CommTrace (v.2.00) User Guide

Lawrence Williams.
lmw@dcs.ed.ac.uk

July 2, 1998

1 Overview

CommTrace is a HASE based utility allowing the visualisation of communication between two simulation entities. CommTrace visualises the communication process as a time series graph detailing the messages sent across a given communication interface. This is of great advantage when, for example, verifying that communication behaviour complies with a particular specification. The use of a diagram offers an improvement over the more traditional approach of manual trace file examination (which can be both tedious and error-prone).

The graph generated is based upon information stored in a communication trace file[1]. Consequently CommTrace is used in a post-simulation diagnostic role.

2 Starting CommTrace

CommTrace can be invoked either from within the HASE environment or as a stand alone utility.

HASE users simply select 'Generate Comms Fig.' from the Simulation menu.

3 The Console Window

Once started CommTrace presents a console window on screen. (Figure 1). The functions of the console are follows:

a. **View Trace Info:** This control launches the trace file viewer.

b. **Comms Diagram:** Pressing this button causes a communications figure to be rendered for the currently selected entities.

c. **Console Output:** Displays any error messages/system messages.

d. **About:** Displays version information.

e. **Exit:** End the current CommTrace session.

[1] Different to that of a standard HASE generated trace.

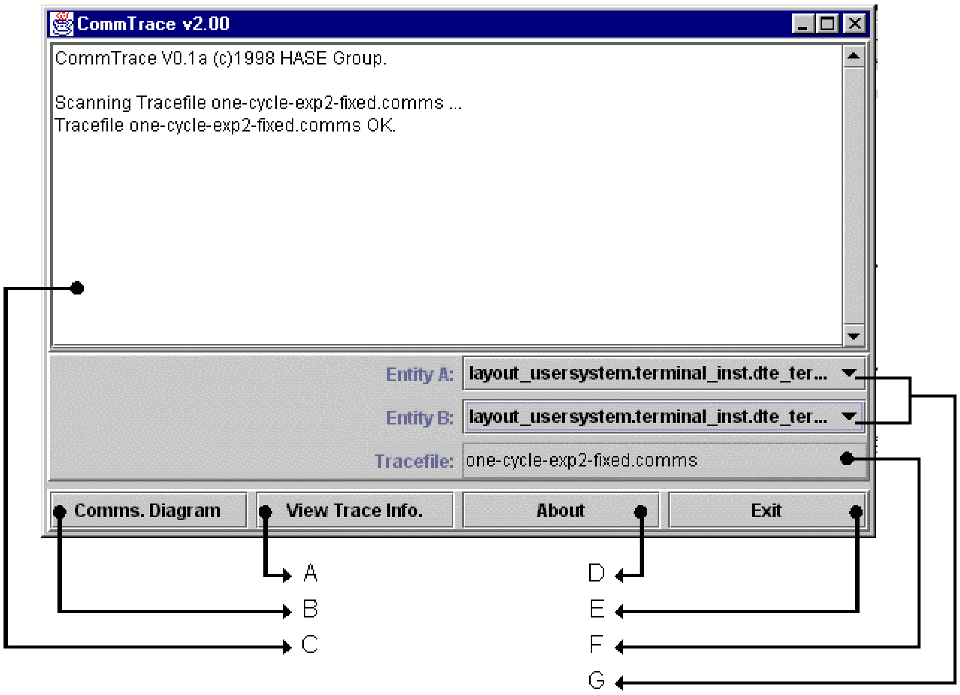

Figure 1: CommTrace Main Console Window.

f. **Tracefile:** This displays the current trace file name. This information is only of real use when CommTrace is run in stand-alone mode (as HASE will use a default name for any trace files it generates).

g. **Entity A/Entity B:** The two list buttons allow the selection of two simulation entities by name. The entities selected will be those referenced in the next communications diagram generated.

4 Trace File Viewer

The trace file viewer (figure 2) allows the user to examine the HASE trace used by CommTrace to generate the communication figures.

a. **No. of sends**: This totals the number of messages found in the current trace file

b. **Total simulation time**: Range of simulation time represented in the trace file.

c. **The trace data**: presented as a table.

d. **OK**: Destroy current trace file viewer.

The trace file used by CommTrace is held in a different format to that of a standard HASE trace. This is because a standard HASE trace file only tracks events being generated at source ports. This is fine for most analysis however we are interested in both source and destination port details here. HASE provides an extended trace format for CommTrace detailing the fields shown in table 1

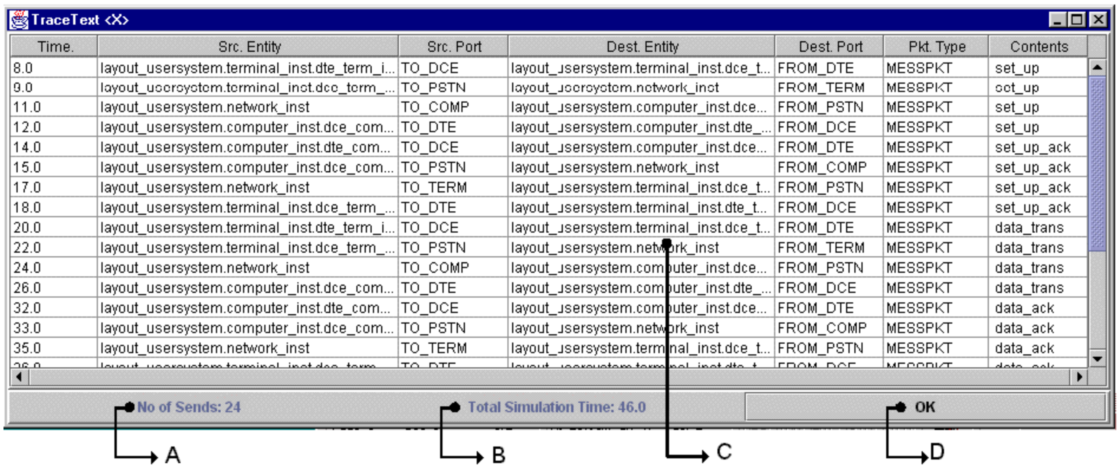

Figure 2: Trace File Viewer.

Field	Notes
Time	The value of the simulation clock when this event was generated.
Src Entity	The entity which generated this event.
Src Port	The name of the source port.
Dest Entity	The entitiy which will receive this event.
Dest Port	The port on which the event will be recieved.
Pkt Type	The type of the message packet (as defined in the model's EDL).
Contents	The packet contents associated with this event.

Table 1: Enhanced Trace File Format - Field Definitions.

5 Communications Diagram

The main function of the CommTrace utility is to provide a meaningful representation of the communication taking place between two simulation entities. This is done via the communication diagram window (figure 3).

The communication diagram is a time-series graph showing the points in simulation time when messages were sent. A message is depicted by an arrow, the direction of which indicates whether an input or output event is taking place. The arrows are labelled with information detailing the message type (in square brackets) and the message contents (display in braces).

Boxes (one blue, one red) which run lengthways down the diagram represent the simulation entities themselves. We note that an entity's messages are drawn in the corresponding entities colour. Simulation time is indexed at the left of the figure and messages are show as arrows running horizontally across the figure.

Events are classified as one of three types (allowing the diagram to be optionally filtered according to event type):

1. **An entity A event**: This type of event is one in which entity A (the leftmost entity in the figure) acts as the message source.

2. **An entity B event**: As above except for entity B.

3. **An external event**: These event are those generated by entities other than the two selected on the console screen. These events are of interest because either entity A or B acts as the destination for these external events.

The controls offered by the communication diagram window are:

a. **OK**: Leave the communication diagram window.

b. **Print**: Make a hard copy of the current diagram.

c. **Refresh**: Force the diagram to be re-drawn.

d. **Copy**: Copy figure to system clipboard[2].

e. **Entity A / B Events**: Switch on/off rendering of events for either of the specified entities.

f. **External Events**: Switch on/off the rendering of external event data.

g. **Communications Figure**: The main display area.

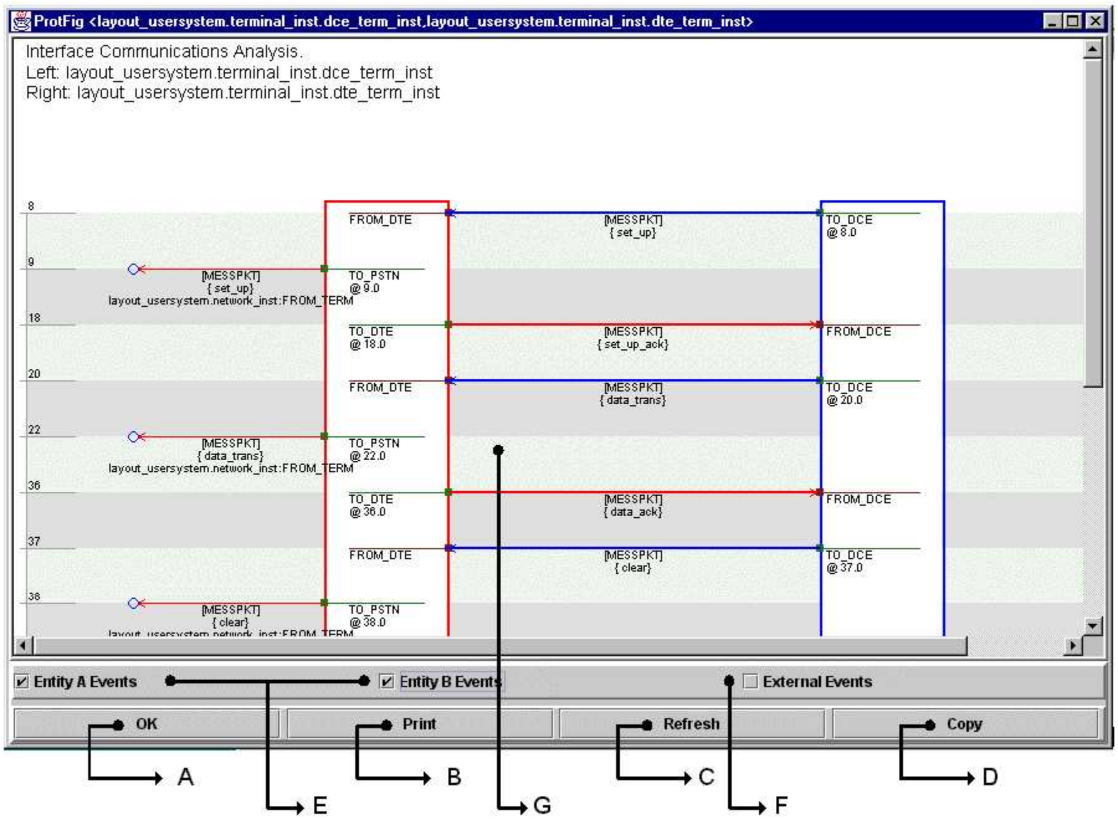

Figure 3: Communications Interface Viewer.

[2]Due to restrictions in the JDK 1.1.6 Solaris copy is not yet implemented.