



**Universelles Videomanagementsystem  
von  
Accellence Technologies GmbH**

---

**vimacc-SDK**  
**http-Interface**

Spezifikation http-Interface (vimacc Web-Server)

Dieses Dokument ist geistiges Eigentum der Accellence Technologies GmbH.  
Änderungen und Irrtümer vorbehalten.  
Dieses Dokument darf nur mit der ausdrücklichen Zustimmung der Accellence Technologies GmbH verwendet,  
vervielfältigt oder weitergegeben werden.

## Impressum

Herausgeber

Gesellschaft: Accellence Technologies GmbH  
Handelsregister: HRB 110799 Hannover  
Geschäftsführer: Dr. Heinz Stephanblome  
Redaktion: Mike Plötz

Tel: +49 (0)511 277 2400  
Fax: +49 (0)511 277 2499

E-Mail: info@accellence.de  
Internet: <http://www.accellence.de>  
Anschrift: Accellence Technologies GmbH  
Garbsener Landstrasse 10, 30419 Hannover, Deutschland

© 2013/2014 Accellence Technologies GmbH

Datum: 27.03.2014

# Inhaltsverzeichnis

---

1	Einleitung.....	5
1.1	Zweck des Dokumentes.....	5
1.2	Aufbau der Dokumentation .....	6
1.2.1	Dokumentenversionierung.....	6
2	Installation .....	7
3	AccVimaccHttpServer.....	8
3.1	Verzeichnisstruktur .\vimacc\data\www .....	8
3.2	Zugriff auf den vimacc http-Server .....	9
3.3	Authentifizierung .....	10
3.3.1	Keine Authentifizierung.....	10
3.3.2	Basic Authentication .....	10
3.3.3	Digest Authentication.....	11
3.3.4	Weitere Parameter des vimacc Web-Servers .....	11
3.4	vimacc Kommandos.....	12
3.4.1	keepAlive .....	12
3.4.2	Zugriff auf live-Videostreams .....	13
3.4.2.1	getDeviceList.....	13
3.4.2.2	getLiveVideo .....	15
3.4.2.3	getLiveVideoStillImage.....	15
3.4.2.4	getLiveVideoHtml .....	15
3.4.3	Steuerung PTZ-Kameras.....	16
3.4.3.1	Aktion STOP .....	16
3.4.3.2	Kamera nach unten.....	16
3.4.3.3	Kamera nach oben.....	16
3.4.3.4	Kamera nach links.....	16
3.4.3.5	Kamera nach rechts .....	17
3.4.3.6	Kamera hinein zoomen .....	17
3.4.3.7	Kamera heraus zoomen.....	17
3.4.4	Steuerung Workstation/Display/Videowand .....	18
3.4.4.1	Video auf einen Videodialog aufschalten.....	18
3.4.4.2	Löschen eines Videodialogs .....	18
3.4.4.3	Setzen eines definierten Grid.....	18
3.4.4.4	Setzen eines Grids für eine Anzahl Videodialoge.....	19
3.4.4.5	Aufschalten eines Szenarios.....	19
3.4.4.6	Anzeigemodus setzen.....	19
3.4.4.7	Aufschalten einer Sequenz .....	20
3.4.4.8	Setzen von Fenstereigenschaften einer Workstation .....	20
3.4.5	Zugriff auf Playback-Videostreams.....	21
3.4.5.1	getPlaybackList .....	21
3.4.5.2	getPlaybackVideo .....	22
3.4.5.3	getPlaybackVideoStillImage .....	22
3.4.5.4	getPlaybackVideoHtml .....	22
3.4.5.5	getPlaybackControlConnectionId.....	23
3.4.5.6	Start Playback .....	23
3.4.5.7	Stop Playback .....	23
3.4.5.8	schneller Vorlauf .....	24
3.4.5.9	schneller Rücklauf.....	24
3.4.5.10	StreamInfo.....	25
3.4.5.11	absolute Position setzen .....	25
3.4.6	Push-Mode / MIME-Type "multipart/x-mixed-replace".....	25
3.5	Beispiel Index.html.....	26

4	Support / Hotline.....	28
5	Abbildungsverzeichnis.....	29

# 1 Einleitung

---

## 1.1 Zweck des Dokumentes

Das vorliegende Dokument ist ein Teildokument der System Dokumentation für das Produkt **vimacc** der Accellence Technologies GmbH.

**vimacc** ist eine universelle Videomanagementsoftware zur Übertragung, Anzeige, Auswertung und Archivierung von Video-, Audiodaten und zugehörigen Metadaten sowie zur Steuerung der Video- und Audiotechnik wie z.B. Kameras und Schaltkontakten eines digital vernetzten CCTV-Systems.

Dieses Dokument beschreibt die Kommandos zur Steuerung eines **vimacc**-System inklusiv der angeschlossenen Kameras und Workstation-/Displayserver-Prozesse, die Möglichkeit des Abrufes von live- und playback-Video-Streams sowie der Anzeige der Video-Streams in einem WEB-Browser.

## 1.2 Aufbau der Dokumentation

Die **vimacc** System Dokumentation besteht aus einer Reihe von Dokumenten, die jeweils einen Teilaspekt behandeln und in sich abgeschlossen sind.

Die folgende Aufstellung beschreibt kurz die zur Verfügung stehenden Dokumente, die in ihrer Gesamtheit die **vimacc** Videomanagementsoftware beschreiben:

- **vimacc Systemdokumentation: Überblick**  
Dieses Dokument skizziert zunächst die Problemstellung eines heutigen digitalen Videoüberwachungssystems und leitet daraus die Notwendigkeit einer universellen Videomanagementsoftware her. Anschließend gibt es einen Überblick über die allgemeinen Eigenschaften von **vimacc** und zeigt einige sich daraus ergebenden möglichen Einsatzgebiete.
- **vimacc Systemdokumentation: Eigenschaften**  
Dieses Dokument liefert eine umfassende technische Leistungsbeschreibung der **vimacc** Videomanagementsoftware.
- **vimacc Systemdokumentation: Architektur**  
Dieses Dokument gibt einen detaillierten Einblick in die Architektur von **vimacc** und stellt die zur Verfügung stehenden Software Komponenten und ihre Funktionen vor.
- **vimacc Systemdokumentation: Schnittstellen**  
Dieses Dokument ist ein Teil der **vimacc** Schnittstellenbeschreibung.
- **vimacc Systemdokumentation: Systemvoraussetzungen**  
Dieses Dokument beschreibt die Minimalanforderungen an Hardware und Betriebssystem-Software, die erfüllt sein müssen, damit **vimacc** auf einer Hardware-Komponente installiert werden kann.
- **vimacc Systemdokumentation: Systemplanung**  
Dieses Dokument beschreibt die besonderen Randbedingungen, die bei der Planung eines modernen und vernetzten Videosystems zu berücksichtigen sind und kann somit einem Systemplaner als Hilfestellung dienen, ein **vimacc** System zu dimensionieren und zu planen.  
Darüber hinaus stellt es die zur Verfügung stehenden Software Editionen und deren Einsatzgebiete vor.

### 1.2.1 Dokumentenversionierung

Das Deckblatt des Dokumentes verweist mit 3 Stellen auf die Versionsnummer des **vimacc**-Steuerprotokolls.

Die letzte Stelle bezeichnet die Versionsnummer des Dokumentes.

(Bsp.: 1.0.0.2 -> Protokollversion 1.0.0, Dokumentrelease: 2)

## 2 Installation

Das Setup der **vimacc**-Editionen bietet die Möglichkeit, einen **vimacc** eigenen http-Server (WEB-Server) zu installieren. Bei entsprechender Auswahl durch den Benutzer wird der WEB-Server unter dem Namen *AccVimaccHttpServer* als Windows-Dienst installiert.

Standardmäßig wird Port 80 benutzt. Der Port kann im ConfigurationCenter individuell angepasst werden.

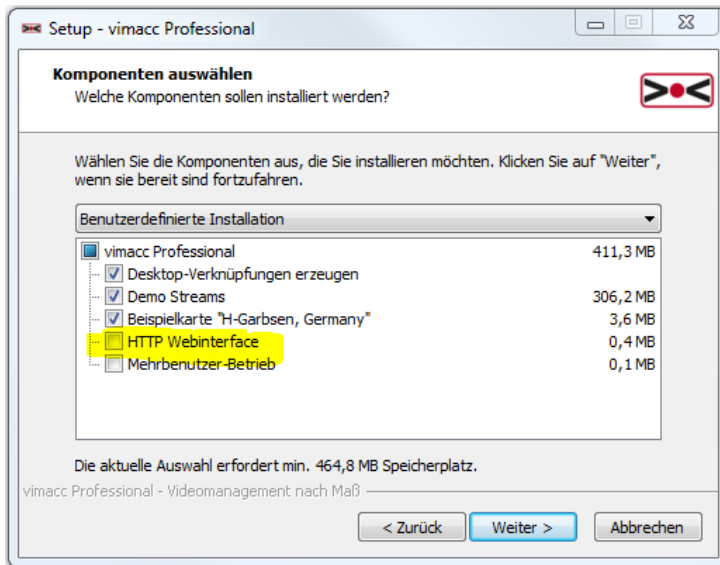


Abbildung 2-1: Installation http-Interface vimacc Professional

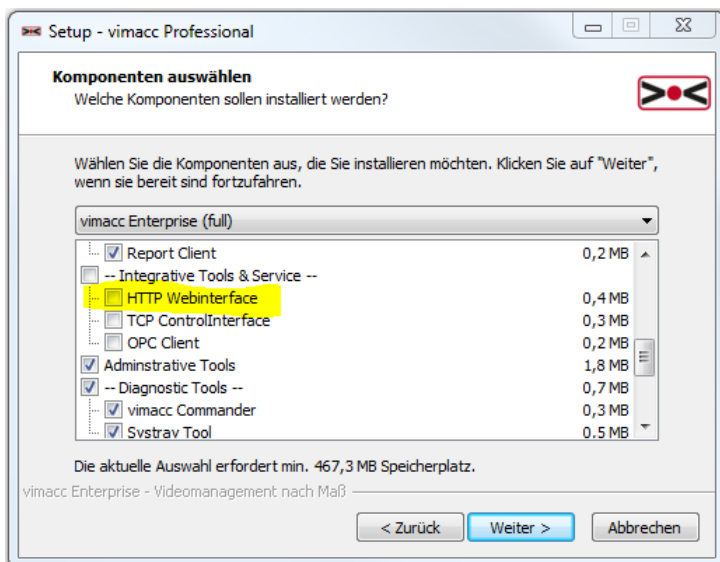


Abbildung 2-2: Installation http-Interface vimacc Enterprise



Der TCP-Port 80 darf nicht durch andere Programme belegt sein

# 3 AccVimaccHttpServer

Der AccVimaccHttpServer stellt Daten über die aktiven **vimacc**-Geräte, deren aktuellen Status, sowie die zugehörigen Video-Streams bereit.

Die Übertragung von Videodaten und Kommandos erfolgt mittels HTTP/1.0-Protokoll. Weitere Ports werden für die Übertragung nicht benötigt.



Bei Nutzung eines individuellen Port (abweichend von Port 80) muss dieser Port bei Anfragen an den Web-Server explizit angegeben werden!

Die jeweils im Originalformat gespeicherten Video-Streams werden in Einzelbilder zerlegt, in das Format JPEG transkodiert und übertragen. Der Abruf der Videodaten kann als JPEG (Einzelbild) oder MJPEG(Stream) erfolgen.

## 3.1 Verzeichnisstruktur `.\vimacc\data\www`

Bei der Installation einer **vimacc**-Edition wird folgende Verzeichnisstruktur angelegt:

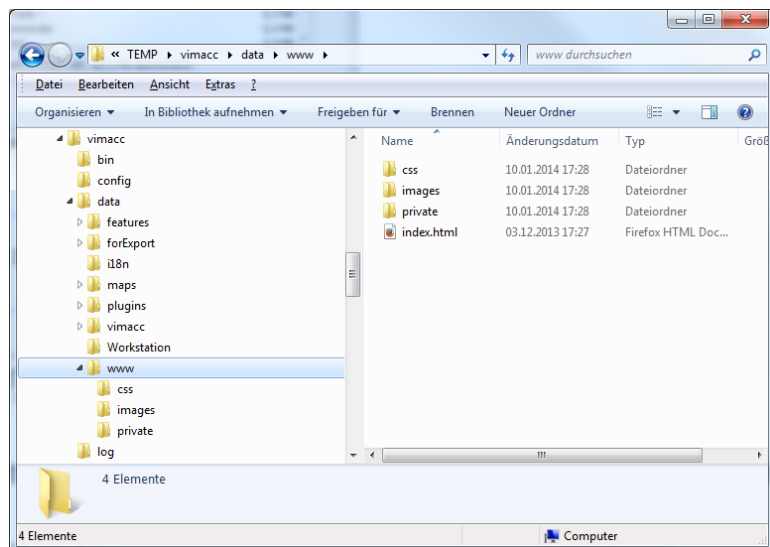


Abbildung 3-1: Verzeichnisstruktur

Der WEB-Sever liefert beim direkten Zugriff immer den Inhalt einer Datei `index.html`, soweit diese Datei vorhanden ist. Im Kapitel 3.5 finden sich weitere Informationen.

Struktur und Daten in diesem Verzeichnis können kundenseitig individuell angepasst werden.



## 3.2 Zugriff auf den vimacc http-Server

Der Zugriff auf den **vimacc** http-Server erfolgt über die URL

<http://ip-address/vimacc/>

Informationen zu Benutzer-Authentifizierungsmethoden finden sich in Kapitel 3.3 Authentifizierung.

Ein direkter Zugriff auf diese URL liefert den Inhalt der Datei index.html zurück, wenn diese Datei im Verzeichnis `.\vimacc\data\www` vorhanden ist.

Im Setup der **vimacc** -Editionen ist ein Beispiel enthalten (siehe Kapitel: 3.5).

Im Fehlerfall wird die Meldung „400 Bad Request“ angezeigt, gefolgt von einer Liste der in dieser Version erlaubten **vimacc** -Kommandos.

In der Kopfzeile sind die Protokollversion und die **vimacc** -Version angegeben.

Bei Fragen zu bestimmten Funktionen oder Erweiterungswünschen sind diese Versionsnummern anzugeben.

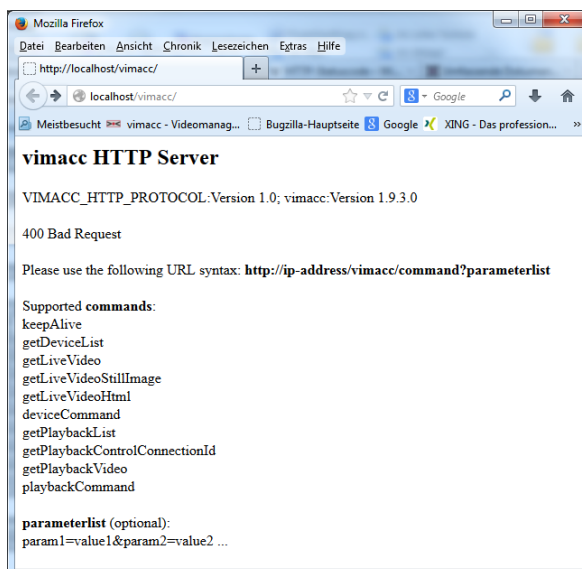


Abbildung 3-2: Fehler 400 Bad Request



Direkte Zugriffe auf die IP-Adresse ohne `/vimacc/` werden als Datei-Zugriffe auf das Verzeichnis `.\vimacc\data\www\` interpretiert.

Existiert die angefragte Datei, dann wird sie vom Server gesendet, alternativ erfolgt eine http-Fehlermeldung „404 not found“

Für den Zugriff auf Geräte- und Videofunktionen gilt folgendes Format:

`http://ip-address/vimacc/command?parameterlist`

Die Parameterliste variiert je nach Kommando und gewünschter Funktion.

## 3.3 Authentifizierung

Der vimacc-http-Server unterstützt neben dem kennwortfreien Zugriff zwei verschiedene Methoden zur Benutzerauthentifizierung gemäß RFC2617 (Basic und Digest-Authentication).

Nach der Installation ist die Benutzer-Authentifizierung für den Web-Server abgeschaltet. Im ConfigurationCenter kann die Authentifizierungsmethode entsprechend aktiviert werden.

### 3.3.1 Keine Authentifizierung

Definition-String im ConfigurationCenter:

```
control="VIMACC_HTTP_SERVER@0.0.0.0:80";authmode=none;maxconnections=100;jpegquality=75;function=CMS,HTTP_SERVER
```

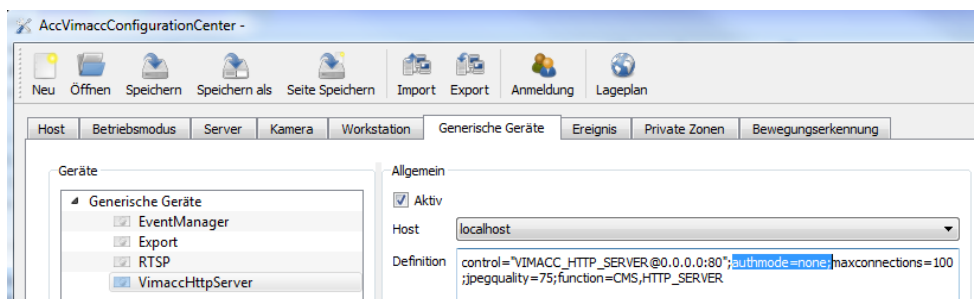


Abbildung 3-3: keine Authentifizierung

### 3.3.2 Basic Authentication

Der Web-Server fordert in diesem Fall eine Authentifizierung an. Bei Nutzung eines Browsers fordert dieser den Benutzer zur Eingabe von Benutzername und Passwort auf. Das Passwort wird Base64 codiert übertragen.

Definition-String im ConfigurationCenter:

```
control="VIMACC_HTTP_SERVER@0.0.0.0:80";authmode=basic;username=TestUser;userpwd=TestPassword;maxconnections=100;jpegquality=75;function=CMS,HTTP_SERVER
```

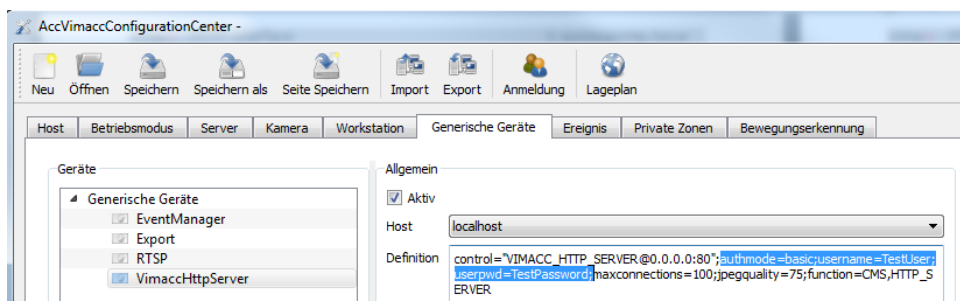


Abbildung 3-4: Basic Authentication

### 3.3.3 Digest Authentication

Der Web-Server fordert ebenfalls eine Authentifizierung für den Zugriff auf die Daten an. Das Passwort wird in diesem Fall als Hashwert übertragen.

Definition-String im ConfigurationCenter:

```
control="VIMACC_HTTP_SERVER@0.0.0.0:80";authmode=digest;username=TestUser;userpwd=TestPassword;maxconnections=100;jpegquality=75;function=CMS,HTTP_SERVER
```

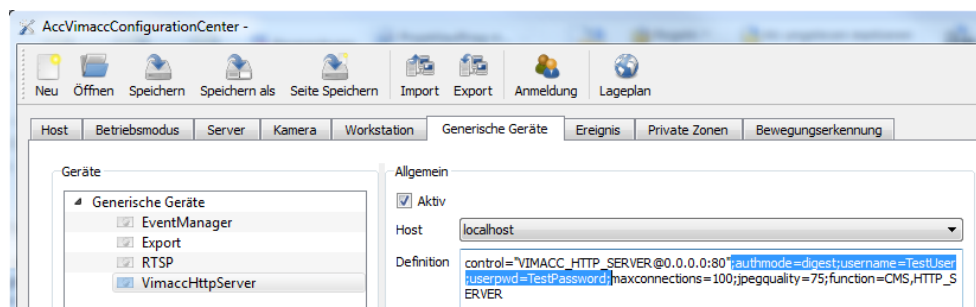


Abbildung 3-5: DIGEST Authentication

Zu Testzwecken kann im ConfigurationCenter ein zusätzlicher Parameter `serverchallenge=75798a683873f75071b7da939173f09a;` angegeben werden, um immer den gleichen Hashwert für das Kennwort zu erzeugen.

### 3.3.4 Weitere Parameter des vimacc Web-Servers

Definition-String im ConfigurationCenter:

```
control="VIMACC_HTTP_SERVER@0.0.0.0:80";authmode=none;maxconnections=100;jpegquality=75;function=CMS,HTTP_SERVER
```

<code>:80</code>	Angabe des Port (hier Port 80)
<code>maxconnections=100</code>	maximale Anzahl der gleichzeitigen Verbindungen zum http-Server
<code>jpegquality=75</code>	Qualität der JPEG-Bilder, wenn der Client keinen Qualitätsparameter gesetzt hat
<code>funktion=CMS, HTTP_SERVER</code>	gesetzte Funktionsgruppen, die vom Client gefiltert werden können

## 3.4 vimacc Kommandos

Alle vimacc Kommandos werden von einem Client (z.B. Browser) als Parameterpaare der HTTP GET-Methode an den **vimacc** WEB-Server gesendet und von diesem entsprechend quittiert.

Kommandos mit Rückgabewerten liefern zusätzlich die angeforderten Informationen.

Fehlermeldungen werden HTTP-konform gesendet.

### 3.4.1 keepAlive

Testen der Verbindung ohne weitere Reaktion.

In bestimmten Fällen kann es erforderlich sein, eine bedarfsorientierte Verbindung offen zu halten, ohne explizit Videodaten anzufordern.

Das Kommando keepAlive bietet eine Option, die Verbindung bei minimaler Bandbreite offen zu halten..

*http://localhost/vimacc/keepalive*

**erforderlicher Parameter:**

keine

**Antwortcode:**

200 OK

## 3.4.2 Zugriff auf live-Videostreams

### 3.4.2.1 getDeviceList

Abruf einer Liste der aktiven Geräte.

Beim Aufruf ohne optionale Parameter wird eine ungefilterte Liste aller aktiven Geräte im **vimacc** geliefert. In diesem Fall werden z.B. auch Workstation- und Display-Prozesse angezeigt.

Über optionale Parameter können Inhalt und Form der Übertragung entsprechend angepasst werden. Optionale Parameter werden über das Zeichen & verkettet.

*http://localhost/vimacc/getDeviceList*

*http://localhost/vimacc/getDeviceList?function=FIX&metainfo=1&streamformat=H264&responsetype=plain*

#### erforderlicher Parameter:

keine

#### optionale Parameter:

<i>function=FIX</i>	Filter auf die Funktionsgruppen. Es werden nur Kameras aus dieser Funktionsgruppe angezeigt. vordefinierte Funktionsgruppen für Kameras: <b>FIX, PTZ</b> Im ConfigurationCenter ( ->Tab:Kamera / Funktionsgruppen ) können weitere benutzerdefinierte Funktionsgruppen definiert werden. Mehrere Funktionsgruppen können kommasetrennt angegeben werden. (z.B.: FIX, PTZ)
<i>streamformat=H264</i>	Filter auf den Typ des Video-Streams (H.264, MJPEG, etc.) -> nicht zusammen mit „responsetype=html“ verwendbar!
<i>metainfo=1</i>	ggf. vorhandene Metainformationen werden zusätzlich übertragen
<i>responsetype=plain</i>	<b>plain</b> (default) je Gerät/Kamera wird eine Zeile übertragen <b>html</b> Übertragung als Liste im HTML-Format mit den Feldern „Kameraname“, „Funktion/Typ“, „Metadaten“ Die Kameranamen werden als HTML-Link übertragen

#### Beispiel Antwortdaten:

*id: "310"; name: "WS1"; streamingtype: "live"; mediatype: "none"; alarm: "false"; function: "WORKSTATION"; streamformat: "none"; control: "state=ok"; connexion: "state=off"; streaming: "state=off"*

*id: "CAM\_Axis206MJPEG\_0001"; name: "Axis 206 MJPEG"; streamingtype: "live"; mediatype: "video"; alarm: "false"; function: "FIX" streamformat: "H264";control: "state=off"; connexion: "state=off"; streaming: "video=ok"*

*id: "CAM\_AxisP5534vapix\_0001"; name: "Axis P5534 vapix"; streamingtype: "live"; mediatype: "video"; alarm: "false"; function: "PTZ"; streamformat: "H264";control: "state=ok"; connexion: "state=off"; streaming: "video=ok"*

*id: "CAM\_BoschAutodomeRCCP\_0001"; name: "Bosch Autodome RCCP"; streamingtype: "live"; mediatype: "video"; alarm: "false"; function: "PTZ"; streamformat: "any";control: "state=ok"; connexion: "state=off"; streaming: "video=ok"*

Je Gerät wird eine Zeile mit Parametern erzeugt, die durch ein Semikolon getrennt sind.

## Parameter am Beispiel der Kamera Axis P5534:

<i>id:</i>	<b><i>CAM_AxisP5534vapix_0001</i></b> eindeutige Datenbank-Id des Videostreams
<i>name:</i>	<b><i>Axis P5534 vapix</i></b> Name der Kamera. Eine Kamera kann mehrere Streams haben.
<i>streamingtype:</i>	<b><i>live</i></b> Typ des Videostreams (live, playback)
<i>mediatype:</i>	<b><i>video</i></b> Typ des Mediums
<i>alarm:</i>	<b><i>false</i></b> Alarmfunktion des Streams false                               -> keine Alarmfunktion true                                   -> Stream hat Alarmfunktion
<i>function:</i>	<b><i>PTZ</i></b> Funktionsgruppen des Streams (siehe ConfigurationCenter) vordefinierte Funktionsgruppen: <b>PTZ, FIX</b>
<i>streamformat:</i>	<b><i>H264</i></b> Format des Videostreams (siehe ConfigurationCenter) H264, MJPEG                       -> Standardformate ANY                                   -> spezielle Formate (z.B. RCPP, USB-Cams) NONE                                  -> kein Videostream (z.B. Workstation)
<i>control:</i>	<b><i>state=off</i></b> Status der Controlverbindung (z.B. PTZ-Steuerung) state=off                           -> nicht benutzt state=ok                             -> Status OK state=disconnected               -> Verbindung unterbrochen
<i>connexion:</i>	<b><i>state=off</i></b> Status einer weiter geleiteten Verbindung state=off                           -> nicht benutzt state=ok                             -> Status OK state=disconnected               -> Verbindung unterbrochen
<i>streaming:</i>	<b><i>video=ok</i></b> Status Videostreaming video=OK                           -> OK Video=no rtp, no rtcp             -> Fehler

### 3.4.2.2 getLiveVideo

Abruf eines Videostreams vom http-Server als MJPEG

Über den geöffneten Kommunikationsport werden im entsprechenden Zeitraster des Originalvideos JPEG-Bilder im Push-Mode (siehe Kapitel 3.4.6 ) übertragen.

*http://localhost/vimacc/getLiveVideo?id=CAM\_Autobahn1\_0001&jpegquality=50*

**erforderlicher Parameter:**

*id=CAM\_Autobahn1\_0001* Datenbank-ID des Videostreams

**optionaler Parameter:**

*jpegquality=50* Qualität der JPEG-Dateien (1: niedrig, 100: beste)  
Default, wenn nicht angegeben: 75

### 3.4.2.3 getLiveVideoStillImage

Abruf eines Einzelbildes

*http://localhost/vimacc/getLiveVideoStillImage?id=CAM\_Autobahn1\_0001&jpegquality=50*

**erforderlicher Parameter:**

*id=CAM\_Autobahn1\_0001* Datenbank-ID des Videostreams

**optionaler Parameter:**

*jpegquality=50* Qualität der JPEG-Dateien (1: niedrig, 100: beste)  
Default, wenn nicht angegeben: 75

### 3.4.2.4 getLiveVideoHtml

Abruf einer HTML-Seite mit einem Videostreams als MJPEG

Über den geöffneten Kommunikationsport werden im entsprechenden Zeitraster des Originalvideos JPEG-Bilder im Push-Mode (siehe Kapitel 3.4.6 ) übertragen. Zusätzlich werden bei PTZ-Kameras Buttons zur Steuerung der Kamerafunktionen (pan/tilt/zoom) übertragen, über die die Kamera gesteuert werden kann. Die Steuerkommandos werden separat beschrieben (siehe 3.4.3 Steuerung PTZ-Kameras).

*http://localhost/vimacc/getLiveVideoHtml?id=CAM\_Autobahn1\_0001&jpegquality=50*

**erforderlicher Parameter:**

*id=CAM\_Autobahn1\_0001* Datenbank-ID des Videostreams

**optionaler Parameter:**

*jpegquality=50* Qualität der JPEG-Dateien (1: niedrig, 100: beste)  
Default, wenn nicht angegeben: 75

## 3.4.3 Steuerung PTZ-Kameras

### 3.4.3.1 Aktion STOP

Alle nachfolgenden Kommandos (siehe 3.4.3.2 ff.) werden abgebrochen.

```
http://localhost/vimacc/devicecommand?id=cam_axisp5534_0001&commandparams=cmd\\=move;stop
```

**erforderlicher Parameter:**

*id= cam\_axisp5534\_0001*                      Datenbank-ID des Videostreams

**Antwortcode:**

200 OK

### 3.4.3.2 Kamera nach unten

Kamera nach unten schwenken. Das Kommando muss mit STOP beendet werden. Je nach Modell und Firmware unterstützen nicht alle Kameras die Übergabe eines Geschwindigkeitsparameters.

```
http://localhost/vimacc/devicecommand?id=cam_axisp5534_0001&commandparams=cmd\\=move;down\\=1
```

**erforderlicher Parameter:**

*id= cam\_axisp5534\_0001*                      Datenbank-ID des Videostreams  
*down\\=1*                                      Geschwindigkeit der Bewegung (1..100)

**Antwortcode:**

200 OK

### 3.4.3.3 Kamera nach oben

Kamera nach oben schwenken. Das Kommando muss mit STOP beendet werden. Je nach Modell und Firmware unterstützen nicht alle Kameras die Übergabe eines Geschwindigkeitsparameters.

```
http://localhost/vimacc/devicecommand?id=cam_axisp5534_0001&commandparams=cmd\\=move;up\\=1
```

**erforderlicher Parameter:**

*id= cam\_axisp5534\_0001*                      Datenbank-ID des Videostreams  
*up\\=1*                                        Geschwindigkeit der Bewegung (1..100)

**Antwortcode:**

200 OK

### 3.4.3.4 Kamera nach links

Kamera nach oben schwenken. Das Kommando muss mit STOP beendet werden. Je nach Modell und Firmware unterstützen nicht alle Kameras die Übergabe eines Geschwindigkeitsparameters.

```
http://localhost/vimacc/devicecommand?id=cam_axisp5534_0001&commandparams=cmd\\=move;left\\=1
```



**erforderlicher Parameter:**

*id= cam\_axisp5534\_0001* Datenbank-ID des Videostreams  
*left\|=1* Geschwindigkeit der Bewegung (1..100)

**Antwortcode:**

200 OK

### 3.4.3.5 Kamera nach rechts

Kamera nach oben schwenken. Das Kommando muss mit STOP beendet werden. Je nach Modell und Firmware unterstützen nicht alle Kameras die Übergabe eines Geschwindigkeitsparameters.

*http://localhost/vimacc/devicecommand?id=cam\_axisp5534\_0001&commandparams=cmd\|=move;right\|=1*

**erforderlicher Parameter:**

*id= cam\_axisp5534\_0001* Datenbank-ID des Videostreams  
*right\|=1* Geschwindigkeit der Bewegung (1..100)

**Antwortcode:**

200 OK

### 3.4.3.6 Kamera hinein zoomen

Das Kommando muss mit STOP beendet werden. Je nach Modell und Firmware unterstützen nicht alle Kameras die Übergabe eines Geschwindigkeitsparameters.

*http://localhost/vimacc/devicecommand?id=cam\_axisp5534\_0001&commandparams=cmd\|=move;zoomin\|=1*

**erforderlicher Parameter:**

*id= cam\_axisp5534\_0001* Datenbank-ID des Videostreams  
*zoomin\|=1* Geschwindigkeit der Bewegung (1..100)

**Antwortcode:**

200 OK

### 3.4.3.7 Kamera heraus zoomen

Das Kommando muss mit STOP beendet werden. Je nach Modell und Firmware unterstützen nicht alle Kameras die Übergabe eines Geschwindigkeitsparameters.

*http://localhost/vimacc/devicecommand?id=cam\_axisp5534\_0001&commandparams=cmd\|=move;zoomout\|=1*

**erforderlicher Parameter:**

*id= cam\_axisp5534\_0001* Datenbank-ID des Videostreams  
*zoomout\|=1* Geschwindigkeit der Bewegung (1..100)

**Antwortcode:**

200 OK

## 3.4.4 Steuerung Workstation/Display/Videowand

### 3.4.4.1 Video auf einen Videodialog aufschalten

Aufschalten eines Video-Streams auf einen Videodialog einer Workstation oder eines Displays einer Videowand.

```
http://localhost/vimacc/devicecommand?id=310&commandparams=cmd\|=show;source\|=cam_autobahn1_0001;videodlg\|=1
```

**erforderlicher Parameter:**

*id=310*

Datenbank-ID der Workstation(Display), auf die der Stream aufgeschaltet werden soll

*source\|=cam\_autobahn1\_0001*

Datenbank-ID des Video-Streams

*videodlg\|=1*

Nummer des Videodialoges

**Antwortcode:**

200 OK

### 3.4.4.2 Löschen eines Videodialogs

Löschen eines spezifischen Videodialogs oder aller Videodialoge einer Workstation oder eines Displays einer Videowand.

```
http://localhost/vimacc/devicecommand?id=310&commandparams=cmd\|=clear;videodlg\|=VD1
```

**erforderlicher Parameter:**

*id=310*

Datenbank-ID der Workstation(Display), auf der gelöscht werden soll

*videodlg\|=VD1*

Nummer des Videodialoges

VD1 – VD24: Löschen eines einzelnen Videodialoges

0: Löschen aller Videodialoge dieser Workstation-Instanz

**Antwortcode:**

200 OK

### 3.4.4.3 Setzen eines definierten Grid

Setzen eines definierten Grids als Matrix von Videodialogen für eine Workstation oder ein Display einer Videowand.

Im Fall, dass die gewünschte Matrix nicht existiert, wird eine Anordnung gewählt, die die Gesamtanzahl der gewünschten Videodialoge anzeigen kann.

Beispiel: rows=10, cols=1 -> Setzen einer Matrix 4x3 (12 Videodialoge)

```
http://localhost/vimacc/devicecommand?id=310&commandparams=cmd\|=setgrid;rows\|=6;cols\|=4
```

**erforderlicher Parameter:**

*id=310*

Datenbank-ID der Workstation(Display), auf der das Grid gesetzt werden soll

*rows\|=6*

Anzahl der Spalten ( 1–6 )

*cols\|=4*

Anzahl der Zeilen ( 1-4 )

**Antwortcode:**

200 OK

### 3.4.4.4 Setzen eines Grids für eine Anzahl Videodialoge

Es wird eine Matrix gewählt, die die gewünschte Anzahl Videodialoge anzeigen kann.

*http://localhost/vimacc/devicecommand?id=310&commandparams=cmd\|=setgrid;dialogcount\|=24*

**erforderlicher Parameter:**

*id=310* Datenbank-ID der Workstation(Display), auf der das Grid gesetzt werden soll  
*dialogcount\|=24* Anzahl der Videodialoge ( 1–24 )

**Antwortcode:**

200 OK

### 3.4.4.5 Aufschalten eines Szenarios

Aufschalten eines Szenarios auf eine Workstation oder ein Display einer Videowand. Der vorherige Zustand wird überschrieben.

*http://localhost/vimacc/devicecommand?id=310&commandparams=cmd\|=showconfiguredscenario;scenario\|=  
 {353ec3fb-de1b-4a2c-a875-f600804e8a60};contextid\|=310*

oder

*http://localhost/vimacc/devicecommand?id=310&commandparams=cmd\|=showconfiguredscenario;scenario\|=  
 QuadGrid;contextid\|=context*

**erforderlicher Parameter:**

*id=310* Datenbank-ID der Workstation(Display), auf der das Szenario ausgeführt werden soll  
*scenario\|=  
 {353ec3fb-de1b-4a2c-a875-f600804e8a60}*  
*scenario\|=QuadGrid* Datenbank-ID oder Name des Szenarios  
*contextid\|=context* Kontextname

**Antwortcode:**

200 OK

### 3.4.4.6 Anzeigemodus setzen

In der Darstellung der Videos kann zwischen 3 verschiedenen Modi gewählt werden.

*http://localhost/vimacc/devicecommand?id=310&commandparams=cmd\|=scalemode;value\|=KEEP*

**erforderlicher Parameter:**

*id=310* Datenbank-ID der Workstation(Display), auf der das Szenario ausgeführt werden soll  
*value\|=KEEP* Anzeigemodus. (Parameter ist case sensitive)  
**KEEP** Anzeige des Original-Bildseitenverhältnisses  
**FIT** Anzeige ohne schwarze Ränder/Streifen  
 Das Bild wird gestreckt, um den gesamten Raum des Videodialogs zu nutzen. Es gehen keine Bildinformationen verloren.  
**ZOOM** Anzeige ohne schwarze Ränder/Streifen  
 Das Bild wird in beiden Richtungen vergrößert, bis keine schwarzen Flächen mehr vorhanden sind.  
 Es gehen Bildinformationen bei der Anzeige verloren.

**Antwortcode:**

200 OK

### 3.4.4.7 Aufschalten einer Sequenz

Eine Sequenz wird auf einen Videodialog einer Workstation oder eines Displays einer Videowand aufgeschaltet.

Der Zustand des Videodialoges wird überschrieben.

```
http://localhost/vimacc/devicecommand?id=310&commandparams=cmd\|=showconfiguredsequence;
sequence\|={3f465281-20b7-4d1a-8cd1-836383581241};contextid\|=WS1;videodlg\|=VD1
```

oder

```
http://localhost/vimacc/devicecommand?id=310&commandparams=cmd\|=showconfiguredsequence;
sequence\|=Autobahn;contextid\|=WS1;videodlg\|=VD1
```

#### erforderlicher Parameter:

<i>id=310</i>	Datenbank-ID der Workstation(Display), auf der das Szenario ausgeführt werden soll
<i>sequence\ ={3f465281-20b7-4d1a-8cd1-836383581241}</i> oder <i>sequence\ =Autobahn</i>	Datenbank-ID oder Name des Szenarios
<i>contextid\ =context</i>	Kontextname
<i>videodlg\ =VD1</i>	Name des Videodialogs ( VD1-VD24 )

#### Antwortcode:

200 OK

### 3.4.4.8 Setzen von Fenstereigenschaften einer Workstation

Das Kommando dient dazu, die Größe, Position und Ansicht von Workstation-Fenstern zu beeinflussen.

Es wirkt gleichermaßen auf Display-Prozesse einer Videowand.

```
http://localhost/vimacc/devicecommand?id=310&commandparams=cmd\|=setgeometry;hidden\|=0;showframe\|=1;shownavigation\|=1;showcontrols\|=1;topmost\|=0;title\|=Workstation01;xpos\|=0;ypos\|=0;width\|=200;height\|=100;contextid\|=context
```

#### erforderlicher Parameter:

<i>id=310</i>	Datenbank-ID der Workstation(Display), die gesteuert werden soll
---------------	--

#### optionale Parameter:

<i>hidden\ =0</i>	<b>0</b> (default) Workstation wird angezeigt <b>1</b> Workstation wird nicht angezeigt
<i>showframe\ =1</i>	<b>0</b> Rahmen des Windows wird nicht angezeigt <b>1</b> (default) Rahmen des Windows wird angezeigt
<i>shownavigation\ =1</i>	<b>0</b> Navigationsbereich wird nicht angezeigt <b>1</b> Navigationsbereich wird angezeigt
<i>showcontrols\ =1</i>	<b>0</b> Bereich der PTZ- bzw. Playback-Controls wird nicht angezeigt <b>1</b> Bereich der PTZ- bzw. Playback-Controls wird angezeigt
<i>topost\ =0</i>	<b>0</b> (default) Workstation-Fenster kann in den Hintergrund <b>1</b> Workstation-Fenster ist immer topmost und wird immer angezeigt
<i>title\ =Workstation01</i>	setzt den Titel des Workstation-Fensters
<i>xpos\ =0;ypos\ =0</i>	Position der linken oberen Ecke des Workstation-Fensters in Pixel Beide Werte müssen >= 0 sein
<i>width\ =200;height\ =100</i>	Größe des Workstation-Fensters in Pixel Beide Werte müssen >=0 sein
<i>contextid\ =context</i>	Kontextname

#### Antwortcode:

200 OK

## 3.4.5 Zugriff auf Playback-Videostreams

### 3.4.5.1 getPlaybackList

Abruf einer Liste der verfügbaren Aufzeichnungen (playbacks).

Beim Aufruf ohne weitere Parameter wird eine ungefilterte Liste aller verfügbaren Aufzeichnungen im vimacc geliefert.

`http://localhost/vimacc/getPlaybackList`

`http://localhost/vimacc/getPlaybackList?function=FIX&metainfo=1&responsetype=plain`

#### erforderlicher Parameter:

keine

#### optionale Parameter:

`function=FIX`

Filter auf die Funktionsgruppen. Es werden nur Kameras aus dieser Funktionsgruppe angezeigt.

vordefinierte Funktionsgruppen für Kameras: **FIX**, **PTZ**

Im ConfigurationCenter ( ->Tab:Kamera / Funktionsgruppen ) können weitere benutzerdefinierte Funktionsgruppen definiert werden.

Mehrere Funktionsgruppen müssen kommagetrennt angegeben werden. (z.B.: FIX, PTZ)

`metainfo=1`

ggf. vorhandene Metainformationen werden zusätzlich übertragen

`responsetype=plain`

**plain** (default) je Gerät/Kamera wird eine Zeile übertragen

**html** Übertragung als Liste im HTML-Format mit den Feldern

„Kameraname“, „Funktion/Typ“, „Metadaten“

Die Kameranamen werden als HTML-Link übertragen

#### Beispiel Antwortdaten:

`id: "cam_autobahn1_0001"; name: "Autobahn1"; streamingtype: "playback"; mediatype: "video"; function: "FIX"; control: "state=ok"; streaming: "state=off"`

`id: "cam_autobahn2_0001"; name: "Autobahn2"; streamingtype: "playback"; mediatype: "video"; function: "FIX"; control: "state=ok"; streaming: "state=off"`

`id: "cam_axis206mjpeg_0001"; name: "Axis 206 MJPEG"; streamingtype: "playback"; mediatype: "video"; function: "FIX"; control: "state=ok"; streaming: "state=off"`

`id: "cam_axisp5534vapix_0001"; name: "Axis P5534"; streamingtype: "playback"; mediatype: "video"; function: "PTZ"; control: "state=ok"; streaming: "state=off"`

`id: "cam_maneki-neko_0002"; name: "Maneki-neko"; streamingtype: "playback"; mediatype: "video"; function: "FIX"; control: "state=ok"; streaming: "state=off"`

`id: "cam_stadtbahn_0001"; name: "Stadtbahn"; streamingtype: "playback"; mediatype: "video"; function: "FIX"; control: "state=ok"; streaming: "state=off"`

`id: "cam_tankstelle_0001"; name: "Tankstelle"; streamingtype: "playback"; mediatype: "video"; function: "FIX"; control: "state=ok"; streaming: "state=off"`

`id: "cam_teststream_0001"; name: "Teststream"; streamingtype: "playback"; mediatype: "video"; function: "FIX"; control: "state=ok"; streaming: "state=off"`

`id: "cam_werkschutz1_0001"; name: "Werkschutz1"; streamingtype: "playback"; mediatype: "video"; function: "FIX"; control: "state=ok"; streaming: "state=off"`

`id: "cam_werkschutz2_0001"; name: "Werkschutz2"; streamingtype: "playback"; mediatype: "video"; function: "FIX"; control: "state=ok"; streaming: "state=off"`

Je Gerät wird eine Zeile mit Parametern erzeugt, die durch Semikolon getrennt sind.

Parameter am Beispiel der Kamera Axis P5534:

`id:` **CAM\_AxisP5534\_0001**  
eindeutige Datenbank-Id des Videostreams

`name:` **Axis P5534**  
Name der Kamera. Eine Kamera kann mehrere Streams haben.

`streamingtype:` **playback**  
Typ des Video-Streams (live, playback)

`mediatype:` **video**  
Typ des Mediums

<i>function:</i>	<b>PTZ</b> Funktionsgruppen des Streams (siehe ConfigurationCenter) vordefinierte Funktionsgruppen: PTZ, FIX
<i>control:</i>	<b>state=off</b> Status der Controlverbindung (z.B. PTZ-Steuerung) state=off                               -> nicht benutzt state=ok                                 -> Status OK state=disconnected                   -> Verbindung unterbrochen
<i>streaming:</i>	<b>state=off</b> keine Funktion

### 3.4.5.2 getPlaybackVideo

Abruf eines aufgezeichneten Playback-Video-Streams als MJPEG

Über den geöffneten Kommunikationsport werden im entsprechenden Zeitraster des Originalvideos JPEG-Bilder im Push-Mode (siehe Kapitel 3.4.6 ) übertragen.

*http://localhost/vimacc/getPlaybackVideo?id=cam\_autobahn1\_0001&jpegquality=50*

**erforderlicher Parameter:**

*id=CAM\_Autobahn1\_0001*                   Datenbank-ID des Videostreams

**optionaler Parameter:**

*jpegquality=50*                         Qualität der JPEG-Dateien (1: niedrig, 100: beste)  
Default, wenn nicht angegeben: 50

### 3.4.5.3 getPlaybackVideoStillImage

Abruf eines Einzelbildes

*http://localhost/vimacc/getPlaybackVideoStillImage?id=cam\_autobahn1\_0001&jpegquality=50*

**erforderlicher Parameter:**

*id=CAM\_Autobahn1\_0001*                   Datenbank-ID des Video-Streams

**optionaler Parameter:**

*jpegquality=50*                         Qualität der JPEG-Dateien (1: niedrig, 100: beste)  
Default, wenn nicht angegeben: 50

### 3.4.5.4 getPlaybackVideoHtml

Abruf einer HTML-Seite mit einem aufgezeichneten Playback-Video-Streams als MJPEG.

Über den geöffneten Kommunikationsport werden im entsprechenden Zeitraster des Originalvideos JPEG-Bilder im Push-Mode (siehe Kapitel 3.4.6 ) übertragen.

Zusätzlich werden bei Playbacks Buttons mit den Funktionen „schneller Rücklauf, Pause, Start, schneller Vorlauf“ übertragen, mit denen das Playback gesteuert werden kann. Die Steuerkommandos werden separat beschrieben.

*http://localhost/vimacc/getLiveVideoHtml?id=CAM\_Autobahn1\_0001&jpegquality=50*

**erforderlicher Parameter:**

*id=CAM\_Autobahn1\_0001*      Datenbank-ID des Videostreams

**optionaler Parameter:**

*jpegquality=50*      Qualität der JPEG-Dateien (1: niedrig, 100: beste)  
Default, wenn nicht angegeben: 50

### 3.4.5.5 getPlaybackControlConnectionId

Beim Abruf von Playback-Daten ist es erforderlich, dass zwischen Client und Host eine eindeutige Connection-ID vereinbart wird, damit auch nach Unterbrechung einer aktiven Verbindung immer wieder auf das korrekte Playback zugegriffen werden kann. Der Parameter wird beim Aufruf von playbackCommands benötigt (siehe: 3.4.5.6 ff)

*http://192.168.200.67/vimacc/getPlaybackControlConnectionId?id=ws1*

**erforderlicher Parameter:**

*id=310*      Verbindungs-ID (wahlfrei festlegbar)

**Antwortcode:**

*playbackcontrolconnectionid: 1389978615455*

### 3.4.5.6 Start Playback

Startet einen Playbackstream

*http://192.168.200.67/vimacc/playbackcommand?id=cam\_autobahn1\_0001&controlconnectionid=1389978615455&commandparams=cmd\|=start&unique\|=0.159209356824763*

**erforderlicher Parameter:**

*id= cam\_autobahn1\_0001*      Datenbank-ID des Video-Streams  
*controlconnectionid\|=1389978615455*      ControlConnectionID (siehe 3.4.5.5)

**Antwortcode:**

200 OK

### 3.4.5.7 Stop Playback

Pause für einen laufenden Playback-Stream

*http://192.168.200.67/vimacc/playbackcommand?id=cam\_autobahn1\_0001&controlconnectionid=1389978615455&commandparams=cmd\|=pause&unique\|=0.159209356824763*

**erforderlicher Parameter:**

*id= cam\_autobahn1\_0001*      Datenbank-ID des Video-Streams  
*controlconnectionid\|=1389978615455*      ControlConnectionID (siehe 3.4.5.5)

**Antwortcode:**

200 OK

### 3.4.5.8 schneller Vorlauf

Start schneller Vorlauf mit Angabe der Geschwindigkeit

*http://192.168.200.67/vimacc/playbackcommand?id=cam\_autobahn1\_0001&controlconnectionid=1389978615455&commandparams=cmd\|=setspeed;value\|=400&unique\|=0.159209356824763*

**erforderlicher Parameter:**

*id= cam\_autobahn1\_0001*

*controlconnectionid\|=1389978615455*

*value\|=400*

Datenbank-ID des Video-Streams

ControlConnectionID (siehe 3.4.5.5)

Geschwindigkeit ( >0 ... 400 )

**Antwortcode:**

200 OK

### 3.4.5.9 schneller Rücklauf

Start schneller Rücklauf mit Angabe der Geschwindigkeit

*http://192.168.200.67/vimacc/playbackcommand?id=cam\_autobahn1\_0001&controlconnectionid=1389978615455&commandparams=cmd\|=setspeed;value\|=-400&unique\|=0.159209356824763*

**erforderlicher Parameter:**

*id= cam\_autobahn1\_0001*

*controlconnectionid\|=1389978615455*

*value\|=400*

Datenbank-ID des Video-Streams

ControlConnectionID (siehe 3.4.5.5)

Geschwindigkeit ( <0 ... -400 )

**Antwortcode:**

200 OK



### 3.4.5.10 StreamInfo

Gibt Informationen über diese Playbackdaten zurück:

- Beginn des Archivstreams
- aktuelle Position
- Ende des Archivstreams

```
http://192.168.200.67/vimacc/playbackcommand?id=cam_autobahn1_0001&controlconnectionid=1389978615455&commandparams=cmd||=getstreaminfo&unique||=0.159209356824763
```

**erforderlicher Parameter:**

<code>id= cam_autobahn1_0001</code>	Datenbank-ID des Video-Streams
<code>controlconnectionid=1389978615455</code>	ControlConnectionID (siehe 3.4.5.5)

**Antwortcode:**

```
streambeginn=2013-01-17T17:54:11:123;currentstreampos=2013-01-17T18:11:22.123;streamend=2013-01-17T18:50:13:123
```

### 3.4.5.11 absolute Position setzen

Setzt den Stream auf eine absolute Zeitposition

```
http://192.168.200.67/vimacc/playbackcommand?id=cam_axisp5534vapix_0001&controlconnectionid=1389980953715&commandparams=cmd||=setabsoluteposition;value||=2013-01-17T18:12:11.278&unique||=0.159209356824763
```

**erforderlicher Parameter:**

<code>id= cam_autobahn1_0001</code>	Datenbank-ID des Video-Streams
<code>controlconnectionid=1389978615455</code>	ControlConnectionID (siehe 3.4.5.5)
<code>value  =2013-01-17T18:12:11.278</code>	absolute Zeitposition

**Antwortcode:**

200 OK

## 3.4.6 Push-Mode / MIME-Type "multipart/x-mixed-replace"

Bei der Übertragung im Push-Mode wird der spezielle MIME-Type "*multipart/x-mixed-replace*" verwendet. Der Server sendet in diesem Fall kontinuierlich JPEG-komprimierte Bilder in der angegebenen Qualität. Die Trennung der einzelnen Bilder erfolgt dabei mit dem „*boundary=vimaccjpeg*“.

```
HTTP/1.0 200 OK
Content-Type: multipart/x-mixed-replace;boundary=vimaccjpeg
```

```
--vimaccjpeg
Content-Type: image/jpeg
Content-Length: 67856
```

```
<binary JPEG image data>
--vimaccjpeg
Content-Type: image/jpeg
Content-Length: 52856
```

```
<binary JPEG Image data>
--vimaccjpeg
...
```

## 3.5 Beispiel Index.html

Im Standard-Setup aller vimacc-Editionen ist ein Verzeichnis `.\www` vorhanden, welches bei entsprechender Auswahl gemeinsam mit dem vimacc Web-Server installiert wird.

Beim Zugriff auf dem Web-Server mit einem Internet-Browser (InternetExplorer, Safari, etc.) wird die integrierte Musterdatei `index.html` geladen und im Browser angezeigt (siehe Abbildung 3-6: Ansicht Beispiel-index.html ).

Die Zugriffsmöglichkeiten auf live- und playback-Videos wird beispielhaft demonstriert.

Die Gestaltung der Website kann auf Basis dieses Beispiels kundenindividuell vorgenommen werden.



Serverseitige Script-Sprachen werden nicht unterstützt.

Die mitgelieferte Datei `index.html` lädt aus dem laufenden vimacc-System automatisch die aktuellen Listen für live- und playback-Kameras und erzeugt Links für die Zugriffe auf die jeweiligen Video-Streams.

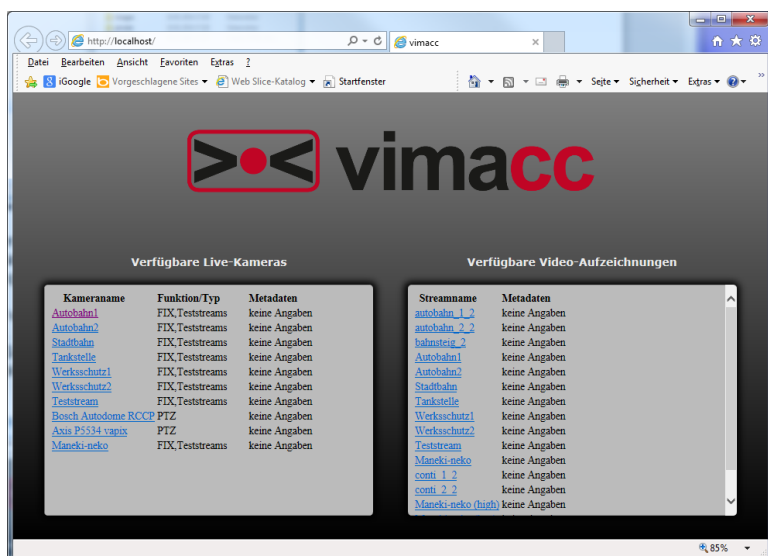


Abbildung 3-6: Ansicht Beispiel-index.html

Beim Klick auf den Link eines Video-Streams wird dieser in einem neuen Browser-Tab angezeigt. (siehe Abbildung 3-7 und Abbildung 3-8)

Je nach Typ des Streams werden verschiedene Bedienelemente angeboten.

Bei live-Kameras mit PTZ-Funktionen werden zusätzlich zum Videobild PTZ-Steuerelemente angeboten. Playback-Videos besitzen Steuerelemente für die Playbacksteuerung und Zeitinformationen für Start-, aktuelle und End-Zeit des Playback-Streams.

Beim Klick auf die Steuerelemente wird die entsprechende Funktion ausgeführt. PTZ-Kommandos müssen mit dem STOPP-Button angehalten werden.



Abbildung 3-7: PTZ-Kamera

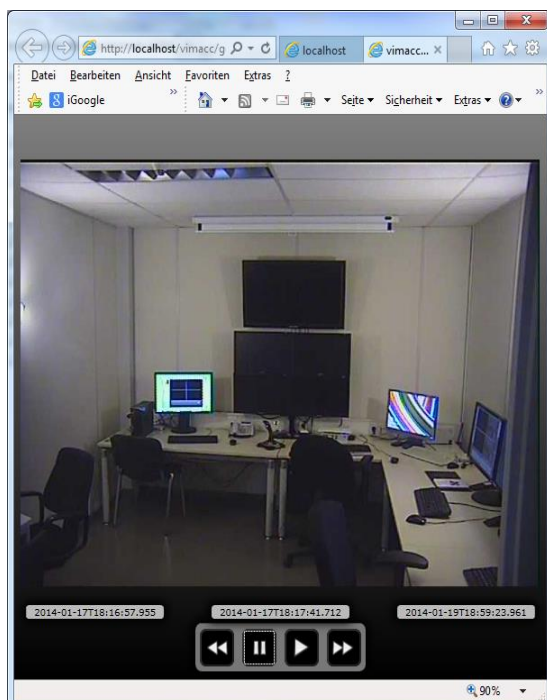


Abbildung 3-8: Playback-Steuerung

# 4 Support / Hotline

---

Haben Sie noch Fragen zu **vimacc**?

Dann kontaktieren Sie unsere Hotline:

- per Email [support@accelence.de](mailto:support@accelence.de)  
oder
- telefonisch unter **+49 (0)511 277 2490**

Unsere Mitarbeiter stehen Ihnen Werktags von 9:00-17:00 Uhr gerne zur Verfügung

# 5 Abbildungsverzeichnis

---

Abbildung 2-1: Installation http-Interface vima <sup>cc</sup> Professional.....	7
Abbildung 2-2: Installation http-Interface vima <sup>cc</sup> Enterprise .....	7
Abbildung 3-1: Verzeichnisstruktur .....	8
Abbildung 3-2: Fehler 400 Bad Request.....	9
Abbildung 3-3: keine Authentifizierung .....	10
Abbildung 3-4: Basic Authentication .....	10
Abbildung 3-5: DIGEST Authentication .....	11
Abbildung 3-6: Ansicht Beispiel-index.html.....	26
Abbildung 3-7: PTZ-Kamera .....	27
Abbildung 3-8: Playback-Steuerung .....	27