

КОНФИГУРИРОВАНИЕ И АДАПТАЦИЯ ПРИЛОЖЕНИЙ РАСПРЕДЕЛЁННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ НА ОСНОВЕ ТЕХНОЛОГИИ АКТИВНЫХ ПАКЕТОВ

С.В. Шибанов, А.А. Мезенков, О.А. Шевченко, А.С. Илюшкин

1. Проблема конфигурирования и адаптации приложений распределенных информационных систем

В настоящее время пользователи особенно заинтересованы в гибких и масштабируемых информационных системах, которые могли бы достаточно быстро подстраиваться под постоянно меняющиеся требования, связанные с изменениями в предметной области, развитием бизнес-процессов предприятия, модернизацией аппаратно-программного окружения. Разнообразие стоящих перед различными компаниями задач столь велико, что не позволяет успешно применять унифицированные информационные системы, так как готовые решения зачастую не позволяют построить полноценную информационную инфраструктуру.

Конфигурирование и адаптация программных приложений необходимы для приведения их в соответствие с изменившимися требованиями. Программное обеспечение можно назвать конфигурируемым, если его можно настроить через стандартные интерфейсы без программирования дополнительных функций и/или без изменения исходного кода программы. Если требования к системе нельзя удовлетворить без программирования или изменения исходного кода, то программное обеспечение принято называть адаптируемым [1].

Адаптация готового решения требует значительно больше ресурсов, причем не только в момент внесения изменений, но и в течение всего жизненного цикла решения. На модификацию программного кода в соответствии с новыми спецификациями потребуется больше усилий квалифицированных специалистов и неизбежно затраты на реализацию будут выше. Тем не менее, зачастую невозможно избежать внесения изменений или проведения дополнительных разработок, что связано с ограниченными возможностями конфигурирования даже достаточно гибких приложений.

Особенно остро проблема конфигурирования и адаптации приложений стоит в распределенных информационных системах (РИС). Это связано со сложностью тиражирования и синхронизации данных, метаданных и приложений между компонентами РИС (подсистемами, узлами), расположенными на разных уровнях организационной структуры компании или ведомства. Причины кроются в территориальной распределенности компонент РИС, их информационной и реализационной гетерогенности. Для поддержания всех элементов распределенной информационной системы в согласованном состоянии необходимо организовать распространение и развертывание обновлений всех видов информационных ресурсов и приложений на узлах РИС (процесс тиражирования), а также сбор сведений с узлов РИС о применении обновлений для контроля и анализа результатов тиражирования (процесс синхронизации).

В связи с этим весьма актуальной является задача разработки программных средств конфигурирования и адаптации приложений РИС, реализующих автоматическое тиражирование и синхронизацию данных, метаданных и приложений. Кроме прочего, средства конфигурирования и адаптации приложений РИС должны обладать возможностью обратной связи, позволяющей целенаправленно собирать сведения с узлов РИС, в том числе, непосредственно (или опосредованно) общаться пользователю с разработчиками.

Для настройки и обновления программных продуктов в настоящее время предлагаются различные решения, но все они не являются комплексными и имеют ряд недостатков и ограничений. Например, анализ средств конфигурирования приложения 1С:Предприятие 8 [2] позволил выделить следующие недостатки:

- отсутствие возможности резервного копирования и сохранения прежних конфигураций системы;
- отсутствие возможности восстановления конфигурации в случае неудачного обновления;
- сложность процесса обновления, требующая обязательное присутствие опытного пользователя.

Ряд компаний разработчиков (например, СТМ, MLS IT Systems) создали системы обновления (такие как, СТМ-Сервис, ArchiUpdate) для своих программных продуктов, но все они решают локальные задачи обновления определенного круга приложений.

Технология ClickOnce позволяет создавать приложения Windows с возможностью установки и запуска обновлений с минимальным вмешательством пользователя [3]. Разработчики ClickOnce постарались снизить сложность процесса развертывания и обновления приложений, уменьшить нагрузку на компьютер пользователя, решить проблему безопасности. Существенным недостатком технологии ClickOnce является обязательное обращение к серверу обновлений при запуске и проверка версий установленных компонент независимо от наличия новых обновлений.

Средства обратной связи можно разделить на системы автоматической генерации и отправки отчетов об ошибках и системы обращения к разработчикам. Средства генерации и отправки отчетов об ошибках

реализованы во многих современных приложениях в виде отдельных модулей: Windows Error Reporting для приложений ОС семейства Windows, Quality Feedback Agent (Mozilla Foundation), Error Report Tools (LibreOffice).

Во многих приложениях имеется возможность написания пользователями писем разработчикам и их отправки средствами электронной почты. Характерными ограничениями современных средств обратной связи являются отсутствие механизмов отправки текущего состояния и конфигурации системы; невозможность подготовки и отправки контекста локализации ошибок для идентификации причин их возникновения; отсутствие механизмов отслеживания обращений пользователей и отправки индивидуальных конфигураций с внесенными модификациями.

Таким образом, можно констатировать, что большинство существующих реализаций средств конфигурирования и адаптации приложений имеют отдельные или комплексные ограничения по видам обновляемых информационных ресурсов и приложений, по архитектурам конфигурируемых и адаптируемых приложений, по наличию активного поведения, по сфере применения и т.д.

2. Технология активных пакетов для тиражирования и синхронизации гетерогенной информации

Для решения задач конфигурирования и адаптации приложений РИС с целью преодоления ограничений и недостатков существующих решений предлагается использовать технологию активных пакетов [5, 6].

Понятие пакета лежит в основе большинства современных систем тиражирования и синхронизации информационных ресурсов. С помощью пакетов управления (Management Packs) в приложении System Center Operations Manager (SCOM) [7] компании Microsoft осуществляется сбор и консолидация данных для мониторинга ИТ-сервисов, приложений, серверов в гетерогенной среде Windows, UNIX и Linux.

В предлагаемой технологии активный пакет консолидирует в себе данные, метаданные и приложения, а также правила обработки пакетов на узлах РИС. Обработка активных пакетов на узле РИС включает:

- анализ и разбор поступившего пакета;
- выделение и актуализация доставленной информации;
- переадресацию пакета на другие узлы РИС;
- отправку информационного пакета узлу-отправителю и пр.

С помощью активных пакетов может не только тиражироваться гетерогенная информация, но и собираться сведения о состоянии приложений на узлах РИС, а также осуществляться обратная связь пользователей с разработчиками.

2.1 Структура и назначение активного пакета

Активный пакет представляет собой логическую сущность, инкапсулирующую в себе:

- данные и метаданные в виде набора каталогов и файлов различного формата, фрагментов баз данных, записей в системный реестр и пр.;
- двоичные и текстовые информационные фрагменты;
- приложения в виде исполняемых файлов, динамически подключаемых библиотек, настроечных файлов и пр., сгруппированных в различные каталоги файловой системы;
- приложения, изменяющие поведение системы управления активными пакетами в узлах РИС.

Активный пакет состоит из заголовка инициализации (Packet Header, PH) и локального репозитория для хранения информационного контента (Packet Local Repository, PLR).

PH необходим для первичной идентификации активного пакета системой управления, последующего анализа содержимого пакета и запуска на исполнение содержащегося в нём кода.

PLR активного пакета представляет собой древовидное хранилище гетерогенной информации. Активный пакет может содержать внутри себя другие пакеты. Для уменьшения размера активного пакета, например, при большом количестве дублирующихся файлов, используются внутренние перекрестные ссылки.

Активный пакет может использоваться, как для распространения, так и для сбора информации, позволяя абстрагироваться при этом от используемых средств связи между узлами РИС. В процессе сбора информации, активный пакет, перемещаясь между узлами РИС, сохраняет внутри себя, кроме данных, адреса узлов РИС, где они были получены.

Активный пакет может модифицировать свою структуру во время нахождения в узле РИС, получая требуемые сведения от системы управления активными пакетами узла.

В современных средствах тиражирования и синхронизации информации, сбора и консолидации данных об ИТ-инфраструктуре, основанных на технологии пакетов, прохождением пакета по сети и получением необходимых от него сведений управляет программное обеспечение узла. В данной технологии предлагается возложить обязанности по управлению пакетом на активный пакет, размещая в нем исполняемые файлы, библиотеки или скрипты интерпретируемом языке (Perl, Bash, Python), которые могут разворачиваться в системе управления активными пакетами узла, изменяя ее поведение. Таким образом, можно рассматривать активный пакет в качестве субъекта управления. В самом простом случае, задачей пакета является копирование собственного тела на узел РИС и/или передача его дальше по сети [1,2].

2.2 Система управления активными пакетами на узлах РИС

Система управления активными пакетами на узле РИС включает среду исполнения активных пакетов (Node Execution Environment, NEE) и систему хранения активных пакетов (System Storage of Active Packets, SSAP).

Узел РИС предоставляет пакету свои вычислительные ресурсы. NEE является необходимым связующим звеном между активными компонентами пакета и вычислительными ресурсами узла, работающими с NEE через прослойку операционной системы [8]. Среда исполнения активного пакета NEE включает набор интерфейсов, в том числе для планирования поведения активного пакета при повторном запуске или в режиме ожидания поступления других пакетов. Активные пакеты в интервалах между запусками хранятся средствами системы хранения активных пакетов (SSAP).

Активность со стороны NEE проявляется только в момент получения нового активного пакета. Обмен пакетами может осуществляться только между узлами РИС, имеющими системы управления активными пакетами и соответствующие интерфейсы.

На рис. 1 представлена обобщенная архитектура системы управления активными пакетами на узлах РИС.

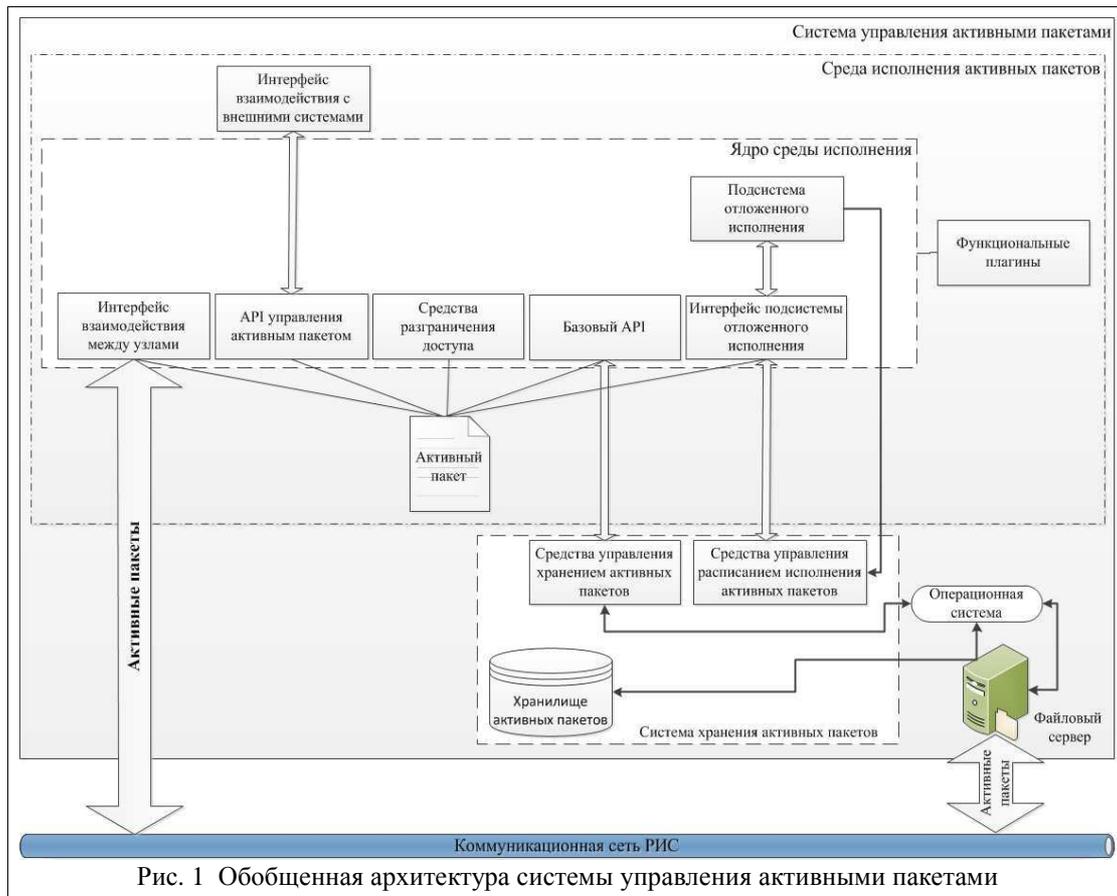


Рис. 1 Обобщенная архитектура системы управления активными пакетами

Разбор, анализ и исполнение активных пакетов на стороне узла РИС реализует среда исполнения активного пакета NEE, которая предоставляет пакету всего один метод Execute для выполнения интерпретируемых команд или использования внутренних программных интерфейсов, обеспечивающих доступ к функциям ядра среды исполнения (NEE API) [8].

Подсистема отложенного исполнения (Deferend Execution Subsystem, DES) входит в состав ядра среды исполнения NEE и позволяет активному пакету программировать NEE для периодической повторной обработки активного пакета. Такое поведение может использоваться для сбора пакетом сведений о состоянии приложений узла РИС. Каждый активный пакет может программировать только собственный повторный вызов и не может напрямую управлять вызовом другого активного пакета. При поступлении активного пакета, NEE, прочитав заголовок инициализации PH пакета, используя метод Execute, выполняет присутствующий там код. После этого пакет сможет управлять «своей жизнью» на узле РИС самостоятельно, планируя время своего следующего запуска системой исполнения при помощи DES, управление которой так же осуществляется через метод Execute.

Возможны несколько вариантов передачи пакета между узлами. В простейшем случае каждая среда исполнения активного пакета знает сетевой адрес всех других сред исполнения в сети, формируемой при

помощи VPN, и пакет отправляется непосредственно на определённый адрес. Можно также определить участки сети с наименьшей латентностью или лучшими скоростными характеристиками, и передать пакет таким образом, чтобы временные затраты на пересылку были минимальны.

Функциональные плагины (модули расширения) реализуются в виде динамически подключаемых библиотек и используются для изменения функциональности среды исполнения, доступной активному пакету во время его исполнения. Хранилище плагинов реализуется в виде каталога файловой системы, в который помещаются плагины, доставленные приходящими на узел активными пакетами. Среда исполнения активных пакетов отслеживает изменения содержимого каталога-хранилища и при появлении нового плагина выполняют его загрузку в систему. Благодаря применению технологии плагинов реализуется динамическое приведение профиля среды исполнения в соответствие с изменившимися внешними условиями без повторной ручной установки дополнительного программного обеспечения.

С помощью активного пакета можно распространять не только обновление среды исполнения, но и адреса узлов. Например, если узлу-отправителю активных пакетов изначально не известны все участники обмена.

2.3 Программная реализация среды исполнения активных пакетов

Среда исполнения активных пакетов на узле РИС реализована в виде нескольких программных компонентов (сборок):

1. ActivePackage.ExecutionEnvironment.Service.exe – главный исполняемый модуль NEE, запускается вручную или автоматически операционной системой и работает в режиме службы, создает экземпляры объектов классов среды исполнения.
2. ActivePackage.ExecutionEnvironment.Core.dll – библиотека ядра NEE, управляет созданием экземпляров классов для работы с сетью и виртуальным окружением среды исполнения, реализует функции запуска и планирования исполнения активных пакетов.
3. ActivePackage.ExecutionEnvironment.Network.dll – библиотека для сетевого взаимодействия между NEE РИС, предоставляет средства для начального этапа обработки поступающих пакетов, а так же интерфейс обмена данными через службу на основе Microsoft WCF.
4. ActivePackage.ExecutionEnvironment.Sandbox.dll – библиотека средств управления виртуальным окружением пакета, контролирует исполнение пакета, отслеживает потребление ресурсов ОС, ограничивает доступ к глобальным библиотекам и функциям системы, управляет записью активных пакетов на диск bkb в выделенную область оперативной памяти.

Диаграмма классов среды исполнения активных пакетов представлена на рис. 2.

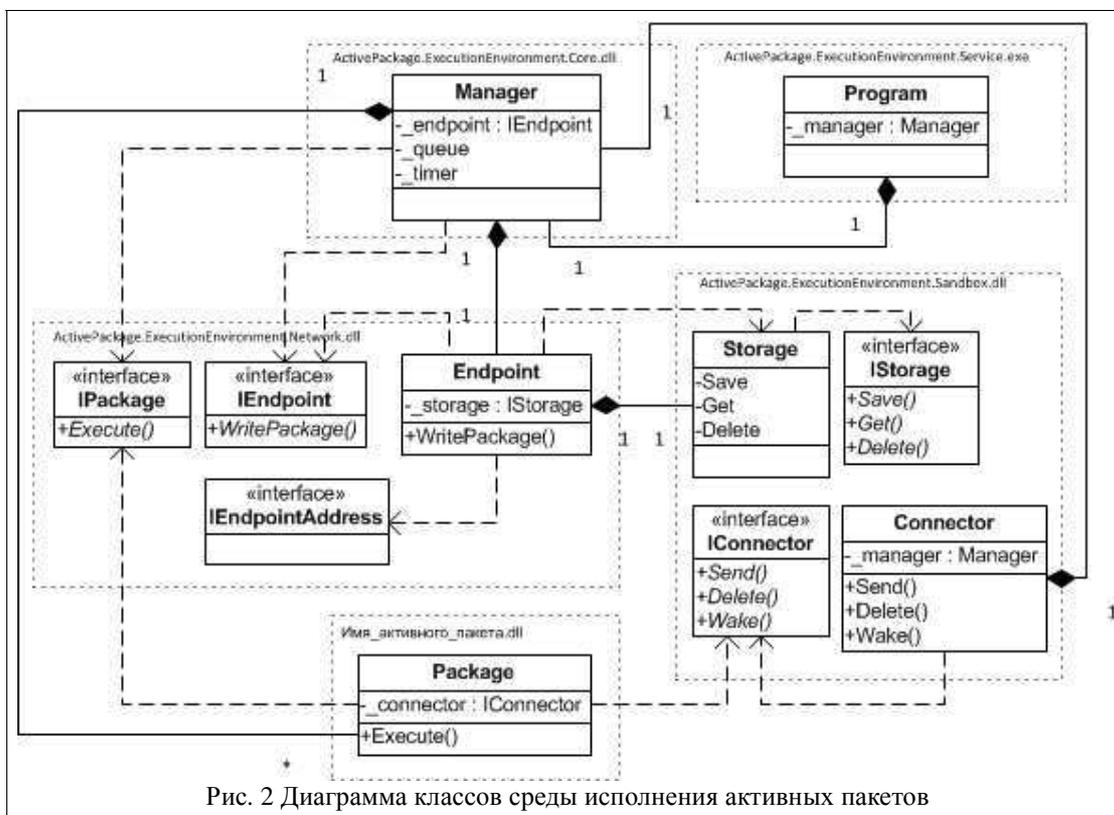


Рис. 2 Диаграмма классов среды исполнения активных пакетов

Основу программного обеспечения среды исполнения активных пакетов составляет статический класс ActivePackage.ExecutionEnvironment.Service.Program с методом Main, через который производится запуск

основной логики приложения. В рамках данного класса создаётся единственный экземпляр управляющего класса `ActivePackage.ExecutionEnvironment.Core.Manager`, который, в свою очередь, инициализирует вспомогательные экземпляры классов для работы с сетью и контроля запуска пакетов на основе файла конфигурации `ActivePackage.ExecutionEnvironment.Service.conf`.

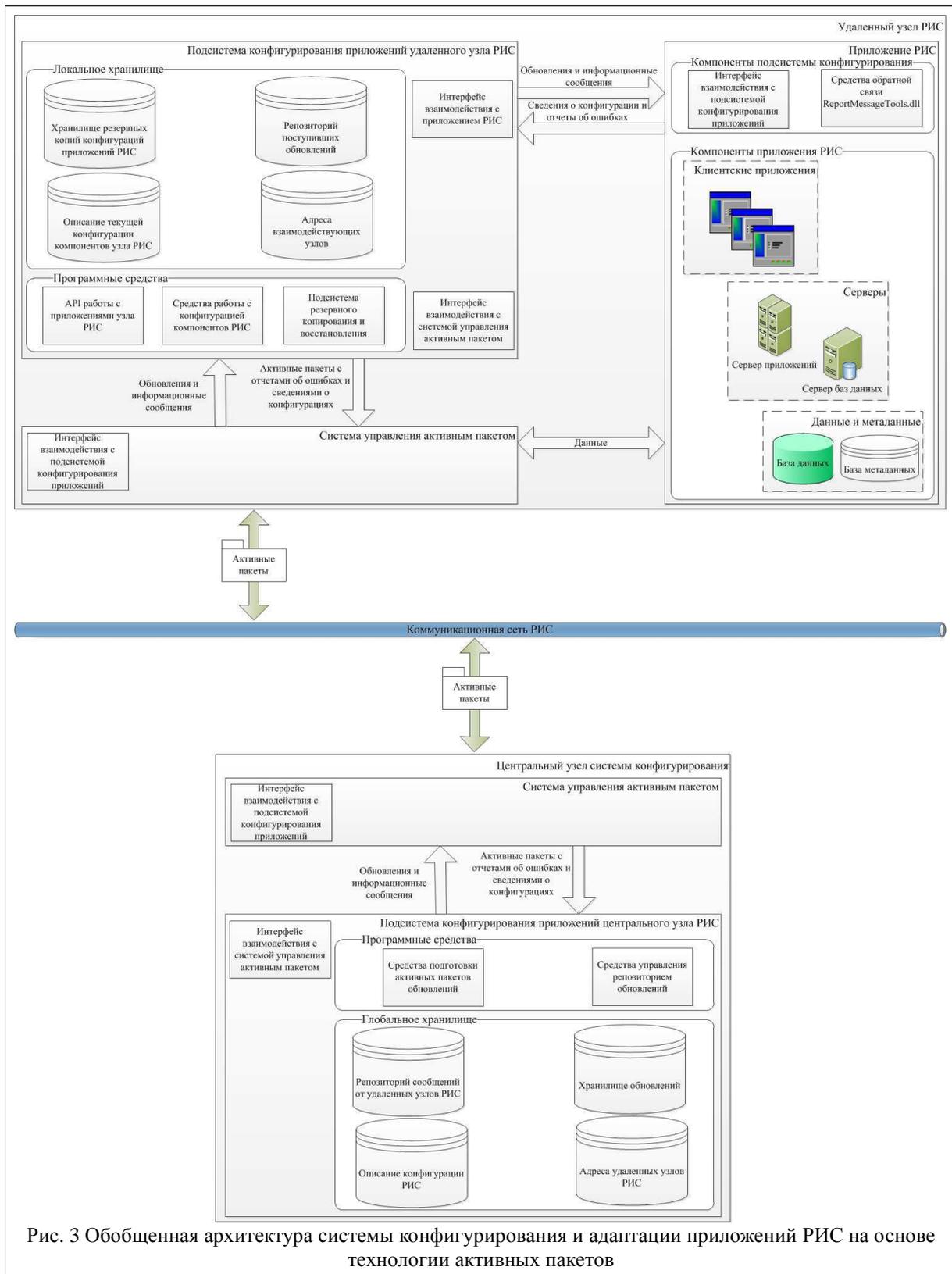


Рис. 3 Обобщенная архитектура системы конфигурирования и адаптации приложений RIS на основе технологии активных пакетов

После выполнения метода инициализации `Initialize` класса `ActivePackage.ExecutionEnvironment.Core.Manager` по адресу из файла конфигурации запускается сетевая

служба для приёма и передачи активных пакетов, а также производится чтение содержимого каталога для хранения активных пакетов и работы виртуального окружения.

Передача активного пакета на удалённый узел осуществляется методом WritePackage. При поступлении пакета на узел РИС выполняются следующие действия:

1. Вызов метода Save класса Storage, который производит запись пакета на диск или сразу выполняет его загрузку пакета в зависимости от его размера.
2. Уведомление управляющего класса Manager о поступлении нового пакета через соответствующее событие.
3. Помещение пакета в очередь первичного вызова в классе Manager.

В методе-обработчике события срабатывания глобального таймера Tick (OnGlobalTimer) класса Manager запускается метод UpdateQueue выборки из очереди состояний соответствующего пакета. Затем производится запуск пакета на выполнение с проверкой состояний других пакетов в очереди. Выбирается пакет с минимальным оставшимся интервалом времени до следующего вызова, и выполняется программирование системного таймера на новый интервал. Активный пакет имеет доступ к ядру среды исполнения через контролирующий интерфейс IConnector из библиотеки виртуального окружения, а сама среда исполнения имеет непосредственный доступ к каждому пакету в очереди внутри класса Manager. Таким образом, выполняется условие замкнутости контура управления ходом выполнения активных пакетов.

3. Система конфигурирования и адаптации РИС на основе технологии активных пакетов

Технология тиражирования и синхронизации гетерогенной информации с помощью активных пакетов положены в основу системы конфигурирования и адаптации РИС (Configuration Management System, CMS), компоненты которой размещаются вместе с приложениями РИС на узлах сети. Обобщенная архитектура CMS представлена на рис. 3.

На центральном узле CMS размещается глобальное хранилище, которое образуют: хранилище обновлений в виде коллекций активных пакетов; описание конфигурации РИС; репозиторий сообщений от удаленных узлов РИС; репозиторий адресов удаленных узлов РИС. На центральном узле CMS также располагаются программные средства подсистемы подготовки активных пакетов для конфигурирования и адаптации приложений РИС.

На удаленных узлах РИС размещаются локальное хранилище, которое включает хранилище резервных копий конфигураций приложений РИС; репозиторий поступивших обновлений; метаданные, описывающие конфигурацию компонент узла РИС, адреса взаимодействующих узлов РИС. Программные компоненты CMS удаленного узла включают: API работы с приложениями узла РИС, средства работы с конфигурацией компонентов РИС, подсистему резервного копирования и восстановления.

Для отправки, получения и исполнения активных пакетов на всех узлах РИС размещается система управления активными пакетами, которая также может использоваться для обмена внутренней информацией РИС.

При создании каждому пакету присваивается уникальный идентификатор GUID. Пакет, созданный в CMS, передается в систему управления активными пакетами центрального узла CMS для отправки на удаленные узлы РИС. При получении активного пакета система управления активными пакетами удаленного узла РИС производит анализ и разбор пакета, а затем передает его CMS узла. CMS удаленного узла РИС осуществляет поиск обновляемых приложений по GUID, сохраняет текущую конфигурацию для отката в случае сбоя или ошибки обновления и выполняет обновление с учетом метаданных описания конфигурации.

Подсистема обратной связи позволяет отправить отчет об ошибке, а также замечания или пожелания пользователя с удаленного узла на центральный узел CMS. Процесс отправки отчета об ошибке или обращения пользователя включает в себя фиксацию сообщения в репозитории, создание информационного активного пакета и передачу пакета системе управления активными пакетами удаленного узла РИС.

Программные средства обратной связи реализованы авторами статьи в виде библиотеки-плагины и применялись в автоматизированной информационной системе «Прокуратура-статистика» 2.0 для прокуратуры Пензенской области [4]. Плагин ReportMessageTools.dll в случае возникновения исключительной ситуации фиксирует состояние приложения, готовит дампы описания исключительной ситуации, передает отчет для создания и отправки информационного активного пакета.

Заключение

В статье рассматривается задача конфигурирования и адаптации приложений распределенных информационных систем. Предлагается использовать технологию активных пакетов, разработанную авторами, для построения средств конфигурирования и адаптации приложений РИС. Разработанные средства управления активными пакетами, а также конфигурирования и адаптации приложений РИС находятся в стадии опытной реализации. В настоящее время проводятся всестороннее тестирование и экспериментальная оценка полученных разработок.

ЛИТЕРАТУРА:

1. Х. Груневеген. Конфигурирование или адаптация?// Открытые системы , № 06, 2008.
2. М. Радченко. Профессиональная разработка в системе 1С:Предприятие 8. СПб.: Питер, 2011. – 808 с.
3. М. Мак-Дональд. WPF: Windows Presentation Foundation в .NET 4.0 с примерами на C# 2010 для профессионалов. М.: Вильямс, 2011 – 1024 стр.
4. А.А. Мезенков, С.В. Шибанов. Повышение надёжности клиент-серверных приложений в информационных системах// Технологии Microsoft в теории и практике программирования. Материалы конференции. Н. Новгород: Изд-во ННГУ, 2010. – с. 286 – 289.
5. А.С. Илюшкин, С.В. Шибанов, О.А. Шевченко. Концепция активного пакета для распространения данных в распределенных системах // Технологии Microsoft в теории и практике программирования. Материалы конференции. Н. Новгород: Изд-во ННГУ, 2010. – с. 158 – 160.
6. С.В. Шибанов, А.С. Илюшкин, О.А. Шевченко. Активная система управления тиражированием и синхронизацией метаданных в корпоративных информационных системах. // Известия высших учебных заведений. Поволжский регион. Технические науки. – № 1. Пенза, 2010. – с. 25 – 37.
7. В. Price, J. Paul Mueller, S. Fenstermacher. Mastering System Center Operations Manager 2007. Published by Wiley Publishing, 2007.
8. А.С. Илюшкин, С.В. Шибанов, О.А. Шевченко. Система исполнения активного пакета в узлах распределенной системы // Технологии Microsoft в теории и практике программирования. Материалы конференции. Н. Новгород: Изд-во ННГУ, 2010. – с. 165 – 167.