Notes from STUDENT RECORDS:

1. Robust class wait list feature.

2. An advising notes module or capacity. This would be an area specific to the record keeping needs of academic advisors; access would be limited to these users. The module would be used to keep track of individual advising records on a per-student basis. The ability to tie a note to one of a series of preidentified flags (for example, “sent e-mail student regarding upcoming enrollment period”; “student requested appointment to discuss major selection”; “student requested signature for add/drop”, etc). A robust deployment would allow for the automatic generation of e-mails or letters to a student selection identified by the advisor (using the reporting structure mentioned earlier) and the posting of a flag to the notes module indicating date e-mail was sent and general content of the message. Advisors could also record free-form notes.

3. Automated calendaring system. This system would rely on both the capacities of SIS and the portal. Students would have the opportunity to sign up for an advising appointment with an individual counselor and later receive an e-mail reminder of the appointment date. Students would also have the option of selecting from a preidentified list of advising needs (discussion of possible major change, discussion of graduation requirements, discussion of career opportunities) and have their appointment directed to an advisor in the appropriate area (college office, major department, Career Services, etc.). Advising periods from the preidentified list would all have established time lengths. Appointments would be reflected on a full-featured calendar used by academic advisors for all other calendaring needs. In a robust implementation, the calendar and note taking system should be tied together in what can be turned an “advisor’s toolkit.”

4. Ability for students to self submit and track the progress of a number of common petitions, such as petition to major change, application for graduation, application for readmission, etc.)

5. Extended integration with other campus systems (Housing, Alumni, etc.).

6. Expanded online transfer evaluation tools for students, including continuing and prospective students. This includes a robust articulation database; a mechanism for prospective students from multiple community colleges to review the applicability of their existing work against UCR admission requirements or against specific degree requirements; online reporting tools for students; etc.

7. The desire for a web-based interface to the system for advisors and records staff and a more intuitive interface for occasional users (e.g., faculty) and expanded help information.
8. *Degree audit tool with user-friendly interface*, the ability to provide “drill-down” reports that focus on a particular portion of the degree (breadth requirements, major requirements, etc). Ability for students to do course planning online and see instant feedback on their plans.

9. Easier *navigation and mining of information* from the financial support and student financials modules, particularly for graduate advisors.

10. A system *that accommodates multiple types of input* (i.e., not limited to keyboard data entry).

11. Extended *audit functions* in some areas, not so much to track the actions a student has taken but the actions of administrative users.

12. Improved handling of *special program admissions*.

13. An *automated mechanism for coding repeated units*.

14. Availability of live *class availability information* for students.

15. Improved *dynamic updating* in a few areas, particularly in the Scheduling screens and the interface between Scheduling and Catalog information.

16. Improved handling of *cross-listed courses*.

17. *Less reliance on batch processes*. 
Notes from STUDENT BUSINESS SERVICES:

1. *Real time, online availability should be as near to 100% as is possible* for 7am – 6pm, Monday thru Friday. Extended hours and weekends must also be available as needed.

2. Billing/Receivable (B/R) module must be *capable of tracking the billing and collecting of student as well as non-student accounts* (Sundry Debtor, Contract Billing).

3. For reconciliation and auditing purposes *credits must be traceable back to the debit transaction(s) and vice versa.*

4. User *access restricted by screen and field as well as inquiry and update. Simple, yet secure process for new user to gain access to system* and existing user to modify their current access.

5. *Online read-only access to batch processing schedule.* View job stream as well as the parameters associated with job.

6. *Web capacity for secure interactive screens* (PAWS, GROWL, Credit Cards, Automated Clearing House)

7. The ability *to properly handle future dated transactions.* Accept them as such and successfully enact upon them with appropriate timing.

8. *SIS screens must be user friendly, provide for easy user navigation, rapid review of data, and rapid key entry of data. SIS screens must update relational database within reasonable time frames* to allow for rapid processing of data and in-person and over-phone student support.

9. Continue *accurate application of student holds.* Comment tracking for phone calls, in person Q & A and general pertinent information.

10. Keep or replicate *Financial Aid’s simulation “Award Disbursement” Screen (324), and SBS/Cashiers “Tuition Calculation” Screen (407).*

11. *The application of credits within a customer’s account must incorporate a hierarchical and/or specific strategy as well as carry forward the offsetting Full Accounting Unit in a manner that is simple, expedient and efficient.*
Notes from FINANCIAL AID:

1. Support **automatic data interchange between UCR’s system and those of external data systems** such as the State of California (California Student Aid Commission), U.S. Department of Education (DOE), Internal Revenue Service (IRS), NCAA software, and other agencies as needed.

2. Enhance **data interchange between campus systems**. Must easily interface in order to automate the process of information exchange between the Personnel Payroll System, General Ledger (PeopleSoft), UCRFSlite, SUPERDOPE, Billings Receivable System, STARRS, PAWS, and departmental databases as necessary.

3. Must perform **standard federal student aid processing**. The system must be flexible enough to quickly respond to the ever-changing nature of financial aid. The system must:
   
   A. Automatically send, receive and log data records from all internal and external sources, ensuring that proper monitoring is available to ensure no data is lost. Ability to tie the upload of records to UCR student status.
   
   B. Automatically load and modify user specified student budgets, and perform all mandated need analysis in accordance with annual federal standards.
   
   C. Ability to monitor records and account for differences in student identifiers due to dissimilar record keys, (e.g., federal SSN vs. institutional ID) and correct inconsistencies. This function must be present to monitor possible differences in internal UCR databases (currently SDUP process).
   
   D. Provide audit trail containing federal, state and institutionally specified data for the necessary retention periods, including counselor comments. Provide for automatic purging of financial aid records and performance of periodic system maintenance (e.g., FAM has reached its segment limit in some areas).
   
   E. Provide automated notification of students’ admission and enrollment, and changes to student status (SIR status, grades, last date of attendance, withdrawal/dismissal actions for computation of federal refunds, and on-line calculation of federal Return to Aid, etc.).

4. Provide **an official third week file for Financial Aid data**. As of now there is no ability to compare third week data historically. This would help to improve the consistency between financial aid data reported to requesting sources by the Financial Aid Office and data of the same type requested/reported by other campus offices. This is currently an ongoing issue in the provision/reconciliation of data reported to the UCOP Corporate Student System.

5. Provide product with a **user friendly interface easily managed by desktop users**, which will empower them to use integrated data from all available sources: STARRS, PAWS, GROWL, web applications, mainframe data, etc.

6. **Perform automated aid packaging, re-packaging, and packaging simulations to optimize the utilization of funds**, incorporating fund restrictions (e.g., federal loan limits). Must be able to track/process aid revisions, cancellations, rejects and re-packaging based on user specified
triggers (e.g., enrollment changes.) The system should be able to monitor the effect of these changes on other departments/system modules, modify data, and notify affected parties as necessary.

7. **Maintain distinct academic year and summer session components**, especially in the aid packaging module. Have a methodology in place to coordinate the timing of data inputs which cross processing years to ensure ‘apples to apples’ data integrity (reduce the negative effect of timing differences in data interpretation).

8. **Track document requests, and receipt of documents in STARRS**. Create document files to easily produce printed tracking letters, award letters, and e-mail correspondence, based on specified system data characteristics for groups of students. Ability to track this output and print duplicates.

9. **Redesign of FAM’s data presentation paradigm to facilitate the grouping of data to mirror the aid processing workflow**. FAM currently has over 40 screens, which were designed at a time when financial aid regulations and procedures were significantly different. Inherent in this redesign is a need for a simple, yet secure, system for the assignment and maintenance of individual system accesses, which compliments the new design.

10. **Allow for timely and efficient disbursement of student financial aid**. The disbursement process must be automated in all areas, adhering to regulations as they pertain to periods of award, allowable disbursement dates, holding of aid payments, eligibility of students to receive aid, etc. This will again require the effective integration of all system modules.

11. **Track system fund expenditures and counts of students awarded by fund, real-time, including automatic system updates on actual payments made from the BRS and/or PeopleSoft system.** Currently the data flow of IA PLUS has no provision for the updating of financial data from other campus systems (exception: PPS updates work-study paid to dates.) Due to system incompatibilities, certain critical data we require is not available in the General Ledger, such as student level payment detail. As we are a feeder system, we only have fund level activity. Financial Aid is required to reconcile to the student as well as the fund level. Due to the sheer number of accounts and transactions involved, a tremendous amount of effort is needed to reconcile ‘across’ FAM, BRS and the General Ledger. This could be greatly reduced by automating the processes between the systems to perform fund monitoring and reconciliation.

12. **Provide a method to monitor UC inter-location activities such as multi-campus enrollment, and facilitate the inter-location transfer of funds and expenses.**
Notes from GRADUATE DIVISION:

1. Improved data dictionary.

2. **Improved tracking of student data from the graduate student perspective** (and improved retrieval) to include the following:
   
   A. Number of enrolled units after third week.
   B. Track departmental committees.
   C. Financial support awarded a student.
   D. Identification of course repeats automatically upon registration
   E. Number of quarters a student has been a TA.
   G. Degrees and dissertation titles.
   H. Identify the number of quarters a student has been enrolled and within that count students on official part time status as 1/2 of a quarter.
   I. Identify part-time status, filing fee, and leave of absence.
   J. Ability to count GPA for those with double majors or those who begin a second major; for example, a student received a degree in Geology and then began a degree in Computer Science (GPA of the second degree be calculated automatically instead of by hand each quarter).
   K. Ability to track English Department Lecturers.
   L. Ability to better isolate grads versus undergrads in Financial Aid.
   M. Ability to name fields, change fields (on screens and web pages).
   N. Availability of “flexible” fields that can be assigned at user discretion.
   O. Improved ability to audit and track changes to data elements (including operator ID of individual making the change).
   P. Want to be able to do timetable tracking.
   Q. Ability to understand student lapses (why is it happening?)
   R. What do students owe, why, and why they have a hold; which students have not completely paid and how much they owe.

3. Improved processing of refunds for withdrawals.

4. Improved processing of residency changes.

5. Improved processing of students who repeat courses (it should be noted on the transcript automatically).

6. Improved processing of students who want to enroll in a course on an S/NC basis (it should only be allowed if the course is offered that way).

7. Improved processing of “backdating” data.

8. Improved processing (and flexibility in data field sizes) relating to identifying lenders.

9. Improved processing of “leaves” after the quarter begins.

10. Improved on-line transcripts (easier to read on one screen).
11. Ability to make batch changes *(multiple changes at one time)*.

12. Ability to e-mail administrators when changes occur in the database *(changes affecting a fellowship)*.

13. Improved fellowship processing *(projecting a student’s fellowship package for six years and having the details appear on the student’s bill)*.

14. Improved “On-Course” capabilities making the module acceptable for graduate students.
Notes from ADMISSIONS:

1. Improved information retrieval and the ability to process / view multiple screens/students at one time.

2. A “user friendly” system that does the following:
   
   A. Does not rely predominately on memorizing screen numbers, codes, etc.
   
   B. Provides more “help” options such as pop-up identifiers, drop-down lists, links, right click, etc.
   
   C. Provides a “Home Page” for each student from which you can retrieve all information captured for that student (drop-down list, links, etc.).

3. Ability to run reports easily from personal computers and have them print on our local printer. (Noted by Graduate Division and to include printing of UCR transcripts.)

4. Improved application management (especially multiple application management). Robust transaction history (effective dated records, etc.) so that each application and all information pertaining to each application remain “current” but auditable over time (and that ability to track any and all changes is facilitated).

5. Improved data management and the ability to track changes to various key data elements (i.e., changing school codes, deleting obsolete school codes, changing school names, adding and deleting majors, etc.).

6. Student access to admissions data. UCR is in the process of developing an Admissions Status Web Site to provide this access. The new SIS should provide live, current information to this site and allow students to make certain changes that will directly change SIS information (i.e. address, email, and certain major changes).

7. Improved electronic transcript processing. The ability to receive transcripts electronically, directly from the institution, and have the information feed directly into SIS.
Notes from INSTITUTIONAL PLANNING:

Data Extraction Capabilities:

1. **Enrollment.** A method to easily identify (and extract) each student’s “initial” enrollment information. By “initial” means “first”. For example: the ability to gather (for each student’s first enrollment as an Undergraduate – and/or Graduate – and/or Credential) his/her First Year/Quarter enrolled – First Degree Program – First Major(s) – First California Residency Status, First US Citizenship or international Visa Status, etc. In theory, then, each student could have a maximum of three records in this file (one for each of his/her first undergrad, grad, and/or credential enrollments). This file could also contain some demographic information such as the student’s name, gender, date of birth, SSN and ethnicity.

   The next requirement is a way to easily identify, retrieve, and extract each student’s academic term information. For each term in which the student is enrolled, I need the individual courses taken, the grades received, an indication of the student’s units attempted, completed, graded, passed/not passed for that term – with year-to-date and total-to-date totals as of the current term – as well as the determination of what units were “quality units”. (The current information displayed on the Transcript Screen (136) contains the requisite information). In addition, each “quarter year” record needs to contain the student’s current name, citizenship/visa and California Residency status, degree program, major – and “student level” (freshman, sophomore, junior, senior, limited, credential, master, doctoral 1, doctoral 2, or doctoral advanced to candidacy more than 9 quarters).

2. **Admissions.** Access to (and ability to retrieve and extract) certain data related to the student’s admission (as either an undergraduate, graduate, or credential student) is also required. Specific data here include the student’s School of Origin (i.e., the school most responsible for his/her admission to the University), Last School Attended, High School GPA (for undergraduate freshmen), Community College or Transfer College GPA (for advanced standing transfers), Degrees conferred (for graduate and credential students) as well as degree-awarding institution. For undergraduates, their centrally assigned ETS number as well as their AP units (and an indication as to whether those units have been applied towards their degree requirement.

   Also needed are the students’ test scores (GRE, GMAT, SAT, SATII, AP, etc.) including type of test, date taken, etc. (see screen #221)

3. **Degrees.** A record for each degree the student has been awarded at UCR is necessary – including the student’s major in which the degree was conferred, the type of degree (BA, BS, PhD, etc.) and date (and year/quarter) in which the degree was conferred.

4. **Financial Support.** The information provided on the current CICS SIS-IA+ screen # 319 contains most of the information necessary for my work with student financial support. Here each financial aid award – uniquely identified by the Corporate Student System award code, the internal Financial Aid office Fund code, fund description, fund abbreviation, as well as the dollar amounts offered – awarded – accepted -- and actually paid to the student.

   Additionally, each award should be tied in to a valid FAU (Account/fund number) in the UCR Financial System.
Since Financial Aid information is really not “finalized” until the end of the academic year, there is no need for “quarterly” versions of the file – however, an agreed upon “version” should be identified as a year-end FINAL version – and this can be used for official reporting.

Much work with student financial support revolves around graduate student support and a major facet of these analyses is the examination of the source of funds for the student. In the specific area of student benefits from TA and GSR earnings (GSHIP, and Fee and Non-Resident Tuition Remissions) it would be most helpful if these various dollar amounts could tie in to the specific “funds” which were used for the student’s salary as a TA or GSR. (Currently only one fund number identifies these awards – making it impossible through SIS to determine whether the funds are campus-based or from extramural sources.)

**Corporate Student System Input Files to UCOP**

1. **Official Statistics.** There are two primary offices which maintain UC Riverside’s “official” statistics – for students, faculty, and staff. On campus, it is the Academic Planning and Budget office. The other “official” residence of the UCR data is the Information Management unit of Information Resources and Communications at the Office of the President. UCOP obtains the detailed UC Riverside information based upon data files which are sent periodically throughout the year.

   Currently, there exists in the SIS-IA+ file no direct way to “map” or “track” just what SIS-IA+ data element is the corresponding match to the elements sent on the Corporate files. (Specifically the Corporate Student System files).

   The new system should provide this cross-walk (and corresponding definitions) so that statistics and published reports which come out of UCOP can be verified and replicated by Academic Planning and Budget. And, the system should be designed to provide this facility easily and without ambiguity.

   The UCOP files this currently includes would be the Third Week and End of Term Enrollment Files, the monthly Undergraduate Admission Files, the quarterly Student Financial Support Files, and the annual Graduate Admission File, Postdoctoral Scholar File, and Degrees Conferred File. Additionally, there have recently been specifications issued for files on “summer” enrollments and financial aid. These, too, should be included in the list above.

**General Comments**

1. **Student ID:** This number should be assigned to each student at the time of his/her Application for Admission == and SHOULD NEVER CHANGE. E.g., if a student applied for admission as an undergraduate in Fall 1997 – and was not admitted – and later applied for admission in Fall 2005 – THE SAME STUDENT ID NUMBER should be assigned to that applicant.

   Additionally, once the student actually enrolls – the number would remain his Student ID number for life.

2. **Interactive Screens:** Similar to the current CICS modules, a quick way to obtain an individual student’s record “on-line” should be made a part of the new SIS system.
3. **Environment:** A facility should be provided for easy cumulative reporting on student statistics – as well as retrieval and extract of “raw” data. For example, a reporting tool which one could use to “count students by graduate and undergraduate level – by major – for a given academic term” – either in aggregate or by specific student. It is essential to be able to obtain individual student records and to be able to save them in a flat ascii format – for eventual incorporation into Excel or Access or SAS.

4. **Accountability and Locking:** The system should have some way of indicating that – at some point – a given year/quarter’s information is Official and is Locked. E.g., if, say, in January of 2004 someone looked at the Winter 2003 quarter’s enrollment file and it showed 1,365 Juniors majoring in Business === the same query made in February of 2015 should show 1,365 Juniors majoring in Business (and they should be the SAME 1,365 people from the 2004 run!) In other words, retroactive adjustments to records should be strictly prohibited – or at least strictly controlled if prohibition is impossible.

5. **Training:** Sufficient training should be provided for users of the new system – and on-going support for users who require access to it.
Notes from Academic Planning and Budget:

1. **Effective Dating of data / event tracking.** RT245 currently has two purposes – (a) a 15\textsuperscript{th}-day flag (the student is to be counted or not counted in CSS reports to UCOP) and (b) a “paid” flag – the student has paid their fees. If a student registers late (after the 15\textsuperscript{th} day), the student is not in the “official census” and is not included in the file to UCOP, but the flag is changed to “Y” after the student pays fees. In other words, the flag did not stay frozen. This means that it is not possible later in time to go back and do an analysis based on the official “frozen” number of students. In the new system, there should be two separate data elements.

2. **Improve Robustness of Data Elements for Reporting and Tracking.** RT020, current registered hours, includes workload units in the non-credit “remedial” courses Math-003, Engl-004D, Engl-005D, and BSWT-001, BSWT-003, and BSWT-003D. However, RT025, current hours earned, excludes these workload units because the courses are not for degree credit. This apples-and-oranges situation makes it difficult to do an analysis comparing units attempted to units completed. The solution is to make RT020 into two data elements. The pair of data elements could be (a) current registered hours including remedial units, and (b) current registered hours excluding remedial units. Alternatively, the pair could be (a) current registered hours excluding remedial units, and (b) current remedial units.

3. **Improve Robustness of Data Elements for Reporting and Tracking.** On the term file, RT040 is the current term GPA, and RT025 is the current term hours earned. On the cumulative file, RA380 is the cumulative GPA, and RA320 is the cumulative hours earned. We would like to have additional data elements on the term file which capture the cumulative GPA and cumulative units as of the end of that particular term. For example, there is no way to go back and extract what students’ cumulative GPA and cumulative units completed were as of the end of a past quarter (other than visually examining screen 136 and manually copying the data for every student). To summarize, on the term file now there is a set of “CURR” data elements RT020, RT025, RA380, RA320, and RT040; there needs to be another set on the term file for end-of-term “CUM” data elements.

4. **Improve Robustness of Data Elements for Reporting and Tracking.** On the degree table, there is a data element for degree date (RA245) but not a data element for degree term. One has to go to another table to “translate” the date into a term, which is inconvenient. (Perhaps the term could be on the degree table, and the date in the translation table?)

5. **Improve Robustness of Data Elements for Reporting and Tracking.** On the term file there is a flag RT240 to indicate whether students are in the University Honors Program that quarter. There are actually two sub-programs – lower division honors and upper division honors. Being in LD honors doesn’t automatically mean a student will also participate in UD honors. So, having two separate flags would be helpful, so one would know which of the two programs the student was currently participating in. Data fields are also needed showing whether the student actually completed LD Honors and/or completed UD honors. (Right now, the only way to see if students completed the requirements for the honors program(s) is to look at screen 136 or extract the transcript text from the RQFILE, which is awkward.) An alternative would be to have, instead of a separate set of “completion” flags, an additional value for the “participation” flags – such as N, Y (participating), C (completed). The University Honors Program folks might have a preference as to the approach.
6. **Improve Flexibility in “Validation Table” Values for Reporting and Tracking.** RT105 –
college code on the term file. For undergraduates, there is no problem – it shows as HS, NA, or
EN. But for graduate students it always shows as GD (Graduate Division), so this is not terribly
useful. It would be much more useful if it showed the actual school/college the graduate student
is in, e.g., HS, NA, EN, ED (Education), MG (Management), or MD (Medicine), for example.
Otherwise the user has to match the major (RT120) to a college-major table to obtain the school
or college.

7. **Improve Robustness of Data Elements for Reporting and Tracking.** On the MMFILE, the
Admit-to College (MM456) and Admit-to Major (MM460) data elements do double duty as both
“admit to” (i.e., admitted) and “accepted to” (i.e., accepted offer of admission) fields.
This is unfortunate because if a student does NOT accept the offer of admission, it is my
understanding that the Admit-to data elements are then blanked out. In other words, the data are
lost. Then it is not possible to answer the question, “how many students were offered admission
to the College of X”. To solve this dilemma, I suggest having separate sets of data elements for
“admit to” and “accept offer to”, so that the “admit to” data will not be lost.

8. **Integration of SIS to PPS and other External Systems.** RC305 – course instructor’s SSN. An
additional data element for the instructor’s non-SSN Employee ID will be needed. It would be
helpful if there was a way to link this data element to a payroll table under the new system to
help avoid the data-entry typos and transposed digits that are sometimes entered. This would
reduce the need for data-cleaning. (Currently, an extract from payroll is made to match by SSN
to a course file extract; the main purpose of this is to be able to know the “pay department of
instruction” for each course and the instructor’s title code. But sometimes the SSNs don’t match
because of data entry errors in RC305.)

9. **Integration of SIS to PPS and other External Systems.** RC255 (Meeting Building) and RC260
(Meeting Room). I would strongly recommend for the new SIS, that the official building and
room codes from the Facilities Data Base (an Oracle-based system developed by Computing &
Communications for Capital Planning) be used instead of the SIS 5-char building code. Having
two parallel systems of building and room codes results in many errors in coding and the need
for a lot of data cleaning. They don’t always neatly map one-to-one. If the new system could tie
in to the Facilities Data System, then data entry persons could perhaps benefit from drop-down
boxes for selecting the building and room in which a course is offered.

10. **Improve Cross Listing Process and Data Management / Retrieval.** Under the current SIS,
cross-listed classes are treated somewhat as if they were separate physical classes. For example,
BSAD-116 and PHIL-116 are really one physical class, where students have the option of
signing up under either BSAD or PHIL. However, they appear in RCFILE as separate records
with different RC010 course section ID. There are flags which say what is cross-listed with
what, but no easy way to consider the two as combined into one. For example, if someone
wanted to know the average number of classes taught by regular rank faculty, they would either
have to write a complicated program or else aggregate by hand, to avoid double-counting classes.
Or if one wanted to list, in alpha order by instructor, every course taught by each instructor, the
list would show cross-listed course as if they were two “courses”. (The same problem occurs
with several “concurrent” courses such as HIST-020-001 and HIST-020H-001, which is one
lecture attended by regular and honors students.) It might be useful if there was an additional
table which was a “parent” to the RCFILE “child” table. In the parent table, each record would
represent a unique physical class, and in the child table, classes and cross-listed classes would appear the way they do now.

In Academic Planning & Budget we have developed a Course Information Reporting System (CIRS), which is kind of a “shadow system” created so that we have course data in a form useful for analysis. It starts each quarter with an extract from SIS. It has the parent-child setup mentioned above for the cross-listed; discussions and labs “tied to” lectures are split off so that the secondary sections appear as individual records; missing building/room/day/time and missing instructor/T.A. data are cleaned up insofar as possible; it includes instructor title codes and instructor departments from payroll; and has some other features such as many calculated fields. I suppose in theory a new SIS could preclude the need for a separate CIRS, but that’s probably unrealistic.
Notes from COMPUTING AND COMMUNICATIONS:

1. The new SIS must have as its foundation a modern relational database. As of 7/17/2002, enterprise databases utilized at UCR include Oracle and DB2. (Noted by Admissions. Noted by Student Business Services and must include imaging capabilities.)

2. The new SIS must deliver a robust, easy-to-use SQL query tool (e.g. facilitates SQL joins) integrated natively into the application. A third party query tool (BRIOS query) is acceptable, if it is tightly integrated into the SIS application. (Noted as top Student Records Request. Noted REPEATEDLY by Admissions as a major requirement. Noted by Student Business Services. Noted by Graduate Division. Noted by Financial Aid.)

3. The new SIS must deliver a robust reporting engine that is separate from the transaction processing system (facilitated via database replication, de-normalized database, etc.). This reporting system must be completely web enabled and tightly integrated with the SIS portal (see below). A third party web reporting tool (BRIOS) is acceptable, if it is tightly integrated into the SIS application.

4. The new SIS must tightly integrate with a vendor supplied portal offering. At a minimum, the portal must provide students with the following:
   
   A. A student “customizable” browser interface.
   
   B. Seamless (e.g. single sign-on) access to student self service functionality (e.g. registration, degree audits, financial aid application, etc.).
   
   C. Seamless (e.g. single sign-on) access to “read only” student data (grades, financial statements, enrolled classes, etc.).
   
   D. Seamless (e.g. single sign-on, correct adds / drops processing) access to the campus learning management system (Blackboard).
   
   E. Seamless (e.g. single sign-on) access to other student electronic services (campus web mail system, calendaring, etc.).
   
   F. Faculty and staff (especially staff involved with advising) must have portal access and access to specific areas of functionality (e.g. a faculty gradebook).

5. The new SIS must, however, allow for the development of a UCR supplied portal. At a minimum, the system must provide XML access to important code paths supporting core SIS processes (e.g. registration).

6. The new SIS must provide XML interfaces to key code paths that will facilitate seamless UCR SIS integration with other institutions of higher education (especially in California) as well as UCOP (e.g. Community Colleges for transfer students, other UCs for Summer Session, etc.).

7. The new SIS must provide robust data warehouses, OLAP (On-line Analytical Processing) tools, and DSSs (Decision Support Systems). However, UCR may decide to develop these systems internally or in collaboration with the vendor.
8. The new SIS must provide sufficient tables, code paths, and functionality to allow for automated "event driven" creation of e-mail sent to targeted "groups" or "classifications" of students. (Noted by Student Business Services. Noted by Graduate Division.)

9. The new SIS must provide sufficient code, tables, and functionality to permit at least XX% of SIS transactions to be processed on a completely "self service" basis.

10. The new SIS must provide sufficient flexibility (in terms of database design, development tools, XML interfaces, etc.) that will allow UCR to relatively quickly develop applications (screens, web pages, business rules) and application extensions aimed at minimizing campus SIS shadow systems. (Noted as second most important Student Records request. Noted as top Graduate Division request.)

11. The new SIS must natively (without "translations" or short cuts) support the complete campus Full Accounting Unit (FAU) for seamless integration with the campus General Ledger.

12. The new SIS must seamlessly integrate with the campus Payroll System so that Student Ids are identical in the campus Payroll System and the campus SIS. (Noted by Student Business Services. Noted by Graduate Division.)

13. The new SIS database must be constructed in such a fashion as to provide a scaleable solution (given distributed, self-service access for 20,000+ students, faculty, staff, and potential students considering attending UCR).