

УДК 004.78

Б.А. Соловьев, В.Т. Калайда, А.И. Елизаров

Компоненты системы безопасности на базе комплекса «Базис»

В работе приводится краткое описание программного комплекса построения распределенных систем «Базис» и построенной на его базе системы безопасности «Лик».

Ключевые слова: распределенная система, программный комплекс, система распознавания, система безопасности.

Интуитивно каждому человеку, имеющему дело с информационными системами, понятно, что переходя от принципов процедурного программирования к понятиям программных компонентов, мы получаем в руки инструмент, который позволит обеспечить новый уровень инкапсуляции и повторного использования кода.

Но как бы не становились более совершенными системы разработки программного обеспечения и операционные системы, в полной мере их преимуществами удастся воспользоваться только программисту. В то же время, переведя понятие программной компоненты на более высокий уровень, мы получаем возможность оперировать «атомами» кода на уровне пользователя или инженера, чья область знаний не затрагивает программирование.

При использовании общепринятого в программировании подхода, когда для решения задачи необходима определенная последовательность операций, мы всегда сталкиваемся с необходимостью составления алгоритма, в то время как при проведении эксперимента или построении системы с известными алгоритмами нас волнуют процессы, которые, как правило, протекают параллельно.

Таким образом, перед нами стоит задача обеспечить условия, при которых разработчик или экспериментатор могут перенести свое представление о решении в структуру программной системы, не вдаваясь в суть процессов, происходящих на уровне кода.

Наиболее близкими абстракциями процесса обработки информации являются модели принятые в теории управления и для создания распределенной системы необходимо пользоваться схожими понятиями. Это означает, что всю задачу необходимо декомпозировать на звенья, представленные в виде черных ящиков, соединенных последовательно и параллельно и решающих атомарные, в рамках данной прикладной области, задачи. При наличии у такого черного ящика набора входов и выходов, ассоциированных с данными, разработчик формирует решение путем составления графа решения.

Авторами ведется разработка программного комплекса «Базис» [1], предназначенного для построения распределенных систем, в основу которого заложены описанные идеи.

Так как задачи коллектива авторов строятся вокруг темы опознавания лиц, основной областью применения «Базис» рассматриваются системы безопасности.

Процесс создания комплексных систем безопасности требует интеграции в них различных подсистем, отвечающих за множество отдельных аспектов безопасности. К таким подсистемам относятся системы пожарной безопасности, охраны периметра, контроля доступа сотрудников, учета рабочего времени и т.д. Хотя интеграция этих систем является актуальной проблемой, они по-прежнему остаются лишь набором слабо связанных между собой подсистем, а иногда и неспособных работать в комплексе. Таким образом, остро стоит вопрос о создании технологии, обеспечивающей унифицированный подход к построению и управлению интегрированными системами безопасности, обеспечивающей возможность постоянного дополнения функциональности систем и изменения списка подсистем, отвечающих за новые аспекты обеспечения безопасности.

На основе изложенного авторами ведется разработка системы безопасности «Лик» [2] под управлением «Базис».

В качестве «строительного блока» распределенной системы в «Базис» используется понятие «прикладного объекта», который представляется черным ящиком и может иметь вхо-

ды и выходы для обработки и генерации информационных потоков. Наличием или отсутствием входов или выходов, определяется тип прикладного объекта:

- генератор – объект, генерирующий новые данные, в результате работы какого-либо, внешнего устройства ввода, имеет выходы, но не имеет входов;
- приемник – объект, предназначенный для окончательной обработки данных, имеет несколько входов, но не имеет выходов;
- комплексный объект – наиболее распространенный типа объекта, имеет как входы, так и выходы, что позволяет генерировать данные на основе анализа входных потоков или изменять их в соответствии со своими функциями.

Администратору задача, решаемая под управлением комплекса «Базис», представляется в виде функциональной схемы, редактируя которую, он легко может распределять отдельные блоки, убирать и добавлять новые, не обращаясь к разработчику всей «прикладной системы» (см. рис. 1). Кроме этого каждый прикладной объект имеет «окно представления» для конфигурации объекта и представления протекающих процессов.

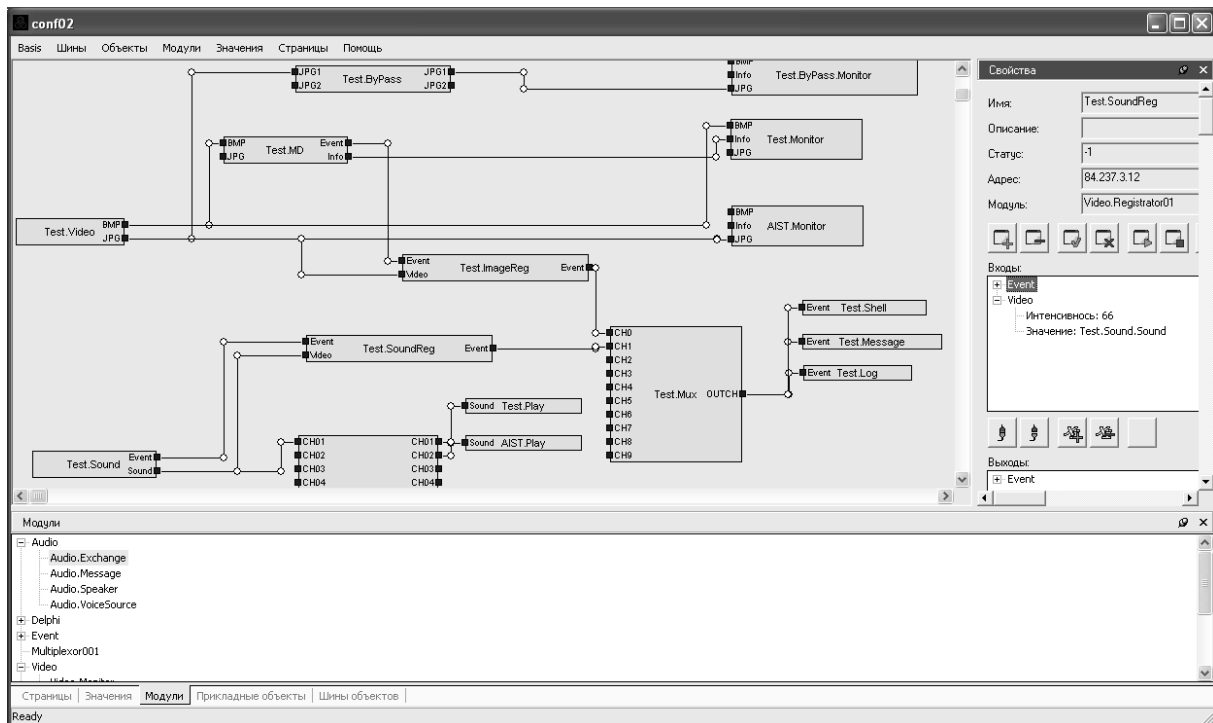


Рис. 1. Окно конфигурации комплекса «Базис»

Каждый объект имеет имя, которое несет информацию о подсистеме, к которой он принадлежит, и его функции в подсистеме. Например, объект с именем «Main building.Level 2.Room 208.Video.EnterCamera» является частью подсистемы видеонаблюдения и представляет собой модуль захвата видео с камеры, расположенной на входной двери 208-го кабинета главного здания.

Каждый прикладной объект решает атомарную в рамках системы задачу. Поэтому для решения своей задачи ему не важно, откуда поступили данные на обработку и куда будут направлены данные, полученные в результате его работы. Направление движения данных, а соответственно и функции подсистемы в целом определяются всей совокупностью прикладных объектов подсистемы и связями между ними. Для этого необходимо разделить генератор и приемник. Это происходит за счет введения понятия объекта-значения. Объект-значение получает имя, которое ему дает администратор, ассоциируя значение с выходом одного из объектов. По этому имени данные генератора становятся доступны всем объектам «Базис» независимо от того, на каком компьютере и в какой подсистеме они находятся [1, 2].

Каждый объект «Базис» представляет собой COM-сервер, реализующий следующие интерфейсы [3]:

- IBasisObject01 – интерфейс, содержащий методы прикладного объекта, является обязательным для всех прикладных объектов;
- IBasisGenerator01 – интерфейс генератора, реализуемый, соответственно, объектами-генераторами и комплексными объектами;
- IBasisReceiver01 – интерфейс объекта-приемника, реализуется приемниками и комплексными объектами.

Для организации жизненного цикла объектов и значений (создание – запуск – работа – остановка – удаление) на каждом компьютере, включенном в процесс под управлением «Базис», создается объект – *шина*, которая в последующем отвечает за объекты, расположенные на подчиненном ей компьютере.

Для организации работы системы компьютеров в виде единой системы, на одном из компьютеров располагается *менеджер конфигураций «Базис»*. Он отвечает за получение информации о системе в целом и управление жизненными циклами шин. Информация о компьютерах, объектах, значениях и связях находится в системной базе данных. Схема взаимодействия компонентов «Базис», показана на рисунке 2, общая информационная среда обеспечивается за счет отделения генератора от приемника посредством объектов-значений.

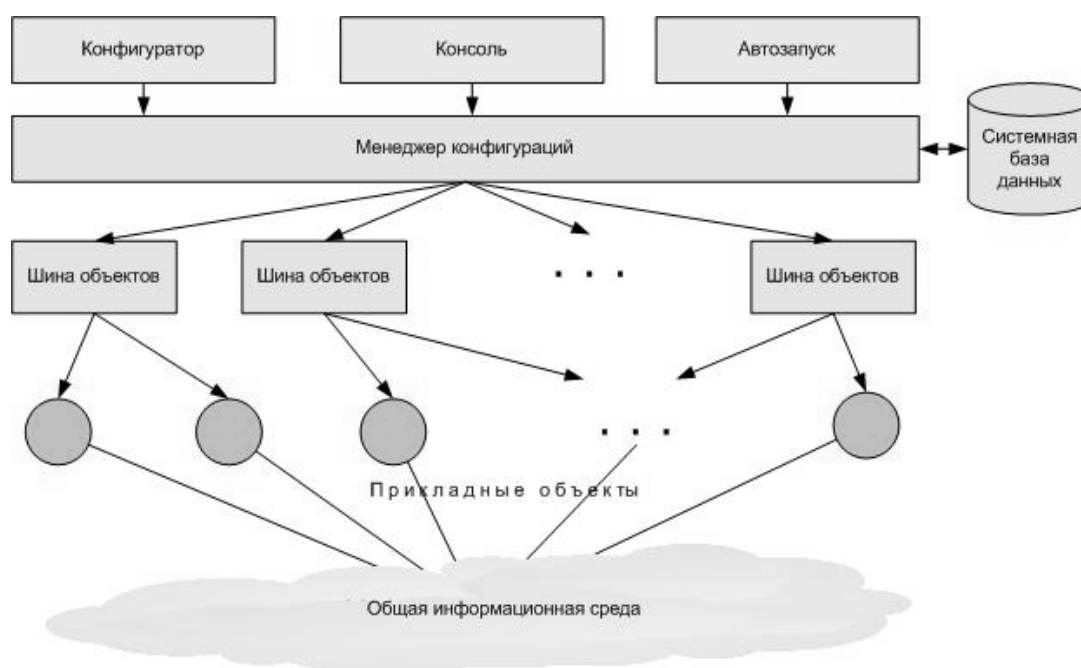


Рис. 2. Компоненты «Базис» и связи между ними

При получении команды на старт, менеджер конфигураций рассылает сигналы на старт всем «подопечным» шинам. При этом администратор может запретить запуск одной из шин по этому сигналу или один из компьютеров может быть недоступен. В случае запрета старта, администратор имеет возможность в последующем дать команду на старт/останов как для отдельной шины, так и для отдельной подсистемы, указав при этом IP- или NetBIOS-адрес компьютера или часть имени подсистемы с символом * на конце (например, start 81.236.4.105 или start MainBuilding.Level 2.*). В случае, если в момент общего запуска шина была недоступна, то при включении компьютера программа автостарта пошлет запрос на старт своей шины автоматически.

При получении команды старт каждая шина просматривает список локальных объектов и производит создание необходимых объектов-значений и запуск прикладных объектов. При подключении входов/выходов объектов им передается короткий идентификатор соответ-

вующего объекта-значения. Каждый из прикладных объектов может также иметь запрет на старт.

Для организации передачи данных объект-значение реализует несколько алгоритмов:

- проецируемые в память файлы, организованные в страничные структуры, для передачи данных локально;
- IP-мультикастинг – для быстрой передачи данных через сеть без гарантии доставки;
- передача данных всем заинтересованным по DCOM-соединению с гарантией доставки и возможностью шифрования.

Все команды и передача данных происходят асинхронно, это не дает зависнуть всей системе при зависании или выходе из строя одного из объектов. На этапе работы также не обязательно наличие менеджера конфигурации. В процессе работы он отвечает только за организацию сеансов перестройки частот генерации данных отдельных объектов, что при использовании компьютеров с удовлетворяющей задаче производительностью – не критично [4].

В процессе работы генерирующий объект передает данные ассоциированному с выходом объекту-значению, который передает информацию о номере страницы размещения и размере данных локальным объектам, использующим данное значение для получения входных данных. При наличии сетевых связей объект-значение обеспечивает доставку данных по сети в зависимости от типа связи.

Система «ЛИК» представляет собой пакет модулей и утилит для построения распределенных систем безопасности под управлением «Базис».

Основными задачами «ЛИК» являются:

- видеонаблюдение;
- контроль доступа;
- охрана территории;
- оповещение;
- ведение протокола событий.

Любая охраняемая территория подразделяется на зоны и источники данных. Описание любой зоны независимо от ее специфики содержит следующую информацию:

- уникальный идентификатор;
- имя зоны, включая имена родительских зон;
- уровень важности;
- текущее состояние;
- словесное описание.

Любой источник может быть описан его именем, уникальным идентификатором и словесным описанием.

В самом простом случае для наблюдения за охраняемой территорией и реагирования на ситуацию достаточно лишь словесного описания события или голосового сообщения с одного из постов. То есть, ключевым сигналом системы безопасности является сообщение о событии, оно является главным носителем управляющей информации в системе. По содержанию сообщения о событии система может включить сигнализацию, оповестить персонал об опасности, открыть или заблокировать дверь. Суть события и реакции на него должны поступать на дисплей охраны. Любые события, циркулирующие в системе, содержат следующие данные:

- уникальный идентификатор;
- имя события;
- зона;
- дата;
- время;
- параметры.

Параметры события могут содержать любую дополнительную информацию специфичную для события, такую как источник сигнала, важность события, идентификатор вошедшего через дверь сотрудника, присоединенные файлы и т.д.

Каждый объект «Лик» наряду с выходами, обусловленными его функциональностью, имеет выход события. Сигнал события представляет собой пару {событие, параметры}. Параметры могут включать в себя имя объекта генератора, зону генератора и т.д. и разделены

символом «;». После генератора сигнал события поступает на вход контроллера зоны, который добавляет необходимые параметры, такие как: имя зоны, важность зоны и режим тревожности зоны. После чего эта информация может быть использована подсистемой оповещения, охраны или протокола. В перечисленных подсистемах для генерации нового события или сигнализации событию сопоставляется уровень опасности по маске или по полному соответствию полей и принимается решение о дальнейших действиях.

Напрямую с оператором взаимодействует подсистема визуализации событий. Эта часть системы принимает на вход сообщение о событии, далее события фильтруются по зоне и источнику и направляются соответствующему элементу отображения события на трехмерной схеме охраняемой зоны. Подсистема отображения событий на схеме выполняется в виде отдельного прикладного объекта «Базис». Для отображения видео, аудио и пр. «неформализованных» данных используются специальные прикладные объекты, такие как объект-дисплей.

Подсистема протокола заносит в базу данных информацию о событии в том виде, в котором оно было сгенерировано с привязкой к зоне, источнику и времени генерации. В случае, когда наряду с формализованным сообщением о событии с ним ассоциируется также и неформализованный фрагмент, он отправляется в специализированное хранилище, в сообщение о событии добавляются идентификатор хранилища и ключ доступа к сохраненным данным. При просмотре события программа анализа протокола получает по этому ключу необходимую видео или аудио информацию.

Для обеспечения звуковой и видео связи используется набор источников, ориентированных на работу с конкретным устройством захвата, после чего полученный поток может быть направлен в архив или воспроизведен на одном или нескольких целевых компьютерах.

Для организации контроля доступа вводится понятие пропуска. Один пропуск может использоваться доступа в разные зоны. Пропуск может действовать в одном или нескольких режимах тревожности помещения. Пропуск содержит информацию о сроке его действия. Для пропуска может использоваться один из дежурных и дополнительных графиков допуска. Пропуск содержит информацию о количестве разрешенных повторных проходов в зону, если ведется контроль повторного прохода. Пропуск содержит уровень проверки личности и может быть связан с пропусками конвоиров.

Дежурный график содержит время разрешения или запрета на доступ принципала в течение одного дня и действует каждый день. Дополнительный график содержит информацию об исключениях таких, например, как выходные и праздничные дни.

В системе могут существовать стандартные и специальные графики, разработанные для группы принципалов или индивидуально. Существуют также графики перехода зоны в различные режимы охраны. В процессе редактирования графиков у администратора создается впечатление работы с одним.

Режимы тревожности представляют собой факт установки зоны «на сигнализацию» и могут содержать несколько уровней, семантика которых определяется администратором системы безопасности.

Правила срабатывания сигнализации и активации тревоги ассоциируются с зоной и ее состоянием. Само правило представляет собой набор строк со значениями параметров события, при котором правило активируется. При срабатывании правила сигнализация активируется путем генерации последовательности ответных событий, управляющих состоянием зоны и исполнительными механизмами.

«Лик» содержит модули «Базис» для построения следующих подсистем:

- подсистема контроля доступа;
- аудио-подсистема;
- подсистема протокола и обработки событий;
- подсистема видеонаблюдения.

Подсистема контроля доступа содержит следующие компоненты.

Модуль Ports.RS485, предназначенный для организации взаимодействия подсистем «Лик» с устройствами, подключенными к одному из компьютеров системы посредством шины RS-485. Он имеет десять входов и выходов. Клиент подключается к одному из входов для

посылки запроса устройству, для получения ответа от устройства он должен быть подключен к выходу с тем же номером.

Модуль взаимодействия с контроллером ЛИК служит для взаимодействия подсистем «ЛИК» с контроллером доступа «ЛИК», подключенным к системе по шине RS-485. Модуль имеет два входа:

- событий, для управления замками;
- данных, для приема данных от контроллера;

и два выхода: для оповещения «ЛИК» о новых ключевых данных на считывателях и данных, для посылки запроса на чтение данных контроллера.

Модуль последовательно формирует запросы на чтение состояния считывателей для каждого источника, прописанного у него в конфигурации, и отправляет их через выход данных. При наличии новых данных генерирует событие о попытке доступа в помещение. В зависимости от реакции системы генерирует управляющую последовательность для замка и светодиодов на выходе.

Модуль контроля доступа Event.SPA служит для проверки состояния зоны, разрешения на вход и условий возникновения тревоги. Модуль имеет один вход и один выход для приема и генерации событий. По приходу события модуль сначала определяет состояние зоны, в которой произошло событие, затем, если событие – попытка прохода, то определяется состояние пропуска данного пользователя в данную зону. Вся информация о состоянии зоны и разрешении/запрете доступа помещается в параметры события. На заключительной стадии происходит проверка условия активации тревоги по содержимому сгенерированной строки события.

В составе аудио-подсистемы на данный момент присутствуют следующие компоненты.

Модуль снятия звуковых данных Audio.VoiceSource предназначен для захвата звука со входа звуковой карты и передаче его дальше для обработки системе «ЛИК». Модуль имеет два выхода: для событий и звуковых данных.

При превышении заданного порога чувствительности модуль начинает передавать звуковые данные на выход, при этом раз в секунду на выходе Sound генерируется сообщение о том, что передаются данные.

Для воспроизведения звука на удаленной машине используется модуль Audio.Speaker. Модуль имеет только один звуковой вход. На машине может располагаться сколько угодно объектов, «порожденных» от этого модуля.

Для коммутации аудио-поток, пришедших из разных мест и предназначенных для воспроизведения на разных машинах, служит модуль Audio.Exchange. Он имеет десять входов и выходов. Выбранный входной поток коммутируются на все выбранные выходные.

В ряде случаев, например при возникновении тревоги, необходимо воспроизвести заранее записанное звуковое сообщение. Для этих целей служит модуль Audio.Message. Модуль имеет один вход – для приема сообщения. К пришедшему событию применяется фильтр, при истинности условия фильтра происходит воспроизведение сообщения в указанном файле. Если сообщение удовлетворяет условиям нескольких фильтров, происходит воспроизведение всех файлов последовательно. Параметр «тишина» указывает на время в миллисекундах, в течение которого повторного реагирования на событие не происходит. При редактировании фильтра предусмотрены следующие условия:

- = – условие равенства для строк и целых;
- != – условие неравенства для строк и целых;
- Есть – условие присутствия параметра в строке сообщения;
- Нет – условие отсутствия параметра в строке сообщения;
- > – больше для строк и целых;
- < – меньше для строк и целых.

Подсистема протокола и обработки событий содержит следующие модули.

Модуль Event.Log служит для записи всех событий, приходящих на его вход в системную базу событий. При записи события в базу происходит индексирование записей по времени возникновения события, идентификатору типа события, порожденного событием тревоге и идентификатору зоны.

Модуль коммутации потоков в один Event.Multiplexor001 имеет 10 входов и 1 выход типа Event. С каждым входом может ассоциироваться зона и источник; в случае отсутствия этой информации в событии модуль коммутации добавляет ее автоматически.

Модуль Visual.Shell организует канал передачи сообщений о событиях, происходящих в «ЛИК», в программу визуализации событий на схеме. Имеет один вход для событий. События передаются без изменений, и вся обработка лежит на программе визуализации.

Для построения подсистемы видеонаблюдения в «Лик» предусмотрены следующие модули.

Для захвата видео с IP камер служит модуль Video.IPCamera. Он имеет два выхода: кадр в формате JPG и кадр в формате BMP. По выходу BMP изображение передается несколькими частями, в зависимости от размера изображения.

Модуль отображения видео отображает входящий видео-поток с дополнительной информацией о потоке. Имеет входы: Info – информации о потоке в виде строки, содержащей описание графических примитивов, BMP – изображение в bmp и JPG – изображение в jpg. Графические примитивы записываются в строчку через точку с запятой. Поддерживаются следующие команды:

rgb(<красный>,<зеленый>,<синий>) – выбрать текущий цвет примитива;

line(<начало по X>,<конец по X>,<начало по Y>,<конец по Y>) – начертить линию;

rect(<начало по X>,<конец по X>,<начало по Y>,<конец по Y>) – начертить прямоугольник;

text(<X>,<Y>,<Текст>) – вывести текст.

Для анализа видео-потока на движение и оставленные предметы используется модуль Video.MotionDetector. Он имеет два входа для принятия видео: JPG и BMP и два выхода: Info – для передачи строки графических примитивов, обозначающих зону срабатывания и Event – для генерации события – движения/оставленных предметов

В модуле предусмотрено три типа чувствительности зон экрана: основная, дополнительная и зона нечувствительности. Для редактирования одной из зон необходимо выбрать ее название в поле «Текущая» панели «Зоны чувствительности» и нажать кнопку «Показать». На изображении появится маска, редактирование которой происходит нажатием правой для добавления или левой для исключения элемента из зоны.

При обнаружении движущегося объекта, он обозначается на изображении прямоугольником. В случае, если объект находится в поле видимости дольше, чем указанное время, его прямоугольник имеет зеленый цвет, в противном случае – красным. Если объект не двигается и находится в поле видимости в течение оговоренного времени, его прямоугольник имеет синий цвет.

Модуль Video.Registrator01 сохраняет в хранилище неформализованных данных входящие пакеты по приходу данных на вход Event. В текст события автоматически добавляется параметр, указывающий тип данных и место хранения. Имеет два входа: Event-событие и Video-данные. На вход Video могут подключаться как видео-, так и аудио-потоки.

Для захвата видео-потока с одного из адаптеров компьютера используется модуль Video.VideoSource. Он имеет два выхода для передачи в «ЛИК» кадров видео-потока: BMP и JPG.

Модуль захвата лица Application.FaceDactTry выполнен в виде «приложения Базис». Имеет вход JPG1 для приема входного кадра. И выход JPG2 для генерации выходных изображений – фрагментов входного, предположительно содержащих лица.

Для организации работы системы «Лик» как целого в нее включены следующие вспомогательные приложения:

- редактор учетных записей, позволяет создавать пользователей и группы, назначать пользователям роли в каждой из групп и создавать пропуска;
- редактор тревог, позволяет создавать правила активации тревог;
- редактор событий, служит для внесения имен событий, которые могут возникать в системе;
- редактор состояний, позволяет задать все возможные состояния зон;

- редактор зон, позволяет создать иерархию зон, назначить им графики состояний и указать установленные в зоне датчики и исполнительные механизмы;
- редактор расписаний, позволяет создавать расписания действия пропусков сотрудников и изменения состояния зон. При вызове этой утилиты из редактора зоны или учетных записей, в главном окне добавляется окно исключений для зоны или пользователя, при этом все расписание отображается в виде единого графика.

Заключение

Разработанная модель взаимодействия по схеме объект-генератор, объект-приемник и объект-посредник для доставки представляет собой наиболее близкий вариант взаимодействия процессов в реальных систем. Работа каждого из вовлеченных в процесс компонентов, в отличие от взаимодействия по схеме клиент-сервер, не направлена на взаимодействие с заранее оговоренным кругом компонентов. Результаты работы одного из генераторов становятся доступны всем заинтересованным в этом результате компонентам, что существенно повышает гибкость программной системы как на этапе разработки, так и во время исполнения. Возможность взаимодействия через объект-посредник позволяет исключить монопольное использование устройств каким-либо из компонентов, в системе появляется возможность в любой момент добавить новые подсистемы обработки информации. В настоящее время ведется работа по созданию набора компонентов «Базис» для управления физико-техническими установками при проведении физических экспериментов.

Литература

1. Соловьев Б.А., Калайда В.Т. Базовое программное обеспечение интегрированных распределенных систем безопасности // Информационные технологии. – 2006. – №1. – С. 43–59.
2. Соловьев Б.А., Калайда В.Т., Елизаров А.И. Распределенная система безопасности «ЛИК» // Известия Том. политех. ун-та. – 2008. – Т.313, №5. – С. 110–116.
3. Соловьев Б.А. Технология создания распределенной системы видеонаблюдения на основе SOM/DCOM // Электронный журнал «Вычислительные сети. Теория и практика». – 2004. – №1(4) [Электронный ресурс]. – Режим доступа : <http://network-journal.mpei.ac.ru/cgi-bin/main.pl?l=ru&n=4&pa=13&ar=11>.
4. Соловьев Б.А., Калайда В.Т. Моделирование работы распределенной системы видеонаблюдения // Доклады ТУСУР. Автоматизированные системы обработки информации, управления и проектирования. – 2005. – №3. – С. 88–93.

Соловьев Борис Александрович

Канд. техн. наук, с.н.с. Института оптики атмосферы СО РАН, Томск
Тел.: (3822) 49-22-42
Эл. почта: sol@iao.ru

Калайда Владимир Тимофеевич

Доктор техн. наук, профессор ТУСУРа
Тел.: (3822) 49-22-42
Эл. почта: kvt@iao.ru

Елизаров Алексей Игоревич

Канд. техн. наук, доцент ТУСУРа
Тел.: (3822) 49-22-42
Эл. почта: alex@iao.ru

B.A. Soloviov, V.T. Kalayda, A.I. Elizarov

The Components of a security system based on use of the «Basis» software complex

This paper is giving a short description of a distributed software systems building complex named «Basis» and a security system built on it named «Lik».

Key words: distributed system, software complex, recognition system, security system.