Installation of Vicki and image capture on the Western Flyer

Mike McCann 17 September 1997

Background

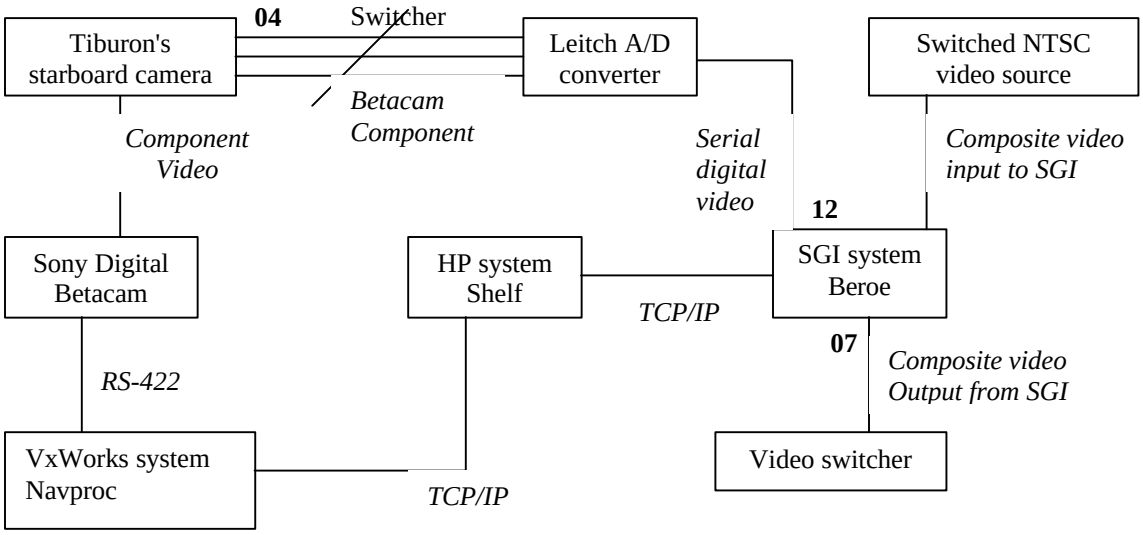
The Vicki program (Video Information Capture with Knowledge Inferring) has been installed on the Western Flyer. Vicki has been used for over a year in MBARI's Video Lab and since July 1997 on the Pt. Lobos to annotate video and snap video frame grabs. Vicki's purpose is to build a database of quantitative observations and to help index our video archive. It was designed to be an easy-to-learn system to create consistent annotations among various video interpreters.

Currently, the Video Lab has a 2-year backlog of ROV expeditions to annotate. A requirement therefore exists to annotate video in real-time as it is being recorded during the ROV dive. We also have a requirement to mark "events" where we will permanently log observations such as vehicle positions, environmental information, and camera parameters. A real-time video annotation system can satisfy this need.

This document describes the technical details of the installation on the Western Flyer so that an operator can understand the system and diagnose problems that might occur. The configuration of the system is very similar what is installed on the Pt. Lobos; in most cases only the names of the machines are different.

Hardware system component diagram

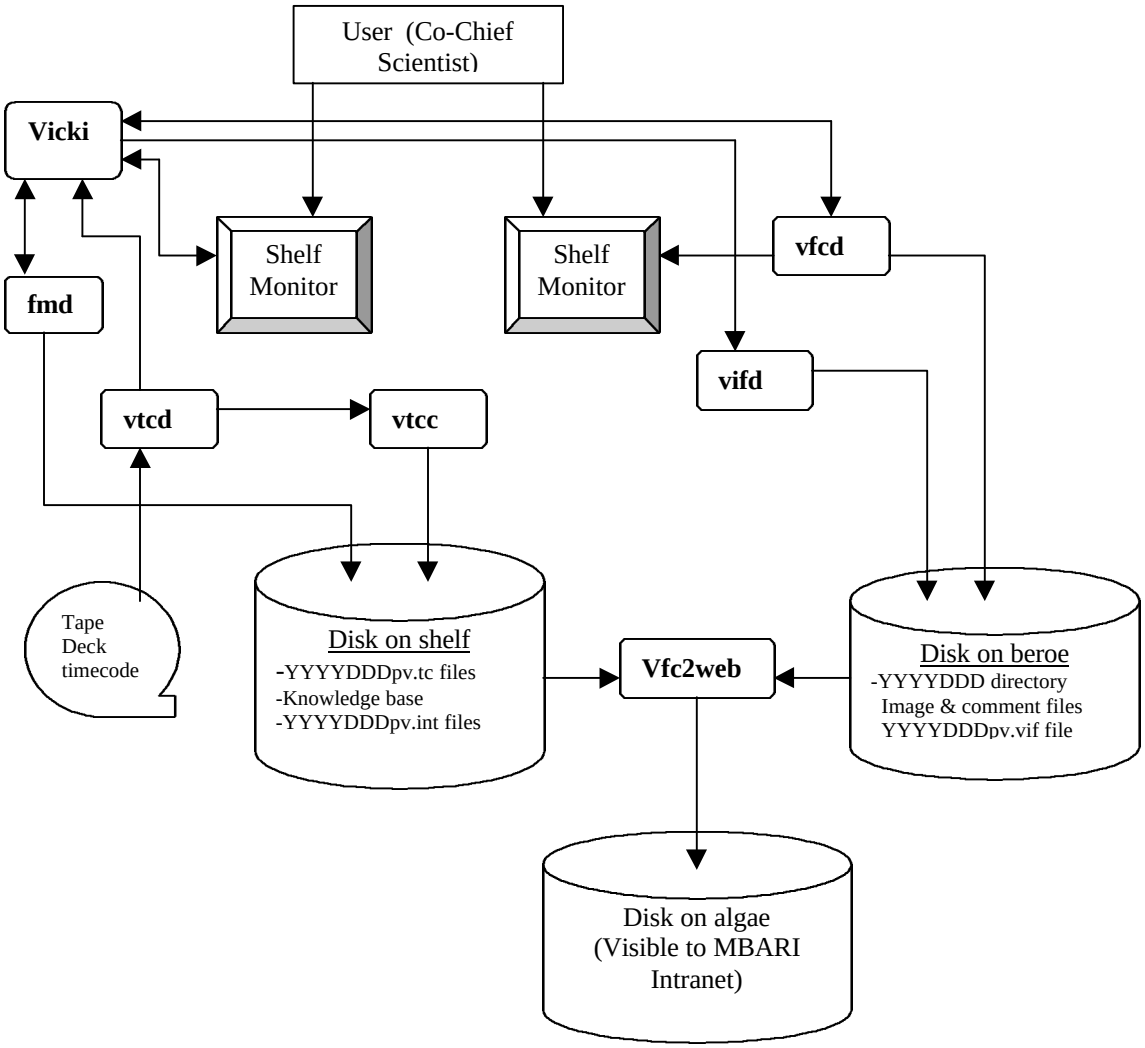
Below is a diagram of some of the hardware components (boxes) connections between them



To configure vicki so that the *Snap Main* button grabs images from Tiburon's right camera video source 04 must be set to destination 12. Any other composite NTSC video source may be mapped to destination XX and the *Snap Aux* button will grab images from that source.

Software system component diagram

Below is a diagram of the programs, clients, servers, and file storage systems that make up the Vicki and image frame capture system. The modules in bold text are described after the diagram.



Vicki

The **Vicki** module is centerpiece for this system. It is the primary program with which the user interacts. It is a Smalltalk program that is started from the *start* script in the vicki account on shelf. It communicates via sockets to other services available on hosts shelf, navproc, and beroe.

Vtcd

The video time code daemon (**vtcd**) is a network service that delivers longitudinal time code from the digital betacam tape deck that is connected to a serial port on the nav processor. The nav processor delivers time code to **Vicki** and to **vtcc** (the video time code client) which is used for logging UTC time and tape

time code to files on the shelf file system. **Vtcd** runs on the nav processor and is registered as service *vicki.vtc 9000/UDP*.

Fmd

The file manager daemon (**fmd**) is used by **vicki** to manage a "file database" for the knowledge-base and interpretations created and edited by **vicki**. It also handles user authentication when annotations are edited. It runs on host shelf and is registered as service *vicki.fm 9003/TCP*. **Fmd** is started for each new connection on shelf by inetd(1), file /etc/inetd.conf is specially configured for this operation.

Vtcc

The video time code client (**vtcc**) is started by user vicki's *start* script on shelf. It is used to provide backup logging of real UTC time and betacam tape time code. The network time protocol daemon (xntpd) is run on shelf for accurate timing on the system.

Vfcd

The video frame capture daemon (**vfcd**) runs on host beroe and serves to capture still digital video images from either the serial digital input (Snap Main) or the composite input (Snap Aux). It is started upon bootup of beroe by the /etc/init.d/vicki script. It receives messages from **vicki** on service *vicki.vfc 9006/TCP*. After an image is captured it is written to disk. A copy of the original is converted to a JPEG preview with overlain text; this image is displayed on shelf's monitor as confirmation that the image had been saved. The environment variable VKSNAPHOST, set by vicki's *start* script, specifies the host where **vfcd** runs. Only the last 4 frame captures are shown on the screen; old ones are killed to preserve system resources.

Vifd

The vicki interchange format daemon (**vifd**) is a simple logging service that runs on host beroe. It is started upon bootup of beroe by the /etc/init.d/vicki script. It receives ANNOTATION blocks from **vicki** on service *vicki.vif 9007/TCP*. **Vifd** will eventually be replaced by a direct connection to the VIMS database so that real-time database updates can be made. The environment variable VKVIFHOST, set by vicki's *start* script, specifies the host where **vifd** runs.

Vfc2web

After each day's operations the video frame capture to (2) web script is run on host beroe. **Vfc2web** takes all the files created by the user with vicki, compresses them, constructs image mapped web pages for the images, and transfers them all to a shore-side system that is visible on the *Where was the Western Flyer?* Intranet web pages. **Vfc2web** is a perl script that is currently run by cron(1) at 4pm each day from NIS account mccann, the program lives in /u/mccann/Media/vfc2web. Network time protocol is run on beroe for accurate timing.

What can go wrong?

Having an understanding of all the connections that the **vicki** system makes (and needs) is helpful for diagnosing and fixing problems as they occur. For instance, if the **vtcd** service is not available on navproc then time code is not received by **vtcc** or by **vicki**. Vicki's VCR window will show "88:88:88:88" if it can't get a timecode from **vtcd**. When the betacam deck is turned off the VCR window will show "99:99:99:99".

If one cannot log into server shelf to start a new annotation then there is a problem with vicki.fm. (Our experience so far shows that this is rarely a problem.) If a *Snap Main* button fails to grab an image then there may be a problem communicating with host beroe's vicki.vfc service. There could be a network problem or something wrong with the beroe system. The specific error messages received would have to be analyzed to locate the problem. The modular design of the system aids in the upgrade and maintenance of it, but adds to the complexity and the need for understanding how it all works.