

ЭКСПЕРИМЕНТАЛЬНАЯ РЕАЛИЗАЦИЯ СИСТЕМЫ ОБЛАЧНЫХ ВЫЧИСЛЕНИЙ «SKYTS» ДЛЯ СЧЕТА T-ПРИЛОЖЕНИЙ В НЕОДНОРОДНОЙ РАСПРЕДЕЛЕННОЙ СРЕДЕ

Г.И. Есин, А.А. Кузнецов, В.А. Роганов

Предпосылками для развития системы «SkyTS» (прежнее название «SkylighTS») явились проведенные в рамках программы «СКИФ» эксперименты по использованию совместно с системой OpenTS таких реализаций MPI, как MPICH-G2, IMPI, PACX. Эти исследования показали, что технология OpenTS может быть доведена до GRID-решения. Под GRID-решением мы будем подразумевать набор средств и технических решений, которые позволяют объединять разрозненные вычислительные средства в территориально-распределенную гетерогенную информационную и вычислительную систему [3].

Все сделанные в последние годы доработки системы OpenTS (улучшение свойств кроссплатформенности, разработка подсистем отказоустойчивости и масштабируемости) позволили ей перейти на новый уровень развития, который состоит в поддержке выполнения T-приложений в неоднородной распределенной вычислительной среде. Результатом этого развития стала проведенная в рамках программ «Триада» и «СКИФ-GRID» разработка архитектуры и экспериментальная реализация системы облачных вычислений, позволяющая объединять разнородные компьютеры для счета T-приложений.

Общая концепция облачных вычислений подразумевает, что пользователь использует свой персональный компьютер в качестве терминального клиента, соответственно, приложения, которые он использует, являются сетевыми сервисами, выполняющимися на удаленной вычислительной установке организации, предоставляющей данный сервис. Примером реализации этой концепции являются системы, созданные компаниями «Google», «Microsoft», «Amazon», «Aptana», и др. Примерно такой же принцип реализует и SkyTS, но ключевым отличием от других подобных систем («Google Apps», «Microsoft Azure», «Xceregion iCloud»,...) является ориентированность на использование T-системы, динамическое объединение вычислительных установок в вычислительные облака, представление параллельных приложений как web-сервисов.

Далее, под вычислительным облаком будем подразумевать группу компьютеров (вычислительных модулей), объединенных по степени близости и/или ресурсному требованию счетного задания; под сервисом системы понимается T-приложение, загруженное пользователем.

Облака бывают двух видов:

- кластер (суперкомпьютер): на использование данного вида облаков накладываются дополнительные ограничения, задаваемые администратором данного ресурса, а также, в целях обеспечения безопасности, сервисы, запускаемые на нем, должны пройти проверку администратором SkyTS или администратором ресурса;
 - группа персональных компьютеров, предоставляемых пользователями-энтузиастами.
- Система обладает следующими свойствами:
- масштабируемость;
 - отказоустойчивость: система задействует механизм отказоустойчивости T-приложений;
 - кроссплатформенность: все модули системы написаны на интерпретируемых языках, что делает их переносимыми;
 - поддержка виртуализации: управляющие и счетные модули могут работать в среде виртуальной машины.

SkyTS состоит из следующих компонентов (см. рис. 1):

- центральный сервер;
- база данных;
- репозитории приложений;
- управляющие модули (менеджеры);
- вычислительные модули (рабочие);
- серверы мониторинга состояния системы.

На центральном сервере находятся: web-интерфейс пользователя (панель управления SkyTS) и подсистемы мониторинга целостности системы, доставки результатов счета, загрузки новых приложений и другие.

Задачами управляющих модулей является выбор задачи из пула заданий, мониторинг состояния рабочих, накопление и передача статистики отказов вычислительных модулей, перезапуск приложения в случае возникновения отказа, доставка приложения и данных для счета, обратная передача результатов. Менеджеры действуют как связующее звено между T-приложениями и ресурсами, необходимыми для счета

пользовательских задач. Они способны выполнять T-приложения в отказоустойчивом режиме и взаимодействовать со счетными модулями посредством обмена сообщениями в определенном формате в

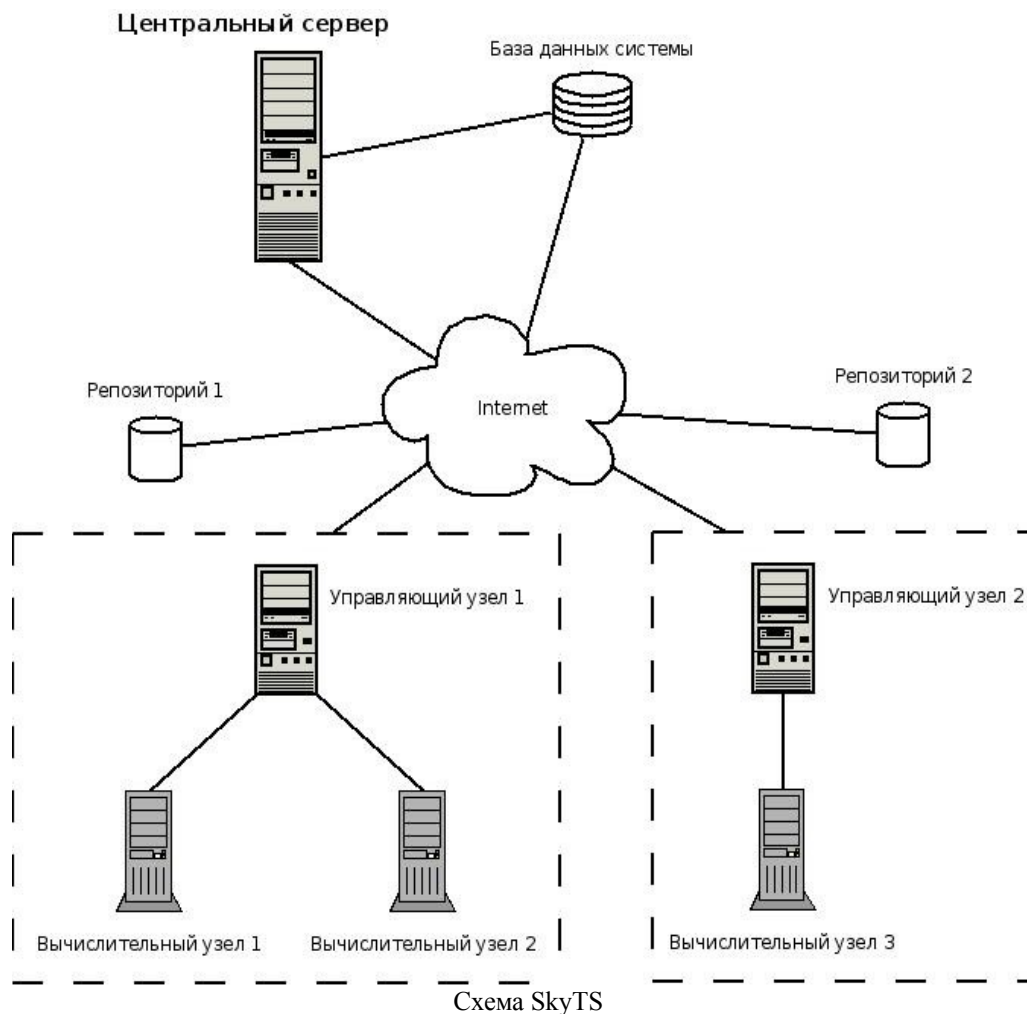


Схема SkyTS

соответствии с сетевым протоколом.

В базе данных хранятся сведения о сущностях системы, пул заданий и общая статистика отказов управляющих и счетных модулей.

Репозитории предназначены для хранения установленных приложений. В качестве средства хранения используется система «subversion» («svn»). Subversion был выбран в качестве сервера репозитория, так как он предоставляет удобный механизм хранения различных версий одного и того же приложения и сетевой протокол доставки данных. Как видно на рисунке, таковых репозиториях используется несколько, это было сделано с целью распределения нагрузки на сетевые каналы для доставки приложений и данных к облакам.

При работе с системой пользователю предлагается выбор из уже загруженных приложений (установленных сервисов), из которых он выбирает требующийся ему, далее происходит параметризация приложения и постановка получившегося задания в пул заданий. Если среди имеющихся в данный момент приложений нет требуемого, то, при наличии у пользователя соответствующего разрешения, есть возможность загрузки собственного приложения. Запуск задания на счет происходит по инициативе облаков: когда у некоторого облака появляются свободные ресурсы, оно обращается к пулу заданий на центральном сервере и «забирает» одно из них, такое, которое максимально заполняет свободные ресурсы облака.

Как уже было сказано выше, вычислительные облака формируются («конденсируются») динамическим образом: после установки на компьютер специальной программы он обращается на центральный сервер системы, передавая информацию о себе: идентификационный номер, IP-номер, сведения о своей программной архитектуре, и центральный сервер приписывает его к какому-либо управляющему модулю. Данный процесс называется динамическим, так как ни один из узлов системы не приписывается жестким образом: в иной момент времени каждый узел может стать как управляющим, так и счетным, роль узла определяется центральным сервером на основании состояния системы, географического расположения и степени близости к уже определенным управляющим узлам. Причин такой динамики множество: наличие в очереди задания, которому требуется большее количество ресурсов, чем те, которыми располагают имеющиеся в данный момент облака, неработоспособность каналов связи или их высокая загруженность и т.д.

В качестве заключения хочется сказать, что по сравнению с существующими аналогами система SkyTS имеет ряд преимуществ, среди которых: простота реализации, безопасность за счет использования виртуальных машин, масштабируемость, динамичность вычислительной структуры и ее топологии, нацеленность на решение прикладных задач в области инженерного анализа и суперкомпьютерного моделирования, поддержка отказоустойчивости.

Работы, положенные в основу данной статьи, были выполнены в рамках:

- проекта «Разработка и реализация языков T++ и соответствующих им средств для эффективной поддержки высокопроизводительного параллельного счета» по Программе фундаментальных научных исследований ОНИТ РАН «Архитектура, системные решения, программное обеспечение, стандартизация и информационная безопасность информационно-вычислительных комплексов новых поколений»;
- программы «СКИФ-ГРИД» «Разработка и использование программно-аппаратных средств ГРИД-технологий и перспективных высокопроизводительных (суперкомпьютерных) вычислительных систем семейства «СКИФ»» (2007–2009 гг.);
- научно-технической программы Союзного государства «Развитие и внедрение в государствах-участниках Союзного государства наукоемких компьютерных технологий на базе мультипроцессорных вычислительных систем (шифр «ТРИАДА»)» (2005–2008 гг.)

ЛИТЕРАТУРА:

1. С.М. Абрамов, А.И. Адамович, А.В. Инюхин, А.А. Московский, В.А. Роганов, Ю.В. Шевчук Т-система с открытой архитектурой // Суперкомпьютерные системы и их применение SSA'2004: Труды международной научной конференции, 26–28 октября 2004 г., Минск, ОИПИ НАН Беларуси. — Минск, 2004, с. 18–22.
2. С.М. Абрамов, Г.И. Есин, И.М. Загоровский, Г.А. Матвеев, В.А. Роганов Принципы организации отказоустойчивых параллельных вычислений для решения вычислительных задач и задач управления в Т-Системе с открытой архитектурой (OpenTS) // Программные системы: теория и приложения (PSTA-2006): Труды Международной научной конференции, 23–28 октября 2006 г., Переславль-Залесский, ИПС РАН. — Переславль-Залесский, 2006, с. 257–264.
3. С.М. Абрамов, А.А. Московский, В.А. Роганов, П.Е. Велихов Пути ученого. Е.П. Велихов: Суперкомпьютерные и GRID-технологии. — М.: РНЦ «Курчатовский институт», 2007. — ISBN 978-5-9900996-1-6. — 314–326 с., Под общей редакцией академика РАН В.П. Смирнова.
4. С.М. Абрамов, А.А. Кузнецов, В.А. Роганов Кроссплатформенная версия Т-системы с открытой архитектурой // Параллельные вычислительные технологии (ПaVT'2007): Труды Международной научной конференции, 29 января–2 февраля 2007 г., Челябинск. — Челябинск: изд. ЮУрГУ, 2007, с. Т.1, 115–121.
5. А.А. Кузнецов, В.А. Роганов Экспериментальная реализация отказоустойчивой версии системы OpenTS для платформы Windows CCS // Суперкомпьютерные системы и их применение (SSA'2008): Труды Второй Международной научной конференции, 27–29 октября 2008 г., Минск. — Минск: ОИПИ НАН Беларуси, 2008. — ISBN 978-985-6744-46-7, с. 65–70.
6. С.М. Абрамов, Г.И. Есин, Ф.А. Коряка, А.А. Московский, В.А. Роганов. 2004. Создание испытательного полигона для grid-приложений в городе Переславле-Залесском, Рос. Научный сервис в сети Интернет: Труды Всероссийской научной конференции, 20–25 сентября 2004 г. Новороссийск, Изд-во МГУ, М., pp. 93–94.
7. Официальный сайт системы программирования OpenTS: Электронный сетевой ресурс, <http://www.opents.net>