

SFML on HPC

What is SFML?

SFML (Simple and Fast Multimedia Library) is a modern, cross-platform multimedia library designed for game development, interactive applications, and other forms of multimedia content creation. It is written in C++ and provides a simple and intuitive interface for accessing various multimedia components, including graphics, audio, and networking.

SFML offers a wide range of features for creating engaging and interactive applications, including:

- **Graphics:** SFML provides support for 2D graphics, including shapes, sprites, and text. It supports a variety of image formats and provides advanced rendering techniques, such as blending and anti-aliasing, to create smooth and visually appealing graphics.
- **Audio:** SFML provides support for playing and streaming sounds and music. It supports a variety of audio formats and provides advanced audio processing capabilities, such as volume control and audio mixing.
- **Networking:** SFML provides support for communication over the network using sockets. This allows developers to create multiplayer games, online applications, and other network-based services.
- **Window management:** SFML provides support for creating and managing windows, including window creation, handling user inputs, and managing the display and rendering of graphics.
- **System:** SFML provides a range of utility classes for accessing system information, such as time, files, and directories, which are useful for a variety of applications.

Overall, SFML is a flexible and robust multimedia library that offers a simple and intuitive interface for creating engaging and interactive applications. It is widely used by game developers and multimedia enthusiasts, and provides a comprehensive suite of features for creating high-quality multimedia content.

Links:

[Official Website](#)

[Tutorial](#)

Versions Available:

The following versions are available on the cluster:

- sfml 2.5.1

How to load SFML?

To load SFML, use the following commands:

```
#Load the SFML module  
module load sfml/2.5.1
```

To verify if the module is loaded correctly, use the following command,

```
# List all the module loaded in the environment  
module list
```

In a fresh environment, this loads SFML and Gcc module.

How to use SFML?

Here are some of the ways in which SFML can be used in HPC:

1. Visualization and Data Analysis: SFML can be used to create visual representations of scientific simulations or other forms of data analysis, making it easier to understand and interpret the results.
2. Multimedia Applications: SFML can be used to create multimedia applications that run on HPC systems, such as games or interactive simulations.
3. User Interfaces: SFML can be used to create user interfaces for HPC applications, making it easier for users to interact with and control the application.
4. Performance Optimization: By utilizing the parallel computing resources available in HPC systems, SFML can be used to create highly optimized and efficient multimedia applications.

Here a sample code to demonstrate the usage of the library,

```
#include <SFML/Graphics.hpp>
#include <cmath>
#include <iostream>

int main()
{
    sf::RenderWindow window(sf::VideoMode(800, 600), "SFML Example");
    sf::CircleShape shape(100.f);
    shape.setFillColor(sf::Color::Green);
    shape.setOrigin(100.f, 100.f);
    sf::Clock clock;

    while (window.isOpen())
    {
        sf::Event event;
        while (window.pollEvent(event))
        {
            if (event.type == sf::Event::Closed)
                window.close();
        }

        float elapsedTime = clock.getElapsedTime().asSeconds();
        shape.setPosition(400 + 200 * std::cos(elapsedTime), 300 + 200
* std::sin(elapsedTime));

        window.clear();
        window.draw(shape);
        window.display();
    }
}
```

```
    return 0;
}
```

To compile and execute use the following slurm script,

```
#!/bin/bash

# Request one node with 1 task and 1 CPU
#SBATCH --nodes=1
#SBATCH --ntasks=1
#SBATCH --cpus-per-task=1
#SBATCH --p main
#SBATCH --qos main
# Set a walltime of 1 hour for the job
#SBATCH --time=1:00:00
# Specify a name for the job
#SBATCH --job-name=SFML_Job
# Set the output file for the job log
#SBATCH --output=SFML_Job.out

# Load the SFML module
module load sfml/2.5.1

# Run the SFML application
g++ test.cpp -o sfml_example -lsfml-graphics -lsfml-window -lsfml-
system

./sfml_application
```

Where to find help?

If you are confused or need help at any point, please contact OIT at the following address.

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

