Planning App ResultsAPI

The ResultsAPI is provided to facilitate a controlled and consistent means for accessing necessary treatment plan details and information. The ResultsAPI is most frequently used by customers wishing to develop custom treatment plan reports for clinical patients. From the ResultsAPI treatment plan data can be accessed, including (but not limited to): patient geometries, prescriptions, beam information, spot placements, and plan dose results. It should be pointed out that when users generate custom plan reports they must include an indication of the coordinate system in which any position information is being displayed and users should be reminded that all ResultsAPI position data is provided in either DICOM Patient or IEC 61217 Beam coordinate systems, as appropriate, at this time. Plan reports must also include a list of any density reassignments that were used in the plan.

Most Planning ResultsAPI functions must be called using Thinknode meta requests. Because of this added complexity, an example function call is provided to aid in end user understanding of proper usage.

The Astroid ResultsAPI and META Requests

A fundamental principle of the Thinknode data environment in which Astroid lives is that Astroid stores INPUT data only, not calculation results. As such, this makes it difficult for end users to directly access the “usual” data from a treatment plan. The astroid ResultsAPI was created to address this difficulty by allowing users to access typical data (e.g.: dose, structure geometry, beam geometry, etc) from a treatment plan without knowing how to explicitly construct the complicated requests needed to generate that data.

Most ResultsAPI functions require a treatment plan as an input and return a form of a *calculation_request* object. This object is typically not directly useful to end users, instead, this object is used to generate the request for the final useful data. This calculation request object could be retrieved from Thinknode ISS, parsed, and then submitted as the body of a new Thinknode calculation request. This is slow, cumbersome, and sometimes impossible, as Thinknode imposes a 5MB limit on the size of the request body. Thinknode META requests can be used to circumvent these difficulties and automatically handle the submission of the resulting calculation request object. The necessary details of the META request submission process are described below.

A META request consists of a function that is flagged as a *Generator* function. The Generator function is a normal Thinknode provider function in all regards. However, the return type MUST be a valid Thinknode calculation request. All Astroid ResultsAPI functions can thus be used as Generators in META calculations. When a calculation request is flagged as a META type and submitted to Thinknode, the Generator function will first be processed, which will produce a new calculation object. This new calculation is automatically submitted and its output assigned back to the original META submission, such that the calculation result returned by the META will directly contain the expected (useful) data that the end user requested from the treatment plan.

In summary, the process flow of using the ResultsAPI is as follows:

1. User calls a ResultsAPI function with a specific treatment plan

2. ResultsAPI returns a calculation request for the final data the user requires
3. A META request is submitted with the returned ResultsAPI calculation request (this can be done by reference ID such that the calculation result does not need to be retrieved)
4. The META request result is the useful data that the user requires

ResultsAPI Functions and Version Compatibility

For a comprehensive list of available Planning Results API functions, please refer to the Results API Function List.

The Planning Results API provides two types of functions for generating the request for a treatment plan: Context based ResultsAPI functions and Non-context based ResultsAPI functions. Each type is described more fully below.

Context based ResultsAPI functions

These are the recommended functions to be used when interfacing with the Planning App ResultsAPI, as they will ensure compatibility with past and future versions of the ResultsAPI.

Context based ResultsAPI functions are signified by names prefixed with “api_” (e.g.: *api_generate_plan_summary_request*). These functions check the Thinknode Planning App version context that was used to publish (approve) or last edit the treatment plan and attempt to return a calculation_request for the Planning App version context that was captured in the treatment plan.

Input Parameters

These functions require input parameters, such as the treatment_plan, as a Thinknode dynamic type. This allows the input data to ignore Thinknode data upgrades when calling the ResultsAPI function for values that can be dependent on upgrades.

Return Data

The context based ResultsAPI functions return a *context_based_calculation_request* that contains the Thinknode Planning App context ID used for the plan and an optional calculation_request that should be submitted using the returned context ID.

These functions use the following logic to attempt to generate a compatible *calculation_request* for the Thinknode Planning App version context ID captured in the treatment plan. This request is created by the following logic:

- If the ResultsAPI function does not exist for the captured Planning App version, the optional<calculation_request> will be returned as null

- If the ResultsAPI function signature has changed, the calculation request will be constructed for the Planning ResultsAPI version that corresponds for the returned context ID.

Processing a Function

These context based ResultsAPI functions return a special type of calculation request object that is not directly usable within a Thinknode META request as they contain the additional context data needed to ensure the desired calculation is performed with the appropriate Astroid Planning App version. The process of running a context based ResultsAPI request consists of the following steps:

- Submit the initial “api_*” function request as a normal (non-META) calculation
- Retrieve the result of the above calculation and get the context ID
- Perform a META request using the above context, setting the META Generator as the calculation object portion of the above “api_*” function result (if it's not null)

Non-context based ResultsAPI functions

These functions are not recommended to be called directly unless the caller specifically wishes to recompute the resulting data with a new Astroid version. These functions construct a request for the end data while not taking into account the captured Planning App context ID of the treatment plan, therefore data upgrades and other changes may be applied to the inputs and outputs, such that the returned data may not exactly match that of the original treatment plan at the time of approval (publishing).

Non-context based ResultsAPI function names are not prefixed with “api_” (e.g.: *generate_plan_summary_request*).

Example

In the .decimal astroid-script-library there is a script called *planning_results_api_example.py* that provides a basic example of calling and accessing the ResultsAPI using META functions.

In this particular example the *api_generate_plan_summary_request* is the ResultsAPI function being called using the current Planning App context installed in the target Thinknode realm. The returned value contains the context ID of the Thinknode Planning App version which created the treatment_plan and the appropriate calculation request for that Planning App version for generating the *plan_summary*. The generator function and META request is submitted using the returned Planning App context ID from the ResultsAPI function. The result of the META request returns the extracted data to use in generating treatment plan reports.