

WEB-ОРИЕНТИРОВАННАЯ СРЕДА ДЛЯ ИНТЕГРАЦИИ МОДЕЛИРУЮЩИХ, ВЫЧИСЛИТЕЛЬНЫХ И ИНФОРМАЦИОННЫХ СЕРВИСОВ

Аноприенко А. Я., Потапенко В.А.

Кафедра ЭВМ ДонНТУ

anoprien@cs.dgtu.donetsk.ua

Abstract

Anoprienko A., Potapenko V. WEB-based environment for integration of simulation, computing and information services. The concepts and structure of universal WEB-based distributed simulation environment system organization are proposed. The common architecture for integration of various web-services as well as separated modules and UML-diagrams of modules interaction are reviewed.

Введение

В развитии информационных технологий может быть выделена примерно десятилетняя периодичность принципиального смещения приоритетов, обусловленного не столько какими-либо разовыми разработками, сколько целой серией технологических решений (рис. 1). В настоящее время, в частности, возрастающей роли развития сетевых физических соединений сопутствует активное формирование на этой базе инфраструктуры логических сетевых соединений, предполагающих гибкую интеграцию различных приложений на базе широкого распространения сервисно-ориентированных архитектур.

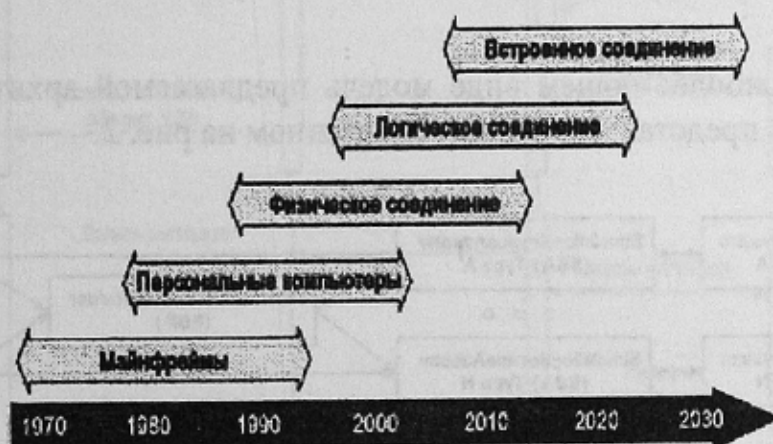


Рисунок 1 – Динамика смещения приоритетов в развитии информационных технологий с 1970 по 2030 гг. [1].

В соответствии с преобладающими на сегодня прогнозами сервисно-ориентированные архитектуры и веб-услуги начнут занимать доминирующие позиции примерно к 2007 г. [1]. При этом они позволят обеспечить недостижимую ранее вариабельность интерфейсов систем и гибкую совместимость с различными стандартами и компонентами. Уже сейчас имеются практически все технические возможности для реализации веб-ориентированных интегрированных сред, объединяющих в единые системы информационные, вычислительные и моделирующие сервисы, наиболее критичными из которых являются именно последние [2,3]. Обусловлено это целым рядом фактором, главными из которых являются существенно повышенные требования к производительности вычислений и интерфейсным средствам. В связи с этим наиболее целесообразной является первоочередная разработка средств и методов веб-интеграции моделирующих сервисов, как наиболее сложных в реализации, с последующим использованием полученных решений для интеграции вычислительных и информационных модулей.

Целью данной статьи является рассмотрение рекомендаций по наиболее рациональной системной организации Web-ориентированных распределенных моделирующих сред (далее ВОРМС), позволяющих в перспективе обеспечить гибкую системную интеграцию различных моделирующих сервисов с вычислительными и информационными сервисами. Предлагаемые решения базируются на многолетнем авторском опыте создания подобного рода систем [4-19] и являются одной из первых попыток обобщения полученных при этом знаний с целью создания концептуальной основы для разработки аналогичных средств и систем нового поколения, максимально использующих возможности и преимущества современной инфраструктуры Интернет.

Обзор архитектуры

В наиболее общем виде модель предлагаемой архитектуры ВОРМС может быть представлена в виде, показанном на рис. 2.

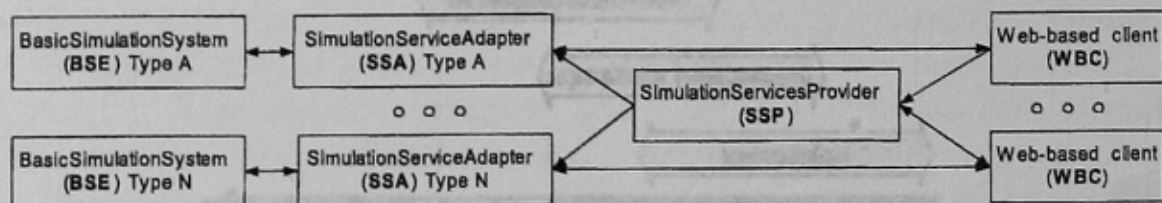


Рисунок 2. – Общая архитектура ВОРМС

Первичной точкой входа всех удаленных запросов со стороны WEB-клиентов (WBC) является провайдер моделирующих сервисов

(SimulationServicesProvider, SSP). Его задачей является предоставление информации о доступных на данный момент базовых сервисах моделирования. WEB-клиент имеет возможность выбрать необходимый сервис из списка предложенных и инициировать начало сеанса работы. Для связи с базовой системой моделирования (BasicSimulationSystem, BSS) предназначен адаптер сервиса моделирования. Поскольку различные BSS имеют свои специфические механизмы доступа и управления, то задачей адаптера SSA является предоставление универсального, прозрачного для клиента механизма доступа к таким BSS. Для каждой BSS реализуется свой собственный адаптер SSA, учитывающий все детали взаимодействия с базовой системой моделирования.

UML-диаграмма взаимодействия перечисленных модулей представлена на рис. 3. Начало сеанса взаимодействия клиентского модуля

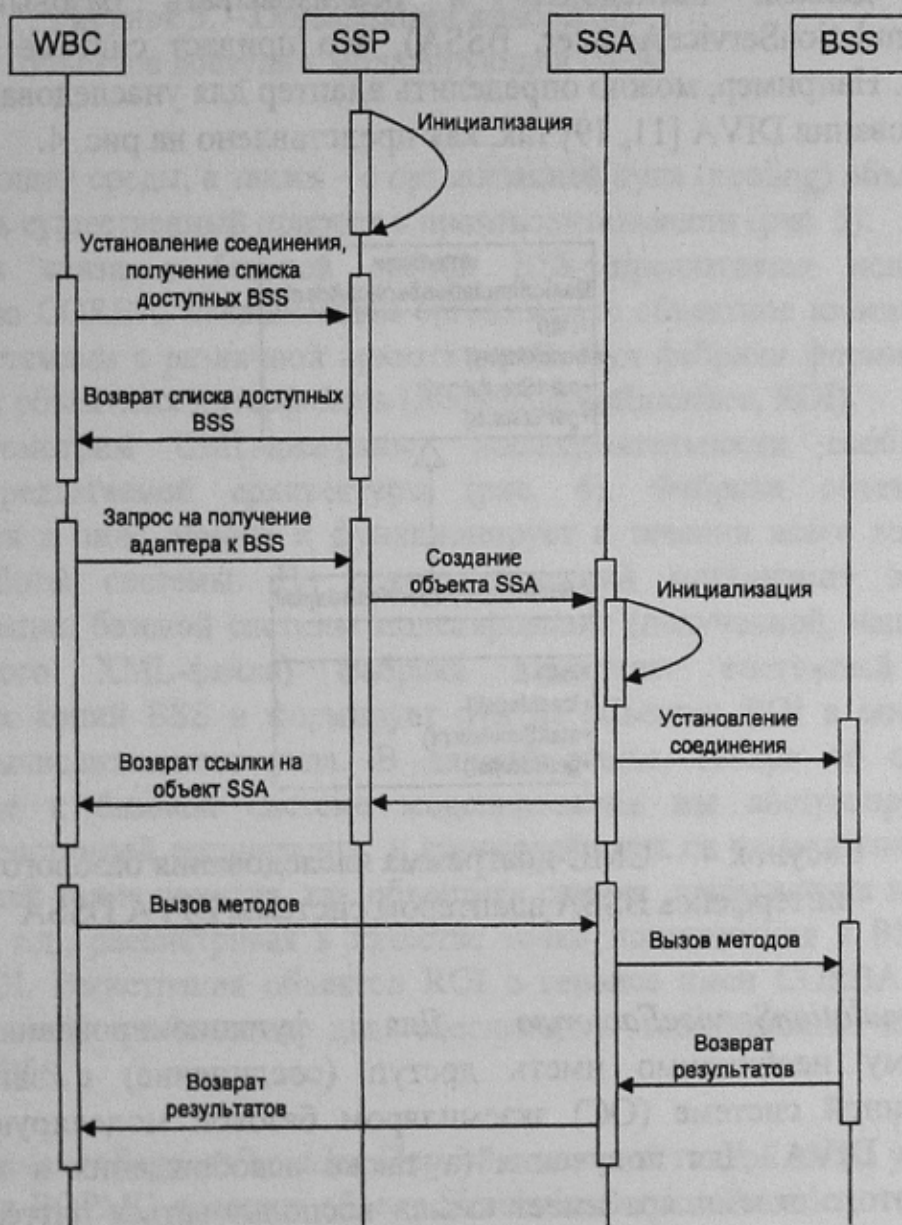


Рисунок 3. – UML-диаграмма взаимодействия базовых модулей

WBC с моделирующей средой начинается с подключения к провайдеру SSP. На основе информации, полученной, например, из описаний в специальном XML файле SSP предлагает WBC доступные сервисы моделирования (Simulation Services, SS). Пользователь системы, согласно его текущим задачам, выбирает тот или иной SS для последующей работы. Провайдер SSP создает экземпляр адаптера SSA и возвращает ссылку на него клиенту WBC. Вся дальнейшая работа клиента происходит именно с ним.

Организация серверной части ВОРМС

Модуль SimulationServiceAdapter. Как было сказано ранее, SimulationServiceAdapter реализует все основные операции взаимодействия клиента с BSS, например, такие, как загрузка модели, установление времени моделирования, получение результатов и т.д. Предполагается, что каждый адаптер должен наследовать и реализовывать базовый интерфейс (BasicSimulationServiceAdapter, BSSA), что придаст системе необходимую гибкость. Например, можно определить адаптер для унаследованной системы моделирования DIVA [11, 19] так, как представлено на рис. 4.

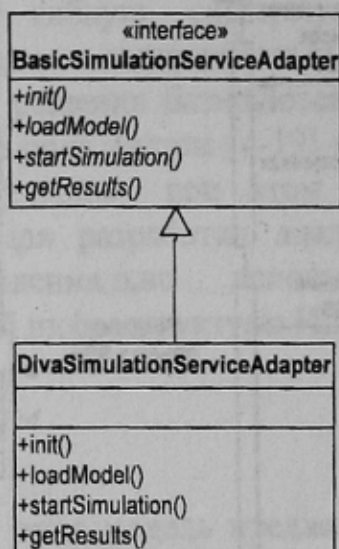


Рисунок 4. – UML-диаграмма наследования базового интерфейса BSSA адаптером системы DIVA DSSA

SimulationServiceFactory Для функционирования адаптера последнему необходимо иметь доступ (соединение) с запущенным в операционной системе (ОС) экземпляром базовой моделирующей среды, например DIVA. Для получения (а также освобождения в конце сеанса работы) этого экземпляра имеет смысл воспользоваться паттерном Factory (фабрика). Основная идея заключается в том, что Factory берет на себя все операции связанные с инициализацией и запуском в ОС экземпляра

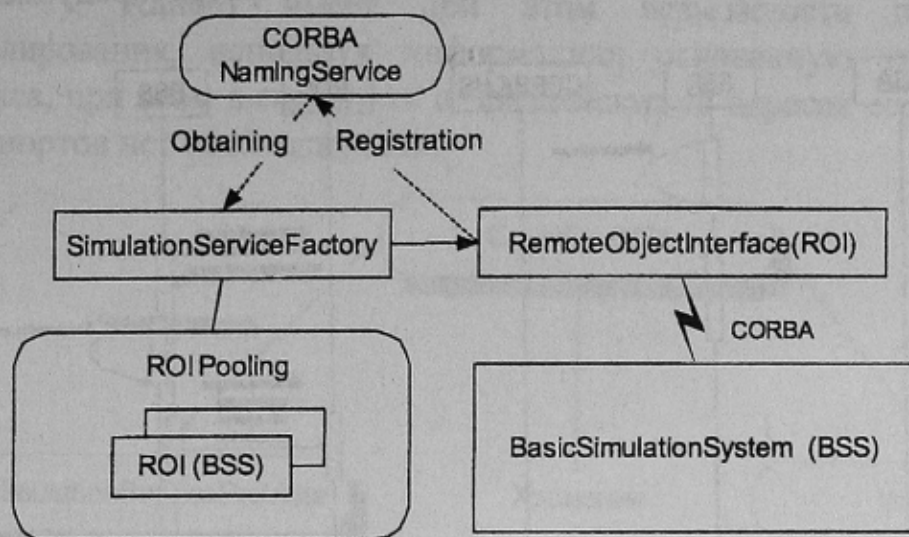


Рисунок 5. – Организации «фабрики» объектов доступа к моделирующей среде

моделирующей среды, а также – с организацией пула (pooling) объектов, что может дать существенный прирост в производительности (рис. 5).

Для связи с базовой средой BSS предлагается использовать технологию CORBA, позволяющей организовать объектное взаимодействие между системами с различной архитектурой. Пул фабрики формируется из удаленных объектных интерфейсов (RemoteObjectInterface, ROI).

Рассмотрим UML-диаграмму последовательности сообщений в рамках предлагаемой архитектуры (рис. 6). Фабрика объектов SSF реализуется в виде демона и функционирует в течении всего жизненного цикла работы системы. На основе описаний механизмов запуска и инициализации базовой системы моделирования (полученной, например, из специального XML-файла) фабрика выполняет системный запуск нескольких копий BSS и формирует пул из объектов ROI в оперативной памяти вычислительного узла. В данной статье, говоря об объектном интерфейсе к базовой системе моделирования мы абстрагируемся от вопросов системной организации и взаимодействия по технологии CORBA, включающей такие понятия, как объектная ссылка, стаб клиента и скелетон сервера и т.д., рассматривая в качестве точки подключения к BSS только объект ROI. Регистрация объектов ROI в сервисе имен CORBA (CORBA NamingService) необходима для обеспечения возможности их поиска фабрикой FFS.

SimulationServicesProvider. Первой и главной точкой входа удаленных клиентов к ВОРМС является объект SimulationServicesProvider, что требует от последнего теоретически бесконечного жизненного цикла для обслуживания входящих запросов клиентов. Традиционно, реализация

серверных демонов в приложениях типа «клиент-сервер» предусматривала

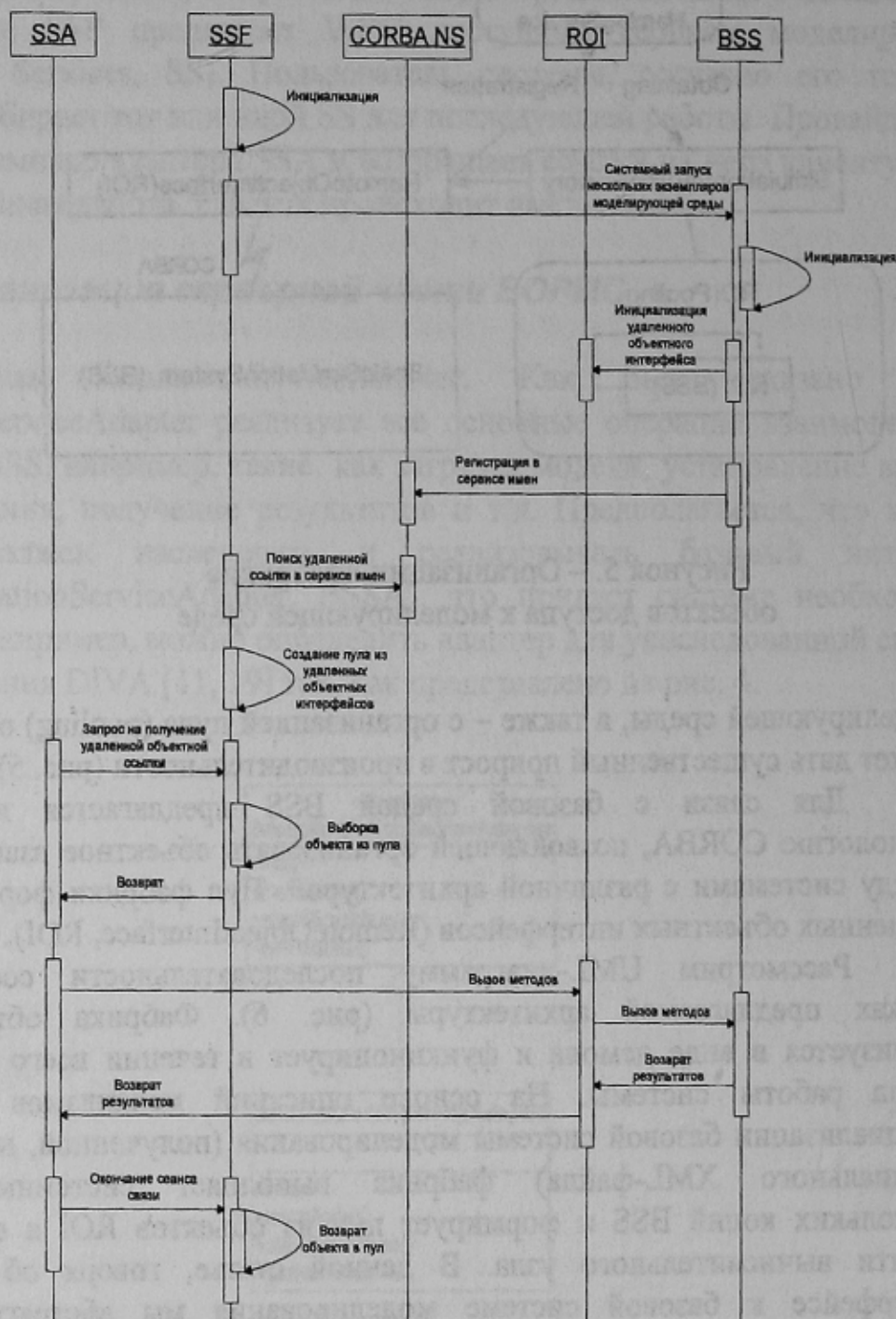


Рисунок 6. – UML-диаграмма последовательностей сообщений при

непосредственное программирование сокетов TCP/IP и сводилась к так называемому механизму "прослушивания" TCP порта в ожидании запросов на подключение со стороны клиентов. Данный механизм по ряду причин в настоящее время не является оптимальным.

Технология J2EE, а именно спецификация JNDI, предоставляет более гибкую модель регистрации и поиска объектов в распределенных системах

(рис. 7). Клиент имеет при этом возможность поиска сервисов моделирования, используя информацию, основанную только на имени сервиса, при этом в сведениях о физических IP адресах серверов и номеров TCP портов нет необходимости.

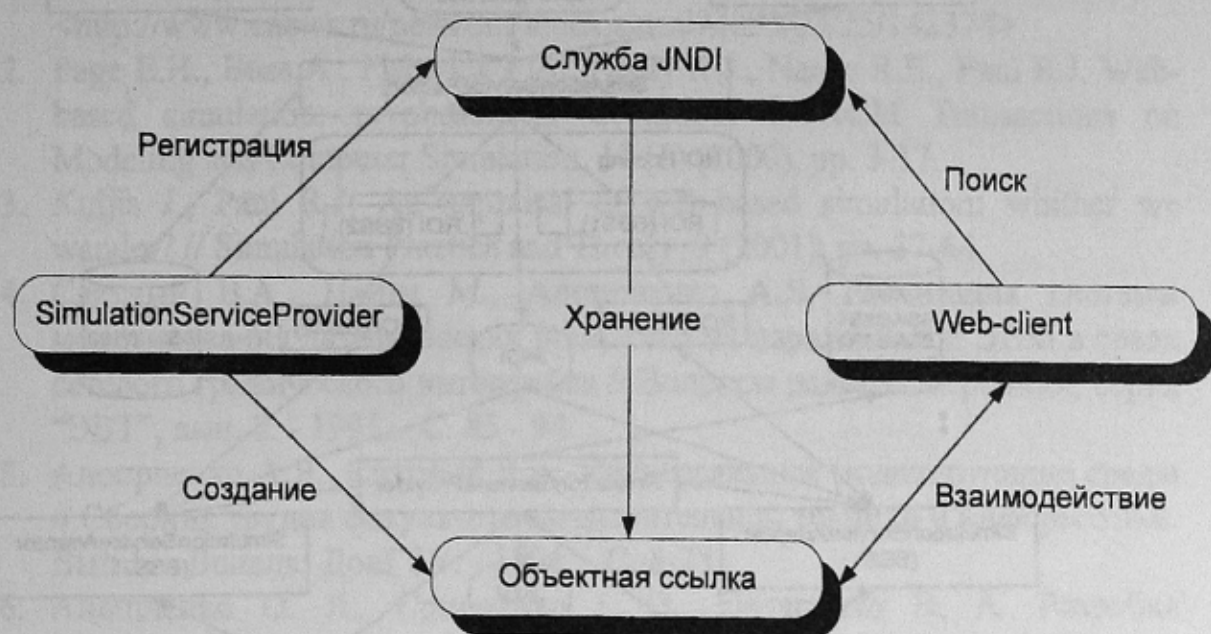


Рисунок 7. – Механизм регистрации и поиска в службе JNDI

Во время первичной инициализации модуль SSP регистрирует себя в службе JNDI под заранее известным клиенту BOPMC именем, что позволяет последнему получить объектную ссылку на SSP для дальнейшего взаимодействия.

Аналогичным образом в реестре JNDI регистрирует себя и объект SSF. Вновь создаваемые объекты SSA используют JNDI для поиска фабрики объектов доступа к базовым моделирующим средам. Таким образом, имеем достаточно гибкое решение, поскольку фабрика SSF может предоставлять ссылку к моделирующим системам, расположенным на разных вычислительных узлах под управлением различных операционных систем.

BasicSimulationSystem. Конечной целью реализации BOPMC является предоставление доступа внешним клиентам к поддерживаемым базовым системам моделирования. Расширение спектра таких систем будет являться важнейшей и наиболее сложной задачей. В настоящее время уже получен положительный опыт организации удаленного взаимодействия с системой DIVA, в ближайшей перспективе планируется организовать аналогичное взаимодействие с системой Matlab/Simulink и другими.

На рисунке 8 представлена полная модель предлагаемой BOPMC, включающая все рассмотренные выше модули.

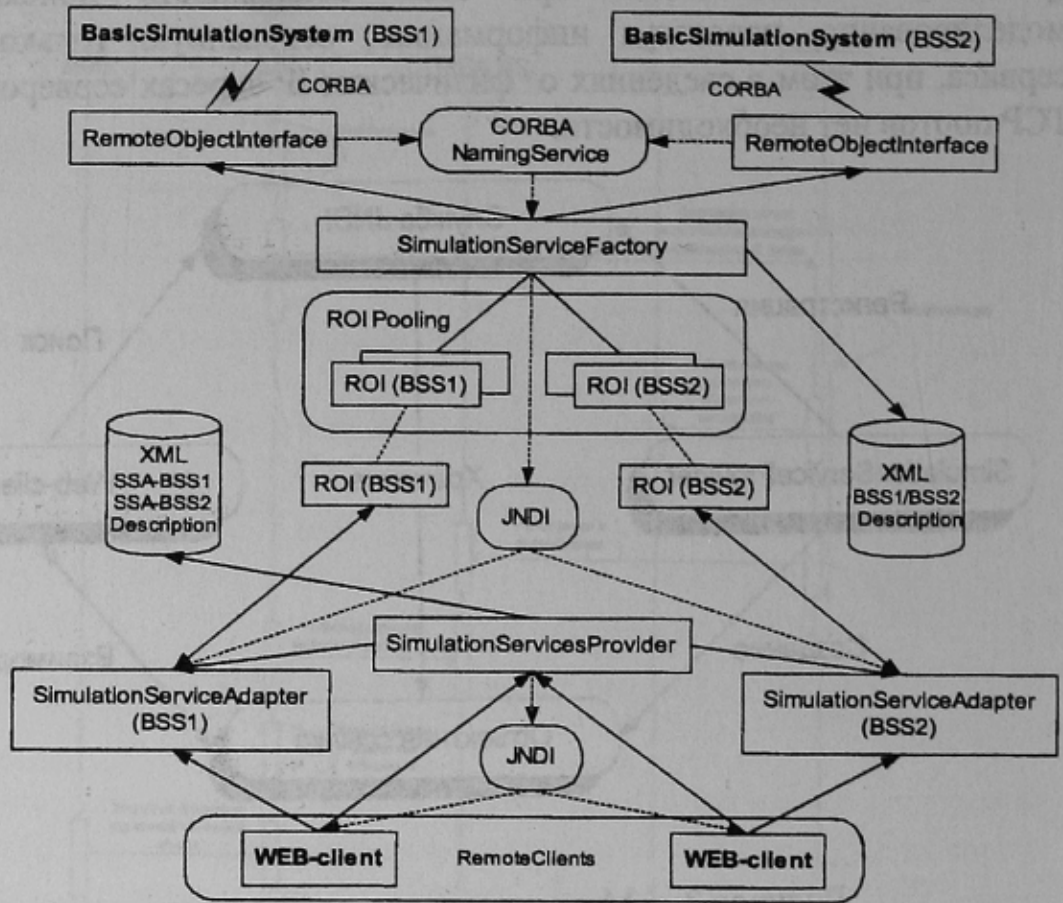


Рисунок 8. – Модель взаимодействия основных модулей WOPMC

Выводы

Предлагаемая структура WOPMC может служить основой для программной реализации реально действующей универсальной WEB-ориентированной распределенной моделирующей среды. Унифицированный либо настраиваемый под конкретную среду графический интерфейс с возможностью доступа из практически любого современного системного окружения, обеспечит прозрачное для пользователя взаимодействие с выбранной системой моделирования. На данном этапе предусматривается, что WOPMC будет предоставлять достаточные широкие, но несколько ограниченные возможности для работы с выбранными сервисами и моделями, т.е. установку различных параметров и режимов моделирования, выбор используемых численных методов и оптимизирующих средств, наглядную визуализацию полученных результатов и т.д. В дальнейшем планируется обеспечить также поддержку всех этапов разработки, тестирования и отладки моделей реальной сложности, а также – интеграцию различных видов сервисов в рамках единой распределенной среды.

Литература

1. Gartner: определены ключевые технологии на 2003-2012 гг. // CNews.ru – Интернет-издание о высоких технологиях, 25.03.2003 <<http://www.cnews.ru/newcom/index.shtml?2003/03/25/142374>>
2. Page E.H., Buss A., Fishwick P.A., Healy K.J., Nance R.E., Paul R.J. Web-based simulation: revolution or evolution? // ACM Transactions on Modeling and Computer Simulation, 10 (1) (2000), pp. 3-17.
3. Kuljis J., Paul R.J. An appraisal of web-based simulation: whither we wander? // Simulation Practice and Theory, 9 (2001), pp. 37-64.
4. Святный В.А., Цайтц М., Аноприенко А.Я. Реализация системы моделирования динамических процессов на параллельной ЭВМ в среде сетевого графического интерфейса // Вопросы радиоэлектроники, серия "ЭВТ", вып. 2. - 1991. - С. 85 - 94.
5. Аноприенко А.Я., Святный В.А. Универсальные моделирующие среды // Сборник трудов факультета вычислительной техники и информатики. Вып.1. - Донецк: ДонГТУ. - 1996. - С. 8-23.
6. Аноприенко О. Я., Ермоленко І. О., Потапенко В. А. Розробка компактних програмних засобів блочно-орієнтованого розподіленого моделювання динамічних систем // Наукові праці Донецького державного технічного університету. Серія «Проблеми моделювання та автоматизації проектування динамічних систем». Випуск 10. – Донецьк: ДонДТУ, 1999. С. 119-128.
7. Аноприенко А.Я., Забровский С.В., Каневский А.Д. Опыт реинжиниринга системы моделирования сложных технологических процессов / Наукові праці Донецького державного технічного університету. Серія "Обчислювальна техніка та автоматизація", випуск 20. - Донецьк, ДонДТУ, 2000, с. 139-148.
8. Аноприенко А.Я., Святный В.А. Высокопроизводительные информационно-моделирующие среды для исследования, разработки и сопровождения сложных динамических систем // Научные труды Донецкого государственного технического университета. Випуск 29. Серия "Проблемы моделирования и автоматизации проектирования динамических систем" - Севастополь: «Вебер». - 2001. - С. 346-367.
9. Забровский С.В., Мацак С.А., Потапенко В.А., Сисюкин В.С. Опыт использования сети Интернет в качестве распределенной среды моделирования // Проблемы моделювання та автоматизації проектування динамічних систем: 29. наук. пр. – Д.: ДонДТУ, 2001. – С. 254-259.
10. Потапенко В.А. Web-ориентированное моделирование как часть распределенной моделирующей среды // Проблемы моделирования и автоматизации проектирования: 52. науч. тр. –Д.: ДонНТУ, 2002. – С. 27-34.

11. Потапенко В.А. Реализация клиент-серверного подхода в рамках реинжиниринга системы DIVA // Информатика, кибернетика и вычислительная техника: Вып. 39. – Д.: ДонНТУ, 2002. – С. 222-226.
12. Анопrienко А.Я., Забровский С.В., Потапенко В.А. Современные тенденции развития тренажерных систем и их модельного обеспечения // Прогрессивные технологии и системы машиностроения: 10. науч. тр. – Д.: ДонГТУ, 2000. – С. 3-7.
13. Анопrienко А.Я., Святный В.А. Компьютерное моделирование: новые возможности и парадигмы // Донбас-2020: наука і техніка – виробництву: Матеріали науково-практичної конференції. м. Донецьк, 05-06 лютого 2002 р. - Донецьк, ДонНТУ Міністерства освіти і науки, 2002. С. 649-652.
14. Анопrienко А. Я., Забровский С.В., Потапенко В.А. Использование технологии CORBA в распределённом моделировании сложных технологических систем // Обчислювальна техніка та автоматизація: 38. наук. пр. – Д.: ДонДТУ, 2002. – С. 186-190.
15. Gilles E.D., Kienle A., Waschler R., Sviatnyi V., Anoprienko A., Potapenko V. Zur Entwicklung des Trainingssimulators einer großchemischen Anlage // Problems of Simulation and Computer Aided Design of Dynamic Systems (SCAD-2002). Scientific Papers of Donetsk National Technical University. Volume 52. Donetsk, 2002, pages 23-26.
16. Waschler R., Kienle A., Anoprienko A., and Osipova T. Dynamic plantwide modelling, flowsheet simulation and nonlinear analysis of an industrial production plant // In J. Grievink and J. van Schijndel, editors, European Symposium on Computer Aided Process Engineering - 12 - ESCAPE-12, 26–29 May, 2002, The Hague, The Netherlands, Amsterdam: Elsevier, 2002, pages 583–588.
17. Святный В.А., Анопrienко А.Я., Потапенко В.А. Модульные среды для сетевого распределенного моделирования сложных динамических систем // Труды международной конференции «Современные проблемы информатизации в технике и технологиях»: Выпуск 8. – Воронеж, 2003. – С. 122-123.
18. Потапенко В.А. Подходы к организации распределенных моделирующих сред на базе промежуточного программного обеспечения технологии CORBA // Труды Межд. научн. практ. семинара «Практика и перспективы развития институционального партнерства». – Донецк, 2003. – С 165-171.
19. Anoprienko A., Potapenko V. Web-basierte Simulationsumgebung mit DIVA-Serverkomponente für komplexe verfahrenstechnische Produktionsanlagen // 17. Symposium "Simulationstechnik" ASIM 2003, Magdeburg, 16.09 bis 19.09.2003. – SCS-Europe, 2003. S. 205-208.