

Д. А. НАСОНОВ, С. В. ИВАНОВ, А. В. БУХАНОВСКИЙ

## РАСПРЕДЕЛЕННАЯ ПРОГРАММНАЯ ПЛАТФОРМА ДЛЯ ПРОТИПИРОВАНИЯ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ УПРАВЛЕНИЯ СЛОЖНЫМИ ДИНАМИЧЕСКИМИ ОБЪЕКТАМИ

Рассмотрены принципы построения инструментальной среды для прототипирования и исследования бортовых систем поддержки оператора сложных динамических объектов в рамках концепции облачных вычислений.

*Ключевые слова:* бортовая система, поддержка оператора, распределенная среда, Грид, облачные вычисления, прикладные сервисы, прототипирование.

Усложнение принципов управления сложными динамическими объектами требует разработки и внедрения бортовых систем поддержки оператора (СПО) в экстремальных условиях эксплуатации. Однако технология разработки таких систем в настоящее время не унифицирована в должной мере. В частности, отсутствуют эффективные средства прототипирования, которые могли бы обеспечить всестороннее экспериментальное исследование характеристик разрабатываемых систем на макетах в составе виртуального испытательного стенда.

В рамках настоящей работы рассматриваются принципы построения распределенной программной платформы (РПП) для прототипирования интеллектуальных систем управления сложными динамическими объектами, функционирующей в рамках концепции облачных вычислений.

РПП представляет собой открытую iPSE-систему [1], функционирующую в распределенной среде. Она включает в себя выделенные вычислительные системы, в том числе обеспечивающие интерфейс с устройствами сбора данных в режиме реального времени, вычислители с графическими процессорами (GPU) и суперкомпьютерные системы, необходимые для решения ресурсоемких задач численного моделирования. Кроме того, в состав среды могут входить уже существующие ресурсы Грид-систем первого поколения. При этом доступ ко всем ресурсам обеспечивается в рамках модели облачных вычислений [2], реализуемой на основе многофункциональной инструментально-технологической платформы CLAVIRE (CLOUD Applications VIRTUAL Environment).

Встраивание компонентов в РПП выполняется на высшем уровне изоляции — в форме прикладных пакетов, взаимодействующих друг с другом через файлы входных и выходных данных. Это обеспечивает возможность выбора и использования уже готовых прикладных пакетов без модификации исходных кодов. Содержательной основой РПП является репози-

торий проблемно-ориентированных сервисов, отвечающих за процесс поддержки принятия решений. В состав репозитория входят прикладные сервисы:

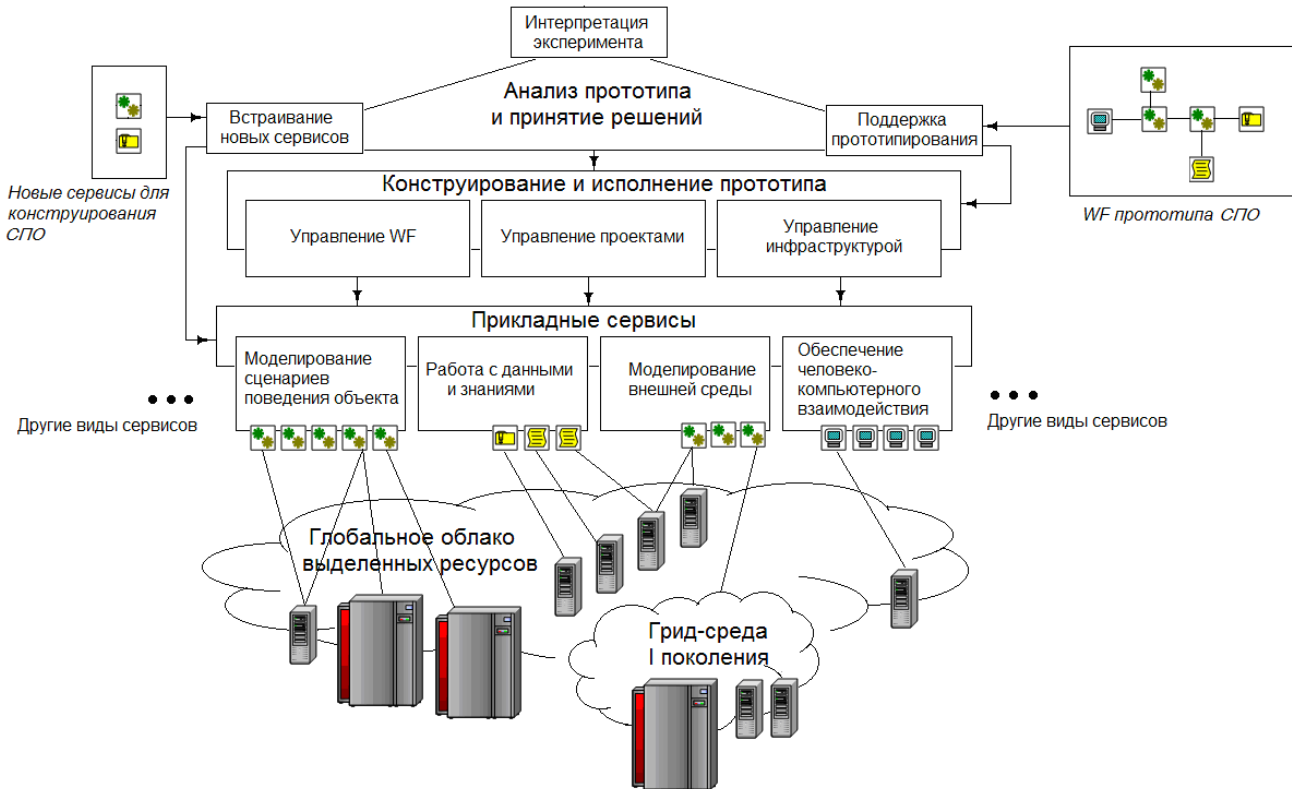
- сбора данных (обращение к датчикам в режиме реального времени через IP или к программам-симуляторам, а также технологические сервисы для эмуляции режимов реального времени);
- обработки и анализа данных (фильтрация и контроль ошибок, статистический анализ, data mining и пр.);
- организации баз знаний (поддержка описания знаний в разных структурах, средства формализации знаний, средства проверки качества баз знаний);
- машин логического вывода (во многом привязаны к структуре баз знаний — вывод по правилам, учет неопределенности, анализ альтернатив, прецедентные технологии и пр.);
- технологий машинного обучения (нейросетевые модели, эволюционное программирование);
- человеко-компьютерного взаимодействия (формирование прикладных интерфейсов системы, визуализации и когнитивной графики);
- компьютерного моделирования объекта и среды (по видам динамических объектов).

В целом проблемно-ориентированные сервисы рассматриваются в связке с вычислительными системами соответствующей архитектуры, управляемыми посредством РПП. Это позволяет пользователю в ходе прототипирования экспериментировать не только с функциональными элементами программного обеспечения СПО, но и с аппаратной составляющей. Пользователь (разработчик конкретной бортовой СПО) взаимодействует с РПП посредством сети Интернет через web-интерфейс. Он имеет возможность создавать распределенные образы-прототипы собственных СПО, описывая их в форме потока заданий (workflow, WF) на языке EasyFlow [3]. Это позволяет в ходе прототипирования исследовать отдельные этапы разработки СПО, постепенно наращивая функционал, заменяя компоненты и пр., в процессе отработки окончательного решения. В том случае, когда доступные в РПП сервисы не могут в полной мере удовлетворить пользователя, он имеет возможность встраивать в WF собственные компоненты; при этом процесс исполнения WF будет замыкаться только на пользовательскую вычислительную систему. Все результаты работы пользователя (создаваемые WF, структуры данных и знаний и пр.) размещаются в соответствующем хранилище, что позволяет географически распределенным группам пользователей работать над проектом поэтапно.

В целом РПП является средством эмуляции работы бортовых СПО с ограничением на реализацию технологий реального времени вследствие существенной инерционности распределенной среды и неопределенности временных характеристик исполнения WF. Дополнительным ограничением является привязка сервисов к конкретным вычислительным ресурсам (например, суперкомпьютерам), которые *a priori* не могут быть использованы в бортовых системах. Вместе с тем РПП позволяет получать информацию о временных характеристиках по отдельным элементам WF и операциям их взаимодействия, а также предоставляет данные о сложности реализуемых алгоритмов, что дает возможность оценить время исполнения WF при изменении архитектуры (без изменения алгоритма). Для моделирования процессов реального времени в состав WF необходимо включать дополнительные компоненты „моделирования времени“, обеспечивающие, в частности, синхронизацию и ранжирование передачи данных между элементами WF.

На рисунке приведена принципиальная архитектура РПП. На схеме отмечены четыре основных уровня. Верхний уровень (анализ прототипа и принятие решений) связан с процессами интеллектуальной поддержки и взаимодействия с пользователем. Следующий уровень относится к автоматическим средствам конструирования и исполнения прототипа. Третий уровень составляют собственно прикладные сервисы (их декларативные описания в репозитории). На четвертом уровне находится облако вычислительных ресурсов, включающих как

выделенные системы, так и существующие распределенные Грид-среды первого поколения. На вычислительных ресурсах установлены пакеты, входящие в состав проблемно-ориентированных сервисов.



Таким образом, посредством РПП разработчик получает возможность в процессе прототипирования сосредоточиться на отработке уникальных частей СПО; для обеспечения функционирования макета в целом используются подходящие сервисы, предоставляемые РПП. Это дает возможность эксперту осуществлять выбор способа реализации различных компонентов СПО путем сравнения работы различных реализаций сервисов в рамках решения отдельных тестовых задач. В целом это позволяет создавать виртуальное окружение для тестирования и обучения уже разработанных бортовых систем; в данном случае сама СПО функционирует на вычислительной системе пользователя, а РПП является лишь эмулятором источников входных данных, характеризующих внешнюю среду. Дополнительно данный подход позволяет минимизировать затраты пользователей на содержание собственной аппаратной инфраструктуры, особенно в области систем ограниченной доступности (например, бортовых вычислителей или суперкомпьютеров нестандартной архитектуры).

Работа выполнена в рамках проектов по реализации Постановлений № 218 и 220 при поддержке ФЦП „Научные и научно-педагогические кадры инновационной России на 2009—2013 гг.“.

#### СПИСОК ЛИТЕРАТУРЫ

1. Бухановский А. В., Ковальчук С. В., Марьин С. В. Интеллектуальные высокопроизводительные программные комплексы моделирования сложных систем: концепция, архитектура и примеры реализации // Изв. вузов. Приборостроение. 2009. Т. 52, № 10. С. 5—24.
2. Foster I., Zhao Y., Raicu I., Lu S. Cloud Computing and Grid Computing 360-Degree Compared // eprint arXiv:0901.0131. 2008 [Электронный ресурс]: <<http://arxiv.org/ftp/arxiv/papers/0901/0901.0131.pdf>>.
3. Князьков К. В., Ларченко А. В. Предметно-ориентированные технологии разработки приложений в распределенных средах // Изв. вузов. Приборостроение. 2011. Т. 54, № 10. С. 36—43.

- Денис Александрович Насонов** — **Сведения об авторах**  
— НИИ Научно-технических компьютерных технологий Санкт-Петербургского государственного университета информационных технологий, механики и оптики; младший научный сотрудник;  
E-mail: denis.nasonov@gmail.com
- Сергей Владимирович Иванов** — канд. техн. наук; НИИ Научно-технических компьютерных технологий Санкт-Петербургского государственного университета информационных технологий, механики и оптики; старший научный сотрудник; E-mail: Sergey.v.ivanov@rambler.ru
- Александр Валерьевич Бухановский** — д-р техн. наук, профессор; НИИ Научно-технических компьютерных технологий Санкт-Петербургского государственного университета информационных технологий, механики и оптики; директор;  
E-mail: avb\_mail@mail.ru

Рекомендована НИИ НКТ

Поступила в редакцию  
15.05.11 г.