

Automatización en el desarrollo de Software Crítico en el Ámbito Científico – Técnico

Alicia Salamon, Patricio Maller, Alejandra Boggio, Natalia Mira, Sofia Perez, Francisco Coenda.

Departamento de Informática, Instituto Universitario Aeronáutico.
Av. Fuerza Aérea 6500, Córdoba, Argentina. Teléfono: 0351 - 4435000
{as.salamon, pmaller, alejandra.boggio, ncmira, sofiabeatrizperez,
franciscocoenda}@gmail.com

Resumen

El presente proyecto se desenvuelve en el marco del proyecto PIDDEF 42/11 en el Departamento de Informática de la Facultad de Ingeniería del Instituto Universitario Aeronáutico (IUA), titulado “Metodología y Framework de Gestión de Líneas Base de Integración de Aplicabilidad en el Desarrollo de Software para el Proyecto UAV”

El proyecto contiene tres etapas claves para el proceso de desarrollo del mismo. Una primera etapa enfocada a la comprensión del entorno en que los científicos técnicos desarrollan software, la segunda al desarrollo e implementación de la arquitectura propuesta guiada por la integración continua para este ámbito particular. Y la última etapa referida al establecimiento de roles, actividades, pautas de trabajo para equipos de desarrollo de software científico-técnico de acuerdo a la arquitectura planteada.

De esta manera se intenta mejorar la calidad de los componentes de software desarrollados por los científicos técnicos.

Palabras clave: integración continua, software científico técnico, calidad de software.

Contexto

El proyecto PIDDEF 42/11, llamado “Metodología y Framework de Gestión de Líneas Base de Integración de Aplicabilidad en el Desarrollo de Software para el Proyecto UAV” se desarrolla en el Departamento de Informática de la Facultad de Ingeniería del IUA, Córdoba. El mismo recibe su financiamiento del Ministerio de Defensa apoyando sus áreas de interés para la defensa.

Este es un proyecto de investigación aplicado en el área de Calidad de Software, especialmente en referencia a la Integración y Gestión de Configuraciones de Software. El objetivo es hacer más eficiente la gestión en cuanto a la etapa de integración en sistemas con múltiples componentes desarrollados por equipos separados (aun geográficamente) y con alta criticidad en cuanto a calidad y respuesta.

El proyecto de gestión de líneas base de integración se piensa como una metodología integral que efficientiza y formaliza el flujo de aprobación y cualificación necesario para que un componente de software alcance la etapa de integración. Estos controles y aprobaciones son instrumentales para no trasladar costos de retrabajo hasta la etapa de integración, donde en grandes sistemas

se presenta más de un orden de magnitud en costos de espera y comunicación.

El proyecto tiene como objetivo entregar una estructura de procesos y prototipo de sistema completamente funcional con todos los mecanismos y con calidad de producto.

Introducción

El software crítico cada vez tiene mayor presencia en nuestro entorno, como aviones, radares, centrales nucleares, etc., siendo necesaria su fiabilidad y seguridad. Sin embargo, hasta en los sistemas más costosos, ampliamente probados y certificados pueden ocurrir fallos, y esto se debe a que estos sistemas son grandes y contienen operaciones en tiempo real, algoritmos complejos, numerosas interacciones, etc.

Esta dificultad de probar la ausencia total de errores en los sistemas críticos seguirá vigente mientras no se modifiquen algunas prácticas, hay que fomentar la reutilización de los componentes de software ya probados en otros proyectos reduciendo así la presencia de fallos.

No obstante, una de las respuestas de la ingeniería de software es la aplicación de un proceso de integración continua que aporta sistematización a todo el ciclo de vida del producto, contribuyendo en el descubrimiento de errores en etapas tempranas del proceso de desarrollo de software [1].

Por otro lado, los desarrolladores de software científico y técnico no cuentan con conocimientos profundos sobre la ingeniería de software, lo que conlleva a que carezca del soporte o fundamentos necesario para implementar las buenas prácticas de esta disciplina. [2, 3].

La integración continua es una práctica que comienza con la organización de los proyectos en una estructura de directorios

adecuada para establecer el orden de compilación y ejecución de los componentes de un proyecto (incluyendo casos de prueba), y de esta manera facilitar la construcción correcta del software cuando se ejecuta el proceso de integración, logrando que el mismo sea transparente para el equipo de desarrollo. Según Martin Fowler, muchos equipos de desarrollo han encontrado que este enfoque reduce significativamente los problemas de integración y permite que los equipos desarrollen software cohesivo más rápido [2].

Líneas de Investigación, Desarrollo e Innovación

La línea de investigación y desarrollo en el cual está sustentado el proyecto es la Calidad de Software, el mismo se compone de tres etapas que se detallan a continuación:

Dominio del ámbito científico técnico en el desarrollo de software.

En esta etapa del proyecto se busca comprender la realidad de los desarrolladores de software científico técnico en el cual se realizaron varias experiencias que contribuyeron a la elicitación de características que resultan críticas en su actividad relacionado a la conformación de los grupos de trabajo, áreas críticas de desarrollo, herramientas de software que utilizan, prácticas de desarrollo que llevan a cabo, normas de calidad en el cual desarrollan sus productos.

En dichas intervenciones se han aplicado distintas técnicas como repertory grid, multivounting, mapas mentales, encuestas para realizar la actividad de descubrimiento y caracterización del ámbito. [4-6]

Arquitectura basada en la Integración Continua

En la segunda etapa se investiga y se desarrolla la arquitectura propuesta guiada por la integración continua, utilizando técnicas y patrones de pruebas. Se ha investigado y seleccionado los componentes de software open source, que mejor se integran a la arquitectura de acuerdo a las características del ámbito científico técnico relevadas.

Los componentes que se analizaron hasta el momento fueron: sistema operativo, repositorio, sistema de control de versión, sistema de integración continua, gestor de incidencias.

La integración continua es una práctica que comienza con la organización de los proyectos en una estructura de directorios (repositorio y sistema de control de versión) adecuada para establecer el orden de compilación y ejecución de los componentes de un proyecto, incluyendo casos de prueba, y de esta manera facilitar la construcción correcta del software cuando se ejecuta el proceso de integración (servidor de integración), logrando que el mismo sea transparente para el equipo de desarrollo. [7-9]

Metodología de trabajo

En la última etapa se establecen los roles, actividades, pautas de trabajo para equipos de desarrollo de software científico-técnico, dentro de un modelo de referencia como la arquitectura elaborada en este proyecto diseñado para facilitar la integración continua. Cabe destacar que el ámbito científico técnico es muy distinto al ámbito comercial, ya que el primero es un ámbito muy específico y normalmente entendido por aquellos que se desenvuelven en él.

Resultados y Objetivos

De acuerdo a las etapas del proyecto de calidad de software se obtuvieron los siguientes resultados.

Con respecto a la primera etapa del proyecto de investigación, los resultados obtenidos en las distintas experiencias realizadas se vieron características diferenciales entre el desarrollo científico técnico y el comercial desde los miembros interdisciplinarios como la complejidad de sus productos de software. Estas particularidades están dadas por: comunicación informal, modo en llevar las pruebas de software, retrabajo en los componentes de software, escasos indicadores de control de los avances de los proyectos, entre otros.

En cuanto al desarrollo de la arquitectura propuesta se ha adoptado un enfoque basado en el aprovechamiento de los recursos y en la usabilidad de la misma. La arquitectura incorpora diversas definiciones metodológicas que se ven reflejadas en las herramientas seleccionadas luego de una ardua investigación. Los componentes que integran la arquitectura fueron configurados logrando una adecuada comunicación entre sí.

La arquitectura que se está construyendo muestra oportunidades que agilizan la forma de trabajo y llevan la mejora de procesos a los desarrollos de software. La utilización de prácticas innovadoras como la integración continua permite la automatización de actividades como la de construcción, ejecución de escenarios, gestión de incidencias, envío de notificaciones de los resultados a los responsables en tiempo y forma abre oportunidades más allá del ámbito de este proyecto.

Las mejoras operativas se han realizado de manera escalonada implementando la difusión de los estándares de líneas base, uso de un

repositorio, servidor de integración, gestión de incidencias para finalmente alcanzar una gestión integrada.

Por último se han definido los roles involucrados en un proceso de desarrollo de software que mejor se adapta al tipo de proyectos que el equipo de desarrollo de software científico técnico.

Los trabajos futuros estarán orientadas a una integración de las distintas fuentes de datos y a la validación y verificación de sistemas críticos.

Formación de Recursos Humanos

El equipo de investigación pertenece al Departamento de Informática de la Facultad de Ingeniería y se desempeña en la línea de investigación Calidad de Software conformada en el año 2010 desarrollando proyectos financiados por el IUA y el Ministerio de Defensa. También se han presentado producciones científicas en diferentes congresos nacionales e internacionales. [4-9].

El grupo está integrado por la directora del proyecto Alicia Salamon, codirector Patricio Maller, y las docentes Natalia Mira y Alejandra Boggio. Sumado a este equipo se integran dos becarios Francisco Coenda y Sofía Pérez.

Ambos becarios han realizado su tesis de grado de Ingeniería de Sistemas relacionadas a las líneas de investigación del proyecto. Una de ellas aportando las líneas bases de desarrollo y la segunda acerca de una metodología para identificar las áreas críticas en equipos de desarrolladores de software científico técnico.

A continuación se realiza una breve descripción académica de cada uno de los integrantes:

Alicia G. Salamon es Licenciatura en Sistema en el Centro de Altos Estudios en

Ciencias Exactas (1990), Buenos Aires, y Magister en Planificación y Gestión (2003) en Diego Portales de Chile.

Patricio Maller es Licenciado en Ciencias de la Computación (1995) en la Universidad Nacional del Sur y Magister en Ciencias de la Computación de la Universidad de Alabama (2000).

Natalia Mira es Licenciada en Informática (1997) de la Universidad Blas Pascal, Córdoba, y es estudiante de Magister en Ingeniería de Software en la Universidad Nacional de la Plata, Facultad de Informática.

María Alejandra Boggio es Ingeniería en Sistema (2003) en el IUA.

Francisco J. Coenda es Ingeniero en Sistema (2011) en el IUA- Córdoba, Argentina y es estudiante del Posgrado de Seguridad Informática en el IUA.

Sofía Beatriz Pérez es Ingeniera de Sistemas (2013) en el IUA- Córdoba, Argentina y es un estudiante de Maestría en Educación en Ciencias Experimentales y Tecnología en la Facultad de Ciencias Exactas, Físicas y Naturales en la Universidad Nacional de Córdoba, Argentina.

Referencias

- [1] D. Spinellis, "Software Builders", *IEEE Software* vol. 25, pp. 22-23, May/Jun 2008.
- [2] M. Fowler, "Continuous Integration" [online]. Disponible en: <http://martinfowler.com/articles/continuousIntegration.html>
- [3] D. Kelly y R. Sanders. "Assessing the Quality of Scientific Software", *First International Workshop on Software Engineering for Computational Science and Engineering*. Leipzig, Germany, 2008.

- [4] A. Salamon, P. Maller, A. Boggio, N. Mira, S. Perez, F. Coenda. “New Practices to Structure and Elicit Information in Scientific Software Development in Teams”, *The 12th International Conference on Computational Science and Its Applications (ICCSA)*, Bahia, Brasil 2012.
- [5] A. Salamon, P. Maller, N. Mira, A. Boggio, J. Giró, J. Cuozzo, F. Coenda, S. Pérez. “Metodología Para Identificar Problemas en Equipos que Desarrollan Software Científico-Técnico”, *ARGERCON*, Córdoba, Argentina, 2012.
- [6] A. Salamon, P. Maller, A. Boggio, N. Mira, S. Perez, F. Coenda, “Aplicación de mapa cognitivo compartido en equipos de desarrollo de software científico técnico”, *2do Congreso Nacional de Ingeniería Informática / Sistemas de Información (CONAISSI)*, San Luis, Argentina, 2014.
- [7] A. Salamon, P. Maller, A. Boggio, N. Mira, S. Perez, F. Coenda, “La Integración Continua Aplicada en el Desarrollo de Software en el Ámbito Científico – Técnico”, *XX Congreso Argentino de Ciencias de la Computación (CACIC)*, La Matanza, Buenos Aires, Argentina, 2014.
- [8] A. Salamon, P. Maller, A. Boggio, N. Mira, S. Perez, F. Coenda, “Testing en el Desarrollo de Software Científico en el Marco de la Integración Continua” *2do Congreso Nacional de Ingeniería Informática / Sistemas de Información (CONAISSI)*, San Luis, Argentina, 2014.
- [9] A. Salamon, P. Maller, A. Boggio, N. Mira, S. Perez, F. Coenda, “Automatización de la generación de indicadores para el seguimiento de proyectos”, *Congreso Argentino de Ingeniería (CADI)*, San Miguel de Tucumán, Argentina, 2014.