



OSTIS-2014

(Open Semantic Technologies for Intelligent Systems)

УДК 004.822:514

СОДЕРЖАНИЕ СИСТЕМНОГО АНАЛИЗА ПРИ АВТОМАТИЗАЦИИ ИНТЕЛЛЕКТУАЛЬНОЙ ДЕЯТЕЛЬНОСТИ НА УРОВНЕ ОТРАСЛИ

Клещев А.С., Шалфеева Е.А.

*Институт автоматизации и процессов управления,
г. Владивосток, Россия*

kleshev@iacp.dvo.ru

shalf@iacp.dvo.ru

Работа посвящена особенностям выполнения, содержанию и результатам системного анализа в составе комплексной технологии автоматизации интеллектуальной деятельности на уровне отрасли. Выполнение всех представленных этапов и получение соответствующих результатов позволяют осуществить поддержку принятия решений на основе знаний и систематическое управление качеством этих знаний. Выполнение такого системного анализа гарантирует семантическую совместимость создаваемых интеллектуальных систем.

Ключевые слова: предметная область; интеллектуальные задачи; системный анализ; управление знаниями.

Введение

Анализ организации повседневной интеллектуальной деятельности и управления ее качеством показал необходимость перехода к новой *парадигме автоматизации* - автоматизации отрасли в целом, т.е. разработке единой для отрасли «облачной» интеллектуальной программно-информационной системы для выполнения всех задач принятия решений на основе знаний и всех задач управления качеством знаний [Клещев и др., 2013].

Ключевыми аспектами этой парадигмы являются: *понятные* специалистам *базы знаний* (БЗ); *полезные* БЗ в течение всего времени их эксплуатации; *возможность включения новых научных* результатов в БЗ; *автоматическое накопление результатов верификации* решений в базе прецедентов для индуктивного формирования и отладки БЗ.

Разработано множество экспертных систем (ЭС) и систем поддержки интеллектуальной деятельности в разных предметных областях. Выполняемый при их создании *системный анализ* (СА) позволяет формировать в таких системах базы декларативных знаний как самостоятельный компонент. Признается необходимость их *развития* в процессе эксплуатации ЭС, в том числе *возможность включения новых научных* результатов в БЗ. Иногда осуществляется автоматическое накопление результатов верификации решений в

базе прецедентов, выявление в них закономерностей и индуктивное обобщение примеров [Witten, 2005; Поспелов, 1987; Черняховская, 2011].

Однако за прошедшие десятилетия, несмотря на значительные успехи в области теории и технологии создания, ЭС так и не начали внедряться в повседневную практику на уровне целой отрасли.

Альтернативой «классическим» экспертным системам являются системы поддержки интеллектуальной деятельности, основанные на «owl-ориентированных» базах знаний. Такие базы знаний формируются в терминах онтологий соответствующих предметных областей, что позволяет *знаниям быть понятными* специалистам и экспертам предметных областей. Наиболее популярным инструментом их построения является система Protégé. В качестве примеров применения таких знаний в литературе удается найти, в основном, использование их для решения задач классификации (они и признаются наиболее распространенными [Бениаминов, 2008]). Не удается найти информацию о накоплении верифицированных решений задач и методах их использования для обновления БЗ Protégé.

Отсутствует технология такой автоматизации поддержки принятия решений в отрасли, которая бы конструктивно решала *проблему правильности применения знаний* при решении задач интеллектуальной деятельности. Не предложены

механизмы непрерывного усовершенствования декларативных баз знаний.

Новый подход к автоматизации повседневной интеллектуальной деятельности и управления ее качеством возможен *с использованием онтологий предметных областей* [Клещев и др., 2013]. Подход требует детальной разработки всех тех этапов, от которых зависит решение проблемы *поддержки интеллектуальной деятельности и правильности применения знаний* при этом. Первый из них - этап *системного анализа*.

Цель настоящей работы – обсудить особенности выполнения и содержание результатов *системного анализа*, позволяющего осуществить автоматизацию повседневной интеллектуальной деятельности и управления качеством знаний на уровне отрасли.

1. Начальные этапы автоматизации интеллектуальной деятельности отрасли

Деятельность в предметной области может быть рассмотрена как [Клещев и др., 2013]:

- *повседневная*, в том числе интеллектуальная (т.е. применение знаний при решении интеллектуальных задач);
- *контроль принимаемых решений* (электронный документооборот, включающий принятые специалистами решения и результаты верификации этих решений);
- *деятельность по управлению знаниями*, т.е. меры по уточнению, расширению и отладке знаний.

Системный анализ должен обеспечивать поддержку всех этих трех видов деятельности. Вопросы автоматизации *повседневной* неинтеллектуальной деятельности в этом исследовании не рассматриваются, а будут рассмотрены этапы, связанные с *базами знаний*.

Поскольку роль *баз знаний* и способ их организации отличаются от других хранилищ данных, они создаются на базе *онтологии предметной области*, которая должна быть построена на этом «раннем этапе». Назовем здесь *инженерией баз знаний* этап, состоящий из *формирования* каждой базы знаний и управления качеством (сопровождения) каждой базы.

Целесообразно различать роль и результаты системного анализа от концептуального проектирования системы и инженерии ее баз знаний.

Системный анализ состоит из нескольких шагов:

- идентификация интеллектуальных задач в предметной области;
- разработка онтологий для каждой такой задачи;
- постановка каждой интеллектуальной задачи;

- разработка или выбор методов решения и соответствующих им алгоритмов для интеллектуальных задач.

По завершении этих работ возможен переход, во-первых, к *концептуальному проектированию системы автоматизации*, во-вторых, к *инженерии баз знаний*. Следует отметить, что переход к *инженерии баз знаний* возможен только после реализации ряда подсистем для инженерии знаний, выделенных при *концептуальном проектировании*.

Концептуальное проектирование системы состоит из шагов:

- идентификация (сколько и каких потребуется) программных подсистем для автоматизации повседневной «обычной» и для повседневной интеллектуальной деятельности;
- идентификация баз знаний;
- идентификация программных подсистем (подсистем-редакторов, подсистем-индуктивного формирования знаний, подсистем проверки соответствия знаний базе прецедентов) для *инженерии баз знаний*;
- идентификация программных подсистем для обучения (и оценивания) принятию решений;
- построение схемы взаимосвязи всех подсистем в единой системе.

Инженерия базы знаний состоит из шагов:

- формирование и проверка качества первого варианта базы знаний;
- расширение базы знаний и проверка качества нового варианта;
- слежение за появлением в повседневной деятельности такого прецедента, для которого база знаний оказалась недостаточна;
- приведение базы знаний в соответствие имеющимся прецедентам.

2. Этап онтологического инжиниринга

При автоматизации интеллектуальной деятельности на уровне отрасли в качестве *системы* обсуждается *предметная область*, в которой имеются сложные *реальные объекты, процессы и явления*, и над этими объектами специалистами решаются (и требуют поддержки) *интеллектуальные задачи*. Вариант *системного анализа* для разработки интеллектуальных программных систем часто называют *онтологическим инжинирингом*.

2.1. Идентификация интеллектуальных задач

В результате проведения типичного анализа повседневной профессиональной деятельности, подлежащей автоматизации, системные аналитики выявляют виды работ, связанных с обработкой и преобразованием информации, последовательность их выполнения, связь работ по информации и, возможно, зависимость их от хранимой информации.

В случае анализа повседневной интеллектуальной деятельности в такой «схеме задач» обязательно появятся интеллектуальные задачи, их зависимость от типов профессиональных знаний, связи задач по данным (как в примере на рис. 1).

Такая модель может быть представлена на языке графов, что способствует и пониманию, и дальнейшему планированию СА.

2.2. Разработка онтологий

Интеллектуальная задача решается специалистом по отношению к (сложно устроенному) объекту действительности, и при ее решении необходимы профессиональные знания. Поэтому типовая схема моделирования одной произвольной интеллектуальной задачи начинается с шага построения *онтологии предметной области*, охватывающей все профессиональные понятия, используемые при решении повседневных задач [Клещев, 2008].

Онтология предметной области включает структуру вербального описания исследуемого и/или управляемого объекта (например, пациента в медицине) и знаний для принятия решений. Другими словами, строятся *онтология действительности* и *онтология знаний*, связываемые *онтологическими соглашениями* между ними [Клещев, 2008].

Творческому процессу выявления онтологий, методологии которого посвящено немало работ [Клещев, 2008; Загоруйко, 2008; Gavrilova, 2007], сопутствует рутинный («технический») процесс их явного формализованного представления. На сегодняшний день для этого имеются универсальные редакторы, такие как Protégé или двухуровневый редактор *иерархических семантических сетей* (ИСС) [Клещев, 2006]. Следует отметить, что представление структуры профессиональных понятий «сверху вниз» через операции соединения, повторения и выбора, как это традиционно делается при создании словарей данных предметной области в технологии разработки программного обеспечения [Pressman, 2001], представляется наиболее естественным (по сравнению с построением сущностей предметной

области из элементов и отношений, как это делается на owl-языке). Формирование информации для медицинских задач в виде иерархий используется уже давно (the hierarchy of the patient model) [Mittal et al, 1979].

Применение редактора ИСС позволяет отдельно создавать и сохранять все онтологии действительности и онтологии знаний, необходимые для рассматриваемой интеллектуальной задачи. В частности, для задачи медицинской диагностики потребуется создать одну онтологию действительности (структуру истории болезни пациента) и более одной онтологии знаний (структура описания клинических проявлений заболеваний, структура описания знаний о воздействующих факторах).

2.3. Постановка задач

В типовой схеме моделирования одной интеллектуальной задачи после построения онтологии предметной области переходят к *постановке задачи* в терминах онтологии. В случае формирования *постановки интеллектуальной задачи* имеются две отличительные особенности.

Одна - явное указание используемых знаний. При этом в постановке задачи, скорее, будут указаны не сами базы, а онтологии этих баз знаний, поскольку профессиональные знания в отрасли с интеллектуальной деятельностью постоянно совершенствуются.

Вторая особенность – формат выходных данных. Поскольку при автоматизации интеллектуальной деятельности не требуется (как правило) получение единственного решения, но требуется ограниченное множество гипотез о решении вместе с объяснением каждой, то формат выходных данных формируется на основе онтологий действительности, и знаний.

Пример структуры постановки задачи медицинской диагностики:

входные данные – заполненная частично *история болезни* (формат - структура ИБ для заполнения паспортных данных, анатомо-физиологических особенностей пациента, жалоб и ведения дневника наблюдений и обследований);

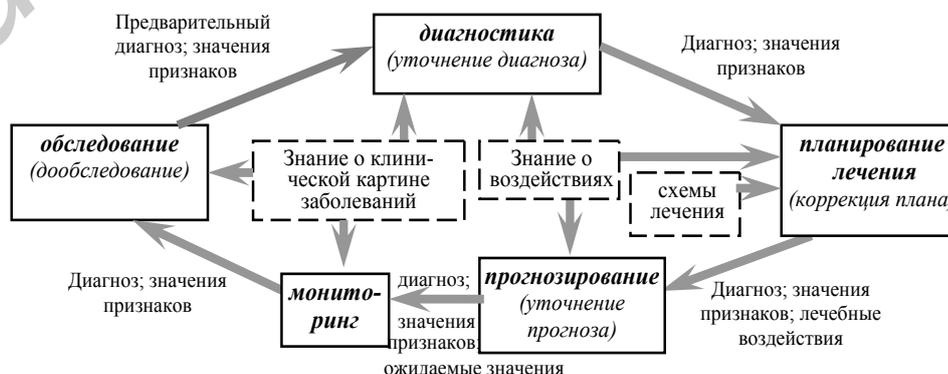


Рисунок 1 – Пример схемы задач для медицинской повседневной деятельности

входные знания - знания о наблюдениях, их возможных и нормальных значениях (формат - структура описания значений признаков при отсутствии заболеваний), знания о клинических проявлениях заболеваний (формат - структура описания вариантов проявлений и периодов развития), знания о воздействующих факторах (формат - структура описания изменения значений признаков при воздействии фактора);

результат - альтернативные диагнозы пациента с их объяснениями (формат каждого объяснения включает название заболевания и объяснение всех наблюдаемых значений каждого признака, подтверждающего клиническую картину заболевания).

Применение редактора ИСС позволяет создать и сохранить онтологию результата, предлагающую необходимый уровень детализации объяснения для понимания специалистом выбранной гипотезы.

2.4. Разработка или выбор методов решения и алгоритмов

Традиционно после того, как выполнена постановка интеллектуальной задачи, осуществляется разработка метода ее решения и доказательство его корректности.

Существенное отличие СА при автоматизации интеллектуальной деятельности состоит в обязательности этого шага, поскольку без гарантий существования реализуемого метода автоматизация решения интеллектуальной задачи нецелесообразна. (В сравнении с этим подбор методов для решения не-интеллектуальной задачи менее сложен и может быть выполнен на более позднем этапе).

В частном случае может быть создан уникальный алгоритм, входные и выходные данные которого задаются в терминах *онтологии действительности*, а составляющие шаги могут использовать термины *онтологии знаний* и проверять выполнение *онтологических соглашений*. Тем не менее, в свое время было предложено несколько общих методов для задач (Taxonomic Classification, Hypothesis Matching и др.) и общий подход декомпозиции задачи на подзадачи, для которых применимы известные методы или дальнейшая декомпозиция [Chandrasekaran, 1992; Gennari et al, 2003]).

Другой опыт дает основания предполагать, что алгоритм решения задачи, для которого доказана корректность, (например, алгоритм медицинской диагностики [Москаленко, 2006]) может быть обобщен до целого класса подобных задач, не зависимо от предметной области.

3. Концептуальное проектирование подсистем для инженерии знаний

При автоматизации деятельности (с использованием знаний) важно обеспечить необходимые средства поддержки для тех, кто

знаниями управляет. Из вышеперечисленных шести этапов концептуального проектирования системы непосредственно к этому имеют отношение два: идентификация баз знаний и идентификация программных подсистем для инженерии знаний.

3.1. Идентификация баз знаний

После того, как в рамках онтологического инжиниринга построены онтология всех знаний для всех задач, можно провести анализ их зависимостей и предложить оптимальную модуляризацию онтологий и баз знаний. Например, в медицине может быть принято решение о соединении (в одной базе) знаний о нормальных признаках пациента со знаниями о клинических картинах заболеваний или, наоборот, об их разделении; может быть принято решение об отделении знаний о возможных значениях всех наблюдений, а также и о возможных названиях самих этих наблюдений в самостоятельную мета-базу, используемую при формировании целевых баз.

Результатом становится решение о том, сколько и каких БЗ должно быть создано в системе, чтобы запланировать для них средства создания, использования и управления.

3.2. Идентификация подсистем-редакторов для формирования каждой базы знаний

Формирование первого варианта базы знаний обычно осуществляется с помощью тех же средств, что и последующее расширение БЗ, в частности, добавление *новых научных* результатов.

Современные инструменты позволяют строить сложные композиции понятий, соответствующие сложно устроенным сущностям и обеспечивать при этом «читабельные» знания [Rogers, 2004]. Распространены универсальные инструменты, например, Protégé, пригодные для формирования любых знаний, представленных как представители классов с конкретными слотами. В их методологии после построения *domain-онтологии* предусмотрена генерация по ней инструмента приобретения знаний, далее – построение с его помощью БЗ и затем – интеграция всех этих компонентов в *knowledge-based system* [Gennari et al, 2003].

Для приближенного к реальности вышеупомянутого представления знаний «сверху вниз» удобен редактор ИСС, который (как и Protégé) автоматически формирует интерфейс редактора знаний после того, как зафиксирована онтология этих знаний с помощью этого же редактора.

Выбор в качестве формы представления знаний ИСС, а в качестве инструмента – соответствующего редактора, позволяет описывать простые и составные признаки, связи между неисправностями (для диагностики), изменчивость признаков во времени и другие варианты сложно устроенных знаний предметной области.

3.3. Идентификация средств программного доступа к содержимому баз знаний

От представления баз знаний зависит и способ доступа к ним, поэтому в концептуальном проекте системы автоматизации уместен «слой» доступа к знаниям, состоящий из операций или запросов достаточно абстрактных. Для доступа к знаниям, хранимым в Protégé, например, импортируют их из Protégé в системы Drools, Eclipse и создают производные правила, работающие с «экземплярами как с фактами» [Глибовец, 2012]. Для доступа к каждой БЗ в виде ИСС удобнее всего была бы своя оболочка, реализующая все типы запросов из всех приложений или подсистем, обращающихся к БЗ. Уже на основе анализа разработанных алгоритмов такой набор запросов может быть составлен.

3.4. Идентификация программных подсистем для управления качеством знаний

Построение первого варианта БЗ, предназначенного для использования в реальной повседневной деятельности специалистов, должно быть основано на методах обеспечения качества, и не может (в частности, в таких областях, как медицина) быть проверенным только вручную.

Дальнейшее *управление знаниями* можно представить так. Обнаруживается задача, для правильной поддержки решения которой знаний в системе оказалось недостаточно. Найденное другим способом решение становится *прецедентом для обучения БЗ* и дополнительным тестом для проверки ее качества.

Но для формирования обучающей выборки может быть недостаточно одного прецедента, поэтому в системе *управления знаниями* может быть запланирована подсистема – «рабочее место» эксперта для поддержки анализа верифицированных решений и отбора прецедентов. Требуется и подсистема для проверки соответствия очередного варианта базы знаний - базе прецедентов; такая подсистема нужна и для контроля добавленных вручную знаний.

Для получения нового варианта базы знаний по обучающей выборке разработан метод индуктивного формирования знаний на основе анализа отдельных представителей, описанных единообразно в терминах онтологии [Kleschev et al, 2012] и опытный инструментарий [Смагин, 2013]. Предшественниками таких средств можно считать систему ФИАКР [Соловьев, 1996], интеллектуальные системы типа ДСМ с анализом данных (knowledge discovery) и автоматическим расширением БЗ [Финн, 1999]. Но эти методы относятся к созданию систем, основанных на правилах или на прецедентах.

4. О концептуальном проектировании подсистем поддержки принятия решений

При автоматизации профессиональной деятельности важно обеспечить необходимые средства поддержки специалистов, принимающих решения на основе *управляемых* знаний. Каждая идентифицированная программная подсистема для поддержки интеллектуальной деятельности соответствует ранее созданной постановке задачи, реализует выбранный алгоритм и обращается при этом к подсистемам («слою» или оболочке) доступа к необходимым базам знаний.

Наличие разработанного метода и алгоритма решения задачи обычно означает реализуемость подсистемы поддержки принятия решений. В процессе концептуального проектирования подсистем проясняются их внешние интерфейсы и параллельно могут обсуждаться и формироваться все виды требований к каждой подсистеме, на основании которых начнется разработка.

Поддержка повседневной деятельности соединена с документооборотом. Этапы идентификации программных подсистем для автоматизации повседневной «обычной» и интеллектуальной деятельности могут проводиться независимо, но очевидно, что в общей модели системы они будут связаны информационными потоками или хранилищами, отличными от БЗ.

Заключение

Возможность автоматизации интеллектуальной деятельности на уровне отрасли зависит от наличия методов и инструментов управления качеством знаний. Решающая роль принадлежит результатам *онтологического инжиниринга*, на основе которых осуществляется проектирование семантически совместимых подсистем решения задач и управления качеством баз знаний и их интеграция.

Предлагаемые для представления знаний и их онтологий иерархические семантические сети не только наиболее удобны для работы специалистов с информацией, но и перспективны с точки зрения управления качеством таких знаний и доступа к ним в процессе поддержки решения задач.

Этот подход позволяет внедрить новые механизмы управления интеллектуальной деятельностью, согласованные с существующими в отраслях механизмами. Такое более эффективное управление возможно только при разработке единого комплекса интеллектуальных программ и сопряженных с ними программ для управления качеством знаний.

Работа выполнена при финансовой поддержке грантов РФФИ № 12-07-00179-а «Интернет-сервисы для преобразования семантических сетей» и ДВО РАН № 12-1-П15-03 «Управление интеллектуальными системами».

Библиографический список

[Бениаминов, 2008] Бениаминов Е.М. Некоторые проблемы широкого внедрения онтологий в ИТ и направления их решений. // Труды Симпозиума "онтологическое моделирование". М.: ИПИ РАН, 2008, с. 71-82.

[Глибовец, 2012] Глибовец Н.Н., Красиков Д.С. Интеллектуализация экспертных систем с помощью онтологий // Information science and computing. Book 28. "Problems of Computer Intellectualization", 2012. 84-90. http://www.foibg.com/ibs_isc/ibs-28/ibs-28-p10.pdf.

[Загорюлько, 2008] Загорюлько Ю.А. Методы и методологии разработки, сопровождения и реинжиниринга онтологий: труды симпозиума «Онтологическое моделирование». М.: ИПИ РАН, 2008.

[Клещев, 2006] Клещев А.С., Орлов В.А. Компьютерные банки знаний. Универсальный подход к решению проблемы редактирования информации // Информационные технологии. - М.: Новые технологии. - 2006. - №5. - С. 25-31.

[Клещев, 2008] Клещев А.С. Роль онтологий в программировании. Часть 1. Аналитика // Информационные технологии. - М.: Новые технологии. - 2008. - №10. - С. 42-46.

[Клещев и др., 2013] Клещев А.С., Черняховская М.Ю., Шалфеева Е.А. Парадигма автоматизации интеллектуальной профессиональной деятельности. Часть 1. Особенности интеллектуальной профессиональной деятельности // Онтология проектирования. Самара: "Новая техника", 2013. № 3(9). 53-69.

[Москаленко, 2006] Москаленко Ф.М. Алгоритм диагностики, основанный на реальной онтологии медицины, для многопроцессорной ЭВМ // Доклад III Междуна. конф. "Расо 2006". М.: Институт проблем управления им. В.А. Трапезникова РАН, 2006.

[Поспелов, 1987] Поспелов Д.А. Послесловие / Горелов И.Н. Разговор с компьютером. Психолингвистический аспект проблемы. М.: Наука, 1987.

[Смагин, 2013] Клещев А.С., Смагин С. В. Комплекс программ для индуктивного формирования баз медицинских знаний в форме, принятой в медицинской литературе // Материалы XX Международной научной конференции студентов, аспирантов и молодых ученых: секция «Вычислительная математика и кибернетика»; МГУ, 9-12 апреля. М.: Издательский отдел факультета ВМК МГУ, 2013. С. 36-37.

[Соловьев, 1996] Соловьев С. Ю. Методы отладки баз знаний в системе ФИАКР. 1996. http://www.park.glossary.ru/serios/read_053.php#p35;

[Финн, 1999] Финн В.К. Синтез познавательных процедур и проблема индукции // НТИ. Сер.2.-1999.- №1-2 С.8-44.

[Черняховская, 2011] Черняховская Л. Р., Федорова Н. И., Низамутдинова Р. И. // Интеллектуальная поддержка принятия решений в оперативном управлении деловыми процессами предприятия / Вестник УГАТУ: научн. журнал Уфимск. гос. авиацион. техн. ун-та, 2011. Т. 15, № 2 (42). С. 172–176.

[Chandrasekaran, 1992] B. Chandrasekaran, T. Johnson, and J.W. Smith, "Task Structure Analysis for Knowledge Modeling," Communications of the ACM, Vol. 33, No. 9, September 1992, pp. 124-136; anthologized in Knowledge Oriented Software Design, ed., J. Cuenca, North-Holland (Amsterdam), 1993, pp. 1-22.

[Gavrilova, 2007] Gavrilova T. Ontological Engineering for Practical Knowledge Work // Lecture Notes in Artificial Intelligence 4693, Proc. of 11th Int. Conf. Knowledge-Based Intelligent Information and Engineering Systems (KES 2007), Italy, Vietri sul Mare, Springer, 2007. pp. 1154–1162.

[Gennari et al, 2003] Gennari J.H., Musen M.A., Fergerson R.W., Grossod W.E., Crubezy M., Eriksson H., Noy N. F., Samson W. The evolution of Protégé: an environment for knowledge-based systems development, International Journal of Human-Computer Studies, 2003. 89-123.

[Kleschev et al, 2012] Kleschev A. S., Smagin S. V. Problems of inductive formation of knowledge in the ontology of medical diagnosis // Automatic Documentation and Mathematical Linguistics. 2012. Vol. 46. Number 1. P. 8-21.

[Mittal et al, 1979] Mittal S., Chandrasekaran B. Smith J. "Overview of MDX - a medical diagnosis system," Proc. III Annual Symposium on Computer Applications in Medical Care, Washington, D.C., October 1979.

[Pressman, 2001] Pressman R.S. Software Engineering: Practitioner's Approach. Fifth edition. McGraw-Hill Inc., 2001. 860 p.

[Rogers, 2004] Rogers J. E. Development of a methodology and an ontological schema for medical terminology. 2004

[http://www.opengalen.org/download/2004-Rogers-Methodology-and-ontological-schema-for-medical-terminology-\(MD-Thesis\).pdf](http://www.opengalen.org/download/2004-Rogers-Methodology-and-ontological-schema-for-medical-terminology-(MD-Thesis).pdf)

[Witten, 2005] Witten, I. H. & Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques*, second edition, Elsevier: San Francisco, 525 p. <http://home.etf.rs/~vm/os/dmsw/Morgan.Kaufman.Publishers.Weka.2nd.Edition.2005.Elsevier.pdf>.

SYSTEM ANALYSIS CONTENTS FOR INTELLIGENT ACTIVITY AUTOMATION AT BRANCH LEVEL

Kleschev A. , Shalfeeva E.

The Institute of Automation and Control Processes, Vladivostok, Russia

kleschev@iacp.dvo.ru

shalf@iacp.dvo.ru

This article is devoted to features of execution, contents and results of system analysis as a part of complex technology of automation of intelligent activity at the branch level. Performance of all presented stages and receiving the corresponding results allow to implement knowledge-based decision-making support and systematic control of the knowledge quality. Implementation of such system analysis guarantees semantic compatibility of created intelligent systems.

Introduction

The analysis of daily intellectual activity's organization and management of its quality revealed the need of transition to a new paradigm of automation – «branch automation», i.e. development of uniform "cloudy" intelligent system for whole branch, its decision-making and knowledge quality control support.

The purpose of this work – to discuss features of execution, contents and results of system analysis, allowing to carry out automation of daily intelligent activity and control of knowledge quality at the branch level.

Main Part

In the main part of the article a system analysis stages' features are presented: identification of tasks in domain, development of ontologies, tasks setting, development of decision methods and algorithms. Also conceptual design of subsystems for knowledge engineering stages are described.

Conclusion

The presented stages of the system analysis are typical for process of automation of intellectual activity. But possibility of such automation at the branch level depends on existence of methods and instruments of quality management of knowledge. The crucial role belongs to results of ontologic engineering on the basis of which all semantic compatible subsystems and instruments are being developed and integrated.