

I Jornadas de Informática y Telecomunicaciones de la APBA
DevOps - Integración continua en el desarrollo de software
Junio - 2015



índice

somos atSistemas



algunos conceptos

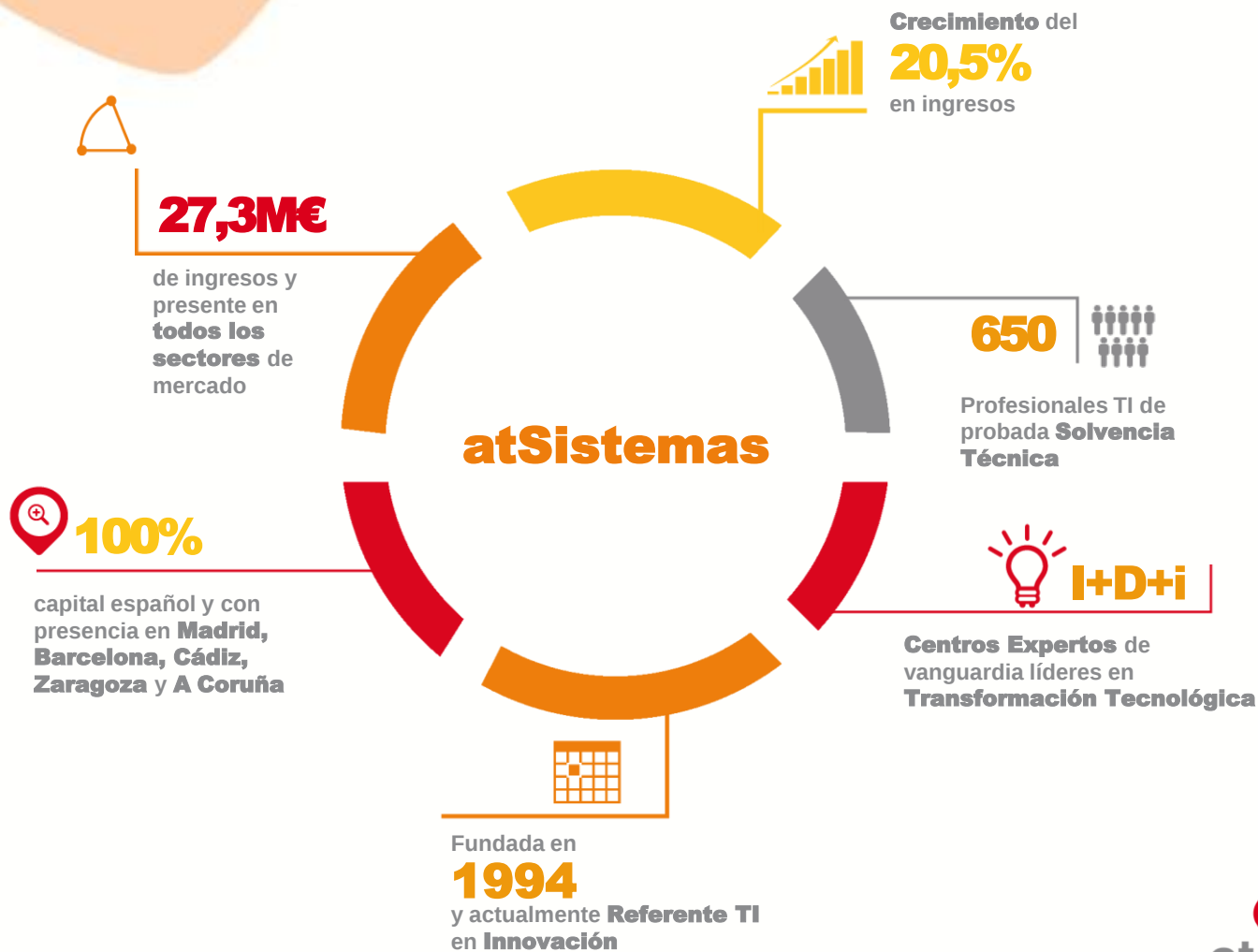


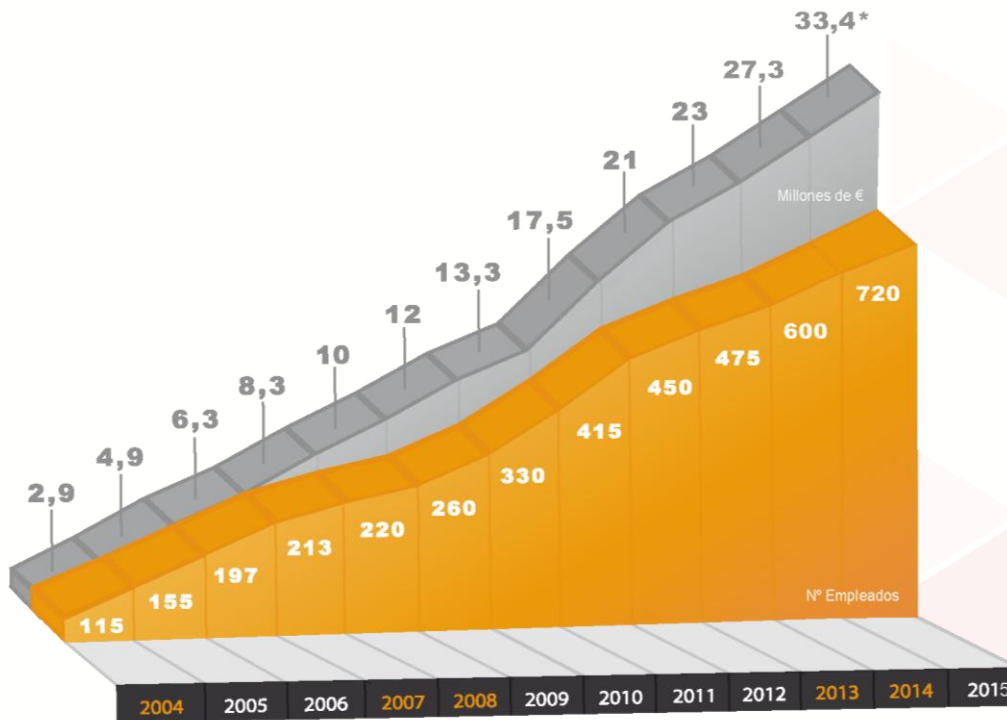
herramientas



conclusiones







* Previsión año fiscal 2015

2013
Apertura Oficina
A Coruña

2008
Apertura Centro de
Servicio de Cádiz

2004
Décimo Aniversario

2013
Apertura Oficina
Zaragoza

2007
Apertura Oficina
Barcelona

solidez financiera



Beneficios sostenibles a lo largo de todos los ejercicios desde su creación en 1994



Reinversión, todos los años, de gran parte de los beneficios para financiar el crecimiento



Bajo nivel de endeudamiento con entidades de crédito, con excelente calificación y total facilidad de acceso al crédito



Amplia holgura financiera que permite hacer frente a las obligaciones y autofinanciar el crecimiento

composición del balance



Activo No Corriente 7,33%

Activo Corriente 92,67%



Patrimonio Neto 72,06%

Pasivo Neto Corriente 0,88%

Pasivo Corriente 27,06%

ratios

liquidez
3,08

solvencia
1,19

endeudamiento
0,46

disponibilidad
0,26

Proyectos Cerrados, Servicios Gestionados o Servicios Profesionales

Asesoramos y Definimos

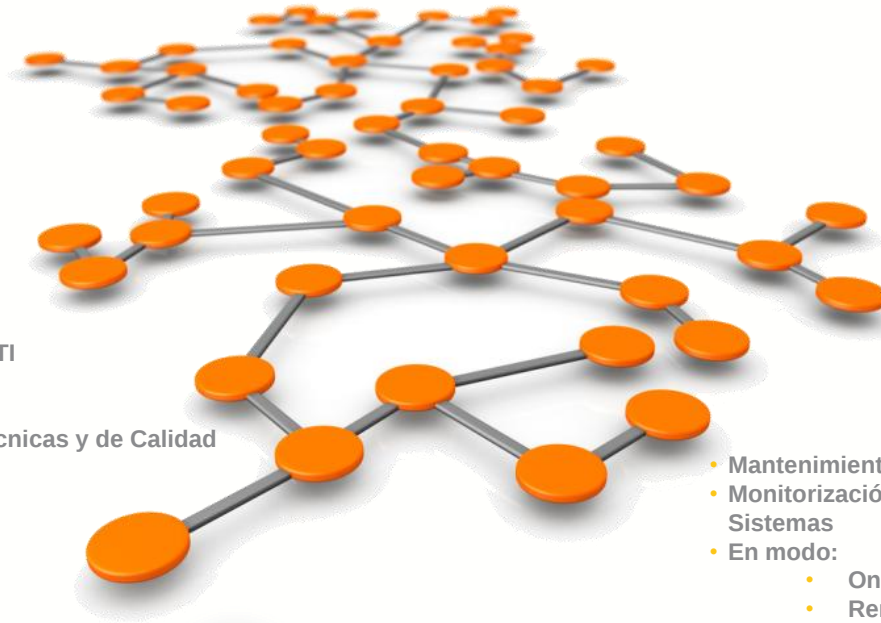
- Definición de Estrategia TI
- Consultoría Tecnológica
- Benchmarking TI
- Oficinas de Proyecto, Técnicas y de Calidad
- Mejora de Procesos TI
- Gestión del Cambio TI

Construimos e Integramos

- Arquitectura TI
- Integración de Sistemas en base a SOA
- Gestión de Procesos de Negocio
- Desarrollo de Aplicaciones
- Testing & Quality Assurance
- Desarrollo de Soluciones y Frameworks

Gestionamos

- Mantenimiento de Aplicaciones
- Monitorización y Administración de Sistemas
- En modo:
 - On Site
 - Remoto
- Cloud Computing



nuestra cadena de valor

Modelo de Relacional Comercial

- Gerentes de Cuentas
- Gerentes Sectoriales
- Gerentes de Líneas de Negocio
- Equipos de Pre-Venta

Modelo de Servicios

- Oficina Técnica y de Proyecto
- Centros Expertos
- Líderes Tecnológicos
- Project Managers
- Service Managers
- Centro Near-Shore

Modelo Tecnológico y Metodológico

- Herramientas Colaborativas
- Herramientas de Gestión del Servicio
- Herramientas de Automatización
- Frameworks de Desarrollo
- Normas y Mejores Prácticas

Modelo de Mejora Continua

- Sistema de Gestión de la Calidad
- Encuestas de Satisfacción al Cliente



Modelo de Gestión de Talentos

- Captación de Talento
- Desarrollo de Talento
- Plan de Carrera
- Plan Formativo
- Formación y Certificación

nuestros partners **estratégicos**

IT **Innovación**
Con los principales
actores tecnológicos
del mercado

 **Especialización**
Foco en áreas
tecnológicas específicas

 **Compromiso**
Programas de certificación

 **Propuestas conjuntas**
Soluciones de valor y
equipos de trabajo conjuntos

Nuestros partners estratégicos



Microsoft Partner
Silver Application Development



Microsoft

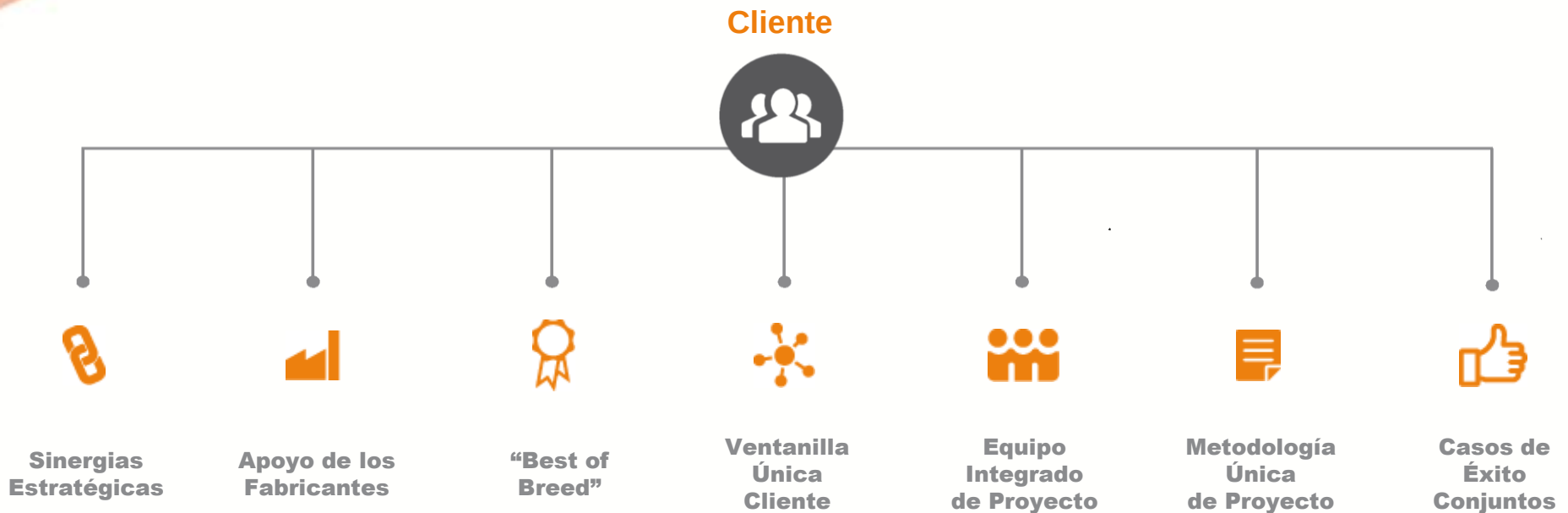


Otros partners



nuestras alianzas estratégicas

Alianzas Estratégicas que nos permiten dar respuesta a Grandes Proyectos de Transformación



nuestros valores



índice

somos atSistemas



algunos conceptos



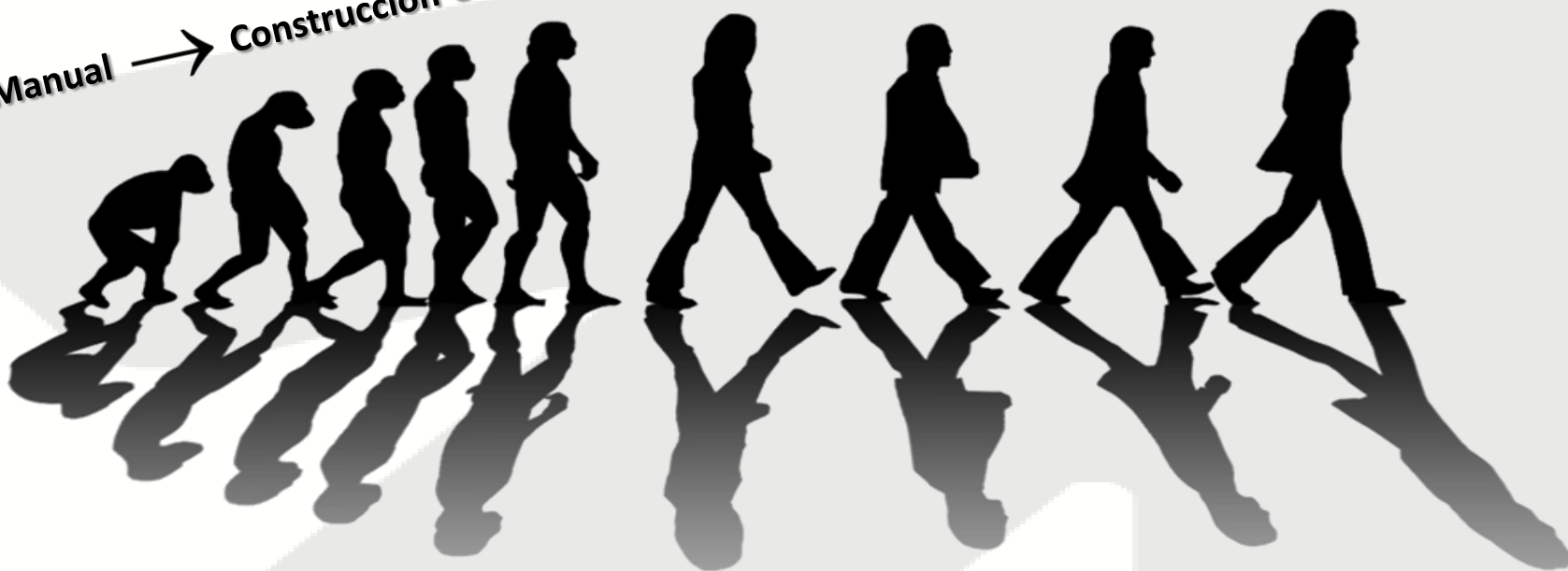
herramientas



conclusiones



Manual → Construcción Continua → Integración Continua → Entrega Continua





sin integración **continua**

Pongamos un ejemplo:

5 programadores durante 3 meses programando
No se ha hecho **ni una sola integración**



Tienes dos días para entregar en producción
¡¡ QUE COMIENZE EL JUEGO !!

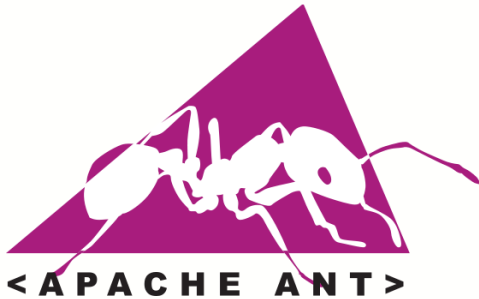


construcción **continua**

Realización de **tareas mecánicas y repetitivas**, normalmente durante las fases de compilación y de build.

¿Qué permite?

- ◆ Automatización de la compilación
- ◆ Multiplataforma (evolución del Make)
- ◆ Fichero Build.xml



Proyecto que contiene:

- objetivo (mínimo 1) que puede contener varias tareas (task) que son fragmentos de código a ejecutar.
- Propiedades (varias) que constan de nombre y valor y son usadas para asignar valores a los atributos de los task.

Limitaciones

- ◆ Dificultades para usuarios noveles
- ◆ Necesidad de definir su ciclo de vida
- ◆ **NO gestiona dependencias**



construcción **continua**

Realización de **tareas mecánicas y repetitivas**, normalmente durante las fases de compilación y de build.

maven

¿Qué permite?

- ◆ Sistema de construcción uniforme (POM.xml)
- ◆ Gestión automática. Lista cambios (Changelog)
- ◆ Gestión automática. Dependencias
- ◆ Gestión automática Informes ejecución Pruebas Unitarias
- ◆ Best Practices
- ◆ Introducir nuevos servicios de manera sencilla

a resaltar

- ◆ Generación de aplicaciones a partir de arquetipos
- ◆ Gestión de la configuración dependiendo del entorno
- ◆ Gestión de dependencias
- ◆ Control de Versiones
- ◆ Ciclo de vida: Compile (genera class) > Test (junit) > Package (genera jar) > Install (deja disponibles) > Deploy (servidor remoto)



construcción **continua**

Realización de **tareas mecánicas y repetitivas**, normalmente durante las fases de compilación y de build.

¿Qué permite?

- ◆ Flexibilidad de un lenguaje de programación
- ◆ Definición de nuevas tareas
- ◆ Gestión de dependencias
- ◆ Multi-Módulo
- ◆ Ficheros Builds más concisos
- ◆ Soporte para entrar lógica



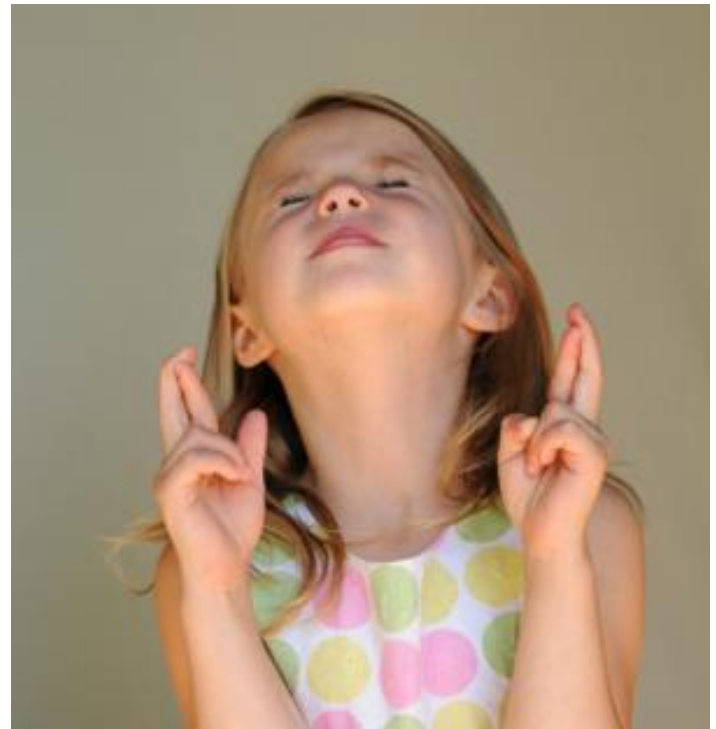
a resaltar

- ◆ Integración con Ant (Antexecutor) y Maven
- ◆ Reutilización. Los archivos se pueden modularizar, importar, compartir
- ◆ Extensibilidad. Podemos ampliar la funcionalidad fácilmente.
- ◆ Integración con sistemas de IC.





Poner en producción una funcionalidad
sin pruebas unitarias





¿Qué es un unit test?

Un test unitario sirve para examinar cada unidad de un código de **forma independiente**



Las reglas del buen TEST

- ◆ Cada test debe ser independiente al resto
 - ◆ Una única afirmación lógica por cada test unitario.
 - ◆ Comprueba solo una unidad de código cada vez
 - ◆ Solo si la arquitectura lo exige, recurriremos a los mocks.
 - ◆ Evitar o reducir al máximo las condiciones previas a la ejecución de una suite
- ◆ Nombra los tests de un modo claro, consistente y unívoco



entrega **continua**

Es la evolución lógica de la integración continua. Llevar la integración continua a los entornos productivos.

Despliegue continuo $\neq$ Entrega continua

Herramientas

Despliegues Automatizados

- Idénticos en ambientes de pruebas y producción
- Ejecución en una línea de comandos
- Repositorios de binarios
- Herramienta de build

Pruebas automatizadas

- Cobertura
- Robustez del conjunto de pruebas
- Manejo de los Data Sets

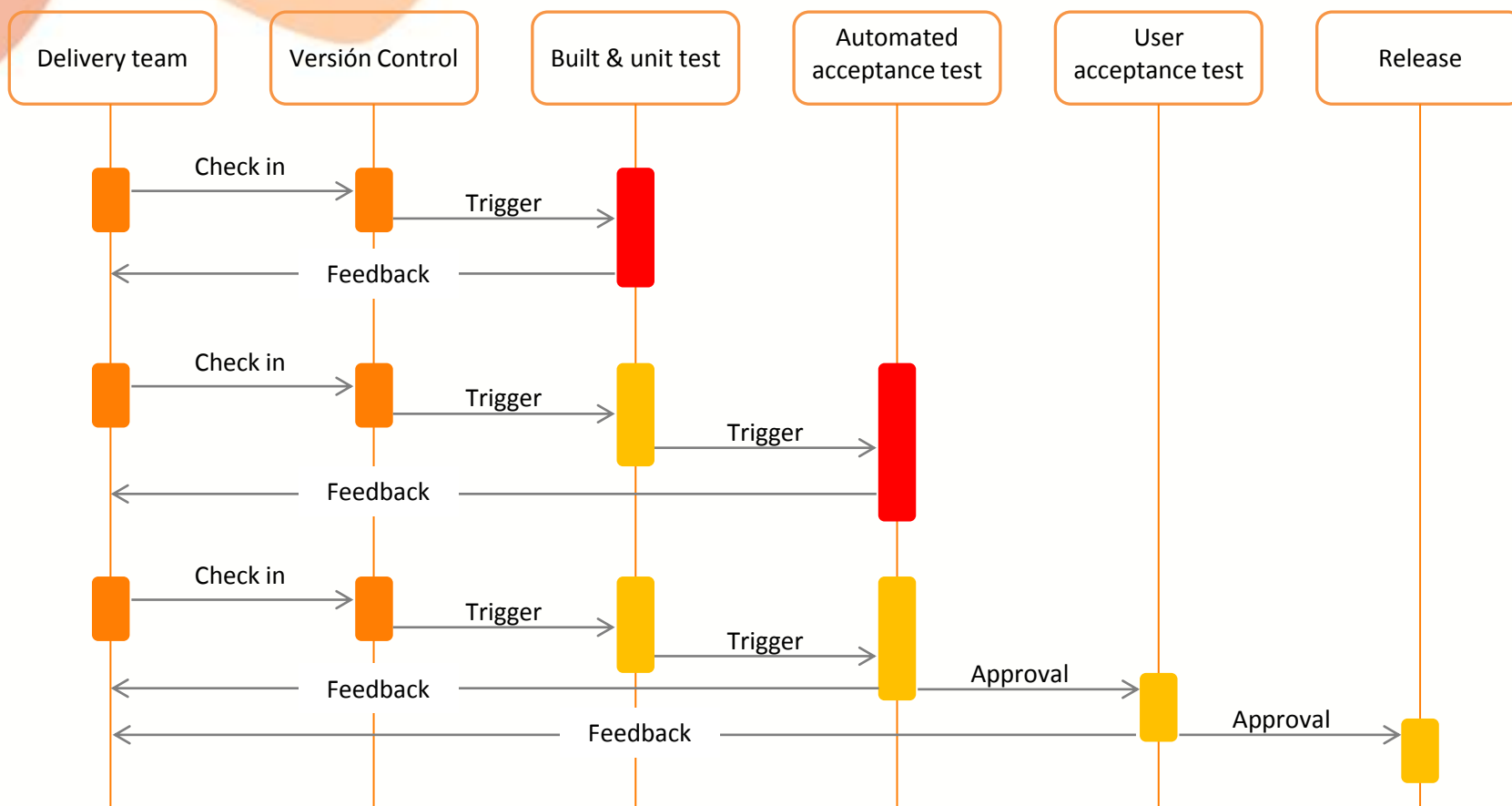
Disciplina en el control de versiones

Integración Continua

Madurez del proceso



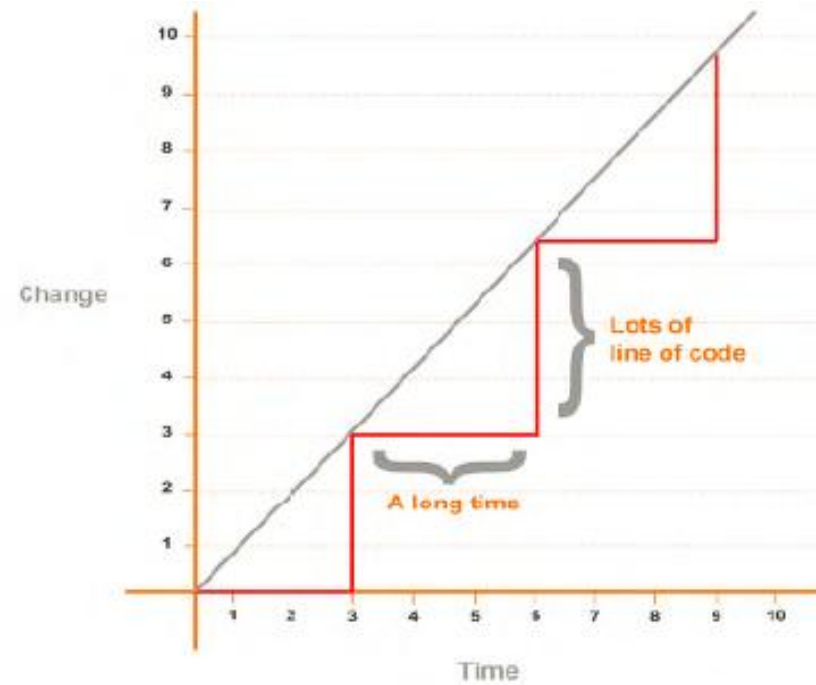
entrega **continua**



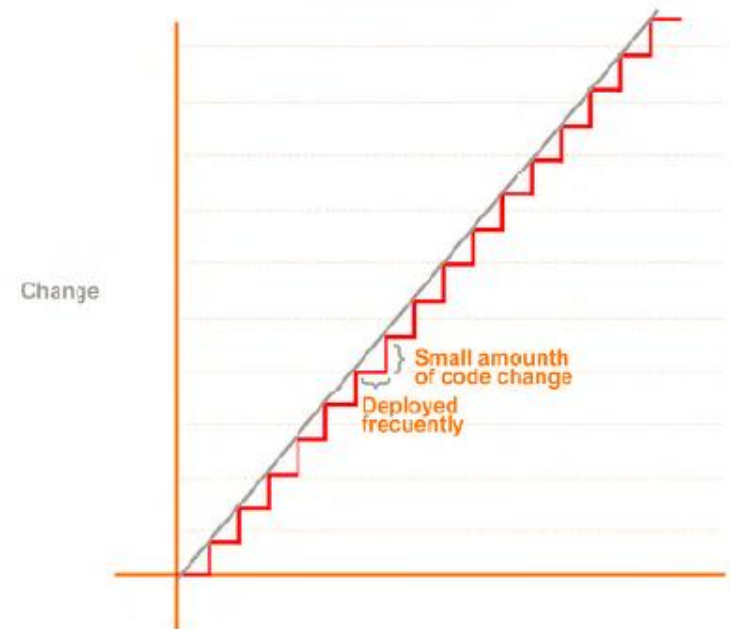


Ventajas de la entrega continua

- Cada entrega es **predecible**
- “Time to resolve” Los problemas aparecen y son abordados en fases más tempranas.
- **Total flexibilidad** para adoptar nuevos cambios
 - Aparece el concepto **Devops**



≠





!!!Imprescindible!!! Definir una política de versionado

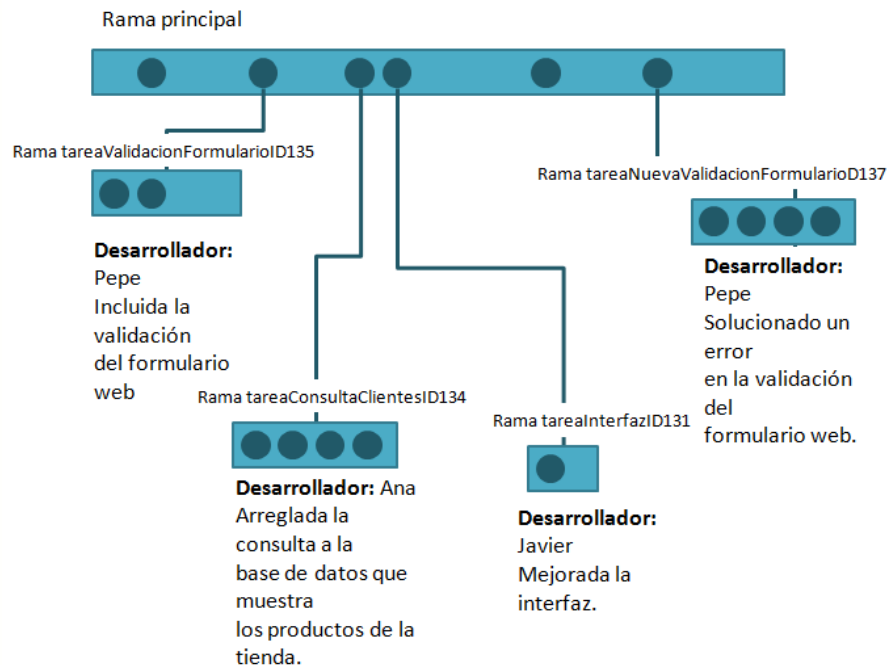


- ◆ ¿Desarrollaremos una historia de usuario por rama y tendremos una rama master o principal de integración? ¿El servidor de integración continua monitorizará los cambios en esa rama master?
- ◆ ¿Optaremos por una estrategia de rama única que será la que monitorizará el servidor de integración continua?
- ◆ ¿Cuándo se cierra la versión de código? ¿Cómo numeramos las versiones?
- ◆ ¿Además de eso tendremos rama por equipo de desarrollo? ¿Ramas de estabilización y bug-fixing antes de pasar a producción?
- ◆ El servidor de integración continua también monitorizará las ramas de historias de usuario que se vayan creando?



Recomendaciones para un desarrollo con metodologías ágiles (Scrum)

- ◇ Crear una rama por tarea, historia de usuario a desarrollar.
- ◇ Al terminar, con el código probado y estable, se integra en la rama principal o línea de desarrollo principal
- ◇ Nombres descriptivos mejoran la trazabilidad del código y establecen el propósito de la rama



Permite

- ◇ Subir código de forma independiente sin miedo a estropear el código principal
- ◇ Reducir la propagación de errores
- ◇ Garantizar la estabilidad del código de la línea principal
- ◇ Independencia de tareas



entrega **continua**

Algunos datos relevantes

flickr

- 3.000 millones de fotos alojadas, con una subida media de 3.000 fotos cada minuto.
- Varios despliegues diarios. Han llegado a informar en su web la hora del último despliegue.

amazon

- Sirve a 137.000 usuarios por semana con 34.000 millones de dólares de beneficios anuales.
- Libera en producción una versión cada 11,6 segundos de media, pudiendo afectar a 30.000 servidores (datos de 2011).

facebook

- Más de 650 millones de usuarios activos. 1 de cada 13 personas de la tierra, tiene facebook.
- Al compilarse el código fuente se genera un binario de 1,5Gb, que se despliega en producción al menos una vez al día con cambios menores y una vez a la semana una versión mayor.

índice

somos atSistemas



algunos conceptos



herramientas



conclusiones



procedimiento entrega continua

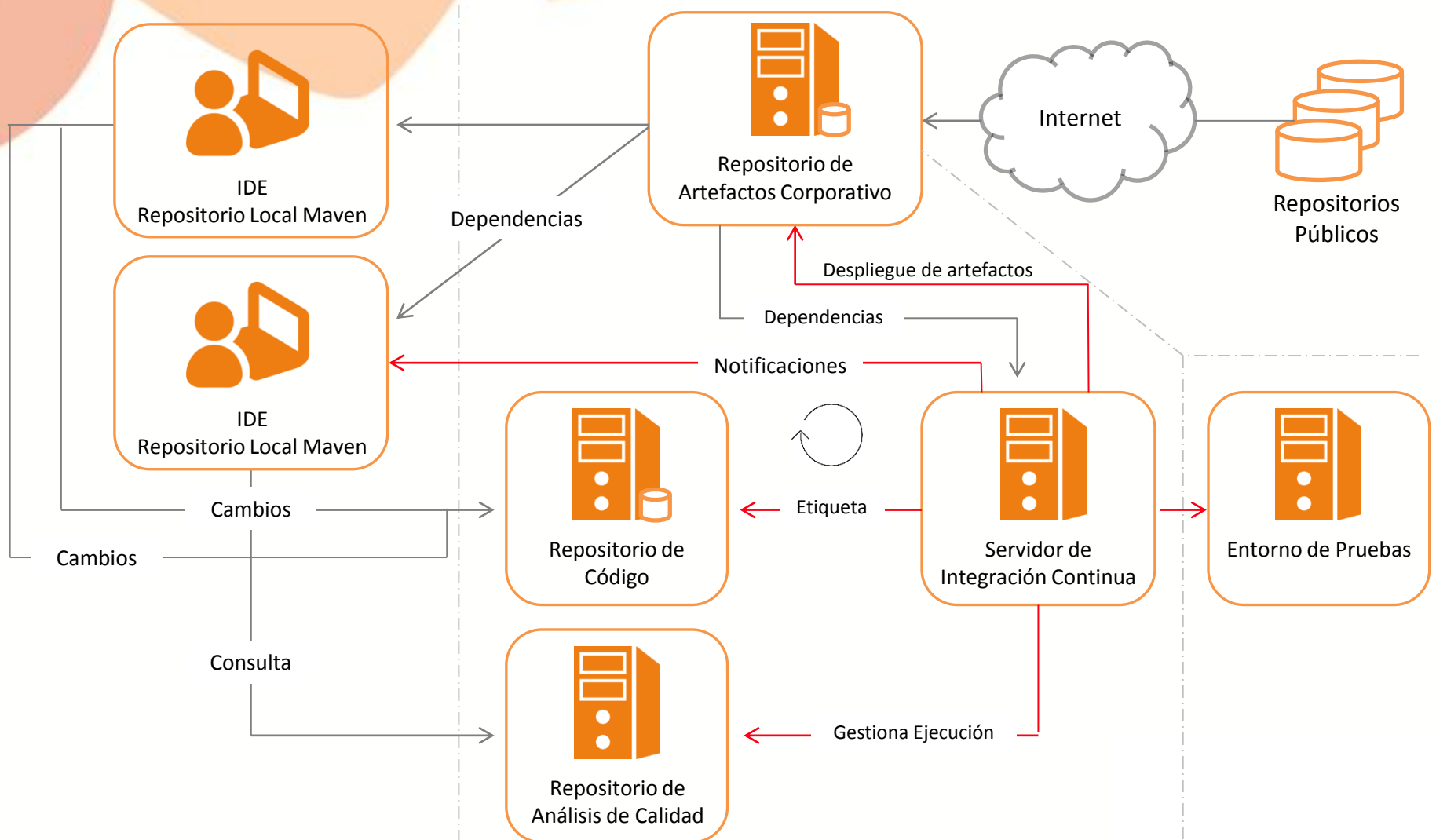
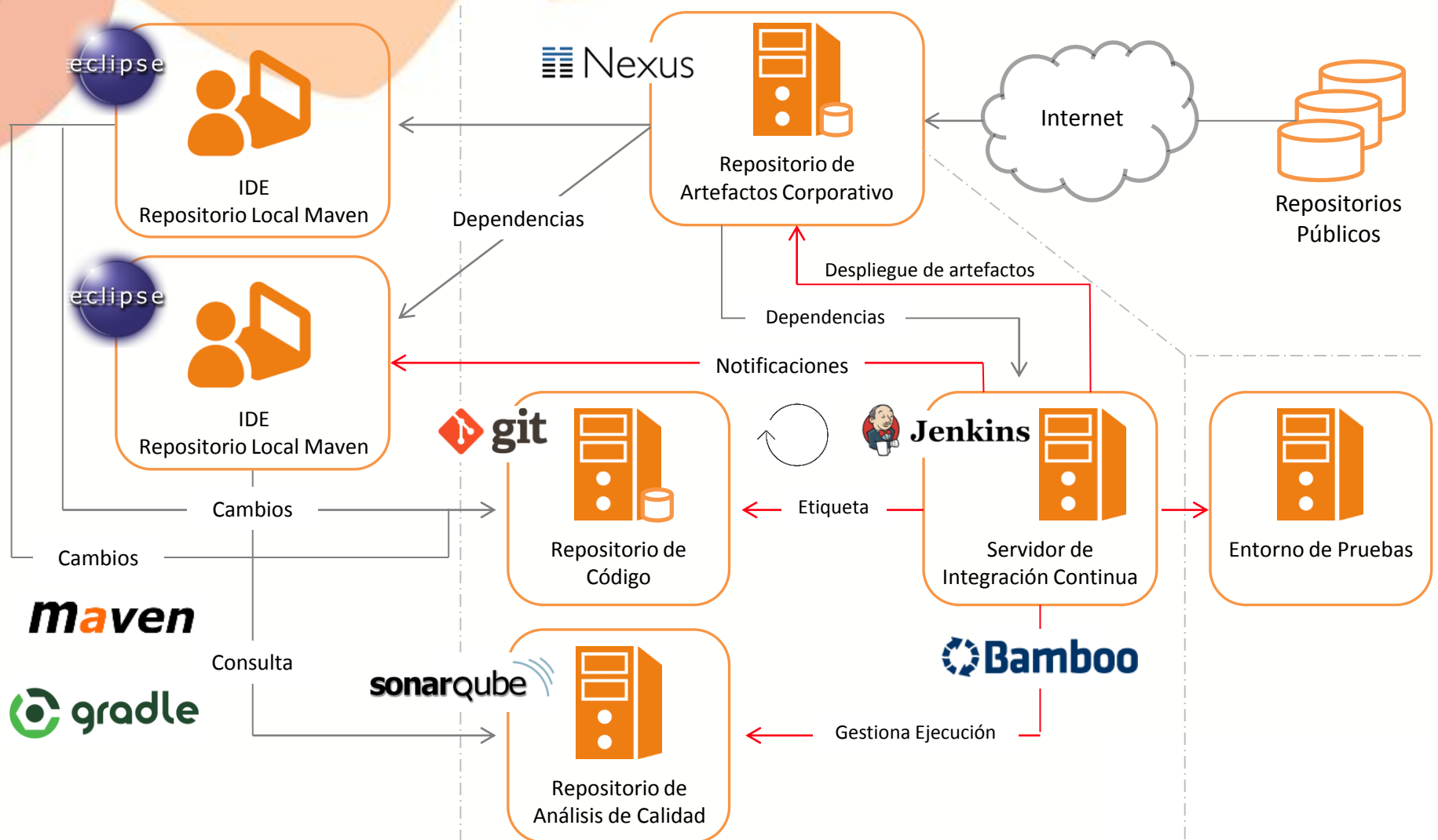
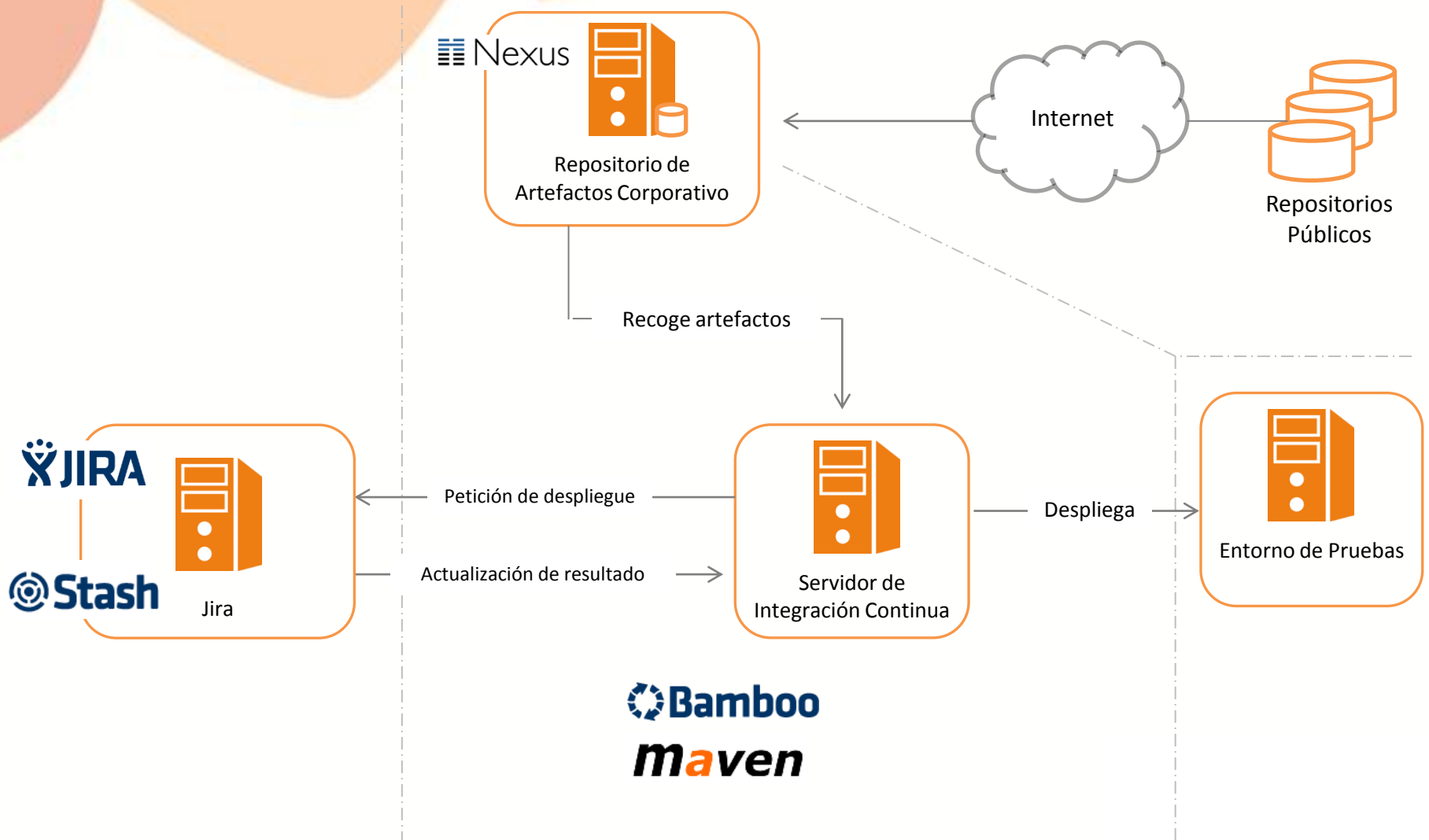


diagrama herramientas



arquitectura caso práctico





☰

JIRA

Dashboards ▾

Projects ▾

Issues ▾

Agile ▾

Capture ▾

Create issue

Quick Search

?

⚙

Scrum: Teams in Space

Plan

Work

Report

Board ▾

⌵

SPRINT: Sprint 1 ▾

QUICK FILTERS: Product UI Server Only My Issues Recently Updated

4 To Do

4 In Progress Min 3 Max 5

1 Code Review

4 Done

▼ TIS Developer Love 4 issues

TIS-46

↑ Update LocalTransportC to handle

6

TIS-40

↑ Update FlightController to handle

6

▼ Everything Else 9 issues

TIS-45

↑ Email non registered users to sign

2

TIS-43

↑ Extend booking experience in UI to include

9

TIS-44

↑ Reward Customers an extra 5-10%

3

TIS-49

↑ Draft network plan for Mars Office

5

TIS-68

↑ Homepage footer uses an inline style -

TIS-42

↑ Extend booking experience in UI to include

9

TIS-67

↑ Developer Toolbox does not display by

TIS-8

↑ Requesting available flights is now taking >

TIS-69

↑ Add a String anonymizer to TextUtils

TIS-49

↑ Engage Saturn Space Tours for Group Travel

TIS-56

↑ Add pointer to main css file to instruct users

Teams in Space / TIS-67

Developer Toolbox does not display by default

People

Reporter: Jennifer Evans

Assignee: Jennifer Evans

Created: 30/Jan/14 3:43 PM

Updated: 03/Feb/14 3:22 PM

Development

5 branches Updated 13/Feb/14 5:4

3 commits Latest a day ago

1 pull request Updated a day ago

1 build Latest 13/Feb/14 6:29

Create branch

[Plan Summary](#) [Branches](#) [Recent Failures](#) [Quarantined Tests](#) [Tracking](#) **Issues**[Run](#) [Actions](#)

JIRA Issues

Showing Last 25 builds

This page lists the JIRA issues that have been linked to builds related to this plan, ordered by last built. This includes issues specified in commit messages, labels or comments for builds, as well as issues manually linked to builds.

JIRA Issues (15 issues)

Type	Key	Summary	Status	Assignee	Fix versions	Related Builds	Last Built
	BDEV-503	Plan Summary pie chart doesnt show in ie 8	To be reviewed	Jason Berry		21 related builds	Last built: 4 hours ago
	BAM-11045	Plan Variables - Descriptive text - Review if plan variables are visible to all builds or just jobs in plan	Resolved	Jesper	4.1.1	20 related builds	Last built: 5 hours ago
	BDEV-397	Redirect loop when clicking top-left nav bar					
	BDEV-346	Duplicate authors in database export cause import to fail from HSQL to MySQL					

Activity

[All](#) [Comments](#) [Work Log](#) [History](#) [Activity](#) [Source](#) [Reviews](#) **Builds**

Tardigrade Server

Showing 25 of 29 builds.

- +core+ Bamboo - CI Tests - atlassian_bamboo_4_1_branch #23**
Relates to: [21 issues](#)
Code changed | Ran: 3 hours ago
Duration: 30 minutes | Tests: 2375 passed
- +core+ Bamboo - Stable CI Tests #971**
Relates to: [21 issues](#)
Code changed | Ran: 3 hours ago
Duration: 29 minutes | Tests: 2356 passed
- +core+ Dogfooding - Bamboo Bitbucket - atlassian_bamboo_4_1_branch #23**
Relates to: [21 issues](#)
Code changed | Ran: 4 hours ago
Duration: 4 minutes | Tests: No tests found



[Build Summary](#) [Tests](#) [Changes](#) [Artifacts](#) [Logs](#) [Metadata](#) [Issues](#) [Sandbox](#)

[Run](#) [Actions](#)

Build Result Summary

Details

Trigger Code has been updated by [Ashwin Srinivasan](#).

Revision [b935f9e...](#)

Completed 25 May 2012, 6:21:29 PM – 2 days ago

Duration 41 minutes

Successful since [#111](#) (12 hours before)

[Labels](#) None [✎](#)

Test Summary

4,020 tests in total

0

New Failures

0

Existing Failures

0

Fixed

3

Quarantined

Code Changes

[See full change details](#)



[Ashwin Srinivasan](#)

25 May 2012, 5:37:23 PM

[BDEV-327](#), CR-BAM-4225: Changed wording of EC2 must be enabled warning message

JIRA Issues

[See full issue details](#)

Issue	Description
BDEV-327	Home page pointing to EC2 config

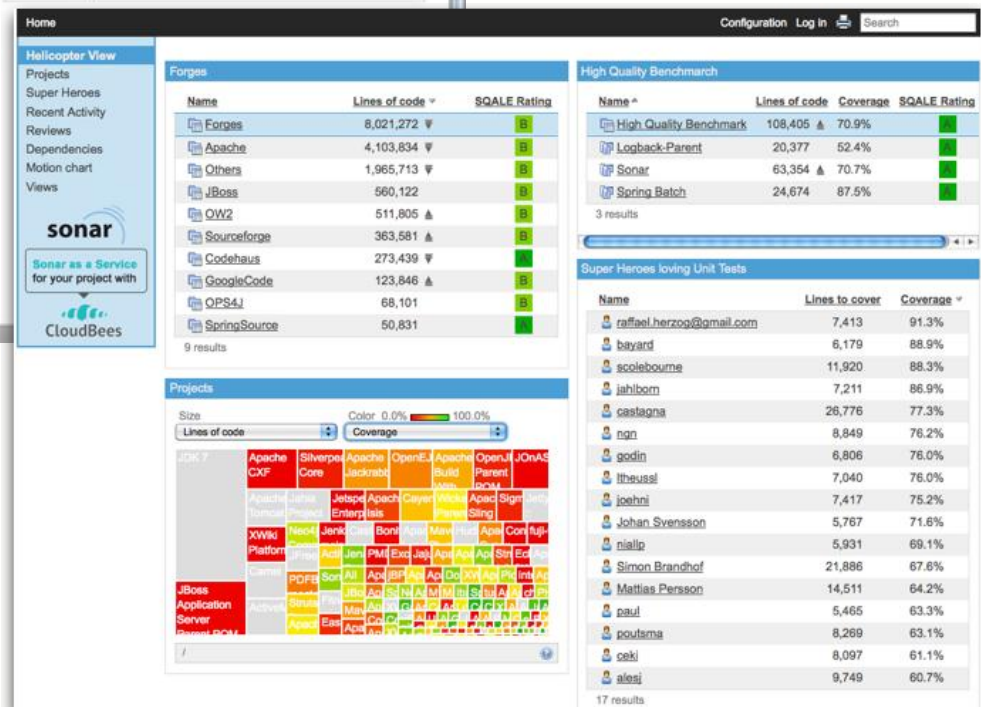
Comments

No comments on this Build.

[Comment](#)

herramientas en detalle

sonarqube



índice

somos atSistmas



algunos conceptos



herramientas



conclusiones







Irene Comalada Juanola

Operations Manager

icomalada@atsistemas.com

Twitter: <https://twitter.com/irenecomalada>

Linkedin: <https://es.linkedin.com/in/irenecomalada>

**¡MUCHAS GRACIAS POR SU
ATENCIÓN!**



**APLAUDAN Y NO HAGAN
NINGUNA PREGUNTA**



MADRID

BARCELONA

CÁDIZ

ZARAGOZA

A CORUÑA

902 888 902