Symplicity Corporation
Security Standards

Symplicity Corporation
1560 Wilson Blvd
Suite 550
Arlington, VA 22209
P: 703-373-7026
F: 703-351-6357
www.symplicity.com

Symplicity Vision

Symplicity is committed to providing excellent products and services. Symplicity is also committed to ensuring the privacy of data by hosting client's data in an Equinix IBX Data Center facility. Symplicity is committed to organically growing our presence in the federal government market. All government systems designed by Symplicity employ the Symplicity e-Business Platform to provide agencies with a comprehensive set of tools to increase the level of coordination between their partners, constituencies, and users.

Physical Hosting

Horizons operates 24 x 7 x 365 in a secure hosting facility operated by Symplicity Corporation and data center provider Equinix. Symplicity retains full control of the hardware, operating systems, and network equipment hosted within the Equinix facility, while Equinix provides redundant power, cooling, connectivity options, fire suppression, and physical site security. Symplicity encourages all clients to review the Equinix Security video for a better understanding of plant security, available at:

http://www.equinix.com/prod_serv/ibx/ibxCenters.php#

Horizons and its components are 100% web based and are accessed via encrypted Secure Socket Layer (SSL) communications providing end to end security from Symplicity to the client. Built off of proven technologies in the SympleObjects Framework that has successfully been deployed at federal agencies, over 600 colleges and universities, and private enterprises, Horizons is built off of a framework that has been tested and refined over eleven years.

Symplicity's hosting provider partner offers multiple world-class facilities. Each center is staffed 24 x 7 with highly trained, experienced support engineers. Support engineers are required to go through annual code of conduct review. All facilities follow a well-defined security policy. This policy is regularly audited and modified, as needed, in order to maintain the highest standards.

- Category 7 and higher cabling throughout the facility
- VESDA®* fire alarm system
- Battery backup and diesel generators with fuel onsite
- Biometric security, with 24x7 Security staffed check-in

For an online tour of the facilities we use, please visit:
http://www.equinix.com/prod_serv/ibx/ibxCenters.php#

Backup Power

Dual power is available to each rack unit from independent power distribution units (PDUs), removing PDU loss as a single point of failure. To ensure stable connectivity, Symplicity offers a redundant Need+1 (N+1) design of uninterruptible power supplies utilizing two separate commercial power feeds from two separate power grids to provide power to the mission critical servers. Redundant stand-by generators guarantee consistent operation in the event of a power failure from commercial power. Generators are equipped with 72 hours of diesel fuel on site, and contracts are in place to refuel the generators in the event that primary power fails.

Fire Protection
Fire detection and suppression systems ensure 24/7 operation of critical systems. Multiple fire detection (photo electronic/ionization and sniffer) systems are in place to constantly monitor for any potential areas of concern. Multi-zone pre-action dry pipe suppression systems allow any suppression necessary to be contained to affected areas while leaving conditions elsewhere uninterrupted.

Infrastructure Built for Reliability and Scalability
Symplicity's leverages Internap (www.internap.com) for internet connectivity and transit. Internap aggregates over 13 different Tier 1 backbone providers (including AT&T, Verizon, Qwest and Level3) over multi-gigabit links to ensure that internet connectivity is maintained in the event of a network outage.

Utilizing a unique network infrastructure and proprietary routing technology, Symplicity can deliver unsurpassed speed, reliability and virtually unlimited scalability.

Fastest Route Available
Symplicity solutions utilize BGP to provide intelligent, fail-safe routing to route users to the fastest available connection to ensure maximum speed and virtually eliminate down time. Additionally, Symplicity utilizes the Internap Route Control Technology that constantly monitors end destinations that automatically routes around slow and highly congested traffic links.

Redundancy
Redundancy is built into the network architecture through redundant network connections and redundant router, switch, firewall, and load-balancer configurations.

Physical Security
Facilities are manned with 24/7 security personnel. Badge/photo ID access screening as well as biometric access screening are employed for an added level of security. Sensitive security areas within the center are protected by motion sensors and security breach alarms, and video cameras installed throughout the center are continuously monitored by security personnel.

Network & Internal Systems Security
All of Symplicity's systems are designed to maintain a quality of service throughout the network by prioritizing packets to maintain an order of precedence of services. Each network device is equipped to filter out DOS attacks. We also offer intelligent, multilayer access control that can be utilized to protect the network from the impact of attacks while reducing the risk of inadvertently discarding legitimate traffic.

Access to all information systems and resources is only provided on a "need-to-know", "least privilege" basis. Access to system/network element management and configuration is controlled with user IDs, strong passwords and proprietary key encryption technologies that identify the specific individual that is attempting access. All passwords are required to meet corporate standards that ensure the password is of sufficient length and complexity to prevent guessing or simple "brute-force" attacks on the account.

Passwords must also be changed on a regular basis. Symplicity periodically audits system accesses and stores audit logs at a remote site. Inactive personnel are disallowed access promptly following their departure or at the customer's request.

Symplicity maintains policies on user ID and password standards and minimum-security configurations for all UNIX and Microsoft® operating systems.

Information Classification and Control
All customer information is considered classified and is handled as such.

Managing the Security
Managed Firewall Services places a layer of security between the application and the Internet. All inbound data traffic flows to a firewall, which filters traffic detecting and deflecting unwanted attempts to penetrate the server security.

Symplicity also offers Managed VPN (Virtual Private Network) Services. The VPN encrypts all traffic between two internet points, providing secure communication channels for individual users, user groups, contractors, vendors and remote offices. Managed VPN also aids in providing a high level of data integrity and protects key corporate information assets.

Other Security Mechanisms

All of Symplicity's solutions include the following security features to ensure client's data remains private, confidential, and secure:

User Connections (HTTPS)
All connections to Horizons by users through a web browser are encrypted using 128-bit SSL encryption. No unencrypted traffic shall be allowed into the system. Connections will also be subject to an authentication challenge before system interaction is granted.

Manager Connections (HTTPS)
Like user connections, management connections, i.e. connections by college systems and department managers, will also occur using 128-bit SSL to ensure secure communications. In addition to this, access to the management interface can be restricted by IP address range and granted only to users to fall within the valid range. This valid range will optionally be client-configurable, through the management interface itself. Connections will also be subject to an authentication challenge before system interaction is granted.

System Administrative Connections (SSH)
All connections by system administrators to the servers used in the infrastructure will be encrypted using the SSH version 2 protocol, and access will be granted only to people connecting with a valid private key. Connections will also be subject to an authentication challenge before system interaction is granted. The system supports the Triple Digital Encryption Standard (3DES).

Server security
Access to servers for maintenance and system administration is compartmentalized on a need basis pertaining to the specific tasks a given system administrator needs to perform. For example, the backup operator has access only to those commands and functions needed in order to perform a system backup to tape. The database administrator has access only to those commands/functions related to database management, and so on.

Users are authorized using secure username/password authentication as is standard on Unix/Linux servers. 100% of remote access is encrypted through the use of the SSH protocol. Terminal access to servers via SSH will be closed to the entire Internet at the firewall level and shall be specifically granted only to management workstations of predetermined and preauthorized personnel.

Personnel Security

Appropriate reference and/or background checks will be conducted for Symplicity employees who have the capability to circumvent controls or who have access to sensitive data. User access will be restricted to the minimum necessary to perform the job.

Application Level Security

User security is accomplished using role based access control. Users are grouped into "roles", the set of which are to be determined during the initial design phase of the project. Also, Symplicity incorporates an incredibly rich audit trail in its solution capturing the following information on all system transactions (who, time, IP address, permission level, web browser used, and operating system). In addition, Symplicity maintains version control on all changes to the database.

All system functions which a user does not have access to are protected at multiple levels: 1) the user interface features such as links and buttons do not show up for users without specific access to those features; 2) the code behind a given feature checks users' access rights before performing any actions or presenting any data; 3) all system objects carry their own access control lists (ACL) which determine who "owns" them, who can see them, who can edit them, and who can annotate them. Additional object level rights may be defined in the design phase of the project. All communications between application components are secured using SSL encryption.

All SympleObjects based systems (including Horizons) have in-built cross site scripting and SQL injection attack recognition technologies which remove these threats.

Horizons includes a login-warning message that can be edited by the Systems Administrator. The users must accept the conditions in the login-warning message by clicking on an acceptance button to gain access to the login screen. If the user does not acknowledge acceptance of the login warning the system must exit the program.

Horizons does not use persistent cookies, nor will it place any type of permanent file on any client.

Database Security

The database servers are behind a pair of highly available firewalls and only accessible to the application through a specific firewall rule, and trusted users through encrypted channels. Communications between application and database are limited to the local network segment and are never exposed to a public network.

Connections to the database server are made using accounts with only the access level necessary for that connection. Connections needing only read-access to data, such as users browsing postings, are made using a database account with only read access to the specific database table they'll be reading. Similarly, update

connections are made through connections granted write access only to those databases and tables they need access to.

Symplicity supports ad hoc database field encryption per client requirements. Every client will be provisioned in their own database with custom defined user access parameters.

Additionally, client data is never stored on removable media and all backups are completed disk-to-disk preventing accidental physical data loss.

Data Security and Ownership

All data is owned and controlled by the customer. Should a server fail that contains client hardware, the server will attempted to be brought back to operating status. If irrecoverable, the hard drives storing customer data will either be low level formatted twice or destroyed.

Development of Application

Symplicity employs all developers and does not outsource any of the development of it's applications to third parties. All developers are trained on the SympleObjects framework and are placed on a one year probationary period. Additionally, all development completed on the SympleObjects platform that powers Horizons undergoes an extensive Quality Assurance process:

Budget: The project director reviews the budget for: cost overrun potential, and actual versus budgeted dollars to date.

Project Plan: The project team monitored by university project management review the project plan: to ensure milestone dates are on target, for any changes in scope, and for possible trouble spots.

Communication: The communications objectives are reviewed to: ensure that key dates are on target, ensure that communications are effective, and to alert the team to changes of scope.

Issue Tracking: The project team is effectively managed by project management staff, review the issue tracker to ensure that: target dates for resolution are being met, high priority items are being resolved, and the project is not in jeopardy.

As Is/To Be Document: The project team reviews project planning documents for completeness, scope of changes, conversion items, interface items, processes to be implemented, users sign off.

Data-mapping Documents: Functional and technical staff review the data mapping documents for completeness.

Functional Requirement Documents: Functional and technical staff review the functional requirement documents.

Technical Requirement Documents: Functional and technical staff review the technical requirement documents.

Programming Code Walk-throughs: Technical staff review programming code walk throughs. The technical lead signs off.

Procedural Documentation: Functional and technical staff review procedural documentation. The functional/technical manager signs off.

Training Materials: Functional managers review and sign off on training materials.

Operational Decision: The entire team reviews all aspects of the implementation to determine readiness to continue. This review examines: the results of all testing, security readiness, hardware readiness, network readiness, database readiness, and production readiness.

Problem Reporting

Clients will report all issues via Symplicity's Issue Tracking system: https://manager.symplicity.com or via phone. All correspondence between Symplicity and the College or University will be logged and tracked in real time.

Software Testing

All code will be tested to ensure that each individual class performs the required functions and outputs the proper results and data. Proper results are determined by using the design limits of the calling (client) function as specified in the design specification defining the called (server) function. Unit testing is typically white box testing and may require the use of software stubs and symbolic debuggers. This testing helps ensure proper operation of a module because tests are generated with knowledge of the internal workings of the module.

Integration Test

There are two levels of integration testing. One level is the process of testing a software capability. During this level, each module is treated as a black box, while conflicts between functions or classes and between software and appropriate hardware are resolved. Test cases must provide unexpected parameter values when design documentation does not explicitly specify calling requirements for client functions.

A second level of integration testing occurs when sufficient modules have been integrated to demonstrate a scenario. During this phase, composite builds, or baselines, of the software are married to the engineering versions of the hardware to evaluate the combined hardware/software performance for each operational function. Both hardware and software documentation is reworked as necessary.

System Testing

System testing begins when sufficient hardware and software has been integrated that enable the operation and functions of the integrated Horizons product.

Monitoring

Symplicity also monitors system operations, network uptime, and system security through a proprietary 24 x 7 network operations center (NOC) developed by Symplicity. The NOC system will alert Symplicity staff to exceptions for expedited recovery and resolution within

one minute of failure. Additionally, all components of Symplicity's infrastructure have redundancy built in to ensure that systems remain online in the event of an isolated server failure. Some of the techniques Symplicity employs:

Intrusion Prevention System

Symplicity employs full-featured IDS (Intrusion Detection System) with IPS (Intrusion Prevention Services) enabled. This is done through deep packet inspection signatures that are updated weekly. For security reasons we can not release the brand of our IDS/IPS solution.

System Status Monitoring

Symplicity employs a full-featured ping, http content checks, database query, file system stat checks, and SNMP queries to ensure that all systems are online. These probes are run continuously every minute to minimize the length of possible outages. Symplicity System Administrators and Network Engineers are alerted via audible and visual alerts with paging and text messaging alerts sent to a group in critical outages. This system allows Symplicity to offer a highly available SLA-backed solution to our end clients.

Reporting

Every component in Horizons is connected to Symplicity's Reporting Engine which allows 100% AD-HOC reports to be built in an intuitive and easy to use interface. Additionally, as all systems have a customizable data structure, each custom added field is also reportable. Reports may be exported to Excel or called via an integration method utilizing web services APIs.

Information Archiving

Symplicity components can be setup to archive and/or remove information on whatever schedule is mandated by the university or state law. Symplicity will work with the client to setup an archival process that accomplishes the goals of the university.

Application Architecture

Symplicity developers are trained in Secure Coding techniques and the company employs all relevant standards and best practices in high performance, highly scalable, n-tier, web applications architecture. Throughout the engagement Symplicity will comply with relevant federal statutory and regulatory requirements (including Privacy Act, 508), legal mandates.

Symplicity's approach builds on the foundation of open source and scalable tools that allow for more flexibility in developing, implementing and deploying the system. Dynamic load balancing across multiple web servers will allow the client to expand for a lower cost as new requirements are implemented and use increases.

Each component of the solution has been specified to provide the most efficient use of projected hardware and software. The approach includes a Secure Socket Layer (SSL) accelerator/reverse proxy cache appliance to allow load balancing at the switch level and to offload processor intensive tasks allowing for a more scaleable infrastructure. Each component can be scaled individually for greater flexibility.

The system will be compliant with University Enterprise Architectural Requirements. The Horizons solution will be comprised primarily of the following components:

1. **Web/Application Servers**

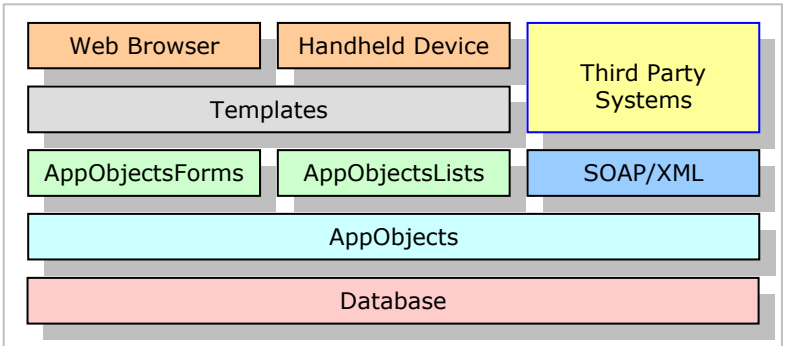
   a. The web servers will be used as the application's "front end", receiving client connections, passing requests to the application server for processing, and sending responses back to clients.
   b. The application servers will be running Symplicity's SympleObjects e-Business Framework and its other COTS solutions.
   c. The application servers will contain the core business logic necessary for the operation of the system – the application servers will pass user queries to the RDBMS layer, process the result sets, and deliver them back to the web server layer for serving to clients.
   d. Symplicity is OS agnostic (can run on any OS) but it recommends employing RedHat Linux as the Operating System for its proven reliability, performance, and security.
   e. Symplicity recommends employing Apache, the world's most used web server. If Microsoft products are preferred, Symplicity has proven experience and expertise in deploying its solution in a Microsoft environment.

2. **RDBMS System**

   a. The RDBMS (database) server will form the core data container for all pieces of the system, including the dataset metadata database, as well as for the integrated datasets.
   b. Symplicity's software applications are database *agnostic* in that they work very well with all major database products including Oracle, DB2, Sybase, as well as popular open source databases such as MySQL.

3. **Integration Systems**

   a. This layer will allow the system to interact with external systems in a secure manner using industry standard "web services" interfaces wherever possible and supporting alternative integration mechanisms to maintain backwards compatibility with legacy systems.



The SympleObjects™ Framework

SympleObjects™

Symplicity has employed this proven open-architecture framework to build and deploy solutions for many of its clients, including the Department of Defense (DOD), Executive Office of the President (EOP), US Senate, the Department of the Treasury, and numerous others. This highly modular, component based application framework enables Symplicity to rapidly design and deploy complex database driven web based information systems, while ensuring high performance, scalability, and reliability.

The SympleObjects™ Framework is built on open standards and is highly customizable and configurable. The Framework provides Symplicity with the necessary building blocks to design and deploy information systems customized to client requirements in a fraction of the time typically required for custom built solutions. In essence, the Framework combines the benefits of customized and off-the-shelf systems into a single solution.

The SympleObjects™ Framework is comprised of four key layers:

Database Layer: SympleObjects™ supports all major databases to be employed as the persistent data store. As per requirements, Symplicity will employ MYSQL running on a supported hardware/OS platform for this project.

Application Objects Layer: Application objects are the heart of the SympleObjects™ Framework – they are custom defined for every information type in a system. A given object is defined by the set of information or *object attributes* that is carried for it.
Presentation Layer:

> Application Object Forms: Application Object forms enables the system to display forms for any *AppObject*, in either display or edit/manage mode. Objects may be edited from a web interface, or from a WAP based handheld device. *AppObjectForms* intelligently renders forms for any required interface incorporating user context and authorization.
>
> Application Object Lists: SympleObjects Application Object Lists allows for simple creation and management of lists of objects -- empowering system integrators with functionality that includes definition of list columns, searching, sorting and filtering (all subject to user context and authorization).

Application Components: At the highest level, the SympleObjects™ Framework provides a set of reusable *application components* which are typical in many complex information applications. Applicant Components are then used as building blocks of the larger solution. This high degree of modularity inherent to using these Application Components in building large systems, is a key factor in the speed of implementation, the high degree of flexibility, as well as in the scalability and performance of SympleObjects™ based solutions. Some of these components include:

1. Document Management
2. Project/Task Management
3. Contact Management
4. Calendaring
5. Collaborative Discussions
6. Email Lists
7. Authentication modules
8. Various Utility modules, including a mail merge capable email system, e-printing subsystem, and object flagging module.

The SympleObjects™ Framework, as described above will enable Symplicity to deploy a best-of-breed system.

Continuity of Operations Plan

Symplicity designs every solution to be highly available exceeding industry levels. Every aspect of our solution, Internet bandwidth, power, server provisioning, and backups utilizes a heartbeat that allows automatic fail over in the event that primary service fails.

Our entire solution is connected via diversified power sources entering the NOC at different physical locations. Additionally, each power source is served by six different uninterruptible power supplies (5+1) and two diesel generators (with 5 days of fuel on site).

Symplicity contracts bandwidth from Internap, which is the premier provider of Internet connectivity. Our solution has diversity inter-connect feeds from Internap, that is fed from IP Services from approximately 15 of the nations largest Internet providers (Qwest, ATT, Level 3).

Every critical server utilizes two different power supplies in an active/passive scenario. Additionally, network level reliability is achieved using a minimum of 4 ports connected to diversified Level 3 Managed Ethernet switches.

Finally, in the event that a server fails, another server is standing by as a passive fail over that will assume the identity of the primary within 10 seconds of a failed node.

Symplicity has vast experience in the deployment and on-going maintenance and support of web-based management information systems. Symplicity services more than 600 universities and numerous U.S. government agencies including NASA, Air Force, US Army, US Treasury, and more.

File Backups

Symplicity leverages the features of the IBRIX file system to ensure that customer data is replicated to two disparate storage nodes. In the event that a node fails, the impact is transparent to the user and data on the failed node is immediately copied from the working version to another storage node to ensure that the replication factor of 2 is maintained. Additionally, the IBRIX file system allows the system to route around storage hot-spots to achieve high performance applications by pulling data from the least loaded servers. Additionally, each storage node is connected with diversified fiberpath to an EMC Storage Area Network.

Full backups are taken weekly with incremental backups being completed every evening. Additionally, a binary log is used on database servers to restore if needed. Symplicity also has a power audit-trail / log at the application layer that allows intra-day restorations.

Nightly backups are stored on a separate SAN and can be pulled if user error or data loss occurs on site at the Equinix data center. Every evening all backups are copied from production (Ashburn, VA) to Symplicity's headquarters (Arlington, VA) via point to point gigabit Ethernet private line service provided solely for Symplicity by Verizon Business Solutions.

Denial of Service & Attacks

Symplicity's network operating status is monitored 24x7 by Symplicity Corporation and our network provider, Internap. Symplicity employs full host based IDS and IPS services and can filter out denial of service attacks. Anti-Virus software is also regularly updated with host scans performed daily.

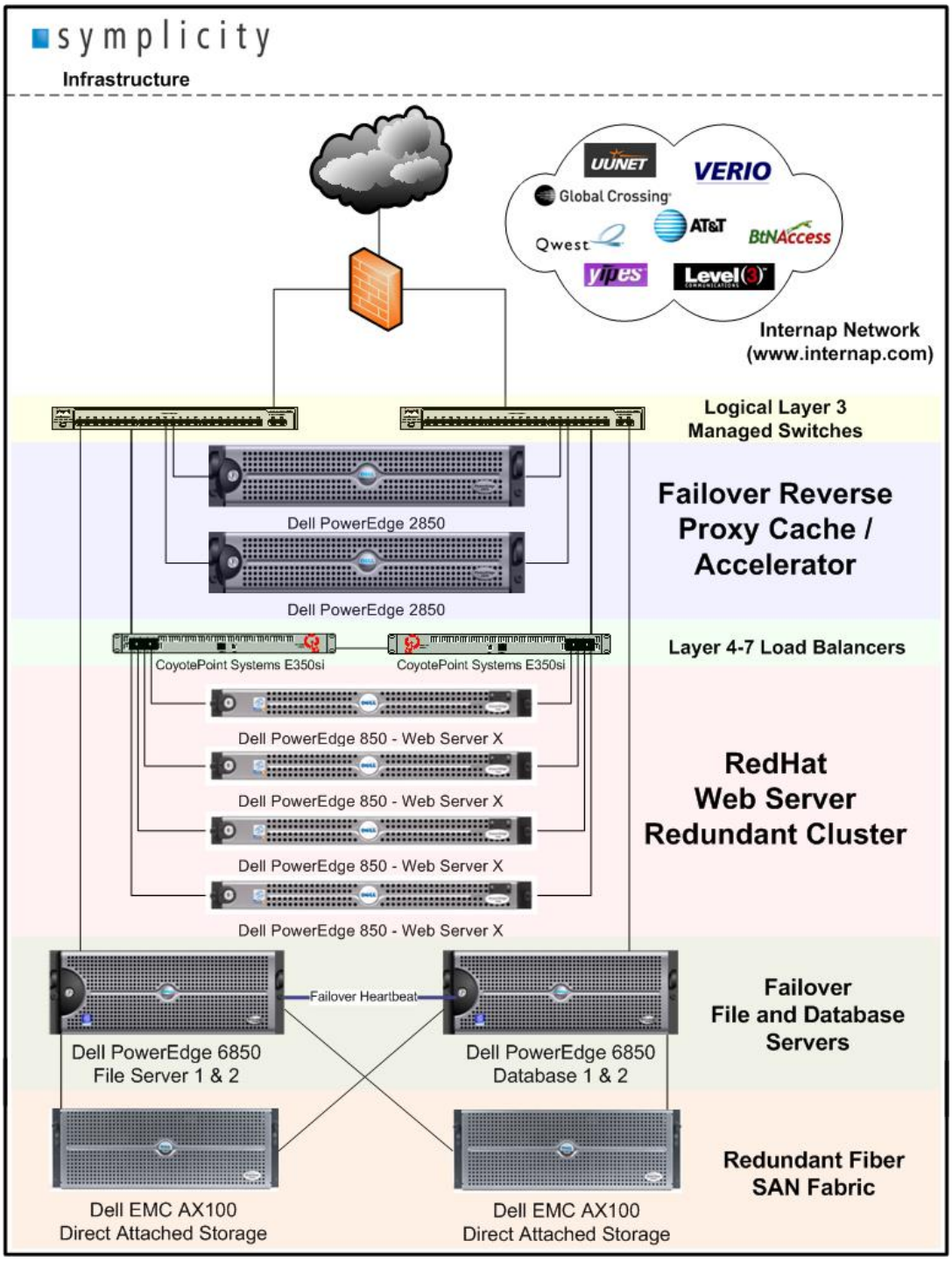
Hardware Architecture

Symplicity offers colleges and universities a reliable and secure hosting and content delivery services. Symplicity's proposed architecture includes a web application server "farm" with redundant servers in load-balanced configuration, using redundant firewalls and routers to eliminate single points of failure and serviced by a different power grid and different Internet backbone.

The proposed hardware multi-tiered architecture consists of multiple backend databases, file servers in a fail-over configuration. In the event that either server fails, the other will assume the functionality of the failed system. Delivery of the static page content will be delivered by a set of Network Appliances Reverse Proxy Caching Servers, to optimize server processes. Separation of the static and dynamic web content allows for dramatic speed increases in the overall application. Because of the light weight of the http protocol, and the relative size of images, static content can account for up to 90% of network traffic. Offloading this traffic to appliances specifically manufactured to deliver cached web content will free vital amounts of processing power on application servers.

Placed behind the caching servers are redundant load balancing switches. They maintain the state with each other and upon loss of communication, perform an automatic failover, causing no disruption to the end user. This allows for flexibility in managing servers in the data center. Servers can be taken offline fir required maintenance, and put back on-line, with no disruption to the production environment.

Logical Network Diagram



Note that there is redundancy at all levels within the hardware architecture. From the server level, within the servers themselves, power, all the way up to the network level. The servers have redundant network ports, redundant power supplies, redundant hard disk subsystems. There are multiple firewalls and load balancers. There are multiple geographic locations. No single component failure will cause disruption in service.

Backup and Disaster Recovery

The infrastructure services team performs daily incremental and full weekly backups for the solution at its primary location. The retention (rotation) period for the solution is one month with monthly snapshots retained for the past 3 years.

Backup and disaster recovery services are available only to Symplicity-hosted applications, not client-hosted applications.

System Capability

High performance for the solution is ensured through multiple mechanisms:

1. The software is designed using current best practices in object-oriented design which provides the highest possible application reliability and performance, while at the same time reducing development time, and increasing maintainability.

2. Intelligent caching techniques ensure that dynamic content is not needlessly generated multiple times. This significantly reduces load on the back end servers by reducing database queries, and reduces load on front end servers by reducing the number of pages that need to be rendered.

3. Load balancers and redundant multiple front and back and servers allow for distributing load amongst several servers, thus ensuring higher performance on a per system basis.

Scalability and Adaptability

Symplicity's architecture, both hardware and software, is designed to ensure "horizontal" and "vertical" scalability at all levels.

Horizontal scalability allows for the adding of additional front end servers, distributing load amongst them, reducing the load on each server, and thus increasing response time and capacity of the infrastructure. Scaling architecture horizontally has no practical limitations in terms of how broadly one can grow it – adding servers, and adding load balancers in front of them as necessary will allow the infrastructure to grow as requirements grow and evolve. Symplicity's architecture can be horizontally scaled simply by adding front end web/application servers and configuring the load balancers to recognize them.

Vertical scalability allows for increasing capacity by adding power to particular architecture components. Symplicity's architecture can be vertically scaled by adding RAM and/or processors to the back-end database servers and/or front end web/application servers.

Furthermore, Symplicity's solution leverages open standards wherever possible in order to assure future interoperability with evolving systems and requirements. Symplicity's architecture uses web services wherever possible to communicate between systems. This will allow a fast, efficient transition to a services oriented architecture where system modules can be exposed into external systems with minimal changes.

The system will have the capability to handle 500+ concurrent users at a minimum without a reduction in performance. The system will save data and move to the next activity in less then 6 (6) seconds, ninety-nine percent of the time. The system will generate reports from simple queries in less than twenty (20) seconds, ninety-nine percent of the time.

Interoperability

Symplicity has extensive experience in developing integration layers for separately managed systems on a variety of platforms, languages, and operating environments.

The integration API will provide generic object access methods for any objects for which the University Registrar office is the authoritative data source.

Standards Compliance

Symplicity's interim integration API shall adhere to Standard Transaction.

The web services API will meet or exceed the following specifications:

- SOAP 1.1
- WSDL 1.1
- ISO 11179 standard names, new names will be selected through cooperation of SME, "Functional Data Expert", and Software Developers -- per Summary XML Guidance #4
- Camel case for XML names
- Avoid acronyms
- No Abbreviations
- Use UPPERCASE acronyms
- Detailed XML comments wherever possible

Symplicity will fully document the database schema.

Integration Transports and Protocol

The primary mechanism of integrating systems will be through FTP, EMAIL, ODBC (existing legacy mechanisms) with an eye towards moving to a Web Services API, using SOAP over HTTP/S.