U+ VA API

2022-05-11

Ver 1.0.2

|  |  |  |  |
| --- | --- | --- | --- |
| 버전 | 작성일 | 내용 | 비고 |
| 1.0.0 | 2022-03-17 | - |  |
| 1.0.1 | 2022-04-08 | 2.1 리소스 정의 수정 |  |
| 1.0.2 | 2022-05-11 | 3.2/3.3 룰 유형/객체 유형 추가  4.2. icon 정의 수정  8.1. Meta-Alarm 정의 수정 | 넥스트케이, 아이브스 반영  8.1. count/level 추가 |
| 1.0.3 | 2022-05-19 | 3.2. 룰 유형 추가  4.1. 시스템 리소스 추가  6.1. 엔진 리소스 추가 | 아이브스 룰 유형 추가 반영  4.1. guid, engine\_count 추가  6.1. engine\_name 추가 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

목차

[1. 개요. 4](#_Toc102131454)

[2. 아키텍처 4](#_Toc102131455)

[2.1. 리소스 4](#_Toc102131456)

[2.2. CRUD 5](#_Toc102131457)

[2.3. 요청 응답 5](#_Toc102131458)

[2.4. HATEOAS 6](#_Toc102131459)

[2.5. FLOW 7](#_Toc102131460)

[3. 코드 및 좌표 정의 7](#_Toc102131461)

[3.1. HTTP 상태 7](#_Toc102131462)

[3.2. 룰 유형 8](#_Toc102131463)

[3.3. 오브젝트 유형 9](#_Toc102131464)

[3.4. 좌표 정의 10](#_Toc102131465)

[4. System 10](#_Toc102131466)

[4.1. 리소스 정의 10](#_Toc102131467)

[4.2. 조회 11](#_Toc102131468)

[5. Authorization 13](#_Toc102131469)

[5.1. 리소스 정의 13](#_Toc102131470)

[5.2. 토큰 발급(로그인) 13](#_Toc102131471)

[5.3. 토큰 재발급 14](#_Toc102131472)

[5.4. 토큰 삭제(로그 아웃) 15](#_Toc102131473)

[6. Analytic Engine 15](#_Toc102131474)

[6.1. 리소스 정의 15](#_Toc102131475)

[6.2. 추가 19](#_Toc102131476)

[6.3. 삭제 21](#_Toc102131477)

[6.4. 수정 22](#_Toc102131478)

[7. Rule Engine 23](#_Toc102131479)

[7.1. 리소스 정의 23](#_Toc102131480)

[7.2. 조회 24](#_Toc102131481)

[7.3. 추가 26](#_Toc102131482)

[7.4. 삭제 28](#_Toc102131483)

[7.5. 수정 28](#_Toc102131484)

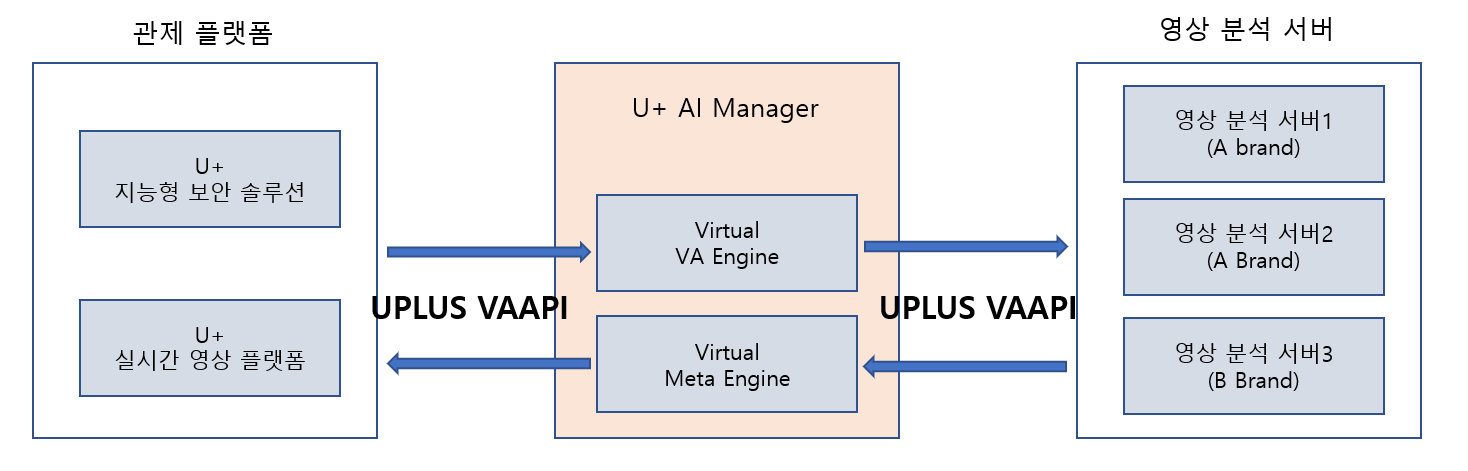
[8. Meta Engine 29](#_Toc102131485)

[8.1. 리소스 정의 29](#_Toc102131486)

[8.2. 전송 30](#_Toc102131487)

# 개요.

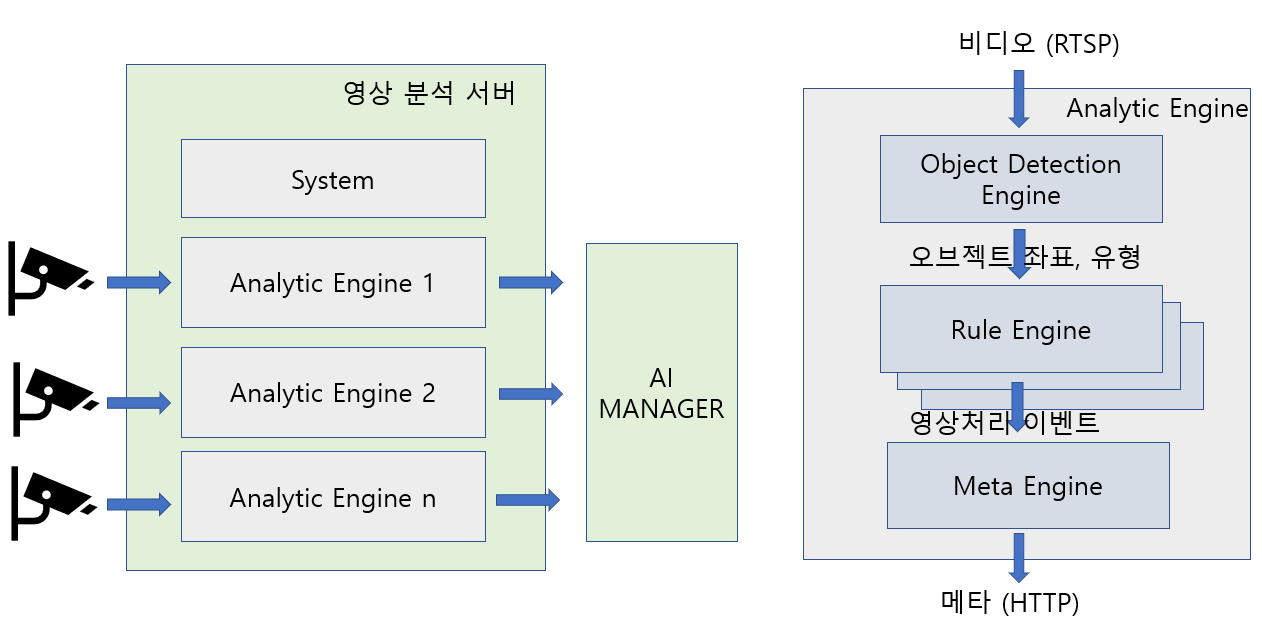
U+ AI Manager는 U+ 관제 플랫폼에서 다양한 제조사의 영상 분석 서버를 연동하기 위한 솔루션으로 UPLUS-VA-API를 사용하여 각 시스템 간의 인터페이스를 정의합니다.



UPLUS-VA-API는 HTTP 기반의 요청/응답 모델을 사용하며, REST의 기본 원칙을 준수합니다.

# 아키텍처

## 리소스



영상 분석 서버의 리소스를 System, Analytic Engine, Rule Engine, Meta Engine으로 구분하며 각 리소스에 고유한 URI를 부여합니다.

* System : 라이선스, 지원 가능한 기능, Analytic Engine / Authorization 리소스에 접근하기 위한 URI 를 제공합니다.
* Analytic Engine : 수신된 영상으로 오브젝트를 검출합니다.
* Rule Engine : Analytic Engine의 하위 리소스로 오브젝트를 기반으로 이벤트를 검출합니다.
* Meta Engine : Analytic Engine의 하위 리소스로 검출된 오브젝트와 이벤트를 전송합니다.
* Authorization : Barear 인증을 위한 JWT 토큰을 관리합니다.

## CRUD

HTTP 메소드를 사용하여 리소스에 대한 CRUD를 제공합니다.

* POST : Create - 새로운 리소스를 추가합니다.
* GET : Read - 리소스의 상태를 조회 합니다.
* PUT : Update – 존재하는 리소스의 상태를 변경합니다.
* DELETE: Delete – 리소스를 삭제합니다.

RESTful에서는 리소스의 부분변경과 전체 변경을 PUT메소드 CTRL 메소드로 구분하지만 본 API에서는 CTRL 메소드는 사용하지 않습니다. PUT 메소드로 전달된 리소스 오브젝트 중 일부 필드가 존재 하지 않을 경우 기존 상태를 유지합니다.

## 요청 응답

요청에 대한 처리 결과는 HTTP 상태 코드를 사용하여 응답합니다. (상태 코드의 정의는 3.1절을 참고합니다). HTTP의 본문은 JSON 포맷을 사용하며, UTF-8로 인코딩 됩니다.

요청에 성공한 경우 요청한 URI에 해당하는 리소스의 오브젝트를 반환하고, 요청에 실패한 경우는 에러 코드와 실패 원인에 대한 상세 내용을 반환합니다.

* Error

|  |  |  |
| --- | --- | --- |
| Name | Value Type | Description |
| error\_code | integer | HTTP 상태 코드와 동일 |
| error\_message | string | 에러 원인에 대해 기술 |

* Sample

|  |
| --- |
| GET /uplus-vaapi/sys  HTTP 1.1 404 Not Found  {  “error\_code” : 404,  “error\_message” : “요청한 경로의 리소스를 찾을 수 없습니다.”  } |

## HATEOAS

API 요청 시 반환되는 오브젝트는 반환되는 리소스 및 연관된 리소스의 링크 정보를 포함합니다. 각각의 링크는 자기 자신을 의미하는 “\_self” 와 연관된 리소스의 이름을 key로 링크 오브젝트를 생성하며, “\_links” 오브젝트로 그룹화 됩니다.

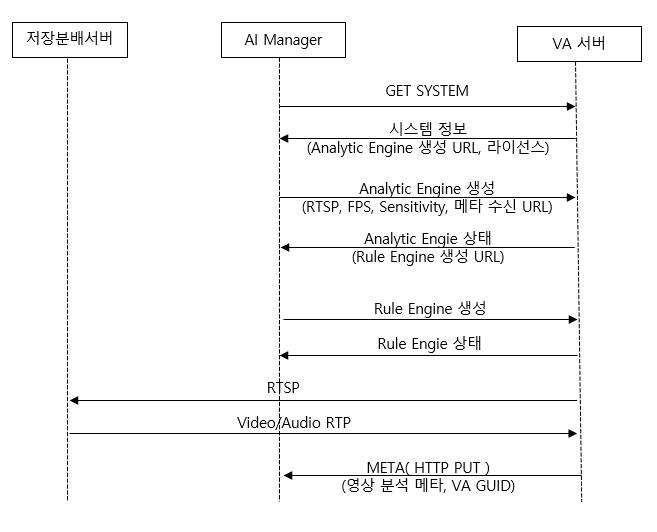
* link

|  |  |  |
| --- | --- | --- |
| key | Type | Description |
| href | string | 링크 주소(URL) |
| method | string array | 해당 링크에서 사용 가능한 메소드  “GET”,”POST”,”PUT”,”DELETE” |

* Sample

|  |
| --- |
| {  “\_links” : {  “\_self” : {  “href” : “/uplus-vaapi/system”,  “method” : [ “GET” ]  },  “engines” : {  “href” : “/uplus-vaapi/engines”,  “method” : [ “GET”, ”POST” ]  }  }  } |

## FLOW



# 코드 및 좌표 정의

## HTTP 상태

|  |  |  |
| --- | --- | --- |
| 코드 그룹 | 코드 | 의미 |
| 2xx | 200 | OK |
| 201 | Created(리소스 추가 대한 응답) |
| 204 | No Content(리소스 삭제에 대한 응답) |
| 4xx | 400 | Bad Request(요청이 유효하지 않음) |
| 401 | Unauthorized(인증 실패) |
| 403 | Forbidden(권한 없음) |
| 404 | Not Found(요청한 리소스가 존재하지 않음) |
| 405 | Method Not Allowed |
| 5xx | 500 | Internal Server Error |

## 룰 유형

|  |  |  |
| --- | --- | --- |
| 코드 | 구분 | 룰 유형 |
| 1 | 일반 | 위험 구역 접근 감지(침입) |
| 2 | 배회 감지 |
| 3 | 불꽃 감지 |
| 4 | 스팀/연기 감지 |
| 5 | 쓰러짐 감지 |
| 6 | 폭력 감지 |
| 7 | 달리기 감지 |
| 8 | 군집 감지 |
| 9 | 유기 감지 |
| 10 | 도난 감지 |
| 11 | 보행자 감지 |
| 12 | 정지차량 감지 |
| 13 | 안전벨트 미착용 감지 |
| 14 | 역주행 감지 |
| 15 | 낙하물 감지 |
| 16 | 불법 주/정차 감지 |
| 17 | 라인 침입 감지(양방향) |
| 18 | 라인 침입 감지(단방향) |
| 19 | 인가자 출입 |
| 20 | 비인가자 출입 |
| 21 | 안전모 착용 |
| 22 | 안전모 미착용 |
| 23 | 안전고리 미착용 |
| 24 | 마스크 미착용 |
| 25 | 이상행위 감지 |
| 26 | 핸드폰 통화 행위 감지 |
| 27 | 핸드폰 촬영 행위 감지 |
| 101 | [음원]스팀 감지 |
| 102 | [음원]타격음 감지 |
| 103 | [음원]비명 감지 |
| 104 | [음원]울음소리 감지 |
| 105 | [음원]차량 충돌음 감지 |
| 106 | [음원]차량 경적음 감지 |
| 107 | [음원]차량 급정거음 감지 |
| 108 | [음원]싸이렌 소리 감지 |
| 109 | [음원]유리 파손음 감지 |
| 201 | 카운트 | 라인 통과 카운트 |
| 202 | 라인 통과 카운트(사람) |
| 203 | 라인 통과 카운트(차량) |
| 204 | 정차 중 차량 카운트 |
| 301 | 레벨 | 객체 혼잡도 |
| 302 | 사람 혼잡도 레벨 |
| 303 | 차량 밀도 |
| 304 | 수위 레벨 |

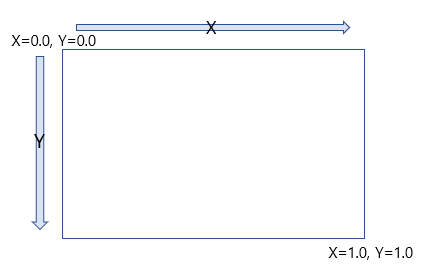
## 오브젝트 유형

|  |  |
| --- | --- |
| 오브젝트 유형 | 코드 |
| 미분류 | 0 |
| 사람 | 1 |
| 자동차 | 2 |
| 얼굴 | 3 |
| 불꽃 | 4 |
| 연기 | 5 |
| 자전거 | 6 |
| 세단 | 7 |
| SUV | 8 |
| 오토바이 | 9 |
| 버스 | 10 |
| 트럭 | 11 |
| 굴착기 | 12 |
| 탱크/트럭 | 13 |
| 지게차 | 14 |
| 레미콘 | 15 |
| 경운기 | 16 |
| 트랙터 | 17 |
| 특수 차량 | 18 |
| 얼굴(남자) | 19 |
| 얼굴(여자) | 20 |
| 얼굴(헬멧) | 21 |

## 좌표 정의

API에서 사용되는 모든 좌표 정보는 영상 좌표계를 사용합니다. 영상의 왼쪽 상단의 모서리를 원점으로, 오른쪽 방향이 x축의 증가방향, 아래쪽 방향이 y축의 증가 방향이 됩니다.

영상의 우측 하단의 좌표는 x=1.0, y=1.0으로 0.0~1.0 사이에 부동 소수점 좌표를 사용합니다.



# System

## 리소스 정의

* system

|  |  |  |
| --- | --- | --- |
| Key | Value Type | Description |
| guid | string | 영상분석 서버의 고유한 아이디 |
| brand\_name | string | 영상분석 서버의 브랜드 이름 |
| icon | object | 영상분석 서버의 브랜드 아이콘 |
| max\_engine | integer | 영상분석 서버에 추가 가능한 최대 엔진 개수 |
| max\_rule | integer | 하나의 엔진에 추가 가능한 룰의 개수 |
| engine\_count | integer | 할당된 엔진의 개수 |
| supported\_authorization | object | 지원 가능한 인증 방식 |
| supported\_rules | object array | 지원 가능한 룰 오브젝트 배열 |
| \_links | object | HATEOAS |

* supported\_authorization

|  |  |  |
| --- | --- | --- |
| Key | Value Type | Description |
| digest | boolean | 다이제스트 인증 지원 여부 |
| basic | boolean | 베이직 인증 지원 여부 |
| bearer | boolean | Bearer(JWT Token) 인증 지원 여부 |

* supported\_rules

|  |  |  |
| --- | --- | --- |
| Key | Value Type | Description |
| rule\_code | Integer | 룰 유형 코드(3.1 절) |
| rule\_name | string | 룰 이름 |
| icon | object | 룰 아이콘 정보  OPTION |

\*icon – optional parameter

* icon

|  |  |  |
| --- | --- | --- |
| Name | Value Type | Description |
| href | string | 다운로드 링크(URL encode) |
| image\_format | string | 아이콘으로 사용할 이미지 포맷(확장자)  “ico”, “png”, “jpg” |

* \_links

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Key(rel) | Description | GET | POST | PUT | DELETE |
| \_self | - | O | X | X | X |
| accept\_token | 토큰 발급 | X | O | X | X |
| refresh\_token | 토근 재발급  OPTION | O | X | X | OPTION |
| analytic\_engines | 영상처리 엔진 | O | O | X | X |

\* “accept\_token” 과 “refresh\_token”은 bearer 인증을 지원하는 경우에만 포함시킵니다.

## 4.2. 조회

* URL: /uplus-vaapi/system
* Method : GET
* Request body: none
* Response body: system object
* Sample

|  |
| --- |
| GET /uplus-vaapi/system  Authorization: Bearer *<accept token string>*  HTTP 1.1 200 OK  {  “brand\_name” : “surv”,  “icon” : {  “href” : “/uplus-vaapi/icon/brand”,  “image\_format” : “png”  },  “max\_engine” : 10;  “max\_rule” : 1,  “supported\_authorization” : {  “digest” : true;  “basic” : false,  “bearer” : true  },  “supported\_rules” = [  {  “rule\_code” : 1,  “rule\_name” : “위험구역 접근 감지”,  “icon” : {  “href” : “/uplus-vaapi/icon/intrusion”,  “image\_format” : “png”  },  {  “rule\_code” : 2,  “rule\_name” : “불꽃감지”,  “icon” : {  “href” : “/uplus-vaapi/icon/flam”,  “image\_format” : “png”  }  }  ],  “\_links” : {  “\_self” : {  “href” : “/uplus-vaapi/system”,  “method” : [ “GET” ]  },  “accept\_token” : {  “href” : “/uplus-vaapi/auth/accept-token”,  “method” : [”POST” ]  },  “refresh\_token” : {  “href” : “/uplus-vaapi/auth/refresh-token”,  “method” : [ “GET”, ”DELETE” ]  },  “analytic\_engines” : {  “href” : “/uplus-vaapi/engines”,  “method” : [ “GET”, ”POST” ]  }  }  } |

.

# Authorization

HTTP 기반의 인증 방식 중 Digest/Basic/Bearer 3가지 방식을 지원하며, 클라이언트는 서버로 지원하는 인증 방식을 조회 후, 사용할 인증 방식을 선택합니다. 지능형 서버는 Digest Authorization과 Basic Authorization 둘 중 하나의 방식은 반드시 지원해야 합니다.

## 리소스 정의

* token

|  |  |  |
| --- | --- | --- |
| Key | Type | Description |
| token | string | 인증 토큰 |
| timeout | integer | 인증 토큰 만료 시간(초단위)( |

## 토큰 발급(로그인)

* URL: <system.\_links.accept\_token.href>
* Method : POST
* Request body: none
* Response body: token object
* Sample

|  |
| --- |
| POST /uplus-vaapi/auth/accept-token  Authorization: Digest username=”admin” …생략…  HTTP 1.1 200 OK  Set-Cookie: refresh\_token=<refresh token string>;HttpOnly;  {  “token” : “*<accept token string>*”,  “timeout” : 3600,  } |

## 토큰 재발급

* URL: <system.\_link.refresh\_token.href>
* Method : GET
* Request body: none
* Response body: token object
* Sample

|  |
| --- |
| GET /uplus-vaapi/auth/refresh-token  Authorization: Bearer *<accept token string>*  Cookie: refresh\_token=*<refresh token string>*  HTTP 1.1 200 OK  {  “token” : “<accept token string>”,  “timeout” : 3600,  } |

## 토큰 삭제(로그 아웃)

* URL: <system.\_link.refresh\_token.href>
* Method : DELETE
* Request body: none
* Response body: none
* Sample

|  |
| --- |
| DELETE /uplus-vaapi/auth/refresh-token  Authorization: Bearer *<accept token string>*  Cookie: refresh\_token=*<refresh token string>*  HTTP 1.1 204 No Content |

# Analytic Engine

## 리소스 정의

* analytic engine

|  |  |  |
| --- | --- | --- |
| Key | Value Type | Description |
| enable | boolean | 영상분석 엔진 활성화 |
| engine\_id | string | 영상분석 엔진 아이디.  READ ONLY |
| engine\_name | string | 엔진 이름 |
| sensitivity | Integer | 영상분석 감도 |
| fps | Integer | 영상 분석 프레임 레이트 |
| video | object | 비디오 수신 파라미터 |
| meta | object | 메타 송신 파라미터 |
| \_links | object | HATEOAS.  READ ONLY |

* video

|  |  |  |
| --- | --- | --- |
| Key | Value Type | Description |
| url | string | 영상 수신 URL |
| login\_id | string | 영상 송신 서버 사용자 아이디 |
| login\_pwd | string | 영상 송신 서버 사용자 비밀번호 |

* meta

|  |  |  |
| --- | --- | --- |
| Key | Value Type | Description |
| url | string | 메타 송신 URL |
| login\_id | string | 메타 수신 서버 사용자 아이디 |
| login\_pwd | string | 메타 수신 서버 사용자 비밀번호 |
| filters | integer array | 메타에 포함시킬 정보 종류  1: frame, 2: alarm |

* \_links

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Key(rel) | Description | GET | POST | PUT | DELETE |
| \_self | - | O | X | X | X |
| rule\_engines | 룰 엔진 설정 | O | O | O | X |
| expansion | 확장 파라미터  OPTION | O | X | O | X |

* 1. 조회
* 전체 엔진 리스트 조회
  + URL: <system.\_links. analytic\_engines.href>
  + Method: GET
  + Request Body: none
  + Response Data: analytic engine object array
* 특정 엔진 조회
  + URL: <analytic\_engines.\_links.\_self.href>
  + Method: GET
  + Request Body: none
  + Response Body: analytic engine object
* Sample

|  |
| --- |
| GET /uplus-vaapi/engines  Authorization: Bearer *<accept token string>*  HTTP 1.1 200 OK  [  {  “enable” : true,  “engin\_id” : “1”,  “sensitivity” : 5,  “fps” : 10,  “video” : {  “url” : “rtsp://192.168.0.10/1/high”,  “login\_id” : “admin”,  “login\_pwd” : “password”  },  “meta” : {  “url” : “http://192.168.0.10/upus-vaapi/meta”,  “login\_id” : “admin”,  “login\_pwd” : “password”,  “filters” : [1,2,3]  },  “\_links” : {  “\_self” : {  “href” : “/uplus-vaapi/engines/1”,  “method”: [“GET”,”PUT”,”DELETE”]  },  “rule\_engines” : {  “href” : “/uplus-vaapi/engines/1/rules”,  “method”: [“GET”,”POST”]  },  “expansion” : {  “href” : “/uplus-vaapi/engines/1/expansion”,  “method”: [“GET”,”PUT”]  }  }  },  {  “enable” : true,  “engin\_id” : “2”,  “sensitivity” : 5,  “fps” : 10,  “video” : {  “url” : “rtsp://192.168.0.10/2/high”,  “login\_id” : “admin”,  “login\_pwd” : “password”  },  “meta” : {  “url” : “http://192.168.0.10/upus-vaapi/meta”,  “login\_id” : “admin”,  “login\_pwd” : “password”,  “filters” : [1,2,3]  },  “\_links” : {  “\_self” : {  “href” : “/uplus-vaapi/engines/2”,  “method”: [“GET”,”PUT”,”DELETE”]  },  “rule\_engines” : {  “href” : “/uplus-vaapi/engines/2/rules”,  “method”: [“GET”,”POST”]  },  “expansion” : {  “href” : “/uplus-vaapi/engines/2/expansion”,  “method”: [“GET”,”PUT”]  }  }  }  ]  GET /uplus-vaapi/engines/1  Authorization: Bearer *<accept token string>*  HTTP 1.1 200 OK  {  “enable” : true,  “engin\_id” : “1”,  “sensitivity” : 5,  “fps” : 10,  “video” : {  “url” : “rtsp://192.168.0.10/1/high”,  “login\_id” : “admin”,  “login\_pwd” : “password”  },  “meta” : {  “url” : “http://192.168.0.10/upus-vaapi/meta”,  “login\_id” : “admin”,  “login\_pwd” : “password”,  “filters” : [1,2,3]  },  “\_links” : {  “\_self” : {  “href” : “/uplus-vaapi/engines/1”,  “method”: [“GET”,”PUT”,”DELETE”]  },  “rule\_engines” : {  “href” : “/uplus-vaapi/engines/1/rules”,  “method”: [“GET”,”POST”]  },  “expansion” : {  “href” : “/uplus-vaapi/engines/1/expansion”,  “method”: [“GET”,”PUT”]  }  }  } |

## 6.2. 추가

* URL: <system.\_links. analytic\_engines.href>
* Method: POST
* Request Body: analytic engine object
* Response Body: analytic engine object
* Sample

|  |
| --- |
| POST /uplus-vaapi/engines  Authorization: Bearer <accept token string>  {  “enable” : true,  “sensitivity” : 5,  “fps” : 10,  “video” : {  “url” : “rtsp://192.168.0.10/1/high”,  “login\_id” : “admin”,  “login\_pwd” : “password”  },  “meta” : {  “url” : “http://192.168.0.10/upus-vaapi/meta”,  “login\_id” : “admin”,  “login\_pwd” : “password”,  “filters” : [1,2,3]  },  }  HTTP 1.1 200 OK  {  “enable” : true,  “engin\_id” : “1”,  “sensitivity” : 5,  “fps” : 10,  “video” : {  “url” : “rtsp://192.168.0.10/1/high”,  “login\_id” : “admin”,  “login\_pwd” : “password”  },  “meta” : {  “url” : “http://192.168.0.10/upus-vaapi/meta”,  “login\_id” : “admin”,  “login\_pwd” : “password”,  “filters” : [1,2,3]  },  “\_links” : {  “\_self” : {  “href” : “/uplus-vaapi/engines/1”,  “method”: [“GET”,”PUT”,”DELETE”]  },  “rule\_engines” : {  “href” : “/uplus-vaapi/engines/1/rules”,  “method”: [“GET”,”POST”]  },  “expansion” : {  “href” : “/uplus-vaapi/engines/1/expansion”,  “method”: [“GET”,”PUT”]  }  }  } |

## 삭제

* URL: <analytic\_engines.\_links.\_self.href>
* Method: DELETE
* Request body: none
* Response body: none
* Sample

|  |
| --- |
| DELETE /uplus-vaapi/engines/1  Authorization: Bearer *<accept token string>*  HTTP 1.1 204 No Content |

## 수정

* URL: <analytic\_engines.\_links.\_self.href>
* Method: PUT
* Request body: analytic object
* Response body: analytic object
* Sample

|  |
| --- |
| PUT /uplus-vaapi/engines/1  Authorization: Bearer <accept token string>  {  “sensitivity” : 3,  “fps” : 15,  }  HTTP 1.1 200 OK  {  “enable” : true,  “engin\_id” : “1”,  “sensitivity” : 3,  “fps” : 15,  “video” : {  “url” : “rtsp://192.168.0.10/1/high”,  “login\_id” : “admin”,  “login\_pwd” : “password”  },  “meta” : {  “url” : “http://192.168.0.10/uplus-vaapi/meta”,  “login\_id” : “admin”,  “login\_pwd” : “password”,  “filters” : [1,2,3]  },  “\_links” : {  “\_self” : {  “href” : “/uplus-vaapi/engines/1”,  “method”: [“GET”,”PUT”,”DELETE”]  },  “rule\_engines” : {  “href” : “/uplus-vaapi/engines/1/rules”,  “method”: [“GET”,”POST”]  },  “expansion” : {  “href” : “/uplus-vaapi/engines/1/expansion”,  “method”: [“GET”,”PUT”]  }  }  } |

# Rule Engine

## 리소스 정의

* rule engine

|  |  |  |
| --- | --- | --- |
| Key | Value Type | Description |
| rule\_id | string | 룰 엔진 아이디.  READ ONLY |
| rule\_type | integer | 이벤트를 발생시키는 룰 유형(3.2 절) |
| object\_type | Integer array | 이벤트를 발생시키는 오브젝트 유형(3.3 절) |
| roi\_type | Integer | 1: 선, 2: 다각형 |
| roi\_positions | object array | ROI 좌표 배열 |
| \_links | object | HATEOAS.  READ ONLY |

* roi\_position

|  |  |  |
| --- | --- | --- |
| Key | Value Type | Description |
| x | float | x좌표 |
| y | float | y좌표 |

* \_links

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Key(rel) | Description | GET | POST | PUT | DELETE |
| \_self | - | O | X | O | X |
| expansion | 확장 파라미터  OPTION | O | X | O | X |

## 조회

* 전체 룰 리스트 조회
  + URL: <analytic\_engines.\_links.rule\_engines.href>
  + Method: GET
  + Request body: none
  + Response body: rule engine object array
* 특정 룰 조회
  + URL: <rule\_engines.\_links.\_self.href>
  + Method: GET
  + Request body: none
  + Response body: rule engine object
* Sample

|  |
| --- |
| GET /uplus-vaapi/engines/1/rules  Authorization: Bearer *<accept token string>*  HTTP 1.1 200 OK  [  {  “rule\_id” : “1”,  “rule\_type” : 1,  “object\_type” : [1],  “roi\_type” : 2,  “roi\_positions” : [  { “x” : 0.25, “y” : 0.25 },  { “x” : 0.75, “y” : 0.25 },  { “x” : 0.75, “y” : 0.75 },  { “x” : 0.25, “y” : 0.75 }  ],  “\_links” : {  “\_self” : {  “href” : “/uplus-vaapi/engines/1/rules/1”,  “method” : [ “GET”, “PUT”]  },  “expansion” :  “href” : “/uplus-vaapi/engines/1/rules/1/expansion”,  “method” : [“GET”, “PUT”]  }  }  },  {  “rule\_id” : “2”,  “rule\_type” : 2,  “object\_type” : [3,4],  “roi\_type” : 2,  “roi\_positions” : [  { “x” : 0.0, “y” : 0.0 },  { “x” : 1.0, “y” : 0.0 },  { “x” : 1.0, “y” : 1.0 },  { “x” : 0.0, “y” : 1.0 }  ],  “\_links” : {  “\_self” : {  “href” : “/uplus-vaapi/engines/1/rules/2”,  “method” : [ “GET”, “PUT”]  },  “expansion” :  “href” : “/uplus-vaapi/engines/1/rules/2/expansion”,  “method” : [“GET”, “PUT”]  }  }  }  ]  GET /uplus-vaapi/engines/1/rules/1  Authorization: Bearer *<accept token string>*  HTTP 1.1 200 OK  {  “rule\_id” : “1”,  “rule\_type” : 1,  “object\_type” : [1],  “roi\_type” : 2,  “roi\_positions” : [  { “x” : 0.25, “y” : 0.25 },  { “x” : 0.75, “y” : 0.25 },  { “x” : 0.75, “y” : 0.75 },  { “x” : 0.25, “y” : 0.75 }  ],  “\_links” : {  “\_self” : {  “href” : “/uplus-vaapi/engines/1/rules/1”,  “method” : [ “GET”, “PUT”]  },  “expansion” :  “href” : “/uplus-vaapi/engines/1/rules/1/expansion”,  “method” : [“GET”, “PUT”]  }  }  } |

## 추가

* URL: <analytic\_engines.\_links.rule\_engines.href>
* Method: POST
* Request Body: rule engine object
* Response Body: rule engine object
* Sample

|  |
| --- |
| POST /uplus-vaapi/engines/1/rules  Authorization: Bearer *<accept token string>*  {  “rule\_type” : 1,  “object\_type” : [1],  “roi\_type” : 2,  “roi\_positions” : [  { “x” : 0.25, “y” : 0.25 },  { “x” : 0.75, “y” : 0.25 },  { “x” : 0.75, “y” : 0.75 },  { “x” : 0.25, “y” : 0.75 }  ],  }  HTTP 1.1 200 OK  {  “rule\_id” : “1”,  “rule\_type” : 1,  “object\_type” : [1],  “roi\_type” : 2,  “roi\_positions” : [  { “x” : 0.25, “y” : 0.25 },  { “x” : 0.75, “y” : 0.25 },  { “x” : 0.75, “y” : 0.75 },  { “x” : 0.25, “y” : 0.75 }  ],  “\_links” : {  “\_self” : {  “href” : “/uplus-vaapi/engines/1/rules/1”,  “method” : [ “GET”, “PUT”]  },  “expansion” :  “href” : “/uplus-vaapi/engines/1/rules/1/expansion”,  “method” : [“GET”, “PUT”]  }  }  } |

## 삭제

* URL: <analytic\_engines.\_links.rule\_engines.href>
* Method: DELETE
* Request body: none
* Response body: none
* Sample

|  |
| --- |
| DELETE /uplus-vaapi/engines/1/rules/1  Authorization: Bearer *<accept token string>*  HTTP 1.1 204 No Content |

## 7.5. 수정

|  |
| --- |
| PUT /uplus-vaapi/engines/1/rules/1  Authorization: Bearer *<accept token string>*  {  “roi\_type” : 1,  “roi\_positions” : [  { “x” : 0.5, “y” : 0.0 },  { “x” : 0.5, “y” : 1.0 }  ],  }  HTTP 1.1 200 OK  {  “rule\_id” : “1”,  “rule\_type” : 1,  “object\_type” : [1],  “roi\_type” : 1,  “roi\_positions” : [  { “x” : 0.5, “y” : 0.0 },  { “x” : 0.5, “y” : 1.0 }  ],  “\_links” : {  “\_self” : {  “href” : “/uplus-vaapi/engines/1/rules/1”,  “method” : [ “GET”, “PUT”]  },  “expansion” :  “href” : “/uplus-vaapi/engines/1/rules/1/expansion”,  “method” : [“GET”, “PUT”]  }  }  } |

# Meta Engine

## 리소스 정의

* meta

|  |  |  |
| --- | --- | --- |
| Key | Value Type | Description |
| engine\_id | string | analytic engine 아이디 |
| utc\_time | string | yyyy-MM-ddThh:mm:ss.zzz |
| frame | object | 화면 구성을 위한 정보 |
| alarms | object array | 알람 발생 정보 |

* frame

|  |  |  |
| --- | --- | --- |
| Key | Value Type | Description |
| rules | object array | 현재 엔진에 설정되어 있는 룰 정보 |
| objects | object array | 화면에 검출된 오브젝트 정보 |

* rule

|  |  |  |
| --- | --- | --- |
| Key | Value Type | Description |
| id | string | rule engine 아이디 |
| type | integer | 이벤트를 발생시키는 룰 유형 |
| roi\_type | Integer | 1: 선, 2: 다각형 |
| roi | float array | ROI 좌표 배열  [x1,y1,x2,y2,x3,y3….] |

* object

|  |  |  |
| --- | --- | --- |
| Key | Value Type | Description |
| id | integer | 오브젝트 아이디 |
| type | integer | 오브젝트 유형(3.3 절) |
| box | float array | 오브젝트 박스 좌표  [top,left,bottom,right] |
| evts | string array | 오브젝트로 인해 이벤트 발생시 이벤트를 발생시킨 룰의 아이디 리스트 |

* alarm

|  |  |  |
| --- | --- | --- |
| Key | Value Type | Description |
| id | string | 이벤트를 발생시킨 룰의 아이디 |
| type | integer | 이벤트를 발생시킨 룰의 코드(3.2 절)) |
| img | string | 이벤트를 발생시킨 오브젝트의 이미지(JPG)  Base64 encode  OPTION |
| count | integer | 룰 구분이 카운트일 경우(3.2 절)  OPTION |
| level | integer | 룰 구분이 레벨일 경우(3.2 절)  OPTION |

## 전송

* URL : <analytic\_engine.meta.url>
* Method : PUT
* Request body : meta object
* Response body : none
* Sample

|  |
| --- |
| POST /uplus-vaapi/meta  Authorization: Bearer *<accept token string>*  {  “engine\_id” : “1”,  “utc\_time” : “2022-03-15T10:30:30.012”,  “frame” : {  “rules” : [  {  “id” : “1”,  “type” : 1,  “roi\_type” : 2,  “roi” : [0.25,0.25,0.75,0.25,0.75,0.75,0.25,0.75]  },  {  “id” : “2”,  “type” : 1,  “roi\_type” : 1,  “roi” : [0.5,0.0,0.5,0.0]  }  ]  “objects” : [  {  “id” : 1,  “type” : 2,  “box” : [ 0.45,0.51,0.52,0.55],  “evts” : [“1”,”2”]  },  {  “id” : 2,  “type” : 1,  “box” : [ 0.1,0.1,0.2,0.2]  }  ]  },  “alarms” : [  {  “id” : “1”,  “type” : 1  },  {  “id” : “2”,  “type” : 201,  “count” : 10  }  ]  }  HTTP 1.1 OK |