Planning App ResultsAPI

The results API is provided to facilitate a controlled and consistent means for accessing necessary treatment plan details and information. The Results API is most frequently used by customers wishing to develop custom treatment plan reports for clinical patients. From the results API 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 Results API position data is provided in IEC 61217 coordinate systems at this time. Plan reports must also include a list of any density reassignments that were used in the plan.

Most Planning Results API functions must be called using Thinknode meta requests. Because of this added complexity, an example function call has been added for end user convenience.

The Astroid ResultsAPI and META Requests

A fundamental princile 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 complicated data (e.g.: dose, structure geometry, beam geometry, etc) from a treatment plan without knowing how to excplictly construct the requests to generate that data.

The ResultsAPI functions take a treatemnt plan as an input and return a form of a *calculation_request* object, which is 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 pulled down, converted to text, and then submitted as the body of a new Thinknode calculation request. This is slow, cumbersome, and sometimes impossible because Thinknode imposes a 5MB limit on the size of the request body. Therefore, META requests where implemented to automatically handle this case.

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 be processed and run. The calcualtion result returned by the META will contain the useful data to the end user from the treatment plan submitted to the original ResultsAPI function.

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
4. The META request result is the useful data the user requires

ResultsAPI Usage and Version Compatibility

The Planning Results API provides two types of functions for generating the request for a treatment plan:

A context based ResultsAPI function

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 with names prefixed with “api_*”. These functions check the Thinknode Planning App version context that was used to publish (approve) or last edit the treatment plan and attempts to return a calculation_request for the Planning App version context that was captured in the treatment plan.

The context based functions provide compatibility for treatment plans that have been created using past versions of the Planning App and thus may have differing ResultsAPI functions or functions with differing signatures. This is achieved by:

- These functions take in input parameters as a dynamic type to ignore Thinknode data upgrade functions when calling this function for values that can be dependent on upgrades.
- These functions return a *context_based_calculation_request* that contains the Thinknode Planning App context ID and an optional calculation_request that should be submitted at the returned context ID.
- These functions uses the following logic to attempt generate a compatible request for the Planning App version captured in the treatment plan:
    - If the ResultsAPI function does not exist 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 for the context ID returned.

A non-context based ResultsAPI function

These functions are not recommended to be called from the ResultsAPI unless the caller wishes to recompute 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.

Non-context based ResultsAPI functions are signified by not having “api_*” prefixed function names.

Available Functions

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

Example

In the .decimal astroid-script-library there is a script called *planning_results_api_example.py* that provides a basic example in calling and accessing the results API meta functions.

In this particular example the *generate_plan_summary_request* is the API function being called by the generator and meta function as this is a common function to use in extracting data for use in generating treatment plan reports.