

# Generalizando un modelo de desarrollo de ecosistemas de software

**MSc. Jessie Castell González**

Universidad de las Ciencias Informáticas, La Habana, Cuba, [jcastell@uci.cu](mailto:jcastell@uci.cu)

**DrC Pedro Yobanis Piñero Pérez**

Universidad de las Ciencias Informáticas, La Habana, Cuba, [ppp@uci.cu](mailto:ppp@uci.cu)

## ABSTRACT

*Software ecosystems represent an evolution from enterprise architectures and software product lines. Based on the combination of their features and benefits provide a more inter-organizational approach to improving productivity, market competitiveness, encouraging reuse and sustainability of the products and the ecosystem as a separate entity. Although in several papers reviewed to date have addressed issues concerning the definition and analysis of software ecosystems, it is considered that there is still no formalized approach to the design and development of software ecosystems. Researchers only define certain elements and their relationships within the ecosystem, as well as best practices for further development. In particular, this paper proposed a generic model that suggests organizing the development of a software ecosystem based on certain assumptions, bases and processes that are presented. The proposed process propose input and output artefacts, as well as a detailed description that guide ecosystem development from conception to post implementation and deployment. Given the particularities of a software ecosystem for such initiatives, much more comprehensive and complex, combining the best practices of software development and takes the best current applied and valued international trends as well. Finally is briefly shown the application of the model in a particular environment.*

**Keywords:** *competitiveness; productivity; reusability; software ecosystems; sustainability.*

## RESUMEN

Los ecosistemas de software representan una evolución desde las arquitecturas empresariales y las líneas de productos de software. Partiendo de la conjunción de sus características y beneficios principales ofrecen un enfoque más inter-organizacional que mejora la productividad, la competitividad en el mercado, fomentando la reutilización y la sustentabilidad de los productos y del ecosistema como un ente propio. A pesar de que en diversos trabajos revisados hasta la fecha se han abordado aspectos sobre la definición y el análisis de los ecosistemas de software, se considera que aún no existe un acercamiento formalizado para la concepción y desarrollo de ecosistemas de software. Los investigadores solo definen ciertos elementos y sus relaciones dentro del ecosistema, así como buenas prácticas para un posterior desarrollo. En particular, en el presente artículo se presenta una propuesta genérica de modelo que sugiere organizar el desarrollo de un ecosistema de software a partir de ciertas premisas, bases y procesos. Los procesos propuestos proponen artefactos de entrada y salida, así como una detallada descripción que guía el desarrollo del ecosistema desde su concepción hasta posterior implementación y despliegue. Teniendo en cuenta las particularidades de un ecosistema de software se elabora dicha propuesta para este tipo de iniciativas, mucho más integral y compleja, que combina las mejores prácticas del desarrollo de software y toma las tendencias actuales mejor aplicadas y valoradas a nivel internacional. Finalmente se muestra de manera breve la aplicación del modelo en un entorno en particular.

**Palabras claves:** competitividad; ecosistemas de software; productividad; reutilización; sustentabilidad.

## 1. INTRODUCCIÓN

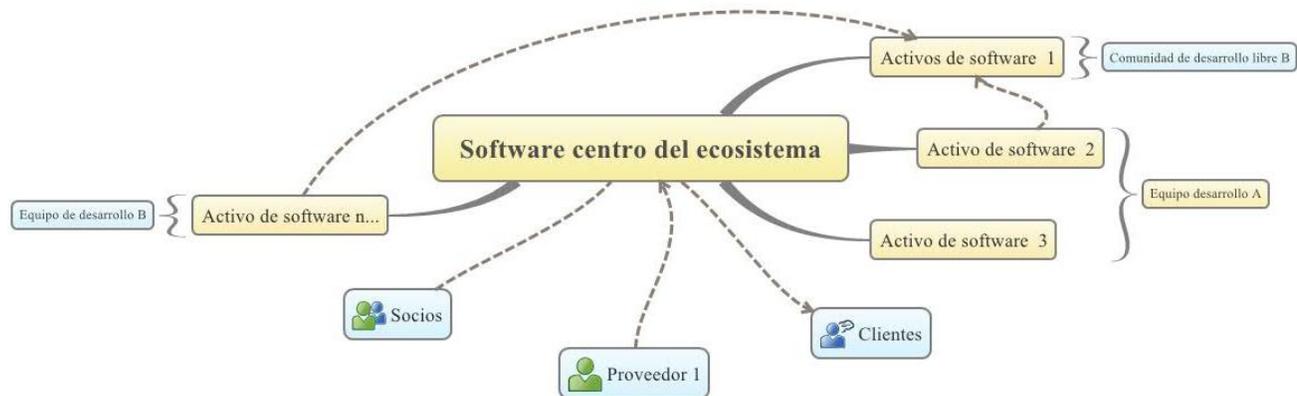
El término ecosistema proviene de la ecología, en la unión de elementos vivientes en un espacio vital donde juntos colaboran como un todo de manera simbiótica. A partir de este concepto han surgido varios tipos de ecosistemas, como es el caso de los ecosistemas humanos, comerciales, sociales y los denominados ecosistemas de software [Bosch 2009].

Los ecosistemas de software son una abstracción de los ecosistemas naturales, puesto que los sistemas que lo integran existen en un medio digital. La propuesta que se presenta en el artículo se centra en la definición y concepción de ecosistemas de software. Se entiende por un Ecosistema de Software (SECO, del inglés Software Ecosystem) como el conjunto de activos que interactúan para compartir un mercado común bajo un sistema ordenado de interrelaciones entre ellos [Frantz 2008]. Se define formalmente como el conjunto de soluciones de software que habilitan, automatizan y soportan cierta necesidad de negocio [Bosch 2009].

Los ecosistemas de software representan una evolución de las arquitecturas empresariales, las denominadas familias de productos de software [Sybren Deelstra 2005] y las líneas de producto de software [Castell 2012]. Varios autores sistematizan el estudio de lo que se considera una natural evolución en el desarrollo de software. Las principales causas o factores han sido la continua necesidad de alinear los procesos del negocio con las tecnologías, la casi urgencia de integración e interoperabilidad entre los sistemas, de manera que se responda con mayor eficiencia a las demandas del mercado y se ofrezca una solución cada día más competitiva.

En los diversos estudios realizados se ha podido constatar que los investigadores asumen desde enfoques diferentes los ecosistema de software. Algunos definen sus componentes y relaciones desde una perspectiva netamente comercial o de mercado, otros desde una perspectiva tecnológica (especifica en términos de integración de los componentes del ecosistema), aunque el enfoque mejor admitido es el que combina ambos enfoques. Se pueden clasificar como ecosistemas a MySQL-PHP, iPhone [Jansen y colectivo de autores 2009], SAP [Heilig y Karch 2008], Google [Jansen 2013], Eclipse [Angeren y colectivo de autores 2011], entre otros.

Existen varios criterios, clasificaciones o taxonomías de los ecosistemas de software, atendiendo a la plataforma en la que se desarrollan, el tipo de enfoque de negocio que brindan, el objetivo del mismo, etc. [Bosch 2009]. Pueden ser ecosistemas de diversa complejidad, según los componentes que integre y las relaciones o conexiones entre estos.



**Figura 1: Principales componentes internos y externos de un ecosistema de software.**

En la Figura 1 se muestran algunos de los componentes del ecosistema, separados entre internos (amarillos) y externos (azules) con sus respectivas relaciones, aunque esta clasificación puede variar según el interés y modelo de negocios de la entidad donde se desarrolle el ecosistema. Diversos autores han abordado las

relaciones entre los componentes de un ecosistema de software [Kjetil Hanssen 2010; Popp 2010], sin embargo en el presente artículo no se han detallado puesto que depende del caso específico del ecosistema de software a construir y del tipo de enfoque que tenga el ecosistema.

Nótese que se supone siempre la existencia de un sistema (activo de software) centro del propio ecosistema, aunque existen algunos ecosistemas como el de Microsoft Office que no tienen un sistema base, sin embargo la propuesta hace un aporte en los tipos de ecosistemas que se desarrollan a partir de cierta necesidad de negocio de un sistema específicos, o ya sea de varios.

Se considera que entre las mayores motivaciones para concebir y desarrollar ecosistemas de software se encuentran la de aumentar significativamente la productividad, alcanzar una mayor robustez de cada uno de sus activos por separado y del ecosistema como un todo, así como la creación de nichos de mercado que propicie la diversidad [Huang 2010; Iansiti y Levien 2002]. Además de ser estas las claves principales se usan comúnmente además como variables para medir y evaluar de la capacidad y evolución de estos.

Existen otros muchos factores que sugieren la necesidad de este tipo de iniciativas, como es caso de personalizar soluciones que incluyan varios tipos de activos que por sí solo no ofrezcan valor en el mercado y que como parte de un ecosistema, en conjunto con las capacidades de otros activos ofrezcan una solución integral.

## 2. DEFINIENDO UN MODELO DE DESARROLLO PARA ECOSISTEMAS DE SOFTWARE

Se ha elaborado el presente modelo con la intención de organizar las etapas de definición y posterior desarrollo de un ecosistema, que intenta abarcar y generalizar las mejores prácticas, definiciones y tendencias, así como los aportes de algunos autores. En otras publicaciones realizadas [Castell 2012; Castell y Piñero 2013], se ha enfatizado que, a consideración de la autora, las investigaciones que en dicha área se han acometido, no completan un ciclo de desarrollo, ni fijan pautas para definir, conformar o evaluar un ecosistema. Las presentes investigaciones abordan aspectos importantes de manera aislada, sin conceptualizar o configurar un proceso de desarrollo propio para ecosistemas, uniendo los principales conceptos que la definen.

La autora se propone refinar una propuesta de modelo para el desarrollo de ecosistemas de software, que en sus inicios se contextualizara y aplicara a los entornos y soluciones de la gestión de proyectos [Castell 2012], véase en la Figura 2 los componentes del modelo. Su finalidad es generalizar una propuesta para diferentes tipos de entornos, que permita su adecuación ante varios tipos de soluciones, aunque se mantendrán los principios y bases del modelo.



**Figura 2: Componentes del modelo propuesto.**

### 2.1 PRINCIPIOS DEL MODELO.

El modelo supone la disposición de premisas o principios que deben considerarse para la aplicación del modelo, pudiendo tener relación directa con su aplicación exitosa o no. Cada instanciación determinará qué principios se ajustan y las posibles consecuencias de no adoptarlas completamente.

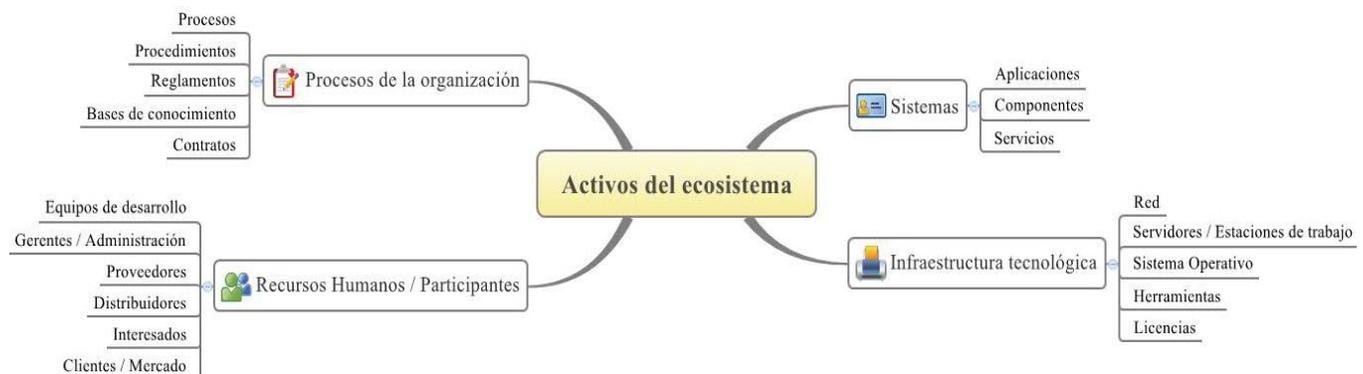
- *Compromiso institucional*: se parte del compromiso en varios niveles de la organización, departamentos y principales participantes que formarán parte del ecosistema.
- *Sustentabilidad*: Determina la característica del ecosistema a basarse en tecnologías y herramientas libres, mantiene la colaboración entre los participantes del ecosistema a través de comunidades abiertas.
- *Centrado en la mejora continua*: garantiza que el ecosistema evolucione y se actualice a sí mismo de manera que siempre represente una alternativa fiable, eficiente y competente en el mercado. Existen varios estudios que señalan este como una elemento indispensable a tener en cuenta en la concepción y desarrollo de los ecosistemas [Recena Soto y colectivo de autores 2012].
- Los activos dentro del modelo *se sustentan en la producción basada en líneas de productos de software*: se considera que los ecosistemas de software representan una evolución en cuanto a la manera que el proceso de concebir y desarrollar software se refiere. Los activos que formarán parte del ecosistema deben ser producto de diferentes líneas de productos que garanticen una eficiencia, una mayor especialización y calidad en los productos obtenidos.

## 2.2 BASES DEL MODELO.

El modelo se sustenta en ciertas bases, muy relacionadas también con los principios antes expuestos. Dichas bases determinan disposiciones importantes a medida que el ecosistema de software en cuestión se va desarrollando. Algunas de estas bases están muy relacionadas de igual forma con los procesos que se describen en la sección siguiente, sin embargo antes es necesario conceptualizarlas para un mejor entendimiento de los mismos.

### 2.2.1 ACTIVOS DE SOFTWARE. CRITERIOS PARA SU SELECCIÓN.

En diversas fuentes revisadas se definen términos como activos de software [Alvarez y colectivo de autores 2012; PMI 2008], activos de los procesos de la organización, sin embargo en la propuesta se abarcan además otros elementos posibles a concebir dentro de un ecosistema, véase la Figura 3.



**Figura 3: Activos del ecosistema propuestos.**

Para la selección final de los activos de software se recomienda tener en cuenta los siguientes criterios:

- *Extensibilidad*: garantiza que los sistemas puedan ser modificados y extendidas sus funcionalidades, por ello se recomienda que sean desarrolladas bajo los preceptos del software libre. Esto garantiza el principio de sustentabilidad.
- *Facilidad de mantenimiento*: deberán contar con un mantenimiento regular, o sea, tener soporte en el tiempo, gestionar su documentación y dar respuesta a solicitudes de cambio.
- *Comprensibilidad*: los activos deberán ser de fácil comprensión por los desarrolladores, el grado de asimilación deberá ser alto, la curva de aprendizaje lo más baja posible para poder garantizar además la extensibilidad y la facilidad de mantenimiento de los mismos.

- *Equilibrio entre costo - beneficio*: se deberá realizar un estudio del costo de cada activo y del beneficio potencial que ofrece al ecosistema como un todo para establecer un equilibrio entre ambas variables.
- *Mecanismos y facilidades para la integración*: evidentemente los activos deberán proveer facilidades para su integración con otros activos del ecosistema. En el siguiente sub-epígrafe se describen brevemente algunos los mecanismos más conocidos.
- *Posicionamiento en el mercado*: es un criterio que aunque parezca banal sería importante considerar que el posicionamiento y buenos resultados de la implantación de un activo puede a su vez darle prestigio y buen posicionamiento al ecosistema como un todo.

### 2.2.1 MECANISMOS DE INTEGRACIÓN. CRITERIOS PARA SU SELECCIÓN.

Todo tipo de iniciativas han sido desarrolladas según las diferentes opciones de integración que se tengan: transferencias de archivos y ficheros, las bases de datos compartidas, invocaciones a procedimientos remotos y mensajería a través de la publicación de mensajes asíncronos. [Hohpe 2012; Hohpe y Woolf 2003].

El modelo propuesto considera ciertos escenarios, diseñados a partir de los patrones de integración antes mencionados, que pueden seleccionarse atendiendo a los distintos enfoques, mecanismos y estilos de integración que cada activo de software pueda proveer.

1. Integración a nivel de ficheros: En este escenario se requieren mecanismos simples siguiendo el estilo de integración de transferencia de archivos y ficheros.
2. Integración a nivel de base de datos: escenario común entre sistemas de la misma organización o que han sido desarrolladas por el mismo equipo de proyecto, donde existe la comunicación y nivel de acceso y seguridad suficientes para permitir el acceso a las bases de datos.
3. Integración a nivel de servicios: Dentro de este escenario en particular se pueden darse varias modalidades o sub-escenarios, pues la integración a nivel de servicios, está relacionada con diferentes conceptos del estilo de mensajería: canales, mensajes, enrutamiento, etc.
  - a. Integración a nivel de servicios A - Conexiones punto a punto: Este escenario puede aplicarse cuando uno y otro sistema intercambian información directamente, ambos funcionan a su vez como servidor y receptor.
  - b. Integración a nivel de servicios B - Bus de Servicios Empresariales sin adopción de SOA: los sistemas se conectarán siguiendo el estilo de mensajería, pero las conexiones serán mediante un ESB. Los sistemas accederán a este para la publicación y consumo de los servicios mediante una infraestructura de seguridad adecuada.
  - c. Integración a nivel de servicios C - Bus de Servicios Empresariales con adopción de SOA: el uso de un ESB no incluye obligatoriamente la adopción de una SOA ni viceversa, sin embargo se considera que el uso de ambos amplía mucho más sus beneficios.

Adicionalmente a los escenarios que se han presentado se pueden concebir escenarios híbridos, que mezclen características de varios, de manera tal que se pueda garantizar la interoperabilidad acudiendo a múltiples mecanismos para ello, según el caso.

## 2.2 PROCESOS DEL MODELO.

Se ha diseñado un modelo para desarrollar un ecosistema de software, que transite de manera general por las fases o procesos que se muestran en la Figura 4 y se describen brevemente más adelante.



**Figura 4: Procesos del modelo.**

Es importante aclarar que a pesar de la presente es una propuesta más genérica, se considera que los ecosistemas de software deberán fijar sus bases y objetivos al comenzar a definirse. Si se revisan los ejemplos de ecosistemas antes mencionados, se puede constatar que cada uno tiene una finalidad y posiblemente un negocio específico, de lo cual dependerá el activo o los activos centrales del ecosistema, el resto de los activos y los participantes del mismo. Es lógico suponer de la premisa que un ecosistema no surge de la nada sino de una necesidad o requisitos para la integración con otros activos.

1. *Identificar los requisitos:* consiste en determinar el propósito general y los objetivos del ecosistema. Sobre todo los requisitos de información del o de los sistemas centro o núcleos del ecosistema. En dependencia de eso se pre-selecciona los activos de software a diagnosticar.
2. *Diagnosticar y seleccionar activos del ecosistema:* a partir del negocio que haya sido seleccionado se realiza un diagnóstico de los tipos de activos existentes en el ámbito de la organización y mercado, y atendiendo a los criterios para la su selección vistos en el apartado Bases, se seleccionan los más factibles para el ecosistema.
3. *Definir y seleccionar los participantes del ecosistema:* aunque los participantes pudieran considerarse un tipo de activo del ecosistema, en sí por participantes se entiende a las personas o grupos de personas que juegan un rol específico dentro del ecosistema, ya sea como proveedores, comercializadores, interesados, etc. Normalmente se les separa en varios niveles [Boucharas y colectivo de autores 2009; Jansen, Finkelstein y Brinkkemper 2009]. Se deberán definir en cada caso el tipo de relación de cada participante con el ecosistema, gestionarse contratos, convenios, colaboraciones, etc.
4. *Seleccionar escenarios de integración:* en la sección de Bases se describieron los diferentes escenarios que se podrían seleccionar a partir del diagnóstico realizado a cada activo. Se aconseja que los criterios para la selección de uno u otro escenario queden documentados para permitir la evolución futura del ecosistema así como para de manera general dejar explícitamente planteado los mecanismos de integración que permiten los activos.
5. *Diseñar el ecosistema:* es el proceso donde se define la arquitectura y los mecanismos para integrar a los activos de software dentro del ecosistema.
6. *Implantar el ecosistema:* durante este proceso, se realiza la implementación de la integración de los activos de acuerdo al escenario seleccionado (suponiendo que los activos estén previamente desarrollados y solo se reutilicen, de no ser así se realiza la integración y el desarrollo independiente de manera paralela, según convenga), se realizan pruebas, se documenta cada fase de la implementación del ecosistema y finalmente se despliega en diferentes entornos, siguiendo la buena práctica de despliegue continuo [Recena Soto, Muñiz, Gomez, Beas, Sutilo y Benitez 2012].
7. *Analizar los resultados:* un proceso vital para la evolución del ecosistema, es cuando se analizan los resultados tras la implantación del ecosistema tanto en entornos virtuales como reales, se identifican no conformidades, se detectan posibles conflictos con los participantes, así como el cumplimiento de los requisitos de información previstos y que originaron la construcción del ecosistema. A partir de este análisis se diseñarán nuevas fases de desarrollo o actualización del ecosistema.

### 2.3.1 ROLES PROPUESTOS PARA EL DESARROLLO DE UN ECOSISTEMAS DE SOFTWARE.

Luego de explorar sobre los roles para el desarrollo de software en general, y partiendo de las características particulares de los ecosistemas de software, se considera que uno de los roles protagónicos durante el

desarrollo de los ecosistemas de software es el Arquitecto [Lazo Ochoa 2011; Unde 2008], en particular, el Arquitecto de integración y el Arquitecto de datos.

Adicionalmente en cada proceso participan analistas, jefe de proyectos (los de cada activo por separado y del ecosistema en general), desarrolladores, diseñadores y probadores. Cada uno de ellos cumple con las responsabilidades especificadas de cada rol por las diferentes metodologías convencionales de desarrollo de software.

### 3. APLICACIÓN DEL MODELO EN UN ENTORNO DE GESTIÓN DE PROYECTOS

Como se explicara en la introducción del artículo la actual propuesta tiene como antecedente un modelo para el desarrollo de ecosistemas de software particularmente definido para soluciones de gestión de proyectos [Castell y Piñero 2013]. Se explicará brevemente algunos de los principales elementos de la aplicación y el entorno de aplicación.

#### 3.1 ECOSISTEMA DE GESTIÓN DE PROYECTOS EN LA UCI

Se seleccionó como sistema de gestión de proyecto base del ecosistema la Suite GESPRO [Piñero Pérez y colectivo de autores 2011], en el entorno de la Universidad de las Ciencias Informáticas (UCI). En la UCI coexisten diferentes soluciones heterogéneas que manejan información relevante de interés institucional y para la gestión de los proyectos.

La UCI, independientemente de los procesos sustantivos relacionados con la formación del profesional, ejecuta proyectos de software para el mercado nacional e internacional dentro de centros de desarrollo de aplicaciones de software. Desde hace algunos años en la misma se ha estandarizado el uso de la Suite GESPRO para gestionar estos proyectos, sin embargo no se había logrado el nivel esperado de comunicación e integración con el resto de los sistemas.

Durante la aplicación del modelo propuesto se ejecutaron las actividades pertenecientes a los procesos descritos. Se identificaron los requisitos de información como base para la identificación y diagnóstico de los activos de software. Se identificaron los mecanismos permitidos y/o necesarios para poder integrar los activos y se definieron los participantes, sus responsabilidades dentro del ecosistema, así como las relaciones entre sí. En el proceso 5 *Diseñar el ecosistema* se obtiene la estructura del ecosistema, véase la Figura 5. Nótese que dicha estructura o arquitectura puede ser menos o más formal en dependencia del nivel de detalle que se tenga en el momento para elaborarlo, se sugiere que sea lo más detallado posible y se vaya actualizando durante el desarrollo del ecosistema. Durante la aplicación se eligió una representación por capas.

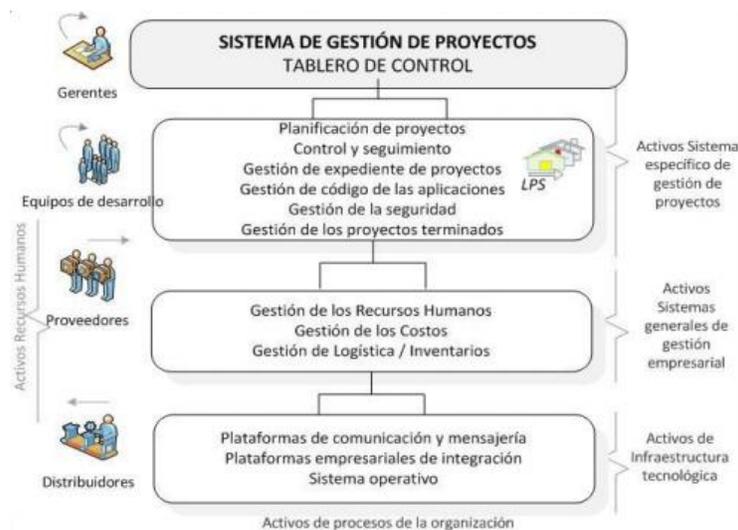
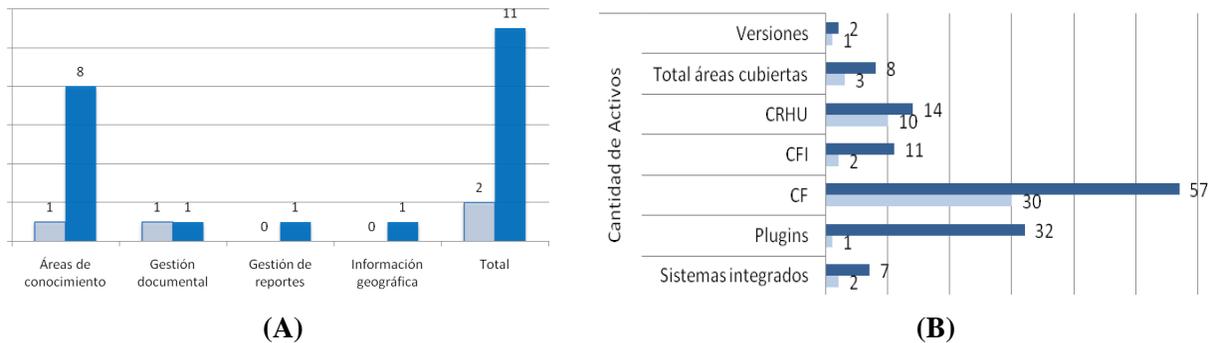


Figura 5. Vista de del ecosistema de software para la gestión de proyectos.

Tras la aplicación del modelo y el análisis de los resultados se evaluaron las variables de productividad, creación de nichos de mercado y robustez, de las cuales se comentaba en la introducción. A partir de la aplicación se definieron los criterios de evaluación para cada variable y se obtuvieron los siguientes resultados por cada uno. Se revisaron y compararon cada variable en dos momentos: anterior a la aplicación del modelo (azul claro) y después de aplicado el modelo (azul oscuro), véase la Figura 6.

- *Productividad*: como criterio básico se tomó la cantidad de funcionalidades de manera general y las funcionalidades de integración desarrolladas como parte del ecosistema.
- *Nichos de mercado*: se ofrecieron un amplio número de servicios adyacentes al desarrollo del ecosistema, como es el caso del apoyo del sistema GESPRO, base del ecosistema, en las asignaturas del Programa de Maestría en Gestión de Proyectos Informáticos. Además se obtuvieron otros productos y componentes durante el desarrollo del ecosistema.
- *Robustez*: se refiere fundamentalmente al crecimiento de producto en cuanto a varios elementos, uno de los establecidos fue el crecimiento de los activos de software.



**Figura 6. Comportamiento de los criterios de evaluación para las variables (A) Productividad (B) Robustez**

#### 4. CONCLUSIONES

A partir de la propuesta de un Modelo para el desarrollo de ecosistemas de software orientado a soluciones para la gestión de proyectos, se ha logrado generalizar una propuesta que sistematice las mejores prácticas del desarrollo de software en general, y que tengan en cuenta los elementos que definen a un ecosistema de software. Se definieron de igual manera principios y bases que deberán dirigir cada uno de los procesos descritos, sirviendo de base para la concepción y desarrollo de este tipo de iniciativas complejas.

La aplicación del modelo en un entorno real y complejo propició un estupendo punto de partida para retroalimentar y perfeccionar la propuesta, a partir de las principales insatisfacciones y lecciones aprendidas. De igual manera se establecieron los puntos fuertes del modelo que permiten asegurar que la propuesta representa una iniciativa novedosa y útil para el desarrollo de los ecosistemas de software.

Se considera que la presente es una propuesta que si bien no está totalmente acabada y probada en diferentes entornos, sí ofrece un primer acercamiento para el análisis de los ecosistemas de software. Pretende organizar un proceso de desarrollo diferente al que tradicionalmente se emplea en otros tipos de escenarios.

#### REFERENCIAS

ALVAREZ, L.G., RUBIERA, I.C. AND JIMÉNEZ, M.S.V. PROPUESTA DE SISTEMA DE ADMINISTRACIÓN DE PRODUCTOS DE SOFTWARE. In *UCIENCIA*. La Habana, Cuba: UCI, 2012, p. 14.

ANGEREN, J.V., KABBEDIJK, J., JANSEN, S. AND POPP, K.M. A Survey of Associate Models used within Large Software Ecosystems. In S. JANSEN, J. BOSCH, P. CAMPBELL AND F. AHMED. *Proceedings of the Workshop on Software Ecosystems 2011*. Brussels, Belgium, 2011, p. 27-39.

BOSCH, J. Ecosystems. In., 2009, vol. 2014.

BOSCH, J. From Software Product Lines to Software Ecosystems. In *13th International Software Product Line Conference (SPLC 2009)*. San Francisco, CA, USA, 2009, p. 10.

BOUCHARAS, V., JANSEN, S. AND BRINKKEMPER, S. Formalizing Software Ecosystem Modeling. In *IWOCE'09*. Amsterdam, The Netherlands: ACM, 2009, p. 10.

CASTELL, J. Modelo para el desarrollo de un ecosistema de software orientado a soluciones para la gestión de proyectos. In *Centro de Consultoría y Desarrollo de Arquitecturas Empresariales, Laboratorio de Investigaciones en Gestión de Proyectos, Facultad 5*. UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS, 2012, vol. master, p. 83.

CASTELL, J. AND PIÑERO, P.Y. EXPERIENCIAS EN EL DESARROLLO DE UN ECOSISTEMA DE SOFTWARE PARA LA GESTIÓN DE PROYECTOS EN UN AMBIENTE EMPRESARIAL. In *“III Taller internacional “Las TIC en la Gestión de las Organizaciones” XV Convención y Feria Internacional. Informática Habana*. La Habana, 2013.

CASTELL, J. AND PIÑERO, P.Y. MODELO PARA EL DESARROLLO DE UN ECOSISTEMA ORIENTADO A SOLUCIONES PARA LA GESTIÓN DE PROYECTOS. In *Congreso Iberoamericano de Ingeniería de Proyectos CIIP*. Pucón, Región Temuco, Chile, 2013, p. 12.

FRANTZ, R.Z. Integración de Aplicaciones: Un Lenguaje Específico de Dominio para el Diseño de Soluciones de Integración. In. Seville, Spain: University os Seville, 2008.

HEILIG, L. AND KARCH, S. SAP NetWeaver®: The Official Guide. 2008, pp. 85. Available from Internet:<[http://www.thali.ch/pdf/p14927\\_1.pdf](http://www.thali.ch/pdf/p14927_1.pdf)>.

HOHPE, G. Enterprise Integration Patterns. In., 2012.

HOHPE, G. AND WOOLF, B. *Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions*. edited by I. PEARSON EDUCATION. Edtion ed. Boston, MA, USA: Addison Wesley, 2003. 736 p. ISBN 0-321-20068-3.

HUANG, P. ESSAYS ON INNOVATION ECOSYSTEMS IN THE ENTERPRISE SOFTWARE INDUSTRY. In *Management*. Georgia, USA: Georgia Institute of Technology, 2010, vol. Doctor of Philosophy in the College of Management, p. 174.

IANSITI, M. AND LEVIEN, R. *The New Operational Dynamics of Business Ecosystems: Implications for Policy, Operations and Technology Strategy*. Edtion ed. Cambridge, Massachusetts, Estados Unidos., 2002. 115 p.

JANSEN, S. On the Necessity of Software Ecosystem Analysis: Can any software company do without? 2013, 2014]. Available from Internet:<<http://elgarblog.wordpress.com/2013/05/22/on-the-necessity-of-software-ecosystem-analysis-can-any-software-company-do-without-by-slinger-jansen/>>.

JANSEN, S., FINKELSTEIN, A. AND BRINKKEMPER, S. A Sense of Community: A Research Agenda for Software Ecosystems. In *31st International Conference on Software Engineering - ICSE 09*. Vancouver, Canada, 2009, p. 4.

KJETIL HANSSSEN, G. From Agile Software Product Line Engineering Towards Software Ecosystems. In *Faculty of Information Technology, Mathematics and Electrical Engineering Department of Computer and Information Science*. Trondheim, Norway: NTNU- Norwegian University of Science and Technology, 2010, vol. Philosophiae Doctor. PhD in Information and Communications Technology, p. 190.

LAZO OCHOA, R. Modelo de referencia para el desarrollo arquitectónico de sistemas de software en dominios de gestión. In. La Habana: Universidad de la Ciencias Informáticas, 2011, vol. Máster en Ciencias Técnicas, p. 97.

PIÑERO PÉREZ, P.Y., PESTANO PINO, H., ACOSTA, M.V., LÓPEZ, S.T., MATIAS, M.I., JORRÍN, M.G., ROLDÁN, I.M., PÉREZ, J.A.L., RIZO, J.M., FRANQUEIRO, E.A.M., SANTANA, F.N.A., VICTORE, R.D., HERNÁNDEZ, M.T. AND CALAÑA, E.S. GESPRO 11.05 Sistema para la Dirección Integrada de Proyectos para la Gestión de la Producción. In D.T.D.L.P. LABORATORIO DE GESTIÓN DE PROYECTOS. *COMPUMAT*. Cuba, 2011, p. 48.

PMI *A guide to the project management body of knowledge*. Edtion ed. Newtown Square, Pennsylvania, EEUU: ANSI/PMI, 2008. 496 p. ISBN 978-1-933890-51-7.

POPP, K.M. Definition of supplier relationships in software ecosystems as a basis for future research. In P. TYRVÄINEN, S. JANSEN AND M.A. CUSUMANO. *International Conference on Software Business - ICSOB 2010*. Jyväskylä, Finland: Springer 2010, 2010, p. 6.

RECENA SOTO, M., MUÑIZ, A., GOMEZ, A., BEAS, J.M., SUTILO, A. AND BENITEZ, F. Clinker Software Development Ecosystem. In. Sevilla , España: klicap <http://clinkerhq.com/>, 2012, vol. 2012.

SYBREN DEELSTRA, M.S., JAN BOSCH Product derivation in software product families: a case study. *The Journal of Systems and Software*.ELSEVIER, 15 January 2005 2005, vol. 74, no. 2, p. 173–194.

UNDE, A. The Role of an Architect. *Becoming an Architect in a System Integrator*. The Architectural Journal. Microsoft Corporation, USA, April 2008 2008, no. 15, p. 7-10.