astroid Dicom App User Guide

Overview

The astroid Dicom App is used to provide access to functions which are used for reading a dicom file and returning more useable standard data types, such as those from the RT Types App available on the thinknode framework. The astroid Dicom App device is not an interactive end user application. Users of the system will write scripts or use a fully interactive software program that makes calls to the functions provided by the astroid Dicom App API. The core functionality includes parsing dicom RT_Plan, Structure Set, CT Image, and Dose file types. The Dicom App adheres to the Dicom PS3.3 2015c dicom specification.

Access to the astroid Dicom App is provided by the thinknode™ framework using http json formatted requests. thinknode™ provides the 'backbone' used to send and receive requests, maintain users, realms and organizations, and provide data storage and management.

The Dicom Manifest Guide contains a complete list of all functions and types supported by the astroid Dicom App through API calls.

Purpose

The purpose of this document is to provide guidance on the setup, access, and usage of the astroid Dicom app.

Getting Started

The primary purpose of the astroid Dicom App is to provide a library of efficient and easily accessible proton dosimetry functions which can serve as the foundational tools for developing higher level radiation treatment planning and analysis software programs. The Dicom App is not intended to be an interactive end-user application; instead it will serve as a source of tested and validated calculation functions which software developers can utilize in other end-user applications. The astroid Dicom App provides access to these calculation functions through the thinknode™ framework which is accessible over the internet and performs all calculations in a cloud based environment. thinknode provides robust, redundant data storage, ensuring industry leading data integrity as well as stable software version management and release process.

Connecting to thinknode™

The Dicom App, and all its released versions live on thinknode™ hosted on the Amazon S3 service. Communication with the App is performed over the internet through thinknode https requests which are

password protected per user and permission based by thinknode organization administrators. The thinknode startup procedure includes full installation of the appropriate application environment onto the virtual servers, which includes hardware and performance checks to ensure sufficient resources are available for the requested application. A full thinknode™ userguide is found at https://developers.thinknode.com.

Basic information needed to communicate with the Dicom App through thinknode™:

- thinknode™ user name and password as a base64 encoded blob (e.g.: userName:PassW0rd1 → dXNlck5hbWU6UGFzc1cwcmQx)
- thinknode™ api url (e.g.: https://api.thinknode.com)
- thinknode™ version the application is released on (e.g.: v1.0)
- application name (e.g.: Dicom)
- thinknode™ realm name

See thinknode™ Examples for sample code projects for connecting to the Dicom App.

Precautions

It is the responsibility of those utilizing this application to ensure all that all usages of this product relating to dicom information handling are performed by trained and qualified personnel only and that such personnel is aware that the quality of any generated output data is highly dependent on the quality and correctness of the input data; therefore if any questions or uncertainties exist regarding the quality, units, or identification of input data arise, they must be investigated and resolved before the data are used.

> **HIPPA Compliance**
>
> Any data posted to thinknode™ is immutable and can not be deleted. Please ensure when posting dicom files that you have anonymized patient data in the dicom files or that you are posting the data to a private bucket in thinknode™.

Functions

The Dicom App exposes a few key functions through its API. Below is a list and brief explanation of a few of the most common functions. A full function list can be found at the Dicom Manifest Guide as well as details of each function, argument parameters, and return values are provided at the.

1. **parse_dicom_filesystem_item**:
   - Parses a dicom *filesystem_item* type into a *dicom_data* type. The filesystem item can contain a thinknode ISS directory of dicom files.
2. **parse_dicom_blob**:

- Parses a single dicom standard formatted file into a *dicom_data* type. This function handles the determination of which type of RT Types to create based on the files modality (e.g. rt plan, structure set, ct image, dose, etc). This function requires the input dicom file to be in a base64 encoded blob.
3. **parse_dicom_blobs_to_CT_iamge**:
   - Parses a list of dicom (blob) files and constructs the combined CT image set as an *image_3* type. Note: Non-CT files are ignored
4. **make_patient**:
   - Takes an array of *dicom_data* types and returns it as a *dicom_patient*. If there were CT Images it will combine them in to an *image_3d*.

See the Calc Provider Examples section for example usage of some of the dicom functions.

Data Types

Details of each datatype used by the Dosimetry App can be found at the Dicom Manifest Guide.

Calc Provider Examples

thinknode™ calculation provider

Official thinknode™ user guide documentation can be found at https://developers.thinknode.com. Below are basic examples of communicating with thinknode™ calculation providers and using the Dicom App calculation functions.

- **Python:** Sample Python project (note: this example requires third party libraries) that details the following thinknode™ api functions:
  - Python: Overview
  - Python: Decimal Libraries

System Usage

Normal system usage requires following the specifications for functions and data types as detailed by the Dicom App Manifest Guide. Making correct calls to the calculation provider requires following the specifications for api requests as detailed by the thinknode™ API guide. Additionally, as most large data structures are returned in base64 encoded format, users can and should perform error checking during decoding to be certain data was not lost or corrupted during transmission.

Details about the intended use of the functions can be found at the Functions section.

Improper System Usage

When using the astroid Dicom App, as with any complex program, there is the potential for misuse. The thinknode™ framework and Dicom App limit most of the potential misuse of exposed functions and data types, but there's still a potential for misinterpretation of the results provided by the app. Below are a few examples of topics that are easily confused or overlooked when performing even basic tasks.

Grid and Image Data Interpretation

The difference between the grid and image origin definitions can result in image or grid data being misinterpreted. Image origins are defined at the corner of the image, while grid origins are defined directly as the position of the corner point as shown below.

For images, it's important to remember that the location of the pixel value is at the pixel center point, so the position of the first cell, for example, should be shifted by 1/2 of the pixel spacing. Grids however typically contain point data (not cell centered), so data and position correspond directly with a shift.
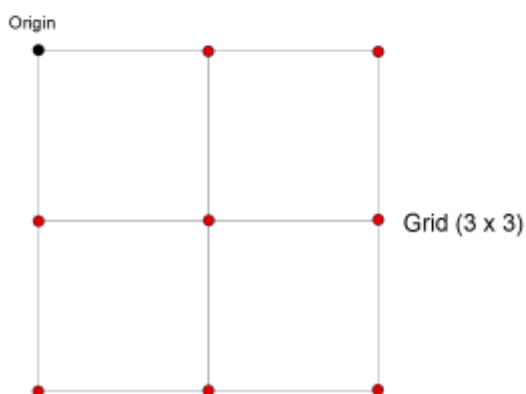
Known Application Limitations

Below are listed the known application limitations, defects, or inconsistencies.

1. None at this time

Should you discover or suspect any additional errors or limitations not noted above, please contact the .decimal astroid Dicom App development lead, Kevin, Erhart, at kerhart@dotdecimal.com with details of your concerns. All concerns will be addressed as quickly as possible and follow up with the submitter will occur throughout the inquire and correction process.

USR-006

.decimal LLC, 121 Central Park Place,
Sanford, FL. 32771

From:
https://apps.dotdecimal.com/ - **decimal App Documentation**

Permanent link:
https://apps.dotdecimal.com/doku.php?id=dicom:userguide:userguide&rev=1436553799

Last update: **2021/07/29 18:21**