DECSAI
Departamento de Ciencias de la Computación e I.A.
Universidad de Granada

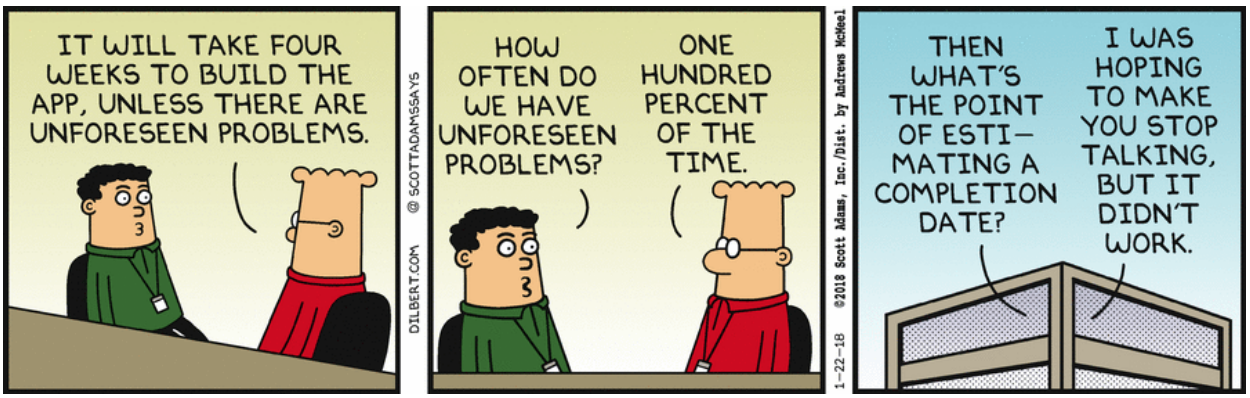
Gestión de riesgos

Fernando Berzal, berzal@acm.org

Gestión de riesgos

- Gestión de riesgos
- Riesgos habituales
- Estudios de viabilidad
- Identificación de riesgos
- Evaluación de riesgos
- Priorización de riesgos
- Plan de gestión de riesgos

Gestión de riesgos



Gestión de riesgos

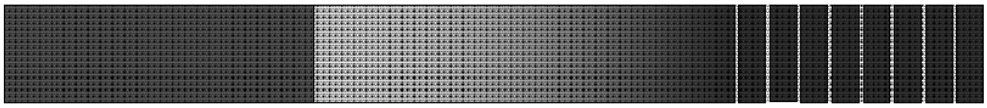
"First, risk concerns **future** happenings. Today and yesterday are beyond active concern, as we are already reaping what was previously sowed by our past actions. The question is, can we, therefore, by changing our actions today, create an opportunity for a different and hopefully better situation for ourselves tomorrow. This means, second, that risk involves **change**, such as in changes of mind, opinion, actions, or places. . . . [Third,] risk involves **choice**, and the uncertainty that choice itself entails. Thus paradoxically, **risk, like death and taxes, is one of the few certainties of life**."

-- Robert N. Charette:
Software Engineering Risk Analysis and Management, 1989.

Gestión de riesgos

Reducción de riesgos

Typical Relationship between Planned Work and Variable Work:



Planned Work

Unplanned, Variable Work
(typically >50% of total)

Better Relationship:

Planned Work

Planned
"Overhead"

Unplanned,
Variable Work

Gestión de riesgos

- El análisis de riesgos
  ayuda a gestionar la incertidumbre en un proyecto.

- **Un riesgo es un problema potencial**,
  que puede ocurrir o no.

  Independientemente de lo que ocurra,
  es aconsejable identificar riesgos,
  estimar su probabilidad de ocurrencia,
  evaluar su posible impacto y
  establecer un plan de contingencia
  por si los problemas realmente se presentan.

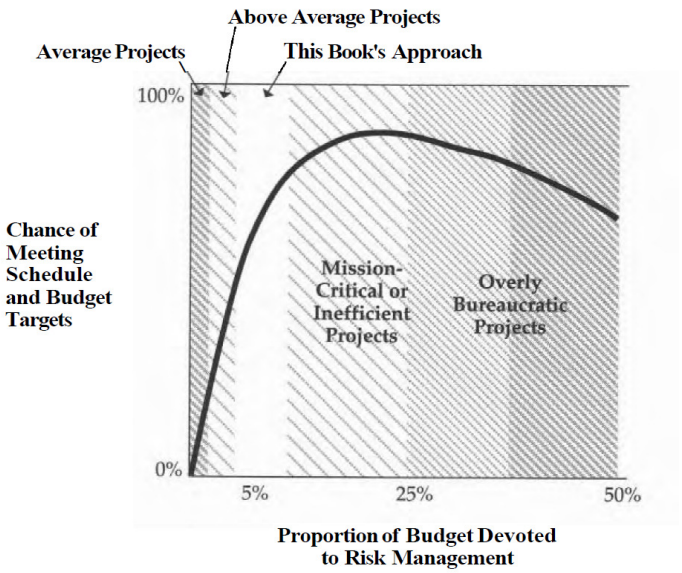
Gestión de riesgos

La gestión de riesgos
pretende responder las siguientes preguntas:

- ¿Qué puede salir mal?
- ¿Cuál es la probabilidad de que salga mal?
- ¿Cuál es el daño que puede causar?
- ¿Qué podemos hacer al respecto?

Gestión de riesgos

"You can't always predict, but you can always prepare."



Steve McConnell: "Software Project Survival Guide", 1997.

Gestión de riesgos

En una actividad compleja, como es un proyecto de desarrollo de software, muchas cosas pueden ir mal (de hecho, a menudo lo hacen), por lo que estar preparado es un elemento clave en la gestión de proyectos:

1. Comprender los riesgos de un proyecto.
2. Tomar medidas proactivas para evitar los riesgos (o gestionarlos/mitigarlos).

Gestión de riesgos

Estrategias para la gestión de riesgos I: Gestión reactiva

Se reacciona ante los riesgos cuando se materializan…

Rob Thomsett:
"The Indiana Jones School of Risk Management," *American Programmer*, vol. 5, no. 7, pp. 10-18, September 1992.

Gestión de riesgos

Estrategias para la gestión de riesgos II: Gestión proactiva

> "If you don't actively attack the risks, they will actively attack you."
> **-- Tom Gilb**

Se realiza un análisis formal de riesgos:

- Mitigación: Plan en anticipación del riesgo.
- Contingencia: Uso de los recursos planificados para contener el riesgo cuando éste se materializa.

Gestión de riesgos

Estrategias para la gestión de riesgos II: Gestión proactiva

- Corrección de raíz de las posibles causas del riesgo.

  p.ej. TQM [Total Quality Management]
  Statistical SQA [Software Quality Assurance]

NOTA: Las causas de los riesgos pueden ir más allá de los límites del proyecto actual y requerir cambios en la organización (lo que no siempre es posible).

Gestión de riesgos

Principios de gestión de riesgos [SEI]

- Perspectiva **global** (riesgos en el contexto del sistema del que el software es sólo un componente y del problema que se pretende resolver).
- Con vistas al **futuro** (riesgos que pueden aparecer en el futuro y planes de contingencia para gestionarlos).
- **Comunicación** abierta (fomentar que todos los "stakeholders" del proyecto puedan sugerir riesgos potenciales)

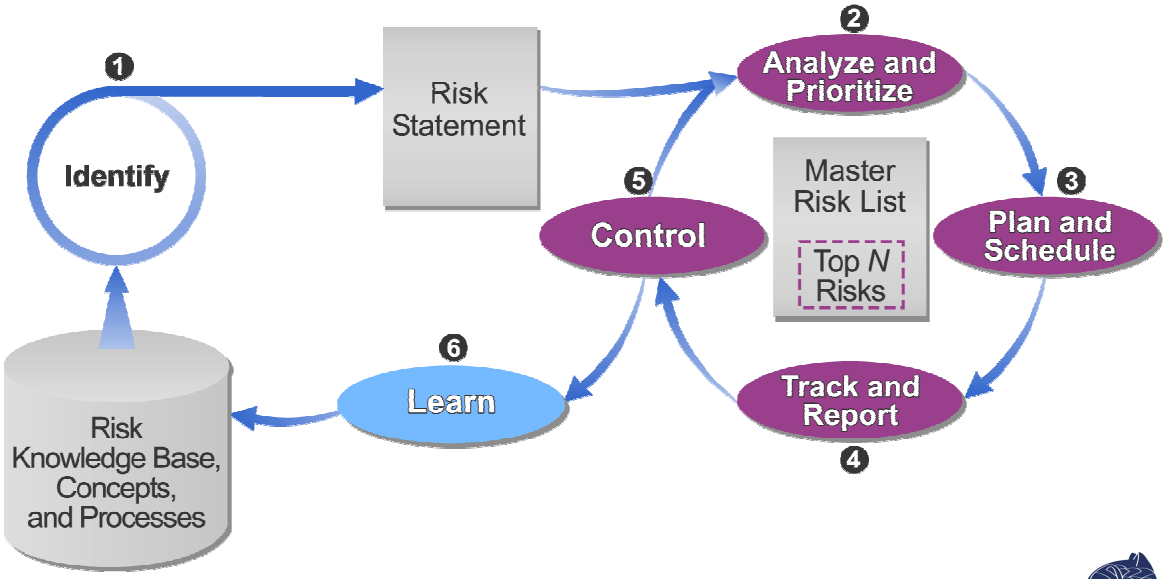
Gestión de riesgos

Principios de gestión de riesgos [SEI]

- **Integración** (de la gestión de riesgos en el proceso de desarrollo de software).
- Proceso **continuo** (actualización de los riesgos identificados conforme se recaba información y avanza el proyecto).
- Visión compartida y trabajo en **equipo** para mejorar la identificación y evaluación de riesgos con la ayuda de todos los "stakeholders" del proyecto.

Gestión de riesgos

Microsoft Solutions Framework [MSF]



MSF Risk Management Discipline, 2002
https://www.microsoft.com/en-us/download/details.aspx?id=721

Gestión de riesgos

- **Identificación de riesgos**
  (reconocer qué puede ir mal)
- **Evaluación/análisis de riesgos**
  (probabilidad de ocurrencia & impacto de cada riesgo)
- **Priorización de riesgos**
  (ranking de riesgos por probabilidad e impacto)
- **Plan de gestión de riesgos**

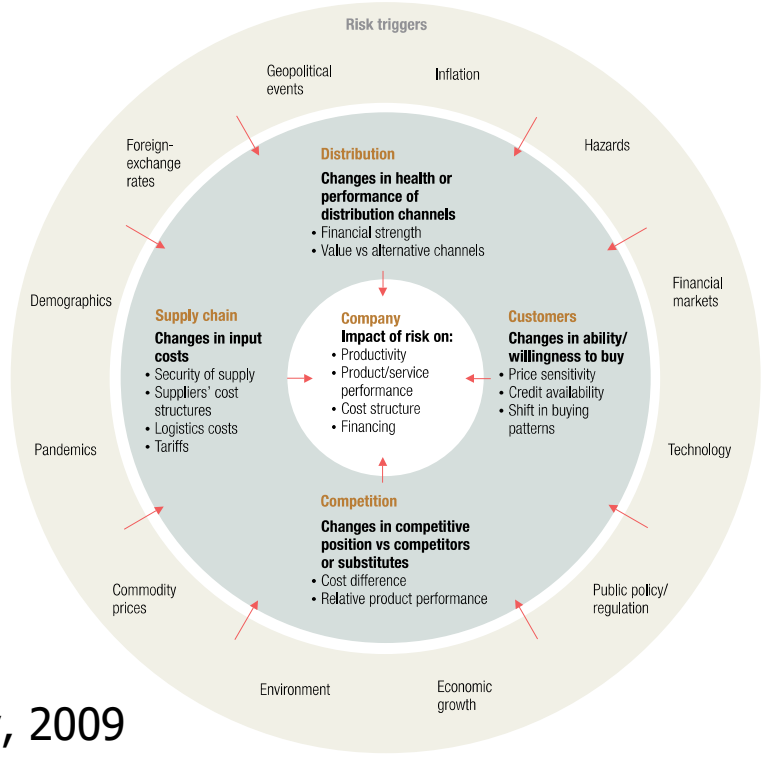
Riesgos habituales

Un proyecto está en riesgo si…

- No se comprenden bien las necesidades del cliente.
- El ámbito del proyecto no se delimita correctamente.
- No se gestionan correctamente los cambios.
- La tecnología cambia.
- Los plazos son poco realistas.
- Los usuarios se resisten al cambio.
- El proyecto carece de "patrocinador".
- El equipo carece de las habilidades necesarias.
- Los gestores olvidan el uso de "buenas prácticas"

Riesgos habituales

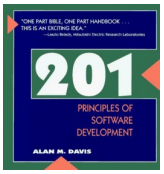
Riesgos habituales



McKinsey, 2009

Riesgos habituales

Top 10



- Falta de personal cualificado.
- Planes poco realistas.
- Comprensión inadecuada de los requisitos.
- Interfaz de usuario poco adecuada.
- Añadir características no necesarias ["gold-plating"].
- Falta de control sobre los cambios en los requisitos.
- Problemas con los componentes reutilizables y API's.
- Problemas en las tareas realizadas externamente.
- Tiempo de respuesta pobre.
- Ir más allá de lo que permite la tecnología.

Barry Boehm: "Software Risk Management: Principles and Practices", IEEE Software 9(1):32-39, January 1991

Riesgos habituales

Ámbito del proyecto

Si no se puede delimitar con precisión el ámbito del proyecto (establecer los límites de alguna de sus características), la incorporación de esa característica al proyecto se convierte en un riesgo.

Riesgos habituales

Tareas ¿completadas?

Regla 90-90

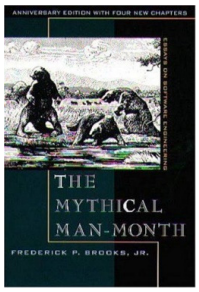
The first 90 percent of the code accounts for the first 90 percent of the development time. The remaining 10 percent of the code accounts for the other 90 percent of the development time.

-- Tom Cargill, Bell Labs

Jon Bentley: "Programming pearls: Bumper-Sticker Computer Science". Communications of the ACM 28 (9):896–901, September 1985
DOI 10.1145/4284.315122

Riesgos habituales

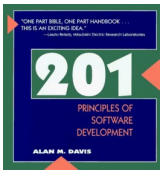
Ley de Brooks: Añadir mano de obra a un proyecto que va con retraso sólo lo retrasa aún más.



Frederick P. Brooks, Jr.:
The Mythical Man-Month:
Essays on Software Engineering
1975

Riesgos habituales

Estimaciones



Las cinco mayores causas de malas estimaciones de costes están todas relacionadas con el análisis de requisitos:

- Cambios frecuentes en los requisitos.
- Requisitos no especificados.
- Comunicación insuficiente con el usuario/cliente.
- Pobre especificación de requisitos.
- Análisis de requisitos insuficiente.

A. Lederer & J. Prasad: "Nine Management Guidelines for Better Cost Estimating", Communications of the ACM 35(2):51-59, February 1992.

Riesgos habituales

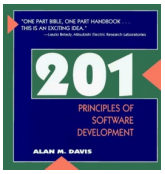
Estimaciones

Reconozca las diferencias existentes entre los objetivos de una empresa y las estimaciones de un proyecto: información valiosa de un riesgo que podría hacer descarrilar el proyecto.

Steve McConnell:
Software Estimation: Demystifying the Black Art
Microsoft Press, 2006. ISBN 0735605351

Riesgos habituales

Plazos poco realistas



Establecer un plazo de entrega poco realista sólo garantiza que el plazo no se cumplirá.

Además:

- Erosiona la moral del equipo y su motivación.
- Crea desconfianza en el gestor del proyecto.
- Anima a abandonar el equipo ["employee turnover"].
- Reduce la calidad (al recortar en QA bajo presión).

Tom DeMarco: "Why does software cost so much?"
IEEE Software 10(2):89-90, March 1993

Riesgos habituales

Incompatibilidades

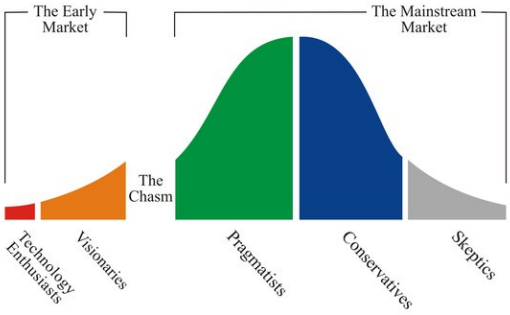
Nunca cambie de compilador o de sistema operativo en las últimas etapas de un proyecto antes de su entrega, ni siquiera de versión.

Riesgos habituales

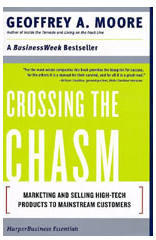
Nuevas tecnologías

Geoffrey A. Moore: "Crossing the Chasm: Marketing and Selling High-Tech Products to Mainstream Customers", 1991

P. N. Golder & G. J. Tellis: "Pioneer Advantage: Marketing Logic or Marketing Legend?" Journal of Marketing Research, 30(2):158–170, 1993.

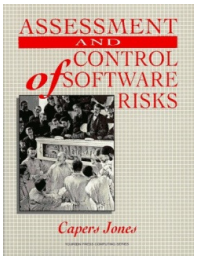


Grupo	Descripción	Tasa de fracaso
Innovadores	Primeros en desarrollar o patentar una idea	
Pioneros	Primeros en tener un prototipo que funcione	
"First movers"	Primeros en vender un producto en el mercado	**47%**
"Fast followers"	De los que entran antes en el mercado, sin ser los primeros	**8%**



Riesgos habituales

Factores de riesgo más graves



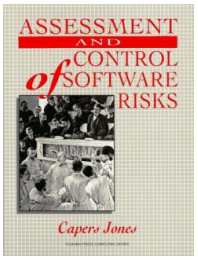
#	Factor de riesgo
1	Métricas inadecuadas (p.ej. LOC)
2	Toma de medidas inadecuada
3	Presión excesiva sobre el calendario del proyecto
4	Malas prácticas de gestión
5	Errores en la estimación de costes del proyecto
6	El síndrome de la "bala de plata" (CASE, OO…)
7	Aparición de nuevos requisitos (1%/mes)
8	Baja calidad
9	Baja productividad
10	Cancelación del proyecto

Capers Jones: "Assessment and Control of Software Risks," 1994

Riesgos habituales

Riesgos más graves por tipo de proyecto

Software de gestión (MIS)

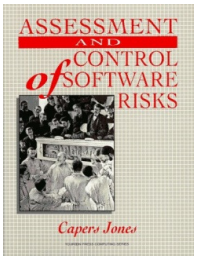


#	Factor de riesgo	%proyectos
1	Aparición de nuevos requisitos	80%
2	Presión excesiva sobre el calendario	65%
3	Baja calidad	60%
4	Sobrecostes	55%
5	Control inadecuado de la configuración	50%

Capers Jones: "Assessment and Control of Software Risks," 1994

Riesgos habituales

Riesgos más graves por tipo de proyecto
Software de sistemas

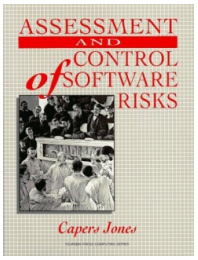


#	Factor de riesgo	%proyectos
1	Calendario demasiado amplio	70%
2	Estimación de costes inadecuada	65%
3	Excesiva burocracia [paperwork]	60%
4	Módulos propensos a errrores	50%
5	Cancelación del proyecto	35%

Capers Jones: "Assessment and Control of Software Risks," 1994

Riesgos habituales

Riesgos más graves por tipo de proyecto
Software comercial



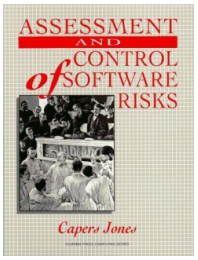
#	Factor de riesgo	%proyectos
1	Documentación de usuario inadecuada	70%
2	Baja satisfacción del usuario	55%
3	Comercialización demasiado tarde	50%
4	Medidas de la competencia	45%
5	Gastos legales	30%

Capers Jones: "Assessment and Control of Software Risks," 1994

Riesgos habituales

Riesgos más graves por tipo de proyecto

Software subcontratado [outsourcing]



#	Factor de riesgo	%proyectos
1	Costes de mantenimiento elevados	60%
2	Fricciones (personal propio-contratista)	55%
3	Aparición de nuevos requisitos	45%
4	Criterios de aceptación no anticipados	30%
5	Propiedad intelectual del software	20%

Capers Jones: "Assessment and Control of Software Risks," 1994

Estudios de viabilidad

Antes de comenzar la realización de un proyecto, para determinar la viabilidad del mismo:

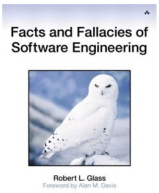
- Viabilidad técnica.
- Viabilidad económica.
- Viabilidad legal.

Estudios de viabilidad

The answer to a feasibility study
is almost always "yes."

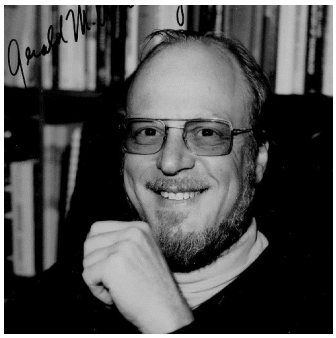
-- **Robert L. Glass**:

"Facts and Fallacies of Software Engineering", 2003



Optimismo incurable: como si ningún problema fuese demasiado difícil para resolverlo o cuando creemos que podemos tener terminado el proyecto "mañana" o, como mucho, "pasado mañana" (hasta que nos damos cuenta de que la fase de corrección de errores se lleva más tiempo que el análisis, el diseño y la implementación combinados).

Estudios de viabilidad



Association for Computing Machinery

SIGSOFT

IEEE computer society

tcse Technical Council on Software Engineering

Gerald M. Weinberg (keynote speaker):
"Overstructured Management of Software Engineering"
ICSE'1982, 6th International Conference on Software Engineering, Tokyo, Japan, September 13-16, 1982

Identificación de riesgos

"While it is futile to try to eliminate risk, and questionable to try to minimize it, it is essential that the risks taken be the right risks."

-- **Peter Drucker**: *Management*, 1975.

Identificación de riesgos

- Intento sistemático de especificar las amenazas que ponen en riesgo el proyecto.
- Hay que considerar tanto riesgos genéricos (amenazas potenciales a cualquier proyecto) como riesgos específicos (dependientes del contexto del proyecto particular).
- El uso de una lista de comprobación en la que se categoricen riesgos habituales puede resultar útil…

Identificación de riesgos

Tipos de riesgos

- Riesgos del proyecto (si se materializan, la terminación del proyecto se puede retrasar y sus costes aumentan)
- Riesgos técnicos (amenazan la calidad del software construido, dificultan su implementación o incluso pueden hacer ésta imposible), cuando algo es más difícil de hacer que lo que se esperaba.
- Riesgos económicos (amenazan la viabilidad del proyecto, poniendo en jaque el producto o servicio).

Identificación de riesgos

Factores de riesgo

Riesgos económicos

- Riesgo de mercado (construir un producto excelente que nadie quiere realmente).
- Riesgo estratégico (construir un producto que no encaja con las necesidades del cliente).
- Riesgo de ventas (construir un producto que el equipo de ventas no sabe cómo vender).
- Riesgo de gestión (perder el apoyo ejecutivo, por cambios de orientación o de responsables).
- Riesgo de presupuesto (perder compromisos presupuestarios o de personal).

Identificación de riesgos

Factores de riesgo

Riesgos técnicos

- Ambigüedad en las especificaciones.
- Herramientas de desarrollo.
- Proceso de desarrollo.
- Obsolescencia técnica.
- Tecnología puntera.

Riesgos del proyecto

- Complejidad del proyecto.
- Tamaño del proyecto.
- Comunicación con el cliente y otros "stakeholders".
- Personal del proyecto.

Identificación de riesgos

Factores de riesgo

No siempre se pueden categorizar fácilmente los riesgos, ya que hay:

- **Riesgos conocidos** (identificables al analizar las fuentes de información disponibles).
- **Riesgos predecibles** (extrapolables a partir de proyectos anteriores).
- **Riesgos impredecibles** (extremadamente difíciles de identificar de antemano).

Identificación de riesgos

Preguntas que hacerse

M. Keil et al.: "A Framework for Identifying Software Project Risks," Communications of the ACM, vol. 41(11):76-83, November 1998

- Have top software and customer managers formally committed to support the project?
- Are end-users enthusiastically committed to the project and the system/product to be built?
- Are requirements fully understood by the software engineering team and their customers?
- Have customers been involved fully in the definition of requirements?
- Do end-users have realistic expectations?

Identificación de riesgos

Preguntas que hacerse

M. Keil et al.: "A Framework for Identifying Software Project Risks," Communications of the ACM, vol. 41(11):76-83, November 1998

- Is project scope stable?
- Does the software engineering team have the right mix of skills?
- Are project requirements stable?
- Does the project team have experience with the technology to be implemented?
- Is the number of people on the project team adequate to do the job?
- Do all customer/user constituencies agree on the importance of the project and on the requirements for the system/product to be built?

Evaluación de riesgos

Factores que determinan las consecuencias de un riesgo

- Naturaleza (posibles problemas que puede causar).
- Ámbito (severidad del riesgo: ¿cómo de grave es? ¿a qué parte(s) del proyecto afecta?)
- Temporización (¿cuándo se puede producir?)

Evaluación de riesgos

Componentes de un riesgo

U.S. Air Force: *Software Risk Abatement*, AFCS/AFLC 800-45, 1988

- Rendimiento (¿el producto satisfará sus requisitos y resultará adecuado para su uso previsto?).
- Soporte (¿será fácil de corregir, adaptar y mejorar?)
- Coste (¿se mantendrá el presupuesto previsto?).
- Plan temporal (¿se entregará a tiempo?)

Evaluación de riesgos

Evaluación del impacto de un riesgo

Categorías generales

- Despreciable [negligible]
- Marginal
- Crítico
- Catastrófico

NOTA: Se evalúa el impacto del riesgo para cada uno de sus componentes (rendimiento, soporte, coste y plan temporal) para determinar su impacto global.

Evaluación de riesgos

Category \ Components		Performance	Support	Cost	Schedule
Catastrophic	1	Failure to meet the requirement would result in mission failure		Failure results in increased costs and schedule delays with expected values in excess of $500K	
	2	Significant degradation to nonachievement of technical performance	Nonresponsive or unsupportable software	Significant financial shortages, budget overrun likely	Unachievable IOC
Critical	1	Failure to meet the requirement would degrade system performance to a point where mission success is questionable		Failure results in operational delays and/or increased costs with expected value of $100K to $500K	
	2	Some reduction in technical performance	Minor delays in software modifications	Some shortage of financial resources, possible overruns	Possible slippage in IOC
Marginal	1	Failure to meet the requirement would result in degradation of secondary mission		Costs, impacts, and/or recoverable schedule slips with expected value of $1K to $100K	
	2	Minimal to small reduction in technical performance	Responsive software support	Sufficient financial resources	Realistic, achievable schedule
Negligible	1	Failure to meet the requirement would create inconvenience or nonoperational impact		Error results in minor cost and/or schedule impact with expected value of less than $1K	
	2	No reduction in technical performance	Easily supportable software	Possible budget underrun	Early achievable IOC

Note: (1) The potential consequence of undetected software errors or faults.
(2) The potential consequence if the desired outcome is not achieved.

Barry W. Boehm: **Software Risk Management**, 1989.

Evaluación de riesgos

Proyección de riesgos (a.k.a. estimación de riesgos)

Evaluación de riesgos atendiendo a dos factores:

- Probabilidad de que el riesgo se materialice.
- Consecuencias de los problemas asociados al riesgo (si éste se materializa).

Evaluación de riesgos

Proyección de riesgos (a.k.a. estimación de riesgos)

Actividades

1. Establecer una escala que refleje la probabilidad de que un riesgo se materialice.
2. Delinear las consecuencias del riesgo.
3. Estimar el impacto del riesgo en el proyecto.
4. Evaluar la precisión de la proyección de riesgos (para evitar malentendidos).

Evaluación de riesgos

Exposición al riesgo
[risk exposure]

RE = Probabilidad x Coste

E.M. Hall:
Managing Risk: Methods for Software Systems Development, 1998.

Evaluación de riesgos

Exposición al riesgo

Identificación	Sólo el 70% de los componentes reutilizables previstos podrán integrarse, por lo que el 30% restante tendrá que hacerse a medida
Probabilidad	**75%** (probable)
Impacto	30 componentes reutilizables previstos x 30% componentes a medida x 100 LOC/componente x 10 €/LOC = **9000 €**
Exposición	RE = 0.75 x 9000 € = **6750 €**

Evaluación de riesgos

- La exposición al riesgo nos sirve de estimación del coste esperado asociado a cada riesgo identificado.
- Podemos utilizar la exposición al riesgo para ajustar la estimación del coste del proyecto.
- Si la suma de las exposiciones al riesgo de los distintos riesgos identificados es significativa en comparación con el coste del proyecto (p.ej. >50%), deberíamos evaluar la viabilidad del proyecto.

Priorización de riesgos

Priorización de riesgos

El objetivo de la evaluación de riesgos es establecer criterios que permitan establecer prioridades.

- Ningún proyecto tiene recursos suficientes para atender todos los riesgos posibles con el mismo grado de rigor.
- Al establecer prioridades, se destinan los recursos disponibles donde tengan un mayor impacto.

Priorización de riesgos

Lista de riesgos

Riesgo	Categoría	Probabilidad	Impacto	Plan RMMM

Probabilidad de ocurrencia (%)

Impacto: marginal, crítico, catastrófico.

RMMM: Risk Mitigation, Monitoring & Management

Priorización de riesgos

Lista de riesgos

- Las técnicas de análisis de riesgos deben aplicarse iterativamente a lo largo del proyecto.
- Las probabilidad e impacto de cada riesgo puede variar conforme avanza el proyecto, por lo que cambia su posición relativa con respecto a otros riesgos.
- Pueden aparecer nuevos riesgos (y desaparecer otros, que ya no sean relevantes).
- La lista de riesgos debe revisarse periódicamente y mantenerse actualizada (p.ej. en un lugar visible de la intranet del proyecto).

Plan de gestión de riesgos

RMMM [Risk Mitigation, Monitoring & Management]

- Una vez identificado un riesgo, evitar que pueda ocurrir es la mejor estrategia [risk avoidance].
- Si no se puede eliminar el riesgo por completo, hay que diseñar un plan de mitigación [risk mitigation], que incluya medidas que disminuyan su probabilidad y/o reduzcan su impacto, antes y durante la ejecución del proyecto.

NOTA: Si la exposición a un riesgo específico es menor que el coste de mitigación de ese riesgo, no tiene sentido tomar medidas de mitigación del riesgo: sólo se monitoriza.

Plan de gestión de riesgos

RMMM [Risk Mitigation, Monitoring & Management]

- Durante el proyecto, se monitorizan los riesgos utilizando indicadores que nos ayuden a determinar si el riesgo es más o menos probable.
- También se monitoriza la efectividad de las medidas tomadas para mitigar riesgos.
- En caso de que el riesgo se materialice, se recurre a las medidas previstas en planes de contingencia.

Plan de gestión de riesgos

RIS [Risk Information Sheet]

Ficha individual para cada riesgo:

- ID
- Probabilidad
- Impacto
- Descripción
- Mitigación
- Monitorización
- Plan de contingencia
- Recursos estimados

Plan de gestión de riesgos

RIS [Risk Information Sheet]

Project: Embedded software for XYZ system
Risk type: schedule risk
Priority (1 low ... 5 critical): 4
Risk factor: Project completion will depend on tests which require hardware component under development. Hardware component delivery may be delayed
Probability: 60 %
Impact: Project completion will be delayed for each day that hardware is unavailable for use in software testing
Monitoring approach:
Scheduled milestone reviews with hardware group
Contingency plan:
Modification of testing strategy to accommodate delay using software simulation
Estimated resources: 6 additional person months beginning in July

Roger S. Pressman:
"Software Engineering: A Practitioner's Approach", 7th edition, 2009

Plan de gestión de riesgos

RIS [Risk Information Sheet]

Risk information sheet			
Risk ID: P02-4-32	Date: 5/9/09	Prob: 80%	Impact: high
Description: Only 70 percent of the software components scheduled for reuse will, in fact, be integrated into the application. The remaining functionality will have to be custom developed.			
Refinement/context: Subcondition 1: Certain reusable components were developed by a third party with no knowledge of internal design standards. Subcondition 2: The design standard for component interfaces has not been solidified and may not conform to certain existing reusable components. Subcondition 3: Certain reusable components have been implemented in a language that is not supported on the target environment.			
Mitigation/monitoring: 1. Contact third party to determine conformance with design standards. 2. Press for interface standards completion; consider component structure when deciding on interface protocol. 3. Check to determine number of components in subcondition 3 category; check to determine if language support can be acquired.			
Management/contingency plan/trigger: *RE* computed to be $20,200. Allocate this amount within project contingency cost. Develop revised schedule assuming that 18 additional components will have to be custom built; allocate staff accordingly. Trigger: Mitigation steps unproductive as of 7/1/09.			
Current status: 5/12/09: Mitigation steps initiated.			
Originator: D. Gagne		Assigned: B. Laster	

Roger S. Pressman:
"Software Engineering: A Practitioner's Approach", 7th edition, 2009

Plan de gestión de riesgos

TABLE 7-3 RISK-MANAGEMENT PLAN

Why?
Why is a risk-management plan needed for this specific risk? Describe the risk's probability of occurrence, consequences, and severity.

How?
How will the risk be resolved in general? Describe the general approach to resolving the risk. List or describe the options that were considered.

What?
What specific steps will be taken to resolve the risk? List the specific steps and deliverables that will be generated in addressing the risk.

Include a description of the conditions under which the risk will be upgraded—for example, if the risk cannot be resolved by a specific date.

Who?
Who will be responsible for completing each step? List the specific person responsible for completing each step.

When?
When will each step be completed? List the completion date for each step.

How much?
How much budget is allocated to the resolution of the risk? List the cost of each step that will be taken to resolve the risk.

Comentarios finales

El efecto Titanic
Pensar que un desastre es imposible
a menudo acaba en desastre.
-- Gerald M. Weinberg: "Quality Software Management"

El gestor del proyecto debe:

- esperar que los riesgos se puedan materializar,
- identificar potenciales amenazas desde el comienzo,
- desarrollar planes de contingencia por anticipado, y
- reevaluar nuevos riesgos continuamente.

Comentarios finales

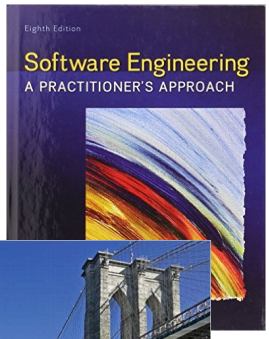
El impacto de los riesgos
debe incluirse en la planificación temporal del proyecto
(y en la estimación de costes)

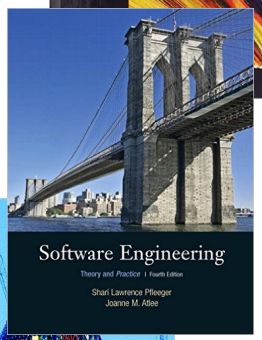
Risk	Probability	Impact	Exposure (RE)
New technology doesn't live up to expectations	25%	8 weeks	2.0 weeks
New technology requires staff training	50%	1 week	0.5 weeks
Demo version of software is required to support trade show	75%	2 weeks	1.5 weeks
Senior staff not available as planned	25%	10 weeks	2.5 weeks
Government regulations change before software ships	10%	2 weeks	0.2 weeks
Total	-	23 weeks	6.7 weeks

Steve McConnell: "10 Deadly Sins of Software Estimation", 2002

Bibliografía

Libros de texto

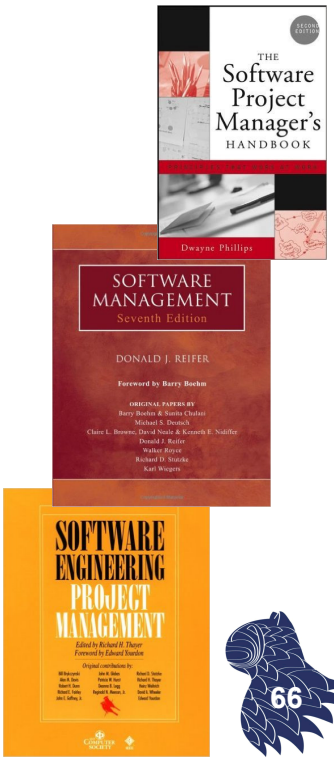




- Roger S. Pressman:
  Software Engineering: A Practitioner's Approach
  McGraw-Hill, 8th edition, 2014. ISBN 0078022126
- Shari Lawrence Pfleeger & Hoanne M. Atlee:
  Software Engineering: Theory and Practice
  Prentice Hall, 4th edition, 2009. ISBN 0136061699
- Ian Sommerville:
  Software Engineering
  Pearson, 10th edition, 2015. ISBN 0133943038

Bibliografía

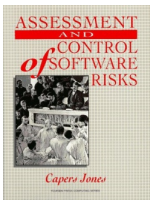
Lecturas recomendadas

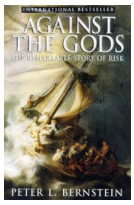


- Dwayne Phillips:
  The Software Project Manager's Handbook: Principles That Work at Work
  Wiley / IEEE Computer Society, 2nd edition, 2004
  ISBN 0471674206
- Donald J. Reifer (editor):
  Software Management
  Wiley / IEEE Computer Society, 7th edition, 2006
  ISBN 0471775622
- Richard H. Thayer (editor):
  Software Engineering Project Management
  Wiley / IEEE Computer Society, 2nd edition, 2000
  ISBN 0818680008

Bibliografía complementaria

Gestión de riesgos

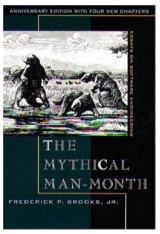


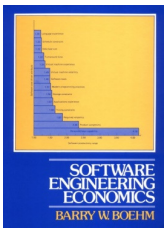


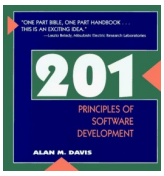
- Tom DeMarco & Tim Lister:
  Waltzing with bears: Managing risk on software projects
  Dorset House, 2003. ISBN 0932633609
- Capers Jones:
  Assessment and control of software risks
  Yourdon Press, 1994. ISBN 0137414064
- Peter L. Bernstein:
  Against the Gods: The Remarkable Story of Risk
  John Wiley & Sons, 1998. ISBN 0471295639

Bibliografía complementaria

Clásicos





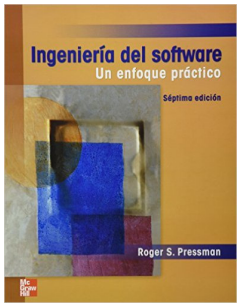


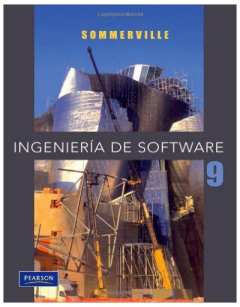
- Frederick P. Brooks, Jr.:
  The Mythical Man-Month:
  Essays on Software Engineering
  Addison-Wesley, 1995. ISBN 0201835959
- Alan M. Davis:
  201 Principles of Software Development
  McGraw-Hill, 1995. ISBN 0070158401
- Barry W. Boehm:
  Software Engineering Economics
  Prentice-Hall PTR, 1991. ISBN 0138221227
- **Manager's Handbook for Software Development**
  NASA Software Engineering Laboratory, SEL-84-101, rev.1, 1990.
- **Software Engineering Laboratory (SEL)**
  Relationships, Models, and Management Rules
  NASA Software Engineering Laboratory, SEL-91-001, 1991.

Bibliografía

Bibliografía en castellano

- Roger S. Pressman:
  Ingeniería de Software: Un enfoque práctico
  McGraw-Hill, 7ª edición, 2010. ISBN 6071503140
- Ian Sommerville:
  Ingeniería de Software
  Pearson, 9ª edición, 2012. ISBN 6073206038





Ejercicios

Busque en Internet una lista detallada de riesgos para proyectos de desarrollo de software (p.ej. SEI, NASA, Banco Mundial...) y describa brevemente la taxonomía utilizada para clasificarlos.