



June 2024

WATS Feature Release Note
Software Package Rest API



Feature Areas

Software Package Rest API 3



Software Package Rest API

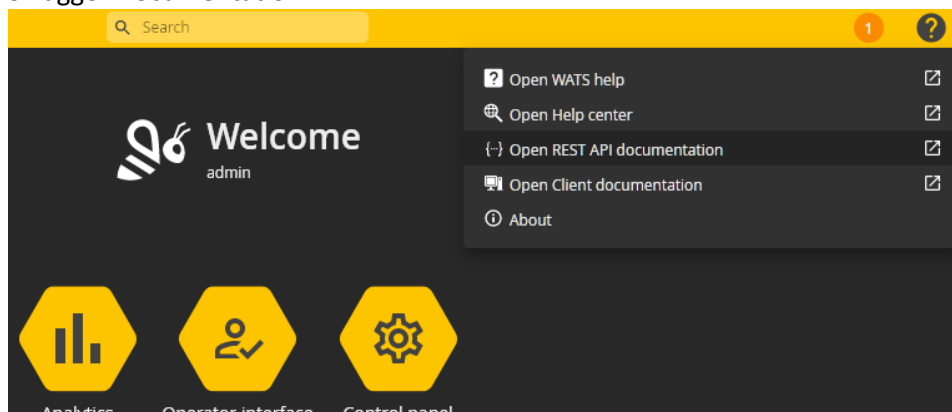
With this release, Enterprise users will have access to our new Rest API for managing their Software Packages directly.

The API will provide various endpoints for creating, reading, updating and deleting Software Packages.

A list of endpoints added:

- POST
 - Api/Software/Package
 - Api/ Software/PackageStatus
 - Api/ Software/Package/UploadZip/{id}
 - Api/ Software/FileAttribute/{id}
- PUT
 - Api/ Software/Package/{id}
- GET
 - Api/ Software/Package/{id}
 - Api/ Software/PackageByName
 - Api/ Software/Packages
 - Api/ Software/PackagesByTag
 - Api/ Software/PackageFiles/{id}
 - Api/Software/VirtualFolders
- DELETE
 - Api/ Software/Package/{id}
 - Api/ Software/PackageByName

More information about limitations and parameters for each endpoint can be found in the [Swagger Documentation](#)





Under the Software section

Software		Show/Hide List Operations Expand Operations
POST	/api/Software/Package	Posts a Package. If name exists, version will be previous version + 1. Key-value pair of tags. Formatted like "tagName=tagValue". Example: Test_Tag=TestValue. Multiple tags can be included, separate with ' '. Example: Test_Tag=TestValue Test_Tag2=NewTestValue.
DELETE	/api/Software/Package/{id}	Deletes the Software Package. Status of Software Package has to be Draft or Revoked before being deleted.
GET	/api/Software/Package/{id}	Gets the specified Software Package by providing the ID of the package.
PUT	/api/Software/Package/{id}	Puts a Package. Note: This will overwrite existing configuration. Use POST api/Software/Package for new package versions. Package in Draft allows for editing of all details. If package is Pending or Released, only Status and Tags can be edited. Status follows these validation rules: Draft to Pending, Pending to Draft OR Released, Released to Revoked. Tags has to be in this format: <PackageInfo><tagName>TagValue</tagName><tagName2>TagValue2</tagName2></PackageInfo>.
POST	/api/Software/Package/FileAttribute/{id}	Updates the file attribute for the specified file. Get file ID by calling /api/Software/GetPackageFiles/{id}
POST	/api/Software/Package/UploadZip/{id}	Post a single Zip file to upload files into Software Package. Will merge files by default, provide query cleanInstall=true to delete all files and folders before installing. Zip cannot contain files on root. All files need to be in folder otherwise: zipFile/myFolder/myFile.txt. Zip has to be encoded using "multipart/form-data", and has to have content-type "application/zip".
DELETE	/api/Software/PackageByName	Deletes the Software Package by using name and version. Status of Software Package has to be Draft or Revoked before being deleted.
GET	/api/Software/PackageByName	Searches for the Software Package with the provided name and status. Can also return a specific version.
GET	/api/Software/PackageFiles/{id}	Returns a list of files assigned to Software Package. Note: Does not contain the actual files.
GET	/api/Software/Packages	Returns a list with all Software Packages.
GET	/api/Software/PackagesByTag	Filters for Software Packages with the provided tag and tag value and status.
POST	/api/Software/PackageStatus/{id}	Updates the status of the Software Package defined by ID. Status can only go from Draft to Pending, Pending to Draft OR Released, Released to Revoked. Any other configuration will be refused.
GET	/api/Software/VirtualFolders	Gets all Virtual Folders registered in the Production Manager.

We also have a Zendesk article that provides some easier to read information:
[Software Package REST API – Help Center \(virinco.com\)](#)