

# ASPECTOS DINÁMICOS EN LA EVALUACIÓN DE LOS PROYECTOS DE SOFTWARE

Luis Moreno

*Universidad de Sevilla*

## Abstract

Software Project have proven challenging to plan, manage and improvement by means of System Dynamics methodology because project conditions evolve over time as a result of feedback responses, many involving non linear relationships. Also, assessment of software projects are limited when detailed dynamic structure information is not available or considered.

At this paper a study of several dynamic structures involving in software projects are realized in order to help in assessing the software projects and improvement the software process.

**Keywords:** *Modelling, Software process improvement, Project simulation, Software Engineering*

## Resumen

La gestión, planificación y mejora en los proyectos de software han sido una constante en la investigación a través de la metodología de Dinámica de Sistemas, debido fundamentalmente a que las condiciones del proyecto de software evolucionan como resultado a las respuestas a los bucles de realimentación, involucrando relaciones no lineales en gran medida. Por otra parte, la valoración de los Proyectos de Software va a estar limitada, si no se dispone de la información dinámica detallada de su estructura.

En este documento se realiza un estudio de las diversas estructuras dinámicas involucradas en los proyectos de software con objeto de ayudar en la valoración de los proyectos de software y favorecer la mejora de los procesos involucrados.

**Palabras clave:** *Modelado, Mejora de procesos de software, Simulación de proyectos, Ingeniería de Software*

## 1. Introducción

Valorar los proyectos de software requiere mucho más que totalizar estimaciones estáticas de proyectos exitosos y proyectarlas o adecuarlas a situaciones presentes, o a normas prescriptivas. Los proyectos de software, como un sistema complejo, tienen dinámicas inherentes que gobierna la evolución de su rendimiento en el tiempo. De ahí que una evaluación de su rendimiento basado en relaciones estáticas, sea de uso limitado. Por lo que los gestores de decisiones deberían ser capaces de valorar sus dinámicas y comprender los efectos de los bucles de realimentación generados en los diferentes subsistemas y elementos que lo conforman, y cuáles pueden operar en diferentes contextos,

con una variedad de objetivos, en orden a valorar el rendimiento en el ciclo de vida del desarrollo de software.

La dinámica de sistemas es una aproximación adecuada para el estudio del comportamiento de los sistemas a este nivel de análisis, especialmente, cuando exhiben complejidad como resultado de las interrelaciones entre sus elementos; complejidad, que dificulta la capacidad para gestionar y determinar el alcance de las decisiones.

El modelado de sistemas dinámicos se concentra en el estudio de los bucles de realimentación como conductores del rendimiento del sistema, y provee guías para decisiones.

La contribución de este documento se dirige en esa dirección. Así como otras aportaciones ven el rendimiento del proyecto como una adecuación a unos procesos de referencias, y normas (punto de vista normativo), nuestro objeto es desde la perspectiva, contexto y escenario en el que se desarrolla el proyecto. Esto es, valorar los diferentes sistemas dada su estructura organizacional y sus dinámicas inherentes y ayudar a desarrollar una estrategia para mejorar el control sobre el proyecto de software según sus cargas específicas, ilustrando estos puntos a través del estudio de importantes estructuras en la literatura.

En los siguientes apartados se expondrá los objetivos y metodología de nuestro trabajo, se realizará una revisión y comparativa de las estructuras dinámica relevantes de los modelos dinámicos aplicados a la Ingeniería de Software, seguirá unas secciones dedicadas a diversos aspectos dinámicos en los proyectos, y finalmente, se expondrán las conclusiones.

## **2. Objetivos**

El objetivo principal del trabajo es explorar las interrelaciones endógenas en los proyectos y contribuir a la mejora de la gestión de los Proyectos de Software por medio del modelado dinámico, utilizando técnicas de dinámica de sistemas, modelos previos y una revisión de la literatura, con objeto de proveer un marco de comprensión de los problemas y cuestiones en los proyectos de software; así como valorar y optimizar las estrategias y toma de decisiones desde un punto de vista sistemático.

Nuestro propósito es evaluar los diversos aspectos dinámicos relevantes en los proyectos de software y sus interrelaciones dinámicas. Y de ahí, las condiciones de alineación con la organización, y ayudar a desarrollar estrategias y planes de acción para mejorar el rendimiento de los proyectos de software.

## **3. Metodología**

En este trabajo nos basamos en la metodología de Sistemas Dinámicos, Pensamiento de Sistemas (Thinking Systems), así como los estándares de Mejoras y Procesos Software, y modelos dinámicos previos para exponer cuestiones planteadas en los objetivos. Para ello, se ha realizado una revisión de la literatura para poder situar el estado actual y cuestiones de las dinámicas de software y se ha analizado las estructuras dinámicas que aparecen en los proyectos de software.

## **4. Estructuras fundamentales en las dinámicas de los proyectos**

En la valoración y desarrollo de las estructuras dinámicas de los proyectos nos basamos en un estudio en profundidad realizado por Lyneis y Ford (2007) sobre las dinámicas de los proyectos. Detalles y definiciones han sido contemplados para establecer las bases de nuestro marco de referencia para proyectos de software.

Las estructuras fundamentales que integran los modelos de proyectos se clasifican en cuatro grupos (Lyneis y Ford, 2007):

- **Factores Característicos de los Proyectos:** Representan los componentes principales encontrados en los sistemas de proyectos como son la colección de tareas para ser ejecutadas en paralelo y en serie, y la aplicación de los recursos a la gestión de flujos en el desarrollo de los procesos basados en la percepción de las condiciones del proyecto. En la representación de estos factores se explota la capacidad de los modelos dinámicos para representar la toma de decisiones, tales como el modelado guiado por desviaciones (*driven by gaps*), retrasos en procesos humanos, relaciones no lineales, y ajustes basados en predicciones (*forecasting*).
- **El ciclo de Re-trabajo:** Uno de los elementos más importantes en los modelos dinámicos de proyectos. Su estructura es recursiva, en donde más re-trabajo genera más re-trabajo, creando comportamientos problemáticos que hacen extender la duración del proyecto, y son fuentes de muchos retos para los gestores de proyectos. Diversos modelos de estructuras del ciclo de Re-trabajo difieren en el número de *stocks*. Los errores no son inmediatamente reconocidos y son detectados como resultado del trabajo sucesivo, o tareas de *testing*. Este re-trabajo descubierto, después de un retraso, requiere la aplicación de un esfuerzo adicional. Asimismo, algún ítem de re-trabajo fluye a través del ciclo de re-trabajo más de una vez. Este ciclo es central en la dinámica adversa del proyecto.
- **Control del Proyecto:** Lo componen los mecanismos de realimentación que los gestores del proyecto utilizan para procesar la información y las acciones para corregir las desviaciones entre objetivos establecidos y situación actual. Los métodos básicos disponibles en los modelos para ajustar estas desviaciones son: (1) mover el comportamiento del modelo más cerca de los objetivos, ó (2) mover los objetivos hacia el comportamiento del proyecto. Estos métodos usan bucles de control negativo, con respuestas proporcionales a las desviaciones; sin embargo, existen límites impuestos en cuanto al tamaño y rapidez de los ajustes, y ambos métodos comportan costes. A veces, los objetivos se establecen para fechas futuras, por lo es necesario incluir un sistema para gestionar la predicción del rendimiento. Las condiciones percibidas y actuales, junto con la combinación del sistema de informe del proyecto, que frecuentemente sobreestima el progreso en las primeras etapas del proyecto y lo subestima en las finales, generaran dinámicas adversas en la forma de efectos "ripple". Asociado al control del proyecto está la *estabilidad del proyecto* y cómo resiste el proyecto a las acciones disruptivas en las actividades que están bajo condiciones de incertidumbres.
- **Efectos adversos Onda y Repercusiones** (*Ripple and knock-on effects*). En los proyectos, generalmente, a pesar de haberse realizado los controles establecidos con objeto de mejorar su rendimiento y adecuación, aparecen comportamientos adversos. Estos comportamientos se puede explicar por medio de unos mecanismos de realimentación reactivos (bucles de realimentación positivos). Esto es, la resistencia a las políticas (ripple) para controlar las desviaciones del rendimiento del proyecto y las consecuencias no intencionadas (knock-on). Estos comportamientos adversos se suelen manifestar con un cierto retraso, y aunque muchos de ellos son internos al proyecto, otros se extienden a su entorno, pudiendo amplificar o iniciar nuevas dinámicas adversas.

La figura 1 (Lyneis y Ford, 2007) ilustra los impactos primarios de los bucles de control (color verde) en el re-trabajo y productividad (efectos *Ripple*, en color rojo). De esta manera, las acciones para intentar corregir los desequilibrios entre el cumplimiento de la evolución del proyecto y los objetivos tienen efectos no intencionados, que generan *resistencia a las políticas* a través de bucles positivos. A su vez los efectos *Ripple* generan bucles de realimentación secundarios y terciarios (efectos *knock-on*, en color morado). Algunos son consecuencia de los procesos físicos que relacionan el flujo de trabajo, mientras otros se deben a reacciones “humanas” a las condiciones del proyecto. Estos efectos reactivos pueden generar significativa dinámica perjudicial para el proyecto, y se manifiestan a través de relaciones no lineales. También se ha representado un ajuste de objetivo (color naranja).

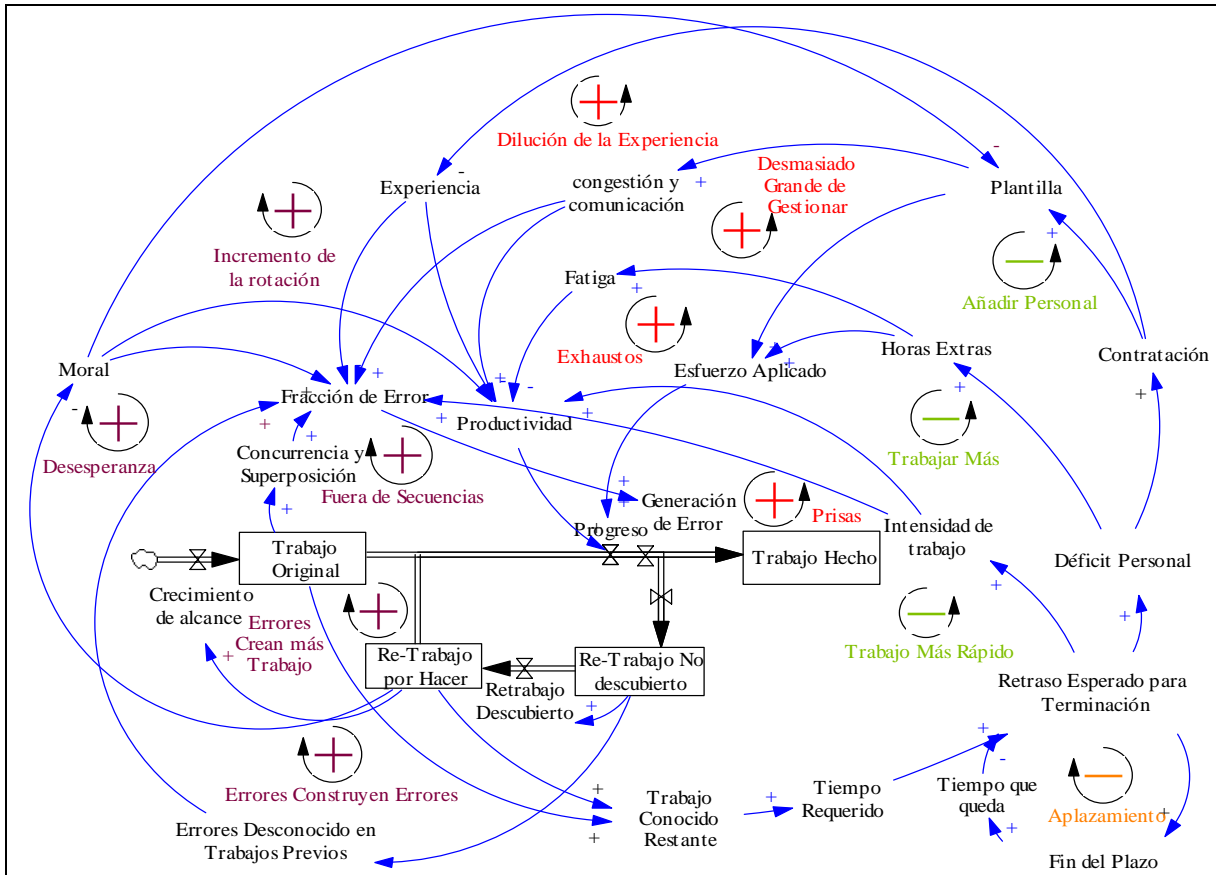


Figura 1. Resistencia a las políticas de control vía efectos *Ripple* y *Knock-on* (Lyneis y Ford, 2007)

Aunque se ha indicado previamente las consecuencias al ejercer una acción de control específica a través de políticas de personal para cumplir los plazos del proyecto, es de señalar la complejidad derivada si se efectúa otro control o ambos simultáneamente. Lo que indica la idoneidad de la metodología de dinámica de sistemas para describir este tipo de interacciones y sus complejidades.

Otros efectos *Ripple* y *Knock-on* (no asociado a controles) aparecen como resultado de cambios de requisitos, sobre todo en las últimas etapas del proyecto, dando lugar a *Scope Creep* del proyecto. También los errores que no han sido detectados en el control de calidad y cambios no previstos u obligados, pueden ocasionar este tipo de reacción. Estos cambios en el alcance del proyecto software ocurren con frecuencia debido a requerimientos no descubiertos que llegan a ser aparentes en la fase de codificación o ejecución.

Desde una perspectiva de sistemas realimentados, el sistema permanece estable en la medida en que los bucles de realimentación que controlan el progreso dominen. Pero en los

proyectos donde hay un cambio de dominancia hacia bucles que *añaden trabajo*, se puede dar lugar a un deterioro en la ejecución del proyecto con la aparición de *tipping points*, y las consecuencias negativas del dominio de ciertos bucles de realimentación (feedbacks) (Taylor y Ford, 2006).

Aunque el estudio anterior se refiere a una gestión operativa del proyecto, es de destacar la dificultad en el contexto de complejidad organizacional e incertidumbre. Así, en el entorno multiproyecto, Yaghootkar y Roos (2007) sugirieron que las ineficiencias en el desarrollo de los proyectos son consecuencia de los efectos sistémicos originados por la complejidades de las interacciones de los factores que intervienen; concluyendo, que una utilización adecuada del modelo dinámico generará una conciencia holística primordial en la toma de decisiones.

## 5. Valoración de los modelos de proyectos de software

La investigación y aplicación de la dinámica de proyectos se ha dirigido hacia la comprensión de los factores y causas que ocasionan los incumplimiento de objetivos (sobre-costes, extensión del plazo de finalización del proyecto, etc.), así como de las acciones para evitarlas o aminorarlas, siendo la dinámica de sistemas un método científico para la valoración de lo que fue bien o mal en un proyecto y así mismo, proveer los elementos necesarios para investigar: (1) las valoraciones post-proyecto para disputas y aprendizaje; (2) estimación de proyectos y valoraciones del riesgo; (3) gestión del cambio, gestión del riesgo y control del proyecto; y (4) gestión de entrenamiento y educación (Lyneis y Ford, 2007).

Con base en los numerosos factores que intervienen en el proyecto y la complejidad de sus interrelaciones, no evaluamos el éxito o fallo del proyecto, primariamente, en la capacidad de éxito de estos factores (existen numerosos estudios empíricos en la literatura referentes a los criterios o causas raíces de éxito (Reel, 1999)); nuestro enfoque se dirige en las condiciones en que éstos intervienen en las dinámicas generadas a través de los bucles de realimentación; de esta manera, hemos reducido la incertidumbre al situar el alcance causal de cada factor, lo que delega a la capacidad de formalización del modelo para la extracción de conclusiones.

La mayoría de los modelos de proyectos software en la literatura estudiada capturan los factores importantes de los proyectos de desarrollo, clásicamente divididos en subsistemas que lo integran junto sus límites, que delimitan su propósito y objeto. Estos modelos reflejan la capacidad cognitiva y de decisión, y significa una simplificación para razonar con un propósito específico. Están vinculados al análisis causal y de ahí su idoneidad para el análisis de las *causas raíces* (Root cause). Un amplio estudio sobre la aplicación y contribución de la dinámica de sistemas a los Sistemas de Información y procesos de software puede verse en (Georgantzas y Katsamakos, 2007), (Madachy, 2008) y (Zhang *et al.*, 2008).

Una valoración de las estructuras dinámicas va a ayudar a entender los procesos y sus capacidades, y comprender los comportamientos emergentes que ocurren en los proyectos, a destacar:

1. Las características y complejidad del ciclo de re-trabajo y nivel de detalle de las fases de trabajo.
2. Los bucles de realimentación de control y los efectos *ripple* (onda o rizo) y *knock-on* (repercusiones secundarias adversas). Y sus consecuencias entre fases, o por cambios en procesos.
3. Efectos de realimentación representados.

4. Formulación de procesos que han sido desarrollados.
5. Mecanismos de ajustes y consecuencias secundarias de ajustes de objetivos, ya señalado en Lyneis y Ford (2007), a través de bucles de realimentación positivos.
6. Latencias de cambios (Park y Peña-Mora, 2003) y errores, para provisión de búferes (Lee y Peña-Mora, 2007).
7. Bucles de Realimentación.

## 6. Aspectos Dinámicos

En nuestro trabajo nos basaremos en los bucles de realimentación ligados a procesos, decisiones, control y operación, como clave para el conocimiento y valoración del proyecto. De acuerdo con los apartados precedentes, a toda acción de control en el proyecto, le corresponde una reacciones primarias y secundarias adversas. Cabe considerar asimismo, para una acción disruptiva en el desarrollo del proyecto, qué tipo de reacciones primarias y/o secundarias favorables habría que aplicar. La importancia al considerar el punto de vista de realimentación es que habitualmente, en la guía de decisiones, se adopta una perspectiva basada en acontecimientos o punto de vista de causalidad en bucle abierto, ignorando los procesos realimentados, los fallos en apreciar los retrasos en las acciones y respuestas, así como en la información, junto con la dificultad de comprender los flujos y niveles e insensibilidades a las no linealidades (Sterman, 2000). Estudios adicionales a través de métodos formales intentan explicar cómo la estructura da lugar al comportamiento, así como la participación de los distintos bucles de realimentación (Ford, 1999; Güneralp, 2006; Gonçalves, 2009).

A continuación se realiza un estudio de diversos aspectos dinámicos a considerar en los Proyecto de Software:

### 6.1 Estimación Inicial del Proyecto

La valoración de la estimación inicial del proyecto es crítica por su influencia en el rendimiento del proyecto, ya que diferentes estimaciones crean un proyecto diferente, así como decisiones y percepciones que afectan al desarrollo del ciclo de vida (Abdel-Hamid y Madnick, 1991). Posibles efectos, como el *síndrome del 90%*, surgen por causa de una subestimación inicial en el tamaño o recursos, junto con imprecisas medidas de progreso y sus interacciones.

En su determinación se han utilizado los modelos algorítmicos (COCOMO) para proveer los valores iniciales base, ajustándose posteriormente a través de la simulación del modelo dinámico; estas técnicas fueron utilizadas para realizar estimaciones de esfuerzos y tiempos en proyectos de software en (Abdel-Hamid y Madnick (1991). Además de esta estimación, *up-front*, se recomienda realizar estimaciones durante el proyecto para detectar cualquier grado de subestimación lo más temprano posible; y después del proyecto, con objeto de valorar el coste del proyecto si otras decisiones se hubiesen tomado. Con lo cual, la estimación ajustada por medio de los modelos dinámicos pueden ayudar a evitar los efectos dinámicos adversos ocasionados, una vez detectada esa subestimación inicial del proyecto. Lo anterior puede extenderse a los proyectos sobreestimados, por los efectos resultantes de una presión de trabajo deficiente (Lyneis y Ford, 2007).

Combinación de técnicas estáticas y dinámicas también fueron utilizadas para la estimación dinámica del rendimiento del proyecto por Choi y Bae (2009).

## 6.2 Comportamientos y Progreso

El progreso adolece visibilidad, especialmente en las etapas iniciales del proyecto (Abdel-Hamid y Madnick, 1991), y se valora en términos de resultados producidos o esfuerzo empleado. Como medida frecuentemente utilizada está la *fracción de trabajo completada* para describir el progreso en los proyectos reales (Lyneis y Ford, 2007). A través de la evolución de ésta variable, se puede explicar interesantes comportamientos del proyecto, especialmente en las últimas etapas, y su relación con los mecanismos de control tratados. Existiendo una relación entre la evolución de ésta y la activación de bucles de realimentación que los gestores utilizan, junto con la subestimación el alcance de las tareas a realizar, trabajo no-descubierto y disminución de la productividad.

La medida imprecisa del progreso viene dada normalmente por medidas *sustitutos*, de aquí la importancia de métricas adecuadas y mediciones más precisas.

## 6.3 Comunicación, cooperación y coordinación

Entre las tres facetas de un buen proyecto, especialmente en los entornos de equipo virtuales, está la comunicación, cooperación y coordinación. Vinculado a ello está el subsistema de recursos humanos para equipos virtuales, ofreciendo una referencia importante (Galindo *et al.*, 2006).

## 6.4 Valor Ganado

Teniendo en cuenta la necesidad de conocer el alcance de proyecto real, y con ello los *stocks* de re-trabajo y cambios generados, para un mejor ajuste en la aplicación de las metodologías de valor conseguido, los modelos dinámicos proporcionan a través de su ciclo de re-trabajo los elementos adecuados para ello. Janamanchi y Burns (2005), empleando estructuras básicas, modelaron la gestión de los indicadores de valor ganado bajo diferentes escenarios, equipo de proyecto variable en el tiempo, productividad variable, sobrecarga en comunicación y gestión de cambio.

## 6.5 Cambios y Errores

Uno de los factores que más alteran el rendimiento de los proyectos de software son los cambios (requisitos, tecnología, procesos, cambios latentes) y errores (en el diseño, código, errores no detectados) y sus consecuencias desfavorables: impactos entre fases, coordinación y efectos *ripple*. Aunque se conoce la “fuerza” que puede tener el impacto a través de simulación (Taylor y Ford, 2006; Lyneis y Ford, 2007), el mecanismo de realimentación y la cuantificación para proyectos de software necesitan ser estudiados. Referencias destacadas son (Park y Peña Mora, 2003) y (Lee y Peña-Mora, 2007).

## 6.6 Simulación y Mejora de Procesos

El área de los Procesos Software y Mejora de Procesos, se ha caracterizado por su constante evolución en orden a cubrir los diversos tipos, tamaños y complejidades del software desarrollado. A pesar de los beneficios que conlleva la implantación de iniciativas de mejora, se encuentran dificultades para mantener sus planes de mejora, aún con éxitos en implantaciones iniciales, e incluso han tenido efectos no deseados. Este problema se ha reconocido como “*Improvement Paradox*”, y se ha sugerido que la inhabilidad para tratar el programa de mejora como un proceso dinámico es el principal determinante del fallo del programa (Keating *et al.*, 1999).

Asimismo, modelos de dinámica de sistemas han analizado la denominada “*paradox productivity*”. De esta manera, Abdel-Hamid (1996) observó que el resultado en la aplicación

de las mejoras de proceso a través de herramientas CASE no lograba las mejoras esperadas en la productividad, debidas fundamentalmente, a las complejidades inherentes del sistema en que se implementa.

Muchos resultados de las decisiones inadecuadas han sido reproducidos en modelos de proyectos. La racionalidad es limitada (*bounded rationality*): Los agentes toman decisiones empleando la rutina y heurística, porque la complejidad del entorno excede su capacidad para optimizar, aún con la poca información de que disponen (Sterman *et al.*, 2007).

La mejora de los procesos es una de las utilidades de los modelos de simulación (Ruiz *et al.*, 2004, Madachy, 2008). Al considerar las dinámicas internas y sus interrelaciones, y ver como afectan los cambios de los parámetros de los procesos en el rendimiento del proyecto, se tiene un papel proactivo del modelo dinámico, que puede ser complementado con la adherencia a normas de referencia de proceso de madurez en las organizaciones, a través de implantación de mejores prácticas (Deek *et al.*, 2006).

### **6.7 Elicitación de Bucles de Realimentación**

Los métodos de participación para construcción de modelos dinámicos en grupo (Luna-Reyes *et al.*, 2006; Winz y Brierley, 2007) y las entrevistas con los involucrados (Yaghootkar y Roos, 2007) para elicitar los mecanismos o procesos dinámicos reactivos con objeto de mejorar la reacción del entorno, serán una herramienta eficaz para conocer e intentar evitar las dinámicas adversas que se generan en los proyectos de Software, especialmente las que afectan al re-trabajo (Reichelt y Lyneis, 1999).

## **7. Conclusiones**

La práctica de la gestión de proyectos está basada, en gran medida, en relaciones causales de bucle abierto y enlaces únicos lineales, reducidas frecuentemente a guías o reglas prácticas que vienen determinadas, principalmente, por las normas y estándares de calidad en los proyectos. Muchas de las decisiones en los proyectos se basan en predicciones o suposiciones sobre un tiempo determinado, ignorando, en gran medida, la capacidad de reacción del entorno en que se desarrolla el proyecto por la complejidad intrínseca del mismo, ampliamente señalado en la literatura estudiada.

Al capturar las causas de las decisiones y sus efectos, y los mecanismos de realimentación en que están involucrados los factores determinantes del proyecto, los modelos de dinámicas de sistemas van a identificar los escenarios en que se desarrolla el proyecto y van a proporcionar advertencias tempranas del impacto de estas acciones.

Sin embargo, nuevos métodos, técnicas y herramientas para determinar esas predicciones y encajar los distintos factores dinámicos, entre ellas, el aprendizaje organizacional, lecciones aprendidas de proyectos pasados y gestión del conocimiento, van a intentar aliviar muchos de los inconvenientes en aras a establecer los mecanismos de control adecuados.

Pocos estudios cualitativos proporcionan un soporte a la vinculación entre bucles de realimentación, selección de factores de éxito o estabilidad del proyecto y rendimiento del proyecto. El modelado explícito de estos bucles de realimentación aportará grandes beneficios a la dinámica de los proyectos de software.



## Referencias

Abdel-Hamid, T. and Madnick, S., "Software Project Dynamics: An Integrated Approach", Prentice Hall, Englewood Cliffs, NJ. 1991.

Abdel-Hamid, T. (1996). The slippery path to productivity improvement. *IEEE Software*, 13(4), pp. 43–52, 1996.

Aracil, J., "Introducción a la dinámica de sistemas". Alianza Editorial Textos, 1986.

Black, L. Reppenning, N., "Why firefighting is never enough: preserving high-quality product development", *System Dynamics Review* 17(1), 2001, pp.33– 62.

Boehm B. 1981, "Software Engineering Economics", Prentice Hall, Upper Saddle River, New Jersey, USA.

Boehm B., Horowitz E., Madachy R., Reifer D., Clark B.K., Steece B., Brown A.W., Chulani S., Abts C. 2000, "Software Cost Estimation with COCOMO II", Prentice Hall, Upper Saddle River, New Jersey, USA.

Chrissis M.B., Konrad. M, Shrum S., 2003. "CMMI: Guidelines for Process Integration and Product Improvement", Addison-Wesley: Boston, MA, USA, 2003.

Choi, K., Bae D-H. "Dynamic project performance estimation by combining static estimation models with system dynamics", *Information and Software Technology*, Vol. 51 pp. 162-172, 2009.

Deek F.P., McHugh J.A.M. and Eljabiri O.M, "*Strategic Software Engineering: An Interdisciplinary Approach*", Taylor y Francis, 2005.

Espinosa, J.A., Carmel E., "The impact of time separation on coordination in global software teams: a conceptual foundation". *Software Process Improvement and Practice*, 2003, 8(4): pp. 249–266.

Ford D. and Sterman J., "Expert knowledge elicitation to improve formal and mental models", *System Dynamics Review*, Vol. 14(4), 1998a, pp. 309-340.

Ford D, Sterman J, "Dynamic modeling of product development process", *System Dynamics Review*, Vol. 14(1), 1998b, pp. 31-68.

Ford D. 1999. A behavioral approach to feedback loop dominance analysis. *System Dynamics Review* 15(1), pp. 3–36, 1999.

Galindo J., Mazzuchi ,T. and Sarkani, S. (2006), "Virtual Software Project Dynamics - The Human Resource Management Sector", *The 24th International Conference of the System Dynamics Society*, July 23-27, 2006 Nijmegen, The Netherlands.

Georgantzas, N. and Katsamakas, E., "Information systems research with system dynamics", *System Dynamics Review*, Vol. 24 (3), pp. 247-264, 2008.

Gonçalves P., "Behavior modes, pathways and overall trajectories: eigenvector and eigenvalue analysis of dynamic systems", *System Dynamics Review* Vol. 25 (1), pp., 35-62, 2009.

Güneralp B. "Towards coherent loop dominance analysis: progress in eigenvalue elasticity analysis", *System Dynamics Review Vol. 22(3)*, pp. 263-289, 2006.

ISO/IEC 15504-5. 2006. Information Technology – Process Assessment – Part 5: An Exemplar Process Assessment Model.

Jacobs D. Accelerating Process Improvement Using Agile Techniques. Crosstalk, March 2004.

Janamanchi, B. and Burns, J.R., "Project Dynamics with Applications to Change Management and Earned Value Tracking", *Conference Proceedings The 23rd International Conference of the System Dynamics Society, July 17-21, 2005 Boston*.

Keating, E., Oliva R., Repenning N., Scott R. and John Sterman, J., "Overcoming the Improvement Paradox", *European Management Journal Vol. 17, No. 2*, pp. 120-134. 1999.

Kellner, M.I., Madachy, R.J., Raffo, D.M., "Software process simulation modeling: Why? What? How?", *The Journal of Systems and Software, Vol. 46 (2/3)*, 1999. pp. 91-105.

Lee, S. and Peña-Mora, F., "Understanding and managing iterative error and change cycles in construction", *System Dynamics Review, Vol. 23*, 2007, pp. 35–60.

Luna-Reyes, L.F., Martinez-Moyano I.J., Pardo T.A., Cresswell A.M, Andersen, D.A., and Richardson G.P., "Anatomy of a group model-building intervention: building dynamic theory from case study research", *System Dynamics Review, Vol. 22*, pp. 291–320, (2006).

Lyneis J.M., "System dynamics for business strategy: a phased approach", *System Dynamics Review, Vol. 19*, pp.37-70, 1999.

Lyneis J.M., Cooper K.G. and Els S.A., "Strategic management of complex projects: a case study using system dynamics". *System Dynamics Review Vol. 17(3)*, 2001, pp. 237– 260.

Lyneis J.M. and Ford, D.N., "System dynamics applied to project management: a survey, assessment, and directions for future research". *System Dynamics Review Vol. 23(2)*, 2007, pp. 157– 189.

Madachy, R. J. , "Software Process Concurrence", *Conference Proceeding, The 20th International Conference of the System Dynamics Society, July 28- August 1, 2002 Palermo, Italy*.

Madachy, R., "Reusable Model Structure and Behaviors for Software Processes". *SPW/ProSim 2006 Workshop May 20, 2006*.

Madachy, R.J., Boehm B. and Lane J.A., "Assessing Hybrid Incremental Processes for SISOS Development, Software Process Improvement and Practice", 2007, Vol.12, pp. 461– 473.

Madachy, R.J., "Software Process Dynamics", Wiley-IEEE Press, 2008.

Mutafelija B., Stromberg H., "Systematic Process Improvement using ISO 9001:2000 and CMMI", Artech House Computing Library. 2003.

Park, M. and Peña-Mora, F., "Dynamic change management for construction: introducing the change cycle into model-based project management", *System Dynamics Review, Vol. 19*, 2003, pp. 213–242.

Reel, J.S., "Critical success factors in software projects", IEEE Software Vol.16(3) pp.: 18-23, 1999.

Reichelt, K., and Lyneis J.M., "The Dynamics of Project Performance: Benchmarking the Drivers of Cost and Schedule Overrun", European Management Journal, Vol. 17 (2), pp. 135-150, 1999.

Ruiz, M., Ramos, I. and Toro, M. "An Integrated Framework for Simulation-based Software Process Improvement", Software Process Improvement and Practice 2004; Vol. 9 pp.81–93.

Setamanit S., Wakeland W. and Raffo D., "Using Simulation to Evaluate Global Software Development Task Allocation Strategies", Software Process Improvement and Practice 2007, Vol. 12, pp. 491–503.

Sterman, J.D. (2000), "Business Dynamics, System Thinking and Modeling for a Complex World", Irwin McGraw-Hill, New York.

Sterman, J.D., Henderson, R., Beinbocker, E.D., Newman, L.I., "Getting Big Too Fast: Strategic Dynamics with Increasing Returns and Bounded Rationality", Management Science, Vol. 53 (4), pp. 683-696, 2007.

Swebok. Guide to the Software Engineering Body. IEEE, Versión 2004.

Taylor, T. and Ford, D.N., "Tipping point failure and robustness in single development projects", *System Dynamics Review*, Vol. 22(1), 2006, pp. 51-71.

Yaghootkar K., Roos, L.U., "The Role of Systemic Effects on Project Performance in Multiproject Environments", *The 2007 International Conference of the System Dynamics Society and 50th Anniversary Celebration July 29 – August 2, 2007*, Boston, Massachusetts, USA, 2007.

Winz I., and Brierley, G., "Participatory Methods in Environmental System Dynamics Projects", *The 2007 International Conference of the System Dynamics Society and 50th Anniversary Celebration July 29 – August 2, 2007*, Boston, Massachusetts, USA, 2007.

Zhang, H., Kitchenham, B. and Pfahl, D., "Reflections on 10 Years of Software Process Simulation Modeling: A Systematic Review", ICSP 2008, LNCS 5007, pp. 345 – 356, 2008.

**Correspondencia** (Para más información contacte con):

Luis Moreno Iglesias.  
Universidad de Sevilla.  
Avenida de Reina Mercedes s/n, 41012 Sevilla, Spain.  
Phone: + 34 95 455 99 46  
Fax: + 34 95 455 65 45  
E-mail : [luism@us.es](mailto:luism@us.es)