УДК 681.518

РОЛИК А.И.,
ВОЛОШИН А.В.,
ГАЛУШКО Д.А.,
МОЖАРОВСКИЙ П.Ф.,
ПОКОТИЛО А.А.

СИСТЕМА УПРАВЛЕНИЯ КОРПОРАТИВНОЙ ИНФОРМАЦИОННО-ТЕЛЕКОММУНИКАЦИОННОЙ ИНФРАСТРУКТУРОЙ НА ОСНОВЕ АГЕНТСКОГО ПОДХОДА

В статье разработана структура системы управления корпоративной информационно-телекоммуникационной инфраструктурой на основе агентского подхода с интеграцией методов активного и пассивного мониторинга и консолидацией централизованного и децентрализованного управления. Определены структуры серверного и агентского модулей, механизмы их взаимодействия. Разработан подход к распределению объектов мониторинга и управления по агентам, а также реактивный метод обновления состояний элементов ИТ-инфраструктуры.

The article describes the structure of the agent-based corporative information-telecommunication infrastructure control system, using the integration of active and passive monitoring methods and the consolidation of centralized and decentralized control methods. The article also determines the structure of the server and agent modules and algorithms of their interaction. The approach to the distribution of monitoring objects among the agents and the reactive method of updating their states were also developed.

Введение

В настоящее время показатели эффективности работы практически всех предприятий и организаций существенно зависят от эффективности функционирования корпоративных информационно-телекоммуникационных систем (ИТС). ИТС представляет собой совокупность информационной системы и телекоммуникационной сети и является базисом ИТ-инфраструктуры крупных предприятий или организаций. ИТС предоставляет аппаратно-программные средства для автоматизации бизнес-процессов, обеспечивает информационное и телекоммуникационное взаимодействие между функциональными и обеспечивающими системами, подразделениями, сотрудниками и пр. Для управления корпоративными ИТС используются универсальные фирменные или разрабатываются собственные адаптированные под потребности предприятия системы управления ИТ-инфраструктурой (СУИ) [1, 2]. СУИ реализуется в виде программного продукта, осуществляющего мониторинг состояния и анализ работоспособности как отдельных элементов ИТС, так и подсистем. Для этого СУИ производит мониторинг сетевого оборудования и решение задач сетевого управления, проверку функционирования и контроль работоспособности сетевых сервисов и приложений, управление аппаратными и программными элементами ИТС, и пр.

Увеличение стоимости информационно-телекоммуникационных ресурсов и технологий, существенная зависимость бизнеса от эффективности работы ИТС, постоянная модернизация и появление новых информационных технологий, ужесточение требований к эффективности и бесперебойности работы ИТС делают актуальной задачу совершенствования существующих или разработки новых перспективных СУИ.

Постановка проблемы

Для решения задач управления ИТС прежде всего необходимо получить информацию о состоянии компонентов ИТС. В настоящее время для решения этой задачи применяются различные методы мониторинга [3], среди которых можно выделить два наиболее простых и очевидных альтернативных метода – активный и пассивный мониторинги. В первом случае со стороны СУИ производится опрос элементов с целью получения значений параметров функционирования, во втором, – элементы ИТС путем отправки специальных сообщений самостоятельно информирует СУИ об изменении своего состояния в случае пересечения текущими значениями контролируемых параметров

установленных пороговых значений. Каждый из этих методов имеет свои преимущества и недостатки, и вполне логично, что перспективным является комбинированный – интегральный подход, сочетающий в себе преимущества обоих методов. При этом возникают задачи распределения функций между методами, интеграции результатов работы каждого из них, разработки структуры центральной и агентской частей СУИ. Это же касается и реализации в СУИ управляющих функций, которые могут в виде команд управления поступать из центра управления или генерироваться в агентах при отслеживании и реализации общей политики управления.

Разработке СУИ с интеграцией методов активного и пассивного мониторинга, а также реализации соответствующих управляющих функций и посвящена данная статья.

Анализ публикаций

В работе [1] определены требования, которым должна удовлетворять перспективная СУИ. Разработана структура серверной части, но не предложены принцип построения агентов и схема взаимодействия серверной и агентской частей СУИ.

В [3] методы мониторинга элементов ИТ-инфраструктуры рассматриваются без привязки к структуре СУИ.

В [4] рассматривается агентская схема построения СУИ, определена концепция мобильных агентов и мета-агентов, однако отсутствуют принципы построения и структура агентов, а также не представлены механизмы взаимодействия агентов с сервером.

Работа [5] посвящена интеграции методов активного и пассивного мониторинга элементов коммуникационной сети в GRID системах. В ней также не рассматриваются принципы построения агентов и вопросы их реализации.

Целью статьи является разработка системы управления информационно-телекоммуникационной инфраструктурой на основе агентского подхода с интеграцией активного и пассивного мониторинга и консолидацией методов централизованного и децентрализованного управления.

Структура СУИ

Современные СУИ представляют собой аппаратно-программные комплексы, предназначенные для мониторинга состояния, анализа работоспособности и управления аппаратными и программными средствами ИТС.

СУИ осуществляют:

— мониторинг и анализ функционирования программно-технических средств функциональных и технологических подсистем;

— организационное и автоматизированное управление функционированием программно-технических средств функциональных и технологических подсистем;

СУИ обеспечивают:

— контроль параметров функционирования аппаратного и программного обеспечения серверов и рабочих станций ИТС;

— анализ данных о функционировании подсистем и распределенных приложений;

— накопление, анализ и отображение результатов контроля состояния и работоспособности элементов ИТС;

— своевременное обнаружение, локализацию и управление устранением неисправностей в ИТС;

— прогнозирование критических состояний системы и предотвращение возникновения отказов в работе ИТ-системы на ранней стадии;

— интеграцию средств администрирования функциональных и технологических подсистем;

— выработку команд и сигналов управления техническими и программными средствами подсистем в автоматическом и ручном режиме для принятия решения системным администратором;

— оптимизацию планирования задач обслуживания системы с учетом загруженности ресурсов ИТС;

— оптимальное распределение общих вычислительных и коммуникационных ресурсов между бизнес-процессами с учетом их значимости;

— автоматизацию планирования деятельности персонала по управлению функционированием при техническом обслуживании и других регламентных работах;

— регистрацию и документирование событий в ИТС, действий администраторов по управлению функциональными и технологическими подсистемами.

Для реализации этих и множества других функций в СУИ используется централизованное или децентрализованное управление, активный или пассивный мониторинг. Как правило, СУИ, построенная по централизованному

принципу, использует метод активного мониторинга, когда сервер СУИ получает информацию от элементов ИТС путем запроса, и метод непосредственного управления элементами путем отправки соответствующих команд. Децентрализованная СУИ реализуется на интеллектуальных агентах, при этом основная часть задач мониторинга и управления реализуется агентом, а сервер собирает информацию о событиях, генерируемых агентами, и вырабатывает политику управления, обязательную для всех агентов.

Наибольший интерес представляет принцип построения СУИ, основанный на рассмотрении в тесной взаимосвязи методов централизованного и децентрализованного управления, активного и пассивного мониторинга. В этом случае производится рациональное разделение задач централизованного и децентрализованного управления и интегрированное использование активного и пассивного мониторинга. Именно такой принцип и реализован в предлагаемой СУИ.

В крупных, территориально распределенных на существенных расстояниях ИТС, содержащих большое количество элементов, СУИ целесообразно строить по иерархическому принципу (см. рис. 1).

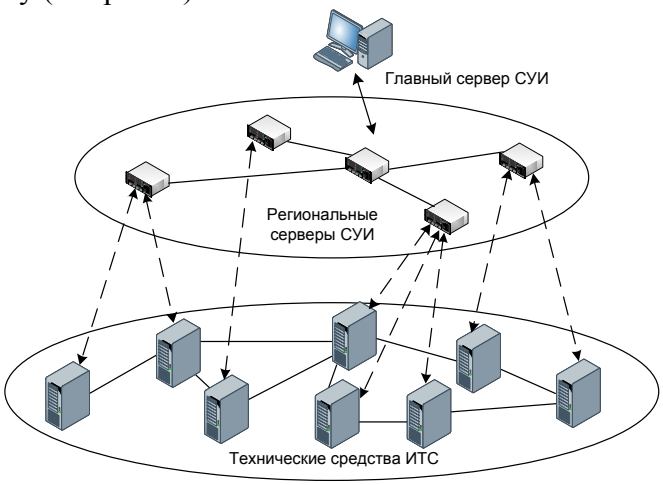

Рис. 1. Иерархия серверов СУИ

В этом случае на верхнем уровне располагается главный или центральный сервер СУИ, определяющий состояние крупных составляющих ИТС – функциональных и технологических подсистем, телекоммуникационной сети и пр., а на нижних уровнях – региональные или периферийные серверы СУИ, каждый из которых осуществляет мониторинг, анализ функционирования и управление элементами ИТС в своей географической или административной зоне ответственности. С нижних уровней на верхние поступают обобщенные данные мониторинга и информация о функционировании подсистем ИТС, расположенных в зоне ответственности периферийных серверов СУИ. С верхних на нижние – распространяются политики управления и макрокоманды. Все серверы СУИ управляются администраторами, которые находятся в состоянии иерархической соподчиненности в соответствии с организационной структурой департамента, обеспечивающего функционирование ИТ-инфраструктуры.

Взаимодействие и совместная работа компонентов СУИ осуществляется через телекоммуникационную сеть. Через сеть осуществляется взаимодействие между серверами СУИ и элементами функциональных и технологических подсистем ИТС, АРМ администраторов СУИ с администраторами функциональных, технологических подсистем и телекоммуникационной сети.

Специальное программное обеспечение разработанной СУИ состоит из трех отдельных программных компонентов – серверного, клиентского и агентского программных модулей.

Серверный программный модуль (сервер) обеспечивает взаимодействие между клиентами, предоставляет доступ к данным, обеспечивает их целостность и защиту, уведомляет администратора о критических или интересующих его событиях в ИТС, осуществляет планирование действий по мониторингу, анализ и управление элементами ИТС. Для сбора данных мониторинга и реализации управляющих воздействий сервер взаимодействует с элементами ИТС через агентские модули. Важной функцией сервера является синхронизация и обработка данных, а также уведомление подключенных агентов и клиентов о произошедших изменениях.

Клиентский программный модуль (клиент) реализует интерфейс администратора для настройки СУИ, отслеживания работоспособности и производительности элементов ИТС, контроля состояния функциональных и технологических подсистем и ИТС в целом, а также для введения команд управления при реконфигурации или проведении восстановительных мероприятий. Непосредственно мониторинг элементов ИТС и реализация управляющих воздействий осуществляется агентскими и серверным модулями, в то время как основная функция клиентского модуля – предоставление графического интерфейса для взаимодействия администратора с системой.

Агентский программный модуль (агент) контролирует элементы ИТС, анализирует их состояние и поведение. Кроме того, агенты используются при моделировании перспективного состояния, как отдельных элементов, так и всей ИТС. Это позволяет предвосхищать и предотвращать функциональные отказы еще при первых признаках их проявления, что, в свою очередь, повышает надежность функционирования ИТС. Агент запускается в виде отдельного сервиса, специализирующегося на выполнении индивидуального набора функций мониторинга и управления, конфигурируемых в зависимости от задач, решаемых агентом и объектами, которые он обслуживает. Собираемые агентом данные могут обрабатываться как на этом же, так и на других агентах, и вместе с результатами их обработки хранятся в локальной БД или отправляются на сервер.

Предлагаемая СУИ оперирует с объектами мониторинга и управления (ОМУ), а вся информация о структуре ИТС представляется в виде дерева ОМУ, при этом каждый ОМУ содержит сведения о конфигурации и текущем состоянии отдельного элемента ИТС. ОМУ – универсальное и обобщенное понятие, элемент логической модели ИТС, основная функциональная единица представления элементов ИТС в СУИ, в которой инкапсулированы те свойства и состояния элементов ИТС, которые имеют отношение к процессам управления ИТ-инфраструктурой. Состояние ОМУ, определяемое агентами СУИ, может зависеть от состояния других ОМУ. В зависимости от физического расположения ОМУ может реально обслуживаться только одним агентом, а для удаленной работы с ОМУ могут создаваться его копии на других агентах и клиентах. В таких случаях копии выступают заместителями реального ОМУ и переадресовывают ему все обращения. Относительно ОМУ определяются параметры мониторинга, механизмы выявления неисправностей и исполнения команд управления. В качестве ОМУ выступают не только физические и логические объекты, но и подсистемы, состояния процессов деятельности, бизнес-процессы и пр. Каждый компонент ИТС всех уровней иерархии содержит дерево ОМУ, листья которого – датчики контроля (сенсорная часть), а остальные вершины – ОМУ. Такой подход позволяет использовать универсальные методы работы СУИ с ОМУ, а также единый или несколько универсальных методов для оценки состояния ОМУ.

Программный сервер СУИ

Структура сервера СУИ приведена на рис. 2.

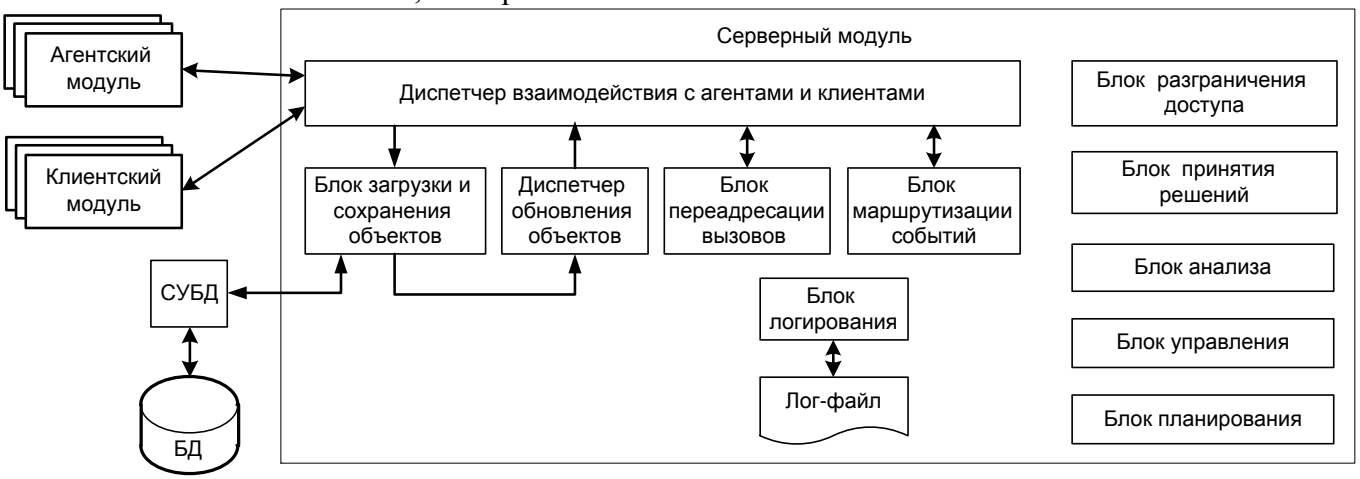

Рис. 2. Структура сервера СУИ

Сервер СУИ обеспечивает сбор данных мониторинга и управление ОМУ через агентские модули, взаимодействие с клиентскими модулями, предоставляет доступ к БД, обеспечивает защиту и целостность данных, осуществляет принятие решений по управлению ИТС, генерирует команды управления ОМУ и сигналы для администраторов, а также выполняет другие функции.

Сервер состоит из диспетчера взаимодействия с агентами и клиентами, а также следующих блоков: загрузки и сохранения объектов, диспетчера обновления объектов, переадресации вызовов, маршрутизации событий, разграничения доступа, принятия решений, анализа, управления, планирования и логирования.

Диспетчер взаимодействия с агентами и клиентами поддерживает взаимодействие и мар-

шрутизацию информационных потоков между программными компонентами СУИ, производит корректное подключение и отключение агентов и клиентов с рассылкой соответствующих уведомлений. Для получения доступа к серверу клиенты проходят процедуру регистрации, при которой проводится аутентификация и проверка прав. Любые запросы незарегистрированных клиентов, кроме запроса на регистрацию, автоматически отклоняются. Для поддержания связи подключенный клиент с заданной частотой путем передачи соответствующего запроса проверяет доступность сервера. В случае недоступности сервера, клиент осуществляет попытки повторного подключения. Сервер производит аналогичную проверку только в случае отсутствия в течение заданного времени запросов от клиента. При недоступности клиента сеанс разрывается. Для сокращения загруженности телекоммуникационной сети запрос доступности не производится при наличии активного информационного обмена между агентами и клиентами с сервером.

Блок загрузки и сохранения объектов осуществляет обработку запросов, кэширование ОМУ и запросов, а также реализует механизм синхронизации. Блок загрузки и сохранения объектов вместе с диспетчером обновления объектов производят загрузку и сохранение логических отображений ОМУ, а также обеспечивают целостность данных и своевременное уведомление клиентов о новых версиях объектов.

При получении некоторого запроса на загрузку ОМУ проверяется его наличие в кэше и, в случае отсутствия, запрос перенаправляется в БД. При этом предоставляются находящиеся в кэше готовые ОМУ, а отсутствующие конструируются из объектов, находящихся в БД, с последующим добавлением в кэш. Благодаря этому значительно повышается скорость выполнения запросов и обеспечивается гарантированное наличие в системе не более одной копии каждого ОМУ. Механизм синхронизации используется для получения от клиента обновленных версий ОМУ, а для применения изменений, внесенных данным клиентом, отслеживаются версии атрибутов ОМУ и его связей. Данный механизм позволяет редактировать один и тот же объект на разных клиентах. После выполнения синхронизации все затронутые изменениями ОМУ сохраняются в БД и передаются диспетчеру обновления объектов для оповещения подключенных клиентов.

Для обеспечения целостности и актуальности данных реализован механизм, отвечающий за своевременное оповещение клиентов о произошедших изменениях структуры дерева ОМУ и его составляющих. Для каждого подключенного клиента диспетчер строит выборки объектов из списка обновленных. Построение выборок сводится к решению NP-полной задачи, т. к. для решения задачи определения необходимости оповещения клиента об обновлении некоторого ОМУ, необходимо решить аналогичную задачу для каждого из связанных с ним объектов. Затраты на определение необходимости оповещения клиента минимальны для объектов, связанных непосредственно с клиентом, и велики для объектов, привязанных к клиентам через цепь других объектов. Поэтому нецелесообразно проводить выборку объектов некоторых типов, отправляя все измененные объекты клиенту. Для снижения сетевого трафика клиентам передаются не измененные объекты, а список их идентификаторов, а клиент запрашивает обновление только тех объектов, которые им используются.

Блок переадресации вызовов и маршрутизации событий, а также диспетчер обновления объектов позволяют клиентам работать с удаленными объектами, своевременно получая уведомления об изменениях их состояния и структуры.

При построении систем клиент-серверного взаимодействия, при котором клиенты взаимодействуют друг с другом только через сервер, возникает проблема обращения к ОМУ, месторасположение которого неизвестно. ОМУ может находиться только на одном агенте, а все его воплощения (копии) на других машинах являются заместителями, поэтому необходимо неявно перенаправлять вызов некоторой функции объекта-копии реальному объекту. При этом обращение к удаленным объектам осуществляется как к локальным, а все механизмы, осуществляющие переадресацию, скрываются от администратора СУИ. Это обеспечивает взаимную анонимность вызывающего и конечного клиентов, поскольку принятие решения о правомерности такого вызова и нахождение конечного клиента осуществляет сервер. Если при обращении к объекту выясняется, что он является копией, то задействуется встроенный механизм переадресации, который обращается к серверу, где происходит определение конечной машины, на которой находится объект, и ее доступность, после чего запрос перенаправля-

ется конечному клиенту. Конечный клиент возвращает результаты запроса на сервер, который, в свою очередь, перенаправляет результаты запросившему клиенту. В случае недоступности сервера или конечного клиента, ошибки или отказа в доступе, клиенту возвращается описание причины неудачного выполнения запроса.

Для обеспечения работы механизма реактивного обновления состояний ОМУ вне рамок одного агента должна осуществляться своевременная передача уведомлений о событиях, произошедших на данном агенте, другим агентам, имеющим зависящие ОМУ. Для подписки на события и их передачи агенты не обязаны знать о местонахождении конкретных ОМУ, обращаясь с находящимися у них копиями, как с реальными ОМУ. Реализация процесса подписки показана на рис. 3.

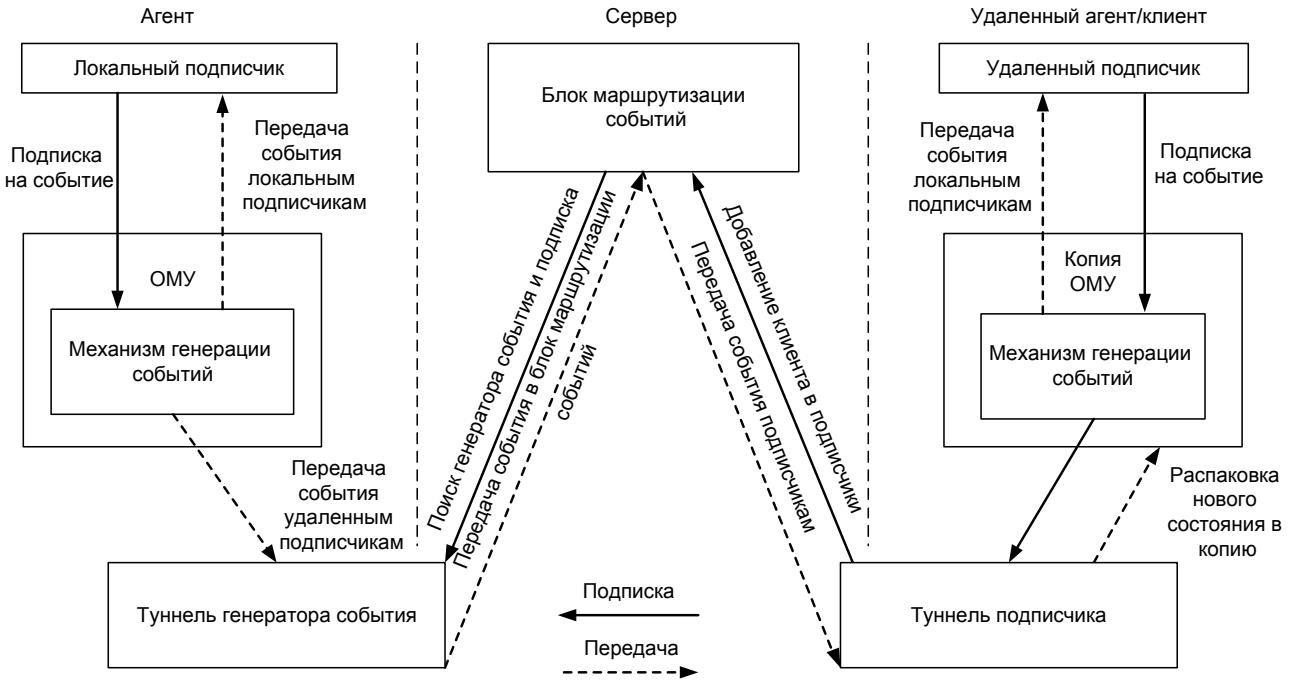

Рис. 3. Процесс подписки на удаленное событие и его вызова

Если при подписке на событие определяется, что данный объект является копией (заместителем), производится подписка на получение данного события на локальном туннеле. Туннель представляет собой общий для данного клиента механизм подписки, отписки и маршрутизации событий. Туннель переадресует подписку на серверный блок маршрутизации событий, где существует реестр событий и клиентов – подписчиков и источников. Сервер определяет клиента, которому принадлежит генерирующий событие объект, и добавляет кортеж (объект–клиент-источник–клиент-подписчик) в реестр событий. Далее диспетчер взаимодействия с агентами и клиентами определяет, подключен ли клиент-источник к серверу, и отправляет ему запрос на отсылку подписываемого события.

Подписка автоматически восстанавливается после разрывов сеансов связи. При отключении клиента-подписчика блок маршрутизации событий автоматически аннулирует все его подписки, а после восстановления подключения клиент заново осуществляет подписку. При отключении клиента-источника, подписки на его события остаются в реестре, и при повторном подключении клиента к сети блок маршрутизации событий сообщает об ожидающих его подписках. При возникновении события обновления состояния, агент делает рассылку локальным подписчикам и, в случае наличия удаленных подписок, происходит отправка этого события и сопутствующих атрибутов на серверный блок маршрутизации событий, который производит его ретрансляцию всем клиентам-подписчикам. Получив событие, клиентский туннель распаковывает его атрибуты в объект-копию, после чего генерирует аналогичное событие на локальной машине от имени объекта-копии. Обычно в качестве атрибутов события отправляется текущее значение или обновленное состояние объекта. Таким образом, использованные механизмы и подходы позволяют реализовать поддержку реактивного обновле-

ния состояний ОМУ и своевременное обновление состояний копий ОМУ на всех подписанных клиентах.

Блок разграничения доступа проверяет права агентов и клиентов при попытках получения доступа, редактирования или удаления объектов и связей, редактирования прав, вызова удаленных функций некоторого объекта или подписки на события об изменении его состояния.

Серверный блок анализа в отличие от агентского, который анализирует состояния отдельных ОМУ, анализирует состояние ИТС в целом. Блок анализирует полученные от агентов данные мониторинга и анализа параметров с использованием различных статистических методов. В частности, выявляются тренды поведения, проводится работа по архивации данных мониторинга и пр. Проактивный мониторинг осуществляется путем анализа и определения тенденции динамики изменения состояния определенных параметров функционирования системы и прогнозирования возможных будущих состояний.

На основании результатов анализа блок принятия решений вырабатывает определенные варианты воздействия на ИТС в соответствии с используемыми критериями, политиками и алгоритмами. Изначально администратор вручную задает алгоритмы поведения системы при возникновении определенных видов неисправностей, а в процессе обучения, запоминая и анализируя действия администратора, блок принятия решений отбирает наиболее часто использующиеся решения, и предлагает их администратору, который в свою очередь может на основании этих рекомендаций совершенствовать алгоритмы поведения системы. Принятые решения могут быть следующих типов: уведомление администратора о существующей или возможной в будущем неисправности; рекомендации администратору по устранению неисправности; автоматизированное принятие решения или предложение выбора из готовых вариантов решения; автоматическое устранение неисправности или реконфигурация ИТС.

Блок управления воздействует на элементы ИТС посредством передачи управляющих запросов соответствующим им ОМУ. При этом блок переадресации вызовов перенаправляет данный запрос агенту, который может непосредственно влиять на поведение данного ОМУ. Блок управления работает в ручном и автоматическом режимах, что позволяет управлять ИТС посредством клиентского интерфейса, а также приводит в исполнение принятые блоком принятия решений автоматические действия по реконфигурации ИТС, перераспределению ограниченных ресурсов [6], управлению информационными потоками [7] и пр.

Блок планирования позволяет выполнять задачи с заданным периодом или в заданное время, а блок логирования сохраняет историю системных событий.

Программный агент СУИ

Мониторинг компонентов ИТС осуществляется посредством программных агентов СУИ. Агенты СУИ – универсальные, гибкие инструменты, с модифицируемым набором функций, позволяющие контролировать элементы ИТ-инфраструктуры, анализировать их поведение и состояние. Каждый агент может специализироваться на выполнении индивидуального набора функций мониторинга и управления, задаваемого администратором, в зависимости от возлагаемых на агента задач.

Модель программного агента СУИ базируется на пяти основных элементах: сущности, действия, отношения, сообщения, проверочные функции. Эти элементы обеспечивают функционирование агентского модуля, а также связи между агентами и сервером или клиентами.

Каждый агент содержит **сущность** – логический объект, активный элемент агента, выполняющий некоторое подмножество его функций, обладающий набором свойств, характеризирующийся определенным шаблоном поведения и способностью взаимодействовать с другими сущностями, в роли которых могут выступать логические объекты агентов, клиентов или сервера СУИ.

Поведение сущностей определяется **действиями** – набором функций, которые определяют работу модели сущности и протокол взаимодействия с другими сущностями. В первом случае инициализируется библиотека функций, которая контролирует и определяет внутреннее поведение агента. Во втором – определяется стратегия внешнего взаимодействия и правила передачи и обмена информацией. Действия между различными сущностями возможны исключительно в случае установленных доверительных отношений между ними.

Отношения определяют возможность взаимодействия между собой агентов и могут быть следующих видов: доверительные и отношения контроля.

Доверительные отношения устанавливаются при необходимости взаимного обмена информацией. Такие отношения используются между группой агентов в случае общего мониторинга и анализа состояния комплексных объектов ИТ-инфраструктуры.

Отношения контроля устанавливаются между двумя сущностями и позволяют одной из них (контролеру) запрашивать и получать необходимую информацию от второй (подконтрольной) сущности. Такие отношения в первую очередь устанавливаются между сервером и агентами СУИ (см. рис. 4) при централизованном управлении.

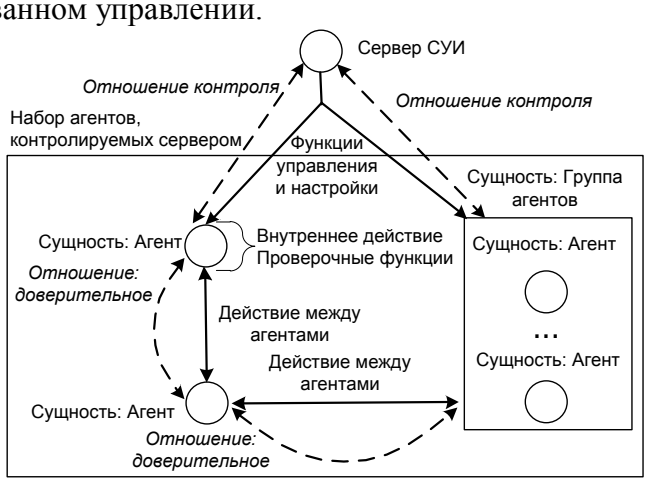

Рис. 4. Связи между сущностями

Агент обменивается информацией с другими агентами, а также с сервером посредством **сообщений**. Сообщения позволяют передавать необходимые данные и осуществлять удаленное управление. Они могут быть двух видов: служебные и системные.

Системные сообщения – сообщения, которые генерируются СУИ, для автоматического или планового обмена системной информацией и для автоматического модифицирования конфигурации отдельных сущностей. Они используются для мониторинга элементов ИТС с возможностью последующего анализа.

Служебные сообщения генерируются администратором СУИ или планировщиком для удаленного управления элементами ИТС или получения информации об их состоянии. С помощью сообщений реализован механизм оперативного реагирования на возможные неисправности.

Для контроля функционирования отдельного агента или группы агентов используются **проверочные функции**. Проверочные функции могут запускаться агентом для проведения самоконтроля и вызываться со стороны сервера. Посредством проверочных функций строится график работы как отдельных элементов, так и всей ИТС, что позволяет в режиме реального времени автоматически определять возможные «узкие места» в производительности системы и перераспределять ресурсы для оперативного решения возникающих проблем. На основе статистической обработки результатов выполнения проверочных функций строятся графики изменения значений ключевых показателей работоспособности как отдельных элементов, так и ИТС в целом, что позволяет автоматически выявлять неполадки в работе ИТС и перераспределять ресурсы для оперативного реагирования на возникающие проблемы.

Структура агентского модуля

Структура агентского модуля СУИ приведена на рис.5.

Рис. 5. Структура агентского модуля

Агентский модуль состоит из исполнительной, сенсорной, элементной частей и блока взаимодействия.

Исполнительная часть обеспечивает штатное функционирование агента, осуществляет управление агентом, планирование действий, вызов функций мониторинга и состоит из блоков мониторинга, управления, планирования и обработки информации.

Блок мониторинга обеспечивает мониторинг ОМУ, находящихся в зоне ответственности агента. Мониторинг может осуществляться стандартными методами с использованием протоколов SNMP или ICMP и инструментария WMI, или с помощью предложенной авторами и реализованной в предлагаемой СУИ технологии DALLF (Dynamic Auto Link Library Function), использующей рефлексию типов. Стандартные средства мониторинга агенты СУИ используют для получения информации с локального элемента ИТС, на который установлен агент, и, при наличии доверительных отношений, с удаленных ОМУ. Расширяемые методы мониторинга, представленные в виде DALL (Dynamic Auto Link Library) функций, выбираются администратором для каждого агента из библиотеки шаблонов поведения. Каждый элемент шаблона соответствует определенному типу агента в зависимости от возложенных на него функций и состоит из набора DALL функций.

Исполнительная часть обеспечивает управление агентом, планирование выполнения функций мониторинга и обработку информации, которая поступает от сенсорной части. При этом контролируются показатели качества функционирования отдельного элемента ИТС, на котором расположен агент и, в случае выхода их за критические пределы, производится локализация потенциальной неисправности. Блок управления обеспечивает работу с деревом ОМУ, к которому привязан агент, и может устранить возникшую, или потенциальную неисправность при наличии необходимых для устранения неисправности алгоритмов, о чем информируется сервер СУИ, который может скорректировать принятые агентским блоком управления решения. При отсутствии необходимых для устранения неисправности алгоритмов, агент уведомляет сервер СУИ о локализированной неисправности, и сервер принимает решение по ее устранению. При недоступности сервера агент принимает решения по устранению неисправностей самостоятельно.

Блок планирования обеспечивает контроль запускаемых функций проверки, обеспечивает непрерывный процесс мониторинга и распределяет ресурсы между всеми решаемыми агентом задачами. Кроме того, для реализации механизмов пассивного мониторинга, можно подписывать DALL функции на внешние системные события.

Блок обработки информации осуществляет обработку данных, поступающих с сенсорной части и от блока мониторинга, и путем статистического анализа выявляет закономерности изменения значений измеряемых параметров и определяет соответствие состояний ОМУ предопределенным политикам, что необходимо при использовании децентрализованного управления.

Элементная часть состоит из дерева ОМУ, которые закреплены за агентом. Состояние каждого ОМУ оценивается по ряду параметров. Блок управления передает ОМУ управляющие команды для поддержания контролируемых параметров в заданных пределах. Расчетные значения параметров поступают на сенсорную часть.

Сенсорная часть состоит из набора датчиков, счетчиков, указателей и пр., которые собирают данные для определения состояний ОМУ, контролируемых агентом, и передает полученные данные в блок обработки информации.

Блок взаимодействия осуществляет информационный обмен между агентами и агентов с сервером. Связь между агентскими модулями может осуществляться напрямую, при установлении доверительных отношений между агентами, или через канал связи агентов в противном случае (см. рис 6.).

Рис. 6. Связи между агентами

Канал связи агентов обеспечивает связь между агентами, когда между ними нет прямых доверительных отношений. При этом создается новое сообщение для передачи, которое поступает в стек сообщений, где обрабатывается посредством планировщика. Каждое сообщение имеет свой уровень важности от 0 до 9 (0 – наивысший уровень). Сообщения с уровнем важности 0 всегда передаются напрямую неза-

висимо от уровня доверенности между агентами.

Каждый программный агент СУИ имеет определенный жизненный цикл, на протяжении которого он исполняет возложенные на него функции, и может находиться в одном из следующих состояний (см. рис. 7): инициализации (идентификатор: 0), активности (идентификатор: 1), перемещения (идентификатор: 2), ожидания (идентификатор: 3), паузы (идентификатор: 4) и уничтожения (идентификатор: 5).

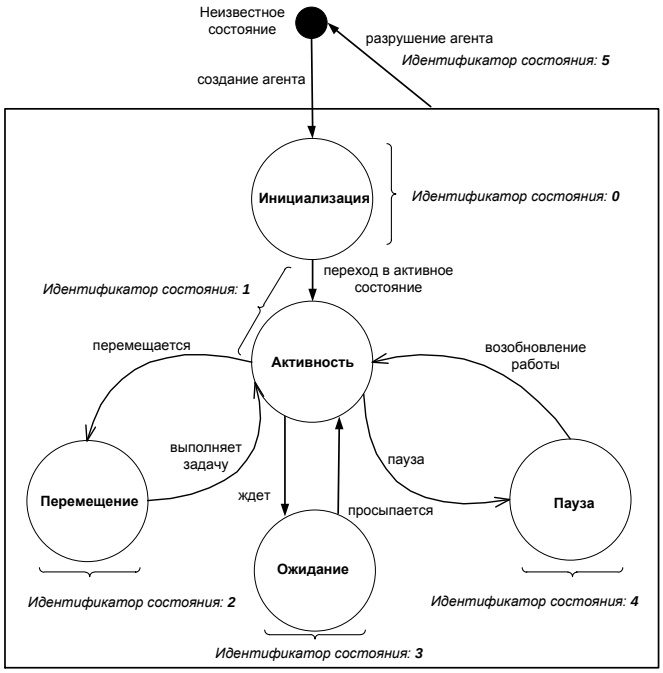

Рис. 7. Жизненный цикл агента

Первоначально программный агент находится в состоянии инициализации, в котором происходит инициализация набора функций самоконтроля, задание параметров для планировщика и загрузка дерева ОМУ. После завершения инициализации агент переходит в состояние активности. В этом состоянии агент запускает планировщик и работает с деревом ОМУ, анализирует состояние всех составных элементов и может производить начальный анализ полученных данных. На данном этапе главной задачей агента является контроль состояний ОМУ и передача необходимой информации на сервер. Если за некоторый промежуток времени не происходит изменения параметров подконтрольных ОМУ, то агент переходит в состояние ожидания.

Главной задачей планировщика становится проверка возобновления активности подконтрольных ОМУ. Если долгое время параметры объектов не изменяются, то по указанию сервера агент может переместиться на другую машину, на которой ОМУ возобновили свою деятельность или на котором повысилась расчетная нагрузка. В этом случае агент переходит в состояние с идентификатором 2 – состояние перемещения и прекращает активность, для чего приостанавливает внутренний планировщик и сворачивает активность дерева ОМУ. Сервер СУИ готовит туннель для перемещения агента и служебным сообщением передает его идентификатор. По идентификатору туннеля блок управления агента находит нужный путь перемещения и мигрирует к заданному месторасположению в ИТС.

Если услуги отдельно взятого агента перестают быть полезными для СУИ, то сервер может принять решение о приостановке работы агента (идентификатор: 4) или разрушить его.

Особенности реализации распределенных агентов

В СУИ использован подход к распределению ОМУ по агентам, при котором все ОМУ закрепляются за агентами в зависимости от производительности агентских машин и пропускной способности линий связи между ними. Логические ОМУ могут обрабатываться на агентах, физическое расположение которых может отличаться от физического расположения ОМУ. К конкретным агентам привязываются только ОМУ, управляющие и мониторинговые функции которых выполняются непосредственно на целевой машине, остальные ОМУ могут обслуживаться любыми агентами. Это позволяет получать структуры, которые просто изменяются и гибко настраиваются.

Такой подход позволяет решить проблему поддержки слабых машин, которые могут не обладать достаточной производительностью и быстродействием для анализа всего дерева находящихся на них ОМУ, а также проблему безопасности, когда возникает необходимость скрыть от агента структуру дерева ОМУ или отдельные его элементы.

Агенты могут быть вынесены за пределы, например, сервера приложения, с тем, чтобы не создавать процессору сервера дополнительную нагрузку решением задач мониторинга и анализа ОМУ. Реализация предлагаемого подхода позволяет также разделить серверный агент на несколько физических машин, что позволяет существенно разгрузить сервер, распараллелить обслуживание возложенных на него ОМУ, и тем самым повысить надежность СУИ.

Упрощенный пример распределения дерева ОМУ по агентам отображен на рис. 8, где тон-

кий агент обслуживает только те ОМУ, для обслуживания которых необходимо их присутствие на целевой машине, толстый агент обслуживает все логические ОМУ своей машины, региональный серверный агент обслуживает ОМУ своей сети, главный серверный агент обслуживает ОМУ всей системы.

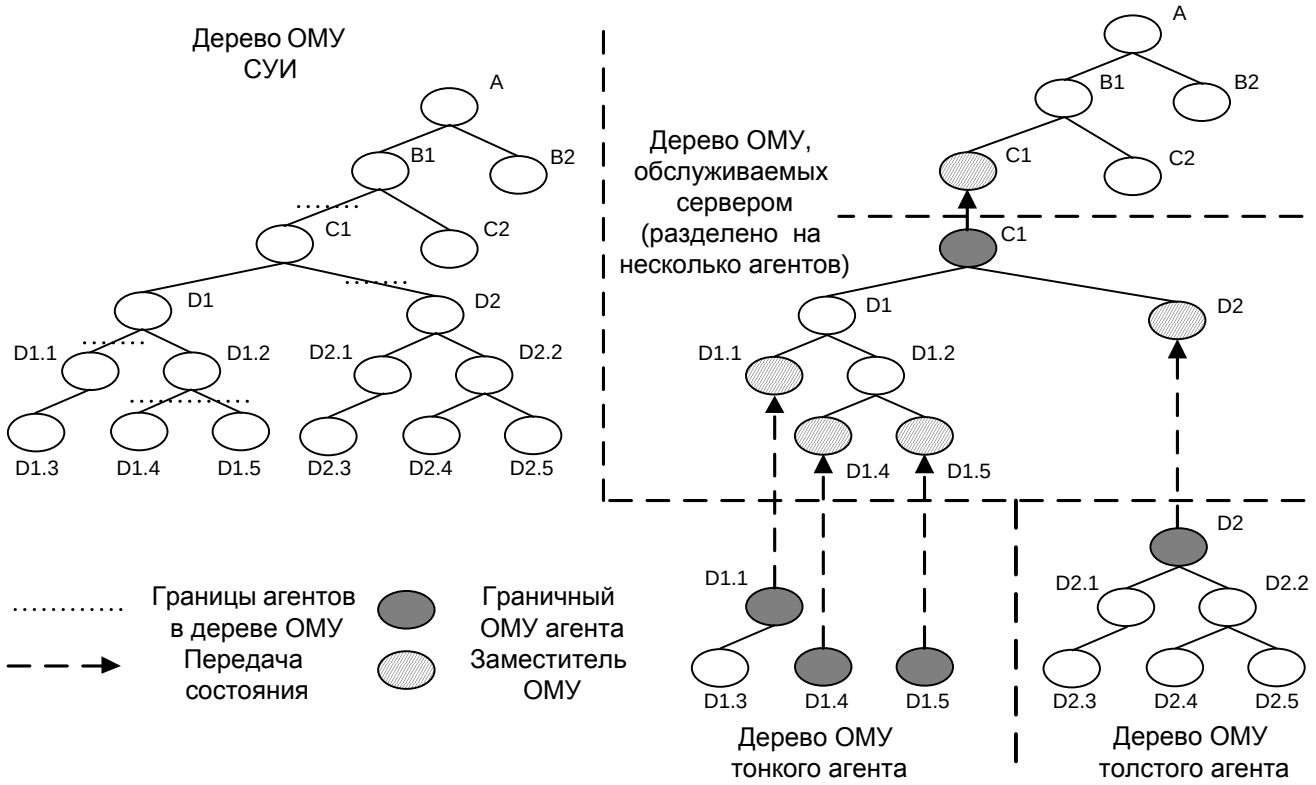

Рис. 8. Пример распределения дерева ОМУ по агентам

Предлагаемый подход позволяет легко реализовать толстые, тонкие и мигрирующие агенты, применяя один и тот же программный модуль, но по-разному задавая граничные ОМУ. Тонкий агент занимается только съемом данных мониторинга с целевой машины и выполнением управляющих функций на целевой машине, а анализ и принятие решений производятся на сервере СУИ. В мигрирующих агентах граница разделения в дереве ОМУ перемещается во время работы СУИ, перенося группы ОМУ с одной машины на другую.

Для реализации предложенного подхода к распределению ОМУ по агентам необходимо, чтобы каждый конкретный ОМУ, вне зависимости от его физического расположения, логически находился (обрабатывался) только на одном агенте, и в то же время обеспечивалась оперативная передача информационных и управляющих сигналов между граничными ОМУ разных агентов. Для этого введены понятия реального ОМУ и его заместителя (копии) соответственно для ОМУ, которые обрабатываются на данном агенте, и тех, которые необходимы на данном агенте для обеспечения взаимодействия с реальными ОМУ. Объекты-заместители, используя встроенный механизм переадресации вызовов, передают запросы реальному объекту, а также синхронизируют свое состояние в соответствии с принятыми событиями.

Программное представление объектов мониторинга и управления

С точки зрения мониторинга, оценки состояния и управления ОМУ, ИТС удобно представлять в виде дерева, узлами которого являются ОМУ, а связи отображают функциональные зависимости. Свойства, наиболее значимые с точки зрения мониторинга и оценки элементов ИТ-инфраструктуры, составляют набор параметров ОМУ.

В дереве ОМУ могут присутствовать одинаковые или схожие ветви. При построении дерева администратор тратит время на дублирование информации, поэтому возникает необходимость стандартизированной обработки однотипных ОМУ. Для этого в предлагаемой СУИ гомогенные ОМУ объединены в шаблоны, которые определяют набор параметров, функций и связей ОМУ данного шаблона. Механизм шаблонов позволяет настроить поведение и структуру шаблонного ОМУ и при создании

новых ОМУ использовать настройки шаблона, при необходимости уточняя и дополняя их. При этом можно легко менять поведение и структуру группы ОМУ путем изменения настроек их шаблона.

Ряд механизмов взаимодействия модулей СУИ инкапсулированы непосредственно в программном отображении ОМУ. Это значительно упрощает работу с объектами в различных программных модулях и позволяет взаимодействовать с удаленными объектами посредством их локальных копий-заместителей, и производить описанные ниже действия без использования отдельных блоков на каждом из программных компонентов СУИ.

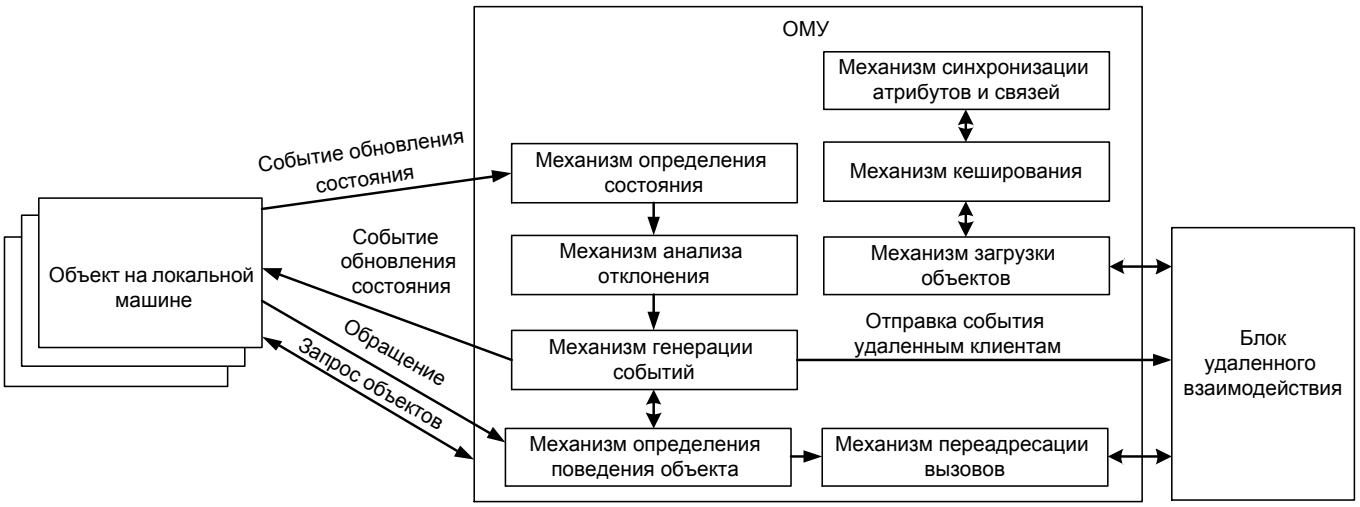

Рис. 9. Инкапсулированные в ОМУ механизмы

В программном отображении ОМУ инкапсулированы следующие механизмы (рис. 9):

— механизм синхронизации атрибутов и связей – осуществляет синхронизацию текущей версии ОМУ с обновленной, полученной с сервера, либо наоборот, сохраняемой клиентом;

— механизм кэширования – обеспечивает хранение используемых на данной машине ОМУ и наличие не более одной копии объекта в системе. В случае получения новой версии уже существующего объекта происходит его синхронизация;

— механизм определения поведения объекта – определяет, является ли копия объекта на данной машине реальной (оперирующей с ОМУ напрямую) или заместителем (переадресовывающим команды на реальный удаленный ОМУ в зависимости от истинного расположения отображаемого им ОМУ), и обеспечивает в системе наличие не более одной реальной копии каждого объекта;

— механизм определения состояния – определяет текущее состояние ОМУ в зависимости от значений мониторинговых показаний или состояний влияющих на него ОМУ, используя различные передаточные функции;

— механизм анализа отклонения – принимает решение о необходимости уведомления зависимых ОМУ об изменении своего состояния;

— механизм генерации событий – сообщает зависимым ОМУ-подписчикам об изменении своего состояния;

— механизм переадресации вызовов – для объектов-заместителей переадресует вызов управляющих функций на сервер, где определяется местонахождение реального объекта;

— механизм загрузки объектов – динамически подгружает необходимые объекты, связанные с данным, используя в качестве источника локальную или удаленную (серверную) базу данных.

Механизм реактивного обновления состояний ОМУ

В ИТС состояние ОМУ N-го иерархического уровня зависит от состояния ОМУ (N-1)-уровня, поэтому возникает задача определения необходимой частоты обновления состояний всех ОМУ.

Если осуществлять пересчет состояний ОМУ по указаниям планировщика, то каждому ОМУ N-уровня необходимо задавать период обновления, а при пересчете использовать текущие значения состояний ОМУ (N-1)-уровня, частота обновления которых может отличаться от частоты опроса ОМУ N-уровня. Недостатки такого метода очевидны: сложность синхронизации обновления сложных иерархий во времени, достаточно медленное реагирование на измене-

ние состояния ОМУ (N-1)-уровня и использование устаревших значений, при низкой частоте опроса, либо излишне высокая требовательность к вычислительным ресурсам при частом опросе.

Реализованный в СУИ механизм обновления построен на принципах реактивного программирования. Планировщик управляет периодом обновления состояний только для ОМУ нижних уровней. ОМУ верхних уровней реагируют на поток событий изменением состояний и не нуждаются в использовании планировщика. При обновлении состояния ОМУ (N-1)-уровня генерирует событие, на уведомление о котором подписываются зависящие от него ОМУ N-уровня. Механизм анализа отклонения ОМУ обладает свойством гистерезиса, что предотвращает генерацию события и излишне частые срабатывания, а также обеспечивает остановку волны перерасчета состояний ОМУ вверх по дереву, если состояние иерархически нижерасположенных ОМУ изменилось незначительно. Кроме того, для пассивного мониторинга в СУИ реализована возможность подписки функций ОМУ на внешние системные события.

Механизм реактивного обновления обеспечивает своевременное обновление состояний всех ОМУ, уменьшает количество избыточных пересчетов состояния верхних уровней иерархии ОМУ, позволяет гибко настраивать чувствительность системы к изменениям состояний ОМУ.

Выводы

Предложенная система управления корпоративной информационно-телекоммуникационной инфраструктурой реализована на основе агентского подхода с интеграцией методов активного и пассивного мониторинга, и консолидацией методов централизованного и децентрализованного управления.

Предложены структуры серверного и агентского модулей СУИ. Рассмотрены особенности реализации мобильных агентов. Разработана схема взаимодействия серверного модуля с агентским и клиентским модулями. Реализованы механизмы поддержания на всех программных компонентах системы актуальных версий объектов мониторинга и управления, позволяющие оперативно отслеживать их состояние и удаленно взаимодействовать. В системе использован реактивный подход обновления состояний ОМУ, что обеспечивает своевременное обновление состояний всех ОМУ, снижает количество избыточных пересчетов состояния верхних уровней иерархии ОМУ. Функциональность системы управления может легко наращиваться за счет использования динамических библиотек, что расширяет возможности ее применения. Благодаря разработанному подходу к распределению объектов мониторинга и управления по агентам нагрузка системы управления на вычислительные ресурсы и телекоммуникационную сеть может перераспределяться.

Предложенная система управления SmartBase ITS Control разработана в НТУУ «КПИ», прошла апробацию в двух системах специального назначения и может быть использована в различных системах управления корпоративными информационно-телекоммуникационными системами для повышения надежности и эффективности использования информационно-вычислительных и телекоммуникационных ресурсов, а также повышения результативности применения информационных технологий.

Список литературы

1. Теленик С.Ф. Система управління інформаційно-телекомунікаційною системою корпоративної АСУ / С.Ф. Теленик, О.І. Ролік, М.М. Букасов, Р.Л. Соколовський // Вісник НТУУ «КПІ». Інформатика, управління та обчислювальна техніка. – К.: «ВЕК+», – 2006. – № 45.– С. 112-126.
2. Теленик С.Ф. Система управления информационной инфраструктурой транспортного предприятия / С.Ф. Теленик, А.И. Ролик, П.Ф. Можаровский, А.В. Волошин // Автомобільний транспорт: зб. наук. праць: Вип. 25. – Харків.: ХНАДУ, 2009. – С. 242-245.
3. Ролик А.И. Управление устранением неисправностей в ИТ-системах / А.И. Ролик, Ю.С. Тимофеева, Н.И. Турский // Вісник НТУУ «КПІ». Інформатика, управління та обчислювальна техніка. – К.: «ВЕК+», 2008. – № 49. – С. 94-107.
4. Ролик А.И. Распределение мобильных компонентов системы управления информационно-телекоммуникационной системой / А.И. Ролик, Р.Л. Соколовский // Вісник НТУУ «КПІ». Інформатика, управління та обчислювальна техніка. – К.: «ВЕК+», 2007. – № 47. – С. 113-124.

5. Andreozzi S. On the Integration of Passive and Active Network Monitoring for Grid Systems / S. Andreozzi, A. Ciuffoletti, A. Ghiselli, D. Antoniades, E. P. Markatos, M. Polychronakis, P. Trimintzios // Integrated Research in GRID Computing, Springer-Verlag, January 2007. – p. 147–161.
6. Ролик А.И. Модель управления перераспределением ресурсов информационно-телекоммуникационной системы при изменении значимости бизнес-процессов// Автоматика. Автоматизация. Электротехнические комплексы и системы. – 2007. – №2 (20). – С. 73–82.
7. Ролик А.И. Система управления информационными потоками в корпоративной IP-сети / А.И. Ролик, В.А. Иосифов // Адаптивні системи автоматичного управління. Міжвідомчий науково-технічний збірник. – Дніпропетровськ: Системні технології, 2009. – Вип. 14 (34). – С. 73-85.