# Purchasing A Workstation Installation

**By Peter Galvin, Systems Manager**

**and Jeff Coady, Director of Facilities**

**Brown University Computer Science Department**

*Paper available from http://www.cs.brown.edu/people/pbg*

**pbg@cs.brown.edu**  **jwc@cs.brown.edu**



## *Overview*

**In 1991, we undertook to upgrade our computing facility. Funding mostly came from an NSF grant. We wrote an RFI and circulated it to some chosen vendors.**

**After one year of work, a vendor was chosen and the order placed. By the end of two years the equipment was tested and installed: more than 160 workstations, 4 servers, and an ethernet-switched network.**

**This talk is about the process, the methods, and the lessons learned to successfully replace the facility.**

**If there is time left at the end...**

## *Overview*

I. What Were We Trying To Do?

II. Requesting Information from Vendors

III. Evaluating Data and Proposals

IV. Determining the Players

V. Picking the Winner

VI. Testing the Installation

VII. Insights





## I. *What Were We Trying to Do?*

- **Increase performance of the current installation, consisting of:**
  - 160 Sparcstation 1 systems (16MB, 100MB disk, 16" color monitor)
  - 2 Sun 4/490 NFS Servers (32MB, 10GB disk)
  - 1 Sun 4/260 Mail/News Server (16MB, 500MB disk)
- **Keep the conversion effort and time to a minimum**
- **Retain a high facility reliability**

## *Approach*

- **Consider this entire process a learning opportunity. Get to:**
  – Learn the future plans of each of the vendors (1 to 2 years out)
  – Determine their strategic directions (ONC, DCE, OS revision plans)
  – Have every vendor help make sure your final installation is the best it can be
  – Learn about previously ignored hardware aspects (3D graphics performance)





## *Who is going to do it?*

- **Technical decisions within the Department are made by the *FACIL committee:***
  – The Chairman and Director of Facilities took the lead on administration and financial issues
  – The Systems Manager and two faculty members were the technical leads
  – The Manager of Research and Educational Software, as well as several other faculty members, had the undergraduates and teaching in mind
  – A faculty member with many industry contacts took the negotiating lead
- **Although much work was done by subcommittees, the committee had the assets to deal with all aspects of the upgrade.**

## *How are we going to do it?*

- **Certainly not by leaving it to the vendors:**
  - SPEC ratings don't tell the whole story
  - Impossible to convey all parameters, design goals, "feelings" to vendors
  - Getting everything we could think of in writing, with penalties
- **Nor your company**
  - Purchasing departments are good at purchasing pencils, but not computer installations
  - Different departments have different needs: what's good for the goose isn't always what the gander wants.





## II. *Requesting Information From Vendors*

- **Select the vendors to solicit**
  - Too few, and you may miss an opportunity.
  - Too many, and the evaluation effort could be overwhelming

## *Write an RFI or an RFP*

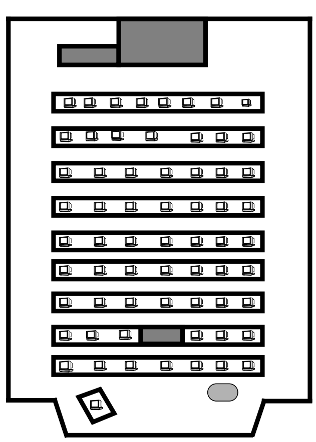
- **Purpose is to ensure the successful vendor can and will provide a platform which meets your specs and expenses.**
- **A Request For Information is less formal and legally binding than a Request For Proposals**
- **Provide background**
  - Why should they be interested in you?
  - What can you do for them?



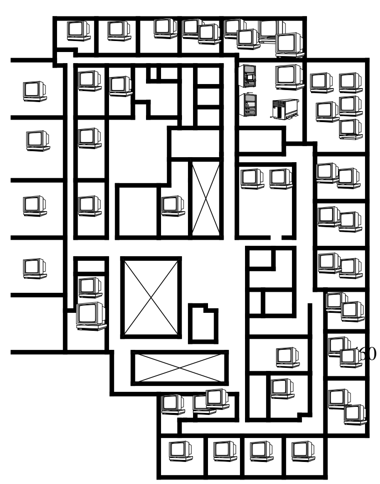


## *Writing an RFI: Describe current facilities - 1*

- **Include fancy diagrams showing how impressive your current installation is...**



**1st Floor Instructional Lab**



**5th Floor**

## *Writing an RFI: Describe current facilities - 2*

- **And pictures showing your facilities in use**

**"Picture your company's workstations here!"**





## *Writing an RFI: Describe current facilities - 3*

- **Describe the current software installation**
  - What commercial software is being used?
  - What pubic domain/shareware/copyleft software is in use?
- **Include network details if servers are included in the proposal**
  - Vendors will need to know interface issues, at the least
  - May also need to know server requirements (NFSops/second, etc)
  - Wiring plan needs if vendors are bidding a full solution

# *Writing an RFI: Specify range of equipment of interest*

**Table 1: Hardware specifications**

| Requirements | Preferences |
|---|---|
| **Workstations**<br>• Workstations must be very quiet, require no special air-conditioning equipment, and be of robust construction<br>• Desktop models must have small footprints to fit on desks and tables<br>• Disks on workstations should be well protected from tampering<br>• Inexpensive, rugged and secure network connections<br>• Network throughput at Ethernet or better speeds<br>• Workstation failure should have a minimal impact on network integrity<br>• CD-quality sound input and output<br>• 8 planes of color with options for 24 or 32 planes<br>• Accelerated 2D color display hardware<br>• Available 3D accelerated color display hardware with Z buffer<br>• High density removable media (such as a floppy disk)<br>• Support for live video<br>• Since a large number of these machines will be in our public auditorium, security for the machine and its peripherals is essential<br>**Servers**<br>• Sustain equivalent of 2400 NFS I/O operations per second; half that if local, persistent caching is provided on the workstation side (as in AFS)<br>• 25GB of disk storage, expandable to over 50GB<br>• Support for routing and transparent file access to well over 10 ethernet subnets, with future expansion desirable<br>• 8mm tape drives | • Binary compatibility across all models is strongly preferred<br>• Several levels of workstation performance<br>• Multiprocessor options<br>• Ease of upgradeability<br>• Wide variety of I/O devices including tablets, pens and 9-dial dial boxes<br>• Twisted-pair ethernet is preferred to thin or thick-net connections<br>• FDDI enhancement possibilities across all platforms<br>• Compatible laptop systems<br>• Interoperable X Window System Terminals<br>**Low-range systems**<br>• 40 to 80 MIPS, >10 Linpack MFLOPS<br>• 32-64MB main memory, >200MB local disk<br>• 16" 1K x 1K pixel displays , >60K Gouraud-shaded triangles/sec<br>**Mid-range systems**<br>• 80-120 MIPS, >20 Linpack MFLOPS<br>• 32-64MB main memory, >400MB local disk<br>• 16" 1K x 1K pixel (or better) displays, >200K Gouraud-shaded triangles/sec, Hardware support for pattern mapping, NURBS,and radiosity<br>**High-end systems**<br>• >100 MIPS , >40 Linpack MFLOPS<br>• 64-128MB main memory, 400MB-1GB local disk<br>• 19"or 21" 1K x 1K pixel (or better) displays, >350K-1M Gouraud-shaded triangles/sec, Hardware support for pattern mapping, NURBS, radiosity<br>• Deskside VME expansion<br>**Servers**<br>• 100 to 200 MIPS<br>• Automatic disk crash recovery<br>• DAT tape drives |





# *Writing an RFI: State software requirements*

- **What commercial software must run?**
- **What public domain/copyleft/shareway must be already ported or easily ported?**

**Table 2: Software requirements**

| Requirements | Preferences |
|---|---|
| • On-line manual pages, document sets and/or help<br>• Support for POSIX-compatible threads<br>**Programming Languages**<br>• ANSI C, $C^{++}$, Common Lisp, Pascal, Fortran, and an assembler<br>• Extensive software development tools<br>• Excellent source-level debugging tools for all languages<br>**Editors**<br>• Variety of editors, such as vi and GNUemacs provided or available<br>**Text Formatting and Printing**<br>• TeX and ditroff<br>• Postscript printing<br>**Third-Party Software**<br>• A large selection of third-party software must be available<br>• Frame's FrameMaker and Informix's WingZ must be immediately available | • Prolog and Scheme<br>• A good Lisp development environment<br>• Postmortem debugging and detailed profiling of C and Pascal programs are desirable<br>• Source, in C, for these tools for local customizing<br>• Good terminal/3270 emulation<br>• IBM PC (MSDOS) and Macintosh emulation software<br>• On-line, X based, hypertext access to all documentation<br>• Integrated office automation software including calendar/scheduling and electronic mail-handling tools<br>• A debugger for multi-threaded applications<br>• Object-oriented Pascal<br>• Support for Interleaf |

## *Writing an RFI: Other requirements to include*

- **Operating system functionality (multimedia support), APIs, performance**
- **Networking functionality (ONC, AFS, standards)**
- **Display software and associated standards (PEX, Motif)**
- **Administrative software (sysadmin tools, network information system (NIS-like))**
- **System maintenance issues (warranty period, cost, on-site spares)**
- **Hardware upgradability (MP, disk, memory)**
- **Backup interroperability (can use old backup tapes, or at least the same media)**





## *Writing an RFI: Declare timeframe and decision method*

- **Vendors want to know the steps you are taking and your timeframe. This was ours:**

**I. Distribution of the RFI to a select set of workstation vendors (Oct 91)**

**II. In-depth technical non-disclosure presentations by interested vendors addressing our requirements and preferences (Oct - Dec 91)**

**III. Invitations to selected vendors for further information, proposals, and negotiations (Jan-Apr 92)**

**IV. Final Selection (May 92)**

**V. Delivery and installation (Scheduled for Fall '92, some arrived Spring '93, and the remainder August '93) (See penalty phase)**

# III. *Evaluating proposals*

**Unfortunately, the work has just started. Now you need to:**

**I. Collect complete information about each proposed system**

- Vendor provided information (glossies, non-disclosure information)
- Benchmarking - going beyond the SPEC numbers
- Background of the company (references, what deals have they cut? any similar installations?)

**II. Produce large comparison decision aid tools (spreadsheets)**

- One for each type of system (low-end desktop, high-end desktop, NFS server, Compute server)
- Weigh each component appropriately and rank the contenders

**III. Iterate 1 & 2**





# *Evaluating Proposals: Collecting information*

- **Non-disclosure information is helpful**

  But it’s usually based on non-existing products:

  Estimated performance ⇔ little information

  SPECmarks for performance measurement

- **Benchmarking**
  - SPECmarks are interesting, but little of what we do stays in the cache

  Dataless systems ➠ NFS performance crucial

  Binaries cached on local disks ➠ Workstation disk speed crucial

  Graphics intensive ➠ Accelerator board performance crucial

  *None of which is captured by SPECmarks*

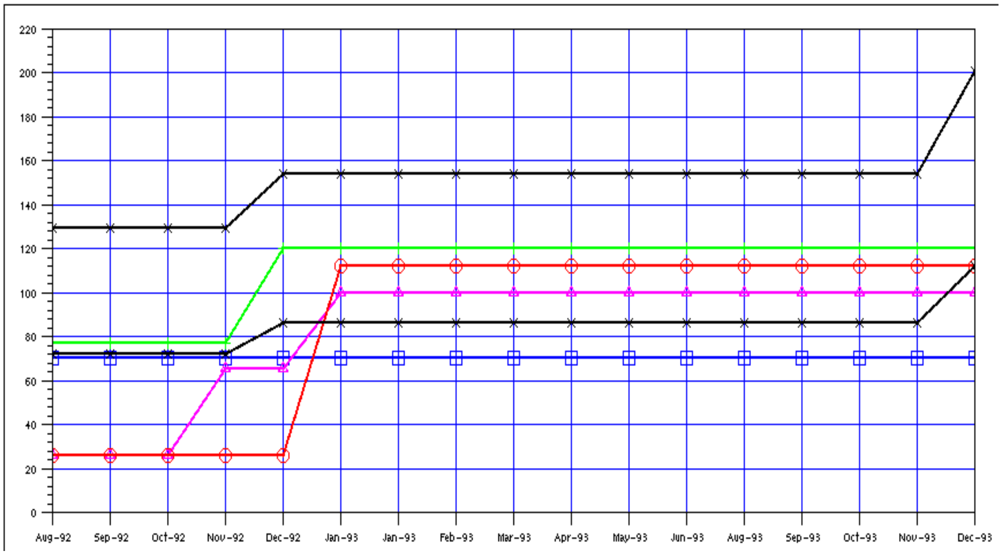
  Living on the bleeding edge ➠ Almost never testing bid systems

## Evaluating Proposals: Benchmarking

- **We did use the SPECmarks in the absence of other information:**

Date VS. SpecInt

Date vs SPECs







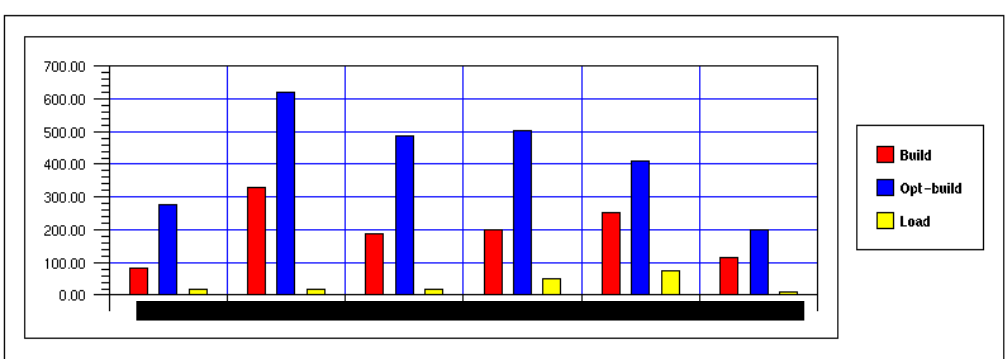

## Evaluating Proposals: Benchmarking - 2

- **For "real" data, we attempted to run our own benchmark suite.**
  - Commonly done tasks, all done over NFS, caching to local disk, where possible. For instance, build time of Emacs:

**Brown Emacs Benchmark**

15-May-92

| Emacs | [redacted] | | | | | | | Clock times in seconds |
|---|---|---|---|---|---|---|---|---|
| Build | 82.40 | 328.00 | 186.00 | 201.00 | 250.80 | 116.40 | 113.90 | Compile with debugging enabled |
| Opt-build | 274.70 | 619.00 | 487.80 | 501.80 | 408.90 | 201.20 | 443.50 | Compile with optimization |
| Load | 20.00 | 19.50 | 17.00 | 48.70 | 73.40 | 8.80 | 5.50 | Resulting emacs loading lisp librarie: |



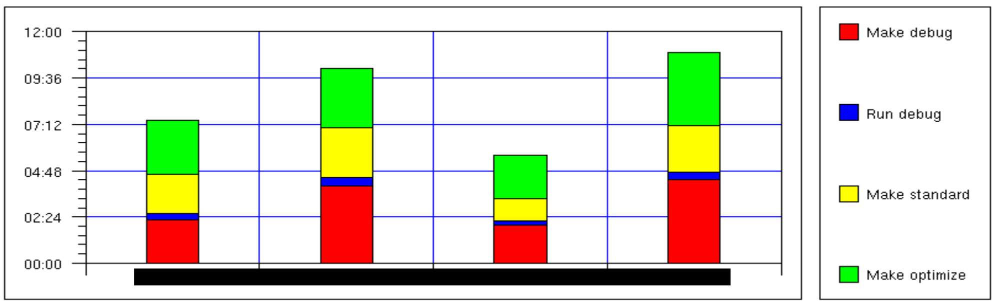
# *Evaluating Proposals: Benchmarking - 3*

- **Another good test was compiling and executing a C++ program:**

Brown C++ Benchmark | 8-May-92

| C++ | [redacted] | | | | |
|---|---|---|---|---|---|
| Make debug | 02:16 | 04:01 | 01:58 | 04:19 | |
| Run debug | 00:18 | 00:25 | 00:14 | 00:25 | |
| Make standard | 02:03 | 02:36 | 01:09 | 02:24 | |
| Make optimize | 02:47 | 03:03 | 02:15 | 03:49 | |
| Run standard | 00:54 | 00:18 | 00:11 * | | * Program failed to run |
| Run optimize | 00:13 | 00:22 | 00:20 * | | |
| Executable size | 98K | 176K | 135K | 140K | |

C++





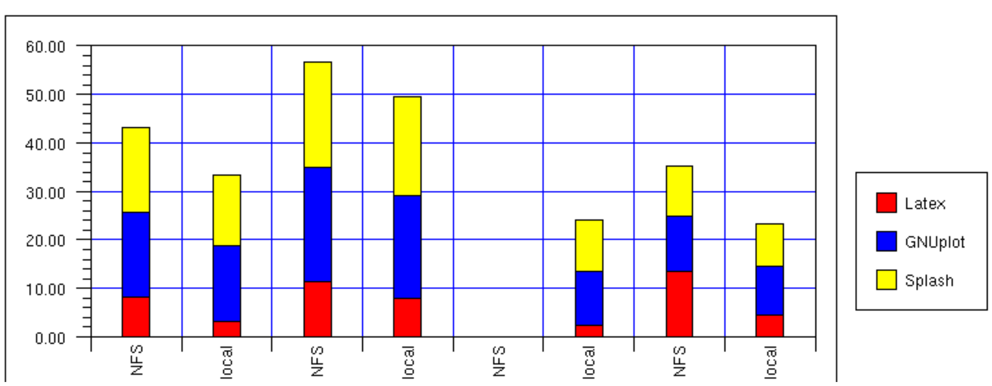



# *Evaluating Proposals: Benchmarking - 4*

- **To eliminate variables, we also built and ran Tex both on local disk and on NFS:**

| Run -g | NFS | [redacted] local | [redacted] NFS | [redacted] local | [redacted] NFS§S | [redacted] local | [redacted] NFS | [redacted] local |
|---|---|---|---|---|---|---|---|---|
| Latex | 8.30 | 3.10 | 11.30 | 7.90 | | 2.50 | 13.50 | 4.6 |
| GNUplot | 17.40 | 15.70 | 23.70 | 21.10 | | 11.10 | 11.40 | 10.0 |
| Splash | 17.50 | 14.70 | 21.60 | 20.50 | | 10.40 | 10.40 | 8.6 |
| Totals | 43.20 | 33.50 | 56.60 | 49.50 | 0.00 | 24.00 | 35.30 | 23.2 |

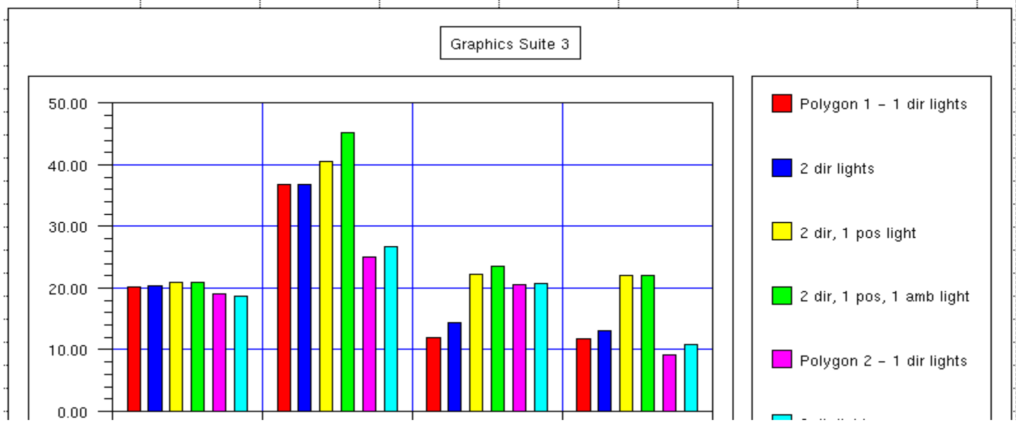
Run -g



# Evaluating Proposals: Benchmarking - 5

- **Finally, we benchmarked graphics performance using Brown's Modeler software. This was in-depth:**

| | | | | |
|---|---|---|---|---|
| Polygon 1 – 1 dir li | 20.25 | 36.80 | 12.00 | 11.70 |
| 2 dir lights | 20.38 | 36.90 | 14.40 | 13.10 |
| 2 dir, 1 pos light | 20.94 | 40.50 | 22.30 | 22.10 |
| 2 dir, 1 pos, 1 amb | 21.01 | 45.20 | 23.50 | 22.10 |
| Polygon 2 – 1 dir li | 19.03 | 25.10 | 20.60 | 9.10 |
| 3 dir lights | 18.63 | 26.80 | 20.70 | 10.90 |








# Evaluating Proposals: Tools

- **The mother of all spreadsheets was used as a first pass:**

**Table 3a. Desktop Hardware Configuration**

| Item | Low-end | Research(Graphics) | Research(Non-graphics) | Details |
|---|---|---|---|---|
| Model | | | | |
| Number Systems | 10 | 108 | 52 | |
| List | | | | |
| #CPU's/Max # CPU's | / | / | / | |
| # /Size internal disk /(Mbytes) | 1 / >400 | 1 / >400 | 1 / >400 | |
| Memory/Maximum (Mbytes) | >= 32 / | >= 32 / | >= 32 / | |
| Monitor Size | 16" | 16" | 16" | |
| Resolution/Refresh rate | | | | Sony Trinitron or equivalent |
| # color Planes | 8 | 24 | 8 | |
| Graphics Acceleration | | | | Double buffered? Other features (i.e. texture mapping, anti-aliasing,) |
| Z Buffer | - | >= 16 Bits | - | |
| Footprint | | | | |
| Audio | | | | |
| Video | | | | |
| Network | | | | |
| Free Slots | | | | |
| Removable media | >= 114 | | | >= 9 in Ed lab and 1 on each of the remaining 105 systems |

**Table 3b. Desktop Performance***

| Item | Low-end | Research(Graphics) | Research(Non-graphics) | Details |
|---|---|---|---|---|
| Specs[a] | >30 | >70 | >70 | |
| MHz | | | | |
| X11perf | | | | |
| Mflops (double precision Linpack) | | | | |
| 2D V/s[b] | | | | |
| 3D V/s[a] | - | | - | |
| 3D Z-buffered, Gouraud Shaded Poly[c] | - | | - | |
| Memory Transfer rate (Mbytes/sec) | | | | |
| Expansion Bus speed(Mbytes/sec) | | | | |
| SCSI rate/width (Mbytes/sec) | | | | |
| Disk Access/Transfer speed(ms/Mbytes/sec) | | | | |
| Audio Sample Rate | 16 Bits, 44K/sec | 16 Bits, 44K/sec | 16 Bits, 44K/sec | |
| Noise (decibels) | | | | |

a. Please attach results of the entire SPEC benchmark suite (SPEC92 results are preferred)
b. 10 pixel, 24 bit color, arbitrary orientation
c. 50 pixel triangles and 100 pixel quads (24 bit, arbitrary orientation, transformed, lit with ambient plus at least 1 directional or point light source, clipped or clip tested with at least 6 clip planes).

* Please include comprehensive and specific graphics details on separate sheet and attach. Please include items such as the





## Table 3c. Network File Server Performance

| Item | Requirements | Details |
|---|---|---|
| SPECs | | |
| Mflops | | |
| Memory Transfer rate(Mbytes/sec) | | |
| Expansion Bus speed(Mbytes/sec) | | |
| SCSI rate/width /# (Mbytes/sec) | | Maximum of 2 disks per channel |
| Disk Access/Transfer speed (ms/Mbytes/sec) | | |
| #Sustained IOPS[a] | 2400 | Total throughput |

a. Please provide description of operations mix.

**Table 3d. Special Systems Configurations**

| Host and Type | fred (spr) | bimini (AI vision) | Details |
|---|---|---|---|
| Model | deskside | deskside | |
| List | | | |
| #CPU's/Max # CPU's | / | / | |
| # /Size internal disk (Mbytes) | >1Gb | >400Mb | |
| Mb Memory/Maximum (Mbytes) | >128Mb / | >128Mb / | |
| Monitor Size | 2 @ 19" | 19" | |
| Resolution/Refresh rate | | | |
| # color Planes | 24 | | |
| Graphics Acceleration | | | |
| Options | | VME Bus | |
| Audio | | | |
| Video | | | |
| Network | CDDI | Enet | |
| Free Slots | | | |
| Removable media | | | |





**Table 3e. Special Systems Performance**

| Host and Type | fred (spr) | bimini (AI vision) | Detail |
|---|---|---|---|
| Specs | >120 | >70 | |
| Mflops (double precision Linpack) | | | |
| 2D V/S[a] | | | |
| 3D V/s[a] | | | |
| 3D Gouraud Shaded Poly[b] | | | |
| Memory Transfer rate (Mbytes/sec) | | | |
| Expansion Bus speed(Mbytes/sec) | | | |
| SCSI rate/width (Mbytes/sec) | | | |
| Disk Access/Transfer speed(ms/Mbytes/sec) | | | |

a. See footnote a on page 1, Desktop Performance
b. See footnote b on page 1, Desktop Performance

| **Product** | **Included in Proposal (check if yes)** | **Availability**[a] |
|---|---|---|
| FrameMaker 3.0 | | |
| Wingz | | |
| Artisan | | |
| Interleaf (>= 5.2) | | |
| TMC CM software | | |
| Ditroff, refer | | |
| Isofax | | |
| Maple V | | |
| Mathematica 2.0 | | |
| Renderman | | |
| Transcript | | |
| UIMX | | |
| Ansi C | | |
| C$^{++}$ | | |
| Native C$^{++}$ | | |
| Fortran | | |
| Pascal | | |
| Common Lisp | | |
| Object Pascal | | |
| LISP Programming Environment | | |
| Postscript previewer | | |
| TeX/DVI pageviewer | | |
| OA tool set | | Specify tools |
| Other | | |

Other

a. Please use the following codes:
1-Currently available
2-Soon to be available (date)
3 -Not available





| **Items** | **Details** |
|---|---|
| Total List Value | |
| Warranty | |
| Future Purchase Warranty (3 years) | |
| Educational Hardware Maintenance/year | |
| Research Hardware Maintenance/year | |
| Software Maintenance/year | |
| Guaranteed Future Discounts (3 years) | |
| Grant | |
| Collaborative Relationship | |
| Disposition of current facility | |
| Conversion support | |
| Delivery Date(s) | |
| Source code provisions | |

## Evaluating Proposals: Conclusions

- **When systems aren't released yet, it is very time consuming to gain access (lots of legal documents to sign) and test the appropriate configurations.**
- **The results of the testing are not complete, nor necessarily accurate: benchmarking is an art, not a science.**
  - Some tests needed to be redone on further review or with further information
  - Be flexible: the results of one test can lead to extra tests or the removal of some planned tests
- **When possible, limit the number of systems and configurations needing testing and the number of tests: each adds time and complication.**
- **Even with these caveats, the testing was mostly worth doing...**
- **Truth to tell - most systems were similar in performance. At least we had proof however.**





## IV. *Determining the Players*

- **Evaluate partial information and choose finalists**
  - Of course, some elimination can be done before the benchmarking, resulting in an easier evaluation
- **Inform the finalists and those not so "fortunate"**
- **Negotiate with each finalist, telling them of their pros & cons**
  - Use benchmarks to improve hardware bid or increase discount.
- **Be vendor specific, but disclose as little as possible regarding the competition.**
- **Keep a running "gripe" list per vendor - open issues, unresolved problems, unanswered questions**
- **Iterate previous line to increase bang for the buck**

## *The Players: Getting the best proposals*

- **Help each vendor develop the best proposal**
  - Frequent contact to be sure they are on the right track
  - Aim high until they tell you otherwise
  - Nothing should be a surprise come "D-Day"





## V. *Picking the Winner*

- **We boiled down all of the information into a comparison chart of important factors**
- **Educate the other decision makers**
- **Get letter of intent or full quote from selected vendor**
- **Specify deliver dates and failure-to-deliver penalties**
  - Make them back up their promises
- **State acceptance and rejection criteria (for failure to meet specifications)**

## *Picking the Winner: Breaking the bad news*

- **Inform other participants of decision and reasons**
  - It's a good idea to cushion the blow, so they will speak do you again
  - Tell them their good points, and the reason they lost. Gently.





## VI. *Testing the Installation*

- **Install the components.**
- **Test the individuals, and the entire facility**
- **Verify that the acceptance criteria were met**

# VII. *Insights*

- **An important step was performing our own set of benchmarks on the exact (or as close as possible) workstations and server models which were under consideration.**
- **SPECmarks don't cut it for us**
- **Also important to have a good, knowledgable, committee to negotiate, explore and decide**
- **The quality of the salespeople and technical people from each vendor made a difference**
- **Get as far deep in the companies as you can**
- **Push until they scream, and then push a little more**