PicoVis: A Dynamic Visualisation Tool for Simulating a Bluetooth Communication Environment Enhancing Student Understanding.

Timothy Doyle
Dr. Chris Exton
Department of Informatics and Electronics
University Of Limerick,
Ireland.
timothy.doyle@ul.ie
chris.exton@ul.ie

Keywords: POP-I.C.education technology, POP-II.A.novices, POP-III.D.visualisation

Abstract

Bluetooth (Bluetooth SIG 2003, Haartsen 1998, Johansson, et al. 2002, Morrow 2002) is a recently conceived universal communication standard that allows wireless connectivity among numerous portable electronic devices resulting in the formation of ad-hoc (Volbert 2002) wireless networks. The paper presents PicoVis, a dynamic simulation and visualisation environment modelling the functionality of a Bluetooth Personal Area Network (piconet), with the aim of aiding student comprehension of this relatively new technology. The simulation environment described in this paper is designed for teaching purposes and not for industrial use, unlike some of the currently available network simulation environments. The aim is that by simulating and visualising a wireless network, the cognitive load (Tuovinen 2000) placed on students when studying a new technology like Bluetooth for the first time will be reduced.

Introduction

Ericssons proposed Bluetooth as a wireless communication standard in 1994 to allow laptops establish phone calls through mobile phones. Since then it has emerged as an important standard for short range, low-power wireless communication. Despite some initial challenges Bluetooth is gaining momentum throughout the world and it is expected that the Bluetooth semi conductor market will grow at a rate of 50% annually to reach $1.46 billion by 2007 (TechXNY CiberLibrary 2003). Because of the huge growth potential mentioned above and the fact that Bluetooth would most definitely be a candidate for ubiquitous computing it would be useful to develop electronic learning aids like PicoVis so as to enhance students understanding of this wireless standard. The development of this standard along with others like the 802.11 Wireless LAN protocols has placed us in the midst of a networking revolution. However tools for analysing these wireless protocols along with educational support tools have lagged somewhat behind. Trying to grasp the complexities of a new communication protocol can place an enormous cognitive load on students. The area of communications is a rather difficult area to teach (Dixon, et al. 1997) because of the dynamic and complex nature of activities that take place in an actual working wireless network. Along with this the Bluetooth specification is itself quite a detailed and complicated document thus making difficult reading for a novice in this area. One possible approach to overcome these issues is to provide a dynamic simulation environment, which visualises the underlying system in question, thus resulting in the users (students) forming an intuitive understanding of the structure and behaviour of the system they are working with. Network simulation is a well-established field and it could be argued that there is not a need for another network simulator. However the problem with existing environments is that they are design tools, developed for professional software engineers, addressing software engineering issues such as testing and aiding in the design of physically complex networks. These tools are generally very powerful and complex environments requiring a great deal of time to master. Also the environments assume that users have a relatively good knowledge of the inner workings of the simulated system. As a result these simulation tools may not be of much use to students who are being introduced to the fundamentals of the technology for the first time and cant really deal with the advanced nature of the existing tools.

A number of surveys have been conducted similar to the one mentioned in (Naps, et al. 2003) which show that visualisation technology can have a positive effect on learning and that learning can be greatly accelerated from active involvement in experience-based activities. Simulation is based on the idea of experimentation (Pidd, et al. 1989). Instead of experimenting on the real system, trials are made on a dynamic model. Simulation tools support what is called "learning by doing" (Rose, et al. 1998) where students are encouraged to explore the technology through the manipulation of variables and the execution of user defined scenarios. In reality visual interactive simulation models will give a better insight into the workings of a system than a static report (like the Bluetooth specification) will ever achieve. Visual interactive simulation has been described as a "voyage of discovery" (Pidd, et al. 1989). Such tools could be used in an academic setting to enhance lectures, supplement labs or engage students. On the other hand it is often the case in an academic or industrial environment that simulation tools are the most cost effective means of providing training and analysis capabilities (Rose, et al. 1998). This paper is concerned with the description of a dynamic interactive simulation tool that visualises the activities that occur in an actual working Bluetooth piconet and discusses its value as a pedagogical aid.

Related Work.

Bluetooth is a leading edge technology and as a consequence there is not an abundance of simulation tools freely available. Most of the research in the area of network simulation has concentrated on fixed networks such as Ethernet simulation tools (Ethernet 2003). However there are similarities between the activities of fixed and wireless networks and therefore a number of fixed network simulators were studied while carrying out the research. Most of the freely available wireless simulation environments that do exist are more analytical based, geared towards professionals, addressing professional software engineering problems that occur in the design of physically complex networks like Local Area Networks. Some of these environments are briefly introduced below.

The most popular Bluetooth open source simulator is called the BlueHoc Simulator (IBM developerworks 2003). This tool was developed at the IBM Research Centre in India. It is based on the Ns2 network simulator (The Network Simulator 2003). BlueHoc is a C++ based simulator that offers developers the ability to test Bluetooth wireless applications. Not alone is it a complex environment to use but it requires a lot of time to set up and configure, therefore not really appealing to students who would prefer having an environment which is easier to use and automatically runs from the command prompt or from clicking on an icon.

BlueWare (Tan 2002) is a simulator developed at MIT that implements most aspects of the Bluetooth protocol stack. This simulation tool is based on the BlueHoc simulator, although it involved the re-writing and re-designing of most of the BlueHoc code. However this is a purely analytical tool, designed for users like professional engineers who have a good working knowledge of the technology.

Both these simulation environments can be described as static or inanimate (Rose, et al. 1998) simulators. They generate large trace files that require the use of a network animation tool called Nam (Estrin, et al. 2000) to replay the simulation results.

Nam is a rather powerful visualisation tool that can handle large amounts of information quickly and generate powerful visualisations. However this tool was designed mainly for analysing network performance characteristics and not for student education.

NetSim (Barnett 1992) is a fixed network simulator, created for student use, simulating the activities of a Local Area Network. However its pedagogical aim is to introduce students to network performance and evaluation, not really aiding the student's comprehension of the inner workings of the technology.

As well as the NAM network animation tool, which as mentioned previously is used to visualise large trace files, there is also SeeNet (Becker, et al 1995), which is a collection of visualisation tools for visualising fixed network data using static displays, interactive controls and animations. Again this is not a real time simulation tool as its focus is on visualising network data rather than offering students the ability to explore the inner workings of the system. However in relation to developing educational support tools SeeNet does have interesting features in that it has a high degree of interactivity and it uses some powerful visualisation techniques.

The simulation tool described in this paper has the characteristics of live or real time simulators. These environments maximise user interaction allowing exploration and provide dynamic real time visualisations of the activities taking place (Rose, et al. 1998).

Pedagogical Requirements.

In order for a tool like PicoVis to be successful as a pedagogical aid for students it must satisfy an increasingly large set of requirements (Freisleban, et al. 2000, Storey, et al. 2002). Specifying requirements for a particular tool like PicoVis is important because the software environment will be designed according to the most important needs of students and not what the developer "thinks" the tool should incorporate.

- The tool must be well designed so that the eventual end users will find it easy to use and most importantly find it beneficial to their learning.
- PicoVis should be simple to navigate.
- All the visualisations and operations of the tool should be based on user-adjusted time.
- The tool should not hinder or frustrate users; the tool must have the best balance between abstraction and detail (Exton 2000).
- Importantly the tool should be relevant to the course that is being thought otherwise it offers no advantages to students.
- Users should have the ability to provide input to the tool and manipulate data while the simulation tool is in use.
- All learning environments should be highly interactive offering students the ability to have complete control while the environment is running. An example would be providing users with "replay" functionality as having to continually restart the simulation causes user frustration.
- The system should provide documentation support in order to maximise students understanding of the mapping between the abstractions of the simulation tool and the visualisations of those abstractions.
- A structured view of the system being visualised should be presented so as to offer students the ability to quickly move to specific parts of the system that are of interest to them.

As can be seen from the above requirements, the design of student learning environments is by no means a straightforward process. It is extremely difficult to develop a tool that has the best balance of all the specified requirements as one may over shadow another.

Bluetooth: A brief Technical Overview.

The simulation tool is built around four lower layers of the Bluetooth protocol stack. These layers are as follows: Baseband, Link Manager, Host Controller Interface and logical Link Control and Adaptation Layer (L2CAP). An overview of the Bluetooth protocol architecture is depicted in figure 1. A Bluetooth enabled device will communicate with other devices using this protocol stack.

The Bluetooth Protocol Stack.

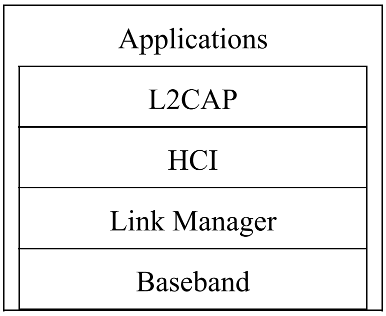


Figure 1: Bluetooth Protocol Stack.

Below is a brief description of the main functionality of each of the four layers of the Bluetooth Protocol Stack, which will also be modelled in the PicoVis visualisation tool, as these are the main topics that a student should really understand if they are to grasp the workings of this technology.

The *Baseband layer* is the physical layer of the stack. It is a state machine that controls the radio layer. Baseband enables the existence of the physical RF links between Bluetooth units. This layer carries out the following piconet functions:

- Provides both circuit and packet switched transport mechanisms.
- Carries out frequency hop selection. Bluetooth uses Frequency Hop Spread Spectrum (FHSS) to avoid interference.
- Device Discovery (Inquiry Procedures) and Connection establishment (Page Procedures). A device uses the Inquiry procedure to discover what other devices are in its range. After the inquiring device has discovered all others within its communication range it may wish to establish a connection, thus forming a piconet.
- It applies a (TDD) Time Division Duplex scheme to the communication links. All communication takes place over time slots that are 625 ms in length.
- Transmission and retrieval of Baseband packets. Information from higher layers is sent over the air using Baseband packets.

The *Link Managers* main priority is piconet management. It is responsible for connection establishment / removal, QoS configuration, Authentication, Security and Power saving.

The *HCI layer* provides a standardized way for higher layer applications to pass commands down to the lower layers of the stack. It eliminates the need for the higher layers to know the detailed functionality of the Bluetooth device.

L2CAP is the protocol with which most higher layer applications interact. This layer provides both connection less and connection orientated services to higher layer protocols along with packet segmentation and reassembly operations, QoS configuration and protocol multiplexing. Services are provided to upper-layer protocols by transmitting data packets over L2CAP channels.

Bluetooth Network Topologies.

Two or more Bluetooth units that share the same channel form a piconet, figure 2.

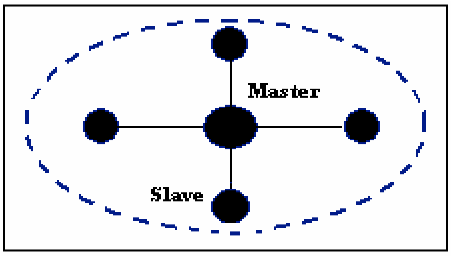


Figure 2: A Bluetooth Piconet.

A piconet can have a maximum of 8 devices active at any one time. The device that initiates the first connection becomes the 'master' while all other devices become 'slaves'. All slaves have their clocks synchronized to the clock of the master. As mentioned earlier the Bluetooth air interface is based on time slots. The master will transmit in even numbered slots while the slave transmits in odd slots. The master controls all communication within a piconet. Communication is organized so that the master polls each slave thus sharing out the bandwidth of the piconet using its intra-piconet scheduling algorithm (Capone, et al. 2001, Yaiz, et al. 2001, Das, et al. 2001). The scheduling algorithms act as the heart of piconet operation by allowing information to be routed from the master device to slaves.

PicoVis: A Discussion.

One of the most difficult yet important tasks to overcome was that of designing the various visualization techniques needed for the simulation environment. It was essential that the visualization techniques used could be easily perceived and interpreted by the user (Storey, et al. 2002). This is important especially if large amounts of complex information has to be visualised. Also a question remained over the level of detail to be used in the visualisations. While it is important to allow for detail, a common problem arises due to tools becoming "saturated" in too much detail. This problem arises because of the developers desire to provide "realistic simulations" (Heidemann, et al. 2000). Thus the goal was to find a level of abstraction that visualises the information in an easy to comprehend way along with insuring that the user would not have to waste too much time learning how to use the tool.

When the simulation tool is run the user is presented with the main simulation window shown in figure 3, which offers a variety of functions. Users have the ability to create devices (nodes) at their own discretion. Newly created devices are placed in the 'Standby' state as defined in the Bluetooth specification (Bluetooth SIG 2003). All these devices operate independently of one another. Users have complete control over their newly created devices. They decide when a device should try and establish a connection, decide which sub state (Page Scan or Inquiry Scan) the device should periodically enter while in the "STANDBY" state and they also have the ability to reconfigure the default variables and timeouts specific to each device. In figure 3 an example scenario is given where the user has created the maximum number of piconets, six. Drawing dashed borders around devices indicates the presence of piconets. The tool allows users to add up to a maximum of nine devices to each piconet, one device being the master while the other eight are slaves. Master nodes are indicated by a purple colour while all slave nodes have a yellow colour as can be seen in figure 3. Also from figure 3 it can be seen that a connection exists in the first piconet between the master device and a slave. Drawing lines between the devices that are in a connected state indicate connections.

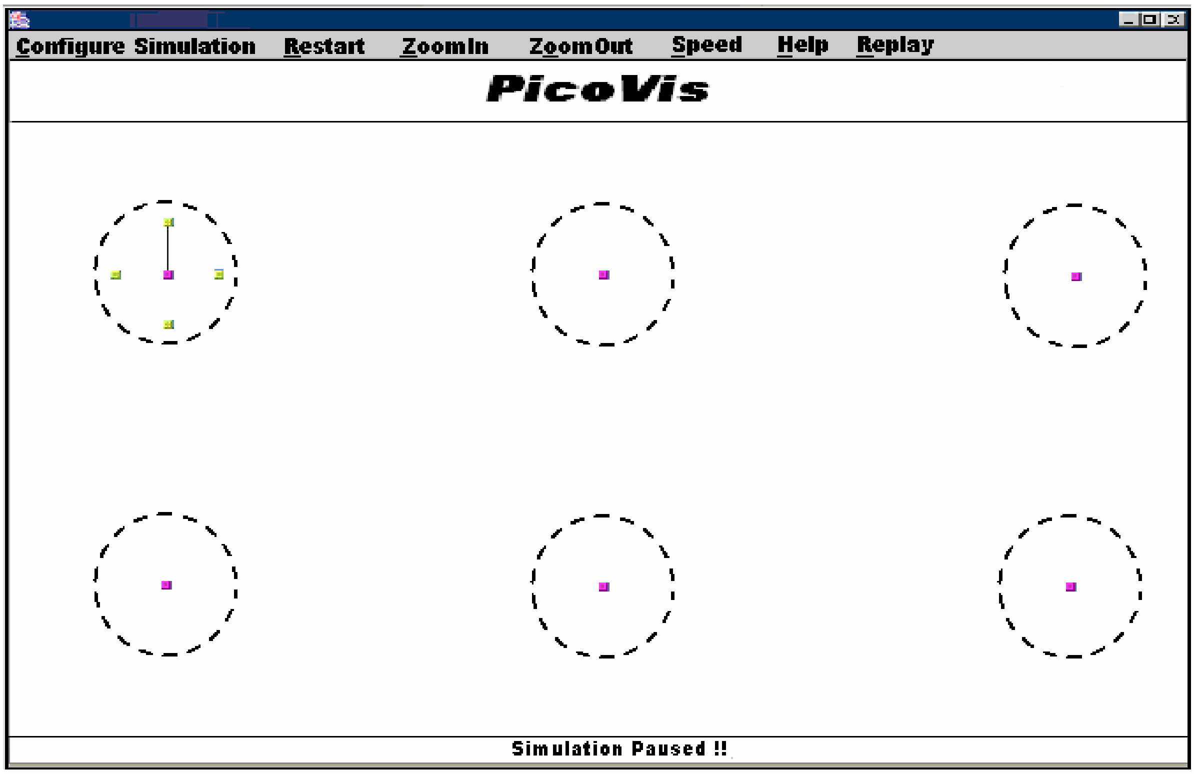
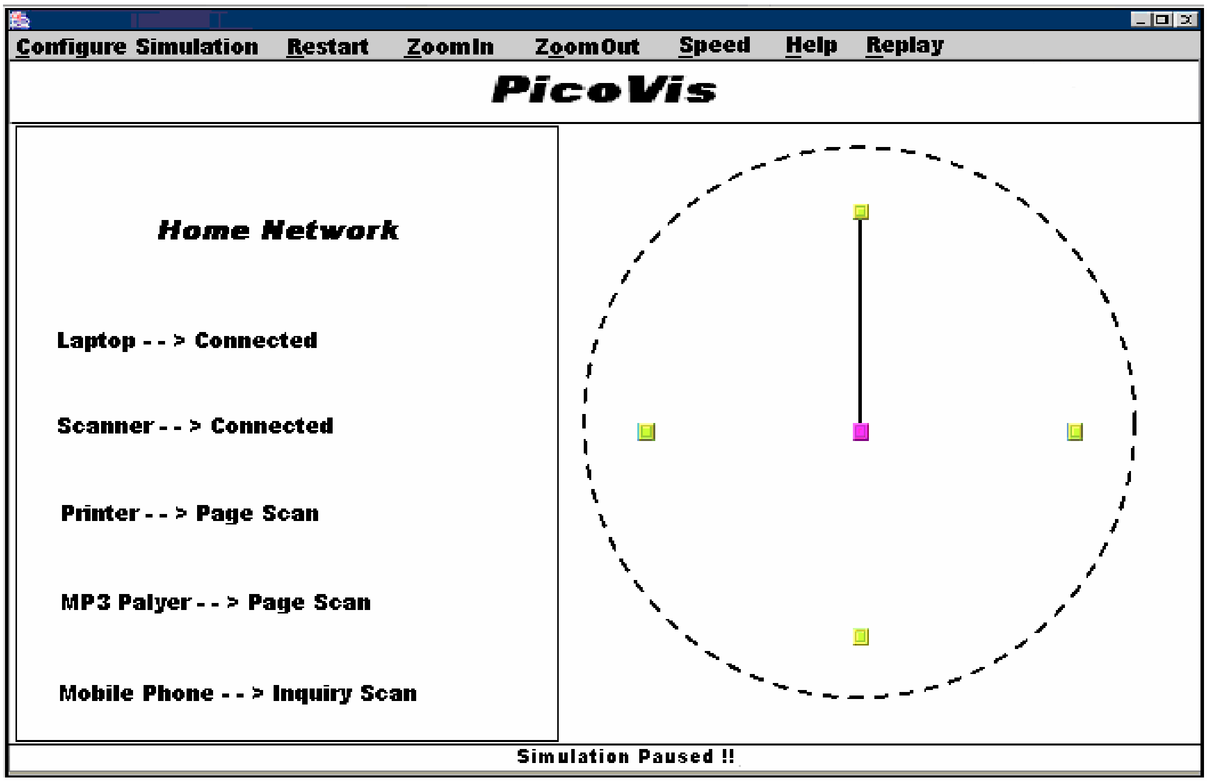


Figure 3, PicoVis Main Window.

The simulation tool incorporates dynamic displays and interactive controls. Users can control the speed of the simulation by setting the speed from the main window. The user can set the speed of the simulation to suit their needs and this can be done at any time while the tool is running. Users can also pause and restart the current simulation. When a node is configured and added to a piconet a trace file will be created and used to store certain activities pertaining to that device during its lifetime. The student may then choose to use this file to replay the activities generated by that device. However all of the trace file does not have to be replayed, the tool will parse the file and pick out the different event sequences that have occurred, then place these in a menu that is used by the student to choose a sequence to replay. The tool also buffers a certain number of events allowing the user to choose a quick replay of previous events associated with a particular device. This is much faster than loading a trace file into the simulation but it will only contain a certain number of events. Both types of replays can be initiated during a running simulation because when the user selects the replay option the current simulation is paused and the tool replays events. When finished replaying events the tool resumes where it left off before it was interrupted to perform a replay. Users can also access help files containing information on how to correctly use the simulation tool.

Monitoring capabilities are presented to the user in the form of dynamic visualisations. However to gain access to these visualisations the user must first select the piconet where the node of interest is located by “zooming in” on that piconet and then selecting the node which is also a “zooming in” operation. An example of selecting a piconet is given in figure 4.



Home Network

Laptop - - > Connected

Scanner - - > Connected

Printer - - > Page Scan

MP3 Palyer - - > Page Scan

Mobile Phone - - > Inquiry Scan

Simulation Paused !!

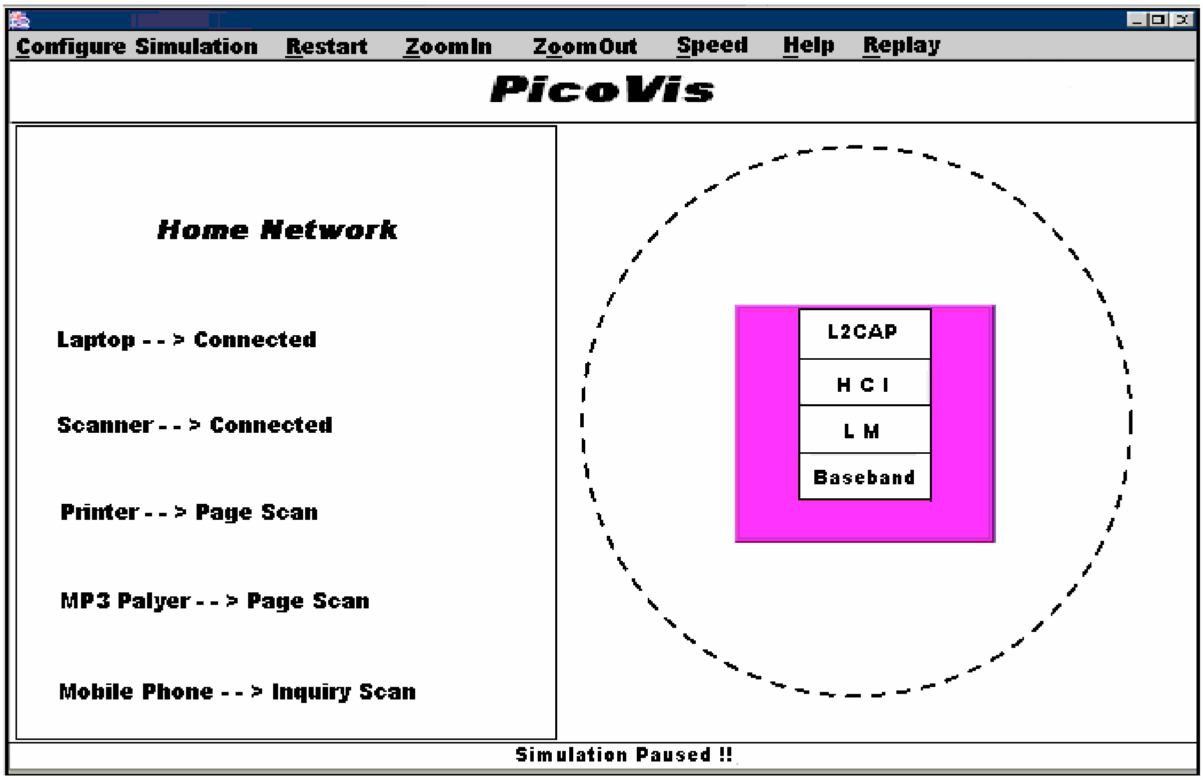
Figure 4: Zooming in on a Piconet.

After a node is selected, (by clicking on it with the mouse) see figure 5, a stack structure of four layers is revealed representing the Bluetooth protocol stack (Bluetooth SIG 2003). However at the moment only the baseband and link manager layers are functional in the simulation tool. By selecting the baseband layer the following functions are available to the user:

- A popup window showing a dynamic visualisation of the activities taking place in the frequency hop selection kernel (Bluetooth SIG 2003) of the device. The selection kernel is made up of a number of components and the student can click on any of the components enabling the viewing of different variables. Also documentation is available for each component so as to insure that the student is able to fully grasp the workings of this complex module.
- The user can view the entire collection of variables and timeouts specific to the baseband layer.
- Building a mental model of how Bluetooth enabled devices use time slots and frequencies to transmit and receive information can be rather changeling especially for novices. Therefore the tool offers students a view of two time slots (a receiver and transmitter timeslot) and shows the frequencies used for listening and transmitting on both slots.
- When describing how connections are established and the message sequences involved, it can be quite difficult to visualise how all this is done from reading textbooks and studying static diagrams. As a result PicoVis provides a popup window called the “Events Viewer”, which brings the whole area of connection set-up and packet transmission to life.
- When the student is studying the “Events Viewer” window, they are able to see actual packets being transmitted from one device to another. As it is important that students understand the structure of these packets and learn how information is represented in the packet fields, they can click on a button from the “Events Viewer” window and be presented with a representation of the packet presently being transmitted or received. The window contains a general baseband packet format containing three entities; by clicking on any one of these entities the user can see the fields and binary data unique to that entity.

- An option is offered to users allowing them to change the substate of the device (i.e. change from Page substate to Inquiry substate (Bluetooth SIG 2003)).
- Finally the baseband layer in the simulation tool offers students the ability to change the variables and timeouts relevant to the current substate the device is operating in, thus resulting in the user having full control over the simulation.

Currently the link manager layer offers only one option, for master devices only it allows the users to initiate the connection establishment process.



Home Network

Laptop - - > Connected

Scanner - - > Connected

Printer - - > Page Scan

MP3 Palyer - - > Page Scan

Mobile Phone - - > Inquiry Scan

Simulation Paused !!

Figure 5: Expanded Master Node.

Conclusion.

In this paper a dynamic interactive simulation tool called PicoVis has been presented. The aim is to try and move away from static or in-animate tools to more dynamic ones with the hope of reducing the amount of cognitive strain that occurs when students are trying to come to grips with a new communication standard like Bluetooth for the first time. The PicoVis tool itself provides students with a highly interactive real time simulation environment containing dynamic visualisations of the activities taking place in the simulation. However it cannot be stated in this paper if PicoVis has been successful in aiding student comprehension of the Bluetooth technology as a usability study has yet to be carried out.

Future work will involve modelling the functionality of the other three lower layers of the Bluetooth stack that were introduced in an earlier section. At the moment only the baseband layer has been simulated. Also the tool could be expanded to allow for multiple interconnected piconets called scatternets (Bluetooth SIG 2003). Finally and most importantly a usability study has to be carried out to determine the effectiveness of the PicoVis tool towards enhancing student understanding of the Bluetooth technology.

References.

Barnett, B.L. (1992) Netsim: a network performance simulator, *Proceedings of the twenty-third SIGCSE technical symposium on Computer science education,* Kansas City, Missouri, United States.

Becker, R. et al (1995) Visualizing network data, *IEEE Transactions on Visualisation and Computer Graphics*, vol.1, no.1, pp. 28-36.

Bluetooth SIG. (2003) *Specification of the Bluetooth System*, version 1.1, Specification Volume 1.

Capone, A. et al. (2001) Efficient polling schemes for Bluetooth picocells. *Proceedings of IEEE International Conference on Communications 2001*, Helsinki.

Das, A. et al. (2001) Enhancing performance of asynchronous data traffic over the Bluetooth wireless ad-hoc network. Proceedings of IEEE INFOCOM 2001.

Dixon, M.W. et al. (1997) Using a network simulation package to teach the client-server model, *Proceedings of ITiCSE 1997*, Uppsala, Sweden.

Estrin, D. et al. (2000) Network visualisation with Nam, the VINT Network animator. *IEEE Computer Magazine*, vol. 33, no. 11, pp. 63-68.

Exton, C. (2000) Concurrency, objects and visualisation, *Australian Computing Education Conference (ACE 2000),* ACM, Melbourne.

Freisleban, B. et al. (2000) The animal algorithm animation tool, *Proceedings of ITiCSE 2000*, Helsinki, Finland.

Haartsen, Jaap. (1998) Bluetooth – the universal radio interface for ad hoc, wireless connectivity, *Ericsson Review*, No. 3, pp. 110 – 117.

Heidemann, J. et al. (2000) Effects of detail in wireless network simulation. *Proceedings of SCS Communication Networks and Distributed Systems Modeling and Simulation Conference 2000*, California.

IBM developerworks: Open Source, (n,d), Retrieved : August 20, 2003, from http://www-124.ibm.com/developerworks/projects/bluehoc

Ethernet: Ethernet Technologies, (n,d), Retrieved : June 2, 2003, from http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/ethernet.htm

Johansson, P. et al. (2002) Bluetooth: an enabler for personal area networking. *Proceedings of OPNETWORK 2002*.

Morrow, Robert (2002) *Bluetooth Operation and Use* 1st Edition, McGraw – Hill, England.

Naps, T. et al. (2003) Exploring the role of visualization and engagement in computer science education, *SIGCSE Bulletin*, Vol. 35, No. 2, p. 131-152.

Pidd, M. et al. (1989) *Computer Modelling for Discrete Simulation* John Wiley & Sons, New York.

Rose, A. et al. (1998) An application framework for creating simulation-based learning environments, HCIL Technical Report, No. 98-07.

Storey, M.A. et al. (2002) Evaluating the usability of Web-based learning tools, *Educational Technology & Society Journal* 5(3).

Tan, G. (2002) Blueware: Bluetooth Simulator for ns MIT Technical Report, MIT-LCS-TR-866, Cambridge, MA.

TechXNY CiberLibrary: Bluetooth, (August, 29, 2003), Retrieved: November 12, 2003, from http://techxny.bitpipe.com/data/detail?id=1062687864_97&type=RES&x=1260093442

The Network Simulator – ns-2, (n, d), Retrieved: August 20, 2003, from http://www.isi.edu/nsnam/ns/

Tuovinen, J.E. (2000) Optimising student cognitive load in computer education. *Proceedings of ACE 2000*, Melbourne, Australia.

Volbert, K. (2002) A simulation environment for ad hoc networking using sector subdivision *Proceedings of the 10th Euromicro Workshop on Parallel, Distributed and Network-based Processing (PDP'02*), Spain.

Yaiz, R. et al. (2001) Polling in Bluetooth - a simplified best effort case. *Proceedings of the 7th Annual CTIT Workshop*.