This document details the steps necessary to load the PHG maize pg_dump files to a postgres instance.   The dump files for this version were created from a postgreSQL 14 instance.

Creating and Loading the Database

Background on the dumps:

For smaller databases, dumping the databases via the simple "pg_dump" command indicating user, host and password is sufficient. Once created, these dumps may be loaded to a database using psql or pg_retore commands.

The PHG maize database is very large which creates problems when dumping to a single file. Because of this, the database was dumped in 10G sections which were then gzipped into individual files.   They were created via the following command:

pg_dump public_maize_2_1 | split -b 10G --filter='gzip > 20230405_public_maize2_1_$FILE.gz

This created a set of 26 files with generated names of:

20230405_public_maize2_1_xaa.gz
20230405_public_maize2_1_xab.gz
20230405_public_maize2_1_xac.gz
…
20230405_public_maize2_1_xaz.gz

To upload these files to a database named public_maize_2_1, the database must first be created in postgres via the "CREATE DATABASE <dbname>" command.

Then, from the command line within the directory which contains pg_dump files, the following command is run:

zcat *.gz | psql public_maize_2_1

Your database should now contain the data from the pg_dumps files.

One thing to note: The PHG maize databases were created and loaded by user lcj34, with some tables created by user phgdev, and user phg also known at pg_dump time. These roles need to be created in the postgres instance prior to loading the pg_dump files, or errors will appear on the screen. The password for each of these 3 users does not matter, but the users (ie roles) must exist in the postgresql db.

Commands to Create the New DB from pg_dump Files:

To create the database and initial roles, run the following steps:

1. Log into your postgres instance using the appropriate host, port and user variables. You must login with a user that has permissions to create a database, add roles and load tables.
   a. **psql -h <host> -p <port> -U <user >**

2. Create the roles "lcj34", "phgdev" and "phg". These 3 users must be created because the pg_dump files that will be loaded have these users defined with permissions on the PHG tables. You will see errors if those roles are not defined. The passwords you provide can be anything as long as the roles are created prior to loading the pg_dump files.
   a. **CREATE user lcj34 with password 'someSecurePwd';**
   b. **CREATE user phgdev with password 'someSecurePwd';**
   c. **CREATE user phg with password 'someSecurePwd';**

3. If not already existing, create the role you want to be a superuser. This role should be the person who is going to do the loading of the db from the pg_dump files. The role is created as above, but grant all permissions:
   a. **GRANT postgres to <user>;**

4. Create any other roles you would like to see in the db, e.g., read-only roles, or roles for specific groups or people.
   a. **CREATE user <username> with password 'someSecurePwd';**

5. (Optional): Log out of postgres, then log back in to postgres as the user you would like to have own the database. If you logout now, and back in as lcj34, then lcj34 owns the db. If you are already logged in as the user of choice, you can skip this step.

6. Create the db you want to load. The pg_dump files do not contain "CREATE DATABASE". This must be done before you load. **NOTE**: because we did not declare a schema, but rather just the database, the db schema is in "PUBLIC". Run the following command. Change the database name if you wish it to be something different:
   a. **CREATE DATABASE public_maize_2_1;**

7. Logout of the DB. "cd" to the folder that has the db dumps you want to load and run the command below (and as shown in the "Background on the dumps" section) to load your files. NOTE: change the port to be the appropriate port for your postgres instance. You can continue to use -U postgres or set the user to any role that has loading permissions. If you created another superuser above and you are that user, you can login with that id.:

zcat *.gz | psql -h \<host\> -p \<port\> -U \<user\> public_maize_2_1

8. Connect to the postgres instance and verify your database tables are loaded.

9. Grant permissions on this database to the other roles created. Below, user "phgdev' gets most permissions, user "phg" gets read only permissions.
    a. GRANT SELECT, INSERT, UPDATE, DELETE on ALL TABLES in SCHEMA public to phgdev;
    b. GRANT SELECT on all tables in schema public to phg;

10. Create any additional users you would like, setting appropriate permissions.

11. If you have questions, please contact Lynn Johnson at lcj34@cornell.edu.