

**Project Title**

**Academic Personnel On-Line Review: A UC San Diego and UC Irvine Collaboration**

**Submitters**

<p><b>UC San Diego</b></p> <p>Viet Truong AP On-Line Team Leader Administrative Computing and Telecommunications (858) 534-9622 viet@ucsd.edu</p> <p>Dawn Reser Director, Academic Information Systems and Data Mgmt. Academic Personnel Services (858) 534-7960 dreser@ucsd.edu</p>	<p><b>UC Irvine</b></p> <p>Max Garrick Project Manager Office of Information Technology (949) 824-7917 max.garrick@uci.edu</p> <p>Joan Tenma Director Academic Personnel (949) 824-5104 jtenma@uci.edu</p>
--	--

**Project Leaders and Team Members**

<p><b>UC San Diego</b></p> <p><u>Administrative Computing and Telecommunications</u></p> <p><b>Executive Leadership:</b></p> <ul style="list-style-type: none"><li>• Emily Deere, Executive Director, IT Applications</li><li>• John Baker, Academic Applications Director</li></ul> <p><b>Project Leader:</b></p> <ul style="list-style-type: none"><li>• Viet Truong, AP On-Line (APOL) Team Leader</li></ul> <p><b>Team Members:</b></p> <ul style="list-style-type: none"><li>• Chris De Rosa, Middleware Architect</li><li>• David Soleno, Application Developer</li><li>• Gail Mikels, Application Developer</li><li>• Jennifer Kramer, Project Manager</li><li>• Lin Chen, Application Developer</li><li>• Matt Elder, Security Developer</li></ul> <p><u>Academic Personnel Services</u></p> <p><b>Executive Leadership:</b></p> <ul style="list-style-type: none"><li>• Kristina Larsen, Assistant Vice Chancellor, Academic Personnel</li></ul> <p><b>Project Leader</b></p> <ul style="list-style-type: none"><li>• Dawn Reser, Director, Academic Information Systems and Data Management</li></ul> <p><b>Team Members:</b></p> <ul style="list-style-type: none"><li>• Lynn Field Karsh, Director, Academic Advancement and Training</li><li>• KC Causey, Senior Administrative Analyst and AP On-Line Training Coordinator</li><li>• Diane Ingalls, Communications Coordinator</li></ul>	<p><b>UC Irvine</b></p> <p><u>Office of Information Technology</u></p> <p><b>Executive Leadership:</b></p> <ul style="list-style-type: none"><li>• Shohreh Bozorgmehri, Director, Academic and Network Applications</li><li>• Briandy Walden, Manager, Academic Web Technologies</li></ul> <p><b>Project Leader:</b></p> <ul style="list-style-type: none"><li>• Max Garrick, Project Manager</li></ul> <p><b>Team Members:</b></p> <ul style="list-style-type: none"><li>• Aldo Sarmiento, Application Developer</li><li>• David Pritikin, Quality Assurance</li><li>• Henry Balanza, Application Support</li><li>• Kim Gerrard, QA and User Support</li><li>• Ray Vadnais, User Experience Architect</li></ul> <p><u>Office of Academic Personnel</u></p> <p><b>Executive Leadership:</b></p> <ul style="list-style-type: none"><li>• Joan Tenma, Director, Academic Personnel</li></ul> <p><b>Project Leaders:</b></p> <ul style="list-style-type: none"><li>• Jean Chin, Technical Analyst</li><li>• Nancy Vongsawat, Senior Academic Personnel Analyst</li></ul> <p><b>Team Member:</b></p> <ul style="list-style-type: none"><li>• Ilona Pak, Senior Administrative Analyst</li></ul>
--	---

## **Summary**

Academic Personnel On-Line *Review* is the second application shared by UCSD and UCI in what is envisioned as a comprehensive, integrated business system that will ultimately allow all academic personnel actions to be completed electronically on both campuses. The first shared application was *Recruit*, a system developed by UCI for submission and review of applications for faculty positions. Development of *Review*, a system for electronically processing academic review files, was initiated by UCSD and will be completed in collaboration with UCI.

Development of single-campus applications based on borrowed code is not uncommon at the University of California, but AP On-Line, and *Review* specifically, represents something new: collaborative development of a single business system for two campuses under the direction of an intercampus management team, with underlying adoption of shared business practices. As a result of this project, UCSD and UCI have already saved years of development time and hundreds of thousands of dollars, a major benefit in the current budget climate. If collaborative development and shared use of AP On-Line can be extended to include other campuses, it will compound the value of the project to the University of California in years to come.

## **Project Description**

### **Introduction**

AP On-Line *Review* is part of a suite of services that also includes AP On-Line *Recruit* (a shared online job application and applicant review system) and *e-Recruitment Plan* (an application for submitting open recruitment plans that is currently used only at UCSD, but will soon be shared with UCI). More services will be added to AP On-Line, as outlined in the [AP On-Line Strategic Plan](#).

*Review* improves the speed and efficiency of the academic review process, makes the process more transparent to all participants, and reduces risk by ensuring compliance with policy and protecting reviewees' due process rights. Because the academic review process is so complex, *Review* is being developed in phases. Phase I, released at UCSD in July 2009, focused on processing actions for which deans have final authority (primarily normal merit advancements). Phase II, released at UCSD in August 2010, focused on processing of actions requiring review by the Committee on Academic Personnel and decisions by the Senior Vice Chancellor.

In the summer of 2009, UCSD and UCI signed a memorandum of understanding agreeing to share the *Recruit* and *Review* applications, and this was rapidly achieved. UCSD received a fully configured version of *Recruit* in October 2009, and UCI received a fully configured version of *Review* in September 2010.

This project description will focus on the shared use and collaborative development of *Review*, including the modifications of business practices necessary to make this possible. Information about the background and features of *Review* is available in the [Appendix](#).

### **The UC San Diego / UC Irvine Collaboration**

In 2009, we set out with a challenging and untested goal: UCSD was to host *Review* for UCI and UCI was to host *Recruit* for UCSD. In 2010 we achieved our goal at a fraction of the cost typical of building locally. We demonstrated that it is possible to launch new applications--built to UC specifications--in a time of severe budget cuts.

Three factors were essential to our success:

1. **An innovative application delivery model** called Software as a Service that has gained popularity in the commercial sector but was mostly untested within UC.
2. **A grassroots, outside-the-box collaboration** that put territoriality and recharge models aside in favor of jointly achieving an audacious goal.

3. **Full executive support** from both campuses' academic personnel and information technology units to change local AP practices and support IT efforts, ultimately providing the right environment to succeed.

### **Innovative Application Delivery Model: Software as a Service**

UC budgets have been slashed, and campuses are responding with no-hire policies, attrition, and layoffs. To offset resulting staff workload challenges, campus units are increasingly turning to IT to provide efficient solutions. IT's challenge is to effectively respond to these needs while facing its own severe cutbacks.

Software as a Service (SaaS) is a model for efficient application sharing across campus lines. Evolving from standard application provider/hosting models, SaaS was popularized by companies like Salesforce.com and is now used widely in the commercial sector.

The UC system has a wealth of online processes. At the local level, however, one campus may use an inefficient paper process while another campus is using a custom-built Web application. SaaS can bridge this disparity in online capabilities if the conditions are right, delivering lower-cost, best-of-breed applications to all UC campuses. SaaS can also efficiently scale from two to ten campuses.

SaaS thrives given the following conditions:

- **Minimal variation in business processes** across the customer base. Differences are usually limited to branding, authentication, and data, with no process or logic customizations. The *Review* Collaboration is committed to creating standard cross-campus AP business practices and creating divergent processes only when absolutely necessary.
- **A well-defined service level** in the form of a Service Level Agreement (SLA). All performance, support, and availability guarantees are clearly defined and agreed to by provider and customer. The UCSD/UCI *Review* [Service Level Agreement](#) documents our commitment to clear service level expectations.
- **Quality metrics that describe overall resource utilization**, allowing for accurate calculation of provider costs and customer charges. The UCSD/UCI Collaboration simplifies costing through a *Review/Recruit* application bartering agreement that assumes similar usage of both tools.
- **Many customers**, resulting in economies of scale that allow for further reinvestment in the product, delivering more value to customer campuses. We have enabled a multi-campus capability for both *Review* and *Recruit*; the next logical step is to leverage our investment by delivering comparatively low-cost service to other UC campuses (as compared to building an application locally to support only local campus needs).

SaaS delivers low-cost service because of the following principles:

- **One hosting campus, many customer campuses.** Resources are focused at the provider campus, allowing for more product enhancements, infrastructure upgrades, and process improvements that benefit all customers. Thus, *Review's* UCSD Administrative Computing and Telecommunications (ACT) development team provides enhancements for both UCSD and UCI.
- **One code base, many branded sites.** All enhancements are assumed to be shared among all sites. No code forking occurs, so campuses are prevented from independently making local changes that are difficult to share. This also eliminates overhead normally associated with maintaining custom-patched open/shared source code. UCSD and UCI *Review* sites share the same code base, with branding and other differences handled via configuration and campus-specific plug-ins where needed.
- **One operations team, distributed support and training.** The provider campus must create operational efficiencies or risk becoming overwhelmed with technical support, work requests, and product change requests from customer campuses. Quality self-service tools and intuitive user interfaces help customer campuses provide effective support and training for end-users. Although *Review* was launched for UCI

only eight months ago as of this writing, the *Review* team has already developed new tools and usability enhancements that improve operational efficiency for both campuses.

### **Grassroots, Outside-the-Box Collaboration**

The UCSD/UCI Collaboration was created to meet two campuses' basic need for more efficient online AP processes. No formal UC committee blessed the partnership, and no UC policies told us how to govern the cross-campus collaboration. We were fully responsible for creating a successful collaborative model. We assumed all the risk associated with an untested model. We faced many challenges and tweaked the rules as we went -- *and we were successful!*

The keys to the UCSD/UCI Collaboration's success were:

- **Full executive support:** This is detailed in the section below.
- **Frequent, high-quality communication:** We overcame communication challenges stemming from geographical distance by implementing weekly project conference calls, online collaboration software (Team Forge Enterprise Edition), and in-person strategy and working sessions.
- **Clear roles and responsibilities:** Both the provider campus and customer campus are accountable for the success of the collaboration. The AP units provide business requirements and verification, while the IT units provide software solutions and integration.
- **Collaborative requirements and verification:** *Review* is in the early stages of a multi-year development program that requires constant decisions, analysis, and testing. Using a collaborative approach ensures that both campuses have input into the future direction of *Review*.
- **Acknowledge and fix what's broken:** Cross-campus collaboration is not always rainbows and sunshine. When something is not working and people are getting frustrated, it is important to recognize it and quickly take action. The collaboration managed to overcome several issues that were hindering our effectiveness.
- **Empowered project managers:** Much of the day-to-day decision making is delegated to the collaboration's AP and IT project managers, allowing for quick action without elaborate approval and review processes. Executive approval is important only for the most critical decisions and resource commitments.
- **Documented vision and purpose:** It is easy to drift off course or become confused about where a project is headed without a formal, written agreement describing the overall vision and purpose of the collaboration. UCSD and UCI recognized this and created a formal [memorandum of understanding](#).

### **Full Executive Support**

From the moment Shohreh Bozorgmehri (UCI), Emily Deere (UCSD), Kristina Larsen (UCSD), and Joan Tenma (UCI) signed the Memorandum of Understanding, UCSD and UCI committed to doing whatever was necessary to achieve our vision of sharing *Review* and *Recruit*. Executives reassigned teams and adjusted priorities as needed, allowing project managers, developers, analysts, designers, and testers to succeed.

Executive support was critical in the following areas:

- **Implementing shared business processes over application customization:** Higher IT costs associated with excessive customization provide the impetus for open dialog about changing long-standing AP practices at both campuses.
- **Selecting best practices over the status quo:** The Collaboration looks first to eliminate steps that provide little value before creating software to support these processes. Over time, this will promote best practices for both campuses.

- **Focusing on collaboration goals over local needs:** It is not possible to launch applications for other campuses without impacting other local projects. UCSD and UCI executives directed resources to the cross-campus collaboration, believing it to be a more effective way of meeting UC-specific AP needs in the long run, as compared to traditional single-campus application development approaches.

### The Collaboration: Looking Forward

Having achieved our initial goal of sharing *Review* and *Recruit*, our future challenges are to implement key enhancements for both applications, enable more applications for the collaboration, and expand *Review* and *Recruit* to additional UC campuses.

Future milestones for *Review*:

- **Online archival support:** Allow Academic Personnel Offices to fully manage review file archives online.
- **Promotion support:** fully support the faculty promotion process.

Future milestones for *Recruit*:

- **UC-wide AP *Recruit*:** AP directors from every campus unanimously support deploying *Recruit* to the entire UC system.

Upcoming opportunities for application sharing:

- ***e-Recruitment Plan (e-RP)***, developed by and deployed at UCSD, is the next logical sharing opportunity. *e-RP* handles approvals for opening faculty recruitments and is the precursor to the academic personnel recruitment and appointment processes.

### Technology utilized in the project

*Review* is a multiple-tier, distributed, end-to-end Java enterprise Web application running on top of the Apache Web server and Red Hat Enterprise Linux operating system. The application is divided into four areas of responsibility: Web Application Presentation, Web Application Business/Data Layer, Integration Layer, and Authentication Layer.

The following table lists the technologies used for this project and their benefits.

Application Area	Technology	Benefits
Web App Presentation Layer	Spring MVC	<p><i>Review</i> is a highly interactive application, with more than 40 points of integration between the application's UI and backend.</p> <p>To more quickly develop this application, we chose the Spring MVC framework. Because of its rich selection of Controller implementations ranging from the simplest (the Controller interface) to the very powerful (AbstractWizardFormController), it provides a consistent approach to handling requests and allows us to concentrate on developing application features.</p>
	Yahoo UI	<p><i>Review</i> has complex user interface requirements. By using Yahoo UI, we are able to take advantage of a rich assortment of ready-to-use cross-browser interface widgets.</p> <p>As a result, we avoided the time-consuming task of user interface development and yet were able to deliver a user interface as specified.</p>

	<b>jQuery/Ajax</b>	To provide users with an interactive experience, <i>Review</i> uses Ajax and jQuery to retrieve data from the database asynchronously in the background without interfering with the display and behavior of the existing page.
	<b>iText</b>	<p>To adhere to the confidentiality of the review process, <i>Review</i> is required to compile different sets of documents into PDF format, which is accessible by users via the application's interface at various stages of the review cycle. A merit review file can have up to 23 different types of PDF documents and will be compiled at least 16 times. A CAP review file can have up to 42 different types of PDF documents and will be compiled at least 45 times.</p> <p>To reduce programming efforts and to avoid developing complex logic for generating PDF documents, we took advantage of iText. This open-source software provides a rich PDF generation library that allows <i>Review</i> to interface via API to create complex PDF files.</p>
	<b>Spring Themes</b>	<i>Review</i> leveraged existing framework components to brand the application according to UCI specifications. More details about Spring Themes are discussed in the sections below.
	<b>Factory Method, Adapter and Strategy patterns</b>	We leveraged three design patterns when implementing campus-specific customizations. More details are discussed in the sections below.
<b>Web App Business/Data Layer</b>	<b>jLink DAO</b>	<i>Review</i> uses UCSD ACT's standard interface for accessing data. This interface provides an object-to-relational persistence and request service for Java.
	<b>UCSD ACT's BPEL Workflow</b>	<p><i>Review</i> has over 50 decision points whereby a review file is routed to different roles within the application. Implementing complex routing requirements would ordinarily be a time-consuming process and would complicate the application business logic.</p> <p>To solve these challenges, we utilize UCSD ACT's existing BPEL Workflow to take over the complex routing logic. The BPEL Workflow works in conjunction with BPEL client software, which provides an intuitive, drag-and-drop interface for building the routing logic. The BPEL Workflow allows us to separate the routing logic from the application's business logic code, which provides the opportunity to perform parallel development and assign technical resources with appropriate expertise to a development task.</p>
<b>Integration</b>	<b>RESTful Web services</b>	Real-time and secured data feed from UCI to APOL. More details are discussed in the sections below.
<b>Authentication</b>	<b>Shibboleth</b>	Allows for cross-campus authentication. More details are discussed in the sections below.

To transform *Review* into a multi-campus software service and provide a framework for future collaboration, we enhanced *Review* in the following areas: branding, customizations, interoperability, and client data security.

As we embarked upon solving these challenges, we agreed to use the following principles in our solutions:

- **Develop campus-agnostic solutions.** Solutions should not be specifically designed to solve UCI's *Review* implementation. Instead, solutions should be reusable so that other campuses may more easily implement APOL.
- **Leverage existing technologies.** When implementing the multi-campus components of APOL, use open source frameworks or design patterns where possible. This approach saves development time by allowing developers to take advantage of proven and well-documented technologies.
- **Maintain one code base** that contains all campus-specific branding and customization required by each campus.

## Branding and Customization

### Challenges

- To provide UCI users with a familiar look and feel, branding UCI's version of *Review* was necessary.
- There are minor differences between business processes at UCSD and UCI. Where UCI was unable to change its business processes, developers customized *Review*'s business logic.

### Solution to branding

The elements that can be branded in *Review* are images (e.g., campus logos), colors, and campus-specific language used on the Web pages. We used Spring Themes as the technology to implement branding. Spring Themes is a subset feature of the Spring Framework, which is the base architecture of *Review*, so it is a natural fit.

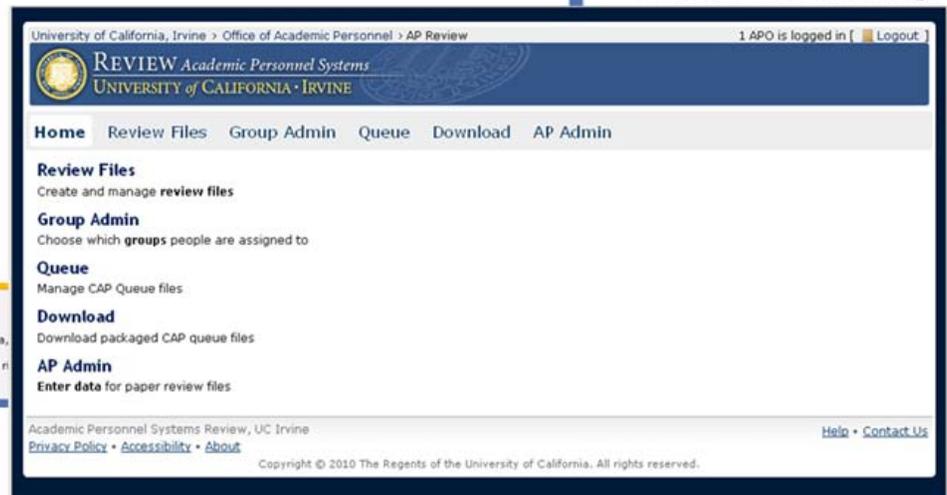
Spring Themes allows us to create generic software that renders Web pages using resource files that can be customized for each campus. All images and text on the Web pages is determined by a campus theme. This approach allows us to group all branding requirements for a campus and associate them to a theme. Thanks in part to the use of Spring Themes, adding more campuses will involve much less future software development because much of the work will be in developing the resource bundles and producing images, not in code changes.

(See application home page branded for UCSD and UCI on the next page.)

### UCSD's APOL Home Page



### UCI's APOL Home Page



### Solution to customization

Some differences between UCSD and UCI business processes were inevitable. We anticipate that other campuses may also need some level of customization. For UCSD and UCI, we abstracted the business processes that differ into their own plug-ins. As *Review* is implemented at more campuses, we will continue to create more abstractions and specific implementations for each campus via customization.

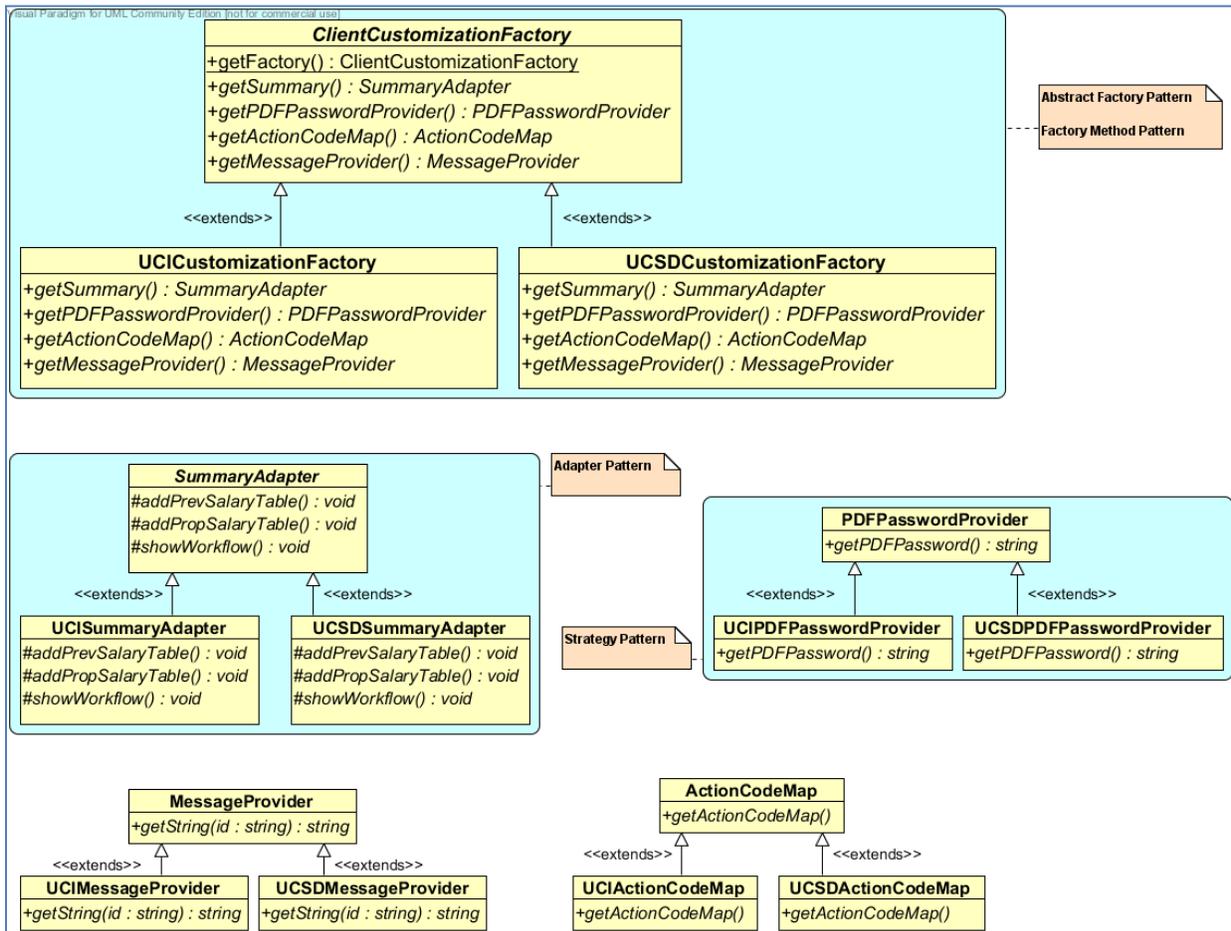
Below are the design patterns that were used to address the customization requirements. Going forward, we will continue to leverage other design patterns as appropriate for future customizations.

**Strategy Pattern:** This pattern provides a way for us to implement features where UCSD and UCI have different algorithms to derive values to the same data type. An example of this implementation in *Review* is the password generation algorithm. UCSD and UCI use different algorithms to generate passwords for CAP Review files. Using this pattern, we are able to extract the incompatible logic from the common code into separate code and then implement each campus's requirement for password generation.

**Adapter Pattern:** This pattern provides a way for us to implement features where UCSD and UCI have incompatible data requirements. One instance in which we used the Adapter Pattern was when a department enters a faculty member's proposed salary: UCSD allows input of a "bonus off-scale" dollar value, but UCI does not. The implementation of this pattern is similar to the Strategy Pattern, but it has the additional task of transforming the data incompatibility.

**Factory Method Pattern:** This design pattern serves as a basis for the Adapter Pattern and Strategy Pattern; it allows us to define the common code and the customization code. This pattern allows us to reuse code where there are common business processes, and to develop code where business processes diverge.

### Customization Class Diagram



All customized code is managed by Spring Themes, allowing us to keep campus-specific code separated.

#### Solution for single code base management

To avoid the complexity of code management tasks and allow for code reuse, we chose to have one code base for all UC implementations. To guarantee that inadvertent mixing of different campuses' customizations (for example, that UCI's customization A is not deployed to the UCSD version), all of the customizations of each campus are treated as a logical set of customizations. The mechanism of grouping customizations by campus is the same as the branding grouping mechanism, which is Spring Themes. Coupling the logical sets of customizations and branding guarantees that each version of Review contains the appropriate branding and customizations.

#### Interoperability

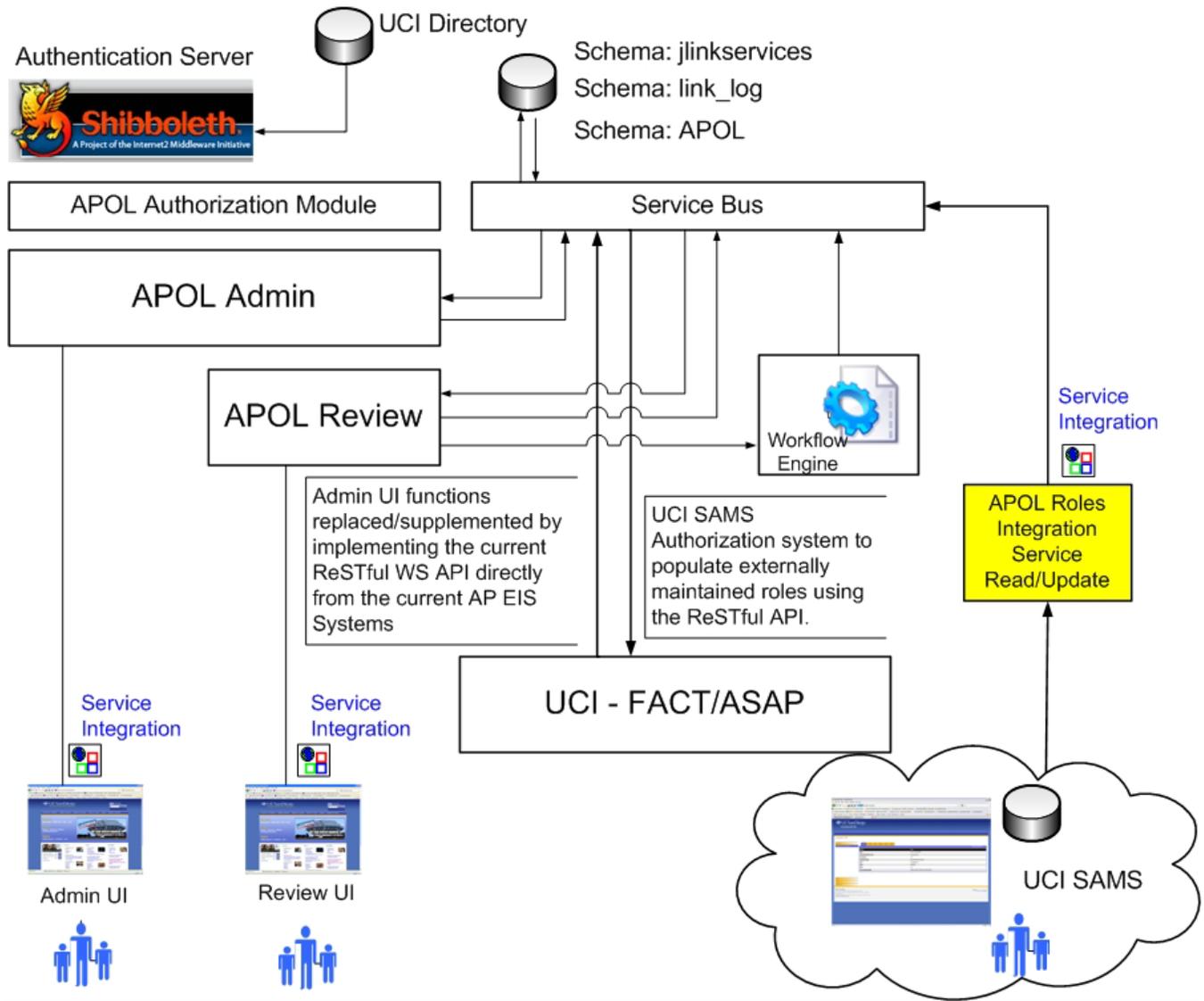
##### Challenges

- Allow UCI applications to be the system of record for APOL's supporting data, such as roles and employee and employment history data. Cross-campus data integration is required to allow Review and UCI to share data.
- Provide UCI users a seamless authentication and authorization experience. To do so, we needed to achieve the following:
  - Provide UCI users with a single sign-on experience similar to other UCI applications.

- o Ensure that *Review* provides users with appropriate access to review information once they enter the application.

Below is a component diagram showing how we met these challenges.

### Interoperability Architecture Diagram



### Solution for cross-campus integration

A bidirectional integration was needed to provide a means for exchanging data between *Review* and UCI databases:

- Transmission from UCI to *Review*
  - Employee data: Provide employment status, directory information, and past review data.
  - Roles: Provide levels of authorization required to meet privacy standards and/or the requirements of the functional office for data security. (e.g., read-only access to the data for selected users, or limited access to certain data elements for selected users).
- Transmission from *Review* to UCI  
When a review file is completed in APOL, the final appointment data can be transmitted to UCI's database. This transmission is currently available in APOL, and UCI is planning to take advantage of this integration in the near future.

We implemented the bidirectional integration with a set of SOAP Web services. The development team took advantage of existing *Review* RESTful objects that provided the plumbing for data access and also satisfied the integration's data access requirements. Instead of starting from scratch, the integration implementation tasks were narrowed to data mapping, UCI's data extraction, and developing the SOAP Web service interface wrapper tasks.

### Solution for authorization

We solved the authorization requirements by developing a cross-campus authentication capability and integrating the *Review* roles module with UCI's access control system.

**Cross-Campus Authentication:** We enhanced APOL's authentication facility to allow users to use both UCSD and UCI Shibboleth identity providers. UCSD and UCI Shibboleth identity providers are responsible for delegating authentication to campus single sign-on systems. Once the user's identity information is authenticated, the identity provider passes it to APOL.

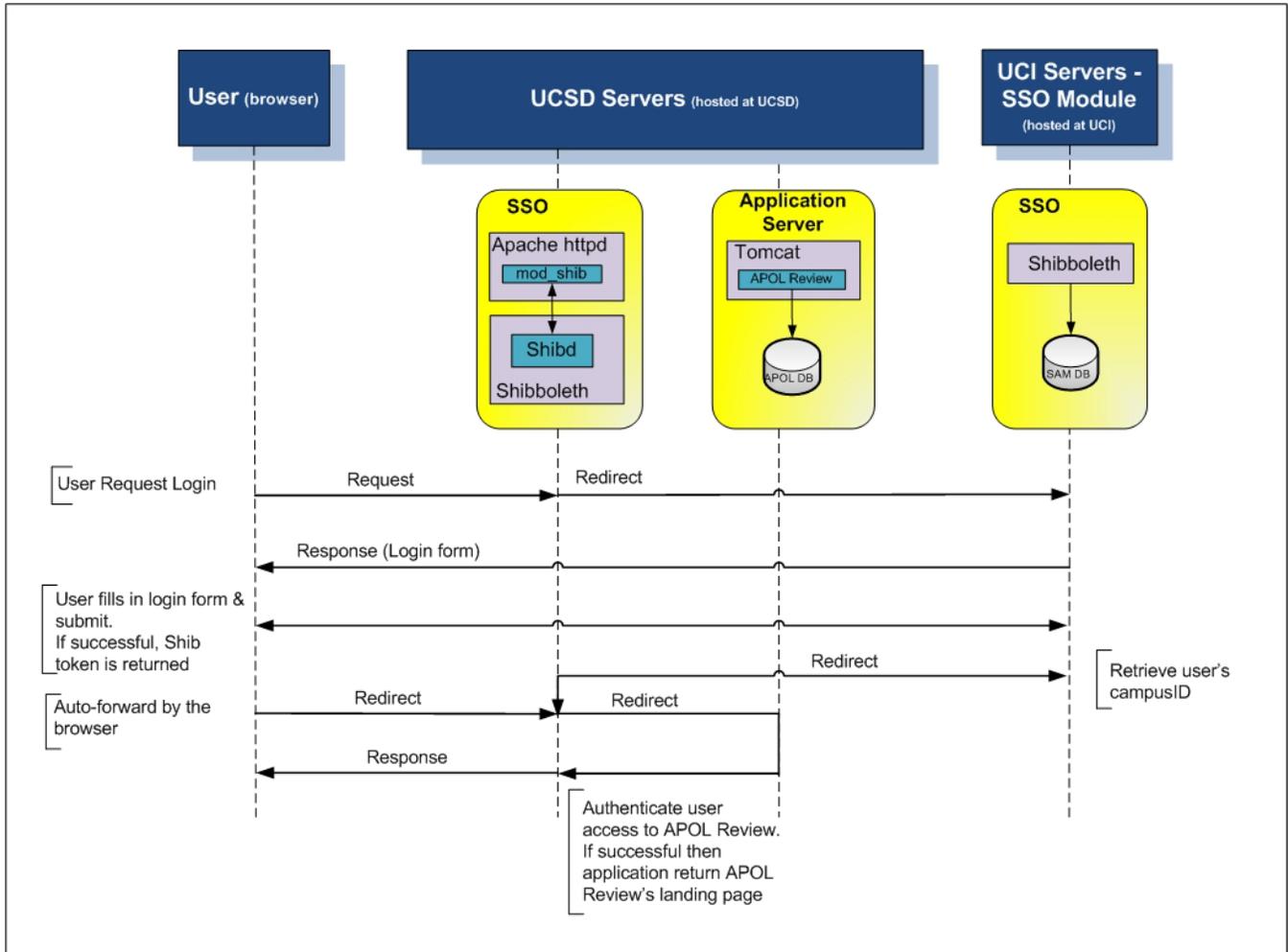
**Roles:** Once *Review* receives a user's identity information from the Shibboleth identity provider, *Review* then determines the user's access within the system using role-based access control. Users with no access are directed to a campus-branded "access denied" page.

Most role information is provided by campus access control systems. UCSD role information is batch imported from the Affiliates application. UCI's role information originates from the SAMS application and is transmitted to *Review* via an hourly process using SOAP Web services.

Going forward, UCSD and UCI will use the experience gained from this project to provide the method of cross-campus authentication in future SaaS deployments.

(See diagram on next page)

Overview of Authentication and Authorization

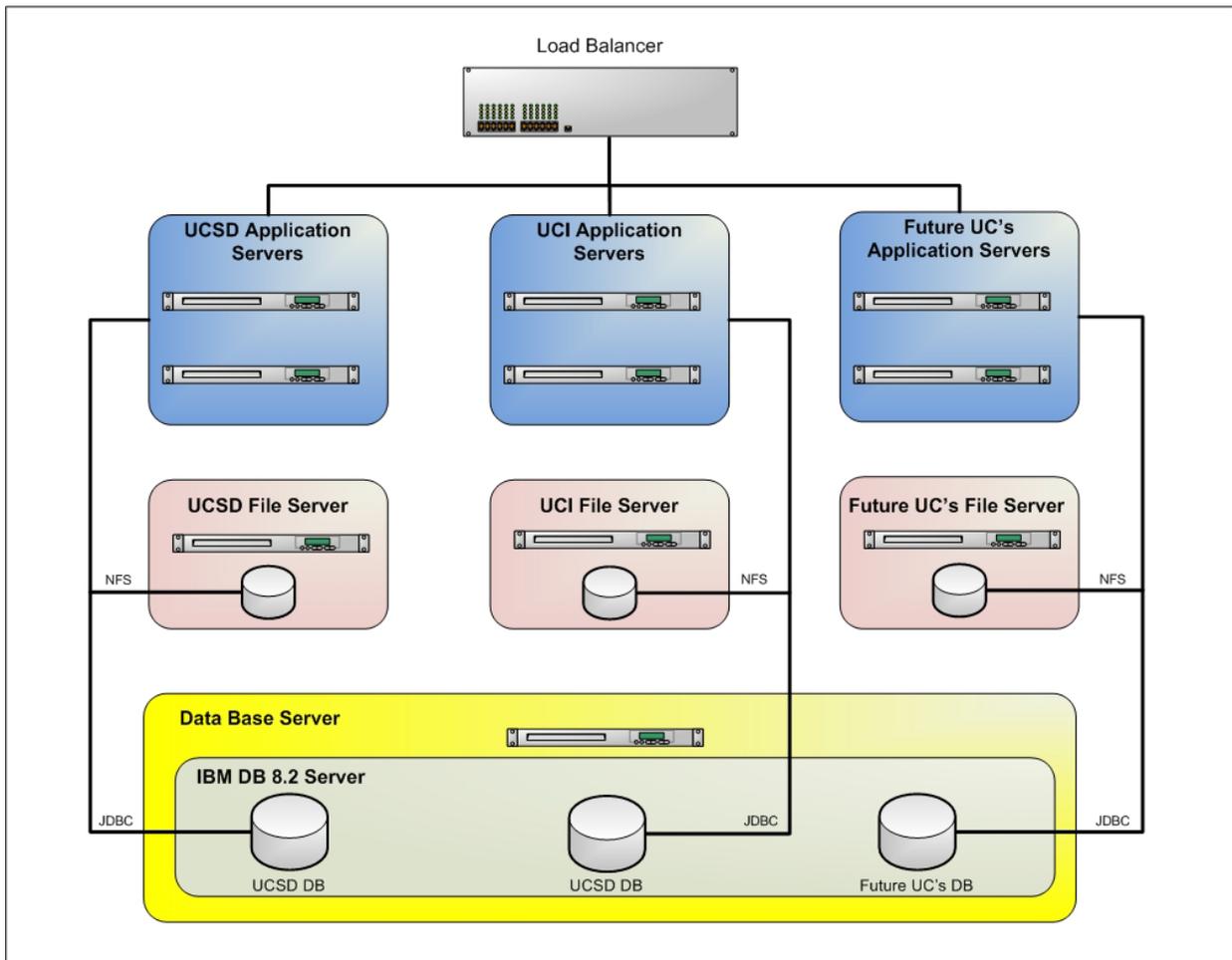


### Client Data Security

One of the concepts of the SaaS model is that data is stored at the hosting campus; in our case, UCSD stored both UCSD's and UCI's data. However, APOL contains sensitive review data that must be secured and isolated. Exposing UCSD and UCI users to each other's data could have serious consequences. Therefore, it was imperative that we build a mechanism to expose each client's data to only its own instance of *Review*.

We addressed this issue by isolating data via dedicated hardware and database partition. We have built dedicated servers and data storage areas for each client campus in order to host non-database data. Database data are isolated by hosting data in different database partitions for each client campus.

### Server Topology



### **Timeframe of implementation**

**Oct 2008:** *Review* 1.0 (UCSD)

Implemented Merit Review feature, which allowed merit review files to be processed in *Review*.

**Sept 2009:** *Review* 2.0 (UCSD)

Implemented *Review*'s Admin feature, which allowed AP staff to record decisions regarding paper review files in the same database as those for online review files.

**May 2010:** *Review* 3.0 (UCSD)

Implemented CAP Prelim feature, which allowed CAP files involving preliminary decisions to be processed in *Review*.

**Aug 2010:** *Review* 3.0 (UCI)

Transformed *Review* into a multi-campus software service and rolled out *Review* to UCI.

**Nov 2010:** *Review* + Single Code based 3.5 (UCSD and UCI)

Merged UCSD and UCI implementations of *Review* into a single code base.

### **Customer Satisfaction**

Since this project involved delivery of a fully configured version of *Review* to UC Irvine, UCI can be regarded as a "customer," even though the two campuses are collaborative partners.

Shohreh Bozorgmehri, director of Academic Network Applications, commented on the success of the Software as a Service model:

*Our initiative focused to bring long-term cost savings by eliminating redundant business processes and IT development efforts across multiple campuses. Our technical team initiated a shared model of development using SaaS (Software as a Service), providing faculty and staff with an interactive enterprise system for processing elaborate faculty merit review files online. This undertaking demonstrates a strong collaboration among key partners to integrate each campus's requirements and priorities into a cohesive shared vision.*

Joan Tenma, director of the UCI Academic Personnel Office, delineated how *Review* meets UCI's business needs:

*The benefits of our collaboration reach beyond IT efficiencies and savings achieved from eliminating the development of redundant systems. We benefit from the synergy generated by our AP and IT collective teams, enabling us to find collaborative solutions and focus on what provides value in our business processes.*

At another level, the customers for *Review* are, of course, the end users in the divisions and departments at both campuses. At UCSD, end users have now had two years of experience with *Review* and have commented favorably on the conveniences and efficiencies it offers.

Kim James, Academic Personnel Manager for the Division of Biological Sciences at UCSD, provided some examples:

*Having the convenience of signing [certifications] via Single Sign-On was very helpful, since I was submitting files during the summer and early fall when some faculty were not*

*on campus (field work). Per policy, faculty have access to see their files, and Review has streamlined this, as faculty can review (with the exception of confidential materials) their complete files. Also, during the review process I have many working documents, and there are notes/annotations that would be included in the paper files. I particularly like that the final file in Review contains the complete file with annotations and all. Also, calculating the next salaries on the summary is a lot easier, since the MOS [market off-scale salary amount] is available. I find that it serves as a double check for my math.*

Collette Isachsen, Academic Affairs Manager for the Department of Political Science at UCSD, offered this assessment:

*I have been using Review at the department level since it was first introduced for the 7/1/09 review cycle and have now processed a dozen files within the system. Based on this experience, I believe Review offers numerous advantages over the previous paper-based process from an administrative perspective and is dramatically more convenient for faculty members being reviewed and faculty voting on review actions. Most notably, it provides an easy-to-use, secure way of sharing file materials with everyone involved the process, regardless of where they might be. Given that most of our faculty work flexible schedules and often travel around the world for their research, this has significantly reduced the amount of coordination and total time required to complete reviews at the department level. There are numerous other specific benefits I could highlight, but will summarize by saying that overall Review is a giant leap forward that has made the processing of academic review files simpler and faster while fully maintaining, and in some ways even enhancing, the integrity of the process.*

AT UCI, pilot-phase users were also enthusiastic. Tracy Calvert, a Personnel Analyst in Pharmaceutical Sciences at UCI, made the following observations:

*I only have positive things to say about my experience with Review. I most appreciated how Review saved me the usual footwork required in coordinating the access to the file for the drafting committee, faculty, and Dean. I also loved saving the trees by having the ability to upload the pubs, which in some cases can be hundreds of pages long. This being a pilot, it was inevitable that some questions and technical issues would arise, but the support staff quickly answered my questions and/or fixed any problems. Overall, I feel that Review is very user-friendly and it shaved the time required for file preparation and distribution. I hope to use AP Review for ALL my cases in the near future.*

Patti O'Dorisio, director of Academic and Staff Personnel at the UCI School of Law, noted significant time savings and improvements in efficiency:

*As a member of the pilot group for Review, I want to let you know how appreciative I am for everyone's effort in launching this much-needed program. I found Review to be highly intuitive for the user, and it greatly aided our School in processing files in a much more efficient and timely manner. All staff and faculty involved in an academic review process will greatly benefit from the use of Review; not only is it easy to use, the cost effectiveness of the program is a value-added feature that is greatly beneficial to the University in an era of financial constraints. Review is an excellent example of the greater good that is realized when strong leadership and creative minds work together. Thank you again for allowing me to be part of such a worthwhile effort.*

## **Appendix: Background and Features of Review**

[\(return to Introduction\)](#)

In response to a Senate-Administration task force recommendation that UCSD develop an online system for creating and processing review files, UCSD Academic Personnel Services (APS) began studying online systems at other UC campuses and identifying its own system needs, which included:

- assembling review files electronically
- collecting data via key entry, uploads from campus databases, and scanning/attaching electronic file documents
- routing files to reviewers electronically
- capturing details of file routing and reviewers' recommendations in a database structure
- maintaining a PDF of the file indefinitely in an archive
- ensuring the confidentiality and security of review materials

UCSD ACT was charged with developing an application to meet UCSD's specific needs. Because the review process is so complex, it was decided to develop *Review* in stages. The initial focus (Phase I) was on processing merit advancement files for which deans have final authority. Phase I was released to the UCSD campus in July 2009 for use during the 2009-2010 academic year. Phase II—processing files requiring review by the Committee on Academic Personnel and a final decision by the Senior Vice Chancellor—was released to the UCSD campus in May 2010.

### **Business needs for an online system**

The paper-based academic review process is complex, time consuming, and labor intensive. Processing a review file can take up to 15 months from the time an academic submits the required information until he or she is informed of the final decision. Drawbacks include the following:

- It can be difficult to respond to recruitment and retention issues in a timely manner, potentially resulting in loss of current faculty or prospective recruits.
- Delays may necessitate retroactive pay increases, which can create significant problems in the financial administration of contracts and grants.
- The status of files is not readily apparent to candidates or departments, which can create impatience and frustration with the academic review process.
- It is difficult to ensure that departmental procedures conform to policy, which can create risk for the university.
- Use of paper (multiple copies of review files and publications) is cumbersome, costly, and wasteful.

Conversely, the expected advantages of an online system were seen to include the following:

<b>Paper process</b>	<b>Advantage of online process</b>
Review files are printed and must be manually routed either by student employees or via campus mail or messenger service for sequential review.	<b>Eliminates printing time and costs; eliminates manual routing of files and associated delays and costs.</b>  Several days can be saved if files do not have to be manually routed for sequential review. All reviewers receive access to the online file at the appropriate time, and simultaneous access is possible if needed. The costs of routing are eliminated.
Multiple copies may be required to accommodate simultaneous review in urgent cases such as retentions and to provide copies for department files.	<b>Eliminates copying time and costs.</b>  All reviewers receive access to the online file at the appropriate time, and simultaneous access is possible if needed. There is no need to retain paper copies.
Departments solicit evaluation letters from external referees. If submitted via e-mail, evaluation letters must be attached to cover letters to ensure authenticity. Cover letters and evaluation letters are printed for the file. Letters are manually redacted if the candidate requests access. A referee identification document is prepared to inform reviewers of referees' qualifications.	<b>Streamlines submission of external letters and identification of reviewers.</b>  External referees receive temporary access to directly upload evaluation letters, eliminating the need for authentication, printing, and copying. Information identifying referees can be placed in fields that can be screened out, providing automated redaction. Referee identification data can be used to automatically populate the required referee ID form.
A set of reprints of the candidate's publications is provided by the department, and the set is routed for sequential review. If simultaneous review is necessary, multiple sets of reprints are required. Routing is cumbersome, especially with large boxes of publications.	<b>Eliminates routing of paper publications.</b>  All reviewers receive access to publications at the appropriate time, and simultaneous access is possible if needed. The cost of copying and routing publications is eliminated.
Departments, divisions, and APS maintain various duplicative file tracking systems or databases.	<b>Eliminates duplication of file tracking.</b>  Central tracking of files reduces the workload, thus speeding up file processing at every level of review.
Candidates typically prepare and submit their review materials up to a year before the proposed action will take effect, and they and their departments often do not know the status of their reviews during the review process.	<b>Improves the transparency of the review process.</b>  The system continually tracks the status of all files. Candidates, departments, and other users can easily see where a file is in the review process at any time.
When file review is complete, APS stores paper files and makes them available for reference at the subsequent review, as well as for administrative research purposes between reviews. Maintaining these files requires significant staff time and effort, as well as substantial storage space. Access to storage space must be restricted to ensure confidentiality. Duplicate files are stored off-site to provide back-up. Older materials are removed from files and microfilmed to conserve space. Files can only be checked out and accessed by one person at a time, and misfiling can occur when files are returned to storage.	<b>Eliminates physical storage, makes files easily retrievable, and ensures confidentiality.</b>  Files are placed in an online archive at the end of the review process, and the archive is protected by the nightly online backup process. Access is controlled by the system to ensure confidentiality. Simultaneous access to files is possible if needed. Saves time, space, and the cost of storing/microfilming paper files, since multiple offices can view the same archived files. Eliminates filing errors.

Paper process	Advantage of online process
Accessing data regarding review files is time consuming, and users may not know where to look for it. Data may be inaccurate due to data entry errors.	<p><b>Improves data accessibility and accuracy.</b></p> <p>Data is easily accessible via reports created by the system. Validity checks at time of data entry ensure improved data accuracy.</p>
There is no way to ensure that departmental procedures comply with university policies, which may expose the university to risk of formal complaints or legal action.	<p><b>Safeguards faculty due-process rights and reduces risk to the University.</b></p> <p>Adherence to policies and procedures is built in to the system. The review process can only be completed in the required sequence and with the required checks and balances, thereby protecting the candidate's due-process rights and reducing risk for the university.</p>

### File review processes currently supported

Six stages of the review process are currently incorporated in *Review*:

- File preparation: Required documents are uploaded as individual PDFs that can be removed and replaced as needed until the file is complete. Candidate must certify that he or she has had the opportunity to submit materials and view all non-confidential materials in the file. All individual file documents are automatically “bundled” into a single PDF, and individual documents can no longer be changed or replaced unless the file is returned to File Prep and recertified.
  - Departmental review: A departmental ad hoc committee may review the file and provide a recommendation. The Candidate is offered the opportunity to view a redacted version of ad hoc report and must certify that access has been offered. Eligible faculty may review the file and vote on the proposed action. The department adds a letter stating its recommended action. The candidate certifies that he or she has been informed of the recommendation and may upload a response and supporting materials before the file is submitted to the divisional dean.
  - Dean’s review and final action: If deans have final authority for the proposed action, the file enters the Final Authority stage of review. The dean’s staff inspects the file and can return it to the department if substantive corrections are needed, or accept it for the dean’s review (and annotate it if there are minor errors). The dean issues a final action letter.
- or*
- Dean’s review and recommendation: If deans do not have final authority for the proposed action, the file enters the Dean’s Review stage. The dean makes a recommendation regarding the proposed action and the file is submitted for campus review.
  - Campus review: The Committee on Academic Personnel (CAP) reviews the file and can request additional information and/or make a recommendation regarding the proposed action. (Future phases of development will provide for review by other entities, such as review panels other than CAP.)
  - SVCAA/Vice Provost’s review and final action: A preliminary decision is issued if campus reviewers disagree with the proposed action, and/or a final action letter is issued.
  - Post-audit: APS staff members audit files for which deans have final authority to ensure that they are complete and in compliance with policy, then enter data regarding the files.

Upcoming additions to these functions are outlined in "[The Collaboration: Looking Forward](#)," and longer-range goals are listed in the [AP On-Line Strategic Plan](#).

### **Security and Confidentiality**

As noted in the Project Description, the security of the Review application is safeguarded by a single sign-on system. To protect the confidentiality of review materials, all users are assigned roles that determine what file materials they can see and what actions they can take, and when.

### **Protection of Due-Process Rights**

To protect their due-process rights under UC policy, candidates are notified to sign certifications acknowledging that they are aware they are under review; have been informed of reviewers' recommendations; have been provided access to non-confidential file materials; and have had the opportunity to comment on all materials in the file. Certifications are signed at prescribed points in the review process, and file processing in *Review* cannot continue unless the certifications are signed. Candidates can also request access to redacted copies of confidential materials, and the system ensures that they are informed of the final outcome of the review process.

### **Transparency**

All roles can see the status of any file by viewing the "File Tracking" screen in *Review*, which shows a list of actions completed or pending (e.g., "Dept Ad Hoc Cert Pending" and "Dept Ad Hoc Cert Signed"), and the date and time of each action. In addition, the majority of roles have access to a "Certs and Notifs" screen, which shows all certifications signed and all notifications sent by one role to other roles, or sent by the system itself, with a date and time stamp for each.

### **Compliance with policy**

In general, compliance with policy is built in to *Review*, because steps in the review process can only be completed in the prescribed order, and only certain roles can complete certain actions. Thus, for example, departmental ad hoc committees are prevented from reviewing incomplete files, and only those assigned the appropriate roles can sign letters (e.g., dean's final action letter). Further, all steps are documented. This automatically enforced compliance helps ensure fair treatment of candidates and reduces risk for the university.