рис. 1).

```

Индекс массы миокарда ЛЖ ^18 R ИММЛЖ {
3 { Норма ^0;
    Увеличение ИММЛЖ ^1 2 3}
2 { Норма ^0 M[30; 125] Ж[30; 110];
    Умеренное увеличение ИММЛЖ ^1 M (125; 150] Ж (110; 135];
    Значительное увеличение ИММЛЖ ^2 M (150; 200] Ж (135; 185];
    Резкое увеличение ИММЛЖ ^3 M (200; 400] Ж (185; 400]}
1 { [30; 400]}
Жалобы на боли в области живота {
D2 {Болей в животе нет (норма) ^N; Жалобы на боли в животе ^b}
D1 {Болей в животе нет (норма) ^N; Постоянные ноющие боли в правом подреберье
^b; Схваткообразные боли в правом подреберье ^b; Опоясывающие боли в эпигастрии
^b; Боли в левом подреберье ^b}}
Возраст {
5 #2 {молодой ^a; не молодой ^b c}
4 #2 {средних лет^b; не средних лет^a c}
3 {пожилой ^c; не пожилой ^a b}
2 {молодой ^a [18;35]; средних лет ^b [36;55]; пожилой ^c [56;65]}
1 {[18; 65]}

```

Рисунок 1. Примеры конфигуракторов

Приведенное выше определение конфигурактора допускает значительное разнообразие в описаниях. Опытным путем установлено, что минимум ошибок пользователи делают в том случае, если и названия доменов, и алиасы элементов доменов кодируются числами, а не так, как в конфигураторе теста «Жалобы на боли в области живота». С целью обеспечения максимальной доступности метода для широкой аудитории пользователей вычислительные схемы, заложенные в конфигуляторах тестов, предложено реализовывать с помощью электронной таблицы так, как показано на рисунке 2. В представленном примере задействовано 3 теста. Видно, что все конфигураторы

тестов линейные, т.е. не содержат ветвлений. После подстановки первичных данных (значений тестов для конкретной клинической ситуации) в ячейки таблицы D2, D6, D8, с помощью встроенных функций Excel рассчитываются значения тестов более высоких уровней общности. Ниже для примера приводится формула из ячейки D4, которая описывает правило пересчета значений из домена 2 в домен 3 теста №2:

=ЕСЛИ(ИЛИ(D3=1;D3=2);1;ЕСЛИ(ИЛИ(D3=3;D3=4);2;ЕСЛИ(D3=0;0))).

База с первичными данными расположена на отдельном листе электронной таблицы.

	A	B	C	D
1	Тест	Домен 1	Домен 2	Значение
2	1	1	1	23,68
3	1	1	2	1
4	1	2	3	1
5	1	3	4	1
6	2	1	1	0
7	2	1	2	0
8	3	1	1	0
9	3	1	2	0

Рисунок 2. Реализация конфигуракторов трех тестов

Для числовых тестов конфигуратор кроме лингвистической иерархии может включать иерархию в виде вложенных разбиений или физического фрактала типа «Канторовская пыль». Иерархия строится между базовым доменом и первым дискретным разбиением (в конфигураторе задается тип разбиения/фрактала и его параметры - скейлинг). Другими словами, конфигураторы числовых тестов не имеют в основании неделимых элементов или «атомов простоты», демонстрируя бесконечность свойства и отсутствие предела процессов делимости материи и информации. Так между доменами 1 и 2 теста ЧИММЛЖ могут быть автоматически сгенерированы дополнительные домены по типу вложенных разбиений/фрактала. Дополнительные домены генерируются программным путем.

Таким образом, конфигураторы стирают границу между непрерывным и дискретным – любой непрерывный числовой тест всегда имеет множественное дискретное представление (интервальное, символическое).

*Задачи, решаемые с помощью конфигураторов тестов.* Различные комбинации доменов для всех тестов определяют различные уровни общности описания клинических ситуаций. Все множество описаний для конкретной целевой задачи (связанной с диагностикой, прогнозированием или оптимизацией лечения) и заданной выборке примеров клинических ситуаций  $\Omega$  распадается на три подмножества: критических, докритических и закритических описаний. Критические описания являются наиболее ценными, так как обеспечивают единственность решения целевой задачи на заданной выборке клинических ситуаций  $\Omega$  и обладают при этом предельными свойствами: их нельзя обобщить ни по одному тесту в рамках заданных конфигураторов без нарушения единственности решения целевой задачи. Докритические описания также обеспечивают единственность решения на  $\Omega$ , но допускают подобное обобщение. Закритические описания нарушают единственность решения целевой задачи. Для критических описаний предпринимается попытка построения истинных минимальных избыточных моделей знаний (множества закономерностей), т.е. таких моделей, которые верны для любых клинических ситуаций данной предметной области (ПрО).

Решение целевой задачи для новой клинической ситуации, для которой априорно

неизвестно заключение, формулируется следующим образом: исходные данные преобразуются в формат критического описания, для которого предположительно построена истинная избыточная модель знаний. С использованием истинной модели знаний находится решение. Данный метод получил название «Метод предельных обобщений» (МПО) [4, 9].

Конфигураторы тестов и приведенная на рисунке схема позволяют максимально просто реализовать следующие системно-когнитивные операции в рамках программного инструментария МПО и онтологии клинической медицины:

- преобразование базового описания клинических ситуаций  $\Omega_0$  (на основе максимально точных доменов) в любое допустимое описание, которое задается списком доменов для каждого элементарного теста (возможно, каждого вхождения элементарного теста);

- вычисление общего числа описаний, числа критических описаний, числа докритических описаний и числа закритических описаний (задано  $\Omega_0$  и конфигураторы);

- определение принадлежности заданного описания  $\Omega$  к одному из множеств: докритических, критических или закритических описаний;

- формирование списка критических описаний (задано  $\Omega_0$  и конфигураторы), каждый элемент списка задается набором доменов;

- формирование полного списка избыточных закономерностей для заданного докритического или критического описания;

- формирование оптимального списка закономерностей для заданного критического описания (по любому из критериев оптимальности: минимальный список; минимальный список с максимальным рейтингом; список с минимальным числом тестов; список с тестами минимального уровня сложности и т.д.);

- построение минимальной избыточной модели знаний для заданного критического описания;

- определение устойчивости состава оптимальной избыточной модели знаний при варьировании параметров конфигураторов;

- определение устойчивости заключения для новой клинической ситуации при варьировании параметров конфигураторов;
- генерация различных моделей конфигураторов тестов при заданных параметрах варибельности (для задач анализа устойчивости);
- для заданного теста (базовый домен которого является числовым непрерывным интервалом) генерация вложенного разбиения или физического фрактала заданной глубины (согласно параметрам скейлинга);
- реализация непрерывных, интервальных, нечетких, нейросетевых математических операций, в которых участвуют значения тестов, выбираемых из конфигураторов;
- построение областей пересечения списков избыточных закономерностей для заданных критических описаний (возможно, всех критических описаний) – формирование ядра закономерностей;
- построение минимальной избыточной модели знаний для ядра закономерностей;
- формирование предпочтений над списком критических описаний в плане целесообразности попытки построения истинных оптимальных избыточных моделей знаний (критические описания

ранжируются в соответствии с уровнем предпочтения);

- построение паретовского множества наборов доменов для заданного заключения с использованием всех типов описаний, включая закритические (для закритических описаний требуется, чтобы для каждого набора доменов множество ситуаций действительности, отвечающее заданному заключению, не содержало артефактов, т.е. не было ситуаций, которые относились бы и к другим заключениям);

- фальсификация заданной избыточной закономерности (добавление новой клинической ситуации, гарантированно фальсифицирующей заданную закономерность).

Перечисленные операции кодируются с помощью VBA. Для реализации операций анализа устойчивости параметры конфигулятора, например, границы возрастных интервалов – 35 и 55 лет, хранятся в отдельных ячейках и, следовательно, легко могут модифицироваться в соответствии с заданным алгоритмом.

Пример применения онтологии для оценки и прогнозирования характера течения заболевания описан в [11].

## Выводы

Рассмотрен базовый уровень онтологии клинической медицины. Предложен способ кодирования элементов конфигураторов тестов, участвующих в описании клинических ситуаций, и простой способ реализации вычислительных схем конфигураторов с использованием электронных таблиц. Выделены системно-когнитивные операции, кото-

рые обеспечивают реализацию метода предельных обобщений на основе онтологии. Все когнитивные операции используют предложенную схему для получения необходимого результата. Практика подтвердила доступность предлагаемой методики для широкого круга пользователей.

## Литература и библиография

1. Клецев А.С., Артемьева И.Л. Математические модели онтологий предметных областей. Часть 1. Существующие подходы к определению понятия «онтология» // НТИ. Сер. 2, 2001, № 2. – С. 20-27.
2. Назаренко Г.И., Осипов Г.С. Основы теории медицинских технологических процессов (в двух частях). – М.: ФИЗМАТЛИТ, 2006.
3. Грищенко В.И., Вовк М.И., Котова А.Б., Белов В.М., Минцер О.П., Кифоренко С.И., Онопчук Ю.Н., Козак Л.М., Ермакова И.И. Биоэкомедицина: единое информационное пространство – К.: Наукова думка, 2001. – 318 с.
4. Прокопчук Ю.А. Применение формального анализа понятий и метода предельных обобщений в задачах обработки медицинских данных и знаний // Укр. ж-л телемедицины та медичної телематики. – 2009. – Т.7, №2. – С.32-36.

5. Прокопчук Ю.А., Белецкий А.С. Проблемы интеллектуализации госпитальных и телемедицинских систем // Укр.ж. телемедицины та медичної телематики. – 2008. – Т.6, №3. – С.244-250.
6. Алпатов А.П., Прокопчук Ю.А., Костра В.В. Госпитальные информационные системы: архитектура, модели, решения. – Днепропетровск: УГХТУ, 2005.- 257 с.
7. Прокопчук Ю.А. Интеллектуальные медицинские системы: формально-логический уровень. – Дн-ск: ИТМ НАНУ и НКАУ, 2007.- 259 с.
8. Информационные технологии в образовании и здравоохранении / А. П. Алпатов, Ю. А. Прокопчук, О. В. Юденко, С. В. Хорошилов. – Дн-ск : ИТМ НАНУ, 2008. - 287 с.
9. Прокопчук Ю.А. Метод предельных обобщений – эффективный принцип работы вычислительного интеллекта // Искусственный интеллект, 2008. - №3.- С. 727 – 736.

10. Прокопчук Ю.А. Архитектура многоцелевых банков знаний в области клинической медицины // Сборник Трудов Всероссийской конференции "Знания-Онтологии-Теории"-2009 в 2-х томах (Новосибирск, 22-24 октября 2009 г.). - Новосибирск: ЗАО "РИЦ Прайс-Курьер", Т.1. - С. 173-177.

11. Прокопчук Ю.А., Сергиени Е.В., Харченко О.А., Татьяненко А.В., Челашеев В.А. Оценка течения заболевания на

основе метода предельных обобщений // Сборник докладов VII – й Международной научно – практической конференции «Математическое и программное обеспечение интеллектуальных систем» (Днепропетровск, 25-27 ноября 2009г.). – Дн-ск: Из-во ДНУ, 2009. – С. 235 – 236.

Надійшла до редакції: 21.02.2009.

© Ю.А. Прокопчук

---

Кореспонденція: Прокопчук Ю.А.,  
пр-т Гагаріна, 8, 49005, Дніпропетровськ, Україна  
E-mail: itk3@ukr.net