March 3, 2004

David W. Pershing
Senior Vice President for Academic Affairs
205 Park
Campus

RE: Proposal to Establish M.S. and Ph.D. Degrees in Computing

Dear Vice President Pershing:

At its meeting of February 23, the Graduate Council voted to approve a proposal to establish M.S. and Ph.D. degrees in Computing within the School of Computing.

The new degrees reflect the growth of computing as a scientific discipline, the broad reach of computing into many other disciplines, and the societal importance of computing. These new degrees will serve our students who are involved with computing but who are outside the traditional boundaries of the current Computer Science degrees.

A copy of the proposal is attached for your approval and transmittal to the Academic Senate.

Sincerely,

David W. Chapman
Assoc. V.P. for Graduate Studies
Dean, The Graduate School

Encl.

XC: Robert B. Roemer, Interim Dean, College of Engineering
Christopher R. Johnson, Director, School of Computing
Martin Berzins, School of Computing

SECTION I - Program Request

Utah State Board of Regents Proposal for the Initiation of New Masters and Doctoral Degrees in Computing

Institution Submitting Proposal:	University of Utah
College in Which Program Will Be Located:	College of Engineering
Department in Which Program Will Be Located:	School of Computing
Program Title:	Graduate degrees in Computing
Recommended Classification of Instructional Programs (CIP) Code:	
Area(s) of Emphasis or Academic Specialty:	Computing
Degree(s) to be Awarded:	Master of Science in Computing Doctorate of Philosophy in Computing
Proposed Beginning Date:	Fall Semester 2004

Institutional Signatures:
March 2004

Chris R. Johnson
Director, School of Computing

Robert Roemer
Interim Dean, College of Engineering

David S. Chapman
Dean, Graduate School

David W. Pershing
Senior VP Academic Affairs

Lorris Betz
Interim President

The University of Utah requests approval to offer M.S. and Ph.D. degrees in Computing effective Fall 2004. This program has been approved by the University of Utah Board of Trustees on ______________.

SECTION II – Program Description

II.A. Program Description

II. A. 1. What is computing

The creation of the School of Computing at the University of Utah reflects a large and sustained growth in the emerging multidisciplinary area of Computing. Computing, as distinguished from traditional Computer Science, is the study and solution of a new class of multidisciplinary problems whose solution depends on the combination of state-of-the-computer science *coupled with domain-specific expertise* in such areas as medicine, engineering, biology, and geophysics. A recent article in the Computing Research News (the newsletter of the academic discipline) was entitled "Computing > (contains) Computer Science" and described Computing as "the integration of Computer Science and other disciplines to address problems of wide interest." The same article (Appendix C) defines such applications as computation-intensive or as data-intensive and shows the relationship between these areas in Figure 1 below.

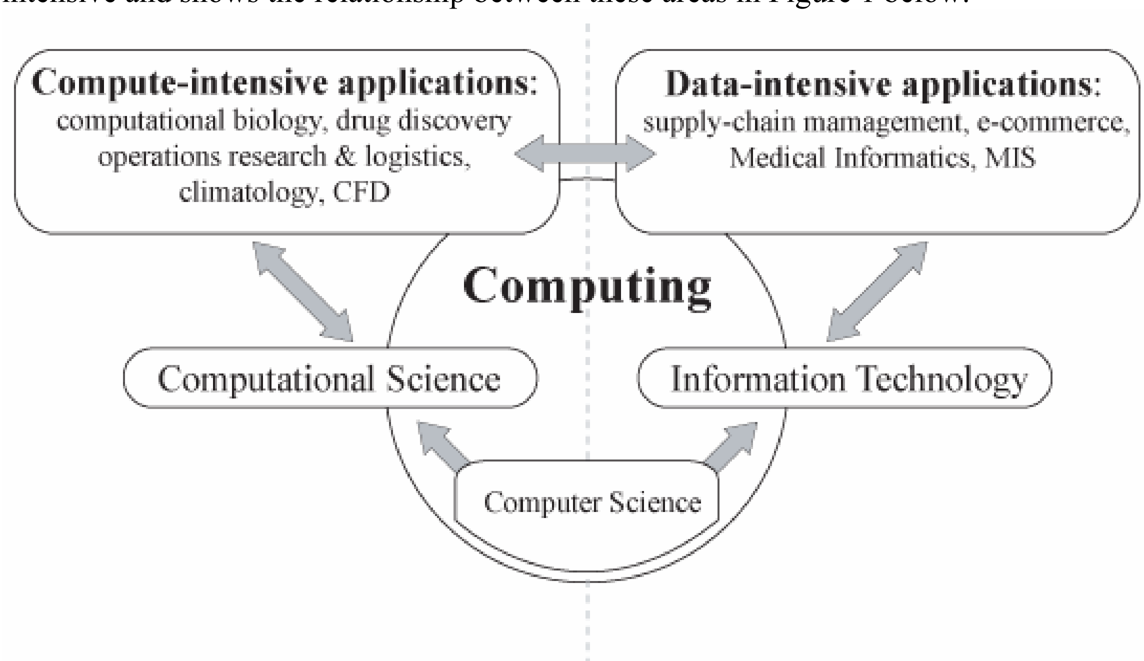

Figure 1: Relationships between Computing, Computer Science and Applications

Substantial growth in Computing at Utah has arisen through the world-leading efforts of several multidisciplinary research and education initiatives, including the Scientific Computing and Imaging (SCI) Institute, the Department of Energy (DOE) CSAFE project on modeling accidental fires and explosions, the NSF Grid-Computing Environment for Research and Education in Computational Science and Engineering, and the newly announced University priority to create the Brain Institute.

In 2000, the University of Utah made the expansion of the Department of Computer Science into the School of Computing an institutional priority. Considerable resources from the College of Engineering, the Senior Vice President for Academic Affairs, and the Utah State Legislature supported this expansion, and it resulted in the growth of the faculty of the School of Computing from 20 members in 2000 to 30 members in 2003. The intended results are well on their way to realization: the School of Computing is now advancing once again into the forefront of the emerging discipline of Computing, and is now poised to reassume the preeminent place it held in Computer Science in the 1960s.

Our faculty expansion has resulted in a significant research growth (largely from federally funded sources: National Science Foundation, National Institutes of Health, Department of Energy, Defense Advanced Research Projects Agency) with external funding increasing from $2M to $14M over the last decade. The bulk of this growth is attributable to research in the multi-disciplinary area of Computing, perhaps the most notable example being the world-leading research of the Scientific Computing and Imaging (SCI) Institute.

In the case of the School of Computing, such growth does not occur in isolation but rather has enormous positive impact on the community as a whole, impact that extends far beyond the simple benefits of infusing millions of dollars annually into the Utah economy. School of Computing and Scientific Computing and Imaging faculty are already working with faculty, industry, the national laboratories and healthcare providers across both the campus and the nation on multidisciplinary problems such as medical applications in cardiology, radiology and neurology, in addition to a large number of applications in engineering and science, such as geophysics, chemical engineering, molecular dynamics, aerospace fluid mechanics, combustion, atmospheric dispersion, and robotics.

II. A. 2. Computing as an emerging academic pursuit

The new faculty who were brought to Utah with the intention that they would direct their efforts toward such cutting-edge research initiatives were also recruited with the expectation that they would train the next generation of computer and computing scientists to do the cross-disciplinary work so important to tomorrow's high-tech industry.

The new faculty who work in multidisciplinary scientific computing and visualization and would be directly associated with the proposed Computing degree are Professors Whitaker (Ph.D. from the University of North Carolina, Chapel Hill), Silva (Ph.D. from SUNY Stony Brook), Kirby (PhD. from Brown University), and Berzins (PhD. from the University of Leeds, UK).

This growth needs to be complemented with graduate programs that reflect the breadth and the multidisciplinary nature of the activities and the aspirations of our students, Utah’s industry and business, and our faculty. The proposed graduate degree program in Computing is also designed to complement existing multidisciplinary programs in areas such as Neuroscience and Computational Engineering and Science. Until we can put into place the educational programs that will permit this training to occur, these scientists will not be fulfilling their potential and Utah will not realize the highest return on its investment.

Multidisciplinary Computing is one of the fastest growing research areas in the US and Europe. Computing is the study and solution of a new class of multidisciplinary problems whose solution depends on the combination of state-of-the-art computer science coupled with domain-specific expertise. Examples of such problems are:

- How can we efficiently store, model, visualize and understand the mass of data generated by the human genome program?
- How might we digitize patient records such that they help medical practitioners rather than drown them in a mass of irrelevant data and baroque procedures?
- How can we model disease transmission in populations that are structured in complicated ways, such as the vectors of the West Nile virus?
- How can we use the extraordinary amount of information recorded by point-of-sale checkout machines to better coordinate retail industries with the shoppers that drive them?
- How might software agents be built such that we trust them to do our shopping online?
- How might we model, simulate and visualize the functions of the heart and brain to better diagnose and treat cardiac and neural abnormalities with a view to improving the quality of life?
- How might we compute solutions to realistic physical models of dangerous situations such as explosions with a view to improving safety?

The next wave of industry growth will focus on business opportunities resulting from the answers to questions such as these. The faculty at the School of Computing are already performing research in these multidisciplinary approaches. This proposal for new graduate degrees in Computing seeks to provide an academic program that encapsulates the pursuit of these challenges into a marketable credential for Utah students.

Two key features of the proposed Computing degree structure are particularly designed to meet this student expectation. Not only is the new Computing degree designed to integrate knowledge from many starting points (engineering, mathematics, physics, medicine), but its track structure makes it possible to build natural and student-centered collaborative academic programs across the University.

II. A. 3 Managing a multi-disciplinary program

The proposed Computing Degree structure will operate at both the Masters and Doctoral level and will be interdisciplinary through its track structure. Each track will have a minimum of six faculty members who will form a Track Faculty Committee or TFC. The TFC will, under the leadership of the Director of Graduate Studies in the School of Computing, be responsible for the creation and subsequent administration of a track. This track structure will make it possible for the Computing degree to be applicable to emerging multidisciplinary problems with a maximum of efficiency in a sound academic manner. We note that academic tracks have been shown to be a successful mechanism for offering a variety of educational opportunities within a larger degree option. For example, the Department of Bioengineering has tracks in Bioinstrumentation, Biomaterials, Biomechanics, Computational Bioengineering, and Neural Interfaces. Similarly, the Department of Biology offers options in Biology, Molecular Biology, Biological Chemistry, and Neuroscience. In a similar vein, the School of Business operates its graduate program under a structure of specializations, which students select at the time of admission.

Examples of possible academic tracks formed under the umbrella of the Computing degree include:

(i) Scientific Computing (currently under development by a TFC)
(ii) Computer Graphics and Animation (currently under development by a TFC)
(iii) Software Engineering
(iv) Applied Robotics

A key role of the TFC will be to ensure that the track has academic standards consistent with and as academically rigorous as those of the participating departments, schools or institutes at both the masters and doctoral level.

An unusual feature of this program is that, because of the multidisciplinary nature of the problems under examination, no single student will arrive from a traditional undergraduate degree having mastered all of the background necessary to study a given problem in depth. The aim is to create a degree that accepts students from a variety of backgrounds. In particular the degree will create a balance between traditional computer science and problem-specific content in the training tailored to the common needs of the students and the multidisciplinary application academic discipline.

Our administrative plan follows a common format used by many programs that are organized into tracks. From the perspective of the university structure, the Computing Degree will be administered through the School of Computing, under the Director's existing administrative structure, including faculty service roles on committees within the school and staff support.

From the perspective of the faculty, we will offer two degrees with full faculty participation in both degrees. Any SoC tenure-track faculty member may join any Track Faculty Committee (TFC) to form and administer a track. Any SoC tenure-track faculty member may advise a student in any track, whether or not a member of the TFC. SoC auxiliary faculty members with long-term instructional relationships to the school may join any TFC so long as the majority of TFC members are tenure-track – this provision ensures the stability of the track program for students in the pipeline.

From the perspective of prospective students, the SoC will offer two graduate degrees, Computer Science and Computing, the latter with a number of tracks. Applicants must choose which degree they intend to pursue at the time of their application; however, the procedure to allow a student to request a transfer from one program to the other is the same the current procedure for transferring between the MS and PhD in Computer Science.

From the perspective of enrolled students, there will be a single operational chain of command that manages their degree programs. Students form thesis or dissertation committees that are monitored by a Graduate Studies Committee. The Graduate Studies Committee is led by a faculty member, the Director of Graduate Studies, who has the day-to-day operational oversight of student programs, reporting in turn to the Director of the school. The role of the TFCs is to focus on establishing track policies for all students within the track, leaving individual student issues first to the student's advisory committee, and second to the Graduate Studies Committee and the Director of Graduate Studies, with final appeal to the Director of the School of Computing.

The formal academic structure has been discussed in detail by the School of Computing faculty; this discussion has resulted in a document describing mechanisms for the management and quality control of tracks in a way that is consistent with the existing standards in the College of Engineering. Guiding principles have been developed for the administration of the Computing Degree tracks:

1. No student shall be awarded a degree outside a properly established track.
2. A track is defined and administered by a group of faculty that will be referred to as the Track Faculty Committee or TFC. In order for a TFC to be *valid* it must meet the following criteria:
   - The minimum size of a TFC shall be 6.
   - The majority of a TFC's members must be tenure track faculty in the School of Computing and all members of a TFC must be long-term instructional faculty at the University of Utah.
   - Any member of the TFC must serve on the TFC for a minimum of one year.
3. Any regular SoC faculty may join any TFC. Any SoC long-term instructional (LTI) auxiliary faculty may join any TFC as long as the act of joining does not invalidate the TFC. Resigning from a TFC is solely up to the discretion of each individual TFC member.
4. The TFC is responsible for the creation and subsequent administration of a track once it has been established. The TFC is also responsible for deciding when to terminate a track. TFC members are expected to participate in teaching courses in the track and in supervision of graduate students in that track. The requirements for a specific degree program track will be individually described in our handbook along with the members of that track's TFC.
5. If a TFC becomes invalid then no new students may be admitted to the track and no change to the existing track requirements can be made. If and when an invalid TFC becomes valid either by member resignations or by the addition of new members then student admissions and requirements changes can once again be made.
6. Creating a new track or eliminating an old track will also require approval by the Director of the School who will consider recommendations from the Graduate Studies Committee and other input as appropriate.

In order to provide consistency for students, should a track be eliminated, students in good standing within the track will be allowed to finish under the track requirements that were in place when they entered the program.

Graduate School requirements will serve as a guide for the formation of tracks; tracks may change as needs arise within the broader School of Computing faculty and student needs.

II. A. 4. Faculty participation

All School of Computing LTI faculty may participate in the Computing degree program, as Track Faculty Committee members, and as department members for issues requiring full faculty oversight. The School of Computing faculty members are listed in Appendix B.

Faculty from other departments may participate as Track Faculty Committee members, but must be long-term instructional faculty (i.e., regular faculty or designated long-term instructional).

II. A. 5 Details on the proposed M.S. in Computing

The M.S. in Computing is a research degree offered through the Graduate School. A student who has been accepted by the Graduate School is formally admitted to candidacy for the M.S. degree at the recommendation of the student's supervisory committee.

Admission to candidacy occurs after the student:

- Selects an approved track within the Computing degree program
- Forms a supervisory committee,
- Files an approved Program of Study form,
- Passes the comprehensive examination, and
- Submits an approved thesis proposal.

Supervisory Committee. An M.S. committee consists of three members. A committee typically consists of School faculty, but may include one qualified external member. The committee should be formed by the second semester of enrollment in the M.S. program, although a committee may be revised later by petition to the Graduate Studies Committee.

Any School of Computing regular faculty member may serve as a supervisory committee chair. Research or clinical faculty may chair supervisory committees if accorded that privilege by the regular faculty. Individuals who are not faculty members may serve on supervisory committees if nominated by the regular faculty on the committee, and endorsed by the Graduate Studies Committee and School Director.
The Dean of the Graduate School must grant final approval of all supervisory committees.

Required Courses. The curriculum requirements for M.S. students will be designed by the TFCs to ensure that all students who receive an M.S. degree have a working knowledge of those topics in their specialty in computing that are deemed fundamental by the faculty.
M.S. students must earn a grade of B or better in each required class, and achieve an overall GPA of 3.5 in those classes.

Program of Study. Course work listed on the approved Program of Study form must consist of at least 30 semester hours of graduate course work and thesis research. At least 6 semester hours of thesis research (CS 6970 equivalent) and 20 semester hours of graduate course work must be included.

At least 24 semester hours must be completed in resident study at the University of Utah. Students must be registered for a minimum of 3 semester hours during the semester in which the thesis is defended.
The Program of Study form should be filed with the School in the second semester of study and with the Graduate School prior to taking the comprehensive examination. The Program of Study form must be submitted to the Graduate School by the last day of the semester preceding the semester of graduation.

Thesis Proposal. The student should prepare and receive approval for a thesis proposal by the end of the third semester of study (not counting summers). A copy of the thesis proposal must be in the student's file. The thesis proposal must be approved at least one semester prior to the semester of the thesis defense.

Comprehensive Examination. The comprehensive examination for M.S. students is coupled to the thesis proposal. It consists of an oral examination on the thesis proposal and research area in a very broad sense. This examination is administered by the student's supervisory committee and should be completed by the end of the student's third semester of study (not counting summers) as a graduate student in the School. The examination should serve as the defense of the student's thesis proposal as well as to establish competence in the research area. The examination must be completed at least one semester prior to the semester of the thesis defense.

Completing Program of Study. An M.S. student is expected to devote the necessary time to courses and research in order to make satisfactory progress toward the degree. Satisfactory progress includes personal participation in the research and teaching environment of the School on a day-to-day basis.

A full time student working on an M.S. program is expected to complete the degree requirements within two calendar years. Beyond this period a student generally does not receive graduate financial support from the School, and the tuition waiver does not apply. A student must petition the Graduate Studies Committee to continue beyond the third year. The Graduate School limits M.S. programs to four years.

II. A. 6. Ph.D. in Computing

The Ph.D. in Computing is a research degree offered through the Graduate School. It is awarded to a candidate who has demonstrated breadth in the areas represented by the School of Computing in general, and depth in a research specialty within the School of Computing. The latter is exhibited through the writing and defense of a dissertation that reports substantial original contributions in an approved area of research.
The University, at the recommendation of the student's supervisory committee, formally admits a student who has been accepted by the Graduate School to candidacy for the Ph.D.. Admission to candidacy occurs after the student:

- Selects an approved track within the Computing degree program
- Forms a supervisory committee,

- Files an approved Program of Study form,
- Completes the core course requirements,
- Passes the qualifying examination, and
- Submits an approved dissertation proposal.

Supervisory Committee. Each student forms a supervisory committee whose members guide the student's research program. The committee conducts the student's qualifying examination and dissertation defense. A Ph.D. supervisory committee consists of five faculty members. At least three faculty members must be from the School, and at least one member from outside the School of Computing. Any School of Computing regular faculty member may serve as a supervisory committee chair. Research or clinical faculty may chair supervisory committees if accorded that privilege by the regular faculty. Individuals who are not faculty members may serve on supervisory committees if nominated by the regular faculty on the committee, and endorsed by the Graduate Studies Committee and School Director.

The Dean of the Graduate School grants final approval of all supervisory committees. Students must form this committee by the end of the second semester of study, although a committee may be revised later by petition to the Graduate Studies Committee.

Required Courses. The curriculum requirements for Ph.D. students will be designed by the TFCs to ensure that all students who receive a Ph.D. degree have a working knowledge of those topics in their specialty in computing that are deemed fundamental by the faculty.

Program of Study. Course work listed on the approved Program of Study form must comprise at least 50 semester hours of graduate course work and dissertation research, exclusive of independent study. Graduate course work applied toward an M.S. degree may be included. At least 14 semester hours of dissertation research (CS 7970 equivalent) and 27 semester hours of graduate course work must be included.
One year of study must be spent in full-time residency at the University (i.e., the student must enroll for a minimum of nine hours per semester for two consecutive semesters, summer optionally excluded). After the residency requirement is fulfilled, registration for three semester hours of Ph.D. Dissertation Research is considered a full load.
The Program of Study form should be filed with the School in the second semester of study and with the Graduate School prior to taking the qualifying examination. The Program of Study form must be submitted to the Graduate Records Office no later than the last day of the semester preceding the semester of graduation.

Comprehensive Examination. The Ph.D. Comprehensive Examination tests for breadth of knowledge across the discipline of computing. Each TFC shall offer a comprehensive examination for its track members.

Dissertation Proposal. The student should prepare and receive approval for a dissertation proposal by the end of the sixth semester of study (not counting summers).

Qualifying Examination. After passing the Comprehensive Examination, all Ph.D. students must pass a Qualifying Examination, as specified by the Graduate School. The Qualifying Exam consists of a written part, to be conducted first, and an oral part. The written part of the Qualifying Examination will cover the candidate's general area of specialization in sufficient depth to demonstrate preparation for conducting Ph.D.-level research.

The oral part comprises the dissertation proposal defense. At the supervisory committee's option, it may also include follow-up questions relating to the written part of the exam. Students should pass their Qualifying Examination by the end of their sixth semester of study, not counting summer enrollment. The Qualifying Examination must be completed no less than one semester prior to defense of the dissertation.

Completing Program of Study. A Ph.D. student is expected to devote the necessary time to courses and research in order to make satisfactory progress toward the degree. Satisfactory progress includes personal participation in the research and teaching environment of the School on a day-to-day basis.

Dissertation. The completed dissertation must be published either in its entirety (through a legitimate publisher of the student's choice or through University Microfilms) or as one or more articles accepted for publication in approved scholarly journals. An abstract of each dissertation must be published in University Microfilms' Dissertation Abstracts International. Other requirements are common with the M.S. degree program and are discussed below.

II.B. Purpose of degree

The success of the research mission reflects society's need for work in these multidisciplinary areas, a need that cannot be fulfilled unless we attend to training the next generation of scientists to work closely with the kinds of industries we most want to bring to Utah: those specializing for example in biomedical technologies, as well as those involved in software development, industrial design, and other high-technology endeavors. To provide this training, we require a degree program to answer growing demands from industry on the one hand and students on the other. Students participating in high-tech research areas with our faculty are at present limited to academic program choices that so far reflect neither the changing multidisciplinary demands of these employers in industry nor the actual breadth and multidisciplinary nature of their training and achievements.

A growing number of students have been forced to respond to the interests of industry and funding agencies by specializing very broadly in multidisciplinary Computing or by finding increasingly unsatisfactory ways to tailor their programs to existing degrees. For example the number of graduate students in the SCI Institute has grown from four to twenty-five over the last decade. While many of these students participate in the high-quality Computer Science graduate program, their multidisciplinary needs and aspirations are somewhat different from those satisfied by conventional Computer Science, which provides more emphasis on learning about computer hardware, operating systems, and theory, and less on how to apply these tools to real-world interdisciplinary problems. Conversely, these students also require serious study in computer science, and their need for this content makes other Engineering graduate programs equally inappropriate. Even the Master's degree in Computational Engineering and Science (CES), which has grown by a factor of four over three years and which has helped somewhat to provide for interdisciplinary education for engineers and scientists, does not fulfill the specific needs of graduate students in computing. For this reason, student inquiries in the proposed new program have been widespread and sustained, and existing students working in the School of Computing, the SCI Institute and the CES program are constantly making inquiries as to whether they might complete their studies under the new proposed computing degree requirements.

II.C. Admission

Admission requirements for the M.S. and Ph.D. in Computing will be similar to the admission requirements for the M.S. and Ph.D. in Computer Science and the admissions process will occur at the same time. Students will select the program (Computing or Computer Science) that they wish to apply for on the admissions form. The School of Computing Graduate Admissions Committee will have representatives from both Computer Science and Computing who will assess the quality of the applicants.

All graduate admissions information will be available on-line on the School of Computing's web site (www.cs.utah.edu). Additionally, there will be a list of Frequently Asked Questions (FAQ). We provide a draft of admission requirements next.

DRAFT admissions requirements

Graduate Admissions for the Computing and Computer Science Degree Programs

Our admissions standards are high and competition for the limited number of positions is rigorous. Admission is based on an evaluation of both an applicant's academic profile and research potential.

Required Application Materials

The deadline for consideration for Fall 200X will be January 15, 200X: the on-line application must be completed, and all other materials must be received by this date. Applicants are evaluated for admission effective Fall semesters. Applications must be received at the School of Computing by January 15 for the following Fall Semester.

Information students will need to complete the on-line form is:

- Whether the student will apply for the Computing or Computer Science graduate program.
- The name, phone number and email address of the three people who will write letters of recommendation.
- Most recent TOEFL (if applicable) scores and dates of the examination.

All application materials should be sent to:
Graduate Admissions Coordinator
University of Utah
School of Computing
50 S. Central Campus Drive, room 3190
Salt Lake City, UT 84112-9205 USA
grad-admissions@cs.utah.edu

More information about the application process, requirements, financial aid, etc., is available from the School of Computing Graduate Handbook, available from our website: www.cs.utah.edu/dept/admissions.

II.D. Student Advisement

Student advisement will be consistent with the well establish advising practices within the School of Computing for the current M.S. and Ph.D. programs in Computer Science. We will update the School of Computing Graduate Handbook to reflect the M.S. and Ph.D. in Computing.

II.E. Justification for Number of Credits

Not applicable: the Number of credit or clock hours does not exceed the requirement of 36 hours beyond the baccalaureate for the MS degree.

II.F. External Review and Accreditation

No external consultants were involved in the development of this proposal.

At this point in time, there are not any formal professional accreditation mechanisms in place for Computing degrees within the national academic sphere. The Computing Degree would be reviewed and accredited in the same manner as the existing Computer Science degree: through the University of Utah Graduate School review and accreditation process.

II.G. Projected Enrollment

Enrollment/recruiting

The School of Computing will recruit graduate students for the Computing degree in concert with its efforts to recruit into the Computer Science degree (e.g., through the school website, advertising as needed, etc.). In addition, TFCs will recruit students through contacts with their colleagues, at national scientific meetings, and through other media relevant to their disciplines.

We base the projections below for enrollment on three data sources: the growth of the carrying capacity of research funding in support of graduate students, the expected increase in advising capacity of the School of Computing faculty as new faculty members ramp up their research programs, and the initial transfer of students from the Computer Science degree to the new program.

	MS	PhD	Faculty participants	Mean student/faculty ratio
Year 1	6	6	12	1:1
Year 2	10	10	15	1.33:1
Year 3	15	15	15	2:1
Year 4	15	20	18	2:1
Year 5	15	25	20	2:1

Table 1: Computing Degree enrollment projections

The current enrollment in the Computer Science degree programs is 34 MS and 64 Ph.D. students; of these we expect approximately 10 students will transfer into the new program.

Based on steady-state projections in the late 1990s, graduate students were admitted at about a one-to-one ratio to the existing faculty into the Computer Science degree. Having increased the faculty by more than 10 since that time, and increased the expected supervisory load of each faculty member, we believe that the enrollment of students in years 2-5 can be projected as above, reaching steady-state by the fifth year.

In the first year of the program, a small fraction of students already enrolled in the M.S. or Ph.D. Computer Science Degree programs may wish to transfer to tracks that better fit their scholastic needs. The following transfer mechanisms are already stated in the School of Computing Graduate Handbook (Sections 1.61 and 1.62):

Transfer into the Ph.D. Program

A student wishing to transfer to the Ph.D. program must submit the following material to the Graduate Studies Committee:

1. A "goal letter" explaining the student's motives and describing the intended research area.
2. Letters from three current School faculty members supporting the application. In the event that the applicant's supervisory committee has been formed, then at least one of these letters must be from the chair of that committee.
3. A University of Utah transcript.

Applications will not be reviewed unless at the time of application the applicant has completed at least two academic semesters of graduate study. Summer term does not count in this tally.

Transfer into the M.S. Program

Students currently enrolled in the Ph.D. program who wish to transfer to the M.S. should make their request in writing to the Director of Graduate Studies. Supporting letters from three faculty members expressing willingness to serve on the student's M.S. Supervisory Committee must also be provided.

II. H. Expansion of Existing Program

Not applicable: This is a proposal for a new program.

II. I. Faculty

No additional faculty will be required to support this program (see section V. – Finance).

The School of Computing faculty are well prepared to launch these degree programs upon approval. The participating faculty working together in interdisciplinary research have already established informal links that will become the foundations for Track Faculty Committees (TFCs). Additional faculty will be mentored into the system as part of the normal faculty mentoring processes for junior faculty in the School of Computing.

II. J. Staff

No additional staff will be required to support this program (see section V. – Finance).

II. K. Library

University of Utah and School of Computing library resources are adequate to support this degree program.

II. L. Learning Resource

University of Utah and School of Computing learning resources are adequate to support this degree program.

II. M. Institutional Readiness

University of Utah and School of Computing administrative structures are adequate to support this degree program.

SECTION III - Need

III. A. Program Necessity

The School of Computing was created to broaden its mission to a wider scope of computing, including:

- a greater range of undergraduate programs (e.g., a departmental Honors degree which has been established),
- multi-disciplinary programs: e.g., Computational Engineering and Science M.S. program (an existing program in which the SoC participates), Bioinformatics (under discussion), Software Engineering (under discussion), Computer Graphics and Visualization (proposed track within the Computing Degree), and
- research institutes (e.g., the SCI Institute grew out of the SoC).

The current graduate degrees offered by the School of Computing include the M.E., M.S., M. Phil. and Ph.D. degrees – all in Computer Science. The School of Computing faculty studied the issue of including the interdisciplinary degrees described above as tracks in our current degrees, and decided after careful deliberation that such an approach would not only damage the content of the CS degree, but would not reflect the nature and substance of the new programs.

The success of the School's research mission reflects society's need for work in these multidisciplinary areas, a need that cannot be fulfilled unless we attend to training the next generation of scientists. Such training requires a degree program to answer demands from industry on the one hand and students on the other. Students participating in high-tech research areas with our faculty are at present limited to academic program choices that so far reflect neither the changing multidisciplinary demands of employers in industry nor the actual breadth and multidisciplinary nature of their training and achievements. A growing number of students have been forced to respond to the interests of industry and funding agencies by specializing very broadly in multidisciplinary Computing or by finding increasingly unsatisfactory ways to tailor their programs to existing degrees.

As an example, the number of graduate students in the SCI Institute has grown from four to twenty-five over the last decade. While many of these students participate in the high-quality Computer Science graduate program, their multidisciplinary needs and aspirations differ from those satisfied by conventional Computer Science. Conversely, the large computing content and emphasis of their work makes other Engineering graduate programs equally inappropriate for their needs. The number of graduate students in the complementary Master's degree in Computational Engineering and Science (CES) has grown by a factor of four over three years, but while this program has helped somewhat to provide for interdisciplinary education for engineers and scientists, it does not fulfill the specific needs of graduate students in computing. For this reason, student inquiries in the proposed new program have been widespread and sustained, and existing students working in both the School of Computing, the SCI Institute and the CES program are constantly making inquiries as to whether they might complete their studies under the new proposed computing degree requirements.

Four new faculty members who have been hired in scientific computing, visualization and graphics are already transferring graduate students from other institutions to work with them on multidisciplinary Computing problems and are also recruiting students here. As a result there are about fifteen new students both in the School of Computing and in the SCI Institute who are working on multidisciplinary topics in addition to the students already here. The creation of an additional joint faculty position with Bioengineering will, when filled, add students to this group. The School of Computing is finding it increasingly difficult to accommodate students through unwieldy existing programs or specially tailored individual programs. We already have the faculty and the courses in place to provide the new degrees; we require only permission to institute our new degree structure. It is essential that we fulfill our promise to put these degrees in place to satisfy the demand we have anticipated and planned for and that we are now experiencing.

It is worth stressing that these are not developments in isolation. Though we were leaders in this area three years ago and are still ahead of other institutions, at this time, our competitors, both national and international, are also reflecting this fundamental change in the multidisciplinary research, teaching and outreach aspects of our subject. We must now begin not only to respond in an ad hoc manner to student demand but also to recruit students from appropriate bachelors degree programs in order not to lose momentum provided by our research funding and by the faculty expansion in the School of Computing. Given that the need is strong and there is no additional cost implication, coupled with the fact that there is campus-wide, statewide and nationwide interest in our new computing programs, it is our firm hope that the Board of Regents will act quickly to permit us to formalize these offerings.

III. B. Labor Market Demand

It is well known that an educated workforce will attract businesses to Utah; a workforce educated in technology will attract jobs that are well paid. Local industries recognize that multidisciplinary appraoches to problems are the future of business growth in Utah. There is a need for graduates who have both an advanced level of computing and who can also work in applications domains at a high level. This is particularly true in areas such as biomedical visualization and advanced biomedical computation. Similarly the latest generations of computer games require high levels of computational and visualization skills to produce the realistic and lifelike action now coming onto the marketplace. There are a number of Utah employers, many of which are spin-offs from the University of Utah, working at this overlap:

- Biotechnology companies such as Myriad and the Watson Labs.,
- Companies using software engineering such as Novell, Altiris, LAN Desk, Legato and Attensity, emWare, Applied Signal Technology and Red Rock Software.
- Aerospace and military contractors, including L3 Communications, ATK Thiokol and TRW, Inc.
- Specialist Engineering companies such as locally-owned Reaction Engineering International and Engineering Geometry Systems, and divisions of petrochemical corporations including BP and Sinclair Oil.
- Computer Gaming Companies such as Microsoft's Games Division, Saffire Software, and Sculptured Software.

These are only a few examples of many such companies with both large and small workforces who are making important contributions to the Utah economy.

At the national level, Computing M.S. and Ph.D. graduates will be highly sought by similar industrial concerns, the national research laboratories and academia. Salaries for graduating Ph.D. students in such companies vary greatly depending on the individual and the area of work but are in the range of $50,000 to $100,000. The higher figures are less common but still used by employers such as DOE National Laboratories.

III. C. Student Demand

Despite of the lack of degree programs that reflect their interests and the demands of the market, a growing number of students are responding to the interests of industry and funding agencies by specializing in multidisciplinary Computing and finding increasingly unsatisfactory ways to tailor their programs to our existing degrees. For example about a dozen graduate students in the Computational Engineering and Science Program have expressed am immediate interest in a multidisciplinary Ph.D, program when polled at a recent meeting. This reflects the widespread and sustained student interest in the proposed new program. As was also noted above, students arriving with recently hired faculty have multidisciplinary research needs that will be best met by the new program. It is essential that we fulfill our promise to put these degrees in place to satisfy the demand for which we have anticipated and planned.

Our proposal is consistent with the creation of the School of Computing and the development of new degree programs and allows us to create new multidisciplinary programs in a sound academic way while responding quickly to present and emerging applications-driven opportunities. We must now begin to recruit the best students from appropriate bachelors degree programs in order not to lose the momentum provided by our research funding and by faculty expansion in the School of Computing. Moreover, as noted above, students participating in these research areas with our faculty are at present limited to academic program choices that as yet reflect neither the demands of employers in industry nor the actual breadth and multidisciplinary nature of their training and achievements. Our current students who are working on multidisciplinary problems are overburdened by trying to meet their own goals in addition to the traditional expectations within program structures that were not created to respond to their needs.

Last year, an external committee made up of nationally recognized leaders in Computer Science and an internal University committee reviewed the Graduate Program at the School of Computing. Using the reports from the review committees along with materials submitted by the School of Computing, the Graduate Council at the University of Utah recently submitted their final review of the School's Graduate Program. In the Graduate Council's summary of School's Graduate Program under the Commendation section, they noted, "The proposal for the new M.S. and Ph.D. degrees in Computing builds on the applied and interdisciplinary strengths of the School and the University of Utah and provides an exciting opportunity for growth."

III. D. Similar Programs offered in the Utah System of Higher Education

There are no comparable programs offered within the Utah System of Higher Education.

III. E. Collaboration with and Impact on Other USHE Institutions

We anticipate that the greatest impact the proposed Computing degrees will have on other USHE institutions will be in offering a new option for which USHE institutions can prepare undergraduate students. Students in engineering and science disciplines at Utah sister institutions who wish to pursue multidisciplinary topics for their graduate studies will be prime candidates for this program.

III. F. Benefits

By implementing this proposal, the University of Utah and USHE will capitalize on the University of Utah's initial investments in the School of Computing. Conversely, if we do not fulfill the demand for this program, other institutions will, and their students, businesses, and communities will reap the benefits. Though we were leaders in this area three years ago and are still ahead of other institutions, at this time our competitors, both national and international, are also reflecting this fundamental change in the research, teaching and outreach aspects of our subject.

Currently, there are an increasing number of Colleges and Schools of Computing and one School of Informatics in the United States (see Appendix C). There are numerous Schools of Computing throughout the world. Other institutions are creating new departments in some of these areas, e.g., scientific computing, robotics, etc. For example, Carnegie Mellon has a very broad School of Computer Science, which has the Computer Science Department and Institutes that offer M.S. and Ph.D. degrees in Human Computer Interaction, Software Research, Language Technology and Robotics within its scope. In a similar vein, Georgia Tech. has a College of Computing with centers for Computer Systems, Graphics Visualization and Usability, and Information Security. The MIT Media Lab is world-renowned for its multidisciplinary work.

The specific areas of scientific computing and bioinformatics in particular are experiencing major growth across the nation. In 1998, 31 graduate programs in computational science at U.S. Universities had been created. As of 2003, the number had grown to 47. For example, Florida State has launched a School of Computational Science with a large investment, as has UC Davis; other schools are devoting major resources to the its development (e.g., UCSD and Texas have created endowed chairs in the area). Bioinformatics departments are also being formed in such major institutions as the University of Maryland, UCSD, Stanford, UC Irvine, UC Santa Cruz, Yale, Georgia Tech, Indiana University, and RPI, among others. Rensselaer has invested heavily in its Master of Science and Doctor of Philosophy in Multidisciplinary Science. UC Davis has a Computer Science and Engineering Major Program.

In addition, since 1998, 16 new undergraduate degree programs in computational science have been created. We note that this only takes into account one aspect of the proposed Computing graduate degree; there are additional increases in other areas such as software engineering and computer graphics.

Utah has the existing potential to become a recognized powerhouse in Computing; this proposal will accelerate that development into a reality.

III. G. Consistency with Institutional Mission

This Computing degree represents the University of Utah's highest priority doctoral request for the 2003-04 academic year. It is essential for the long-term health and growth of the School of Computing and the College of Engineering. The demand for the degree is high as are the potential rewards for the students who complete the degree.

The mission of the University of Utah is to educate the individual and to discover, refine, and disseminate knowledge. As a major teaching and research university and the flagship institution of the Utah state system of higher education, the University of Utah strives to create an academic environment in which the highest standards of scholarship and professional practice are observed and where responsibilities to students are conscientiously met. It recognizes the mutual relevance and interdependence of teaching and research as essential components of academic excellence.

The proposed degree also fits with the threefold mission of the College of Engineering. The faculty, staff, and students join together to:

1. provide modern, relevant engineering education for undergraduates and graduates that is nationally recognized for its excellence;

2. conduct scholarly research involving both graduate and undergraduate students and make the results available to the industrial sector, government agencies and the general public via presentations, publications, software, patents, technical advice, and graduates;

3. contribute directly to the economy of the State of Utah by providing a resource base for local industry and by assisting with the commercialization of new technologies to improve the quality of life in Utah and the nation.

As a multidisciplinary program originating in the School of Computing, our aim is to develop a graduate program that fits within the University and College Missions and is defined by its own mission statement, which is as follows:

To provide an internationally excellent graduate program in multidisciplinary computing that is broad in its applications, high in its level of academic achievement and capable of addressing the multidisciplinary challenges inherent in Science, Medicine and Engineering.

SECTION IV - Program and Student Assessment

IV. A. Program Assessment

The metrics by which the Computing degree program will be assessed include creation of new tracks, level of faculty participation in tracks, student enrollment and student graduation. The following table shows the goals that have been set for the first year and the fifth year of the Computing degree combined M.S. and Ph.D. program. These goals will be incorporated into the School of Computing strategic planning process and reviewed annually as part of the ongoing commitment to program assessment within the School.

Goal	Year 1	Year 5
Tracks approved	2	2
Faculty participants	12	20
Students enrolled	12	40
Annual graduation rate	1	11

Table 2: Assessment metrics

IV. B. Expected Standards of Performance

Graduates of the Computing MS and PhD program will be held to the international standards of their respective degrees. At completion of a Masters Degree in computing, a student should be able to conduct independent research on a topic directed by an employer or senior academician. The holder of a PhD is expected to possess the ability to lead an independent research program on the topic of his/her choice. These standards are assessed through the traditional examination process for all graduate-level programs, including the defense of an independent dissertation by the doctoral candidate.

IV. C. Student Assessment

Formative assessment: Responsibility for formative assessment will rest with the student's advisor, the student's committee, and the DGS. Metrics will include grades in classes, due progress against degree milestones (program of study, oral exam, dissertation proposal, proposal, etc.), quality of ongoing research efforts, and participation in the intellectual life of the SoC.

Summative assessment: This will be measured in two ways: first, by the quality of the research performed while in graduate school (measured by how well the dissertation is received by the research community, publications produced, talks given, etc.); and second, by the success of the student in the years following graduation (measured by success in landing appropriate jobs, advancement in visibility and stature in the research community, post-graduation publication record, etc.)

IV. D. Continued Quality Improvement

Issues identified by the formative assessment will become evident first. The Director of Graduate Studies and his/her committee will review the progress of each student each semester, based on the formative guidelines and will flag problem areas for individual students. If the DGS discovers that certain kinds of problems are occurring widely, some sort of system-wide correction will be devised.

Issues identified by the summative assessment will become evident only years down the road but are potentially much more serious. If problems with students' education are preventing them from advancing in the field, the entire School will have to become involved in revising the degree requirements and the delivery methods.

SECTION V – Finance

As we noted above, in 2000, the University of Utah responded to requests from students and industry by making the expansion of the Department of Computer Science into the School of Computing an institutional priority. Considerable resources from the College of Engineering, the Senior Vice President for Academic Affairs, and the Utah State Legislature supported this expansion, and it resulted in the growth of the faculty of the School of Computing from 20 members in 2000 to 30 members in 2003. The intended results are well on their way to realization: the School of Computing is now advancing once again into the forefront of the emerging discipline of Computing, and is now poised to reassume the preeminent place it held in Computer Science in the 1960s.

Substantial resources have been made available to the School of Computing to facilitate growth in academic programs such as proposed here. In other words, the investment in this new degree program has already been made. The Computing degree program will be administered through the existing graduate student services mechanisms of the School of Computing, so no new staff will be required. The School of Computing has a shared commitment to admit a graduate class matched to the size of its resources so that essentially all entering students are awarded support for the first year of their programs. This will help make the program accessible to everyone admitted, especially that large percentage of Utah students who already have families to support and may find funding their ongoing educations a challenge. Growth in the graduate programs of the School will result from the ramp-up of research funding of new faculty who have joined the School in the last three years. In FY02, the School generated expenditures of more than $14M in research funding beyond the state appropriation, largely from competitive grants funded by federal research agencies (National Science Foundation, National Institutes of Health, Department of Energy, Defense Advanced Research Projects Agency). These funds primarily support graduate students and salaries for research staff. This figure does not yet reflect research income generated by recently hired faculty and so funding levels are expected to grow substantially in the next few years.

V. A. Budget

The following table presents the costs associated with supporting graduate students at the funding level that is the current norm for graduate students in the Computer Science program. We would expect students in the Computing degree to be treated comparably to students in the Computer Science degree programs

	Year 1	Year 2	Year 3	Year 4	Year 5
Number of students	12	20	30	35	40
Salaries and Wages	180000	300000	450000	525000	600000
Benefits	16200	27000	40500	47250	54000
Current Expense	0	0	0	0	0
Library	0	0	0	0	0
Equipment*	0	0	0	0	0
Travel*	0	0	0	0	0
Total	196200	327000	490500	572250	654000

Table 3: student support costs for the first 5 years of the graduate Computing degree programs.

* Equipment and travel costs will vary by the individual needs of the student and are generally provided by external funding within the research mentor's program

V. B. Funding Sources

No new state funds are requested in support of this program. Costs displayed in Table 3 will generally be born by external funding sought and obtained by the participating faculty, which includes a component to support graduate students. Some first year students may be offered teaching assistantships if their skills match needs currently funded by state funds allocated to the School of Computing. To make an outstanding program, the faculty will seek external (federal and industrial) funding to support fellowships for recruiting top students.

V. C. Reallocation

Not applicable: no internal reallocation in support of this program is requested.

V. D. Impact on Existing Budgets

As noted above, substantial resources have been made available to the School of Computing to facilitate growth in academic programs such as proposed here. In other words, the investment in this new degree program has already been made. The Computing degree program will be administered through the existing graduate student services mechanisms of the School of Computing, and thus no new staff will be required. The School of Computing has a shared commitment to admit a graduate class matched to the size of its resources so that essentially all entering students are awarded support for the first year of their programs. Growth in the graduate programs of the School will result from the ramp-up of research funding of new faculty who have joined the School in the last three years. In FY02, the School generated expenditures of more than $14M in research funding beyond the state appropriation, largely from competitive grants funded by federal research agencies (National Science Foundation, National Institutes of Health, Department of Energy, Defense Advanced Research Projects Agency). These funds primarily support graduate students and salaries for research staff. This figure does not yet reflect research income generated by recently hired faculty and so funding levels are expected to grow substantially in the next few years.

Appendix A: Curriculum

We expect that, as new members join the School of Computing faculty, they will wish to offer additional courses at the graduate level. By organizing our programs into tracks, we will have the opportunity to focus and prioritize the need for new classes based on student demand and a rational, planned curriculum. We list below the existing courses offered within the School.

New Courses to be Added in the Next Five Years –

This course is currently being offered as a special topic, and is an example of the types of courses that shall be developed to meet specific track needs.

Course Number	Title	Credit Hours
CS6XXX	Distributed and Parallel Computing	3

The course consists of a theoretical and practical introduction to high performance parallel computing using shared and distributed memory computers.

All Program Courses

6010 Writing Research Proposals (2 credit hours)

Fundamental aspects of writing computer science research proposals, including thesis, dissertation, and grant proposals. Form, style, substance, and marketing of effective proposals will be considered. Emphasis is placed on developing and presenting clear and compelling ideas. Substantial writing and class presentations is required of all participants. (This is a half-semester course.)

6020 Conducting, Publishing, and Presenting Early-Career Research (3)

This is an independent study offering designed to encourage beginning graduate students to conduct, publish, and present original research early in their graduate careers. A graduate student can earn credit for CP SC 6020 by having a first-authored paper accepted for publication in a top-tier journal or conference and by subsequently presenting the published work in a one-hour research colloquium. The research must be conducted while a graduate student at Utah; the paper must be accepted within two years of enrolling in the graduate program; the journal or conference must be approved by the student's graduate committee; the colloquium must be presented as soon as possible after the acceptance of the paper; and the student must complete these requirements and register for CP SC 6020 within three years of enrolling in the graduate program. CP SC 6020 may not be repeated for credit.

6100 Foundations of Computer Science (3)

Finite Automata and related topics (BDDs, Presburger Arithmetic, and decidable fragments of first-order logic). Automata on Infinite Words, connections with Specification and Verification of Systems. Push Down Automata, Turing Machines, Proofs by Reduction, Diagonalization, Problems in Computability. First-order Logic and Decidability. NP Completeness, P-space Completeness.

6110 Formal Methods for System Design (3)

Study of methods for formally specifying and verifying computing systems. Specific techniques include explicit state enumeration, implicit state enumeration, automated decision procedures for first-order logic, and automated theorem proving. Examples selected from the areas of superscalar CPU design, parallel processor memory models, and synchronization and coordination protocols.

6210 Advanced Scientific Computing I (3)

An introduction to existing classical and modern numerical methods and their algorithmic development and efficient implementation. Topics include: numerical linear algebra, interpolation, approximation methods and parallel computation methods for nonlinear equations, ordinary differential equations, and partial differential equations.

6220 Advanced Scientific Computing II (3)

A study of the numerical solution of two and three dimensional partial differential equations that arise in science and engineering problems. Topics include: finite difference methods, finite element methods, boundary element methods, multigrid methods, mesh generation, storage optimization methods, and adaptive methods.

6300 Artificial Intelligence (3)

Introduction to field of artificial intelligence, including heuristic programming, problem-solving, search, theorem proving, question answering, machine learning, pattern recognition, game playing, robotics, computer vision.

6310 Robotics (3)

The mechanics of robots, comprising kinematics, dynamics, and trajectories. Planar, spherical, and spatial transformations and displacements. Representing orientation: Euler angles, angle-axis, and quaternions. Velocity and acceleration: the Jacobian and screw theory. Inverse kinematics: solvability and singularities. Trajectory planning: joint interpolation and Cartesian trajectories. Statics of serial chain mechanisms. Inertial parameters, Newton-Euler equations, D'Alembert's principle. Recursive forward and inverse dynamics.

6320 Computer Vision (3)

Basic pattern-recognition and image-analysis techniques, low-level representation, intrinsic images, ``shape from'' methods, segmentation, texture and motion analysis, and representation of 2-D and 3-D shape.

6340 Natural Language Processing (3)

Computational models and methods for understanding written text. Introduction to syntactic analysis, semantic analysis, discourse analysis, knowledge structures, and memory organization. A variety of approaches are covered, including conceptual dependency theory, connectionist methods, and statistical techniques. Applications include story understanding, fact extraction, and information retrieval.

6350 Machine Learning (3)

Techniques for developing computer systems that can acquire new knowledge automatically or adapt their behavior over time. Topics include concept learning, decision trees, evaluation functions, clustering methods, explanation-based learning, language learning, cognitive learning architectures, connectionist methods, reinforcement learning, genetic algorithms, hybrid methods, and discovery.

6360 Virtual Reality (3,)

Human interfaces: visual, auditory, haptic, and locomotory displays; position tracking and mapping. Computer hardware and software for the generation of virtual environments. Networking and communications. Telerobotics: remote manipulators and vehicles, low-level control, supervisory control, and real-time architectures. Applications: manufacturing, medicine, hazardous environments, and training.

6470 Advanced Topics in Compilation (3)

Compilation of modern languages. Optimization techniques, register allocation and instruction scheduling, garbage collection, exception handling. Linkers and late-stage compilation and optimization.

6480 Data Communications and Networks (3)

A comprehensive study of the principles and practices of data communication and networks. Topics include: transmission media, data encoding, local and wide area networking architectures, internetwork and transport protocols (e.g., IPv4, IPv6, TCP, UDP, RPC, SMTP), networking infrastructure (e.g., routers, name servers, gateways), network management, distributed applications, network security, and electronic commerce. Principles are put into practice via a number of programming projects.

6520 Programming Languages and Semantics (3)

Examination of the formal and pragmatic ideas behind programming language design. Imperative, functional, logic, object-oriented, and multi-paradigm languages. Lambda calculus, fixpoints, type systems, and predicate logic. Denotational semantics and models of concurrency.

6530 Database Systems (3)

Representing information about real world enterprises using important data models including the entity-relationship, relational and object-oriented approaches. Database design criteria, including normalization and integrity constraints. Implementation techniques using commercial database management system software. Selected advanced Topics such as distributed, temporal, active, and multi-media databases.

6540 Human/Computer Interaction (3)

Fundamentals of input/output devices, user interfaces, and human factors in the context of designing interactive applications.

6610 Advanced Computer Graphics I (3)

Interactive 3D computer graphics, polygonal representations of 3-D objects. Interactive lighting models. Introduction to interactive texture mapping, shadow generation, image-based techniques such as stencils, hidden-line removal, and silhouette edges. Introduction to image-based rendering, global illumination, and volume rendering.

6620 Advanced Computer Graphics II (3)

Introduction to ray-tracing. Intersection methods for 3-D objects, reflection and refraction. Introduction to surface and solid texturing. Introduction to continuous-tone pictures and the aliasing problem. Special effects such as soft shadows, depth-of-field, motion-blur, and indirect lighting.

6630 Scientific Visualization (3)

Introduction to the techniques and tools needed for the visual display of data. Students will explore many aspects of visualization, using a "from concepts to results" format. The course begins with an overview of the important issues involved in visualization, continues through an overview of graphics tools relating to visualization, and ends with instruction in the utilization and customization of a variety of scientific visualization software packages.

6650 Image Synthesis (3).

Using camera and sensor simulation along with physical simulation to generate realistic synthetic images.

6670 Computer-Aided Geometric Design I (3)

6680 Computer-Aided Geometric Design II (3)

Introduction to current concepts and issues in CAGD systems with emphasis on free- form surface design; mathematics of free-form curve and surface representations, including Coons patches, Bezier method, B-splines, triangular interpolants, and their geometric consequences; classical surface geometry; local and global design tradeoffs and explicit and parametric tradeoffs; subdivision and refinement as techniques in modeling; current production capabilities compared to advanced research. Laboratory experiments with current CAD systems.

6710 Advanced Integrated Circuit Design I (3)

Introduction to basic concepts of the design of CMOS integrated circuits for students with a wide range of backgrounds. Static and dynamic properties of CMOS circuits, composite layout of CMOS circuits, and modeling of transistors for use in SPICE simulations. Commonly encountered CMOS circuits. Introduction to CMOS analog/digital circuits. Students complete design, composite layout, and digitization of a simple integrated circuit using computer-aided design tools.

6720 Advanced Integrated Circuit Design II (3)

Design of mixed signal (analog/digital) CMOS integrated circuits. Fundamental building blocks for analog circuits, including the basic principles of opamp, current mirror and comparator design. Basics of discrete-time signals and filters. Implementation of switched capacitor circuits and discussions of various implementations of D/A and A/D converters, oversampled converters and phase locked loops.

6740 Computer-Aided Design of Digital Circuits (3)

Introduction to theory and algorithms used for computer-aided synthesis of digital integrated circuits. Topics include algorithms and representations for Boolean optimization, hardware modeling, combination logic optimization, sequential logic optimization and technology mapping.

6750 Synthesis and Verification of Asynchronous VLSI Systems (3)

Introduction to systematic methods for the design of asynchronous VLSI systems from high-level specifications to efficient, reliable circuit implementations. Topics include specification, controller synthesis, optimization using timing information, technology mapping, data path design, and verification.

6770 Advanced Digital VLSI Systems Design (3)

Full custom, high speed, high performance CMOS circuit design issues, methodologies, and techniques. Failure modes, modeling techniques, testing, clock skew analysis, clock distribution, power analysis, power line distribution, electrical rules checking, megacell design flow, and other important design issues.

6810 Advanced Computer Architecture (3)

Principles of modern high performance computer and micro architecture: static vs. dynamic issues, pipelining, control and data hazards, branch prediction and correlation, cache structure and policies, cost-performance and physical complexity analyses.

6820 Parallel Computer Architecture (3)

Architecture, design, and analysis of parallel computer systems: vector processing, data vs. control concurrency, shared memory, message passing, communication fabrics, case studies of current high performance parallel systems.

6830 VLSI Architecture (3)

Project-based study of a variety of Topics related to VLSI systems. Use of field programmable gate arrays to design, implement, and test a VLSI project.

6930-6944 Seminar (1-3)

Current Topics in Computer Science. May be repeated for credit.

6950 Independent Study (1-4)

6960-6969 Special Topics (1-4)

6970 Masters Thesis Research (1-12)

6980 Faculty Consultation Masters (1-12)

7120 Information-Based Complexity (3)

Analysis of optimal computational methods for continuous problems. Introduction to the general worst case theory of optimal algorithms, linear problems, and spline algorithms as well as selected nonlinear problems. Examples include optimal integration, approximation, nonlinear zero finding, and fixed points.

7240 Sinc Methods (3)

Sinc methods for solving difficult computational problems, such as partial differential and integral equation problems, that arise in science and engineering research. Emphasis on parallel computation. Applications vary, depending on participants in the class. Students are given projects--whenever possible in their areas of research--that lead to publishable research articles.

7310 Advanced Robotics (3)

Covers the kinematics, dynamics, and control of robotic manipulators. Projects controlling robots will be an integral part of the course.

7460 Advanced Operating Systems (3)

Practical distributed operating systems concepts from basics through the state of the art. Topics include interprocess communication, client-server systems, distributed shared memory, distributed file systems, distributed databases, portable computing, software fault tolerance, and wide-area (e.g. web) applications. Work includes individual oral presentations, a group project, and a written research report.

7940 Seminar (1-3)

May be repeated for credit.

7950 Independent Study (1-4)

7960 Special Topics (1-4)

7970 PhD Dissertation Research (1-12)

7980 Faculty Consultation PhD (1-12)

7990 Continuing Registration: PhD (0)

Appendix B. Faculty participants

All School of Computing Long-Term Instructional (LTI) faculty may participate in the Computing degree program, as Track Faculty Committee members, and as department members for issues requiring full faculty oversight.

Faculty from other departments may participate as Track Faculty Committee members, but must be long-term instructional faculty (i.e., regular faculty or designated long-term instructional).

The tables below list all LTI faculty in the School of Computing, followed by those faculty in the Scientific Computing and Imaging Institute and other associated programs who may choose to participate in the tracks within the Computing programs.

Name	Type	Rank	Research Area
School of Computing Assistant Professors			
Rajeev Balasubramonian	Regular	Assistant	Computer architecture: clustered processors, memory hierarchy bottlenecks, instruction-level parallelism, power-efficient processors
Matthew Flatt	Regular	Assistant	Programming languages and systems
Sneha Kasera	Regular	Assistant	Computer networks and systems – mobile systems and wireless networks, network security
Michael Kirby	Regular	Assistant	Scientific computing and visualization
Emil Praun	Regular	Assistant	Computer Graphics
John Regehr			Embedded systems,consumer real-time, design and implementation of operating systems
Konrad Slind	Regular	Assistant	Formal Verification, higher order logic, functional programming
Cynthia Thompson	Regular	Assistant	Machine learning, natural language processing, and artificial intelligence
School of Computing Associate Professors			
Erik Brunvand	Regular	Associate	Computer architecture and VLSI systems
John Carter	Regular	Associate	Operating systems parallel and distributed computing, and multiprocessor computer architecture
Wilson Hsieh	Regular	Associate	Compilers, programming languages, systems, and architecture
Charles Hansen	Regular	Associate	Visualization, computer graphics, parallel computation, computer vision
Ellen Riloff	Regular	Associate	Natural language processing, information retrieval, and artificial intelligence
Peter Shirley	Regular	Associate	Computer graphics, visualization, visual perception, software methodologies for graphics
Claudio Silva	Regular	Associate	Scientific visualization, computer graphics, computational geometry

Ross Whitaker	Regular	Associate	Computer vision, visualization, and image processing
School of Computing Full Professors			
Martin Berzins	Regular	Professor	Adaptive Numerical Methods and software, Parallel Algorithms, Computational Fluid and Solid Mechanics Applications.
Elaine Cohen	Regular	Professor	Computer graphics, scientific visualization, geometric modeling, and mechanical design
Al Davis	Regular	Professor	Parallel computer architecture asynchronous circuits and systems, high performance multiprocessor communications
Ganesh Gopalakrishnan	Regular	Professor	Formal verification, asynchronous circuits and systems
Tom Henderson	Regular	Professor	Computer vision, mobile robotics
Lee Hollaar	Regular	Professor	Digital intellectual property law
John Hollerbach	Regular	Professor	Robotics, teleoperation, virtual reality, and human motor control
Chris Johnson	Regular	Distinguished Professor	Scientific computing, visualization, imaging, and problem solving environments
Robert Kessler	Regular	Professor	Systems software and software engineering
Gary Lindstrom	Regular	Professor	Programming language design, specification and implementation
Rich Riesenfeld	Regular	Professor	Computer graphics, geometric modeling, design
Kris Sikorski	Regular	Professor	Parallel scientific computation and computational complexity
Frank Stenger	Regular	Professor	Numerical analysis
William Thompson	Regular	Professor	Computer vision, visual perception
School of Computing Designated Long-Term Instructional Faculty			
Art Lee	Clinical	Associate	Aspect-oriented programming, scientific data management, distributed object systems
David Hanscom	Clinical	Professor	Computer architecture and data communications
Joe Zachary	Clinical	Professor	Application of computers to education
Steven Parker	Research	Assistant	Parallel component architectures, scientific visualization, and computer graphics
Sam Drake	Research	Associate	Integrated process planning and computer aided manufacturing, design, industrial robotics

Jay Lepreau	Research	Associate	Operating systems, components and languages, networks, security
	Faculty from other academic units		
Sarah Creem-Regehr	Psychology	Assistant	Cognitive and neural mechanisms underlying: object and space perception, perception-action dissociations and interactions, visual and motor imagery, spatial cognition
Stephen C. Jacobsen	Mechanical Engineering	Professor	Robotics, teleoperation, virtual reality, human motor control
Rob S. MacLeod	Bioengineering	Associate	Computational Electrophysiology
Chris Myers	Electical and Computer Engineering	Associate	Digital VLSI systems, computer architectures
Jeff Weiss	Bioengineering	Associate	Experimental and Computational Biomechanics, with applications in Orthopaedics and Cardiovascular Mechanics

Appendix C. CRA Article on Computing

Appendix D. Letters of Support