

# Main

```
int main()
{
    settings.load("my_config_file.cfg")

    std::string const txt_filename = settings.get_value("txt_filename");
    std::string const output_path  = settings.get_value("output_path", "" );

    util::LogInfo::on()           = util::to_value<bool>(settings.get_value("logging","true"));
    util::LogInfo::console()      = util::to_value<bool>(settings.get_value("console","true"));
    util::LogInfo::filename()     = output_path
                                + "/"
                                + settings.get_value("log_file","log.txt");

    logging << "### " << util::timestamp() << newline;

    parameters = grit::make_parameters_from_config_file(settings);

    assert(grit::is_valid(parameters));

    grit::init_engine_with_mesh_file(
                                util::get_data_file_path(txt_filename)
                                , parameters,engine
                                );

    for (unsigned int i = 0; i < max_steps; ++i)
    {
        write_svg_files(output_path, i);

        do_simulation_step();

        engine.update(parameters);
    }

    logging << "Done" << newline;

    return 0;
}
```

# Write SVG files

```
void write_svg_files(  
    std::string const & output_path  
    , unsigned int const & frame_number  
    )  
{  
    std::string const filename = output_path  
        + util::generate_filename(  
            "/my_own_file_name"  
            , frame_number  
            , "svg"  
            );  
  
    glue::svg_draw(filename, engine, parameters);  
}
```