

Васильев А.В. – диссертант кафедры Автоматизированных систем управления

РГУ нефти и газа имени И.М. Губкина

ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ КОМПОНЕНТНОЙ СРЕДЫ
ПАРАЛЛЕЛЬНОГО И РАСПРЕДЕЛЁННОГО МОДЕЛИРОВАНИЯ СИСТЕМ
ГАЗОСНАБЖЕНИЯ

Введение

Система поддержки принятия диспетчерских решений (СППДР) автоматизированной системы диспетчерского управления (АСДУ) Единой системой газоснабжения (ЕСГ) России нуждается в модернизации. Перед ней ставятся новые задачи: интеграция с создаваемым единым информационным пространством (ЕИП) АСДУ ЕСГ, функционирование в распределённом режиме, быстрое решение вычислительно-сложных расчётных задач моделирования систем газоснабжения (СГ) [1].

На сегодняшний день СППДР АСДУ состоит из разнородных программно-вычислительных комплексов (ПВК), создаваемых независимыми коллективами разработчиков. В составе большинства ПВК выделен расчётный модуль, выполняющий численное решение задач, основанных на моделировании СГ. Расчётные модули, как правило, разрабатываются в течение длительного времени с использованием монолитной архитектуры – высока степень взаимозависимости, взаимопроникновения между частями модуля, что крайне затрудняет внесение в него существенных изменений, таких как переход к использованию параллельных, и, тем более, распределённых вычислений.

Существующие ПВК СППДР испытывают затруднения с решением наиболее вычислительно-сложных задач, таких как моделирование нестационарных режимов крупных СГ (с десятками тысяч моделируемых объектов) в реальном режиме времени; расчёт плана перехода от одного режима транспорта газа к другому; идентификация параметров модели СГ; расчёт квазиоптимальных режимов транспорта газа, и других. Необходимым условием быстрого решения таких задач является использование параллельной и распределённой обработки данных [2].

Наличие многочисленных разнородных ПВК затрудняет согласование результатов их работы, интеграцию в единую систему. Монолитный характер расчётных модулей затрудняет переход от однопоточных вычислений - к параллельным; от однопользовательского локального режима работы - к многопользовательскому распределённому; от использования собственных форматов представления данных, расчётных процедур - к единым открытым форматам, согласованным расчётным процедурам, интеграции с ЕИП АСДУ.

Решением перечисленных проблем модернизации может стать разработка набора вычислительных сервисов, обеспечивающих вычислительные потребности ПВК СППДР, функционирующих в распределённой иерархии АСДУ – и переход ПВК к их использованию. Разработка таких сервисов возможна на базе единой среды моделирования, обеспечивающей единство и согласованность вычислительных сервисов, модульность и гибкость системы, поддержку параллельных и распределённых вычислений, функционирование в распределённой среде. Такая среда моделирования описана в данной статье.

Основные архитектурные решения построения среды моделирования

Использование сервис-ориентированного подхода

Предложенная среда моделирования построена на базе сервис-ориентированной парадигмы, то есть состоит из взаимодействующих сервисов – распределённых, слабо связанных компонент, обладающих стандартизированным описанием интерфейса и взаимодействующих по стандартизированным протоколам.

Из различных альтернативных средств реализации сервис-ориентированной архитектуры (web-сервисы, Java EE, WCF, CORBA, DCOM и других) на основании сравнительного анализа выбрано программное обеспечение промежуточного уровня ZeroC ICE (Internet Communication Engine).

На рис. 1 представлена схема взаимодействия сервисов в распределённой среде.



Рис. 1. Схема взаимодействия сервисов в распределённой среде

Основными элементами распределённой системы являются вычислительные узлы N_i . Каждый из узлов содержит определённый набор вычислительных ресурсов, работает под управлением той или иной операционной системы. Узлы связаны между собой сетями передачи данных.

Сервис состоит из трёх частей: описания интерфейса, объекта-исполнителя и объекта-посредника. Интерфейс сервиса представляет собой описание множества предоставляемых им методов, сделанное на специализированном языке описания интерфейсов. Объект-исполнитель сервиса выполняется на сервере и реализует все методы, объявленные в интерфейсе сервиса. Объект-посредник используется клиентом для удалённого вызова методов исполнителя.

Среда времени выполнения (Runtime) ZeroC ICE играет роль адаптера между исполнителем и посредником сервиса, что позволяет использовать для их реализации весь набор поддерживаемых языков программирования: C++, Java, C#, Objective-C и др.; операционных систем: Windows, Linux, Mac OS и др.

Использование сервис-ориентированного подхода обеспечивает прозрачность расположения и реализации компонент распределённой системы, высокую степень модульности и повторного использования компонент системы.

Сопоставление вычислительных сервисов и вычислительных ресурсов на основе типов

Все сервисы в разработанной среде разделены на две группы: вычислительные сервисы, непосредственно выполняющие вычисления; и служебные сервисы, организующие вычислительные процессы в распределённой среде.

На сегодняшний день сформировался ряд различных типов вычислительных ресурсов, обладающих различными областями применения и технологиями программирования. Они представлены в табл. 1.

Табл. 1

Типы вычислительных ресурсов, их назначение и средства программирования

Тип вычислительного ресурса	Средства программирования	Назначение вычислительного ресурса	Применение для моделирования СГ
Многоядерный процессор	OpenMP, TBB, PPL, библиотеки численных методов и др.	Вычисления общего назначения	Организация вычислительного процесса; моделирование сложных объектов
GPGPU-видеокарта, математический сопроцессор	CUDA, OpenCL, C++ AMP, OpenACC, библиотеки численных методов и др.	Высоко-параллельные вычисления с применением модели программирования SIMD	Моделирование многочисленных простых объектов, например, трубопроводов
Вычислительный кластер; супер-компьютер	MPI и др.	Наиболее вычислительно-сложные задачи	Решение вычислительно-сложных задач: идентификации параметров модели и др.

Для сопоставления вычислительным сервисам необходимых для их выполнения вычислительных ресурсов введена концепция типа комплекта вычислительных ресурсов. Из вычислительных ресурсов узлов системы формируются комплекты, каждому из которых присваивается тип. В свою очередь, каждому вычислительному сервису

сопоставляется тип комплекта вычислительных ресурсов, подходящего для его выполнения.

Сопоставление комплектов вычислительных ресурсов и вычислительных сервисов выполняется следующим образом. На каждом комплекте вычислительных ресурсов запускается служебный сервис управления комплектом ресурсов. Его интерфейс предоставляет методы для запуска в рамках управляемого вычислительного ресурса множества вычислительных сервисов, совместимых с ним по типу.

Явное управление вычислительными ресурсами

В распространённых реализациях сервис-ориентированной архитектуры для клиента работа с вычислительными ресурсами прозрачна – клиент работает с сервисами, удалёнными объектами, но не управляет вычислительными ресурсами. Такой подход позволяет упростить реализацию балансирования нагрузки, масштабирования, отказоустойчивости.

В разработанной среде предложен и реализован механизм явного управления вычислительными ресурсами. Это позволило минимизировать накладные расходы удалённого взаимодействия сервисов и добиться контроля над организацией вычислительных процессов в распределённой среде.

Основным механизмом реализации явного управления вычислительными ресурсами является сервис диспетчеризации комплектов вычислительных ресурсов [3]. Схема явного управления вычислительными ресурсами представлена на рисунке 2.



Рис. 2. Схема явного управления вычислительными ресурсами; У.С. – управляющий сервис

На этапе настройки системы определяется множество узлов, управляемых сервисом диспетчеризации. В рамках этих узлов запускаются служебные сервисы управления комплектами вычислительных ресурсов. Объекты-заместители управляющих сервисов передаются сервису диспетчеризации и собираются им в типизированные пулы.

Для использования вычислительного сервиса клиент выполняет следующую последовательность операций.

1. Клиент запрашивает у сервиса диспетчеризации вычислительных ресурсов комплект вычислительных ресурсов необходимого типа.
2. Сервис диспетчеризации передаёт клиенту объект-посредник сервиса управления комплектом вычислительных ресурсов необходимого типа. При этом посредник изымается из соответствующего пула – поэтому другой клиент не сможет обратиться к этому же вычислительному ресурсу в это же время – тем самым достигается монопольный характер захвата вычислительного ресурса. Далее клиент работает напрямую с захваченным вычислительным ресурсом.
3. Клиент использует методы сервиса управления захваченным комплектом вычислительных ресурсов для запуска необходимых вычислительных сервисов.

4. Клиент использует запущенные вычислительные сервисы.
5. Клиент освобождает захваченный вычислительный ресурс. Объект-посредник его управляющего сервиса возвращается в пул сервиса диспетчеризации и может быть захвачен другими клиентами.

Предложенный механизм диспетчеризации позволяет достичь монопольного характера захвата вычислительного ресурса клиентом, обеспечить прямую работу клиента с вычислительным ресурсом, а также предоставляет средства управления конфигурацией вычислительного процесса в распределённой среде.

При запросе комплекта вычислительных ресурсов клиент выбирает, к какому из доступных сервисов диспетчеризации обратиться с соответствующим запросом – тем самым он определяет множество ресурсов, в рамках которого будет выполняться необходимый ему вычислительный сервис. Различные режимы работы системы в зависимости от взаимного расположения клиентов, вычислительных ресурсов и сервисов диспетчеризации представлены в таблице 2.

Табл. 2.

Различные режимы работы системы в зависимости от взаимного расположения клиентов, вычислительных ресурсов и сервисов диспетчеризации

Режим работы	Расположение сервиса диспетчеризации и вычислительных ресурсов	Особенности режима работы
Локальный	На компьютере пользователя	Расчёты происходят локально; набор вычислительных сервисов ограничивается комплектом вычислительных ресурсов пользователя
Клиент-серверный	В рамках вычислительного сервера; в облаке	Клиенты свободны от вычислительной работы; возможно использование эластичных облачных вычислительных ресурсов
С использованием ресурсов клиентов	Диспетчер – в локальной сети; ресурсы – на компьютерах клиентов	Используется вычислительный потенциал оборудования клиента; возможности каждого клиента расширяются за счёт использования ресурсов соседей
Распределённый	В распределённой среде, в соответствии с иерархией АСДУ	Создание вычислительных процессов, соответствующих иерархии АСДУ, например, для многоуровневых диспетчерских тренировок

На основе комбинирования приведённых режимов работы достигается как наиболее полное использование вычислительных ресурсов, так и управление конфигурацией вычислительного процесса в распределённой среде.

Проектирование и реализация вычислительных сервисов

В качестве апробации предложенной среды были разработаны вычислительные сервисы расчёта режима СГ и идентификации эмпирических параметров модели СГ, проведены вычислительные эксперименты в различных режимах работы.

Вычислительный сервис расчёта режима систем газоснабжения

Алгоритм расчёта режима СГ приведён в [4]. Он содержит два основных вычислительных элемента, требующих использования параллельных вычислений: решение системы линейных алгебраических уравнений (СЛАУ) и моделирование дуг расчётной схемы.

Схема организации вычислительного процесса расчёта режима СГ представлена на рис. 3.

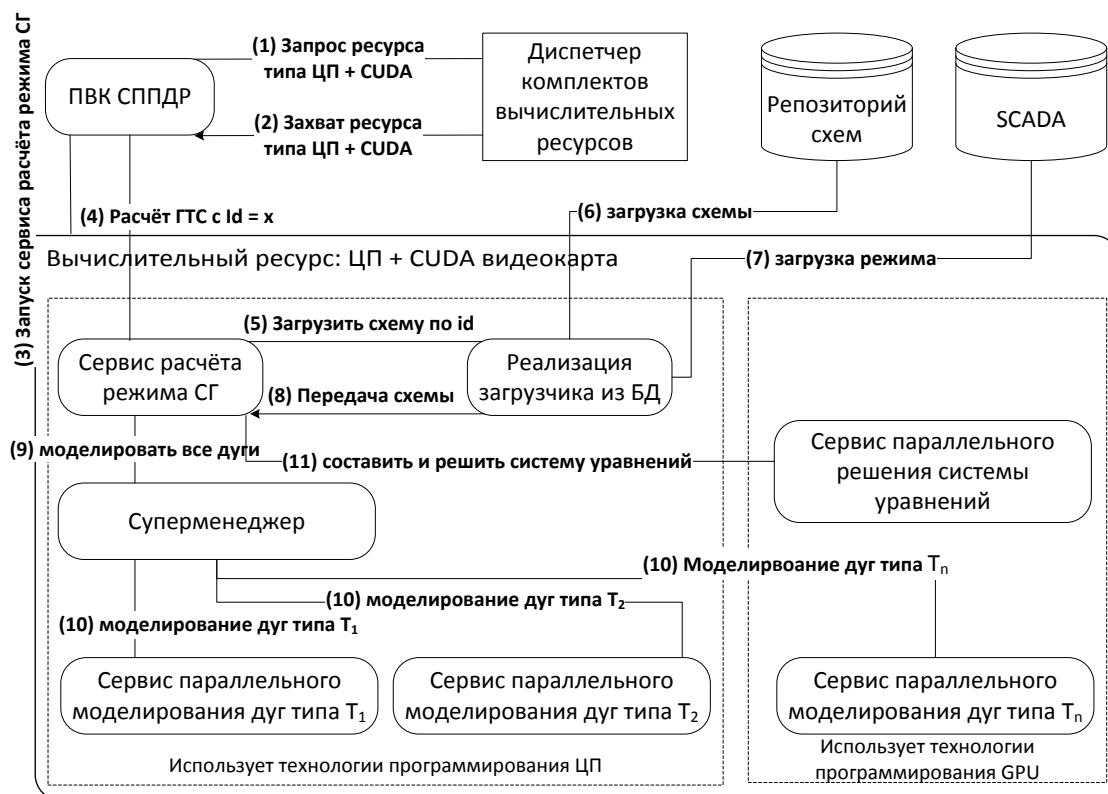


Рис. 3. Схема вычислительного процесса расчёта режима системы газоснабжения

Для эффективного решения СЛАУ разработано семейство взаимозаменяемых вычислительных сервисов, реализующих общий интерфейс на базе различных численных библиотек: Intel MKL как для плотных, так и для разреженных матриц; CUSP – для решения разреженных систем на CUDA-видеокартах.

Для параллельного моделирования объектов-дуг расчётной схемы разработано специализированное решение [5]. Оно основано на группировке однотипных объектов в векторы и выделении для каждого типа объектов менеджера параллельного моделирования, осуществляющего эффективное размещение и параллельное моделирование объектов каждого типа. Для координации менеджеров параллельного моделирования используется сервис Суперменеджер.

Представленный на рис. 3 комплект вычислительных ресурсов содержит центральный процессор и GPGPU-видеокарту. На процессоре выполняются сервисы, организующие вычислительный процесс, осуществляющие загрузку и выгрузку данных, сервис суперменеджер, часть сервисов параллельного моделирования дуг. На графической карте выполняется сервис решения систем уравнений и сервис моделирования трубопроводов – так как это многочисленные и простые объекты, подходящие для моделирования с использованием высокопараллельных вычислений.

Организация вычислительного процесса идентификации параметров модели СГ в распределённой среде

Алгоритм решения задачи идентификации эмпирических параметров модели представлен в [4]. Этот алгоритм использует задачу расчёта режима СГ в качестве основного вычислительного элемента – алгоритм итерационный, и на каждой итерации осуществляются многочисленные расчёты режимов СГ при различных значениях вектора эмпирических параметров модели. Так как расчёт даже одного режима СГ может целиком использовать вычислительные ресурсы одного узла, целесообразно разделить вычислительную нагрузку на несколько узлов в распределённой среде. Схема организации такого вычислительного процесса представлена на рис. 4.

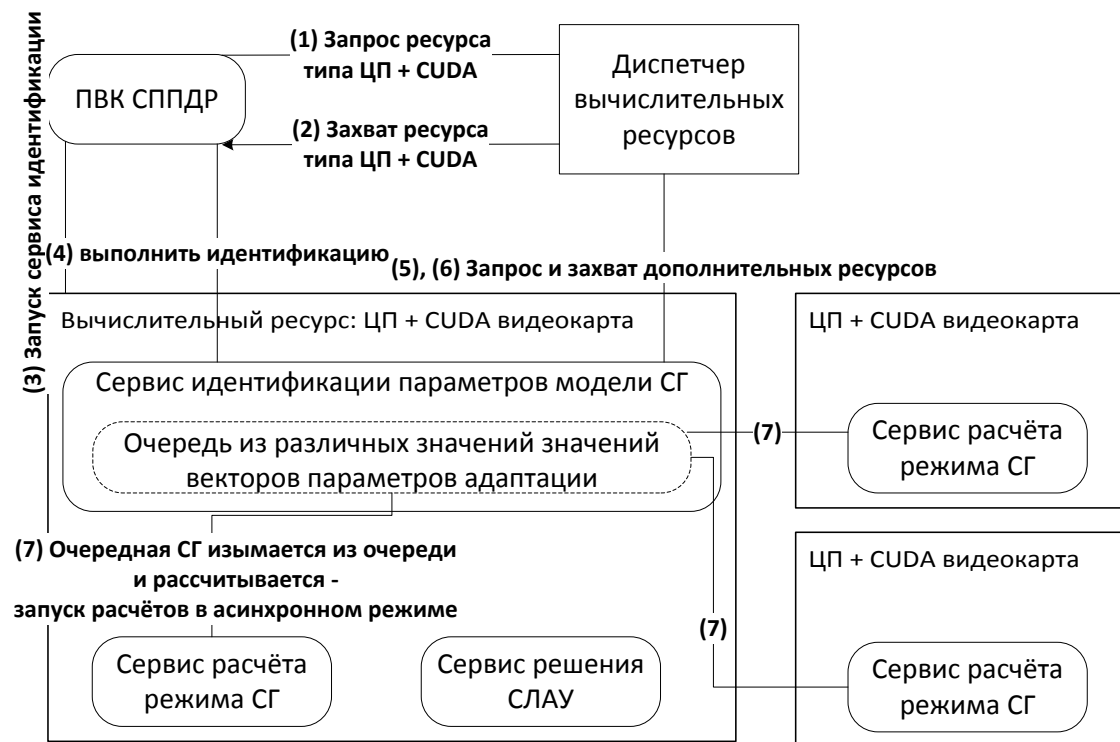


Рис. 4. Схема вычислительного процесса идентификации эмпирических параметров модели

Вычислительный процесс идентификации эмпирических параметров модели организован следующим образом.

1. Клиентский ПК СППДР обращается к сервису диспетчеризации вычислительных ресурсов для захвата вычислительного ресурса, приспособленного к выполнению сервиса идентификации эмпирических параметров модели.
2. Диспетчер выделяет запрошенный вычислительный ресурс.
3. Клиент запускает на захваченном ресурсе сервис идентификации.
4. Клиент вызывает метод «Выполнить идентификацию» запущенного сервиса.
5. Сервис идентификации обращается к сервису диспетчеризации для захвата дополнительных вычислительных ресурсов, приспособленных к выполнению сервиса расчёта режима СГ.
6. Диспетчер выделяет запрошенные ресурсы.
7. Сервис идентификации на каждой итерации формирует очередь векторов эмпирических параметров СГ, при которых необходимо осуществить расчёт режима СГ. В асинхронном режиме на захваченных ресурсах запускаются задачи расчёта режима СГ с очередным значением вектора адаптации вплоть до исчерпания очереди.

Описанная организация вычислительного процесса сервиса идентификации эмпирических параметров модели СГ иллюстрирует как в разработанной среде сочетаются параллельные и распределённые вычисления.

Параллельные вычисления активно используются на уровне базовых вычислительных сервисов, таких как расчёт режима СГ, решение СЛАУ. Сервисы более высокого уровня комбинируют базовые сервисы в распределённой среде, добавляя к параллельным вычислениям распределённые.

При этом распределение нагрузки может достигаться не только за счёт комбинирования сервисов, но и непосредственно за счёт использования технологий распределённого программирования. Например, сервис идентификации может быть реализован на основе технологии программирования MPI и запущен на комплекте вычислительных ресурсов, представляющем собой вычислительный кластер с настроенной средой выполнения MPI.

Интеграция разработанной среды с ПВК СППДР

Разработана общая методология поэтапного перехода ПВК СППДР к использованию вычислительных сервисов, разработанных на базе предложенной среды. Интеграция осуществляется в два этапа.

На первом этапе вводится модуль-адаптер, реализующий интерфейс взаимодействия графического интерфейса ПВК с его расчётным модулем. Это позволяет сделать переход к использованию нового вычислительного обеспечения прозрачным для графического интерфейса. Модуль-адаптер определяет доступность запрошенной интерфейсом пользователя вычислительной задачи в новой вычислительной среде – если доступен соответствующий вычислительный сервис, используется он; если нет – используется старый расчётный модуль.

На втором этапе функциональность адаптера постепенно включается в графический интерфейс пользователя - он начинает работать с новыми вычислительными сервисами, используя предоставленный программный интерфейс.

По данной методологии осуществляется интеграция с ПВК «Веста-тренажёр», разрабатываемым на кафедре Автоматизированных систем управления РГУ Нефти и газа имени им. Губкина и внедрённого в различных газотранспортных и газодобывающих предприятиях.

Проведение и анализ результатов вычислительных экспериментов

Интеграция с ПВК «Веста-тренажёр» позволила удостовериться в корректности разработанных вычислительных сервисов – результаты их расчётов хорошо согласуются с результатами расчётного модуля ПВК «Веста-тренажёр».

Для измерения производительности вычислительных сервисов разработан специальный механизм тестирования, позволивший осуществить тестирование сервисов в различных режимах работы – локальном, сетевом, распределённом. Наиболее показательны результаты, приведённые в табл. 3 и табл. 4.

Табл. 3.

Эффект применения параллельных вычислений к задаче расчёта режима СГ

Использование ПВ	Доля времени выполнения			Ускоре- ние (раз)
	Решение системы уравнений	Модели- рование объектов	Работа с расчётным графом	
Без ПВ	До 98%	2%		1
ПВ для решения системы уравнений	5%	85%	10%	20 раз
ПВ для моделирования расчётных объектов	15%	5%	80%	До 120 раз

В табл. 3 представлены результаты профилирования вычислительного сервиса расчёта режима СГ без использования параллельных вычислений; с использованием параллельных вычислений для решения системы уравнений; дополнительно с использованием параллельных вычисления для моделирования расчётных объектов. Тестирование производилось на современном персональном компьютере Intel Core i7, 2xGeForce GTX 460.

Анализ эксперимента показывает, что использование параллельных вычислений позволяет достичь многократного ускорения решения расчётных задач даже при локальной работе на обычном персональном компьютере.

В табл. 4 представлены результаты решения задачи идентификации эмпирических параметров модели СГ. Клиент всегда выполнялся на компьютере Notebook, а сервер располагался соответственно локально на Notebook; в локальной сети на Desktop; в облаке Amazon Web Services (AWS) на виртуальной машине; на группе из 4x виртуальных машин AWS, соединённых скоростной сетью передачи данных.

Табл. 4.

Время решения задачи идентификации параметров модели СГ в различных режимах работы

Расположение вычислительных ресурсов	Время на клиенте (сек)	Время на сервере (сек)	Доля накладных расходов на передачу данных	Ускорение (раз)
Локально на Notebook	155,53	155,54	0%	1
В локальной сети на Desktop	46,88	46,11	1,64%	3,31
На виртуальной машине AWS	59,6	55,96	4,56%	2,6
На группе из 4х виртуальных машин в AWS	16,07	14,9	7,28%	9,56

Анализ этого эксперимента показывает, что для клиента целесообразно использование превосходящих по мощности удалённых вычислительных ресурсов для решения вычислительно-сложных задач. При этом в зависимости от расположения удалённых вычислительных ресурсов доля накладных расходов на передачу данных может быть значительной.

Задача идентификации, основанная на многократном расчёте режима СГ, хорошо масштабируется в распределённой среде. Это достигается во многом за счёт минимизации накладных расходов, благодаря прямому взаимодействию клиента и сервера, монопольному захвату вычислительного ресурса и поддержке операций с сохранением состояния, реализованных на основе явного управления вычислительными ресурсами.

Заключение

Предложенная компонентная среда организации вычислительных процессов моделирования СГ предоставляет полную поддержку параллельных и распределённых вычислений, реализует единую платформу для создания и предоставления вычислительных сервисов ПВК СППДР.

Благодаря высокой степени модульности, система приспособлена к дальнейшему развитию. Она имеет как краткосрочную, так и долгосрочную перспективу. На 2013й год запланирован переход ПВК «Веста-тренажёр» к использованию вычислительных сервисов, реализованных на основе разработанной среды. В то же время предложенная среда создаёт предпосылки для безболезненной интеграции с создаваемым единым

информационным пространством АСДУ, для обеспечения ПВК СППДР согласованными вычислительными сервисами и формирования единого вычислительного пространства СППДР.

Литература

1. Функциональные требования к модернизированной автоматизированной системе диспетчерского управления единой системы газоснабжения (АСДУ ЕСГ). – 2009.
2. *Herb Sutter*. The Free Lunch Is Over: A Fundamental Turn Toward Concurrency in Software // *Dr. Dobbs's Journal*. – 2005. - №.30(3).
3. *Васильев А.В., Леонов Д.Г., Сарданашвили С.А., Швечков В.А.* Архитектурные решения интеграции расчетных комплексов моделирования режимов систем газоснабжения в единое информационное пространство автоматизированной системы диспетчерского управления ОАО «ГАЗПРОМ» // Газовая промышленность. М: 2012, № 12.
4. *Сарданашвили С. А.* Расчётные методы и алгоритмы (трубопроводный транспорт газа). М.:ФГУП Изд-во «Нефть и газ» РГУ нефти и газа им. И.М. Губкина, 2005.
5. *Васильев А.В. Сарданашвили С.А.* Программная архитектура для использования параллельных вычислений при численном балансировании потоков газа на расчётном графе газотранспортной системы. // Автоматизация, телемеханизация и связь. – 2012. - №1. 38-44 с.

Иллюстрации

- Рис. 1. Схема взаимодействия сервисов в распределённой среде.
- Рис. 2. Схема явного управления вычислительными ресурсами.
- Рис. 3. Схема вычислительного процесса расчёта режима системы газоснабжения.
- Рис. 4. Схема вычислительного процесса идентификации эмпирических параметров модели