

Yacaré: Una experiencia de desarrollo colaborativo de software

Daniel Bollo (dbollo@psi.unc.edu.ar)

Resumen. Las Universidades Nacionales, a pesar de algunas diferencias en su tamaño, estructura, organización interna y procedimientos, tienen necesidades de información similares que se reflejan en requerimientos de sistemas de información análogos. Estas necesidades han sido tradicionalmente satisfechas por distintas vías, tales como la utilización de sistemas desarrollados por SIU, la incorporación de software desarrollado por terceros y aplicable a la realidad de las universidades, o el desarrollo a medida, ya sea con recursos propios o mediante contratación de terceros. El presente trabajo propone otro mecanismo, el desarrollo colaborativo entre instituciones, que a nuestro entender no ha sido suficientemente utilizado en el ámbito público.

Si bien la iniciativa concreta analizada aquí y sobre la que están trabajando cuatro universidades, es el desarrollo de un sistema de gestión de colegios preuniversitarios, no es objetivo de este trabajo su presentación. Este sistema sólo constituyó la oportunidad para establecer un marco de trabajo colaborativo, que consideramos innovador y que creemos que puede trasladarse a otros desarrollos en el marco de las universidades, o en el más amplio de organismos del Estado. En este sentido, vemos con agrado la iniciativa de Software Público encarada por la Subsecretaría de Tecnologías de Gestión de la Jefatura de Gabinete de Ministros.

Palabras Clave: Desarrollo, software, colaborativo, universidades

1 Introducción

Las Universidades Nacionales, dado su objetivo común y a pesar de algunas diferencias en su tamaño, estructura, organización interna y procedimientos, tienen necesidades de información similares que se reflejan en requerimientos de sistemas de información análogos. Estas necesidades han sido tradicionalmente satisfechas por distintas vías, tales como la utilización de sistemas desarrollados por SIU, la incorporación de software desarrollado por terceros y aplicable a la realidad de las universidades, o el desarrollo a medida, ya sea con recursos propios o mediante contratación de terceros.

El presente trabajo propone otro mecanismo, el desarrollo colaborativo entre instituciones, que a nuestro entender no ha sido suficientemente utilizado en el ámbito de las universidades, o incluso en el más amplio del Estado Nacional.

El SIU (Sistemas de Información Universitario) nació para proveer información homogénea y confiable del sistema universitario al gobierno nacional y luego derivó en el desarrollo de aplicaciones que resultaban imprescindibles y eran un paso previo para cumplir con ese cometido, ya que la obtención de información agregada, presupone la existencia de sistemas homogéneos que generen el insumo de esos informes.

Hoy día nos encontramos en las Universidades con una apreciable cantidad de sistemas desarrollados o facilitados por el SIU. La gestión de alumnos, presupuesto, finanzas, inventarios, etc. es viabilizada por sus sistemas, adoptados o en vías de implementación por la gran mayoría de las casas de estudio.

Pero estos sistemas cubren, como es de suponer en un esquema lógico de prioridades, las necesidades de información más relevantes, de mayor impacto y amplia cobertura, por lo que resta abarcar áreas con necesidades particulares o sectores que no están presentes en la mayoría de las universidades.

Desde hace poco tiempo, otra alternativa que se está abriendo para la obtención de soluciones de manera colaborativa, es la iniciativa de Software Público encarada por la Subsecretaría de Tecnologías de Gestión de la Jefatura de Gabinete de Ministros. La creación de un repositorio de proyectos, la infraestructura de coordinación y el esquema de licencias propuestos, es otro hito importante en la búsqueda de eficientizar y dar sinergia al desarrollo de sistemas comunes de la Administración Pública y Universidades.

Algunas de estas necesidades pueden satisfacerse con sistemas genéricos, es decir, pensados para resolver un problema, pero sin tener a las universidades como objetivo. En muchos casos, aunque no siempre, se trata de software libre. Podemos citar el caso de Koha, como sistema integrado de bibliotecas, o Comdoc, desarrollado en el Ministerio de Economía y usado en varias universidades para el seguimiento de expedientes.

Sin embargo hay casos en los que los requerimientos particulares conducen al desarrollo de sistemas a medida de una universidad. Este hecho, sumado a la asincronía en la definición de prioridades, atenta contra un crecimiento coordinado del parque de aplicaciones universitarias. Solicitudes urgentes de soluciones

informáticas para determinadas áreas, abandono o retraso de proyectos, financiamiento insuficiente para otros, delinean un mapa de información propio para cada casa de estudios, que se ve reflejado en esfuerzos repetidos en distintas universidades para el desarrollo de sistemas que cubren los mismos requerimientos, pero muchas veces incompatibles, dada la diversidad de arquitecturas y herramientas y recursos humanos con disímil capacitación y habilidades.

Creemos que esta situación conduce a un uso ineficiente de los recursos del Estado, y que es necesario desarrollar mecanismos orgánicos de colaboración que permitan el desarrollo de sistemas reutilizables en distintos ámbitos. Ese es el objetivo de este proyecto.

2 Proyecto Yacaré

2.1 Necesidades comunes.

Muchas universidades nacionales gestionan colegios preuniversitarios. Esos colegios suelen contar con sistemas de información ad-hoc, de complejidad variable. La Universidad Nacional de Córdoba (UNC) cuenta con dos colegios de este tipo: el Colegio Nacional de Monserrat y la Escuela Superior de Comercio Manuel Belgrano. Ambos cuentan con sistemas basados en tecnologías obsoletas, y que requieren su urgente reemplazo por sistemas más adecuados. La primera aproximación para resolver este problema consistió en identificar sistemas existentes en otras universidades. Este relevamiento mostró que había muchas universidades en la misma situación que la UNC, y sólo se encontró un sistema, el desarrollado por la Universidad Nacional de La Plata, que reunía los requisitos mínimos. Sin embargo, la utilización de este sistema creaba una dependencia de un equipo externo, lo cual ponía en duda la sustentabilidad del proyecto en el largo plazo.

Tampoco se consideró una solución adecuada el desarrollar un sistema propio para la UNC, por lo que, a la vista de que no era esta universidad la única en tener esta necesidad, se planteó a otras universidades la posibilidad de emprender un proyecto de desarrollo colaborativo.

Este tipo de desarrollo, impensable hace algunos años, se ha vuelto posible por la difusión de tecnologías de comunicación y colaboración: chat, videoconferencia, sistemas distribuidos de control de versiones, herramientas para desarrollo colaborativo de documentos, etc.

Si bien hubo muchas universidades interesadas, finalmente se constituyó un grupo de cuatro dispuestas a aportar recursos al proyecto

- Universidad Nacional de Córdoba
- Universidad Nacional del Litoral
- Universidad Nacional de Rosario
- Universidad Nacional del Sur

Cabe destacar que el objetivo de este trabajo no es presentar un sistema de gestión de colegios preuniversitarios, sino el marco de trabajo colaborativo, que consideramos innovador y creemos que puede trasladarse a otros desarrollos en el marco de las universidades, o en el más amplio de organismos del Estado.

Universidad Nacional de Córdoba - UNC

Fundación: 19 de junio de 1613

Facultades: 13

Carreras de Grado: 83

Alumnos: 102.766

Web:: <http://www.unc.edu.ar>

Universidad Nacional de Rosario - UNR

Fundación: 29 de noviembre de 1968

Facultades: 12

Carreras de Grado: 64

Alumnos: 76.006

Web:: <http://www.unr.edu.ar>

Universidad Nacional del Sur - UNS

Fundación: 5 de enero de 1956

Facultades: 17

Carreras de Grado: 61

Alumnos: 24.546

Web:: <http://www.uns.edu.ar>

Universidad Nacional del Litoral - UNL

Fundación: 17 de Octubre de 1919.

Facultades: 10

Carreras de Grado: 46

Alumnos: 28.450

Web:: <http://www.unl.edu.ar>

2.2 Solución

Tras una serie de operaciones de coordinación previas, el trabajo conjunto comenzó mediante una reunión presencial, realizada el 13 de diciembre de 2011 en la Universidad Nacional de Córdoba, y a la cual asistieron representantes de todas las universidades involucradas. En esa reunión se definieron la arquitectura, las herramientas a utilizar y los recursos que cada universidad aportaba.

La selección de la arquitectura y de las herramientas de desarrollo es extremadamente importante. Una herramienta tiene dos extremos: uno que se adapta al problema, y otro que se adapta al usuario de la herramienta. Es necesario elegir una herramienta que permita resolver el problema, pero que al mismo tiempo sea adecuada para quienes van a utilizarla, por lo que es relevante que el proceso de selección de las herramientas sea participativo.

Conviene distinguir entre dos tipos de herramientas: las seleccionadas para resolver el problema objetivo, es decir, el sistema de gestión de colegios preuniversitarios, por un lado, y las necesarias para gestionar el trabajo colaborativo en sí.

Con respecto al problema objetivo, se seleccionó un conjunto de herramientas basadas en Java, con GlassFish como servidor de aplicaciones, JPA e Hibernate como estándares de persistencia de objetos y Vaadin como framework de presentación. La base de datos elegida fue PostgreSQL, y se optó por Pentaho como herramienta de análisis y Business Intelligence. Una enunciación completa de las herramientas elegidas se detalla en la última sección de este trabajo.

Tal como se expresara más arriba, estas herramientas se seleccionaron en virtud de su aptitud para resolver el problema en concreto, y de su adecuación a las capacidades de cada universidad. Analicemos, por ejemplo, la decisión de utilizar un stack Java. Sin duda se trata de una alternativa adecuada para el tipo de problema a resolver, pero podrían haberse seleccionado otros lenguajes de programación o frameworks que también hubieran sido adecuados; por lo que los antecedentes de cada universidad en el uso de las distintas alternativas terminan constituyendo una parte muy importante de las razones que conducen a la decisión final. Una vez seleccionado el stack, se definieron, también mediante consenso entre los participantes, los componentes específicos a utilizar. Por ejemplo, en la selección del servidor de aplicaciones, fue relevante la experiencia de la Universidad Nacional del Litoral en GlassFish, experiencia que se transmitió a las otras universidades mediante una capacitación realizada durante el año 2012.

En lo referente a las herramientas tecnológicas para la gestión del trabajo colaborativo, se seleccionó Git como sistema de control de versiones, y Redmine como software de administración de proyectos. En ambos casos, la selección se realizó por consenso entre los desarrolladores, basado en experiencias previas con estas herramientas y otras de funcionalidades similares (particularmente Subversion y Trac). En ambos casos el software está desplegado en servidores de la Universidad Nacional de Córdoba, donde también se utiliza para otros proyectos, tanto internos como colaborativos. Por ejemplo, la UNC participa en el desarrollo del software de bibliotecas Koha, en el que se también se utiliza Git.

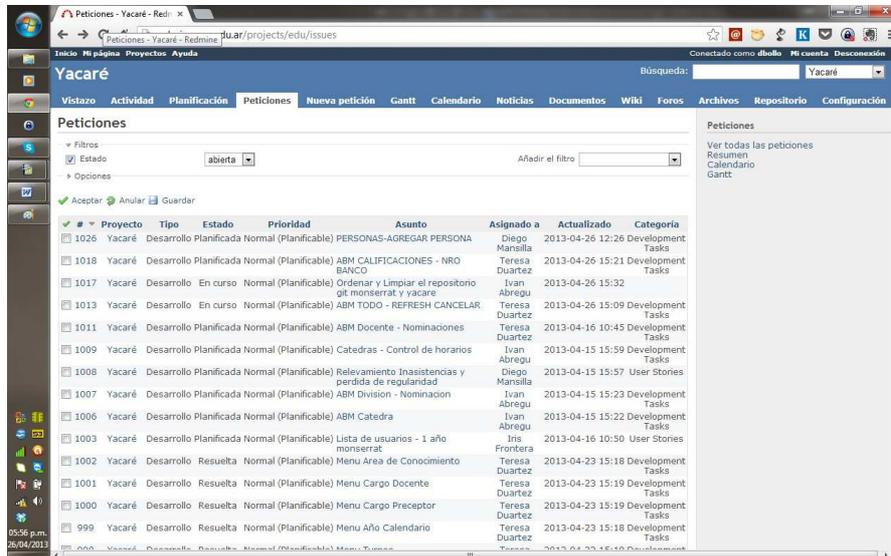


Fig. 1: Redmine como gestor de proyectos

Las mencionadas herramientas están orientadas al trabajo asíncrono. También se consideró conveniente un mecanismo de reuniones periódicas, las que se concretan semanalmente utilizando Skype como herramienta de videoconferencia. Estas reuniones abarcan dos aspectos, uno eminentemente técnico, por lo que convoca a analistas, arquitecto y programadores en la consideración de temas referentes a la construcción del sistema y otras reuniones congregan a usuarios claves del sistema en la discusión y acuerdo sobre modalidades y funciones a incluir. El siguiente cuadro presenta un resumen de los acuerdos logrados en estas reuniones, así como el estado de avance de cada universidad.

1. Relevamiento:

- Las cuatro universidades presentaron relevamiento y documentación completa de las instituciones a implementar.

2. Modelado:

- UNC: se encargó de estudiar y ajustar el modelo de datos de acuerdo a los datos relevados por cada universidad, el mismo se encuentra documentado. Además armó estructura de proyecto para comenzar a desarrollar prototipos de interfaz pero con la tecnología propuesta por Córdoba (Hibernate y Vaadin).

3. Tecnologías:

- JPA: UNR: se encarga de migrar las clases generadas por UNC a la tecnología propuesta JPA.

- EJB: UNL: tomó a cargo la migración a EJB, pero los recursos asignados al proyecto se encuentran bajo capacitación sobre dicha tecnología.
- Glassfish: UNL: Desarrolló un taller de capacitación para UNC, se presentaron los referentes en ambas universidades.
- Redmine: UNC: se encarga de la configuración y administración.

4. Fecha estimada de implementación:

- UNC: Marzo 2013
- UNL: Marzo 2014
- UNR: Agosto 2013
- UNS: Marzo 2013

2.3 Factibilidad del proyecto

Para analizar la viabilidad del modelo colaborativo que aquí proponemos, debemos compararlo con las otras alternativas. Es decir, asumimos que este modelo es aplicable cuando existe la necesidad de adquirir un sistema para resolver un problema común a varias instituciones y la ausencia de otras soluciones implica encarar un desarrollo. En ese caso ¿es una alternativa válida al desarrollo individual en cada una de las instituciones? ¿No son los costos de coordinación superiores a los de la duplicación de esfuerzo? Creemos firmemente que la respuesta es afirmativa en el primer caso y negativa en el segundo.

Desde el punto de vista de factibilidad técnica, la complejidad de un proyecto de este tipo no es esencialmente superior a la de un proyecto similar centrado en una única institución. En cualquier caso, el desarrollo de un sistema de mediana complejidad debe ser realizado por un equipo, y las herramientas actuales hacen que la dispersión geográfica del equipo no sea un problema severo.

Desde el punto de vista financiero, los costos de desarrollo totales no son sustancialmente superiores al desarrollo individual, y son considerablemente inferiores para cada una de las instituciones participantes, ya que solo deben afrontar una parte de los costos del proyecto.

Quizás el único aspecto problemático es la viabilidad desde el punto de vista de política organizacional. Un proyecto de este tipo requiere un compromiso institucional por parte de cada uno de los organismos participantes. Este tipo de compromiso exige que o bien las áreas (de informática o sistemas) involucradas, sea cual sea su denominación organizacional, tengan objetivos claros y autonomía suficiente para encarar un esfuerzo de este tipo, o bien que se involucren en el proceso autoridades con esas características.

Este último aspecto ha sido uno de los escollos con los que se ha tenido que luchar, tema que tratamos con mayor amplitud en el apartado de lecciones aprendidas.

Dada su factibilidad, un objetivo de este trabajo es animar a la asociatividad entre áreas universitarias de tecnología. Como decíamos al comienzo, el objetivo final sería que incorporemos, en el escenario de las decisiones habituales, no sólo los recursos propios, sino los del conjunto de universidades. En el camino a ese cambio cultural,

prácticas como la que proponemos, irán constituyendo hitos imprescindibles de aprendizaje, acostumbramiento y conocimiento mutuo.

Un insumo básico para la concreción y replicación de proyectos como este es la comunicación, objetivo hacia el cual conducen, además de formas tradicionales como simposios y jornadas, iniciativas más específicas que han nacido hace relativamente poco:

- Comité Técnico del Consorcio SIU, ámbito donde los responsables de las áreas de sistemas de información de todas las universidades nacionales se encuentran periódicamente en encuentros presenciales y a través de su lista de correo, extranet, video y teleconferencias y otras herramientas colaborativas y de comunicación.
- Tícar, encuentros de responsables TIC de universidades nacionales que ya lleva dos ediciones en Villa María (Córdoba) en 2012 y recientemente en Córdoba, en el mes de Abril de 2013, a la que concurrieron representantes del 90% de las Universidades Nacionales, además de otras instituciones relacionadas.
- Tical, encuentros que también abarcan a directivos de tecnología, en este caso, con un ámbito ampliado a las universidades de América Latina.

2.4 Alcance e impacto

Es de relevancia el beneficio que este proyecto aportará a las universidades nacionales, en la medida que se puedan establecer metodologías y procedimientos colaborativos que permitan aunar esfuerzos en el logro de soluciones cooperativas a problemas comunes. Más allá de estas herramientas, un paso muy importante sería que, a través de experiencias exitosas de este tipo, se genere en las áreas de tecnología de las universidades una cultura de trabajo que incorpore estas prácticas dentro de sus estrategias de desarrollo y adquisición de sistemas, de la misma manera que ha transformado este proceso de decisión la experiencia del consorcio SIU y la adopción de sus soluciones, luego de más de 15 años de existencia de este organismo.

Cambios culturales de esta envergadura, difíciles de lograr, luego de incorporados generan una inercia positiva que se refleja en decisiones cotidianas que incorporan como recursos propios los del conjunto.

El proyecto beneficiará, sin duda, a sus destinatarios directos: autoridades y estamentos docentes y de apoyo administrativo de las escuelas donde se implemente, y fundamentalmente, los nueve mil alumnos de los colegios pre-universitarios de las cuatro universidades que forman parte del proyecto. Esta cifra es sólo una primera instancia, ya que luego se proyectará a los alumnos de colegios de otras universidades que adopten el sistema en el futuro. Ya manifestaron su voluntad en ese aspecto, las universidades de Tucumán y San Luis, que, careciendo de recursos para sumarse al desarrollo, tienen interés en adoptar el sistema para sus escuelas una vez terminado. El sistema Yacaré se implementará en los siguientes colegios pre-universitarios:

Universidad Nacional de Córdoba

- Colegio Nacional de Monserrat: 1600 alumnos activos.
- Colegio Nacional Manuel Belgrano: 2000 alumnos activos

Universidad Nacional del Litoral

- Escuela Industrial Superior: 1016 alumnos activos.
- Escuela de Agricultura, Ganadería y Granja: 290 alumnos activos.

Universidad Nacional de Rosario

- Instituto Politécnico Superior: 1013 alumnos activos.
- Colegio Superior de Comercio: 1029 alumnos activos.
- Escuela Agrotécnica: 241 alumnos activos.

Universidad Nacional del Sur

- Escuela de Ciclo Básico Común: 781 alumnos activos.
- Agricultura y Ganadería “Adolfo J. Zabala”: 355 alumnos activos.
- Escuela Normal Superior “Vicente Fatone”: 321 alumnos activos.
- Escuela Superior de Comercio “Prudencio Cornejo”: 426 alumnos activos.

El alcance del sistema excede a las instituciones que beneficia, ya que con un uso más racional y eficiente de los recursos públicos, evitará la duplicación de esfuerzos y su consecuente gasto improductivo. Asimismo, el uso de aplicaciones y metodologías para la comunicación de los equipos de trabajo reducirán al mínimo sus necesidades de traslado y viajes.

Otro beneficio será la despapelización que generará en los colegios el uso de herramientas informáticas que reemplazarán planillas, libretas, etc. y que también evitarán traslados, en algunos casos, al facilitar la comunicación de docentes y alumnos a distancia.

La reproducción de iniciativas como la presentada no sólo es deseable sino necesario y en su replicación es secundaria la necesidad de recursos, siendo las actitudes su componente fundamental.

2.5 Hardware y Software

El hardware y software requeridos para el proyecto no difieren de los necesarios para cualquier proyecto de desarrollo de software. Las características colaborativas no agregan requerimientos fundamentalmente diferentes: todo proyecto de desarrollo de software requiere sistemas de control de versiones, de gestión de proyectos, de seguimiento de incidentes. El repositorio central se encuentra alojado en la UNC, pero podría estar alojado en instalaciones de cualquiera de los miembros del proyecto. Los

servidores están virtualizados, pero esto es consecuencia de la infraestructura existente y no es un requerimiento del sistema.

En lo que hace específicamente al software, se optó por un conjunto de herramientas destinadas a resolver distintos aspectos del problema:

- **JAVA:** es un lenguaje de programación orientado a objetos, desarrollado por Sun Microsystems <http://www.java.com/es/download/faq/whatis_java.xml>

- **JPA** (estándar de persistencia): permiten mapear una clase Java (Bean) con una tabla de la base de datos y facilitan mucho el trabajar con la persistencia de los objetos (usando metodos del estilo a “select”, “insert”, “update” y “delete”). <<http://www.oracle.com/technetwork/articles/javaee/jpa-137156.html>>

- **Drools** (motor de reglas) : es un sistema de gestión de reglas de negocio (BRMS, por las siglas en inglés de business rule management system) con un motor de reglas basado en inferencia de encadenamiento hacia adelante (forward chaining), más correctamente conocido como sistema de reglas de producción, usando una implementación avanzada del algoritmo Rete. Es software libre distribuido según los términos de la licencia Apache. <<http://www.jboss.org/drools/>>

- **Hibernate** (estándar de persistencia): es una herramienta de Mapeo objeto-relacional (ORM) para la plataforma Java (y disponible también para .Net con el nombre deNHibernate) que facilita el mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicación, mediante archivos declarativos (XML) o anotaciones en los beans de las entidades que permiten establecer estas relaciones. Hibernate es software libre, distribuido bajo los términos de la licencia GNU LGPL. <<http://www.hibernate.org/>>

- **EJB** (componentes Java de la capa de negocio): Los Enterprise JavaBeans (también conocidos por sus siglas EJB) son una de las API que forman parte del estándar de construcción de aplicaciones empresariales J2EE (ahora JEE 6.0) de Oracle Corporation (inicialmente desarrollado por Sun Microsystems). Su especificación detalla cómo los servidores de aplicaciones proveen objetos desde el lado del servidor que son, precisamente, los EJB:

- Comunicación remota utilizando CORBA
- Transacciones
- Control de la concurrencia
- Eventos utilizando JMS (Java messaging service)
- Servicios de nombres y de directorio
- Seguridad
- Ubicación de componentes en un servidor de aplicaciones.

La especificación de EJB define los papeles jugados por el contenedor de EJB y los EJB, además de disponer los EJB en un contenedor. <<http://www.oracle.com/technetwork/articles/javase/index-138213.html>>

- **Vaadin** (framework de presentación, con posibilidad de AJAX) es un código abierto de web de marco de aplicación para las aplicaciones de Internet. A diferencia de JavaScript cuenta con una arquitectura de servidor, lo que significa que la mayoría de la lógica se ejecuta en los servidores. Ajax se utiliza la tecnología en el navegador de un lado a garantizar una experiencia de usuario e interactiva. En el lado del cliente Vaadin se basa en la parte superior y se puede ampliar con Google Web Toolkit. <[https:// vaadin.com/learn](https://vaadin.com/learn)>
- **Maven** (framework de despliegue de aplicaciones) es una herramienta de software para la gestión y construcción de proyectos Java. <<http://maven.apache.org/>>
- **PostgreSQL** (Base de Datos) : es un sistema de gestión de base de datos relacional orientada a objetos y libre, publicado bajo la licencia BSD. <<http://www.postgresql.org/about/>>
- **Redmine** (herramienta para la gestión de proyectos): es una herramienta para la gestión de proyectos que incluye un sistema de seguimiento de incidentes con seguimiento de errores. Otras herramientas que incluye son calendario de actividades, diagramas de Gantt para la representación visual de la línea del tiempo de los proyectos, wiki, foro, visor del repositorio de control de versiones, RSS, control de flujo de trabajo basado en roles, integración con correo electrónico, etcétera. <<http://www.redmine.org/>>
- **Git**: repositorio de código fuente y sistema distribuido de control de versiones. <<http://git-scm.com/>>
- **IReports**: herramienta de diseño de generación de reportes. <<http://jasperforge.org/projects/ireport>>
- **JasperReports**: es una herramienta de creación de informes que tiene la habilidad de entregar contenido enriquecido al monitor, a la impresora o a ficheros PDF, HTML, XLS, CSV y XML. <<http://jasperforge.org/projects/jasperreports>>
- **Spring Security** (ex Acegi) (Framework de seguridad) es un framework que proporciona servicios de seguridad para aplicaciones J2EE. Está basado íntegramente en Spring Framework, por lo que se recomienda estar familiarizado con las principales características del mismo (inyección de dependencias, AOP, etc). <<http://static.springsource.org/spring-security/site/>>
- **Pentaho**: es un conjunto de programas libres para generar inteligencia empresarial (Business Intelligence). Incluye herramientas integradas para generar informes, minería de datos, ETL, etc. La plataforma Open Source Business Intelligence cubre muy amplias necesidades de Análisis de los Datos y de los Informes empresariales. <<http://www.pentaho.com/>>
- **Alfresco**: es un sistema de administración de contenidos libre, basado en estándares abiertos y de escala empresarial para sistemas operativos tipo Unix y Otros <<http://www.alfresco.com/es/products/document-management/>>
- **GlassFish** es un servidor de aplicaciones de software libre desarrollado por

Sun Microsystems, compañía adquirida por Oracle Corporation, que implementa las tecnologías definidas en la plataforma Java EE y permite ejecutar aplicaciones que siguen esta especificación. La versión comercial es denominada Oracle GlassFish Enterprise Server (antes Sun GlassFish Enterprise Server) . Es gratuito y de código libre, se distribuye bajo un licenciamiento dual a través de la licencia CDDL y la GNU GPL.<<http://glassfish.java.net/>>

3 Lecciones aprendidas

A la fecha, Abril de 2013, Yacaré se encuentra implementado parcial y gradualmente en las Universidades Nacionales de Córdoba y del Sur, mientras que Rosario y Litoral esperan hacerlo en el curso de 2013 y Marzo de 2014 respectivamente.

De la experiencia acumulada hasta el momento, surgen los siguientes puntos relevantes:

- Los aspectos metodológicos y técnicos, a pesar del desafío que impone un equipo de desarrollo disperso geográficamente, son relativamente similares al de un proyecto estándar encarado localmente, aunque la distancia entre integrantes y usuarios de cada institución, exige hacer hincapié sobre dos puntos :
 - Se deben afinar las comunicaciones, manteniendo el ritmo de las mismas, algo que a veces no es tan fácil sostener.
 - La figura de líder de proyecto cobra una mayor dimensión, ya que su papel de coordinación se vuelve imprescindible en un grupo disperso, con diferentes responsabilidades, asignación de tiempo y distintas aptitudes. En el caso de Yacaré, ese rol fue asumido desde Córdoba luego de una primera etapa donde la inexistencia de ese liderazgo produjo un vacío que ralentizó y distorsionó el curso del desarrollo
- La asincronía en la percepción de sus necesidades por parte de cada institución participante y por ende, su fijación de prioridades, es el problema que subyace en la mayoría de los inconvenientes con los que se ha topado este proyecto. Como ocurriera anteriormente con otros sistemas, cada institución percibe la necesidad de soluciones en diferentes tiempos y con distintos niveles. Este fenómeno ha sido diagnosticado por el SIU hace tiempo como una de las principales dificultades en su trabajo de coordinación en la implementación de sus sistemas. La prioridad en la obtención de una solución es el acelerador del aporte de recursos humanos y técnicos a los proyectos y ha sido la razón de la asignación dispar de recursos por parte de las universidades participantes. Este último aspecto, dada su relevancia, debe ser encarado firmemente desde un principio, generando compromisos irreversibles en los máximos niveles de decisión.

4 Conclusión

Las universidades hacen de la práctica colaborativa un valor importante en el desarrollo de sus actividades primarias. En docencia, extensión e investigación, permanentemente vemos manifestaciones e intervenciones colectivas, intercambios de alumnos, docentes e investigadores, equipos interdisciplinarios llevando a cabo proyectos, programas conjuntos de becas y subsidios, etc.

En ámbitos propios de la tecnología, también conocemos organismos donde la colaboración es el sentido mismo de su existencia, como las comunidades de software libre y otros, donde la eficiencia y descentralización, como el caso de grandes empresas de software y hardware, exige el trabajo cooperativo de áreas o divisiones, a veces también distribuidas geográficamente. Es por ello que el ambiente colaborativo no es extraño a las áreas de sistemas de las universidades.

Sin embargo, esta iniciativa tiene carácter innovador al no conocerse proyectos con las características del que proponemos. Seguramente sería motivo de un estudio de otro tipo el reconocer las razones de nuestro habitual comportamiento endogámico, el que podría ayudar, a partir de tal conocimiento, a generar líneas de acción que rompan o al menos debiliten las barreras que existen, para, de ese modo, lograr una actividad más abierta y abarcativa de nuestras áreas de tecnología.

Ante comunión de objetivos, necesidades y restricciones, debería surgir naturalmente la colaboración