

Networked Objects

Patrick Dwyer

Spring 2006

Week 2 - January 26th

Calm Technology

Most people would not consider technology a “calming” influence in their lives; in fact quite the opposite. Networked technologies tend to be inherently disruptive; pagers, cell phones, instant messages. All of these bring distant information into the foreground of our concentration, disrupting or intruding upon the task at hand. A calming technology still attempts to bring distant or extra information to our attention, but through the use of peripheral awareness.

Individually we can typically focus on only one or two pieces of information at a time, and there is a considerable time investment in breaking or beginning our attention. On the other hand we can monitor the periphery, visually, audibly, or through any of our senses, for changes and slight alterations, without sacrificing our primary attention. The monitoring of periphery works only so long as the periphery allows us to maintain a suitable attentive distance. For example, consider two scenarios, one in which we monitor conditions external to our primary attention, and one in which the periphery enhances our primary attention.

The introduction of easy to use video chat has opened up a world of communication previously relegated to those with significant technological skill. Before then most people had to make due with text chatting. In itself textual chat was useful, but there was a noticeable lack of contextual information regarding the people participating; hence the introduction of emoticons, short-hand context phrasing (LOL, etc), and many miscommunications of humor, intention, and sarcasm. With video chat we retain one of the most familiar peripheral sources of data; human physical expression. Being able to see a person’s face, watch their body language, while chatting informs us in a far more detailed and accurate manner than text ever could. This peripheral, calm, technology (the video chat) is laden with information that we unconsciously interpret; but the periphery can intrude on the primary, breaking our attention. When things in the background change too quickly, or the video is sudden altered, we break concentration.

When our primary attention is costly to break, it is important to push distraction as far to the periphery as possible. But there are some things we still may need to monitor, and we need to keep some things at a suitable distance at which we can monitor them, without having them become a primary focus. Email is quite typically a distraction for people, pulling them away from the task at hand every few minutes to check for new mail. This simple act of checking for new email can be a significant break in attention, involving a mental context switch between programs. By creating methods to acknowledge the receipt of new email in a peripheral manner, we can remove the distraction of constantly checking our email. We can inform ourselves of new email in just about any sensory manner, the most common of which are audible background sounds, or small visual cues.

Network Layers - The OSI Model

For two people to communicate a number of things must be agreed upon before information can be exchanged. We can define these things with a few questions; In what medium will we communicate? In what language will we communicate? Who should initiate the exchange? What does a piece of communication look like? These questions apply to human communication as much as computer communication. In the case of two people talking, we answer these questions based largely on context; if we are speaking to one another we use the medium of sound, and the language we speak will be decided by our common environment. The first person to speak initiates the conversation, and either person can speak first. Hope-

fully the context of the conversation will be guided by the grammar and convention of the language we're speaking.

Answering these questions allows us to determine the protocol of communication, the steps and details that guide a successful exchange of information. When dealing with computer networks these questions can be further refined into a communication model, in this case the OSI, or Open Systems Interconnection, model. The OSI Model breaks communications into seven layers, where each layer is responsible for a specific component of communication. The benefit of this layered model is that each layer hides the details of communication from the layer above it, so that certain layers can change their mode of operation without the layers (or programs) higher in the model needing to know the details.

Check the readings detailed below for more detailed information on the OSI model. For now we'll look at the basics of the OSI networking model:

Physical

The physical layer defines the actual medium across which information travels. In our example of two people speaking this the medium of air. In a computer network this might be ethernet cable or serial cable.

Datalink

The datalink layer defines the format of data on the physical network. For our two people talking this would be sound waves, and for our network a packet. The datalink layer also handles logical addressing of data; for our network this would be the MAC (Media Access Control) address of our computer.

Network

The network layer controls the layout and routing of information. For two people talking this is a rather amorphous concept, but we could consider physics the routing principle. For our network this is typically Internet Protocol, where our computer receives an IP address when it connects to the network.

Transport

On top of the network layer we have a transport layer, which defines the mechanics of sending and controlling data across the network. We'll be using TCP (Transmission Control Protocol) and UDP (User Datagram Protocol) to transport data.

Session

The session layer controls the format of the data being transported over the network. This might be grammar for two people talking, or FTP commands across a network.

Presentation

Because different operating systems can represent data in different ways we use a presentation layer to handle the conversion between network representation and local representation of data. When two people are talking a common word may generate two different mental pictures, one for each person. In our network a Mac and PC may order the bits in a byte of information differently.

Application

This is the application data as represented to the end user. This could be a program or service running on a computer, or the data as presented to the code we write and run.

The beauty of the OSI network model resides in the distinct separation of these layers. When running an FTP client (application layer), my network could be based on Internet Protocol (network layer) over Ethernet (physical layer), or Internet Protocol (network layer) over Bongo-drums (physical layer)¹.

Control Issues

Working with networked communication requires a different approach than working with local sensors and attached devices. As we function in a networked world we become affected by the network; disruptions, delays, other programs, mishandled information. When we open our devices to the network we need to address the control and communication issues that arise.

Network Delay

In Intro to Physical Computing we interface with devices and sensors that give us an immediate response when we request data; not so for the network. When we request information, or respond to a request from another device, we need to take variable time delays into account. This means that our code controlling our network device can't be constructed in such a way that it demands immediate data or fails (crashes). Most of the time this isn't a problem, but it is an issue of which we need to be aware.

Information Formatting

Part of the idea behind a protocol is a specific format for the information being transmitted across the network. This means that our programs, if sending data to a remote program, need to correctly format the information, otherwise we cannot expect the remote program to correctly interpret what we desired it to do. Likewise when receiving data from a remote program we need to know of and correctly process the formatted data we get. The issue of formatting can be somewhat mitigated by writing our own server programs that handle the formatting details, which might be beyond the ability of our micro-controller.

Problems do occur, and information isn't always in the format we expect it to be. Remote programs can malfunction, produce errors, mis-format data, the possibilities are endless. Our hardware (or our server programs) need to be aware of this possibility, and correctly deal with malformed data.

Network Failure

Network outages are a way of life, and just as we get on with our day when the network isn't functioning properly, we need to design our devices to function in the eventuality of a network failure. We'll look into specific details when we come across them, but a typical solution is to make your devices capable of generating their own "faked" data to use when the network isn't responding. Of course this doesn't work in all cases, so our solutions will need to be tailored to the circumstances.

Introduction to Lantronix Devices

This week we look at two of our Lantronix devices, the Cobox-Micro and the XPort. Excellent tutorials on configuring and using these two devices can be found at:

<http://tigoe.net/pcomp/cobox/>

This week we only need to get familiar with the setup and configuration of our devices. Next week we get into communicating with our embedded network controllers. We'll cover four basic steps to make sure our XPort is working properly; configuring the XPort through a Terminal program, configuring the XPort over Telnet, echo data between telnet and a serial terminal, and finally configuring the XPort over the web. Specific notes on the configuration steps we take with the XPort are available at:

<http://tigoe.net/pcomp/cobox/cobox-config-notes.shtml>

¹ <http://eagle.auc.ca/~dreid/>

Assignment

Readings

- The Coming Age of Calm Technology, Mark Weiser
 - <http://www.ubiq.com/hypertext/weiser/acmfuture2endnote.htm>
- Intro to the OSI Model, Anthony P. McCullough
 - <http://users.erols.com/amccull/osi.htm>
- Detail on the OSI Model
 - http://www.uwsg.iu.edu/usail/network/nfs/network_layers.html
- Equator Project
 - <http://www.equator.ac.uk/>
- Disappearing Computer Initiative
 - <http://www.disappearing-computer.net/>
- Using Digital but Physical Surrogates to Mediate Awareness, Communication and Privacy in Media Spaces, Greenberg and Kuzuoka
 - <http://www.cpsc.ucalgary.ca/grouplab/papers/2000/00-Surrogate.PersTechnology/dig-phys-perstech.pdf>

Start Journal

This week I would like to receive a link from everyone in the class to the journal they will be maintaining throughout the semester. As well you should start keeping journal notes on the readings and technology we cover.

Order Network Controller

This week we should begin ordering network co-processors. We'll do a bulk order of XPort devices from Lantronix, which we'll organize at the end of class. Once we have XPorts to work with we can start building XPort boards, and configuring the devices.