



MECALUX

Software Solutions

Arquitectura y Calidad de un producto Software – EasyWMS

Arquitectura de un producto real y extensible

Parte I

Logística y Almacenes

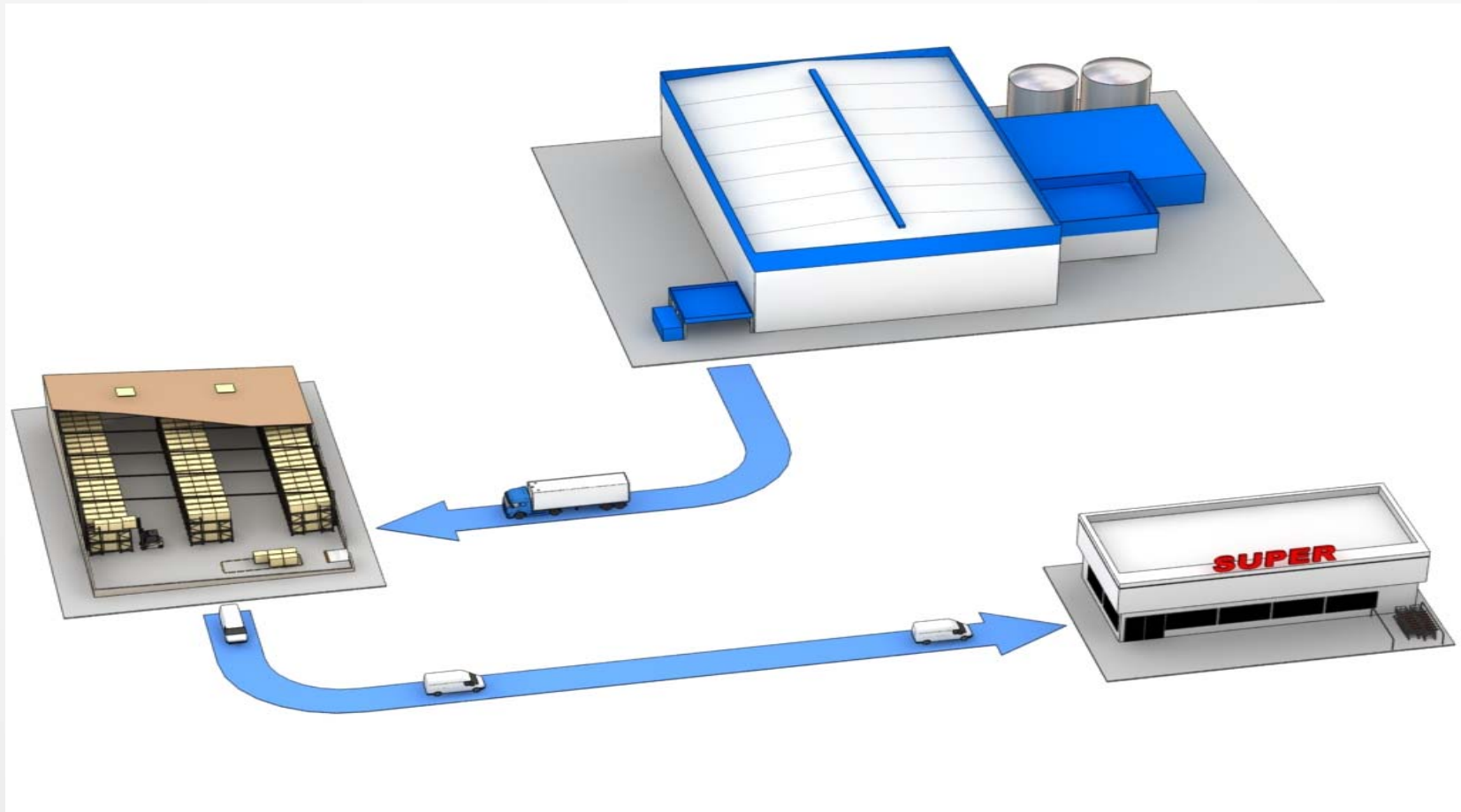
¿Para qué sirve un almacén?



Su uso es muy diverso, depende de las necesidades de la organización:

- Evitar costes de rotura de stock
- Permite operativas de organización (etiquetado, inventario) de forma mucho más controlada y cómoda que teniendo que hacerlo en el destino final (tiendas, por ejemplo).
- **Abastecimiento de cadena de suministro de forma eficiente**

Almacenes y Cadena de Suministro (Supply Chain)



No perder de
vista los objetivos

- Hacer más con menos
- Hacer cosas mejor (o en menos tiempo)
- Obtener más beneficios

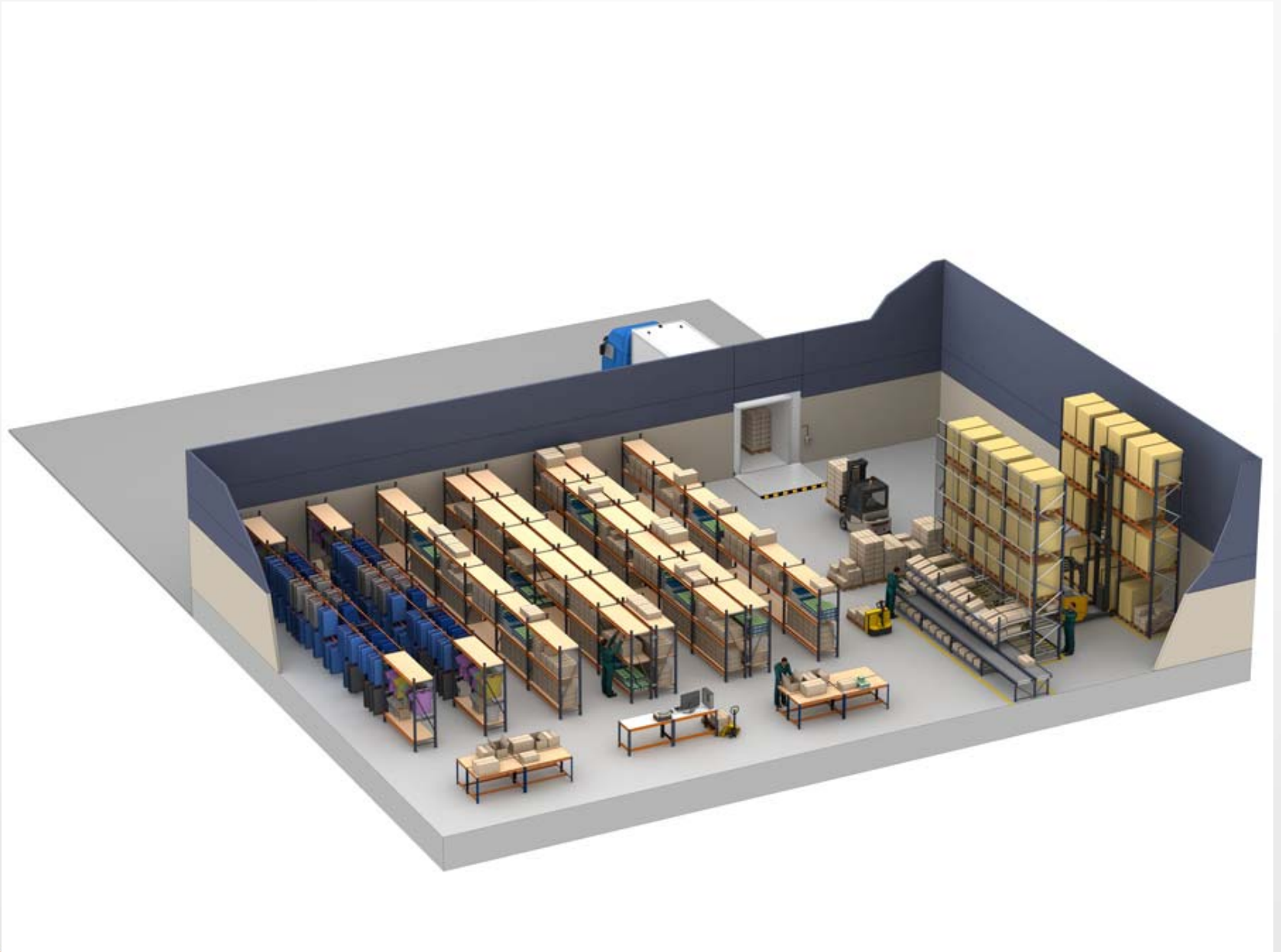
• 0P1GUGL W92 D6U6JIC102

ERP y otros subsistemas empresariales



El almacén puede integrarse en una organización que es la que gobierna la cadena de suministro

- El almacén y su contenido son recursos de una empresa / organización, luego son susceptibles de ser gestionados por un ERP (Enterprise Resource Planning).
- Íntimamente ligados con Supply Chain
- Ejemplo: Los contenidos del almacén pueden ser directamente relacionados con las ordenes de compra – ventas a cliente, luego están relacionados con un CRM (Customer Relationship Management)



Tipologías de Almacenes - Automáticos



Tipologías de Almacenes por tipo de estantería, algunos ejemplos



Atributos buscados en un WMS

- Trazabilidad
- Interconexión
- Fiabilidad de las operativas
- Rapidez de implantación*
- Facilidad de uso*
- Escalable*
- **Adaptable a los cambios de mercado***

Retos que plantea

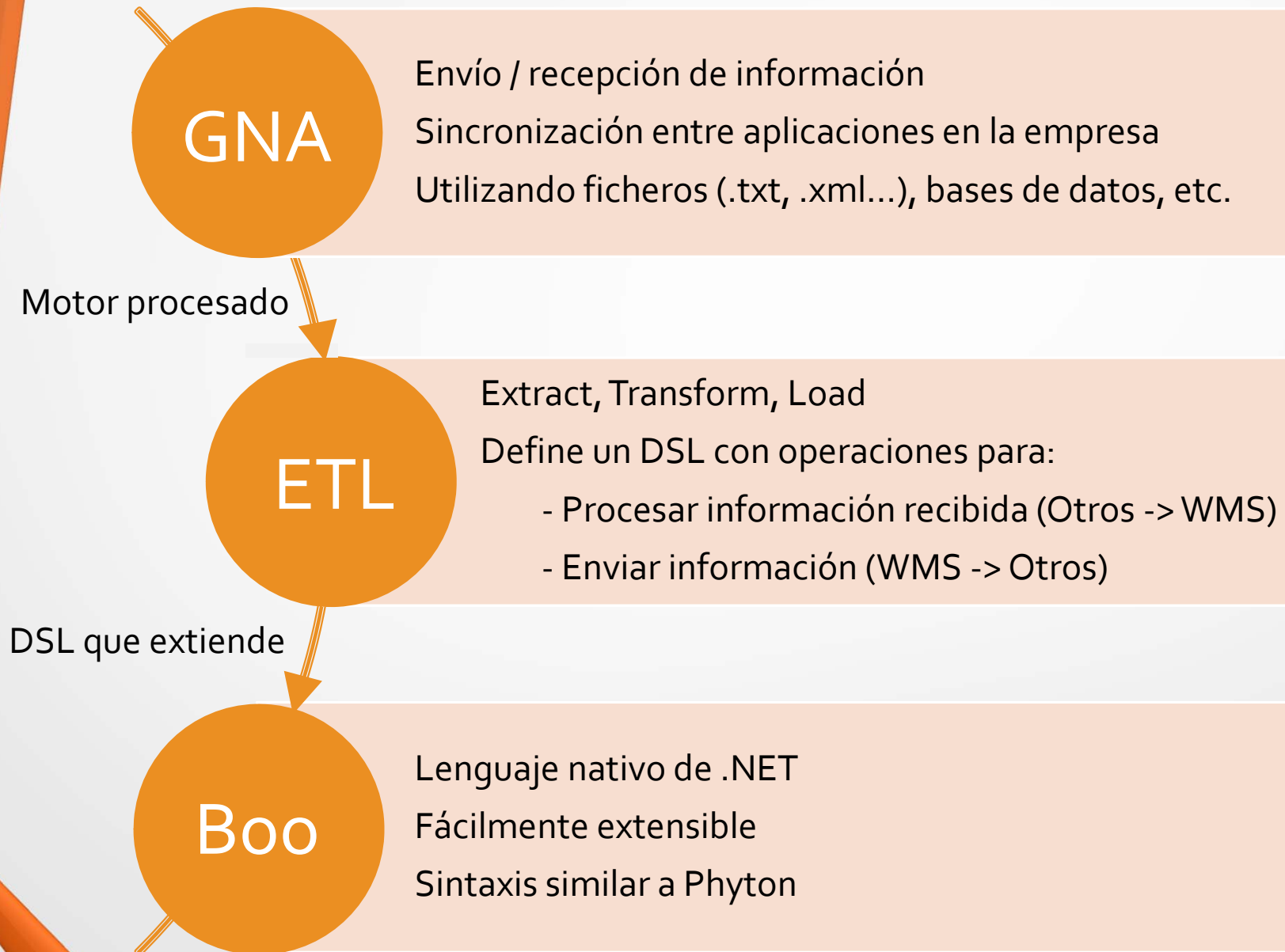
- Cantidad de datos a almacenar muy alta
- Arquitectura lo más abierta posible
- Herramientas de ayuda a la configuración
- Interface de usuario muy simple
- Infinitas posibilidades de configuraciones

Importante: El ciclo de vida del software, va unido al del mercado

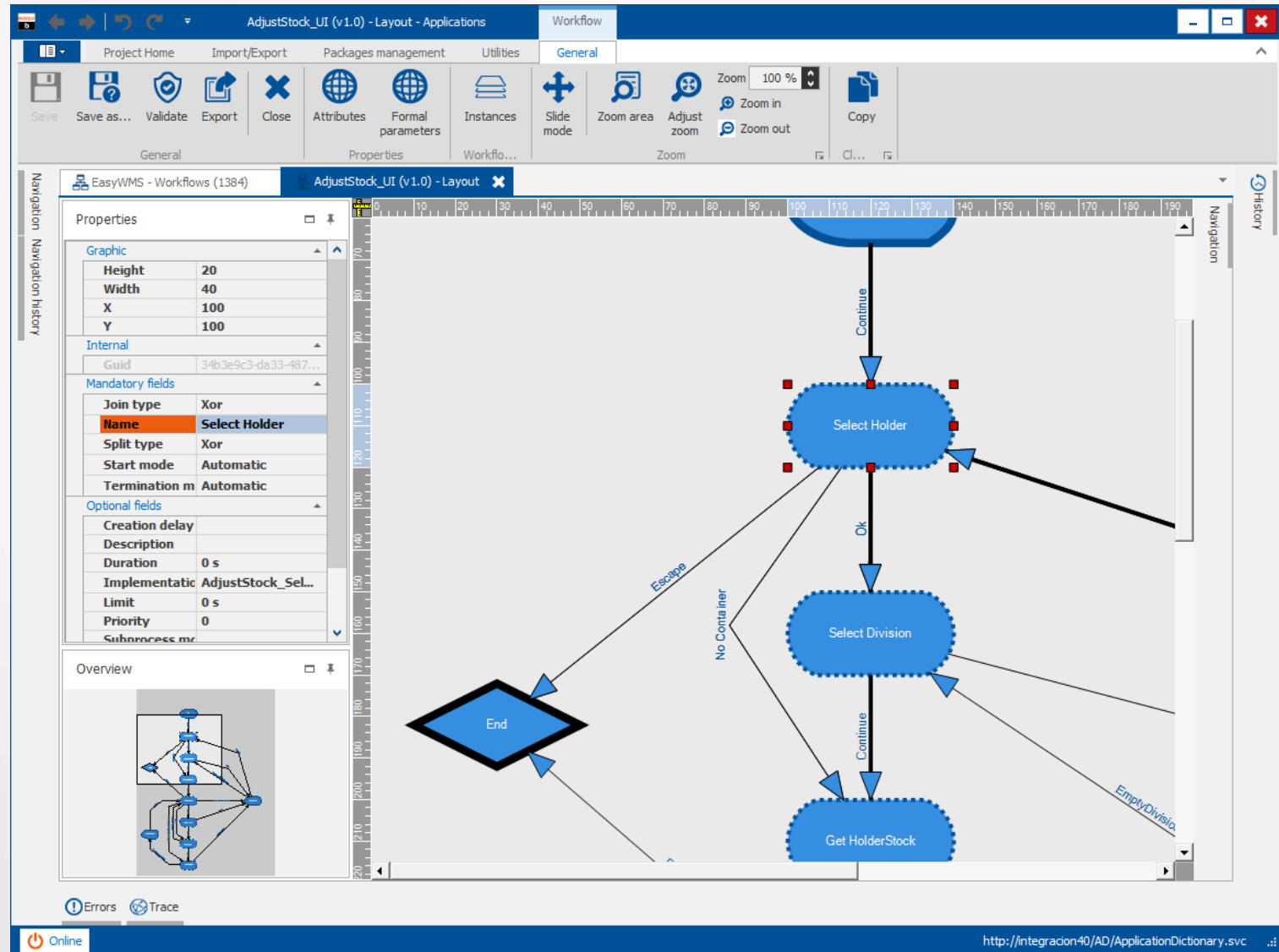
Parte II

Arquitectura de Easy WMS

Comunicaciones: Interconexión con sistemas empresariales



Flexibilidad y Adaptación al entorno: EasyBuilder - WorkFlows



Flexibilidad y Adaptación al entorno: EasyBuilder – Edición de vistas

View view - CarriersVList (1.0) - Applications

Project Home Import/Export Packages management Utilities View

Add parameter Add default parameters View uses Add attachments and notes Add command Add export Add group Add link Add link assistant Add massive edit Add print Add report Add audit

Parameters: CarriersVList Visibility: C... Actions: CarriersVList

Navigation

- test_tenant
 - Groups (4)
 - Jobs (13)
 - MenuItems (100)
 - Resources (163)
 - RFMenuItems (61)
- Applications
 - Common (v1.0)
 - EasyWMS (v1.0)**
 - Commands (1143)
 - Dialogs (434)
 - Entities (194)
 - Events (1275)
 - Field types (112)
 - Formats (1)
 - Lists (115)
 - Queries (912)
 - Records (158)
 - Relationships (33)
 - Reports (24)
 - Resources (17128)
 - Subscriptions (180)
 - Validators (11)
 - View groups (115)
 - Views (419)
 - Workflow actions (23)
 - Workflows (1384)
 - Writing model (126)
 - Notifications (v1.0)
 - SmartUI (v1.0)

EasyWMS - Workflows (1384) AdjustStock_UI (v1.0) - Layout EasyWMS - Views (419) View view - CarriersVList (1.0)

CarriersVList

- modeEdit
- modeNew
- Actions
 - Left
 - EditCarrier - Left (ActionLink)
 - NotesAttachSupplier - Left (ActionAttachNotes)
 - MassEditCarrier - Left (ActionMassiveUpdate)
 - PrintAll - Left (ActionPrint)
 - ExportCarriers - Left (ActionExport)
 - AgencyRemove - Left (ActionCommand)
 - Audit - Left (ActionAudit)
 - Right
 - AddCarrier - Right (ActionLink)
- CarriersVGrid (ViewGrid)
 - SupplierDescription
 - SupplierName
 - Code (ViewField)
 - Name (ViewField)
 - Description (ViewField)
 - NumSuppliers (ViewField)
 - NumAccounts (ViewField)
 - IsActive (ViewField)
 - Contact_Comment (ViewField)
 - Contact_Extension (ViewField)
 - Contact_Fax (ViewField)
 - Contact_Name (ViewField)
 - Contact_Phone (ViewField)
 - Address_ZipCode (ViewField)
 - Address_AddressLine (ViewField)

Cancel

Data

Attach entity	[EditValue is null]
Description	CarriersVList
Entity	Agency
Name	CarriersVList
Notes entity	[EditValue is null]
Row limit	100
Title	Agencias
View group	Agency

Filter

Filter script

Use filter property ☐

Internal

Sql

Sql order by	Code ASC
Sql where	

Visibility

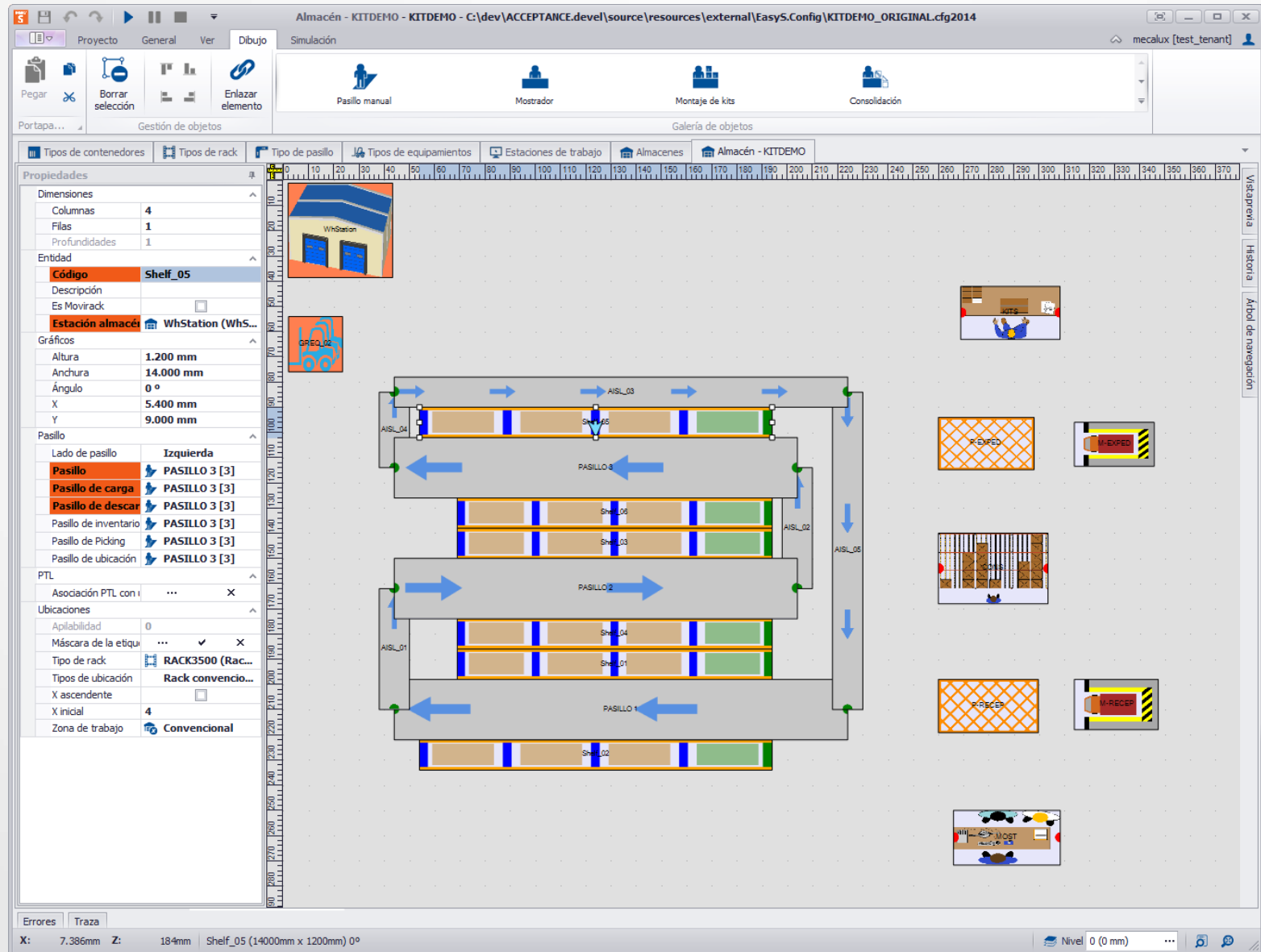
Visible	<input checked="" type="checkbox"/>
---------	-------------------------------------

Errors Trace

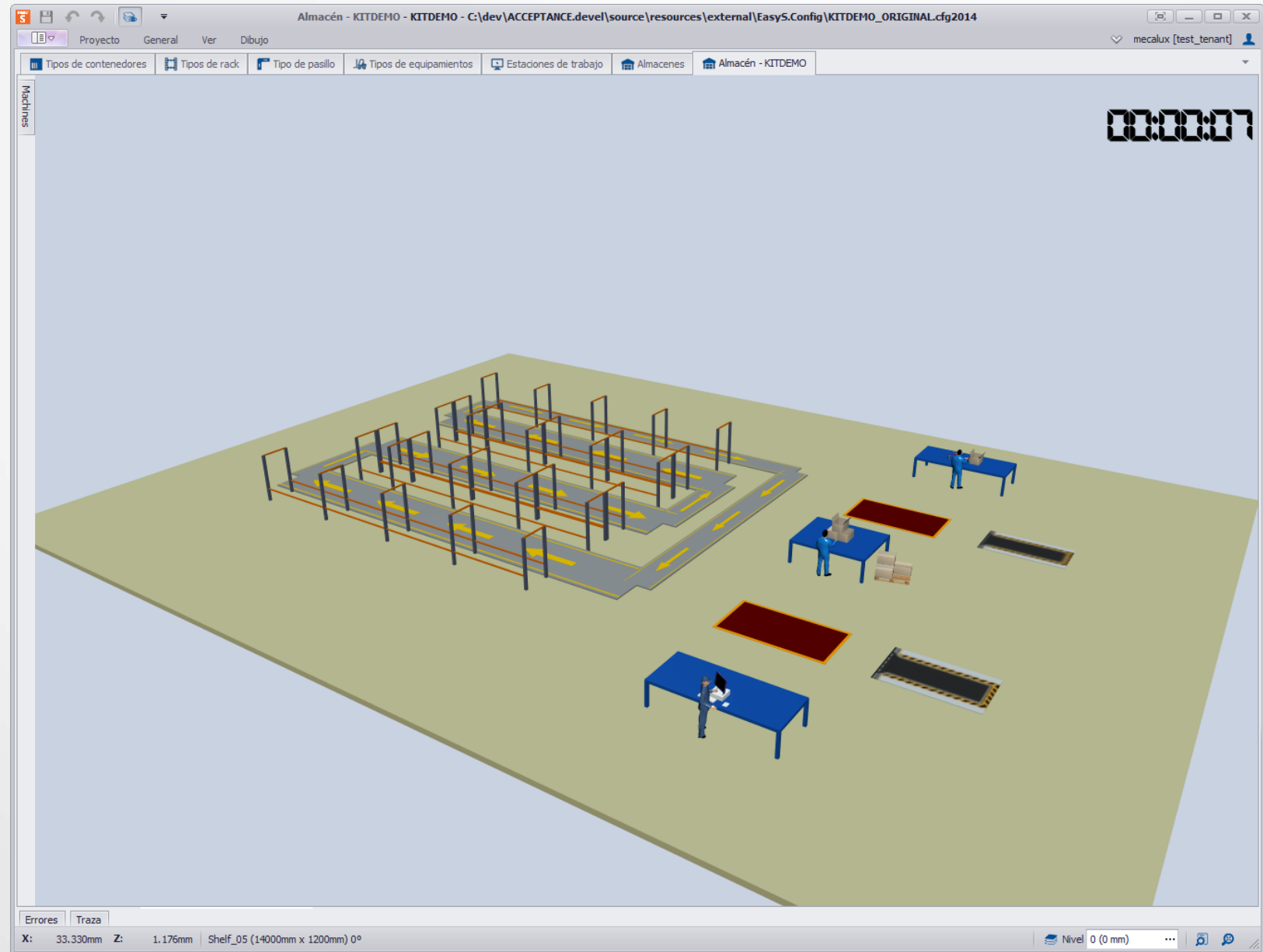
Online

<http://integracion40/AD/ApplicationDictionary.svc>

Flexibilidad y Adaptación al entorno: EasyS – Interfaz gráfica de configuración



Flexibilidad y Adaptación al entorno: EasyS – Simulación



Parte III

Calidad y Desarrollo

Control de versiones

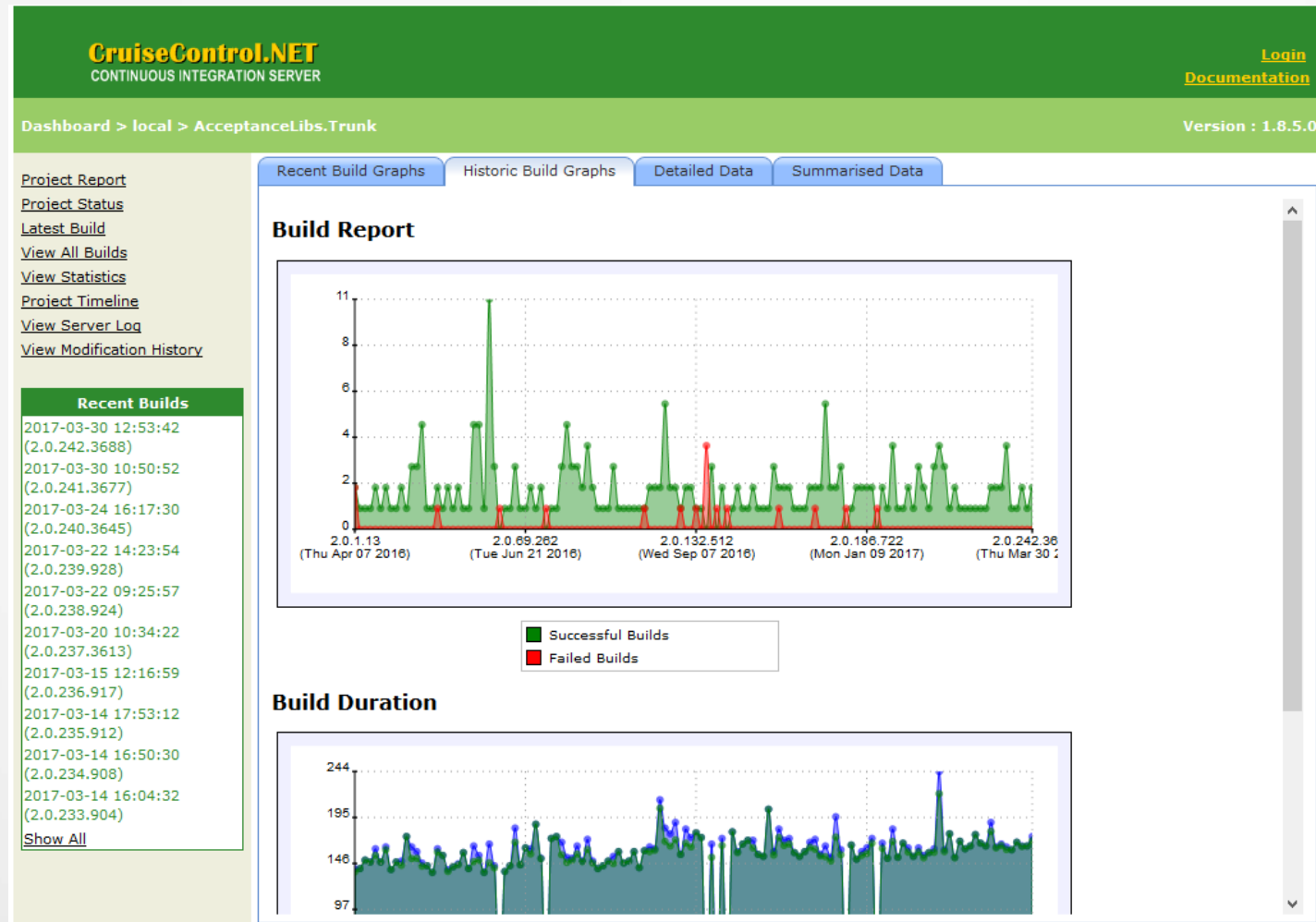
- SVN
- GIT
- Operativas en Ramas

Sistema de CI/CC

- Control continuo del estado del producto
- Ayudan a sincronizar el trabajo entre equipos
- Evitan DLL Hell

Test Automatizados

- Control de cambios / Test de regresión
- Permiten probar muchas casuísticas en poco tiempo
- Uso de tecnología de virtualización en contenedores



<https://jenkins.io/>
<http://cruisecontrolnet.org/>



Integración Continua

- Check-Out
- Build
- Test
- Rinse and Repeat
- **Prioridad:** Arreglar el build lo antes posible

CruiseControl.NET
CONTINUOUS INTEGRATION SERVER

[Login](#)
[Documentation](#)

Dashboard > local > Acceptance.870 > 2017-03-26 09:45:05 (1.0.30.934)
 Version : 1.8.5.0

[Latest](#)
[Next](#)
[Previous](#)

[Build Report](#)
[MSTest Report](#)
[MsBuild Report](#)
[Modifications](#)
[View Build Log](#)
[Specflow Test Report](#)
[RF revision tree](#)

Recent Builds

2017-03-26 09:45:05 (1.0.30.934)
 2017-03-19 09:45:05 (1.0.29.11102)
 2017-03-05 09:45:05 (1.0.28.3803)
 2017-02-19 09:45:05 (1.0.27.10627)
 2017-02-12 09:45:05 (1.0.26.864)
 2017-02-06 16:39:32 (1.0.25.853)
 2017-02-06 15:02:50 (Failed)
 2017-02-05 09:45:05 (Failed)
 2017-01-29 09:45:05 (1.0.24.10286)
 2017-01-22 09:45:05 (1.0.23.781)
[Show All](#)

solution Test Execution Report
 Generated by SpecFlow at 03/26/2017 12:46 (see <http://www.specflow.org/>).

[View this report in its own window](#)

Summary

Features	Success rate	Scenarios	Success	Failed	Pending	Ignored
3 features	100%	14	14	0	0	0

Feature Summary

Feature	Success rate	Scenarios	Success	Failed	Pending	Ignored
CheckViewsQueryPerformance	100%	2	2	0	0	0
Stress_02AccumulateSorders	100%	1	1	0	0	0
Stress_03AssignComplexity	100%	11	11	0	0	0

Feature Execution Details
Feature: CheckViewsQueryPerformance

Scenario	Category	Status	Time(s)
[show] Open all the web views and check the log for consuming queries	ViewQueries, SmartUIWeb, KitDemo.cfg2014, longRun, LogParser, DemoTest	success	259.482
[show] Check new predicates for using query files instead of entity names	ViewQueries, SmartUIWeb, KitDemo.cfg2014, longRun, LogParser, queryinfilesample, DemoTest	success	49.791

Feature: Stress_02AccumulateSorders

Scenario	Category	Status	Time(s)
[show] 02_stress, asignacion de stock, medicion en funcion de las ordenes liberadas con anterioridad variante 2	AcceptanceTest, test_asignación_completos_conPK.cfg2014, LongRun, AccumulateSordersVsStockAssignment, Manual, Stress, AssignStress05	success	577.201

Test Automáticos

- Criterios de aceptación
- SpecFlow (notación Gherkin)
- Test distribuidos (Pruebas reales)

Retos

- Comprobación de escalado
- Validación de virtualmente, infinidad de configuraciones
- Múltiples motores de BBDD soportados
- Riesgo de no focalizar

```
CountManualCount_01CheckLoc  X CheckViewsQueryPerformance.f  HelperStep.cs  MyFluentTest.cs  ContainersActionsAutomatic_Prc
@Automatic @AcceptanceTest @RF @Count @ManualCount @CheckLocationFrom @KitDemo-ManualCount.cfg2014
Feature: CountManualCount_01CheckLocationFrom
    Para poder verificar el funcionamiento del flujo de trabajo de recuento fisico
    como usuario
    quiero poder comprobar la ubicacion seleccionada como origen del recuento

    Background:
        Given el fichero de configuracion de EasyS AcceptanceTest\ManualCount\KitDemo-ManualCount.cfg2014
        Given la aplicacion uRFConsole ejecutada
        Given una sesion iniciada
        Given seleccionamos 1 Recuentos en el menu actual

    #####
    ##### UBICACIÓN ORIGEN VACÍA #####
    #####
    #Recuento/Recuento físico de ubicación -> Ubicación vacía confirmando que lo está
    Scenario Outline: 03_Recuento fisico de ubicacion, Ubicacion vacia con confirmacion
        Given seleccionamos 3 Recuento Físico en el submenu actual
        ## Fin preparación del escenario
        When escribimos <LocationCode> en el formulario Ubicacion.1 como la ubicacion sobre la que hacer el recuento
        Then el formulario ConfirmacionUbiVacía.1 muestra una pregunta con las palabras clave <LocationCode>,vacía a la que respondemos SI
        And finalizamos el proceso actual regresando al formulario de login
        Then la tabla Stocks NO contiene ninguno de los siguientes datos
            | LocationCode |
            | 00100100121 |

    Examples:
        | LocationCode |
        | 00100100121 |
```

<https://cucumber.io/docs/reference>

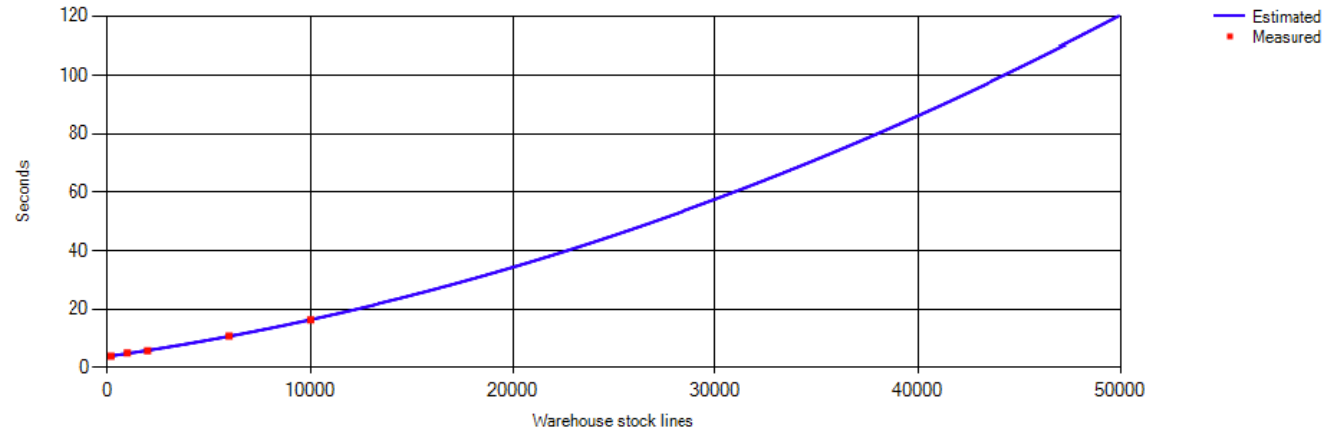
<http://specflow.org/>

Análisis de escalado

```

When los batch de escalado en el directorio Process\stress\comsign\complety\02stocklines\convencional\10000stocklines
-> done: Background_uGNSteps.GivenSceneLoadBatchFromFolder("Process\Stress\03...") (343,4s)
When finalizamos la orden de salida SORDER_100Lines
-> Time to FinalizeCreation Outbound Order command 'SORDER_100Lines': 501 ms.
Given la tabla OutboundOrders contiene los siguientes datos
--- table step argument ---
| Code | OutboundOrderStatus |
| SORDER_100Lines | Waiting |
-> done: Background_uGNSteps.DataIsFoundInEntity("OutboundOrders", <table>) (0,0s)
-> done: Process_OutboundOrdersSteps.finalizar("SORDER_100Lines") (0,5s)
When liberamos la orden de salida SORDER_100Lines generando estadísticas con la clave 10000
-> Time to Release Outbound Order command 'SORDER_100Lines': 16415 ms.
-> done: Process_OutboundOrdersSteps.liberarGuardandoResultados("SORDER_100Lines", "10000") (16,4s)
Then la tabla Tasks contiene los siguientes datos
--- table step argument ---
| ProductCode | TaskStatus |
| Art_001 | Pending |
| Art_100 | Pending |
-> done: Background_uGNSteps.DataIsFoundInEntity("Tasks", <table>) (0,1s)
Then guardamos las estadísticas en el fichero SORelease_100SOLines_Picking.csv
See the log
-> done: StatisticsSteps.GeneramosLasEstadisticasEnElFichero("SORelease_100SOLi...") (0,0s)
Then calculamos las funciones de regresion y generamos una estimacion con titulo Warehouse stock lines desde 100 hasta 50000 con incrementos de 100
Polynomial of order 2: 3.81665446872905 + 0.000992742628524679*X + 2.6758856981572E-08X^2 ; Coefficient of determination(R^2): 0.999691947016896
Polynomial of order 1: 3.52821345376753 + 0.00126442357974804*X ; Coefficient of determination(R^2): 0.997065625230544
-> Best determination coefficient: R2= 0.99969 (R1= 0.99707)

```



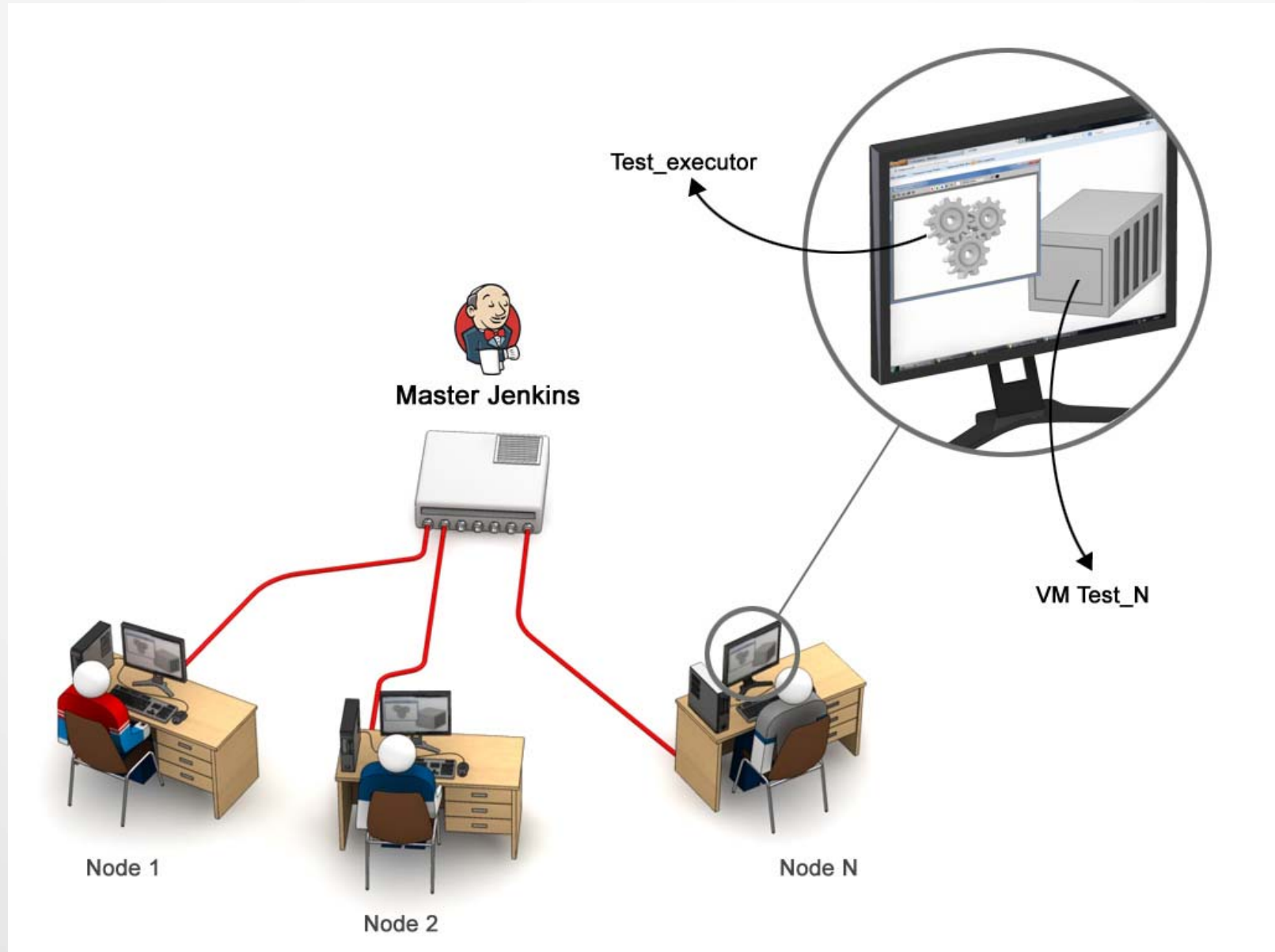
```

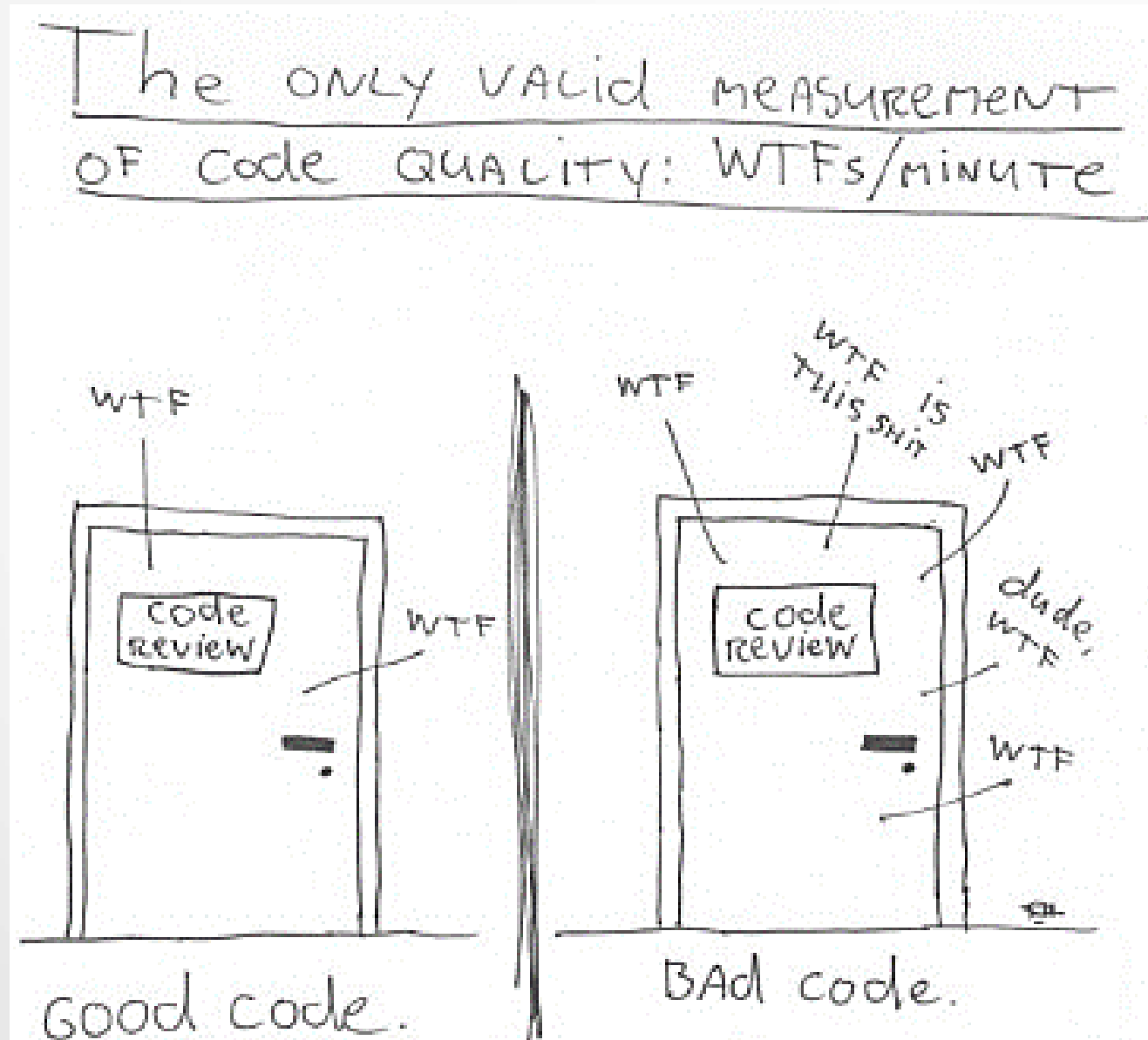
-> Quadratic Time estimated for 50000 = [00:02:00.3510000] sec.
-> Linear Time (best case) estimated for 50000 = [00:01:06.7490000] sec.
-> done: StatisticsSteps.ThenCalculamosLasFuncionesDeRegresion("Warehouse stock l...", 100, 50000, 100) (0,0s)

```

[\[show\]](#)02_stress, asignacion de stock, medicion en función de las lineas de stock existentes en el almacen: Shipping(Shipping)

Test distribuidos





(c) 2008 Focus Shift/OSNews/Thom Holwerda - <http://www.osnews.com/comics>

Issue tracker (redmine)

- Fundamental para trazar la historia de modificaciones
- Enlaza los test de aceptación con las historias de usuario
- Permite seguimiento de problemáticas de clientes
- <http://www.redmine.org/>

Parte IV

Evolución de la organización del desarrollo

Metodologías usadas

ICONIX

(<http://www.iconixsw.com/>)

Scrum

(<http://www.scrumguides.org/>)

TDD / BDD

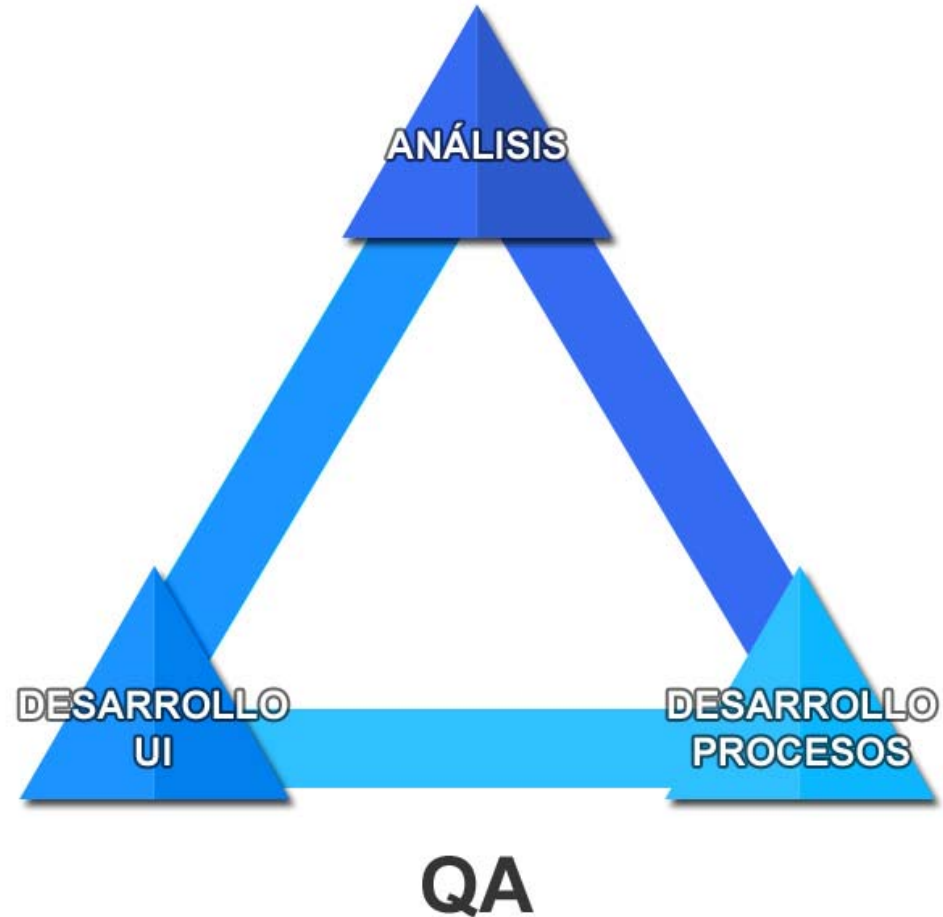
Kanban / JiT

Desarrollo en equipo único – monolítico - ICONIX



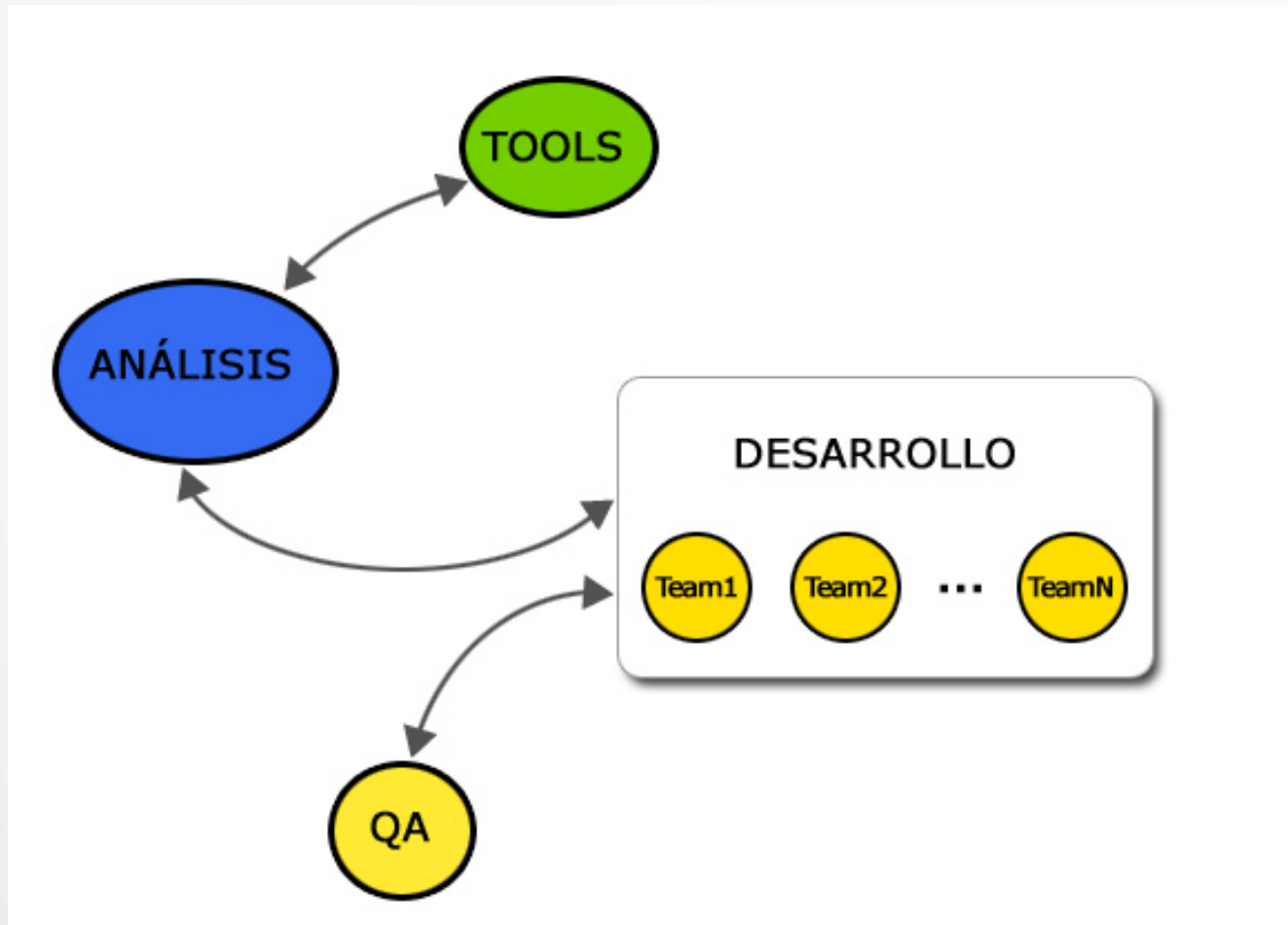
Pros - Cons

- Ciclos de desarrollo muy largos
- Tiempos de análisis demasiado largos para los requerimientos del mercado
- Fase de pruebas estresante y con muchos retrasos
- La documentación se quedaba obsoleta con facilidad



Pros - Cons

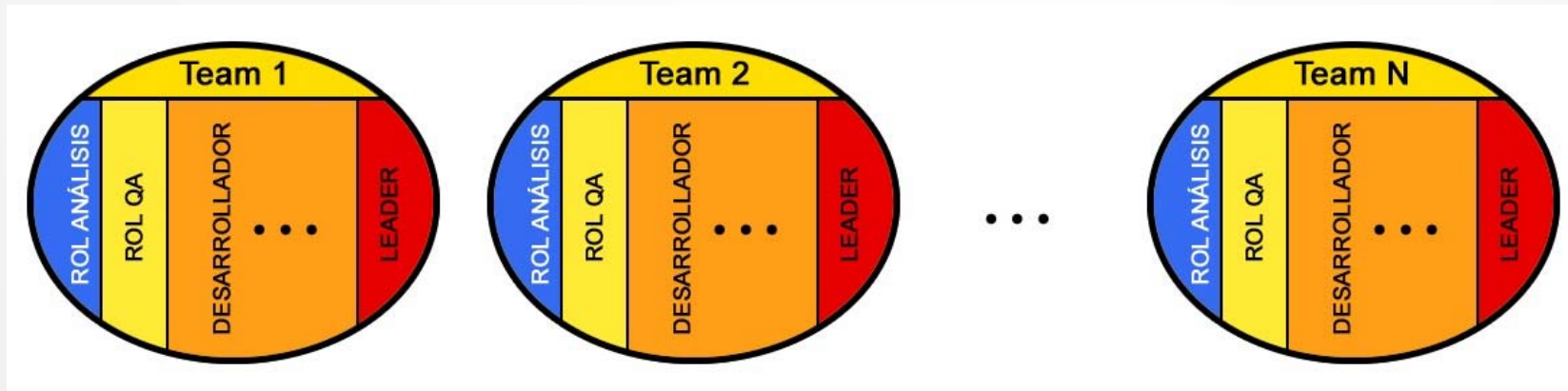
- Ciclos de desarrollo aun muy largos
- Pérdida del sentido de propiedad para los desarrollos
- Se crean islas de conocimiento muy específico
- Los primeros test automáticos son muy complejos y muy dependientes. El mantenimiento de los mismos es demasiado costoso



Pros - Cons

- Calidad empieza a llevarse a desarrollo, su papel cambia
- Los tiempos de iteración no son todo lo efectivos que quisiéramos
- Se consiguen ciclos de publicación más cortos

Desarrollo en equipos – Kanban / JiT



Pros - Cons

- Equipos autónomos, sentido de la propiedad del desarrollo
- Los equipos deciden los plazos
- Roles, no personas
- Se evitan los valles en roles muy específicos, el equipo decide si pueden apoyar otras partes del desarrollo



¡Fin!

¡Gracias por asistir!