

Relación entre Innovación y metodologías de desarrollo. En empresas de software de Entre Ríos ^ξ

Rafael Blanc *

Leandro Lepratte **

Rossana Sosa Zitto ***

Resumen

En el presente trabajo se mide la innovación en software en una muestra de firmas de la provincia de Entre Ríos. El software tiene características que complican su clasificación como un producto o servicio; por ello se abordó su caracterización como producto, servicio o proyecto. Por lo tanto, será difícil aplicar al software los métodos tradicionales de medición de la innovación (manuales), los cuales fueron creados para productos industriales. Para la medición de la innovación se usan dos corrientes teóricas: la tradicional, basada en los manuales de Oslo y Bogotá, y la corriente creada para la medición de servicios, también llamada enfoque de diferenciación. Se realizó el contraste entre ambos métodos y se obtuvieron diferentes resultados para la muestra. Como conclusión, queda abierta la discusión sobre cómo medir la innovación en proyectos de software.

Palabras claves: Software, Innovación, Métodos de Medición, Entre Ríos, Servicios

Abstract

This article analyzes the software innovation in a sample of firms in the province of Entre Rios. It is difficult to categorize the software as a product or service in view of its features. Therefore, we are dealing with its characterization as a product, service or project. Accordingly, the traditional methods of innovation measurement (manuals), which were created for industrial products, are difficult to apply to the software sector. In this context, there are two theoretical approaches: the traditional manuals, based on Oslo and Bogotá, and the approach aimed at measuring services also called also called differentiation approach. Both methods were contrasted and different results were obtained for the sample. Consequently, the discussion about how measuring innovation in software projects is still open.

Key Words: Software, Innovation, Measure Method, Entre Ríos, Services.

Clasificación JEL: O31, O32, L86

^ξ - Recibido 15 de agosto 2014 / Aceptado 27 de Septiembre 2014.

* Universidad Tecnológica Nacional, Facultad Regional Concepción del Uruguay (FRCU UTN). Grupo de Investigación en Desarrollo, Innovación y Competitividad (GIDIC). Correo electrónico: rlblanc@frcu.utn.edu.ar, rafaellujanblanc@yahoo.com.ar.

** Universidad Tecnológica Nacional, Facultad Regional Concepción del Uruguay (FRCU UTN). Grupo de Investigación en Desarrollo, Innovación y Competitividad (GIDIC).

*** Universidad Tecnológica Nacional, Facultad Regional Concepción del Uruguay (FRCU UTN). Grupo de Investigación en Desarrollo, Innovación y Competitividad (GIDIC).

ISSN: 2344-9195 <http://www.redpymes.org.ar/index.php/nuestra-revista> / <https://revistas.unc.edu.ar/index.php/pid/index>

Pymes, Innovación y Desarrollo – editada por la Asociación Civil Red Pymes Mercosur

This work is licensed under a Creative Commons Attribution 3.0 License.

1. Introducción.

Los estudios sobre innovación en el sector software se han desarrollado inicialmente con un esquema de medición inspirado en la industria tradicional de tipo manufacturera. Esto es, modalidades de medición que responden perfectamente a productos repetitivos con servicios definidos y con poca variación en el tiempo. Modalidad que es la adoptada por los manuales de innovación (Oslo, 2006). En paralelo a esto se ha comenzado a desarrollar un nuevo esquema de medición destinado a servicios (Saviotti y Metcalfe, 1984; Gallouj y Weinstein, 1997; Djellal and Gallouj, 2000; Edison, BinAli y Torkar, 2013). El mismo, busca dividir un servicio en diferentes componentes y evaluar si hay variación, agregado o disminución de los mismos a fin de determinar si hubo o no innovaciones.

En este trabajo se intenta explorar la siguiente cuestión: ¿existen diferencias en la medición de la innovación según el enfoque que se adopte?, ¿cuál de los modelos es más adecuado para este tipo de industria? Para esto se construyeron dos modelos de medición conforme a los enfoques mencionados que se aplicaron a una muestra de firmas del sector software en la provincia de Entre Ríos, Argentina, entre los años 2013 y 2014.

Dada la importancia que las formas de organización del trabajo adquieren en los resultados en innovación, según los estudios realizados en diferentes sectores (Borello et al., 2005; Erbes, Roitter y Delfini, 2011), se decidió evaluar si las metodologías de desarrollo, que determinan cómo debería organizarse el trabajo en las empresas de software, tienen impacto en los resultados en innovación de las firmas. Los modelos de metodologías van desde las rígidas (más antiguas) hasta las ágiles (las actuales) y toda una gama de comportamientos organizacionales híbridos (Coriat, 1992) realizados por las firmas para lograr los objetivos de las mismas. En este estudio se intenta determinar si hay o no relación entre las metodologías de desarrollo implementadas por las firmas de software de Entre Ríos y los resultados de innovación obtenidos por las mismas.

Para caracterizar a las firmas de acuerdo a su metodología se elaboró un *índice de agilidad* que dará una posición relativa para cada firma según su posición entre los paradigmas de desarrollo tradicionales y los ágiles. Se utiliza este método, dado que la mayor parte de las firmas no aplican metodologías en su forma codificada, si no que en un proceso tácito toman partes de los modelos y los adaptan a sus rutinas de trabajo. Finalmente se analizan otras variables de corte a fin de contextualizar la muestra, tales como el tamaño, situación de exportación y antigüedad de las firmas.

En cuanto a la selección del sector, se estudia la industria del software de la provincia de Entre Ríos porque representa una alternativa a los sectores tradicionales, como son el sector de materias primas y el de primera transformación de las mismas, y dado que el mismo ha experimentado un crecimiento sostenido a nivel nacional que también se vio reflejado a nivel provincial.

A su vez, la provincia cuenta con una variada oferta de formación en diferentes disciplinas relacionadas a las TIC, lo que impacta positivamente en la disponibilidad de mano de obra calificada al alcance de la firmas. El costo de las horas de programación es inferior al de los grandes centros de desarrollo nacionales como Córdoba, Rosario y Capital Federal, lo cual también abre interrogantes sobre el papel del marco institucional en la potenciación del valor agregado en el sector.

Entre Ríos tiene aglomeraciones de empresas de desarrollo de software en diferentes departamentos como Paraná, Uruguay y Gualeguaychú, y recientemente se ha creado el Polo Tecnológico Paraná que, con apoyo del gobierno nacional y provincial, logró reunir a veinticinco empresas del sector tecnología, de las cuales el 84% se dedica a la realización de productos y servicios de software. Hay otro desarrollo más espontáneo y que cuenta con gran impulso por la dotación de recursos humanos (oferta de profesionales aportadas por dos universidades UTN-FRCU y UADER-FCYT) que se dio en el departamento Uruguay (en particular la ciudad de Concepción del Uruguay) donde hay nueve firmas y múltiples programadores independientes que satisfacen clientes locales y del exterior.

La estructura del artículo es la siguiente: en el apartado 2 se enumeran los principales aportes teóricos del tema, tanto en innovación como en metodologías de desarrollo. En el apartado 3 se definen los objetivos principales del trabajo y la metodología aplicada. En la sección 4 se detallan los resultados del estudio. Finalmente en la sección 5 se establecen las conclusiones alcanzadas por el mismo.

2. Marco de referencia

La medición de la innovación en software presenta una dificultad de base que es la caracterización de este como un producto o un servicio. Debido a esto se encuentran dos grandes corrientes para su medición. La primera que lo trata sin hacer diferencia con respecto a los productos manufacturados “industriales”, es la corriente que se refleja en el llamado *enfoque de Asimilación*. El segundo enfoque, destaca la necesidad de un marco de medición propio para servicios el cual es denominado *enfoque de Diferenciación*. En este apartado se analizará qué enfoques de medición de innovación alternativos se pueden utilizar para medir la innovación en las firmas de software (2.1). Luego se justificará por qué surge la dificultad del enfoque de medición, o sea la dificultad de posicionar al software como producto, servicio o proyecto (2.2) y finalmente se abordará la importancia de las metodologías de desarrollo en la innovación de las firmas de software (2.3).

2.1. Enfoques de medición de innovación.

2.1.1. Enfoque de Asimilación

Este enfoque, representado por los estudios que utilizan para medir innovación en software principalmente las directrices del manual de Oslo (Oslo, 2006), consiste en aplicar el mismo marco de medición de la industria a los servicios. Observando el manual surge una serie de definiciones en la forma de recoger los datos de innovación, a partir de dos opciones: una basada en el sujeto y otra basada en el objeto. El primer caso basado en el sujeto, y el más común en los estudios por sector o las encuestas nacionales de conducta tecnológica, consiste en seleccionar un panel de firmas mediante cierto procedimiento estadístico y realizar una serie de preguntas sobre el comportamiento innovador, estrategias, incentivos y barreras a la innovación, entre otras variables. Estos datos son codificados y tratados estadísticamente a fin de obtener indicadores que midan

innovación. Mientras que, la basada en el objeto, consiste en analizar empresas innovadoras o que han tenido un comportamiento innovador comprobado por medios descriptivos y cualitativos, con énfasis en el *proceso* que da lugar a innovaciones. El manual de Oslo considera que:

“la innovación implica la utilización de un nuevo conocimiento o un nuevo uso o una combinación de conocimientos existentes. El nuevo conocimiento puede haber sido generado por la empresa innovadora en el curso de sus actividades de innovación (es decir con la I+D interna) o adquirido externamente a través de determinados canales (es decir, comprando la nueva tecnología). La utilización de un nuevo conocimiento o de una combinación de conocimientos existentes requiere esfuerzos innovadores por la empresa que pueden ser diferenciados de las rutinas habituales.” (Oslo, 2006: 44).

Hace referencia a la existencia de servicios y reconoce su importancia en la economía actual. A su vez los clasifica en cuatro grupos: los servicios relativos principalmente a los bienes (como el transporte y la logística), los servicios vinculados a la información (como los centros de atención al cliente), los servicios basados en el conocimiento y los servicios orientados a las personas (como la asistencia sanitaria). A su vez, en el apartado de tecnologías de la información nombra la existencia de software como producto, ya sea su presentación como producto separado, o el resultante del mismo como parte embebida en equipos de diferentes tipos. El manual de Oslo explicita el problema del software en los siguientes apartados:

...“elaborar el índice del desarrollo, la producción, la adaptación y la utilización de los programas informáticos es un asunto más complicado ya que se trata de actividades que se valoran en el conjunto de la economía. Se han efectuado encuestas en empresas desarrolladoras de productos TIC tanto sobre el uso de las TIC como sobre las actividades de I+D.” (Oslo, 2006:32).

Hace referencia a la problemática de la innovación en servicios, destacando que la distinción entre productos y procesos es a menudo poco clara, ya que la producción y el consumo tienden a ocurrir simultáneamente. También destaca que la innovación en servicios es un proceso incremental con menos posibilidad de cambios. Así, menciona que “La actividad de innovación en los servicios es un proceso generalmente continuo, consistente en una serie de modificaciones introducidas progresivamente en los productos y en los procesos” (Oslo, 2006:48). El Manual tiene dificultades para dividir un producto o servicio como se ve en las siguientes afirmaciones:

“Si la innovación implica características nuevas, o significativamente mejoradas, del servicio propuesto a los clientes, es una innovación de producto. Si la innovación implica la utilización de métodos, de equipos y/o de unos conocimientos nuevos o significativamente mejorados para prestar el servicio, es una innovación de proceso” (Oslo, 2006:64).

Así, el manual tiende a encasillar todo en las categorías preestablecidas: producto, proceso, mercadotecnia y organización. Djellal, Gallouj y Miles (2013) describen el *enfoque de asimilación* como un formato de medición de la innovación en servicios, en este caso el software, *del mismo modo que se mide en industria de manufactura*. Estos autores son importantes exponentes del enfoque que se explica a continuación, como alternativa crítica al descripto hasta aquí.

2.1.2. Enfoque de Diferenciación

Este enfoque comúnmente conocido como *demarcation approach* o *differentiation approach*, plantea que la diferenciación entre producto y servicio no es clara en muchos casos. Y que el enfoque de asimilación tiende a minimizar la diferencia entre industrias de manufactura y las industrias de servicios. El enfoque de diferenciación, critica al enfoque de asimilación y considera que los servicios deben ser medidos con herramientas de diseños específicos para los mismos (Archibugi et al, 1994; Coombs and Miles, 2000; Djellal and Gallouj, 2000). Gallouj y Weinstein (1997) y Saviotti y Metcalfe (1984) definen un producto o servicio como un conjunto de características que separan en vectores; los cuales al ser modificados pueden resultar en una innovación. Un servicio o producto se traducirá en el vector de características de estos. Es decir, un servicio se define como una serie de características (Y_1, Y_2, \dots, Y_N) que son la suma de una serie de competencias y características técnicas. Las *competencias* (C_1, C_2, \dots, C_N) son de carácter intangible y pueden definirse como la capacidad para hacer uso de tecnologías tangibles o intangibles para desarrollar el servicio requerido. Las *características técnicas* (X_1, X_2, \dots, X_N) como por ejemplo procedimientos, equipamientos y lenguajes pueden ser tangibles o intangibles y movilizan las competencias. Ambas, competencias y características técnicas, se retroalimentan del modo que nuevas capacidades llevan a nuevas características técnicas y estas últimas llevan a la construcción de nuevas capacidades. Cabe destacar que este modelo tuvo sucesivas mejoras, la última versión del mismo se encuentra en Gallouj y Toivonen (2011), que divide al modelo en dos partes: una respecto al cliente y otra orientada por el productor del servicio.

De esta forma se le otorga una dimensión propia al cliente, y se toman en cuenta, las características de las competencias de los clientes. Este enfoque encuentra su justificación por considerarse a sí mismo como más acertado al momento de medir un producto que generalmente no es repetitivo, como se da en la industria de software, donde cada producto está muy condicionado por los requerimientos de los clientes.

Así, conforme a este modelo, dependiendo del grado de modificación de los vectores, será el nivel de innovación resultante. Gallouj y Weinstein (1997) a partir del modelo de vectores definen una serie de tipos de innovaciones:

Innovación radical: supone la creación de un producto o servicio completamente nuevo, es decir, cuyo vector de características (Y_1, Y_2, \dots, Y_N) es completamente distinto al de los productos ya existentes, también su vector de características técnicas (X_1, X_2, \dots, X_N). En cambio, el vector de competencias (CL_1, CL_2, \dots, CL_N) se ve incrementado o ampliado por las nuevas competencias traídas por el cambio.

Innovación de mejora: consiste en la mejora en uno de los vectores (características técnicas o competencias), sin variar la estructura de estos, o sea sin cambiar el sistema. Mantiene los mismos vectores de entrada (X_1, X_2, \dots, X_N) y (CL_1, CL_2, \dots, CL_N) pero cambiando de sus componentes X_i o C_i es mejorado, pero el vector de características del producto final (Y_1, Y_2, \dots, Y_N) permanece con los mismos elementos donde al menos uno de ellos (Y_i) ha mejorado.

Innovación incremental: la estructura del sistema, es decir, los vectores $\{X, C, Y\}$, permanecen casi constantes pero se ve modificada de manera marginal por la adición de un elemento nuevo en $\{X\}$, y en $\{Y\}$ o mediante la sustitución de alguno de sus

elementos. Sin que la nueva característica traccione competencias específicas. Esto puede llevar a mejoras del vector {Y} o simplemente una mejora en el proceso. La diferencia entre las innovaciones incrementales y las de mejora puede ser considerada como una cuestión de matiz, al referirnos a la mejora de una característica es una innovación de mejora, y a la adición de una nueva característica en el caso de la innovación incremental.

Es de destacar que la mayor parte de los estudios empíricos utilizando este último enfoque son de carácter cualitativo buscando profundizar las ventajas y desventajas del mismo. Hay escasas publicaciones sobre intentos de utilizar los vectores, y los mismos en general constan de pocos componentes para hacer estudios cuantitativos (Hertog, Gallouj y Segers, 2009; Fairhurst, 2011)

2.2. Peculiaridades del software: ¿Producto o servicio? ¿O ambos?

El primer problema que surge al analizar software, es la definición sobre si es un producto o un servicio tal como lo venimos explicitando aquí.

En nuestro estudio, para comenzar se tratará de definir producto y, luego se pasará a la problemática de estas definiciones en la industria del software.

Si vamos a definiciones tradicionales de producto, sus características son: replicable (que se puede hacer muchas veces exactamente de la misma forma), estandarizado (esto responde a la condición anterior ya que si un producto se quiere lograr exactamente igual cada vez se debe tener estándares sólidos que permitan su replicación exacta), diseño (los productos tienen fase de diseños previos, donde por métodos científicos y empíricos se logra fijar los estándares a alcanzar para un producto sea reproducible y homogéneo) y tangible (la idea de producto está fuertemente asociada a lo palpable del mismo (Hartman y Lindgren, 1993). Por lo tanto la palabra producto, tiene una connotación física en el pensamiento del usuario o cliente.

Si vamos a definiciones tradicionales de servicio se caracteriza por: diseño (la mayoría de los servicios cuenta con diseño, pero el mismo tiene menos detalle que el de los productos, ya que los servicios implican mayor personalización que los productos), personalización (una de las principales características de los servicios que tienen éxito es que tienen un alto nivel de importancia la opinión del cliente y sus gustos y necesidades), repetible (dados los puntos anteriores podemos deducir que un servicio puede ser repetible, pero el resultado no será exactamente el mismo como ocurre en los productos), y tangible o intangible (los resultados de un servicio no siempre son tangibles, siempre responden a cierta necesidad del cliente, pero estas no siempre se presentan en forma física). Un ejemplo de esto puede ser la formación dado que, si bien hay un entorno físico para realizarla, los resultados en el cliente son intangibles. (Hartman y Lindgren, 1993; Wolak, Kalafatis y Harris, 1998).

Tabla 1: Características principales de productos y servicios

Producto	Servicio
Replicable	Diseño
Estandarizado	Personalización
Diseño	Repetible
Tangible	Tangible
	<i>Intangible</i>

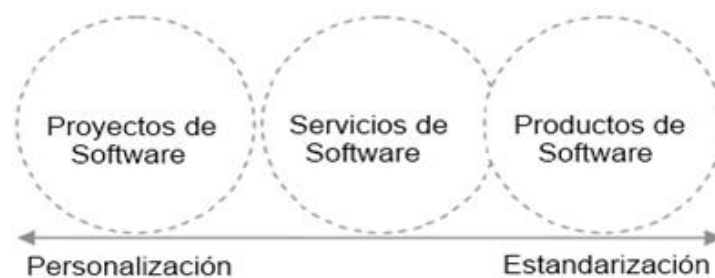
Fuente: elaboración propia en base a Hartman y Lindgren (1993); Wolak, Kalafatis y Harris (1998).

A estas definiciones, que son de difícil interpretación en muchos casos se suma que, en la dinámica económica actual, la necesidad de competir ha llevado a las firmas a otorgar productos con un componente alto de servicio a fin de satisfacer a los clientes y lograr obtener diferenciación en el mercado. De ahí surge el concepto *producto ampliado, extendido o aumentado*, esto responde a un producto o servicio básico con una serie de productos o servicios accesorios que incrementan la satisfacción del cliente con el mismo. Lo que, en algunos productos o servicios, se vuelve complejo diferenciar qué genera más valor si lo principal, que cumpla con la necesidad primaria o el conjunto de accesorios que amplían al mismo.

La identificación del software como un producto o servicio puede ser dificultosa, debido a sus características relacionadas fuertemente con el cliente, el cual puede solicitar un producto muy específico o un software masivo que resuelva sus problemas (enlatado). Muchas empresas no tienen muy definida la frontera entre producto y servicio, desarrollándose en ambos sectores al mismo tiempo (Nambisan, 2002).

Aramand (2008) proporciona una clasificación que resulta conveniente para aplicar al caso de software en lo que respecta a su categorización como producto y servicio. Divide el software en tres sectores que van desde lo personalizado a lo estandarizado. Estas tres divisiones son Proyectos de Software, Servicios de Software y Productos de Software. Siendo el primero el más personalizado a las necesidades y requerimientos del cliente; en pocas palabras, hecho a medida del mismo. En el otro extremo, se encuentra el producto de software que es altamente estandarizado y es independiente de los requerimientos del cliente (un ejemplo de esto serían los paquetes desarrollados por multinacionales como Microsoft, Adobe, Oracle, etc. en la cual el cliente compra un paquete llave en mano con ciertas prestaciones prefijadas por un convenio de licencia y no puede modificar el paquete). Entre estos dos extremos (los proyectos de software y los productos de software) se encuentran los *servicios de software que son productos con cierto grado de personalización hacia al cliente que no llegan a ser totalmente desarrollados para él ni tampoco un producto de consumo masivo*.

Grafico 1: Tipos de software por personalización



Fuente: Aramand (2008).

A su vez la diferencia entre un producto masivo o personalizado puede estar dada por la cantidad de código que sea reutilizado en el producto. Si un producto utiliza muchos módulos, objetos o funciones de otros programas anteriormente desarrollados este no será nuevo ni tan personalizado como para ser un proyecto de software. En la cantidad de código propio podría definirse las características de personalizado o estándar del producto. Por otra parte es interesante la visión de Dossani (2005) en la que esquematiza mediante el siguiente cuadro ejemplos de grados de personalización de software dependiendo el tipo de usuario al que va destinado.

Tabla 2: diferentes tipos de software

	Producto Usado Por	Personalizado para:
Sistema Operativo	Todos los usuarios	Ningún usuario
Herramientas	Muchos usuarios	Pocos Usuarios
Aplicaciones	Pequeños y Grandes Usuarios	Muchos Usuarios

Fuente: Dossani (2005)

Lo anterior impacta en la forma que se miden los resultados de innovación, ya que las preguntas clásicas como ¿Si desarrollo un nuevo producto o no? ¿Si desarrollo un nuevo servicio o no? tienen graves dificultades en industrias como las de software donde la oferta al cliente es tan amplia y personalizable.

En este estudio bajo el enfoque de asimilación, se considerará al software como producto, y bajo el enfoque de diferenciación, al software como servicio, a fin de contrastar ambos y tratar de determinar sus ventajas y debilidades.

2.3. Metodologías de Desarrollo de Software

Dada la importancia de las formas de trabajo en los resultados en innovación (Coriat, 1992; Borello et al., 2005; Erbes, Roitter y Delfini, 2011), en el presente estudio se relacionará las *formas de medición* con las *metodologías de desarrollo* utilizadas por la firma. De esta forma se busca establecer si las metodologías tienen relación con los resultados referentes a innovación de la muestra.

Una *metodología* puede definirse como un conjunto integrado de técnicas y métodos que permite abordar de forma homogénea cada una de las actividades del ciclo de vida

de un proyecto de desarrollo. Es un proceso de software detallado y completo. Las metodologías se basan en una combinación de los *modelos de proceso genéricos* (Cascada, Prototipos, Incremental, V, Espiral, RUP, CRYSTAL, DSDM, SCRUM y Extreme Programming), definen artefactos, roles y actividades, junto con prácticas y técnicas recomendadas.

Por *ciclo de vida* se entiende el conjunto de fases por las que pasa el sistema que se está desarrollando desde que nace la idea inicial hasta que el software es entregado o reemplazado. Los dos principales componentes de los ciclos de vida son las fases y los entregables. Las fases son un conjunto de actividades relacionadas con un objetivo en el desarrollo del proyecto, las mismas pueden ser lineales o recursivas o mezclas de ambos. Por su parte, los entregables corresponden a objetos físicos o virtuales los cuales son entregados entre fases o de las fases a clientes internos o externos. Las etapas del ciclo de vida cambiarán de acuerdo a la metodología que sea seleccionada para el desarrollo del proyecto.

La metodología para el desarrollo de software es un modo sistemático de realizar, gestionar y administrar un proyecto. Esta metodología comprende los procesos a seguir sistemáticamente para idear, implementar y mantener un producto software desde que surge la necesidad del producto hasta que se cumple con el objetivo por el cual fue creado. Dentro de las mismas se puede encontrar cómo dividir un proyecto en etapas, qué tareas se llevan a cabo en cada etapa, qué restricciones deben aplicarse, que técnicas y herramientas se emplean, cómo se controla y gestiona un proyecto. Las metodologías podrán cambiar la forma que se toman los requerimientos, la interacción con el cliente la técnica de programación y el testeado entre otros. Las metodologías han evolucionado a medida que se expandió el uso de software, y se volvió un producto de consumo masivo incorporado a diferentes tipos de dispositivos que se emplean en la vida diaria.

En los esquemas de desarrollo tradicionales el proceso de desarrollo llevaba asociado un marcado énfasis en el control del proceso mediante una rigurosa definición de roles, actividades y artefactos, incluyendo modelado y documentación detallada. Este método para abordar el desarrollo de software ha demostrado ser efectivo y necesario en proyectos de gran tamaño (respecto a tiempo y recursos), donde por lo general se exige un alto grado de burocracia en el proceso de control y seguimiento. Sin embargo, este enfoque no resulta conveniente para muchos de los proyectos actuales donde el entorno del sistema es de alto dinamismo, y en donde se exige reducir drásticamente los tiempos de desarrollo pero manteniendo una alta calidad y bajos costos.

El dinamismo y variabilidad de la actual industria del software requirió la adaptación y transformación de las metodologías convencionales a las nuevas características del mercado caracterizado por el rápido desarrollo de aplicaciones y la reducción de la vida de los productos (Boehm, 2006). El ciclo de vida de los productos y servicios de software se acorta, lo que obliga a incrementar la productividad, a disminuir el tiempo de reacción, y a adaptarse rápidamente a los cambios y a las nuevas necesidades de los clientes.

Hubo cambios en relación con la postura de los clientes lo que actualmente tienen más conocimiento de lo que pueden hacer los productos de software y están dispuestos a realizar más requerimientos y de mayor nivel de complejidad. Kent Beck 2004 señalaba la importancia del cambio señalando que todo en el software cambia. Los requerimientos, el diseño, el negocio, la tecnología, los miembros del equipo. El problema no es el cambio en sí mismo, puesto que sabemos que el cambio va a suceder;

el problema es la incapacidad de adaptarnos a dicho cambio cuando éste tiene lugar. Estamos obligados a abordar este cambio con nuevas alternativas que permitan adaptarnos mejor a esta situación.

A mediados de los años 90 surgen los métodos de desarrollo ágil de software (Agile Software Development ASD) como una reacción contra los métodos tradicionales considerados excesivamente pesados y rígidos, por su carácter normativo, fuerte dependencia de planificaciones detalladas costosas en tiempo y recursos previos al desarrollo. Los métodos ágiles perciben cada cambio como una oportunidad para mejorar el sistema e incrementar la satisfacción del cliente; la gestión del cambio se convierte en un aspecto inherente al propio proceso de desarrollo software mejorando así su adaptación a entornos turbulentos. Qumer y Henderson-Sellers (2007) definen *agilidad* como un comportamiento persistente o habilidad, de entidad sensible, que presenta flexibilidad para adaptarse a cambios, esperados o inesperados, rápidamente; persigue la duración más corta en tiempo; usa instrumentos económicos, simples y de calidad en un ambiente dinámico; y utiliza los conocimientos y experiencia previos para aprender tanto del entorno interno como del externo.

Las metodologías tradicionales imponen una fuerte disciplina de trabajo sobre el proceso de desarrollo de software, con el objetivo de asegurar que el software que se obtenga satisfaga los requerimientos del usuario y reúna estándares aceptables de calidad. La planificación del trabajo es rigurosa, muy documentada tanto en alcance como en resultados, con roles rígidos bien marcados de difícil cambio durante el proyecto.

En contraposición, las metodologías ágiles aportan nuevos métodos de trabajo que apuestan por una cantidad apropiada de procesos. Disminuye la excesiva cantidad de cuestiones administrativas (planificación, control, documentación), sin que esto les lleve al extremo de la ausencia de documentación tanto del código como de los procesos (Canós, Leteliery y Penadés; 2003). Se tiene en cuenta que se producirán cambios durante el ciclo de desarrollo y cuanto menos rígidos sean los marcos administrativos menores serán los costos asociados a los mismos. Se cambia la figura del cliente de un rol de aportar requerimientos y recursos económicos, en el que se encuentra en las metodologías tradicionales, a un lugar importante en el proceso de desarrollo con una comunicación fluida con el mismo a fin de asegurar resultados de calidad. Otra gran diferencia es que las metodologías ágiles funcionan en forma adaptativa, mientras que las tradicionales buscan ser predictivas. Asimismo, el proceso de desarrollo en las metodologías ágiles está unido al par usuario-desarrollador, en cambio en las metodologías tradicionales esto se ve remplazado por la noción de cliente-producto.

Tabla 3: diferencias entre metodologías ágiles y tradicionales

Metodologías Ágiles	Metodologías Tradicionales
Basadas en heurísticas provenientes de prácticas en la producción de código.	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo.
Especialmente preparados para cambios durante el proyecto.	Cierta resistencia a los cambios
Impuestas internamente (por el equipo)	Impuestas externamente
Proceso menos controlado, con pocos principios.	Proceso mucho más controlado, con numerosas políticas/normas.
El cliente es parte del equipo de desarrollo, forma en muchas ocasiones parte del equipo de test del proyecto asegurando la calidad de los resultados.	El cliente interactúa con el equipo de desarrollo mediante reuniones periódicas para ser informado y para entregas de resultados.
Grupos pequeños (<20 integrantes) y generalmente trabajando el mismo sitio	Grupos de cualquier tamaño, mayores facilidades para realizar proyectos distribuidos
Pocos artefactos	Más artefactos
Poca documentación	Mucha documentación, estructura más burocrática.
Pocos roles	Más roles
Menos énfasis en la arquitectura del software.	La arquitectura del software es esencial y se expresa mediante modelos
Ciclos cortos de desarrollo	Ciclos largos de desarrollo

Fuente: elaboración propia

Teniendo en cuenta lo anterior se pueden encontrar los siguientes modelos ágiles y tradicionales de desarrollo de software que se muestran en la tabla a continuación.

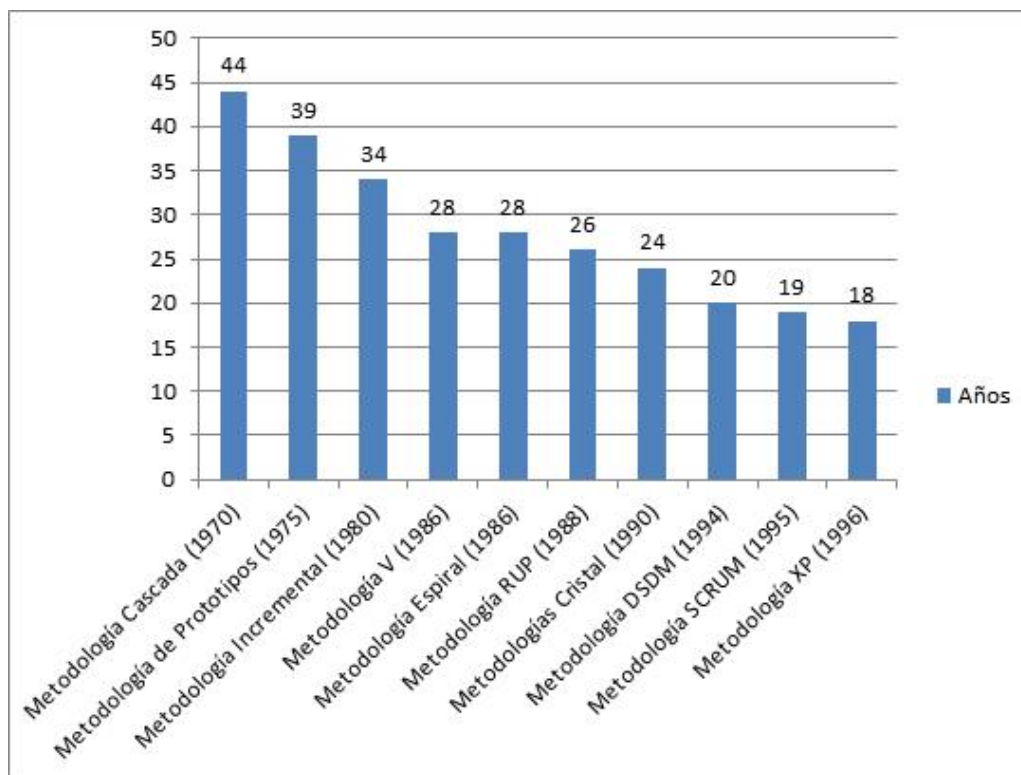
Tabla 4: Clasificación de las metodologías en tradicionales y ágiles

	Tradicionales	Ágiles
Metodología de Cascada	X	
Metodología de Prototipos	X	
Metodología Incremental	X	
Metodología V	X	
Metodología Espiral	X	
Metodología RUP	X	
Metodologías CRYSTAL		X
Metodología DSDM		X
Metodología SCRUM		X
Metodología Extreme Programing XP		X

Fuente: elaboración propia

Las metodologías de desarrollo fueron evolucionando de un paradigma tradicional (rígido) hacia uno ágil. En el siguiente gráfico se establece la antigüedad aproximada de cada metodología (Ver Gráfico 2).

Grafico 2: De tradicional a Ágil Evolución. Años desde su origen.



Fuente: elaboración propia

A continuación se enuncia la metodología utilizada en el trabajo, a fin de relacionar los diferentes elementos mencionados en el marco de referencia. Además, se detallan los objetivos y las principales variables del estudio.

3. Metodología

Los objetivos del trabajo son los siguientes: contrastar los enfoques de medición de innovación en las firmas de software de la provincia de Entre Ríos, y determinar si hay relación entre las metodologías de desarrollo implementadas por las firmas analizadas y sus resultados en innovación.

Para el logro de dichos objetivos, se realizó un análisis de una muestra de firmas del sector de desarrollo software en la provincia de Entre Ríos (exceptuando las firmas que no producen software propio o adaptan software de terceros: como distribuidores, servicios técnicos y mesas de ayuda o soporte). De acuerdo al relevamiento de datos secundarios, datos de cámaras, información de polos y búsquedas en internet se contactó una serie de empresas. A las mismas se les aplicó la metodología Snow Ball a fin de conseguir nuevos casos. La muestra cuenta con 23 firmas relevadas, sobre un universo estimado en base a datos primarios y secundarios de 60 firmas. Las firmas analizadas fueron entrevistadas, con un formulario diseñado para obtener datos generales de la empresa, grado de innovación y metodologías de desarrollo. Asimismo fueron excluidas las firmas con menos de 3 ocupados, a fin de considerar sólo aquellas que pudieran estar en condiciones de implementar alguna metodología de desarrollo. Se descartaron los freelancers, que trabajan en forma remota por horas para matrices en el exterior u

otras provincias. La muestra fue obtenida desde abril del 2014 y aún se encuentra en desarrollo el muestreo (agosto 2014).

Variables principales del estudio:

Grado de innovación: esta variable representa los resultados en innovación de las empresas alcanzados durante el último año. Para realizar la misma se utilizó el enfoque de diferenciación conformando un vector de diferentes variables de un producto de software. Los componentes del vector fueron sumados y promediados a fin de obtener una variable continua para el presente análisis.

Tabla 5: Componentes variable grado de innovación

Variable	Tipo
Cambio en forma en que vende el producto.	Binaria
Cambio en Forma en que distribuye el producto.	Binaria
Cambio en Forma de pago del producto.	Binaria
Cambio en Contrato de licencia del producto.	Binaria
Cambio en Servicios que ofrece al cliente	Binaria
Cambio en Toma de Requisitos	Binaria
Cambio en Planeación del proyecto	Binaria
Cambio en Seguimiento del proyecto.	Binaria
Cambio en Modelado (análisis y diseño)	Binaria
Cambio en Testing	Binaria
Cambio en Implementación	Binaria
Cambio en Soporte del software	Binaria
Nuevas funcionalidades al programa	Binaria
Nuevos módulos al programa	Binaria
Modificación Parcial Interfaz	Binaria
Modificación Radical Interfaz	Binaria
Ha hecho cambios en los dispositivos de los cuales se accede a su software.	Binaria
Sistema sobre el cual corre su software	Binaria
Lenguaje de programación en el cual desarrolla	Binaria
Framework o herramientas	Binaria
Motor de base de datos	Binaria
Rango de 0 a 1	

Fuente: elaboración propia

Flexibilidad en el Desarrollo: es una variable continua compuesta por la suma de una serie de variables de escala que evalúan la flexibilidad de la metodología que utiliza la firma.

Tabla 6: Componentes de la variable flexibilidad en el desarrollo

Variable	Tipo
Grupo Estable Coordinado por Team-Leader	Binaria
Grupos conformados especialmente para un proyecto	Binaria
En su empresa los empleados realizan horas extras	Binaria
En su empresa realiza teletrabajo	Binaria
Existe rotación entre los roles	Binaria
Existe rotación entre proyectos	Binaria
Se realizan reuniones de equipo:	Escala
El equipo de trabajo se encuentra en el mismo edificio	Binaria
El equipo que se encuentra radicado en la empresa, trabaja	Binaria
La metodología utilizada varía en función del proyecto	Binaria
Frecuencia de comunicación con el cliente.	Escala
El cliente prueba versiones incompletas.	Escala
El cliente prueba versiones finales.	Escala
¿El cliente aporta ideas para nuevos productos o servicios?	Escala
Se planifica en forma	Escala
Se realiza el seguimiento en forma	Escala
Se revisan los planes	Escala
La cantidad de documentación en su empresa es:	Escala
Frecuencia con la que se documentas	Escala
Rango de 0,1711 a 1	

Fuente: elaboración propia

Dada la importancia del Tamaño, Exportación y la Antigüedad de las firmas para los resultados, tanto de innovación como para el uso de ciertas metodologías de desarrollo, se medirán las siguientes variables.

Tamaño: es una variable de escala que representa la cantidad de empleados que tiene una firma, siendo grandes las de más de 20 empleados, medianas entre 20 y 10 empleados, y pequeñas menos de 10 empleados. Se debe aclarar que estas clases de fueron seleccionadas y ajustadas a la realidad de las empresas de software de Entre Ríos que poseen pocos empleados en comparación a otras zonas del país como Buenos Aires. Se descartó la opción tradicional del volumen de facturación por ser un dato muy sensible al momento de realizar la encuesta.

Exporta: es una variable que toma dos valores. Uno (1) para las empresas que exportan y cero (0) para las empresas que no exportan.

Antigüedad: representa la antigüedad en años desde el inicio de actividades de la firma en la provincia de Entre Ríos.

A continuación se presentan los principales resultados de la muestra analizada por el presente estudio a fin aceptar o rechazar las hipótesis planteadas.

4. Resultados

En el análisis de los resultados, la primera variable a examinar es la antigüedad de las firmas de la muestra. La mayor parte de las firmas se concentra en la clase de 0 a 10 años (60,9% de los casos), lo cual coincide con la etapa de mayor crecimiento de la industria de software a nivel nacional. En menor medida se encuentran empresas de más de 10 años de antigüedad con un 39,1%. Dentro de las firmas más antiguas, una mínima parte supera los 20 años en el mercado (8,6% del total de la muestra).

Tabla 7: Antigüedad

Años	Frecuencia
Menos de 5 años	26,10%
De 5 a 10 años	34,80%
Entre 10 y 20 años	30,40%
Más de 20 años	8,60%

Fuente: elaboración propia

En cuanto al tamaño de las empresas según la cantidad de empleados, la mayor parte de las firmas está por debajo de los 5 empleados (26,1%). Se puede ver que las de menor peso porcentual son las que emplean a más de 20 personas con el 8,7%, las cuales podríamos considerar como grandes para los parámetros provinciales. En promedio, las firmas poseen aproximadamente 13 empleados y ocupan un total de 300 puestos de trabajo a nivel provincial.

Tabla 8: Tamaño de las firmas.

Empleados	Frecuencia
Hasta 5 empleados	26,10%
De 5 a 10 empleados	21,70%
Más de 10 hasta 20 empleados	43,50%
20 empleados y más	8,70%

Fuente: elaboración propia

En la muestra se observa que solo el 52,2% de las firmas exportan. En esta dirección, se infiere que la mayor parte las firmas combina el mercado interno con el externo. Aunque, en menor medida encontramos empresas que se dedican al mercado interno a nivel provincial y nacional.

Tabla 9: Exportadora o no

Exporta	Porcentaje
No	47,8
Si	52,2

Fuente: elaboración propia

Si relacionamos las tres variables anteriores (tamaño, situación de exportación y antigüedad de las firmas) podemos observar que las empresas que más exportan (70%) son aquellas cuya estructura ocupacional es de 5 a 20 empleados. Por otra parte, las firmas de mayor tamaño son aquellas que tienen una trayectoria menor a 10 años; las empresas más pequeñas tienen una media de 14 años de antigüedad y en general se dedican a un producto que mantienen en el tiempo. Esto contrasta con las empresas modernas las cuales en general no tienen productos bien definidos y responden a las demandas de los clientes.

Tabla 10: Exportadora o no

Empleados	De 0 a 5	De 5 a 10	De 10 a 20	Más de 20
Antigüedad (Media)	14	8	12	9
Exporta (% por clase)	17,39%	39,13%	30,43%	13,04%

Fuente: elaboración propia

Observando el grado de innovación, y teniendo en cuenta el rango de valores que puede asumir la variable (0 a 1), se destaca en la muestra un valor máximo de 0,45 y un valor mínimo de 0,05 con lo que se puede descartar la presencia de empresas con innovaciones radicales. Por otro lado, se evidencia que la mayor parte de los casos en la muestra (73.9 %) no supera el 0,25. Por lo tanto, podríamos considerar a estas firmas como no innovadoras.

Tabla 11: Resultados variable grado de innovación

Grado de innovación	Frecuencia	Frecuencia acumulada
0,05	8,7	8,7
0,1	13	21,7
0,15	17,4	39,1
0,2	21,7	60,9
0,25	13	73,9
0,3	4,3	78,3
0,35	4,3	82,6
0,4	4,3	87
0,45	13	100

Fuente: elaboración propia

Si transformamos la tabla anterior a una escala similar a la utilizada por Gallouj y Weinstein (1997) obtenemos los siguientes niveles:

Tabla 12: variable grados de innovación por clase

Clase	Grado de innovación	Frecuencia
No innovadora	0 a 0,25	73,90%
Innovación de mejora	0,26 a 0,50	26,10%
Innovación incremental	0,51 a 0,75	0,00%
Innovación radical	0,76 a 1	0,00%

Fuente: elaboración propia

Lo anterior muestra que no hubo innovaciones radicales ni incrementales en el período analizado (un año antes de la realización de la encuesta). Sin embargo, se identificaron innovaciones de mejora en el 27,8% de las firmas. Caso contrario, si utilizamos el indicador tradicional de los manuales de innovación (tabla siguiente), observamos que los resultados mejoran notablemente, mostrando que más del 34% de las empresas poseen nuevos productos, el 26% posee nuevos servicios, el 8,7% de las firmas presenta nuevos modelos de comercialización y finalmente el 17,4% realizó cambios organizacionales. Esto muestra, tal como se esperaba, que los sistemas de medición no arrojan los mismos resultados.

Tabla 13: Indicador de innovación (enfoque de asimilación)

	Si	No
Nuevos productos	34,80%	65,20%
Nuevos servicios	26,10%	73,90%
Nuevos modelos de comercialización	8,70%	91,30%
Cambios organizacionales	17,40%	82,60%

Fuente: elaboración propia

Por otra parte, si analizamos los resultados de la variable flexibilidad en el desarrollo, teniendo en cuenta su rango de 0,17 a 1, podemos observar tres niveles: un primer nivel con el 27,8% de los casos que corresponde con la aplicación de metodologías tradicionales, un segundo nivel con el 11,1% denominado semi-ágiles que tiene gran parte de las conductas pero aún le falta algunos puntos, y un último nivel con el 16,7% llamado ágiles que cumple con casi todos los postulados de las metodologías ágiles. Debe tenerse en cuenta, que durante las entrevistas varias empresas declararon utilizar una metodología concreta como Scrum o Rup. Pero solo dos declararon utilizarlas en su totalidad, las demás declararon hacer adaptaciones en pos de mejorar su sistema de trabajo.

Tabla 14: Resultado de la variable Flexibilidad en el Desarrollo

Clase	Flexibilidad en el Desarrollo	Frecuencia
Tradicionales	0,17 a 0,44	21,70%
Semi Ágiles	0,45 a 0,72	69,60%
Ágiles	0,73 a 1	8,70%

Fuente: elaboración propia

Por otro lado, si relacionamos metodologías de desarrollo e innovación por ambos métodos, se observa que las empresas que aplican metodologías ágiles tienden a ser menos innovadoras en todos los niveles. En cambio, las que muestran un mejor resultado son las semi-ágiles con un buen desempeño en ambos enfoques.

Tabla 15: Innovación y Nivel de Agilidad

Enfoque	Clase	Tradicional	Semi Ágiles	Ágiles
Diferenciación	Innovación de Mejora	33,33%	66,67%	0,00%
Asimilación	Producto	25,00%	50,00%	25,00%
	Servicio	16,67%	50,00%	33,33%
	Comercialización	50,00%	50,00%	0,00%
	Organización	25,00%	75,00%	0,00%

Fuente: elaboración propia

En lo que se refiere a la correlación de las variables, se observa que la variable Tamaño tiene una relación positiva con ambos enfoques de innovación. Esto es coherente con la mayoría de la bibliografía especializada que encuentra que, a mayor tamaño mejores resultados en la variable innovación. En cuanto al enfoque de asimilación, el tamaño se correlaciona positivamente con las innovaciones de Servicios, Comercialización y Organización. Por su parte, la Innovación en Servicios se encuentra asociada con el Nivel de Agilidad, denotando que a mayor agilidad mayor servicio. No se encontró una relación entre las variables Tamaño y Exportación como sucede en otro tipo de industrias, ni entre agilidad y tamaño, que por definición, a mayor tamaño más rígidas son las metodologías. Bajo este contexto, se debe tener en cuenta que las empresas son de pequeño tamaño comparadas con las descripciones bibliográficas descritas en los trabajos donde se analizan empresas que operan a nivel global.

Tabla 16: Correlaciones entre las principales variables del estudio

	Antiguo	Exporta	Tamaño	Innovación Producto	Innovación Servicios	Innovación en Comercialización	Innovación en Organización	Indicador Agilidad	Índice de innovación
C. de Pearson	1	0,065	-0,061	-0,151	-0,001	0,268	-0,395	-0,078	-0,013
Sig. (bilateral)		0,77	0,782	0,491	0,995	0,216	0,062	0,723	0,952
C. de Pearson	0,065	1	0,291	-0,032	0,371	0,295	-0,25	0,099	-0,098
Sig. (bilateral)	0,77		0,177	0,886	0,082	0,171	0,251	0,653	0,656
C. de Pearson	-0,061	0,291	1	0,344	0,477	0,422	0,434	0,214	0,431
Sig. (bilateral)	0,782	0,177		0,108	0,021	0,045	0,039	0,328	0,04
C. de Pearson	-0,151	-0,03	0,344	1	0,398	0,099	0,387	0,205	0,307
Sig. (bilateral)	0,491	0,886	0,108		0,06	0,654	0,068	0,347	0,154
C. de Pearson	-0,001	0,371	0,477	0,398	1	0,519	-0,011	0,426	0,25
Sig. (bilateral)	0,995	0,082	0,021	0,06		0,011	0,959	0,043	0,25
C. de Pearson	0,268	0,295	0,422	0,099	0,519	1	0,265	-0,259	0,574
Sig. (bilateral)	0,216	0,171	0,045	0,654	0,011		0,221	0,233	0,004
C. de Pearson	-0,395	-0,25	0,434	0,387	-0,011	0,265	1	-0,099	0,382
Sig. (bilateral)	0,062	0,251	0,039	0,068	0,959	0,221		0,652	0,072
C. de Pearson	-0,078	0,099	0,214	0,205	0,426	-0,259	-0,099	1	-0,181
Sig. (bilateral)	0,723	0,653	0,328	0,347	0,043	0,233	0,652		0,41
C. de Pearson	-0,013	-0,1	0,431	0,307	0,25	0,574	0,382	-0,181	1
Sig. (bilateral)	0,952	0,656	0,04	0,154	0,25	0,004	0,072	0,41	

*La correlación es significativa al nivel 0,05 (bilateral).

Fuente: elaboración propia

5. Conclusiones

La relevancia del sector software en la provincia de Entre Ríos, evidencia un crecimiento incipiente de esta industria, en virtud de los resultados obtenidos en la muestra estudiada. Sobre todo a partir del dato de cantidad empresas de menos de 5 años de antigüedad que representan el 26% de la misma. En términos del tamaño de las firmas, se observa que hay un claro predominio de empresas que tienen entre 3 y 20 ocupados, las cuales representan el 91% de la muestra. Solo el 9% de las empresas tiene más de 20 empleados lo que se vincula a que en la provincia de Entre Ríos hay una mayor presencia de empresas pequeñas en comparación con otras zonas de producción del país como Córdoba, Santa Fe y Buenos Aires. En lo que respecta a la exportación de las firmas, la muestra se encuentra dividida casi a la mitad entre firmas orientadas al mercado interno y firmas orientadas al mercado externo. Los destinos de exportación son variados; entre los países que declararon que exportaban se encuentran Paraguay, México, Perú, Venezuela, Bolivia, Guatemala, Panamá, Estados Unidos, España y Francia (Blanc, et al, 2014).

En el apartado de metodologías de desarrollo de las firmas, se comprueba que hay una implementación de las modalidades tradicionales y se observan pocos casos de empresas ágiles. *Sin embargo, la situación dominante es la de empresas del tipo semi-ágiles*, esto se comprueba en las entrevistas realizadas en donde los participantes mencionaron metodologías pero especificaron que hacían *adaptaciones* dado que les era más útil a sus propósitos. Esto obliga a plantearse formas híbridas de organización del trabajo para las empresas de este tipo.

En el caso de la relación entre agilidad e innovación, otro de los objetivos del presente trabajo, se verifica una correlación positiva bajo el enfoque de asimilación en el apartado servicios, aunque no bajo el de diferenciación. El objetivo de contrastar los enfoques de medición de innovación en la muestra seleccionada, se observan *resultados claramente diferentes*. La medición del grado de innovación por el enfoque de diferenciación evidencia menor frecuencia de innovación que el enfoque de asimilación, logrando solo innovaciones incrementales. Este enfoque, aporta datos que permiten determinar con más certeza si hubo cambios en el software o servicios accesorios. A su vez, permite hacer análisis posteriores con partes del mismo, como por ejemplo: evolución de las interfaces, cambio en los lenguajes de programación, etc.; cuestiones que son totalmente imposibles con los datos recabados bajo el enfoque de asimilación. Por lo cual, a pesar de ser más lento al momento de la entrevista y requerir de un nivel elevado de capacitación en la temática software del entrevistador, la información relevada es de suma importancia para comprender los resultados y procesos de innovación sobre todo si se utilizan preguntas abiertas que pongan énfasis sobre la trayectoria sociotécnica de las firmas (Lepratte, Yoguel, Thomas, 2011).

Por su parte, las preguntas tradicionales de los manuales de innovación bajo el enfoque de asimilación, obtuvieron mejores resultados demostrando que hay innovación en todas las dimensiones tradicionales: productos, procesos, comercialización y organizacionales; lo cual es diferente al resultado del enfoque de diferenciación. El enfoque de asimilación presenta ventajas al momento de realizar la encuesta, ya que es mejor la tasa de respuestas dado la menor cantidad de preguntas y se necesita menos capacitación del encuestador en cuestiones relacionadas con software. Otra ventaja a tener en cuenta es que permite comparar este sector con otros tradicionales que utilizan el mismo sistema de medición.

Ambos enfoques tienen problemas para los nuevos esquemas de trabajo de las firmas de software que están tendiendo a ser proveedores especializados de diferentes clientes, los cuales realizan servicios o productos de acuerdo a la necesidad de los mismos; eliminando la idea de producto modular parametrizable con una serie de servicios que ha sido lo dominante en las antiguas industrias de software. Estos nuevos esquemas de trabajo dificultan determinar el grado de novedad, y definir qué es novedoso a nivel firma, ya que todos los proyectos son diferentes y únicos; pudiendo ser clasificados como un producto, una solución, una capacitación dependiendo la necesidad del cliente.

Esto pone de manifiesto las limitaciones de los enfoques cuantitativos usados para el estudio de los procesos de trabajo, generación de conocimientos, creación y/o adaptación de productos y servicios en este tipo de industrias sin incorporar perspectivas convergentes con otras metodologías de tipo cualitativas. Los aportes de los estudios organizacionales de corte constructivistas – sociotécnicos (Kallinikos, 2012) representan una alternativa complementaria interesante a los del tipo evolucionista neo-schumpeterianos orientados a sistemas complejos, en particular en el plano de análisis micro y meso.

Otra cuestión a profundizar que surge a partir de este tipo de estudios, es el papel que cumplen los sectores orientados a TIC en la transformación o generación de nuevos perfiles de especialización productiva, sobre todo cuando estos se caracterizan por empresas del tipo centrales de desarrollo (empresas que se dedican a escribir códigos a pedido de terceros, caracterizadas por el precio de su hora desarrollo) y en su extremo opuesto, por consultoría especializada (desarrollo de soluciones a medida para clientes que cuentan con capacidades de desarrollo, caracterizadas por las habilidades y trayectoria de sus recursos humanos). Una posible exploración de trabajos de investigación a futuro podría triangular enfoques provenientes de aportes de los estudios organizacionales al *estilo gobernanza a través de tecnologías* (Kallinikos, 2012), y *sistemas sociotécnicos* de producción e innovación (Lepratte, Yoguel, Thomas, 2011)

6. Bibliografía

Abrahamsson, P.; Salo, O.; Ronkainen, J.; Warsta, J. (2002). Agile software development methods—review and analysis.

Agile Manifesto (2001). <http://www.agilemanifesto.org/>

Akman, GülSen y Yilmaz, Cengiz (2008). Innovative capability, innovation strategy and market orientation: an empirical analysis in turkish software industry, *International Journal of Innovation Management*, 12.

Aramand, M. (2008). Software products and services are high tech? New product development strategy for software products and services, *Technovation*, vol 28, pp. 154-160

Archibugi, D., Cohendet, P., Kristensen, A., Schäffer, K.-A., (1994). Evaluation of the Community Innovation Survey (CIS). Phase 1, EIMS Publication, p. 11.

Barletta, F.; Pereira, M., Robert, V. y Yoguel, G. (2012). *Capacidades, vinculaciones, y performance económica. La dinámica reciente del sector de software y servicios informáticos argentino*. 41 JAIIO - SSI

Beck, K. y Fowler, M. (2001). *Planning Extreme Programming*, Addison-Wesley/Pearson, Upper Saddle River, NJ, USA.

Beck, K., Andres, C. (2004). *Extreme Programming Explained: Embrace Change*. Second Edition. Addison Wesley Professional, Boston, November.

Blanc, R.;Lepratte, L; Hegglin, D y Pietroboni, R (2014).*Empresas de software en Entre Ríos: innovación, exportación, calidad y empleo*. XIX Reunión Anual Red Pymes Mercosur PYMEs, Innovación y desarrollo.

Boehm, B. (1988). A Spiral Model of Software Development and Enhancement, *IEEE Computer*, Vol. 21, No. 15 May, pp. 61-72.

Boehm, B., Turner, R. (2004). *Balancing Agility and Discipline—A Guide for the Perplexed*, Addison-Wesley, Boston, MA, USA.

Borello, J; Erbes, A; Robert, V; Roitter, S y Yoguel, G (2005). Competencias técnicas de los trabajadores informáticos. El caso de Argentina. *Revista de la Cepal*, 87.

Burger-Helmchen T. y Cohendet P. (2011). User communities and social software in the Video Game Industry. *Long Range Planning* 44 (2011). pp 317 – 343.

Cockburn, A. (2006). *Ágile Software Development: The Cooperative Game* (2nd Edition)., Addison-Wesley Professional.

Cohen y Levinthal (1989). Innovation and learning: The two faces of R&D, *The Economic Journal*, Volume 99, September pg. 569-596.

Canós, J.; Leteliery, P. y Penadés, C. (2003). Metodologías Ágiles en el Desarrollo de Software. DSIC -Universidad Politécnica de Valencia.

Coombs, R., Miles, I. (2000). Innovation, measurement and services: the new problematic In: Metcalfe, J.S., Miles, I. (Eds.), *Innovation Systems in the Service Economy. Measurement and Case Study Analysis*. Kluwer Academic Publishers, Boston, pp. 85–103.

Coriat, B. (1992). *Pensar al revés; Trabajo y organización en la empresa japonesa*. México/España, Siglo XXI.

De Carvalho, L. (2003). Planejamento da Alocação de Recursos Humanos em Ambientes de Desenvolvimento de Software Orientados à Organização, PhD Thesis, Universidade Federal do Rio de Janeiro, COPPE, Brazil.

Den Hertog, P y Rubalcaba, L (2010). Policy frameworks for service innovation: a menu-approach. En *The handbook of innovation and services A Multi-disciplinary Perspective*, Edited by FaizGallouj y FaridahDjellal Edward Elgar Publishing, Inc

Djellal, F., Gallouj, F. (2000). *Innovation surveys for service industries: a review*. Paper presented at the DG Enterprise Conference on Innovation and Enterprise Creation: Statistics and Indicators. Sophia Antipolis, France, November.

Djellal F.; Gallouj, F y Miles, I. (2013). Two decades of research on innovation in services: Which place for public services? *Structural Change and Economic Dynamics*. Vol. 27, pages 98-117.

Dossani, R (2005). Origins and Growth of the Software Industry in India, Asia-Pacific Research Center, Stanford University.

Drejer, I (2004). Identifying innovation in surveys of services: a Schumpeterian perspective. *Research Policy* 33, pp 551–562.

Edison, H.; BinAli, N y Torkar, R (2013). Towards innovation measurement in the software industry. *The Journal of Systems and Software* 86 (2013). pp. 1390 - 1407.

Erbes, A; Roitter, S y Delfini, M (2011). Organización del trabajo e innovación: un estudio comparativo entre tramas productivas argentinas. *Revista Economía: Teoría y Práctica*, N° 34, enero-junio, pp. 101-132. Universidad Autónoma Metropolitana, México. ISSN: 0188-8250.

Fairhurst, E. (2011). *Innovation in public health care: Diabetes education in the UK en Handbook of Innovation and Services: A Multi-disciplinary Perspective*. Edward Elgar Publishing Ltd.

Floyd, C. (1984). A Systematic Look at Prototyping. In Budde, R., Kuhlenkamp, K., Mathiassen, L. y Züllighoven, H. (Eds.), *Approaches to Prototyping*, pp. 1-18

Franke, N. y Von Hippel, E. (2003). Satisfying heterogeneous user needs via innovation toolkits: the case of Apache security software. *Research Policy* 32 pp 1199 – 1215

Gallouj, F. y Savona, M. (2009). Innovation in services: a review of the debate and a research agenda. *Journal of Evolutionary Economics*, 19, pp. 149–172.

Gallouj, F y Toivonen, M (2011). Elaborating The Characteristics-Based Approach To Service Innovation: Making The Service Process Visible. *Journal of Innovation Economics*. Nro 8 (2).

Gallouj F y Weinstein O. (1997). Innovation in services. *Research Policy*, vol. 26, pp. 537-556.

Gomaa, H. (1983). The Impact of Rapid Prototyping on Specifying User Requirements, ACM SIGSOFT. *Software Engineering Notes* 8, 2, 17-28.

Gorla, N y Wah, Y. (2004). Who should work with whom?: building effective software project teams. *ACM*.47 (6).pp 79–82.

Hartman, D. y Lindgren, J. (1993). Consumer Evaluations of Goods and Services - Implications for Services Marketing, *Journal of Services Marketing*.

Hertog P, Gallouj F y Segers, J. (2009). Measuring innovation in a ‘low-tech’ service industry: the case of the Dutch hospitality industry, *The Service Industries Journal*, vol. 31(9)., pages 1429-1449, November.

IEEE (2004). Guide to the Software Engineering Body of Knowledge-SWEBOK IEEE Computer Society.

Johansson, C y Bucanac, C (1999). *The V-Model IDE*, University Of Karlskrona, Ronneby.

Jong, J. y Von Hippel, E. (2009). Transfers of user process innovations to process equipment producers: A study of Dutch high-tech firms. *Research Policy* 38 pp. 1181–1191.

Kallinikos, J. (2012). Form, function and matter: crossing the border of materiality. En: Leonardi, Paul M. and Nardi, Bonnie A. and Kallinikos, Jannis, (eds.) *Materiality and Organizing: Social Interaction in a Technological World*. Oxford University Press, Oxford, UK, 67-87. ISBN 9780199664061.

Lepratte, L.; Thomas, H.; Yoguel, G. (2011). Sistemas Sociotécnicos, innovación y desarrollo. WP 1-2011 GIDIC UTN FRCU. http://mpira.ub.uni-muenchen.de/33559/1/Working_Paper_2011_Lepratte_UTN_FRCU.pdf

Manual De Bogotá (2001). Normalización de Indicadores de Innovación Tecnológica en América Latina y el Caribe. Red Iberoamericana de Indicadores de Ciencia y Tecnología (RICYT). / Organización de Estados Americanos (OEA). / Programa CYTED COLCIENCIAS/OCYT.

Manual de Oslo (2006). Guía Para La Recogida E Interpretación De Datos Sobre Innovación publicación conjunta de OCDE y Eurostat, Tercera edición.

Mills, H (1988). *Software Productivity*, Editado Dorset House.

Motta, J; Zavaleta, L; Llinás, I; Luque, L y Jones, C (2012). *Procesos de Innovación y Competencias de los Recursos Humanos en la Industria Del Software En Argentina*.

Nambisan, S. (2002). Software firm evolution and innovation – orientation. *J. Eng. Technol. Manage* 19 pp 141 -165.

Nelson, R. y Winter, S. (1982). *An Evolutionary Theory of Economic Change*, Harvard University Press, Cambridge.

Nelson, R. (1995). Recent evolutionary theorizing about economic change, *Journal of Economic Literature*, Vol. 33.

PMBOK (2004). Guía de los Fundamentos de la Dirección de Proyectos. Tercera Edición. Project Management Institute, Four Campus Boulevard, New town Square, PA 19073-3299 EE.UU.

Qumer, A., Henderson-Sellers, B. (2007). An evaluation of the degree of agility in six agile methods and its applicability for method engineering. *Information and Software Technology*.

Royce, W (1970). Managing the Development of Large Software Systems. *Proceedings of IEEE Westcon*.

RUP (2002). Rational Unified Process. Version 2002.05.00. Rational, Software Corporation,.

Saviotti P. y Metcalfe J. (1984). A theoretical approach to the construction of technological output indicators, *Research Policy* 13, pp. 141-151.

Schumpeter, J (1934). *The Theory of Economic Development*. McGraw-Hill, Nueva York.

Schwaber, K y Beedle, M (2001). *Ágile Software Development with Scrum*, Prentice Hall.

Segelod, E. y Jordan, G. (2004). The use and importance of external sources of knowledge in the software development process. *R&D Management*, 34, pp. 239-52.

Teece, D. (1992). Competition, cooperation and innovation. *Journal of Economic Behavior and Organization*, 18, pp 1–25.

Van der Have, R., Toivonen, M., y Tuominen, T. (2007). *Dimensions of Service Innovation*. Presented at the ISMOT 2007 conference “Managing Total Innovation and Open Innovation in the 21st Century,” Hangzhou, China.

Williams, L (2010). Agile Software Development Methodologies and Practices. *Advances in Computers*, Vol. 80 (1).

Wolak, R.; Kalafatis S. y Harris P. (1998). An Investigation Into Four Characteristics of Services. *Journal of Empirical Generalizations in Marketing Science*, Volume Three.