Filegroups Table

The filegroups table holds a recursive structure that contains points to the entries in the files table. Similar in some ways to the directory structure on your computer, each file in in a filegroup, and each filegroup may be inside another filegroup or may attach directly to a project.

Definition:

```
      Column        |       Type        | Collation | Nullable |                      Default
---------------------+-------------------+-----------+----------
+---------------------------------------------------
 filegroup_id        | integer           |           | not null | nextval('filegroups_filegroup_id_seq'::regclass)
 project_code        | character varying |           |          |
 groupname           | character varying |           | not null |
 parent_filegroup_id | integer           |           |          |
 datasize            | bigint            |           |          |
 type                | character varying |           |          |
Indexes:
    "filegroup_id" PRIMARY KEY, btree (filegroup_id)
Check constraints:
    "check_one_exists" CHECK ((
CASE
    WHEN project_code IS NOT NULL THEN 1
    ELSE 0
END +
CASE
    WHEN parent_filegroup_id IS NOT NULL THEN 1
    ELSE 0
END) = 1)
Foreign-key constraints:
    "filegroups_filegroups_fk" FOREIGN KEY (parent_filegroup_id) REFERENCES filegroups(filegroup_id) ON UPDATE
CASCADE ON DELETE CASCADE
    "projects_filegroups_fk" FOREIGN KEY (project_code) REFERENCES projects(project_code) ON UPDATE CASCADE ON
DELETE CASCADE
Referenced by:
    TABLE "execution_blocks" CONSTRAINT "filegroups_execution_blocks_fk" FOREIGN KEY (filegroup_id) REFERENCES
filegroups(filegroup_id) ON UPDATE CASCADE ON DELETE CASCADE
    TABLE "filegroups" CONSTRAINT "filegroups_filegroups_fk" FOREIGN KEY (parent_filegroup_id) REFERENCES
filegroups(filegroup_id) ON UPDATE CASCADE ON DELETE CASCADE
    TABLE "files" CONSTRAINT "filegroups_files_fk" FOREIGN KEY (filegroup) REFERENCES filegroups(filegroup_id) ON
UPDATE CASCADE ON DELETE CASCADE
    TABLE "scans" CONSTRAINT "filegroups_scans_fk" FOREIGN KEY (filegroup_id) REFERENCES filegroups(filegroup_id)
ON UPDATE CASCADE ON DELETE CASCADE
```

Columns:

filegroup_id: an auto-generated id to uniquely identify the filegroup.

project_code: the alphanumeric code for the project this filegroup is attached to, if any. either this or parent_filegroup_id must have a value.

groupname: a string that can be displayed to aid in identifying the filegroup. not used for any internal purpose; this is solely for the convenience of the user.

parent_filegroup_id: the filegroup_id or the parent filegroup, if any. either this or project_code must have a value. This is where the recursiveness of the structure comes in; the parent filegroup may have a parent filegroup, etc.

datasize: the total size of all files in the filegroup, when it exists. this was not part of the original design, and for the most part only VLA data ingested by the new archive will have a value here.

type: a convenience for the ingestion and retrieval of VLA data; this helps the data fetcher identify which groups to download.