

С.В. Войцеховский, Е.Г. Мысливец, И.А. Пучкова

К ВОПРОСУ О ВОЗМОЖНОСТИ ИНТЕЛЛЕКТУАЛИЗАЦИИ ПРОЦЕССОВ В СИСТЕМАХ УПРАВЛЕНИЯ БАЗАМИ ДАННЫХ

Исследуются методологические основы синтеза модели процессов в серверных системах управления базами данных (СУБД) на основе закона сохранения целостности объекта с использованием алгебраической операции, связанной со свойствами несущих множеств. Среда радикалов рассматривается как инструмент моделирования процессов управления и обеспечивает технологичность подхода, поскольку учитывает элемент логики в пространственно-временных состояниях, что позволяет отобразить условия существования вычислительного процесса.

Ключевые слова: имитационная модель, серверная СУБД, интеллектуальное управление, среда радикалов, индуктивно-дедуктивные выводы.

S.V. Voichevskij, E.G. Mislivec, I.A. Puchova

TO THE QUESTION ABOUT THE POSSIBILITY OF INTELLECTUALIZATION OF THE PROCESSES IN DATABASE MANAGEMENT SYSTEMS

The methodological basis of the synthesis of the process model in the server database on the basis of the law of preservation of the integrity of the object, using an algebraic operation associated with the properties of bearing sets. The radical environment is considered as a tool for modeling control processes and provides technological approach, because it takes into account the element of logic in space-time States, which allows to display the conditions of the existence of the computational process.

Keywords: simulation model, server DBMS, intelligent control, radical environment, inductive-deductive conclusions.

1. Современные решения в области интеллектуализации СУБД

Использование баз данных (БД) и информационных систем (ИС) становится неотъемлемой составляющей управленческой деятельности и функционирования крупных организаций. От правильного выбора инструментальных средств создания ИС, определения адекватной модели данных, обоснования рациональной схемы построения БД, организации запросов к хранимым данным и ряда других моментов во многом зависит эффективность функционирования разрабатываемых систем. Все это требует осознанного применения теоретических положений и инструментальных средств разработки БД и ИС [1].

В области интеллектуализации вычислительных процессов можно отметить несколько новейших тенденций, возникающих при разработке крупными корпорациями программного обеспечения для ИС с применением серверных СУБД. Общую тенденцию интеллектуализации и расширения функций СУБД обеспечивают две важные возможности, которыми обладают практически все современные реляционные корпоративные СУБД, – поддержка хранимых процедур и применение триггеров [2]. Известны СУБД, которые управляют низкоуровневыми процессами в ИС, в том числе ресурсами вычислительной системы [1].

© Войцеховский С.В., Мысливец Е.Г., Пучкова И.А., 2018.

Сегодня общемировой тенденцией становится разработка специализированного программного обеспечения, ориентированного на платформу LINUX. Informix стал одной из основных СУБД, работающих в среде UNIX-подобных операционных систем (ОС). Работа сервера Informix подразумевает запуск специальной программы, которая обеспечивает функционирование всех SQL-операторов. Для каждого клиента запускается процесс ОС, использующей эту программу. Однако в случае если клиент прервал работу, но не вышел из своей задачи, его процесс занимает ресурсы системы, снижая ее производительность [1]. Реляционный подход, изначально выбранный при создании программных продуктов СУБД Informix, был определен тем обстоятельством, что базы данных в этом случае имеют надежную структуру и все действия, предпринимаемые над данными, дают предсказуемые результаты. При этом модель данных понимается как алгебраическая система – множество всевозможных типов данных, а также определенных на них отношений и операций [2].

Достижением Informix стал выпуск нового сервера БД OnLine Dynamic Server, который входит в состав системы, начиная с версии 6.0 (2017 г.). Этот продукт основан на так называемой динамической масштабируемой архитектуре (Dynamically Scalable Architecture – DSA), которая специально ориентирована на работу с многопроцессорными системами. OnLine Dynamic Server обеспечивает повышение производительности за счет гибкости использования ресурсов СУБД и многопоточной архитектуры. Фактически OnLine Dynamic Server берет на себя многие связанные с распределением ресурсов функции ОС. В результате уменьшается нагрузка на ОС, что в конечном счете приводит к росту производительности. Для обслуживания клиентов запускаются «виртуальные процессоры» – процессы ОС, которые устанавливают связь между клиентом и ядром Informix.

Связь устанавливается с помощью специальных программных модулей под названием «нить» (thread), которые активизируются только если клиент обращается к серверу БД. В случае, когда клиент неактивен, «нить» может обслуживать других клиентов. Число виртуальных процессоров определяет администратор БД, исходя из реальных ресурсов вычислительной системы и сети клиентов. Если вычислительная система является многопроцессорной, то разные виртуальные процессоры могут обслуживаться разными реальными процессорами [3].

2. Отечественные программные решения в области интеллектуального управления вычислительными процессами

За последние годы в области интеллектуального управления были представлены программные работы отечественных ученых. Статья А.В. Чечкина, М.В. Пирогова и В.А. Васенина посвящена принципиально новой парадигме нейрокомпьютерных систем, не имеющей зарубежных аналогов, в виде активаторов и регуляторов среды радикалов исходя из применения теории квантовой интерференции к построению нейронных сетей. Также предложена модель проблемной области сложной системы в форме среды радикалов и сформулирована ключевая проблема критических систем – проблема информационно-системной безопасности. Таким образом, был предложен совершенно новый тип моделирования и показана возможность алгебраизации теории нейронных сетей. Это позволит поднять ее на качественно более высокий уровень, а также улучшит согласование идеи нейрокомпьютера с современными тенденциями суперкомпьютеров, GRID вычислений, агентных систем, компьютерных кластеров и т.д. [4].

На основе отечественных достижений предлагается рассмотреть возможность создания серверной СУБД, которая обеспечит условия существования вычислительных процессов с требуемой производительностью. Именно в серверных СУБД стоят вопросы экономного обращения с ресурсами и ограничения по времени обслуживания клиентов [2].

Благодаря новейшим исследованиям А.В. Чечкина [4; 5], В.Г. Бурлова [6; 7], О.М. Лепешкина [7] представляется возможным создание имитационной модели (ИМ) СУБД, основанной на новом подходе в области бесконфликтного использования ресурсов с применением новой технологии проектирования. Работы В.Г. Бурлова продолжают классические труды П.К. Анохина в области синтеза функциональных систем [8].

Согласно исследованиям П.К. Анохина, об интеллектуализации управления можно говорить исключительно при наличии комплекса обратных связей, корректирующих поведение функциональной системы на каждом этапе достижения цели. При использовании разработанного А.В. Чечкиным нового типа моделирования среди прочих функций возможно рассчитать «интеллектуальное» поведение СУБД в отношении распределения вычислительных ресурсов. В работах О.М. Лепешкина решены технологические вопросы на основе фазирования процессов управления, кроме того, в его работах к новейшим достижениям относится разработка координатной сетки для проектирования сложных систем в среде радикалов. За счет процессного подхода и на основе закона сохранения целостности, разработанного В.Г. Бурловым, могут быть решены задачи бесконфликтного управления ресурсами.

Рассмотрим, каким образом проектируемая на отечественной базе ИМ СУБД может создать альтернативу достижениям создателей сервера БД OnLine Dynamic Server. Последние сообщают, что количество «нитей» и время их жизненного цикла определяются человеком (администратором), исходя из наблюдения за состоянием вычислительных ресурсов, но ничего не сообщается о возможности автоматизировать указанные действия в отношении ОС и о принципах организации обратной связи для управления запрошенными вычислительными ресурсами.

В работе [9] доказано, что в ОС должно существовать понятие приоритета потока. Речь идет об ОС реального времени (ОСРВ). Проблема в том, чтобы определить, какой задаче ресурс требуется более всего. В идеальной ситуации ОСРВ отдает ресурс потоку или драйверу с ближайшим крайним сроком (deadline driven OS). Чтобы реализовать это временное ограничение, ОС должна «знать», сколько времени требуется каждому из выполняющихся потоков для завершения. ОС, построенных по этому принципу, практически нет, так как он слишком сложен для реализации.

Поэтому разработчики ОС принимают иную точку зрения: вводится понятие уровня приоритета для задачи, и временные ограничения сводят к приоритетам. Так как умозрительные решения чреваты ошибками, показатели системы реального времени снижаются. Чтобы эффективно осуществить указанное преобразование ограничений, проектировщик может воспользоваться теорией расписаний или имитационным моделированием.

Так как на сегодняшний день не имеется иного решения, понятие приоритета потока неизбежно. В ОС должна существовать система наследования приоритетов. Именно этот механизм синхронизации и тот факт, что различные потоки используют одно и то же пространство памяти, отличают их от процессов. Процессы почти не разделяют одно и то же пространство памяти, а в основном работают в своих локальных адресных пространствах. Комбинация приоритетов потоков и разделения ресурсов между ними приводит к другому явлению – классической проблеме инверсии приоритетов.

Например, есть три потока. Когда поток низшего приоритета захватил ресурс, разделяемый с потоком высшего приоритета, и при этом начал выполняться поток среднего приоритета, выполнение потока высшего приоритета будет приостановлено, пока не освободится ресурс и не отработает поток среднего приоритета. В этой ситуации время, необходимое для завершения потока высшего приоритета, зависит от низших приоритетных уровней, это и есть инверсия приоритетов. Ясно, что в такой ситуации трудно выдержать ограничение на время исполнения.

Чтобы устранить такие инверсии, ОСРВ должна допускать наследование приоритета как повышение уровня приоритета до уровня потока, который его вызывает. Наследование означает, что блокирующий ресурс поток наследует приоритет потока, который он блокирует (разумеется, это справедливо лишь в том случае, если блокируемый поток имеет более высокий приоритет) [9].

Если вернуться к рассмотрению OnLine Dynamic Server, то можно заметить некоторые аналогии между поведением «потока» и «нити». Его создатели не раскрывают в открытых публикациях принципов построения своего программного продукта. Можно предположить, что оптимальное управление «нитеями» способно обеспечить экономию вычислительных ресурсов. Поскольку «нити» также конкурируют за вычислительные ресурсы, имеет значение время срабатывания «нити». Допустим, некая активная, но задержанная во времени «нить» сработает немного позже, и это обстоятельство позволит не конкурировать за ресурсы, что в итоге (для клиентов БД) обеспечит выигрыш и по времени, и по вычислительным ресурсам.

Допустим, что пространство «нитей» будет представлять собой множество элементов с тремя возможными состояниями: 1 – «активен», 2 – «пассивен», 3 – «активен, но задержан». Предположим, что существуют возможные состояния «поля нитей», которые способствуют выполнению целевых задач вычислительного процесса с требуемой производительностью. Но также существуют состояния, препятствующие вычислительному процессу.

Управление «нитеями» отличается от управления потоками в ОС, поскольку «пространство нитей» представляет собой условно статическую систему, в отличие от управляемых ОС динамических процессов и потоков. Пространство «нитей» вместе с разработанными ограничениями и поставленными для этого пространства целями напоминает модель возможных состояний аксонов мозга живого существа и соответствует модели «искусственного интеллекта», описанной в трудах Анохина П.К. Это обстоятельство в свою очередь позволяет говорить об «интеллектуальном» поведении проектируемой ИМ СУБД.

Исследования Бурлова В.Г. доказывают возможность синтезировать систему с условно бесконфликтным доступом к ресурсам при заранее заданной производительности системы (закон сохранения целостности). Причем классические труды Анохина П.К. говорят о комплексе обратных связей, необходимых для реализации интеллектуального управления (так называемые афферентации). Кроме того, новейшая отечественная технология проектирования, не имеющая аналогов в зарубежных странах, разработанная Лепешкиным О.М. (фазирование процессов, координатная сетка) и Чечкиным А.В. (теория радикалов), дает возможность учитывать одновременно задачи автоматизации процесса управления, выявления конфликтных ситуаций, реагирования на деструктивные факторы. Также среда радикалов при проектировании сложной системы позволяет одновременно учитывать аппаратную и программную составляющую в отличие от ранее известных зарубежных инструментальных средств разработки [1].

3. Формирование рационального облика ИМ СУБД и ее функционирование

Существующая концепция управления вычислительными ресурсами приводит в силу неполноты учета динамики состояния ИС к возникновению нештатных состояний по управлению ресурсами (например, замедление работы СУБД при большом количестве записей в таблицах).

Пусть задачей ИМ СУБД будет обеспечение заданной производительности, которая является опосредованным показателем производительности вычислительной системы в целом и характеризуется количеством задач, выполняемых вычислительным средством в единицу времени. Условием существования вычислительного процесса положим бесконфликтное использование вычислительных ресурсов.

Объектом исследования при решении поставленной задачи является проектируемая ИМ СУБД с элементами различного уровня иерархии.

Предметом исследования при решении задачи являются закономерности построения рационального облика ИМ СУБД и ее функционирование.

Известно, что СУБД может быть функциональным элементом ИС [2]. На рис. 1 отображена соответствующая схема.

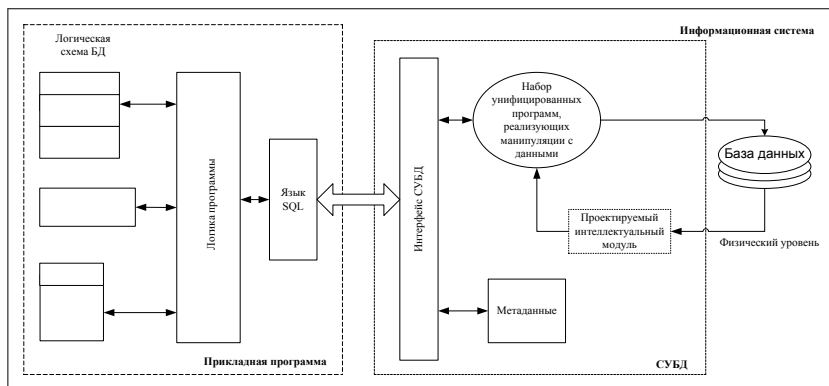


Рис. 1. СУБД как функциональный элемент ИС

Как правило, под интеллектуализацией подразумевается автоматизация поддержки принятия решений [2]. Для реализации автоматического управления в ИМ СУБД необходимо получить условия существования вычислительного процесса и реализовать обратную связь [7].

Введение учета обстановки, мониторинга и нейтрализации проблем в ИС создает необходимые условия автоматического управления процессами ИС и ее отдельных элементов в соответствии с изменившимися требованиями внутренней и внешней среды за счет реализации обратной связи. В результате таких постоянных преобразований и перехода к новому качественному состоянию ИМ восстанавливает нарушенные в результате накопившихся изменений внутреннее и внешнее равновесия, которые определяются как требуемая производительность.

Учитывая реализацию новой концепции поддержания функционирования ИС, ИМ СУБД «формирует цель» и «разрабатывает план действий». Далее устанавливается последовательность регламентированных операций (регламент) и осуществляется запрос ресурсов через обращение к ОС. При условии обеспечения ресурсами ИМ СУБД сразу инициирует выполнение регламента для достижения поставленной цели, не затрачивая времени на согласования с другими потребителями ресурсов.

Существующее положение в области имитационного моделирования ИС [1] показывает отсутствие единого формализма и метода моделирования процессов. Есть возможность представить, что сложная система и ее составляющие, а также все составляющие проблемной области рассматриваются как математические объекты единой системы. Тогда частные математические модели управления, основанные на единой базовой математической модели, могут дать специалистам полное представление о ситуации, складывающейся в области вычислительных процессов, и обеспечить решение всех целевых задач в заданные сроки с требуемой эффективностью.

Для формирования ИМ СУБД авторами предлагается новый подход, суть которого заключается в следующем. Модель внутренней структуры ИМ СУБД может быть отображена логико-алгебраической концепцией математики, а вычислительные

процессы – в структурно-математической концепции. Как основа моделирования создан системно-образующий агрегат, состоящий из алгебраической операции, связанной со свойствами несущих множеств, и операциями интегрирования. Использование данного агрегата позволяет применить формально-аксиоматический метод для построения дедуктивно-индуктивных выводов и реализовать обратную связь в управлении вычислительными процессами [7].

Задача синтеза модели интеллектуализации процессов в серверных СУБД рассмотрена:

– на методологическом уровне – получение условий существования процесса управления вычислительным процессом в виде устойчивой повторяющейся связи свойства объекта и свойства действия при фиксированном предназначении;

– на методическом уровне – решение за счет среды радикалов, которая позволяет описывать активацию управления вычислительным процессом в пространстве и времени, исходя из сложившейся обстановки;

– на технологическом уровне – решение за счет реализации концепции фазирования управления вычислительным процессом.

Далее рассмотрим три уровня решения задачи.

Методологический уровень

В основе теории синтеза облика (модели) и способов применения системы находится закон сохранения целостности объекта: устойчивая, объективная, повторяющаяся связь свойств объекта и свойств его действия при фиксированном предназначении. В работах Бурлова В.Г. разработана математическая структура, которая на основе новой аксиоматики, включающей язык (базовые понятия, ключевые слова и отношения между ними), аксиому (уравнение синтеза) и теоремы (базовые зависимости достижения результата), позволила строго математически определить систему и тем самым учитывать конструкцию, применение и целевое предназначение системы [6].

Проектируемая ИМ СУБД обладает свойствами функциональной системы и создается для поддержания вычислительных процессов с использованием серверной СУБД на требуемом уровне.

Разработанная алгебраическая операция позволяет рассмотреть применение формально-аксиоматического метода к синтезу ИМ СУБД [6].

Обозначим через R множество всех возможных пространственно-временных состояний (ПВС) проектируемой системы.

За основу формирования условия существования процесса положен закон сохранения целостности объекта в интегральной форме:

$$\int_Q \varphi(r) dr = \int_Q \Phi(u(r), v(r), r) dr = I(r), \quad (1)$$

где Q – множество требуемых (в смысле достижения цели) ПВС системы;

r – пространственно-временное состояние системы, здесь $r \in R$, $R \supset Q$;

$u(r)$ – вектор управления системой;

$v(r)$ – вектор возможностей системы (ресурсы, силы и средства);

$\varphi(r) = \Phi(u(r), v(r), r)$ – функция, потенциал поля эффективности разрабатываемой системы;

$I(r)$ – показатель заданной эффективности функционирования системы при устранении дестабилизирующих воздействий на вычислительные процессы и выполнение задач с учетом понесенного ущерба от этих воздействий, а также затрат материальных средств, ресурсов и времени.

В качестве меры для оценивания адекватности проектируемой системы по критерию полноты учета закономерностей предметной области предлагается вероятностная мера.

В работах Бурлова В.Г. [6] отражены соотношения финальных вероятностей состояний проектируемой системы, полученных путем решения системы линейных алгебраических уравнений, которые в свою очередь получаются из дифференциальных уравнений Колмогорова. Эти соотношения определяют вероятность нахождения системы в заранее обусловленных состояниях; как следствие можно выработать требования к свойствам процесса распознавания проблемы, возникшей в ИС при обращении к вычислительным ресурсам и к свойствам процесса нейтрализации этой проблемы.

Вероятность нахождения проектируемой системы в заранее обусловленном состоянии в зависимости от времени проявления деструктивного воздействия и времени его преодоления, рассчитывается так:

$$P_{\text{обусл.}} = \frac{v_1 v_2}{\lambda(\lambda + v_1 + v_2) + v_1 v_2}, \quad (2)$$

где λ – величина, обратная среднему времени проявления проблемы;

v_1 – величина, обратная среднему времени идентификации проблемы;

v_2 – величина, обратная среднему времени нейтрализации проблемы.

На рис. 2 показано графическое представление зависимости вероятности нахождения системы в заданном состоянии от характеристики аналитической деятельности и характеристики процесса нейтрализации проблемы при выбранной $\lambda = 170$.

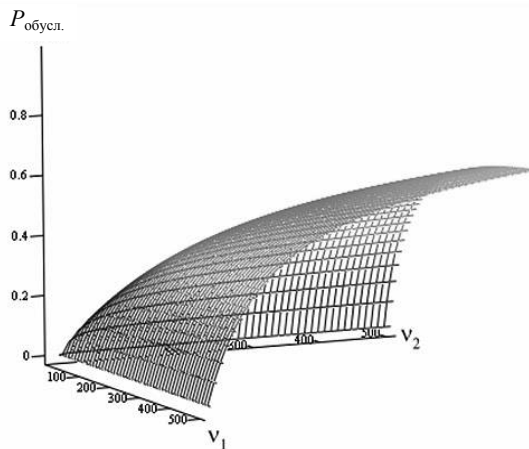


Рис. 2. Вероятность нахождения проектируемой системы в заранее обусловленном состоянии

Таким образом, установлена аналитическая зависимость обобщенных характеристик обстановки ($\Delta t_{\text{ин}}$), аналитической деятельности ($\Delta t_{\text{ин}}$) и нейтрализации проблемы ($\Delta t_{\text{ин}}$), возникшей в ИС.

$$\text{Соответственно, } \Delta t_{\text{ин}} = \frac{1}{v_1}, \Delta t_{\text{ин}} = \frac{1}{v_2}, \Delta t_{\text{ин}} = \frac{1}{\lambda}.$$

Методический уровень

Для реализации процессного подхода предлагается моделирование на основе формализма среды радикалов из области математической информатики [4; 5; 7]. Среда радикалов обеспечивает технологичность подхода, поскольку учитывает элемент логики в пространственно-временных состояниях, что позволяет отобразить условия

существования вычислительного процесса. Среда радикалов рассматривается как инструмент моделирования процессов управления.

Подход к проектированию сложных систем, разработанный профессором А.В. Чечкиным [5], позволяет реализовать «интеллектуальную» надстройку СУБД. Среда радикалов как инструмент моделирования процессов управления базируется на понятиях: уникум, контейнер, ультраконтейнер и активатор, которые формируют описание поля реализации двух состояний элементов системы («активное» и «пассивное»), как во времени, так и в пространстве.

Под радикалом понимается любая функциональная система, имеющая два доступных извне состояния: «активное» и «пассивное». Активный радикал функционирует согласно своему предназначению, а пассивный не проявляет себя. Множество радикалов со связями между собой являются средой радикалов. Вопросами активации среды радикалов занимаются системы, называемые активаторами. Система всех активных радикалов среды радикалов образует так называемый системоквант, который определяет квант поведения среды на данный момент.

Любой объект управления имеет различные состояния в зависимости от параметров управления. Наборы возможных управляющих параметров образуют среду радикалов, система управления объектом является активатором, а реализованные в данный момент значения управляющих параметров образуют системоквант в этой среде радикалов.

Теория радикалов включает в себя три взаимодействующих подсреды: опорная среда, ультрасреда, активирующая подсистема [4; 5].

Опорная среда представляет собой описание элементов сложной системы при помощи опорных радикалов – уникумов и контейнеров. Уникумы обозначают составляющие (компоненты) сложной системы, контейнеры обозначают свойства уникумов.

Ультрасреда содержит знания и сведения об элементах опорной среды, которые описаны в виде логики работы ультраконтейнеров. Ультраконтейнер является компонентом ультрасреды, содержит последовательность задач, связанных как с внесением изменения в опорную среду, так и с получением информации об опорной среде.

Активирующая подсистема предназначена для осуществления взаимодействия «пользователя» со средой радикалов. «Пользователю» доступны операции создания радикала, цепи, запроса, ультра-контейнера, задач ультраконтейнера и удаления радикала.

Понятие радикала позволяет описать всю проблемную область сложной системы и ее окружение с учетом жизненного цикла на среде радикалов. Радикалами такой среды являются составляющие сложной системы и ее окружения, их связи, функции, задачи жизненного цикла, средства и методы решения таких задач. При таком подходе создаваемая интеллектуальная надстройка сложной системы должна принимать такие решения, которые обеспечивают управление системой на протяжении всего ее жизненного цикла.

Интеллектуальная надстройка сложной системы строится из двух частей: рабочей и активирующей. Рабочая подсистема будет символьным представлением в интеллектуальной системе радикалов проблемной области сложной системы, т.е. будет моделью проблемной области сложной системы. Активирующая подсистема содержит средства анализа, синтеза и активации рабочей области. Она отвечает за решение задач жизненного цикла системы и постоянно разрешает конфликты в среде радикалов.

Используя модель проблемной области, активирующая подсистема должна создавать в рабочей подсистеме системоквант, который определяет поведение сложной системы. Пример геометрической схемы радикалов управления ресурсами приведен на рис. 3.

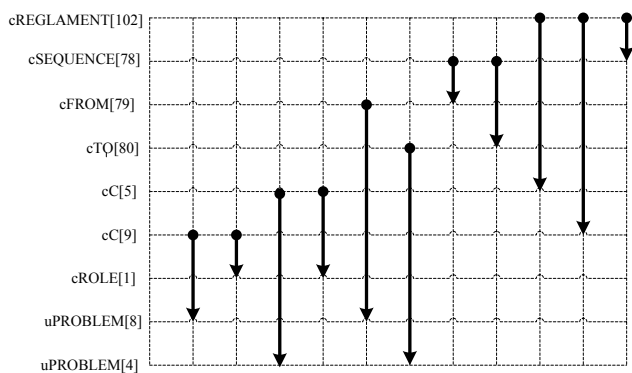


Рис. 3. Пример геометрической схемы радикалов управления ресурсами

Технологический уровень

Для разработки ИМ СУБД следует соотнести этапы проектирования и основные положения модели функционально-ролевого управления, провести однозначное отображение существующих методов моделирования информационных систем на среду радикалов, предложить рекомендации по формированию коллективных информационных задач и использованию результатов активации схем управления в динамике функционирования ИМ СУБД при одновременной разработке политики безопасности СУБД.

Указанными методами успешно реализована интегрированная платформа управления социальными и экономическими системами жизнедеятельности региона «Ароганит» [10], где на основе формально аксиоматического метода разработаны языковые средства, которые представляют среду радикалов и позволяют строить из них необходимые структуры. Взаимодействие уникамов описывает поведение системы и их состояние в контейнере («штатное», «нештатное» и «аварийное»). Отношения контейнеров прописаны в ультраконтейнерах.

Сообщается, что главным достоинством нового метода является механизм взаимодействия ультраконтейнеров, контейнеров и уникамов на координатной сетке. В данной координатизации не только прописано взаимодействие уникамов, но и учитывается взаимодействие контейнеров и ультраконтейнеров на основе параметров уникамов как между собой, так и при вертикальной композиции-декомпозиции отношений уникам-контейнер-ультраконтейнер.

При выполнении требований процессного подхода реализовано управление доступом к ресурсам на основе возможности трансформации субъектно-объектного подхода в среду радикалов для построения системы управления реального времени. Для этого проведена формализация функции в субъектном подходе на основе среды радикалов, что позволило развить новое направление, учитывающее сочетание функциональности и безопасности, которое рассматривает режимы работы протекающих процессов на основе прописанных регламентов [7]. Следует отметить, что благодаря опыту практической разработки узкоспециального программного обеспечения (платформа «Ароганит») в рамках разработанной методики удалось создать удобное и простое в использовании ПО [10].

Заключение

На основе структурно-математического и формально-аксиоматического подходов разработана методика синтеза ИМ СУБД для LINUX-подобных систем (с откры-

тым кодом), что соответствует решению задач, поставленных перед МО РФ руководством государства, по переводу информационных и управляющих систем на отечественное ПО.

Предлагаемая методика позволяет обеспечить единый и непротиворечивый подход к разработке моделей как объектов управления, так и систем управления в сложных и больших организационно-технических структурах. Показана возможность рассмотреть вопросы информационной безопасности непосредственно на этапах проектирования.

Признаком интеллектуализации процессов в СУБД является построение на множестве требуемых состояний ИМ непротиворечивых индуктивно-дедуктивных выводов [7].

Учет в ИМ СУБД комплекса обратных связей с вероятностным оцениванием показателей эффективности функционирования целевых систем трансформирует традиционные гомеостатические системы управления (на основе обратных связей) в перспективные аллостатические (на основе обратных-прямых связей и упреждающего прогноза). Именно аллостатические ИМ позволят в ближайшем будущем синтезировать СУБД, а также перспективные системы безопасности и управления с малым потреблением ресурсов в робототехнике.

Литература

1. *Хомоненко А.Д., Цыганков В.М., Мальцев М.Г.* Базы данных: учебник для высш. учеб. завед. / под ред. проф. А.Д. Хомоненко. 6-е изд., доп. СПб.: КОРОНА-Век, 2009. 736 с.

2. *Бурлов В.Г.* Синтез модели управления информационной безопасностью // Информационные управляющие системы и технологии: материалы IV Международной научно-практической конференции (ИУСТ-ОДЕССА-2015). 2015. С. 147–150.

3. *Бурлов В.Г., Лепешкин О.М.* Моделирование процесса управления на основе теории радикалов // Нейрокомпьютеры: разработка, применение. 2016. № 3. С. 54–61.

4. Архитектура и программное обеспечение современных компьютерных систем и сетей войск ВКО: учеб.-метод. пособие / С.В. Войцеховский, С.В. Калиниченко. СПб.: ВКА им. А.Ф. Можайского, 2013. 352 с.

5. *Лепешкин О.М., Бурлов В.Г., Кирилова Т.В.* Моделирование процесса управления социальными и экономическими системами региона на основе потенциально активированных элементов пространства и времени // Проблемы экономики и управления в торговле и промышленности. СПб., 2013. № 3. С. 82–85.

6. *Новиков А.Н., Нечай А.А., Малахов А.В.* О подходе к обоснованию рациональной номенклатуры эталонной базы измерительных комплексов на основе нечетких моделей // Вестник Российского нового университета. Сер. «Сложные системы: модели, анализ и управление». 2017. № 1. С. 72–79.

7. *Калиниченко С.В., Котиков П.Е., Нечай А.А.* Решение репликационных проблем в базах данных для повышения устойчивости программного обеспечения автоматизированных систем // Вестник Российского нового университета. Сер. «Сложные системы: модели, анализ и управление». 2017. № 4. С. 18–21.

8. *Котиков П.Е., Нечай А.А.* Репликация данных между серверами баз данных в среде геоинформационных систем // Вестник Российского нового университета. Сер. «Сложные системы: модели, анализ и управление». 2015. № 1. С. 88–91.

9. *Котиков П.Е., Нечай А.А.* Решение проблемы управления параллельным выполнением транзакций в распределенных базах данных для устранения опасной противоречивости // Вестник Российского нового университета. Сер. «Сложные системы: модели, анализ и управление». 2015. № 2. С. 62–64.

10. Лохвицкий В.А., Калининченко С.В., Нечай А.А. Подход к построению системы автоматизированной интеграции информации в базу данных для ее своевременной актуализации // Мир современной науки. 2014. № 2 (24). С. 8–12.

References

1. Khomonenko A.D., Tsygankov V.M., Mal'tsev M.G. Bazy dannykh: uchebnyk dlya vyssh. ucheb. zaved. / pod red. prof. A.D. Khomonenko. 6-e izd., dop. SPb.: KORONA-Vek, 2009. 736 s.
2. Burlov V.G. Sintez modeli upravleniya informatsionnoy bezopasnost'yu // Informatsionnye upravlyayushchie sistemy I tekhnologii: materialy IV Mezhdunarodnoy nauchno-prakticheskoy konferentsii (IUST-ODESSA-2015). 2015. S. 147–150.
3. Burlov V.G., Lepeshkin O.M. Modelirovanie protsessa upravleniya na osnove teorii radikalov // Neyrokomp'yutery: razrabotka, primeneniye. 2016. № 3. S. 54–61.
4. Arkhitektura i programnoye obespecheniye sovremennykh komp'yuternykh sistem I setey voysk VKO: ucheb.-metod. posobie / S.V. Voytsekhovskiy, S.V. Kalinichenko. SPb.: VKA im. A.F. Mozhayskogo, 2013. 352 s.
5. Lepeshkin O.M., Burlov V.G., Kirilova T.V. Modelirovanie protsessa upravleniya sotsial'nymi i ekonomicheskimi sistemami regiona na osnove potentsial'no aktivirovannykh elementov prostranstva i vremeni // Problemy ekonomiki i upravleniya v torgovle i promyshlennosti. SPb., 2013. № 3. S. 82–85.
6. Novikov A.N., Nechay A.A., Malakhov A.V. O podkhode k obosnovaniyu ratsional'noy nomenklatury etalonnoy bazy izmeritel'nykh kompleksov na osnove nechetkikh modeley // Vestnik Rossiyskogo novogo universiteta. Ser. "Slozhnyye sistemy: modeli, analiz i upravleniye". 2017. № 1. S. 72–79.
7. Kalinichenko S.V., Kotikov P.E., Nechay A.A. Resheniye replikatsionnykh problem v bazakh dannykh dlya povysheniya ustoychivosti programmnoy obespecheniya avtomatizirovannykh sistem // Vestnik Rossiyskogo novogo universiteta. Ser. "Slozhnyye sistemy: modeli, analiz i upravleniye". 2017. № 4. S. 18–21.
8. Kotikov P.E., Nechay A.A. Replikatsiya dannykh mezhdru serverami baz dannykh v srede geoinformatsionnykh sistem // Vestnik Rossiyskogo novogo universiteta. Ser. "Slozhnyye sistemy: modeli, analiz i upravleniye". 2015. № 1. S. 88–91.
9. Kotikov P.E., Nechay A.A. Resheniye problemy upravleniya parallel'nym vypolnениem tranzaktsiy v bazakh dannykh dlya ustraneniya opasnoy protivorechivosti // Vestnik Rossiyskogo novogo universiteta. Ser. "Slozhnyye sistemy: modeli, analiz i upravleniye". 2015. № 2. S. 62–64.
10. Lokhvitskiy V.A., Kalinichenko S.V., Nechay A.A. Podkhod k postroeniyu sistemy avtomatizirovannoy integratsii informatsii v bazu dannykh dlya ee svoevremennoy aktualizatsii // Mir sovremennoy nauki. 2014. № 2 (24). S. 8–12.