

# Mejoras en el Proceso de Testing

**Fernando Calles Gato**  
*Indra Sistemas*  
*fcalles@indra.es*



## Innovate2010

The Rational Software Conference

Let's **build** a smarter planet.

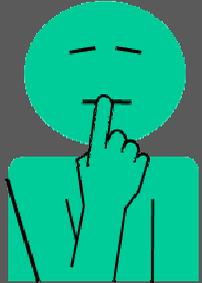
The premiere software and product delivery event.  
4 de Noviembre, Madrid



# MARCO CONCEPTUAL



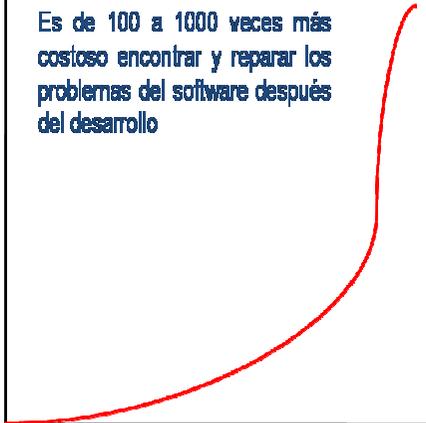
# ¿Por qué es necesario el Testing?



¿Coste?  
○  
¿Beneficio?

**Coste**

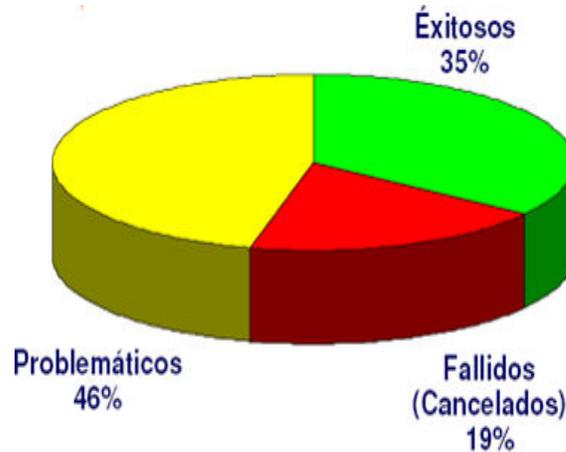
Es de 100 a 1000 veces más costoso encontrar y reparar los problemas del software después del desarrollo



**Tiempo**

✓ “Los bugs de software cuestan \$59,5 mil millones de dólares anuales a la economía de EE.UU.”  
(Fuente [www.calidaddelsoftware.com](http://www.calidaddelsoftware.com))

Un estudio de Standish Group indica que se producen graves consecuencias en la calidad del software por no implementar el proceso de pruebas en los proyectos.



- ⊕ De una inversión en proyectos de \$255 billones, se desperdician \$55 billones
- ⊕ Al liberar un producto, tan sólo están incluidas el 52% de las funciones y propiedades requeridas.
- ⊕ De media los costes de los proyectos suponen el 143% de lo estimado, y el 82% se pasa de plazos

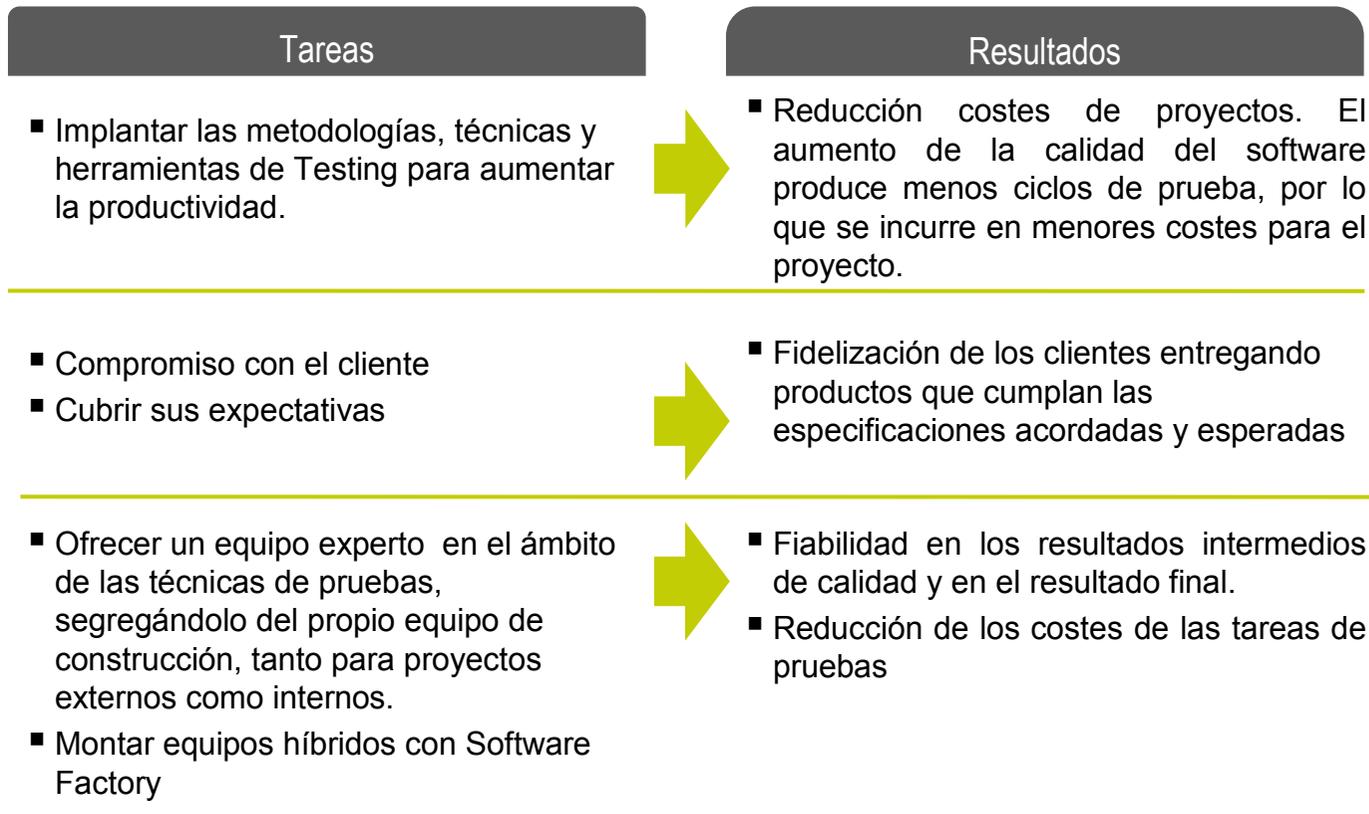
Definiciones	
<b>Éxitosos</b>	En tiempo, en presupuesto, en funcionalidad prometida
<b>Problemáticos</b>	Tarde, sobrepasado el presupuesto, falta funcionalidad
<b>Fallidos</b>	Proyectos cancelados

Source: Standish Group Chaos Report  
© ESI 2009

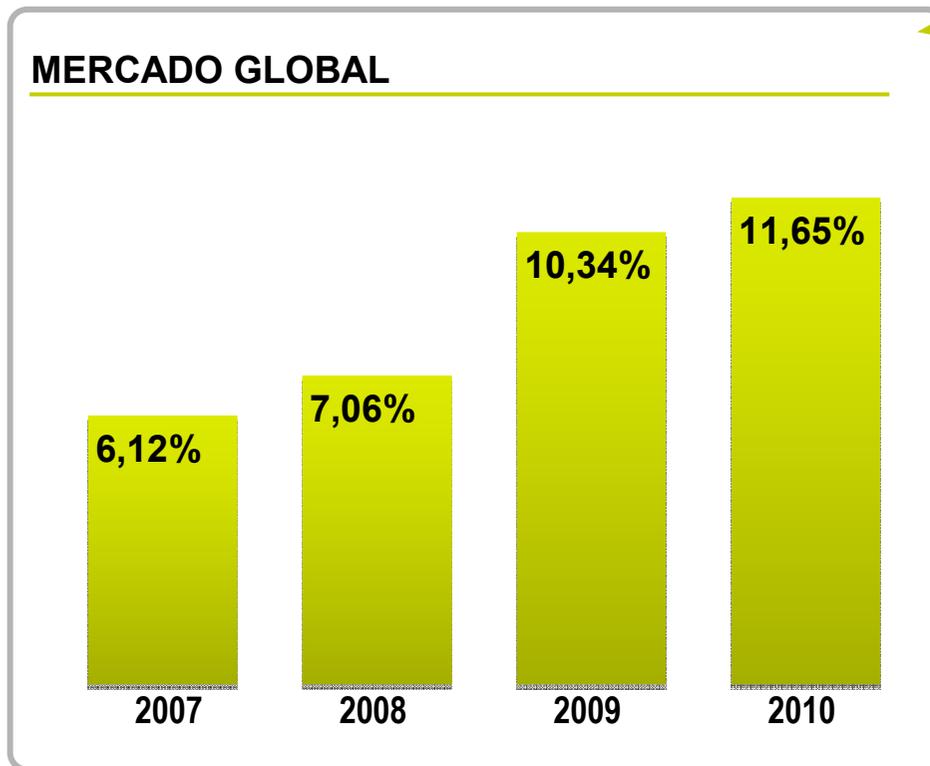


## Misión de Indra

Ofrecer a sus clientes Servicios de Testing y Certificación de Software orientados a garantizar que tanto el proceso de desarrollo como el producto entregable cumplen con los compromisos pactados y las expectativas del usuario final, respetando la planificación definida y el presupuesto asignado.



## Evolución del Mercado Mundial de Testing con respecto al mercado de TI



La evolución anual en el incremento de la inversión en tareas de Testing, refleja la importancia que los clientes dan a la etapa de aseguramiento de la calidad

- Objetivo Indra:**
1. Referente como certificador de aplicaciones.
  2. Reducir las incidencias en los Proyectos.
  3. Ofrecer a los mercados una relación coste/beneficio altamente satisfactoria.
  4. Ofrecer a los Mercados especialización e industrialización de la etapa de pruebas en proyecto.

Fuente: INFINITY RESEARCH

Porcentaje de Testing con respecto al global de TI



## Objetivos de Indra

¿Qué son las pruebas del software?

Técnicas utilizadas para certificar si un producto software puede ser puesto en funcionamiento a nivel productivo.

Un buen test de prueba es aquel que logra detectar errores con el objetivo de ser subsanados.

Para ser más efectivas, las pruebas deben ser conducidas por un equipo independiente

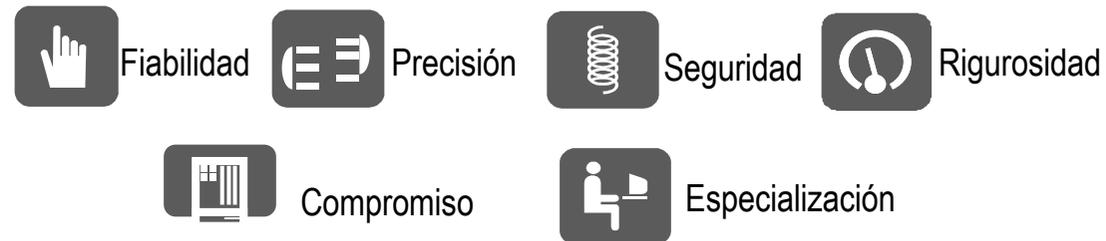
### Problemática

---

El esfuerzo de implementar Testing en los proyectos supone un 15% del esfuerzo total de un DaM.

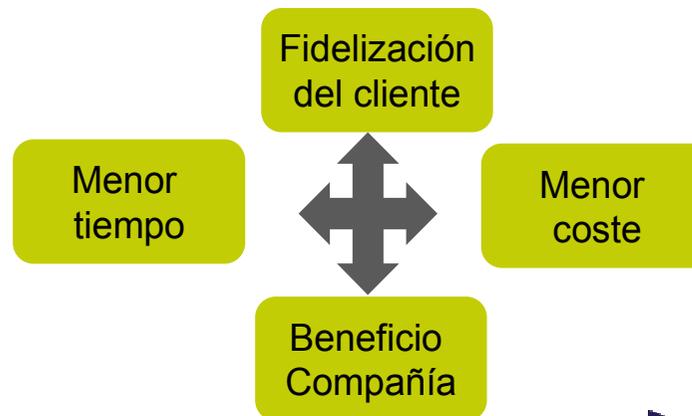
### Objetivos

---



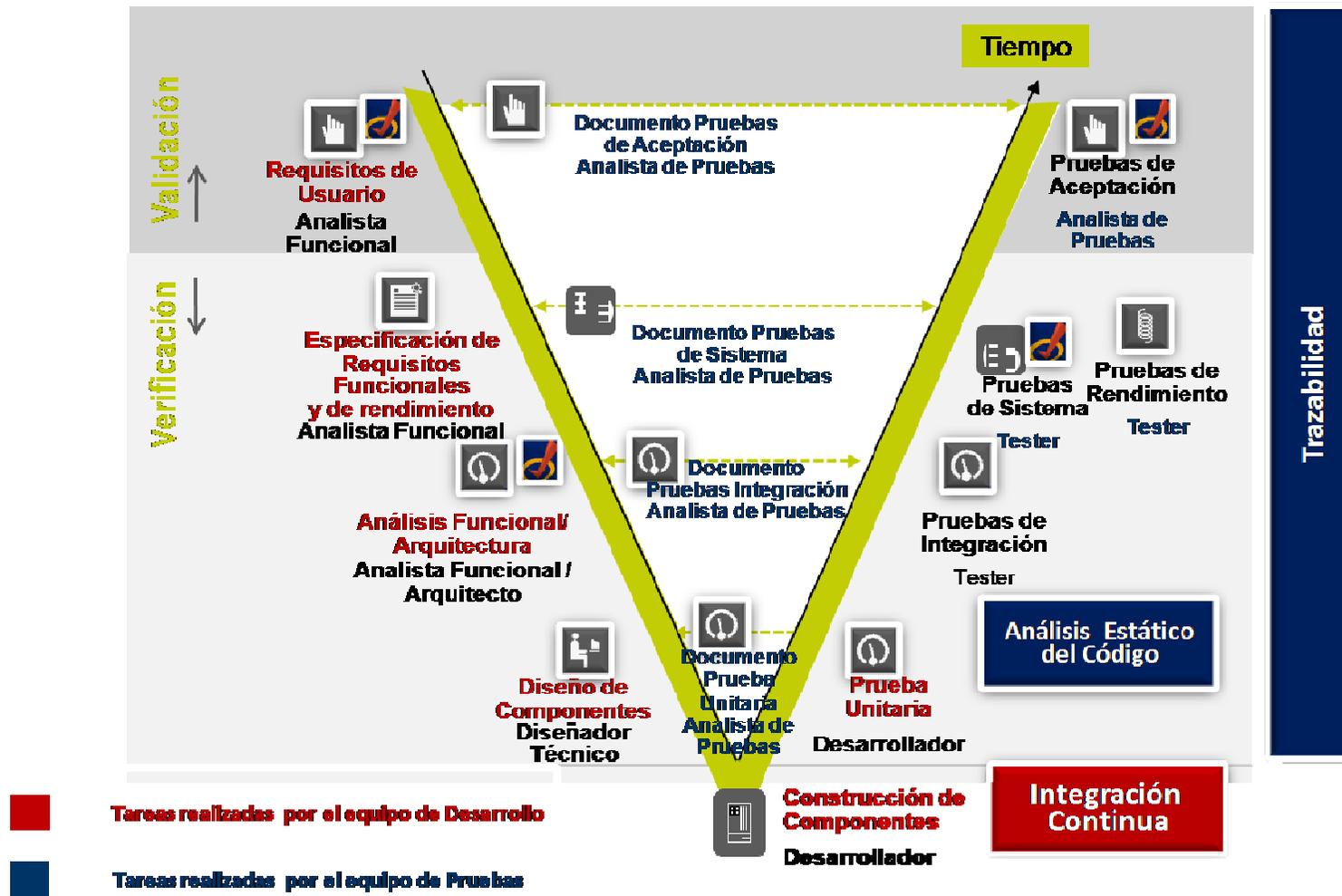
### Beneficio

---

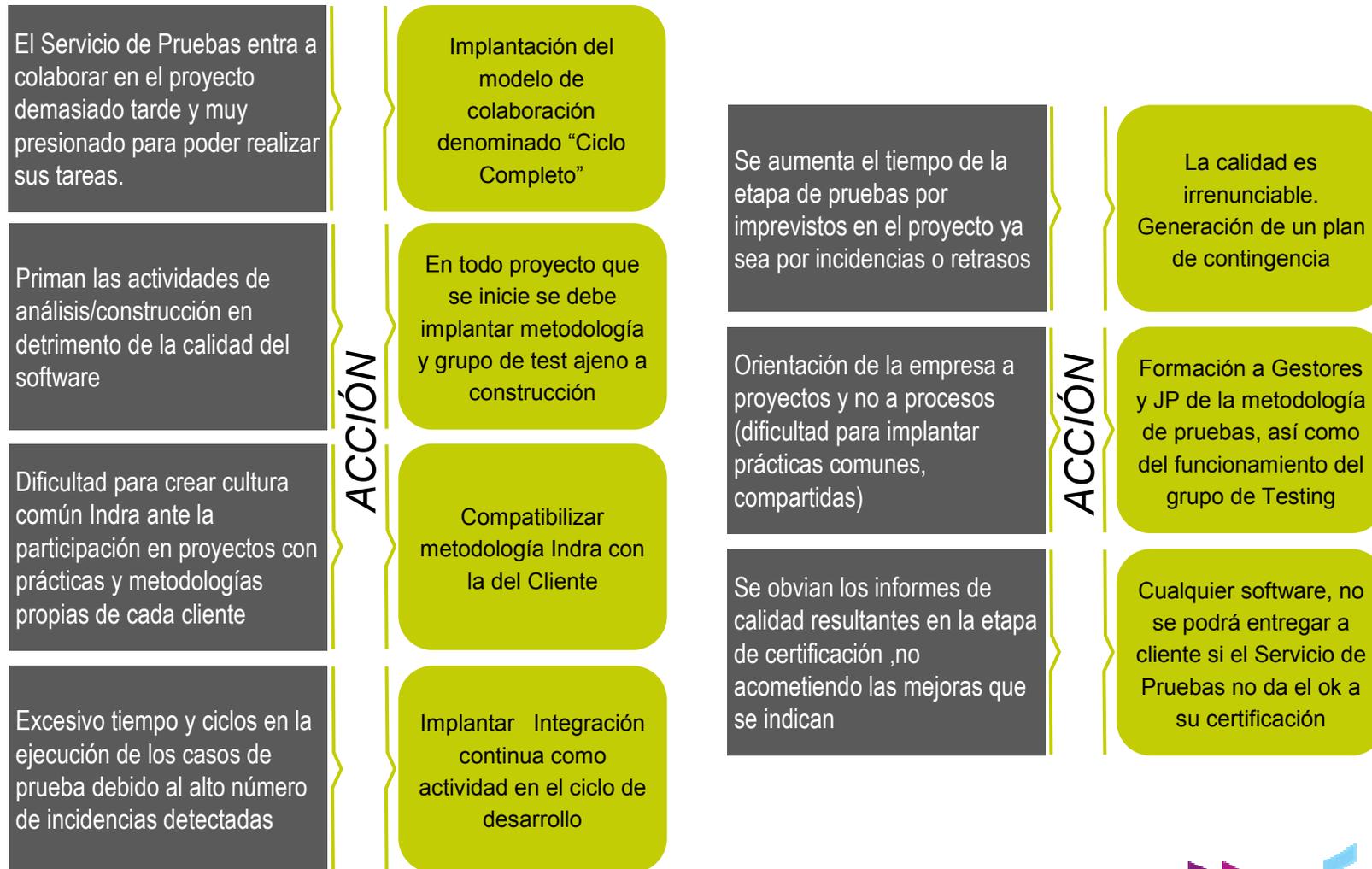


# Relación entre productos de desarrollo y niveles de prueba

Proporcionar al cliente una solución global en el ámbito de la calidad dentro del ciclo de vida del software



## Diagnóstico: Áreas de mejoras vs actuación



# Propuesta de Valor



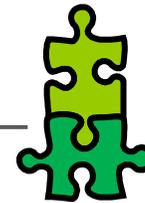
# Offering de Calidad del Software



## Interno

---

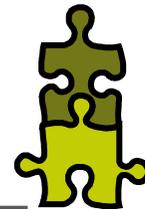
Mejorar la calidad de nuestros proyectos dentro de la compañía, asumiendo la etapa de certificación dentro del ciclo de vida del software aportando recursos y buenas prácticas: reducción de riesgos, cumplimiento de presupuestos y calendarios, identificación temprana de errores, etc.



## Externo

---

Ofrecer a nuestros clientes un servicio orientado a garantizar que tanto el proceso de desarrollo del software como el producto final (propio o de terceros) cumplen con las expectativas del usuario, respetando la planificación definida y asegurando la calidad esperada en función del coste presupuestado.



## Áreas de Actuación

	Objetivos	Beneficios
<b>Pruebas Funcionales</b>	<p>Comprobar que el sistema cumple con las especificaciones funcionales y de negocio acordadas.</p> <p>“ El sistema hace lo que tiene que hacer ”</p>	<ul style="list-style-type: none"> <li>• El coste de solucionar dichos errores es menor.</li> <li>• Aumenta el grado de disponibilidad de los sistemas y, por tanto, disminuyen las pérdidas de oportunidad de negocio</li> <li>• Se consolida y fideliza al cliente</li> </ul>
<b>Trazabilidad</b>	<p>Recopila y enlaza información entre requerimientos, casos de uso y casos de prueba.</p> <p>“Todo requerimiento ha de ser diseñado, codificado y probado”</p>	<ul style="list-style-type: none"> <li>• Todos los requerimientos están cubiertos por los casos de prueba y viceversa.</li> <li>• Obtención de todos los documentos de requerimientos, diseño, y guiones de prueba</li> <li>• Gestión del control de cambios</li> </ul>
<b>Análisis de Código</b>	<p>Verificar si el código fuente, se ajusta a métricas parametrizadas (desde el punto de vista de la sintaxis y numérico) y consensuadas con el cliente</p>	<ul style="list-style-type: none"> <li>• Proporcionará mejoras sustanciales en la calidad de la programación y mantenimiento</li> </ul>
<b>Pruebas No Funcionales</b>	<p>Pruebas de Seguridad Pruebas de Disponibilidad Pruebas Usabilidad, Accesibilidad Pruebas de Rendimiento</p>	<ul style="list-style-type: none"> <li>• Comprobar la vulnerabilidad del sistema</li> <li>• El sistema funciona de forma continua</li> <li>• Detección de cuellos de botella</li> <li>• Validación de los niveles de acceso</li> <li>• Detección de carga máxima del sistema</li> </ul>
<b>Integración Continua</b>	<p>Integraciones automáticas continuas basadas en compilaciones y ejecución de test de forma frecuente con el objetivo de detectar fallos lo antes posible</p>	<ul style="list-style-type: none"> <li>• Reducción del Time to Market</li> <li>• Reducción del tiempo de integración</li> <li>• Detección de errores lo más pronto posible</li> <li>• Pruebas inmediatas tras cambios en código</li> </ul>
<b>Selección de Herramientas</b>	<p>Asesorar al cliente en la selección de las herramientas de testing que mejor se adapten a sus necesidades</p>	<ul style="list-style-type: none"> <li>• Implantación</li> <li>• Formación</li> <li>• Mantenimiento de versiones</li> </ul>

## Seis objetivos

El objetivo principal de la Unidad de Calidad de Indra es ser el referente de todas las actividades tecnológicas y comerciales que se desarrollen dentro del contexto del aseguramiento de la calidad.

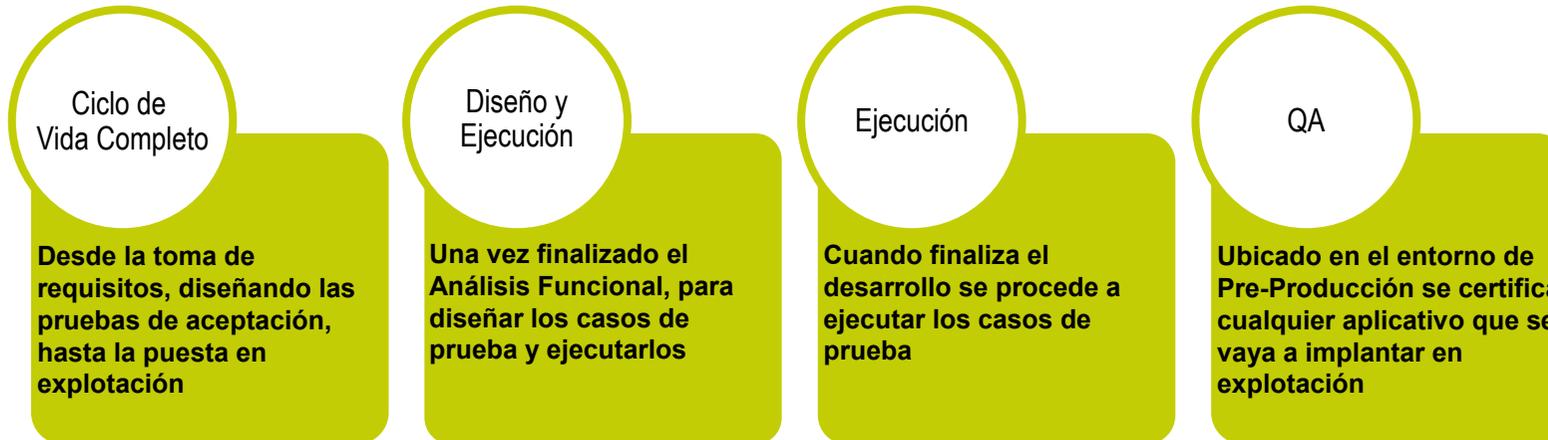
Con ese objetivo, dispone de los recursos técnicos y humanos necesarios para satisfacer las demandas de cada cliente.

Su objetivo, en todo momento, es consolidar, difundir y evolucionar una estrategia asegurando la calidad que espera su cliente, consiguiendo reducir costes.

Esta Unidad de Pruebas ofrece a sus clientes Servicios de Testing y Aseguramiento de la Calidad de Software orientados a garantizar que tanto el proceso de desarrollo como el producto final cumplen con las expectativas del usuario, respetando la planificación definida y el presupuesto asignado.



## Cuatro medios de colaboración



### “Una Necesidad por cubrir”

En España, las empresas que se dedican a abordar la calidad del software son escasas.

No hay grandes empresas que se estén centrando en ofrecer este tipo de servicios especializados al cliente español que lo demanda.

Es un negocio muy dependiente del conocimiento del dominio del producto con el que se trabaja.

Incrementar la calidad del software, mientras se reducen costes.

Más de 100 profesionales en Indra asegurando la calidad de los Sistemas de Información



## Seis modalidades de colaboración

### Testing funcional

Con el testing funcional se persigue la detección del mayor número de errores en un ambiente controlado, previo a Producción, así como la validación de la funcionalidad de acuerdo a los requisitos del usuario, asegurando la trazabilidad entre requisitos, casos de uso y casos de prueba

### Testing de Performance

Detectar cuellos de botella en el sistema y diagnosticar el origen de los mismos en base a los datos recolectados. Medir los tiempos de respuesta de una aplicación de extremo a extremo. Asegurar la capacidad cuando el sistema soporta la carga máxima de usuarios concurrentes

### Análisis Estático del Código

Establecer el conjunto de criterios, indicadores y umbrales, obtenidos del análisis estático del software que sirvan de base para establecer un sistema de evaluación del software con el objeto de minimizar: El número de errores en el código, el esfuerzo que requiere mantenerlo y el tiempo que requiere su desarrollo.

### Integración Continua

Realización de integraciones automáticas de un proyecto lo más frecuentemente posible, y así poder anticiparse a los posibles fallos que en éste se puedan dar. Aseguramiento de la ejecución de pruebas unitarias Última versión siempre disponible y ensamblada

### Trazabilidad

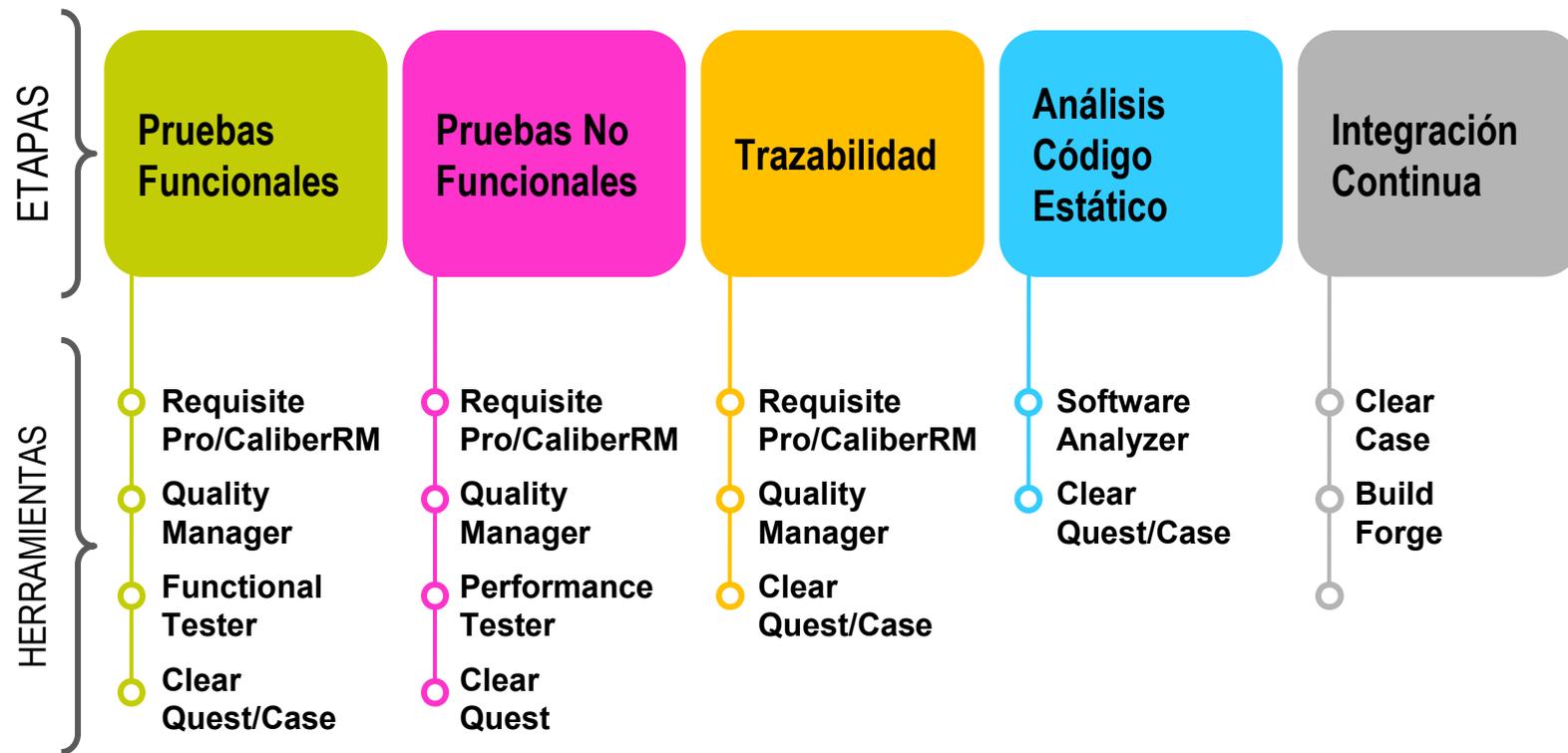
Permite fácilmente detectar si hay conflictos entre requerimientos, diseño, lógica de codificación y casos de prueba, consiguiendo que todos los requerimientos estén cubiertos por los casos de prueba y viceversa y detectando los cambios a realizar en el software cuando sean modificados.

### Selección de Herramientas

El conocimiento y la amplia experiencia de Indra con las diferentes herramientas de testing, permiten asesorar a cada cliente en la selección del producto que mejor se adapte a sus necesidades, además de implantarlas, impartir formación y aportar best practices sobre ellas.

La Unidad de Pruebas pone a disposición de los clientes de Indra el modo de conseguir que este cumpla con los objetivos clave del negocio, mediante un proceso de calidad del software efectivo y eficiente. Es efectivo porque intenta encontrar a tiempo, los defectos y tiene en cuenta los riesgos del producto, y es eficiente, porque es un método aplicable universalmente, que se centra especialmente en la reutilización.

## Herramientas del Servicio



# Experiencias



## Necesidad de Probar Software

- Surge la necesidad de probar el software antes de entregar al cliente:
  - ¿Funciona la aplicación correctamente?
  - ¿Se han probado todos los requisitos/CUS?
  - ¿Está la infraestructura dimensionada correctamente?
  - ¿Se puede entregar el software?



## Situación anterior

- **Testing Funcional:**

- En el apartado de pruebas funcionales se han estado utilizando hasta ahora, herramientas Opensource adaptadas a las necesidades de los proyectos con el objetivo de gestionar, casos de prueba e incidencias. Estas herramientas aunque cubrían el flujo del desarrollo están limitadas en cuanto a capacidades.

Mantis  
indra  
Consultoría Tecnológica

Conectado como: agalfonso (Alex - desarrollador) 2009-10-26 13:47

Principal | Mi Vista | Ver Incidencias | Reportar Incidencia | Log de cambios | Roadmap

@ Buscar:  Filtrar [ Filtros avanzados ] [ Crear Enlace ]

Mostrando Incidencias (1 - 7 / 7) [ Imprimir Informes ] [ Exportar a CSV ]

	P	ID	#	Categoría	Severidad	Estado	Actualizada	
<input type="checkbox"/>		0002909	2	Web	grave	Reabierta (jbermejov)	2009-10-06	Inicio de
<input type="checkbox"/>		0002958	1	Web	mejora	Rechazada (rdeandres)	2009-09-23	En la may. Entidad
<input type="checkbox"/>		0002983	1	Web	mejora	Rechazada (daportillo)	2009-09-23	Mensaje
<input type="checkbox"/>		0002957	2	Web	mejora	Corregida (rdeandres)	2009-09-21	Listas de
<input type="checkbox"/>		0003000		Web	bloqueante	Abierta (imdiaze)	2009-08-	Problema

Testlink TestLink 1.8.2 : Testlink Administrator [admin]

Inicio | Casos de Prueba | Ejecutar | Resultados | Usuarios | Eventos | ID del Caso

-documentation-

Plan de Prueba : plan 1 [Versión : build 2]

Filtro de Navegación y Opciones

- Testing / plan 1 (6)(3, 1, 1, 0)
  - a (5)(2, 1, 1, 0)
    - ab
    - bdf
    - dsfgh
    - sdb
    - asdf
  - b (1)(1, 0, 0, 0)

Suite de Pruebas : a

Caso de Prueba ID tst-6 :: Versión: 1 asdf

No hay ningún probador asignado

Última ejecución (cualquier versión)

Aún sin probar

Última ejecución (versión actual) - \

Fecha	Versión Probad
22/09/2009 17:47:51	build 2 Testlink Adminis



## Situación anterior

- **Testing de Performance:**

- Para las pruebas de rendimiento se usaban herramientas como LoadRunner y Jmeter. Con estas herramientas la posibilidad de automatizar las pruebas es limitada.

```

Action()
{
    lr_start_transaction("Login_y_Home");

    web_set_user("{User}",
    lr_decrypt(lr_eval_string("{Password}")),
    "indraweb.indra.es:443");

    // web_set_certificate_ex(
    // "D:\PROYECTOS\INDRAWEB 2.0\Certificados\indra509.cer");
    web_set_certificate_ex(
    // "CertFilePath=d:\INDRA.PEM",
    // "CertFormat=PEM",
    // "KeyFilePath=c:\ncpm.pem",
    // "KeyFormat=PEM",
    // "Password=Yo8JQ6i",
    // "Password=indra",
    // "CertIndex=1",
    // "LAST");

    web_url("indraweb.indra.es",
    "URL=https://indraweb.indra.es/",
    "Resource=0",
    "RecContentType=text/html",
    "Referer=",
    "Snapshot=t1.ini",
    "Mode=HTML",
    "LAST");
}
    
```

Group Name	Script Path	Quantity	Load Generators
Indraweb_ie8_login	D:\Proyectos\Indraweb\scripts\Indraweb_IE8_Login	20	localhost

**Scenario Schedule**

Schedule Name: Schedule 1

Schedule by: Scenario (selected) / Group

Run Mode: Real-life schedule (selected) / Classic schedule

**Global Schedule**

Total: 20 Users

Action	Properties
Initiate	Initiate all Users simultaneously
Start Users	Start 20 Users: 1 every 01.00.03 (HH:MM:SS)
Duration	Run for 00:58:00 (HH:MM:SS)
Stop Users	Stop all Users simultaneously

**Interactive Schedule Graph**

The graph shows a constant number of 20 users over a period of 58 minutes.





**indra**

Crean una potente alianza para cubrir el área de calidad y pruebas que demanda el mercado desarrollando prototipos y proyectos basados en la plataforma Jazz y mas concretamente bajo

**RATIONAL**

**REQUISITEPRO, QUALITY MANAGER, Functional Tester , Performance Tester**

**Rational.** software

Powered by



# Plataforma de Testing



## Caso Practico

En el ejemplo siguiente, se muestran los datos clave en la ejecución de un proyecto real sin realizar la fase de aseguramiento de la calidad sin un equipo especializado y otro proyecto en el cual sí se incorporó dicho equipo bajo el modelo de ciclo de vida completo.

Proyecto	Elementos	Recursos	Recursos Pruebas	Duración Prevista	Duración Real	Incidencias
Proyecto 1	90 Ventanas 1.140 Programas	20	0	3 Años	4.2 Años	1.923 (126 tras Implantación)

Proyecto	Elementos	Recursos	Recursos Pruebas	Duración Prevista	Duración Real	Incidencias
Proyecto 2	56 Ventanas 920 Programas	13	4	1,5 Años	12 meses	178 (0 tras Implantación)



## Resultados

- RQM ofrece a todos los miembros de un equipo, las herramientas para obtener los informes y resultados necesarios para responder a la pregunta ....

¿Se puede entregar el software?



## Conclusiones

- IBM Rational presenta al equipo de pruebas una oportunidad ideal para evolucionar en los procesos de calidad de software.
- Se cubre desde el principio hasta el final todo el ciclo de vida de una sistema. Desde la captura de requisitos hasta la entrega del software.
- RQM integra en una sola aplicación las utilidades necesarias para gestionar online el testing funcional y de performance de un sistema y gracias a su interfaz de usuario sencilla permite una colaboración online de todos los integrantes del equipo en tiempo real.
- RFT permite con la automatización de pruebas un ahorro de tiempo en las pruebas de regresión y aceptación del usuario.
- RPT permite una grabación y ejecución más ágil de los scripts y escenarios. También viene integrado todo en una misma herramienta, lo que facilita el trabajo.



## Conclusiones

Implantar un proceso de Testing en los sistemas en desarrollo garantiza el retorno sobre la inversión en TI y ofrece beneficios tangibles:

- ✓ Implantar un proceso de calidad del software consigue reducciones de hasta un 80% en el coste del mantenimiento del sistema en el entorno de explotación
- ✓ Reducción del 30% del coste total de la fase de pruebas
  - ✓ Tiempo de entrega reducido en un 15%.
  - ✓ Asistencia técnica y riesgos de negocio reducidos.
- ✓ Niveles de calidad mejorados mediante la prevención de más del 97% de graves defectos



# Gracias por su Atención



[www.indra.es](http://www.indra.es)

