

DVR SDD

From EIVEES CCTV Dep. Community

Как кандидаты в творцы, мы должны сначала заняться хаосом. Что есть хаос? Если при данном событии X в А могут произойти всевозможные события в В и если такая независимость наблюдается повсеместно, то перед нами хаос. Если же событие X в А ограничивает определенным образом то, что может произойти в В, то между А и В возникает связь. Если событие X в А ограничивает события в В однозначно (поворачиваем выключатель - зажигается лампа) связь А и В будет детерминированной.

С. Лем "Сумма технологий"

Contents

- 1 Введение
 - 1.1 Назначение документа
 - 1.2 Назначение системы
 - 1.3 Термины и сокращения
 - 1.4 Ссылки на другие документы
 - 1.5 Общие сведения
 - 1.6 Общие требования к проектам
 - 1.6.1 Настройки в Win-версии
 - 1.6.2 Поддержка национальных языков
 - 1.6.2.1 Строки
- 2 Обзор системы
- 3 Архитектура системы
 - 3.1 Описание используемого оборудования
 - 3.2 Компоненты системы
 - 3.2.1 Распределение по компонентам
 - 3.2.2 Сторонние компоненты
- 4 Данные системы
 - 4.1 Структура БД
 - 4.1.1 Таблица System
 - 4.1.2 Таблица Computers
 - 4.1.3 Таблица Devices
 - 4.1.4 Таблица Cameras
 - 4.1.5 Таблица Frames
 - 4.2 Реестр(Windows-версия)
 - 4.2.1 Режим работы CHCSVA с БД
 - 4.2.2 Режим работы CHCSVA с реестром
 - 4.3 Формат архива
- 5 Структура компонентов
- 6 DVRClient
 - 6.1 Назначение
 - 6.2 Функции
 - 6.3 Дополнительные элементы управления
 - 6.3.1 Элемент просмотра тревожных ситуаций
 - 6.3.1.1 Входные данные
 - 6.3.1.2 Выходные данные
 - 6.4 Используемые компоненты
 - 6.5 Зависимости
 - 6.6 Интерфейсы
 - 6.7 Данные
- 7 DVRLauncher
 - 7.1 Задачи
- 8 DVREngine
 - 8.1 Назначение
 - 8.2 Функции серверной части
 - 8.3 Функции клиентской части
 - 8.4 Используемые компоненты
 - 8.4.1 DVRDBBridge
 - 8.4.2 DVRAIgo
 - 8.4.3 DVRBackup
 - 8.5 Зависимости
 - 8.6 Интерфейсы
 - 8.6.1 DVRLauncher
 - 8.7 Данные
- 9 CHCSVA
 - 9.1 Задачи
- 10 DVRBackup
 - 10.1 Назначение
 - 10.2 Функции
- 11 DVRAIgo

- 11.1 Алгоритмы на уровне камеры
 - 11.1.1 Входные данные
 - 11.1.2 Задачи
 - 11.2 Алгоритмы на уровне системы
 - 11.2.1 Входные данные
 - 11.2.2 Задачи
 - 12 DVRRender
 - 12.1 Функциональные требования
 - 12.1.1 Нарисовать текст
 - 12.1.2 Нарисовать прямоугольник
 - 12.1.3 Нарисовать закрашенный прямоугольник
 - 12.1.4 Нарисовать растровое изображение
 - 12.1.5 Нарисовать кадр видео реального времени
 - 12.1.5.1 Альтернативная последовательность "Сбой при инициализации видеоподсистемы"
 - 12.1.5.2 Альтернативная последовательность "Функция прорисовки не установлена клиентом"
 - 12.1.5.3 Альтернативная последовательность "Сбой при запросе видеопотока"
 - 12.1.6 Нарисовать кадр видеоархива
 - 12.2 Пройграть архив
 - 12.3 Нефункциональные требования
 - 12.3.1 Производительность
 - 12.3.2 Надежность и доступность
 - 12.3.3 Обработка ошибок
 - 12.3.4 Интерфейсные требования
 - 12.4 Диаграмма предметной области
 - 12.5 Последовательности действий в вариантах использования
 - 12.5.1 Нарисовать кадр видео реального времени
 - 12.5.2 Нарисовать кадр видеоархива
 - 12.5.3 Пройграть архив
 - 12.6 Работа архива
 - 12.6.1 Состояния архива
 - 12.6.2 Алгоритм получения текущего времени и синхронизации архива между камерами
 - 13 DVRDBBridge
 - 13.1 Задачи для CHCSVA
 - 13.2 Задачи для серверной части
 - 13.3 Задачи для клиентской части
 - 13.4 Задачи для DVRBackup
 - 13.5 Режимы работы
 - 14 Пользовательский интерфейс
 - 15 Изменения и дополнения

Введение

Назначение документа

Документация следующих элементов проекта DVR

1. Модули проекта
 1. Задачи
 2. Зависимости между ними
 3. Предоставляемые интерфейсы
2. Внешние данные, используемые в проекте

Назначение системы

Термины и сокращения

Ссылки на другие документы

№	Путь	Комментарии
1	DVR.mdl	UML-диаграммы DVR
2	ISR\Interface_dvr.doc	Аналоги и существующие DVR
3	ISR\InterfaceDVR_vr3.xls	Примерный вид DVR GUI v3
4	ISR\InterfaceDVR_vr4_1.xls	Примерный вид DVR GUI v4
5	ISR\ Описание функций удаленного просмотра.doc	
6	ISR\ Функции системы ЦБ3.doc	
7	Own\ DVR - функциональная спецификация.doc	Прототип GUI клиента
8	Own\ DVR UI.vsd	Visio-диаграммы GUI клиента
9	Own\DVR UI.htm	Экспортированная в html DVR UI.vsd
10	DVR T3.doc	Техническое задание (предварительный вариант)

Все пути относительно директории проекта в SVN

Общие сведения

Общие требования к проектам

Согласно документу Knowledge base, если не оговорено другое.

Настройки в Win-версии

1. Для всех проектов существуют две обязательные конфигурации – Debug(отладочная версия) и Release(релиз). Проекты должны компилироваться и работать без ошибок в обеих указанных конфигурациях.
2. Каждый проект должен иметь файл *.sln с установленными зависимостями от других проектов.
3. Результирующие DLL и EXE файлы должны в результате компиляции появляться в каталоге BIN\Debug(для отладочной версии) и BIN\Release (для релиза) согласно структуре SVN (следует использовать только относительные пути).
4. Character Set = MBCS.

Поддержка национальных языков

Строки

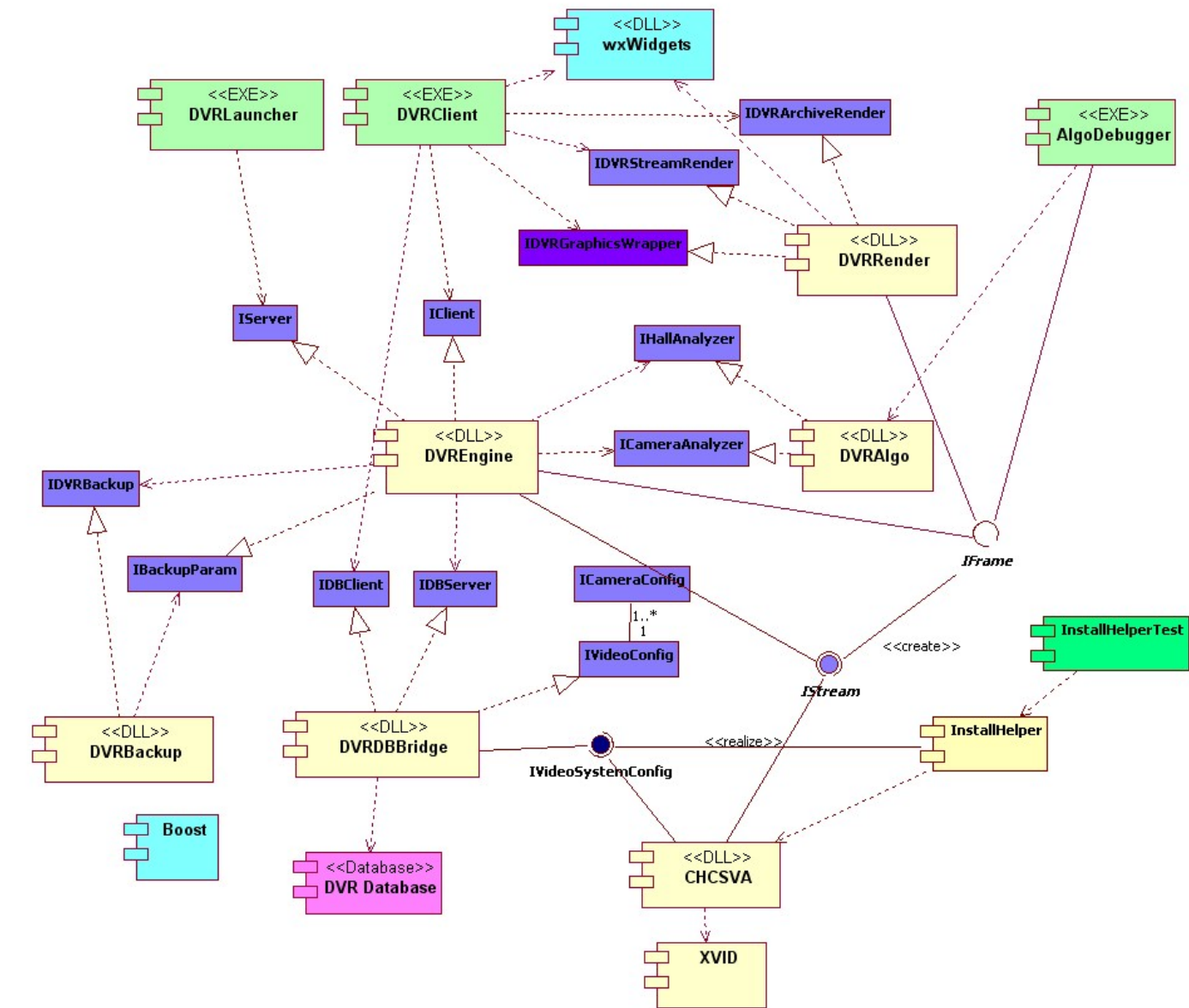
Для поддержки строк на различных языках для разных платформ (Win, *Nix) используется библиотеки gettext и iconv.

Обзор системы

Архитектура системы

Описание используемого оборудования

Компоненты системы



Распределение по компонентам

№	Ф.И.
1	Горбачев Е.
2	Ахриев А.
3	Болтнев А.

Название	Исполнитель	Примечание
DVRClient	1	Клиентское приложение системы
DVRRender	1	Компонент отображения графики
DVREngine	1	Серверная обработка
DVRLauncher	1	Серверное приложение системы
InstallHelper	1	Функции установки DVR - показ видео с камер, создание конфигурации в БД
CHCSVA	1	Развитие проекта chesva из CinemaSec
DVRDBBridge	1	Компонент работы с БД
DVRBackup	1	Компонент удаления архива и данных из БД
DVRAlgo	2, 3	Алгоритмы детектирования
AlgoDebugger	2, 3	Отладчик алгоритмов (в рабочей системе не используется)

Сторонние компоненты

Название	Версия	Назначение	URL
wxWidgets	2.7.0	Кроссплатформенная GUI-библиотека	http://www.wxwindows.org/
XviD	1.1.0	MPEG-4 видеокодек	http://xvid.org/
Boost	1.33.1		http://www.boost.org/

Данные системы

Структура БД

Структура БД в случае использования конфигурации системы полностью на БД

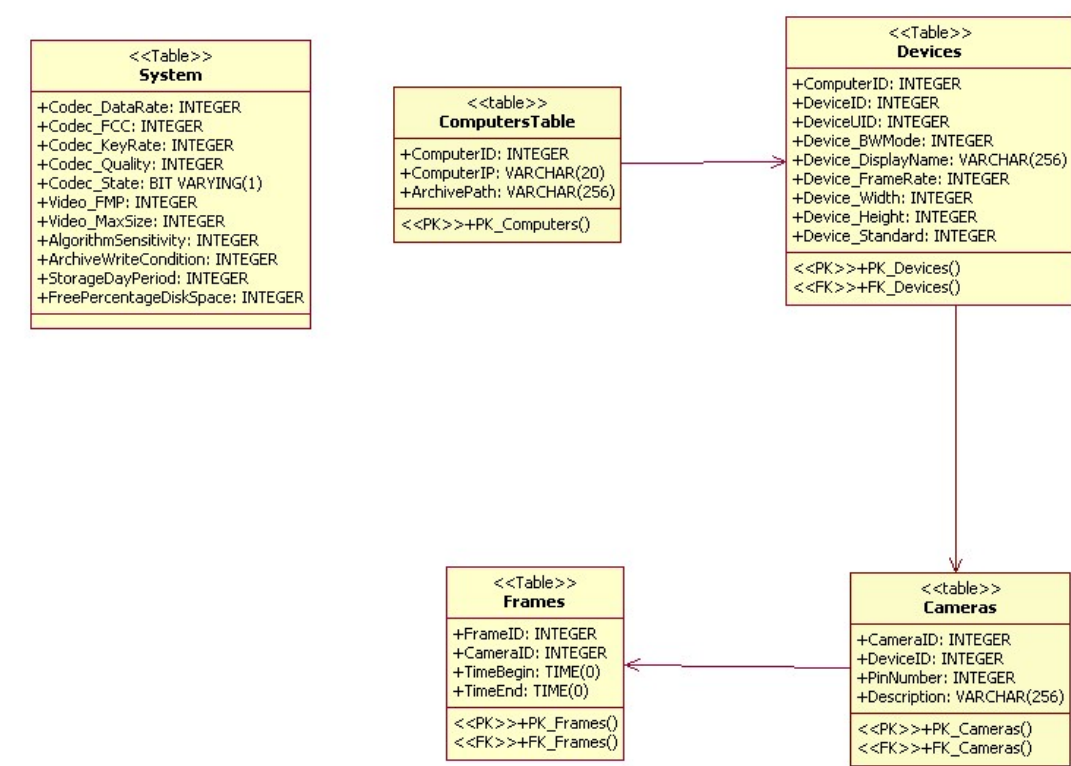


Таблица System

Системные настройки (кодек, алгоритмы детекции движения). Таблице должна чувствовать только одна запись - опции на всю систему.

Название	Тип	Значение по умолчанию	Назначение
Codec_DataRate	int	0	Параметр chcsva
Codec_FCC	int	1145656920('X','V','T','D')	Параметр chcsva (FourCC кода)
Codec_KeyRate	int	0	Параметр chcsva
Codec_Quality	int	10000	Параметр chcsva
Codec_State	blob	NULL	Параметр chcsva
Video_FMP	int	120	Параметр chcsva, frame per minutes частота захвата кадров камеры
Video_MaxSize	int	300	Параметр chcsva, размер видеофайла архива в сек
AlgorithmSensitivity	int	50	Величина чувствительности алгоритмов детекции, 0 - 100
ArchiveWriteCondition	int	0	Условие записи архива (0 - всегда, 1 - только по движению, 2 - никогда)
FreePercentageDiskSpace	int	15	кол-во минимального свободного места на диске, куда пишется архив (в % от общей емкости раздела)
StorageDayPeriod	int	21	кол-во дней хранения видеоархива архива с текущего момента времени

Таблица Computers

Параметры компьютеров в системе

Название	Тип	Значение по умолчанию	Назначение
ComputerID	int		Идентификатор компьютера (PK)[AUTOINC]
ComputerIP	string		IP адрес
ArchivePath	string		Локальная директория, в которой хранится видеоархив

Таблица Devices

Параметры входов на платах захвата, подключенные к компьютерам. Поскольку одна плата захвата может содержать несколько выводов (Orwell4L), то она может участвовать в системе как несколько устройств захвата

Название	Тип	Значение по умолчанию	Назначение
DeviceID	int		Идентификатор платы захвата в системе (PK)[AUTOINC]
ComputerID	int		Идентификатор компьютера, в котором установлена плата захвата (FK)
DeviceUID	string		Уникальный строковый идентификатор (GUID), пока для совместимости с видеоподсистемой
Device_BWMode	int	0	Цветной(0) или черно-белый(1) режим захвата
Device_DisplayName	string		Параметр chcsva, device moniker для DirectShow
Device_FrameRate	int	2	fps захвата
Device_Width	int	352	ширина изображения захвата
Device_Height	int	288	высота изображения захвата
Device_Standard	int	16	Параметр chcsva, видеостандарт (PAL, NTSC, ...)

Таблица Cameras

Парметры камеры или входа на платы захвата

Название	Тип	Назначение
CameraID	int	Идентификатор камеры (PK) [AUTOINC]
DeviceID	int	Идентификатор платы захвата(FK)
PinNumber	int	Номер пина для одного вывода на плате
Description	string	Пользовательское имя камеры

Таблица Frames

Периоды времени по каждой камере, где наблюдалось движение. Сохраняются сервером для клиента системы.

Название	Тип	Назначение
FrameID	int	Идентификатор рамки (PK)[AUTOINC]
CameraID	int	Идентификатор камеры, на которой появилась рамка (FK)
TimeBegin	datetime	Время начала периода
TimeEnd	datetime	Время окончания периода

- PK

ключевой поле
- FK

ключевое поле из другой таблицы
- AUTOINC

автоинкрементное поле (или через последовательность)

Вид Computers создан для скрытия хранения параметров на уровне системы и на уровне компьютера

```
SELECT
dbo.ComputersTable.ComputerID,
dbo.ComputersTable.ComputerIP,
dbo.ComputersTable.ArchivePath,
dbo.System.FreePercentageDiskSpace,
dbo.System.StorageDayPeriod,
dbo.System.Codec_DataRate,
dbo.System.Codec_FCC,
dbo.System.Codec_KeyRate,
dbo.System.Codec_Quality,
dbo.System.Codec_State,
dbo.System.Video_FMP,
dbo.System.Video_MaxSize,
dbo.System.AlgorithmSensitivity,
dbo.System.ArchiveWriteCondition
FROM      dbo.ComputersTable CROSS JOIN dbo.System
```

Реестр(Windows-версия)

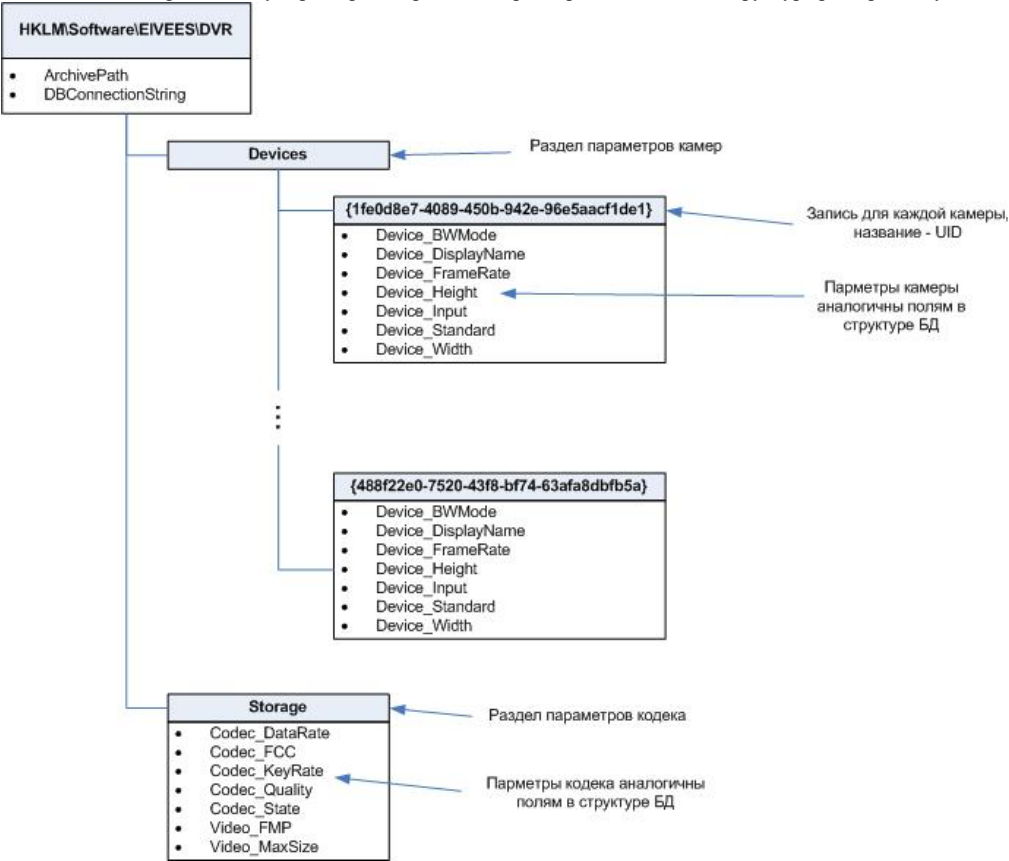
Режим работы CHCSVA с БД

Используется DVRDBBridge для локального хранения параметров работы системы. Корневая директория – HKLM\Software\EIVEES\DVR



Режим работы CHCSVA с реестром

Если DVRDBBridge Использует реестр для хранения параметров CHCSVA, то структура реестра следующая



Формат архива

Видео с камер пишет по требованию сервера и складывается в файловый архив, который сегментируется по времени. Каждый файл в архиве имеет имя в следующем формате: %s\%ld\%02d-%02d-%02d\%02d-%02d-%02d.avi, \$BASEPATH, \$CameraID, \$Year%100, \$Month, \$Day, \$Hour, \$Minute, \$Second

Параметр	Комментарий	Значение
BASEPATH	Полный путь к корневой папке архива	Строка в формате, воспринимаемой файловой системой
CameraID	Идентификатор камеры из БД	1...2 ³¹ -1
Year	Год	2006...2 ³¹ -1
Month	Месяц	1-12
Day	День	1-31
Hour	Час	0-23
Minute	Минута	0-59
Second	Секунда	0-59

Это имя указывает на время, к которому относится начало видеозображения в этом файле.

Структура компонентов

DVRClient

Назначение

Основное приложение системы.

Функции

- 1. **Пользовательские задачи.** Описаны в функциональной спецификации.
- 2. **Протоколирование сообщений.** Ведение файла протокола для всех компонентов системы, которые поддерживают данную возможность.

Дополнительные элементы управления

Элемент просмотра тревожных ситуаций

Обеспечивает визуальное отображение различных типов тревожных ситуаций по камерам.

Входные данные

1. Период времени, за который отображаются ситуации.
2. Список ситуаций (период времени и тип).
3. Список пометок пользователя (текст и время).
4. Текущее архивное время (дата и время).

Выходные данные

Сообщение родительскому окну, если был двойной щелчок на ситуации/пометке.

Используемые компоненты

Зависимости

Нет

Интерфейсы

Данные

DVRLauncher

Выполняет функции главного серверного приложения, запуская серверную часть.

Задачи

1. **Протоколирование сообщений.** Аналогично клиенту. Отображает журналы сообщений в отдельных вкладках.
2. **Определение уже запущенных экземпляров приложения.** При попытке запуска более одного экземпляра выдается сообщение пользователю.

DVREngine

Назначение

Компонент реализации работы детектирования тревожных ситуаций. Подразделяется на

1. **Серверная часть.** Основная логика работы сервера.
2. **Клиентская часть.** Поставка информации от сервера для клиента (межпроцессное и сетевое взаимодействие клиент-сервер).

Функции серверной части

1. **Запуск серверного поставщика видео.** Интерфейсы CHCSVA с флагом сервера.
2. **Запуск алгоритма детектирования на уровне камеры.** Поставка изображений с камер алгоритмам, загрузка маски (планируется)
3. **Запуск алгоритма детектирования на уровне всей системы.**
4. **Обеспечение взаимодействия уровней камера-система.** Для обнаружения тревожных ситуаций обеспечивать сетевое или локальное дуплексное взаимодействие рабочих потоков алгоритмов уровней камера-система
5. **Сохранение и передача тревожных ситуаций.** Происходит периодический опрос алгоритма на предмет тревожной ситуации. Выходные данные (тип ситуации, время, список камер) при необходимости сохраняются в БД и передаются клиенту.
6. **Передача данных телеметрии системы клиенту.** Отслеживать состояние удаленных и локального серверов, камер системы. Данная информация по запросу передается клиенту.

Функции клиентской части

1. **Прием данных телеметрии информации от серверной части.** Не реализовано.
2. **Чтение/установка параметров хранения архива**
3. **Чтение/установка параметров алгоритмов**

Используемые компоненты

DVRDBBridge

IDBServer

запись рамок в БД

IVideoConfig

чтение конфигурации системы

DVRAlgo

ICameraAnalyzer
интерфейс детектирования на уровне камеры
IHallAnalyzer
интерфейс детектирования на общем уровне

DVRBackup

IDVRBackup
интерфейс удаления архива

Зависимости

1. DVRClient
2. DVRLauncher

Интерфейсы

IClient
предоставляет интерфейс клиентской части

DVRLauncher

IServer
предоставляет интерфейс серверной части

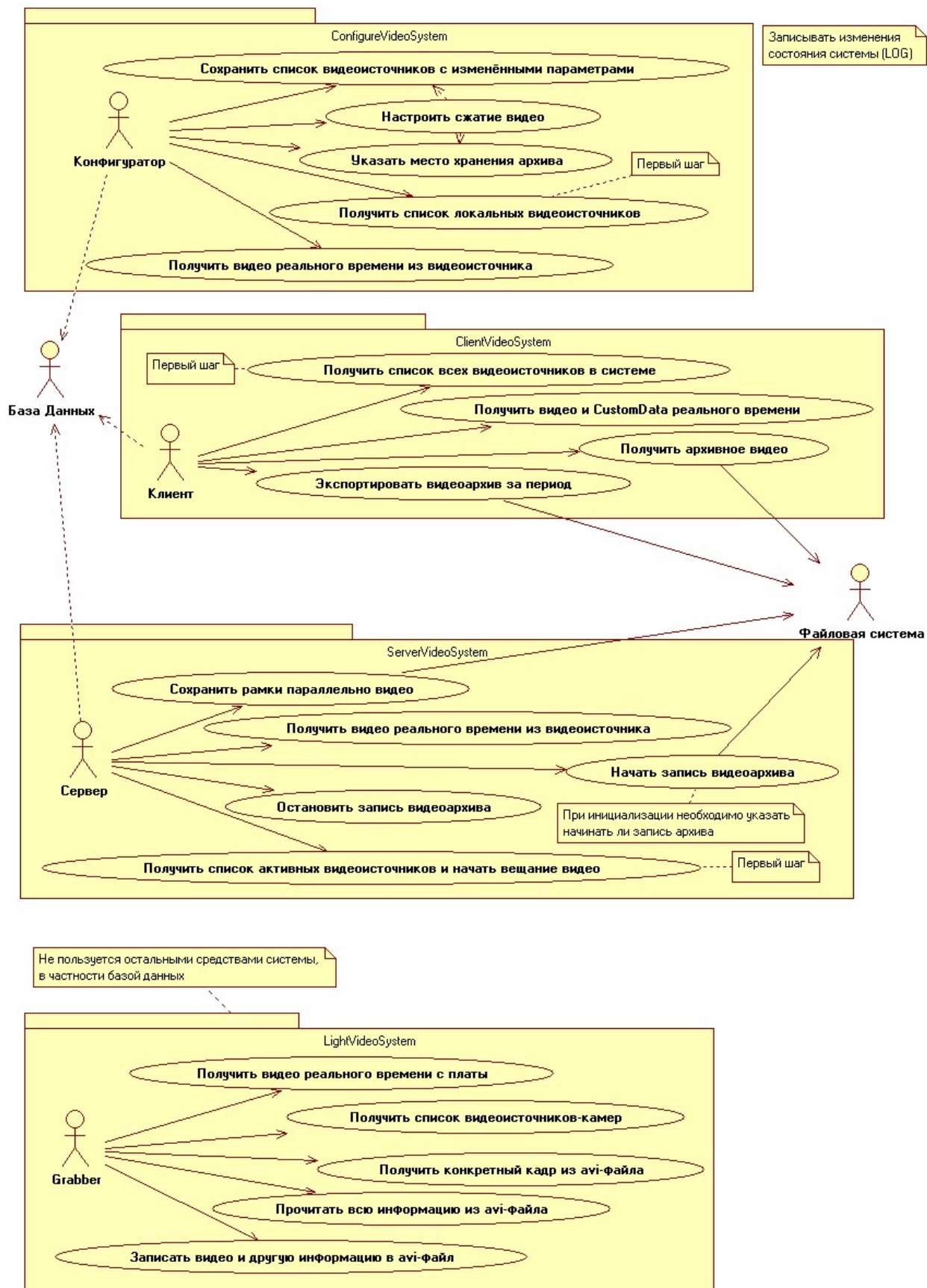
Данные

CHCSVA

Поставщик видеок кадров камеры (сервер, клиент) и файлового архива (клиент). Серверная часть должна быть запущена на каждом компьютере.

Задачи

1. **Обновление списка плат.** При старте компонента на серверной части должен происходить опрос всех устройств и сопоставление их с имеющимися. При необходимости добавить/удалить устройство.
2. **Получение текущего видеок кадра для заданной камеры.** По заданному идентификатору камеры синхронно (по запросу) передавать текущий кадр. Сервер получает кадр без сжатия в формате YUY2, клиент – сжатый подходящим качеством в формате RGB24.
3. **Начать/остановить запись видеоархива.** Производить запись видеоархива в заданном формате и по заданному пути. Запись должна начинаться и заканчиваться по команде извне.
4. **Получение видеок кадра из архива.**
 1. Переходить на произвольное время, при этом переходить в режим последовательной выборки кадров.
 2. По заданному идентификатору камеры синхронно (по запросу) передавать текущий кадр клиенту в формате RGB24.
5. **Экспорт архива** в заданный формат.
6. **Протоколирование событий.** Для отладки, просмотра истории работы системы необходимо передавать сообщения о своей работе клиенту или серверу.

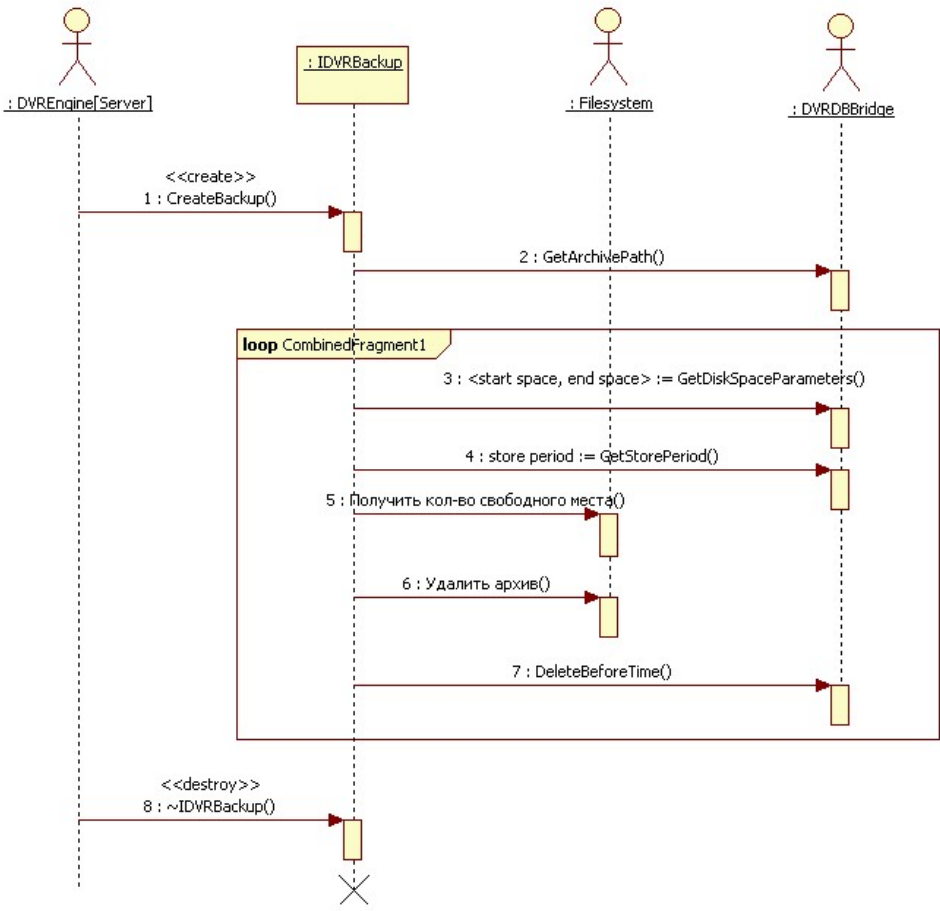


DVRBackup

Назначение

Функции

- 1. **Удаление по параметрам.** Данные (файловый архив и записи БД) должны удаляться синхронно, т.е. если происходит удаление видео до времени **T**, то все записи БД должны относиться к времени позже **T**. Время архивного файла высчитывается исходя из его имени.
 - 1. **По процентному заполнению диска.** Периодически (5-10 мин) запрашивать у DVRDBBridge параметры диска - **A**-параметр начала удаления и **B**-параметр конец удаления (при этом **A**≤**B**, иначе ошибка) и определять **X**-текущее процентное соотношение свободного дискового пространства. Если **X**<**A** , то начать удаление пока **X**<**B**.
 - 2. **По дате.** Периодически запрашивать у DVRDBBridge параметр даты удаления, и все данные, относящиеся ко времени раньше должны быть удалены.
- 2. **Протоколирование событий.** При осуществлении удаления по п.1 или его невозможности выводить сообщение через заданный интерфейс IMessage.



DVRAlgo

Алгоритмы на уровне камеры

Входные данные

- 1. Заголовок и данные в формате YUY2.
- 2. Маска изображения – двоичный массив, где каждый байт соответствует точке на изображении.

Задачи

- 1. **Поставлять информацию для детектирования на уровне системы.** По запросу поставлять двоичный массив данных.
- 2. **Протоколирование.** Выводить сообщение через отладочный интерфейс.

Алгоритмы на уровне системы

Входные данные

Данные от алгоритмов камер – двоичный массив.

Задачи

DVRRender

Компонент прорисовки изображения с камер в клиентском окне. Обеспечивает 2 режима работы – реального времени (отображение изображений реального времени с камер) и архивный (отображение архивного видеозаписи) по совокупности камер

Функциональные требования



Общие задачи выполняются в обоих режимах.

Нарисовать текст

Клиент задает

1. текстовую строку
2. прямоугольник вывода
3. цвет текста
4. параметр выравнивания текста в прямоугольнике по горизонтали (лево, право, центр)
5. параметр выравнивания текста в прямоугольнике по вертикали (верх, низ, центр)
6. параметр шрифта.

Система отрисовывает на окне текст и возвращает результат отрисовки клиенту

Нарисовать прямоугольник

Клиент задает

1. прямоугольник вывода
2. цвет рамки
3. толщину рамки.

Система отрисовывает на окне прямоугольник и возвращает результат отрисовки клиенту

Нарисовать закрашенный прямоугольник

Клиент задает

1. прямоугольник вывода,
2. цвет

Система отрисовывает на окне прямоугольник и возвращает результат отрисовки клиенту

Нарисовать растровое изображение

Клиент задает параметры изображения

1. ширина
2. высота
3. расположение цветовых каналов (RGB, BGR)
4. данные изображения
5. прямоугольник вывода на окно

Система отрисовывает на окне прямоугольник и возвращает результат отрисовки клиенту

Нарисовать кадр видео реального времени

1. Система инициализирует видеоподсистему в режиме клиента
2. Клиент устанавливает процедуру обратного вызова
3. Клиент устанавливает таймаут ожидания прорисовки для камеры (опционально, иначе значение по умолчанию)
4. Установить данный режим как активный
5. Система по таймеру вызывает функцию прорисовки
6. Получить интерфейс рендера
7. Клиент вызывает функцию прорисовки текущего кадра, задав идентификатор камеры, прямоугольник вывода
8. Система запрашивает видеопоток для камеры у подсистемы
9. Система возвращает после прорисовки код результата : успех, таймаут ожидания кадра, другая ошибка
10. Клиент выполняет варианты прорисовки текста, прямоугольников
11. Если для какой-то камеры превышено время от последнего вызова функции прорисовки ее кадров, то система освобождает видеопоток

Альтернативная последовательность "Сбой при инициализации видеоподсистемы"

1. В случае прорисовки возвращается соответствующий код результата

Альтернативная последовательность "Функция прорисовки не установлена клиентом"

1. Система не запрашивает видеопотоки

Альтернативная последовательность "Сбой при запросе видеопотока"

Аналогично "Сбой при инициализации видеоподсистемы"

Нарисовать кадр видеoarхива

1. Система инициализирует видеоподсистему в режиме клиента
2. Клиент устанавливает процедуру обратного вызова
3. Клиент устанавливает таймаут ожидания прорисовки для камеры (опционально, иначе значение по умолчанию)
4. Установить данный режим как активный
5. Система по таймеру вызывает функцию прорисовки
6. Получить интерфейс рендера
7. Клиент вызывает функцию прорисовки текущего кадра, задав идентификатор камеры, прямоугольник вывода
8. Система запрашивает архивный видеопоток для камеры у подсистемы. В зависимости от состояния камеры (не инициализирована, стоп, пауза, проигрывание) система получает кадр.
9. Система возвращает после прорисовки код результата : успех, таймаут ожидания кадра, другая ошибка
10. Клиент получает текущее время архива
11. Клиент выполняет варианты прорисовки текста, прямоугольников
12. Если для какой-то камеры превышено время от последнего вызова функции прорисовки ее кадров, то система освобождает видеопоток

Проиграть архив

Показывает как работает клиент с архивной видеоинформацией

1. Указать системе время начала и конца работы. Все действия с архивом производятся в некотором временном интервале $[T_1, T_2]$, в котором воспроизведение архива находится в состоянии остановки, паузы, проигрывания. При старте архива автоматически переходит в состояние проигрывания, по достижению конца периода автоматически останавливается. Время архива определяется исходя из времени поступающих видеок кадров. Поскольку одновременно может происходить проигрывания нескольких камер, времена кадров которых не обязаны совпадать, то требуется синхронизация покамерно.
2. Во время проигрывания система определяет текущее время архива по всем камерам
3. Клиент переводит архив в одно из состояний остановки, паузы, проигрывания
4. Клиент передвигает время архива на произвольное время в пределах диапазона $[T_1, T_2]$

Нефункциональные требования

Производительность

1. Потребление не более 20% процессорного времени среди всех потоков (в гросехр), с запуском алгоритмов детектирования на 4х камерах

Надежность и доступность

Обработка ошибок

1. Невозможность отображения какого-либо примитива сигнализируется кодом возврата из функции прорисовки.
2. Обработка аварийного завершения клиента или сервера CHCSVA, повторное соединение с видеопотоками.

Интерфейсные требования

1. Кроссплатформенность (оконная подсистема Windows и X11).
2. Отображение информации посредством OpenGL (возможно с помощью wxWidgets).
3. Оба режима работы отрисовывают в одно окно, в зависимости от текущей установки. Это сделано для избежания проблем с режимом, когда существуют одновременно 2 и более окон отображения OpenGL.
4. Поддержка до 32 камер. При этом максимальная величина кадр/секунда варьируется в зависимости от камер.
5. В качестве источников входных изображений используется CHCSVA. Привязка камер производится по идентификаторам.
6. Синхронизация по отображаемым камерам (время всех кадров должно быть в допустимом диапазоне).
7. Все примитивы отображаются в окне программы в зависимости от заданных координат вывода. Координаты вывода задаются в относительных величинах прямоугольником – координаты левого верхнего угла, ширина и высота. Все параметры лежат в диапазоне $[0,1]$. При этом $(0, 0)$ - левый верхний угол окна, $(1, 1)$ - правый нижний угол.

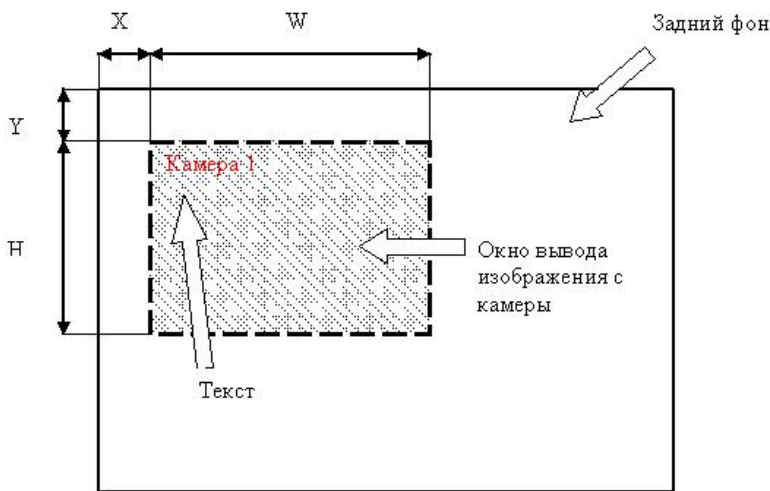
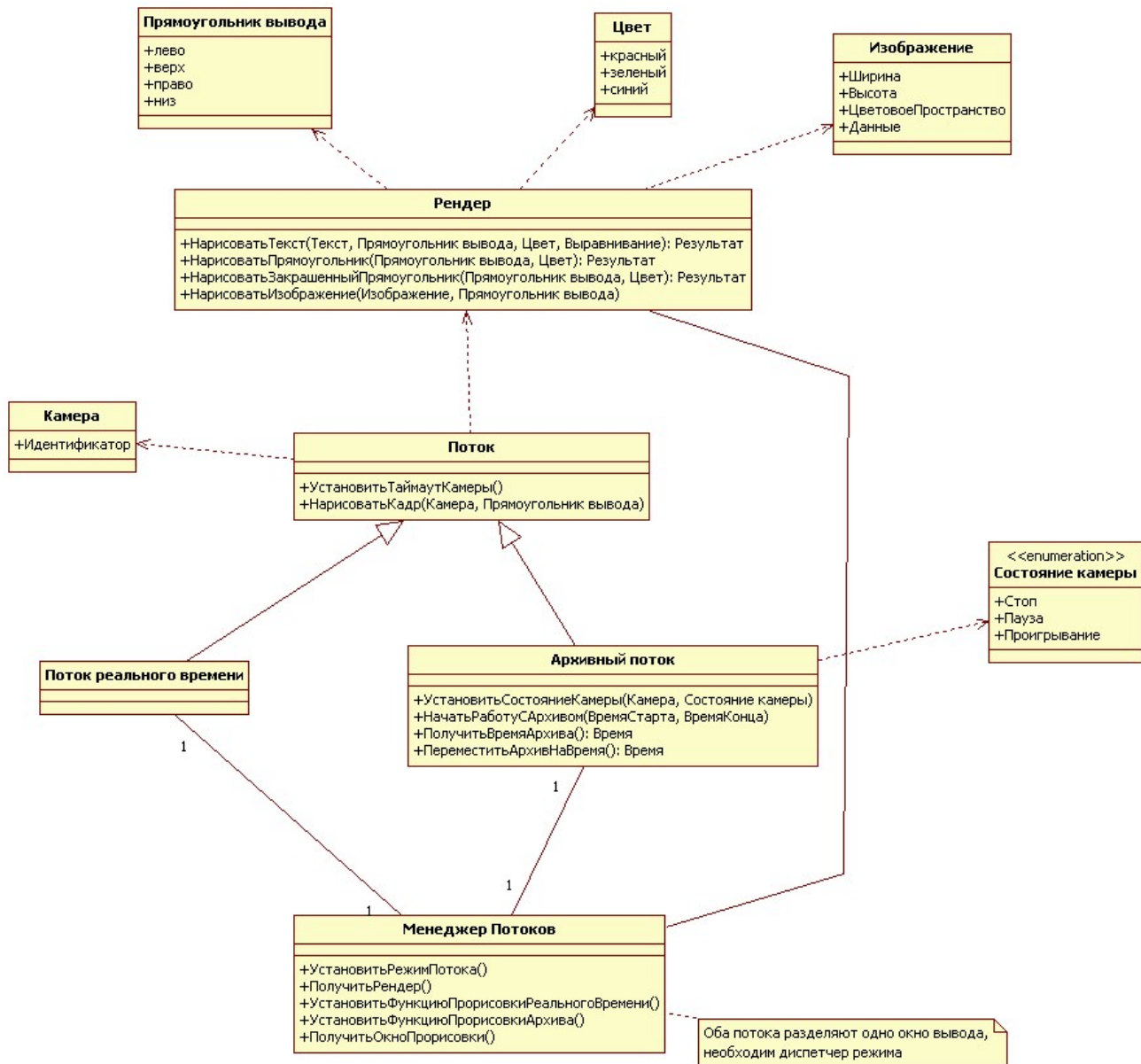


Диаграмма предметной области

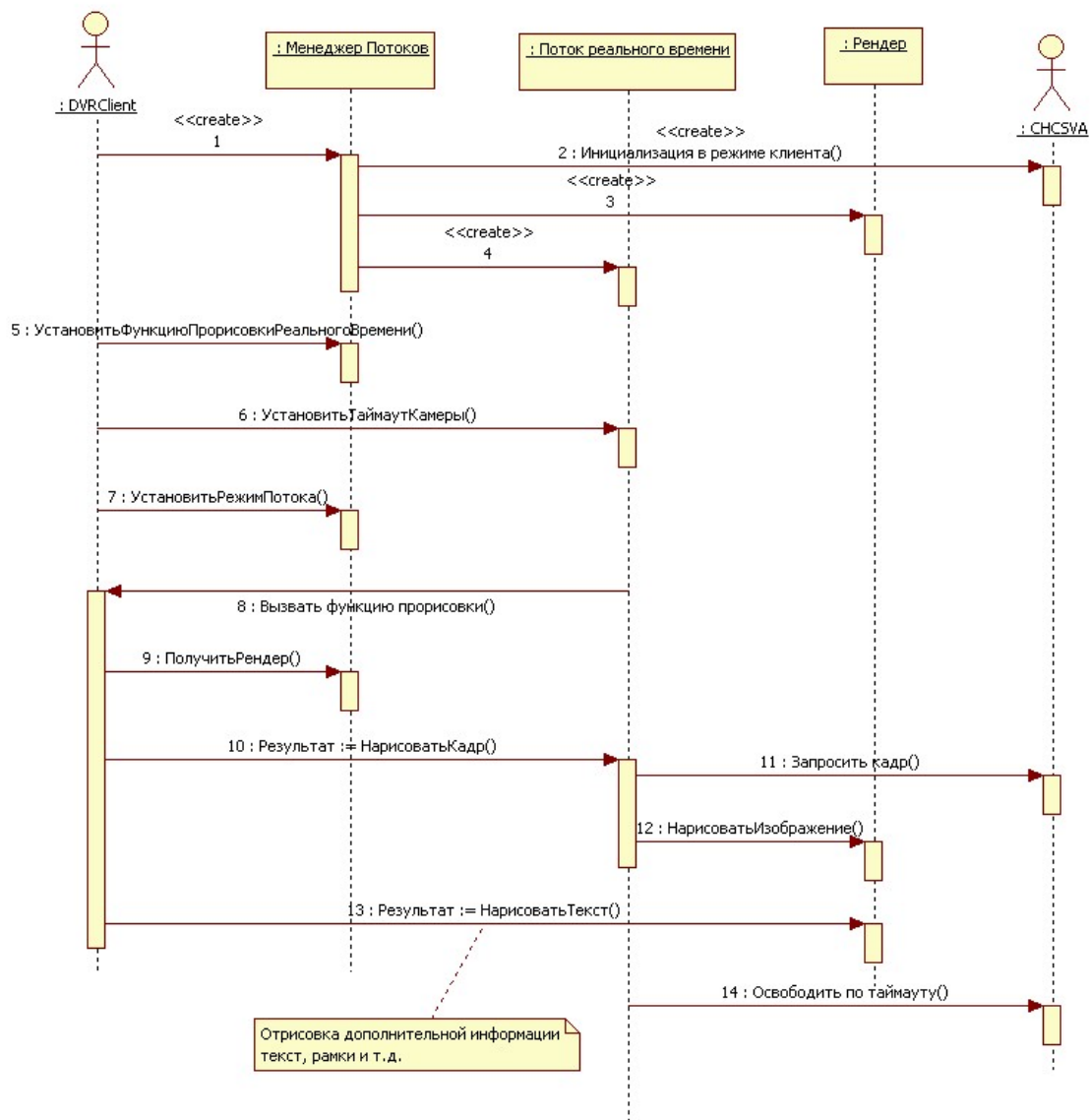
Для спецификации интерфейса к DVRRender и описания последовательностей вызовов в вариантах использования



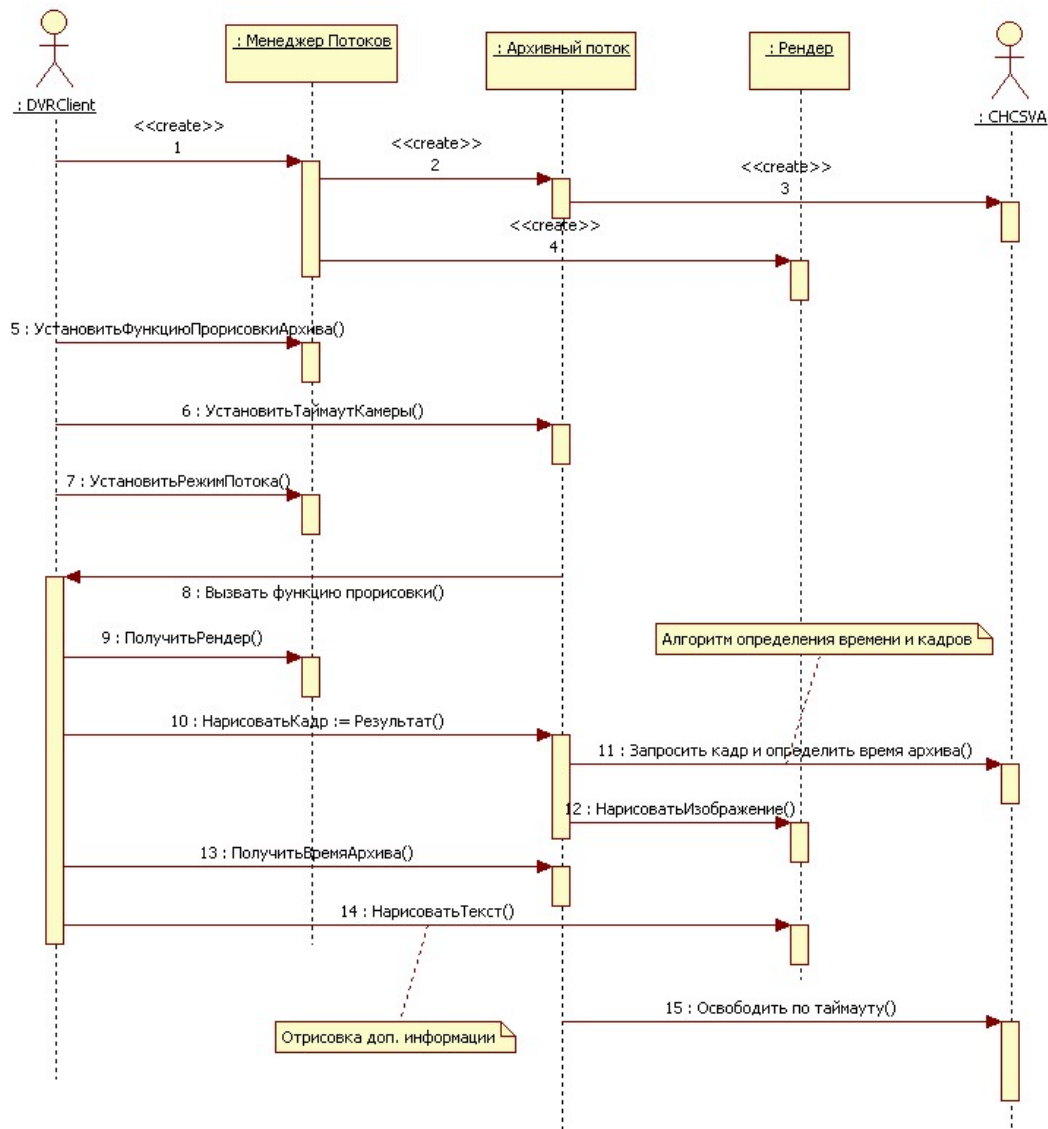
Последовательности действий в вариантах использования

Нарисовать кадр видео реального времени

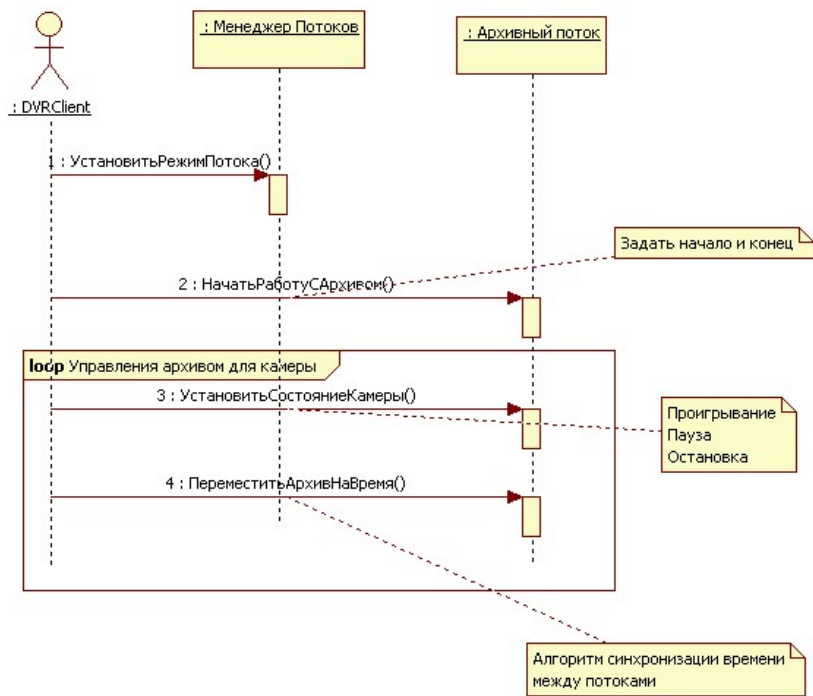
Последовательность согласно варианту использованию и диаграмме предметной области



Нарисовать кадр видеоархива



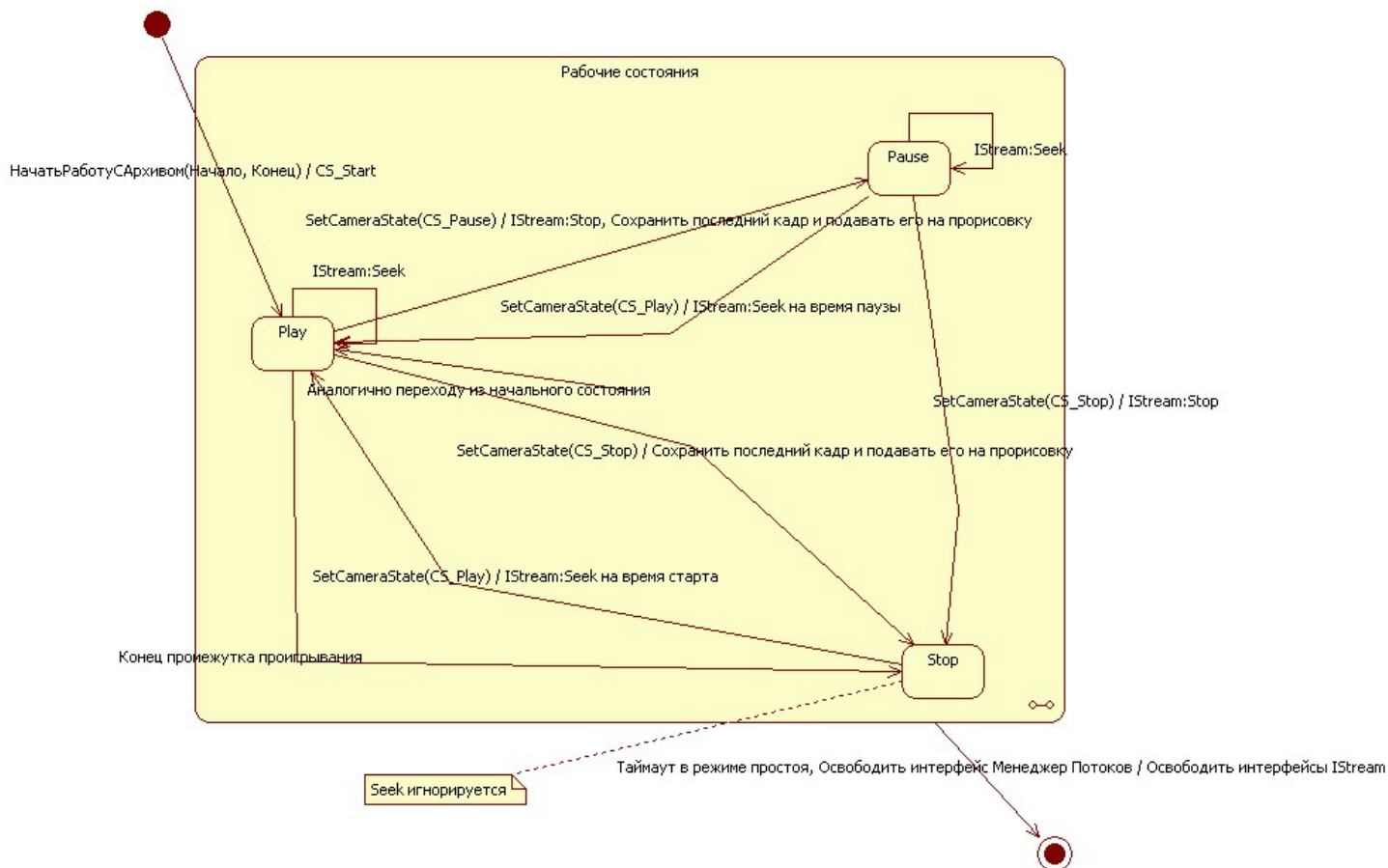
Проиhrать архив



Работа архива

Состояния архива

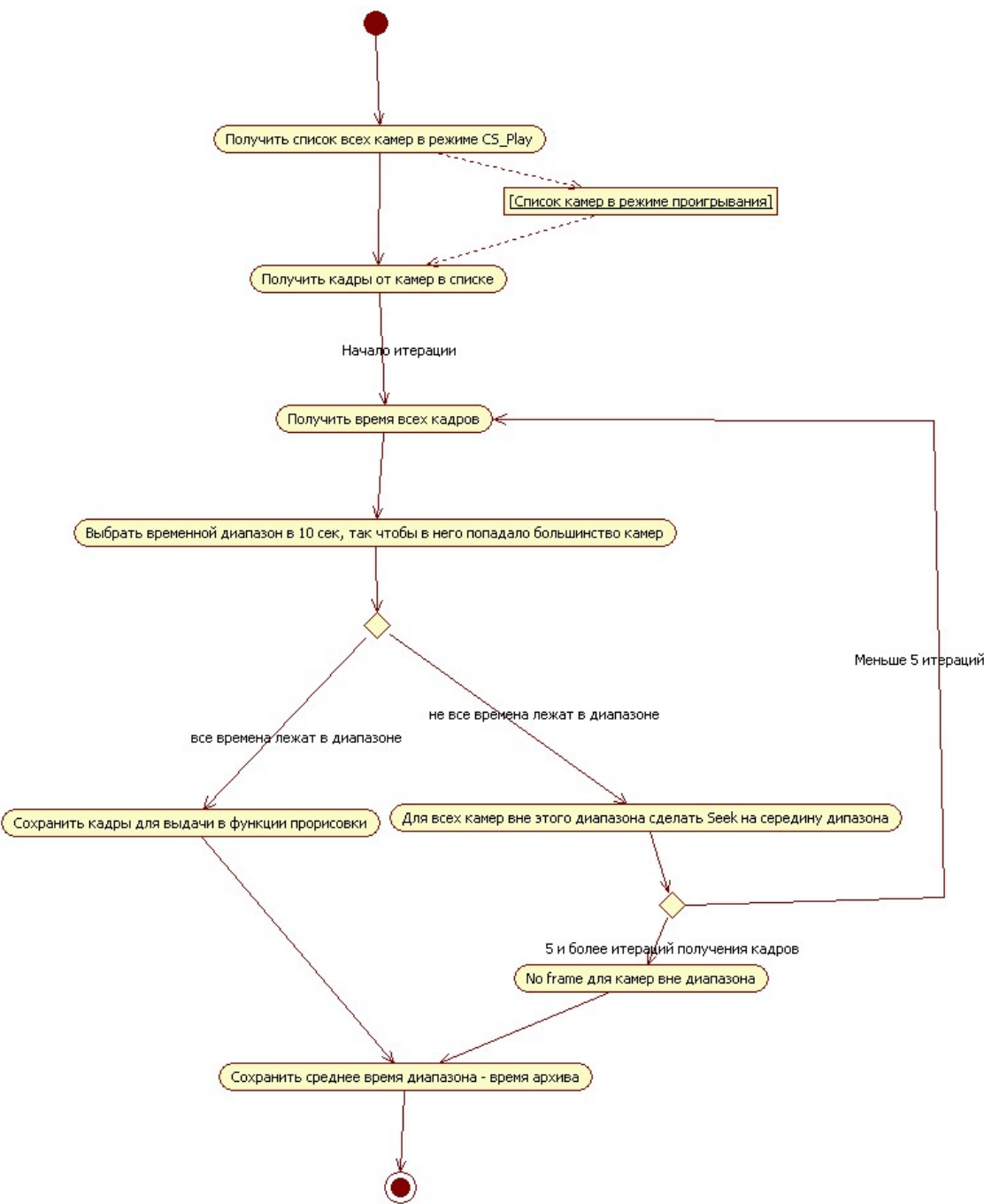
При работе с архивом каждая камера проходит состояния на диаграмме



Алгоритм получения текущего времени и синхронизации архива между камерами

Видеопровайдер не гарантирует

- 1. Сплошное хранение архива
- 2. Период, когда доступно изображения со всех камер



DVRDBBridge

Компонент доступа к конфигурации системы и работы с БД.

Задачи для CHCSVA

- 1. Запись и чтение инициализационных данных, уникальных для компьютера. Доступ производится по имени, поддерживаемые типы – целое длиной 4 байта, строка, двоичный массив.

Название	Тип	Назначение
Codec_DataRate	int	Параметр chcsva
Codec_FCC	int	Параметр chcsva (FourCC кодека)
Codec_KeyRate	int	Параметр chcsva

Codec_Quality	int	Параметр chcsva
Codec_State	Blob	Параметр chcsva
Video_FMP	int	Параметр chcsva
Video_MaxSize	int	Параметр chcsva
ArchivePath	string	Локальная директория, в которой хранится видеоархив

Запись и чтение информации о камерах на данном компьютере

- Получение идентификаторов камер
- Добавление новой камеры
- Удаление камеры по идентификатору

Запись и чтение информации по каждой камере

- Получение IP адреса компьютера, где подключена камера
- Получение строкового уникального идентификатора камеры (только для совместимости с chshva 1.0)
- Доступ к именным параметрам, поддерживаемые типы – целое длиной 4 байта, строка.

Название	Тип	Назначение
Device_BWMode	int	Параметр chcsva
Device_DisplayName	string	Параметр chcsva
Device_FrameRate	int	Параметр chcsva
Device_Width	int	Параметр chcsva
Device_Height	int	Параметр chcsva
Device_Input	int	Параметр chcsva
Device_Standard	int	Параметр chcsva

Задачи для серверной части

1. Получить список локальных камер
2. Получить список удаленных камер с IP адресами серверов
3. Запись в БД информации о тревожных ситуациях
4. Чтение из БД настроек алгоритма

Задачи для клиентской части

1. Получить список всех камер
2. Получить IP адресами серверов
3. Чтение из БД информации о тревожных ситуациях
4. Запись в БД настроек алгоритма
5. Чтение/запись в БД комментариев пользователя
6. Чтение/запись в БД событий системы

Задачи для DVRBackup

1. Передать файловый путь к архиву
2. Передать параметры удаления архива по дисковому пространству - 2 числа типа float от 0 до 100
3. Передать параметры удаления архива по дати - кол-во дней хранения архива.
4. Удалять записи о тревожных ситуациях раньше определенной даты

Режимы работы

1. Основной режим работы, когда все необходимые данные сохраняются в БД. Используемые СУБД:
 1. Базовая Win-версия: MS Access
 2. Расширенная Win-версия
 3. Unix-версия (базовая, расширенная): PostgreSQL
2. Режим работы с реестра. Данные по конфигурации хранятся в соответствии с форматом реестра

Пользовательский интерфейс

Изменения и дополнения

Версия	Дата	Комментарий
0.1	03.10.2005	Начальная версия
0.2	21.11.2005	Компоненты, исполнители, требования к DVRRender
0.3	12.12.2005	Структура БД, реестр

0.4	14.12.2005	Задачи и требования к компонентам, структура реестра
0.5	15.12.2005	Элемент ввода даты/времени, алгоритм, chcsva
0.6	02.02.2005	1. Перехали на wiki 2. Упорядочил структу в соответствии с IEEE 1066-1998
0.7	12.04.2006	Обновлены требования к DVRRender (варианты использования + алгоритм синхронизации)

Retrieved from "http://zork/wiki/index.php/DVR_SDD"