



Программный комплекс «Аххон Next»

SDK интеграции (HTTP API)

Версия 1.1



EXPERIENCE THE NEXT™

Ай Ти Ви групп
Москва 2012



Содержание

1	ОБЩИЕ СВЕДЕНИЯ О ДОКУМЕНТЕ	4
2	ОБЩИЕ СОГЛАШЕНИЯ	4
3	ИНФРАСТРУКТУРА	4
3.1	Получение уникального идентификатора	4
3.2	Получение списка серверов	4
3.2.1	Список всех серверов в домене	4
3.2.2	Информация о конкретном сервере	4
3.3	Получение списка источников видео (камер)	5
3.3.1	Получение всех доступных источников	5
3.3.2	Получение всех доступных оригинальных источников для конкретного сервера	6
3.3.3	Получение информации о конкретном источнике	6
3.3.4	Получение информации обо всех источниках	6
3.4	Получение живого потока от видеокамеры	7
3.5	Получение снимка с видеокамеры	7
3.6	Получение архивного потока	7
3.6.1	Получение информации об архивном потоке	8
3.6.2	Управление архивным потоком	8
3.7	Покадровый просмотр архива	9
3.7.1	Получение кадра по моменту времени	9
3.7.2	Получение времени регистрации кадра	9
3.8	Получение содержания ММ архива	9
3.9	Получение информации о сработках детекторов и тревогах	10
3.9.1	Получение списка тревог	10
3.9.2	Получение списка срабатываний детекторов	11
3.10	Управление телеметрией	13
3.10.1	Получение списка устройств телеметрии для источника видео	13
3.10.2	Управление степенями свободы	13
3.10.2.1	Получение информации о степенях свободы	13
3.10.2.2	Изменение наклона и поворота	14
3.10.2.3	Изменение одной из степеней свободы	14
3.10.2.4	Переход по точке на экране	14
3.10.2.5	Увеличение области изображения	15
3.10.2.6	Автоматическая фокусировка и диафрагма	15
3.10.3	Управление пресетами	15
3.10.3.1	получение списка пресетов	15
3.10.3.2	Создание и изменение пресета	15
3.10.3.3	Переход к пресету и удаление пресета	16

3.11 Получение статистики	16
---------------------------------	----

1 Общие сведения о документе

Документ «Программный комплекс Аххон Next. SDK интеграции (HTTP API)» содержит API (интерфейс программирования приложений) для работы web- и iOS-клиента программного комплекса Аххон Next.

Данный документ является собственностью компании ООО Ай Ти Ви групп. Компания ООО Ай Ти Ви групп оставляет за собой право в будущем менять данный API в одностороннем порядке.

2 Общие соглашения

HTTP сервер NGP возвращает результаты вызова методов в формате JSON.

3 Инфраструктура

3.1 Получение уникального идентификатора

GET `http://server/prefix/uuid` - на каждый запрос генерирует уникальный идентификатор (UUID).

Уникальный идентификатор может понадобиться, например, для получения информации о последнем отображенном кадре архивного видео или для управления архивным потоком.

Пример ответа:

```
"f03c6ccf-b181-4844-b09c-9a19e6920fd3"
```

3.2 Получение списка серверов

3.2.1 Список всех серверов в домене

GET `http://server/prefix/hosts/` - получить список всех хостов в домене.

Пример ответа:

```
[ "SERVER1", "SERVER2" ]
```

3.2.2 Информация о конкретном сервере

GET `http://server/prefix/hosts/HOSTNAME` - получить информацию о конкретном хосте.

Пример ответа:

```
{
  "hostname": "SERVER2",
  "domainInfo":
  {
    "domainName": "DomainName",
    "domainFriendlyName": "Пользовательское название домена, если есть"
  },
}
```

```
"platformInfo" :  
{  
  "machine" : "ARM9",  
  "os" : "Linux"  
},  
"licenseStatus" : "Expired",  
"timeZone" : "+180" // GMT+3  
}
```

3.3 Получение списка источников видео (камер)

3.3.1 Получение всех доступных источников

GET <http://server/prefix/video-origins/> - получить все доступные оригинальные источники (камеры). Возвращаемые идентификаторы будут иметь вид "HOSTNAME/ObjectType.Id/Endpoint.Name". Попутно будут возвращены friendly name и прочая метаинформация.

Пример ответа:

```
{  
  "SERVER1/DeviceIpint.3/VideoSource.0:0" :  
  {  
    "origin" : "SERVER1/DeviceIpint.3/VideoSource.0:0",  
    "friendlyNameLong" : "Камера 3",  
    "friendlyNameShort" : "3"  
  },  
  "SERVER2/DeviceIpint.5/VideoSource.0:0" :  
  {  
    "origin" : "SERVER2/DeviceIpint.5/VideoSource.0:0",  
    "friendlyNameLong" : "Камера 5",  
    "friendlyNameShort" : "5"  
  }  
}
```

3.3.2 Получение всех доступных оригинальных источников для конкретного сервера

GET http://server/prefix/video-origins/HOSTNAME/ получить все доступные оригинальные источники (камеры) только для конкретного хоста.

3.3.3 Получение информации о конкретном источнике

GET http://server/prefix/video-origins/VIDEOSOURCEID - получить информацию о конкретном источнике. VIDEOSOURCEID - трехкомпонентный идентификатор endpoint-а источника (HOSTNAME/ObjectType.Id/Endpoint.Name).

Пример запроса:

```
GET http://server/prefix/video-origins/SERVER1
```

Пример ответа:

```
{
  "SERVER1/DeviceIpint.3/VideoSource.0:0" :
  {
    "origin" : "SERVER1/DeviceIpint.3/VideoSource.0:0",
    "friendlyNameLong" : "Камера 3",
    "friendlyNameShort" : "3"
  }
}
```

3.3.4 Получение информации обо всех источниках

GET http://server/prefix/video-sources/* - получить список всех доступных источников, не только оригинальных.

Пример запрос:

```
GET http://server/prefix/video-sources/SERVER2
```

Пример ответа:

```
{
  "SERVER2/DeviceIpint.5/VideoSource.0:0" :
  {
    "origin" : "SERVER2/DeviceIpint.5/VideoSource.0:0",
    "friendlyNameLong" : "Камера 5",
    "friendlyNameShort" : "5"
  }
}
```

```

},
"SERVER2/VideoDecoder.0/VideoSource" :
{
  "origin" : "SERVER2/DeviceIpint.5/VideoSource.0:0",
  "friendlyNameLong" : "SERVER2/Видеодекодер 0",
  "friendlyNameShort" : "Видеодекодер 0"
}
}

```

3.4 Получение живого потока от видеокамеры

GET <http://server/prefix/live/media/VIDEOSOURCEID?параметры>.

Параметры:

format – параметр может принимать значения "mjpeg", "webm" либо "h264". В случае, если формат не указан или не распознан, сервер выбирает нативный формат (чтобы избежать перекодирования). Если при этом нативный формат не поддерживается клиентом, сервер выбирает WebM.

w – значение ширины кадра.

h – значение высоты кадра.

Пример запроса:

```
GET http://server/prefix/live/media/TEST-1/DeviceIpint.23/VideoSource.0:0?format=mjpeg&w=640&h=480
```

3.5 Получение снимка с видеокамеры

GET <http://server/prefix/live/media/snapshot/VIDEOSOURCEID?параметры>.

Параметры:

w – значение ширины кадра.

h – значение высоты кадра.

Пример запроса:

```
GET http://server/prefix/live/media/snapshot/TEST-1/DeviceIpint.23/VideoSource.0:0
```

```
GET http://server/prefix/live/media/snapshot/TEST-1/DeviceIpint.23/VideoSource.0:0?w=640&h=480
```

3.6 Получение архивного потока

GET <http://server/prefix/archive/media/VIDEOSOURCEID/STARTTIME?параметры>,

где STARTTIME - время в формате ISO.

Параметры:

speed - скорость воспроизведения, может принимать отрицательные значения.

format - параметр может принимать значения "mjpeg", "webm" либо "h264". В случае, если формат не указан или не распознан, сервер выбирает нативный формат (чтобы избежать перекодирования). Если при этом нативный формат не поддерживается клиентом, сервер выбирает WebM.

В случае, если не задано значение ни одного из двух вышеперечисленных параметров, скорость считается равной 0, а формат - JPEG, и запрос интерпретируется как запрос на покадровый просмотр архива.

id - уникальный идентификатор архивного потока (может отсутствовать). Необходим для получения информации о потоке или для управления им.

w – значение ширины кадра.

h – значение высоты кадра.

Пример запроса:

```
GET http://server/prefix/archive/media/TEST-1/DeviceIpn23/VideoSource.0:0/20110608T060141.375?speed=1&w=640&h=480
```

3.6.1 Получение информации об архивном потоке

GET http://server/prefix/archive/media/rendered-info/UUID - получение информации о последнем отображенном кадре.

Здесь UUID - уникальный идентификатор архивного потока для которого запрашивается информация.

Доступна следующая информация о кадре:

timestamp - временная метка кадра.

Пример запроса:

```
GET http://localhost:8000/asip-api/archive/media/rendered-info/%22996cea31-91c4-9a46-9269-48b998fd2f29%22
```

Пример ответа:

```
{
  "timestamp": "20110408T103627.048"
}
```

3.6.2 Управление архивным потоком

GET http://server/prefix/archive/media/stop/UUID - остановка архивного потока с идентификатором UUID.

После успешного выполнения возвращается информация о последнем кадре.

3.7 Покадровый просмотр архива

3.7.1 Получение кадра по моменту времени

GET <http://server/prefix/archive/media/VIDEOSOURCEID/STARTTIME> - получение кадра, соответствующего моменту времени STARTTIME. Кадр возвращается в формате JPEG.

3.7.2 Получение времени регистрации кадра

GET <http://server/prefix/archive/contents/frames/VIDEOSOURCEID/ENDTIME/BEGINTIME?limit=COUNT> - получение времени регистрации кадров, находящихся в MM архиве. Семантика параметров описана в разделе *Получение содержания MM архива*. Значение по умолчанию для параметра limit равно 250. Сервер не обязан следовать заданному клиентом значению limit и может по своему усмотрению вернуть меньшее количество результатов.

В возвращаемом ответе (json объекте) массив, содержащий моменты времени, соответствующие кадрам, помещается в свойство с именем **frames**.

В возвращаемом ответе (json объекте) отдельное свойство **more** - булевское значение, сигнализирующее о том, был ли выбран указанный в запросе отрезок времени полностью (false), или остались кадры, timestamp-ы которых не были возвращены из-за достижения предельного числа возвращаемых значений.

Пример запроса:

```
GET
http://server/prefix/archive/contents/frames/SERVER1/DeviceIpint.2/VideoSource.0:0/20101230T103943.000/20101230T103952.000?limit=3
```

Пример ответа:

```
{
  "frames" :
  [ "20101230T103951.800", "20101230T103951.760", "20101230T103951.720" ],
  "more" : false
}
```

3.8 Получение содержания MM архива

GET <http://server/prefix/archive/contents/intervals/VIDEOSOURCEID/ENDTIME/BEGINTIME?limit=COUNT&scale=SIZE> - получение содержания архива, начиная от момента времени BEGINTIME, заканчивая моментом времени ENDTIME.

В случае, если BEGINTIME не указан, подразумевается значение, соответствующее бесконечному будущему. Если при этом ENDTIME также не указан, подразумевается его значение, соответствующее бесконечному прошлому. Вместо BEGINTIME и ENDTIME могут быть использованы слова "past" или "future", означающие бесконечное прошлое и бесконечное будущее соответственно.

Необязательный параметр `limit` указывает, каким количеством интервалов следует ограничиться. Значение `limit` по умолчанию равно **100**.

Необязательный параметр `scale` указывает, при каком минимальном временном расстоянии между двумя интервалами они будут выданы как два различных интервала (а не склеены в один). Значение `scale` по умолчанию равно **0**.

Порядок выдачи интервалов соответствует отношению между заданными `BEGINTIME` и `ENDTIME` (по возрастанию, если `BEGINTIME < ENDTIME`, и по убыванию, если `ENDTIME < BEGINTIME`). При этом начало и конец интервала всегда выдаются в естественном порядке, т.е. значение времени начала интервала будет меньше либо равно времени конца интервала).

В возвращаемом ответе (json объекте) массив, содержащий интервалы, помещается в свойство с именем **intervals**.

В возвращаемом ответе (json объекте) отдельное свойство **more** - булевское значение, сигнализирующее о том, был ли выбран указанный в запросе отрезок времени полностью (`false`), или остались интервалы, которые не были возвращены из-за достижения предельного числа возвращаемых интервалов (`true`).

Пример запроса:

GET

`http://server/prefix/archive/contents/intervals/SERVER1/DeviceIpint.2/VideoSource.0:0/20101230T103904.000/20101230T103959.000?limit=3`

Пример ответа:

```
{
  "intervals" :
  [
    { begin: "20101230T103950.000", end: "20101230T103955.230" },
    { begin: "20101230T103923.110", end: "20101230T103941.870" }
  ],
  "more" : true
}
```

3.9 Получение информации о сработках детекторов и тревогах

3.9.1 Получение списка тревог

GET

`http://server/prefix/archive/events/alerts/VIDEOSOURCEID/ENDTIME/BEGINTIME?limit=COUNT&offset=COUNT` - получение списка тревог. В случае, если значение `limit` не указано, оно считается равным 100. Поле **raisedAt** не уникальное, поэтому может понадобиться пропуск уже полученных ранее тревог от начала интервала поиска.

Пример ответа:

```
{
  "events" :
  [
    {
      "type": "alert",
      "id": "42D43A79-90D6-4ba7-BD23-1714996A2F88",
      "raisedAt": "20101230T103950.000",
      "zone": "SERVER1/DeviceIpint.3/VideoSource.0:0",
      "reasons": ["ruleAlert", "videoDetector"],
      "initiator": "4359EC93-EF31-4de0-9EDE-AA5C5803D6F8",
      "reaction":
      {
        "user": "root",
        "reactedAt": "20101230T103958.000",
        "severity": "alarm"
      }
    },
    ...
  ],
  "more": true
}
```

Возможные значения в массиве **reasons**: armed, disarmed, userAlert, ruleAlert, videoDetector, audioDetector, ray.

Возможные значения поля **severity**: unclassified, false, notice, warning, alarm.

3.9.2 Получение списка срабатываний детекторов

GET

<http://server/prefix/archive/events/detectors/VIDEOSOURCEID/ENDTIME/BEGINTIME?limit=COUNT&of fset=COUNT> - получение списка срабатываний детекторов. В случае, если значение limit не указано, оно считается равным 100. Поле **timestamp** не уникальное, поэтому, может понадобиться пропуск уже полученных ранее срабатываний детекторов от начала интервала поиска.

Пример ответа:

```
{
  "events":
  [
    {
      "id": "433d45ec-0b7f-aa43-8491-c8acb7d0ac56"
      ,"source": "hosts/SERVER1/SituationDetector.0"
      ,"origin": "hosts/SERVER1/DeviceIpint.0/SourceEndpoint.video:0:0"
      ,"detectorId": "1"
      ,"type": "CrossOneLine"
      ,"alertState": "ended"
      ,"timestamp": "20120314T121512.597"
      ,"rectangles":
      [
        {
          "index": "1"
          ,"left": "0.622086710929871"
          ,"top": "0.68798337459564196"
          ,"right": "0.65736908435821495"
          ,"bottom": "0.79889315128326399"
        }
      ]
    },
    ...
  ],
  "more": true
}
```

3.10 Управление телеметрией

3.10.1 Получение списка устройств телеметрии для источника видео

GET `http://server/prefix/control/telemetry/list/OBJECTID` - Получение списка устройств телеметрии для источника видео, где OBJECTID – двухкомпонентный идентификатор объекта (HOSTNAME/ObjectType.Id).

Пример ответа:

```
[  
  "SERVER1/DeviceIpint.2/TelemetryControl.0"  
]
```

В дальнейшем для обозначения устройств телеметрии вида HOSTNAME/ObjectType.Id/TelemetryContol.n будет использоваться шаблон TELEMETRYCONTROLID.

3.10.2 Управление степенями свободы

3.10.2.1 Получение информации о степенях свободы

GET `http://server/prefix/control/telemetry/info/TELEMETRYCONTROLID` - Получение информации о степенях свободы, управление которыми поддерживается, способах управления ими (непрерывный, относительный, дискретный), предельно допустимые значения.

Пример ответа:

```
{  
  "degrees":  
  {  
    "tilt":  
    {  
      "relative": {"min": "-45", "max": "45"},  
      "continuous": {"min": "-10", "max": "10"}  
    },  
    "pan":  
    {  
      "absolute": {"min": "-170", "max": "170"},  
      "continuous": {"min": "-10", "max": "10"}  
    },  
    "zoom":
```

```

    {
      "absolute": {"min": "0", "max": "20"}
    }
  },
  "feature": ["autoFocus", "areaZoom", "pointMove"]
}

```

degrees - информация о степенях свободы (tilt, pan, zoom, focus, iris). Каждая степень свободы содержит список поддерживаемых способов управления (absolute, relative, continuous).

feature - список поддерживаемых функций (autoFocus, autoIris, areaZoom, pointMove).

3.10.2.2 Изменение наклона и поворота

GET <http://server/prefix/control/telemetry/move/TELEMETRYCONTROLID?параметры> - изменение наклона, поворота (tilt, pan).

Параметры:

mode - способ управления (absolute, relative, continuous);

pan, tilt - значение для соответствующих степеней.

Пример запроса:

GET <http://server/prefix/control/telemetry/move/TEST-1/DeviceIpint.25/TelemetryControl.0?mode=absolute&pan=-99&tilt=10>

3.10.2.3 Изменение одной из степеней свободы

GET <http://server/prefix/control/telemetry/{степень}/TELEMETRYCONTROLID?параметры> - изменение одной из степеней (zoom, focus, iris).

Параметры:

{степень} - изменяемая степень свободы (zoom, focus, iris);

mode - способ управления (absolute, relative, continuous);

value - значение.

Пример запроса:

GET <http://server/prefix/control/telemetry/zoom/TEST-1/DeviceIpint.25/TelemetryControl.0?mode=absolute&value=6> - меняется zoom;

GET <http://server/prefix/control/telemetry/focus/TEST-1/DeviceIpint.25/TelemetryControl.0?mode=relative&value=3> - меняется focus;

GET <http://server/prefix/control/telemetry/iris/TEST-1/DeviceIpint.25/TelemetryControl.0?mode=continuous&value=1> - меняется iris.

3.10.2.4 Переход по точке на экране

GET <http://server/prefix/control/telemetry/move/point/TELEMETRYCONTROLID?параметры> - переход по точке на экране.

Параметры:

x,y – значения координат точки по горизонтали и вертикали, указываемые относительно размера изображения;

Пример запроса:

GET http://server/prefix/control/telemetry/move/point/TEST-1/DeviceIpint.23/TelemetryControl.0?x=0.14&y=0.32

3.10.2.5 Увеличение области изображения

GET http://server/prefix/control/telemetry/zoom/area/TELEMETRYCONTROLID?параметры – увеличение (zoom) выбранной области изображения.

Параметры:

x,y - левый верхний угол области;

w,h - ширина и высота области.

Координаты и размеры указываются относительно размера изображения.

Пример запроса:

GET http://server/prefix/control/telemetry/zoom/area/TEST-1/DeviceIpint.24/TelemetryControl.0?x=0.23&y=0.089&w=0.25&h=0.25

3.10.2.6 Автоматическая фокусировка и диафрагма

GET http://server/prefix/control/telemetry/auto/TELEMETRYCONTROLID?параметры - авто фокусировка/диафрагма.

Параметры:

degree – принимает значения focus или iris.

Пример запроса:

GET http://server/prefix/control/telemetry/auto/TEST-1/DeviceIpint.24/TelemetryControl.0?degree=iris

3.10.3 Управление пресетами

3.10.3.1 получение списка пресетов

GET http://server/prefix/control/telemetry/preset/info/TELEMETRYCONTROLID- получить список существующих пресетов.

Пример запроса:

GET http://server/prefix/control/telemetry/preset/info/TEST-1/DeviceIpint.23/TelemetryControl.0

Пример ответа:

```
{  
  "0": "Коридор",  
  "1": "Входная дверь",  
  "4": "Дырка в заборе"  
}
```

3.10.3.2 Создание и изменение пресета

GET http://server/prefix/control/telemetry/preset/set/TELEMETRYCONTROLID?параметры - создать/изменить пресет.

Параметры:

pos - индекс;

label - наименование пресета.

Если пресет с указанным индексом уже существует, то его метка будет изменена.

Пример запроса:

GET http://server/prefix/control/telemetry/preset/set/TEST-1/DeviceIpint.23/TelemetryControl.0?pos=0&label=Exit

3.10.3.3 Переход к пресету и удаление пресета

GET http://server/prefix/control/telemetry/preset/{действие}/TELEMETRYCONTROLID?параметры - перейти или удалить пресет.

Параметры:

{действие} – принимает значение **go** или **remove**, используется чтобы перейти к/удалить пресет; pos - индекс пресета.

Пример запроса:

Осуществляется переход к уже созданному пресету с индексом 1:

GET http://server/prefix/control/telemetry/preset/go/TEST-1/DeviceIpint.23/TelemetryControl.0?pos=1
– данный запрос удаляет пресет с индексом 2:

GET http://server/prefix/control/telemetry/preset/remove/TEST-1/DeviceIpint.23/TelemetryControl.0?pos=2

3.11 Получение статистики

GET http://server/prefix/statistics/VIDEOSOURCEID - выдача статистики по указанному источнику видео.

GET http://server/prefix/statistics/webserver - выдача статистики о работе сервера.