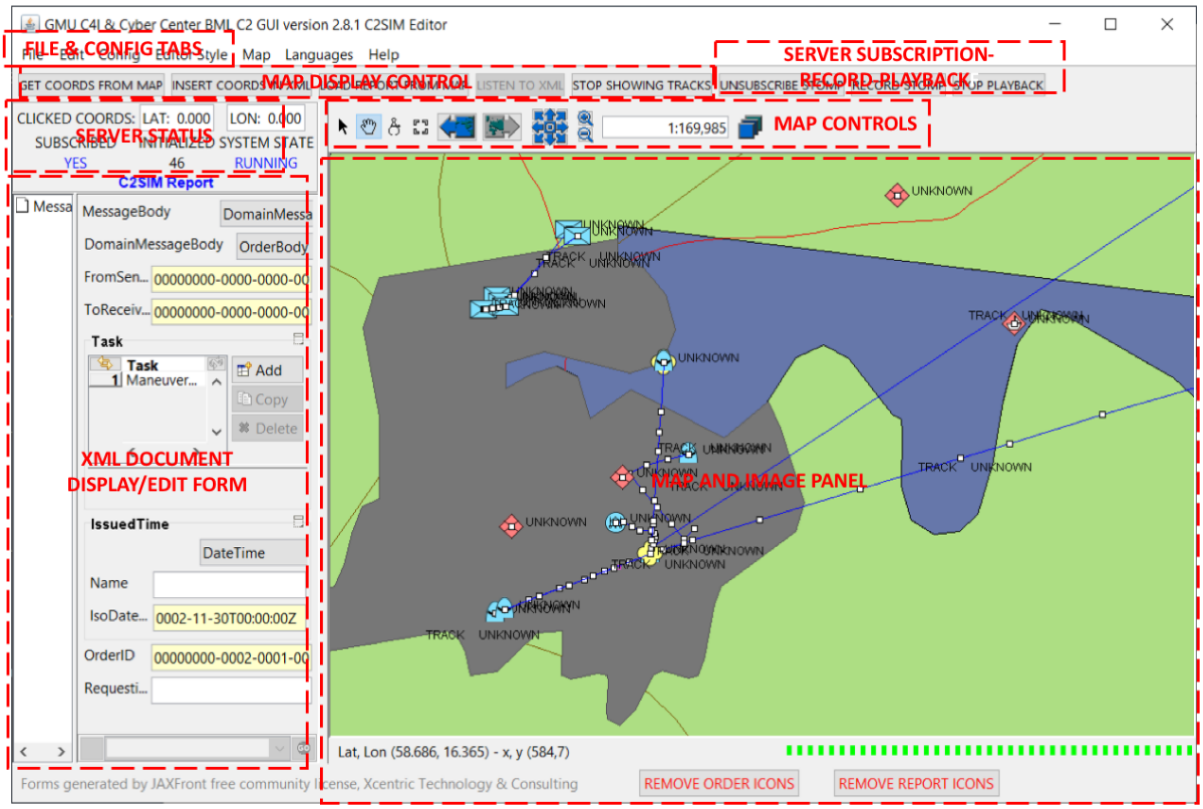
USER'S GUIDE
C2SIM GUI Version 2.10.4

The BML C2 GUI editor was developed in 2009-2010 by Lt Col Mohamamad Ababneh, Jordanian Air Force (now retired), as graduate student work during his PhD program at George Mason University. The GUI is written in Java for portability. It went through a major revision to version 2.0 in 2013 to include the STOMP and REST protocols for use with the WISE-SBML server in NATO MSG-085 and was called BMLC2GUI. Recently it has been upgraded to version 2.9, including the draft SISO Command and Control-Simulation (C2SIM) standard version 1.0.0 modified for NATO MSG-145 and balloted by SISO, and renamed C2SIMGUI; this makes it compatible with C2SIM Reference Implementation Server version 4.8.0.0 and later. The GUI download includes all code needed to build in Apace NetBeans. This document supersedes the original User's Guide by Dr. Ababneh; however, the original will continue to be included with the GUI open source for continuity since it contains information not included in this document.

Purpose of the C2SIM GUI

The GUI is open source software patterned after the C2LG GUI, similar software developed by Fraunhofer-FKIE that is not available open source. Like the FKIE GUI, C2SIMGUI is intended to be a tool that will interact with a C2SIM server by creating and/or editing XML Order and Report files, pushing such files into the server, subscribing to receive such files, and displaying their tactical graphics (unit icons and related graphical control measures) on a map or image.

Major Features of the GUI



The GUI is shown above with labels on its features. The features are:

- File and Config tabs: these select the File and Config dropdown menus; File provides different tabs for each message format because the XML must be associated with a schema when it is loaded

- Load Coords (Coordinates), Insert Coords in XML, and Reports: these buttons let you capture the coordinates of a point on the map, which can then be copied or inserted in the form below. You can also load into the form area the contents of a report represented by an icon on the map, by clicking on the icon (if this doesn't work, the icon represents an initial position or order not a report).

- Server Status: shows subscription status, initializations pending, and server running/paused/stopped/initializing/uninitalized status. If you change the C2SIM state from another application the toolbar buttons enable/disable do not update with the status message from the server.

- Map Display Control: turns on/off listening to STOMP for reports; turns on/off display of tracks

- Server Subscribe Control: subscribes to/unsubscribes from server

- Record and Playback of the incoming STOMP XML file stream can be started from the GUI.

    - Recording is in the same logfile format as that made by the server.

    - Recording: requires the GUI to be subscribed to server STOMP, because it captures STOMP from the server.

    - Playback: when the GUI is connected to the server, Playback is sent to the server via REST and then displayed on the map when returned through STOMP; this results in all clients connected to the server receiving the playback. If not connected to server, Playback displays contents of the logfile locally. NOTE: Whether connected or not, only the report format configured in AutoDisplayReports will be displayed locally. This can be ALL, C2SIM, IBML09, or CBML.

    - There is a timescale factor in Config that speeds up play rate (0 plays all log documents as quickly as possible) and a time limit factor in Config that automatically stops the player after a preset time. If time limit is zero the file is played to its end.

    - Playing automatically turns on LISTEN FOR REPORTS.

- XML Document Form: auto-drawn form from JAXFront (under free community license, limited to non-commercial use) that displays and updates Config, Cyber, Initialization, Order and Report data. This can be configured to auto-load on receipt of Orders and Reports coming from the network, via STOMP.

- Map Controls: rosette pans map; displays layers; manipulates map graphics (hand icon enables mouse-drag panning but takes away cursor arrow; to get it back click on arrow). There is a distance tool that can be used to measure distance on the map and a tool that lists contents of layers show in the map. Choose the arrow to select icons on the map and the hand icon to pan with the mouse. There is a layer for Bogaland that adds details.

- Map and Image Panel: displays a map or image file with superimposed layers and graphics; has buffer for layers of 500 Orders and/or Reports; when full it clears itself; when 80% full it warns. NOTE: the icons are the same military standard for Initialization, Orders and Reports. At the bottom of the map you can see the coordinates of where the arrow cursor is pointing and a set of buttons to clear categories of icon (Initial, Order or Report that will let you declutter the map).

REST and STOMP in the GUI

Representational State Transfer (REST) is the new protocol for input to Web Services, replacing the less efficient Simple Object Access Protocol (SOAP). Since version 2.0, the server uses only REST for input, so the GUI uses only REST to provide input to the server.

Stream Oriented Message Protocol (STOMP) provides output of messages distributed by the server. The server has a capability to filter the message streams by Topic; the GUI listens to all topics. It displays on the map icons for all reports in C2SIM, IBML09, and CBML formats. When a new report is posted for a unit, the old one is removed; however, if SHOW TRACKS is turned on, the display shows the track of the unit as a sequence of small white boxes. Tracks also are erased when ERASE MAP GRAPHICS is selected.

File Menu of the GUI

The File dropdown menu controls input/output of Initialization messages, Orders, and Reports from the host computer file system and the XML display/edit panel and the map. When an Order or Report is read from the file system, it is automatically displayed in text on the JAXFront panel. Orders and Reports also are displayed graphically on the map when they are read. Options exist for C2SIM, IBML09 and CBML message formats.

The options allow creation of a new Initialization message, Order or Report; reading of Initialization message, Order, or Report from the file system; pushing the last Initialization message, Order or Report that was read to the server via REST; and saving an Order, Report or Initialization message from the editing form to the file system.

STOMP connection to the server happens automatically on startup if configured in the configuration panel. When connected, if config entry AutoDisplayReports is set to IBML, CBML, C2SIM, or ALL and LISTEN FOR REPORTS is selected, each incoming report of the type configured will be displayed on the map, the same holds for AutoDisplayOrders and AutoDisplayInit. The map auto-centers to the first report received; thereafter it is possible for received reports to lie beyond the part of the map shown – use zoom to see them. There is a button LOAD REPORT FROM MAP in the map control panel that will open a report into the display/editing form, when its icon is clicked on the map. If Config entry AutoDisplayOrders is set to ALL, C2SIM, IBML, or CBML, orders are displayed as they are received from STOMP rather than as they are sent to the server. As a result, both locally entered orders and those from other participants in the C2SIM Coalition are displayed (CAUTION: they will be overwritten by the next XML document to be posted. If you want to keep a copy, use the File menu to save it. Also be aware that if LISTEN TO XML is selected an incoming order or report might overwrite your editing before you can save it.)

The File menu also includes the ability to send control messages to the C2SIM Reference Implementation: SHARE+START, SHARE, INITIALIZE, STOP, START, PAUSE, RESET, STOP+RESET+INITIALIZE. A password is required (see Config Menu below). These controls are important for assembling and distributing C2SIMInitialization messages, which take the place of MSDL under C2SIM. In the space between the File menu tab and the JAXFront panel there are status displays for the server connection and the status of the coalition of C2 and simulation systems supported by the server. A shortcut button to INIT/SHARE/START SERVER or to STOP/RESET SERVER will appear in the MAP CONTROLS panel if the server is connected and in a state to take these steps in combination.

JAXFront Form Use With the GUI

JAXFront provides an open source capability to draw input/output forms based on an XML Schema (XSD) file. The GUI uses this to produce a form automatically for Initialization Messages, Orders and Reports created new or read from the file system. For C2SIM format these can be edited and saved to the file system. There are two subpanels; the left one provides a top-level structure to control what part of the message is shown in the form, while the right one provides the form itself. To reset either of these, click on the small box on the vertical border and drag the box left or right. It may be necessary to click on elements within the form or “New” to display the form that is needed. Also, in some cases it is necessary to click on “…” to display more of the form. After creating or editing a form, it can be saved by selecting “Save JAXFront” under the “File” dropdown.

Config Menu of the GUI

The Config dropdown menu uses JAXFront to display GUI options in a form and allow them to be updated. Use Load to display the options; after editing them use Save to persist the values. The contents of the form are set by files in the C2SIMGUI/Config folder by main module C2SIMGUI.java. The options are described near the top of the main module. Here are some of the most important options:

- Map Scale: at startup
- Center Unit: (1 or 0) center on the map the first unit loaded from file system or subscription
- Start Subscribed (yes or anything else) causes GUI to subscribe to the STOMP server at startup
- CBMLns, IBMLns and C2SIMns: namespace to be used in interpreting messages of these types
- Server Name: IP address or DNS name of the server where the GUI will push orders and subscribe for reports
- AutoDisplayReports: ALL, C2SIM, IBML or CBML will show reports of that type when LISTEN FOR REPORTS is selected (this also applies to playback of logfiles).
- Server Password: required for server control messages.
- Init Map Lat and Lon: Value for center of the map at startup
- Warn On Report Seq: 1 to display a warning when report send time goes "backward" i.e. is less than that shown in previous report
- blueSideName: the C2SIM/MSDL definition of "ForceSide" does not define "blue" and "red", so the C2SIMGUI allows any of the ForceSides to be designated as "blue" by name, with the result that the ForceSide hostile to that one is "red". If blueSideName is left blank, the first ForceSide in the C2SIMInitialziation file is treated as "blue"
- checkForDuplicateID: if this has value "yes" the GUI will catch and drop Orders and Reports that have duplicate unique identifiers (for C2SIM, OrderID and ReportID); these may come from applications that do not assign unique IDs or from the server sending multiple formats of the same message (for example, C2SIM plus CBML)
- askAboutCheckingDuplicateID: if checkforDuplicateID is "yes" then if askAbout… is also "yes" a message will popup when the first duplicate is found, asking whether checking for duplicates should continue

Maps in the GUI

Zoom within the map can be achieved using the mouse wheel. Panning can be done either by using the pan rosette icon (arrowheads at NSEW) or using mouse (select using small hand icon). Layers, if present, can be turned off and on using the layers icon (multiple rectangles stacked). In principle the map can be replaced by .shp ("shape") files and graphics such as TIFF. (See *The Open Map Developer's Guide.*) To employ a particular .shp file, it must be configured in the bmlc2gui.properties file and included in the /data directory.

Initialization

Initialization can come from the server, local load of initialization file, or playback of a logfile that captured initialization. When an initialization file is processed, the initialized units are displayed on the map and tables are built so that Orders and Reports will show unit names on the map. The initialization icons can be removed separately from orders and reports.

- If the GUI is started when server is not yet initialized, the GUI waits for a C2SIM Initialization message; if the server has been initialized the GUI requests a late join, which includes unit positions updated by the server from PositionReports.

- If the GUI is not connected to the server, when an Order or Report is loaded using the file menu the GUI will offer to initialize. If the option is not taken, map icons will not show unit names.
- If the GUI plays back a logfile that contains a server initialization message, the GUI initializes from that.

Schemata in the GUI

The C2SIMGUI/Schema folder has schemata for all of the XML messages with which the GUI is intended to work. The latest of these are in C2SIM, which contains the schema, based on the SISO C2SIM schema 1.0.0, ASX (which contains NATO MSCOE modifications for the experimental Autonomous Systems Extension used in CWIX 2019), and Cyber which contains the schema for an experimental cyber effects interface also used in CWIX 2019. The "flat" versions of schemata are functionally identical to the original produced by the SISO C2SIM Product Development Group but employ a simpler XML structure compatible with JaxFront.

Server Validation

The C2SIMGUI now includes a process that will exercise the C2SIM Reference Implementation Server (also open source from GMU C4I & Cyber Center). This can be used after software changes, to confirm that the server is still functioning as designed. The validation function is selected at the bottom of the File menu. Its operation is documented in the ValidateServer.java class source file and uses files and subfolders in the ValidateServer folder. It includes scripted functions that perform a server restart, but those can be invoked only if the current server password is in the C2SIMGUI configuration described above.