DOCUMENT RESUME

ED 456 463 CS 510 596

AUTHOR	Tetzlaff, David
TITLE	Non-Linear Editing for the Smaller College-Level Production Program, Rev. 2.0.
PUB DATE	2001-07-00
NOTE	90p.; Paper presented at the Annual Meeting of the University Film and Video Association (54th, Colorado Springs, CO, August 16-19, 2000).
AVAILABLE FROM	For full text: http://www.filmclass.com/nonlinr2.pdf.
PUB TYPE	Opinion Papers (120) -- Speeches/Meeting Papers (150)
EDRS PRICE	MF01/PC04 Plus Postage.
DESCRIPTORS	*Editing; *Film Production; Higher Education; Instructional Effectiveness; *Liberal Arts; Small Colleges; Technological Advancement
IDENTIFIERS	Discourse Aims; Media Courses; *Nonlinear Models; *Technology Integration

ABSTRACT

This paper focuses on a specific topic and contention: Non-linear editing earns its place in a liberal arts setting because it is a superior tool to teach the concepts of how moving picture discourse is constructed through editing. The paper first points out that most students at small liberal arts colleges are not going to wind up working professionally in the field of film production, and questions why they need to be taught about operating any kind of editing system, since the professional environment will almost always afford them the services of an expert technician. It also notes, however, that the separation of theory and practice is deeply problematic in terms of social philosophy, much less pedagogical philosophy, and that there is both practical and conceptual value in learning how to make a video edit. The paper is divided into the following detailed parts: (1) Non-Linear Editing and Liberal Arts Pedagogy; (2) General Issues in Technology; (3) An Ideal System...; and (4) Specific Systems. Appended are: G3/G4 Genealogy; Using Removable Hard Drives with a Mac; and Review of the Matrox RT2000. (NKA)

Non-Linear Editing for the Smaller College-Level Production Program, Rev. 2.0.

by David Tetzlaff

Non-linear Editing for the Smaller College-level Production Program, rev. 2.0

David Tetzlaff
Assistant Professor, Film Studies Program
Connecticut College
July, 2001

Part 1: Non-Linear Editing and Liberal Arts Pedagogy

Does NLE = Not for Liberal Arts Education?

Last year (2000) an article in the trade magazine *Video Systems* profiled several production facilities that have recently switched platforms for their non-linear editing systems, moving from Mac OS to Windows NT. The featured example was a video editing / multimedia lab at Northern Oklahoma College, managed by a self described die-hard Mac enthusiast, that had nevertheless just installed 15 Discrete Edit workstations, all connected to a huge central storage server via fibrechannel.

This is the not the world I live in. I've held several teaching positions in media production in various parts of the country, and I've interviewed for many more. Never have I encountered such a facility, or even the hint of a budget adequate to create something like that. The schools in my world might have the funds to by *one* Discrete Edit package, though a good number are not even that flush. And an expert full-time facility manager for an editing lab? Fuhgeddabottit!!

This is not really a case of rich versus poor, but rather of difference in purpose. Regional colleges like Northern Oklahoma are looking to provide concrete pre-professional training to students who will seek employment as edit-system operators in professional media facilities. The colleges in my world, on the other hand, situate production as only a relatively small part of some larger general communication or media studies curriculum, under the umbrella of the liberal arts. Thus, the scale of facilities and budgets are naturally dialed down considerably.

Which brings us to the initial question: if Northern Oklahoma is an example of what a good NLE education facility looks like, do schools with only a fraction of that budget have any business looking into non-linear? And a corollary question: the Northern Oklahoma program frames the act of operating a non-linear systems as a highly technical vocational specialization, so is non-linear equipment even appropriate for a broader liberal arts curriculum, where the focus is on "education for life" rather than training for a entry-level job?

In other words, we might conclude that non-linear has no place within a liberal arts program because first, we can't afford to do it properly, and second, it's too technical anyway. And so we may imagine a scene where two people sit in front of an Avid. One, with a degree in Sociology from UC Berkeley, is a producer/director giving instructions on how to cut the program at hand — say a documentary on urban problems — while the other, with a certificate from a local JuCo, is an Avid editor who makes the instructions come to life on the computer, but has little or no creative input in the construction of the show. This is pretty much how things often are in the "real world." So we can reframe our question in a more practical mode: if, as a liberal arts program, we recognize that many of our students aren't going to wind up working professionally in the field, and if our aim for those that do is to wind up as writers, producers, and directors, why do we need to teach them anything about operating *any* kind of editing system, since the professional environment will almost always afford them the services of an expert technician? Why don't we just teach them theory and aesthetics and chuck production altogether? (Obviously, this is an approach that many liberal arts programs have adopted — not all, thank heavens, or more people in my line of work would be unemployed.)

Well, first of all, the separation of theory and practice is deeply problematic in terms of social philosophy, much less pedagogical philosophy. A proper discussion of this would take a whole different essay, so I'll just ask you to imagine such a paper — complete with references to Harry Braverman's *Labor and Monopoly Capital* on how political control is achieved by separating conception and execution, as well as references to the Frankfurt School on the value of *praxis* — and note on a similar but more mundane level that liberal arts programs have a mission that extends beyond the realms of industry practice. Among other things, we generally want to be a place where truly independent media makers can develop. The independent film or video maker necessarily wears many hats, and often works without specialized technical help. There is a point, of course, where a certain level of complexity leaves the independent artist behind — only a very few filmmakers doing short-form experimental work develop and print their own filmstock, for example. We can agree that a video-maker need not know how to assemble the computer components of a NLE, or write the software code of the editing program. However, there is both practical and conceptual value in learning how to make a video edit.

The conceptual value is the heart of the matter. More than 25 years of studying and making media have led me to the conclusion that conception and execution are two sides of the same coin. If you want to learn how to make a film, you should study how other films are made. If you want to learn how films work, you should try making one. *Production courses exist within liberal arts programs not to train people how to make professional quality media artifacts, but to help them learn how the medium works.* The hands-on experience is only valuable to the extent that it is also a minds-on experience — but in this capacity it enriches the education of those minds in a way that a purely theoretical pedagogy cannot match. The majority of our students are *not* going to become media professionals. Yet in our media-saturated society, where the ubiquitous solicitations of mediated messages not only function as art, information, diversion or spectacle but also as primary vehicles of power — the idea that an educational institution needs to justify media studies on some sort of vocational basis is just as absurd as arguing that, in an age of threatened ecosystems, the biology curriculum can only be warranted by the generation of professional scientists. And just as the biologists know that every student — whether working on the major or just fulfilling a gen ed requirement — needs to experience a lab section in order to truly appreciate the principles of the science, so, I believe, it is with film/video production.

Why NLE?

To return, then, to the specific topic of the essay: *Non-linear editing earns it place in a liberal arts setting because it is a superior tool to teach the concepts of how moving picture discourse is constructed through editing.*

First, a NLE is actually easier to learn than a comparable tape-based system. Realizing that some of you may take issue with that statement, let me point to 'comparable' as the key term. There is no question that a cuts-only tape-based edit system is easier to learn than an NLE — unless you limit the NLE to cuts only as well. If all you need to learn is how to trim clips and drag them one after the other into the timeline, there's not much to NLE. The success of iMovie, which almost anyone can learn in about 15 minutes, demonstrates this. Even in cuts-only work, once you advance beyond straight cuts of video and single audio track at the same point, linear gear becomes more complicated than a NLE. Setting up split edits, adding and mixing additional tracks of sound, are quite difficult on linear systems. Finally, a cuts-only system necessarily removes two key elements of moving picture language from your working vocabulary — the fade and the dissolve. To add these in a linear system, you need to go to an A/B roll setup with three VCRs. A/B roll controllers are quite daunting for students to learn.

The subject of transitions brings up another possible objection to non-linear editing in a liberal arts context. NLEs, after all, are often judged in the marketplace on how many transitions they offer and the spectacle value of these effects. True 3D page peels and PIP spinning inside a 3D box!! Many educators may feel that this effect bonanza is all a load of crap, a substitute for knowledge and creativity in the more fundamental aspects of the medium, and seek to keep students away from this sort of thing — I happen to be one of them. However to keep students away from NLE just to keep them away from the cheesy effects that come with the NLE is throwing out the baby with the bathwater. Effects generally exist in the form of software plug-ins, and we can simply delete them from the system — forcing students to ask to get one reinstalled for a specific use — which they would need to be able to justify in aesthetic or semiotic terms other than "I just thought it would look cool!"

The second benefit of NLE in the classroom is that the representation of the production depicted in the timeline allows students to visualize the structure of a piece, and see how it changes dynamically as different edit choices are made. In the "old days" when I taught production using linear tape-based systems, I was always drawing out what amounted to timelines on the blackboard to explain how different pieces of the show are fit together. Having the computer do this, individually with each student, is a definite improvement over my crude drawings and general examples. This is especially helpful in terms of teaching the editing concepts that are most difficult to learn on linear systems — how split edits and video inserts construct a more integrated continuity than using sound and image recorded at the same time; how sounds from multiple tracks combine in a mix depending on relative audio levels; how timing adjustments affect the feel of dissolves and fades.

Third, and by far most important, a NLE allows students to learn filmic language the way human beings normally learn things — by experimentation, trial-and-error. Once a choice has been made in a linear system, it gets locked in. A student shows you a rough cut of a short piece containing 15 shots or so. You comment that the timing on the second cut is awkward and suggest she play around with it some more. On a linear system this is very difficult for her to do. After finding a new spot for the cut, she must either redo all the cuts she made later, or dub them the edit master and re-edit a copy of the old end back onto the new beginning at a loss of at least one generation — which doesn't allow for too many opportunities to redo things before the quality of the image degrades below acceptability. Non-linear, on the other hand, allows the editor to change her mind about any part of the program at any time during the editing process. It also makes the creation of multiple alternate versions extremely easy. Students learning the medium can easily explore these variations for themselves, and share them with their peers or their teachers. Of course, this freedom of decision making was something film editors always had, and was so central to their creative process that they demanded it be replicated in video once film production moved into electronic form — which is how NLE got developed in the first place.

The Problem with NLE

NLE may be a wonderful tool for teaching the basic principles of film/video editing, but every benefit comes with a cost. The most daunting of these may seem to be the price of the equipment involved. Non-linear systems have traditionally been very expensive, with decent packages starting at around $15,000. In contrast, the JVC Edit Desk system — a very good, linear, cuts-only S-VHS edit system targeted to the education market — runs around $6,000. Lower cost NLE products have been available for awhile, but they've largely been trouble-prone, low performance systems that are far more pain than they're worth. Fortunately, around 1998, advances in computer technology caught up to more modestly-priced NLE offerings, and a number of very-capable packages can now be acquired in the price range of the Edit Desk, or even less. The bulk of this paper discusses several such systems in detail.

However, the cost of NLE is not a matter of equipment purchase alone. There is a considerable amount of work and time demanded in terms of set-up and maintenance — more so than tape-based gear requires. This may eventually translate back to budget figures in terms of added labor cost. However it's more likely to materialize in the form of added time and pressure commitments on already busy salaried employees, either for technical services support staff or, more likely, for any faculty member teaching with the NLE.

The two most important questions specific to creating a NLE lab in a college environment are: 1) Where are the students going to store their media files? 2) Who is going to take care of the equipment?

NLE systems are generally designed for stand-alone, self-contained applications —

with two or three users and two or three projects at most occupying the system during any period of time. The standard design of an NLE does not easily accommodate itself to the comings and goings of a large number of users, as is the case in a typical college lab. A larger number of users would be better accommodated by multiple systems, with the ability for users to work on a given project at any open station. However, short of the sort of costly fibre-channel setup mentioned in the introduction, standard professional NLE design offers no easy way for a user to move from one computer to another. If you create a small tape-based video editing lab with three stations, for example, the lab can always accommodate three students working at once — they just pop their tapes into whatever station may be available. If their work is stored instead as files on a fixed hard drive, this efficiency is lost. Three students may come to the lab to work at the same time, only to find their projects are all stored on the same computer. While one can work, the other two cannot, and the two other systems in the room stand unused. This can be addressed to some degree with a precise reservation system for each station in the lab, but this creates an extra administrative workload, and given the over-booked schedules typical of today's students, conflicts will still be a significant problem. A better plan is to find an economical way to allow students to move their work to any available machine. There are several technological solutions for this, to be discussed below, but most of them involve setting up the NLE system in atypical ways, going somewhat against its inherent design. This means, first of all, more work on the part of whoever designs the lab — you may not be able to just order a standard package off the shelf — and it also results in a system that may require more frequent maintenance and troubleshooting.

A certain measure of maintenance and troubleshooting, of course, is just part of the computer age. It's not that NLE systems get sick that much more often than linear systems. It's the nature of the illness. Problems with a linear system are usually either very simple — someone unplugged something, or flipped a switch into the wrong position — or very serious — something breaks inside the machine. Neither requires too much labor on the part of the college staff. In the first case it's relatively easy to trace the error and reset things to normal, and in the second case you just box up the machine and send it off to the repair center. Once the system becomes computerized, however, the software issues analogous to mispatching and flipped switches are not only far greater in number but much harder to track down. There is also really no particular set of behaviors you can teach students to avoid instances of corrupted files and memory leaks. To be fair, we must note that NLE systems eliminate some of the time consuming technical problems that plague linear tape editors — especially control track issues. Still, they need regular attention, and more importantly, they need attention from someone with a very particular expertise. NLE is a very specialized computer application, and the people who perform maintenance on the general population of desktop computers on campus most likely do not have the knowledge to address the problems attendant to the specifics of computer-video technology. Again, this burden tends to get shifted back upon the faculty teaching in the lab, which can result in a serious overload on top of a full schedule of classes, committee work, and expectations for publication.

Given the typical culture and institutional practices of liberal arts programs, the support issue may be the most formidable obstacle in the quest for a functional NLE lab. Colleges just don't seem to have a framework for providing technical assistance on anything other than a large scale — the lab at Northern Oklahoma, for example. At large universities, while the role and scale of production in the curriculum may be limited within the boundaries of a liberal arts approach, there still may be enough students passing through the program to justify a larger facility and full-time engineering support. There's also a legacy at large schools of academic departments employing engineers to service television studios used in broadcasting classes. At a small liberal arts college, or a limited program within a small to mid-sized university, there simply is not the economy of scale to operate in that manner. While the rest of this paper addresses many of the issues in equipment cost and facility design that apply to bringing NLE to a liberal arts setting, I do not mean to suggest that these are the only factors that must be considered, or that what follows is a complete blueprint that can enable any program to succeed in adding NLE to its teaching resources. These are pieces of the puzzle, no more.

General Criteria for NLE in a Small College Environment

Within a liberal arts film/media curriculum, NLE is not a proper subject in itself — we are not aiming to teach students the technical process of using a computer-based production system. Rather, NLE is a *tool* for teaching students how to use the language of moving images. If our goal was to train edit-system operators for the production industry, we would want our lab to replicate, to the best of our ability, the technical environment of a professional facility. If we were not able to obtain the exact types of equipment and software that define the "industry standard," we would seek substitutes that operate in a similar manner.

Unfortunately, too many school programs look to "how the pros do it" as a model, even when their pedagogical goals are not defined by pre-professional training. Technologies have certain biases, which they tend to with them when installed in a teaching facility. That is, a specific technology tends to enable certain teaching goals, while providing obstacles to others. If we approach production education from a liberal arts perspective, our lab needs to meet a different set of needs than we would find at a professional shop.

Multiple Users

First, a liberal arts production lab needs to easily allow every member of the class to work on projects where they have creative ownership and full opportunity to experiment with the tools of the medium. In most cases, this means the lab must accommodate many more users at one time than any professional system would bear. We might be able to get by with a single workstation for a single very small class, if it is accessible 24 hours a day. But with more typical class sizes or restricted access times, our lab will need multiple workstations, and the ability for users to move their work easily from station to station, in order to provide the kind of authorship experience that suits the goals of the pedagogy. As I've already noted, this creates a variety of technical headaches since NLE systems are not designed this way.

Ease of Use

Ideally, the lab should place the least amount of technical routine possible between the student and the creative act. Professionals demand a wise range of options and tweaks — which leads to a level of complexity in hardware and software that can be very intimidating to the beginner. We should aim to simplify wherever we can. Unfortunately, this is not easy to do. Available solutions to the problems of multiple users add elements of technical complexity, and the more simple editing systems created for 'home movie' users may be *too* simplified once students move beyond the most basic techniques. iMovie, for example, offers no way direct way to change the in and out points of a clip once you've set them initially. The concept of simplicity should be evaluated comprehensively, taking into account all tasks students will need to perform on a regular basis. Simplicity in one area may lead to problematic complexity in another. Thus, what appears simple on the surface may turn out to have hidden intricacies, while what appears imposing at first look may be a more usable tool overall.

One extremely important use factor in a student lab is the time it takes to perform necessary tasks on the equipment, especially those of a merely technical nature. Students have other classes and often work to help pay tuition. They have only a limited amount of time to devote to your production class, to spend working in the lab. You want as much of that time as possible to be used working with the *medium itself*, not the technology of the medium. That is, you want them to spend their time making edits, not setting up the machine to make an edit, or watching the machine laboriously create an edit.

Appropriate Conceptual Complexity

The system should enable working with any aspect of textuality that the faculty consider *essential* to the medium. What these are exactly will vary from situation to situation. For example, I consider the ability to mix multiple audio tracks and to create split edits to be essential in my classes, which are based in narrative form. A faculty person working in a more experimentalist mode may place more importance on the ability to add certain kinds of filtering or effects to an image.

The problem is that almost all video tools are designed for one of two primary markets: professional users who want lots of technical control, and home users who want the process as automated as possible. The professional tools tend to be overly complicated, and complicated in non-useful directions — overloaded with tacky 3D transition effects, for example. The home movie tools, though, tend to be too simplified — cameras without manual focus or exposure control, editing systems with no audio flexibility.

I emphasized the word *essential* above, because I believe we should be prepared to sacrifice capabilities we merely *like* for the sake of simplicity. However, even after we pare our needs down to an essence, we may find ourselves needing to ratchet the overall complexity of the system up quite a bit in order to obtain a key feature. This means that the *Ease of Use* goal cannot be embodied solely in the tools in the lab themselves, and falls back of the ability of the instructor to develop instructional methods and materials that help guide students through a maze of commands and controls to the ones they need to use — and allow them to turn of or ignore other functions.

Stability and Disaster Prevention

NLE systems are somewhat temperamental beasts in general, but some are more temperamental than other. In a school lab, where the systems will likely be handled by more users and receive less routine maintenance than they would in a professional shop, stability is more difficult to maintain, though just as important come deadline or due date. The best route to stability is to build a lab out of components that have been out for awhile and gained a proven track record. Unfortunately, in the fast-moving world of computer technology, products have a short market life, and something newer, better and cheaper is always being introduced. When these improvements are incremental, it's easy to stay with older, proven choices. However, some new products offer truly significant advances over previous technologies. These benefits may be impossible to resist, but you should realize that the latest and greatest is going to some with its share of bugs that will take some time to be resolved, and be prepared to deal with them.

Computer media is a more fragile form than videotape. Tapes can be 'eaten' by machines or accidentally erased, of course, but these are fairly rare occurrences in comparison to the kinds of calamities that can befall a hard drive. One of the advantages of computer-based editing is the creation of an Edit Decision List (EDL) that allows an edited program to be re-created fairly easily from the original tape if the digitized media on the hard drive gets removed or destroyed. (The EDL is built into the project or program file that constitutes the basic working document of NLE software.) This can be a marvelous policy for student projects, and mitigates against just-before-due-date laments of "the hard drive crashed" either as traumatic calamity or manipulated excuse. In order to make use of EDLs, the hardware components of the system must be equipped with timecode capability, and the students may need to perform one or two steps in the production process to generate or use timecode properly. This does add more complexity to the system, and the value of the trade-off may be arguable. Like any insurance policy, the cost seems an annoyance until you need it, but if the need arises in the absence of insurance, the result is far worse than annoyance. In my experience within small programs a problem comes up between once a semester and once a year, which is often enough that I feel timecode is a necessity. (Timecode is also required to edit in offline/online mode, which is a key for accommodating multiple users on limited budgets.)

The most important thing you can do to help keep a NLE system stable and functioning is to focus the computer system on doing only that one thing. Most of the hopelessly messed-up NLE systems I've seen have a variety of other boards and other kinds of software installed in them, and most of the NLE systems I've see equipped to do lots of other tasks besides editing have been hopelessly messed up. Install only the software you need for video. Put the web design or DTP, or multimedia authoring stuff on another machine. Do NOT connect your NLE computer to the network (this also keeps people from using it to read email), and delete the network software. This not only makes the system more likely to stay healthy, it also makes tracking down problems a lot easier when it does get sick.

PART 2: General Issues in the Technology

Certain types of technological choices must be made in equipping a NLE editing lab. Each has consequences not only for the budget of the project, but for the pedagogy that comes out of it.

Quantity

How many stations do you need? This is not a simple matter determined by how many students will be enrolled in production courses. It also depends on what hours the facilities will be available to students. Ideally, students should have access to the lab 24 hours a day, 7 days a week. However, at some schools, security concerns for the campus at large or the building housing the lab may restrict the hours to 9-9, or even 9-5. (Few programs will be able to afford to have a staff person on duty in the lab even for these limited hours, so some other measures will need to be employed to make sure only authorized students use the lab and no equipment gets stolen.) Obviously, the fewer hours the lab is open, the more equipment it needs to contain in order to provide equivalent access. Access is a key to production pedagogy. When you teach a production class with too many students and too little equipment, this greatly limits the kind of assignments you can give, and the learning methods you can encourage. Students invariably wind up working in large groups in these situations. While this may provide certain benefits in terms of developing their social skills, or teaching them something about the specialized roles typical in professional production, it's hardly the ideal method for getting them to explore the workings of the medium. Students need more experience in authorship than a series of large group assignments can provide. I believe this is an essential part of a liberal arts approach to production. Could we imagine a creative writing class where students were assigned to create a short story in groups of four, or a history class where research projects were assigned to teams of five? In any type of class, I find it somewhat problematic to throw students into new activities completely by themselves, so where appropriate I often have them work on projects in pairs — for example in making class presentations on readings in a theory class. In my introductory production class, I like to have the students do four short projects — two individually and two with one partner.

So how much equipment is enough for a moderately individualized pedagogy? Based on my teaching experiences, which includes a variety of access situations, if the lab is open 24 hours, I would say: a single class of 10 students or less would be slightly constrained but could get by with one edit system and two cameras, though two edit systems and three cameras would be the most reasonable figure. With more than 10 students, I would say multiple edit stations are a necessity. If the lab is to accommodate 12-16 students, I would say two edit systems and three cameras minimum, with three editors and four cameras a more proper number. As the number of students grows from here, additional cameras would be the first priority. (I wouldn't add a fourth edit station until a sixth camera had been acquired). At Connecticut College we have designed our facility to handle up to 40 students enrolled in production courses during a single semester, and we have four edit systems and six camera kits. I must note that these facilities are dedicated to class use only and not shared with other users on campus who may be pursuing some sort of production project for a class in another department or for their own individual purposes. If the hours are shorter, the classes more numerous, the facility shared with other users, the numbers ought to increase in rough proportions.

Where do the files go? Fixed vs. removable storage; forms of removables

If you have only a single very small class at any one time, and a tight budget, you may find yourself working with only a single editing system instead of multiple stations. At least this makes the storage question easier by removing the primary advantage of removable storage, the ability to move projects around. You can just add more capacity to your system in the form of additional internal or external hard drives and divide it equally among the students. This is how NLE systems are designed to be used, and you'll have little problem finding appropriate devices to use and vendors who can set up a system in this manner. Fixed drives are also more reliable than any type of removable system, and create fewer technical headaches in setup and maintenance.

For even a single 'normal' sized production class though, much less multiple classes or a lab open to other users, a single system is not enough. Fixed drives make a poor choice for a multi-station lab. If you have 15 projects divided among three stations, inevitably three different users with their work on the same station will want to come in and edit at the same time. Of course they'll be the *only students* who want to work at that time, and you wind up with one student working, two others waiting, and two stations sitting unused because the files can't be moved. You can try to set up an elaborate reservation system to try to plan around these conflicts, but this adds greatly to the hassle factor for both the students and the staff. One way or the other, using fixed drives tends to subvert the purpose of building a multi-station lab — which is to provide ready access to creative tools.

What a multi-station lab really ought to have, then, is some sort of removable file storage system. This can take several forms.

The simplest form to set up technically is to equip each user with an external hard drive that can be plugged into the system with a simple cable. (A 'user' may be an individual, or two or more students working on the same project together...) This is made possible by Firewire technology (also known as iLink or IEE1394). Firewire is available on either Macs or PCs, though its implementation is more mature and more stable on Macs (Apple co-developed the spec with Sony). The main problem with external Firewire drives is cost. A news post on the Internet (www.2-pop.com, May 12, 2000) noted that Columbia University film school was requiring students using the school's Final Cut Pro editing lab to purchase their own 37GB Firewire hard drives — to the tune of $645 apiece. Since then, prices of these drives has dropped considerably. As of the summer of 2001, a 45 GB drive costs around $250. While a production student at a pre-professional program like Columbia's might be expected to support their education with a $645 purchase, few students in a general liberal arts program are going to be able or willing to cough up $250 just to take a video class. This places the burden on the college to acquire drives that can be checked out to students who are enrolled in a production course, and while I would like to think any school seeking to offer video ought to be willing to invest $250 per seat into the class for storage I realize this may be a bit to strong for many programs. Another problem with external Firewire drives: they're fairly bulky, and fragile enough that you wouldn't want students carrying them around campus. Which poses the question, where do you store the devices students use to store their data?

An inexpensive solution that is also easy to set-up simple is to place the digitized audio and video on a Castlewood Orb cartridge drive — a kind of muscled up Zip drive with much higher capacity (2.2GB) and throughput. Unlike the media for Iomega's competing Jaz drive, Orb disks are inexpensive, around $30, easily within the range of materials liberal arts students might be expected to spend on a class. Physically, Orb disks are just a bit bigger than Zip disks, thus easily kept in the students possession and providing no physical storage issue for the lab. Having students own their own storage also eliminates administrative work for the school personnel in charge of the lab. Unfortunately, there are several significant problems with cartridge drives. 2GB isn't a lot of space for digital video files, and the drives aren't fast enough for high quality video anyway. What's more, cartridge drives aren't anywhere near as reliable as standard hard drive mechanisms — a disk just dies every so often. (My experience with Orb drives and disks has been much better than my experience with Jaz drives and disks, though, for what that's worth.) So, all in all, working with cartridge media limits students to short projects of sub-VHS quality, and introduces a measure of instability into the lab that may well offset the system's benefits.

The third means of removable storage is some form of 'mobile rack' system: standard hard drive mechanisms mounted in small trays that slide in and out of a receiver frame mounted in the computer itself or in an external housing. A mobile rack achieves the same result as the external Firewire drive through different, and potentially more economical means. The external drives have a separate case and power supply for each unit, adding to the cost. The mobile rack allows working guts of the drive to be exchanged into a common case and power supply, thus lowering the cost somewhat. The mobile rack tray is also physically smaller than a complete external drive unit, and presents less of a storage problem. In our lab at Connecticut College we have installed small lockers made from standard post-office-box units, which are just the right size to hold a mobile rack tray and a few videotapes. While a certain effort on the part of the Physical Plant was required to turn the mailboxes into lockers — creating a freestanding frame to hold them and sealing the back — the resulting unit takes up minimal space in the lab.

The problem with removable hard drives is that you may need to be willing to engage in some tinkering to realize the savings. External Firewire drives are ready to go out-of-the-box, there's little system set-up involved. Unfortunately, there appears to be little market for simple inexpensive, pre-assembled removable drive systems. Instead, pre-packaged drive systems employing mobile racks have tended to be directed at high end professional users — UltraWide SCSI drives in heavy-duty cases arranged in multiple drive arrays (RAIDs). Such configurations are overkill and over budget for the sort of NLE a small college program ought to be considering.

For most of the young history of NLE, expensive SCSI interface hard drives have been prescribed for video work, and less expensive IDE interface drives have been considered too slow. But by 1999 the IDE spec had advanced to the point where the newer 'UltraATA' drives were fast enough for good quality video, especially DV format video streams. All currently existing Firewire drives (6/01), in fact, actually contain UltraATA mechanisms connected to the Firewire interface by means of a 'bridge chip' that translates between the two standards. This actually slows down the data throughput somewhat — any Firewire drive fast enough for video work is built around an UltraATA mechanism that would be even faster if connected to a standard EIDE port.

High performance SCSI drives usually generate a good of heat. This is one reason for the heavy duty, metal cases used in professional mobile rack RAIDs. IDE drives don't run as hot, and can be housed in plastic cases without problems due to lack of heat dissipation. By 1999, low cost plastic mobile racks for IDE drives appeared on the market as an add-on for internal mounting in tower computer cases. These may give the appearance of being a cheap and chancy substitute for cast metal housings, but we have been using them at Connecticut College for 2 years now with no real problems — and I wish I could say that for most of the rest of the gear in the lab.

A removable drive solution appropriate and affordable for a college NLE video lab using IDE drive mechanisms and inexpensive plastic mobile racks can be created out of component parts, if you have someone on hand with the minimal technical skills needed to put it together. Adding mobile racks to a PC based system is a fairly easy task, since Wintel machines are readily available in tower cases with open 5 1/2" bays to hold the receiver frames, and good interface cards to add a couple extra IDE ports are cheap and easily acquired. The problem here is that PCs in general are more trouble prone than Macs as video platforms, harder to service and maintain, and more difficult to get set-up properly in the first place. As I shall note again below, I strongly recommend any school acquiring a Windows based NLE to get a 'turnkey' system, assembled and tested by a video professional, rather than trying to put the system together yourself (with or without the aid of your school's Information Services staff). However, since turnkey vendors generally commit themselves to providing support for systems, but want to keep their support tasks manageable, they usually offer only a limited set of options in their systems, and — you guessed it — mobile racks aren't one of them. Its not *too* hard a do-it-yourself (DIY) project to add them to an appropriately configured system yourself, though.

Changing platforms shifts the issues. While Mac computers are in almost all other respects superior video editing platforms — more user friendly, more stable, easier to maintain — it is difficult to configure an IDE-based removable drive system on a Mac, since the computer has no open drive bays. Solutions exist, as I will detail in the below, but most of them require a fair amount of DIY tinkering. This is fine is you have the time and the skill, not so great for the not-so-handy.

In 2001 the Mac removable picture got brighter as Granite Digital introduced a new prepackaged removable drive system using a Firewire interface and IDE mechanisms in an external housing, perfect for digital video on a Mac. While not as cheap as the DIY solutions it is still modestly priced compared to similar systems based on high-speed SCSI mechanisms. For a lab of twenty users, the cost of the Granite Firewire removable system would be fairly close to the cost of fitting each user with an external Firewire removable, though the Granite system would be a more elegant solution, with higher performance and increasing economy of scale for higher numbers of users.

Summary

The best way to provide students storage for their projects in a multi-station lab is for each user to have their own movable hard drive. Cartridge drives are advisable only in cases where very small budgets or large numbers of users makes it highly impractical for each user to have a hard drive. Fixed storage is appropriate only for a small group of users working on a single station.

Online vs. Offline

Online Editing refers to working on a program at the full quality level to be employed in the final master tape; *Offline Editing* refers to the first part of a two-step process, working out the program's shape in a lower-quality format that will then be used as a template to re-create a high-quality master version later — much as film editors cut a workprint which then is used a guide to conform the camera original once the final creative decisions have been made. As terms, *online* and *offline* are not specific to non-linear editing. They derive, in fact, from the days when linear edit masters were made on 1" videotape — requiring very expensive studio time (thus 'online' because the final edit tied up the primary signal path of the studio). This led to editors using small format VCRs away from the studio ('off line') to do their creative work. Once the offline version was finished, the editor created a numerical Edit Decision List (EDL) matching time locations on the source tapes to time locations in the finished production. The EDL then served as a template to re-create the program online in the studio.

An *online* system is one that only works at the desired master-tape quality level. An *offline* system is one that only works below the desired master-tape quality level. An *offline/online* system is capable of both. The terms are relative: if you shoot in VHS and your goal is to make a VHS master, then a VHS edit system is an online system. If you shoot in Digibeta, and your goal is to make a Digibeta master, then a DV edit system is offline.

Given a liberal arts production pedagogy that emphasizes applied semiotics over technical processes, it's best to keep students working in online mode, as they can devote more energy to making meaning if they don't have to deal with the conform/convert step of re-creating their program in a higher-quality mode. This process is semi-automated in NLE systems, but it still takes time, not the least because the automated results are seldom perfect and generally require some tweaking. None of this involves brainwork in the medium, thus it is something of a waste of the student's limited lab time. It is also easier for beginners to develop edit decision skills in online mode, where they can work with the clearest possible copies of their images, and don't have to struggle with degraded images in an offline copy, wondering how the finished product will really look. On a simpler, but perhaps just as important note, beginners are also likely to have more fun working online, carry more enthusiasm into their work, and be less dogged by technical frustration.

The only reason I can imagine for an undergraduate production course in a liberal arts curriculum to employ an online/offline editing strategy is to save money, to try to stretch a very limited budget so that it can cover more users. Offline editing is a distinct compromise, and one I would suggest you try to avoid, but given financial realities the argument for compromise may be compelling.

Let's compare offline editing to an online editing system using DV video. DV footage requires 3.5 MB of drive space for each second of video. That's works out to approximately 4 minutes and 45 seconds of footage per gigabyte of storage. Professional editors develop the discipline to transfer only the footage they may need to their hard drives in order to conserve storage space. Students, on the other hand, don't yet *know* what they need and haven't had enough experience to develop efficient work habits. They need space to knock around in, try things out, make mistakes. This means that an online DV edit system requires something like the removable hard drive system I described above to work effectively, with each user having their own individual drive. Since the drives are probably too expensive to require students to buy them (~$175 including the mobile rack tray for 46GB IBM we use at Conn College as of 6/01), and will likely have little use to the students once they leave the class, the school is going to have to foot the bill. The more users that need to be accommodated at any one time, the more expensive it gets. And in practical terms, an online system based on removable drives is going to accommodate only a fixed number of users — determined by how many hard drives are available. With linear editing systems you could always squeeze out lab access for one more user — all she needed to do was bring her own tapes and wait for an open time slot. But once all the hard drives are issued, that's the end of that unless you have a system that has offline capability as well.

Typically, you would base the user limit for an online-only lab on the enrollment caps of your production courses, and only students currently enrolled in the courses would be able to use the facility. Such a restriction may already be in force anyway, based on the limited time available to use the edit systems in relation to course demands. In some cases however, you may want or need to open the video lab to students beyond the immediate production class — including students who have taken the class previously and want to keep working independently, or students from other departments who have good reasons to incorporate video into their work for other classes.

A strict offline/online NLE system would require each user to work on her project in an a low-resolution draft mode, with files stored an Orb disk or on a smaller, less expensive removable hard drive. The Orb disks function like videotape — anyone who has their own disk becomes a potential user of the system. The school is relieved of the expense of providing everyone a hard drive. However, the startup savings and openness come at the expense of making student work more difficult, adding more technological processes that may distract students from more important conceptual issues, and making maintenance of the lab somewhat more time intensive. So again, I would recommend a lab where everyone works offline only to those programs absolutely forced that direction by the numbers game.

The best of both worlds would be a lab that allowed a good number of users to work in online mode, yet also could admit extra users beyond that via offline/online capability. That is, you could set up a kind of tiered system where students enrolled in production class will work online with a school-issued drive, and anyone else would work offline using their own storage. The drawback to this idea is that a few single systems work well both as online and offline editors, and the equipment required for the tiered approach will cost a bit more than it would for the more limited online-only facility.

I will discuss offline/online editing in more detail shortly, but first we need to examine a few other aspects of the NLE production process.

Recording Format 1: Analog vs. Digital

The price of DV cameras has dropped to the point — entry level units between $600 - $700 — where it is foolish to consider any other format for new purchases. (I include Sony's Digital 8 under the DV umbrella here although it's not MiniDV per se.) DV cameras provide excellent integration with low-cost NLE systems via Firewire, but even if you chose not to edit in DV, it's by far the best choice for acquisition. Image quality is vastly superior to either S-VHS or Hi8, and moving up to a more 'professional' format — not that many liberal arts programs could afford to even consider it — brings very little extra bang for a great deal of extra buck. Just as importantly, time code, our disaster insurance policy and a necessary feature of our bargain-basement offline/online NLE systems, is integrated into the DV spec and present in every DV camera and deck.

But what if your school already has a stock of functioning VHS, SVHS, 8mm, Hi8 cameras and decks — or maybe even U-Matic field units and editors? The advantages of

DV are so great that you should only consider using existing analog gear if you are working with an *extremely* tight budget.

That noted, anything that captures a decent image can be used as a source for an online-only NLE system. However, if the equipment lacks timecode, and most analog stuff does, there's no way for students to recreate their projects easily after a hard drive crash.

There are VCRs that can add timecode to an already-recorded tape in both Hi8 and SVHS formats. Sony professional Hi8 editing decks (EVO-9850, EVO-9720, CVD-1000) can post-stripe RCTC. The JVC BR-S800U and SR-S365U can post-stripe JVC's proprietary CTL timecode. The Hi8 machines are all discontinued, and still fetch a good price on the used market, if you can find them. The SR-S365U is current, with a street price of about $1000. The cost of these machines may offset any savings you would garner by using older cameras. In addition, adding time-code requires another pass of the tape, taking more time in the lab. Hopefully, students could use this time to begin logging their footage. The method for adding timecode on some of these machines is not exactly straightforward — so another layer of complexity gets added to the system.

If you already have any of these models on-hand to use as edit system feeders, well and good. This will allow you to make use of an existing stock of timecode-less cameras as acquisition sources for a NLE. However, if no such decks are on-hand, given their price, as an alternative to acquiring timecode striping decks, you might consider buying DV machines and having students dub their footage to DV before editing it. The extra time involved in dubbing or post-striping is the same. Dubbing is easier. There is no significant generation loss involved (you're going to have to digitize the signal to get it into the NLE anyway). DV tape is more robust than Hi8. The main benefit, of course, is this lets you avoid making a significant new investment in a dying format and gets you moving toward the future. In fact, one of the lowest priced (about $1500 at discount) desktop DV VCRs available as I write is the JVC SR-VS20U, which includes both SVHS and miniDV mechanisms, and thus has self-contained dubbing capability.

In any case, before sinking any significant chunk of money into an effort to make existing analog gear usable with a NLE system, you should compare the cost, the methods required to work in such a hybrid environment, and the quality of the results with those of the lower-cost all-digital set-ups described in the appendix. Even if they are a bit more costly, they may be far more cost-effective.

Recording Format 2: The Many Flavors of 'Prosumer' Digital

Within the digital realm there are no less than four format variations — all using the same DV 4:1:1 codec but writing the signal to tape differently. The basic version is properly named just plain "DV" although it is usually called "miniDV" for clarification purposes. This is a consortium standard, and is the only variant produced by more than one manufacturer. Both Sony and Panasonic decided that since miniDV was a "consumer" format, they wanted to offer a upgraded version for their industrial and "prosumer" customers. Sony calls theirs DVCAM, Panasonic DVCPRO. These differ from straight DV in two ways: 1) a wider track pitch, which means the tape moves through the recorder faster — a standard mini size tape runs 60 mines recording DV, but only 42 minutes recording DVCAM or DVCPRO; 2) DV records 'unlocked" audio, while DVCAM and DVCPRO record 'locked' audio.

The wider track pitch is designed to make DVCAM and DVCPRO more robust under the repeated playback of editing, but I am aware of no complaints about DV in this regard. The difference between 'locked' and 'unlocked' audio is probably the least understood and most confusing aspect of digital video among its users. The best explanation I have found comes from Adam Wilt's website (www.adamwilt.com). Locked audio means the clock controlling the audio sample is exactly in sync with the clock controlling the video sample at all times. Wilt likens it to walking a dog with the dog tied rigidly to you waist, so you are exactly side-by-side at all times. In unlocked audio, audio and video samples are controlled by separate clocks. This does *not* mean the two can drift perceptibly out-of-sync, however. The clocks are linked, but not rigidly so. Wilt likens it to walking your dog on a leash. At any moment the dog may be a step behind you, or a step ahead of you, but ultimately you both get where you're going at the same time. While locked audio is the technically superior method, its advantages are negligible for most production purposes.

In short, all other things being equal, regular old DV is fine, and there is no compelling reason to go to DVCAM or DVCPRO. Of course, all other things are not equal. Comparable models are not always available in each variation. Device control interface and Firewire compatibility with the computer system especially need to be considered.

DV, DVCAM and DVCPRO all use the same tape stock. This is a certain amount of confusion about this since the manufacturers market tapes separately, as being distinctly for one format or another. This is done, first of all, so timing lengths on the tape can correspond to the track pitch — a tape labeled as a DV60 would be labeled a DVCAM42 — and also so the manufacturers can push their industrial/professional customers toward their higher tape grades. There are also three size of DV tape cartridge shells: mini, standard and extra large. The largest size is only used by Panasonic DVCPRO. Most (but not all) DV transports will only accept the mini-size tapes. Most (but not all) DVCAM transports will accept both mini size tapes and standard tapes.

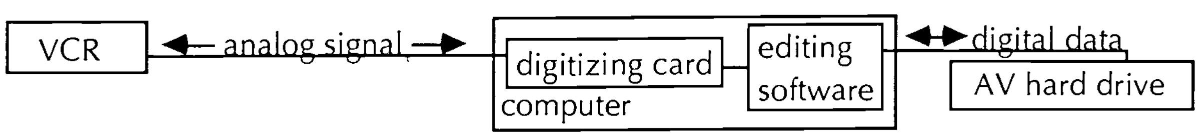
To make all of this even more confusing for the buyer, the formats are not compatible. DVCAM machines should play back DV, but most will not play DVCPRO. DVCPRO machines should play back DV, but may not play DVCAM. Sony DV machines should play back mini size DVCAM tapes, but DV decks from other manufacturers play DV only. All other bets are off, and some tech experts suggest that even the stated compatibilities are hit and miss. In any case — except for the Sony DSR-11, which records in either DV or DVCAM — digital decks only record in their native format.

No compatibilities are even possible with the fourth variation, Sony Digital 8. This records the same digital information as regular DV (unlocked audio), but on a Hi8 tape instead of a DV tape. Sony created this format to cover the lower end of their consumer camcorder line, in order to stretch the utility of their investment in Hi8 technology, and to provide a bridge to digital video for their old Hi8 customers (Digital 8 camcorders will play back standard Hi8 tapes, allowing a user with a library of home movies on Hi8 to still have a means to view them after switching to a new digital camera). At present, Digital 8 is only used in camcorders, and no tabletop models are available.

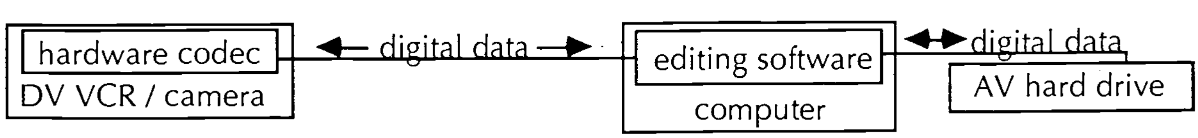
Since all the digital formats offer essentially the same image quality, choosing between them is largely a matter of price and features. However, the variations do not mix and match well, and you should choose carefully, as once you have selected a format you may be effectively stuck with it.

Compression Format (Codec)

There are currently four choices available in codecs for low-to-mid-priced computer video editing systems: Motion-JPEG (M-JPEG), MPEG-2 I-frame, and MPEG-2 IPP/IBP frame, and DV, which is significantly different from the other three. (The stand-alone non-linear 'editing appliances' from Applied Magic use yet another scheme, called wavelet compression.)



Traditional NLE systems use M-JPEG or MPEG-2 and operate as shown in the block diagram above. Whether the VCR is digital or not, analog signals are sent to a digitizing card installed inside the computer. This extra hardware converts the images and sounds into digital data which can be manipulated by the editing software and stored on the system's hard drive. All codecs *compress* the digital signal in some way so more data can fit on the hard drive, and so the computer can pass enough bits through its data buses fast enough to register the 29.97 frame-per-second video image ('codec' is an abbreviation of 'compression-decompression'). Compression always involves some loss of quality, though at lower amounts of compression (higher data rates) the effect is negligible. A key feature of M-JPEG and MPEG-2 codecs is that their compression level is *scalable*, meaning the user can increase adjust the file size / image quality trade-off to suit their particular needs.



DV is different. The signal is digitized and compressed by a hardware chip built-in to the camcorder or VCR. A digital data stream carries this information to the computer, which receives it through a standard Firewire port that may be used for other for other forms of data I/O as well. There is no special video hardware in the computer. For purposes of cost and simplicity, DV is not scalable. DV images come in a fixed size and fixed image quality, neither of which can be adjusted to trade off different qualities.

M-JPEG is the old standard, used by everything from Avid, Media 100 and other professional systems down to $200 PC digitizing cards. It is the least efficient codec in terms of perceived quality — and thus takes comparatively more storage space or bandwidth to achieve images that reach any given level of pleasing the eye. Professional M-JPEG systems are usually equipped with fast SCSI RAID storage systems. However, results good enough for teaching purposes can be achieved with somewhat higher compression rates and a single SCSI drive, or a fast UltraATA drive. M-JPEG compression is very flexible. Most M-JPEG cards can not only capture at varying compressions, but also at varying frame sizes, and scale images back up to full-screen on playback. For example, while a standard frame is 640x480 pixels — you can capture 320x240, but the system will still be able to display interpolated 640x480 while you work. The advantage to this is that highly compressed full-frame images often show large, blocky digitization artifacts that obscure the subject of the shot — while quarter-frame images at the same data rate may remain intelligible. The trade-off here is that movement in the blown-up quarter frame may appear to have just the slightest pixilated quality. Some editors may find this effect annoying, but most would likely find it preferable to blocky artifacts.

Thus, M-JPEG is useful at both the top and bottom ends of the quality scale. In high performance systems, its offers enough quality to serve in online editing of network-quality source material. In low-end systems, its flexibility makes it useful in offline editing where the goal is to stuff footage into the smallest file size that still shows enough of the image to be workable. In the middle though, for online work where 'very good' quality is enough, other codecs are more cost effective.

DV is a more efficient codec. By compressing chroma (color) data more highly — which seldom affects the picture in a visible way — DV takes a lower data rate than M-JPEG to produce comparable images. DV gets along perfectly well with low cost IDE drives. However, since DV compression is fixed, you can't increase the data rate to get an even higher quality image, nor can you increase compression to trade away image quality for savings in storage space. While DV images can't match the quality that can be produced by the very best M-JPEG systems, the quality is still *very good*, comparable to the output of a mid-level Avid or Media 100. Final quality of the edited image depends in part on the quality of the source material naturally, and high end M-JPEG systems won't show superiority to DV unless very high end cameras are being used for acquisition.

Using DV as an editing format can lower the cost of a system if DV is also used as the acquisition format. Since the camera or deck performs the AD and DA conversions — as long as an external DV machine is connected to the system, the computer does not need to be equipped with a digitizing card. If the footage is shot in DV, the system can edit it without any degradation at all — what goes in is exactly what comes out. If DV source material is edited using any of the other codecs, no matter what quality level selected, some degradation of the image occurs as the image is decompressed, converted to analog, redigitized and recompressed in the new format. This also requires an additional digitizing board, tending to make the system more expensive.

High end M-JPEG systems are designed for the professional world where Digital Beta or Beta SP are the primary acquisition formats. If DV is your source material, there is no reason to use M-JPEG or MPEG-2 codecs in editing other than for the scalability they offer. Scalability, in turn, is only necessary to save storage space, especially in offline/online editing.

As such, high quality M-JPEG systems have no real benefit to a liberal arts production education program, while adding greatly to the budget. To put this another way: no matter what anyone tells you, you do not need an Avid!!

With the cost of large UltraATA hard drives continuing to drop, it becomes easier to acquire enough storage to make scalability a non-issue. Thus, M-JPEG hardware may now only make sense at the lowest budget level for a teaching system.

MPEG-2 edit systems are the newest entrants to the field. MPEG-2 is designed primarily as a distribution format, where it can achieve a great deal of efficiency. Whereas DV or M-JPEG compresses each frame of video in its entirety, MPEG-2 can compare two consecutive frames and make a record of one by marking only the differences it has to its neighbor, which takes up much less bandwidth. This is fine for program playback, where we watch the images in a steady stream, but poses a problem for editing, where we may need to shuffle back and forth starting at any particular frame. MPEG-2 allows a variety of schemes for *inter*-frame compression (in contrast DV and M-JPEG have only *intra*-frame compression — each frame compressed only in terms of itself), but in all of these a complete frame must be recorded every so often. These are called 'I' frames. It is possible to create an MPEG-2 system that records nothing but I-frames, giving up inter-frame compression, and this is what most MPEG-2 edit systems employ.

MPEG-2 I-frame is a more complex and efficient codec than M-JPEG and produces better results at a given data rate and frame size. However, MPEG-2 only operates in full frame 720x480 mode, which limits MPEG-2 I-frame's usefulness for low-res offline work to some extent. Low to mid-priced MPEG-2 I-Frame systems have a maximum data rate of 25Mb/sec — that's mega*bits*, and since there are about 8 megabits in a megabyte, this is very close to DV 3.5MB sec.

MPEG-2 IPP and IBP are inter-frame compression methods, so they are significantly more efficient than any of the other compression methods. Pinnacle Systems offers two video editing systems using these compression formats. The DC1000 is the base model, with S-Video in/out. The DC2000 adds component video in/out. (The cards are also available as the DVD1000/DVD2000 including full-featured DVD authoring software.) A DV in/out daughtercard is available — it transcodes the DV signal into MPEG-2, rather than using native DV as the editing format. Pinnacle employs a proprietary technology to address the problems of using of inter-frame compression in an editing environment. This system enables outstanding capabilities in offline/online editing, as discussed below in the review of the DC1000.

In conclusion: overall DV is the most useful editing codec in terms of achieving high quality at low price and integrating the editing system with acquisition equipment.

However, DV is not scalable. Although dropping drive prices make online-only labs more practical, a teaching lab on a tight budget may still demand scalability to deal with storage issues. Either M-JPEG or MPEG-2 can serve as a low-res capture format. The best low-data-rate results come from Pinnacle's IPP/IBP MPEG-2 systems, but the savings in storage costs is somewhat offset by the higher cost of the digitizing hardware. M-JPEG and MPEG-2 I-frame are similar in low data-rate usability, though M-JPEG is generally a bit more flexible in this regard, and can be found in lower-cost packages.

It's possible that for some teaching labs, a single codec may not fill the bill, and the best design may be to employ systems that offer both DV and one of the scalable codecs as well. Here the choice should be made not on the basis of the codec employed per se, but on the available features, cost, reliability etc. of any specific product.

Online-Only Variations

Remember that 'online' is a relative term, and means only that the work is edited at the same quality level as the finished product. All 'online' qualities then are not equal. Assuming we want to build an online-only edit lab for sake of simplicity, we still have some options of what that online quality level will be, which can change the budget for the system as well. The easiest way to build a stable, functional good quality online-only facility is to stay within the DV format. This means that you spare the expense of a capture card, since the images and sounds are digitized when the tape is recorded. However, although DV is an efficient codec, since it's not scalable, building a DV online-only lab commits you to providing a certain amount of storage space to each user, and the cost the all those individual good-sized drives begins to mount up. If you use MJPEG editing, however, you can lower the data rate of the capture, which means that each user will need less space — thus a smaller and less costly hard drive will suffice. Since this is online editing, we won't be compressing the data into the sort of draft-quality image that will easily go onto an Orb disk. We looking for a manageable image-quality/space-saving trade-off. I would say MJPEG images remain presentable at about half the data rate used by DV — somewhere in the VHS vicinity. Whether this compromise will save enough money to make it worthwhile will depend on other aspects of the production facility, which I will discuss in more detail in the section on specific systems. One quick note here though, the latest generation of editing software — Premiere 6 and Final Cut 2 — have abandoned support for low cost MJPEG capture cards, and thus an MJPEG online-only system will need older versions of the software and will have a more limited upgrade path. (Not that this matters necessarily, as long as the system works.)

One desirable feature of an online-only system that is not strictly required and can be dispensed with to save money is timecode capability. Timecode is the essential backbone in an offline/online setup, but it has a valuable function in an online-only environment since it may enable a user to recover from a hard-disk disaster without completely redoing all the work. A NLE does not store information in the same way a word processor does — a Microsoft Word file is self-contained, it contains all the characters in the document. Non-linear editors employ two sets of files. The actual pictures and sounds captured are placed in *media* files. These are what take up all the disk space. Once captured, the media files themselves are not actually edited. The editing software generates what are called *program* or *project* files based on the decisions the editor has made in the timeline or storyboard windows. These amount to a set of complex instructions about how the computer should play back the media files — a detailed EDL in computer language. Thus, for example, the software makes a cut by telling the computer to playback file X from point A to point B, followed immediately by playback of file Z from point D to point F. This method gives NLE one of its major advantages over linear work, editing is *non-destructive* — since the media files aren't erased, you can always change your mind about any aspect of the program by making changes in the timeline. This also allows editors to make practical backups of their work, since program files are small enough to fit on a floppy.

This point cannot be emphasized enough: Students must keep backups of their program files on a separate disk for safety's sake, not just leave the only copy of their program file on the computer or on their removable disk.

Low cost NLE software (iMovie, MovieShaker, etc.) — and stand-alone editing appliances don't stamp media files with source timecode, don't record source timecode in the program file, and don't have a batch recapture function. Instead, they link the instructions in the program file to the content in the media files based on the media files' records in the directory of the hard disk. This means that those links are lost if the media file is accidentally deleted on the hard disk. If a hard disk suffers an unrecoverable crash, the project is lost even if the program file has been backed-up elsewhere. (A side note: you'll want a copy of Norton Utilities, Diskwarrior, Tech Tool, or some other recovery software available in the lab.)

On the other hand, if source timecode is captured and stamped by the NLE, a project remains recoverable independent of whatever happens to its media files as long as the program file survives. Let's say a hard drive dies as a student is about to complete a semester long senior project. Without timecode, she has no choice but to start over from the beginning. With timecode, as long as the program file remains, she need only execute a 'batch recapture' function. The system will then instruct her to insert each source tape in order, and automatically recapture all the missing media using the timecode on the tape as a reference. In practice, the recapture process is a bit more complicated. First, there's generally a measure of user error involved, as the editor may have forgotten to enter the proper name of the source tape when making the initial capture. Thus, the system will capture it's defined timecode region from the wrong tape, inserting a completely different shot into the program. Second, depending on the sophistication of the device control interface between the computer and the VCR, timecode stamps may not be precisely consistent, with some clips being a frame or two off — the length of the recaptured clip won't change but it might start and end a frame earlier than the original clip, or start and end a frame later. For some cuts, these deviation may make an important difference, and the editor will need to carefully review the recompiled program. Nevertheless, recapturing is a far less time-consuming process than redoing each edit point from scratch.

Fortunately, the newer generations of hard drives are pretty reliable. In two years at Connecticut College we have yet to face a situation where all the media files for a project have been lost. We have encountered many situations where students have accidentally deleted a file or two, but in these cases the students have generally just re-logged the files since they're familiar with that process. Thus, while timecode does save time in this sort of situation, the extra time to work around its absence is hardly a crisis level problem.

Running a video lab without timecode is like driving without insurance — the odds are you'll get away with it over any limited period, but you'll feel more comfortable if you have coverage in case of accident. And in the long run, accidents may happen.

Choosing a Path for the Tight Budget

I have laid out two paths of compromise to lower the cost of an NLE setup.

Forced Offline/Online

Benefits: Finished projects of high quality. Disaster insurance. Unlimited user pool.

Drawbacks: Less pleasant working method. More technology intensive for student and faculty. Lower reliability.

Budget Online-only

Benefits: Ease of use. General stability.

Drawbacks: Reduced image quality or project length restriction. No disaster recovery. Limited user pool.

I think the basis for choosing one of these strategies over the other is fairly evident in these bullet points.

The forced offline/online approach is better suited to a program where 1) the students are more 'into' production work — where they have the desire to wind up with better looking, more ambitious projects, the drive to learn and execute the extra steps involved, and the dedication to work through a few more technical problems; 2) the faculty or staff have the know-how and time to deal with these issues.

The budget online-only approach is better suited where 1) the course is an elective rather than part of a core curriculum, students are more casual about moviemaking and excited just to be able to make any kind of production; 2) the faculty or staff have less time and technical background.

How Offline/Online Editing Works

Earlier, I gave a brief outline of a forced offline/online editing process, in which all users work in low-resolution offline mode, but then can recreate their project in a high-quality online mode after their Edit Decision List is finalized. Edit environments capable of both steps of the process must meet these requirements:

1. Each station should be equipped with a large fixed hard drive, fast enough to deliver online quality and large enough to hold a finished project of the required length at

that quality. However, students do not edit onto this drive, since it is not large enough to hold everyone's work, and must be kept free to conform finished projects.

2. The system must also include hardware for some scalable codec (e.g. M-JPEG or MPEG-2) — it can't be straight DV only. Students will capture their files with this codec at the lowest usable resolution to minimize file size and maximize storage space.

3. The source tapes must be time-coded, and the VCR feeding the footage into the NLE must deliver time-code to the system.

4. The editing software must stamp captured files with time code information, force users to label each clip with the name of the source tape, and offer a 'batch recapture' function.

5. The system must provide some option for users to store their low-res media files, either via small removable hard drives, Orb drives, or a partition on a second fixed drive for a single-station facility.

When the student finishes the low-res version, it becomes a template for the re-assembly of a high-res version. Unlike the disaster recovery process outlined in the 'Online-only Variations' section above, the user does not need to recapture all the defined clips in the editing bin to enable further manipulation of the EDL. Instead, the program is set to recapture only the footage actually used in the timeline, with short 'handles' extending beyond the in and out points used in the timeline, to allow for minor adjustments. The hi-res clips created in this step are captured to the large fixed drive in the computer. They will stay there only temporarily, in a single session long enough for the student to make any minor adjustments to the program and output it to tape. (The same sort of minor problems with mislabeled clips and timecode stamp inaccuracies I noted with disaster recovery will probably lead to some tweaking here as well.) Once the adjustments are made, and the project is recorded back to tape, the hard drive is cleared for the next user.

If the final output is to DV, the student can make additional 'first generation' copies of the program from the tape master. If the output is to an analog tape, in order to make additional highest-quality copies, the student would need to have saved the program file, and held onto all the original source tapes. If so, the student could then reload the program file into the editing software, run the batch recapture routine again and reassemble the hi-res version onto the fixed hard drive again, going through the same sort of adjustments as before if necessary. This is not a simple or quick process, but it is possible. (In a no-timecode online-only format, once the media files are erased from the hard drive the project is essentially gone.) A student would follow a similar procedure when deciding, after a time, to revise a previous project — perhaps polishing a class assignment to make it serve as a portfolio piece.

Where any timecode-based system is used, students should be strongly encouraged not only to save back-ups of their program files, but to keep all their original footage intact. Beginning students especially think nothing of recording over their camera original once the assignment it was taken for has been turned in. They want to save money on videotape. They see no value in their footage because they don't consider the possibility of revising anything — for a portfolio, a festival submission, a chance at a better grade — and don't consider the possibility that their master edited copy may get lost or accidentally damaged. Videotape is cheap. It costs little for students to preserve their materials, which are ultimately irreplaceable and may well turn out to have value to them at a future date.

While an Orb drive is in many ways the easiest option for media file storage in a low budget offline/online system, anyone creating such a facility ought to consider having some users (more advanced students), or even *all* users capture to a removable hard drive — albeit smaller and less expensive models than you would use with an online-only system. A search of Internet vendors often yields close-out specials on discontinued drive models. I've seen IBM 20GB/5400RPM mechanisms for just over $80. 20GB is actually large enough for online work with modest projects, and figuring another $8-$10 for the mobile rack tray leaves us at under $100 per user — perhaps within the range of the school budget, perhaps within the range students could be asked to purchase — certainly a better value than a less reliable 2GB cartridge at $30. Since students aren't likely to have any use for a mobile-racked hard drive outside of the video class, it might seem unfair to ask them to buy them. However, each new semester would bring a new group of students who need drives, creating a market for the former students to sell theirs and recoup their investment. Or the school could achieve the same end of having students fund the drive purchases over time by charging a rental fee, which for the smaller hard drives could fall in the same $30 range as the purchase of an Orb disk, yielding improved reliability, much more capacity, and the ability to work in online mode for shorter projects.

One advantage to offline editing on smaller hard drives is that for most projects the files need not be too compressed. At the kind of compression needed to stuff files into an Orb disk, images are definitely workprint-only quality. They're legible enough to make editing choices, but not the kind of thing a student would likely care to show family or friends. On the other hand, a small hard drive would allow a higher data rate for similar projects, and while the offline image would clearly be inferior to the source, it may acceptable enough for casual showing. Especially for initial projects of limited ambition or success, a student might want to have something to show but not care if it was the finest quality. Using a kind of intermediate compression in the offline working mode, students can have the choice of going the extra step to produce a high-quality master, or not. Each choice not to create an online version is less lab-time and less hassle for student and instructor.

Even in offline mode, an Orb disk limits the size of project students can work on comfortably. My experiments place the lowest usable data rate for M-JPEG low-res at ~750KB/sec (using 320x240 frame size) which would work out to 48 minutes assuming you could fill the disk. This is fine for editing a couple projects of 4 minutes or less, but is obviously not enough space to cut a 20 or 30 minute documentary. Depending on the kinds of pieces you want students to make, whether you offer advanced classes or just an introduction, the Orb may or may not be adequate.

Whatever system you implement will need to fit your school's student culture, your capability to administer a supply of drives (you would at least need to locate a source of inexpensive drives for students to purchase — they aren't the sort of thing one can pick up at the local computer store), the possibility of getting an additional outlay of funds to begin with if you can recoup them later via rental/use fees.

Device Control

There is a great deal of confusion about what is meant by calling a VCR or a device control interface 'frame accurate.' The term derives from linear editing, and refers to a system's ability to place a new piece of video onto a tape containing existing footage, so that the exact desired first frame of new picture replaces exactly the first no-longer-desired frame on the tape. This is not that important an issue in nonlinear editing, where all the edits are typically done inside the computer and sent out to tape as a whole from beginning to end. Thus, very few digital VCRs are 'frame accurate' in this sense — generally professional models outside of the price range of most schools. However, there is a second form of accuracy involved — whether a set of instructions to capture a given range of timecode from a tape actually captures exactly the same footage every time, or whether the window might move a bit forward or backward from time to time. This is what I call timecode stamp accuracy, and it's very difficult to get information about it. Most 'expert sources' simply note 'Firewire device control is not frame-accurate' and leave it at that. But what does that mean? What kind of inaccuracies do reasonably-priced DV VCRs produce?

Either kind of accuracy is in part dependent on the device control protocol employed, the communication system that carries instructions back and forth between the VCR and the computer. There are three basic systems in use in DV VCRs. The most common is Firewire device control, which uses the same data-stream carrying the digital images and sounds. Every DV VCR or camcorder with a Firewire port can be controlled via Firewire device control. The other two forms are the common computer serial communications standards, RS-422 or RS-232. To use serial control, the VCR must be equipped with the proper port, and these are usually found only on more expensive models.

Most low cost analogs VCRs have no device control at all. Sony VCRs and camcorders usually include a rudimentary control interface called 'Control -L" or 'LANC'. Sony also made Hi8 decks for use with computer systems using yet another control interface, VISCA. In lower-end analog gear, the presence of a control port does not necessarily mean the device is equipped with timecode. To repeat, offline/online editing depends on timecode, and device control is the means to get timecode from the VCR into the computer.

Device control accuracy is not determined solely by the protocol employed. It may also be affected by the implementation of that protocol in specific VCR designs, by mechanical features of the VCR, and by the quality of the device control software. Final

Cut includes built-in device control for Firewire, RS-422, and RS-232. Premiere 5.1c and Premiere 6 include Firewire device control, and serial control is available in third-party plug-ins from Pipeline Digital and Diaquest.

The rule of thumb advanced throughout the collected video expertise to be found on the Internet is that the only guarantee of complete accuracy is to use an RS-422 device with approved software control. RS-232 is considered a notch down but still reasonable accurate, and Firewire a little shaky. Unfortunately it's hard to find specific information about what these general characterizations mean in practice. Again, none of this matters as long as you're just capturing footage, editing in the computer and laying a completed program back to tape in an uninterrupted process. But it matters a good deal for any offline/online process, or any effort to recover an edited program following a drive crash or accidental deletion of files. So I decided to make some limited tests

In the summer of 2000 I had the opportunity to review two reasonably priced non-linear systems based on Windows PCs. In contrast to Macs, Windows computers do not have Firewire ports built in, nor is Firewire software part of the operating software. The two products I reviewed, the Matrox RT2000 and the Pinnacle DV-1000 both included Firewire ports, and supplied Firewire drivers and Firewire device control software to go along with them. In both cases the timecode stamping reliability was horribly inadequate — with almost no correlation between the time-code numbers recorded and the footage captured from one pass to the next. I concluded that neither product was suitable for offline/online work without the use of a serial controlled VCR and a serial device control plug-in. I was using a Sony DSR-20 VCR for these tests, and as it is equipped with an RS-232 port, I acquired the RS-232 capable ProVTR device control plug-in from Pipeline and re-checked the accuracy of the systems with this addition. (ProVTR handles *all* device control varieties other than Firewire: RS-422, RS-232, VISCA and LANC.) This solved the problem completely. That was a year ago, newer drivers have been issued since, and Firewire control on the PC side may well have improved since then, though I have no knowledge of this either way. I remain suspicious of Firewire implementations on Windows machines.

Moving to the Mac, I tested Firewire device control in a variety of hardware/software combinations, and also RS-232 device control with FCP 2.0 a Targa 2000 capture card, and a Sony DSR-20 VCR. Contrary to expectations, Firewire and RS-232 showed similar accuracy in simulated offline/online recaptures. If anything, Firewire control was a little better.

Both Premiere 6 and Final Cut 2.0 yielded absolutely accurate re-captures time after time with Firewire device control and DV video input via Firewire whether I used a Sony DSR-20, DSR-11, or a JVC SR-VS20U as a source deck. With earlier versions of the software, or with analog captures made through a Targa 2000 board, Firewire device control yielded a modest percentage of very small inaccuracies on a regular basis. I made my tests by capturing ten clips from a tape with 'burned-in' timecode, trimming them into a timeline sequence, deleting the original media, recapturing, and then noting whether the burned-in numbers were the same as before at the edit points. I did this enough times with each setup to establish a pattern. Where deviations occurred, typically two or three clips in a ten clip batch capture sequence would show slippage. However, the change was never more than a single frame, and all of the clips showing errors in the same batch capture sequence would be off in the same direction. (For comparison, successive RS-232 controlled analog captures from the DSR-20 had three or four deviations, though again of only one frame, and all in the same direction.)

Note, though, that re-capture inaccuracies may be compounded by the errors in digitizing the original clips. That is, if your original capture of a clip is one frame off, and you edit the sequence on that basis, any time a redigitize operation recaptures that clip correctly, the new media will be one frame off in relation to the edit points you set for that clip in the timeline.

The question, though, is how many edits in a student production will be thrown off aesthetically by moving a single frame one way or another? In my own work I run across edits that need to be truly frame accurate often enough, but I have also observed that beginning students are usually unable to tell whether edit points have shifted a frame or two. In other words, to even an experienced observer occasional one frame deviations will only be an issue every once in a while, and the average observer is unlikely to notice the difference. Again, this problem is only an issue in offline/online situations. Given that a school is most likely to adopt this strategy as a cost-saving compromise in the first place, it probably doesn't make sense to spend the extra funds necessary to generate superior offline/online accuracy by upgrading to RS-422 feeder decks — the money would probably be better spent enabling more users to work online to begin with. If those funds are not available, those one-frame shifts are probably best seen as reasonable trade-off, inconvenient but hardly debilitating.

Real Time?

I noted earlier that one of the primary values of NLE is it allows students to work easily with dissolves and fades. However the pedagogical benefit of having A/B transitions that can be easily made and revised is offset somewhat by the necessity of rendering the effects on most NLE systems. Since students only have a limited time to spend in the lab, we not only want to minimize the time they need to spend figuring out the software, we also want to minimize the time they spend waiting for the software to work. Every moment the students sits watching the slow movement of the progress bar on a render is learning time wasted. Short (one or two second) fades or dissolves don't eat up *that* much time — usually rendering in 30 seconds to 2 minutes. But anything involving longer overlaps, or keying of titles especially, really drags out the working process. A "real time" edit system — one that plays back basic transitions and title overlays immediately, without rendering — is a great benefit to the college lab because it allows students to spend more time focusing on the creative tasks at hand.

Unfortunately, until recently, real time editing has been available only on more expensive professional systems — high-end Avids, Media 100s etc. Starting in mid-2000, however, several products have been introduced that promise to bring real time into the range of more modest budgets. These particular systems bring other pluses and minuses along with real time — depending on how you teach and how your students work, real time may be important enough to sacrifice other desired features.... or not. I will discuss a couple of lower-end real time products in the section on specific systems below.

Standardization Across Campus

For years I have approached the idea of how a proper video facility for production education should be equipped from a narrow perspective, defined only by an attempt to meet the needs of scheduled classes in the most budget-efficient manner. However, since arriving at my current teaching job, I've realized a broader perspective may be in order. Video equipment is becoming inexpensive enough, and an interest in media has penetrated our cultural generally enough, that editing stations may start popping up on campus in places beyond the labs designated for production classes offered by Film/Media/Communication departments. At Connecticut College, there are no less than three other computer-based video editing systems on campus, besides our Film Studies lab. The Dance department has a system to edit dance videos. As a result of a grant, a new system has been purchased solely to aid faculty in adding media content as part of course development. Finally, the library has a system open to use by anyone in the campus community, and is looking to add two more.

The problem with all this, of course, is that the left hand never knows what the right hand is doing, and as things stand now each corner of our little video world is fitted with different hardware and software. Each of us creating these separate spaces has tried to outfit them according to the demand of the specific purposes for which they will be used. In doing this, we have imagined our populations of users to be distinct groups — The Film Students, The Dance Students, The Faculty Assistants, The Non-Media Students. What we seem to have forgotten here is that we don't have *that* many students, and as a good liberal arts school our students move around the college rather than staying in one little corner. A student who has some generic interest in video may wind up wanting to work in any or all of these areas — take a film production class, take a dance and technology class, make a video for a philosophy assignment, get a part-time job helping a professor make a video to supplement a lecture. However, our present designs sabotage this mobility by demanding that the student learn a new technology at each stop.

Now, we may observe that in and of itself, adapting from one NLE to another is not that difficult a task. If you know how to use Premiere, you can figure out how to use a Final Cut Pro pretty quickly, and vice versa. However, again, we need to recognize the time demands on our students. The effort of learning yet another new software package may be just the extra difficulty that keeps a student from taking up a project. We also need to remember that in terms of our educational goals, time spent by students learning new software and hardware is time wasted — time not spent on the substantive aspects of the project itself. Once a student has learned the technology of editing in one corner of the college, it's to no one's benefit if she has to relearn it, simply to move to another area of content.

Thus, at any campus with multiple video editing facilities, standardization, or lack thereof, becomes an ease of use issue. After just one semester of dealing with some of the problems that can result from having different technologies in every corner, I became a champion of some degree of standardization, and concluded that it would be in our students and our program's best interest to make reasonable sacrifices from my own ideal preferences if we could everybody on the same page by doing so. For example, I may prefer Final Cut Pro to Premiere, but I find Premiere serviceable enough, so if the other units with video on campus all want to use Premiere I might go along, since whatever unique benefits FCP may offer the Film students might be less valuable overall than the ability to enable more student work by removing the extra software training barrier.

Choosing Computer Hardware

Mac or PC? In the last analysis, you should chose the computer platform you will have the least trouble with, and will enable you to get problems resolved more quickly. This depends in part on your knowledge and experience, and the resources available from your campus computing services department. You may be a die-hard Mac user, but unless you want to do *all* your own tech support, if you wind up at one of the many campuses where the techs don't do anything *but* Windows, it may be wise to go with the flow. On the other hand, if the computing staff doesn't have any video-specific expertise (and most don't) they are unlikely to be able to help you with your video-specific problems. Left to your often resources, as you may well be, you are certainly better off with the platform *you* know the best. Many more video products are offered for the PC, and prices are generally lower. However, Macs are far easier to set up and keep running than PCs — there are no IRQ conflicts, and Quicktime is simply a much more stable file architecture than AVI. The Mac OS also has better integration of DV in general, and DV device control in particular, which makes it more hospitable to affordable systems that fit our multi-user criteria.

NLE systems can be acquired two ways. You can buy everything separately and put it together yourself — the computer, the boards, the drives, the software... Or you can buy a "turnkey" system from a vendor who does all that stuff for you. A turnkey system costs more, of course, but in general it is well worth the money, especially if you chose a PC platform. Assembling a NLE system from component parts on a Mac is a challenge, on a PC it is like entering the fifth circle of hell. In building a do-it-yourself Mac-based NLE there is often struggle, but usually success at the end. In building a do-it-yourself PC-based NLE, too often the torture of shifting IRQ conflicts, flaky drivers, and corrupted Registry entries never resolves itself into a usable result. PC computers cobble together a variety of different components from different sources — motherboard, controller chip, BIOS, display card, I/O card. Any one of these, or any combination of them, can cause a conflict with NLE hardware or software. PC NLE systems generally work properly only within a very narrow range of the larger pool of PC parts that work together otherwise. This is why so many PC NLE users have trouble with their systems, and why it is routine for Web forums devoted to PC NLEs to ask each participant to create a profile that includes a list of the components in their PC, which then gets attached as a sig to all the user's posts, so when either success or failure is reported the other readers of the board don't have to keep asking, "what sound card are you using with that?" A knowledgeable, experienced technician can usually find a way through the pitfalls. PC NLE systems should be configured by just such an expert — a PC *Vide*o expert. No, the networking tech from college computer services is not going to be able to do it. Find a turnkey vendor with a good reputation and a good guarantee. If this is unavailable, at the least you should call the maker of the NLE hardware and find out what computers *they* use to develop, test and show their products at trade shows, and then purchase *exactly* that configuration.

Anyone who forsakes the turnkey route for DIY needs to observe some general caveats. Video systems demand a lot out of a computer, demands that often create problems where other computer uses do not. Manufacturers may promote the idea that video editing is simply an optional feature for your home computer — the computer is the primary purchase and the video editing package is the add-on. However, the truth is more the opposite. The video elements are the picky parts of the combo, so anyone who wants computer video editing to work properly must view the editing package as the primary purchase and the computer as a secondary component. *Choose the editing package first, then buy the host computer to fit.*

Be aware that computer standards are anything but absolute. You might think that one ATA66 hard drive can substitute for another, any PCI SCSI adapter work fine in any PCI bus, and any Firewire VCR connect properly to a Firewire enabled computer. Not so. I have already noted how resistance to cable interference varies among brands of hard drives. In the case of both PCI and Firewire 'standards', exact implementation varies enough from different makers of controller chips and different revisions of chips that compatibility is anything but guaranteed. Advertising claims of "works with any xxx system" need to be taken with a grain of salt. You can't assume that tab "A" and slot "A" actually do fit together, just because they're supposed to adhere to the same spec or standard. This means you can't just pick out parts for an editing system based on catalog descriptions and expect the assemblage to work properly. You need to experiment, or find out what specific combinations have worked for others. Try to arrange for trial use of any combination of products you plan to configure into a system, before committing your precious budget to purchasing the stuff and setting students down in front of it to work.

The low-cost systems listed below are all do-it-yourself projects to one degree or another, since they involve either used parts or non-standard configurations. As such, they are Mac based. They do not necessarily have PC analogues, and even if they did, troubleshooting functional component lists would be extremely time-consuming. As you'll see if you read all the details below, it took me long enough to troubleshoot the

Mac systems. My hope is that my experiments have yielded results other instructors might reproduce without going through all the trial and error. However, since even a slight change in hardware or software can throw a previously stable NLE system into conniptions, anyone trying to cook up a system based on these recipes should follow them as closely as possible without substitution...

Choosing Software

Assuming that high-end professional editing packages are not only out of our price range, but too complicated and difficult besides, we are left with a choice between very simple but limited software tools such as iMovie, and more full-featured mid-level programs such as Premiere or Final Cut Pro.

Apple and Sony provide simple software tools for DV editing, bundled with their computers. Macs come with iMovie and VAIO PCs come with the similar Moviegoers. These programs are easy to learn, and iMovie especially has a reputation for being very appealing to beginners. As a more experienced maker, I find their capabilities too limiting, making certain things I want students to know difficult or impossible to do. That said, if these were the only tools available for teaching, beginning students could certainly learn a great deal about visual communication using them. I've spoken or corresponded with instructors who would like to set up an a modest introductory production course within a high school or college English curriculum, though they have no real background in film production, only in film analysis. A technology tyro attempting to shepherd young videomakers is likely to be for an ordeal no matter what equipment is chosen, but the goal of providing students a valuable learning opportunity they wouldn't have otherwise is laudable. In this sort of situation, something like iMovie just might work.

One factor that limits iMovie or MovieShaker is that they only work with DV. This means they require a decent amount of storage for each individual user. This effectively restricts their usefulness to situations where a very small class uses a single workstation, or situations where financial and technical resources can provide a removable drive system (and its probably safe to say that most non-expert faculty looking to set up a little production class on the side aren't going to be graced with that kind of budget). For a simplified system that allows you to select a quality level with a scalable codec, you might look at the stand-alone editing appliances from Applied Magic. Editing appliances have proprietary editing software etched into permanent memory, and that's all these systems do — they have no other computer functions besides video editing. This makes them more stable than software programs loaded in temporary memory on top a general purpose computer operating system. Editing appliances are a small step above iMovie in complexity and a significant step up in functionality. Still very easy to learn, they offer enough additional features that they may satisfy the needs of intermediate students as well as beginners. Personally, I find appliances to be just a bit too limiting in areas of technique I find important, but they promise to be much less of a hassle to students and instructors than a computer based system, and this may well be worth the trade-off. Editing appliances ought to appeal to the sort college faculty person who has been hired to teach a mixture of theory and production, but who has devoted most of their study to scholarship, gaining a minimal hands-on background, perhaps by being assigned to TA production classes while in grad school. I will say a bit more these systems in the specific systems section below.

Moving into the area where I think all the important creative needs of a liberal arts production curriculum will be satisfied, we have what video folks refer to as 'software editing systems' meaning the software turns your native computer hardware into an editing machine, or will work with a range of third party capture hardware. This distinguishes these products from more costly high-end professional systems (Avid MC, Media100, Discreet Edit) that use software tied to a specific piece of proprietary hardware.

The 'software editing' market has long been dominated by Adobe Premiere, on both Mac and PC platforms. Many people, including me, have had unpleasant experiences with edit systems using older versions of Premiere. The interface was easy to learn and usable enough, but getting anything done took forever since *everything* had to be rendered before it could be viewed. Watching even a sequence of straight cuts required a Make Movie command. The problem here was not so much Premiere itself, but rather that computer systems were not powerful enough to allow a software-based system like Premiere to work very fast. In contrast, the speed of working on an Avid or Media 100 was not a function of the software itself, or the hardware itself, but the very specific integration between them. There was one definite software problem in Premiere up through version 4, though. It was based on a true 30fps time base — assuming the edited video would wind up as a multimedia file designed for viewing on a computer — and never quite handled 29.97fps NTSC properly — the result being that audio and video would often go out of sync in Premiere after a period of time.

Premiere 5 changed that, adopting a true 29.97fps time base. By 1999 computer power had caught up to the concept of software-based editing, and Premier systems could play straight cut sequences out of the timeline without rendering. Thus, with the proper modestly priced hardware, Premiere 5 was/is a very usable editing tool. Premiere 5 was designed with MJPEG systems in mind, though DV support was added in version 5.1c. Adobe introduced Premiere 6 in early 2001, and the new version has been redesigned especially to integrate with DV (although support for many M-JPEG capture cards has been abandoned in the process). Most Premiere systems still need to render transitions and effects, but a special PC version of Premiere supports the lower-priced real-time boards available for that platform. While Premiere lacks many of the refinements of higher-end editing software in terms of customizing the interface and enabling shortcuts for common tasks, I would still consider it a full-featured editor in the sense that there is very little you *can't* do with it, though the process may not be particularly elegant. While Premiere enables an appropriate level of conceptual complexity, its basic functions are still quite easy to learn. I've seen many students with no prior editing experience of any kind pick up the basics in Premiere almost immediately (as long as they had a minimal computer literacy to begin with).

Premiere has the advantage of being an almost universal standard, albeit one not granted much esteem. Some of your students will likely already have been exposed to it. This can only make the time you spend in class going over editing procedures more productive. Your college may also own an available Premiere license already, since many schools have purchased Adobe software in multi-user bundles.

The most significant competitors to Premiere are Final Cut Pro (FCP) which is published by Apple and available only for the Mac, Avid Xpress DV, a 'low-end' incarnation of the industry-standard Avid interface available only for the PC, Media100 Cinestream for both PC and Mac, and Ulead Media Suite, available only for the PC.

I have never seen Media Suite, nor heard of a college using it, though it has received some positive user comments on the Web. I have also not had the opportunity to work with DV Xpress specifically, though I have worked with other Avid products that share many features with the Xpress DV. Xpress DV is the most expensive of the three products, though not outrageously priced at $1500. It works with DV only, making it suitable only for a higher-quality online-only system. There are no real-time options available for DV Xpress at this time, and I would guess Avid would keep it that way to protect their more expensive real-time products. The Avid interface also has a decidedly more imposing learning curve than either Premiere or Final Cut, though it is certainly more sophisticated and powerful than Premiere, at least. The primary benefits of the Xpress DV in comparison to Final Cut Pro come in its unique project management tools, and the fact that it serves to train users in the Avid interface, which dominates the world of professional video editing. (A useful comparison of XpressDV and FCP is available online at [http://www.2-pop.com/library/articles/2000-07-14.html, http://www.2-pop.com/library/articles/2000-07-24.html, http://www.2-pop.com/library/articles/2000-08-11.html] Thus, I think the Xpress is best suited to a pre-professional program that seeks to prepare students for careers as Avid operators. I take the goals of a liberal arts program to lie in a different direction, and thus feel little remorse in my inability to discuss Xpress DV in further detail.

Cinestream is a revision of the old Radius/Digital Origin EditDV software. The new version adds many important features — including the ability to easily combine different sequences of clips together in a combined timeline. I have not had the opportunity to use Cinestream, but Edit DV was a fast, clean, stable program that lacked real contender status mainly because it was missing the features Cinestream has now added. If Cinestream has retained EditDV's strengths while addressing its shortcomings, it could be a valuable tool. It remains to be seen, though, whether Cinestream will be able to secure a niche in the NLE marketplace in light of Final Cut's surging popularity. Like Xpress DV, Cinestream is a DV only editor, and is thus suitable only in an online-all-the-time lab.

Like Xpress DV, Final Cut Pro offers a more sophisticated and powerful interface in comparison to Premiere, though one that remains fairly easy to learn. Students will likely have a bit easier time getting started on Premiere, but by the end of a semester with FCP they should be reaping quite a few benefits of its improved workflow. Written by the same software engineer who originally created Premiere, FCP has more similarities to Premiere than the hype might lead you to expect. However, many important refinements have been added to the basic Premiere concepts. The most important, I think, are multiple levels of undo, and the ability to open multiple timelines and bins and copy back and forth between them. The later is especially important for long projects, as it easily enables the editor to work on the project as a series of more manageable sections, experiment with different ways of combining them, and make changes to the sections even after they've been combined. Premiere can combine timelines, but the process is labored and offers much less flexibility once everything has been merged together. FCP has more capability to create visual effects, incorporating a number of features of Adobe After Effects, absent from Premiere. FCP is also far easier to navigate through an offline/online process than Premiere, which has an obtuse sequence of steps required to redigitize a project as a different resolution or with a different codec.

FCP has recently (4/01) advanced to version 2.0, which adds real-time capability in conjunction with the Matrox RTMac board (see below), as well as some minor improvements to the interface. As with Premiere 6, however, support for low-cost MJPEG boards that worked with earlier versions has been dropped. However reports on the Internet indicate that Final Cut 2.0 does still work with Targa 1000/2000 capture cards, despite Apple having abandoned official support for them. In a very brief and simple test, I was able to get FCP 2.0 working with a Targa 2000 card without encountering problems. New MJPEG boards fully certified to work with FCP 2.0 and Premiere 6 start at around $1000. Adobe has gone so far as to indicate Premiere compatibility with the Targa boards was tested and failed. Again, I made my own brief test, and indeed I was unable to get Premiere 6 to recognize video from the Targa during capture, and though it would playback previously captured MJPEG files through the Targa, the Targa did not respond properly to changes in the Premiere 6 playback settings.

Premiere 6 has narrowed the overall usability gap between FCP and Premiere. Premiere 5.1 had an all but useless Ripple edit tool, and lacked the ability to create keyboard shortcuts for several common commands. The Ripple edit tool in Premiere 6 works as it should, and the implementation of keyboard shortcuts is somewhat improved, though still well short of FCP. The awkward routines for combining timelines and redigitizing remain unchanged, though.

Premiere 6 does best FCP in at least one area — working with audio. It has a Mixer window that allows real-time level adjustment with 'faders', and the capability to automate the level changes — recording them into the EDL of the timeline. The Mixer window also shows LED type VU meters. Most importantly, I think, Premiere allows you to hear the effect of audio filters — eq, delay, chorus, etc. — and changes in filter settings without rendering, enabling immediate preview of the filtered sound. With FCP, unless you have an unusually powerful computer with very fast drives (i.e. probably not the kind of machine you're going to wind up with in your teaching lab) audio adjustment is much more a matter of guesswork, as you usually have to render audio clips before you can hear the results of any changes involving filters.

Both Premiere 6 and Final Cut 2 offer improved device control. All versions of Final Cut include RS-232 and RS-422 control built-in, as well as two versions of Firewire control. Firewire control in Final Cut 1.25 and earlier is excellent with Sony VCRs, good with most Canon camcorders, and chancier with JVC and Panasonic products. Apple's list of approved devices for FCP is fairly short and omits many JVC and Panasonic models. That doesn't mean unlisted models won't work — just that Apple isn't promising anything. Premiere 6 has newly re-engineered Firewire device control with specific settings for virtually every available model of DV camcorder and VCR. Despite Apple's minimal list for FCP, version 2.0 appears to work fine with JVC and Panasonic decks that did not work with earlier versions. Improvements in compatibility (as opposed to accuracy) appear to be due more to the OS and driver updates required to run FCP 2.0 and Premiere 6 than to changes in the editing software itself. For example, in my tests the JVC SR-VS20U had problems establishing a solid Firewire connection using FCP 1.2, Firewire 2.3.3, Quicktime 4.12 and Mac OS 8.6, but worked fine with FCP 1.2.5, OS 9.1, Quicktime 5 and Firewire 2.8.1.

Different people have different experiences with the relative bugginess of FCP and Premiere. In general, we could sum up the bug differences between the two programs by saying that Premiere tends to crash and FCP tends to have problems with batch capture and dropped frames. Neither is totally debilitating, both are frustrating if you're plagued by them. Apparently, a good number of users switched from Premiere 5.1 to FCP because they got tired of Premiere quitting in the middle of a work session. FCP almost never does that, but it has other quirks. For example, even slight mis-communication between FCP and the hard drive during capture may cause the capture to abort and the system to report a spurious error message indicating faulty timecode on the source tape (sending the user off in the wrong direction in an attempt to remedy the problem...) This is not a universal problem — many users do not experience it, — but it is a problem that Premiere does not have.

In my experience, on a Mac platform, Premiere 5.1c was a bit less troublesome than FCP 1.0 - 1.2.1 on a Mac G3 using OS 8.6. However, Premiere 5.1 got much more difficult with an upgrade to OS 9, and I do not recommend this combination. FCP 1.2.1 remained about the same with OS updates through ver. 9.0.4. FCP 1.25 under OS 9.04 was a wee bit more cranky than the older combinations, but significantly improved with OS9.1. This is the most trouble-free FCP combination. Premiere 6 on OS 9.1 appears to significantly improve stability over any previous Premiere combo. FCP 2.0, on the other hand, has exaggerated the program's hardware quirks. A number of users who did not previously experience the phantom timecode break bug now do after upgrading (meaning FCP 2.0 seems to have gotten pickier about a data storage path 1.25 negotiated properly). However, we now use FCP 2.0 at Connecticut College, and have had no problems with it whatsoever, so far. Still, as far as present versions go on a Mac, Premiere 6 gets a slight nod for stability.

I cannot speak to the stability of Premiere on a PC platform, though reports in Internet forums indicate users of Premiere 6 PC are very happy with the improvements made in that area over previous PC versions, which have tended to be more troublesome than their Mac counterparts. The more important comparison here probably remains the superior stability of Macs in general as a multimedia content-creation platform

One minor issue that dogged both FCP and Premiere for the Mac until early 2001 was that they both relied on Apple's Quicktime DV software codec to render DV footage in transitions and titles. Versions of this codec through 4.1.2 contained two celebrated deficiencies — "luma clamping" and poor text renders. I found that while these effects were noticeable to the watchful eye, they did not cause any significant problems in the projects produced by our students. In the Fall of 2000, Apple developed a much improved version of their codec, included first in their Quicktime 5 public beta package, and then in the OS 9.1 upgrade (though the QT package in 9.1 is still designated 4.x.x, it does have the new codec...). FCP 2.0 comes with a special version of Quicktime 5 that enables real-time editing with the appropriate card, as well as including the new and improved codec.

The choice of a software package may come down to the type of system you wish to build. Users seeking to build an ultra low cost offline/online system may wish to drop back to Premiere 5.1c, which has the advantage of working with almost every MJPEG card available over the last 5 years, as well as with DV. It's also less demanding in terms of processor speed or available RAM than Premiere 6 or any version of FCP. A slightly spiffier offline/online system can be constructed using FCP, since the price of used Targa 1000/2000 cards — the standard around which FCP's MJPEG support was designed — has dropped dramatically, to $300-500 as of 6/01. Though requirements for the host computer system are greater for any version of FCP than for Premiere 5, this too becomes more affordable every three months, as Moore's Law predicts. (Premiere 6 is similar to FCP in practical system requirements.)

For DV-based online-only systems, I would say the choice between FCP and Premiere 6 is a close call, weighing FCP's more sophisticated editing interface and visual effects against Premiere's superior audio tools. At present (6/01), although Premiere supports several PC real-time cards, it does not include real-time support on the Mac, while FCP 2.0 does, making it the only choice for those who wish to add real time support to the benefits of the Mac platform.

Summary

For a DV based online-all-the-way system, if the budget is available, I personally prefer Final Cut Pro 2.0 because 1) its ability to open multiple timelines makes it better suited to longer projects, including documentaries 2) it has the only real-time support for Macs (I think real-time is important, and I much prefer working with Macs compared to PCs). But then, I have enough technical know-how to have been able to deal with the FCP hardware quirks that have popped up in our lab. Premiere is less picky about hardware setups, and the improved stability of version 6 may make it a better choice where 1) real-time is not required or not within the budget 2) tech support is left to a faculty or staff member with little ability or desire to get under the hood now and then.

In order to build a more budget-conscious offline-online lab, you will probably want to go back to previous versions of Premiere or FCP. Premiere 5.1 is the best choice for an ultra-low-cost MJPEG system, since it will work with extremely inexpensive capture cards and computers. Final Cut Pro would be the choice for an improved yet still cost conscious MJPEG offline/online system.

A Few Potential Issues with Mobile Racks and/or Firewire Devices

Mobile rack systems have their downside, though I would argue a minimal one. A removable hard drive is more likely than a fixed unit to be dropped or have soda spilled on it — but the very premise of a video production class is that we must trust students to handle expensive equipment with proper care. Migrating hard drives can sometimes lead to software installed on the computer acting up in minor ways, (usually cured by a re-boot).

While computer components are promoted as standardized interchangeable parts, the reality is often a bit more complicated. You would think, for instance that any IDE hard drive mechanism of the appropriate physical size should be able to be placed in an IDE mobile rack, and once the rack system is attached to any standard IDE port, it ought to work properly. However, adding a mobile rack to an IDE drive system may narrow the range of real-life component compatibility — specifically, a hard drive that may work fine when connected directly to the interface may not work reliably when installed in a mobile rack. Hard drive connections are touchy things, as anyone who had dealt with termination or ID problems on a SCSI chain knows. IDE connections don't use terminators — which is why they are restricted to short cables and internal installation only. Placing an IDE drive in a mobile rack adds another connector and a couple more inches of wire to the cable path — which may be just enough extra length to create enough additional interference on the cable to cause little glitches in the data transfer. Different drive makes and different IDE controllers are more or less susceptible to interference. Most combinations *will* work fine in a mobile rack. At this point, the best guarantee of success is to use IBM Deskstar drives, which have proven to be top performers in just about every category, including resistance to cable interference.

Individual external Firewire drives may be a useful storage option now that prices have come down, and the Granite Firewire removable system is an attractive possibility. However there are a few things anyone considering Firewire storage ought to know. By my evaluation, it is only with the release of new Firewire drivers in OS 9.1 (Jan, 2001) that Firewire drives have become fully reliable DV capture devices for the Mac. (I don't know what the situation is for Wintel machines, but in general PC Firewire support lags behind Apple.) As I noted earlier, at this point (6/01), native Firewire hard drives do not exist and all Firewire hard drives on the market are actually IDE drives in an external housing with a special IDE-to-Firewire bridge chip mounted on a small circuit board between the IDE connection on the raw drive and the Firewire port on the back of the case. All of the bridge chips used in external Firewire cases manufactured prior to 2001 produced a significant performance drop compared to the same raw drive connected directly to an IDE port on the computer motherboard. However, in early 2001 the new Oxford high-performance IDE-Firewire bridge chip was released. There is little if any performance drop between an UltraATA drive mounted via an Oxford-equipped bridge board and the same drive connected to a motherboard IDE port. At this point (6/01) most manufacturers of Firewire drives and Firewire drive kits (you get the case, power supply, bridge board and cable; everything but the drive mechanism itself, which is easily installed) are still using the older bridge boards, though a few, most notably Granite Digital and Firewire Depot have moved to the Oxford chip.

Do not invest in Firewire technology unless you are getting the Oxford chipset.

Of course, new competitors of comparable performance may well appear after this writing. It's not that Firewire drives based on the older bridge-chips don't work for DV, they do. They just don't have much room for error. Think of video data streaming from an older Firewire drive as a package that has to move at 60mph being carried in a car with a top speed of 65. It goes along, but you're pushing kind of close to the limit and there's not much room for error. You'd feel more comfortable trying to sustain 60mph in a car that can cruise at 100mph if it has to. The older bridge-boards are more likely to produced dropped frames when other parts of the system get stressed, and may not have the throughput for real-time effects with a Matrox RT Mac, Matrox RT2000, or Pinnacle DV1000. Oxford bridgeboards, on the other hand, offer plenty of speed for these purposes.

Unfortunately, at this point, the Granite pre-packaged removable system in the only form of mobile rack Firewire storage I can recommend. Granite offers a component bridgeboard that can attach to the back of any IDE device housed in a case with a bit of room at the back. One ought to be able to construct a do-it-yourself Firewire mobile rack system by mounting standard IDE mobile racks in standard enclosures, and attaching Granite bridgeboards to the back of the rack receivers. The reason for doing this would be to save some money over the pre-packaged system. For example, an extra drive tray for a standard mobile rack can be purchased for $8, the special Granite trays cost $30. I tried building such a DIY enclosure with the Granite boards, and it failed to work properly. We have yet to determine whether the problem arose from some manufacturing defect, or is a generic incompatibility with the components involved. Not that this problem matters much — for most users the time and effort required to assemble the DIY system is worth more than the extra cost of the elegant pre-packaged solution. (I'm a tinkerer by nature, and have taken on the challenge of exploring different options for the purpose of creating this document.)

See the discussion of "Adding removable drives to a Mac" in the appendix for additional information, including non-Firewire options.

Choosing a VCR

All DV VCRs are essentially identical in image quality, so choosing one comes down to issues of features, reliability and price. There are more expensive machines, but none better suited to school lab than the Sony DSR-20 (street price around $2700). It's narrow design is efficient with desktop space, yet it's very well built, and has proven to be extremely reliable. It records in DVCAM, but plays back DV as well. A digital tape recorded in one make or model of VCR may not play back perfectly in another, but I have yet to find a DV-SP or DVCAM tape that won't play in a DSR-20. Another issue that pops up with DV VCRs is how their audio D/A converters handle overdriven audio tracks, and the DSR-20 handles them gracefully. It has a responsive transport, making it easy to shuttle tape back and forth and find a particular spot, and an RS-232 port for users who need serial device control. Audio level controls allow for making proper adjustment when dubbing from an analog source. The front panel displays timecode, audio level, and audio sample rate readouts, all of which are useful.

I must digress here for a bit on the subject of audio sample rates. A common technical error for DV production beginners is to digitize audio at a different sample than the audio was recorded. The DV spec allows for audio to be digitized at either 32K (12 bit) or 48K (16 bit) sample rates. Most camcorders are set to 32K by default, but editing software is more likely to set the higher quality 48K as a default. Once a camera is set, it should stay set — in theory. But in the real world of cameras changing hands from student to student, settings the users aren't supposed to know exist wind up getting changed or reset somehow. The problem is that when 32K audio is captured with a 48K setting, the resulting sound track will play back with obnoxious pops and crackling noises. Students may wind up with a mixture of 32K and 48K audio on their source tapes, and not be aware of it. The only way to deal with the different sample rates in FCP or Premiere is to adjust the capture settings to the appropriate rate for each chunk of tape. You have to do this manually — the software doesn't auto-recognize the sample rate. Thus, you need to know what the rate on the tape is, which you can only find out from the playback deck. Most DV VCRs display this info in some way, usually in the form of an overlay display on top of the video output. Having it displayed on the front of the VCR itself, however, is more convenient and less distracting.

The only drawback to the DSR-20, and it is very minor, is that since it records only in DVCAM format, finished projects laid back to tape will not be playable on DV camcorders that student's families may own. This has not been a problem for us, as DV camcorders are still fairly rare in the general public, and our students generally make VHS dubs to keep and show copies of their projects. However, I expect DV camcorders to become more common in the next two years. Nevertheless, if a school was equipped with DV camcorders and DVCAM editing decks it would be a fairly easy task for students to check out a camera, hook it to the DVCAM deck via Firewire, and dub the DVCAM master of their projects over to a mini-DV tape.

Three possible substitutes for the DSR-20 at lower prices are the Sony DSR-11

(~$2000), the JVC SR-SV20U (~$1500), and the Panasonic AG-DV1000. .

The DSR-11 is small but sturdy, and like the DSR-20 it accepts both larger 'standard' DV tapes and regular 'mini' size tapes. It is unique in that it can record in both DVCAM and DV format. The transport of the DSR-11 is not as responsive as the DSR-20's. When you press 'stop' or 'play' from 'fast-forward' or 'rewind' the deck continues to shuttle another several minutes tape as it slows down before stopping. This makes locating specific sections of tape rather bothersome. In my experience, the DSR-11 also may create small read errors when playing back DVCAM tapes recorded in a different model. It lacks the audio level controls and front panel readouts of the DSR-20. Timecode, format and audio sample frequency are displayed on an overlay, but this can only be turned on or off by the remote control, not from the deck itself (of course, the overlays do not appear on the video images passed to the computer via Firewire.)

I strongly recommend anyone using Final Cut in versions prior to 1.2.5 and/or versions of the Mac OS prior to 9.1 to use one of the Sony decks. Sony developed the Firewire standard along with Apple, and Sony gear has rock solid Firewire support. While Sony remains (7/01) the Firewire standard and the closest thing to a guarantee of trouble-free operation, as Apple continues to develop Firewire driver software, and as the device control within Final Cut and Premiere has been refined, more devices from other manufacturers become more stable and usable.

The JVC SR-VS20U is a dual-deck model, containing both a mini-DV transport and an SVHS transport. It is a more 'consumer-ish' product, lacking audio level controls and audio meters. Some of its controls are only accessible from the remote control, making it somewhat less suited to a lab setup. The timecode display in both the front panel and the screen overlay displays only minutes and seconds, not frames. One key feature of the SR -VS20U (lacking from its predecessor, the SR-VS10U) is DVCAM playback compatibility, valuable on any campus where Sony DVCAM gear might be owned by another department. Unlike the Sony decks, which accept both DV tape sizes, the SR-VS20 only accepts the more common 'mini' sized tapes. The SRV-S20 performed perfectly in Firewire device control tests using Mac OS 9.1, Final Cut 2.0, Quicktime 5 and Firewire 2.8.1. Successive batch captures of a series of clips produced absolutely identical media. However, problems appeared with earlier software versions. Under OS 8.6, Firewire 2.1, Quicktime 4.1.2 the deck wouldn't respond at all to FCP 1.2. Communication was established by upgrading to Firewire 2.3.3, but proved to be intermittent — though device control engaged at all times, sometimes the video image came through during batch capture, but sometimes it didn't, resulting in the production of a series of small empty clips in the capture folder.

I have not tested the Panasonic AG-DV1000, but I would expect its Firewire performance to be similar to the JVC's. It's a single transport DV deck, without DVCAM playback capability, that accepts only 'mini' sized tapes.

Choosing Camcorders

The Sony TRV900 cameras in system 'A' are the 'prosumer' standard. They have a three-chip pickup system, producing a beautiful image, and a full range of controls with reasonable ergonomics by mini-camcorder standards. This is clearly the most desirable camera for a college class — spending more won't yield any significant improvements, and the TRV-900 offers definite advantages over lower-priced units. That said, however, less expensive one-chip DV cameras still produce very nice images, and some models have enough control options to get by, so this is definitely an area where the budget can be tightened.

While several manufacturers offer DV cameras, I recommend sticking with Sony or Canon. All mini-sized camcorders include some sort of image stabilization mechanism to combat the increased shake that occurs when you hold a small camera in front of you, as compared to bracing a larger one on your shoulder. There are two basic kinds of image stabilization systems — optical, used by Sony and Canon, and electronic, used by Panasonic, JVC, and others. The optical systems work well, the electronic ones do not. On top of this, Sony and Canon have a track record of producing fine all around video cameras — their cameras have consistently good pictures and often contain important control features lacking in competitors models.

Sony and Canon have both recently revamped their product lines. I have not been able to examine the new models in person, but the best choices appear to be the Sony TRV-17 at a discount street price of around $875 and the Canon Optura Pi at a discount street price of around $900 (6/01). The TRV-17 is probably the preferable choice as it has a more conventional manual focus control on the ring of the lens, while the Optura Pi uses the same multi-purpose dial at the rear of the camera that it uses to navigate the menu system and adjust manual exposure. A lower priced Canon, the ZR20 at a street price of $565, may be an option, though it lacks manual exposure control. While it does have an AE shift function, this doesn't lock the setting, thus offering no good way to shoot scenes where reflectivity changes with movement in and out of the frame — a person dressed in bright white walking through the foreground, for example. If you don't consider this flaw debilitating, you should find the ZR20 adequate if budget dictates the lower price. Both Canons can set white balance and shutter speed manually, but only via menu options (a bit clumsy).

PART 3: An Ideal System...

An ideal NLE system for a liberal arts program would:

1. Be capable of online quality output at approximately DV image quality.
2. Accommodates multiple users via the use of removable storage media.
3. Enable low-res offline work by:
    A. Including timecode, device control and batch redigitizing.
    B. Providing an efficient codec yielding viewable images at low data rates.

(~750 MB sec. or less)

4. Be easy to learn, with an "intuitive" interface, and easy to use.
5. Enable split edits, audio mix and level adjustments and other fundamental edit procedures without workarounds.
6. Offer real time fades, dissolves and text overlays.
7. Be highly reliable, and maintenance free.
8. Not cost a lot of money.

Of course, the ideal system does not exist.

PART 4: Specific Systems

Below are some examples of NLE systems that might be used in a small, liberal arts oriented video production program. There are three subsections. I will begin with more desirable configurations. Second, I discuss some systems appropriate to a single small class (10 students or less) where students might all use a single editing station to keep the budget restrained. Finally, I discuss some more radical and compromised alternatives for shoestring budgets with multiple students. I do not want to give these budget-basement concepts an unqualified endorsement, or necessarily even any endorsement. They presume a dedicated and technology-savvy instructor, and are offered primarily to show what's possible under meager circumstances. That is, you shouldn't base anything on these models unless you absolutely have to, but if you do have to, you should be able to make these designs work.

For most of the systems I have included a rough minimum 'start-up' budget for a single class of 16 students, including not only the base edit systems, but also camcorders and removable storage. I must note however that these budgets do not account for things like tripods, external mics, camera bags or cases, furniture for the lab, and many other items that may be needed to set up a production class. All of the budget figures are offered for comparison, though hopefully they represent a starting point for gauging an actual budget, depending on the precise needs of your situation. All of the prices listed reflect typical discount 'street prices' as of the summer of 2001, and are, of course, subject to change.

The premises at work here are as follows. The class is devoted to aesthetic issues in filmmaking, and the tools are means to an end, rather than the subject of the class. As such, most class time will be devoted to discussions of conceptual issues, and students will work on their production projects on their own time outside of class. In contrast to a lab designed for a course focused on the technical processes of NLE itself, we do not need enough workstations to have every student sit down in front of their own station during class. What we need is enough workstations, available for enough hours during the day, that students will able to complete the number and kind of projects we want them to do, without forcing them to work in unwieldy large groups. I consider 16 students to be a good average size for a production class. Assuming the editing lab has 24 hour access, I believe two editing systems and three camcorders are the minimum stock of gear needed to support a class of that size. This also assumes this equipment is devoted exclusively to the class, and not used for other purposes. As the number of students enrolled in any one semester increases, the number of available workstations and cameras should also go up, but not necessarily in direct proportion as certain economies of scale are involved. At Connecticut College, for example, we have designed our facility to accommodate up to 40 users during any term, and we have 4 edit stations and six camcorders.

I want to emphasize that I base the bulk of the following discussion on some form of concrete experience, not "book knowledge". Still, in order to present a reasonably thorough survey of the terrain I will need to mention possibilities outside of my direct experience. However I have tried to be clear about what I have actually tested and used, and what I merely conclude should work OK.

However, in saying "I have actually assembled this or used this and it worked for me" I also need to note the limits of such a statement. Poke around technology discussion boards on the Net, and you'll find that for just about any product, there are a group of people who seem to have no problems with it, and others who have nothing but trouble. Reading these posts, one tries to gauge how the numbers balance on either side. I can only offer a single report. Just because something works for me is no guarantee it will work for you. Conversely, just because I had a problem doesn't mean someone else hasn't gotten the same configuration to perform splendidly.

For many of the systems described below, I have assembled prototypes, and performed tests designed to simulate a typical student project. That's a decent predictor of long-term success, but hardly an ironclad guarantee. In every case, I have set up the systems as fresh installs, with nothing else loaded on the computer beforehand, and nothing else added later. Again, I cannot emphasize enough that the quickest way to a sick NLE system is to put a bunch of other applications on the computer. Different software packages put different little pieces of code in the system folder that may conflict with one another, or eat up a necessary resource. Otherwise healthy SCSI chains go bonkers if you add a scanner. And so on. Ask your system to do the minimum variety of tasks possible.

4.1: Recommended Options

A: Mac DV Real-time

This is my own first choice for schools that do not have severe budget restrictions. It's a DV online-only system, so it presumes each user of the lab will have their own hard drive . 'User' may not necessarily mean an individual, as you might have students working on assignments in pairs or even groups of three. I am completely convinced that online editing with removable drives is the best way for students to work, but this does mean you need to be able to acquire lots of hard drives. Again, though, drive prices have fallen so far that this is nowhere near the impediment it used to be.

The system includes Final Cut Pro 2.0 running on a Mac G4/466 upgraded to at least 512MB of RAM. Real-time processing for dissolves and titles is provided by a Matrox RTMac board. Individual storage is provided by mobile racks in external enclosures, connected to the Mac via Firewire. Editing software is easier to use the more screen real estate you have to spread out the various windows. A 15" LCD main monitor provides the minimum recommended screen resolution for FCP without requiring users to squint. Since the Matrox card also drives a second monitor we add another 15" LCD. A 13" Sony monitor provides NTSC video display. A Sony DSR-20 DVCAM VCR provides DV input and output.

RTMac

The Matrox RTMac board offers a limited set of real-time effects for FCP at a reasonable price. At first glance, one may be disappointed at the number of transitions *not* supported for real-time, or the fact that no image filters are supported in real-time. Yet, what the RTMac does do is still significant. It handles two video layers and a graphic layer without rendering. Thus it produces fades, dissolves and title overlays — even a title overlay over a dissolve — without rendering. These are the most common 'effects' in either narrative or documentary production, and eliminating the need to render them cuts out maybe 70% of the render time involved in a typical student project in these forms.

The RTMac is a digital/analog hybrid. It uses DV input, but actually creates the real-time output as an analog signal available at the card's breakout box. The system will not create real-time effects in the DV stream sent back out the Firewire port. This is not a serious disadvantage, since we simply connect the NTSC monitor to the Matrox board rather than to the DV deck in order to view the image during editing. For final DV output back to tape all the effects do need to be rendered, but this need not be done during the actual editing work process, only once at the end. The problem with needing to render bread and butter transitions such as fades and dissolves is that one render is seldom enough. As with a straight cut, it often takes several trial and error attempts to get the timing of a transition just right. Adjust, render, view, adjust, render, view... the time adds up quickly eating into the productivity of the editing session. With the RTMac, the feedback on fades dissolves and supers (and wipes, if you go for that sort of thing) is immediate, which makes a huge difference.

Approximate Budget

PowerMac G4 466 w Zip /no modem	1600
Additional 512MB RAM	100
Matrox RTMac Board	1000
(2) NEC LCD 1530V 15" monitors	
@ $400 ea	800
Sony DSR-20 DVCAM Deck	2600

Final Cut Pro	2.0	250
Sony 13" NTSC monitor	300	
Granite Digital 2 bay Firewire case	530	
price per edit station	7,180	
(2) edit systems	14,360	
(4) Sony *TRV-900 (3 chip)* camcorders @ $1900	7600	
(16) IBM Deskstar 75GXP 45GB ATA100 hard drives @175 ea.	2800	
(14) Granite Digital Mobile Rack Trays @ $27ea.	378	
total budget for 16 student class	25,346	

Evaluation

Strong Points: DV offers excellent image quality. Real-time features allow students to work on the important things instead of waiting for renders. Removable drives allow each individual user to work in online mode at reasonable cost. FCP is easy enough to get started, yet offers considerable power as students become more familiar with it. First choice for a closed lab with a fixed number of users.

Weak Points: No offline mode as configured. Some parts of FCP, including batch capture, may still a bit cranky (though so far we have had no issues with our new FCP 2.0 setup, very similar to what is described here).

Variations

I have chosen the Sony DSR-20 because it is simply the most reliable, usable DV deck at a reasonable price. Possible substitutions at a lower price are the Sony DSR-11 (~2000) and the JVC SR-SV20U (~1500). I have also chosen three-chip Sony cameras due to their outstanding picture quality, and their full range of manual controls. One chip cameras from Sony or Canon still offer excellent image quality at much lower prices, though with reduced feature sets.

Unlike most DV only editing setups, with the Matrox board installed, it is not necessary to have a DV device connected to the system at all times to view NTSC output. The only time the DV deck need be connected is for capturing and final layback to tape. It would be possible, with appropriate cable switches, to share a DV VCR between two edit stations (though assignment schedules often dictate that all members of the class may be capturing footage at similar times, which could be a problem with shared decks.) It would also be possible to forego having decks dedicated to the edit systems at all, with students needing to check out a camcorder and bring it into the lab to do captures (this may or may not interfere with their availability for shooting, and it is probably not desirable to add the wear of regular capture sessions to the lightweight transports used in camcorders).

I chose the Granite FireVue storage devices to hold the removable drive racks due to their ease of assembly. The FireVue enclosure contains two mobile racks, making 2 drive RAIDs possible. This system should not demand RAID performance, and students should need no more than a single drive. Having the second bay is not strictly necessary, but it can be a valuable option as it provides for easy transfer of media files between hard drives, and allows you to issue a second hard drive to the deserving student with a long form project who has filled up her first drive.

There are other ways to get removable drives working on a Mac, which are somewhat less expensive if requiring a good bit more work. They are discussed in the appendix, through I would guess that for most labs the benefits of the FireVue will outweigh the added cost.

B. Straight Mac DV on a Somewhat Smaller Budget

Real-time is nice, but not absolutely necessary, and a school constructing a lab on a more modest budget may choose to eliminate the Matrox boards. Some of the other components of system 'A' above can also be replaced by lower cost substitutes.

I chose LCD flat panel displays for system 'A' because 1) they don't interfere with one another as CRTs do, thus work better in a tight space where two data monitors and an NTSC monitor sit side by side, and 2) they take up less desk space, which is often tight in a lab. That said, LCDs are still fairly expensive, and some savings could be generated by substituting CRT monitors. Video editing apps demand a main monitor that can display 1024 x768 without eyestrain — which means a 19" CRT. While two monitors are definitely preferable, you can by with a single monitor as long as it's big enough. If you do add a second monitor, it may be smaller, as its main function is to hold smaller extra windows. A 17" or even a 15" CRT would be adequate. With the Matrox board driving a second monitor, interference between CRTs is definitely an issue as the Matrox refresh is clocked slightly differently from the standard graphics boards in the Mac. Without the Matrox board, you can avoid interference problems by using a second card of the same type and make as the main card. In the case of a new Mac G4, this probably means substituting an ATI AGP card for the stock NVIDIA card, and getting a second PCI ATI card.

Approximate Budget

PowerMac G4 466 w Zip /no modem	1600
Additional 512MB RAM	100
Generic 19" monitor	250
JVC SR-S20U DV deck	1500
Premiere 6	350
Sony 13" NTSC monitor	300
2-bay DIY external case for removable drives (w. SCSI card)	380
price per edit station	4,480

(2) edit systems	8,960
(4) Sony TRV17 camcorders @ $875	3500

(16) IBM Deskstar 75GXP 45GB ATA100 hard drives @175 ea.	2800
(14) ViPower Mobile Rack Trays @ $11ea.	155
total budget for 16 student class	15, 515

Evaluation

Strong Points: DV offers excellent image quality. Removable drives allow each individual user to work in online mode at reasonable cost. Premiere 6 is stable and user-friendly to beginners.

Weak Points: No real-time. No offline mode as configured.

Variations

This system does not require a new G4, and should work fine with a Blue and White (Yosemite) G3/350 or better Mac. G3/400s should be available on eBay for around $900. Moving to a two-monitor setup sends the budget the other direction, of course. This may be an upgrade you seek to add down the road. Note that 2 monitor set-ups are a lot more pleasant to look at if the video cards and the monitors match (not necessarily in size, but in manufacturer and series.)

If you expect more ambitious work out of students, especially long form projects, it would be wise to use Final Cut instead of Premiere.

C. Online/Offline Real-time on a PC

I'm very Mac-centric, but some of the newer NLE packages for Windows PCs offer intriguing benefits — specifically real-time editing and an integration of DV input with a scalable MPEG-2 codec in the same package. Thanks to kind persons in the media relations staffs at Matrox and Pinnacle, during the summer of 2000 I was able to borrow a card from each company and test it in preparation for this article.

I have moved my review of the Matrox RT2000 to the appendix, as it is now outdated by changes in the product, and Matrox's introduction of a successor, the RT2500. I have not had the opportunity to examine the updated Matrox systems, so I can't really say anything about them one way or the other. In my initial reviews, I concluded that although the Pinnacle DC1000 had some important advantages over the RT2000 in an educational setting, specifically a clear superiority in acting as an *offline* system. I cannot say whether changes in the Matrox line have addressed the differences I found to be important, but I include my original observations on the Matrox in the appendix so anyone investigating their current line will know the strengths and weaknesses I found in the earlier offering, and be able address whether those things have changed, or stayed the same.

The DC1000 captures and edits in MPEG-2 intra-frame compression — either IPP or IBP — which means it can compress images much more efficiently than M-JPEG, MPEG-2 I-frame, or DV. The DV input/output option of the DC1000 does not enable it to edit native DV, like the Mac systems with or without the Matrox RTMac, or the Matrox

RT2000 for the PC. Rather it uses a hardware codec to transcode DV to MPEG-2 in real-time.

Pinnacle claims that by transforming the image from the 4:1:1 coded colorspace of DV to the 4:2:2 coded color space of MPEG-2, the DC1000 can create cleaner effects. However, Matrox notes that transcoding the DV stream can't make it any better, and necessarily introduces *some* measure of artifacting that would not be present in a pure DV system. Like the RTMac (and the RT2000), the DC1000 provides monitoring during editing in the conventional manner — analog outputs from a break-out box — rather than using Firewire and requiring a connected DV device to do the DA conversion. I noticed that the image ouput from the breakout-box of footage captured from DV did look distinctly different from the DV original — with softer edges and somewhat washed-out appearance. However, after sending the footage back out the DC1000's Firewire connection — thus transcoding it twice, once going into the system and once going out — the DV ouput was indistinguishable from the original in casual viewing. I concluded that whatever degradation the transcoding may introduce was not enough to worry about, especially for the purposes of a teaching lab.

The great benefit of the DC1000 is its combination of basic real-time effects with a high-efficiency codec that allows offline work on smaller drives, including removable cartridge media such as the Castlewood Orb. The DC1000 lets you chose between variable bit rate compression, and constant bit rate compression (CBR). Thus, you can set your capture data rate so it never exceeds the capacity of the Orb. What's more, with the intra-frame compression, at the upper end of the Orb's data rate range the image actually looks quite nice, with very little artifacting, if any. The efficiency of the codec also lets you push the data rate lower before the image becomes unusable — 4Mbit captures (~500KB/sec!) are good enough for working offline. This makes employing Orb drives more practical by increasing their usable capacity in minutes of captured video. At a typical working bit rate determined by the Orb's capacity, the compression is efficient enough that the drive is capable of delivering enough data to form two video streams — so you can use real-time effects even with the cartridge drive!!

The Pinnacle system does a good job of overcoming the potential problems of intra-frame compression. I was worried that all the extra calculation needed to fill-in-the-blanks of partial frames might slow down the system during editing. There's even a warning in the manual that with very long and complex projects, the system may take "several minutes" to respond the first time you click in the timeline. I did not have time to create an "organic" complex project to test the system, but I did attempt to simulate a more complex piece by throwing over a half hour of chopped-up video into the timeline, and adding a mess of dissolves. Yet, Premiere performance did not drag at all (this was 5.1; the system now uses version 6). I had no problems playing or scrolling anywhere in the timeline. I should note, too, that my test DC1000 was installed in an old IBM Intellistation equipped with only the minimum recommended RAM and processor speed.

The real-time capabilities of the DC1000 are strong, if not quite as extensive as the

RT2000 or as adjustable as the RTMac. There are none of the real-time 3D effects offered by the RT2000 — though for many of us this may seem more a benefit than a deficit (if you add a page peel, you deserve to suffer the wait for it to render!!). Nor can the DC1000 match the keyframing abilities of the FCP/RTMac combo. The DC1000 also uses some of its "real-time" resources to perform smooth playback of MPEG partial frames. That is, since a cut is unlikely to occur at two I-frames, in order to make the cut, the system essentially backs up to the last complete frames, and then composes the two shots from those reference points, using two video streams through the board, and makes the cut in the appropriate spot on the fly. For simple transitions — fades, dissolves, two-layer composites — this has no practical effect on performance. However, in a more complex sequence — perhaps involving a title keyed in over an effect — the DC1000 may have to do some rendering where the RT2000 would not. These limitations never gave me problems in my brief test of the board — I only became aware of it by reading the documentation. In terms of the kinds of work we do in our program, I was quite pleased by the real-time function of the DC1000.

With this system, you can set up the lab so everyone works offline, with either an Orb cartridge or a small removable drive, or you set up a tiered system where some users — say students enrolled in a particular class — work in online mode with a large removable drive while other users with less available storage work offline. In any case, once an offline user finishes a project, she can check out a larger removable hard drive to redigitize a high res version of her project and output it to tape.

Adding removable drives to a PC is not as much of a challenge as it is on a Mac, as PCs are available in tower cases with open, front-facing 5 1/4" drive bays — ready to accept one or (preferably) two mobile rack frames, and have enough PCI slots to easily add an additional IDE interface card to which the mobile racks can be connected. A Firewire storage solution provides no benefit here, and is not recommended.

In building a system around the DC1000 a couple of the general points I mentioned earlier about PC-based systems need to be reprised. First is device control. The Pinnacle Firewire device control plug-in on my test system was woefully inadequate. Pinnacle may have improved its plug-in or Firewire drivers since then, and Premiere 6 now offers built-in Firewire device control for most common DV devices. Given the degree to which Adobe is now emphasizing Firewire In/Out, I am sure the situation has greatly improved, though I have been unable to test it personally. Less due to my remaining qualms about PC based Firewire, than to the importance of highly accurate time code to any offline/online system, I must strongly recommend third-party RS-422 device control for this package. I have also specified the Sony DSR-40 DVCAM deck because of its RS-422 interface. Second is the relative difficulty of putting a PC-based system together from scratch — some component parts, especially different motherboards, will be compatible with the DC1000 and others will not. There are also certain tricks required in setting up the IRQs and other system parameters. So, again, I strongly recommend anyone acquiring a DC1000 system to purchase it from a turnkey vendor.

The problem here is that turnkey vendors may only be willing to offer specific configurations to which they will refuse to make any alterations. These stock configurations will inevitably be based on typical NLE design, and not include removable drives. If you can't get exactly what you want, the trick is to find a vendor who will send you the tested, working system that is as close to what you want to wind up with as possible so you'll reduce the risk of mucking things up as you make your own tweaks. For example, one turnkey vendor I spoke to refused to set up a system without using SCSI media drives, which just won't do. I found another vendor though, who likes to use IDE media drives, and offers an additional ATA 100 controller card as a standard option. In this case all one has to do is order the system with two fixed media drives attached to the ATA 100 controller, then pull the fixed drives and attach the mobile rack trays to the same cable positions. Change the drive settings in Windows to 'removable' and you should be ready to go, with no deep goofing around in the hardware settings.

What you want is:

- the system drive as master on the primary IDE port of the motherboard
- a zip drive as slave on the primary IDE port of the motherboard
- the cd drive as master on the secondary IDE port of the motherboard
- an orb drive as slave on the secondary IDE port of the motherboard
- a mobile rack as master on the primary IDE port of ATA 100 card
- a mobile rack as master on the secondary IDE port of ATA 100 card

Approximate Budget

1GHz+ Pentium PC Tower w. Matrox dual-head graphics card	1600
Pinnacle DC 1000 (software inc.)	1500
(2) NEC LCD 1530V 15" monitors @ $400 ea	800
Sony DSR-4 DVCAM deck	3500
Sony 13" NTSC monitor	300
Internal IDE Orb drive	150
Internal IDE Mobile Rack frames	30
ATA 100 dual port IDE interface card	50
Diaquest Timecoder RS-422 device control software	200
price per edit station	8,130

(2) edit systems	16,260
(4) Sony *TRV-900 (3 chip)* camcorders @ $1900	7600
(16) IBM Deskstar 75GXP 45GB ATA100 hard drives @175 ea.	2800
(14) Mobile Rack Trays @ $11ea.	155

total budget for 16 student class 26,815

Evaluation

Strong Points: The flexibility and savings it allows in storage are unique in a real-time system. Meets more criteria of the "ideal NLE for a liberal arts college" than any other system. Very good image quality. Real-time features allow students to work on the important things instead of waiting for renders. Removable drives allow each individual user to work in online mode at reasonable cost. Premiere is easy to learn. First choice for an open lab that must accommodate a flexible number of users.

Weak Points: PCs not as user friendly as Macs, more trouble prone overall, and higher difficulty of setup.

Variations

Some savings can be accomplished to fit a tighter budget by making the same sort of camera and monitor substitutions I noted in the more budget minded Mac system above.

As described above in the 'Online, Offline/Online and Storage Format' section of part 1, you may want to forego Orb drives for offline work in favor of smaller removable hard drives in mobile rack trays. Since the cost of these (~$90) is in the range that could conceivably be supported by the students in the form of a lab fee if not outright purchase, this would reduce the budget for this system by about $3000 by covering/eliminating the expenditure line for removable drives.

D. Basic Mac Offline/Online System (not tested)

This is for a lab with an large or unpredictable number of users, or a situation where the college cannot afford storage devices, and students will be responsible for their own storage, either by buying cartridges or 'renting' removable hard drives.

Students would shoot with digital camcorders, digitize from the analog outs of a DV deck to low-res M-JPEG stored on an Orb cartridge or small removable hard drives. Firewire device control will control digital cameras or decks whether the software is capturing DV through Firewire or another codec through a digitizing card. Students would create the project, then batch re-capture to the large fixed hard drive in digital mode via Firewire and export the all-digital conformed version back out through Firewire to a digital tape master on the camcorder again.

We'll use FCP 2.0 for FCP's easier offline/online operations. We'll also pony up for an officially supported capture card, for peace of mind. The Aurora Video Systems Igniter card is overkill in the sense that it can handle much higher data rates than you'll ever need, but it the cheapest scalable card with ongoing development and support. We'll add an Orb drive so anyone can use the system, but we'll also include a base for removable hard drives, as this is the preferred method for students to store files. We assume for this system that you devise some means of having students support the cost of these drives — through purchase, rental or lab fees — so they are not listed in the budget.

Again, please note that tripods, mics, bags, tax, shipping and other important whatnots are not included in these figures.

Approximate Budget

PowerMac G4 466 w. Zip/ no modem	1600
Additional 512MB RAM	100
Aurora Igniter MJPEG capture card	1100
IBM Deskstar 60GB 7200RPM/ATA100	200
Firewire Orb drive	220
Generic 19" monitor	250
JVC SR-S20U DV deck	1500
Final Cut 2.0	250
Sony 13" NTSC monitor	300
2-bay DIY external case for removable drives (w. SCSI card)	380
price per edit station	5,900

(2) edit systems	11,800
(4) Sony TRV17 camcorders @ $875	3,500
total budget for 16 student class	15, 300

Evaluation

Strong Points: Very good online image quality via DV. Orb drives allow multiple users on the system at low cost to them and no added cost to the school. Removable drive case provides for higher reliability, higher quality student-supported option with smaller hard drives.

Weak Points: No real time. Having everybody work offline is a bit messier technically and conceptually than pure online.

Variations

Upgrades would include moving to 3-chip TRV-900 cameras or a two monitor set-up.

There are several ways to lower the cost of the system. Used Yosemite G3s could be substituted for the G4s. A Targa card, preferably a Targa 1000, could be substituted for the Igniter. Though officially unsupported, Targas seem to work fine with FCP 2.0. As a worst case scenario you could downgrade to FCP 1.2.5 to regain full support of the Targa, since 2.0 offers few advantages over 1.2.5 in this configuration. Finally, it would be possible to eliminate the JVC source decks entirely. Instead, you would buy one extra camcorder and give it priority for use in the lab as a capture and record device. The online FinalCutPro system described back in section 'B' requires each station to be equipped with a dedicated source deck, since the VCR does the D/A conversion that allows the editor to see playback on the NTSC monitor. In an offline mode system like this one, though, a VCR doesn't absolutely need to be present at all times, only during capture and final output of the online version back to tape. During normal editing the digitizing card creates the NTSC output While having a feeder at each station would certainly be preferred, if budget demands it, it's not entirely impractical to have students use the camcorders for capture sessions.

E. A Pre-packaged Dedicated Editing Appliance (not tested)

All of the other NLE systems discussed in this paper are based upon general purpose computer hardware and operating systems. Install the right software and hardware in a Mac or PC, and it becomes an editing machine. Yet, creating systems in this manner contributes profoundly to perhaps the most common and troublesome deviation from our ideal — the lack of reliability. NLE engineers need to make their products work on top of a basic platform that wasn't designed for the purpose, and is composed of varying other software and hardware parts, many created by different engineers in different companies, that will be assembled in constantly varying combinations, with each component likely to have a short market-lifespan before being replaced by something newer and better — which inevitably creates a series of new incompatibilities amongst formerly harmonious parts. It's a wonder anything works at all.

There are more possibilities for error in a system that can do many things than in a system that can only do one thing. A desktop computer stores application files on re-writable magnetic media, then loads them into RAM when you want to use them. There are three potential problems here. The file containing parts of the program code can become corrupted. An error can be made loading the program. Finally, the RAM itself is subject to memory leaks that can create errors in the program's operation after it has been running awhile.

An alternative approach is presented by non-linear "editing appliances." These are specialized computers, in effect, that do nothing but act as video editors. All the software and hardware are created from the ground up by a single team of engineers. Since there is no need for a general OS, the software can use less code and still yield powerful functions. Since the device is only designed to handle one set of instructions, they can burned into a flashROM. There's no disk file to get corrupted, no time required to load the program file, no memory leaks. These benefits don't make the system perfect, but they certainly help.

The first low-cost editing appliance in the US market was the Draco Systems Casablanca, built around a now fairly ancient CPU (Motorola 68040). Draco has since changed its name to MacroSystems, and replaced the "Cassie" with the more up-to-date Kron. MacroSystems now also offers the Avio, a lower priced model, but one based on newer technology. Unfortunately, neither Kron nor Avio have the ability to handle multiple users using removable disks gracefully.

The "new kid" in the editing appliance business is Applied Magic, which offers the DV capable Screenplay "premium" NLE appliance, and the Sequel "entry-level" NLE appliance (no DV I/O). Both are reasonably priced — the Screenplay had a street price of about $3500, the Sequel around $2000 — and are offered in special 'Academic Editions' that include an addition to the standard software code for the purpose of accommodating an unlimited number of projects on removable disks. (This function is not available on her standard models, nor on the Kron or Avio.) For any editing appliance, all you need to add is a feed/record VCR and an NTSC monitor — no data monitor is required since the editing interface appears as an overlay on the NTSC video output. Appliances promise much greater reliability than conventional NLEs, and also feature simplified editing interfaces designed to be easy-to-learn. Both MacroSystems and Applied magic include a variety of testimonials to their systems' stability and ease-of-use in their sales literature and on their websites.

Screenplay and Sequel come with an internal fixed hard drive, and an ultrawide SCSI port to attach external drives. Until recently, this meant a removable drive system suitable for these products would need to employ costly SCSI mechanisms. However, a company called ACARD now produces a bridge chip board that attaches to the back of an IDE drive, allowing it to be connected to a SCSI chain. This device can be attached to the back of an IDE mobile rack in an external enclosure, which should allow the Screenplay to be used with relatively low-cost, high-capacity UltraATA removable drives, much like the other systems described above. (I have tested the ACARD board with IBM drives and ViPower Mobile Racks successfully, connected to an ATTO SCSI card installed in a Mac. I have not tried to connect such a drive system to an editing appliance, but it should definitely work.)

None Screenplay or Sequel employ timecode in a way that allows a user to batch re-capture a crashed project or create a low-res offline edit that can be conformed to hi-res later. This is especially unfortunate, since these systems employ an efficient wavelet codec that should be able to create legible workprint files at very low data rates. Appliances offer no easy way to move sections of clips between one timeline and another, an absence likely to make long-from work more laborious.

As I briefly noted earlier, another problem with editing appliances is that they have rather restricted audio capabilities. If the user seeks to go beyond 'boxcar' edits where audio and video are always cut together, to perform split edits ('L Cuts' and 'J Cuts'), the system becomes more complicated. Screenplay and Sequel can make split edits, but not directly — a workaround is required. The workaround isn't that complicated, as long as you get the edit points for the split right the first time. If you need to play around with the cut, though, you'll wind up reduplicating the workaround steps again and again.

The differences between the Sequel and Screenplay go beyond the DV I/O as the price difference suggests. Screenplay can capture video at a higher-quality data rate. Sequel lacks capabilities to do insert editing, keyframe transitions, fade titles, do custom color correction and combine audio tracks, all present in Screenplay. (The lack of insert editing and audio track mixing are quite limiting.) Another extra feature provided with the Screenplay is password protection, which can help greatly in lab management, especially if you wind up with a single-station situation where several users are sharing fixed drives.

The Screenplay should be a good choice for an online-only lab, using DV acquisition, especially attractive for courses that rely exclusively on a faculty member with little technical expertise or troubleshooting time to run the lab. The primary technical hurdle here would be creating the DIY external drive enclosures for the mobile racks — a one-time task (with instructions below). Compared to the Mac DV system "B", at a similar budget, the Sequel system offers greater simplicity and stability as a trade-off against the power and flexibility of the FCP interface.

Approximate Budget

Screenplay	3500
JVC SR-VS20U DV deck	1500
Sony 13" NTSC monitor	300
1-bay DIY external case for removable drives	230
price per edit station	5530

(2) edit systems	11060
(4) Sony TRV-17 camcorders @ $875	3500
(16) IBM Deskstar 75GXP 45GB ATA100 hard drives @175 ea.	2800
(14) ViPower Mobile Rack Trays @ $10 ea.	140
total budget for 16 student class	15, 500

Evaluation

Strong Points: Very good image quality. Stable and very easy to learn. Removable drives allow each individual user to work in online mode at reasonable cost.

Weak Points: No timecode-based disaster recovery. Common audio editing tasks require workarounds. Not as conducive to long form work as other systems.

4.2: Small Class, Small Budget

F. Single-Station Mid-Quality Online (not tested)

This is for programs with limited ambitions, looking to gain just a starting point in moving image making as an option in a much broader curriculum. We'll assume:

1) only one section is offered, and enrollment is limited to 10 students or less. (10 is doable, but pushing it with only one station, 8 would be a more comfortable figure)

2) neither the instructors nor the students have much technical background in either video or AV computers.

3) an absolutely minimal budget must be constructed

4) in that light, VHS quality is just fine for the finished projects

5) the school already owns a decent VHS/8mm deck and small TV with monitor inputs it can devote to the system, and VHS/8mm cameras are already available for the students to use

In light of our non-existent budget we will have to live with some distinct limitations, so we select the Screenplay editing appliance as the basis of the system. We gladly suffer its lower image quality, since the wavelet compression used lowers the requirement for storage space. We'll aim to provide each user with enough drive space to complete a reasonably complex project — 14GB. We'll achieve this by adding two IBM 60GB Deskstar drives with ACARD SCSIDE bridgeboards in an external enclosure — the Sequel has a 20GB drive mounted internally. Our external drives will be fixed, not removable, since we only have one station. The fairly simple mechanics of assembling the external drive should be the most technically challenging aspect of setting up the system.

Approximate Budget

Sequel	2000
(2) IBM 60GB Deskstar hard drives	360
(2) ACARD bridge boards	150
External SCSI 3 case for drives	100
total budget	2610

Assuming no associated gear was available we might add:

13" TV with monitor inputs	200
(4) Canon ZR20 DV cameras @ $550	2200
combined total budget	5010

That's 3 cameras to check out, and one to leave with the system for capture and export (via analog connections). If you have to buy new cameras, getting anything but digital is just a wasted investment at this point.

One important note about this system: it's upgradable to system "D" above. A Sequel can be upgraded to a Screenplay, and the external case can be fitted with mobile racks, using the same bridge-chips.

Evaluation

Strong Points: Stable and very easy to learn. Upgradable. Efficient codec saves storage space.

Weak Points: Somewhat reduced image quality. No timecode-based disaster recovery. Audio and video editing limitations will restrict range of basic creative choices. Not as conducive to long form work as other systems.

Alternatives

It would be better to start with the Screenplay, rather than the Sequel, if you can swing it, especially if you wind up with DV cameras. Screenplay's password feature is especially valuable where several users will be storing their files on the same fixed drives. Using a camcorder for capture and recording is a money-saving compromise with a couple drawbacks — a camcorder left in the lab is a theft target, and is likely not to hold up to heavy use as well as a desktop deck. Thus, it would be desirable to replace that lab camcorder with a JVC SR-VS20U, which would add about another $1000 to the bottom line.

G. DV Online Single Station (not tested)

For a single small class, a single Mac-based DV system with lots of fixed internal storage presents an alternative to the Screenplay/Sequel system described above. You can make this system more simple by using iMovie if you're willing to accept it's limitations, or you can go for a more full featured system by using Premiere 6 or Final Cut.

One good thing about iMovie is you download it for free, and if you have or can borrow a Mac and a digital video camera, you can check it out at length without investing anything. If you discover you can live with iMovie's limitations in terms of the kind of work you think students need to be doing, you can build a system around it. iMovie is well engineered for what it does, and it eliminates a lot of potential technical problems via simplicity and automation. For example, it figures out what sampling rate the audio signal was recorded at and adjusts it properly, where Premiere and FCP require the user to know the sample rate and set a preference field accordingly — an area that often crosses students up leading to distorted soundtracks. Again, I think iMovie would primarily be of interest to faculty who are new to production teaching, don't have that much time or energy to devote to technology, and will be offering only a single introductory course. Faculty with a bit more video background and ambitions for more sophisticated student work will probably want to opt for Premiere 6 or FCP.

This is a DV only editor, so we'll need more storage. For simplicity's sake, we'll use external Firewire drives. We'll get a total of five 60GB drives. Firewire can still be tricky with multiple devices, so it might or might work to attach all of these drives at once. However, it's no complex task to have students switch the cables to whatever drive their project is on, so we can get by connecting them one at a time if need be.

To keep the budget down, I have specified a used G3 computer, as it has plenty of power for the purpose at hand. iMovie displays video playback on the Data monitor nicely, and doesn't really require an additional NTSC monitor. For Premiere or FCP you'd want to add an NTSC monitor and a bit more RAM. Note that I have specified a desktop DV deck here, rather than the camera-in-the-lab compromise I used in figuring the budget for the appliance system above. This accounts for the difference between the iMovie system's cost, and the Sequel system's cost — using the same sort of video feeder the two configurations are very similar in cost.

Approximate Budget -iMovie 300GB external Firewire, desktop feeder

PowerMac G3/400 w. 256MB ram (used)	900
(5) IBM 60GB Deskstar hard drives @ $180	900
(5) Granite Digital 3.5" Firewire drive cases	750
(3) Canon ZR20 DV cameras @ $550	1650
JVC SR-VS20U DV deck	1500
total budget	5800

Evaluation

Strong Points: Very good image quality. Very easy to learn. Upgradable to more advanced software.

Weak Points: No timecode-based disaster recovery. Audio and video editing limitations will restrict range of basic creative choices somewhat. Not as conducive to long form work as other systems.

Approximate Budget -Premiere 300GB external Firewire, desktop feeder

PowerMac G3/400 w. 384MB ram (used)	900
Premiere 6	350
(5) IBM 60GB Deskstar hard drives @ $180	900
(5) Granite Digital 3.5" Firewire drive cases	750
(3) Canon ZR20 DV cameras @ $550	1650
JVC SR-VS20U DV deck	1500
Panasonic 13" TV w. line inputs	160
total budget	6210

Evaluation

Strong Points: Very good image quality. Easy to learn. Timecode-based disaster recovery.

Weak Points: Higher technological hassle level than iMovie system.

Alternatives

You can lower the cost by lowering the amount of storage each student receives. Rather than drop to 4 external Firewire drives, you might want to use 3 75GB drives mounted internally. There's a bracket to mount one drive above the system drive in the Mac, acting as a slave on the primary IDE port. (The earliest Blue and White G3s do not have this, you need to make sure yours has a revision 2 motherboard. See the G3 notes in the appendix.) There are physical mounting positions for two more drives in the bottom of the G3, but to connect them you'll need a Sonnet Tempo PCI ATA66 controller card. Where the budget list above offers 300GB storage at cost of $1650, this method would yield 225GB of storage at $850. You might also squeeze the budget further by replacing the SR-VS20U with a ZR-20, dropping the bottom line even further (I'd say do this only if you're really pinched...).

Approximate Budget -iMovie 225GB internal, camcorder feeder

PowerMac G3/400 w. 256MB ram (used)	900
(3) IBM 60GB Deskstar hard drives @ $250	750
Sonnet Tempo ATA66 controller	100
(4) Canon ZR20 DV cameras @ $550	2200
total budget	3950

4.3: How Low Can You Go?

The sad fact is that at many colleges where video production courses are offered, a budget sufficient to teach them properly is simply not available. By 'properly' I do not mean the replication of professional facilities or the availability of 'industry standard' gear. Rather, I mean the ability for every student to gain enough access to some sort of usable production tool they can use to develop their creative abilities in the medium. That is, the issue is less the sophistication of the equipment than whether there is enough of it to go around.

Having held a variety of short-term positions at different colleges between 1988 and 1998, and subsequently being on the job market seven of those ten years — going out on no less than twenty-six on-campus interviews — I was able to see a fairly representative cross-section of production education at U.S. colleges. It wasn't a pretty picture. Many schools maintained a three-camera TV studio for classroom use. However, running crew rotations through live interview programs is not my idea of moving picture education. Whether they had a studio or nor, most schools had only a couple field cameras for students to use, and one or two VHS edit systems, often shared with a media services department, available to students only during the limited time that media services was open *and* not using the gear themselves. This just isn't enough access for students in a typical class of 12 to 16 to work on projects individually, or even in pairs. Thus, the instructor needs to have the students work in large groups, where typically one or two people do all the work and nobody else learns anything. Again, not my idea of media pedagogy.

So I would always find myself asking the Dean or Chair or Vice-President or whoever was in charge whether there was any money available to acquire some camcorders and simple edit systems. Sometimes the answer was a straightforward "No!" but more often the answer was, "Maybe, a little..." meaning maybe a few thousand dollars. Now, in those days I figured $25,000 to $30,000 as the bare minimum to get a decent program going, using JVC Edit Desk editing systems and Panasonic SVHS camera, figuring 2-3 new editors and 3-5 new cameras. So 'a little' wasn't really enough.

Although I was invited for many interviews, I was rarely offered a job. Strangely, my lack of success seemed to correlate with the number of times I told administrators they would need to spend more money than they had planned in order to teach a new production course properly. This has been my career crisis: most schools with openings in production simply do not have the resources to teach production the way I believe it should be taught. So what do you do? Stay unemployed? Resign yourself to teaching a crappy class?

Well, one of the things I've done is spend an inordinate amount of time trying to find ways to stretch 'a little' money in acquiring facilities. Assuming, as a given, that there will never be enough money to buy enough new equipment to give all the students reasonable access, I have spent countless hours researching the questions, "Just how cheap can you get by? Are there any atypical products or configurations that can do the job at a lower cost?" I have scoured the Internet, making daily visits to video-technology newsgroups, gone to numerous trade-shows and grilled manufacturers representatives. I have also done a great deal of tinkering, assembling different systems at my own expense to get some practical idea of how they would work in an educational setting.

Where the Deals Are

The general operating mode for school purchases is always to buy new equipment from fairly large vendors who cater to the education market and accept POs. This doesn't really work for building a computer video system on the cheap. We need the option to buy the equipment that will work the best for our program and budget, even if that means obtaining used gear from an individual private party.

Once you get the power to buy used equipment, a whole world of low-cost video opens up, thanks to eBay. This market is not without peril, and looking for used gear extracts a high price in terms of the time it takes to locate the right stuff and make deals to acquire it. Via no other means, though, can you acquire so usable a facility for so little money.

The next best thing to eBay are Internet price-search engines, which let you locate the lowest available prices on new gear, and surplus dealers who have set up shop on the web. Shopper.Com, MySimon.Com and Pricewatch.Com are excellent sources for finding discount vendors for anything computer-related. Deal-Mac.Com and Ramseeker (macseek.com) also offer useful info for Mac-based shoppers. You'll need to do some searching to find surplus vendors who have the sort of thing you need, since their stock is very here-today-gone-tomorrow. (I've found some good deals at computergeeks.com, and www.halted.com is a good source for cables, cases and other parts for systems using older technology.)

Then there's scavenging. High tech companies and some schools with high tech programs use their computer equipment until it becomes obsolete, and may just give it away or in some cases dump it with a 'computer recycler.' (This is in contrast to most colleges and universities, which use stuff until it falls apart.) Given the relatively modest cost of system 'B' above, the budget systems described below should appeal most to programs with a certain amount of the 'older technology' prescribed already on hand or available in the community at small or no expense.

H. The Original Bargain Basement NLE — Analog Offline/Online

Systems 'F' and 'G' are designed to be simple to use, simple to assemble, and low in cost. They are good for situations where the faculty and staff who will teach with and support the system have a low geek factor. The problem is, they just can't accommodate many students. It's possible to expand the number of users yet stay within a very modest budget, to acquire two stations for a about the same price as one 'F' or 'G' type station. However, the geek factor goes up considerably — as we enter the world of offline/online editing, and begin working with some out-of-the-mainstream components. I do not recommend any of the remaining systems unless the faculty or staff person in charge has used some sort of offline/offline system before, has assembled some sort of computer video system before, or is very brave and has lots of spare time.

I spent much of a year between 1998 and 1999 researching and experimenting in an attempt to come up with an ultra low cost NLE system that actually worked. My early attempts, though they seemed sound on paper, ended in failure. Finally, in the summer of 1999, after much trading on eBay supplemented by key purchases from surplus vendors — I achieved success. While I will describe my Original Bargain Basement prototype (OBB Sr.) in this section, the steady advance of the computer market has brought more goodies into the range of our mini-budget, as components become available second-hand. Thus, since in the years since creating OBB Sr. I have also devised a Son-of-OBB (Jr.), to be discussed in the next section, and even an OBBIII, to be discussed in the section after that. OBB Sr. is capable of somewhat better online image quality, and is probably more stable than Jr. in the long run, while BBIII definitely represents a step-up in image quality, though it makes other compromises.

All three of these systems should only be considered by programs seeking to make use of an existing stock of analog camcorders. The more usable gear you have on hand, and the better its condition, the more attractive these systems should be. If you're starting from scratch, or even if you would need to purchase a couple Hi8/SVHS camcorders and feed decks to flesh out these systems, a digital tape based offline/online system makes more sense. This is probably the best overall choice for anyone on a tight budget needing to cover more users than a single system will handle. I guess we have to call it New Budget Basement (NBB), and it will also outlined in section 'K' below.

I have placed OBB Sr. first in the discussion because the story of how it got built is the most illustrative of the process involved in building a low budget NLE, and introduces several points that will come up later in discussing other configurations.

As a budgetary move, especially within the technology available at the time, I decided a low-cost system would need to be capable of lo-res offline editing, and would have to be based on a scalable codec. My initial approach to building the core of a low cost system was to acquire a low-cost M-JPEG digitizing card, and then find the least expensive computer that would support the card properly. Matching a card to a computer is no easy matter, and this is a prime area where the card manufacturer's specifications simply cannot be trusted. Not that the makers tell outright lies, but the system requirements for a digitizing card may list it as being compatible with any computer in which it can do any kind of video work at all — including the kind of small-frame, slow frame-rate video used in CD-ROM or on the Internet. That is, the manufacturers may base their compatibility list on something far less demanding than 640x480, 30 fps NTSC that can be output to tape. Low end M-JPEG cards achieve their degree of performance, and their lower price, by shifting much of the workload to the CPU of the computer. This means they need a pretty powerful host machine to do 'real' video editing. While current computers may easily fit the bill, I was looking at older machines to keep the cost down, and that was where things got tricky.

The first card I acquired was an Aurora Fuse. This is a Mac-only product, sort of the little brother to the Igniter card specified in system 'D' above, which had received good initial press. I installed it in a PowerMac 9600, which had been a top-of-the-line Mac only six months before. (I had actually been hoping I could get it to work on a cheaper, less powerful Mac, but I used the 9600 for the test because it was what I had on hand.) The results were disappointing. The overall system simply could not process data fast enough to make Premiere run smoothly handling 640x480/30fps. At first I thought I just might not have set up the system correctly, but after several not-very-illuminating e-mails with Aurora tech support , I concluded I had the proper settings entered. I also concluded that the Fuse — which in other respects I found to be a well-thought out and executed product — really needed to be housed in a G3 to do 30fps, full-frame video. G3 machines were new and still quite costly at the time, and seeking to preserve my minimal cost concept, I sought another strategy.

I had purchased the Fuse from the internet vendor that offered the lowest price. I had so hoped it would work, I had not thought ahead to what I would do if it didn't meet my expectations. When I called the vendor seeking an RMA to return the Fuse, I discovered that since I had had it just over a week, they would charge me a 35% restocking fee. I had no choice but to swallow hard and take the loss. (This vendor is now out of business). The moral of the story, unless you *know* the item you are purchasing will work, buy from a vendor that will allow you to return it, even if that means you pay a bit more. For any expensive item you buy on eBay, you should request iEscrow. This protects you from unscrupulous sellers by allowing you inspect merchandise before paying for it. There's a small fee for the service, usually paid by the buyer (i.e. you).

The major caveat to using eBay to acquire components for an editing system is simply the considerable amount of time and effort it takes to participate in the auctions. In addition, since you're now buying your system in parts, rather than as an already assembled whole, you have to know what you're doing, and there's even more time involved in putting it together, troubleshooting, and so on, and. All things being equal, all this effort may *not* be worth the money saved. However, the premise here is that you just don't have the money, and you're desperate enough to get some usable tools for your students that you're willing to spend the time. This, in fact, is my motivation for writing this article. Having spent a great deal of time on my quest, I want to share what I've learned so that others might be enabled to do something similar in far less time.

Having concluded that 1) a low-cost M-JPEG card in a low-cost computer didn't work well enough, and 2) a low-cost M-JPEG card in a good enough computer was going to be too expensive, I turned to eBay to pursue my alternative strategy — putting a better M-JPEG card in a cheaper computer. This more or less reflects the path originally taken by Avid and Media 100, who after all, had devised hardware components that could handle high-quality video in old NuBus Macs with 040 processors. My specific target on eBay was the Truevision Targa series of M-JPEG boards. The Targa 2000 PCI (hereafter T2K) was developed in the mid-'90s. It was designed to be capable of 'broadcast quality' output, using the computers available at the time. As such, it handles far more of the processing of the video signal on the card itself, and makes much lower demands on the CPU. Unlike Media 100 or Avid, it is not an integrated hardware/software system, but an open architecture component that will run Premiere. It was widely used by professionals in its time, and is rock solid stable. T2K boards appear regularly on eBay. At the time I built the OBB prototype T2K boards were going between $900 and $1500 at auction. (By 6/01, as DV advances continue to squeeze out demand for M-JPEG, prices had dropped to $300-$500.)

I bought a Targa on eBay, in a package including an unspecified Mac as host. When I received the package, I discovered the Targa was installed in a Motorola Starmax 3000/180 Mac clone — a machine *not* listed as being supported by Truevision. However, the system worked fine, far superior to the combination of the Fuse and the 9600. I was able to digitize, edit and playback using Premiere 5.1 without a hitch. I later verified that this was no fluke by installing the board in another Starmax, also with excellent results. The computer had been all but a throw-in in the auction, and StarMax machines continue to have the lowest value in the Mac resale market.

There are several models in the Targa line. The stock Targa 2000 PCI is the board of choice here. It has s-video input and output connections. There is a T2K Pro model that adds Betacam component in/out — but we're assuming our bargain system is not using Beta source material! Further up the scale are the Targa 2000 RTX and DTX which add higher data rates, real-time/dual-channel video streams and other high end features our student lab doesn't need. There is also a model *below* the T2K: the Targa 1000 (aka T1K). The primary difference between the two is that the T2K supports a second computer monitor, and will display video playback on that monitor at actual resolution. The T1K does neither. Since the T2K worked so well, I wondered if the T1K, which could be found somewhat cheaper, would also do the job. I acquired one, and determined that it was not an adequate substitute, at least in the context of my test system. By drawing the playback window itself, the T2K takes a great load off the CPU. Even with the tiniest window size chosen for the T1K, the system remained sluggish, where operation with the T2K had been smooth.

(Like the Fuse card, the T1K would do better in a more powerful computer. In fact, with a Blue and White G3 or newer Mac, the second-monitor support of the Targa 2000 becomes a kind of disadvantage. The Targa draws the screen more slowly than the high-speed standard video card, dragging down the overall performance of the system in every task other than displaying the video window overlay. While the Targa 2000 is the much superior card in this all-MJPEG system, the Targa 1000 would actually preferable in a system designed to go from low-res MJPEG offline to DV online, built on a Firewire native G3 or G4.)

With the StarMax working so well, I made no move toward a more expensive host computer for my prototype. One reason I might have done so though, is that by using a clone I probably cut myself off from potential tech support assistance from the campus IS staff — the Apple experts at most colleges are strictly *Apple* experts, and will usually refuse to touch a Mac clone. The Starmax series was based on an Apple motherboard, but a rather oddball one — the PowerMac 4400, which uses a different sort of memory than any other Mac.

For online mode storage, I added an 18.2GB ultrawide SCSI drive, driven by an ATTO PCI-PSC ultrawide PCI SCSI host adaptor. The built-in SCSI port on the Starmax is too slow for good-quality video, and the IDE ports are too slow to support UltraATA. When I constructed my prototype bargain system in summer 1999, I was able to buy a used 18.2 UW SCSI drives on eBay for about $400. A year later, *new* drives of that size from reputable manufacturers were selling at street prices under $250.

Truevision/Pinnacle recommends Atto SCSI cards for use with Targa boards. Atto PCI/PSC are regularly available on eBay or from surplus/discount resellers on the internet for $100 or less. These are usually pulls from OEM systems, without documentation or software, and often with older firmware. However, the firmware is on a flashROM, easily updatable, and Atto keeps up-to-date drivers, utility software and ROM flashers available on their website. This continuing strong support for the card makes it an excellent value.

For offline storage I used an Orb drive, in this case an external SCSI model. Installed in the Starmax, I was able to get the Orb running quite well off the Atto card, though I had to adjust the speed of the Atto down using the Atto tools software. (If an interface tries to transfer data faster than the drive or bus allows it can bottle-neck the system and actually cause performance to decline.) *Note, as I will discuss below, that the Targa/Atto/Orb SCSI combination did not work properly in Yosemite (Blue and White) G3 or newer Macs.* Again, I recommend you find a way to use small removable hard drives — either purchased directly by students or supported through student fees — as an alternative or addition to the Orb for offline storage. This is best incorporated into the system, by using a mini-tower Starmax or PowerMac, which should have an open 5 1/4" drive slot. (The more horizontal desktop case models do not.) You would place a ViPower mobile rack in this slot, and use an ACARD bridge board to attach the mobile rack to the ATTO's SCSI chain. See the Appendix section on adding removables to a Mac for more info about these items.

One of the nice features of the T2K is that it supports a second computer monitor, as well as providing output for an NTSC monitor. A video editing application takes up a lot of screen space, and I highly recommend a two-monitor setup. Ideally, the main monitor should have a 1024x768 screen for Premiere — comfortable on a 19" CRT, a little squinty on a 17". You can get by with 832X624, though, and in setting up the prototype I used 15" monitors as a cost factor. I was able to buy decent used 15" monitors for $55 apiece from a local computer recycling company. This was a rather unique opportunity afforded by my location in the San Francisco bay area. The recyclers let me poke through their stock, test the monitors by plugging them into a Mac — and pull out the few decent ones. Unless a similar congenial source is at hand, you will want to buy new monitors. I most strongly urge you to *never* buy a monitor on eBay. First of all, they're heavy — you'll pay far too much to have it shipped. They're also fragile internally, and don't travel well through freight systems. More importantly, more used monitors than not have awful pictures — bad focus, bad geometry, bad luminance.... *Never* buy any computer monitor new or used you have not actually seen work, or that you cannot easily return. The best move then for most people is to shop local dealers for a low-end monitor that looks decent. Some of the generic units available stink, but some are pretty good and constitute excellent values. You just have to look around — adding more of your time and energy to the project.

Two things that I did *not* have to buy for the prototype system were an NTSC monitor and a copy of Premiere. My college already had multiple Premiere licenses bought as part of a general Adobe package, and the AV department had a stock of old but still functioning video monitors. I suspect the situation at many campuses is similar.

When I created my prototype system in 1999, DV cameras were too expensive to fit the rock-bottom budget mode. I rejected VHS and SVHS because of the difficulty of getting timecode capability in the format. So I wound up with Hi8 models equipped with RCTC. (My college at the time already had a few of these, so it was a natural.)

The value of Hi8 equipment has dropped dramatically with the introduction first of mini DV, then digital 8. Camcorders are the sorts of things some people purchase with high expectations but wind up using very little, and it's possible to find used camcorders on eBay in excellent condition at decent prices. Only a few Sony and Canon Hi8 camcorder models were equipped with the RCTC time-code system, but these were not necessarily extravagantly priced to begin with. Used models with RCTC should be available between $250 and $500 (6/01), depending on age and other features, if you can find them.

For a feeder deck to the edit system, either the Sony CVD-1000 (VISCA interface) or the Sony EVO-9850 (RS-422) will read and write RCTC. As I noted earlier, one of the nice things about RCTC is that it be striped onto an existing tape without erasing any material already recorded on it. This way, if students happen to have their own Hi8 cameras without RCTC, they can still use their tapes in the edit system by engaging this extra step. Used CVD-1000s should be under $1000, used EVO-9850s between $1000-$15000, though again both of these models may be scarce in the used market (prices ~ as of 8/2000). I used a CVD-1000 in my prototype system, though the 9850 is definitely preferable in features, quality and for having RS-422 device control. Note: Premiere does not include device control software for non-Firewire VCRs. This function is performed by the Pipeline ProVTR plug-in. ProVTR comes with a cable to connect the Mac to the deck — you would need to specify the VISCA or RS-422 interface when ordering it, depending on which VCR you acquire.

There are several drawbacks to building a system around Hi8. Although Hi8 cameras generally produce very nice images, the quality is not as good as digital, and Hi8 tape is subject to a notable amount of dropouts. While used Hi8 camcorders in excellent condition may be found without great difficulty, recording decks (the CVD-1000 or EVO-9850) in good condition are harder to come by. They are not only more rare to begin with, but many of the used units available have seen heavy institutional use. Overall, locating enough equipment in proper condition is likely to be yet another time consuming activity — probably requiring bi-weekly checks of eBay over a period of several months. Finally, some of this equipment is just old enough in the fast-moving world of consumer electronics obsolescence that it may now be expensive to repair, if parts are still available at all.

So, the complete system slightly updated, would look something like this, with approximate prices of what these things have sold for on the Web lately (6/01). I have listed items I suspect you may have already and thus not have to buy in italics, and provided totals that both include (in italics) and leave out these amounts.

Targa 2000 PCI board (used)	400
Atto PCI PSC UW SCSI card (used)	100
StarMax mintower	
upgraded to 96MB RAM (used)	225
18.2 GB UW SCSI drive (new)	225
Orb Drive, SCSI ext.	150
Pipeline ProVTR device control software	200
(2) 17" Monitors @ $150 ea. (new)	300
Sony EVO-9850 Hi8 deck (used)	850
Removable drive bay	90
Adobe Premiere 5	*350*
Panasonic 13" TV w. line inputs	*160*
price per edit station	2540 *(3050)*
(2) edit systems as described above	5080 *(6100)*
(4) Hi8 camcorders with RCTC	
(used @ $400 ea.)	*(1200)*
total budget for 16 student class:	5080 *(7300)*

I actually built not one but two of these, tested them, and they *did* work. However, my tests were relatively limited. I did not have the opportunity to see how the systems would hold up over time under regular use by students, since — due to rather complicated academic politics, the school I built the prototypes for declined to use them. I was especially curious to see how the Orb drives would hold up over time, and how the system would handle the batch redigitizing as students remade their low-res projects into hi-res online versions. Unfortunately, with no opportunity to put them in place, I wound up disassembling them and selling back the unwanted components on eBay.

In the time since then, new lower-priced digital camcorders have appeared, as well as reasonably priced computers with built-in Firewire. Thus, I would not recommend this configuration to anyone starting from scratch at this time — though it would remain useful for anyone planning to use an available stock of analog cameras, especially if you already have time-code capable feeder decks on hand.

Evaluation

Strong Points: Good image quality. Targa is stable, proven hardware with good support. Orb drives allow multiple users on the system at low cost to them and no added cost to the school. Premiere is enough of a standard that some users will already be familiar with it. The basics are easy to learn, yet the software allows advanced conceptual complexity. ProVTR device control should yield accurate offline/online recaptures.

Weak Points: More expensive than low-cost digital system unless you already have some usable analog camcorders and or VCRs. No tech support available for Starmax. No real time. No DV capability. Analog format equipment may be at the end of its service life. Having everybody work offline is a bit messier technically and conceptually. Redigitizing and advanced techniques are more awkward on Premiere than on FCP.

Alternatives

You might abandon the Starmax and move to an Apple-brand computer for support reasons. Not that the Apple will work any better, just that you probably have an easier time finding someone to talk to if you have a problem. You would need a PowerMac of similar or better processing power — a 604/120 or better should do. This makes the PowerMac 8500, 9500, 7600, 8600, 9600 or 7300 logical candidates, with the 8600 probably the best combination of power and value. (I have not tested the Targa/Atto combo in any of these machines, but if it works in the Starmax, which is *not* certified by Traversing, you'd think it would work in the Macs that *are*...) This summer (01), I was able to scrounge a working 7300 through a connection at a local junior college that was giving old equipment away...

You might use Hi8 cameras not equipped with RCTC, in which case students would need to take the extra step of post-striping the time-code with the feeder deck. You might also use VHS or SVHS cameras in a similar fashion, in which case you would need to employ the JVC SR-S365U as the feeder deck. However, a better choice for using VHS/SVHS source material is probably system #4 below.

I. Sub-Basement: The Worlds Cheapest Multiple-User NLE, Analog Offline/Online

This system is for people with *very* limited budgets, a good amount of technical skill, and a sense of adventure. It uses a video digitizing component for which little or no useful tech support is available. It's not really upgradeable. It's not as good in terms of image quality or stability as the system just described. It *should* be solid enough for teaching purposes, but if you can possibly afford one of the other systems, don't even think about this. This is really an exercise in scraping the bottom of the barrel. Again, this is a system for someone trying to make use of an existing stock of analog camcorders, and anyone starting from the ground up would be better off with the budget digital offline/online system described below.

The "World's Cheapest NLE" is based on an Iomega Buz card installed in a first generation (beige) Mac G3 computer. An internal fast-SCSI hard drive provides online storage for conforming, while an IDE Orb drive provides for project construction in the offline mode. Backup copies of timeline files can be made with the Mac's floppy drive (the beige G3 being the last Mac to come equipped with a floppy). The editing software is Adobe Premiere 5.1. Again, two monitors are used to provide a suitable screen workspace, one connected to the on-board video of the G3, the other to an additional PCI video card. Source material is fed from a Sony CVD-1000 Hi8 deck or a JVC SR-S365U SVHS machine, using the Pipeline ProVTR device control plug-in for Premiere. These decks are specified since they are capable of striping timecode onto existing tapes. They are likely to be the most expensive parts of the package. This system would be especially practical for anyone who already has one or two of these VCRs on hand. By the same token, the beige G3 is now old enough that you may be able to acquire a hand-me-down from someone on campus who has upgraded to a G4, dropping the budget even further.

The Buz was a combination M-JPEG digitizer and ultraSCSI (narrow) adapter card that Iomega once sold in both Mac and PC versions at low prices — around $200 for the PC version and $300 for the Mac version. The Buz was very popular with PC owners due to its low cost, but it became one of the most hated products in computerdom since few of its owners could get it to work properly. The drivers Iomega supplied with the PC version were hopelessly buggy. Storage, not video, is Iomega's core business, and their engineering staff apparently had neither the time or the ability to address the problems adequately. After a couple attempts at revised drivers that proved little better than the originals, Iomega just gave up and discontinued the PC Buz, leaving many frustrated and angry owners waiting for better drivers that would never be developed.

The Mac version of the Buz was much more stable. The quirk here is that it would only work in beige G3 machines. It would not function in any earlier machines, even with G3 CPU upgrades, and when the newer Blue and White G3s appeared, the Buz hardware proved to be incompatible with them (the computer simply would not boot with the Buz installed) — a fact that Iomega was astonishingly slow to realize and address. Iomega eventually released an upgraded version of the Buz for Mac supposedly compatible with the Blue G3 and the Yikes G4 — I have no idea if it works with newer Macs and don't care, since this is ultra-low-budget project only and we're staying with the cheap stuff.

In almost all the PC Buz packages sold, the hardware was identical to that used in the Mac package. The only difference was the drivers and the bundled software. In other words, all you need to get a PC-packaged Buz to run on a Beige G3 is the Mac drivers and a copy of Premiere 5. Your school probably already has a copy of Premiere. While Iomega has never made Mac drivers for Buz available on their website, the latest drivers for the original Mac hardware (that is, the hardware shared with the PC version) are available at the BuzInfo site at http://www.geocities.com/ResearchTriangle/Lab/7357/software.html. (There may be new drivers in the packages designed for the Yosemite and Yikes Macs).

Since there were lots of PC Buzes sold, and few that ever wound up working well, there are lots of them available quite cheaply on eBay. I purchased a used PC Buz on eBay ($65), and installed it in a beige G3/233 using the drivers from the BuzInfo site. The system ran Premiere 5 without sluggishness, using either OS 8.1 with Quicktime 2.5 or 3.02, or OS 8.6 with Quicktime 3.02 or 4.03. Results were slightly better with the older OS/Quicktime combination. The Buz/Premiere/Quicktime combo showed some quirks — sometimes the capture window in Premiere appeared in the wrong aspect ratio, and captured clips sometimes showed blank screens in their immediate playback windows, although they played fine once placed in the timeline. None of these oddities affected the actual usability of the system. I also cannot say whether they are products of the this configuration generically, or just my specific test system. I only had one beige G3 available in which to try out the configuration, and it was somewhat battered and underpowered (it had only 66MB RAM, and 128Mb would be a more practical figure). An earlier, more limited trial with a different Buz card in a different G3/233 did not show these problems.

At it's higher data rates, the Buz produces good quality images, though not on a par with a Targa card. With the settings applied after installation, the Buz tends to push the contrast too hard, and blow out highlights a bit — nothing immediately apparent, but not hard to see if you take a critical look at the output and source images side by side. This effect can be countered to a considerable degree by adjusting the Image settings under "Video Input" in the Capture mode of Premiere.

(My Buz card also introduced a bright green horizontal line 2 or 3 pixels high across the bottom of the raster, when installed in a Yosemite G3. On a properly adjusted NTSC monitor, this would never be visible, being well within the area masked by the CRTs overscan. I discovered it because I had an underscan monitor connected to the G3. I noticed no problem when the Buz was installed in the beige G3, but I was viewing the image through a normal monitor. I do not know whether this flaw is generic to all Buz cards installed in a Yosemite, or a defect in the one I acquired. I'm guessing the later, but I cannot be sure.)

For individual storage we will again use a combination of small removable hard drives and Orb disks. Again, we also include a large internal SCSI drive for conforming online versions of projects. Since we're in extreme budget mode, we will forego an additional UltraWide SCSI card, and use the SCSI port on the Buz to connect the hard drives. The SCSI-2 interface on the Buz is faster than the built-in SCSI on the G3, which is only the standard 'slow' SCSI-I. Though the Buz port, being 'narrow' SCSI, is not capable of the same data rates as Ultrawide, it should still provide usable online resolutions as long as the connected hard-drive is fast enough. (Note: attach the Orb to the built-in SCSI port, not the port on the Buz.)

Again, in the approximate budget below I have listed things I expect you may have on hand already in italics — a copy of Premiere, an NTSC monitor, and analog camcorders. Again, let me note that if you have to go out and *buy* all this stuff, I do not recommend this system at all, since at this point significant investment in analog camcorders is a poor decision. The only possible reason you might want do this is if you can find several RCTC equipped used Hi8 camcorders at a price far enough below digital models to make a compelling difference in a tight budget. Nevertheless, I have figured the total system based on Hi8, though it could be used with VHS by using the JVC 365 as a feeder. This makes sense if and only if you have decent VHS cameras on hand. Again, I have also not included tripods, external microphones, carrying bags for the camcorders or other incidentals in these rough budget figures. Again, you may have some of these things on hand already...

Buz card (used)	65
PowerMac G3 mintower (beige, used) w. 128MB RAM & 4MB video RAM	400
18.2 GB SCSI drive (ebay)	125
Orb Drive, SCSI external	150
Pipeline ProVTR device control software	200
Generic 19" monitor	250
Sony EVO-9850 Hi8 deck (used)	850
Removable drive bay	90
Adobe Premiere 5	*375*
Panasonic 13" TV w. line inputs for NTSC monitor (new)	*160*
price per edit station	2135 *(2670)*
(2) edit systems as described above	4270 *(5340)*
(4) Hi8 camcorders with RCTC (used @ $400 ea.)	*1200*
total budget for 16 student class:	4270 *(6560)*

Evaluation

Strong Points: Decent image quality. Orb drives allow multiple users on the system at low cost to them and no added cost to the school. Premiere is enough of a standard that some users will already be familiar with it. The basics are easy to learn, yet the software allows advanced conceptual complexity. ProVTR device control yields should yield accurate offline/online recaptures. G3 Mac is supportable platform.

Weak Points: No real time. No DV capability. Analog format equipment may be at the end of its service life. Having everybody work offline is a bit messier technically and conceptually. Advanced techniques tend to be more awkward on Premiere than on FCP. No support of any kind for Buz. Buz/ Quicktime/Premier combo exhibits minor quirks.

Alternatives

I have included the cost of an ACARD IDE-SCSI bridge board in the budget to support the removable hard drives, but this may not be necessary, depending on the revision of the G3 minitower. Some of these computers support only one drive on the IDE ports (the system drive, and the CD-ROM drive). These machines are equipped with SCSI internal Zip drives, or connectors for a SCSI Zip. These would need the bridge-board. Later versions of the beige G3 supported 2 devices on at least one of the IDE ports. This would allow you to connect an IDE mobile rack directly to the IDE port in the slave position (if a Zip is installed, you might have to remove it to gain access to the bay).

J. Low Cost Analog Offline —> DV online (for VHS camcorders) (not tested)

This package is designed for schools who have a large stock of usable VHS or SVHS camcorders on hand now, but want to prepare for DV in the future, and avoid further investment in 'old' technology.

The feeder in this system is a JVC SR-VS20U dual transport deck, with built in SVHS to miniDV. Students would need to dub their VHS footage to miniDV as a first step, in order to stripe it with timecode. The computer is a G3 running FCP, equipped with a Targa card for low-res capture. With the analog outs of the JVC deck connected to the Targa, students would capture from the DV dub to small removable hard drives or an Orb disk, using the Firewire device control built into FCP. Final creation of the online master would be done purely in DV, recaptured to a large fixed IDE drive inside the computer. Note, that the total budget for this package is higher than system 'K' below, which includes new digital camcorders, but does not include feeder decks. 'K' makes better sense if you're getting one modest chunk of money now, but a dry well for several years after. 'J' here makes sense only if you can reasonably expect the funds to upgrade to DV cameras in the next couple of years.

PowerMac G3 400 w. 384MB RAM (used)	900
Targa 1000 (used)	350
45GB IBM ATA66 drive	150
Orb Drive, EIDE internal	150
1-bay external removable drive case (w. SCSI card)	290
Generic 19' monitor	250
JVC SR-VS20U DV/SVHS deck	1500
Final Cut	250
Panasonic 13" TV w. line inputs (new)	*160*
price per edit station	3,840 / *4,000*

(2) edit systems	7,680/*8,000*
camcorders (on hand)	*NA*
total budget for 16 student class	$7,680/*8,000*

Evaluation

Strong Points: Upgradable. Ready to accept DV camcorder source. Orb drives allow multiple users on the system at low cost to them and no added cost to the school.

Weak Points: Mediocre image quality from VHS source (though no degradation involved in editing). Required dub from VHS to DV is time consuming. No real time. Having everybody work offline is a bit messier technically and conceptually.

Alternatives

You can make a small dent in the budget by replacing the SR-VS20U with a combination of a standard VHS tabletop deck and a ZR-20 camcorder — making the system somewhat less user friendly and less reliable as a trade-off. This would bring the price of this setup closer to the costs of OBB Sr. Again, you might want to add a removable drive bay to substitute smaller IDE hard drives in mobile racks for the Orb, as a reliability upgrade and removing the necessity for conforming to create a viewable project. (Higher quality offline). Have students buy or rent the hard drives to keep the budget down. See the section in the appendix about adding removable drives to a late-model Mac.

K. Ultra-Low Cost Analog Offline —> DV Online NLE

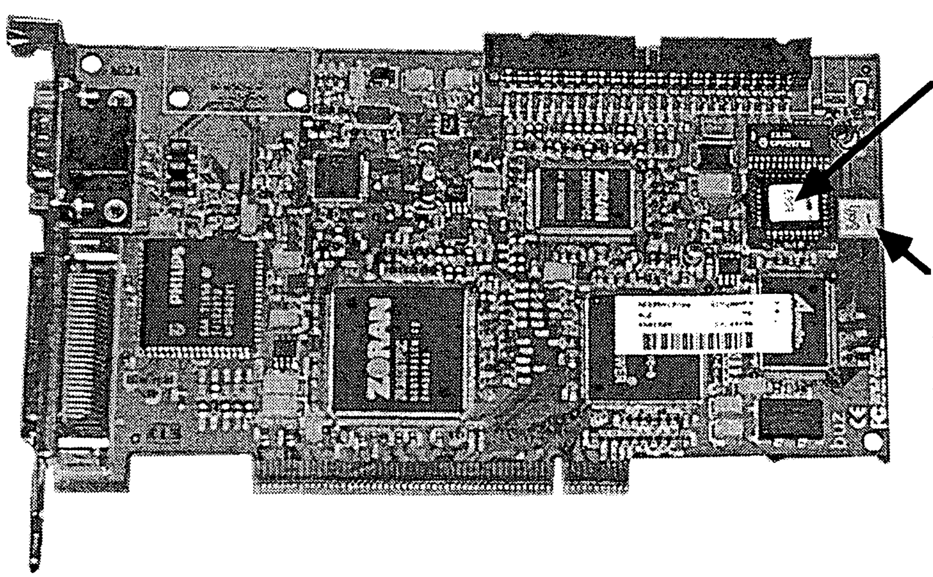
If you have to start from scratch, or *can* start from scratch, buying new camcorders as well as edit systems, you can create an offline/offline system of significantly better quality for a low price using DV gear if you're willing to make certain compromises.

The first compromise is to forego the convenience of dedicated feeder VCRs with the edit system. Students will capture footage and lay-back finished projects by using a camcorder, checked out on a temporary basis for that purpose. The online FinalCutPro DV system described back in section 'B' requires each station to be equipped with a dedicated feeder VCR, since the VCR does the D/A conversion that allows the editor to see playback on the NTSC monitor. However, in an offline mode system, the VCR doesn't absolutely need to be present at all times, only during capture and final output of the online version back to tape. The digitizing card creates the NTSC output used during editing.

Students will shoot with digital camcorders, and digitize from the analog outs of the cameras to low-res M-JPEG stored on an Orb cartridge or small removable hard drives. Firewire device control will control digital cameras or decks whether the software is capturing DV through Firewire or another codec through a digitizing card. Students will create the project, then batch re-capture to a fixed hard drive in digital mode via Firewire and export the all-digital conformed version back out through Firewire to a digital tape master on the camcorder again.

The downside of this method is that it puts extra wear on the camcorders, and makes them unavailable for shooting while they're in the lab as source decks. Thus, for this system we spec an extra camcorder, with the idea that this unit will have a priority for lab use, though it can be checked out for shooting when not reserved for the lab, and the rest of the camcorders would have the reverse. If there are periods when the whole class is shooting and not working in the lab, the camera can go out on shoots. When the group moves en masse to capture and begin editing, the other cameras can be drawn into the lab. As before, the Canon ZR-20 is our low-end DV camera of choice.

The second compromise is the use of an unsupported MJPEG capture card. This reduces your options for getting help if something goes wrong. Again, the daring technologist returns to the Buz card. The limitations of its's image quality don't much matter in this configuration, since it is used only for the offline version — the online version will be pure DV and won't pass through the Buz at all. There's a trick required here. We want to build this system on a Yosemite G3 so we have native Firewire support for device control and DV capture. However, as it comes from the factory, the Buz is incompatible with this computer, and will cause the machine to hang at startup. Fortunately, the problem lies in the SCSI adapter part of the card, not the digitizing function. (Apple changed PCI chips between the beige G3 and the Yosemite. The SCSI BIOS in the Buz is compatible with the older Apple PCI spec, but not the newer one.) If you remove the BIOS chip on the Buz the SCSI part of the card is disabled and it will no longer mess with the operation of the computer. It will operate simply as a capture card, and properly so. The chip is in a socket; you can just pry it out with a small screwdriver.



SCSI BIOS chip pry this out to use the Buz with a Yosemite G3

I put a small paper label here, marked to remind me how to put the chip back in

With the BIOS chip removed, I installed the PC Buz I bought from eBay in a Yosemite G3, and successfully captured and played clips using the BuzInfo drivers, Premiere 5 and Mac OS 8.6. (Results with Premiere on a G4 Sawtooth, or FCP on a G3 were unacceptable.)

The third compromise is using Premiere 5.1, which is the only editing software with which the Buz is fully functional. The process of moving from offline to online versions is more complicated in Premiere than it is in Final Cut.

That's a fair amount of compromise, but the point is the price, and it's down there... We'll assume again that you can scrounge NTSC monitors, and that your school has an old Premiere 5 licence you can use.

PowerMac G3 400 w. 256MB RAM (used)	900
Buz card (used)	70
45GB IBM ATA66 drive	150
Orb Drive, EIDE internal (mount in Zip bay)	150
Generic 19" monitor	250
DIY external case for removable drive	160
Premiere 5.1	*350*
Panasonic 13" TV w. line inputs (new)	*160*
price per edit station	1,680 / *2,190*
(2) edit systems	3,360/*4,380*
5 Canon ZR-20 camcorders @ 550 ea.	2,750
total budget for 16 student class	$6,110/*7,130*

Evaluation

Strong Points: Very good online image quality via DV. Digital gear good investment for future. Orb drives allow multiple users on the system at low cost to them and no added cost to the school. Premiere is enough of a standard that some users will already be familiar with it. The basics are easy to learn, yet the software allows advanced conceptual complexity. Stable enough not to get in the way of projects getting completed.

Weak Points: No real time. Having everybody work offline is a bit messier technically and conceptually. Buz/ Quicktime/Premier combo may exhibit a few minor quirks. Capture and mastering on camcorder more awkward than use of dedicated feeder deck.

APPENDIX

A1: G3/G4 Genealogy

Apple doesn't always change the name or number of a computer when they make important changes inside the box.

Original (Beige G3)
The original PowerPC G3 computers (the cases are actually platinum, not beige, but they're the off-white ones) came with 233Mhz or 266MHz processors, in mini-tower or desktop configurations. All models have a 66Mhz system bus, built-in SCSI, and conventional Mac serial ports (a serial port is needed for RS-422 or RS-232 device control). Firewire can only be added via an add-on PCI card — these are not great models for DV based systems. As host for an M-JPEG editor, the mini-tower is by far the preferable model, since it has more room for expansion, including an open 5 1/2" drive bay in the front where a mobile rack can be installed. Not to be confused with the different *models* are two revisions of the motherboard. All beige G3s come with two IDE ports. However, on the first revision motherboards, each port only supported a single drive — thus two total for the system. These were already used by the system drive and the CD-ROM drive — leaving no ability to attach another IDE drive (without adding a TurboMax card). The revision two motherboard, though, *does* support the standard master/slave IDE configuration, making it far more useful for our low cost editing system. The only way I know to tell the two revisions apart is to look inside the case. On computers equipped with a Zip drive, if the Zip is SCSI its a rev. 1 board, if it's IDE it's a rev. 2 board. If the machine does not have a Zip drive, it should have a cable running to the spot where the Zip would go. If it's a SCSI cable, it's revision 1. An IDE cable indicates revision 2. You would install either a mobile rack or an IDE Orb drive in this position, so you would probably remove the IDE Zip if you have one to free up the port.
Look for a 266 Minitower with a rev. 2 motherboard as a host for a low cost capture card

Yosemite (Blue and White) G3
Many a change from the Beige models: the Yosemite machines have a 100MHz system bus and built-in Firewire, but no serial ports and no on-board SCSI. Again there were two revisions of the motherboard — differing primarily in the IDE capability. For the first Yosemites, Apple went back to a single drive configuration on the primary IDE port. The three hard drive positions were also contained on a single, large removable metal plate in the bottom of the case.
The revision 2 Yosemites support both master and slave on the primary IDE bus. The computer has smaller individual mounting plates for each hard drive position. The system drive comes with a bracket attached, allowing a second drive to be installed immediately above it. There is also a second IDE connector in the middle of the short flat cable leading

to the system drive, for connecting a drive placed in the bracket.
All Yosemite Macs have ADB ports for older Mac keyboards and mice, as well as built in USB and Firewire ports. They do not have built-in SCSI or serial ports. Serial ports can be added by replacing the modem, with adapters from GeeThree (Stealth Port) or Griffin Technology (g port). All Yosemites have a special 66MHz PCI video card slot (not compatible with standard PCI cards), and three additional PCI slots.
Look for a rev. 2 motherboard for an offline online system — you'll probably want to use the slave position above the main drive to mount a 'conforming' drive for assembling re-digitized projects.

Yikes G4 (PCI Graphics)
The first generation of G4 Macs were built on slightly modified versions of the Yosemite motherboard. They do not have ADB ports, but otherwise share the characteristics of the rev. 2 Yosemite, including PCI graphics, which is often used to distinguish them from the next generation of G4s.

Sawtooth G4 (AGP graphics)
These machine have a new motherboard, designed for the G4 processor. The most important distinction between the original Sawtooth and the Yikes is the move to AGP graphics cards, (which require OS 9 to function).
Without (as far as I know) generating a new nickname, Apple made several significant changes to its second generation of AGP G4 computers (the third series to share the same silver and gray case and outward nomenclature). First, Apple abandoned the compact keyboard and awkward round mouse they had been using since the Yosemite debuted, replacing them with a full-size keyboard and a conventionally oblong optical mouse. Under the hood, Apple dropped the 'microphone' analog audio input, raised the bus speed to 133MHz, and offered a built-in CD-RW drive for the first time.

Quicksilver G4
Introduced in July 20001, a somewhat modified all silver case design, and a big jump in processor speed appear to be the primary the changes. The new machines are shipping with, and apparently require, an incremental update to the OS, version 9.2. Whether any compatibility problems arise from this is as yet unknown.

A2: Using removable hard drives with a Mac.

The Mac is my platform of choice for computer-based video editing. I also believe removable IDE hard drives are the answer to the storage problems posed by a multi-user, multi-station editing facility, like a video lab serving a college production class (removable for functionality, IDE for cost effectiveness). There's a problem though: the case design of recent Macs makes it very difficult to add a removable hard drive 'mobile rack.' There are no open-accessible drive bays in the case, and IDE drives depend on internal mounting and relatively short cables. How do you get the mobile rack to work with the Mac? I have tried quite a number of schemes to do this, all of which seemed like they *should* work, but many of which failed for one reason or another (maybe being OK for general purpose computing, but unable to meet the sustained throughput demands of video editing.) I have discovered two methods that do work (though only one has been torture tested over time), but both of those involve a certain amount of DIY labor in collecting and assembling a variety of component parts. Fortunately, in 2001 a reputable computer hardware vendor released a product that answers this need exactly, offering a third option, and the first one we'll consider here.

1). Buy the Granite Digital FireVue Firewire RAID
(http://www.scsipro.com/catalog/pg22_firewireidehotswapraid.htm).

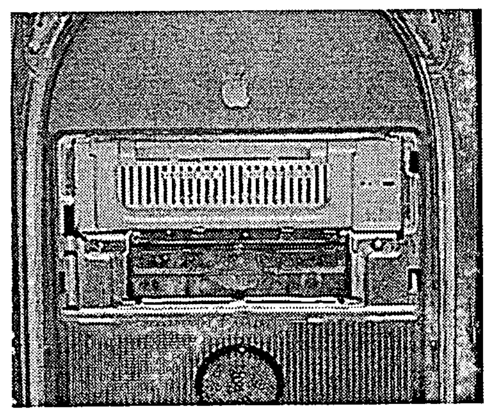
Like all current Firewire drives, the Granite uses IDE drive mechanisms connected to a Firewire interface via a small IDE-to-Firewire bridge board housed in the case. The Granite differs from other Firewire housings in that it employs a mobile rack system — the bridge board is built-in to the rack receiver frame. Although Granite has designed this product primarily as pre-packaged RAID, they also sell a two-bay case without drive mechanisms (model 0200). The Granite has the advantage of using Oxford chips in the IDE-Firewire bridge boards — a newer technology that yields much higher performance than earlier bridge designs (close to the speed of connecting the drive directly to an UltraATA port on the computer's motherboard).

So, why would anyone want to forsake this neatly packaged solution for one of my DIY schemes? The Granite is a bit more expensive. The major difference lies in the cost of the additional trays. A Granite tray is $30, while standard mobile rack trays can be found for $8. Depending on how many students you need to equip with drives, this may eat up enough budget to make some extra effort worthwhile, or not.

The only way to get the Granite as a complete package (minus drives) is to order the 2 bay model ($527). Although it's nice to have two removable bays per computer — just in case you need to copy media from one drive or another, or to accommodate that special student with a huge project who has legitimately outgrown the capacity of her first drive — you only *need* one. You can buy a single Granite mobile rack, a separate case to put it in, and the necessary case connectors, in order to produce a more economical package, but then you have to put it together. BTW, even if you go with only a single external removable bay per computer, you might want to put it in a two-bay case just in case you want to add-on in the future.

(I should also note here that I have tried to install standard IDE mobile racks in first-generation (non-Oxford-chip) external Firewire cases — such as the ADS Pyro Drive Kit. The mobile rack didn't fit in the case without modification, and the combined device was subject to constant phantom TC break errors and regular dropped frames with Final Cut and system 9.04. An upgrade to OS 9.1 cured the major problems, and allowed me to complete a substantial project using the system, but my overall impression was that the data throughput of the combination was just above what DV demanded with little room for error. Given the serious problem of fitting the mobile rack into the case in the first place, I don't recommend this method. I must also note that an attempt to put standard mobile racks into a case equipped with Granite Digital Oxford-chip bridge boards was a particularly spectacular failure. There may have been a defective component responsible for the problems here, but I cannot verify this, and thus cannot recommend this combination. The Granite hot-swap mobile racks are the only Firewire-based removable I can endorse on operational grounds, and they have the advantage of being a much easier package to get assembled and working since the bridgeboard is built in to the mobile rack frame itself.

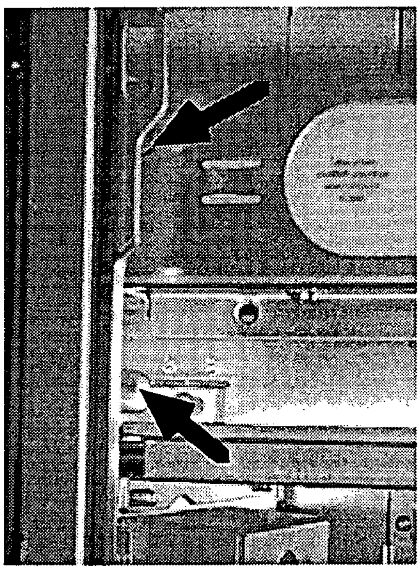
2. Remove the CD-ROM drive from the Mac, and put a mobile rack in it's place.



This is the tried and true method. The secondary IDE port in the Mac, where the CD-ROM and optional Zip are normally connected, is not UltraATA, so a hard drive in a mobile rack installed there will not be quite as fast as one connected to the primary IDE port, nor as fast or via an Oxford chip Firewire connection. However, there is plenty of headroom for a DV stream — this is definitely faster and more reliable than placing an identical hard drive in a first-generation external Firewire case.

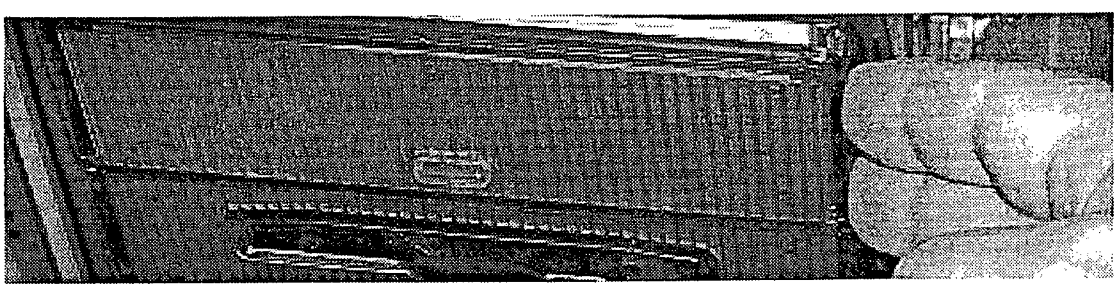
Using this method has two minor disadvantages: first, there's only room for one drive; second, adding back CD-ROM capability in external form is an additional small expense and likely tinkering project. Ability to mount two removables at once is a convenience, not a necessity. CD-ROM capability, though, is more crucial, if only to be able to restart the computer from the system CD in case of trouble.

So, you'll need to add an external CD-ROM drive. This means, first of all, equipping the Mac with a bootable SCSI card. The UltraSCSI card that has come as a $47 option on Firewire Macs works fine. This is basically an Adaptec 2930, which is also available separately. You don't have to get a deluxe card — it's not going to be shuttling high-speed data. But whatever card you get, make sure a drive connected to it can boot the system. (You need to boot from the system or repair CD to troubleshoot the system.) You also need a CD-ROM drive that will boot the Mac, which means it needs to have been made with certain Apple-specific instructions in its ROM. Most standard CD-ROM drives don't have this, and though they will work fine to run CD-ROMs once you start up from the hard drive, they won't be able to startup the system in all cases themselves. Unfortunately, you can't just go out and order a nice new external SCSI CD-ROM drive that meets these specifications. You'll probably need to go to eBay and look for an old Apple-branded bare CD-ROM drive. These will be slow (2X-8X) but cheap (under $25). Next, you'll need a standard external 5 1/4" SCSI case to put it in. These are available from many sources. Look in the back of *Computer Shopper* for dealers that specialize in cases, or do searches for "scsi case" and "scsi enclosure" in Yahoo shopping. Halted Electronics (www.halted.com) often has low priced surplus cases.

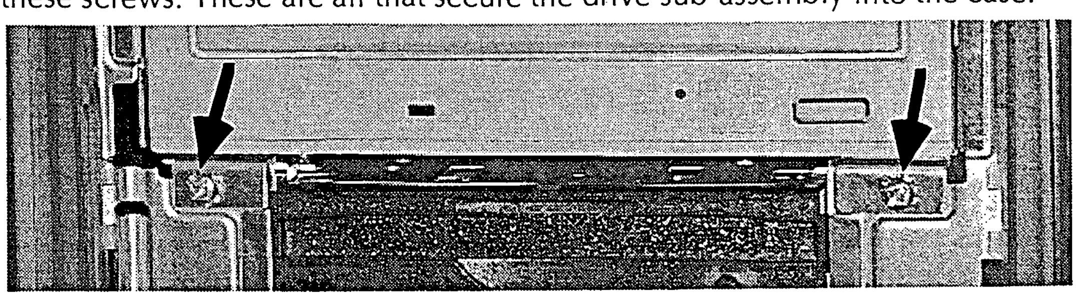


To remove the CD-ROM from the Mac. Open the side, then look for the two plastic tabs that hold in the drive trim panel. Push them in gently — toward the drive — and then forward. The trim plate should pop out just a bit.

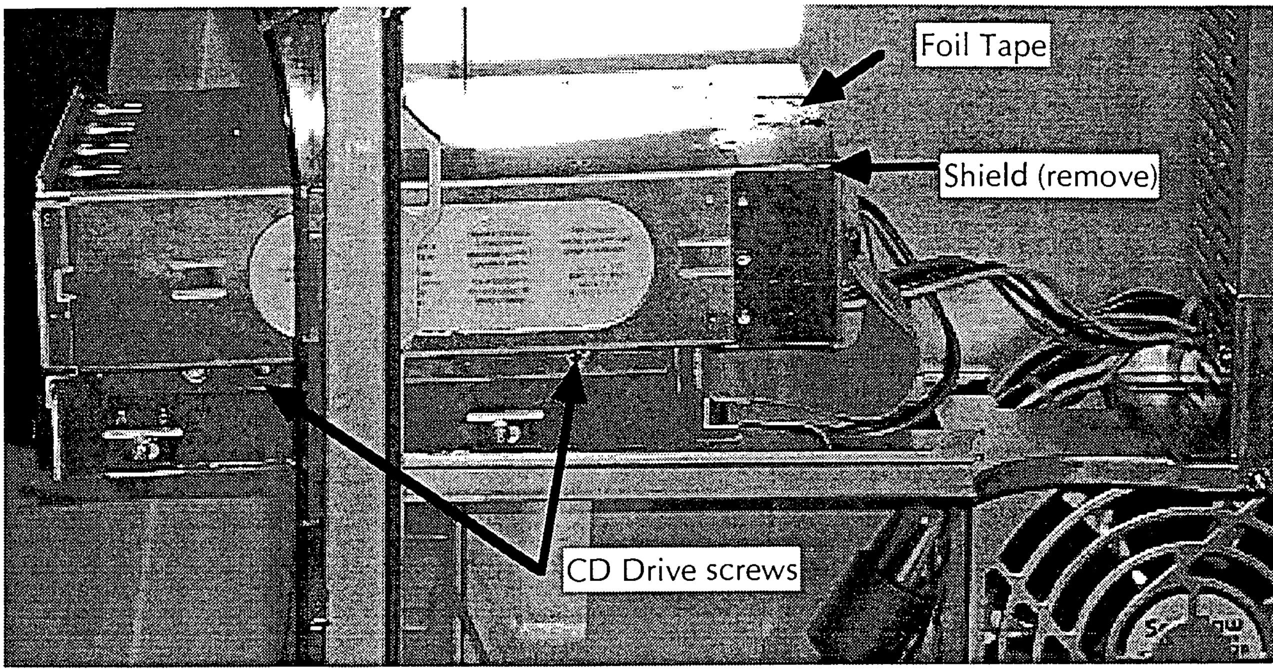
The trim plate is hinged on the left side (looking at the front). Swing it out gently until you feel resistance, then wiggle it up and down gently until it comes out.



This reveals two screws, one on each side of the Zip drive (or empty Zip bay). Remove these screws. These are all that secure the drive sub-assembly into the case.



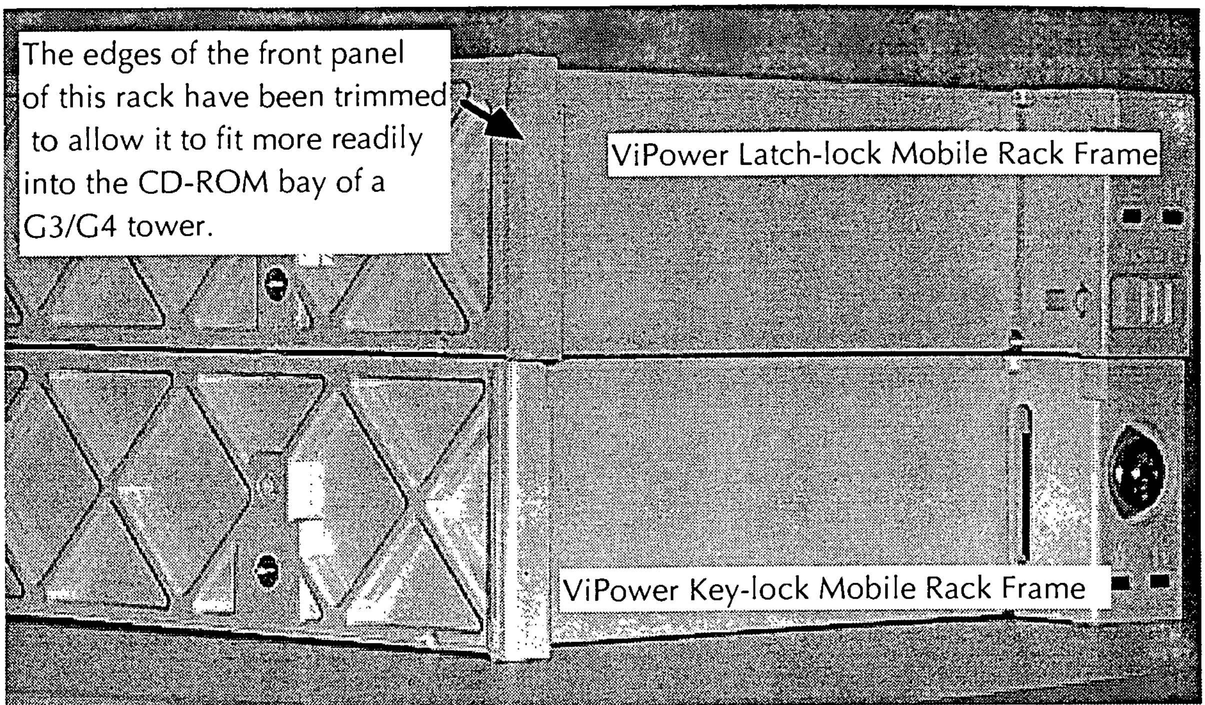
Reach behind the assembly, between the back of the CD drive and the power supply, and push the sub-assembly forward. (Don't try to unhook the data and power cables with the assembly in place. It's too cramped.) You may need to give it a good shove to get it to move. After it has slid forward about halfway, you can remove the data and power cable(s) from the CD drive (and Zip, if so equipped).



The data cable may be secured to the top of the sub assembly with foil tape. You should peel this back carefully from the top, leaving it attached to the cable, so you can stick it back on when you put everything back. Take the assembly all the way out. The CD drive is secured by 4 screws on the bottom. Remove the screws, and slide the drive out of the assembly. Save the screws by taping them to the removed CD drive, so you'll know where to find them if you want to put the CD drive back in someday. The back of the sub-assembly will have a metal shield clipped over the back of the CD bay, possible secured with more foil tape. Remove this.

Plastic mobile racks generally come from one of two Taiwanese manufacturers — ViPower and Lian-Li — although they may be repackaged and sold under a variety of names. ViPower-made racks generally retain model numbers beginning with 'VP' and Lian-Li-made racks generally have model numbers beginning with RH. I've only used the ViPower racks personally, and I've been quite content with them. Racks from both companies come in a number of variations. Most racks use a circular key to lock the tray into the frame. This strikes me as a bit awkward for a lab, as keys could get lost or someone could walk off with one. (The frames don't seem to be keyed individually, though. Any key seems to work on any lock.) ViPower also makes a model with a sliding latch in place of the key lock. This seems more practical for a student lab, and is what we've used at Connecticut College. These are harder to find at dealers, but can be purchased directly from ViPower's US sales representative (510-623-0958) with a school

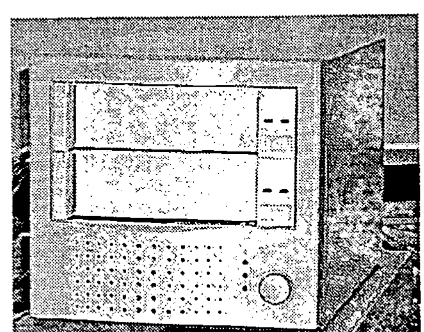
PO. (www.vipower.com)



The Mac CD bay is a very tight fit, and in fact the ViPower mobile rack frame is just a little too wide to fit comfortably. You can get it all the way in, but the front pinches just enough to cause the dust door to bind. To create a better fit, you need to shave or sand just a bit of the sides of the front panel on each side. In the photos here, the keyed frame is unmodified, and the slide-latch frame has had its sides trimmed to make a better fit in the CD bay.

Installation is the reverse of removal. Use the self-tapping screws that come with the mobile rack (not the machine screws) to attach the frame to the sub-assembly. Slide the assembly half-way in and reattach the cables. The hardest part of the job is getting the connector back onto the Zip drive correctly (if you have a Zip drive). It helps to have a flashlight. Angle the sub-assembly up to give your fingers more room.

Once the mobile rack is in the Mac, you can't put the front trim panel back on. This disturbs that Mac's space age cosmetics, giving your lab a more touch of tech-grunge aesthetics.



3. Attach IDE-WideSCSI bridgeboards to the back of the mobile rack frames, mount the frames in external SCSI enclosures, and connect the enclosures to the Mac via a wide SCSI card and 68 pin cable.

For a bit less money and a bit more effort than the FireVue requires, you can assemble a similar device using a SCSI rather than Firewire interface. The heart of this method is the IDE-UltraWideSCSI bridge board made by ACARD. This provides fast performance similar to an Oxford bridge board. ACARD is a Taiwanese company that specializes in interfaces and interface adapters for the Mac. They were the OEM for the ProMax TurboMax PCI ATA card, and are OEM for the current Sonnett Tempo ATA card. There is only one good source for ACARD branded products in the US: Microland USA (www.microlandusa.com) with offices in texas and California.

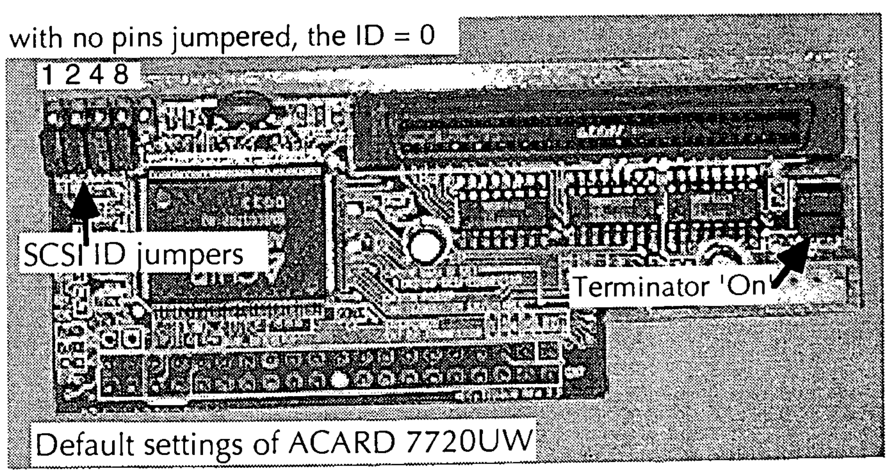
You will also need a wide-SCSI adapter in the Mac to connect the external enclosure. ACARD offers an economical model, which certainly ought to interface properly with the ACARD bridge board.

Though designed primarily to be attached to a bare drive, the ACARD 7720UW bridge board fits neatly on the back of a mobile rack, leaving the fan unobstructed.

In order to house the combination of mobile-rack and bridgeboard, which is deeper than a standard CD-ROM drive, we need a fairly deep case. Most of the single-bay cases for 5 1/4" devices aren't deep enough. The case should be at least 11" deep. A typical two-bay SCSI case offers enough room. You should order a case with 68 pin SCSI-III (wide SCSI) connectors and cables already fitted. Dirt Cheap Drives (www.dirtcheapdrives.com) offers an appropriate model, FHCASEW, for $95.

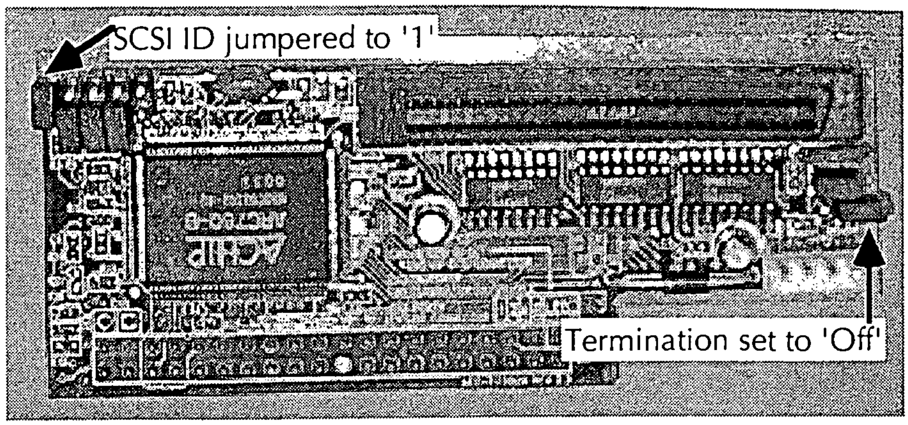
We have just switched our lab at Connecticut College to this system, after using the mobile-rack-in-the-CD-bay method described above for a year-and-a-half. We decided to build two-drive cases, for the convenience reasons described earlier. You don't *need* a dual bay setup, and you may go with a single mobile rack per station, but you should still mount it in a two-bay case, for the depth and in case you want to add a second mobile rack later.

If you do build a dual-drive case, you need to set the bridge boards for different SCSI IDs, and set termination properly. The adapter at the end of the cable should have termination on, the one in the middle of the cable should have termination off. ID number is set by the 4 leftmost jumpers at the left of the bridge board. The default is all jumper off for an ID of '0'. Adding the first jumper adds '1' to the ID, the second jumper adds '2', the third '4' and the fourth '8'. Since there are going to be no other devices on our SCSI chain, we left one adapter at ID '0" and set the other to ID '1'. Termination is controlled by the jumper nearest the power connector. The default is for the jumper to be on, meaning termination is on. Only the last device on the cable should be terminated, so we remove the jumper from the adapter that will connect in the

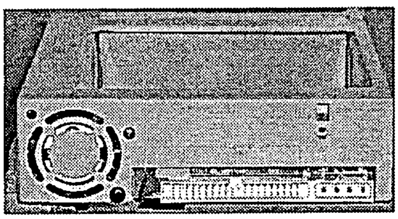


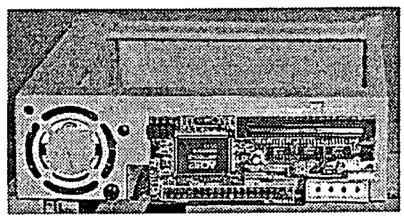
Default settings of ACARD 7720UW

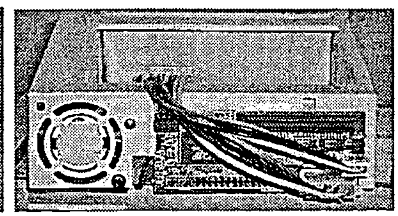
middle position on the cable. (This is the top position in our case, since our cable runs over the side and goes from the top down. Your cable may go the other way.)



Once the jumpers on the boards have been set, attach them to the mobile racks.

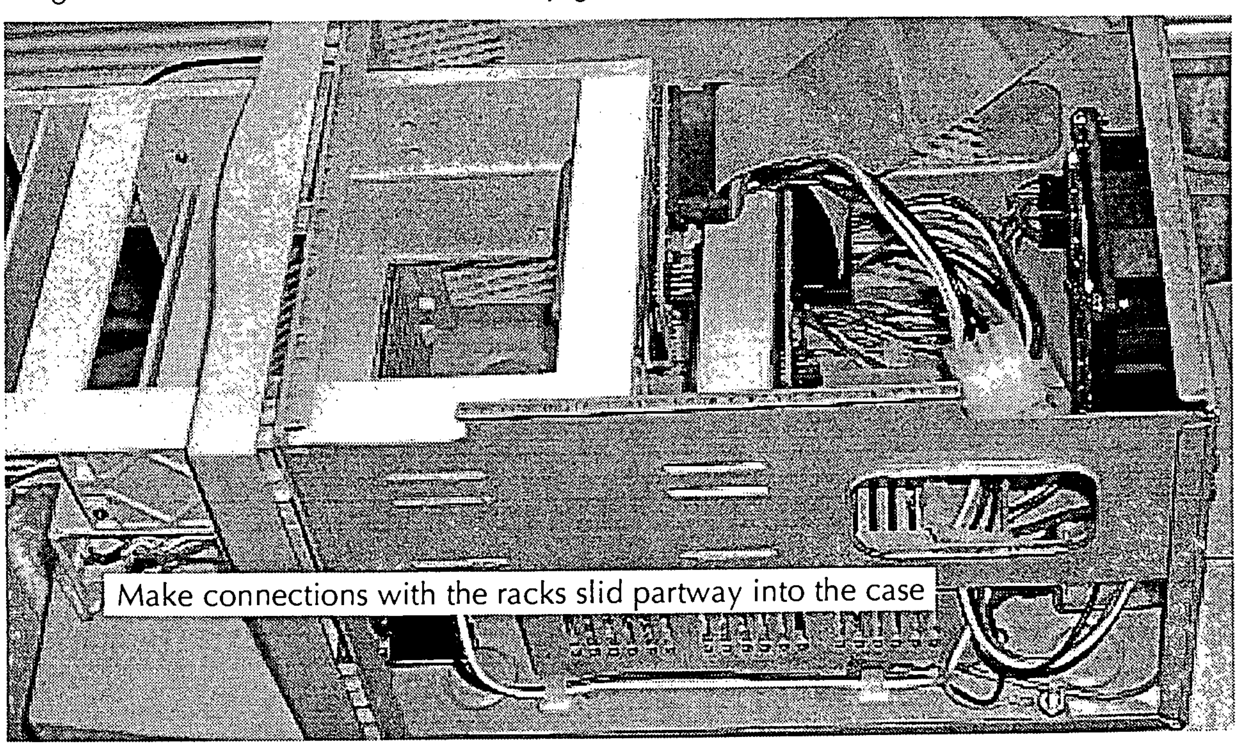




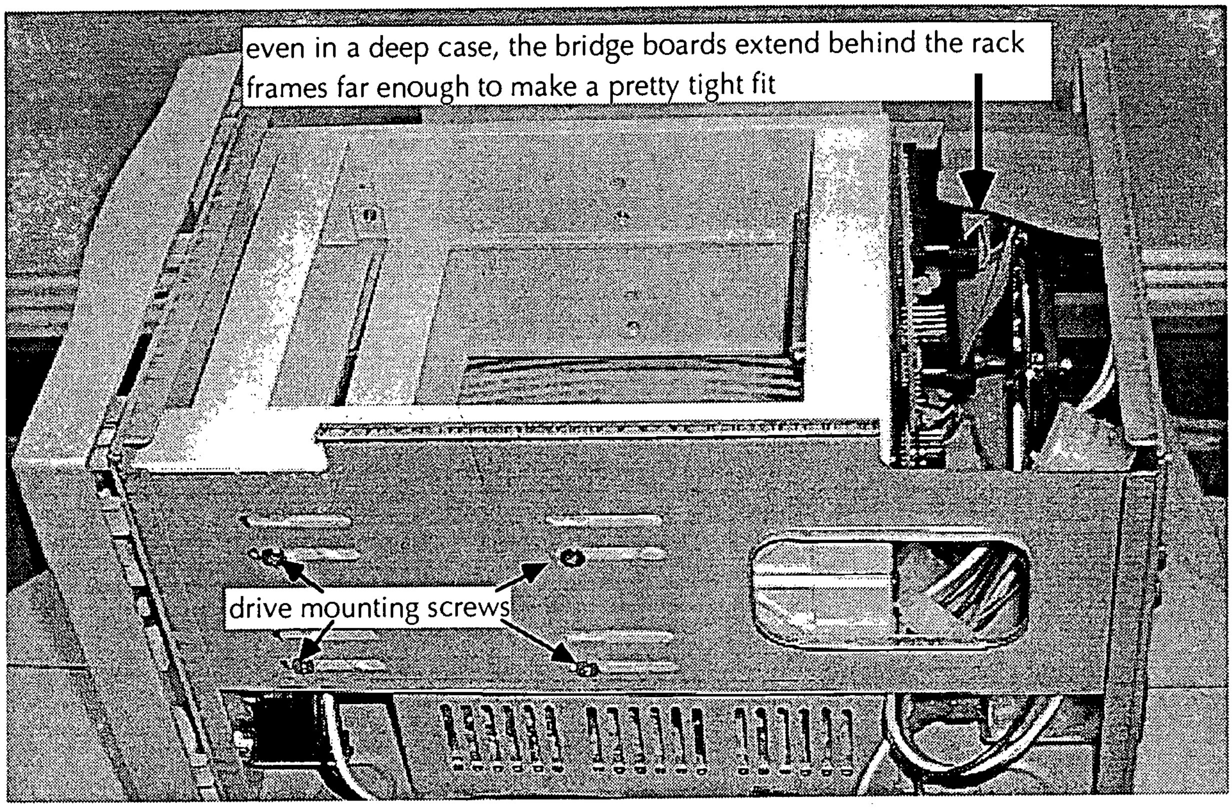


Next attach the pigtail power lead, plugging the respective ends into the bridgeboard and the hard drive. The case has several wires we're not going to use, for SCSI ID switches, for audio out jacks, for drive LEDs. Just tuck these out of the way.

Slide the racks half-way into the case. Attach the data cable connectors to the bridgeboards and attach the ends of the pigtail cables to the power leads of the case.


Make connections with the racks slid partway into the case

Carefully push the mobile racks the rest of the way in, carefully routing your now-folding cables so they don't get pinched or stick into the exhaust fan. Secure the mobile racks in position with drive mounting screws and replace the case cover.



Parts List

ACARD AEC-6712UW Wide SCSI PCI card / www.microlandusa.com/ $85
(2) ACARD AEC-7720UW IDE/SCSI bridge board @ $74 / $148
(2) ViPower VP10LSF mobile racks @ $17 ea. / $34
FHCASEW 2 bay wide SCSI case / www.dirtcheapdrives.com / $95
SC3TOSC3-3 3' wide SCSI cable www.dirtcheapdrives.com / $20
Total: $382

A3: Reviewof the Matrox RT2000

(In my initial review of the RT2000, I noted several shortcomings — the system was difficult to install and configure properly; the real-time effects were not keyframable, limiting their usefulness; DV device control was useless, and compression levels for MPEG-2 captures could not be set accurately. My impression was that Matrox had rushed the software release. In fact, the company issued two bug-fix 'service-pack' software updates in the brief time I had the board. Since, then Matrox has continued to diligently work on improving the software, addressing most of issues I just noted. While I have not have the chance to examine the updates myself, reports on the Web are encouraging.)

This system features real-time editing of three layers in Premiere, using either native DV or MPEG-2 I-frame capture. The RT2000 is very modestly priced — under $1000 (2/2001) for the board and software. It is unique is its ability to create 3-D effects without high priced hardware. However, most of the 3D effects Matrox supplies with the system are not likely to be too useful within a film-style production curriculum. They are more likely to appeal to event videographers and production houses creating local advertising.

To be fair to even the earlier software release, I must qualify my initial difficulties with the RT2000 with 2 caveats. First, I'm not particularly Windows savvy, and second I did get it to work — after three tries and two calls to tech support. This fumbling around probably would have been eliminated if I could have acquired a turnkey system — which again I strongly recommend for anyone acquiring ANY Windows based NLE.

Once I did get the system running I was very impressed with its real-time editing capabilities. The RT2000 is designed to edit native DV video, captured via Firewire, but it does not display video playback back out the Firewire port during editing. Rather, it has a more conventional breakout box with video and audio outputs for direct connection to a monitor. This means that, unlike a pure DV system such as the FinalCut package discussed below, a DV device does not have to be present at all times during editing — only for capture and final layback to tape. This may allow some schools to save money by forgoing dedicated feeder decks at the edit stations, instead just bringing in camcorders when necessary. In displaying real-time transitions and filters, the RT2000 does not actually create a new DV stream, and thus, when the edit is finished, the effects must be rendered before the completed program can be sent out Firewire, but the RT2000 does this *very* quickly, and since it only has to be done once, this is basically a non-issue.

The Matrox hardware shines in bringing real-time transitions to DV video editing at a very affordable price. It also has very strong real-time title generation and title animation features, which I find more useful than the 3-D effects. The RT2000 is a strong contender for a DV based, online editing lab.

The RT2000 also includes a scalable codec — MPEG-2 I-Frame — giving it potential as an offline/online system. Unfortunately, this potential is limited by other aspects of the RT2000's design. The most significant of these, is that the RT2000 can not be set to capture MPEG-2 video at a fixed frame rate — e.g. using the "limit each frame to

___KB" setting in Premiere is not available. You can only adjust the compression rate using a sliding "quality" bar. This is "variable bit rate" compression (VBR) where the system attempts to maintain a consistent visual quality. Thus, more complex images will compress less, and simpler images will compress more. The problem is that with only VBR available, it is impractical to capture low-res files to a removable cartridge drive, such as an Orb. When a complex image comes along, the data rate will surge beyond the Orb's speed, and the capture will fail. You can always reset the slider to a lower setting and try again, but this begins to add way to much fiddling around to a process (offline/online) that is already more complicated than we'd like for students. In addition, the quality slider in the RT2000 is poorly implemented. The scale is divided into several segments — Very Low, Low, Medium, High and Very High. Not that there are only five settings on the scale — it's continuously variable. However, there is no numerical readout of your position on the scale (a percentage number, say), just a readout of the general quality categories that changes as you cross one of the borders. This makes it difficult to duplicate a particular setting with any consistency. The scale is also poorly defined — the Medium range settings, and even the lower end of the High range yielded highly compressed images that displayed noticeable blocky digitization artifacts. This is not to say that the RT2000 doesn't digitize well, only that the actual range of useful settings are confined to the upper end of the quality slider. Unfortunately, at the upper end a small movement of the slider can result in large changes in the data rate. At the very top of the scale, the system digitizes at the maximum 25Mbits/sec (about 3.25MB/sec). Move the slider just slightly off the top end and the rate may drop to 10Mbits/sec.

If you use small removable hard drives for offline work, as I have recommended, these problems are lessened somewhat, since you can easily find a setting that is within the drives data range and avoid a failed capture. However, setting the compression rate is still awkward, and the results will not yield the most efficient use of your storage space.

What all this adds up to is that the RT2000 is probably not a good fit for a lab with a flexible number of users, where you are expecting a lot of offline work. With the RT2000, every user should have her own hard drive, whether working online or offline. While the basic mode of operation of an RT2000 lab would be online editing, it would still offer the alternative of MPEG-2 offline work for students working on longer, more complicated projects. In any case, the native DV codec will provide excellent image quality, and the real-time functions will improve student workflow.

While this system might appeal to someone working in a Windows-only environment, I prefer the Matrox RTMac. While the RT2000 offers 3D effects and a limited scalable compression offline mode, and the RTMac does not, the RTMac has the advantage of running on a more stable and user friendly platform, and supporting a more sophisticated editing program in the form of Final Cut Pro.

U.S. Department of Education
Office of Educational Research and Improvement (OERI)
National Library of Education (NLE)
Educational Resources Information Center (ERIC)

ERIC

CS 510 596

Reproduction Release
(Specific Document)

I. DOCUMENT IDENTIFICATION:

Title: Non-Linear Editing for the Smaller College-level Production Program, rev. 2.0
Author(s): DAVID J. TETZLAFF
Corporate Source:
Publication Date: 2001

II. REPRODUCTION RELEASE:

In order to disseminate as widely as possible timely and significant materials of interest to the educational community, documents announced in the monthly abstract journal of the ERIC system, Resources in Education (RIE), are usually made available to users in microfiche, reproduced paper copy, and electronic media, and sold through the ERIC Document Reproduction Service (EDRS). Credit is given to the source of each document, and, if reproduction release is granted, one of the following notices is affixed to the document.

If permission is granted to reproduce and disseminate the identified document, please CHECK ONE of the following three options and sign in the indicated space following.

The sample sticker shown below will be affixed to all Level 1 documents	The sample sticker shown below will be affixed to all Level 2A documents	The sample sticker shown below will be affixed to all Level 2B documents
PERMISSION TO REPRODUCE AND DISSEMINATE THIS MATERIAL HAS BEEN GRANTED BY SAMPLE TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)	PERMISSION TO REPRODUCE AND DISSEMINATE THIS MATERIAL IN MICROFICHE, AND IN ELECTRONIC MEDIA FOR ERIC COLLECTION SUBSCRIBERS ONLY, HAS BEEN GRANTED BY SAMPLE TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)	PERMISSION TO REPRODUCE AND DISSEMINATE THIS MATERIAL IN MICROFICHE ONLY HAS BEEN GRANTED BY SAMPLE TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
Level 1	Level 2A	Level 2B
X		
Check here for Level 1 release, permitting reproduction and dissemination in microfiche or other ERIC archival media (e.g. electronic) and paper copy.	Check here for Level 2A release, permitting reproduction and dissemination in microfiche and in electronic media for ERIC archival collection subscribers only	Check here for Level 2B release, permitting reproduction and dissemination in microfiche only

Documents will be processed as indicated provided reproduction quality permits.
If permission to reproduce is granted, but no box is checked, documents will be processed at Level 1.

I hereby grant to the Educational Resources Information Center (ERIC) nonexclusive permission to reproduce and disseminate this document as indicated above. Reproduction from the ERIC microfiche, or electronic media by persons other than ERIC employees and its system contractors requires permission from the copyright holder. Exception is made for non-profit reproduction by libraries and other service agencies to satisfy information needs of educators in response to discrete inquiries.

Signature: [signature]
Printed Name/Position/Title: DAVID J. TETZLAFF, ASST. PROF.
Organization/Address: Connecticut College New London CT 06320
Telephone: 860-439-5253
Fax:
E-mail Address: djtet@conncoll.edu
Date: 7-24-01

III. DOCUMENT AVAILABILITY INFORMATION (FROM NON-ERIC SOURCE):

If permission to reproduce is not granted to ERIC, or, if you wish ERIC to cite the availability of the document from another source, please provide the following information regarding the availability of the document. (ERIC will not announce a document unless it is publicly available, and a dependable source can be specified. Contributors should also be aware that ERIC selection criteria are significantly more stringent for documents that cannot be made available through EDRS.)

Publisher/Distributor:
Address:
Price:

IV. REFERRAL OF ERIC TO COPYRIGHT/REPRODUCTION RIGHTS HOLDER:

If the right to grant this reproduction release is held by someone other than the addressee, please provide the appropriate name and address:

Name:
Address:

V. WHERE TO SEND THIS FORM:

Send this form to: ERIC Clearinghouse on Reading, English, and Communication (ERIC/REC).

ERIC/REC Clearinghouse
2805 E 10th St Suite 140
Bloomington, IN 47408-2698
Telephone: 812-855-5847
Toll Free: 800-759-4723
FAX: 812-856-5512
e-mail: ericcs@indiana.edu
WWW: http://eric.indiana.edu