This tool takes a set Synthea input files with a given ratio of various FHIR resource types like, for example, Patient, Practitioner and Encounter. It then generates output files with the same resource types, but with changed ratios of resources. For example, we can take a set of input files with 20% of Patient, 10% of Practitioner and 70% of Encounter resources, and then generate a set of output files that contains 40% of Patient, 5% of Practitioner and 55% of Encounter resources.

Both input and output files are in the standard Synthea format, with resource files organized into resource groups, with each resource group being stored into its own directory. Each resource group contains one .ndjson file per resource type. Within each file there are any number of resources in json format, with single resource per file line:

[Root Dir]

[0059fe72-e91d-4e05-be6e-869e0ca8a20b] (resource group directory)

* AllergyIntolerance.ndjson
* CarePlan.ndjson
* Encounter.ndjson
* ExplanationOfBenefit.ndjson
* Patient.ndjson
* Practitioner.ndjson
* Etc…

[00603305-c6c0-40e9-b573-0c5b70d712a8] (another resource group directory)

* AllergyIntolerance.ndjson
* CarePlan.ndjson
* Encounter.ndjson
* ExplanationOfBenefit.ndjson
* Patient.ndjson
* Practitioner.ndjson
* Etc…

This tool can process either sets of files stored in a file system or a set of Azure blobs.

Once the output set of resource files is generated by either deleting or cloning the input files, references to resources are updated to guarantee that there will not be any orphaned references.

When reading the input resource files, the tool verifies that subset of resource data generated by Synthea compiles with the FHIR specifications. Also, when reading the input resource files, the tool verifies that there are no invalid references to nonexistent resources in the resource group.

Tool can process numerous resource groups in parallel by specifying task count in the command line parameters.

# Synthea files are processed in several steps:

For detailed description of the command line options, execute the tool from command line without any command line options.

## 1. Generate a set of input blobs by using Synthea tool, in the above-described format.

## 2. From the Synthea input blobs generate a set of info files.

Blend tool command – **blobinfo**

This step needs to be taken only once for a set of input Synthea blobs. You must have ready an input blob container with Synthea resource blobs organized into resource groups as described above. You must also download one of the resource groups into your local drive. Generated info files will contain basic information such as number of resources per type, per group and average size of resource per resource type. The purpose of info files is to make it easier for the tool to generate output blend of desired size and resource ratios, without loading large amounts of data into the memory.

## 3. Create blend target ratio file.

Blend tool command – **target**

Takes as input the info files from previous step and blend ratio file, and then creates target ratios files. Input blend ratio file contains the desired ratios of resources in the output resources, in percentages. Output target ratios file contains ratios at which input resources will be deleted or cloned to achieve the given ratio of output resource types and total output size. Thanks to this file the blend tool can modify the ratios of the input files as it reads, without loading large amounts of data into the memory.

## 4. Generate output files or blobs with modified resource ratios.

Blend tool command – **file**

or

Blend tool command – **blob**

Takes a set of input resource files and generates a set of output resource files in the above-described format. Subset of data generated by the Synthea in the input files if validated against FHIR specification and all the resources are validated against resources present in the resource group.

## 5. OPTIONAL - Verify a set of resource files.

Blend tool command – **verifyfile**

or

Blend tool command – **verifyblob**

## 6. Upload files into FHIR server DB (only blobs)

Blend tool command – **import**

## 7. Periodically test if FHIR server import task completed

Blend tool command – **isfinished**