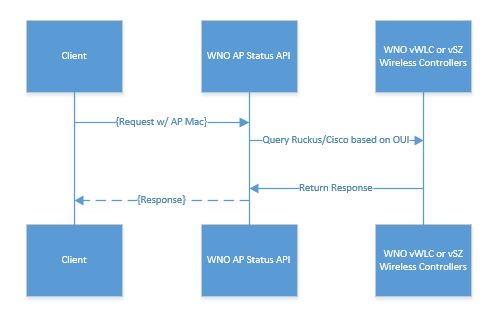
1. Introduction  
   1. Executive Summary  
      1. The WNO AP Status API is written and maintained by the Wireless Analytics and Automation department at Charter (‘WAA’) and will return the status of a given Access Point (‘AP’) mac on the Wireless Network Operations (‘WNO’) public Wi-Fi controllers by querying the various WNO Wi-Fi controller systems where the AP would connect. It queries various systems in real-time and then returns a response detailing information regarding the AP’s operational status. It will give details as to the AP’s connection status, configuration status (readiness to transmit), and also return other various status codes in the event of a failure. The API is vendor-agnostic, so it will return a normalized payload for both Ruckus and Cisco APs, and will also be updated if any new AP vendors are deployed by WNO.
   2. Sequence Flow Diagram



* 1. Target Audience

The API is intended to support any Charter development requiring real-time status of Charter APs.

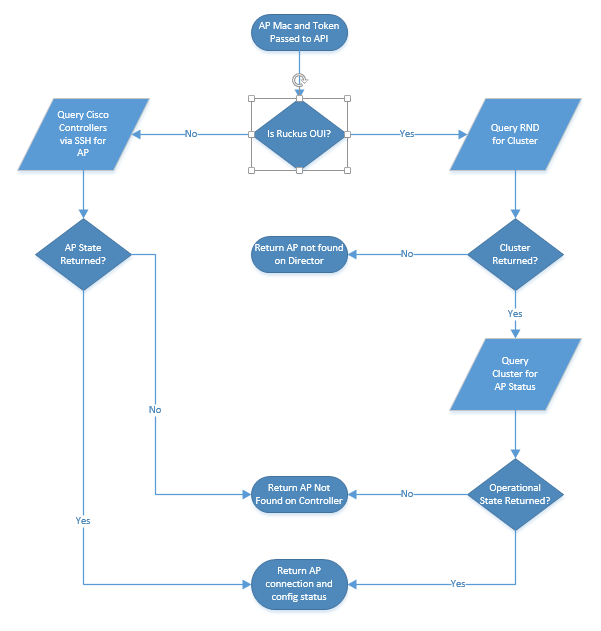
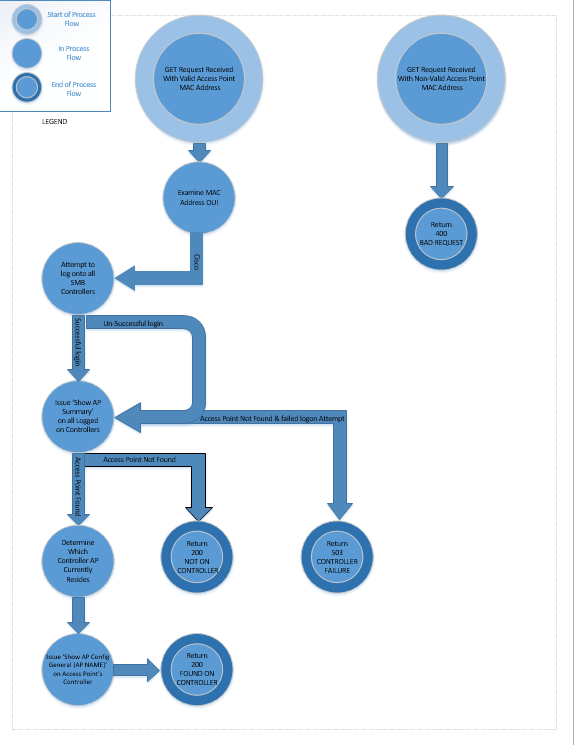
* 1. API Location

The production API will be hosted on a VM in the Wi-Fi Pod in NDCW (‘RED’ network).

* 1. Planned Production URLs
     1. API Endpoint  
        <https://wno-waa-api-self-install.charter.com/api/v1/self-install/accessPointStatus>
     2. Swagger API documentation:  
        <https://wno-waa-api-self-install.charter.com/api/v1/ui>
  2. Service Interface and Methods Supported

Name: WNO AP Status API  
Interface: Rest  
Main URI: /api/v1/self-install/accessPointStatus  
Method: get (include request headers, detailed below)

Format: JSON

* 1. Basic API Flow  
     
  2. Cisco flow detail  
     

1. Scope  
   1. Given AP Mac, will return AP status of any Ruckus or Cisco AP within WNO control.
2. AP Plug-In to Controller Join Chronology
   1. Cable Modem (‘CM’) with ISIW tag joins CMTS
   2. AP connects to CM and gets ISIW space IP from DHCP. IP is registered to DLPQS API, available from APO
   3. Depending on AP vendor, AP retrieves vendor-specific option 43 from DHCP
      1. Ruckus:
         1. AP is directed to Ruckus Network Director (RND)
         2. RND examines subnet-based ruleset and directs AP to a regional controller
         3. AP Joins Controller and Zone specified by the Subnet Rule. AP goes from ‘connectionState’ ‘Discovery’ to connectionState ‘Connect’
         4. Controller pushes firmware and configuration to the AP. Several different configState values are returned during this process.
         5. AP configState changes to ‘completed’ and connectionState ‘connect’ to signal it is transmitting
            1. Process takes about 22 minutes
      2. Cisco:
         1. AP joins east or west provisioning controllers
         2. AP connection state changes to ‘Connect’
         3. AP configuration state changes to ‘DOWNLOADING’ while downloading firmware and configuration
         4. AP configState changes to ‘REGISTERED’ when transmitting if connectionState is ‘Connect’
         5. [Insert approximate time to completion]
3. Requirements  
   1. Input Data
      1. The request header should include two keys: ‘mac’ and ‘Authorization’. ‘Authorization’ is the access token provided by WAA, and ‘mac’ is the AP mac in the following regex format: '([0-9A-Fa-f]{2}(:)){5}([0-9A-Fa-f]{2})'
      2. Header example:  
         {'Authorization':'bearer JWT token','mac':[APMAC]}
4. Response Elements
   1. HTTP Status Codes and status texts
      1. 'BAD\_MAC\_FORMAT' - HTTPStatus.BAD\_REQUEST #400  
         This error is returned when the AP Mac is not passed to the API in the proper regex format.
      2. 'NOT\_IN\_DIRECTOR' - HTTPStatus.OK #200  
         This status text currently applies only to Ruckus APs. It means that the AP has not yet reached the Ruckus Network Director for its controller assignment
      3. 'DIRECTOR\_FAILURE' - HTTPStatus.SERVICE\_UNAVAILABLE #503  
         This code is returned in the event that there is a system error querying the Ruckus Network Directors.
      4. 'CONTROLLER\_UNAUTHORIZED' - HTTPStatus.UNAUTHORIZED #401  
         Returned when the API is unable to authenticate directly to a WNO controller. The authorizations are stored within a secured location in the API. If this status is returned, please contact WAA.
      5. 'DIRECTOR\_UNAUTHORIZED' - HTTPStatus.UNAUTHORIZED #401  
         Returned when authorization fails to a Ruckus Network director. Please contact WAA if this error is encountered.
      6. 'CONTROLLER\_FAILURE' - HTTPStatus.SERVICE\_UNAVAILABLE #503  
         This error occurs when 1) the AP mac is not found on a controller, and 2) one of the controllers cannot be contacted. This can happen because of system failure, system maintenance, etc.
      7. 'NOT\_ON\_CONTROLLER' - HTTPStatus.OK #200  
         Returned if the AP is not found on a controller.
      8. 'FOUND\_ON\_CONTROLLER' - HTTPStatus.OK #200  
         This status text is returned when the AP mac is found on one of the WNO controllers. The response payload (detailed below) should be examined for further information regarding the AP’s readiness.
      9. 'SYSTEM\_EXCEPTION' - HTTPStatus.SERVICE\_UNAVAILABLE #503  
         Returned when there is an exception in the API.
      10. 'NO\_DIRECTOR\_SESSIONS' - HTTPStatus.SERVICE\_UNAVAILABLE #503  
          Returned when no Ruckus Network Director sessions are available.
   2. Payload Response Keys
      1. Example:  
         {  
          "mac":"58:93:96:27:C8:30",  
          "model":"ZF7762-AC",  
          "name":"2140533\_VENUEBASE\_COAX",  
          "ip":"67.49.162.181",  
          "configState":"completed",  
          "connectionState":"Connect",  
          "vendor":"’Ruckus’"  
         }
      2. 'mac': same as the input passed
      3. ‘vendor’: the AP’s manufacturer, currently ‘Ruckus’ or ‘Cisco’
      4. 'model': the AP’s model as reported by the controller
      5. ‘name’: the AP’s name on the controller
      6. ‘ip’: the IP address of the AP as reported by the controller
      7. 'connectionState': details whether the AP is in one of the following statuses: ‘Discovery’, ‘Connect’, or ‘Disconnect’. ‘Disconnect’ implies that the AP is not currently attached to a controller. Note that Discovery connectionState applies to Ruckus controllers and is normally returned when the AP first joins to the controller.
      8. ‘configState’
         1. Ruckus:

"configState" : {

"description" : "State of the AP configuration.",

"enum" : [ "newConfig", "fwApplied", "fwDownloaded", "fwFailed", "configApplied", "completed", "configFailed" ]

* + - 1. Cisco:
         1. ‘DOWNLOADING’ or ‘REGISTERED’
      2. configState of ‘completed’ or ‘REGISTERED’ indicates that the AP is fully operational if the connectionState is ‘Connect’

1. System Access
   1. System access is granted by Wifi Analytics and Automation for a specific application. A request should be sent to [DL-NTO-Wireless-Analytics@charter.com](mailto:DL-NTO-Wireless-Analytics@charter.com). Authorization tokens should be protected and not shared between individuals or other departments. Upon review, we will issue an authorization token that should include in the request header (detailed below)
2. System Logging
   1. Housed in NCDW, under WNO Control
   2. Relevant status messages will be logged to syslog sever and can be examined in the event of API exceptions