

**БОГАТИКОВ Валерий Николаевич** – доктор технических наук, ведущий научный сотрудник Института информатики и математического моделирования технологических

**КРЕМЕНЕЦКИЙ Вячеслав Георгиевич** – кандидат химических наук, старший научный сотрудник, Института химии и технологии редких элементов и минерального сырья им. И. В. Тананаева КНЦ РАН

**СИДЕЛЬНИКОВ Дмитрий Владимирович** – аспирант Института информатики и математического моделирования технологических процессов КНЦ РАН процессов КНЦ РАН

## **ПРОЕКТИРОВАНИЕ КОМПЛЕКСА СЧЕМОБЪЕКТОВ. ПРОЕКТИРОВАНИЕ ПОДСИСТЕМЫ ДОСТУПА К ИНФОРМАЦИОННЫМ РЕСУРСАМ.**

Усиление процессов интеграции в науке и технике, необходимость решения сложных комплексных проблем, стимулирующих научно-технический прогресс, обуславливают актуальность организации комплексных научных исследований [1]. Это неизбежно связано с одновременным или последовательным применением различных методов исследований и привлечения математических моделей явлений и вычислительных экспериментов (ВЭ).

Реализация комплексных исследований, часто, приводит к созданию многоуровневых распределенных информационных систем научных исследований, в которых средства хранения, обработки и анализа информации интегрируются со средствами подготовки и реализации ВЭ. Часто программные комплексы поддержки и проведения научных исследований реализуются в виде, так называемых, пакетов прикладных программ состоящих программных модулей и средств управления их работой, совместимых по структуре данных и способам управления и объединяемых функционально-целевым назначением [1].

### ***Анализ предметной области***

В ходе проведенного обзора химической литературы по кластерным моделям расплавов и твердых тел было установлено, что даже для такого классического объекта исследований, каким являются расплавы галогенидов щелочных металлов, до сих пор – несмотря на сорокалетнее интенсивное изучение – наблюдается регулярное появление все новых структурных моделей. Подобная ситуация имеет место и для расплавов других веществ, не говоря уж о многокомпонентных системах. Однако в большинстве случаев существующие модели неполны или недостаточно конкретны для того, чтобы их можно было подвергнуть проверке на физическую обоснованность методами молекулярной статистики или динамики, и нуждаются в существенной доработке. Эта доработка опирается на применение различного рода программных комплексов.

В связи с этим, был проведен обзор программных средств для физико-химического моделирования и визуализации, в том числе была рассмотрена продукция таких известных корпораций и фирм как Accelrys, Tripos, HyperCube Inc., ChemAxon Ltd., Cambridge Soft, ACD Labs, Scienomics Inc., CompuChem, Crystal Impact. Были рассмотрены также такие программные комплексы, как GULP, ChemCraft, ChemViz, Jmol, Viewmol, Crystal Studio, ARITVE, MOLDRAW, OSCAIL X, Q-mol, Crystal Maker, gOpenMol, Vega ZZ, YASARA, LigBuilder, CaRIne Crystallography, CRYSTAL-03, MOLDA, HIMERA, Convert, Babel File Conversion. По результатам обзора было найдено программное обеспечение, позволяющее в целом, с различной степенью автоматизации и предоставляемого сервиса, проектировать, генерировать, визуализировать и анализировать различные химические системы. Тем не менее, рассмотренные программные продукты не предоставляют всего необходимого инструментария для реализации молекулярно-статического подхода, включающего в себя не только анализ готовых моделей, но и их достройку или корректировку. Недостаточно развит оказался также инструментарий, необходимый для конструирования структурных моделей, и особенно – для анализа многомерных поверхностей.

Эти факты и послужили отправной точкой для формулирования требований и последующей разработки программного комплекса ChemObjects [2, 3].

### ***Подходы и технологии***

Существует достаточно много подходов и информационных технологий, нацеленных на поддержку и реализацию этапа проектирование жизненного цикла информационных систем. Одной из перспективных и общих методологий построения систем является концептуальное проектирование [4]. В его рамках концептуальные модели (КМ) предметной области (ПО) используются для формализации и представления экспертных знаний об информационных объектах и процессах их обработки [5]. Одним из подходов к созданию концептуальных моделей предметной области является функционально-целевой подход, применяемый для класса задач с древовидными моделями предметной области [6].

Первая версия комплекса ChemObjects, была разработана с использованием системного подхода, объектно-ориентированного анализа, структурного, модульного, компонентного и плагиносного подходов. Эксплуатация показала возможности для развития комплекса как сточки зрения недостаточность функционального наполнения, связанного с предметной областью, так и по алгоритмическому и информационному обеспечению.

Ввиду достаточно весомых требований, предъявляемых к новой версии комплекса, повышается актуальность привлечения новых, эффективных средств проектирования.

В настоящее время широко применяется шаблонное (паттерное) проектирование и программирование [7-9], которое называют еще метапрограммированием. Данный подход является воплощением уровня абстрактности, при котором большинство частей системы строятся из библиотеки заранее отработанных шаблонов (часто независимых от типов данных тех элементов, которые входят в состав шаблона), и при этом может достигаться относительно высокая и управляемая гибкость системы, как на уровне исходного кода, так и возможно на уровне реализации. Эффективность разработки во многом зависит от того, насколько хорошо освоена библиотека шаблонов проектирования.

Сейчас активно развивается в различных прикладных областях технология построения информационных систем на базе интеллектуальных агентов и мультиагентов [10-11]. Технология агентных систем является новой парадигмой информационной технологии, ориентированной на совместное использование научных и технических достижений и преимуществ, которые дают идеи и методы ИИ, современные локальные и глобальные компьютерные сети, распределенные БД и распределенные вычисления, аппаратные и программные средства поддержки построения распределенных и открытых ИС [12].

Сложно представить сколько-нибудь развитую ИС по управлению, накоплению или обработке структурированных данных, совсем не использующую технологии БД. Несмотря на активное развитие новой объектно-ориентированной парадигмы организации БД [13] все еще широко распространенной (и хорошо проверенной временем) является технология реляционных БД [14-16].

Существует достаточно много наработок в области построения КМ с использованием функционально-целевого подхода и их применения в разных прикладных областях [4-6, 17, 18]. Однако законченных и доступных программных продуктов, подходящих для автоматизации построения ИС с использованием концептуального проектирования на базе функционально-целевого подхода с привлечением распределенных вычислений найдено не было.

Поэтому представляется актуальным сначала разработать вспомогательный, несколько упрощенный инструментарий, интегрирующий рассмотренные выше подходы и технологии, а потом использовать его для реализации автоматизированного комплекса ChemObjects, как частного случая построения ИС.

При проектировании, как вспомогательной программной оболочки, так и комплекса ChemObjects уделяется отдельное внимание их модульности, расширяемости и открытости, так как, это способствует, как вторичному использованию существующих частей ИС при построении новых версий, так и адаптации исходного кода таких систем к новым платформам и технологиям.

Вычислительные системы, обладающие “прозрачным” программным интерфейсом являются более универсальными, так как обладают, широкими возможностями расширения. Они могут быть использованы в научных группах и инженерных лабораториях, разрабатывающих собственные расширения для решения своих задач [19]

В настоящее время уделяется большое внимание вопросам построения ИС для распределенных вычислений, как на базе суперкомпьютеров, так и с распараллеливанием по компьютерным сетям (например, с использованием сети Интернет [20]), несмотря на сложности, которые неизбежно преследуют проектирование и реализацию таких систем [21]. Поэтому при разработке вспомогательного инструмента, также делается попытка включить в него возможности построения распределенных ИС для проведения параллельных вычислений.

### ***Проектирование***

Поставленная выше цель построения вспомогательного инструментария для разработки комплекса ChemObjects реализуется в виде последовательности отдельных задач. Рассмотрим их подробнее.

1. Проектирование и реализация автоматизированной системы проектирования ИС (АСПИС) Lolita с гибкой поддержкой идеологии проектирования и подсистем доступа к данным.
2. Разработка и реализация подсистемы доступа данных для АСПИС Lolita для обеспечения возможности удаленного или распределенного доступа к данным.
3. Разработка и реализация идеологии проектирования ИС (в рамках АСПИС Lolita) для целей проведения научных исследований и вычислительных экспериментов с возможностью построения удаленных и/или распределенных ИС с поддержкой проведения параллельных вычислений.
4. Разработка и реализация комплекса ChemObjects с помощью реализованного инструментария в АСПИС Lolita.
  - 4.1. Выбор модели исполнительной среды.
  - 4.2. Разработка декларативной КМ ПО для комплекса ChemObjects.
  - 4.3. Построение системы объектно-ориентированных типов данных для моделирования объектов ПО. Определение совокупности функций-членов классов и исполнителей, которые должны будут их выполнять.
  - 4.4. Построение иерархии задач для ВЭ
  - 4.5. Формирование совокупности покрывающих действий и их сопоставление с исполнителями в соответствии с выбранной моделью исполнительной среды.
  - 4.6. Генерация исходного кода исполнителей и реализация необходимых покрывающих действий (смысловое наполнение обработчиков событий в исполнителях) в сторонних средах разработки.

#### 4.7. Прогон и тестирование комплекса ChemObjects и возможно отделение от АСПИС Lolita.

##### *Автоматизированная система проектирования ИС Lolita*

В качестве вспомогательного инструментария предназначенного для использования при построении комплекса ChemObjects была реализована автоматизированная система проектирования ИС Lolita.

Свойства расширяемости и открытости сделали возможным реализовать требование к построению вспомогательного инструментария, который бы интегрировал концептуальное проектирование на базе функционально-целевого подхода с распределенными вычислениями, как один из возможных подходов, используемых для проектирования ИС в АСПИС Lolita.

АСПИС Lolita базируется на двух понятиях:

1. идеологии проектирования ИС (software designing ideologies);
2. подсистемы доступа к данным (data access systems).

В рамках АСПИС Lolita любая идеология проектирования (ideology) может состоять из произвольного количества этапов (stages) разработки ИС, которые в свою очередь могут быть автоматизированы произвольным количеством инструментов (tools). Инструментами являются исполнимые файлы ОС Windows.

Подсистемы доступа к данным обеспечивают централизованный (и возможно удаленный) доступ к информационным ресурсам, и используются как инструментами проектирования на этапе разработки ИС (в рамках АСПИС Lolita), так и реализуемой программной системой на этапе ее эксплуатации.

Идеологии проектирования ИС и подсистемы доступа к данным разрабатываются отдельно от АСПИС Lolita и (как плагины) подключаются к ней, проходя через процесс регистрации.

На рис. 1. показана схема взаимодействия разработанной идеологии проектирования ИС для научных исследований (НИ) и ВЭ (Software Development Ideology for SE and CE) и подсистемы доступа к данным (Data Access System for SDI). Светлым цветом выделены стрелки показывающие потоки передачи данных, черным цветом отображена передача управления (вызов функций). Многие акты передачи управления сопровождаются также передачей входных данных (параметров вызываемой функции) и заканчиваются возвращением результатов (данных) в тот блок из которого был произведен вызов функции. Для того, чтобы не загромождать рисунок этой информацией такие стрелки на нем не отображены, а только показываются главные направления передачи данных.

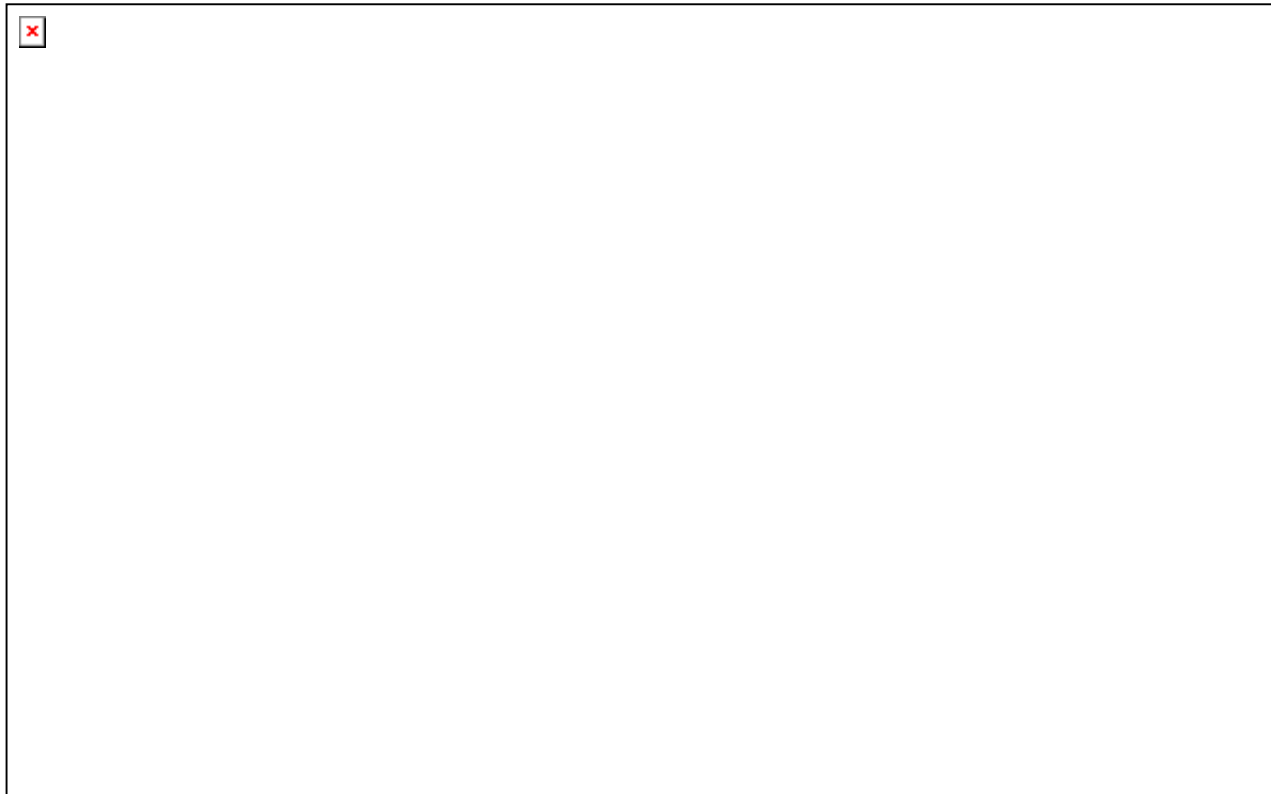


Рис. 1. Схема взаимодействия разработанных идеологии проектирования ИС и подсистемы доступа к данным.

##### **Состав и структура подсистемы доступа к данным**

Подсистема доступа к данным состоит из трех частей названных и описанных ниже.

1. **Менеджер для настройки источника данных (DSMan.exe)** – используется в АСПИС Lolita при создании нового проекта с помощью идеологии проектирования ИС для НИ и ВЭ (рис. 2). Главное назначение – специфицировать источник данных (серверное приложение доступа к данным и имя используемой БД КМ ИС) для приложений (инструментов) автоматизации проектирования.

2. **Серверное приложение** (DASrvApp.exe) – его главной задачей является обработка запросов, получаемых от клиентов, использующих модуль DAUnit, для доступа к базам данных (рис. 3).

3. **Модуль доступа к данным** (DAUnit) – содержит все необходимые механизмы для доступа к серверному приложению (DCOM-интерфейс серверного приложения, компоненты для установления соединений и пр.), а также компоненты для хранения данных на стороне клиента полученных в результате его запросов к серверу.

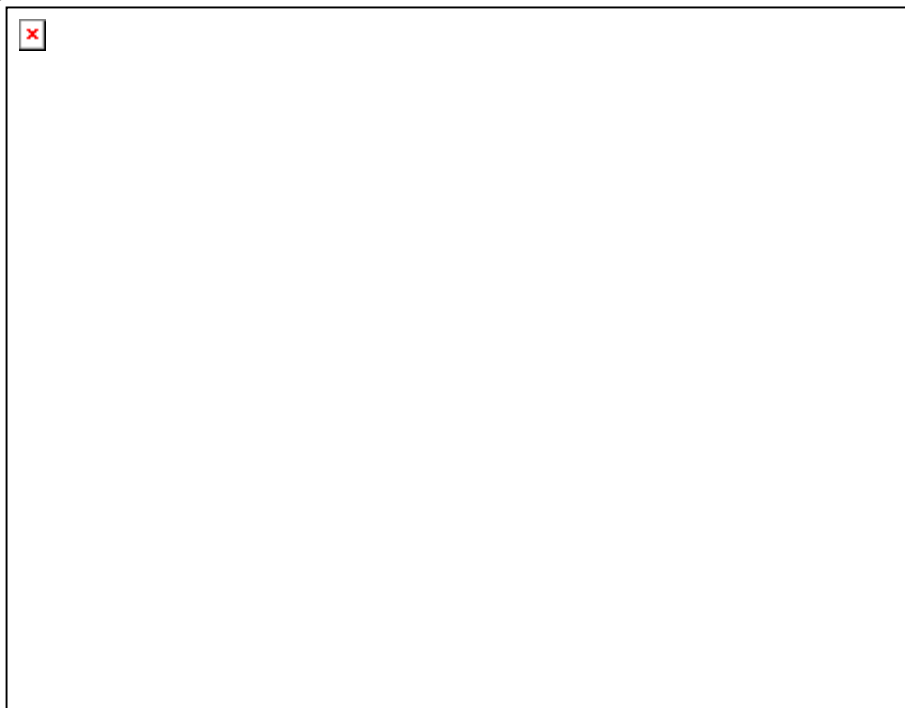


Рис. 2. Серверное приложение подсистемы доступа к данным.

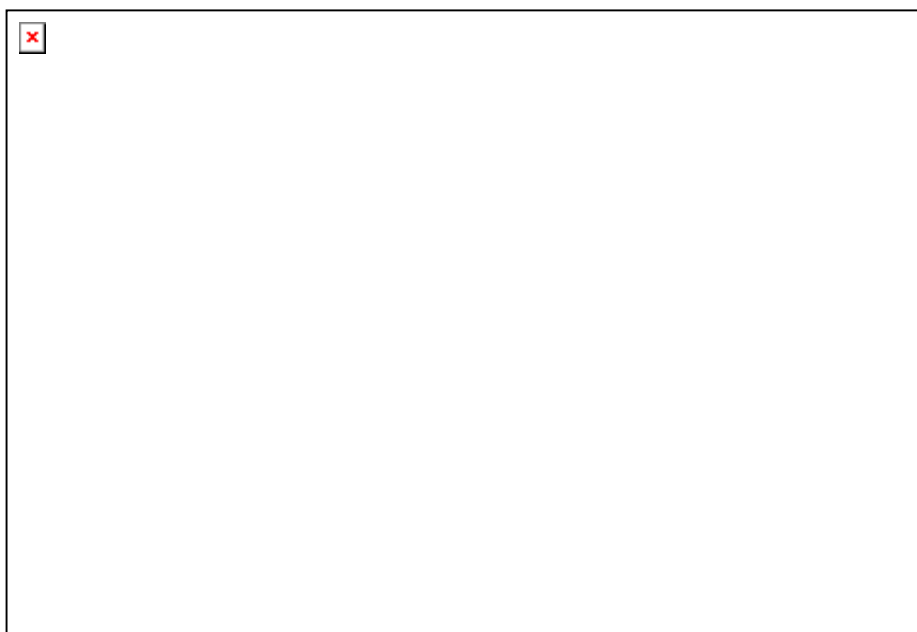


Рис. 3. Серверное приложение подсистемы доступа к данным.

Структуру разработанной подсистемы доступа к данным можно видеть на рис. 1.

Модуль доступа к данным встраивается в приложения, которым будет необходимо подключиться к серверу, т.е. в инструменты автоматизирующие проектирования ИС в АСПИС Lolita и в удаленные исполнители действий (используемые в идеологии проектирования ИС для ВЭ, для покрытия задач КМ ИС).

На стороне сервера все БД можно разделить на два типа: БД КМ ИС и шаблоны БД КМ ИС. Шаблоны предназначены для создания новых, "пустых" БД КМ ИС.

### **Сетевые технологии доступа**

Подсистема доступа к данным опирается на те или иные сетевые протоколы, которые инкапсулируют все специфические особенности организации каналов для доступа к информационным ресурсам, организованных в виде баз данных. В некоторой степени она отвечает за надежность и удобство соединений, предлагая определенный интерфейс взаимодействия с данными.

Реализованная подсистема доступа к данным поддерживает две сетевые технологии доступа: **Microsoft DCOM** (Distributed Component Object Model) [22-25] и **Windows Sockets Architecture** (опирающаяся на сетевой протокол TCP/IP). При использовании соединений по технологии DCOM, компьютерная сеть должна быть основана на технологии Microsoft Windows NT (т.е. в сети должны быть созданы группы пользователей, компьютер, на котором запускается сервер, должен иметь доступ к спискам всех пользователей сети или должна быть установлена сетевая служба каталогов Active Directory и пр.). Для технологии WinSocket достаточно, того, чтобы сеть поддерживала протокол TCP/IP. При использовании TCP/IP подсистема доступа к данным будет функционировать и на сетях основанных на других информационных технологиях (не Microsoft NT), например Novell NetWare.

Для взаимодействия клиентского и серверного приложения подсистемы доступа к данным посредством обеих технологий используются Object RPC (Remote Procedure Call), поэтому на обеих сторонах должна быть включена его поддержка.

Реализация подсистемы доступа к данным была ориентирована на ОС семейства Windows. Основной акцент делается на использование в ОС Windows XP. При использовании в ОС Windows 95/98, может потребоваться наличие дополнительных DLL библиотек для поддержки технологий DCOM, WinSocket и/или технологии БД MIDAS.

### **Основные функции подсистемы**

Рассмотрим основные функции, разработанной и реализованной подсистемы доступа к данным для НИ и ВЭ.

1. Предоставляется сервис, который заключается в обеспечении объектно-ориентированного интерфейса для доступа к БД (поддерживаются БД в формате Microsoft Access Database).
2. Поддерживается доступ к данным, как для локального, так и для удаленного или распределенного по сети варианта вычислительной системы (используется технология построения многозвенных программных систем MIDAS [26-28]).
3. Поддерживается многопользовательский режим доступа к данным как к одной, так и к разным БД КМ ИС.
4. Поддерживается несколько протоколов (DCOM, TCP/IP) для доступа к данным. Обеспечивается независимость ИС и исполнителей действий от технологий и протоколов доступа к данным (для этого используется модуль доступа к данным – DAUnit).
5. Предоставляются сервисные функции для манипуляции файлами БД и шаблонами БД, как для пользователей сервиса (реализуется за счет функций модуля DAUnit), так и для администратора БД (используется графический интерфейс пользователя серверного приложения).
6. Проводится регистрация всех действий и событий, которые инициируют пользователи сервиса в специализированной серверной БД, и предоставляется графический интерфейс пользователя для администратора серверного приложения для их мониторинга (рис. 5).

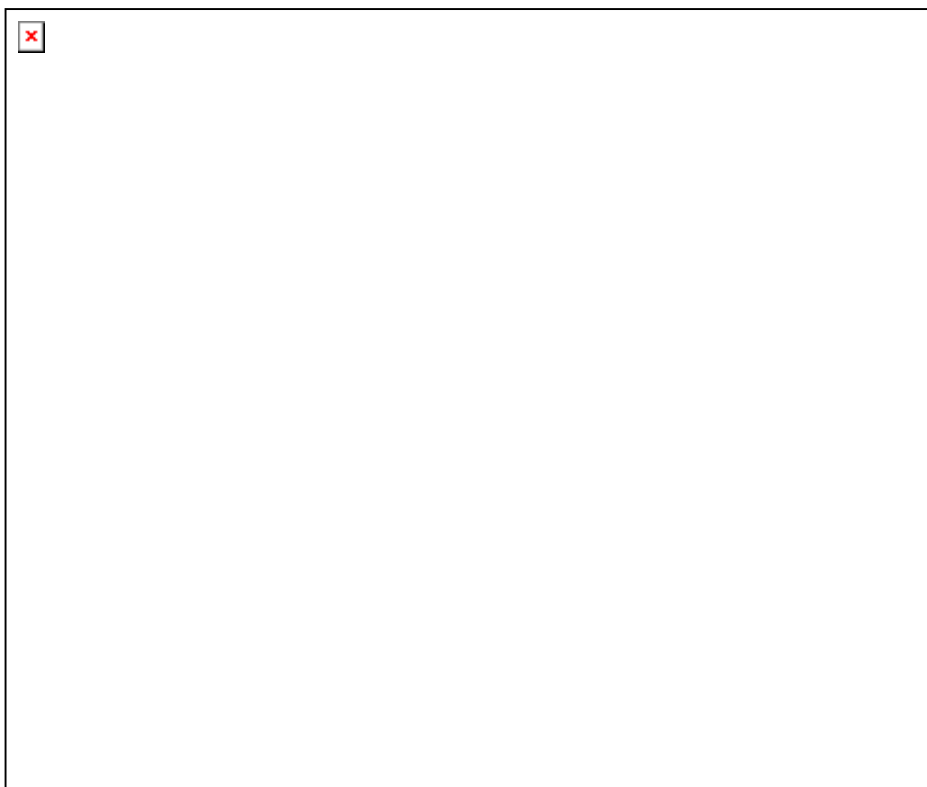


Рис. 5. Интерфейс для мониторинга событий на сервере БД КМ ИС.

7. Поддерживается протокол FTP (File Transfect Protocol) для скачивания и добавления файлов БД и шаблонов БД с и на серверное приложение.

8. Поддерживается динамическое создание/удаление компонент для организации произвольного количества независимых каналов для SQL-запросов для любой БД, которая была открыта клиентом, в рамках одного подключения (возможность реализуется на серверной стороне и модуля DAUnit).

9. Поддерживается возможность организации произвольного количества параллельных, независимых каналов доступа (подключений) из одного клиентского приложения. Это может быть использовано при параллельном выполнении нескольких действий, одним исполнителем.

Для реализации подсистемы доступа к данным использовалась интегрированная среда программирования, поддерживающая методологию RAD и библиотеку визуальных компонент VCL: Borland C++ Builder v 6.0 + Update1,2,4.

К недостатку реализации данной подсистемы доступа к данным для НИ и ВЭ можно отнести отсутствие на стороне серверного приложения и в модуле доступа к данным программного кода, обеспечивающего разрешение конфликтных ситуаций связанных с многопользовательским режимом доступа. Для устранения данного недостатка можно использовать форму обслуживающую ошибки пользователя (Reconcile Error Dialog) и событие инициируемое на стороне клиента OnReconcileError [29, с. 520].

### **Заключение**

Построение АСПИС Lolita, которая интегрирует перспективные подходы и информационные технологии проектирования и реализации ИС, позволяет получить инструментарий для построения комплекса ChemObjects, реализующий программные компоненты для создания КМ ПО, КМ ИС ChemObjects, объектно-ориентированной системы классов, иерархии целей вычислительных экспериментов, совокупность распределенных программных исполнителей.

### **Список использованных источников**

1. Кузьмин И. А., Путилов В. А., Фильчаков В. В. Распределенная обработка информации в научных исследованиях. Л.: Наука, 1991. 304 с.

2. Богатиков В. Н., Сидельников Д. В., Кременецкий В. Г., Программная среда для комплексного анализа структурных моделей кристаллического и расплавленного состояний. Информационные технологии в региональном развитии. Выпуск V, – Апатиты: изд. КНЦ РАН, 2005. с. 129–131.

3. Сидельников Д. В., Богатиков В. Н., Кременецкий В. Г., Построение автоматизированного комплекса ChemObjects для верификации кластерных моделей расплавов. – Информационные технологии в региональном развитии. Вып. IV. – Апатиты, 2004. – С. 77–82.
4. Емельянов С. В., Олейник А. Г., Попков Ю. С., Путилов В. А., Информационные технологии регионального развития. – М.: Едиториал УРСС, 2004. – 400 с.
5. Путилов В. А., Фильчаков В. В., Фридман А. Я., CASE-технология вычислительного эксперимента. – Апатиты: КНЦ РАН, 1994, том 1, 249 с.
6. Путилов В. А., Горохов А. В., Системная динамика регионального развития. Мурманск: НИЦ “Пазори”, 2002. 306 с.
7. Аплтон Б. Образцы и программное обеспечение: основные понятия и определения. <http://www.softcraft.ru/design/patintro/content.shtml>
8. Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влиссидес. Приемы объектно-ориентированного программирования. Паттерны программирования. СПб: Питер, 2001. - 368 с.
9. О. Дубина. Обзор паттернов проектирования. <http://citforum.ru/SE/project/pattern/>
10. Агенты. <http://uchcom.botik.ru/nut/theory.html>
11. Даниил Кальченко. Агенты приходят на помощь. <http://www.compress.ru/Archive/CP/2005/10/7/>
12. Тез. докл. IX Региональной научной конф. (Апатиты 13-14 апр. 2006 г.), Апатиты: изд. КФ ПетрГУ, 2006.
13. Обработка объектный БД в С++. Программирование по стандарту ODMG. - М.: "Вильямс", 2001. - 384 с.
14. Зеленков Ю.А. Центр Интернет ЯрГУ, ВВЕДЕНИЕ В БАЗЫ ДАННЫХ <http://www.mstu.edu.ru/education/materials/zelenkov/toc.html>
15. В. Н. Петров Информационные системы. Учебник. СПб.: Питер, 2002. – 688 с.: ил.
16. С. Д. Кузнецов, Центр Информационных Технологий, Проектирование и разработка корпоративных информационных систем. <http://www.citforum.ru/cfin/prcorpsys/>
17. Путилов В. А., Фильчаков В. В., Фридман А. Я., CASE-технология вычислительного эксперимента. – Апатиты: КНЦ РАН, 1994, том 2, 169 с.
18. Информационные технологии в региональном развитии: Сборник научных трудов ИИММ КНЦ РАН, вып. V. – Апатиты: Издательство КНЦ РАН, 2005. – 147 с.
19. Балашов М.Е. Автоматизированная система подготовки и анализа данных для решения задач вычислительной гидродинамики.: Автореф. дис. на соискание уч. степени кандидата технических наук. Тверь, 2006.–20с.
20. Распределенные вычисления в Интернете. <http://distributed.ru/>
21. Петренко В., Бирюков С. Параллельные вычисления. <http://sigma.math.rsu.ru/content/004/03/>
22. Л. Зельцер. Будущее распределенных вычислительных систем. <http://beda.stup.ac.ru/psf/ziss/wmaster/books/magazine/pcmag/9706/069716.htm>
23. Н. Дубова. COM или CORBA? Вот в чем вопрос. <http://www.osp.ru/os/1999/03/06.htm>
24. Хейфец И. М. DCOM Распределенная COM. <http://ilyailyahome.chat.ru/dcom.htm>.
25. COM / COM+ / DCOM – DISTRIBUTED COMPONENT OBJECT MODEL. <http://www.datahousecorp.com/rus/technology/dcom.htm>
26. MIDAS. Практическое применение. Игнатъев Р. <http://www.rsdn.ru/article/db/midas.xml>
27. Шамис В. А. Borland C++ Builder 6. Для профессионалов – СПб.: Питер, 2005. с. 507 – 524.
28. Технология DataSnap. Механизмы удаленного доступа. [http://www.kib.ru/lib/resources/src19/Secrets\\_Delphi\\_7\\_Book/Chapter20/Index.html](http://www.kib.ru/lib/resources/src19/Secrets_Delphi_7_Book/Chapter20/Index.html)
29. Шамис В. А. Borland C++ Builder. Для профессионалов. – СПб.: Питер, 2005. – 798 с.