MINUTES OF NETWORK GRAPHICS GROUP MEETING

Sunday evening, 15 July

   The meeting came to order around 1930, Jim Michener presiding. After
   introductions, an agenda was constructed for the rest of the meeting.

   Elaine Thomas distributed copies of an Alternative Network Graphics
   Protocol for attendees to read overnight prior to discussion.

   Because some individuals were absent who had definitely indicated
   that they were coming Monday morning, the meeting was adjourned at
   2030 after deciding to meet at 0930 the next morning.

Monday Morning/Afternoon, 16 July

   The meeting was called to order at 0930

   Jim Michener distributed an outline of a paper describing desirable
   facilities for the use of two dimensional input devices with a
   hierarchically structured display program.

   Ken Victor distributed copies of RFC 553: A Proposed Network
   Text/Graphics Protocol. (LJOURNAL,17810,)

   Ken Pogran described the history of the NGG and how the "levels"
   approach of RFC 493 came about. In particular, the "level 0"
   protocol was an attempt to define something to experiment with, but
   with the thought that it should be possible to imbed "level 0"
   meaningfully in any later protocol.

   Reports of Network Graphics Experiences

      Jon Jervert described the installation at CAD/CAM (Fort Monmouth).
      They have a spectrum of display terminals and have tried several
      via a Telnet connection to MIT-DMCG. They experienced
      unacceptable slowness with a 300 Baud bandwidth.

      Austin Henderson described an Air Traffic Control experiment in
      which the simulator receives codes describing changes in state and
      generates descriptions of the air space (region) being controlled

and aircraft position and velocity. These descriptions are highly
encoded--they are not pictures in any general sense. The rate at
which the simulation proceeded was adequate.

Jim Michener described the results of an experiment in which the
E&S LDS-1 at MIT-DMCG was used to generate stylus inking input for
a character recognition program at SDC. The experiment was
plagued with difficulties including bugs in SDC's NCP and
scheduling of experimental/debugging sessions. When the
experiment was finally terminated (due to planned extensive
hardware modifications at DMCG) a clear understanding had not yet
emerged, but apparently network transmission delays had been
experienced of up to 20 seconds.

Dan Cohen described an Aircraft Flight Simulator which interacts
with a user at the Harvard PDP-1. The simulation takes place on a
PDP-10. Network traffic is approximately 200 bits from the PDP-1
to the PDP-10 and several thousand bits in the opposite direction.
It has been found that at least 5 updates are required per second
to give the "pilot" an adequate feeling of control. The Harvard
PDP-10 and one at BBN have been used, the latter at 6 AM to avoid
loading problems.

John Pickens described UCSB's status regarding output in level 0
Network Graphics Protocol (NGP-0).

Steve Bunch reported that he has an Imlac monitor which accepts
NGP-0 directly. Programs have been developed at CCN (using
subroutine packages modeled after plotter packages) which build
files containing pictures in NGP-0. Other programs output the
pictures either to a Gould plotter or a storage display (in device
specific code) or to an Imlac (in NGP-0 form).

Steve Holmgren briefly described a Fancy Arpa Network Graphics
System (FANGS) under development at UCSD.

Discussion of Modifications in the Graphics Protocol

David Egli reported that he and Jim Foley (of Univ. of North
Carolina) thought that the graphics protocol should have the
ability to replace items, and that 3 dimensional data should be
allowable. Jim Foley also thinks that a subpicture call should be
able to specify a rate of rotation, scaling, and translation, in
addition to initial values for these.

An extended coffee break followed to allow perusal of the
documents distributed.

Elaine Thomas summarized her protocol proposal for a
hierarchically structured, editable display file.

Discussion related to the levels approach of RFC 493 concluded
that levels were inappropriate; we would henceforth think in terms
of negotiable options.

Ken Victor stressed that NLS was particularly desirous of being
able to make use of the graphics protocol; that was the reason for
their developing RFC 553.

Ken Pogran observed that a structures display system as is being
proposed is more a distributed graphics system than a protocol,
and that he thought this a good idea. General consensus agreed
with him.

Jim Michener described proposals for input. He emphasized the
necessity of transmitting position information in figure
coordinates as opposed to screen coordinates or top level figure
coordinates.

Bob Sproul described two different ways in which a graphics
application in a serving host can communicate to a using host
controlling a display device.

   If the using host has complex enough software or hardware, a
   structured definition of the display may be sent.

      A structured display definition consists of figures (also
      called pictures or groups) which consist of units. A unit
      is either a call to another figure or a collection of one or
      more text or graphic commands. (Other special purpose units
      may exist, also.) Figures and units have names and may be
      created, replaced and deleted (and other things).

A simpler scheme for the using host is that transformed segmented
display information be sent across the network.

Segments have names and can be individually created, replaced and
deleted.

Either the application works directly in terms of segments, or it
works in terms of a structures display definition and software at
the serving host has the responsibility of evaluating the
transformations and the sub-figure calls.

It seems likely that such transformation software might have to
exist at the serving host anyway if that host has any graphics
terminals of small to moderate capability.

It was agreed to restrict our attention to the simpler
"transformed-segmented" scheme, and delay consideration of the
"hierarchically structured" scheme until another meeting.

It seemed to the meeting that a significant number of
applications would need nothing more powerful than a segmented
scheme.

One desirable mechanism is an "end batch of updates" command. It
can help optimize the use of a storage terminal and it can let a
user program causes fixes to occur on a refresh tube all at once.

After lunch, Ira Cotton pointed out that it would be easy enough to
allow NGP-0 to be upward compatible with a segmented, transformed
scheme. Bob Sproul agreed and said that that was a good argument for
sending only device independent data on the net. (This idea was
modified in discussion on Tuesday.)

Ken Victor discussed TTY units, a mechanism for displaying characters
which are "unescorted" i.e., are not part of a graphics "text"
command. In particular they are for spontaneous messages from the
operating system (like "out of funds" or "going down in 5 min").
General discussion was undecided on whether TTY units should really
be part of a graphics protocol. (This was later decided
affirmatively.)

It was noted that unescorted characters coming from the serving
host could probably be handled, but that those coming from the
using host might not be.

Discussion of Network Connection for Graphics

A graphics connection may start out with a Telnet connection. We
will request a DO GRAPHICS telnet option.

Multiplexing on the Telnet connection vs using a separate connection
pair.

Dan Cohen stated that his Flight Simulator uses a second pair.

Alex McKenzie pointed out that some hosts have only "read and
block" input commands, not "read and continue". This means we
cannot demand to have separate connection pairs with graphics on
one and telnet-type information on the other.

Jim Hansen called for a show of hands of preferences: NLS was the
only site against using multiple connection. Several sites were
against multiplexing graphics information on the Telnet
connection. Issues included:

It is easier to merge two streams at the user than to split one
into two. The latter requires "smart" programming.

TIP users may lose if multiple connections are required.

It should be possible to do it on one connection.

In summary: two connections are better than one, the number
shall be negotiated over the Telnet connection.

Ira Cotton asked for a discussion of connection initiation other
than via a Telnet connection. It was agreed that we did not know
enough at this time to specify this and that it was a matter for
experimentation.

Someone commented that what we have is a Network Virtual Graphics
Terminal which has a Network Virtual Keyboard and a Network Virtual
Printer (in the Telnet sense) and a Network Virtual Display Unit.
The printer and the display unit may be the same.

Ira Cotton announced that Jim Foley (of Univ. of North Carolina) is
planning to have a workshop on machine independent graphics under the
auspices of SIGGRAPH in Washington D.C. around mid-April (cherry
blossom time).

Discussion of Graphics Input

Dan Cohen summarized the use of input in his flight simulator:
since it comprises only approximately 200 bits in toto, all
switches, knobs, and stylus position are transmitted. This takes
place about five times per second.

Austin Henderson described the input facilities on the LL TX-2.

Attentions are enabled. What information will be desired when
a particular attention occurs is described at the time the
attention is enabled.

When an attention occurs, the system records the desired
information in a queue for the application program.

When the application program is next scheduled it examines the
queue and responds as it sees fit.

It was generally agreed to adopt the TX-2 strategy. Input devices
will not be enabled unless the server does so.

   No restriction is placed on any "lies" the using host wishes to
   make regarding disguising one device as another.

Network connections for input follow the same rules as for output.

What input attentions are implemented at the using host may be
determined by the serving host in response to an inquiry.

Inking will be provided by the using host (but only one inking
input can be specified at a time; no buffering ahead shall be done
by the using host).

Tracking means the feedback of the current two dimensional input
device position to the user.

   This is automatically turned on by Inking, Positioning, and
   Targeting (hitting) attentions.

What data are reported at the time of an attention is specified by
the application at the server when the attention is enabled.

Types of attentions were listed and also what additional optional
information could be specified with each.

Deactivating Inputs was discussed.

   It is possible for the application to explicitly deactivate an
   attention.

   When an attention is enabled it shall be possible to specify when
   it should be deactivated. Three modes were mentioned: Never
   turned off (until the application explicitly does so), turned off
   when it occurs (self-destruct), turned off when any attention
   occurs.

   The need for a synchronization message was agreed upon.

It was agreed that the serving host - using host relationship would
be one of master - slave. Among other things, the using host would
never volunteer input information which the serving host
(application) had not asked for.

It was decided to meet the next morning at 0830

The meeting adjourned about 1830

Monday Evening, 16 July

    About 2030 seven of us met in Ken Victor's room

    Bob Sproul led the meeting and kept track of the various aspects of
    the protocol.

    Protocol topics which had been discussed during the day's meeting
    were covered again. Most aspects were firmed up based on the day's
    discussions. Several topics were identified for discussion in the
    morning.

    Operations on and attributes of segments were defined.

    The server should be able to enquire for various information from the
    using host.

        Whether the using host has all the features implemented (which the
        application needs).

        What input devices the human has at his disposal.

        What sort of terminal is being used, not so as to send device
        specific code to it, but so that the application does not try to
        use some graphics programming technique on a terminal which can
        not handle it (e.g., some sort of dynamics on a storage tube).

    The server may request that the using host report what segments have
    been defined, their status, and what is contained in then. This is
    good for debugging, and also provides a limited facility of building
    a picture then dumping it to some storage medium other than a
    graphics device.

    It was pointed out that the effect of multiple changes in the display
    (replacing, inserting and deleting segments) should occur "all at
    once" when an "end batch of updates" command is received by the using
    host.

        For a refreshed display, this means keeping old and new copies of
        segments until the "batch" command is received.

        This rule may be waived if storage limitations dictate.

There was considerable discussion on input. It was felt to be the
least firm of any aspects of the protocol.

The meeting broke up around 0030?

Tuesday Morning/Afternoon, 17 July

Bob Sproul presented the results of the previous evening's discussion
to the whole meeting.

The features required of a graphics user program under the proposed
protocol were divided into three classes:

Required features included segment manipulation, primitive
graphics output operations, and response to queries from the
server regarding what is implemented at the using host, what input
devices the human has available, etc.

Optional features included TTY units, reporting the contents of a
segment back to the server at his request.

Experimental features included Input.

It was assumed that after some experience, experimental
features would become either required or optional.

A full list of required, optional, and experimental features will
be issued as a supplement to the description of the protocol.

A graphics server program need only implement those features which
applications at that site make use of.

There was some discussion regarding how and when the graphics
protocol should be published.

The protocol is still regarded as experimental, and we wouldn't
want any site to assume otherwise, to their later dismay.

Some worry was expressed about finally presenting this protocol to
the Network Community in a form that would not frighten too many
people.

Ira Cotton advised us to include a glossary.

Bob Sproul will put an initial version (skeleton) of a description
of the graphics protocol for transformed-segmented scheme into NLS
and will invite everybody in the group to edit it (in normal NLS
fashion).

When one does editing normally, one's ident, the date and the
time are associated with each statement one touches. This
information can be seen via the viewspec (capital) K.

There was some discussion of whether Level 0 NGP could be imbedded in
the Transformed-segmented graphics protocol.

One unfortunate part of NGP-0 was that an End-Picture the is not
explicitly required in order to see something. If it were
required, then it could act like an end-batch-of-updates command.

UCSB assumes that NGP-0 works like a storage tube. They append
a new function plot to an existing picture never having sent an
End-Picture operation.

This ability to append in a storage tube fashion struck the
processors of refresh tubes as quite a drawback, because of
implementation difficulties.

It was decided to allow a using site to have NGP-0 compatibility,
but not to require it.

At least the NGP-0 opcodes would not be reused.

Except for the End-Picture problem, and possibly also a coordinate
system problem (coordsys), NGP-0 can be imbedded in the transformed-
segmented protocol with the entire NGP-0 picture corresponding to a
single segment.

The following sites hope to achieve implementations of the
experimental segmented protocol:

UCSB hopes to have a server running for OLS and Signal Analysis
(speech processing).

SRI-ARC hopes to have NLS operate in this protocol.

MIT-DMCG may have some simple serving programs.

Several people plan to implement user programs, at least as far as
the required features go.

(coordsys) A discussion arose concerning what coordinate system
should be used in sending graphics output primitives from the server
to the user.

The following problems were addressed:

What happens if the display segment terminal screen area to be used by the application is not rectangular?

What happens if the basic unit delta X is not the same as the unit delta y? The application might want a 45 degree line to really be at 45 degrees.

Various answers to the first question:

Use the largest square within the rectangle (centered?, adjusted to the left, top, right, or bottom?)

Use the smallest square surrounding the rectangle. (How is the rectangle positioned in the square?)

NGP-0 standard coordinates (-1/2 to +1/2) used and mapped into the whole rectangle.

The user reports left, bottom, right, and top physical coordinates and the server sends coordinates within the range given.

This is compatible with the attitude that the transformed (!) segmented graphics data are sent.

It is also saves the using host (which might be an Imlac) from doing a multiply.

John Pickens observed that if a graphics server for a finicky application transmits characters as strokes, then the application is assured of having the characters positioned in exactly the right place (e.g., for a numeric label on a tic mark on the axis of a graph. If characters are sent as text (not strokes) positioning is not necessarily guaranteed.

Ken Victor and Jim Michener will look into ways of keeping the NGG apprised of progress (in terms of what sites have experimental/operational graphics protocol servers or user programs) using a pointer file in the NIC.

The next NGG meeting is tentatively scheduled for the first Sunday in February 73, at 8PM. It will either be at the NIC or partly there and partly at Xerox PARC.

The meeting was adjourned at 1500.

Appendix: Meeting Participants/ Affiliation/ Online mailing address/
    Attendance (S=Sunday, M=Monday day, E=Monday Evening, T=Tuesday)

    Steve Bunch   ILL-ANTS
        BUNCH@ISI
        SMT

    Dan Cohen   Harvard
        DCOHEN@ISI or COHEN@HARVARD
        SMET

    Ira Cotton   National Bureau of Standards
        NBS-TIP@NIC attention Ira Cotton
        SMET

    John Day   ILL-ANTS
        S

    David Egli   CAD/CAM (Fort Monmouth)
        ECOM@BBN
        SMT

    Jim Hansen   ILL-ANTS
        HANSEN@ISI
        SMT

    Jim Hart   NASA/Ames
        MT

    Austin Henderson   Lincoln Labs
        DAH@TX2 or DAH@BBN
        SMET

    Steve Holmgren   ILL-ANTS
        HOLMGREN@ISI
        MT

    John Jervert   CAD/CAM (Fort Monmouth)
        ECOM@BBN
        SMT

    Alex McKenzie   BBN
        AAM in the journal or MCKENZIE@SRI-ARC
        SMT

    James Michener   MIT-DMCG
        JCM in the journal or JCM@DMCG
        SMET

    John Pickens   UCSB
       JRP in the journal or UCSB@ISI (attn: John Pickens)
       MT

    Ken Progran   MIT-Multics
       Pogran.CompNet at MIT-MULTICS
       SMT

    Bob Sproul XEROX
       SPROUL@MAXC
       MET

    Elaine Thomas   BBN
       THOMAS@BBN
       SMET

    Ken Victor   SRI-ARC
       VICTOR@NIC
       SMET

           [This RFC was put into machine readable form for entry]
              [into the online RFC archives by Via Genie]