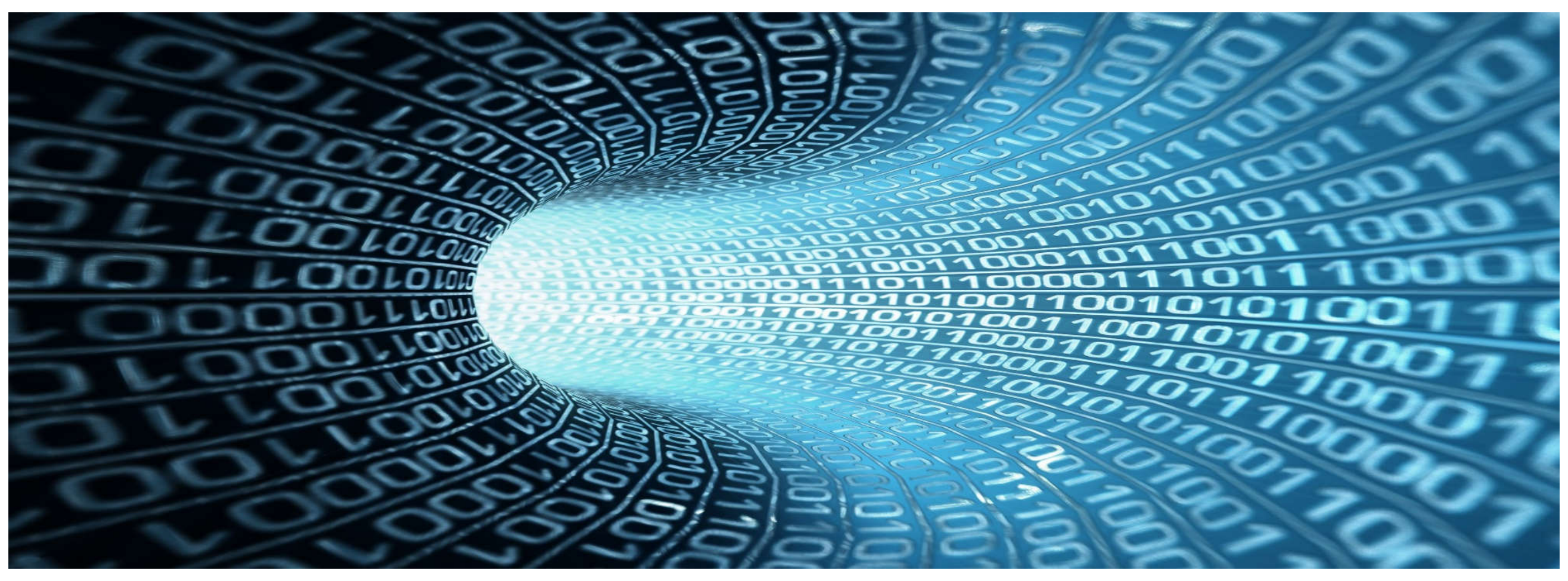
Data Streaming Client Development in SIMULINK for LTB Real-time Application

Clarence L. Jackson II
Center for Ultra-wide-are Resilient Electrical Energy Transmission Networks (CURENT)
Knoxville, TN

Motivation

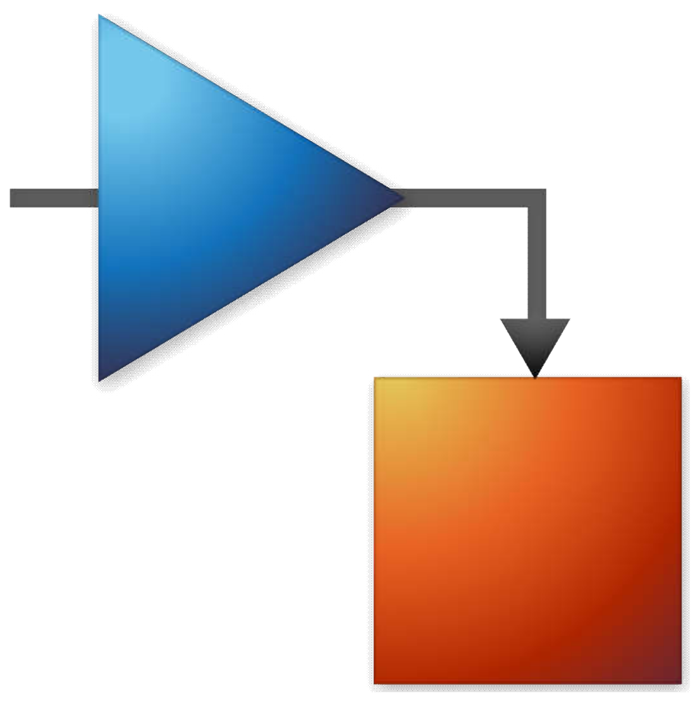
The Large Test Bed simulator is the simulation software used at CURENT to simulate and visualize large areas of the grid. The LTB software consists of a modular architecture in which multiple pieces of software interact and communicate with each other. The DiME server is one mechanism used for inter-process communication within the LTB. Although it is possible to run some LTB simulations on common pc machines, these machines aren't always capable of running complex simulations in real-time. In addition to its current functionality it would be beneficial if the DiME server could be utilized to stream data for additional simulations ran in real time on specialized hardware. This unit will focus on the implementation of a SIMULINK library containing DiME server methods for use in a SIMULINK environment and application on the LTB simulator's real-time simulations.

DiME Methods



The DiME server is initiated by a python script which creates an echo server for the data streaming. Once the server has been created users may stream data between separate instances of Matlab by first creating a DiME object connecting it to the same address and port of the server and then using its methods to stream data content. Those methods include the ability to send Matlab variables, receive Matlab variables, and also designate how many queued Matlab variables to receive upon receiving those variables.

S-Function



SIMULINK, the simulation Modeling software within Matlab, uses function blocks to create numerical data. SIMULINK also provides u the means to create custom user defined function blocks called S-functions in which users may provide functionality via code in C, Fortran, or Matlab. Since DiME methods are callable within Matlab the blocks created for data streaming were developed using solely Matlab code and SIMULINK's level 2 s-function.

In order to implement DiME Methods in a SIMULINK environment the numerical data created by the model would first have to be formatted. This was accomplished by the use of a Matlab structure.

Results

The final product was a SIMULINK library containing three custom blocks that may be used for successful data streaming in a SIMULINK environment; DiME Send, DiME Receive, and DiME Object Initialization.

- The DiME Object Initialization block is required and must be used in combination with either the DiME Send block or the DiME receive block. This block takes as a parameter the address and port of the echo server and creates a DiME object to handle the data streaming.

- The DiME Send block takes as a parameter the amount of input data to be streamed. This block is responsible for the formatting of the numerical simulation data into Matlab structures and also the sending of that data.

- The DiME Receive block takes a numerical input of how much data to be received at a time as well as the amount of data to be received as a parameter.

Acknowledgments: This material is based upon work supported by the National Science Foundation and Department of Energy under Grant No. 1041877. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.