GEOS on HPC

What is GEOS?

GEOS is a C/C++ library for computational geometry with a focus on algorithms used in geographic information systems (GIS) software. It implements the OGC *Simple Features* geometry model and provides all the spatial functions in that standard as well as many others. GEOS is a core dependency of PostGIS, QGIS, GDAL, and Shapely.

Links:

Official Website

Documentation

Versions Available:

The following versions are available on the cluster:

- GEOS-v3.3.3
- GEOS-v3.10.1

How to load GEOS?

To load GEOS, use the following commands:

```
#Load the GEOS module
module load geos/3.10.1-gcc540
```

To verify if the module and dependencies are loaded correctly, use the following command

```
#Show all the modules loaded
module list
```

This should list all the GEOS and dependencies that are loaded. In this case it is geos and GCC 5.4.0.

How to use GEOS?

For this demonstration, I am going to use C API of the program. This piece of code is excerpted from the documentation,

```
/* geos_hello_world.c */

#include <stdio.h>  /* for printf */
#include <stdarg.h> /* for va_list */

/* Only the CAPI header is required */
#include <geos_c.h>

/*
* GEOS requires two message handlers to return
* error and notice message to the calling program.
*
*   typedef void(* GEOSMessageHandler) (const char *fmt,...)
*
* Here we stub out an example that just prints the
* messages to stdout.
*/
static void
geos_msg_handler(const char* fmt, ...)
{
    va_list ap;
    va_start(ap, fmt);
    vprintf (fmt, ap);
    va_end(ap);
}

int main()
{
    /* Send notice and error messages to the terminal */
```

```
    initGEOS(geos_msg_handler, geos_msg_handler);

    /* Read WKT into geometry object */
    GEOSWKTReader* reader = GEOSWKTReader_create();
    GEOSGeometry* geom_a = GEOSWKTReader_read(reader, "POINT(1 1)");

    /* Convert result to WKT */
    GEOSWKTWriter* writer = GEOSWKTWriter_create();
    char* wkt = GEOSWKTWriter_write(writer, geom_a);
    printf("Geometry: %s\n", wkt);

    /* Clean up allocated objects */
    GEOSWKTReader_destroy(reader);
    GEOSWKTWriter_destroy(writer);
    GEOSGeom_destroy(geom_a);
    GEOSFree(wkt);

    /* Clean up the global context */
    finishGEOS();
    return 0;
}
```

Create a test.c file and copy the code. Use the following code to compile the code,

```
#Compile and link geos library
gcc test.c -o test.out -l geos_c
```

Execute the executable, the output should be

```
#Output on stdout
Geometry: POINT (1.0000000000000000 1.0000000000000000)
```

This was just a simple demonstration of geos API. Follow the official documentation on website for more.

Where to find help?

If you are stuck on some part or need help at any point, please contact OIT at the following address.

https://ua-app01.ua.edu/researchComputingPortal/public/oitHelp