

# Программная и аппаратная архитектура сервера визуализации сеточных данных программного комплекса GIMM\_NANO\*

А.И. Плотников

Институт математического моделирования РАН, Москва

В работе рассматривается принципиальная схема архитектуры сервера визуализации сеточных данных программного комплекса GIMM\_NANO, включая интерфейс клиент-серверного взаимодействия. Также рассмотрен принцип работы данной архитектуры на примере разрабатываемой программы визуализации сеток, которая будет включена в программный комплекс сервера визуализации.

## 1. Постановка задачи

Разработать аппаратную и программную архитектуру сервера визуализации сеточных данных. При этом сервер визуализации должен выполнять следующие функции:

1. Обеспечение интерактивного взаимодействия с ПО клиентской части программы визуализации (получение параметров визуализации и отправка обработанных данных, корректное завершение работы по требованию клиента)
2. Работа с файловым хранилищем комплекса GIMM\_NANO для получения входных данных для анализа.
3. Осуществлять параллельную обработку сеточных данных, с учетом полученных от пользователя параметров.

Для обеспечения масштабируемости программного обеспечения сервера визуализации сеточных данных, необходимо было разработать единый программный интерфейс для всех используемых процедур визуализации (изоповерхностей, сеток, изолиний и т.д.)

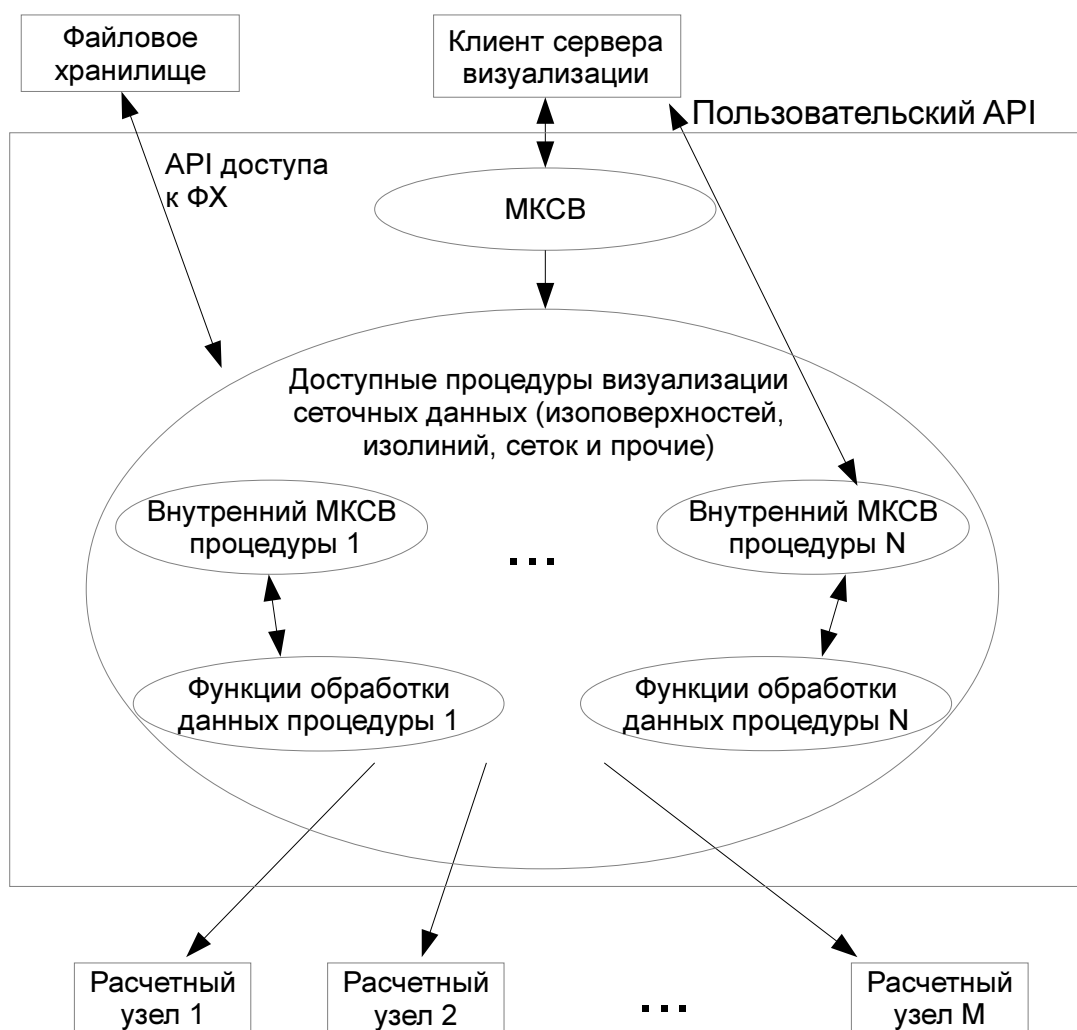
Далее в работе рассматривается архитектура программы визуализации сеточных данных и реализация программного интерфейса на примере разрабатываемого модуля визуализации сеток.

## 2. Архитектура сервера визуализации

Принципиальная схема разработанной архитектуры сервера визуализации сеточных данных проекта GIMM\_NANO представлена на диаграмме (рис. 1).

---

\* Работа выполнена при поддержке РФФИ, проект 11.07.00.779\_a



**Рис. 1.** Принципиальная схема архитектуры сервера визуализации. МКСВ – модуль клиент-серверного взаимодействия. Элементы программной архитектуры изображены в форме эллипса, аппаратной – в форме прямоугольника

Таким образом, можно выделить два основных модуля данного сервера:

- Модуль клиент-серверного взаимодействия (МКСВ)
- Процедуры обработки данных (каждая процедура состоит из внутреннего МКСВ и функций обработки данных)

Основная задача модуля клиент-серверного взаимодействия – предоставить программе-клиенту интерфейс (далее API) для доступа к функциям сервера визуализации, объединив API различных программных компонентов последнего в единый интерфейс. Данный интерфейс описан ниже. Также модуль осуществляет соединение с клиентом, корректное завершение работы сервера визуализации и разрыв соединения по требованию. В качестве клиента могут выступать и две сущности (один на прием данных, другой на передачу). Это полезно в случае действия на клиента определённых лицензионных ограничений. В случае с GIMM\_NANO, в качестве клиента, осуществляющего передачу данных серверу визуализации, выступает управляющая программа, которая контролирует права доступа пользователей всего программного пакета.

Каждая процедура обработки данных состоит из внутреннего МКСВ и набора функций обработки данных. Внутренний МКСВ процедуры получает на вход параметры соединения и имя файла сетки, считывает данную сетку из файлового хранилища, получает от пользователя все

необходимые параметры и запускает функции обработки сетки, возвращая результат пользователю. Требования к данному модулю описаны в следующем разделе. Функции обработки данных используют для проведения необходимых вычислений расчётные узлы сервера визуализации, осуществляя распределение данных между выделенными узлами и балансировку нагрузки.

## 2.1 API модуля клиент-серверного взаимодействия

Каждая процедура визуализации должна иметь функцию с унифицированным именем, которая получает на вход параметры соединения и осуществляет взаимодействие с клиентом, получая от него необходимые параметры, проверяя их, вызывая функции параллельной обработки данных и отправляя результат клиенту. Возможные возвращаемые значения функции описаны ниже. В терминологии данной статьи описываемая функция называется внутренним модулем клиент-серверного взаимодействия (внутренний МКСВ). Далее представлен алгоритм взаимодействия клиента программы визуализации с сервером и описаны основные требования к внутреннему МКСВ

Взаимодействие клиента программы визуализации с сервером осуществляется посредством МКСВ следующим образом:

1. Клиент запрашивает соединение. В случае успеха, сервер высылает приветственное сообщение и ждет команды.
2. Клиент передаёт название процедуры визуализации, которой собирается воспользоваться и имя файла сетки, на которой будет производиться работа. Сервер анализирует файл сетки на корректность и отправляет результат пользователю.
3. Далее вызывается внутренний МКСВ выбранной процедуры. Возвращает функция одно из следующих значений:
  - a. Флаг успешного выполнения всех инструкций
  - b. Флаг ошибки выполнения и её код, позволяющий определить тип ошибки, среди которых: неверные входные параметры, ошибка самой программы обработки данных, потеря связи с клиентом.
  - c. Сигнал завершения работы процедуры по требованию клиента. В самой же процедуре визуализации реализован функционал максимально быстрой остановки обработки данных по требованию клиента. Команда завершения работы универсальная для всех процедур.
  - d. Сигнал завершения работы сервера.
4. Соответственно, в зависимости от результата вызова предыдущей функции, МКСВ делает одно из следующих действий:
  - a. В случае успешного выполнения, переходит к п.5.
  - b. В случае возникновения ошибки, сообщает об этом пользователю и переходит к п.5.
  - c. В случае сигнала о завершении работы процедуры, сообщает пользователю, что процедура успешно остановлена, переходит к п.5.
  - d. Закрывает соединение, сообщая об этом клиенту, и завершает свою работу.
5. Ожидание одной из следующих команд от пользователя:
  - a. Изменение файла сетки и/или процедуры обработки (визуализации). При этом клиент передает оба параметра, а на стороне сервера осуществляется проверка того, какой параметр изменился. Далее идет переход к п.3.
  - b. Передача новых параметров. В данном случае программа переходит к п.3.
  - c. Выход из программы.

Далее рассмотрена архитектура процедуры визуализации сеток, разрабатываемой на основе данного архитектурного шаблона.

### 3. Архитектура процедуры визуализации сеток

Программное обеспечение визуализатора сеточных данных позволяет анализировать сетки большого объема, осуществляя огрубление данных удаленно на сервере визуализации (в качестве которого может выступать многопроцессорная система). Система визуализации позволяет перемещаться по трехмерной сетке большого объема по принципу карт, позволяя выделять и уточнять необходимые участки, изображая лишние участки лишь в общих чертах. Данная программа хорошо приспособлена к решению таких задач, как поиск элементов сетки, удовлетворяющих (или не удовлетворяющих) определенным условиям; проверка качества алгоритмов декомпозиции и балансировки нагрузки при решении сеточных задач.

Полученные сеточные данные программа приводит к внутреннему формату, в котором структура доступа к доменам сетки имеет вид дерева и который позволяет максимально быстро получать различные области сетки с заданным уровнем детализации, существенно экономя, благодаря этому, время и вычислительные ресурсы аппаратного обеспечения сервера визуализации. Недостатком такого подхода является относительно долгое время первоначального анализа сетки. Алгоритм хорошо адаптирован для применения на многопроцессорных системах, что позволяет решить описанную проблему при условии наличия достаточных вычислительных ресурсов.

Программное обеспечение данной системы визуализации состоит из 4-х базовых модулей:

1. Внутренний МКСВ. Функциональное назначение описано в разделе 2. Описание API приведено ниже.

2. Средство декомпозиции сеток. Данный модуль позволяет агрегировать узлы сетки в домены с приблизительно одинаковым количеством узлов внутри каждого домена, благодаря чему можно оценивать плотность сетки в наблюдаемой точке, не отображая всю сетку целиком. Также, цветовая маркировка домена может показывать интересующие интегральные характеристики сетки внутри данного домена (количество не удовлетворяющих определенному условию элементов и т.д.). Данный модуль имеет как последовательную, так и параллельную реализации, осуществляя максимально равномерное разбиение между доменами по количеству узлов.

3. Модуль огрубления поверхностей. Позволяет осуществлять огрубление поверхностей доменов для сокращения объема данных, хранящихся на сервере и пересылаемых клиенту. Модуль имеет параллельную реализацию и осуществляет огрубление с заданной точностью.

4. Ядро программы — процедура построения дерева иерархии доменов. Данный модуль рекурсивно применяет процедуру декомпозиции сеток (сначала параллельно, разбивая исходный объект на домены, а потом и каждый домен в отдельности, уже последовательно на вычислительных узлах многопроцессорной системы). После этого осуществляется параллельное огрубление получившихся поверхностей и объединение доменов в дерево по принципу вложенности.

Блок-схема работы сервера визуализации имеет вид, изображенный на рис. 2.

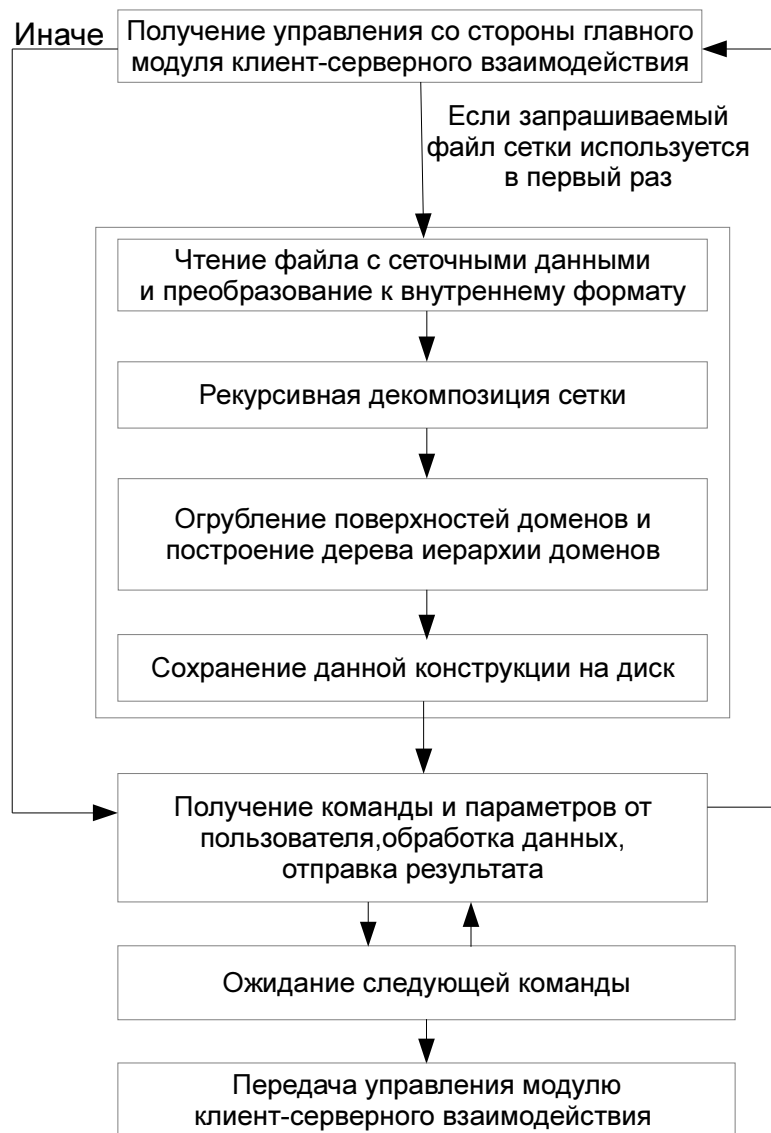


Рис. 2. Блок-схема работы процедуры визуализации сеток

### 3.1 API внутреннего МКСВ процедуры визуализации сеток

На основе разработанного архитектурного шаблона, в соответствии с описанием процедуры визуализации сеток, приведённым выше, API внутреннего МКСВ данной процедуры имеет следующий вид:

1. Получение управления от МКСВ, ожидание параметров от клиента
2. Клиент отправляет параметры отображаемой области сетки (прямоугольная область, задающаяся двумя точками). В случае наличия ошибок в данных, в МКСВ передается флаг ошибки входных данных (всю остальную работу МКСВ делает самостоятельно)
3. Клиент отправляет характерный размер поддомена (эффективный диаметр). Осуществляется проверка, как в п. 2.
4. Получение со стороны клиента дополнительных параметров отображения (если используются функции выделения цветом и т.п.)
5. Осуществляется обработка данных сетки по алгоритму, описанному выше. В случае возникновения ошибки в МКСВ передается её флаг в соответствие с п 3.b раздела 2.1 данной статьи. Аналогично реализованы функции пунктов 3.c и 3.d.

## 6. Передача управления МКСВ с флагом успешного выполнения процедуры

### Заключение

Таким образом, разработана программная и аппаратная архитектура сервера визуализации сеточных данных, удовлетворяющая следующим требованиям:

1. Система реализует все функции, заявленные в п.1 «Постановка задачи»
2. Обладает хорошим потенциалом масштабируемости за счет единого интерфейса всех программных модулей

Данная архитектура легла в основу программной реализации визуализатора GIMM\_Data\_Visualizer, являющегося частью комплекса GIMM\_NANO, разрабатываемого в институте математического моделирования РАН, Москва.

### Литература

1. Б. Н. Четверушкин, В. А. Гасилов, С. В. Поляков, Е. Л. Карташева, М. В. Якобовский, И. В. Абалакин, В. Г. Бобков, А. С. Болдарев, С. Н. Болдырев, С. В. Дьяченко, П. С. Кринов, А. С. Минкин, И. А. Нестеров, О. Г. Ольховская, И. В. Попов, С. А. Суков. Пакет прикладных программ GIMM для решения задач гидродинамики на многопроцессорных вычислительных системах // Матем. моделирование, 17:6 (2005), 58–74.
2. С. В. Поляков, Т. А. Кудряшова, А. А. Свердлин, Э. М. Кононов, О. А. Косолапов. Параллельный программный комплекс для решения задач механики сплошной среды на современных многопроцессорных системах // Матем. моделирование, 22:6 (2010), 132–146.
3. Якобовский М.В. Обработка сеточных данных на распределенных вычислительных системах // Вопросы атомной науки и техники, Сер. «Математическое моделирование физических процессов», 2 (2004), 40-53.
4. S.Polyakov, M. Iakobovski. GEOMETRICAL SIMULATION AND VIZUALIZATION IN NANO-ELECTRONICS PROBLEMS // COMPUTER GRAPHICS & GEOMETRY, 11:1 (2009), 46-68.
5. M. Iakobovski, I. Nesterov, P. Krinov. LARGE DISTRIBUTED DATASETS VISUALIZATION SOFTWARE, PROGRESS AND OPPORTUNITIES // COMPUTER GRAPHICS & GEOMETRY, 9:2 (2007), 1-19.