








Article

# OpenGDC: Unifying, Modeling, Integrating Cancer Genomic Data and Clinical Metadata

Eleonora Cappelli <sup>1,†</sup> , Fabio Cumbo <sup>2,\*†</sup> , Anna Bernasconi <sup>3</sup> , Arif Canakoglu <sup>3</sup> ,  
Stefano Ceri <sup>3</sup> , Marco Masseroli <sup>3</sup>  and Emanuel Weitschek <sup>4</sup> 

<sup>1</sup> Department of Engineering, University of Roma Tre, Via della Vasca Navale 79/81, 00146 Rome, Italy; eleonora.cappelli@uniroma3.it

<sup>2</sup> Department of Cellular, Computational and Integrative Biology (CIBIO), University of Trento, Via Sommarive 9, Povo, 38123 Trento, Italy

<sup>3</sup> Department of Electronics, Information and Bioengineering, Politecnico di Milano, Piazza L. da Vinci 32, 20133 Milan, Italy; anna.bernasconi@polimi.it (A.B.); arif.canakoglu@polimi.it (A.C.); stefano.ceri@polimi.it (S.C.); marco.masseroli@polimi.it (M.M.)

<sup>4</sup> Department of Engineering, Uninettuno University, Corso Vittorio Emanuele II 39, 00186 Rome, Italy; emanuel.weitschek@uninettunouniversity.net

\* Correspondence: fabio.cumbo@unitn.it

† These authors contributed equally to this work.

Received: 31 July 2020; Accepted: 3 September 2020; Published: 12 September 2020



**Abstract:** Next Generation Sequencing technologies have produced a substantial increase of publicly available genomic data and related clinical/biospecimen information. New models and methods to easily access, integrate and search them effectively are needed. An effort was made by the Genomic Data Commons (GDC), which defined strict procedures for harmonizing genomic and clinical data of cancer, and created the GDC data portal with its application programming interface (API). In this work, we enhance GDC harmonization by applying a state of the art data model (called Genomic Data Model) made of two components: the genomic data, in Browser Extensible Data (BED) format, and the related metadata, in a tab-delimited key-value format. Furthermore, we extend the GDC genomic data with information extracted from other public genomic databases (e.g., GENCODE, HGNC and miRBase). For metadata, we implemented automatic procedures to extract and normalize them, recognizing and eliminating redundant ones, from both Clinical/Biospecimen Supplements and GDC Data Model, that are present on the two sources of GDC (i.e., data portal and API). We developed and released the OpenGDC software, which is able to extract, integrate, extend, and standardize genomic and clinical data of The Cancer Genome Atlas (TCGA) from the GDC. Additionally, we created a publicly accessible repository, containing such homogenized and enhanced TCGA data (resulting in about 1.3 TB). Our approach, implemented in the OpenGDC software, provides a step forward to the effective and efficient management of big genomic and clinical data of cancer. The strong usability of our data model and utility of our work is demonstrated through the application of the GenoMetric Query Language (GMQL) on the transformed TCGA data from the GDC, achieving promising results, facilitating information retrieval and knowledge discovery analyses.

**Keywords:** data modeling; data integration; next generation sequencing; cancer; knowledge extraction

## 1. Background

The large amount of genomic data generated by Next Generation Sequencing (NGS) technologies [1,2] and their related clinical data brings significant value for medical research, especially for cancer studies [3]. Thanks to NGS techniques, different types of experimental data are

produced, whose storage and analysis can be very demanding [4–6]. More and more often researchers have to face big biological data [7,8], frequently lacking integrated data models and accessible schema representations. Thus, storing, retrieving, integrating, comparing, and analyzing heterogeneous biomedical data becomes a major challenge.

In cancer research, several organizations are involved in the collection, management and publication of genomic and clinical data. In particular, the Genomic Data Commons (GDC [9,10]) is a recent initiative of the National Cancer Institute (NCI) with the aim of creating a unified system to promote the sharing of these data. The GDC supports several programs and defines bioinformatics pipelines: it provides Clinical/Biospecimen Supplements and genomic data harmonization procedures related to DNA-sequencing [11], RNA-sequencing [12,13], miRNA-sequencing [14], Copy Number Variation [15] and DNA-methylation [16]. The processed data is publicly available through the GDC portal, which deals with different cancer programs; The Cancer Genome Atlas (TCGA) [17] is the most relevant project within the GDC, collecting genomic and clinical data of 33 different tumor types of over 11,000 patients [18].

TCGA data were available at its own portal until late 2016, but since early 2017 they were migrated to the new GDC portal, resulting in a major change of genomic and clinical/biospecimen formats and schema. In the GDC portal, experimental data (i.e., DNA-sequencing, RNA-sequencing, miRNA-sequencing, Copy Number Variation and DNA-methylation data) are produced from harmonization procedures applied on different analysis strategies, improving the quality of data available at the old TCGA portal. Indeed, the GDC provides a programmatic access to interact with these harmonized data through Application Programming Interfaces (APIs), e.g., to obtain aliquot Universal Unique Identifiers (UUIDs) that identify uniquely GDC experiments. The harmonization procedures provide standardized and comparable data, depending on the type of NGS experiment, regardless of the program where they were generated.

For what concerns metadata, Clinical/Biospecimen Supplements were represented in an unstructured format in the TCGA portal; conversely, the GDC introduced a new structured data model (i.e., the GDC Data Model). The transition is however still incomplete: the GDC provides some relevant clinical/biospecimen information only in the old unstructured format and some other only or also in the new format. Correspondingly, the GDC exposes two different methods for retrieving clinical and biospecimen information. The first one is the direct download of Supplements from the GDC portal in XML format, which is semi-structured and does not adhere to a specific data model. The second one is through the GDC APIs, which allow downloading structured information according to the GDC Data Model and provide output in JSON format. These methods allow reaching two different materializations of the metadata, partly overlapping with each other.

The GDC is proceeding with the migration from the first representation to the second one, importing and inserting the data contained in the first within the second. However, in this transitory phase (that has lasted for several months and will probably last for a long time), much of the information in the first model is not yet replicated to the second, and there is no single source that provides information from both models. In order to obtain a comprehensive representation of such information, it is therefore necessary to extract data using a pipeline that deals with model differences and identifies, manages and removes the overlapping information. The first contribution of our work is the design and development of such a pipeline, such that the clinical and biospecimen data (referred to as metadata) are represented with a common format.

In our work, we solve the issues arisen in the transition from the TCGA data portal to the GDC one, providing genomic data and their associated clinical/biospecimen metadata in a standardized format, making both of them seamless, straightforward and easy to be used. We enhance GDC harmonized data by applying a state of the art data model for genomics, in order to uniform genomic and clinical/biospecimen data. We automatically standardize data by mapping them to such unique common schema, thereby supporting scientists in data integration and analyses [19–21]. We also

integrate information extracted from external public databases, i.e., GENCODE [22], HGNC [23], miRBase [24] and NCBI genome annotations [25], enriching the content of the experimental data.

This work is an evolution of another project, TCGA2BED [26], which faced partially similar but much simpler issues, focusing on the old TCGA portal. Unlike TCGA2BED, beside extending TCGA genomic data and standardize the format in which they are provided by the GDC, we integrate, normalize and make non-redundant their multiple metadata available with different representations; we do so by mapping them to a unique data model and widely exploiting the GDC APIs to interact with and extract GDC data. Our main contribution is the integrative representation of experimental and clinical/biospecimen data by applying the Genomic Data Model (GDM [27]); this then allows querying them, together with other data from multiple sources, uniformly and comprehensively through the GenoMetric Query Language (GMQL [28,29]) directly on a new publicly available repository of standardized data. GDM consists of two parts, one describing processed datasets with a genomic region-based format, and one describing the metadata. For the former one, we map the content of GDC data to GDM, thereby transforming the experimental data of the GDC into a new data collection, which we denote as OpenGDC, harmonized and extended by linking with other public databases. For the latter one, the Clinical and Biospecimen Supplements (which are semi-structured, not part of a data model) are extracted and merged with all the information on clinical and biospecimen data available through the GDC APIs (which is structured and adheres to the GDC Data Model), and finally converted to the metadata format of GDM, used by OpenGDC.

Other works have dealt with the problem of storing, retrieving and enhancing data of the GDC, almost all of them are focused on the TCGA program. Among them, we mention: (i) TCGA-assembler 2 [30], a software pipeline which allows downloading TCGA data from the GDC defining filtering criteria to merge the extracted data files of samples into a single data table, and finally to process them; (ii) The International Cancer Genome Consortium (ICGC [31]), which provides a data portal to characterize genomic abnormalities in different cancer types, including data from TCGA; (iii) The Seven Bridges Cancer Genomics Cloud (CGC [32]), which allows accessing data from public cancer genomic datasets (e.g., TCGA) and analyzing them in the cloud by using bioinformatics tools and workflows. All these works are of great interest and improve the access to GDC data; in particular they aggregate them, identify important genomic features, and analyze them with cloud computing resources. Moreover, there are several state-of-the-art tools to retrieve and analyze TCGA data, including some R packages like (i) TCGAbiolinks [33], which provides algorithms for data mining and analysis of cancer genomics, (ii) cbiportal [34], an open platform for interactively exploring multidimensional cancer genomics data sets in the context of clinical data and biologic pathways, (iii) Xena [35], an easy-to-use cancer genomics visualization tool for large public data resources of the GDC. Conversely, our approach is different, as it aims at facilitating the use of TCGA data of the GDC by providing it in a standardized and extended format, and enriched with multiple integrated metadata. In particular, OpenGDC provides a structured data format of the different types of genomic experiments through a single schema, and considers the clinical and biospecimen information as strict defined structured metadata. For a more detailed overview of the available tools for TCGA data we refer to the work [36], where the authors identify two main categories of TCGA tools, for data *Extraction* and for *Integrative data analysis*. We can use this distinction and classify our novel system in the first category.

The rest of this manuscript is organized as follows. Section 2 presents the Genomic Data Model and its application to the different data types retrieved from the GDC. Here, we also describe the pipeline used to build metadata from Clinical/Biospecimen Supplements and from additional information retrieved through GDC APIs, as well as we illustrate the detection and removal procedure of redundant metadata attributes. In Section 3 we show the architecture of our novel software system, OpenGDC, for the extraction, harmonization and extension of genomic data and metadata from the GDC. We also describe the structure of the created FTP repository, containing all the public accessible genomic and clinical data of the TCGA program of the GDC and their harmonized and extended OpenGDC version

already produced using our software system. Section 4 shows examples of querying and processing of the new OpenGDC data with GMQL, highlighting the advantages provided by the performed harmonization and extension. In Section 5 we discuss the main aspects of our contribution, summarize our final remarks and mention future developments.

## 2. Methods

In this section we describe the standardization of GDC experimental data and Clinical/Biospecimen Supplements through the application of GDM, which provides a representation of the genomic experimental data in Browser Extensible Data (BED) format [37] and of its biological/clinical properties (i.e., metadata) in a key-value format. Genomic data are extended with additional information extracted from external public databases. Using GDM, experimental data are unified into a single format, thus becoming homogeneous, coherent, and comparable. Metadata are also unified, as the original GDC metadata formats are all associated with a single format of key-value pairs, although the keys and the number of pairs may vary across different datasets. Because of the heterogeneous nature of data, it is not possible to know a priori all the clinical, biological and experimental properties of the experimental samples; these are produced as a result of metadata mapping. Furthermore, to generate metadata we develop intelligent procedures for identifying redundant metadata information that are present on the two different sources of the GDC: Clinical/Biospecimen Supplements from the data portal and GDC Data Model information from the API.

In the next two subsections we detail the genomic data and metadata formats obtained by applying the Genomic Data Model to all open data types provided by the GDC.

### 2.1. The Genomic Data Format

For genomic data, we use a free-BED data representation, in which fix coordinate fields (chromosome, start position, end position, strand) and we include additional fields according to the specific type of experiment; for every data type we provide a specific ready-to-use schema in XML format. We implemented automatic procedures for converting the original GDC genomic data into such free-BED format; to index our BED output files, we introduce *opengdc\_id*, an extension of the aliquot Universal Unique Identifier (UUID, that is the unit of analysis for GDC genomic data identifying a sample analyzed portion). Since in the GDC an aliquot relates to different data types, *opengdc\_id* concatenates the *aliquot uuid* with the specific *data type*. In the following, we provide an overview of the input and output data of our standardization procedures; for a detailed description of all input and output fields of each data type, the reader may refer to the OpenGDC Format Definition (Supplementary File 1).

*Gene Expression Quantification* data are provided in the GDC for each aliquot in three tab-delimited files, each of which presents the Ensembl ID of the gene and one of the following values:

1. *FPKM*, the number of Fragments Per Kilobase of transcript per Million mapped reads;
2. *FPKM-UQ*, the Upper Quartile normalized FPKM value;
3. *counts*, the number of reads aligned to each gene, calculated by HT-Seq.

We merge the content of these files using the common *Gene\_Ensembl* field. Then, we extract additional information to describe the gene regions. In the final free-BED structure we include the genomic coordinates (i.e., *chromosome*, *start position*, *end position* and *strand*), the *gene\_symbol* from GENCODE (human genome version GRCh38 annotation), and the corresponding *entrez\_gene\_id* from the NCBI genome annotation.

*MiRNA Expression Quantification* data are derived from the sequencing of the micro RNA (i.e., miRNA). They contain information about the nucleotide sequence and the expression of miRNAs. One file per aliquot is provided by the GDC, where each row refers to a single miRNA and contains its expression computed on all reads aligning to that particular miRNA. In the free-BED output we

consider all fields provided in input, with the addition of the miRNA genomic coordinates extracted from miRBase and the corresponding *entrez\_gene\_id* and *gene\_symbol* extracted from HGNC.

*Isoform Expression Quantification* data contain expression profiles calculated for each isoform of the miRNA sequence. The GDC provides one file for each aliquot, where each row refers to a single isoform. For the free-BED structure, all input fields are left unchanged with the exception of the *isoform\_coords* field, which is parsed to obtain separate genomic coordinate fields. As an addition, we retrieve the corresponding *entrez\_gene\_id* and *gene\_symbol* from HGNC.

A copy number variation (CNV) is a variation in the number of copies of a given genomic segment per cell. The GDC provides two data types related to CNVs: *Copy Number Segment* (including both germline and somatic CNVs) and *Masked Copy Number Segment* (including only somatic CNVs). The internal representation is the same for both data types. A single experiment is represented by a tab-delimited file, where each row refers to a single CNV. For the free-BED representation we reuse all input fields except for the sample id; we add the *strand* field—required for the BED standard—which we always set to ‘unknown’ using the wildcard character ‘\*’.

*Masked Somatic Mutation* experiments discover mutations by aligning DNA sequences derived from tumor samples to sequences derived from normal samples and to a reference sequence. A Mutation Annotation Format (MAF) file is used to specify, for each sample, the discovered putative or validated mutations and to categorize those mutations (SNP, deletion, or insertion) as ‘somatic’ (i.e., originating in the tissue) or ‘germline’ (i.e., originating from the germline), as well as to specify additional information about the mutations. Four MAF files for each tumor sample are provided by the GDC, each representing DNA-sequencing data. Each file is generated by a specific analysis pipeline [38–41] and includes 125 attributes. By merging the four input files, we defined a free-BED structure with 18 fields including the main information, such as genomic coordinates, the corresponding *gene\_symbol* and *entrez\_gene\_id* (if the mutation involve a gene), the type of mutation, the tumor and matched normal sequencing alleles 1 and 2, and the aliquot barcode/UUID for the tumor and matched normal samples.

A DNA methylation experiment consists in deep sequencing of bisulfite-treated DNA. It can be obtained as the covalent modification of cytosine bases at the C-5 position, generally within a CpG sequence context. If DNA methylation occurs in promoter regions, it is an epigenetic mark that represents the repression of the transcripts of the promoter gene. We consider both Illumina Infinium HumanMethylation27 (HM27) and HumanMethylation450 (HM450) DNA methylation platforms, used for measuring the level of methylation at 27,578 and 485,577 known CpG sites as beta values (respectively for HM27 and HM450). By using probe sequence information provided in the manufacturer manifest, HM27 and HM450 probes are remapped to the GRCh38 reference genome. These probe coordinates are then used to identify the associated transcripts from GENCODE, the associated CpG island (CGI), and the position of the CpG site in reference to the island. For each methylated site the GDC reports a list of gene symbols; the genes that fall within 1500 bp from the methylated site are used, considering the gene as starting from the transcription start site (TSS) to the end of the gene body. For each aliquot, the GDC provides a tab-delimited **Methylation Beta Value** data file with 11 fields. We define a free-BED structure composed of 18 fields, which includes all original fields with the addition of the *strand*, the *entrez\_gene\_id* retrieved from GENCODE or HGNC, the *ensembl\_transcript\_id*, the *position\_to\_tss* (distances in base pairs of the CpG site from each associated transcript start site; negative values indicate that the CpG site is located downstream with respect to the TSS), and the *cgi\_coordinate* (i.e., the start and end coordinates of the CpG island associated with the CpG site). Moreover, we filtered out the methylation sites with missing beta values (i.e., not measured or with unreliable measurement) and reported the *gene symbol* that is at minimum bp distance from the methylated CpG dinucleotide, in case this is outside a gene region.



## 2.2. Metadata Format

Each experimental BED file is associated with a metadata file containing a list of key-value pairs. Also metadata files are indexed with an *opengdc\_id*, which identifies the pair of BED-metadata files. To populate the OpenGDC metadata files, we retrieve clinical/biospecimen information from the GDC data type called Clinical and Biospecimen Supplements. In addition, we consider other properties retrieved using the GDC APIs (specifying *aliquot uuid* and *data type* as parameters).

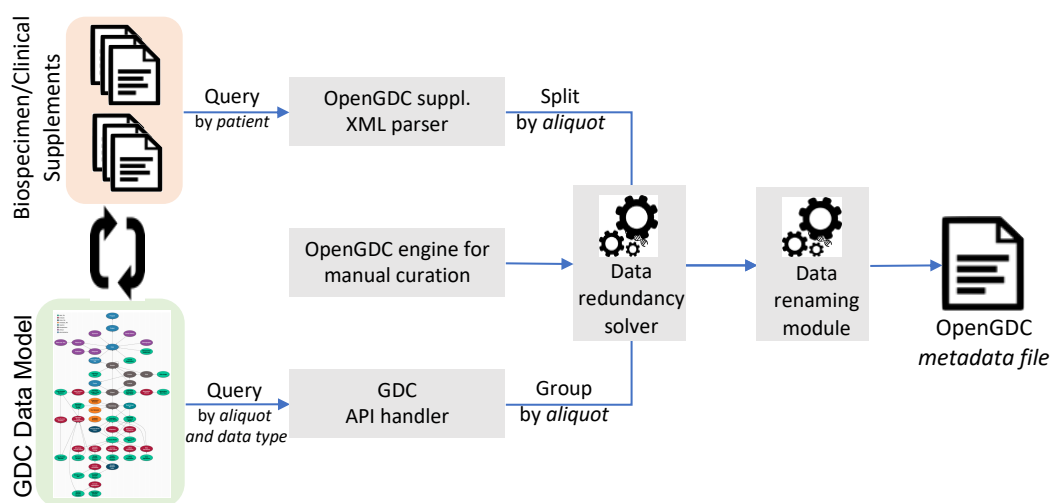
Clinical and Biospecimen Supplements are a special data type that contains data documentation; this information is stored in two different XML format files, originally provided by Biospecimen Core Repositories (BCRs) under contract of the National Cancer Institute (NCI). A **Clinical Supplement** is a collection of information about demographics, medical history (i.e., diagnosis, treatments, follow ups, and molecular tests), and family relationships (i.e., exposure and history) of a particular patient. A **Biospecimen Supplement** instead includes information associated with the physical sample taken from a patient and its processing.

## 2.3. Metadata Extraction And Composition

The content of an OpenGDC metadata file is obtained by taking into account:

1. the BCR *Biospecimen* and *Clinical Supplements*,
2. the information retrieved through the GDC APIs,
3. additional manually curated attributes computed within our standardization pipelines.

Given a converted experimental data file in free-BED format, identified by an *opengdc\_id*, the corresponding metadata file is generated according to the pipeline shown in Figure 1.



**Figure 1.** Metadata pipeline overview. The procedure starts with the download of the *Biospecimen* and *Clinical Supplement* files by using the Genomic Data Commons (GDC) application programming interfaces (APIs) according to a *patient uuid*. Aliquot uuids are extracted from the *Biospecimen Supplement* file, whose content is split by aliquot. Based on these *aliquot uuids* and their associated *data types*, the *GDC Data Model* is queried through the GDC APIs, in order to obtain additional metadata information. Finally, inside a unique metadata file, we merge together: clinical data, a portion of biospecimen data, GDC Data Model metadata, and manually curated attributes (automatically generated by the pipeline). The obtained metadata attributes and their values are processed by two additional components of the pipeline: the *Data redundancy solver*, which deals with removing redundant attributes and their values, and the *Data renaming module*, which applies rules for renaming attributes. The remaining and possibly renamed attributes, along with their values, compose the final OpenGDC metadata file.

On the top left corner of Figure 1, we consider **Biospecimen and Clinical Supplements**; they are organized by *patient* (identified by the *bcr\_patient\_uid* attribute), with a patient typically related to many aliquots. Multiple OpenGDC metadata files are created, one for each aliquot reported in the patient biospecimen file. We replicate the full content of the Clinical Supplement of a patient over all metadata files regarding the aliquots of the patient. The resulting metadata attribute keys start with the *clinical\_\_* prefix. A Biospecimen Supplement, instead, contains a unique section on the patient, but also distinct sections on multiple samples, their portions, and the resulting aliquots. In each aliquot metadata file we replicate the common parts about the patient (and, in case, about related samples/portions), while the remaining content of the biospecimen file is divided among the different metadata files according to the specific aliquot each of them refers to. The resulting metadata attribute keys start with the *biospecimen\_\_* prefix.

On the bottom left corner of Figure 1, we query **GDC Data Model elements** using the GDC RESTful APIs. We call the API services once for each aliquot listed in a Biospecimen Supplement and each data type of interest, by specifying the *aliquot uid* and the *data type*, and then associate with each OpenGDC data file all information retrieved in the obtained response. The extracted attributes describe a data file along different GDC Data Model conceptual areas (i.e., administrative, biological, clinical and analysis). Relevant administrative entities include the PROGRAM (i.e., the broad framework of goals to be achieved by multiple experiments, such as TCGA), the PROJECT (i.e., the specifically defined piece of work that is undertaken or attempted to meet a single requirement, such as TCGA-LAML—which refers to Acute Myeloid Leukemia), the CASE (i.e., the collection of all data related to a specific subject in the context of a specific project, such as a patient). Among biological entities there are SAMPLE (i.e., any material sample taken from a biological entity for testing, diagnostic, propagation, treatment, or research purposes) and ALIQUOT (i.e., pertaining to a portion of the whole; any one of two or more samples of something, of the same volume or weight). Clinical entities include TREATMENT (i.e., therapeutic agents provided, or to be provided, to a patient to alter the course of a pathologic process) and DIAGNOSIS (i.e., data from the investigation, analysis and recognition of the presence and nature of disease, condition, or injury from expressed signs and symptoms). Analysis entities include harmonization pipelines such as “Copy Number Variation” and “Methylation Liftover”, each related to one data type.

In case an OpenGDC data file corresponds to  $n$  original GDC files, the JSON response to the corresponding API call is divided in  $n$  partitions, each containing information on one single GDC original file and on the related aliquot (the information of the latter one is replicated in each partition). Then, in the final OpenGDC metadata file, we group the information from the original files (by concatenating multiple values in a single key-value pair), while we consider the aliquot information only once. All these metadata attribute names are prefixed with *gdc\_\_* and obtained by flattening the hierarchical structure of the JSON responses, i.e., through concatenation of JSON keys at each traversed level of the response structure.

As an addition to GDC inputs, we generate a set of **manually curated key-value pairs** (gathered in the group of metadata keys prefixed with *manually\_curated\_\_*). These contain information that is missing in the GDC and derived from other sources or specified by our system. We add the data format (e.g., BED file textual format), URLs of the data and metadata files on the FTP server publicly offered by OpenGDC (see Section 3 for details about the OpenGDC software and the FTP repository), the genome built (i.e., reference assembly), the *id*, *checksum*, *size* and *download date* of the data file, and the status of the tissue, which indicates if it is of a normal or control sample.

Combining Clinical/Biospecimen Supplement information with GDC Data Model information leads to value redundancy, which is due to the fact that there does not exist a specific data model for the Supplement data and it is impossible to determine a priori which information are non-overlapping. We ascertained the presence of attributes holding different names but same semantics and associated values. We profiled all input data, obtaining sets of different keys that present same values within a

same metadata file. Example groups of key-value pairs with different keys and same value, along with the corresponding chosen candidate key preserved in each group, are shown in Table 1.

**Table 1.** Example of choices produced by the *Data redundancy solver*.

Preserved	Different Attributes	Values
×	biospecimen__bio__analyte_type	RNA
	gdc__cases__samples__portions__analytes__analyte_type	RNA
×	biospecimen__admin__day_of_dcc_upload	31
	clinical__admin__day_of_dcc_upload	31
×	gdc__cases__primary_site	Ovary
	gdc__cases__project__primary_site	Ovary
×	gdc__cases__samples__portions__analytes__aliquots__concentration	0.17
	gdc__cases__samples__portions__analytes__concentration	0.17

The preliminary profiling activity was used to provide guidance to create a list of data redundancy heuristics—with the aim to remove the redundant metadata attributes and their values—applied by the *Data redundancy solver* (at the center of Figure 1).

The heuristics have been primarily devised as a result of a long email exchange with the GDC Support team (support@nci-gdc.datacommons.io) that helped us to understand how the ingestion process works: a restricted number of attributes from the supplements are already provided with a defined mapping to the data model attributes, while for others the relation is still uncertain (i.e., not curated yet by the GDC)—for these we reconstructed common semantics through a semi-automated approach.

Moreover, clinical and biospecimen supplements cover overlapping semantics spaces (as it can be understood by their definitions in Section 2.2). Thus we make the deliberate decision of extracting only one of them.

Finally, the new data model entities are non overlapping but the APIs provide their content in a nested fashion. For example, a project is related to a case with a functional dependency, therefore the project information can be uniquely reached through the case entity. As a consequence, any information related to the case\_\_project group is redundant w.r.t. the one given by a dual attribute with the same suffix. Analogously, aliquots are comprised in analytes (N aliquots are in 1 analyte), therefore we keep the information that is most specific, pertaining to the aliquot.

We have summarized our approach to solve redundancy in four rules. These cover the whole space of possibilities at the time of writing this manuscript; however this set will be updated as the need for new rules will arise, in conjunction with updates of OpenGDC scheduled releases: The preliminary profiling activity was used to define the following list of heuristics to remove the redundant metadata attributes and their values, which is applied by the *Data redundancy solver* (at the center of Figure 1):

1. verify mappings on the official GDC GitHub repository available at [https://github.com/NCI-GDC/gdcdatamodel/tree/develop/gdcdatamodel/xml\\_mappings](https://github.com/NCI-GDC/gdcdatamodel/tree/develop/gdcdatamodel/xml_mappings), specifying which fields from the BCR Supplements correspond to the GDC API fields: when redundant, keep the second ones;
2. when a field from the BCR Biospecimen Supplement is redundant w.r.t. a field of the BCR Clinical Supplement, keep the first one;
3. when a field belonging to the *case* group is redundant w.r.t. a *case\_\_project* group field, keep the first one;
4. when a field belonging to the *analytes* group is redundant w.r.t. a *analytes\_\_aliquots* group field, keep the second one.

To facilitate the use of metadata key-value pairs, in case keys are very long and cumbersome, we simplify them through the *Data renaming module*, which applies renaming rules



according to a match-and-replace strategy based on regular expressions. With respect to the original keys retrieved from the GDC APIs, we usually leave unchanged the rightmost part (i.e., last subgroup and name of the attribute); this ensures that the attributes remain uniquely identified. As an example, `gdc__cases__samples__portions__analytes__aliquots__aliquot_id` becomes `gdc__aliquots__aliquot_id`. The three levels of the resulting attribute, separated by double underscore, identify respectively an attribute retrieved through the GDC APIs (“gdc”), belonging to the “aliquots” entity of the GDC Data Model, and indicating specifically the identifier of the represented aliquot (i.e., “aliquot\_id”). Examples of renaming rules and their results are shown in Table 2.

**Table 2.** Examples of metadata attribute renaming rules and their results.

GDC Naming	OpenGDC Flattened	OpenGDC Renamed
<code>cases.diagnoses.age_at_diagnosis</code>	<code>gdc_cases_diagnoses_age_at_diagnosis</code>	<code>gdc_diagnoses_age_at_diagnosis</code>
<code>analysis.input_files.data_category</code>	<code>gdc_analysis_input_files_data_category</code>	<code>gdc_input_files_data_category</code>
<code>cases.project.program.name</code>	<code>gdc_cases_project_program_name</code>	<code>gdc_program_name</code>

Column 1: attribute names as they are specified in GDC APIs parameters; Column 2: OpenGDC naming convention; Column 3: results of the renaming phase applied to the attributes in Column 2.

### 3. Results

In this Section we present the created OpenGDC software, which implements, for the GDC data, the mapping to the Genomic Data Model and the retrieval, extension and standardization procedures described in Section 2. Additionally, we illustrate the FTP repository where we provide the standardized genomic data and metadata obtained by applying OpenGDC to the GDC data of the TCGA program.

#### 3.1. *Opengdc Software Architecture*

OpenGDC is an open-source and cross-platform software, written in Java programming language; (The corresponding code is openly available on the GitHub repository: <https://github.com/DEIB-GECO/OpenGDC>) it allows the extraction, extension and standardization of publicly available data from the GDC. The software is provided as a standalone desktop application with a friendly user interface and supports the BED, GTF, CSV, JSON and XML standard formats as output. Its architecture has been implemented following the Model-View-Controller (MVC) design pattern, as shown by the flowchart in Figure 2. The software is composed of two main pipelines: (i) the GDC data download procedure and (ii) the data conversion one.

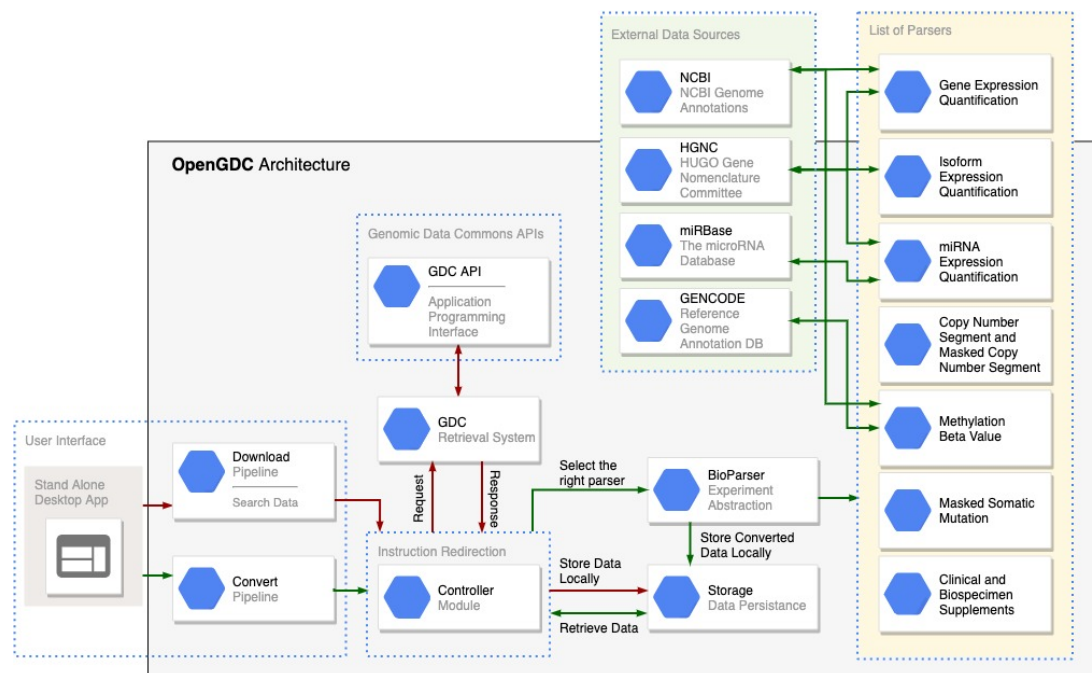
The whole system can be summarized by three main software components:

- *Controller*: it redirects the user instructions to the correct software module and initializes an instance of the software able to download and/or convert the GDC data;
- *Data Download*: it manages the process of search and retrieval of the public GDC data, taking advantage of the GDC APIs;
- *Data Standardization*: it allows to easily convert and standardize data according to a specific data type. The process is facilitated by the ad-hoc class *BioParser*, which provides an abstract representation for all GDC data types; this class can be extended to support new data types in case of future extensions of the GDC repository.

OpenGDC benefits from the public GDC APIs during the data download procedure to retrieve the original genomic, clinical and biospecimen data. It also makes use of the GDC APIs during the conversion procedure of the Clinical and Biospecimen Supplements to extract additional information such as the size of the downloaded files, their MD5 checksum, as well as the last creation and update timestamps, which are then added in the metadata files.

Data conversion uses a different parser depending on the type of converted data. Additionally, the process retrieves complementary information from a set of external data sources, such as NCBI

Genome and Gene databases, GENCODE, HGNC and miRBase, to extract the genomic coordinates, Entrez Gene ID and gene symbols starting from the information already existing in the original data.



**Figure 2.** OpenGDC architecture. Graphical representation of the flowchart describing the OpenGDC software architecture. Every feature is differentiated in two pipelines, i.e., Download and Convert, represented by red and green arrows, respectively. Software modules are additionally enclosed in a dotted line to delineate their function (i.e., User Interface, Instruction Redirection, Genomic Data Commons APIs, External Data Sources, and the List of Parsers).

### 3.2. Interacting with the GDC Public Apis

We search and extract data and other information from the GDC through its public APIs. In particular we use three main API endpoints:

- *cases*: to find all files related to a specific case (i.e., sample donor);
- *files*: to find all files with specific characteristics such as the file name, MD5 checksum and data format;
- *data*: to download GDC data files.

As an example of interplay among two of these endpoints, consider a scenario where we want to download all publicly available *Gene Expression Quantification* data for the tumor *Breast Invasive Carcinoma* in the context of the *TCGA* program. First, we query the GDC for all file unique identifiers (UUIDs) related to this particular case. To this end, we make an HTTP POST request to the *files* endpoint. As a result, the GDC returns a list of file UUIDs (*file\_id* fields). Starting from this list, we then download the associated files; this is done by querying the *data* endpoint specifying a single file UUID, e.g., <https://api.gdc.cancer.gov/data/1837ad2a-4edf-4d80-9050-f78115e54454> (i.e., one HTTP GET request for each result *file\_id* in the previous query response). For a detailed description about the syntax of the payload and the other ways to query the GDC, the interested reader may refer to the GDC API documentation available at <https://docs.gdc.cancer.gov/>. For additional details about the OpenGDC software and its usage we point the reader to the user guide and readme file, available as Supplementary File 2 and Supplementary File 3, respectively.

### 3.3. Data Repository

We created an open access FTP repository containing all the publicly available data of the TCGA program of the GDC in their original and new standardized extended version (genomic data in BED format and metadata in key-value format). The repository is available at [42]. The data are firstly divided in two branches, original GDC data and extended BED ones (*original* and *bed* folders, respectively). The structure of the FTP space is then organized within the two branches using the following structure: *program* (e.g., TCGA), *tumor* (e.g., TCGA-BRCA, TCGA-KIRP, TCGA-OV, etc.), and finally *data type* (e.g., *gene\_expression\_quantification*, *methylation\_beta\_value*, *clinical\_and\_biospecimen\_supplements*, etc.). For each data type, the genomic and metadata are separately provided for each aliquot. Currently, a total volume of 2.7 TB of data (1.4 TB of original GDC data and 1.3 TB of converted data) of 33 different tumors is maintained. Table 3 shows the number of aliquots, patients and samples available for each tumor.

**Table 3.** List of processed tumors with the related number of involved aliquots, samples and patients.

Tumor	Aliquots	Samples	Patients
Acute Myeloid Leukemia	1605	1605	1211
Adrenocortical Carcinoma	771	771	595
Bladder Urothelial Carcinoma	3786	3762	2873
Brain Lower Grade Glioma	4674	4674	3590
Breast Invasive Carcinoma	10,305	10,280	7520
Cervical Squamous Cell Carcinoma and Endocervical Adenocarcinoma	2706	2706	2118
Cholangiocarcinoma	401	401	267
Colon Adenocarcinoma	4358	4244	3121
Esophageal Carcinoma	1705	1701	1271
Glioblastoma Multiforme	3347	3282	2190
Head and Neck Squamous Cell Carcinoma	4955	4951	3636
Kidney Chromophobe	667	667	462
Kidney Renal Clear Cell Carcinoma	5322	5155	3499
Kidney Renal Papillary Cell Carcinoma	2812	2784	2023
Liver Hepatocellular Carcinoma	3604	3602	2610
Lung Adenocarcinoma	5245	5146	3722
Lung Squamous Cell Carcinoma	4780	4736	3460
Lymphoid Neoplasm Diffuse Large B-cell Lymphoma	423	423	327
Mesothelioma	775	775	603
Ovarian Serous Cystadenocarcinoma	4825	4777	3586
Pancreatic Adenocarcinoma	1659	1659	1267
Pheochromocytoma and Paraganglioma	1652	1652	1253
Prostate Adenocarcinoma	4778	4778	3473
Rectum Adenocarcinoma	1462	1453	1124
Sarcoma	2341	2335	1797
Skin Cutaneous Melanoma	4197	4197	3242
Stomach Adenocarcinoma	4108	4080	3018
Testicular Germ Cell Tumors	1377	1377	1045
Thymoma	1120	1120	862
Thyroid Carcinoma	4827	4827	3523
Uterine Carcinosarcoma	504	504	398
Uterine Corpus Endometrial Carcinoma	5088	5058	3860
Uveal Melanoma	720	720	560

Structure and content of the created FTP repository are described in detail in the Supplementary File 4 and Supplementary File 5, respectively.

#### 4. Use Case Examples

In this Section, we show some examples of application of the GenoMetric Query Language (GMQL [28]) on the OpenGDC standardized data in order to highlight the advantages of our data representation in terms of information retrieval and integrative processing. GMQL is a high-level domain-specific query language. It can be executed in the system architecture described in [29], which is specific for genomic data processing. The current available version of the GMQL system uses Apache Spark [43] as its backbone; along with other design choices, this provides high scalability in cloud computing. The GMQL system contains a multiplicity of public genomic datasets from a variety of sources [44], ready to be used within tertiary analysis pipelines (as shown in [29]); among other sources, it features all the datasets available in the OpenGDC FTP service, providing an interface for browsing and processing data curated in OpenGDC. The produced datasets are also made available within another system, GenoSurf (GenoSurf is available at [45]) [46], a semantic search engine based on a Conceptual Model [47] that integrates TCGA data, imported by OpenGDC, with several sources such as ENCODE [48], Roadmap Epigenomics [49], and 1000 Genomes [50], among others, using the META-BASE integration framework [51].

In the following, we propose three use cases along with their GMQL queries (the corresponding GMQL queries are available in the Supplementary File 6, ready to be executed on [52]) (alternatively expressible using the Python package [53]); we focus on query aspects, acting on both region data and metadata, which highlight the strengths of the datasets produced by OpenGDC, i.e.: (1) enabling the combined use of metadata derived from the GDC Data Model, the Clinical/Biospecimen Supplements, and our manually curated additions; (2) providing positional information (i.e., genomic coordinates) in a standardized structure, which encourages data inter- and intra-source interoperability; (3) allowing joined use of different data types even from different sources (e.g., gene expression and methylation, or mutations and gene annotations) based on common gene identifiers (e.g., the HUGO gene symbol), or genomic positions.

##### 4.1. Use Case 1: for Kidney Cancers, Find Mutations and Their Number in Each Exon

For this example, we consider TCGA public somatic mutation data samples of Kidney Adenoma and Adenocarcinoma patients—which are contained in three TCGA projects, i.e., Kidney Chromophobe (KICH), Kidney Renal Clear Cell Carcinoma (KIRC) and Kidney Renal Papillary Cell Carcinoma (KIRP)—and extract novel mutations (i.e., not listed in dbSNP [54]) in gene exons. For each sample, we count the mutations occurring in each exon, filter-out the exons without any mutation, and finally return the remaining mutated exons, equipped with their number and the maximum number of mutations in one exon.

In this example: (1) we use GDC mutation data in combination with a GENCODE annotation dataset—demonstrating the interoperability of OpenGDC curated data with other sources; (2) we use seamlessly metadata from GDC APIs (i.e., first and second conditions in line 2 in Listing 1) and Clinical Supplements (third and fourth conditions in line 3 and 4 in Listing 1)—this is not possible on the GDC portal, where only the former are supported; (3) we select three TCGA projects together by using the characterization of the tissue and the classification of diseases (note that the OpenGDC normalized metadata attribute *gdc\_disease\_type* represents the type of malignant disease (The disease is categorized by the World Health Organization's (WHO) International Classification of Diseases for Oncology (ICD-O).), while the attribute *gdc\_project\_disease\_type* contains the full name for the project. The output dataset contains in total 227 samples with 15,517 exon regions and 296 distinct metadata attributes.

```

1 #Select mutation data based on both region and metadata attributes
2 MUT = SELECT(gdc__primary_site == "Kidney" AND gdc__disease_type == "Adenomas and Adenocarcinomas" AND
3 clinical__shared__history_of_neoadjuvant_treatment == "No" AND
4 clinical__clin_shared__followup_treatment_success == "Complete Remission/Response";
5 region: dbsnp_rs == "novel") GRCh38_TCGA_somatic_mutation_masked_2019_10;
6 #Select known human protein-coding and non-protein-coding exon regions of the GENCODE annotation release 27
7 EXON = SELECT(annotation_type == "exon" AND release_version == "27") GRCh38_ANNOTATION_GENCODE;
8 #Map the mutations to the exons and count how many they are in each exon of each sample
9 EXON_MUT = MAP(count_name: MUT_count) EXON MUT;
10 #Remove exons that do not contain mutations
11 EXON_MUT_SELECT = SELECT(region: MUT_count > 0) EXON_MUT;
12 #In the metadata of each sample add the count of how many exons remain and the maximum number of mutations in
13 #an exon of the sample
14 EXON_RES = EXTEND(exon_count AS COUNT(), max_mut AS MAX(MUT_count)) EXON_MUT_SELECT;
15 MATERIALIZE EXON_RES INTO result1_exons_mutations;

```

**Listing 1.** Example of GenoMetric Query Language (GMQL) query to find exons with somatic mutations in kidney cancers.

#### 4.2. Use Case 2: in Breast Invasive Carcinoma, Find the Genomic Regions Whose Mirna Expression Counts Result above Average in at Least 10 % of Tumoral Samples

We translate these specifications into selecting TCGA miRNA expression samples corresponding to patients who are affected by primary tumors of Breast Invasive Carcinoma, and into selecting the miRNA regions that exhibit a value of *reads\_per\_million\_mirna\_mapped* (In the miRNA Expression Quantification data type, it is the read normalized count in reads-per-million-miRNA-mapped associated with each miRNA ID.) above the average of the dataset in 10% or more of such samples. We first use a simple query (lines 3–8 in Listing 2) to evaluate the average of miRNA normalized reads. In order to obtain the lightest query possible in terms of computational time, from the selected TCGA dataset we PROJECT only the required field, MERGE all samples into one, compute the average as a metadata attribute (*avg\_reads*) and MATERIALIZE a small dataset in order to get the required average value (531.6 for the considered data). We then perform a query to filter out miRNA regions that present a *reads\_per\_million\_mirna\_mapped* value equal or below the calculated average of the dataset (lines 11–13 in Listing 2). In addition, we use COVER to extract in one sample only the remaining miRNA regions that are present in at least 10% of the dataset samples and equip each extracted region with: (1) the number of samples in which the region is expressed above average; (2) the list of co-located genes, using specifically the *entrez\_gene\_id* region attribute—which is a new attribute added in the OpenGDC data, with respect to the original GDC data. The output dataset contains a sample with 102 miRNA regions (with *reads\_per\_million\_mirna\_mapped* above average) out of the 1881 distinct ones considered in the initial dataset.



```

1 #This first query materializes a dataset from whose single metadata attribute (avg_reads) we read the average
2 #number of reads in the dataset, which is then used as threshold in the second query
3 S0 = SELECT(gdc__project__disease_type == "Breast Invasive Carcinoma" AND
4 gdc__samples__sample_type == "Primary Tumor") GRCh38_TCGA_miRNA_expression_2019_10;
5 P = PROJECT(reads_per_million_mirna_mapped; metadata: none) S0;
6 M = MERGE() P;
7 E = EXTEND(avg_reads AS AVG(reads_per_million_mirna_mapped)) M;
8 MATERIALIZE E INTO result2_reads_threshold;

10 #Find miRNA regions with a number of reads above the average in the dataset
11 S = SELECT(gdc__project__disease_type == "Breast Invasive Carcinoma" AND
12 gdc__samples__sample_type == "Primary Tumor"; region: reads_per_million_mirna_mapped > 531.6)
13 GRCh38_TCGA_miRNA_expression_2019_10;
14 #Find genomic regions present in more than 10% of samples; for each
15 #region report a list of overlapping genes and the number of samples in
16 #which it occurs
17 C = COVER(ALL / 10, ANY; aggregate: num_samples AS COUNT(), all_genes AS BAGD(entrez_gene_id)) S;
18 MATERIALIZE C INTO result2_cover;

```

**Listing 2.** Example of GMQL query that finds miRNA regions with expression above average in more than 10% of samples and the associated genes.

#### 4.3. Use Case 3: in a Comparative Study, For Both Normal and Tumoral Tissue Samples of Each Patient Affected by Cholangiocarcinoma Extract the Expression and Average Promotorial Methylation Levels of Each Gene

In the OpenGDC standardized data of TCGA, using with value “normal” our *manually\_curated\_\_tissue\_status* metadata attribute, added with respect to the original GDC data, we can select normal samples of five different types at once (i.e., Blood Derived Normal, Solid Tissue Normal, Buccal Cell Normal, EBV Immortalized Normal, Bone Marrow Normal—corresponding to sample type codes 10–14 <https://gdc.cancer.gov/resources-tcga-users/tcga-code-tables/sample-type-codes>). Similarly, the value “tumoral” of the same attribute refers to ten different types of cancer samples (corresponding to sample type codes 01–09 and 40). Since methylation sites of interest for gene expression regulation are typically located in the surroundings of a gene TSS, we consider methylation values only in the promotorial region of each gene, extracted around the gene TSS from 2000 bases upstream to 1000 bases downstream (lines 5 and 19 in Listing 3); for gene expression data we only keep the *fpkm* values and the *gene\_symbol* (line 5 in Listing 3), while for methylation data only the *beta\_values* (line 10 in Listing 3).

Note that the code described in Listing 3 lines 1–13 for normal samples is repeated in lines 15–27 for tumoral samples. For methylation data, we compute the average *beta\_value* in each gene promoter. With the MAP at line 13 we associate each gene expression and promotorial region (in each sample of the normal N\_EXPR dataset) with the average of the methylation *beta\_values* in the gene promotorial region (in a sample of the normal N\_METH dataset); N\_EXPR and N\_METH samples are matched only if belonging to the same tissue sample (uniquely identified by the *gdc\_\_samples\_\_sample\_id*).

At Listing 3 line 30 the datasets resulting from line 13 and line 27 are combined using a JOIN operation, which allows associating each gene promotorial region with the *gene\_symbol* and the gene expression *fpkm* value and methylation *avg\_beta\_value* from both the normal and tumoral samples of a patient. Note that the equi predicate on attributes can only be applied thanks to the addition of the *gene\_symbol* attribute in the OpenGDC gene expression data (as original GDC data did not include it).

Lines 33–38 in Listing 3 are only needed for shaping results into a convenient format, as it can be appreciated in Table 4, which contains an excerpt from the result dataset (in the column names of Table 4 we use the subscripts *n* and *t* for *normal* and *tumoral*, respectively).



Occurrences of null in the average beta values correspond to cases where no methylation probes are located in the specified gene promotorial region. Overall, the output dataset contains 9 samples with about 60,670 distinct regions each.

## 5. Conclusions and Future Work

In this work, we presented a novel approach and its implementation in a set of automatic procedures able to extract, integrate, extend and standardize genomic and clinical data of The Cancer Genome Atlas as included in the Genomic Data Commons portal. Our approach and software were applied to multiple data types obtained from different types of NGS experiments (i.e., Gene-, miRNA-, Isoform-Expression Quantification, Masked Somatic Mutation, Copy Number Segment, Masked Copy Number Segment, Methylation Beta Value). Additionally, we considered clinical and biospecimen information about the experimental data.

To reach our objective, we took advantage of the Genomic Data Model, which allowed us to represent an experimental sample by its genomic regions and its related metadata. The genomic regions are defined by their genomic coordinates (chr, left, right, strand) and genomic features, which are produced by the specific NGS experiment. Conversely, metadata report clinical and biological properties in attribute-value pair format.

Based on the GDM representation, we implemented OpenGDC, a software for retrieving TCGA experimental data in the GDC portal, which is then processed with ad-hoc procedures for each data type. Our standardization procedure provides all the data in free-BED format, which contains a set of experiment-specific fields in addition to the genomic coordinates. In order to obtain this standardized format, the software is able to automatically extract additional features from external data sources (e.g., GENCODE, HGNC and miRBase), which are not provided in the original GDC data files. The software also integrates experimental data with clinical and biospecimen information derived from different GDC sources.

Our pipeline extracts metadata attributes from the original Clinical and Biospecimen Supplements and from the GDC RESTful APIs. The obtained attributes are merged in a single metadata file for each experiment, using a tab-delimited key-value format. Then, two software components are used in the metadata pipeline: (i) the Data Redundancy Solver, to detect and remove redundant metadata attributes, and (ii) the Data Renaming Module, to redefine attribute names. In particular, data profiling is performed to identify redundant attributes, i.e., with the same values and different names. All these procedures and their input/output data types are thoroughly described in the OpenGDC Format Definition document, available as Additional File 1.

We collected the standardized genomic data and metadata in a FTP repository, which we made publicly available at [42]. We also showed usage examples of these data through the application of GMQL queries, to highlight the validity and utility of our approach. They demonstrate that our data representation facilitates data retrieval, integrated processing and analyses, especially thanks to the combination of the filtering on specific clinical/biospecimen attributes and the extraction of genomic features.

Future work concerns the application of our data representation and software pipeline to other projects integrated in the GDC portal and to other cancer-related repositories, in order to facilitate knowledge discovery over multiple cancer data. Additionally, we plan to use our approach and software in order to further enhance the data integration among different biomedical public repositories. Finally, we are going to take advantage of the standardized data, which is easily processable by several state of the art bioinformatics tools, in order to perform new knowledge extraction analyses about cancer.

**Supplementary Materials:** The following are available at <http://www.mdpi.com/2076-3417/10/18/6367/>, Supplementary File 1—OpenGDC\_format\_definition.pdf, PDF file that contains, for each considered data types included in OpenGDC: the format definition, the format conversion details from the GDC original TCGA format into OpenGDC format, and external database integration specifications. (Available at <http://www.mdpi.com/2076-3417/10/18/6367/>)

[//www.bioinformatics.deib.polimi.it/opengdc/](http://www.bioinformatics.deib.polimi.it/opengdc/)); Supplementary File 2—OpenGDC\_User\_Guide.pdf, PDF file that contains a practical guide to introduce OpenGDC software use; Supplementary File 3—OpenGDC\_readme.txt, Text file that includes installation and execution details of the OpenGDC software package; Supplementary File 4—OpenGDC\_repository\_description.txt, Text file that details the content and the structure of the OpenGDC data repository; Supplementary File 5—OpenGDC\_statistics.xlsx, Spreadsheet file that contains 3 sheets: (i) *All\_statistics*, including patient, sample and aliquot counts for each tumor and experiment type; (ii) *Counts\_for\_each\_experiment*, including the occurrences of the patients, samples and aliquots for each experiment; (iii) *Total\_counts\_for\_each\_tumor*, including the occurrences of the patients, samples and aliquots for each tumor; Supplementary File 6—OpenGDC\_GMQL\_queries.txt, Text file including the ready-to-run GMQL queries described in Section 4 to reproduce the experiments on [52].

**Author Contributions:** E.W., M.M., and S.C. directed the work. F.C., E.C., E.W., M.M., A.B., and A.C. wrote the manuscript. F.C. and E.C. performed software and F.T.P. server design and implementation. A.B. analyzed and structured the metadata. F.C. and E.C. downloaded, converted and processed the data. M.M., S.C., and E.W. conceived the research. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work has been supported by the ERC Advanced Grant 693174 “Data-Driven Genomic Computing (GeCo)” project (2016-2021), funded by the European Research Council, which covered also the publication costs.

**Acknowledgments:** The results reported here are based upon the data maintained by the National Cancer Institute’s Genomic Data Commons: <https://gdc.cancer.gov/>.

**Availability of Software and Data:** OpenGDC software is freely available at <http://www.bioinformatics.deib.polimi.it/opengdc/>. A repository with the homogenized and enhanced TCGA data from the GDC is publicly accessible at <ftp://geco.deib.polimi.it/opengdc/bed/>. Finally, the source code of OpenGDC is also available on GitHub at <https://github.com/DEIB-GECO/OpenGDC> and the version used to perform the experiments described in this manuscript is available on Zenodo at <https://doi.org/10.5281/zenodo.4000250>.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

API	Application Programming Interface
BED	Browser Extensible Data format
BCR	Biospecimen Core Repository
CGC	Cancer Genomics Cloud
CNV	copy number variation
GDC	Genomic Data Commons
GDM	Genomic Data Model
GMQL	GenoMetric Query Language
ICD-O	International Classification of Diseases for Oncology
ICGC	International Cancer Genome Consortium
KICH	Kidney Chromophobe
KIRK	Kidney Renal Clear Cell Carcinoma
KIRP	Kidney Renal Papillary Cell Carcinoma
MAF	Mutation Annotation Format
MVC	Model-View-Controller
NCI	National Cancer Institute
NGS	Next Generation Sequencing
TCGA	The Cancer Genome Atlas
TSS	transcription start site
UUID	Universal Unique Identifier
WHO	World Health Organization.

## References

1. Metzker, M.L. Sequencing technologies—The next generation. *Nat. Rev. Genet.* **2010**, *11*, 31. [[CrossRef](#)] [[PubMed](#)]
2. Weitschek, E.; Santoni, D.; Fiscon, G.; De Cola, M.C.; Bertolazzi, P.; Felici, G. Next generation sequencing reads comparison with an alignment-free distance. *BMC Res. Notes* **2014**, *7*, 869. [[CrossRef](#)] [[PubMed](#)]

3. Kamps, R.; Brandão, R.; Bosch, B.; Paulussen, A.; Xanthoulea, S.; Blok, M.; Romano, A. Next-generation sequencing in oncology: Genetic diagnosis, risk prediction and cancer classification. *Int. J. Mol. Sci.* **2017**, *18*, 308. [[CrossRef](#)] [[PubMed](#)]
4. Ozsolak, F.; Milos, P.M. RNA sequencing: Advances, challenges and opportunities. *Nat. Rev. Genet.* **2011**, *12*, 87. [[CrossRef](#)] [[PubMed](#)]
5. Zhang, Y.; Jeltsch, A. The application of next generation sequencing in DNA methylation analysis. *Genes* **2010**, *1*, 85–101. [[CrossRef](#)] [[PubMed](#)]
6. Alkan, C.; Kidd, J.M.; Marques-Bonet, T.; Aksay, G.; Antonacci, F.; Hormozdiari, F.; Kitzman, J.O.; Baker, C.; Malig, M.; Mutlu, O.; et al. Personalized copy number and segmental duplication maps using next-generation sequencing. *Nat. Genet.* **2009**, *41*, 1061. [[CrossRef](#)]
7. Bellazzi, R. Big data and biomedical informatics: A challenging opportunity. *Yearb. Med. Inform.* **2014**, *23*, 08–13. [[CrossRef](#)]
8. Luo, J.; Wu, M.; Gopukumar, D.; Zhao, Y. Big data application in biomedical research and health care: A literature review. *Biomed. Inform. Insights* **2016**, *8*, BII-S31559. [[CrossRef](#)]
9. Grossman, R.L.; Heath, A.P.; Ferretti, V.; Varmus, H.E.; Lowy, D.R.; Kibbe, W.A.; Staudt, L.M. Toward a shared vision for cancer genomic data. *New Engl. J. Med.* **2016**, *375*, 1109–1112. [[CrossRef](#)]
10. Jensen, M.A.; Ferretti, V.; Grossman, R.L.; Staudt, L.M. The NCI Genomic Data Commons as an engine for precision medicine. *Blood* **2017**, *130*, 453–459. [[CrossRef](#)]
11. Timmermann, B.; Kerick, M.; Roehr, C.; Fischer, A.; Isau, M.; Boerno, S.T.; Wunderlich, A.; Barmeyer, C.; Seemann, P.; Koenig, J.; et al. Somatic mutation profiles of MSI and MSS colorectal cancer identified by whole exome next generation sequencing and bioinformatics analysis. *PLoS ONE* **2010**, *5*, e15661. [[CrossRef](#)] [[PubMed](#)]
12. Mortazavi, A.; Williams, B.A.; McCue, K.; Schaeffer, L.; Wold, B. Mapping and quantifying mammalian transcriptomes by RNA-Seq. *Nat. Methods* **2008**, *5*, 621. [[CrossRef](#)] [[PubMed](#)]
13. Trapnell, C.; Williams, B.A.; Pertea, G.; Mortazavi, A.; Kwan, G.; Van Baren, M.J.; Salzberg, S.L.; Wold, B.J.; Pachter, L. Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nat. Biotechnol.* **2010**, *28*, 511. [[CrossRef](#)] [[PubMed](#)]
14. Zeng, Y.; Cullen, B.R. Sequence requirements for micro RNA processing and function in human cells. *RNA* **2003**, *9*, 112–123. [[CrossRef](#)]
15. Conrad, D.F.; Pinto, D.; Redon, R.; Feuk, L.; Gokcumen, O.; Zhang, Y.; Aerts, J.; Andrews, T.D.; Barnes, C.; Campbell, P.; et al. Origins and functional impact of copy number variation in the human genome. *Nature* **2010**, *464*, 704. [[CrossRef](#)]
16. Bibikova, M.; Barnes, B.; Tsan, C.; Ho, V.; Klotzle, B.; Le, J.M.; Delano, D.; Zhang, L.; Schroth, G.P.; Gunderson, K.L.; et al. High density DNA methylation array with single CpG site resolution. *Genomics* **2011**, *98*, 288–295. [[CrossRef](#)]
17. Weinstein, J.N.; Collisson, E.A.; Mills, G.B.; Shaw, K.R.M.; Ozenberger, B.A.; Ellrott, K.; Shmulevich, I.; Sander, C.; Stuart, J.M.; Network, C.G.A.R.; et al. The Cancer Genome Atlas pan-cancer analysis project. *Nat. Genet.* **2013**, *45*, 1113. [[CrossRef](#)]
18. Liu, J.; Lichtenberg, T.; Hoadley, K.A.; Poisson, L.M.; Lazar, A.J.; Cherniack, A.D.; Kovatich, A.J.; Benz, C.C.; Levine, D.A.; Lee, A.V.; et al. An integrated TCGA pan-cancer clinical data resource to drive high-quality survival outcome analytics. *Cell* **2018**, *173*, 400–416. [[CrossRef](#)]
19. Cappelli, E.; Felici, G.; Weitschek, E. Combining DNA methylation and RNA sequencing data of cancer for supervised knowledge extraction. *Biodata Min.* **2018**, *11*, 22. [[CrossRef](#)]
20. Celli, F.; Cumbo, F.; Weitschek, E. Classification of large DNA methylation datasets for identifying cancer drivers. *Big Data Res.* **2018**, *13*, 21–28. [[CrossRef](#)]
21. Weitschek, E.; Cumbo, F.; Cappelli, E.; Felici, G. Genomic data integration: A case study on next generation sequencing of cancer. In Proceedings of the 2016 27th International Workshop on Database and Expert Systems Applications (DEXA), Porto, Portugal, 5–8 September 2016; pp. 49–53.
22. Harrow, J.; Frankish, A.; Gonzalez, J.M.; Tapanari, E.; Diekhans, M.; Kokocinski, F.; Aken, B.L.; Barrell, D.; Zadissa, A.; Searle, S.; et al. GENCODE: The reference human genome annotation for The ENCODE Project. *Genome Res.* **2012**, *22*, 1760–1774. [[CrossRef](#)] [[PubMed](#)]
23. Eyre, T.A.; Ducluzeau, F.; Sneddon, T.P.; Povey, S.; Bruford, E.A.; Lush, M.J. The HUGO gene nomenclature database, 2006 updates. *Nucleic Acids Res.* **2006**, *34*, D319–D321. [[CrossRef](#)] [[PubMed](#)]



24. Griffiths-Jones, S.; Saini, H.K.; van Dongen, S.; Enright, A.J. miRBase: Tools for microRNA genomics. *Nucleic Acids Res.* **2007**, *36*, D154–D158. [[CrossRef](#)] [[PubMed](#)]
25. Sayers, E.W.; Agarwala, R.; Bolton, E.E.; Brister, J.R.; Canese, K.; Clark, K.; Connor, R.; Fiorini, N.; Funk, K.; Hefferon, T.; et al. Database resources of the national center for biotechnology information. *Nucleic Acids Res.* **2019**, *47*, D23. [[CrossRef](#)]
26. Cumbo, F.; Fiscon, G.; Ceri, S.; Masseroli, M.; Weitschek, E. TCGA2BED: Extracting, extending, integrating, and querying The Cancer Genome Atlas. *BMC Bioinform.* **2017**, *18*, 6. [[CrossRef](#)]
27. Masseroli, M.; Kaitoua, A.; Pinoli, P.; Ceri, S. Modeling and interoperability of heterogeneous genomic big data for integrative processing and querying. *Methods* **2016**, *111*, 3–11. [[CrossRef](#)]
28. Masseroli, M.; Pinoli, P.; Venco, F.; Kaitoua, A.; Jalili, V.; Palluzzi, F.; Muller, H.; Ceri, S. GenoMetric Query Language: A novel approach to large-scale genomic data management. *Bioinformatics* **2015**, *31*, 1881–1888. [[CrossRef](#)]
29. Masseroli, M.; Canakoglu, A.; Pinoli, P.; Kaitoua, A.; Gulino, A.; Horlova, O.; Nanni, L.; Bernasconi, A.; Perna, S.; Stamoulakatou, E.; et al. Processing of big heterogeneous genomic datasets for tertiary analysis of Next Generation Sequencing data. *Bioinformatics* **2018**, *35*, 729–736. [[CrossRef](#)]
30. Wei, L.; Jin, Z.; Yang, S.; Xu, Y.; Zhu, Y.; Ji, Y. TCGA-assembler 2: Software pipeline for retrieval and processing of TCGA/CPTAC data. *Bioinformatics* **2017**, *34*, 1615–1617. [[CrossRef](#)]
31. Zhang, J.; Baran, J.; Cros, A.; Guberman, J.M.; Haider, S.; Hsu, J.; Liang, Y.; Rivkin, E.; Wang, J.; Whitty, B.; et al. International Cancer Genome Consortium Data Portal — A one-stop shop for cancer genomics data. *Database* **2011**, *2011*, bar026. [[CrossRef](#)]
32. Lau, J.W.; Lehnert, E.; Sethi, A.; Malhotra, R.; Kaushik, G.; Onder, Z.; Groves-Kirkby, N.; Mihajlovic, A.; DiGiovanna, J.; Srdic, M.; et al. The Cancer Genomics Cloud: Collaborative, reproducible, and democratized—A new paradigm in large-scale computational research. *Cancer Res.* **2017**, *77*, e3–e6. [[CrossRef](#)] [[PubMed](#)]
33. Colaprico, A.; Silva, T.C.; Olsen, C.; Garofano, L.; Cava, C.; Garolini, D.; Sabedot, T.S.; Malta, T.M.; Pagnotta, S.M.; Castiglioni, I.; et al. TCGAAbiolinks: An R/Bioconductor package for integrative analysis of TCGA data. *Nucleic Acids Res.* **2016**, *44*, e71. [[CrossRef](#)] [[PubMed](#)]
34. Cerami, E.; Gao, J.; Dogrusoz, U.; Gross, B.E.; Sumer, S.O.; Aksoy, B.A.; Jacobsen, A.; Byrne, C.J.; Heuer, M.L.; Larsson, E.; et al. The cBio cancer genomics portal: An open platform for exploring multidimensional cancer genomics data. *Cancer Discov.* **2012**, *2*, 401–404. [[CrossRef](#)] [[PubMed](#)]
35. Goldman, M.; Craft, B.; Brooks, A.; Zhu, J.; Haussler, D. The UCSC Xena Platform for cancer genomics data visualization and interpretation. *BioRxiv* **2018**, 26470. [[CrossRef](#)]
36. Settino, M.; Cannataro, M. Survey of main tools for querying and analyzing TCGA data. In Proceedings of the 2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Madrid, Spain, 3–6 December 2018; pp. 1711–1718.
37. Quinlan, A.R.; Hall, I.M. BEDTools: A flexible suite of utilities for comparing genomic features. *Bioinformatics* **2010**, *26*, 841–842. [[CrossRef](#)]
38. Fan, Y.; Xi, L.; Hughes, D.S.; Zhang, J.; Zhang, J.; Futreal, P.A.; Wheeler, D.A.; Wang, W. MuSE: Accounting for tumor heterogeneity using a sample-specific error model improves sensitivity and specificity in mutation calling from sequencing data. *Genome Biol.* **2016**, *17*, 178. [[CrossRef](#)]
39. Larson, D.E.; Harris, C.C.; Chen, K.; Koboldt, D.C.; Abbott, T.E.; Dooling, D.J.; Ley, T.J.; Mardis, E.R.; Wilson, R.K.; Ding, L. SomaticSniper: Identification of somatic point mutations in whole genome sequencing data. *Bioinformatics* **2011**, *28*, 311–317. [[CrossRef](#)]
40. Cibulskis, K.; Lawrence, M.S.; Carter, S.L.; Sivachenko, A.; Jaffe, D.; Sougnez, C.; Gabriel, S.; Meyerson, M.; Lander, E.S.; Getz, G. Sensitive detection of somatic point mutations in impure and heterogeneous cancer samples. *Nat. Biotechnol.* **2013**, *31*, 213. [[CrossRef](#)]
41. Koboldt, D.C.; Zhang, Q.; Larson, D.E.; Shen, D.; McLellan, M.D.; Lin, L.; Miller, C.A.; Mardis, E.R.; Ding, L.; Wilson, R.K. VarScan 2: Somatic mutation and copy number alteration discovery in cancer by exome sequencing. *Genome Res.* **2012**, *22*, 568–576. [[CrossRef](#)]
42. OpenGDC FTP Repository. Available online: <ftp://geco.deib.polimi.it/opengdc/bed/> (accessed on 31 May 2020).
43. Apache Spark. Available online: <http://spark.apache.org/> (accessed on 31 May 2020).

44. Bernasconi, A.; Canakoglu, A.; Masseroli, M.; Ceri, S. The road towards data integration in human genomics: Players, steps and interactions. *Briefings Bioinform.* **2020**, *bbaa080*. [[CrossRef](#)]
45. GenoSurf. Available online: <http://www.gmql.eu/genosurf> (accessed on 31 May 2020).
46. Canakoglu, A.; Bernasconi, A.; Colombo, A.; Masseroli, M.; Ceri, S. GenoSurf: Metadata driven semantic search system for integrated genomic datasets. *Database J. Biol. Databases Curation* **2019**, *2019*, baz132. [[CrossRef](#)] [[PubMed](#)]
47. Bernasconi, A.; Ceri, S.; Campi, A.; Masseroli, M. Conceptual Modeling for Genomics: Building an Integrated Repository of Open Data. In *Conceptual Modeling*; Mayr, H.C., Guizzardi, G., Ma, H., Pastor, O., Eds.; Springer International Publishing: Cham, Switzerland, 2017; pp. 325–339.
48. Encode, C. An integrated encyclopedia of DNA elements in the human genome. *Nature* **2012**, *489*, 57–74.
49. Kundaje, A.; Meuleman, W.; Ernst, J.; Bilenky, M.; Yen, A.; Heravi-Moussavi, A.; Kheradpour, P.; Zhang, Z.; Wang, J.; Ziller, M.J.; et al. Integrative analysis of 111 reference human epigenomes. *Nature* **2015**, *518*, 317. [[CrossRef](#)] [[PubMed](#)]
50. Consortium, G.P. A global reference for human genetic variation. *Nature* **2015**, *526*, 68–74. [[CrossRef](#)]
51. Bernasconi, A.; Canakoglu, A.; Masseroli, M.; Ceri, S. META-BASE: A Novel Architecture for Large-Scale Genomic Metadata Integration. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **2020**. [[CrossRef](#)]
52. GMQL. Available online: <http://genomic.deib.polimi.it/gmql-rest/> (accessed on 31 May 2020).
53. Nanni, L.; Pinoli, P.; Canakoglu, A.; Ceri, S. PyGMQL: Scalable data extraction and analysis for heterogeneous genomic datasets. *BMC Bioinform.* **2019**, *20*, 560. [[CrossRef](#)]
54. Sherry, S.T.; Ward, M.H.; Kholodov, M.; Baker, J.; Phan, L.; Smigielski, E.M.; Sirotkin, K. dbSNP: The NCBI database of genetic variation. *Nucleic Acids Res.* **2001**, *29*, 308–311. [[CrossRef](#)]



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).