

Аноприенко А.Я. (Донецк, ДонНТУ)

Потапенко В.А. (Донецк, ДонНТУ)

Опыт создания распределенных моделирующих сред

Рассмотрены вопросы организации распределенных моделирующих сред, предназначенных для моделирования сложных технологических процессов. Главная идея заключается в применении принципа повторного использования существующих наработок в области моделирования сложных процессов совместно с технологией объектно-ориентированного программирования для реализации Web-ориентированных средств моделирования. Приведен пример организации распределенной моделирующей среды на базе системы DIVA.

The ways for development distributed simulation environments for simulation complicated technological processes are reviewed. The main idea is to use existents native simulation systems and combine object-oriented technology to design Web-based simulation environments. The example of such type systems is given.

Введение. Эволюция методов и средств компьютерного моделирования характеризуется двумя наиболее весомыми тенденциями: во-первых, интенсивным развитием графических интерфейсов с развитыми возможностями многооконной визуализации результатов и непосредственного визуального синтеза моделей; во-вторых, стремительной эволюцией средств параллельного и распределенного моделирования с максимальным использованием возможностей, предоставляемых глобальной сетью Интернет.

Существующие в настоящее время системы моделирования представляют собой достаточно мощные и универсальные среды, предназначенные для решения широкого круга задач. Тем не менее, применение таких систем не всегда возможно или оправдано. Среди имеющихся недостатков можно выделить такие, как высокая стоимость лицензий, значительная и порой неоправданная ресурсоемкость, отсутствие возможности эффективного использования в сетевой среде. В то же время современные информационные технологии в последнее время наиболее интенсивно развиваются именно в направлении расширения сетевых возможностей, особенно в области распределенных вычислений, базирующихся на объектно-ориентированном подходе с использованием таких технологий, как CORBA, Java RMI, COM/DCOM и др.

В последнее время одной из наиболее значимых проблем является разработка систем, способных взаимодействовать друг с другом с целью создания интегрированной моделирующей среды, объединяющей множество вычислительных серверов и конечных пользователей.

В данной статье рассматриваются подходы к организации подобного рода моделирующих сред на примере использования для этих целей системы моделирования сложных технологических процессов DIVA [1-3].

Распределенная моделирующая среда. Нарастающая сложность систем моделирования привела к необходимости в координации усилий в направлении стандартизации архитектур, протоколов и общих подходов в организации процесса моделирования, что привело к появлению распределенных моделирующих сред (РМС).

Распределенная моделирующая среда представляет собой достаточно сложную систему, состоящую из множества программных или программно-аппаратных компонент [4-5]. Декомпозиция типичной РМС позволяет выполнить анализ ее основных структурных элементов. На рисунке 1 изображены основные компоненты предлагаемой модели взаимодействия распределенной моделирующей среды.

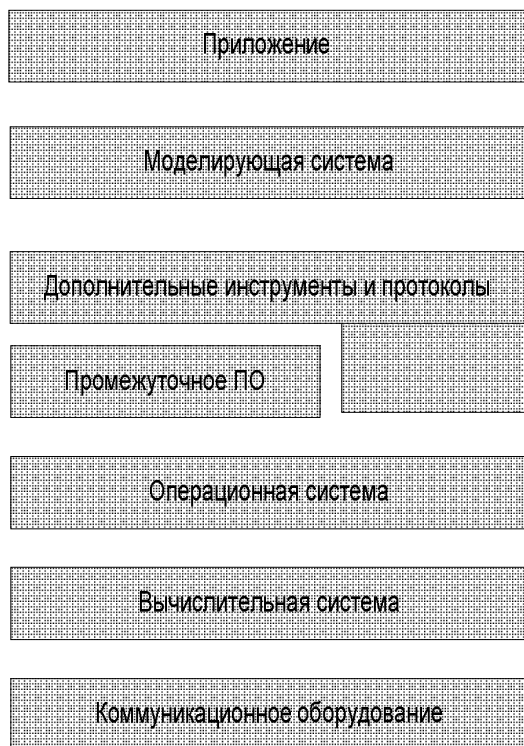


Рис. 1 Основные компоненты предлагаемой модели взаимодействия распределенной моделирующей среды

уровневыми объектно-ориентированными интерфейсами.

В зависимости от реализации конкретной моделирующей среды уровень «*дополнительные инструменты и протоколы*» может иметь различную функциональность. Тем не менее, основное предназначение это обеспечить компонентное взаимодействие между моделирующей системой и удаленными клиентами. Дополнительно могут быть использованы модули мониторинга, тестирования и оптимизации.

Непосредственно процесс моделирования осуществляется на следующем уровне. При этом в качестве системы моделирования могут быть использованы уже существующие решения, изначально не предназначенные для использования в распределенной моделирующей среде.

Верхний уровень – *уровень приложений* используется для организации интерфейса пользователя, установки различных параметров РМС, управления сессиями.

Одним из наиболее важных этапов определения будущей архитектуры распределенной моделирующей среды является выбор промежуточного ПО, а также дополнительных средств и протоколов. Среди современных технологий поддерживающих распределенную архитектуру можно выделить такие, как CORBA [6-8] и HLA (High Level Architecture, архитектура высокого уровня) [9-10].

Организация распределенной моделирующей среды на базе системы DIVA.

Система моделирования DIVA первая версия которой была разработана в 1987-1990 гг. в институте системной динамики и управления Штутгартского университета для обеспечения исследовательских работ в области разработки моделирующих систем сложных химических процессов и моделирования отдельных технологических аппаратов и установок [11-13].

Система DIVA позволяет проводить моделирование в различных областях химической промышленности. С ее помощью можно принимать решения при выборе варианта реализации как технологического процесса в целом, так и организации отдельного технологиче-

Нижний слой содержит коммуникационное оборудование, задача которого на программно-аппаратном уровне обеспечить сетевую поддержку взаимодействующих сторон. К данной группе относятся сетевые карты, маршрутизаторы, линии связи и т.д.

Уровень *вычислительных систем* (ВС) охватывает рабочие станции, на которых осуществляется непосредственный процесс компьютерного моделирования. Распределенная моделирующая среда может включать множество ВС, таким образом объединяя несколько гетерогенных вычислительных узлов.

Вычислительные системы, входящие в состав РМС могут иметь различную аппаратную основу (Intel, Macintosh, Sparc и т.д.) и соответственно отличные типы *операционных систем* (Windows, Linux, MacOS, SparcOS и т.д.).

Для облегчения *взаимодействия* между различными системными компонентами, а также поддержки распределенных вычислений вводится *промежуточное программное обеспечение*. Роль промежуточного ПО заключается в сокрытии низкоуровневых протоколов обмена данными, таких, как сокеты TCP/IP, более высоко-

ского аппарата. При моделировании можно учитывать энергозатраты процессов, материалоемкость и их влияние на окружающую среду.

DIVA реализована на языке программирования FORTRAN для операционной системы UNIX и традиционно поставляется в виде исходных текстов программных модулей с необходимостью последующей компиляции. Несмотря на интенсивное развитие инструментальных средств системы в последнее десятилетие, ее архитектура принципиально не изменилась, что создает определенные неудобства для пользователей.

В рамках договора о научном сотрудничестве между Донецким национальным техническим университетом и институтом им. Макса-Планка (Магдебург, Германия) в настоящее время ведутся совместные работы по созданию математической модели процесса производства уксусной кислоты - одного из наиболее сложных промышленных химических процессов такого рода, а также соответствующего программного обеспечения модульной распределенной моделирующей среды, позволяющей использовать все преимущества современной инфраструктуры Интернет.

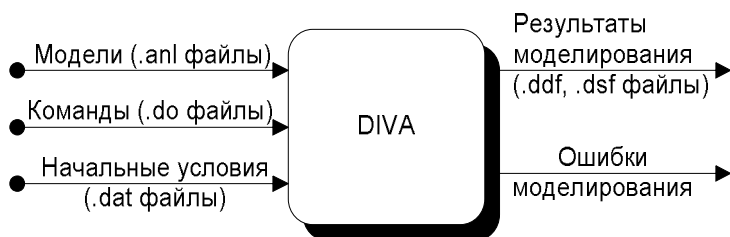


Рис. 2 Исходная модель системы DIVA

нет.

Изначально система DIVA не была ориентирована на работу в сетевой среде и не имеет в своем составе достаточно развитых интерфейсных средств. В общем виде работу с этой системой можно представить в виде упрощенной модели, изображенной на рисунке 2.

В данной модели можно выделить вектор входных моделей:

$$\bar{M} = (m_1, m_2, \dots, m_n), \quad (1)$$

где m_1, m_2, \dots, m_n - predetermined набор моделей описанных в файле <имя модели>.anl, n – количество моделей. Модель представляет собой набор систем алгебраических и обыкновенных дифференциальных уравнений, а также систем уравнений в частных производных с заданными начальными и граничными условиями.

Для каждой модели существует вектор начальных условий:

$$\bar{D} = (d_1, d_2, \dots, d_n), \quad (2)$$

где d_1, d_2, \dots, d_n - predetermined набор описаний начальных условий для соответствующих моделей, n – количество файлов. Файл обычно имеет название <имя модели>.dat.

Набор всех команд системы DIVA можно представить вектором команд:

$$\bar{C} = (c_1, c_2, \dots, c_n), \quad (3)$$

где c_1, c_2, \dots, c_n - все реализованные команды DIVA, n – количество всех команд. Команды можно разделить на управляющие, информационные и команды получения помощи. Управляющими командами задается тип модели, параметры моделирования и т.д. При помощи информационных команд можно получить значения текущих параметрах моделирования. Для получения справки по командам DIVA используются команды помощи.

Результаты моделирования сохраняются в файлах <имя модели>.ddf и <имя модели>.dsf.

В процессе работы системы DIVA могут возникать разнообразные ошибки. Их можно описать вектором ошибок:

$$\bar{E} = (e_1, e_2, \dots, e_n), \quad (4)$$

где e_1, e_2, \dots, e_n - стандартный набор ошибок системы DIVA, n – количество всех возможных ошибок. Ошибки могут быть исправимыми и фатальными. Примером исправимых ошибок может быть ситуации с ошибочным вводом команд, неправильным заданием параметров моделирования и т.д. При этом система DIVA продолжает функционировать. При возникновении фатальных ошибок выполнение программы прерывается.

Описанным интерфейсам соответствуют программные модули на языке FORTRAN,

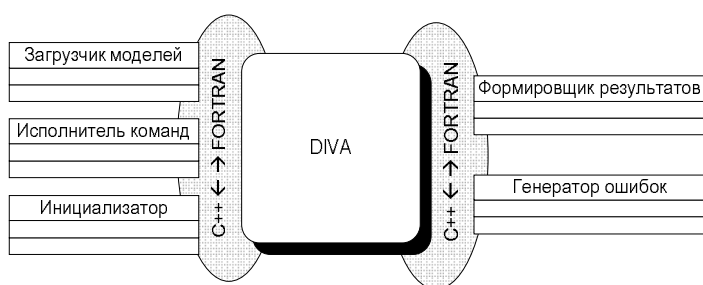


Рис. 3 Объектная модель системы DIVA

являющимся основным для системы DIVA. Язык FORTRAN не является объектно-ориентированным и в исходном виде не позволяет организовывать распределенное взаимодействие. Для решения данной проблемы система DIVA была дополнена разработанными программными интерфейсами, реализованными на современном объектно-ориентированном языке программирования C++. Связь между языками C++ и FORTRAN реализована при

помощи имеющихся встроенных средств обоих языков. Таким образом, была получена модель взаимодействия изображенная на рисунке 3.

Следующим этапом является формирование внешних объектных интерфейсов, предназначенных для предоставления доступа к системе удаленным станциям, используя коммуникационные возможности современных средств связи, в том числе сети Интернет. Для этого, при помощи языка описания удаленных интерфейсов IDL [7], были разработаны соответствующие программные компоненты (рис. 4.). IDL - язык однородной спецификации интерфейсов разнообразных информационных ресурсов, инкапсулируемых посредством CORBA и

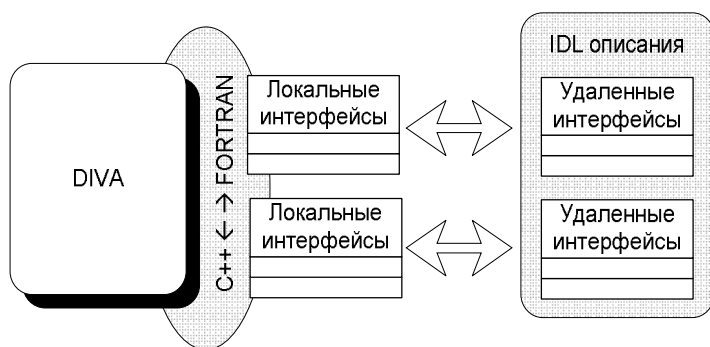


Рис. 4 Объектная модель системы DIVA с использованием технологии CORBA

является чисто описательным языком. Определения на IDL могут быть отображены стандартным образом в конкретные языки программирования, такие как C, C++, Java, Smalltalk. Репозиторий интерфейсов, содержащий определения интерфейсов на IDL, позволяет находить интерфейсы доступных сервисов в сети и программировать их использование в программах-клиентах с учетом возможностей брокера объектных запросов. Роли "клиент" и "сервер" следует рас-

сматривать как относительные: клиент (сервер) в архитектуре CORBA может быть сервером (клиентом) по отношению к другим клиентам (серверам).

Для доступа к моделирующей среде при помощи технологии Java Applets [17] был разработан графический интерфейс пользователя.

Данный интерфейс позволяет:

- наглядно визуализировать процессы, происходящие в наиболее важных узлах системы (реактор, очистительные колонны и т.д.);
- интерактивно управлять различными параметрами технологического процесса производства уксусной кислоты;
- получать результаты моделирования в виде многооконного режима с графиком для каждого исследуемого параметра;
- осуществлять выбор математической модели для дальнейших экспериментов.

Применение технологии Java Applets для организации графического интерфейса пользователя дает возможность работы с системой широкому кругу людей, в том числе в сети Интернет. В этом случае доступ к среде моделирования аналогичен доступу к традиционным ресурсам Интернет, таким, как страницы HTML, рисунки и т.д.

Выводы. В задачах повышения эффективности работы сложных промышленных комплексов, например, процесса производства уксусной кислоты, необходимо более широко использовать современные средства компьютерного моделирования, которые позволяют более эффективно контролировать такие свойства сложной системы, как ее динамика, устойчивость, целостность и др. При этом функциональная мощь используемых средств моделирования должна гармонично сочетаться с современными коммуникационными технологиями. В этом случае инфраструктура Интернет может, по сути, рассматриваться в качестве беспрецедентно мощной единой массивно-параллельной среды.

Использование описанной методики построения распределенных вычислительных систем, в частности распределенных моделирующих систем, позволит практически на порядок повысить масштабируемость, функциональность и быстродействие процесса моделирования. Использование Web как среды развертывания распределенной моделирующей системы позволило перейти на качественно новый уровень организации вычислительной среды, в том числе для целей обучения и тренинга. Web-ориентированные системы моделирования означают фактически новый качественный уровень в развитии современных информационных технологий.

ЛИТЕРАТУРА

1. Mohl K.D., Spieker A., Kohler R., Gilles E.D., Zeitz M. *DIVA - A simulation environment for chemical engineering applications. In Informatics, cybernetics and computer science (ICCS-97). Donetsk State Technical University. Donetsk, 1997, pp. 8-15.*
2. Kröner A., Holl P., Marquardt W., Gilles E. D. *DIVA - an open architecture for dynamic simulation. Computers and Chemical Engineering, 14, 1990, pp. 1289-1295.*
3. Anoprienko A., Kienle A., Svyatnyi S., Osipova T. *Modelling of acetic acid reactor for simulation on the base of DIVA environment. In Informatics, cybernetics and computer science (ICCS-97). Donetsk State Technical University. Donetsk, 1997, pp. 26-32.*
4. Pasquarelli A. *Technologies for Distributed Simulation: CORBA and HLA. Winter Simulation Conference, 1998, pp.13-20.*
5. Потапенко В.А. *Web-ориентированное моделирование как часть распределенной моделирующей среды. // Научные труды Донецкого государственного технического университета. Серия: Проблемы моделирования и автоматизации проектирования: - Донецк: ДонНТУ, 2002.*
6. Аноприенко А. Я., Забровский С.В., Потапенко В.А. *Использование технологии CORBA в распределённом моделировании сложных технологических систем. В кн. "Вычислительная техника и автоматизация». Сборник научных трудов ДонНТУ." Донецк, ДонНТУ, 2002.*
7. Калиниченко Л.А., Козаловский М.Р. *Стандарты OMG: Язык определения интерфейсов IDL в архитектуре CORBA // Системы Управления Базами данных, №2, 1996г., С.115-129.*

8. Shen C. *A CORBA Facility for Network Simulation*. // *Proc. of the 1996 Winter Simulation Conf.*, pp. 613-619, 1996.
9. Ritter K.C., Klein U., Strassburger S., Diessner M. "Web-based animation of distributed simulations based on HLA". // *In Proceedings of Simulation and Visualisation, Magdeburg, Germany, March 5-6, 1998*.
10. E. H. Page, "The Rise of Web-Based Simulation: Implications for the High Level Architecture", in *Proceedings of the 1998 Winter Simulation Conference*, pp. 1663-1668, Washington, D.C., 13-16 December 1998.
11. Waschler R., Kienle A., Sviatnyi S., Gilles E.D., Anoprienko A., Osipova T. *Modeling and Simulation of a Chemical Reactor for Production of Acetic Acid – III. Dynamic phase transitions*. In *Problems of simulation and computer aided design of dynamic systems*. Donetsk state technical university. Donetsk, 1999, pp. 102-109.
12. Kienle A., Pfisterer F., Waschler R., Sviatnyi S., Gilles E.D., Anoprienko A., Osipova T. *Modeling and simulation of a chemical reactor for the production of acetic acid — II. two-phase model*. In *Informatics, cybernetics and computer science (ICCS-99)*. Donetsk state technical university. Donetsk, 1999, pp. 15-23.
13. Sviatnyj V.A. *Problems of concurrent simulation of complex dynamic systems*. In *Informatics, Cybernetics and computer science (ICCS-99)*. Donetsk state technical university. Donetsk, 1999, pp. 6-14
14. Святный В.А. *Проблемы параллельного моделирования сложных динамических систем*. // *Наук. Труды ДонГТУ, серия «Информатика, кибернетика и вычислительная техника», вып. 6, 1999, с.6-14*.
15. Потапенко В.А. *Реализация клиент-серверного подхода в рамках реинжиниринга системы DIVA*. // *Научные труды Донецкого государственного технического университета. Серия: Информатика, кибернетика и вычислительная техника, (ИКВТ-2002) выпуск 39: -Донецк: ДонНТУ, 2002*.
16. Аноприенко А.Я., Забровский С.В., Потапенко В.А. *Современные тенденции развития тренажерных систем и их модельного обеспечения // Прогрессивные технологии и системы машиностроения: Международный сборник научных трудов. Вып. 10. – Донецк: ДонГТУ, 2000, с. 3-7*.
17. *Java Technology*. <http://www.java.sun.com>.

Как правильно ссылаться на этот доклад:

Аноприенко А.Я., Потапенко В.А. Опыт создания распределенных моделирующих сред // Труды международной научно-технической конференции «Современные средства автоматизации и компьютерно-интегрированные технологии». – Краматорск, Донбасская государственная машиностроительная академия, 11-13 февраля, 2003.