



Universidad Cenfotec

Maestría en Tecnología de Bases de Datos
Documento final de Proyecto de Investigación Aplicada 2

Tema:

“INTEGRACIÓN DE SCRUM E ITIL, USANDO INTELIGENCIA DE
NEGOCIOS PARA EL CONTROL DE INDICADORES CLAVE DE
DESEMPEÑO”

Estudiante:

PLÁ ROJAS, JOSÉ AGUSTÍN

San José, Costa Rica

JUNIO 2018

Declaratoria de derechos de autor

Yo, José Agustín Plá Rojas, estudiante de la Universidad Cenfotec, de la carrera Maestría en Tecnologías de Bases de Datos, declaro bajo juramento y consciente de las responsabilidades penales de este acto, que soy el autor intelectual de esta tesis titulada “Integración de Scrum e ITIL, usando inteligencia de negocios para el control de indicadores clave de desempeño”. Autorizo a la Universidad Cenfotec, a sus estudiantes activos y cualquier ente gubernamental que utilice el presente documento para ser consultado única y exclusivamente con fines académicos.

José Agustín Plá Rojas

Cédula: 303730264

Dedicatoria

A Dios, gracias nuevamente por darme perseverancia.

A mi esposa, por el tiempo que no le di y todo lo que me escuchó.

A Don Ignacio, quien logró orientarme para ponerle orden a mis ideas.

Tribunal examinador

Este trabajo fue aprobado por el Tribunal Examinador de la Universidad Cenfotec como requisito para optar por el grado de Master en Tecnologías de Bases de Datos.

Ignacio Trejos Zelaya

Tutor

Elmer Enrique Fonseca Umaña

Lector

Luis Rolando Gonzalez Sandoval

Lector

Aprobación de filóloga

San José, 03 de mayo de 2018.

Señores Cenfotec

Estimados señores,

Por medio de la presente declaro que leí y corregí el Proyecto Final de Graduación del señor José Agustín Plá denominado *"INTEGRACIÓN DE SCRUM E ITIL, USANDO INTELIGENCIA DE NEGOCIOS PARA EL CONTROL DE INDICADORES CLAVE DE DESEMPEÑO"*.

Revisé el trabajo en aspectos, tales como: construcción de párrafos, ortografía, puntuación y otros relacionados con el campo filológico como formato de citas, índice y bibliografía. Desde ese punto de vista considero que está listo para ser presentado como Proyecto Final de Graduación; en este sentido, el documento cumple con los requerimientos de presentación, pero la edición final del texto, que incluirá o excluirá las correcciones filológicas, queda bajo la completa responsabilidad del autor.

Suscribe cordialmente,

Mra. Fernanda Masís D.

María Fernanda Masís Díaz
304320913
Carnet ACFIL 094
Filóloga

Tabla de contenido

Declaratoria de derechos de autor	ii
Dedicatoria	iii
Tribunal examinador.....	iv
Aprobación de filóloga.....	v
Tabla de contenido	vi
Índice de tablas	xi
Índice de imágenes	xi
Resumen ejecutivo.....	14
Capítulo 1. Introducción.....	14
1.1 Generalidades	14
1.2 Antecedentes del problema	15
1.3 Definición y descripción del problema	17
1.4 Justificación	18
1.5 Viabilidad.....	19
1.5.1 Punto de vista técnico.....	20
1.5.2 Punto de vista operativo	20
1.5.3 Punto de vista económico.....	20
1.6 Objetivos.....	20
1.6.1 Objetivo general	21
1.6.2 Objetivos específicos.....	21
1.7 Alcances y limitaciones.....	21
1.7.1 Alcances	21
1.7.2 Limitaciones.....	22
1.8 Estado de la cuestión	22
1.8.1 Resumen de ‘Tres expertos en gerencia de calidad’	25

1.8.2	Resumen de 'Guía de Scrum'	26
1.8.3	Resumen de 'El nuevo XP'	28
1.8.4	Resumen de 'Guía de ITIL para DevOps'	32
1.8.5	Resumen de 'Medición de ROI en transformaciones ágiles del negocio'	
	34	
Capítulo 2.	Marco teórico.....	36
2.1	Descripción teórica de BI.....	36
2.1.1	BI tipo DW 1.0	37
2.1.2	BI tipo DW 2.0	38
2.2	Metodologías para el negocio.....	38
2.3	Metodología ITIL.....	39
Capítulo 3.	Marco metodológico	42
3.1	Tipo de investigación	42
3.2	Alcance investigativo	42
3.3	Enfoque	42
3.4	Diseño	42
3.5	Población y muestreo	43
3.6	Instrumentos de recolección de datos	43
3.7	Técnicas de análisis de la información	44
3.8	Descripción de Scrum en SGI	44
3.9	Estrategia de desarrollo de la propuesta	48
Capítulo 4.	Implementación de ITIL.....	49
4.1	SIPOC	49
4.2	Necesidad de formalizar el proceso entre departamentos.....	50
4.3	Acuerdo a nivel operacional	51
4.3.1	Diagrama de flujo del proceso	52

4.3.2	Roles y responsabilidades.....	62
4.3.3	Métricas del OLA	63
Capítulo 5.	Implementación de inteligencia de negocios	64
5.1	Acceso a fuente de datos	65
5.2	Etapa Interactiva.....	65
5.3	Etapa Integrada	66
5.4	Etapa Archivada	68
Capítulo 6.	Implementación de reportes	70
6.1	Reporte Métricas OLA	70
6.1.1	Métrica OLA CR's completados.....	70
6.1.2	Métrica OLA CR's cerrados sin cambios	71
6.1.3	Reporte seguimiento de CR's abiertos	72
6.1.4	Reporte de calidad de CR's entregados.....	73
6.2	Reporte fuera del OLA.....	74
6.2.1	Reporte de CR's pendientes de Sprints previos	75
Capítulo 7.	Análisis de resultados.....	78
Capítulo 8.	Conclusiones y recomendaciones	80
8.1	Conclusiones	80
8.1	Recomendaciones	81
Capítulo 9.	Reflexiones finales	82
Bibliografía	84
Apéndices.....	91
Ap 1.	Diagrama de Kimball DW 1.0.....	91
Ap 2.	Diagrama de Inmon DW 1.0.....	92
Ap 3.	Diagrama de Inmon DW 2.0.....	93
Ap 4.	Diagrama completo SAFe 4.5.....	94
Ap 5.	Diagrama de flujo OLA.....	95

Ap 6.	Función SQL para pasar a columna el texto dividido por coma.....	96
Ap 7.	Vista SQL de CR's del Sprint que acaba de pasar	97
Ap 8.	Vista SQL de CR's de previos Sprints que no se hayan cerrado	98
Ap 9.	Procedimiento SQL para crear dimensión de fechas.....	99
Ap 10.	Acceso a fuente de datos	100
a.	Ingresar a QUIX.....	100
b.	Acceso a base de datos de Oracle	100
c.	Oracle driver para Windows	101
d.	Probar la conexión con SQL*Plus.....	103
e.	Vincular servidor Oracle a SQL Server.....	105
Ap 11.	Etapla Interactiva.....	110
a.	Creación de base de datos QUIX_SA	110
b.	Creación de tablas.....	111
c.	Actualización de la tabla BO_QX_CR.....	112
i.	Procedimiento almacenado para insertar	112
ii.	Procedimiento almacenado para actualizar.....	115
iii.	Procedimiento almacenado para borrar.....	117
iv.	Ejecución automática de procedimientos de sincronización.....	118
Ap 12.	Etapla Integrada	120
a.	Creación del ETL de la base de datos de multidimensional SPRINT_DW	120
b.	Agregar los Sprints actuales a la dimensión DimSprint	120
c.	Hacer los llenados de las demás dimensiones.....	122
d.	Carga de la tabla de hechos.....	124
Ap 13.	Etapla Archivada	125
a.	Ejecución automática del ETL de SPRINT_DW	125

b. Creación de ETL's de la base de datos multidimensional SPRINT_ARC.....	126
c. SPRINT_ARC en SSAS	128
d. Procesamiento del cubo SPRINT_ARC.....	130
Ap 14. Consulta SQL en Tableau para CR's completados	133
Ap 15. Consulta seguimiento de CR's abiertos.....	134
Glosario.....	135
Acrónimos	139

Índice de tablas

Tabla 1 - Marzano y Kendall, PDCA Deming, Estructura Proyecto	21
Tabla 2 - Fases de implementación.....	48
Tabla 3 - Plan de acción para implementar BI.....	64
Tabla 4 - Dimensiones etapa integrada	67
Tabla 5 - Beneficios por implementación.....	78
Tabla 6 - Lista de acrónimos	139

Índice de imágenes

Ilustración 1 - Ejemplo Línea de producción de servidores (Kelleher, 2012) ...	15
Ilustración 2 - Ejecución de pruebas de calidad en servidores (Vanian, 2015)	16
Ilustración 3 - Ejemplo de diagrama de partes de un servidor (Integrity Global, 2016)	16
Ilustración 4 - Proceso de manufactura	19
Ilustración 5 - DevOps Software de Automatización.....	32
Ilustración 6 - Software de Pruebas Funcionales	32
Ilustración 7 - Software de Integración Continua	32
Ilustración 8 - Ciclo de ITIL (ProactivaNET, 2017).....	40
Ilustración 9 - Gráfico de Burndown.....	44
Ilustración 10 - Scrum en SGI.....	46
Ilustración 11 - SIPOC de SGI.....	50
Ilustración 12 - Tablas SPRINT_DW	67
Ilustración 13 - Cubo SPRINT_ARC.....	69
Ilustración 14 - Tabeau conexión a QUIX_SA	70
Ilustración 15 - Tableau hoja de trabajo CR's completados.....	71
Ilustración 16 - Tableau Server CR's CNC	72
Ilustración 17 - Tableau Server CR's abiertos	73
Ilustración 18 - Medida de calidad de CRs en Tableau	73
Ilustración 19 - Reporte de Calidad de CR's en Tableau Server	74
Ilustración 20 - Conexión de Tableau Server a cubo SPRINT ARC	75

Ilustración 21 - Cubo SPRINT ARC en Tableau Desktop	76
Ilustración 22 - CR's abiertos por encargado de Sprints pasados	77
Ilustración 23 - Detalle de datos de gráfico de cubo en Tableau	77
Ilustración 24 - Diagrama de Kimball DW 1.0	91
Ilustración 25 - Diagrama de Inmon DW 1.0	92
Ilustración 26 - Diagrama de Inmon DW 2.0 (Inmon B. , 2010, pág. 2)	93
Ilustración 27 - Esquema SAFe (Scaled Agile Framework, 2017)	94
Ilustración 28 - Ventana de QUIX usuarios SQL	100
Ilustración 29 - Solicitud acceso a QUIX DB.....	100
Ilustración 30 - Búsqueda de ODAC en Bing.....	101
Ilustración 31 - Licenciamiento ODAC	101
Ilustración 32 - ODAC a descargar con SQL*Plus.....	102
Ilustración 33 - ODAC descomprimido.....	102
Ilustración 34 - Ignorar advertencias ODAC	103
Ilustración 35 - SSMS nuevo servidor vinculado	106
Ilustración 36 - Crear servidor vinculado	107
Ilustración 37 - Seguridad servido vinculado	107
Ilustración 38 - Tablas en servido vinculado	108
Ilustración 39 - Prueba conexión servidor vinculado.....	108
Ilustración 40 - Reconstrucción índice de almacenamiento de columnas.....	114
Ilustración 41 - Paso SyncDelQUIX	118
Ilustración 42- Pasos de Job QUIX_SA-SyncingWithLinkedServer	119
Ilustración 43 - Agregar Sprint actual a DimSprint	120
Ilustración 44 - Proceso de Agregar Sprint	121
Ilustración 45 - Conditional Split Sprint actual	122
Ilustración 46 - Llenar otras dimensiones	122
Ilustración 47 - Proceso de llenar demás dimensiones.....	123
Ilustración 48 - Proceso de llenado de dimensión Usuarios	124
Ilustración 49 - Llenar tabla de hechos de Sprint.....	124
Ilustración 50 - Proceso llenado tabla de hechos del Sprint	124
Ilustración 51 - Job SPRINT_DW-DailySprintSnapshot.....	125

Ilustración 52 - Conditional Split de etapa archivada.....	126
Ilustración 53 - Job SPRINT_ARC-SprintEndDay21.....	127
Ilustración 54 - Declaración de SQL para actualizar CRs no cerrados.....	127
Ilustración 55 - Proyecto SPRINT_ARC en Visual Studio 2013.....	128
Ilustración 56 - Vista de fuente de datos SPRINT_ARC.....	129
Ilustración 57 - Tabla de hechos como dimensión.....	130
Ilustración 58 - Dimensión de fechas.....	130
Ilustración 59 - Cubo SPRINT_ARC cache MOLAP.....	131
Ilustración 60 - SSAS tarea de procesamiento en SSIS.....	131
Ilustración 61 - Propiedades tarea de procesamiento del cubo.....	132
Ilustración 62 - Ejecución del JOB eliminar duplicados y actualizar SPRINT_ARC.....	132
Ilustración 63 - Consulta SQL en Tableau para CR's completados.....	133

Resumen ejecutivo

El enfoque de este proyecto es demostrar cómo una empresa transnacional y de manufactura, cuyos procesos internos son complejos, es capaz de alcanzar los niveles buscados de calidad tanto en productos como en servicios, gracias a una implementación armonizada de ITIL (para la gestión de servicios) junto con Scrum (para el desarrollo ágil de software) que incorpora inteligencia de negocios para el control de las métricas que exigen ambas metodologías.

Capítulo 1. Introducción

1.1 Generalidades

El foco de una empresa de manufactura es la producción y para ello viene de la mano un ideal de calidad, es decir, que de la línea de manufactura salgan productos que cumplan con las necesidades de calidad requeridas por el cliente. Los tiempos de entrega también influyen en la calidad, ya no como producto, sino como servicio; y por lo tanto, entre más complejos los procesos de manufactura mayores serán los tiempos de entrega.

ITIL es una metodología que establece muchos procesos que ayudan con el alcance del concepto de calidad a un nivel de servicio; para ello utiliza una amplia cantidad de procesos, flujos de trabajo, definición de roles y métricas, todo lo anterior contribuye con la implementación de dicha metodología y lograr una calidad en el servicio. Por otro lado, las metodologías ágiles como el caso de Scrum buscan una relación más directa con el cliente y de esta forma brindar resultados más rápidos y a la vez más precisos en el cumplimiento de los requerimientos.

Una sola metodología no es suficiente para lograr calidad de servicio en los procesos internos, calidad del producto que se despachará al cliente y un tiempo aceptable en la resolución de órdenes de compra. Por ello es necesario buscar una implementación variada que ayude a fortalecer las debilidades encontradas en cada metodología y en cada nivel del negocio. Las metodologías ágiles como Scrum ayudan mucho a un bajo nivel directo en el trabajo del día a día, mientras que ITIL trae beneficios a un nivel más intermedio entre gerencias de varios departamentos.

La empresa donde se aplicó este proyecto y cuyo nombre no se mencionará por asuntos legales, es una reconocida corporación multinacional manufacturera de equipo de cómputo empresarial, con más de cincuenta mil empleados, y que cumple con el estándar de calidad en manufactura ISO9001; además posee varias fábricas a nivel global.



Ilustración 1 - Ejemplo Línea de producción de servidores (Kelleher, 2012)

1.2 Antecedentes del problema

El fin de la manufactura de un producto es que cumpla con las expectativas de calidad del cliente. Para alcanzar este objetivo, cada línea de producción contiene etapas de pruebas de calidad, y en casos de que el producto falle dicha prueba, tendrá que volver a repetir la etapa de manufactura que no pasó la prueba hasta corregir el fallo.

Cuando los clientes están distribuidos alrededor de todo el mundo, los productos se manufacturan lo más cerca de los diferentes nichos de mercado que existan. Pero a su vez donde la fabricación sea de menor costo, de nuevo la calidad tiene que ser la misma indiferente de cuál haya sido la fábrica de donde salió el producto final.

No solo se debe considerar la calidad en la línea de producción sino también la de inventario de proveedores, almacenaje y logística de entrega, entre otras; esto exige

una necesidad de comunicación y entendimiento de procesos entre cada departamento, para así lograr una pronta entrega al cliente final. En un aspecto más amplio el cliente de la compañía es quien pagó por el producto, pero entre departamentos y considerando el flujo de trabajo de los procesos puede que un departamento sea cliente de otro.



Ilustración 2 - Ejecución de pruebas de calidad en servidores (Vanian, 2015)

Al tratarse de equipo de cómputo empresarial, hay múltiples proveedores de discos duros, tarjetas de red, procesadores, memorias, entre otras muchas partes y es necesario garantizar que cada solución que se esté vendiendo al cliente trabaje en conjunto y sin errores.

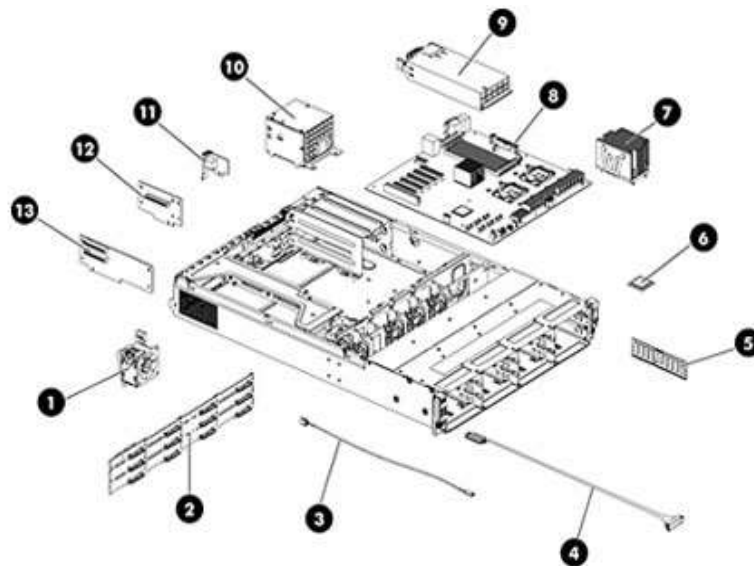


Ilustración 3 - Ejemplo de diagrama de partes de un servidor (Integrity Global, 2016)

Cada vez que se lanza un nuevo servidor se registra como una Nueva Introducción de Producto (*NPI*, por sus siglas en inglés), a tal NPI hay que generarle un combo de pruebas que garanticen su buen ensamblaje y funcionamiento al salir de la línea de producción. Pero después de que un equipo catalogado como NPI sale al mercado, siguen generándose cambios en las partes que lo conforman por lo que las pruebas también cambian.

El equipo encargado de desarrollar las pruebas de calidad que se aplican en cada una de las más de una docena de fábricas a nivel global, se llama Servicios Globales de Ingeniería (SGI). A pesar de que las pruebas que desarrolla son las que se aplican en cada una de las fábricas, las solicitudes vienen de otros departamentos que son los ingenieros responsables de cada producto (PE's por sus siglas en inglés), y están al tanto de cada una de las órdenes de compra y los cambios que estas requieren en dichas pruebas.

1.3 Definición y descripción del problema

Anteriormente, el departamento de SGI trabajaba en modo estación de bomberos, este símil se refiere a que su día a día se basaba en apagar incendios o mejor dicho incidentes que ocurrían en alguna de las fábricas; el concepto de incidente es algo que esté deteniendo una orden en la línea de producción de fábrica. Un alto ejecutivo quien posteriormente tomó el mando sobre SGI, vio la necesidad de cambiar la forma de trabajo y buscó implementar la metodología SAFe de la cual se habla más adelante en la sección *Metodologías para el negocio*. Pero su adopción no fue tan fácil e internamente lo que se logró fue la primera fase de SAFe, referente a metodologías ágiles y en este caso dándole un mayor énfasis a Scrum, de la cual se puede encontrar más información en la sección de Estado de la Cuestión *Resumen de 'Guía de Scrum'*.

La implementación de Scrum trajo en su momento una mejor definición de roles y responsabilidades dentro del equipo, así como una más eficiente asignación de trabajo a los recursos disponibles. Aun así, esta metodología carece de la creación de documentación que formalice el proceso de negocio; sin dicho proceso que exija los requerimientos de cada *User Story* o solicitud de cambio para este caso, se empezó a notar un conflicto con los clientes de SGI que son los ingenieros responsables de cada producto y quienes generaban solicitudes cuando éstas todavía no podían ejecutarse o

escalaban como incidente cualquier solicitud para buscar priorizarla. Prácticamente hay un Ingeniero de Producto por cada modelo de servidor que produce la compañía, por lo que personalmente, para cada uno de ellos sus solicitudes son las prioritarias; pero como SGI recibe las solicitudes de todos los ingenieros de producto a nivel global, el departamento tiene que establecer una nueva prioridad según el impacto para la compañía.

Los clientes que asignan solicitudes de cambio o CR's por sus siglas en inglés al departamento de SGI, no estaban al tanto del proceso interno de SGI y las razones de porqué funciona de esta manera. La falta de entendimiento del proceso interno de SGI generó conflictos entre departamentos y es aquí donde se ve la necesidad de cubrir con ITIL lo que le falta a Scrum para crear acuerdos entre departamentos a nivel de gerencia y esclarecer las necesidades del solicitante como proveedor del servicio, buscando ahora una calidad en el servicio y no solo en el producto final.

1.4 Justificación

La implementación de metodologías ágiles de manera interna en el departamento de SGI, así como el compromiso de servicio con la organización padre, que en este caso es la Cadena de Suministro (Supply Chain en inglés); requiere de informes constantes, algunos enfocados en la cola de solicitudes pendientes o Product Backlog según Scrum, otros midiendo el avance de cada día o Daily Scrum como su nombre lo indica también según Scrum y los reportes mensuales para comparar el nivel de servicio o KPI's según ITIL.

Los Project Managers, Change Managers, Product Owners, entre otros roles definidos por ITIL y Scrum, son personal muy valioso y con una alta carga de trabajo como para que una parte importante de su día a día se gaste creando gráficos para alta gerencia.

Además, se busca una fuente única de datos que ofrezca un historial de cada reporte y no tener que ir a buscar historial de correos para comparar y ver si ha habido una mejora en el servicio.

Por último, las metodologías de SAFe y Scrum no han sido una solución completa como tampoco de la comprensión de todos los involucrados.

Fallos en la calidad del desarrollo de las pruebas de calidad, se transfieren en fallos en la calidad del producto en la línea de producción. Para comprender de manera visual cuál es la posición de *calidad* en el proceso de manufactura se presenta el siguiente gráfico.

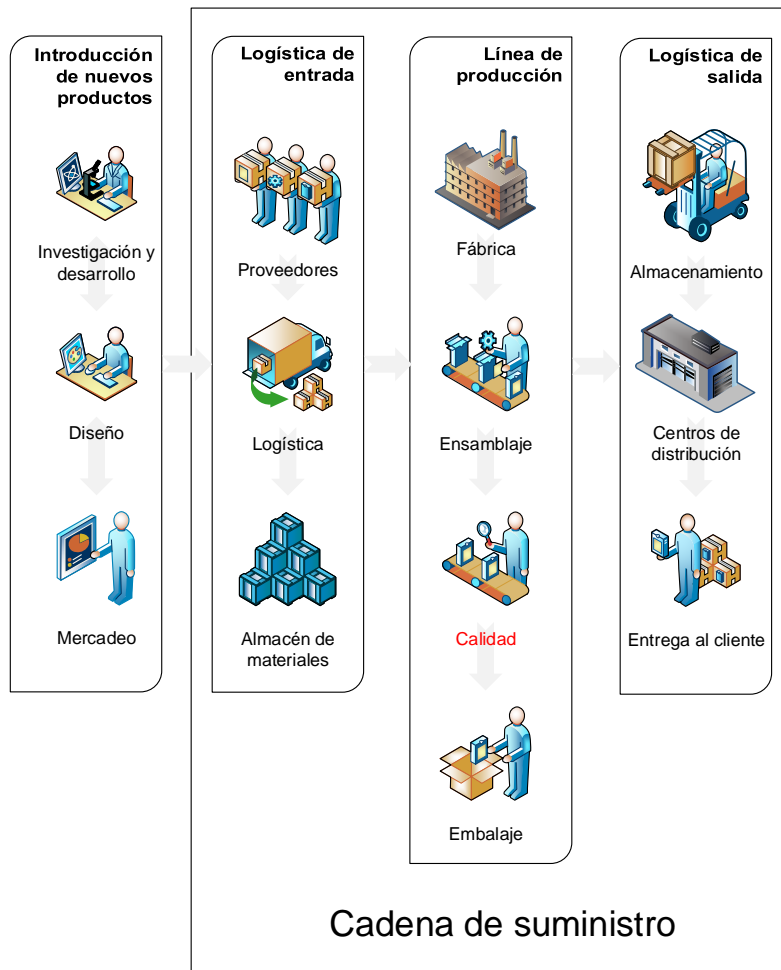


Ilustración 4 - Proceso de manufactura

1.5 Viabilidad

Las herramientas con las que se cuenta para llevar a cabo este proyecto, están atadas en forma circunstancial de manera técnica, operativa y económica. Consecuente a tener las herramientas para su ejecución, la viabilidad también va atada a los beneficios ya anteriormente descritos que brinde la ejecución del proyecto.

1.5.1 Punto de vista técnico

Las herramientas con las que se cuenta para este proyecto son:

- Cuenta de lectura a la base de datos que maneja el sistema de Administración del Negocio llamado QUIX.
- Acceso a un ESXi host donde se pueden crear máquinas virtuales en VMware.
- Software de instalación de Windows Server 2012, SQL Server 2014 Enterprise y Tableau 10.
- Acceso a un sitio en Tableau Server preparado para las necesidades de la Cadena de Suministro.

1.5.2 Punto de vista operativo

Los reportes son una necesidad constante de Gerencia y dependiendo el nivel de gerencia así cambia el tipo de reporte.

El recurso humano disponible es de una sola persona y tal persona no estará dedicada totalmente a este proyecto. Para el negocio el foco es que las fábricas corran y mantener una calidad en los productos; los reportes a pesar de ser necesarios según las metodologías, no superan lo que es el núcleo del negocio.

1.5.3 Punto de vista económico

El equipo a utilizar es hardware disponible que de no ser usado sería marcado para reciclaje.

Las licencias son a nivel empresarial por lo que no afectan el presupuesto de la organización de Cadena de Suministro, ni de su equipo de desarrollo de pruebas de calidad.

El personal que colaborará con este proyecto lo hará como segunda o tercera prioridad en su lista de tareas, sin dejar de lado lo que afecte directamente la calidad en cada lanzamiento de pruebas.

1.6 Objetivos

La taxonomía seleccionada para enunciar los objetivos de este proyecto es la de Marzano y Kendall 2006, debido a un enfoque de resolución de problemas en cuatro fases, Conocimiento, Comprensión, Análisis y Utilización. Cada etapa lleva subniveles que ayudan a esclarecer los alcances del proyecto.

Las cuatro fases sugeridas en el modelo de Marzano y Kendall, se asemejan al modelo de Deming (J. Gerald, 1992, pág. 7) de calidad - usado en ITIL.

Tabla 1 - Marzano y Kendall, PDCA Deming, Estructura Proyecto

Marzano y Kendall	Ed. Deming	Documento
Conocimiento	Actuar	Introducción y marco teórico
Comprensión	Planificar	Marco metodológico
Análisis	Hacer	Implementación y análisis de resultados
Utilización	Revisar	Conclusión y recomendaciones

1.6.1 Objetivo general

Implementar modificaciones en los servicios relacionados con la cadena de suministro, mediante acuerdos operacionales e inteligencia de negocios, a fin de lograr la coordinación entre procesos de negocios y la mejora continua basada en datos.

1.6.2 Objetivos específicos

1. Establecer un acuerdo de nivel operacional entre el equipo de PE's y el de SGI, satisfaciendo los requerimientos de servicios y de procesos, con salvaguarda de los métodos internos de cada departamento.
2. Integrar la herramienta existente de gestión de procesos con un nuevo sistema de inteligencia de negocios.
3. Examinar el funcionamiento del acuerdo de nivel operacional mediante métricas, con el fin de auditar la calidad del servicio y fundamentar su mejora.

1.7 Alcances y limitaciones

A continuación, se detalla lo que se pretende con este proyecto, así como sus limitaciones. De buscar mayores alcances requerirá del planteamiento de otro proyecto que dé continuidad a este.

1.7.1 Alcances

La orientación de este documento se divide en dos etapas principales, la primera es realizar un acuerdo a nivel operacional entre departamentos, OLA por sus siglas en inglés y descrito en la segunda fase de ITIL, que es la etapa de diseño del servicio, y que garantice continuidad del proceso de Scrum y metodologías ágiles adoptadas por SGI;

la segunda es implementar una solución de BI que brinde datos y métricas que promuevan la continuidad y mejora del servicio pactado en el OLA.

1.7.2 Limitaciones

Los reportes para el proyecto inicial serán los definidos en el OLA.

La implementación de BI, brindará muchas mayores posibilidades de estudio y un análisis más profundo, pero eso conlleva un trabajo continuo en conjunto con las necesidades del negocio, por lo que no permitiría dar un fin al proyecto.

La empresa al ser una corporación pública que cotiza en la bolsa, además de factores como múltiples competidores en el mercado, tiene restricciones en cuanto a la publicación de datos y procesos internos que puedan inferir en el mercado de bolsa o ser usados por otros competidores para copiar estrategias de negocio.

Los datos usados en este documento no deben publicar información interna restringida como sean errores en manufactura o el proceso de prueba de productos en la línea de manufactura.

1.8 Estado de la cuestión

La descripción del problema, sus antecedentes, historia y tipo del negocio, trae una mezcla de diversas metodologías bastante populares en el mundo informático y otras en el campo de producción industrial. En la historia de la industria a través del tiempo se han dado muchos cambios; Henry Ford de la Ford Co. y Alfred Sloan de General Motors que trajeron el concepto de producción en masa con la línea de ensamblaje en movimiento, posterior a su logro vinieron Eiji Toyoda y Taiichi Ohno de Toyota con un proceso de manufactura ágil (Womak, Jones, & Roos, 2007, págs. 24-28, 141-143, 47-66), pero en la actualidad también hay que unir la parte de manufactura con la de desarrollo de software y es en esta sección que se procura explicar la teoría de los métodos de mejora que se han implementado o se vayan a usar durante la ejecución de este proyecto.

Las metodologías se diferencian de los estándares en que las primeras son maleables, son una guía de consejos a seguir de cómo pueden hacerse las cosas; en cambio un estándar exige cómo deben hacerse.

Encontrar documentos académicos que describan la unión entre Cadena de Suministro, SAFe, Scrum, LEAN, Six Sigma e ITIL, adaptándose a los sistemas utilizados de forma interna no es posible hacerlo fuera de la misma compañía.

El propósito de la Inteligencia de Negocios es justamente generar beneficios de manera inteligente al negocio; por ello innovar en herramientas famosas en el mercado como R, librería Pandas en Python, MapReduce, o tantas otras sin tener una necesidad real para los objetivos del negocio es un claro egocentrismo del lado de ingeniería que se enfoca en probar nuevas tecnologías en vez de ir directo a solucionar los problemas del negocio.

Al tratarse de una empresa manufacturera, esta debe obedecer a estándares internacionales de calidad como el ISO 9001, que define calidad como *“el grado al cual un conjunto de características heredadas cumplen los requerimientos”*, mientras que para los tres gurús de gerencia de la calidad, esta varía, para Crosby la calidad debe cumplir los requerimientos, para Deming la calidad debe ser sostenible y para Juran es una mezcla de cero defectos mientras cumpla las características (J. Gerald, 1992, págs. 3-4). Con base en estas cuatro definiciones es que se justifica la necesidad de un departamento que vele por la calidad de los servidores que salen de cada fábrica, sin varianza entre ellas.

Los cambios en tecnología son muy veloces y a ese mismo ritmo es que tienen que cambiar las pruebas de calidad, además, la empresa no produce todo el hardware que va dentro de sus servidores, este depende de terceros, semejante a la manufactura de vehículos, por ello hay variables externas a tomar en cuenta. La ausencia de pruebas de calidad termina parando la fábrica, afectando el inventario o perjudicando la satisfacción del cliente.

La rapidez con la que se necesitan los cambios, es lo que ha llevado a implementar metodologías ágiles dentro del departamento de calidad. Estas son SAFe y Scrum, así como con este proyecto el uso de ITIL para implementar una comunicación con base en acuerdos con otros departamentos como el de cadena de suministro y línea de producción.

Al buscar sobre BI y Scrum en conjunto, lo que se encuentra son diferentes intentos de aplicar proyectos de BI de manera ágil. Pero eso no es lo que busca este

proyecto, donde el fin es aplicar BI para el análisis de un grupo de desarrollo que utiliza Scrum en un proceso interno en la línea de producción, que se enfoca en manufactura LEAN siguiendo el método TOYOTA que es la base de Six Sigma, y además a nivel de gerencia un impulso en el uso de ITIL para brindar sincronía entre departamentos. Es por ello que las siguientes referencias solo brindan un aporte de manera individual en su área y será parte de este proyecto el explicar su integración, a continuación, siguiendo el formato tomado de un estudio de la Universidad de Castilla-La Mancha del Departamento de Tecnologías y Sistemas de Información, se detalla referencias encontradas relacionadas en algunos tópicos al tema de este proyecto. (Blanco, et al., 2008, pp. 25-33).

1.8.1 Resumen de ‘Tres expertos en gerencia de calidad’

Identificación	
Título	Tres expertos en gerencia de calidad
Publicación	Oficina de Liderazgo Total en Calidad
Autores	J. Gerald Suarez
Descripción	
Área	Metodologías de calidad en gerencia
Resumen	Es un resumen de los ideales de calidad de los primeros 3 gurús en la materia, Philip B. Crosby, W. Edwards Deming y Joseph M. Juran
Aspectos a destacar	
<p>Crosby:</p> <ul style="list-style-type: none"> • Calidad: “conformidad con requerimientos” • Prevención sobre detección de errores. • Días con cero defectos • Cuatro absoluciones de calidad: <ol style="list-style-type: none"> 1. Cumplir con requerimientos 2. Prevenir errores 3. Hacerlo bien a la primera 4. La medida de la calidad es el precio de la no conformidad • Proceso de prevención • 6 C’s de Crosby <p>Deming:</p> <ul style="list-style-type: none"> • Calidad: “La dificultad en definir calidad, está en trasladar futuras necesidades del usuario en características medibles, para que así un producto pueda ser diseñado para que su resultado brinde satisfacción a un precio que el usuario pueda pagar”. • Planificar-Hacer-Revisar-Actuar • Sistema de conocimiento profundo <ul style="list-style-type: none"> ○ Teoría del sistema: conocer el negocio. ○ Teoría de variación: saber de estadística para el negocio. ○ Teoría de conocimiento: entender los procesos. ○ Conocimiento de psicología: comprender la cultura del equipo. • Causas comunes y causas especiales <p>Juran:</p> <ul style="list-style-type: none"> • Calidad: “Idoneidad para el uso” • Espiral de progreso en calidad • Trilogía de Juran <ul style="list-style-type: none"> ○ Planificación de calidad es a presupuesto. ○ Control de calidad es a control de costos. ○ Mejora de calidad es a reducción de costos. • Resolver según Pareto de errores 	

(J. Gerald, 1992)

1.8.2 Resumen de 'Guía de Scrum'

Identificación	
Título	La Guía de Scrum
Publicación	Organización Scrum y Corporación Scrum
Autores	Jeff Sutherland y Ken Schwaber
Descripción	
Área	Metodología de Scrum
Resumen	Es la guía oficial de Scrum, disponible en el sitio de la Alianza de Scrum
Aspectos a destacar	
<p>Scrum es:</p> <ul style="list-style-type: none"> • Ligero. • Simple de comprender. • Difícil de dominar. <p>Tres aspectos clave:</p> <ol style="list-style-type: none"> 1. Transparencia: todo observador tiene derecho a saber qué está pasando 2. Inspección: ayuda a detectar retrasos, pero se puede convertir en un retraso en si misma si se hace con exceso de frecuencia. 3. Adaptación: debido a fallas detectadas en la inspección se procede a cambios en el proceso para poder alcanzar un producto aceptable. <p>Cuatro eventos para Adaptación e Inspección:</p> <ol style="list-style-type: none"> 1. Planificación del Sprint <ol style="list-style-type: none"> a. ¿Qué se va a hacer? b. ¿Cómo se va a hacer? c. ¿Cuál es el producto meta? 2. Scrum Diario: No debe durar más de 15 minutos, dirigido por el Scrum Master quién inspecciona ante el equipo de desarrollo <ol style="list-style-type: none"> a. ¿Qué hice ayer? b. ¿Qué voy a hacer hoy? c. ¿Qué obstáculos tengo? 3. Revisión del Sprint <ol style="list-style-type: none"> a. Demostración por el Scrum Master y el equipo de desarrollo ante el Product Owner y sus invitados sobre cuáles fueron los resultados del Sprint. 4. Retrospectiva del Sprint: reunión dirigida por el Scrum Master para el equipo de desarrollo, donde se discute: <ol style="list-style-type: none"> a. ¿Qué se hizo bien? b. ¿Qué se debe mejorar? <p>Cinco valores del equipo de Scrum</p> <ol style="list-style-type: none"> 1. Compromiso 2. Coraje 3. Enfoque 4. Apertura 5. Respeto 	

Roles del equipo de Scrum:

- Dueño del Producto (Product Owner en inglés): responsable de la Cartera de Productos (Product Backlog en inglés) que es donde se definen los requerimientos del producto; único con autoridad para cancelar Sprint.
- Equipo de desarrollo: se les indican los requerimientos, no el cómo hacerlos; multifuncional con todas las habilidades necesarias para culminar el producto; no hay subequipos y son responsables como equipo NO como individuos del desarrollo del producto. Un equipo muy pequeño es de tres miembros, uno muy grande es de nueve, seis sería la cantidad ideal.
- Maestro Scrum (Scrum Master en inglés): Evangelizador de Scrum para la organización, intermediario entre el Product Owner y el equipo de desarrollo, encargado de inspección del progreso y coordinar con el Product Owner para la adaptación.

Ningún Sprint debe durar más de un mes.

(Schwaber & Sutherland, 2016)

1.8.3 Resumen de 'El nuevo XP'

Identificación	
Título	El nuevo XP
Publicación	Universidad de Cagliari
Autores	Michele Marchesi
Descripción	
Área	Metodología XP
Resumen	Es un resumen de la segunda edición del libro Extreme Programming Explained (Explicación de Programación Extrema en inglés) de Kent Beck.
Aspectos a destacar	
<p>Valores de XP:</p> <ul style="list-style-type: none"> • Comunicación: directa entre miembros del equipo, así como entre el equipo y los clientes. • Simplicidad: haga que funcione de la forma más simple. No hacer pensando en futuras opciones, si se hace simple añadir futura funcionalidad es fácil • Retroalimentación: mostrar métricas de progreso y escuchar opiniones del cliente. • Coraje: no tenerles miedo a los cambios, siempre y cuando se cumplan los tres valores anteriores. • Respeto: solidaridad entre los miembros del equipo. <p>Principios de XP:</p> <ul style="list-style-type: none"> • Humanidad: <ul style="list-style-type: none"> ○ Seguridad de mantener el trabajo. ○ Realización de conseguir un logro. ○ Pertenencia a un equipo. ○ Crecimiento en oportunidades para expandir habilidades y perspectivas ○ Intimidad para ser entendido y entender a otros. • Economía <ul style="list-style-type: none"> ○ Si se produce software, también se debe producir valor para el negocio. ○ Un dólar hoy vale más que un dólar mañana. • Beneficio mutuo <ul style="list-style-type: none"> ○ Solucionar más problemas de los que se generan. ○ Toda actividad debe beneficiar a toda la organización. • Auto-similitud <ul style="list-style-type: none"> ○ Soluciones similares para contextos diferentes. • Mejora <ul style="list-style-type: none"> ○ En cada iteración mejorar el sistema tanto en calidad como en funcionalidad, usando retroalimentación del cliente y del equipo, así como también de pruebas automáticas. • Diversidad <ul style="list-style-type: none"> ○ Equipos donde todos son iguales puede traer conformidad, pero no efectividad. 	

- Reflexión
 - Analizar las razones detrás del éxito o del fracaso, sin ocultar errores y discutirlos de manera explícita para el aprendizaje.
- Flujo
 - Desarrollar constantemente software útil, realizando todas las actividades de desarrollo.
 - Las prácticas de XP asumen un flujo continuo de actividades, no una secuencia de fases diferentes, con el lanzamiento del software solamente después del último.
- Oportunidad
 - Los problemas deben verse como una oportunidad de mejora.
 - Si un solo desarrollador genera demasiados errores, entonces que se trabaje en parejas todo el tiempo.
- Redundancia
 - Para evitar que los problemas resurjan hay que solucionarlos de múltiples maneras, programación en parejas, pruebas automáticas de calidad, uno al lado del otro, involucrar al cliente, etc.
- Fracaso
 - Dividir un fracaso en partes y ver cuáles pasaron, así aislando el fallo.
- Calidad
 - Hacer, probar, fallar y luego corregir, los tres últimos son claves en calidad.
- Pequeños pasos
 - Dividir trabajo en pequeñas y simples tareas.
 - Un error en un pequeño paso es solo un pequeño retraso.
- Aceptar responsabilidad
 - Tarea que se acepta se debe cumplir.

XP se conforma de 13 prácticas primarias y 11 corolarios

Las prácticas se dividen en:

1. Análisis de requerimientos y planificación.
2. Equipo y factores humanos.
3. Diseño.
4. Codificación de código y lanzamiento.

Prácticas primarias:

- Análisis de requerimientos y planificación
 - Historias: descripción corta de las funcionalidades que ve el cliente.
 - Ciclo semanal: el cliente decide que historias se van a realizar en la semana.
 - Ciclo trimestral: el desarrollo por lanzamiento se planea de forma trimestral.
 - Suelto: evitar hacer promesas que no se pueden cumplir, clasificar lo que pueden ser historias descartables.

- Equipo y factores humanos
 - Sentarse juntos: los equipos de desarrollo deben trabajar en un ambiente abierto, capaz de albergar a todo el equipo para maximizar la comunicación.
 - Todo el equipo: el equipo en conjunto debe poseer todas las habilidades requeridas por el proyecto. Los miembros deben ayudarse entre sí.
 - Espacio de trabajo informativo: pantallas, pizarras o posters deben mostrar de forma visible el estado del proyecto según las tareas a cumplir
 - Trabajo con energía: limitar la posibilidad de horas extra por individuo, para garantizar un buen desempeño diario.
 - Programación en parejas: el código siempre se escribe por dos personas en una sola máquina.
- Diseño
 - Diseño incremental: el diseño previo es indispensable para obtener buen código, pero solo diseñar lo asignado al ciclo a trabajar.
 - Probar primero: escribir pruebas y verificar código antes de agregarlo al repositorio, pruebas exitosas antes de subir código promueve una mayor confianza entre el equipo.
- Codificación de código y lanzamiento
 - Integraciones en 10 minutos: integraciones largas y complejas dificultan la demostración y retroalimentación.
 - Integración continua: los desarrolladores deben subir código cada dos horas para evitar problemas de integración.

Prácticas corolarias:

- Análisis de requerimientos y planificación
 - Involucración real del cliente: todos los afectados por el sistema deben formar parte de la planificación semanal y trimestral.
 - Desarrollo incremental: desarrollar una función a la vez, no pensar en “todo o nada”.
 - Negociar el contrato de alcance: es mejor una secuencia de contratos cortos para reducir el riesgo.
 - Pago-Por-Uso: promover pago por funcionalidad.
- Equipo y factores humanos
 - Continuidad del equipo: los miembros deben permanecer juntos en varios proyectos, ayuda a establecer una mejor relación.
 - Encoger equipos: conforme el equipo se vuelve más capaz y productivo, la carga puede permanecer igual; pero se envían otros miembros a fortalecer otros equipos.
- Diseño
 - Análisis de causa raíz: también conocido como RCA (Root Cause Analysis, en inglés), cada vez que se encuentra un defecto, se deben eliminar él y sus causas, evitando que el problema se repita de nuevo.

- Codificación de código y lanzamiento
 - Codificar y probar: es el concepto clave de XP.
 - Compartir código: cualquier miembro del equipo de desarrollo debe ser capaz de cambiar cualquier parte del sistema en cualquier momento
 - Única base del código: evitar diferentes ramas de repositorios, procurar encontrar pronto los errores al unir código en el repositorio.
 - Despliegue diario: cada noche nuevo software debe ser puesto en producción. Es riesgoso y costoso tener brechas entre versiones de software en prueba y la que está en producción.

(Marchesi, 2005)

1.8.4 Resumen de ‘Guía de ITIL para DevOps’

Identificación	
Título	Guía de ITIL para DevOps
Publicación	Upguard
Autores	Compañía Upguard
Descripción	
Área	Mezcla de ITIL con Desarrollo de Operaciones
Resumen	Explica la relación que puede tener ITIL al involucrarlo con metodologías ágiles e implementarlo con el desarrollo de operaciones dentro de una gran empresa.
Aspectos a destacar	
<p>Servicio de Transición -> Gerencia de Cambios: mejorar comunicación de ITIL en el manejo de CR's (Change Records por sus siglas en inglés), al traer aspectos como el rol de Product Owner y la actividad de Backlog Grooming (limpieza de pendientes, traducción del inglés) en Scrum, el Slack de XP, Planificación del Sprint en Scrum o del Ciclo en XP.</p> <p>Servicio de Transición -> Gerencia de Lanzamiento e Implementación: antes del lanzamiento existe la Prueba de Aceptación del Usuario (User Acceptance Test [UAT] en inglés), dependiendo de los resultados de tales pruebas se procede a la implementación. Diferentes herramientas tanto libres como propietarias están disponibles para cada uno de estas fases:</p> <ul style="list-style-type: none"> • Automatización 	
	
<p><i>Ilustración 5 - DevOps Software de Automatización</i></p>	
<ul style="list-style-type: none"> • Pruebas funcionales 	
	
<p><i>Ilustración 6 - Software de Pruebas Funcionales</i></p>	
<ul style="list-style-type: none"> • Integración continua 	
	
<p><i>Ilustración 7 - Software de Integración Continua</i></p>	

Servicio de Operación -> Gerencia de Incidentes: este es el manejo de Bugs o fallos en software que ya haya sido implementado; en la categoría de Diseño, XP maneja los incidentes de manera prioritaria con RCA, a diferencia con Scrum la calidad no es tan prioritaria como tampoco el arreglo de fallos.

Mejora Continua del Servicio -> Gerencia del Conocimiento: La sugerencia con DevOps es no solo documentar lo que da más errores, sino también automatizar dichas tareas.

Tanto ITIL como las diferentes metodologías ágiles, son sugerencias a implementar; durante los años la práctica ha vuelto que la implementación se más y más estricta y ese no fue el propósito inicial de ninguna de estas; la falta de orden a un alto nivel del negocio del que sufren las metodologías ágiles puede beneficiarse de ITIL, así como también ITIL puede beneficiarse del enfoque de mejora a bajo nivel tanto en comunicación como desempeño y calidad que brindan las metodologías ágiles.

(UpGuard, 2017)

1.8.5 Resumen de ‘Medición de ROI en transformaciones ágiles del negocio’

Identificación	
Título	Midiendo el Retorno de la Inversión de Transformaciones Agiles
Publicación	Project Management Institute
Autores	Equipo PMI India
Descripción	
Área	Metodologías Ágiles, Scrum, Transformación, Adopción, Métricas, ROI
Resumen	Presenta los diferentes modelos de medición para medir la salida de una transformación hacia métodos ágiles en la organización.
Aspectos a destacar	
<p>Identificar los beneficios del Cliente: el cliente va primero, tendrá respuesta más rápida y menos burocracia para solicitar cambios en requerimientos previos</p> <p>Identificar los beneficios del Negocio:</p> <ul style="list-style-type: none"> • Empresas de Servicios <ul style="list-style-type: none"> ○ Mayor satisfacción del cliente: se puede medir con Índice de Satisfacción del Cliente capturado por medio de encuestas de satisfacción del cliente. ○ Compromiso a largo plazo: se trabaja en equipo junto al cliente con sinergia hacia una meta común. • Empresas de Producción: <ul style="list-style-type: none"> ○ Tiempo más rápido de salida al mercado: ayuda a responder rápidamente a las solicitudes de los clientes sobre nuevas características, como también a responder ante lanzamientos de la competencia. ○ Ciclo de realización de ingresos más corto: lanzamiento de nuevas características en intervalos cortos. • Empresas Emergentes <ul style="list-style-type: none"> ○ Rápida validación de idea del negocio: es importante validar la idea del negocio con potenciales clientes, para asegurar el potencial del negocio evitando invertir en lo que el mercado no desea. ○ Afluencia de ingresos: en este tipo de empresas, el flujo de ingresos estable es aún más importante para mantener su existencia. Lanzamientos más frecuentes de productos o servicios son necesarios. <p>Darse cuenta de más rápido de los beneficios del proyecto.</p> <p>Costos de implementar metodologías ágiles:</p> <ul style="list-style-type: none"> • Infraestructura para integración continua. • Herramientas y licencias para automatización y manejo de tareas. • Entrenamiento y asesoría. • Reconocimiento de equipos exitosos. <p>Parámetros de Retorno de la Inversión:</p> <ul style="list-style-type: none"> • Cambio de fechas en entregables de proyectos. • Valor para el cliente. • Mejora de eficiencia del equipo. • Reducción de basura (5's). 	

Métricas para ambientes ágiles:

- Métricas del negocio
 - Ganancia para el negocio.
 - Retorno de la inversión.
 - Ciclo de lanzamientos.
 - Costo de retrasos.
 - Plazo de ejecución.
 - Tiempos de ciclos.
- Métricas de procesos
 - Velocidad.
 - Tamaño del Backlog (Lista de requerimientos).
 - Requerimientos completados en la iteración.
 - Requerimientos sin terminar en la iteración.
- Métricas de calidad
 - Defectos por requerimiento.
 - Pruebas de aceptación por requerimiento.
 - Defectos sin solucionar.

(PMI India Team, 2015)

Capítulo 2. Marco teórico

2.1 Descripción teórica de BI

El propósito de la Inteligencia de Negocios, no es desarrollar el mejor almacén de datos (DW por sus siglas en inglés), mercado de datos (DM por sus siglas en inglés), el Manejo Maestro de Datos (MDM por sus siglas en inglés) o el mejor sistema de Extracción Transformación y Carga de datos (ETL por sus siglas en inglés); sino su fin real es brindar información para procurar tomar decisiones más inteligentes para el negocio mediante el análisis histórico de los datos del mismo.

Profundizando un poco más en la descripción de BI (Business Intelligence en inglés) y citando un anterior trabajo de postgrado, "... Una parte que va muy unida a la Inteligencia de Negocios es la fuente, limpieza, veracidad y actualidad de los datos; también involucra un poco de análisis para detectar patrones de datos o grupos que presenten semejanzas en comportamiento y sean considerados para la creación de reportes finales, mucho de esto conforma la minería de datos (Data Mining en inglés)" (Plá Rojas, 2013, pág. 18).

Algunos términos han alcanzado una alta popularidad y con ello ha crecido una confusión al mezclar términos de análisis como el de Minería de Datos con el de Inteligencia de Negocios. En un anterior proyecto final de graduación realizado por este mismo autor se da una sencilla explicación para individuos ajenos al tema sobre los conceptos mínimos de BI, para ello se recomienda una ligera lectura rápida a la sección de *Glosario y Acrónimos* para garantizar una mejor comprensión de este documento.

Algunas de las tareas de BI son: generación de reportes, proceso en vivo de análisis, minería de datos, manejo de desempeño del negocio, modelos de referencia (Benchmarking en inglés) y predicción de comportamiento. O sea, BI es el punto de intersección entre la tecnología y las decisiones empresariales. Pero el negocio siempre es continuo por lo cual el análisis del mismo también debe serlo, no es seguro confiar solo en datos finales cuando el momento de tomar acciones correctivas ha pasado; sino en el proceso actual del negocio, así como también en el estudio del conjunto histórico que muestre tendencias de comportamiento.

Es importante mencionar que el Análisis de Negocios hace uso extenso de los datos, estadísticas, análisis cuantitativos, modelado predictivo, y análisis de hechos relevantes para poder dirigirse a la toma real y actual de decisiones.

Las decisiones son siempre para enfrentar el futuro, en gran parte considerando los factores históricos, así como también tomando en cuenta posibles acontecimientos; esto es lo que se considera como Análisis Predictivo.

Antes de profundizar más en BI, es importante tener presente que la misma requiere una comprensión amplia del negocio, para posteriormente poder ahondar en temas de agrupación, análisis y reporte de datos; tal comprensión es vital antes de iniciar un trabajo de BI. Se trata de hacer un negocio más inteligente, no necesariamente de más reportes dinámicos con hermosos diseños. Por ello un Análisis Empresarial, un Plan Estratégico o siguiendo la metodología de ITIL un Acuerdo a Nivel de Servicios (SLA por sus siglas en inglés) son bases importantes para orientar la recopilación de datos y métricas o Indicadores Claves de Desempeño (KPI's por sus siglas en inglés) sobre los mismos.

2.1.1 BI tipo DW 1.0

Los precursores y evangelistas iniciales de BI son Ralph Kimball y Bill Inmon, aunque entre ambos existen algunas discordancias; Kimball promueve que varios departamentos usualmente generan sus propios mercados de datos y donde el siguiente paso es crear un almacén de datos que incluya todo para ser la única fuente de los análisis, su diseño se puede ver en el *Apéndice Ap 1*. Por otro lado, Inmon sugiere una unificación inicial de todos los datos, de donde cada departamento tomará del almacén solo lo que les interese y crearan su propio mercado de datos, una mayor descripción se puede encontrar en el *Apéndice Ap 2*.

La versión de Kimball tiende a ser más usada entre más grande sea la compañía, viéndolo como cada departamento responsable de su DM posterior a ello la alta gerencia solicita a un equipo de análisis de datos la unificación de cada DM para la creación de un DW que permita una visión general e integrada del negocio.

2.1.2 BI tipo DW 2.0

Bill Inmon en el 2008 lanzó una nueva propuesta de arquitectura para BI, la denominó DW 2.0. Esta propuesta deja de lado los DM's y se enfoca en DW's, solo que en lugar de tener como en DW 1.0 un solo DW, con la nueva propuesta se sugiere un mínimo de 2 DW. La razón es que los DW's serán como tablas particionadas por tiempo, en el primer DW se guardan los datos recientes con los que el negocio se está llevando a cabo, inventario actual, ventas del mes, etc. Mientras que para el segundo DW se encarga de almacenar datos ya históricos.

Igual que el DW 1.0, el proceso de ETL sigue siendo requerido y se da hacia un Staging Area (Área de escena). El SA es una copia directa de los datos transaccionales de una forma que no afecte el desempeño de las operaciones del negocio, la limpieza como estandarización de datos se realiza en esta etapa.

El diagrama de DW 2.0 se puede encontrar en el *Apéndice Ap 3*.

2.2 Metodologías para el negocio

¿Cuál escoger ITIL, Kanban, XP, Scrum? La realidad es que son metodologías, no prácticas estandarizadas. Son más bien sugerencias de qué hacer, algunas son pensadas hacia tipos de negocios específicos como el de pequeños proyectos de desarrollo de software en el caso de Scrum o en la misma área, aunque más orientado a calidad, diseño y compartir conocimiento, como es el caso de XP. Por otro lado, dos metodologías que nacieron en ambientes de mejora de calidad de servicios como es el caso de ITIL, o en el ambiente de manufactura para una forma rápida de manejar y velar por el cumplimiento de tareas como es el caso de Kanban.

El fin del equipo de SGI es el desarrollo de pruebas de calidad, para ser aplicadas en la línea de producción, al igual para la aprobación de cuáles equipos, partes opcionales o firmware pueden pasar a manufactura. XP posee más aspectos orientados a calidad que la metodología de Scrum, pero por otro lado esta última tiene más aspectos de planificación para cada Sprint de Scrum o ciclo de XP, ambas tienen la parte de la poda de la pila de requerimientos (Backlog grooming en inglés), pero es Scrum quien tiene un rol más de maestro de obras refiriéndose al Scrum Master a diferencia de XP donde el cliente se involucra de forma directa con todo el equipo. Sin

dejar de lado también está Kanban, que sirve para enfocar el trabajo del día a día, trabajar en lo que sea prioritario.

La metodología de SAFe procura integrar varias metodologías ágiles para desarrollo de servicios para su manejo intermedio y de PMP para la planificación inicial, esto puede constatarse en el diagrama que se encuentra en el *Apéndice Ap 4*. En el primer nivel, se utiliza la definición de roles de Scrum, la parte de diseño y calidad de XP, así como la de control diario de esfuerzo que recomienda Kanban. En la parte media se declaran la línea de programas y de soluciones grandes, aquí se podría referir a ITIL control de cambios y control de lanzamientos (Change Management, Release Management, en inglés) que son parte de la etapa de transición (Service Transition, en inglés) de ITIL, y por último el nivel superior el cual es más de manejo de proyectos semejante a los portafolios realizada por un PMP.

2.3 Metodología ITIL

Semejante a Prince2 o COBIT, ITIL también nació en el Reino Unido; aunque lo que lo diferencia es la implementación. ITIL es una metodología maleable en su implementación según las necesidades del negocio, no un estándar rígido. A pesar de nacer en los años 1980's siguiendo enseñanzas de Edwards Deming sobre control de procesos, fue hasta los años 1990's que tomo importancia en el sector privado, seguido de varias versiones, v2 en el 2001, v3 en 2007 y la 2011 en respectivo año. Con cada versión se procura brindar más ejemplos de forma directa sobre casos de implementación y sus beneficios.

ITIL consta de cinco ciclos:

1. Estrategia del Servicio
2. Diseño del Servicio
3. Transición del Servicio
4. Operación del Servicio
5. Mejora Continua del Servicio

Como se aprecia en la siguiente imagen, la piedra angular de la metodología es la Estrategia del Servicio, es en esta etapa donde se definen los servicios que se le proveerán al cliente. Y en la siguiente etapa de Diseño del Servicio se crea el Acuerdo a Nivel de Servicio (SLA, por sus siglas en inglés) con el cliente, así como de ser

necesario el Acuerdo a Nivel Operacional (OLA, por sus siglas en inglés) entre departamentos.



Ilustración 8 - Ciclo de ITIL (ProactivaNET, 2017)

Al ser un proceso de pruebas de calidad en la línea de manufactura, lo más esencial son los cambios y las nuevas implementaciones. ITIL dentro de la etapa de Transición consta de roles administración de cambios, administración de lanzamientos, validación y pruebas; como también en la etapa de Operación, los roles de administración de incidentes, administración de problemas y administración de solicitudes. Existen muchos otros procesos y sub-procesos, pero de nuevo, ITIL no se trata de tomarlo en su totalidad, sino de ir tomando los bloques más necesarios para el negocio.

Difícilmente se puede catalogar ITIL como una metodología ágil, pero sí se puede adaptar para acaparar la etapa media de SAFe, donde se brinda un mayor orden a la solicitud de cambios y lanzamientos, además de la administración de incidentes y un esfuerzo para evitar que estos se repitan que es la parte de manejo de problemas

y de conocimiento y asemeja la práctica de diseño en XP. En la fábrica la solicitud de cambios es continua, por lo que no puede ser considerado un proyecto debido a que no se tiene un principio y un fin. Los pilares de un servicio son su constancia, confianza y calidad, y ahí es donde entran a funcionar los acuerdos y roles de ITIL que brindan una garantía en el servicio acordado.

ITIL promueve el control de procesos, mediante la medición periódica de métricas acordadas entre los líderes de organizaciones de parte del solicitante y el proveedor de servicios. Un proyecto se puede definir como una carrera planeada con un principio y un fin previamente establecido; ahí es donde difiere principalmente de un servicio, donde se inicia, pero no se estima un fin, sino una relación mutua que pueda mejorar a largo plazo.

Capítulo 3. Marco metodológico

Este apartado pretende dar a conocer la metodología que se utilizó en este proyecto manteniendo en el mismo un enfoque que procura mejorar la calidad en los procesos del negocio.

3.1 Tipo de investigación

Aun cuando lo que se busca es mostrar métricas, esto sería un resultado del proyecto, en cambio en su transcurso lo que se pretende es comprender la calidad por separado y en conjunto de diversas metodologías ágiles. Es por ello que el enfoque de este proyecto es cualitativo, donde se busca no solo eficiencia, sino además eficacia de la metodología a usar entre el cliente y el proveedor.

3.2 Alcance investigativo

La investigación evaluativa pretende determinar las diferentes metodologías a utilizar en el departamento de SGI que puedan beneficiar su relación con los solicitantes y clientes.

3.3 Enfoque

Al tratarse de una investigación sobre el accionar real de un equipo de trabajo, no se dispone de estudios con semejanzas suficientes como para ser comparados, al contrario, lo que se busca es una mejora en los procesos utilizados específicamente en el departamento de SGI y que además muestren un impacto en relación con la Cadena de Suministro, como entrada de solicitudes y las fábricas como los ejecutores del trabajo aportado por SGI. El impacto de la calidad es vital para un departamento que lo que busca es mejorar la calidad de los productos en cada una de las fábricas.

3.4 Diseño

El modelo a utilizar es el CIPP de Stufflebeam, dicho modelo tiene un gran parecido a procesos de mejora de calidad como SIPOC. Donde se inicia con una serie de preguntas simples cuyas respuestas dan una visión más clara del proceso.

- Identificando los extremos:
 - ¿Quién es el proveedor y quién el cliente final?
R/ Para SGI los proveedores son los ingenieros de la cadena de suministro, mientras que el cliente final es fábrica. Puede que haya

casos donde fábrica se transforme en proveedor, pero esos son casos fortuitos.

- Identificar entradas y salidas:
 - ¿Qué entra y qué sale de SGI?
R/ Entran solicitudes de cambio, que pueden ser interpretadas como CR's viéndolo como ITIL o User Stories bajo una visión de Scrum. Y la salida sigue siendo pruebas de calidad para los productos en la línea de manufactura.
- Identificar el proceso:
 - ¿Cuál es el proceso?
Aquí es donde se vuelve importante la definición de responsabilidades entre los diversos involucrados y como medir tanto la eficacia como eficiencia del proceso acordado entre las partes.

3.5 Población y muestreo

La población es justo el departamento de SGI y el muestreo es la recopilación de datos de cada lanzamiento a fábrica. Para el caso del departamento de calidad de SGI, toda solicitud de cambio asignada a este departamento en el área de sostenibilidad de producción forma parte de la población a contabilizar.

3.6 Instrumentos de recolección de datos

La base de datos de QUIX a la cual se tiene acceso con una cuenta que solo mantiene permisos de lectura, es sumamente importante para la empresa, por lo que se debe procurar que las lecturas para las copias a la etapa interactiva, sean rápidas y eficientes, para que afecten en lo menos posible el desempeño de la base de datos de producción.

Lo más importante es copiar la tabla que mantiene el detalle de cada CR, aunque siguen existiendo muchas otras tablas relacionadas pero cuyo valor sería solo de minería de datos y están fuera del alcance de este proyecto, como lo son las tablas de archivos adjuntos que tiene cada CR o la tabla de notas, donde igual un CR puede tener muchas notas relacionadas.

3.7 Técnicas de análisis de la información

El análisis de la información será el producto final de este proyecto, pero las técnicas a utilizar iniciales puede que cambien según la necesidad del negocio o el nivel jerárquico de los interesados en los reportes.

La utilidad de diferentes técnicas de análisis depende de a quién se les muestren; el gráfico de Burndown, la velocidad del Sprint y el peso de los User Stories; son técnicas muy valiosas para un Scrum Master, pero pierden algo de interés para el Product Owner, y aún más para el vicepresidente de la organización o gerente a nivel de servicios (SLM, por sus siglas en inglés), por último, son casi inentendibles para gerentes del lado del cliente. Es por ello que, a altos niveles organizacionales, las métricas más valiosas son aquellas que se acordaron entre organizaciones o en contratos con clientes y ya no tanto las sugeridas a bajo nivel por metodologías ágiles.

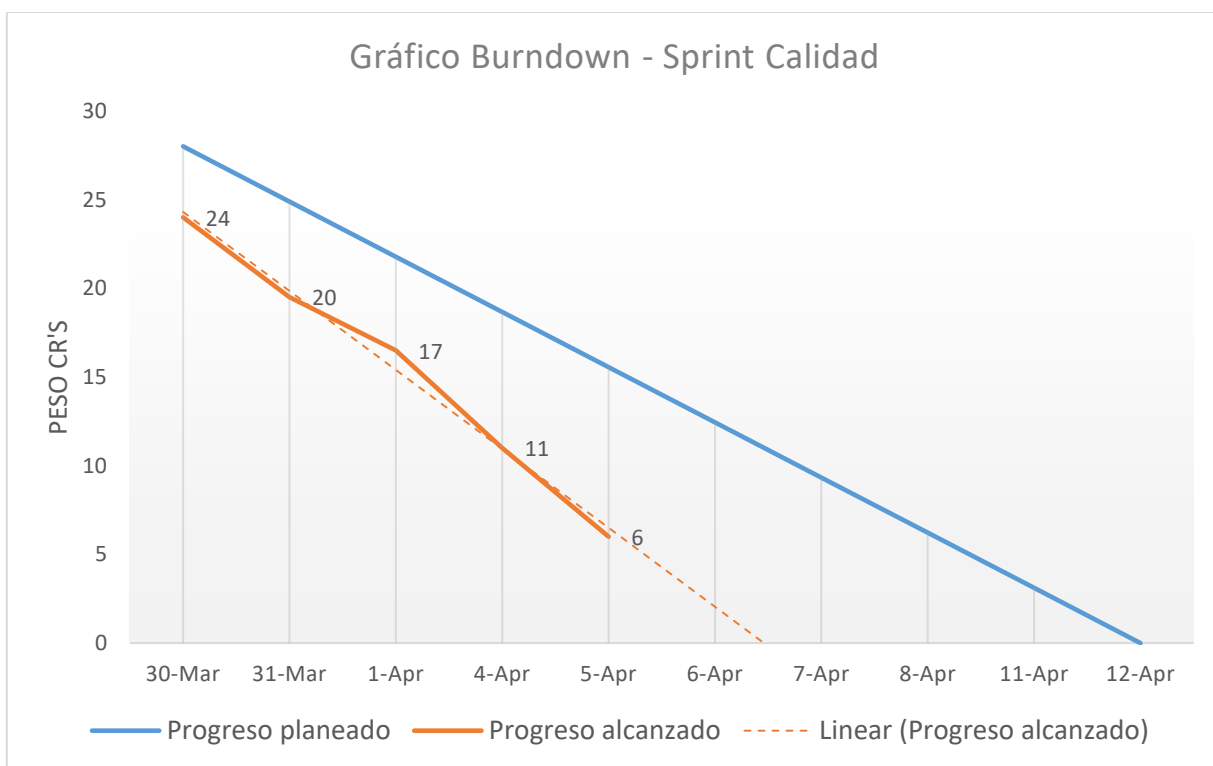


Ilustración 9 - Gráfico de Burndown

3.8 Descripción de Scrum en SGI

Cuando SGI empezó con entrenamiento de SAFe, lo que se abrazó fue el primer nivel de esta metodología y se inclinó por el uso de Scrum. El primer nivel de SAFe va más dirigido al equipo de desarrollo y ligeramente también se aplicó el segundo nivel

mediante la práctica de la reunión de Scrum de Scrums, igual para unir el trabajo de diferentes equipos de desarrollo dentro del departamento.

El proceso de Scrum dentro de SGI trajo cierto orden a las operaciones internas de la organización. Pero al carecer Scrum de aspectos imprescindibles para SGI como el manejo de errores en fábrica, que sí son considerados como Bugs por XP, se introdujo un modelo de WarRooms que viéndolo desde un punto de vista de ITIL es el manejo de Incidentes; mientras se mantuvo Scrum para el manejo de Cambios. También la parte de “Go – No Go” es una reunión de validación donde se decide que pruebas de calidad de cada producto fueron aprobadas para implementarse en fábrica, lo que viene a ser en ITIL el manejo de Validación de la etapa de Transición.

Toda la definición del proceso anterior era comprendida en su mayoría de manera interna por SGI, pero no más allá con el proveedor quien en este caso son los Ingenieros de Producto de la Cadena de Suministro (SC GPE's, por sus siglas en inglés).

Antes de profundizar más en la descripción de la adaptación de Scrum en el departamento de SGI, es importante recordar los pasos que conforman esta metodología mencionados en el Resumen de ‘Guía de Scrum. Primero hay una cola general de trabajo que se le llama Product Backlog; de ahí el Product Owner revisa el trabajo en la cola y se escoge qué es lo más importante a trabajar para el siguiente Sprint, a esta actividad se le llama Sprint Planning que se hace junto con el Scrum Master porque este dice cuántos son los recursos disponibles para el siguiente Sprint. Después de que el PO (*Product Owner*) y PM's (*Project Managers*) realizan el *Sprint Planning*, se sugiere el *Sprint Backlog* a los SM's (*Scrum Masters*) de cada equipo, estos lo revisan con los desarrolladores y equipo de calidad en busca de obstáculos obvios iniciales y sugieren rechazarlos y que el PO los retire del Sprint Backlog.

Dentro de cada Sprint hay un seguimiento diario de cómo va el trabajo, si hay algún obstáculo, qué se hizo el día anterior y qué se va a hacer hoy, estas preguntas las dirige el Scrum Master a cada uno de los miembros del equipo; con esta reunión se lleva un control de la velocidad del equipo y un gráfico que se llama Burndown el cual es una medición del trabajo pendiente en una línea de tiempo que es la duración del Sprint. Al final de cada Sprint hay una reunión con el Product Owner y los clientes

donde se presenta la culminación del trabajo realizado, y seguida otra reunión más interna donde se discute qué se puede mejorar para el próximo Sprint.

Ahora un lanzamiento o Release en inglés, puede estar compuesto por varios Sprints, en el caso de SGI que hay varios equipos desarrollando pruebas a la vez, el primer Sprint es de varios grupos desarrollando y el segundo es de otro grupo validando todo en conjunto, esta parte de calidad parece algo más a XP o administración de validación en ITIL.



Ilustración 10 - Scrum en SGI

Las fábricas siempre están activas y con órdenes entrantes por lo que no cabe posibilidad de que esperen largo tiempo por nuevos paquetes de pruebas. Es por ello que en este caso los Releases (lanzamiento de paquetes de pruebas), son en general de tres semanas. Las primeras dos semanas que son de desarrollo y las últimas dos semanas de validación, por lo que la segunda semana de desarrollo se traslapa con la primera de validación y esto genera una mayor interacción entre ambos equipos.

El segmento de desarrollo consiste de un equipo de herramientas, uno de firmware, otro de pruebas comunes en todos los equipos y por último uno que se especializa en cada uno de los tipos de servidores. Por otro lado, el equipo de calidad lo valida todo en conjunto, y es su decisión el qué se lanzará o no a fábrica.

El *Sprint Planning* procura cubrir todos los defectos urgentes, además de todas las órdenes especiales y distribuirlos entre los equipos de desarrollo.

Cada requerimiento se da por completado por el equipo de desarrollo cuando se marca como listo para pruebas de calidad, esto debe ser antes del día 14 de cada

Sprint. El día 14 los equipos de desarrollo pasan a trabajar en el siguiente *Sprint* del próximo *Release*, aunque dando prioridad a correcciones que encuentre el equipo de validación en la última semana de su *Sprint* y del *Release*. Puesto el día 21 se da la aprobación de que se puede enviar a fábrica y qué debería corregirse en el siguiente *Release*.

Durante el *Sprint* se pueden encontrar diversos obstáculos o también emergencias que vengan de fábrica u órdenes especiales que vengan de clientes importantes, es por ello que los requerimientos pueden variar durante el *Sprint*, eso es lo que debe contemplarse entre el colchón de trabajo por cada *Scrum Master* y medirse contra la volatilidad del *Sprint*.

La capacidad de trabajo que los *Scrum Masters* aceptan para sus equipos se calcula según el promedio de requerimientos exitosos en un histórico de Sprints, a esto se le conoce como la velocidad de cada equipo de Scrum.

La velocidad de desarrollo también se contempla según el progreso de los desarrolladores con cada requerimiento asignado; esto puede verse en el gráfico *Burndown* que puntúa según la evolución de los requerimientos de acuerdo a su estado.

Al contrario, el interés de los *Product Owners* y gerencia no es el progreso de cada requerimiento, sino lo que se ha completado durante el transcurso del *Sprint*, esto igual se mide, pero en un *Burndown* estratégico que considera los requerimientos solo como pendientes o completados.

Lo anterior trae varios problemas, entre ellos, la ausencia de una planificación ideal donde no se requieran cambios en el ciclo de *Sprint* debido a urgencias en órdenes o incidentes en fábricas. Además, la necesidad de mantener una auditoría constante sobre el proceso conlleva mucho trabajo a los diferentes *Scrum Masters* que deben dedicar gran parte de su día a día a la elaboración manual de gráficos. Tales gráficos actualmente se limitan a *Burndown* de progreso del *Sprint*, que son comprendidos por gerencia directa pero no a nivel ejecutivo, lo que demuestra una ausencia de herramientas que brinden métricas en tiempo real sobre el negocio a nivel de requerimientos en pruebas de calidad.

3.9 Estrategia de desarrollo de la propuesta

La estrategia de desarrollo de la propuesta en este caso se dividirá en tres fases.

Tabla 2 - Fases de implementación

Fase	Descripción
Primera fase	Identificar las necesidades del negocio semejante a un Plan Estratégico, planteado en el marco metodológico de ITIL, en busca de un acuerdo de nivel operacional (OLA, por sus siglas en inglés). El objetivo es darle una mayor fortaleza de aceptación a la implementación de un proceso ágil con una visión más formal y comprensión de los procesos entre departamentos.
Segunda fase	Implementación de Inteligencia de Negocios, inclinándose por seguir el planteamiento de DW 2.0 promovido por Bill Inmon (Inmon B. , 2010) y descrito en el Marco Teórico en la sección <i>BI tipo DW 2.0</i> .
Tercera fase	La etapa de reportes, cuya importancia es vital para el éxito y la valoración del proyecto por parte del negocio. Sin una visión de los reportes de métricas acordados en la primera etapa y conseguidos mediante la segunda etapa, el seguimiento de lo acordado se vuelve inalcanzable. Para esta etapa se procura utilizar Tableau como herramienta de reporte; aunque otros usuarios también hagan uso de herramientas como Excel con PowerPivot, para acceder de forma directa a los datos en cada una de las etapas de BI.

Capítulo 4. Implementación de ITIL

Previo a empezar una implementación de Inteligencia de Negocios, es necesario tener un claro entendimiento del negocio. Entender qué es lo que el negocio quiere alcanzar y cómo lo va a lograr, es el pilar para tener una idea de qué es lo que se necesita medir y es aquí donde se incluye ITIL.

Las variables en un negocio son muchas y puede que el cambio en su mayoría no sea algo que traiga un mayor efecto en los resultados. Por ello comprendiendo el negocio, se puede escoger mejor que es lo que hay que mejorar, que haya sinergia entre todos los involucrados de cuál es la meta a conseguir, y así entender qué es lo que se debe medir.

4.1 SIPOC

Los procesos de mejora como Kaizen, 6Sigma, TQM, hacen referencia en un inicio de mejora de calidad, a tener un claro entendimiento del SIPOC para el negocio. Haciendo referencia a un ensayo de Michael Scarborough en Global Knowledge, “SIPOC se puede utilizar como un medio para definir cualquiera de los procesos de administración de servicios presentados en las mejores prácticas de ITIL®” (Scarborough, 2015, pág. 3).

- **S**upplier, en español Proveedor
- **I**nput, en español Entrada
- **P**rocess, en español Proceso
- **O**utput, en español Salida
- **C**ustomer, en español Cliente

Algo tan simple como esta definición, logra traer a flote las posibles confusiones que existan en el negocio; muchas veces el mismo que solicita algo va a ser quien lo vaya a utilizar, pero ese no es el caso en el proceso de desarrollo de pruebas de calidad.

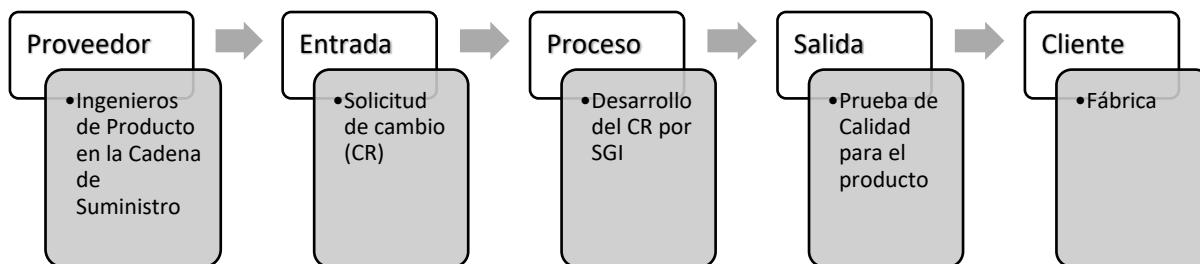


Ilustración 11 - SIPOC de SGI

En este caso la prontitud en la resolución de un CR es lo que espera el solicitante, pero el proceso dentro de SGI tampoco es sencillo y conlleva la coordinación de varios sub-equipos para poder llevar a cabo las tareas asignadas. Por otra parte, del lado de SGI se percibe una confusión de importancia entre fábrica y los Ingenieros de Producto de la Cadena de Suministro.

La confusión se origina ya que es frecuente que fábrica también envíe solicitudes de cambio. Ello se debe a fallos encontrados en la línea de manufactura, pero fallos no en los productos en sí que según ITIL se considerarían como eventos, sino en las pruebas de calidad, lo que se considera como un incidente que hay que corregir con urgencia según el nivel de impacto en producción. Es por ello que cuando el solicitante era fábrica, el modelo de SIPOC no calzaba y las solicitudes de parte de Ingenieros de Producto pasan a un segundo plano.

4.2 Necesidad de formalizar el proceso entre departamentos

Un problema que trajo la implementación de Scrum solo dentro de SGI, sin comprensión para el solicitante GPE, ni para el cliente que son las fábricas; es el rechazo de CR's ya avanzado el proceso de desarrollo. El no ser claros en el motivo de un rechazo, como tampoco en la tardanza de dos semanas en empezar a trabajar un CR, trajo escalaciones de parte de GPE hacia SGI, como también búsquedas de cómo hacer las cosas por debajo como por ejemplo procurar clasificar todo nuevo CR como si fuera un Incidente de WarRoom.

Las faltas de entendimiento del proceso trajeron faltas a su seguimiento o una falta de respeto al mismo, violaciones a las prácticas de Scrum, como el irrespetar al Product Owner así como también al Scrum Master al contactar de forma directa al Desarrollador e involucrarlo en muchas más reuniones que solo el Daily Scrum.

4.3 Acuerdo a nivel operacional

ITIL en la etapa de diseño, dentro de lo que se define como Mejora Continua del Servicio (CSI, por sus siglas en inglés), sugiere la implementación de acuerdos internos entre departamentos de una misma organización; todo con el fin de alcanzar una sinergia en búsqueda de una misma meta. Dichos acuerdos se conocen en inglés como Operational Level Agreement (OLA) y a pesar de no ser documentos legales con multas entre el solicitante y el proveedor del servicio, si fomentan un compromiso entre las partes, un respeto por los procesos y requerimientos del mismo.

Los alcances de Scrum y XP en SGI sufrían de la amenaza externa de una falta de entendimiento por parte del proveedor, que como se mencionó anteriormente, son los Ingenieros Globales de Producto de la Cadena de Suministro (SC GPE, por sus siglas en inglés). Una falta de comprensión a nivel externo, descalifica las mejoras realizadas a nivel interno en SGI.

El OLA consta de varios apartados, entre ellos los más relevantes para la solución del problema son:

- Alcance: indica las limitaciones de la extensión del acuerdo, en este caso solo entre SC GPE y SGI para servidores en producción, excluyendo fábricas, laboratorios y proveedores externos.
- Descripción del servicio: aclarar el proceso entre los interesados, un diagrama de flujo del proceso ayuda a conceptualizar el servicio.
- Roles y responsabilidades: detallar las responsabilidades para cada rol, tanto del lado de SC como de SGI.
- Métricas: son las mediciones a utilizar para procurar una calidad en el servicio.

4.3.1 Diagrama de flujo del proceso

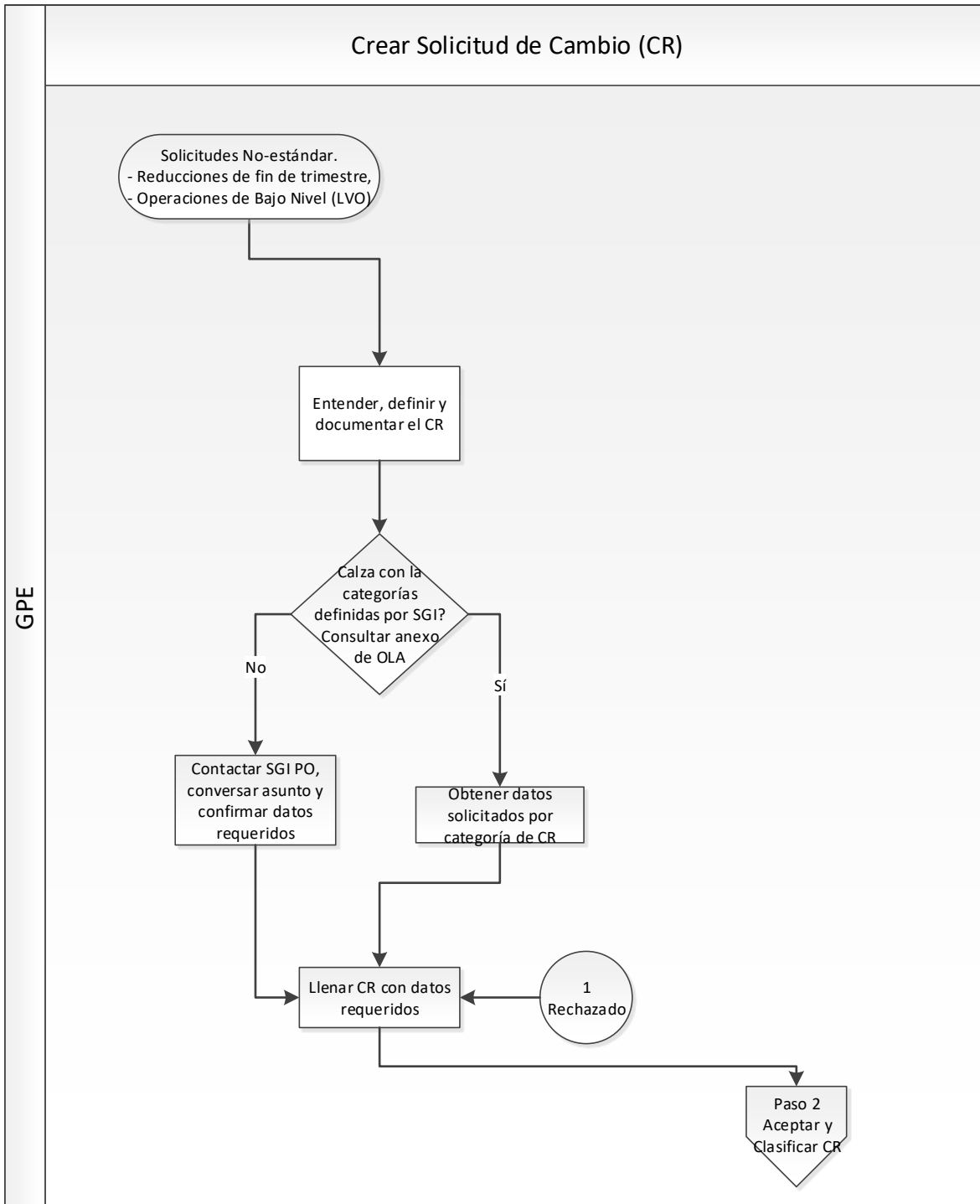
El diagrama se dividió en cinco etapas con dos actores. Los actores son los departamentos de GPE como el proveedor y el de SGI como quien brinda el servicio. Este diagrama busca abarcar dentro de ITIL, los conceptos utilizados por SGI de Scrum y XP.

Etapas:

1. Crear Solicitud de Cambio (CR)
2. Aceptar y Clasificar CR
3. Scrum Trabajo en Proceso
4. Incidentes de WarRoom
5. Lanzamiento del CR

El diagrama puede verse en su totalidad en el *Apéndice Ap 5*, para este apartado se profundizará en cada una de las etapas.

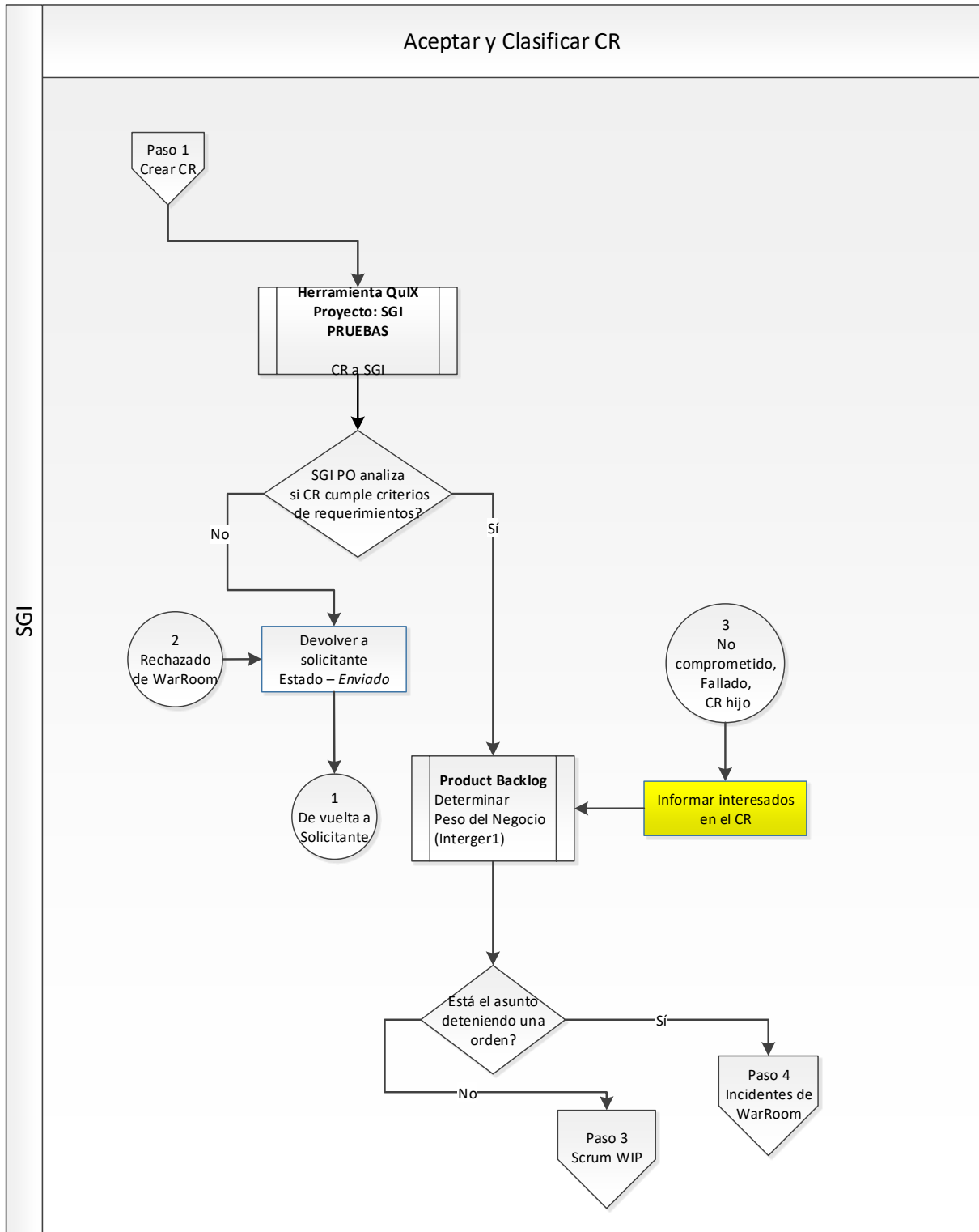
4.3.1.1 Crear solicitud de cambio (CR)



Esta primera etapa detalla, a como lo describe el SIPOC, el momento cuando el proveedor ingresa la entrada o, en palabras más propias de este proceso, cuando el ingeniero de producto va a ingresar la solicitud de cambio en la herramienta de QUIX.

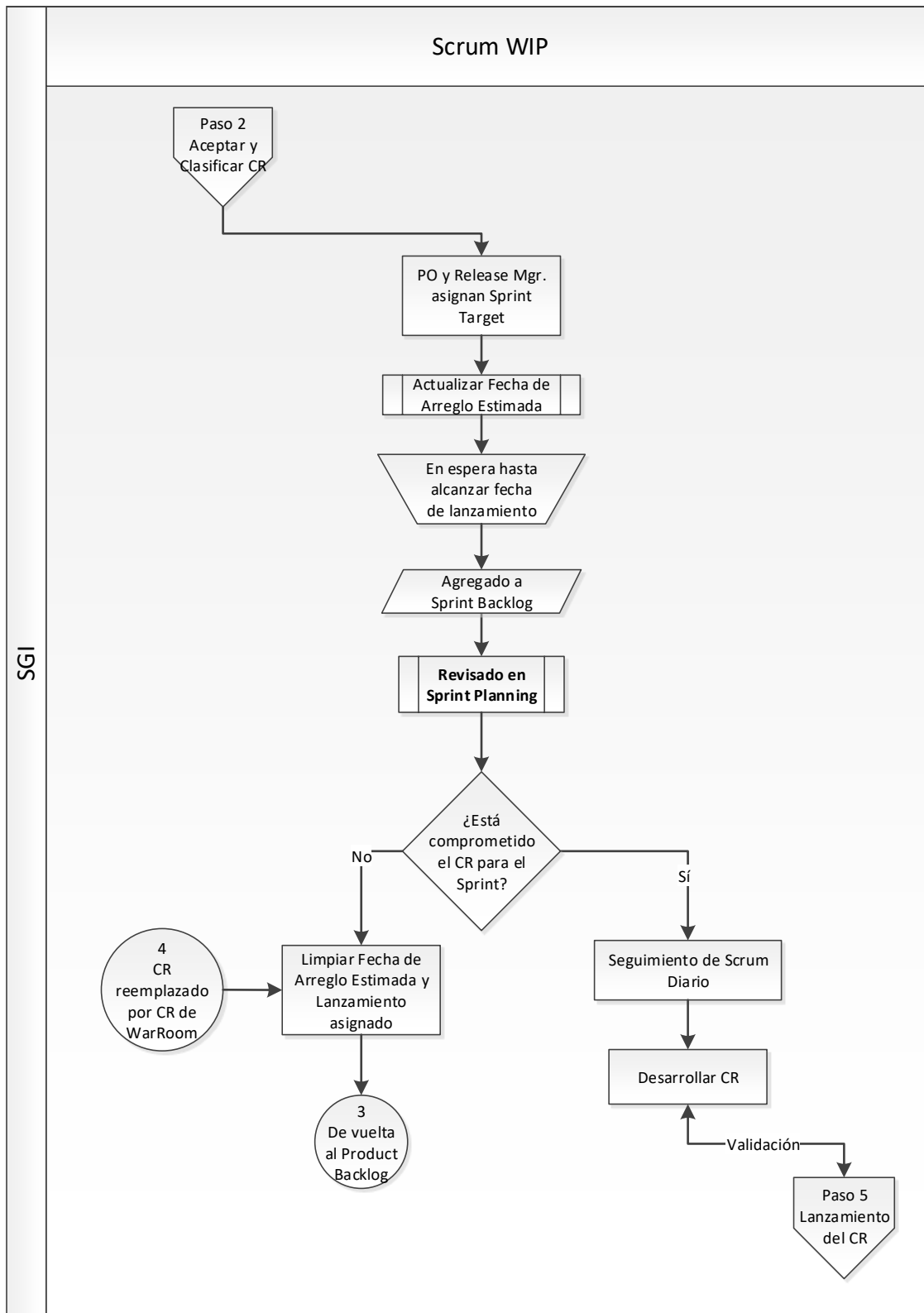
Indica que se deben seguir los requerimientos para generar un CR y asignarlo a la organización de SGI, para el desarrollo o modificación de una prueba de calidad. Para evitar un futuro rechazo el CR debe estar debidamente documentado y de manera entendible por el dueño de producto de SGI.

4.3.1.2 Aceptar y clasificar CR



Antes de que el CR pase a formar parte del *Product Backlog*, el dueño del producto del lado de SGI debe asegurarse que el CR enviado por el ingeniero de producto cumple con los requisitos según la categoría del CR. También valida si la solicitud de cambio es debida a un incidente en fábrica, lo que llevaría el CR a que se discuta en WarRoom, y de confirmarse que está deteniendo una orden en línea de producción, el CR entra directo al Sprint actual con carácter prioritario.

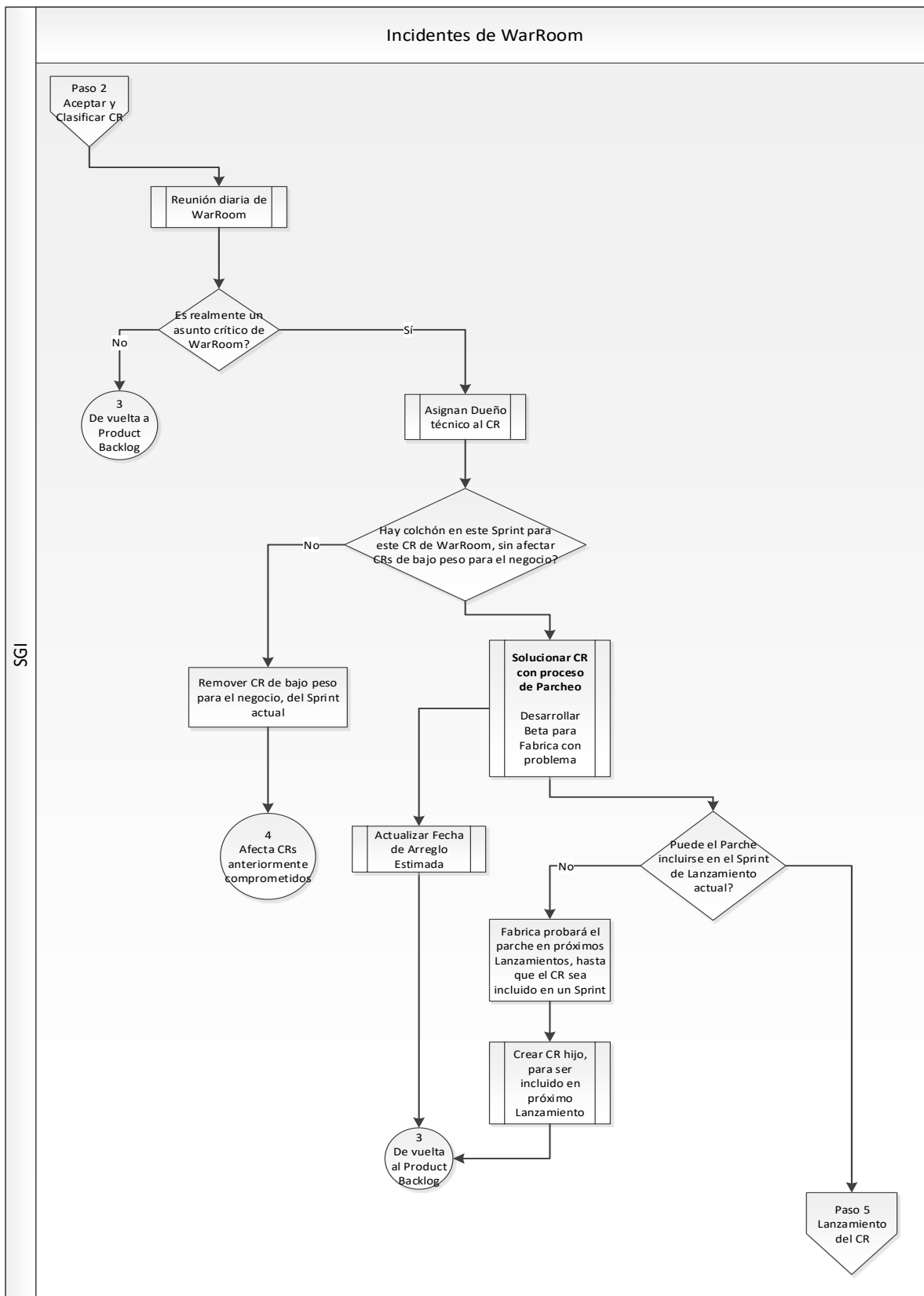
4.3.1.3 Scrum trabajo en proceso (WIP)



El CR que no haya sido clasificado como incidente, pasa a la cola de producto (*Product Backlog*) donde, con base en el valor para el negocio, el dueño de producto y el jefe de lanzamientos lo asigna a un próximo Sprint.

Una vez empieza el Sprint al que se asignó el CR, se vuelve a validar si el equipo de desarrollo se compromete a alcanzarlo, esto se conoce como *Sprint Planning* o, en español, planificación del ciclo. De no comprometerse a desarrollar el CR para el Sprint por comenzar, se le devuelve al *Product Backlog*, de lo contrario es aceptado y se empieza a trabajar en su desarrollo y validación, con la expectativa de que entre en el próximo lanzamiento.

4.3.1.4 Incidentes de WarRoom

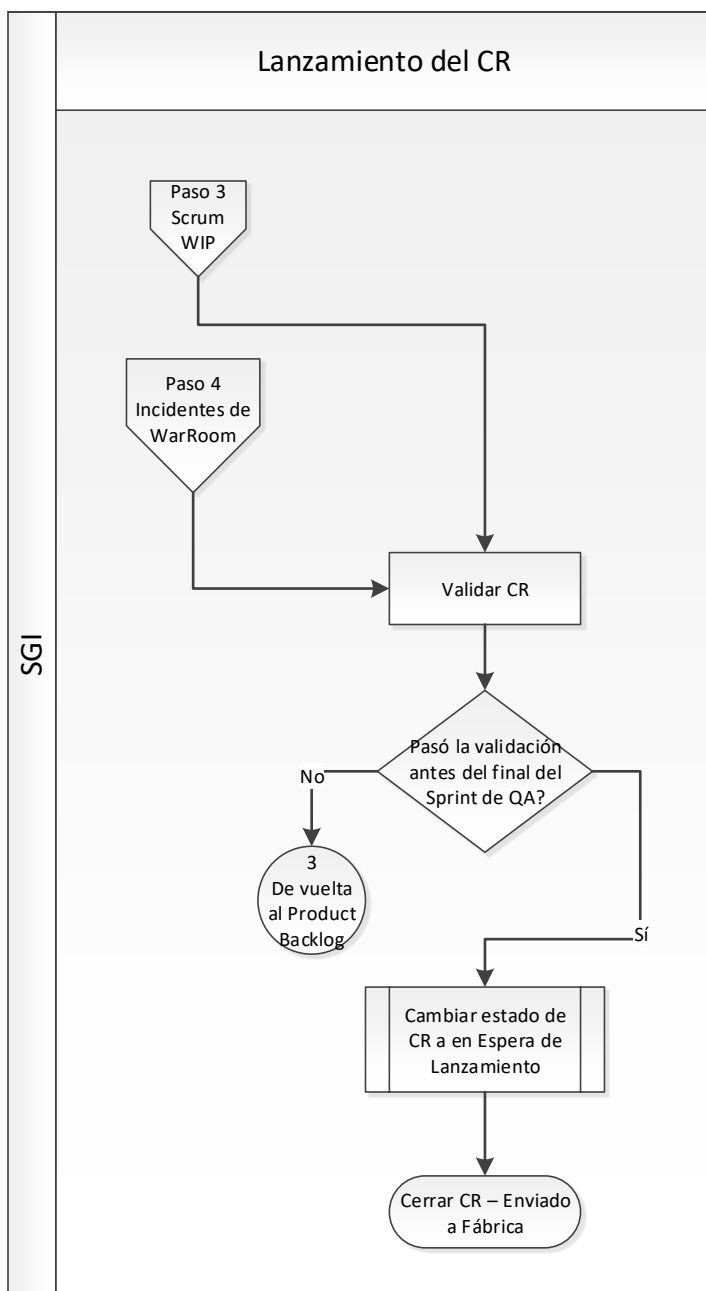


Los incidentes es algo que no se considera en Scrum, pues se los ve como defectos, que serán arreglados en el próximo Sprint. El proceso XP, también ágil, trabaja los defectos con algo más de prioridad, pues da énfasis en la calidad y genera un análisis de causa raíz para evitar que el problema surja de nuevo. Esto último se asemeja a la parte de manejo de problemas que describe ITIL en la fase de operaciones, donde también se maneja la parte de incidentes.

Si el Sprint no tiene colchón de recursos para recibir un incidente de WarRoom, entonces el Scrum Master debe sacar un CR con poco valor para el negocio pero que tenga el suficiente peso de trabajo para poder recibir y priorizar el incidente.

La urgencia que hay en una fábrica donde una o más órdenes están detenidas debido a un error en las pruebas de calidad, genera la necesidad de implementar parches, siendo la misma fábrica que reportó el incidente la que se encargará de validar si la solución propuesta es efectiva; para el siguiente Sprint el parche se incluye dentro del lanzamiento oficial para que esté disponible para las demás fábricas a nivel global. En casos donde la solución del incidente calce con el lanzamiento del Sprint, se puede agregar el parche al mismo lanzamiento global, pero para esto las fechas deben ser muy exactas.

4.3.1.5 Lanzamiento del CR



El CR pasa a la parte de validación, donde se da una comunicación constante entre equipo de validación y de desarrollo. Las pruebas de validación generan defectos con base en los errores encontrados con la implementación del CR; en caso de no poder solucionar todos los defectos antes a la fecha de lanzamiento, el CR no será incluido en el paquete y volverá al Product Backlog.

4.3.2 Roles y responsabilidades

SGI necesita que cada CR les llegue completo con todos los datos requeridos, para evitar entrar en un ciclo repetitivo de solicitud de información.

4.3.2.1 Responsabilidades del Ingeniero Global del Producto (GPE):

- El GPE encontrará la categoría del CR en los apéndices del OLA, llamada Tipos de Solicitud de Cambio, que coincide con la solicitud que se enviará a SGI, y seguirá la lista correspondiente de requisitos.
 - En caso de no encontrar una categoría adecuada, se aplicará la categoría general, seguida de ponerse en contacto el SGI Product Owner correspondiente.
- En caso de que GPE desee priorizar un CR específico, el Product Owner de SGI correspondiente deberá ser contactado.

4.3.2.2 Responsabilidad del SGI Product Owner (PO):

- El PO mantendrá actualizada la fecha estimada de arreglo (EFD, por sus siglas en inglés) y comunicará al remitente en caso de cambios.
- EL PO validará que el CR cumple con la plantilla de categoría correspondiente.
- El PO establecerá una reunión entre GPE y el Jefe Técnico correspondiente de SGI, cuando sea necesario para aclarar detalles técnicos.
- El PO dará seguimiento con el Embajador de Fábrica correspondiente, para cuando el CR haya sido clasificado como Incidente de WarRoom.

4.3.3 Métricas del OLA

Métricas e Indicadores Clave de Desempeño (KPI, por sus siglas en inglés), son medidas que cubren el propósito de este acuerdo y están destinadas a llevar un control del desempeño y el cumplimiento del OLA.

- Medir mejoras históricas en el tiempo para resolver CR's asignados a SGI. La resolución se da por los CR's clasificados como Completados o en Espera de Lanzamiento. La intención es disminuir el tiempo de resolución de los CR's.
- Medir la precisión histórica entre EFD y Fecha Completado de los CR's. La intención es buscar disminuir la brecha entre EFD y Fecha de Completado.
- Medir la cantidad histórica de CR's cerrados como Cerrado Sin Cambios (CNC, por sus siglas en inglés). Un CR cerrado como CNC, es un CR que tomó el espacio de otro CR más beneficioso en el Sprint.
- Medir la cantidad histórica de CR's por gravedad. Velando que no todo CR sea clasificado como Crítico, con el único propósito de priorizarlo.

Capítulo 5. Implementación de inteligencia de negocios

Los capítulos anteriores trataron temas de aclaración de la teoría de la cual trata el proyecto, investigación de posibles semejanzas y diseño de la solución para el negocio involucrando ITIL sobre Scrum. En este capítulo se expondrá el detalle de los pasos técnicos realizados para alcanzar la solución propuesta de Inteligencia de Negocios.

La implementación de BI orientado a DW 2.0, se subdivide en cuatro etapas:

Tabla 3 - Plan de acción para implementar BI

Etapas	Acción
Acceso a fuente de datos	<ul style="list-style-type: none"> Desarrollar la consulta contra la base de datos de producción.
Etapas Interactivas	<ul style="list-style-type: none"> Generar la tabla de los CR's semejante a la de QUIX. Actualizar dicha tabla todos los días con lo que ha entrado nuevo o ha cambiado en el último día.
Etapas Integradas	<ul style="list-style-type: none"> Estructurar las dimensiones necesarias y la tabla de hechos, que permitan la creación de los gráficos y reportes citados en los requerimientos. Insertar los datos de los sprints que estén corriendo, una vez al día, después de la reunión de Scrum de Scrum que sucede todos los días entre semana a la 1pm y termina a las 2pm (GMT -6). Los datos insertados cada día tendrán tres indicadores: el id del Sprint, el día del Sprint (entre el día 1 y 21) y el id del CR. El cubo debe procesarse todos los días en SSAS, después de que se haya agregado el día del Sprint al cubo relacional en SSMS.
Etapas Archivadas	<ul style="list-style-type: none"> Guardar en una base de datos semejante a la integrada, solo los registros de los días 21, que es el último CR para cada Sprint.

Cada etapa contiene un detallado desarrollo técnico y se procura dar una explicación a cada uno de sus pasos. Aun así, hay parte del código que para fomentar un entendimiento más directo, se excluyó de este capítulo y se incluyó en la sección de *Apéndices Ap 6, Ap 7, Ap 8 y Ap 9*.

5.1 Acceso a fuente de datos

Siguiendo el diagrama de SIPOC, donde la entrada de trabajo la hace el equipo de Ingenieros de Producto del departamento de Cadena de Suministro, lo que ingresa son solicitudes de cambio. La entrada de tales solicitudes se realiza por medio de una herramienta que internamente se llama QUIX pero comercialmente es conocida como *Serena Business Management*, actualmente la compañía MicroFocus es la dueña de dicha herramienta.

QUIX es manejado por un equipo fuera de SGI, además la base de datos del producto es manejada en Oracle, mientras que la solución a implementar del lado de SGI es en SQL Server.

La asignación de trabajo en Scrum, tanto el Product Backlog, como también el Sprint Backlog, vienen de la herramienta de QUIX; también la parte de análisis de causa raíz y la de administración de cambios para cada sub-equipo dentro de SGI, es generada mediante los datos que se extraen de la base de datos de QUIX en Oracle.

Los datos son de los pilares principales en una solución de BI. En este caso, esta parte se dividió en dos etapas: la primera consiste en buscar acceso a la herramienta y autorización a cada uno de los proyectos de los que se quiera extraer datos, la segunda es generar un servidor vinculado en SQL Server para poder extraer los datos. Una descripción detallada de cómo se logró esta etapa se puede encontrar en el *Apéndice Ap 10*.

5.2 Etapa Interactiva

La base de datos de QUIX es sumamente importante para la empresa, por lo que se debe procurar para las copias a la etapa interactiva, que las lecturas sean rápidas y eficientes, afectando en lo menos posible el desempeño de la base de datos de producción.

Lo más importante es copiar del servidor remoto, por medio de la vinculación que se hizo en la etapa anterior, la tabla que mantiene el detalle de cada CR (BO_QX_CR, nombre de tabla a copiar), aunque siguen existiendo muchas otras tablas relacionadas pero cuyo valor es útil para realizar minería de datos y están fuera del alcance de este proyecto (tablas de archivos adjuntos que tiene cada CR o la tabla de notas, donde un CR puede tener muchas notas relacionadas).

El nombre de la base de datos para esta etapa interactiva, donde se guardarán los datos que se sincronicen del servidor vinculado, es QUIX_SA (Staging Area, en inglés) y se sincronizarán 227 columnas, cada 15 minutos, de una lista de proyectos que se mantiene en otra tabla llamada ProjectsToSync. Para mayor información sobre la implementación técnica de esta etapa interactiva, consultar el *Apéndice Ap 11*.

5.3 Etapa Integrada

Lo que se esté trabajando en su momento es lo que compone la etapa integrada. En este caso son los CR's que se han aceptado dentro del Sprint de Scrum. Para la conformación de esta etapa, se estructuraron las dimensiones necesarias y la tabla de hechos correspondiente a partir del análisis ejecutado en la etapa anterior y se desarrolló una base de datos de Data Warehouse llamada SPRINT_DW.

La composición de esta etapa consta de siete dimensiones, que incluyen una dimensión de fecha y la respectiva tabla de hechos. Se utilizó la topología de estrella para su diseño. Esta etapa lleva un registro de todos los CR para cada uno de los 21 días del Sprint. La sincronización entre esta etapa integrada y la etapa anterior (etapa interactiva) se da una vez al día, mientras que la etapa interactiva con la base de datos de producción es cada 15 minutos.

El modelo utilizado para la base de datos multidimensional llamada SPRINT_DW fue el de estrella. Además, el nombre de las dimensiones se hace en inglés, debido a que los datos que se extraen de la herramienta de gestión de negocios también están en inglés.

A continuación, se procede a brindar una breve descripción de cada una de las dimensiones:

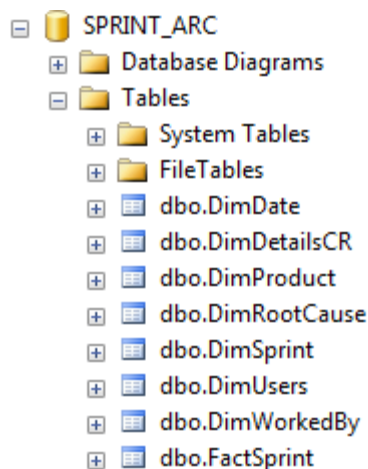


Ilustración 12 - Tablas SPRINT_DW

Tabla 4 - Dimensiones etapa integrada

Dimensión	Descripción
Date (Fecha)	Permite agrupar por las diferentes fechas para determinar aspectos asociados a fecha de creación, última vez que se modificó, fecha de cierre.
DetailsCR (Detalles de CR)	Contiene la descripción de los detalles de cada CR.
Product (Producto)	Permite hacer análisis referente a los productos que más CR's generan.
RootCause (Causa Raíz)	Permite categorizar las distintas raíces causas de fallos y demás.
Sprint (Ciclo)	Administra el Sprint que se encuentra actualmente en ejecución.
Users (Usuarios)	Contiene el correo de cada uno de los usuarios que interactúan con el CR, el dueño actual, el solicitante, el encargado de pruebas, etc.
WorkedBy (Trabajado por)	Permite definir quién lo trabajo categorizando en grupos y subgrupos.

El proceso de creación del paquete de SSIS para extracción, transformación y carga de los datos en la tabla de hechos y en cada una de las dimensiones, se detalla en el *Apéndice Ap 12*.

5.4 Etapa Archivada

Esta etapa genera una base de datos de Data Warehouse denominada SPRINT_ARC, la estructura tanto de las dimensiones como de la tabla de hechos es exactamente la misma que se desarrolló para el DW de la etapa integrada, con la diferencia que está no guarda una copia por cada día del Sprint, en cambio solo mantiene el día 21 que es el último día y solo actualiza los CR's que sigan en el Sprint al que pertenecían en un inicio y no se hayan terminado.

La razón principal de pasar la base de datos del motor de *SQL Server* a *Analysis Services* y a como se indica en la página web de documentación de Modelado Multidimensional de Microsoft (Microsoft, 2017) "...es lograr el rendimiento rápido de consultas en los datos empresariales...". La composición de un modelo multidimensional se da por medio de cubos y dimensiones que permiten consultas complejas, algo semejante a una tabla pivote.

La vista del cubo, difiere no tanto en los datos pero sí en aspecto. Además, el cubo incluye ciertas mediciones que se hacen con base en la tabla de hechos.

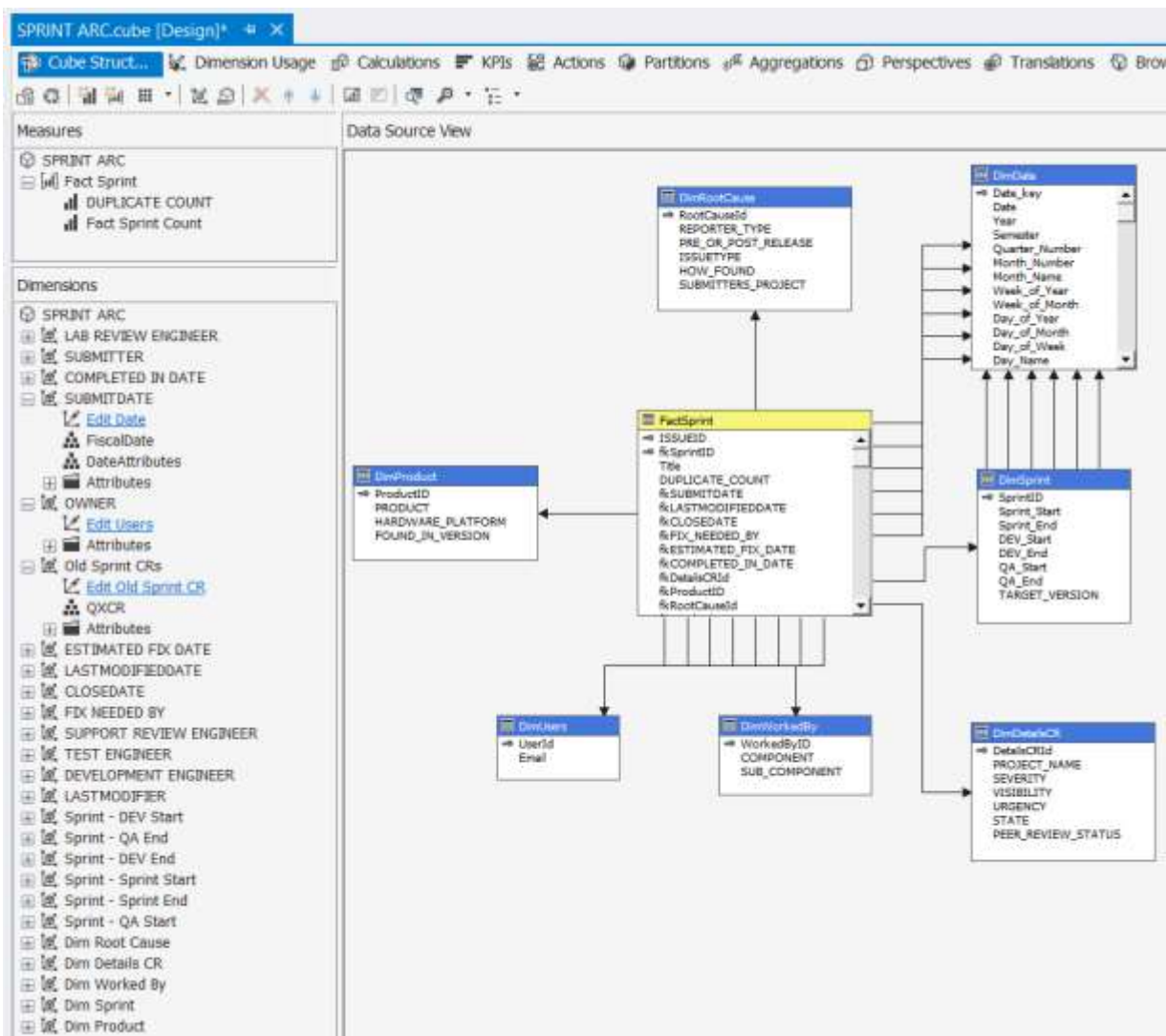


Ilustración 13 - Cubo SPRINT_ARC

El cubo desarrollado se hizo siguiendo un modelo estrella; cada llave foránea dentro de la tabla de hechos se convierte en una dimensión; en este caso hay varias llaves foráneas apuntando a la dimensión de fechas y a la de usuarios; esto permite ahondar más en el valor por buscar, como las fechas: puede ser año, mes, día, semestre, año fiscal, etc.

Para un mayor detalle de cómo se alcanzó esta etapa archivada, consultar el *Apéndice Ap 13*.

Capítulo 6. Implementación de reportes

Los principales reportes por desarrollar, son los indicados en la sección *Métricas del OLA*, que ya fueron acordados como una manera de control sobre la calidad del servicio. Otros reportes creados fueron para análisis más individuales de los equipos, tanto con el uso de Tableau como de Excel.

6.1 Reporte Métricas OLA

El propósito de las métricas en un OLA es que sean vistas constantemente por los jefes de los departamentos involucrados en el acuerdo. Para lograr esto se utilizó una implementación de Tableau Server, donde se publicaron los reportes de métricas desarrollados con Tableau Desktop.

La primera acción en una herramienta de reportes como Tableau es indicar de dónde se van a tomar los datos. En este caso es de QUIX_SA.



Ilustración 14 - Tableau conexión a QUIX_SA

6.1.1 Métrica OLA CR's completados

El propósito de esta métrica es medir los CR's completados en cada mes, por los últimos 13 meses, mes actual más 12 anteriores, como también el promedio de cantidad de días que las solicitudes cerradas permanecieron abiertas desde un inicio.

Y por último el promedio de la fecha en donde se definió un compromiso de solución contra la meta de equipo de días para brindar una solución.

La consulta realizada a la base de datos QUIX_SA para poder sacar el reporte descrito se puede ver en el *Apéndice Ap 14*.

Después de tener la consulta, se procede a generar cada reporte por separado, en una hoja de trabajo.

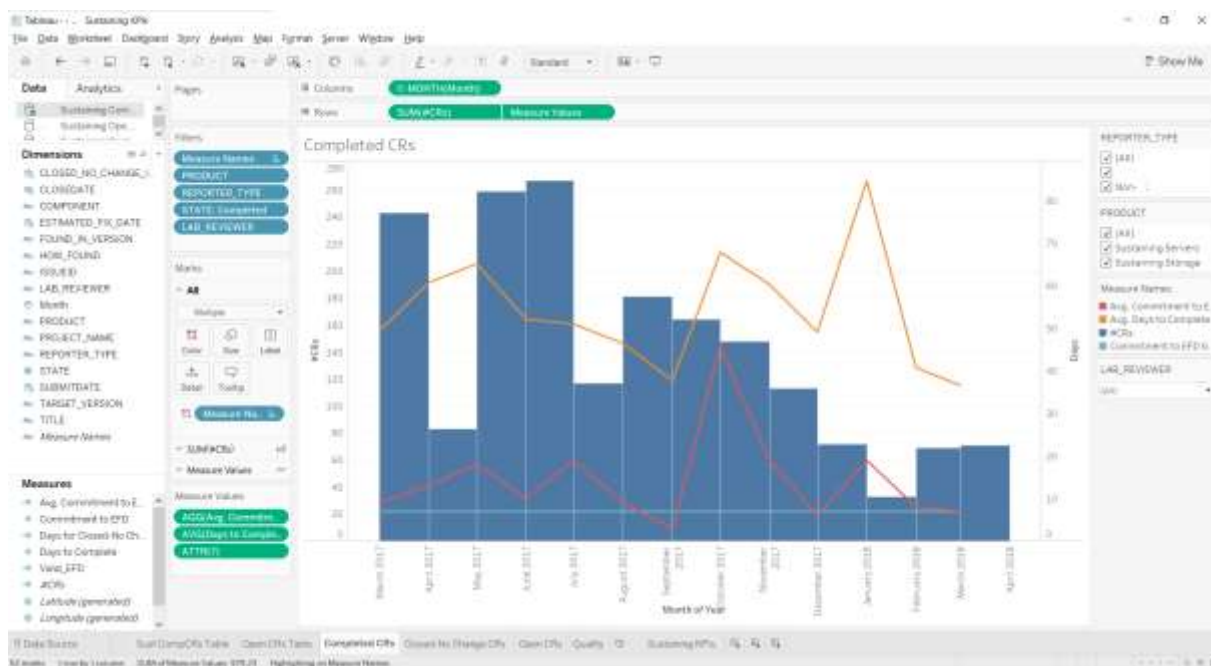


Ilustración 15 - Tableau hoja de trabajo CR's completados

Como se puede apreciar, el gráfico consta de dos ejes Y, el primero a la izquierda es la cantidad de solicitudes de cambio que se cerraron en el mes, que se representa con las barras. Mientras el de la derecha se mide en días y es el que corresponde a las líneas; la naranja es el promedio de días que se necesitó para cerrar los CR's, la roja es el promedio de días que se excedió la estimación de fecha de completado contra la fecha real en la que se completó el CR, y la celeste es la meta de no pasarse más de 7 días de lo que se comprometió en cerrar el CR.

6.1.2 Métrica OLA CR's cerrados sin cambios

Los CR's que se cierran sin cambios, son solicitudes que entraron al Sprint, pero no cumplieron los requisitos técnicos o de negocio para poder completarse. Estos CR's denotan falta de comunicación entre los ingenieros de producto del lado de la cadena de suministro y el dueño de producto del lado de SGI, también puede ser un fallo de

este último al no haber detectado el CR a tiempo, validando los requerimientos, antes de haber entrado al Sprint.

Este reporte se basa en la misma consulta que el de CR's completados.

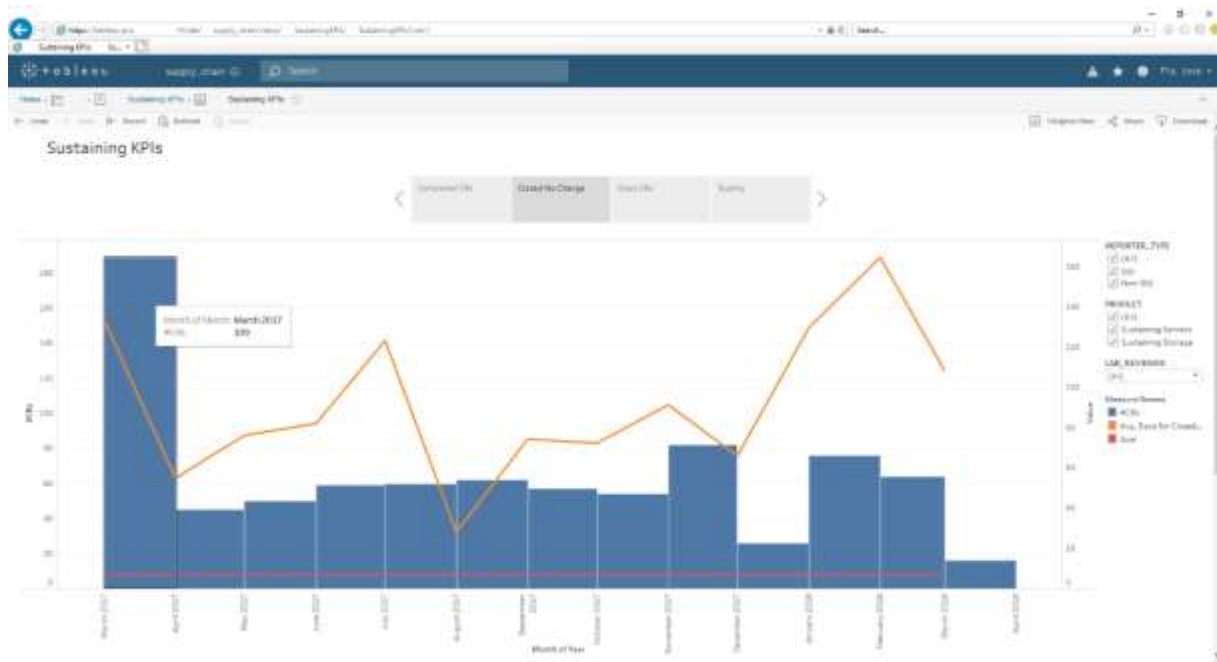


Ilustración 16 - Tableau Server CR's CNC

La imagen anterior es el reporte ya publicado en el servidor de Tableau. Las barras muestran los CR's que se cerraron sin cambios, mientras que la línea naranja indica el promedio de días que esos CR's estuvieron abiertos, la línea roja es la meta procurar detectar en menos de siete días un CR que no cumpla con los requerimientos.

6.1.3 Reporte seguimiento de CR's abiertos

El propósito de este reporte es evitar ir dejando viejos CR's abiertos, para permitir hacer una limpieza de antiguos CR's que con el tiempo hayan perdido valor para el negocio.

La consulta para este reporte, aunque es a la misma base de datos, difiere de los anteriores dos reportes, puesto esta trata de los CR's que siguen abiertos y no los que ya se hayan cerrado. El código SQL de la consulta se encuentra en el *Apéndice Ap 15*.

El reporte indica la cantidad de CR's abiertos, el promedio de días que han estado abiertos, y en los casos donde se haya aceptado el CR también el promedio de días posterior a la fecha estimada de solución.

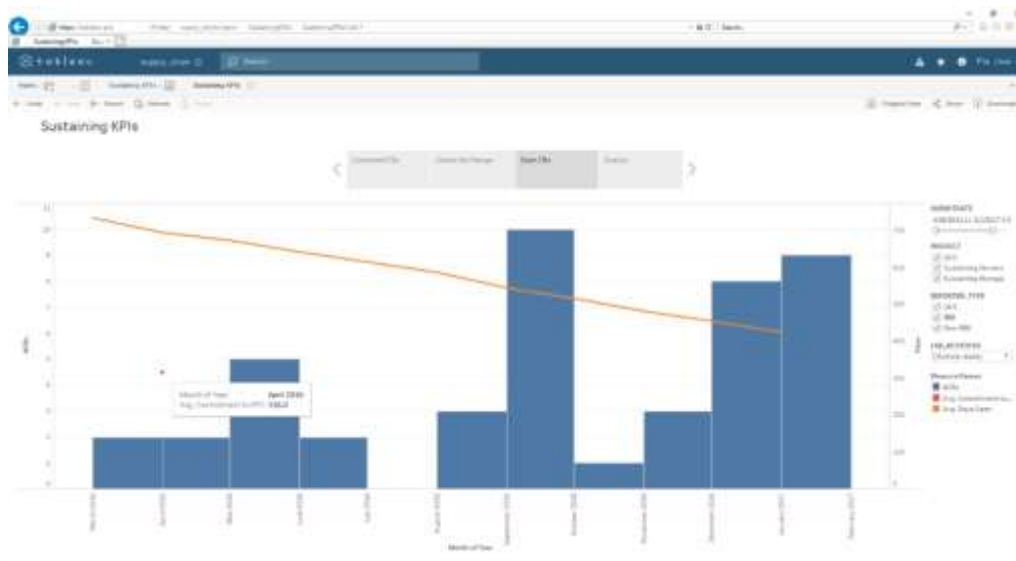


Ilustración 17 - Tableau Server CR's abiertos

6.1.4 Reporte de calidad de CR's entregados

El SIPOC, descrito en la sección de *Implementación de ITIL*, indica que el cliente es la fábrica. En el diagrama de flujo de proceso se detalla el manejo de incidentes, que según Scrum se dejan para un siguiente Sprint, pero al tratarse de un ambiente de manufactura su solución es prioritaria.

La intención es establecer una mejora continua, pero inicialmente el que representaría el rol de Jefe a Nivel de Servicio (Service Level Manager, en inglés) dentro de SGI, optó como meta inicial que no más del 50% de los CR's del Sprint sean defectos en validación o incidentes en fábrica.

El cálculo de la calidad se hace directamente en la hoja de trabajo en Tableau y no por medio de SQL.

Quality

Sustaining Quality

```
1-SUM(IF CONTAINS([ISSUETYPE], "Issue")
OR CONTAINS([ISSUETYPE], "Defect")
THEN 1
ELSE 0
END)/SUM([#CRs])
```

The calculation is v... Sheets Affe... Apply OK

All

Enter search text

- ABS
- ACOS
- AND
- ASCII
- ASIN
- ATAN
- ATAN2
- ATTR
- AVG

ABS(number)

Returns the absolute value of the given number.

Example: ABS(-7) = 7

Ilustración 18 - Medida de calidad de CRs en Tableau

El reporte de calidad, solo tiene un eje Y que representa el porcentaje de CR's del mes que fallaron en calidad ya sea en validación o en fábrica.

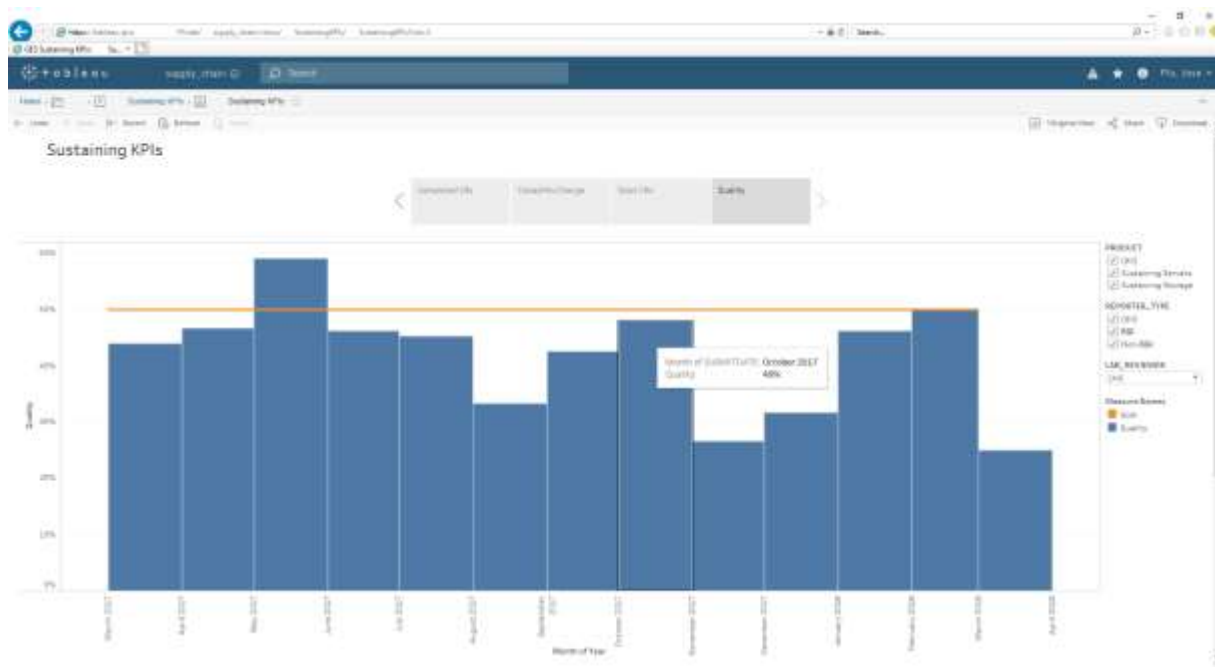


Ilustración 19 - Reporte de Calidad de CR's en Tableau Server

El porcentaje indicado solo indica qué tanto de ese mes fueron defectos en validación o incidentes en fábrica, el porcentaje restante es lo que se trabajó en solicitudes de cambio acordadas en la planificación del Sprint.

6.2 Reporte fuera del OLA

Cada equipo hace uso de reportes propios con el fin de mejorar de manera interna antes de aparecer mal en los reportes de OLA. Mientras más detalle tenga un reporte menor es el nicho de mercado que lo va a entender.

6.2.1 Reporte de CR's pendientes de Sprints previos

Este reporte es parecido al Reporte seguimiento de CR's abiertos, pero brinda más datos para poder identificar tanto el Sprint como el ingeniero asignado al CR pendiente. Para brindar más detalle, en este reporte la fuente de datos ya no es la tabla de QUIX_SA, sino el cubo de la etapa archivada.

Microsoft Analysis Services
SPRINT ARC (AS-MD_SPRINT_ARC)

Locate the cube to connect to:

Server:

Local cube file:

Enter information to log on to the server:

Use Windows Authentication (preferred)

Use a specific username and password:

Username:

Password:

Ilustración 20 - Conexión de Tableau Server a cubo SPRINT ARC

La presentación de la dimensión es algo diferente a cuando la conexión se hace directo a una base de datos relacional, en este caso no hay consultas opcionales de SQL.

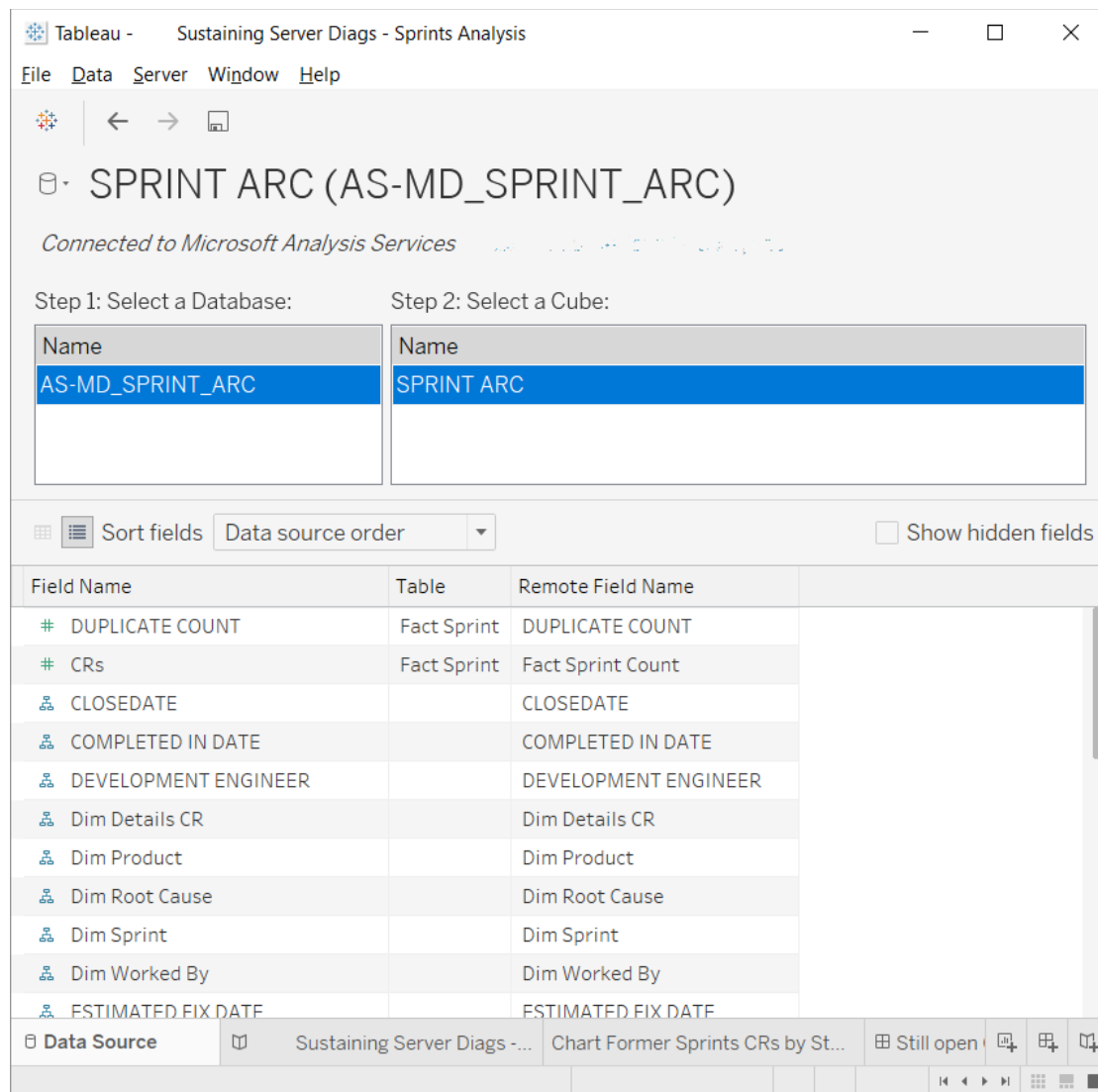


Ilustración 21 - Cubo SPRINT ARC en Tableau Desktop

La mayor ventaja que ofrece este reporte es el poder ahondar en los datos que lo componen, además de la previa agrupación de características de los hechos por medio de dimensiones y del aumento en la velocidad de respuesta en la visualización del reporte.

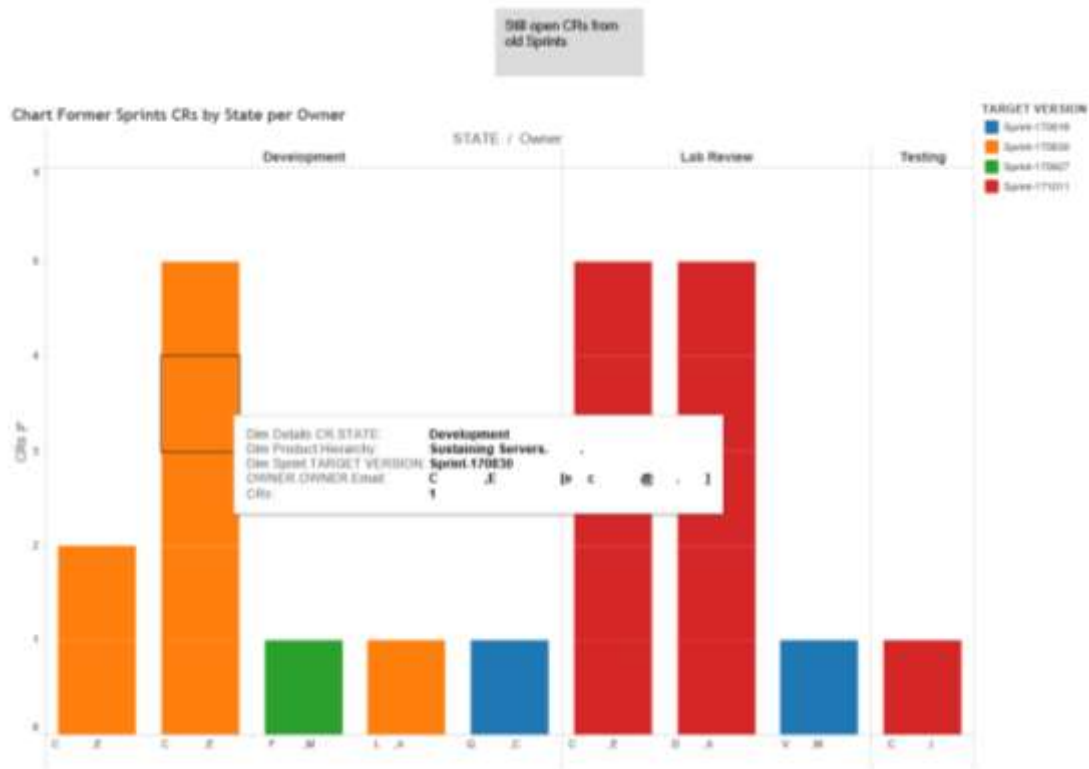


Ilustración 22 - CR's abiertos por encargado de Sprints pasados

Semejante a una tabla pivote, con las dimensiones se pueden agrupar las columnas tanto por encargado del CR, como también por el estado. De cada selección de columnas en el gráfico se pueden extraer fácilmente los datos que lo conforman.

Ver datos - Google Chrome

Secure | https://tableau-pro/vizql/t/..._supply_chain/w/SustainingServerDiags-SprintsAnalysis/

Resumen

Mostrando las primeras 4 filas.
[Descargar todas las filas como fila de texto](#)

STATE	PRODUCT	HARDWARE PLATFORM	FOUND IN VERSION	TARGET VERSION	Owner	CRs
Lab Review	Sustaining Servers	ARM	30.000.000	Sprint-171011	C ,E	1
Lab Review	Sustaining Servers	ARM	30.000.000	Sprint-171011	C ,E	1
Lab Review	Sustaining Servers	ARM	30.000.000	Sprint-171011	C ,E	2
Lab Review	Sustaining Servers	ARM	30.000.000	Sprint-171011	C ,E	1

Mostrando las primeras 4 filas.
[Descargar todas las filas como fila de texto](#)

Ilustración 23 - Detalle de datos de gráfico de cubo en Tableau

Capítulo 7. Análisis de resultados

Cada etapa de implementación: ITIL integrado a Scrum, inteligencia de negocios usando DW 2.0 y la de generación de reportes con Tableau; todas por separado trajeron beneficios palpables desde su desarrollo y hasta la puesta en marcha. Sin embargo, el enfoque de este trabajo es subrayar no solo como beneficio sino más como necesidad el contar con una implementación completa de inteligencia de negocios, el requisito de que las tres etapas expuestas en el *Capítulo 5*. trabajen en conjunto y sinergia con el fin común de mejorar el negocio.

Tabla 5 - Beneficios por implementación

Implementación	Productos	Beneficios
Definición de procesos	Implementación de ITIL por medio de OLA, respetando metodología ágil de Scrum usada en el desarrollo de pruebas de calidad	<ul style="list-style-type: none"> Entendimiento de actores involucrados. Entendimiento de requerimientos para iniciar el proceso. Entendimiento de tiempos de entrega, calendarización y asignación de trabajo. Indicación de dueños del proceso, para revisiones de mejora y escalaciones.
Almacenamiento de datos que conforman el proceso	<ul style="list-style-type: none"> Acceso a los datos Recolección de datos Base de datos multidimensional 	<ul style="list-style-type: none"> Reportes por Excel o personalizados en otra herramienta que no son solo los generados por QUIX. Mayor capacidad de análisis de los datos. Más rapidez en la extracción de datos de CR's.
Reportes dinámicos	<ul style="list-style-type: none"> Reportes en Tableau 	<ul style="list-style-type: none"> Muchos usuarios con capacidad de revisar un solo reporte, sin necesidad de compartir el mismo archivo de Excel. Seguimiento de métricas, que permite exigir una mejora continua.

Los beneficios de cada etapa son varios, pero es el conjunto lo que brinda una solución efectiva para el negocio. Scrum ayudó a dejar de ser un “apaga incendios”, pero al ser una metodología de primera capa, seguía vendiendo a SGI como una caja negra y estación de bomberos, donde no se sabía quién era el causante del incendio.

El proponer, conforme a un modelo SIPOC, el operar un servicio de forma constante y con una mejora continua de la calidad y no solo del producto brindado; ayuda a estandarizar procesos y entenderse entre organizaciones.

Un acuerdo a nivel operacional sin un seguimiento de métricas, lo convierte en un interesante documento de referencia que será guardado en alguna gaveta. Las métricas habilitan el accionar oportuno, basado en datos.

Un almacenamiento de datos sin un propósito definido, se convierte en un gasto de recursos, sin un uso constante, solo para fines de investigación.

Un conjunto de reportes que no sean exigidos mes a mes por gerencia, se convierten en una base para decisiones que se tomaron una sola vez, para atacar un solo problema; este problema puede provocar una solicitud del mismo reporte una segunda vez solo para confirmar que la solución aplicada haya sido efectiva. Pero una correcta implementación de ITIL busca una mejora continua del servicio y para saber qué mejorar se requiere una revisión constante de las métricas, esto logrará una madurez en los procesos y un mejor servicio, es por esto la necesidad de reportes que tengan una revisión constante.

ITIL, aunque parezca poco práctico en un inicio, brinda una visión clara de los procesos y logra evitar que las organizaciones trabajen como silos individuales. Pero el seguimiento de un servicio definido requiere de una revisión constante y para ello los datos y los reportes son vitales, de otra forma la parte de mejora continua sería muy subjetiva.

Capítulo 8. Conclusiones y recomendaciones

8.1 Conclusiones

Analizar la Cadena de Suministro como un todo es algo complejo y hasta abrumador, este proyecto se enfocó en el departamento de desarrollo de pruebas de calidad y se logró un acuerdo con el departamento global de ingenieros de producto. El acuerdo requirió de un seguimiento de varias métricas, para lo que fue necesaria la implementación de Inteligencia de Negocios y un sistema de reportes que permita una auditoría rutinaria, garantizando una mejora en el servicio.

- Antes de definir un flujo de trabajo involucrando a otros departamentos como el proveedor y el cliente, es necesario documentar y entender el proceso interno, citando a Edward Deming “Si uno no puede describir lo que está haciendo como un proceso, no sabe lo que está haciendo”. El OLA obligó a aclarar el flujo interno de trabajo de SGI, sirviendo para una mejor relación con los ingenieros de producto y disminuyendo las solicitudes cerradas sin cambios por falta de requerimientos. La integración de ITIL, ayudó a formalizar la documentación de procesos que no calzaban exactamente con la metodología de Scrum, como la parte de manejo de incidentes y gestión de la validación.
- La integración de la herramienta de gestión de negocios como fuente de datos a un ambiente de inteligencia de negocios usando SQL Server, junto con sus funciones de SSIS y SSAS, requirió un gran esfuerzo inicial en el proceso de acceso y recolección de datos. Esto obligó a una interacción con el departamento global de bases de datos así como con el encargado de la herramienta QUIX. En la etapa interactiva, se dificultó encontrar el peso de las solicitudes para el equipo de Scrum puesto que el indicador en los CR's corresponde a la urgencia para el negocio y no la complejidad de desarrollar cada CR. Por último, la etapa archivada generó un alto beneficio en el seguimiento de las solicitudes por Sprint, lo que ayudó en las métricas del OLA y una mejor relación con los ingenieros de producto, para evitar las ‘escalaciones’.

- Los reportes se desarrollaron en su mayoría en Tableau fueron valorados como ‘muy amigables’ por los usuarios. Sin embargo, los elevados costos de licenciamiento, dificultó que todos los usuarios tuvieran acceso a la herramienta; ello disminuyó el impacto de los reportes sobre la mejora del servicio.

8.1 Recomendaciones

El negocio es lo que debe ir primero, no la urgencia de implementar metodologías que estén en apogeo en el mercado. Buscar lo mejor para el negocio no necesariamente tienen que ser las últimas tendencias, el enfoque debe ser la mejora y no la implementación estricta de alguna metodología.

- Así como la iniciativa de SAFe vino desde un nivel ejecutivo y aunque lo que se implementó fue más una personalización de Scrum, lo mismo debió haber sucedido con ITIL. Hay roles dentro de ITIL como el Jefe a Nivel de Servicio (SLM, por sus siglas en inglés), que deben ser definidos en un nivel de autoridad en la organización, en vez de ser sugeridos desde un rango de ingeniero.
- La implementación de DW 2.0 requiere de arduo trabajo inicial y también de un mantenimiento constante, lo que conlleva a la necesidad de analizar primero la complejidad técnica de las métricas acordadas en el OLA. Las métricas son lo que debe definir la implementación de BI y no implementar BI y después buscar qué métricas se pueden obtener.
- El beneficio de los reportes depende directamente de que llegue a los individuos que puedan tomar decisiones con fundamento en los resultados de cada reporte. En la elección de la herramienta para reportes debe haber un balance entre facilidad de uso, alcance de usuarios, presentación de datos y los costos, tanto iniciales como de mantenimiento y por usuario.

Capítulo 9. Reflexiones finales

La mejora en la calidad de servicio entre los departamentos de la Cadena de Suministro, es algo que al final se verá reflejado en la mejora de la calidad del producto que recibe el cliente.

Centrarse en buscar qué medir o en realizar mejoras como las 5s no es lo que enseña Edward Deming: primero se deben entender bien los procesos para después poder medirlos; de lo contrario, muchas métricas podrían resultar innecesarias y hasta traer decisiones erróneas sobre el negocio.

La medición de datos debe ir enfocada en la mejora del equipo y no en un castigo de los individuos; utilizar los datos para la micro gerencia podría afectar el clima organizacional. Y en un ambiente donde cada actor tiene múltiples tareas es difícil juzgar a un individuo por una sola métrica.

Cambios de alta gerencia en el último período fiscal del 2017 en puestos relacionados al apoyo brindado al OLA, tanto por el lado de SGI como también en la organización de GPE, además de varias divisiones de la empresa, perjudicaron la continuidad de esta solución.

Posterior a los cambios en la alta gerencia, SGI decidió fortalecer el uso de Scrum por medio de una herramienta de Computer Associates (CA) llamada Rally. Usando API's de Rally, se migran los CR's del Product Backlog en QUIX a convertirse en User Stories en Rally. En esta herramienta ágil los casos de prueba para validación solo pueden asociarse a un único User Story aun cuando el caso de prueba esté abarcando varios User Stories; misma restricción al intentar asociar defectos encontrados en validación. Los casos de prueba, como los defectos y también los incidentes de WarRoom, se registran únicamente en Rally; por lo que se pierde control o seguimiento del trabajo por el lado de QUIX.

Rally es una herramienta de acceso en línea y se accede de forma individual por usuario, con un costo por cada uno. Esto evita que todos los ingenieros de producto, que no forman parte de SGI, puedan tener acceso al User Story asociado a su CR en QUIX. Esta situación evita un seguimiento del avance de trabajo en el CR por medio del solicitante, debido a que la sincronización de datos no es bilateral entre QUIX y Rally.

Al repasar los resultados de la solución de este trabajo, junto con las destituciones recientes, la división de la empresa, más la seguida implementación de Rally con un acceso restringido a los datos por usuarios ajenos a SGI, llevan al autor preguntarse ¿qué tanto SGI desea volver a la seguridad de una caja negra en el proceso de cadena de suministro?

Bibliografía

- Ali, M. M. (s.f.). *Get SAFe Now*. Amazon Digital Services LLC. doi:B01M1GI80M
- Araya Elizondo, I., & Muñoz Alcázar, J. (2016). *Construir un modelo de negocios que consolide las diferentes fuentes de datos, con el fin de mejorar la calidad de la información, actualmente utilizada por la organización WW TS Support Solutions Pricing en HPE para la toma de decisiones*. Tesis, Universidad Cenfotec, Maestría en Tecnologías de Bases de Datos, San José.
- AXELOS. (2011). ITIL Glossary of Terms English - Spanish (Latin America). (1). Recuperado el 20 de 4 de 2018, de [https://www.axelos.com/Corporate/media/Files/Glossaries/ITIL_2011_Glossary_ES-\(Latin-America\)-v1-0.pdf](https://www.axelos.com/Corporate/media/Files/Glossaries/ITIL_2011_Glossary_ES-(Latin-America)-v1-0.pdf)
- Barton, B., Schwaber, K., & Rawsthorne, D. (Enero de 2015). Reporting Scrum Project Progress to Executive Management through Metrics. Recuperado el 14 de Mayo de 2017, de http://www.danube.com/docs/Scrum_Metrics_for_Management.pdf
- Beasley, R. (2013). Agile Development & BI Methodology – "Bridging the Gap". En IBM (Ed.), *BusinessObjects User Conference*. ASUG SAP BusinessObjects. Recuperado el 14 de Mayo de 2017, de http://events.asug.com/2013BOUC/Trends%20&%20Strategy,%20Roadmaps%200&%20Product%20Direction/8_26_13/0707%20Agile%20Development%20and%20BI%20Methodology%20Bridging%20the%20Gap.pdf
- Beck, K., Beedle, M., Bennekum, A. v., Cockburn, A., Cunningham, W., Fowler, M., . . . Sutherland, J. (2001). *12 Principios Manifiesto Ágil*. Recuperado el 14 de Mayo de 2017, de Manifiesto Ágil: <http://agilemanifesto.org/iso/es/principles.html>
- Beck, K., Beedle, M., Bennekum, A. v., Cockburn, A., Cunningham, W., Fowler, M., . . . Thomas, D. (2001). *Manifiesto Ágil*. Recuperado el 14 de Mayo de 2017, de <http://agilemanifesto.org/iso/es/manifesto.html>

- Blanco, C., Lasheras, J., Valencia-García, R., Fernández-Medina, E., Toval, A., & Piattini, M. (Julio de 2008). *Ontologías de Seguridad: Revisión sistemática y comparativa*. (U. d.-L. Mancha, Ed.) Murcia, España.
- Byod, R. (21 de Noviembre de 2014). *9 Scrum Metrics to Keep Your Team on Track*. Recuperado el 14 de Mayo de 2017, de Pragmatic Marketing:
<http://pragmaticmarketing.com/resources/9-scrum-metrics-to-keep-your-team-on-track>
- Chaves, J. (2017). *Análisis y uso de técnicas de minería de datos para optimizar la distribución y abastecimiento del recurso hídrico en el cantón de Belén*. Tesis, Universidad Cenfotec, Maestría en Tecnología de Bases de Datos.
- Fowler, M. (13 de Diciembre de 2015). *New Methodology*. Recuperado el 14 de Mayo de 2017, de Martin Fowler website:
<https://www.martinfowler.com/articles/newMethodology.html>
- Fowler, M., & Highsmith, J. (Agosto de 2001). *The Agile Manifesto*. Recuperado el 14 de Mayo de 2017, de
http://dimoiv.uqac.ca/8INF851/web/part1/introduction/The_Agile_Manifesto.pdf
- Green, P. (2011). *Measuring the Impact of Scrum on Product Development at Adobe Systems*. (A. Systems, Ed.) Proceedings of the 44th Hawaii International Conference on System Sciences. Recuperado el 14 de Mayo de 2017, de
https://www.agilealliance.org/wp-content/uploads/2016/01/Adobe_HICSS44.pdf
- Inmon, B. (Abril de 2010). *Data Warehouse 2.0 Modeling and Metadata Strategies for Next Generation Architectures*. (L. Forest Rim Technology, Ed.) Recuperado el 26 de Agosto de 2017, de Embarcadero Technical Papers:
<https://www.embarcadero.com/images/dm/technical-papers/bill-inmon-data-warehousing-2-0-whitepaper.pdf>

- Inmon, W. (Octubre de 2009). *Data Warehousing 2.0 and SQL Server: Architecture and Vision*. Recuperado el 15 de Julio de 2017, de Microsoft White Papers:
<https://msdn.microsoft.com/en-us/library/ee730351.aspx>
- Integrity Global. (24 de Junio de 2016). *Unraveling the Mystery of HP/HPE Part Numbers*. Recuperado el 14 de Enero de 2018, de Integrity Global Solutions:
https://cdn3.bigcommerce.com/s-3jjekk/product_images/uploaded_images/hp-part-numbers.jpg?t=1466716914
- ISO. (15 de 12 de 2015). *Guidance on the relationship between ISO/IEC 20000-1:2011 and service management frameworks: ITIL*. Reporte Técnico, ISO/IEC.
- J. Gerald, S. (1 de Julio de 1992). Three Experts on Quality Management. (No. 92-02). (T. Q. Office, Ed.) Arlington, Virginia, Estados Unidos de America: Office of the Under Secretary of the Navy. Recuperado el 24 de Abril de 2017, de
<http://dtic.mil/dtic/tr/fulltext/u2/a256399.pdf>
- Kelleher, K. (20 de Setiembre de 2012). *HP and IBM: Two paths, one future*. Recuperado el 14 de Enero de 2017, de Fortune:
<http://fortune.com/2012/09/20/hp-and-ibm-two-paths-one-future/>
- Kim, D. (2013). *State of SCRUM Report 2013*. Recuperado el 14 de Mayo de 2017, de Scrum Alliance:
https://www.scrumalliance.org/scrum/media/ScrumAllianceMedia/Files%20and%20PDFs/State%20of%20Scrum/2013-State-of-Scrum-Report_062713_final.pdf
- Mahnic, V., & Zabkar, N. (25 de Enero de 2008). *Measurement repository for Scrum-based software development process*. (U. o. Ljubljana, Ed.) Recuperado el 24 de Abril de 2017, de Faculty of Computer and Information Science:
https://www.researchgate.net/profile/Viljan_Mahnic/publication/268326181_Measurement_repository_for_Scrum-based_software_development_process/links/54b640680cf2318f0f9a291a.pdf

- Marchesi, M. (2005). *The New XP*. (D. d. Universidad de Cagliari, Ed.) Recuperado el 2 de Julio de 2017, de https://www.academia.edu/2862372/The_new_XP
- Meier, J. (29 de Enero de 2012). *ITIL Stages, Processes, and Sub-Processes*. Recuperado el 16 de Agosto de 2017, de Microsoft Developer J.D. Meier's Blog: <https://blogs.msdn.microsoft.com/jmeier/2012/01/29/itil-stages-processes-and-sub-processes/>
- Microsoft. (27 de Octubre de 2014). *Microsoft SQL Server Data Tools - Business Intelligence for Visual Studio 2013*. Recuperado el 1 de Marzo de 2018, de Microsoft Download Center: <https://www.microsoft.com/en-us/download/details.aspx?id=42313>
- Microsoft. (1 de Marzo de 2017). *Modelado multidimensional (SSAS)*. Recuperado el 6 de Marzo de 2018, de Microsoft Docs: <https://docs.microsoft.com/es-es/sql/analysis-services/multidimensional-models/multidimensional-models-ssas>
- Microsoft Docs. (9 de Febrero de 2018). *Download SQL Server Data Tools (SSDT)*. Recuperado el 1 de Marzo de 2018, de Microsoft Docs: <https://docs.microsoft.com/en-us/sql/ssdt/download-sql-server-data-tools-ssdt>
- Microsoft. (s.f.). *Instalar SQL Server Data Tools*. Recuperado el 1 de Marzo de 2018, de MSDN Library: [https://msdn.microsoft.com/es-es/library/hh500335\(v=vs.103\).aspx](https://msdn.microsoft.com/es-es/library/hh500335(v=vs.103).aspx)
- Payton, T. (2014). *IMPLEMENTING SCRUM PART 5 of 5: SCRUM SUCCESS METRICS*. (AXOSOFT, Ed.) Recuperado el 14 de Mayo de 2017, de <http://www.scrumhub.com/wp-content/uploads/2014/10/scrumhub-implementing-scrum-part5.pdf>
- Plá Rojas, J. A. (2013). *Estudio de viabilidad de implementación de Inteligencia de Negocios en el sector público de Costa Rica*. Maestría en Administración de Empresas énfasis en Banca y Finanzas, Universidad Latina de Costa Rica.

- PMI India Team. (10 de Setiembre de 2015). Measuring ROI of Agile Transformation. Bangalore, Karnataka, India. Recuperado el 14 de Mayo de 2017, de <https://pmi.org.in/events/media/technical-paper/measuringr-b509462068e6cc5.pdf>
- ProactivaNET. (3 de Julio de 2017). *ITIL en 3 minutos*. Recuperado el 16 de Agosto de 2017, de ProactivaNET Blog: <https://www.proactivanet.com/blog/proactivanet/itil-en-3-minutos/>
- ProjectsAtWork, S. A. (Dirección). (2013). *The State of SCRUM 2013: Benchmarks and Guidelines, How do you compare?* [Película]. Recuperado el 14 de Mayo de 2017, de <https://fast.wistia.net/embed/iframe/lmzsqdbi30?popover=true>
- Prowareness Scrum Professional. (26 de Febrero de 2017). Agile Metrics. Recuperado el 14 de Mayo de 2017, de http://www.scrum.nl/wp-content/uploads/2017/04/whitepaper_agile_metrics.pdf
- Radigan, D. (s.f.). *The kanban methodology*. Recuperado el 02 de Julio de 2017, de Jira an Atlassian Company: <https://www.atlassian.com/agile/kanban>
- Scaled Agile Framework. (15 de Junio de 2017). *Poster*. Recuperado el 19 de Julio de 2017, de Scaled Agile Framework: <http://www.scaledagileframework.com/posters/>
- Scarborough, M. (19 de Junio de 2015). *Using SIPOC to Define ITIL® Processes*. Recuperado el 25 de Abril de 2018, de Global Knowledge: <https://www.globalknowledge.com/ca-en/resources/resource-library/white-paper/using-sipoc-to-define-itil-processes/>
- Schwaber, K., & Sutherland, J. (Julio de 2016). Guía de Scrum. (O. Scrum, Ed.) Recuperado el 14 de Mayo de 2017, de <http://www.scrumguides.org/docs/scrumguide/v2016/2016-Scrum-Guide-US.pdf>
- Scrum Alliance. (2013). *The State of SCRUM 2013: Benchmarks and Guidelines, How do you compare?* Recuperado el 14 de Mayo de 2017, de Scrum Alliance:

<https://www.scrumalliance.org/scrum/media/ScrumAllianceMedia/Files%20and%20PDFs/State%20of%20Scrum/StateOfScrumPresentation.pdf>

Scrum Alliance. (2016). State of SCRUM Report 2016. Recuperado el 14 de Mayo de 2017, de

https://www.scrumalliance.org/scrum/media/ScrumAllianceMedia/Files%20and%20PDFs/State%20of%20Scrum/StateOfScrum_2016_FINAL.pdf?aliId=233097731

Scrum Alliance Core Scrum. (13 de Diciembre de 2012). SCRUM una descripción. (S. Alliance, Ed.) Recuperado el 14 de Mayo de 2017, de

<https://www.scrumalliance.org/scrum/media/ScrumAllianceMedia/Files%20and%20PDFs/Why%20Scrum/Core%20Scrum%20Translations/Core-Scrum-Spanish.pdf>

Scrum Alliance. (s.f.). *How to measure Scrum success*. Recuperado el 14 de Mayo de 2017, de Scrum Alliance:

<https://www.scrumalliance.org/scrum/media/ScrumAllianceMedia/Files%20and%20PDFs/Articles/How-to-Measure-Scrum-Success.pdf>

Scrum Alliance. (s.f.). The Scrum framework in 30 seconds. Recuperado el 14 de Mayo de 2017, de

<https://www.scrumalliance.org/scrum/media/ScrumAllianceMedia/Files%20and%20PDFs/Why%20Scrum/ScrumAlliance-30SecondsFramework-HighRes.pdf>

SOGETI LAB. (28 de Diciembre de 2015). *Measuring performance in Scrum teams is an act of sabotage!* Recuperado el 14 de Mayo de 2017, de Top 10 Blogs of 2015: <http://labs.sogeti.com/measuring-performance-in-scrum-teams-is-an-act-of-sabotage/>

Srivastava, B. (16 de Enero de 2017). *Agile Methodology In Software Development And Drawbacks Of The Waterfall Model*. Recuperado el 14 de Enero de 2018, de C# Corner: <http://www.c-sharpcorner.com/article/agile-methodology-in-software-development-and-drawbacks-of-waterfall-model/>

- Stein, S., Rabeler, C., Blome, M., Vance, J., & Guyer, C. (9 de Febrero de 2018). *Previous releases of SQL Server Data Tools (SSDT and SSDT-BI)*. Recuperado el 1 de Marzo de 2018, de Microsoft Docs: <https://docs.microsoft.com/en-us/sql/ssdt/previous-releases-of-sql-server-data-tools-ssdt-and-ssdt-bi>
- The Standish Group International. (2013). *The CHAOS Manifesto 2013*. Recuperado el 14 de Mayo de 2017, de <http://dimoiv.uqac.ca/8INF851/web/part1/problematique/ChaosManifesto2013.pdf>
- Turban, E. (2011). *Decision Support and Business Intelligence Systems* (Vol. IX). USA: Content Technologies and Cram101.
- Unadkat, A. (11 de Marzo de 2014). *JIRA Agile: Scrum or Kanban?* Recuperado el 6 de Julio de 2017, de Compañía Valiantys: <http://blog.valiantys.com/en/jira-en/jira-agile-scrum-kanban>
- UpGuard. (10 de Junio de 2017). *ITIL Guide to DevOps*. Recuperado el 10 de Junio de 2017, de <https://www.upguard.com/ebooks/itil-guide-to-devops>
- Vanian, J. (27 de Agosto de 2015). *A bright spot for big tech companies*. Recuperado el 14 de Enero de 2018, de Fortune: <https://fortunedotcom.files.wordpress.com/2015/05/hewlett-packard.jpg>
- Womak, J. P., Jones, D. T., & Roos, D. (2007). *La máquina que cambio al mundo*. Free Press. doi:B001D1SRRS

Apéndices

Ap 1. Diagrama de Kimball DW 1.0

“El concepto que se mantiene al basar el DW en la extracción de datos directa de los DM's; es más dada en casos donde ya existen sistemas de información maduros. Y la reestructuración de los DM's sería casi inaudita, debido al alto costo y a la dependencia que herramientas o sistemas ya existentes tengan de estos. Aquí es evidente como se duplica el proceso ETL entre las fuentes de datos y los DM's, seguido de los DM's y el DW.” (Plá Rojas, 2013, pág. 27)

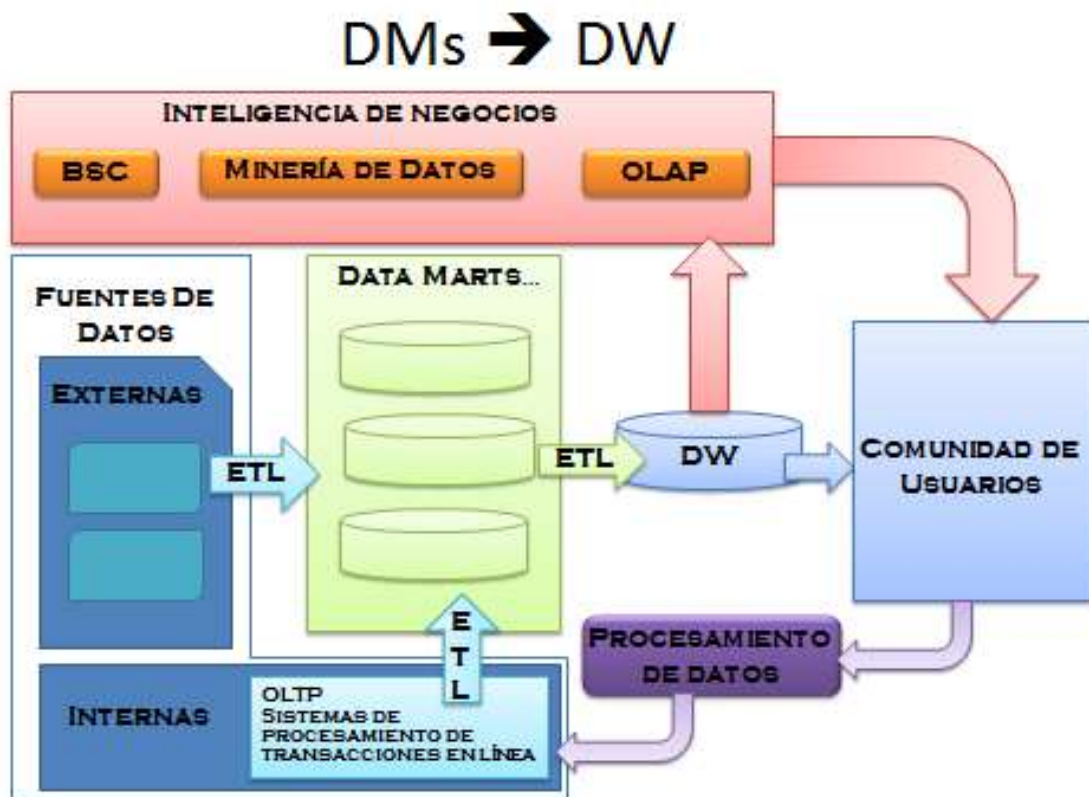


Ilustración 24 - Diagrama de Kimball DW 1.0

Ap 2. Diagrama de Inmon DW 1.0

“La mayor ventaja de esta implementación se basa en el ahorro que se obtiene al evitar un doble proceso de ETL; siendo el almacén de datos (DW) la fuente estandarizada y única de información de los mercados de datos (DM). “ (Plá Rojas, 2013, pág. 26)

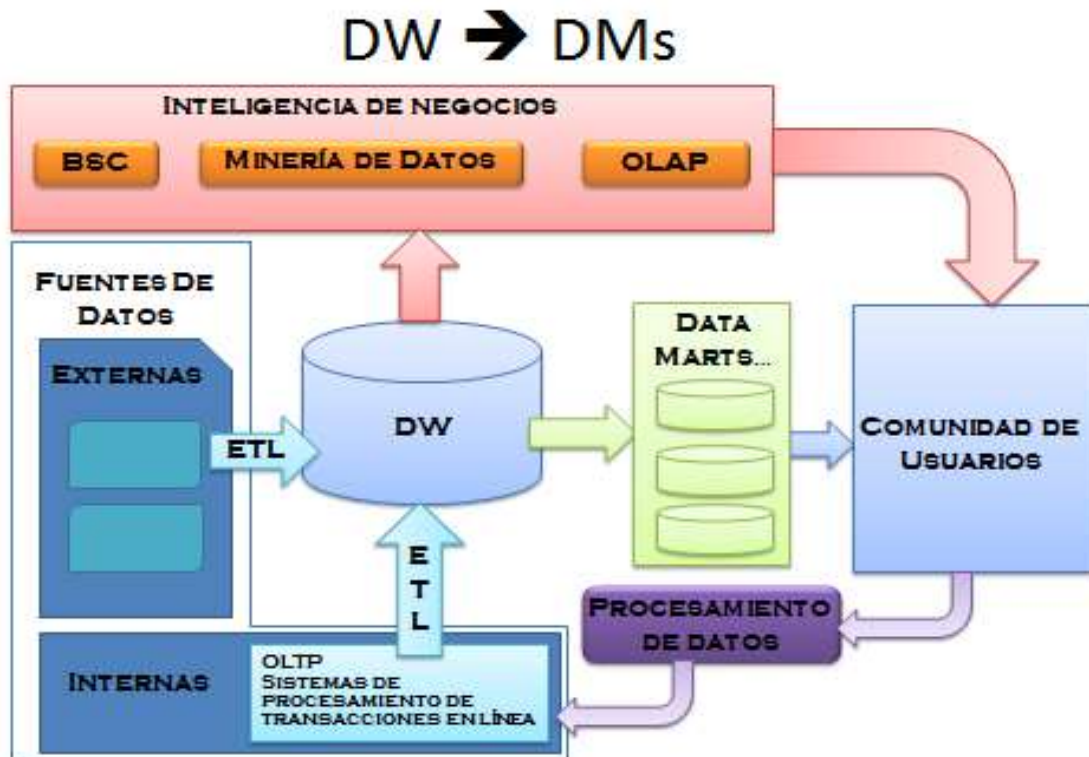
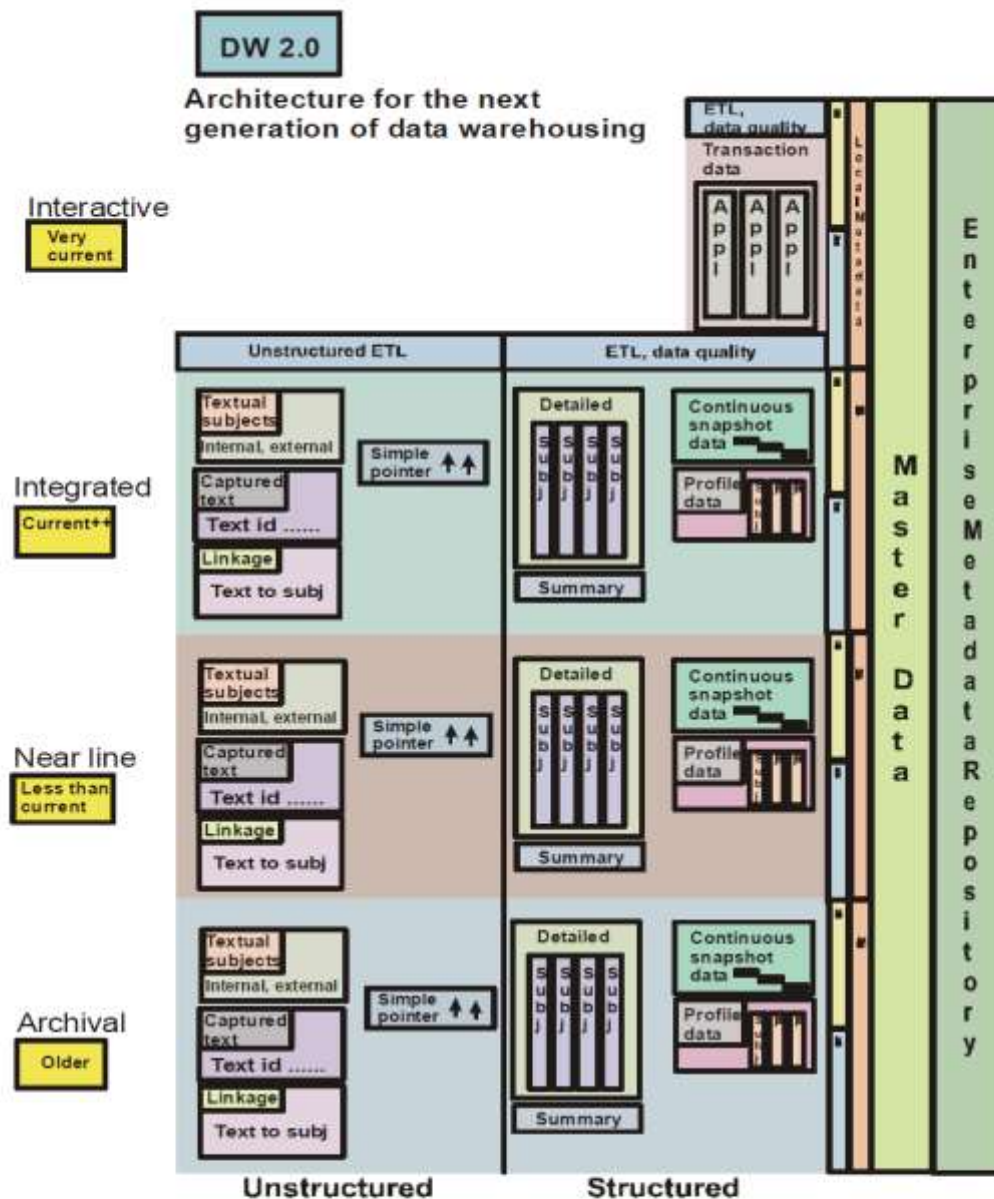


Ilustración 25 - Diagrama de Inmon DW 1.0

Ap 3. Diagrama de Inmon DW 2.0



Then there was DW 2.0, architecture for the next generation of data warehousing

Figure 2

Ilustración 26 - Diagrama de Inmon DW 2.0 (Inmon B. , 2010, pág. 2)

Ap 4. Diagrama completo SAFe 4.5

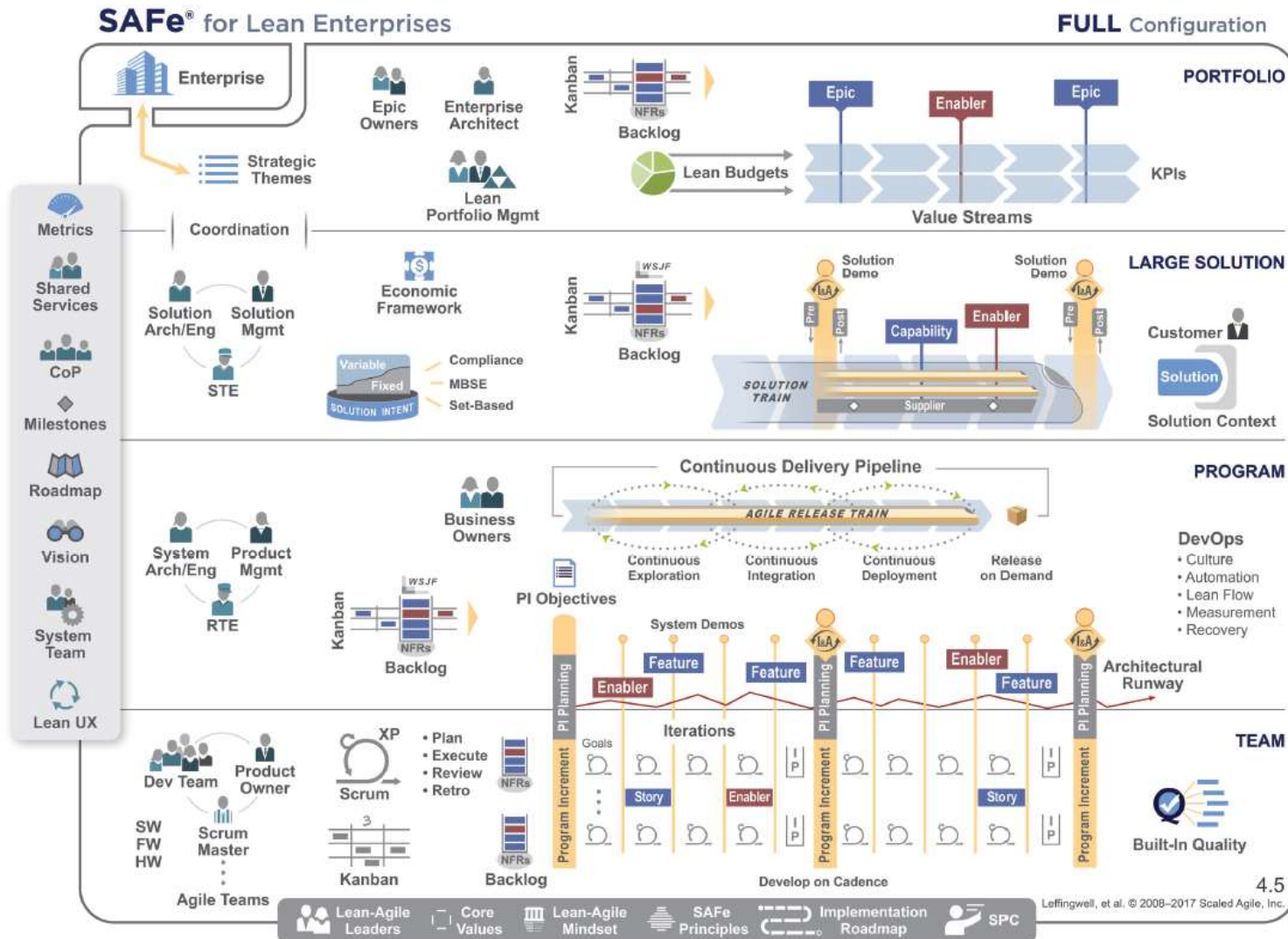
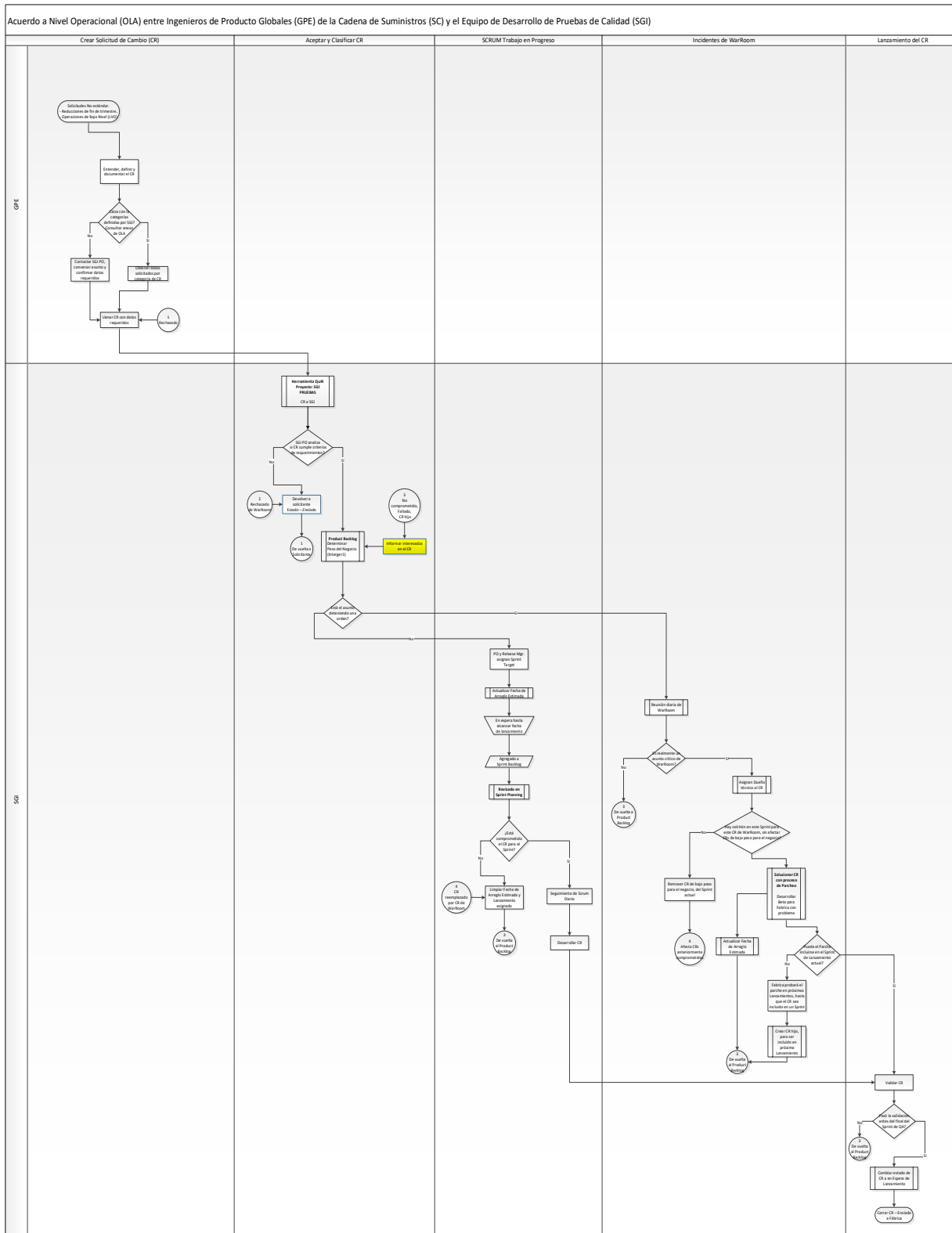


Ilustración 27 - Esquema SAFe (Scaled Agile Framework, 2017)

Ap 5. Diagrama de flujo OLA



Ap 6.Función SQL para pasar a columna el texto dividido por coma

```

/***** Object: UserDefinedFunction [dbo].[ftSplitStrings_Modern]
Script Date: 3/10/2018 7:08:06 PM *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO

CREATE FUNCTION [dbo].[ftSplitStrings_Modern]
(
    @List NVARCHAR(MAX),
    @Delimiter char(1)
)
RETURNS TABLE
WITH SCHEMABINDING AS
RETURN
    WITH E1(N)          AS ( SELECT 1 UNION ALL SELECT 1 UNION ALL SELECT 1 UNION ALL
SELECT 1
                        UNION ALL SELECT 1 UNION ALL SELECT 1 UNION ALL SELECT 1
                        UNION ALL SELECT 1 UNION ALL SELECT 1 UNION ALL SELECT 1),
    E2(N)              AS (SELECT 1 FROM E1 a, E1 b),
    E4(N)              AS (SELECT 1 FROM E2 a, E2 b),
    E42(N)             AS (SELECT 1 FROM E4 a, E2 b),
    cteTally(N)        AS (SELECT 0 UNION ALL SELECT TOP (DATALENGTH(ISNULL(@List,1)))
ROW_NUMBER() OVER (ORDER BY (SELECT NULL)) FROM E42),
    cteStart(N1)       AS (SELECT t.N+1 FROM cteTally t
WHERE (SUBSTRING(@List,t.N,1) = @Delimiter OR t.N = 0))
    SELECT Item = SUBSTRING(@List, s.N1,
ISNULL(NULLIF(CHARINDEX(@Delimiter,@List,s.N1),0)-s.N1,8000))
    FROM cteStart s;

```


Ap 7. Vista SQL de CR's del Sprint que acaba de pasar

```

USE [SPRINT_ARC]
GO

CREATE view [dbo].[vwGetCRsofjustEndedSprint] as
with lastnewSprint as (
select [SprintID],[TARGET_VERSION],[Sprint_Start] from DimSprint
except
select [SprintID],[TARGET_VERSION],[Sprint_Start] from DimSprint ds
join [dbo].[FactSprint] fact on ds.SprintID=fact.[fkSprintID]
)
select
ltrim(rtrim(bo.[ISSUEID])) [ISSUEID]
, isnull(bo.[Title], 'None') [Title]
, isnull(bo.[DUPLICATE_COUNT], 0) [DUPLICATE_COUNT]
, CONVERT(INT, REPLACE(CONVERT(VARCHAR(11), bo.[SUBMITDATE], 112), ', ', '')) [SUBMITDATE]
, CONVERT(INT, REPLACE(CONVERT(VARCHAR(11), bo.[LASTMODIFIEDDATE], 112), ', ', '')) [LASTMODIFIEDDATE]
, CONVERT(INT, REPLACE(CONVERT(VARCHAR(11), bo.[CLOSEDATE], 112), ', ', '')) [CLOSEDATE]
, CONVERT(INT, REPLACE(CONVERT(VARCHAR(11), bo.[FIX_NEEDED_BY], 112), ', ', '')) [FIX_NEEDED_BY]
, CONVERT(INT, REPLACE(CONVERT(VARCHAR(11), bo.[ESTIMATED_FIX_DATE], 112), ', ', '')) [ESTIMATED_FIX_DATE]
, CONVERT(INT, REPLACE(CONVERT(VARCHAR(11), bo.[COMPLETED_IN_DATE], 112), ', ', '')) [COMPLETED_IN_DATE]
, isnull(bo.[PROJECT_NAME], 'None') [PROJECT_NAME]
, isnull(bo.[SEVERITY], 'None') [SEVERITY]
, isnull(bo.[VISIBILITY], 'None') [VISIBILITY]
, isnull(bo.[URGENCY], 'None') [URGENCY]
, isnull(bo.[STATE], 'None') [STATE]
, isnull(bo.[PEER_REVIEW_STATUS], 'None') [PEER_REVIEW_STATUS]
, isnull(bo.[PRODUCT], 'None') [PRODUCT]
, isnull(bo.[HARDWARE_PLATFORM], 'None') [HARDWARE_PLATFORM]
, isnull(bo.[FOUND_IN_VERSION], 'None') [FOUND_IN_VERSION]
, isnull(bo.[REPORTER_TYPE], 'None') [REPORTER_TYPE]
, isnull(bo.[PRE_OR_POST_RELEASE], 'None') [PRE_OR_POST_RELEASE]
, isnull(bo.[ISSUETYPE], 'None') [ISSUETYPE]
, isnull(bo.[HOW_FOUND], 'None') [HOW_FOUND]
, isnull(bo.[SUBMITTERS_PROJECT], 'None') [SUBMITTERS_PROJECT]
, isnull(bo.[TARGET_VERSION], 'None') [TARGET_VERSION]
, isnull(bo.[COMPONENT], 'None') [COMPONENT]
, isnull(bo.[SUB_COMPONENT], 'None') [SUB_COMPONENT]
, isnull(bo.[SUBMITTER], 'None') [SUBMITTER]
, isnull(bo.[OWNER], 'None') [OWNER]
, isnull(bo.[LASTMODIFIER], 'None') [LASTMODIFIER]
, isnull(bo.[SUPPORT_REVIEW_ENGINEER], 'None') [SUPPORT_REVIEW_ENGINEER]
, isnull(bo.[DEVELOPMENT_ENGINEER], 'None') [DEVELOPMENT_ENGINEER]
, isnull(bo.[LAB_REVIEW_ENGINEER], 'None') [LAB_REVIEW_ENGINEER]
, isnull(bo.[TEST_ENGINEER], 'None') [TEST_ENGINEER]
, ds.[SprintID]
, datediff(dd, sofs.Date, GETDATE())+1 [DAYOFSPRINT]
from QUIX_SA.[dbo].[BO_QX_CR] bo
join lastnewSprint ds on bo.[TARGET_VERSION] = ds.[TARGET_VERSION]
join [SPRINT_ARC].[dbo].[DimDate] sofs on ds.[Sprint_Start] = sofs.[Date_key]
where bo.[ISSUEID] is not null
and PROJECT_NAME = 'GES TEST'
and component in ('Diagonal Common Component', 'Diagonal Product Component'
, 'Diagonal Tools', 'Test Content')
and product in ('Sustaining Servers')
GO

```

Ap 8. Vista SQL de CR's de previos Sprints que no se hayan cerrado

```

USE [SPRINT_ARC]
GO

CREATE view [dbo].[vwUnfinishedUnmatchedCRS] as

with nonClosedCRs as (
    select [ISSUEID]
    ,sp.TARGET_VERSION
    ,det.[STATE]
    ,[fkCLOSEDATE]
    ,[fkCOMPLETED_IN_DATE]
    ,lsmod.Email 'LASTMODIFIER'
    ,sp.[Sprint_Start]
    ,sp.SprintID
from [dbo].[FactSprint] fs
join [dbo].[DimDetailsCR] det on fs.[fkDetailsCRId] = det.[DetailsCRId]
join [dbo].[DimSprint] sp on fs.[fkSprintID] = sp.[SprintID]
join [dbo].[DimUsers] lsmod on fs.fkLASTMODIFIER = lsmod.UserId
where
    det.[STATE] not in ('Completed','Closed')
)
select nonClo.[ISSUEID]
,nonClo.TARGET_VERSION
,nonClo.[STATE]
,nonClo.LASTMODIFIER
,bo.[STATE] STATE_SA
,bo.TARGET_VERSION TARGET_VERSION_SA
,bo.LASTMODIFIER LASTMODIFIER_SA
,cast(convert(char(8),bo.CLOSEDATE,112) as int) 'CLOSEDATE'
,cast(convert(char(8),bo.COMPLETED_IN_DATE,112) as int) 'COMPLETED_IN_DATE'
,nonClo.[Sprint_Start]
,nonClo.SprintID
from [QUIX_SA].[dbo].[BO_QX_CR] bo
join nonClosedCRs nonClo on bo.ISSUEID = nonClo.ISSUEID
where
nonClo.STATE != bo.[STATE]
OR (nonClo.fkCLOSEDATE is null and bo.CLOSEDATE is not null)
OR (nonClo.fkCOMPLETED_IN_DATE is null and bo.COMPLETED_IN_DATE is not null)
OR (nonClo.fkCLOSEDATE != cast(convert(char(8),bo.CLOSEDATE,112) as int))
OR (nonClo.fkCOMPLETED_IN_DATE != cast(convert(char(8),bo.COMPLETED_IN_DATE,112) as int))
OR nonClo.LASTMODIFIER != bo.LASTMODIFIER
OR (bo.TARGET_VERSION is null and nonClo.TARGET_VERSION is not null)
OR bo.TARGET_VERSION != nonClo.TARGET_VERSION
GO

```

Ap 9.Procedimiento SQL para crear dimensión de fechas

```
--http://social.technet.microsoft.com/wiki/contents/articles/22776.t-sql-calendar-table.aspx
CREATE PROCEDURE [dbo].[uspPopulateDimDate]
@start_date date,@end_date date,@firstFYmonth int
AS
BEGIN
SET NOCOUNT ON;
SET DATEFIRST 1;
TRUNCATE TABLE DimDate
DECLARE
    @fiscalMonth int, @fiscalYear int;
SET @firstFYmonth = @firstFYmonth - 1

WHILE (@start_date<=@end_date)
    BEGIN
        SELECT @fiscalMonth = CASE WHEN DATEPART(MONTH, @start_date) > @firstFYmonth THEN DATEPART(MONTH,
@start_date) - @firstFYmonth
            ELSE DATEPART(MONTH, @start_date) + (12 - @firstFYmonth) END
        SELECT @fiscalYear = CASE WHEN ((12 - @firstFYmonth) + DATEPART(MONTH, @start_date)) > 12 THEN
DATEPART(YEAR,@start_date)+1 ELSE DATEPART(YEAR,@start_date) END

        INSERT INTO DimDate(
            [Date_key],[Date],[Year],[Semester],[Quarter_Number],[Month_Number],[Month_Name],[Week_of_Year],[W
eek_of_Month],
            [Day_of_Year],[Day_of_Month],[Day_of_Week],[Day_Name],[Fiscal_Year],[Fiscal_Semester],[Fiscal_Quar
ter],[Fiscal_Month],
            [Weeks_Monday_Date],[isweekEND]--, [isHoliday]
        )
        SELECT
            [Date_key]=CONVERT(int,CONVERT(VARCHAR(20), @start_date, 112)),
            [Date]= @start_date,
            [Year] = DATEPART(YEAR,@start_date),
            [Semester] = CASE WHEN DATEPART(MONTH, @start_date) <7 THEN '1' ELSE '2' END,
            [Quarter_Number] = DATEPART(QUARTER, @start_date),
            [Month_Number] = DATEPART(MONTH, @start_date),
            [Month_Name] = DATENAME(MONTH, @start_date),
            [Week_of_Year]= DATEPART(WEEK, @start_date),
            [Week_of_Month]= DATEPART(WEEK, @start_date) - DATEPART(WEEK, CONVERT(CHAR(6), @start_date, 112) +
'01') + 1,
            [Day_of_Year]= DATEPART(DAYOFYEAR, @start_date),
            [Day_of_Month]= DATEPART(day, @start_date),
            [Day_of_Week]= DATEPART(WEEKDAY, @start_date),
            [Day_Name]= DATENAME(WEEKDAY, @start_date),
            Fiscal_Year = @fiscalYear,
            [Fiscal_Semester] = CASE WHEN @fiscalMonth <7 THEN '1' ELSE '2' END,
            [Fiscal_Quarter] = DATEPART(QUARTER,
CONVERT(varchar,@fiscalYear)+'/'+CONVERT(varchar,@fiscalMonth)+'/'+'01'),
            [Fiscal_Month] = @fiscalMonth,
            [Weeks_Monday_Date] = DATEADD(dd,-1*(DATEPART(WEEKDAY, @start_date)-1),@start_date),
            isweekEND = CASE WHEN DATENAME(WEEKDAY, @start_date) in ('Saturday', 'Sunday') THEN '1' ELSE '0'
        END
        --,isHoliday = CASE WHEN @start_date IN (SELECT LOH.DateOfHoliday FROM ListOfHolidays LOH ) THEN 1
        ELSE 0 END
        SET @start_date =DATEADD(dd, 1, @start_date)
    END
END
GO
```

Ap 10. Acceso a fuente de datos

a. Ingresar a QUIX

El acceso a la herramienta web tiene que ser mediante a un usuario físico, puesto su uso va más allá de solo el departamento de SGI y por cada proyecto que se desee consultar, hay una solicitud de permisos asociada.



BO SQLNet Access Setup

BO SQLNet Login Setup. The BO SQLNet Login ID and password are to be used to login BO Database

Name	
SQLNet Login ID	SQLNet Login ID

→ UPDATE

Ilustración 28 - Ventana de QUIX usuarios SQL

La base de datos a la cual se otorga acceso es en realidad una vista materializada, donde se muestra solo los CR's de los proyectos que el usuario tiene permisos.

b. Acceso a base de datos de Oracle

El objetivo es extraer los datos directos de la base de datos, para ello hay que solicitar al equipo de Base de Datos dueño de aplicación QUIX, por un usuario y clave.

Debido a que lo que se desea es solo extracción de datos, con permisos simples de lectura es más que suficiente.

La solicitud tiene que ser aprobada por el jefe directo.

Items currently in selection.

Portfolio ID	Database Asset	Roles Selected	Justification
108884	quix-gcu42094	Read-Only	Data Analysis and Reporting for EG SC

Ilustración 29 - Solicitud acceso a QUIX DB

Después de la aprobación, la clave de acceso será recibida por correo electrónico.

c. Oracle driver para Windows

SQL Server viene con muchos drivers para conectarse a bases de datos externas, pero Microsoft no es responsable de crear drivers para que bases de datos Oracle tengan acceso externo, por lo tanto, SQL Server no viene con ningún driver de Oracle predeterminado.

Lo primero es descargar el paquete Componentes de Acceso a Datos de Oracle (ODAC, por sus siglas en inglés). Para ello se procede a buscar en Internet por “64-bit odac for windows x64” e ingresar al resultado que lleve a la página de Oracle.



Ilustración 30 - Búsqueda de ODAC en Bing

<http://www.oracle.com/technetwork/database/windows/downloads/index-090165.html>

Previo a cualquier descarga hay que aceptar las condiciones de licenciamiento en la página de Oracle.

64-bit Oracle Data Access Components (ODAC) Downloads

Important: The [32-bit Oracle Developer Tools for Visual Studio download](#) is required for Entity Framework design-time features. The downloads that support Entity Framework deployment below do not contain design-time tools, only run-time support.

You must accept the [OTN License Agreement](#) to download this software.


Accept License Agreement | Decline License Agreement

Ilustración 31 - Licenciamiento ODAC

Para facilitar pruebas de conectividad y permisos de acceso es útil que el paquete que se descargue contenga SQL*Plus el cual es un cliente de SQL en línea de comando.

64-bit ODAC 12.2c Release 1 (12.2.0.1.0) for Windows x64

[Released June 1, 2017]

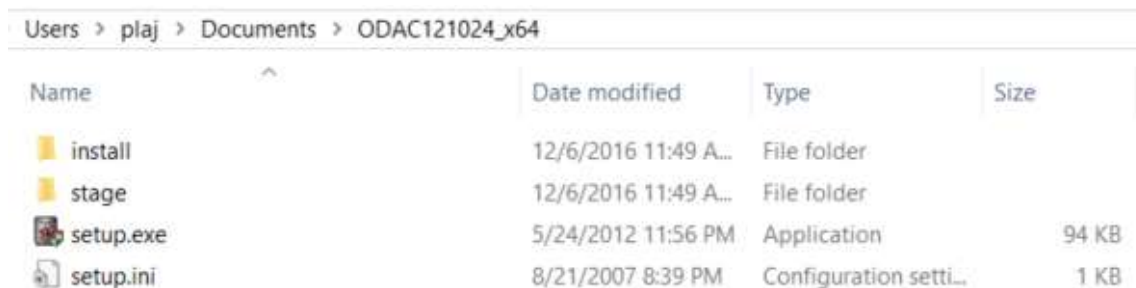
 [ODAC122010_x64.zip](#) - 415 MB (435,541,788 bytes)

Download Includes

- 64-bit Oracle Data Provider for .NET 4 12.2.0.1.0
- 64-bit Oracle Data Provider for .NET 2.0 12.2.0.1.0
- 64-bit Oracle Providers for ASP.NET 4 12.2.0.1.0
- 64-bit Oracle Providers for ASP.NET 2.0 12.2.0.1.0
- 64-bit Oracle Provider for OLE DB 12.2.0.1.0
- 64-bit Oracle Services for Microsoft Transaction Server 12.2.0.1.0
- 64-bit Oracle ODBC Driver 12.2.0.1.0
- 64-bit Oracle SQL*Plus 12.2.0.1.0
- 64-bit Oracle Instant Client 12.2.0.1.0

Ilustración 32 - ODAC a descargar con SQL*Plus

Posterior a la descarga y descomprimir el paquete en un directorio limpio, se procede a ejecutar el archivo *setup.exe*.



Name	Date modified	Type	Size
install	12/6/2016 11:49 A...	File folder	
stage	12/6/2016 11:49 A...	File folder	
setup.exe	5/24/2012 11:56 PM	Application	94 KB
setup.ini	8/21/2007 8:39 PM	Configuration setti...	1 KB

Ilustración 33 - ODAC descomprimido

El único aspecto a tener en consideración durante la instalación es en el Paso 7, donde hay que marcar la caja para ignorar todas las advertencias y aceptar en la ventana emergente.

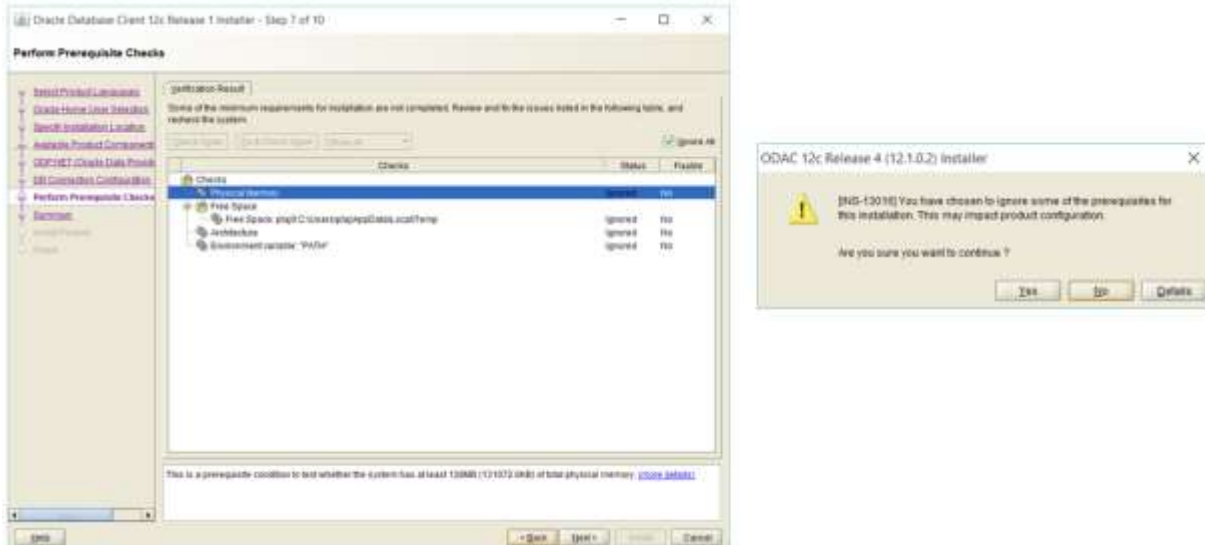


Ilustración 34 - Ignorar advertencias ODAC

Después es solo terminar el proceso de instalación.

d. Probar la conexión con SQL*Plus

SQL*Plus es una utilidad de línea de comando; una vez instalada la ruta del ejecutable debe estar dentro la variable de ambiente PATH.

```
C:\Users\plaj>set Path
Path=C:\app\client\plaj\product\12.1.0\client_1;C:\app\client\plaj\product\12.1.0\client_1
bin;C:\ProgramData\Oracle\Java\javapath;...
```

Ahora se procede a verificar el acceso a base de datos de QUIX.

```
C:\Users\plaj>sqlplus <USUARIO>/<PASSWORD>@<SERVIDOR>:<PUERTO>/QUIX
SQL*Plus: Release 12.1.0.2.0 Production on Tue Dec 6 15:03:45 2016
Copyright (c) 1982, 2014, Oracle. All rights reserved.
Connected to:

Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit Production
With the Partitioning, Real Application Clusters, OLAP, Data Mining and Real Application
Testing options

SQL>
```

Luego de corroborar el acceso al servidor, hay que confirmar la base de datos y las tablas requeridas, además hay que tomar en cuenta que es usando la variante del lenguaje SQL que pertenece a Oracle y es PL/SQL, y no Transact-SQL de Microsoft.

Finalmente, para cerrar la sesión es con *exit* o *quit*.

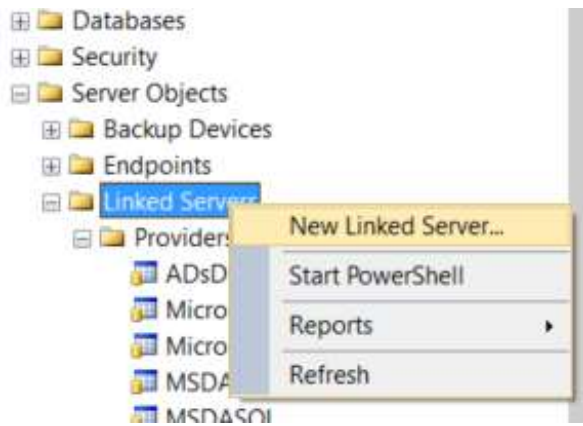
```
SQL> exit
Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit
Production
With the Partitioning, Real Application Clusters, OLAP, Data Mining
and Real Application Testing options
-----
SQL> quit
Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 - 64bit
Production
With the Partitioning, Real Application Clusters, OLAP, Data Mining
and Real Application Testing options
```

e. Vincular servidor Oracle a SQL Server

El propósito de instalar el ODAC es poder vincular el servidor Oracle a un motor de SQL Server en Windows. Así como se mencionó, SQL Server no viene con dicho driver, el cual es necesario para vincular el servidor Oracle.

SQL Server normal	Con ODAC instalado
<ul style="list-style-type: none"> + Databases + Security - Server Objects <ul style="list-style-type: none"> + Backup Devices + Endpoints - Linked Servers <ul style="list-style-type: none"> - Providers <ul style="list-style-type: none"> ADsDSOObject MSDAOSP MSDASQL MSOLAP SQLNCLI11 SQLOLEDB 	<ul style="list-style-type: none"> + Databases + Security - Server Objects <ul style="list-style-type: none"> + Backup Devices + Endpoints - Linked Servers <ul style="list-style-type: none"> - Providers <ul style="list-style-type: none"> ADsDSOObject MSDAOSP MSDASQL MSOLAP OraOLEDB.Oracle Search.CollatorDSO SQLNCLI10 SQLNCLI11 SQLOLEDB

OraOLEDB.Oracle es el proveedor necesario para conectar el motor de SQL Server a una base de datos en Oracle.



Haciendo un clic derecho en la carpeta de “Linked Servers” y escogiendo la opción “New Linked Server”.

Ilustración 35 - SSMS nuevo servidor vinculado

La ventana de instalación para el servidor a vincular, preguntará por:

- Linked server: es el nombre que se le quiere dar a la conexión
- Server type: en este caso es escoger “*Other data source*” debido a que la base de datos se encuentra en Oracle y no SQL Server.
- Provider: se selecciona el ODAC que se acaba de instalar, en este caso “*Oracle Provider for OLE DB*”.
- Product name: el nombre de la base de datos a la cual se va a conectar
- Data source: similar a cuando se hizo la conexión con SQL*Plus,
<SERVIDOR>:<PUERTO>/<DB NAME>

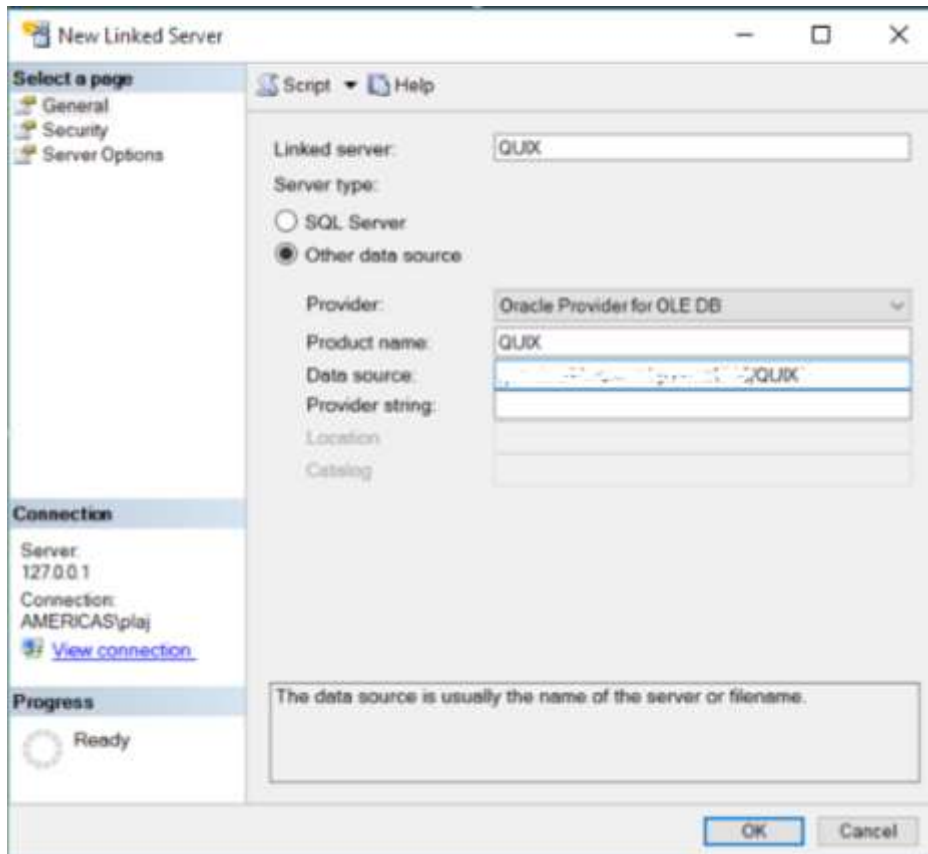


Ilustración 36 - Crear servidor vinculado

Antes de pulsar **OK**, hay que configurar el usuario y clave a utilizar para realizar la conexión. Para ello se va a la página de seguridad, se escoge la última opción y se ingresa el usuario y clave.

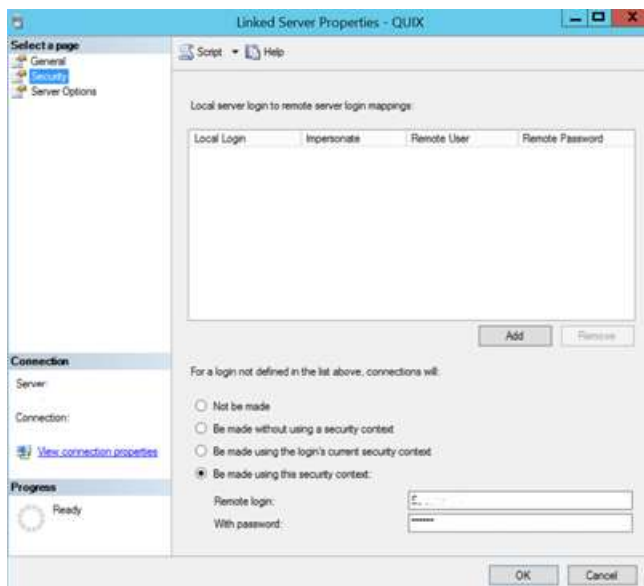
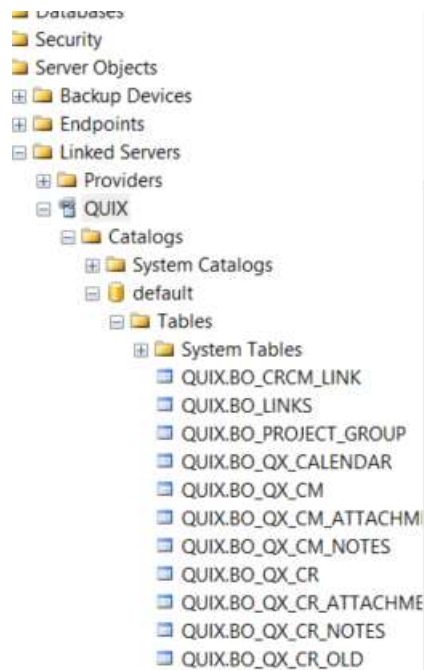


Ilustración 37 - Seguridad servido vinculado



Como se hizo con SQL*Plus, hay que confirmar la conexión y acceso a las tablas. Expandiendo el directorio de Tablas debajo del servidor vinculado.

Ilustración 38 - Tablas en servidor vinculado

La clave por razones de seguridad no es permanente, esta expira, por lo que es bueno probar que la conexión todavía funcione.

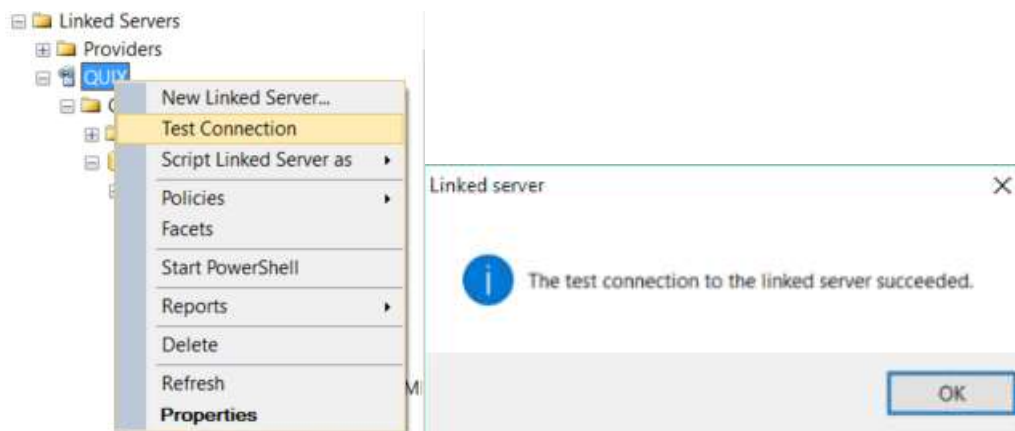


Ilustración 39 - Prueba conexión servidor vinculado

Un ejemplo de una consulta directa al servidor vinculado:

```
select TOP 10 ISSUEID,PROJECT_NAME, LASTMODIFIEDDATE from QUIX..QUIX.BO_QX_CR  
order by LASTMODIFIEDDATE desc
```

(10 row(s) affected)

ISSUEID	PROJECT_NAME	LASTMODIFIEDDATE
QXCR1001522732	ROM UEFI	2016-12-08 19:22:12.0000000
QXCR1001506260	NSP	2016-12-08 19:22:03.0000000
QXCR1001519253	GES TEST	2016-12-08 19:21:01.0000000
QXCR1001522028	GES TEST	2016-12-08 19:20:20.0000000
QXCR1001522730	ROM UEFI	2016-12-08 19:19:30.0000000
QXCR1001521166	GES TEST	2016-12-08 19:18:45.0000000
QXCR1001522726	ROM UEFI	2016-12-08 19:17:07.0000000
QXCR1001515047	ROM UEFI	2016-12-08 19:14:44.0000000
QXCR1001522527	Windows Server 2012 R2	2016-12-08 19:14:44.0000000
QXCR1001423555	QLogic-CNA-HBA	2016-12-08 19:13:47.0000000

Ap 11. Etapa Interactiva

a. Creación de base de datos QUIX_SA

La base de datos lleva el nombre de QUIX, puesto esta etapa es casi una copia, pero se le agrega el sufijo de SA por las siglas en inglés de la etapa interactiva del DW 2.0.

Además, para un mejor rendimiento, el Log de la base de datos va en un disco duro aparte, como también el archivo de datos crecerá a nivel de 1Mb y no de forma porcentual.

```
CREATE DATABASE [QUIX_SA]
CONTAINMENT = NONE
ON PRIMARY
( NAME = N'QUIX_SA', FILENAME = N'F:\MSAS12.MSSQLSERVER\OLAP\QUIX_SA.mdf' , SIZE =
759808KB , MAXSIZE = UNLIMITED, FILEGROWTH = 1024KB )
LOG ON
( NAME = N'QUIX_SA_log', FILENAME =
N'E:\MSAS12.MSSQLSERVER\OLAP\Log\QUIX_SA_log.ldf' , SIZE = 291904KB , MAXSIZE =
2048GB , FILEGROWTH = 10%)
GO
```

b. Creación de tablas

La base de datos contiene dos tablas, la principal es BO_QX_CR con 227 columnas y la otra es solo para llevar control de cuáles proyectos son los que hay que extraer y su nombre es ProjectsToSync.

```
CREATE TABLE [dbo].[BO_QX_CR](
  [ISSUEID] [varchar](32) NOT NULL,
  [LASTMODIFIEDDATE] [datetime2](7) NULL,
  [SUBMITDATE] [datetime2](7) NULL,
  [PROJECT_NAME] [varchar](64) NULL,
  ... --Otras 223 columnas
  CONSTRAINT [PK__BO_QX_CR__ISSUEID] PRIMARY KEY CLUSTERED
(
  [ISSUEID] DESC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]

CREATE TABLE [dbo].[ProjectsToSync](
  [PROJECT_NAME] [varchar](100) NOT NULL,
  [DateAdded] [smalldatetime] NULL CONSTRAINT [DF_ProjectsToSync_DateAdded] DEFAULT
(getutcdate()),
  CONSTRAINT [PK__Projects__BCBE781DD530BE96] PRIMARY KEY CLUSTERED
(
  [PROJECT_NAME] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
```

La rapidez de respuesta de la tabla BO_QX_CR en la base de datos QUIX_SA de la etapa interactiva, debe ser menor a 15 segundos, tanto para los reportes directos como para la sincronización con la base de datos de producción. Para ello se genera un índice no agrupado (nonclustered, en inglés), por aparte del agrupado (clustered, en inglés) de la llave primaria, con un almacenamiento materializado de ciertas columnas claves para hacer las consultas.

```
CREATE NONCLUSTERED COLUMNSTORE INDEX [ix-strd_PROJECT_LASTMODIFIED_SUBMITTED_ISSUEID]
ON [dbo].[BO_QX_CR]
(
  [PROJECT_NAME], [LASTMODIFIEDDATE], [SUBMITDATE], [ISSUEID]
) WITH (DROP_EXISTING = OFF)
GO
```

c. Actualización de la tabla BO_QX_CR

El proceso de sincronización con la tabla de producción se dividió en 3 pasos: insertar, actualizar y borrar filas, con el fin de tener un mejor control de fallos y facilidad de repetir o corregir uno de los pasos sin necesidad de repetir los demás.

i. Procedimiento almacenado para insertar

El primer procedimiento almacenado se llama *spSyncInstQUIX*.

```
CREATE procedure [dbo].[spSyncInstQUIX]
    @range int = -5,
    @rangeType char(2) = 'hh'
as
begin
    SET NOCOUNT ON;

    declare @rmttime datetime,
            @syncDate datetime,
            @cmd nvarchar(255),
            @Parms nvarchar(255);

    select @range = case when @range > 0 then @range * -1 else @range end

    select @rmttime=rmttime
    from OPENQUERY([QUIX], 'select sysdate as rmttime from dual') as t;

    set @Parms = '@Units int, @date datetime, @dtOutput datetime OUTPUT'

    set @cmd = 'set @dtOutput = Dateadd(' + @rangeType + ',@Units, @date)';

    exec sp_ExecuteSQL @cmd, @parms, @Units = @range, @date = @rmttime, @dtOutput =
    @syncDate OUTPUT
```

Debido a que la fuente está en un servidor remoto en Oracle, es basándose en la hora de ese servidor y no en la del servidor local la que se usa para sincronizar las filas de acuerdo al campo LASTMODIFIEDDATE, el cual indica la última fecha y hora en la que se modificó la fila. La consulta busca la hora actual del servidor remoto, menos el rango de tiempo que se quiera consultar; de manera predeterminada se indica un rango de 5 horas, como se puede apreciar en el parámetro @range de -5 y @rangeType de 'hh'. También se asegura de que el valor del rango siempre sea en negativo, porque no se quiere extraer filas de un tiempo que no ha pasado. El resultado es @syncDate, que es la fecha y hora desde la que se quieren extraer filas del servidor vinculado.

El siguiente paso es sacar las filas que están en la fuente de datos en el servidor vinculado pero todavía no se han agregado a *QUIX_SA* y guardarlas en una tabla temporal llamada *#InstQUIX*.

```

if OBJECT_ID('tempdb..#InstQUIX') IS NOT NULL
    drop table #InstQUIX

select * into #InstQUIX From (
    select
        QXCRs.[ISSUEID]
        ,QXCRs.[PROJECT_NAME]
        ,QXCRs.LASTMODIFIEDDATE
        ,QXCRs.SUBMITDATE
        ... --Otras 223 columnas
    from [QUIX].[QUIX].[BO_QX_CR] QXCRs with (nolock)
    left join [QUIX_SA].[dbo].[BO_QX_CR] qxsa on QXCRs.[ISSUEID] = qxsa.[ISSUEID]
    join [QUIX_SA].[dbo].[ProjectsToSync] pts
        on coalesce(QXCRs.[PROJECT_NAME], '') = pts.[PROJECT_NAME]
    where qxsa.[ISSUEID] is null
    and QXCRs.LASTMODIFIEDDATE>=@syncDate
    and QXCRs.[ISSUEID] is not null
    and (QXCRs.LASTMODIFIEDDATE is null or not
        QXCRs.LASTMODIFIEDDATE < QXCRs.SUBMITDATE)
    ) realQuix;

```

Es importante destacar que esta consulta se hace sin bloquear la base de datos de producción, como lo establece la opción *nolock* en el *Select*, las ventajas del *nolock* que logran un más rápido rendimiento en la consulta, mientras que eliminar bloqueos en un servidor de SQL con una alta carga tiene la desventaja de que los datos puedan estar desactualizados con respecto de *inserts* o *updates* que se estén realizando al mismo tiempo que el *select*, lo cual es atenuado porque la consulta se encarga de sincronizar cada 15 minutos todo lo modificado en las últimas 5 horas (lo que no haya sido cambiado en una primera sincronización lo será en las siguientes). El filtro de la consulta se hace con base en solo los proyectos que estén en la tabla *ProjectsToSync*, donde la llave primaria *ISSUEID* no se encuentre ya en la etapa interactiva y donde *LASTMODIFIEDDATE* sea mayor o igual a *@syncDate*.

La última parte de este procedimiento almacenado es la transacción donde se da la inserción de los nuevos CR's. Para poder hacer la escritura, es necesario quitar antes el índice no agrupado con almacenamiento de columnas. Este tipo de índice brinda una ventaja en tiempo de lectura, ya que guarda las columnas que se le indican como otra

tabla y al ser solo cuatro columnas en vez de las 227 que tiene la tabla real, baja el tiempo de lectura. Pero tiene la desventaja de que la tabla real no puede ser modificada al estar el índice activo, primero hay que desactivarlo, hacer los cambios en las filas y después reconstruir el índice.

```

SET XACT_ABORT ON
begin transaction syncInstQUIX;

ALTER INDEX [ix-strd_PROJECT_LASTMODIFIED_SUBMITTED_ISSUEID] ON [dbo].[BO_QX_CR]
DISABLE

insert into [BO_QX_CR]
OUTPUT inserted.[ISSUEID],inserted.SUBMITDATE,inserted.[PROJECT_NAME]
select iq.* from #InstQUIX iq

ALTER INDEX [ix-strd_PROJECT_LASTMODIFIED_SUBMITTED_ISSUEID] ON [dbo].[BO_QX_CR]
REBUILD PARTITION = ALL

commit transaction syncInstQUIX
SET XACT_ABORT OFF

drop table #InstQUIX

END

GO

```

Después de que se hace el *insert* se reconstruye el índice de nuevo, se cierra la transacción y se elimina la tabla temporal.

Sin el índice con columnas almacenadas el costo de lectura y escritura en la consulta pasa de 21% a un 74%. Mientras que el tiempo de reconstruir el índice es solo de un segundo.

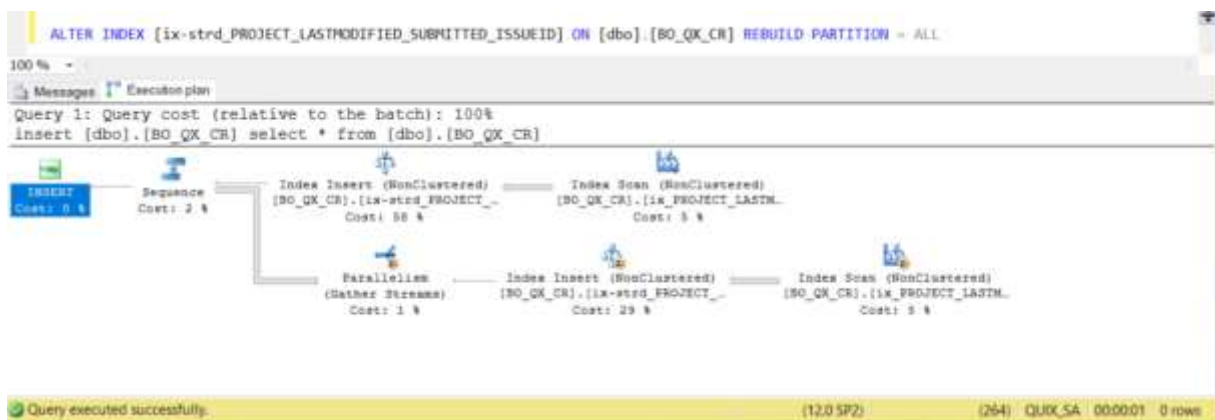


Ilustración 40 - Reconstrucción índice de almacenamiento de columnas

ii. Procedimiento almacenado para actualizar

El procedimiento almacenado lleva el nombre de spSyncUptQUIX e incluye la parte de la hora remota con la validación del rango de tiempo.

```
CREATE procedure [dbo].[spSyncUptQUIX]
    @range int = -5,
    @rangeType char(2) = 'hh'
as
begin
    SET NOCOUNT ON;

    declare @rmttime datetime,
        @syncDate datetime,
        @cmd nvarchar(255),
        @Parms nvarchar(255);

    select @range = case when @range > 0 then @range * -1 else @range end

    select @rmttime=rmttime
        from OPENQUERY([QUIX], 'select sysdate as rmttime from dual') as t;

    set @Parms = '@Units int, @date datetime, @dtOutput datetime OUTPUT'
```

La tabla temporal en este caso se llama #UptQUIX y ya no considera los proyectos por sincronizar, pues solo se basa en actualizar lo que ya haya sido insertado.

```
if OBJECT_ID('tempdb..#UptQUIX') IS NOT NULL
    drop table #UptQUIX

select * into #UptQUIX from (
    select
        QXCRs.[ISSUEID]
        ,QXCRs.LASTMODIFIEDDATE
        ,QXCRs.SUBMITDATE
        ... --Otras 224 columnas
    from [QUIX]..[QUIX].[BO_QX_CR] QXCRs with (nolock)
        join [dbo].[BO_QX_CR] qxsa on QXCRs.[ISSUEID] = qxsa.[ISSUEID]
            and QXCRs.LASTMODIFIEDDATE <> qxsa.LASTMODIFIEDDATE
    where QXCRs.LASTMODIFIEDDATE>@syncDate
        and QXCRs.LASTMODIFIEDDATE>=QXCRs.SUBMITDATE
        and QXCRs.[ISSUEID] is not null
) realQuix;
```

La transacción de actualización se lleva a cabo con una instrucción de *update*.

De igual forma se borra y reconstruye el índice.

```

SET XACT_ABORT ON
begin transaction syncUptQUIX;

ALTER INDEX [ix-strd_PROJECT_LASTMODIFIED_SUBMITTED_ISSUEID] ON [dbo].[BO_QX_CR]
DISABLE

update target
SET target.[INTERNAL_SYSTEM_ID] = source.INTERNAL_SYSTEM_ID
, target.[SYSTEM_TYPE] = source.SYSTEM_TYPE
, target.[PCT_URL] = source.PCT_URL
, target.[ISSUEID] = source.ISSUEID
, target.[PROJECT_NAME] = source.PROJECT_NAME
, target.[STATE] = source.STATE
, target.[TITLE] = source.TITLE
, target.[LASTMODIFIEDDATE] = source.LASTMODIFIEDDATE
OUTPUT
deleted.ISSUEID,deleted.LASTMODIFIEDDATE,inserted.ISSUEID,inserted.LASTMODIFIEDDATE
FROM [BO_QX_CR] as target
join #UptQUIX as source on target.ISSUEID = source.ISSUEID
and target.LASTMODIFIEDDATE <> source.LASTMODIFIEDDATE

ALTER INDEX [ix-strd_PROJECT_LASTMODIFIED_SUBMITTED_ISSUEID] ON [dbo].[BO_QX_CR]
REBUILD PARTITION = ALL
commit transaction syncUptQUIX

SET XACT_ABORT OFF
drop table #UptQUIX

END
GO

```

iii. Procedimiento almacenado para borrar

El tercer procedimiento de la parte de sincronización se llama *spSyncDelQUIX*.

```

CREATE procedure [dbo].[spSyncDelQUIX]
    @range int = -1,
    @rangeType char(2) = 'mm'
as
begin
    SET NOCOUNT ON;

declare @rmvertime datetime,@syncDate datetime,@cmd nvarchar(255),@Parms nvarchar(255);

select @range = case when @range > 0 then @range * -1 else @range end

select @rmvertime=rmvertime
from OPENQUERY([QUIX], 'select sysdate as rmvertime from dual') as t;

set @Parms = '@Units int, @date datetime, @dtOutput datetime OUTPUT'

set @cmd = 'set @dtOutput = Dateadd(' + @rangeType + ',@Units, @date)';

exec sp_ExecuteSQL @cmd, @parms, @Units = @range, @date = @rmvertime, @dtOutput =
@syncDate OUTPUT

if OBJECT_ID('tempdb..#DelQUIX') IS NOT NULL
    drop table #DelQUIX

select * into #DelQUIX
From (
    select qxsa.[ISSUEID] from [dbo].[BO_QX_CR] qxsa
        left join [QUIX].[QUIX].[BO_QX_CR] QXCRs with (nolock)
            on qxsa.[ISSUEID] = QXCRs.[ISSUEID]
        where QXCRs.[ISSUEID] is null and qxsa.[ISSUEID] is not null
            and (qxsa.SUBMITDATE>=@syncDate or qxsa.LASTMODIFIEDDATE>=@syncDate)
) realQuix;

ALTER INDEX [ix-strd_PROJECT_LASTMODIFIED_SUBMITTED_ISSUEID] ON [dbo].[BO_QX_CR]
DISABLE

SET XACT_ABORT ON
begin transaction syncDelQUIX;

DELETE bo
    OUTPUT deleted.*
    FROM [dbo].[BO_QX_CR] AS bo
        INNER JOIN #DelQUIX AS dq ON bo.[ISSUEID] = dq.[ISSUEID]

commit transaction syncDelQUIX
SET XACT_ABORT OFF
drop table #DelQUIX

ALTER INDEX [ix-strd_PROJECT_LASTMODIFIED_SUBMITTED_ISSUEID] ON [dbo].[BO_QX_CR]
REBUILD PARTITION = ALL

END
GO

```

La finalidad de este procedimiento de borrado es eliminar los CR's que ahora pertenezcan a un proyecto al cual no se tenga acceso, por lo que los datos presentados podrían ser erróneos.

iv. Ejecución automática de procedimientos de sincronización

La sincronización se da mediante trabajos calendarizados o *Jobs* como se le conocen en *SQL Server*.

La parte de borrado se da todos los días a las 10pm hora local del servidor y llama al procedimiento *spSyncDeIQUIX*.

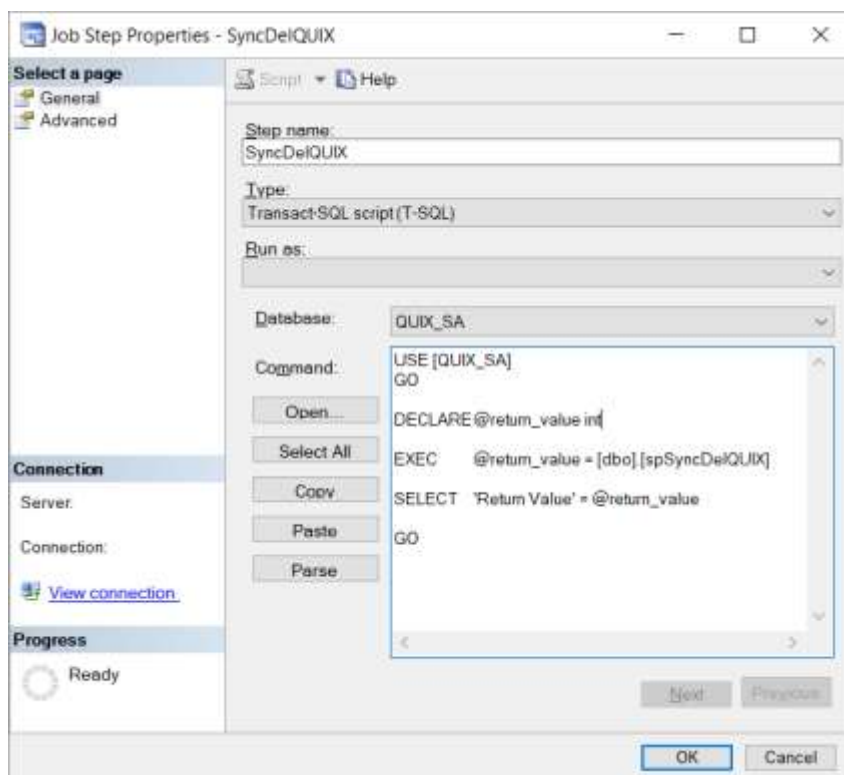


Ilustración 41 - Paso SyncDeIQUIX

El *Job* de insertar y actualizar corre cada 15 minutos y se llama *QUIX_SA-SyncingWithLinkedServer*, tiene dos pasos, en el primero ejecuta al procedimiento *spSyncInstQUIX* y en el segundo a *spSyncUptQUIX*.

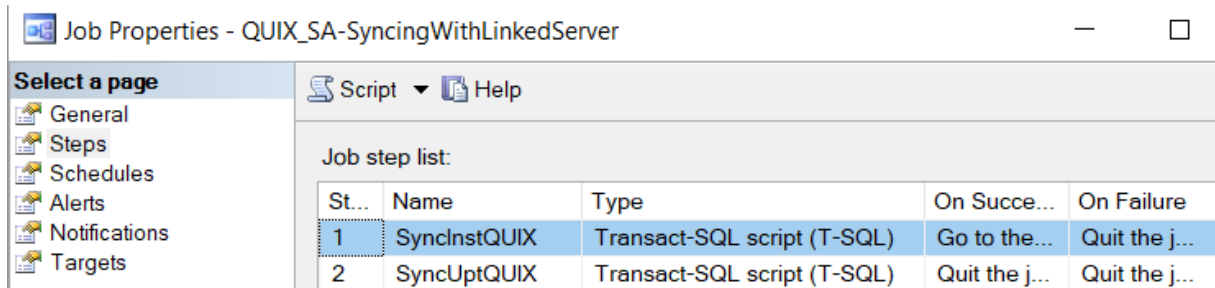


Ilustración 42- Pasos de Job QUIX_SA-SyncingWithLinkedServer

En caso de fallo el paso *SyncInstQUIX* concatenará los datos de salida al archivo ***F:\MSSQL12\MSSQLSERVER\Log\jobSyncInstQUIX.log***; mientras que *SyncUptQUIX* lo hará en el mismo directorio al archivo *jobSyncUptQUIX.log*.

Ap 12. Etapa Integrada

a. Creación del ETL de la base de datos de multidimensional SPRINT_DW

La herramienta utilizada fue *SQL Server Integration Services*, más conocida por sus siglas *SSIS*. Esta herramienta ya viene junto con la versión Enterprise de *SQL Server 2014* como se detalló en la sección *Punto de vista técnico*. El *SSIS 2014*, a diferencia del *SSIS 2012*, no viene con un editor de paquetes llamado *SQL Server Data Tools (SSDT)*, por sus siglas en inglés). Seguimos las instrucciones en *Microsoft Docs*, para instalar *SSDT*; en el caso de *SSIS 2014* hay que descargar la versión 16.5 de *SQL Server Data Tools* para usarla en conjunto con *Visual Studio 2013* (*Microsoft Docs*, 2018).

Después de haber estructurado las respectivas dimensiones y tabla de hechos para la base de datos correspondiente a la etapa integrada, se procedió con la creación de los ETLs para llevar a cabo la carga de información entre la etapa integrada y la interactiva, se llevó a cabo la siguiente estrategia para la creación del *ETL_SPRINT_DW.dtsx*.

b. Agregar los Sprints actuales a la dimensión DimSprint

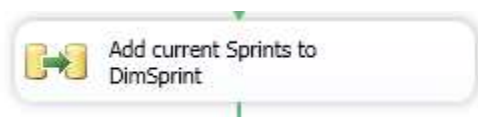


Ilustración 43 - Agregar Sprint actual a DimSprint

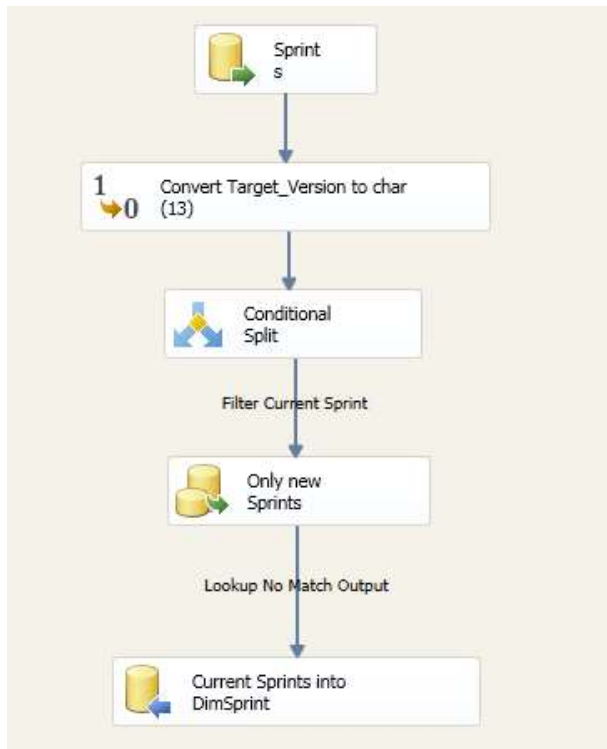


Ilustración 44 - Proceso de Agregar Sprint

De acuerdo con lo anterior se procede a configurar el componente “Data Flow Task” agregando diversos componentes que permiten hacer la lectura de la información contra la base de datos de la etapa interactiva QUIX_SA permitiendo filtrar por los Sprints actuales que se deben de insertar en la dimensión DimSprint.

Esta es la dimensión más importante de todo el diseño debido a que administrara los rangos de extracción y movimiento de datos entre los sectores, de acuerdo con la fecha del Sprint. El componente Conditional Split es el que valida dicho proceso; para esta etapa se validó que el rango de fechas para el Sprint_Start fuera mayor a 20 días en el pasado y menor a la fecha actual, esto posibilita determinar los Sprints que están actualmente en ejecución.

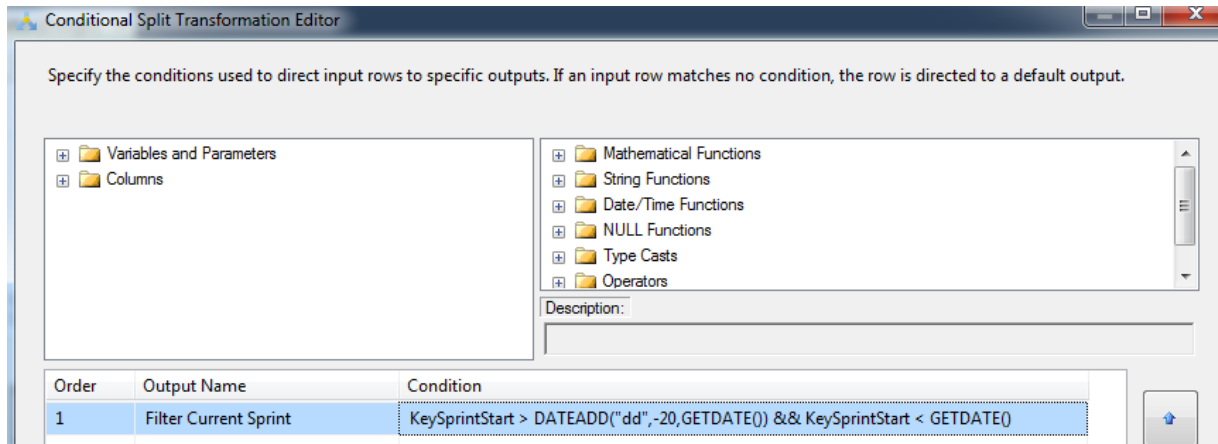


Ilustración 45 - Conditional Split Sprint actual

c. Hacer los llenados de las demás dimensiones

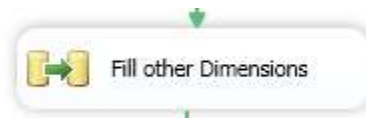


Ilustración 46 - Llenar otras dimensiones

Para llenar las siguientes dimensiones: DimWorkedBy, DimRootCause, DimProduct y DimDetailsCR, se utilizó un componente SORT por cada una de ellas con el fin de filtrar ciertos campos del Data Flow y con ello pasar al siguiente paquete las columnas asociadas a la dimensión, seguidamente un componente LOOKUP por cada dimensión para insertar lo que aún no se encuentre en la dimensión y con ello evitar los duplicados y por último el OLEDB DESTINATION para hacer la respectiva carga de los datos generados como salida a los procesos anteriores.

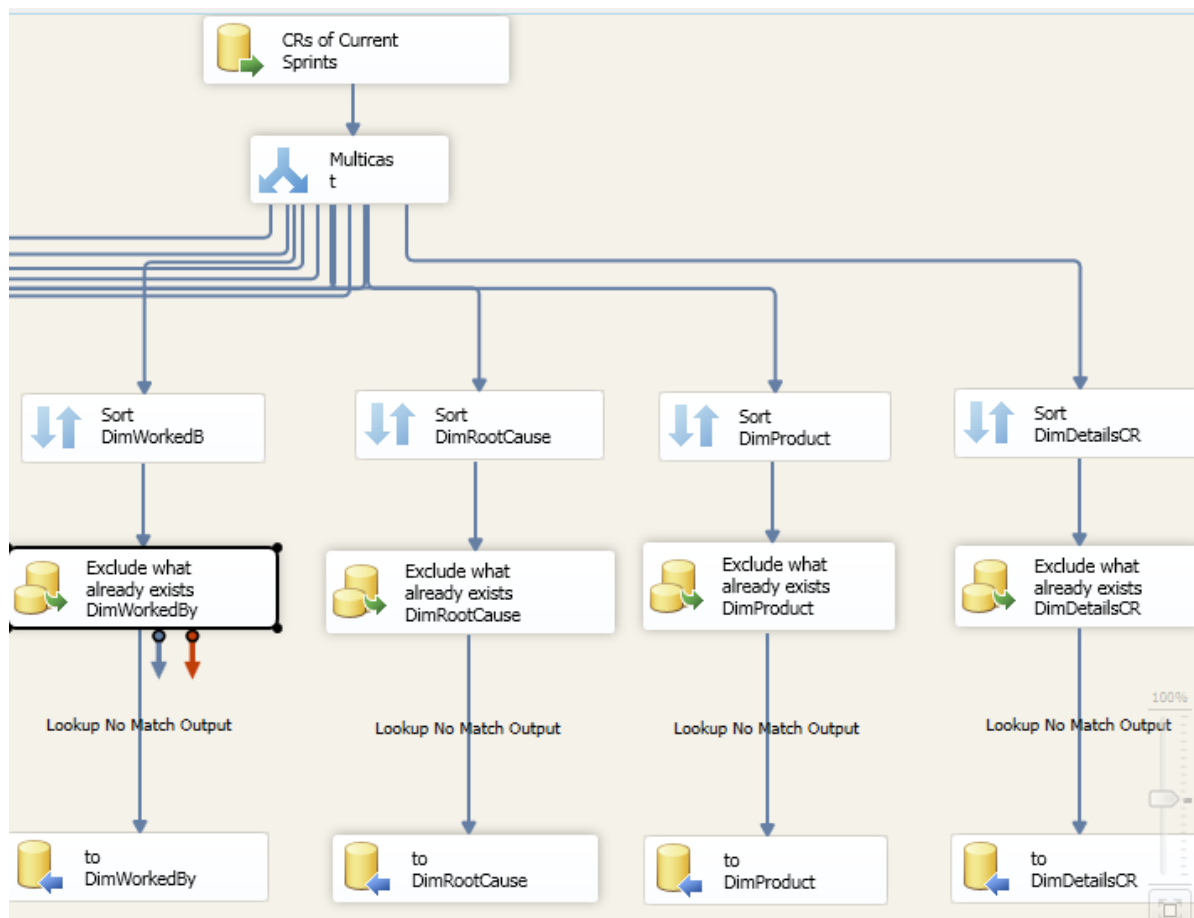


Ilustración 47 - Proceso de llenar demás dimensiones

Respecto del llenado de la dimensión DimUsers, semejante a las anteriores dimensiones, se utilizó un componente SORT para cada una de los campos referentes a: Submitter, Owner, Lastmodifier, Development Engineer, Lab Review Engineer, Test Engineer con el fin de filtrar únicamente dichas columnas y quitar los duplicados, seguidamente un componente LOOKUP por cada dimensión para insertar únicamente lo que no se encuentre en la dimensión, después un componente de UNION ALL para hacer la unión de todos los campos anteriores en uno solo seguidamente de un SORT para hacer la revisión de duplicados y por el ultimo el OLEDB DESTINATION para hacer la respectiva carga de lo generado anteriormente.

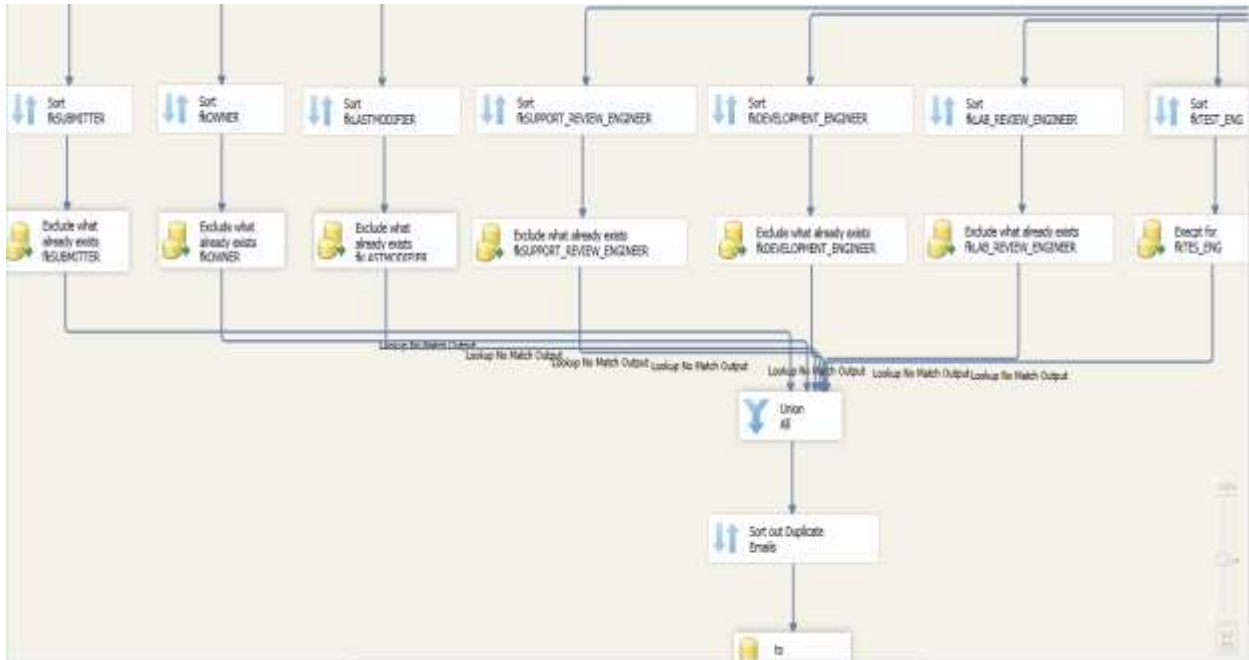


Ilustración 48 - Proceso de llenado de dimensión Usuarios

d. Carga de la tabla de hechos



Ilustración 49 - Llenar tabla de hechos de Sprint

Los cuatro pasos de este proceso son:

1. Extraer datos CR's de etapa interactiva.
2. Convertir tipo de datos.
3. Asegurar el no incluir duplicados.
4. Cargar los datos a la tabla de hechos.

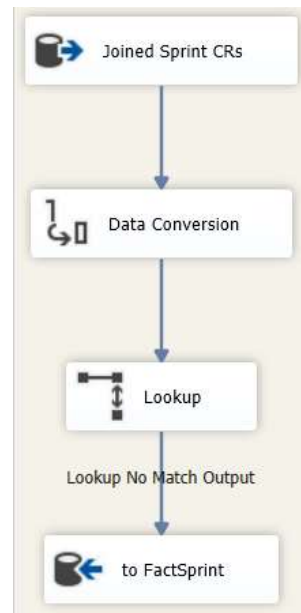


Ilustración 50 - Proceso llenado tabla de hechos del Sprint

Ap 13. Etapa Archivada

a. Ejecución automática del ETL de SPRINT_DW

El JOB encargado de llevar a cabo la sincronización de la parte Integrada SPRINT_DW, se llama SPRINT_DW-DailySprintSnapshot, y corre todos los días a las 3pm hora local después de la reunión de Scrum, en donde se define si existe algún CR que haya que eliminarlo, suspenderlo del sprint por falta de recursos o por poca prioridad o, al contrario, agregarlo por ser una urgencia para evitar incidentes, corregir un problema en manufactura o priorizar una orden de compra. Esta reunión es informativa de los cambios realizados durante el transcurso del día por el dueño técnico de cada CR, y luego de esta reunión se generan reportes a gerencia, como un resumen del progreso del día.

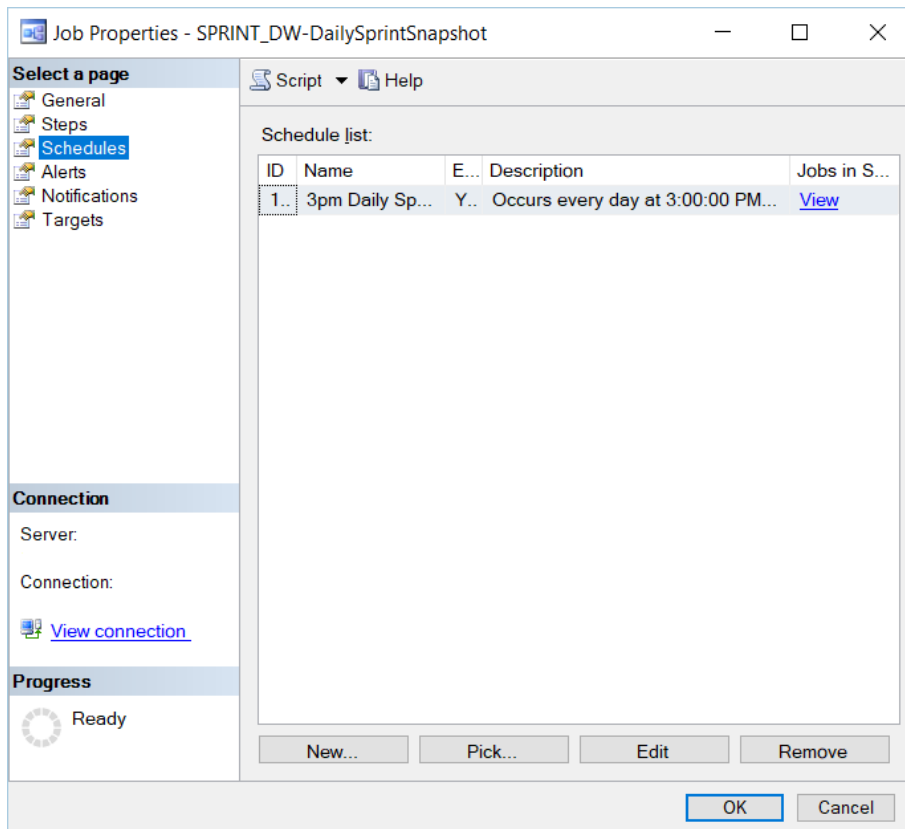


Ilustración 51 - Job SPRINT_DW-DailySprintSnapshot

b. Creación de ETL's de la base de datos multidimensional SPRINT_ARC

El proceso de ETL que se llama ETL_OLD_SPRINT_ARC.dtsx de esta etapa es exactamente el mismo al de la etapa integrada, con la salvedad de que el componente Conditional Split valida un diferente rango de fecha, ya que la etapa archivada maneja todos los CR's de Sprints pasados, por ende el rango de fecha para este condicional valida que la fecha Sprint_start sea menor a 20 días en el pasado a partir de la fecha actual; con lo que se excluye el Sprint actual.

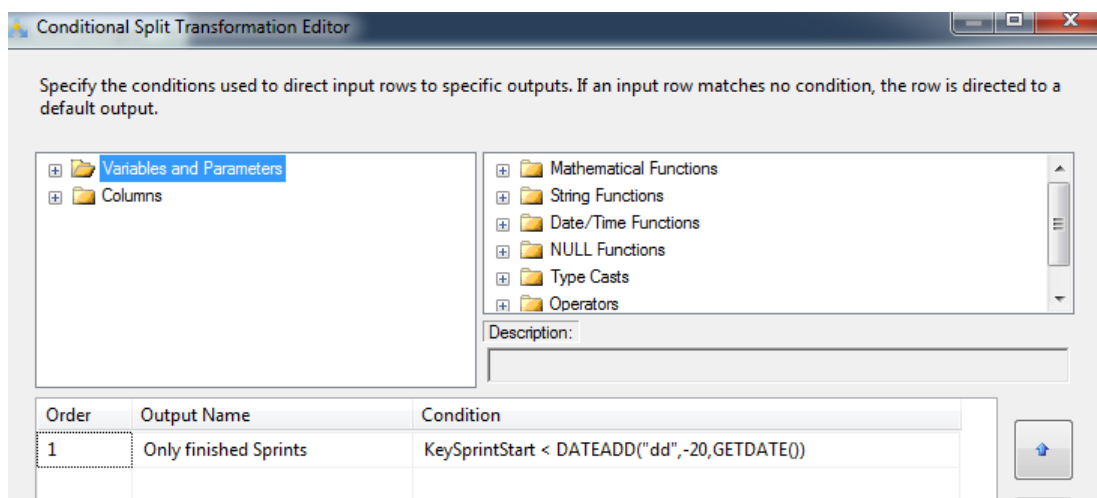


Ilustración 52 - Conditional Split de etapa archivada

Del lado de SQL Server, el Job encargado de ejecutar este paquete se llama SPRINT_ARC-SprintEndDay21 y está calendarizado para correr cada dos semanas.

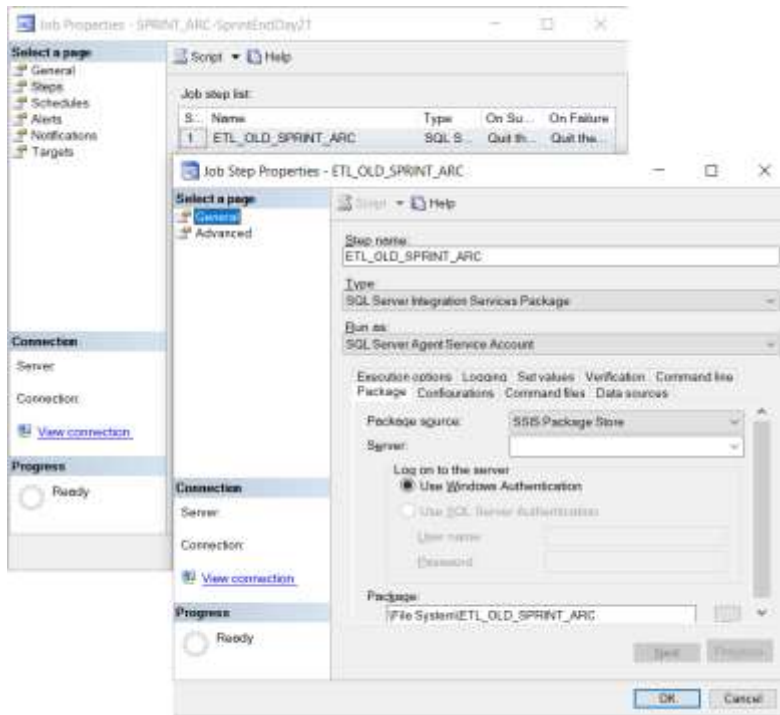


Ilustración 53 - Job SPRINT_ARC-SprintEndDay21

También se crea otro paquete ETL_UPDATE_UNFINISHED_CRs_ARC.dtsx, donde se eliminan los CR's que no se hayan terminado y que hayan sido migrados a SPRINTs más nuevos o que el estado del CR haya cambiado.

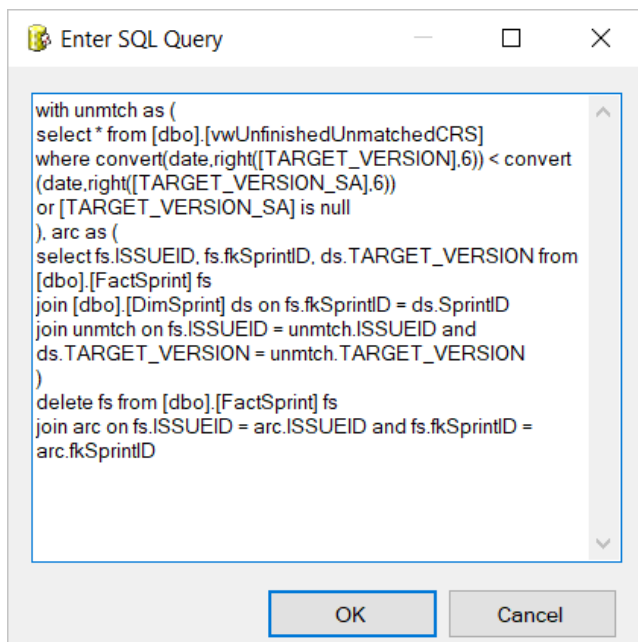


Ilustración 54 - Declaración de SQL para actualizar CRs no cerrados

c. SPRINT_ARC en SSAS

La solución en SSAS muestra desde la fuente de datos en SQL Server, la vista, el cubo, las dimensiones y los roles de usuarios para acceder al cubo.

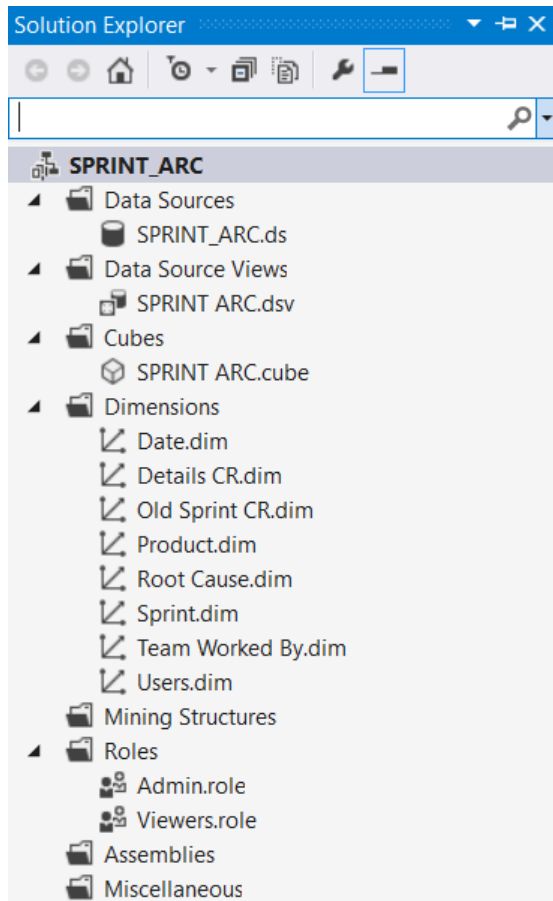


Ilustración 55 - Proyecto SPRINT_ARC en Visual Studio 2013

Posterior a indicar la fuente de datos, sigue la vista, si en la base de datos en SQL se establecieron las relaciones entre tablas, entonces estas igual se pueden apreciar en la vista en SSAS, de lo contrario hay que crear las relaciones de forma manual. También se pueden hacer consultas por aparte, lo que viene siendo como una vista en SQL.

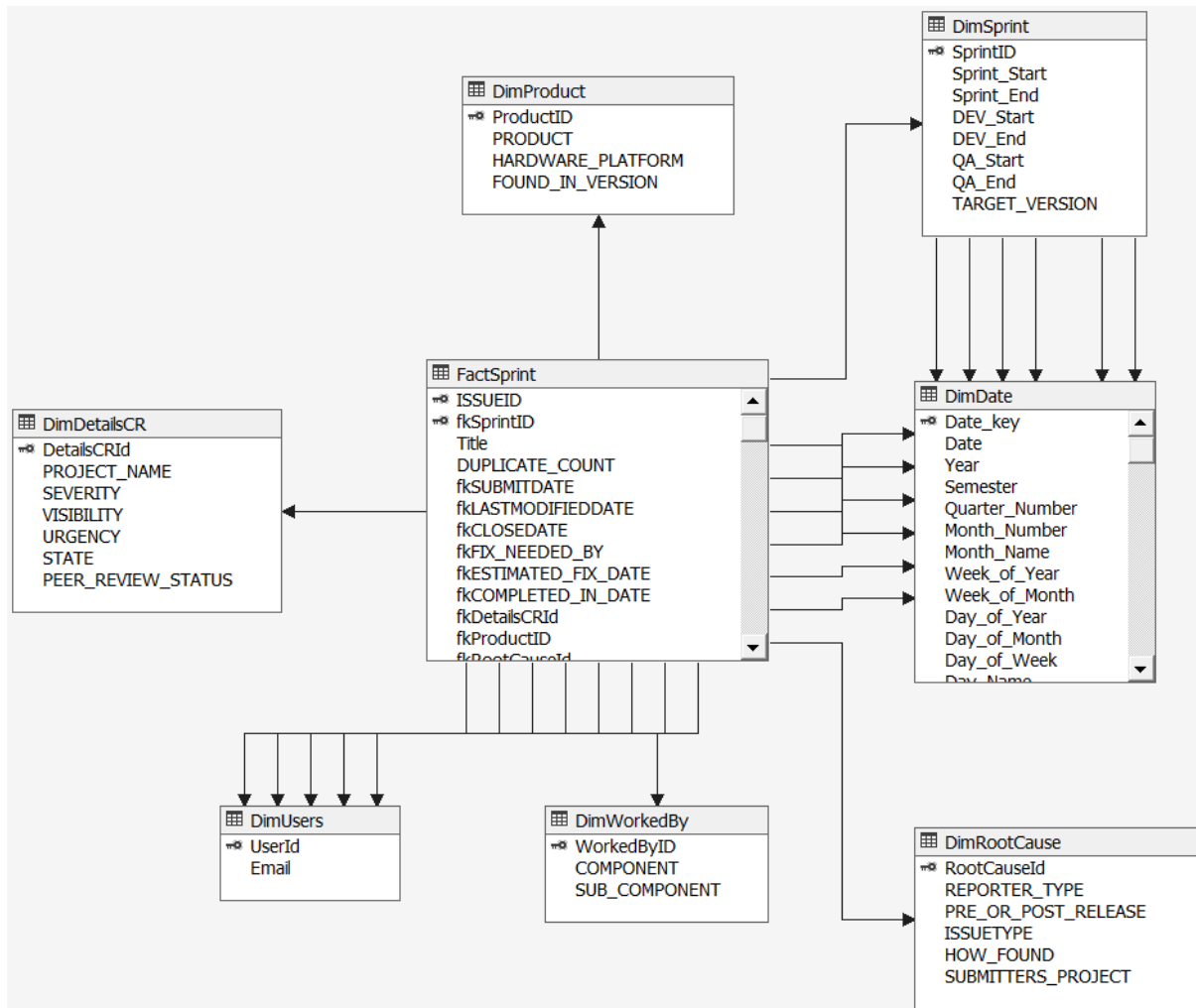


Ilustración 56 - Vista de fuente de datos SPRINT_ARC

La vista de la fuente de datos no es lo mismo que la vista del cubo, ya que no se espera que sean los mismos usuarios. En la fuente de datos, se sigue una nomenclatura más acorde al formato usado en bases de datos relacionales, mientras que en la vista del cubo se procura dar nombres de más fácil comprensión para usuarios de análisis y toma de decisiones del negocio.

Por ejemplo, el nombre de las dimensiones difiere del nombre de las tablas; además, en este caso no todos los datos de la tabla de hechos son para medidas, por lo que es necesario incluir la tabla como una dimensión más.

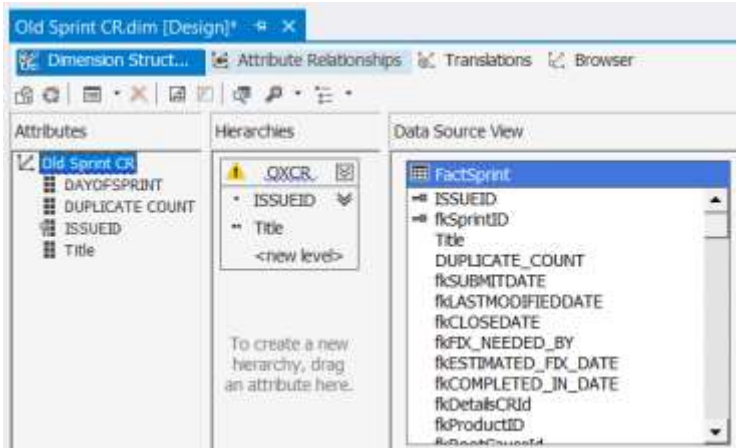


Ilustración 57 - Tabla de hechos como dimensión

Los atributos es lo que los usuarios podrán ver de la dimensión 'Old Sprint CR', mientras se esconden las llaves foráneas.

Otro ejemplo es la dimensión de fechas, donde los atributos se agrupan según la funcionalidad, en este caso lo que es la fecha en el calendario gregoriano y la fecha según el año fiscal.

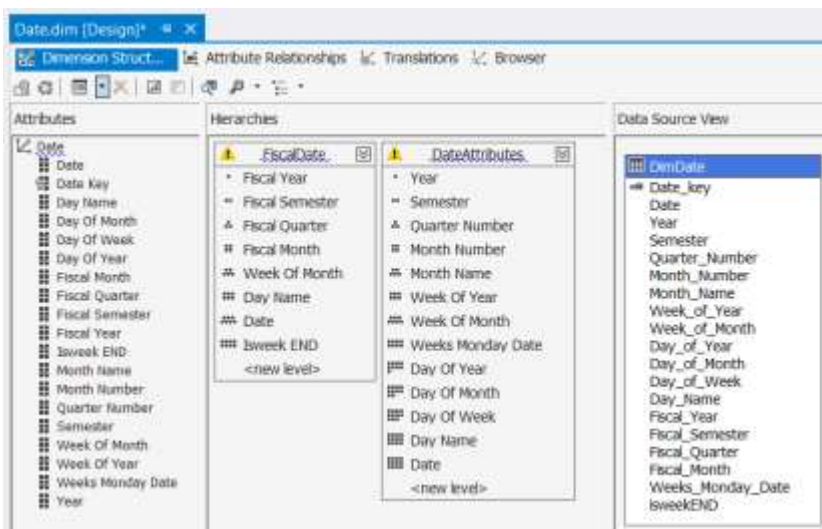


Ilustración 58 - Dimensión de fechas

d. Procesamiento del cubo SPRINT_ARC

El tipo de procesamiento que se escogió para este cubo es MOLAP, debido a que permite ejercer un mayor control de cuándo se hacen las actualizaciones del cubo.

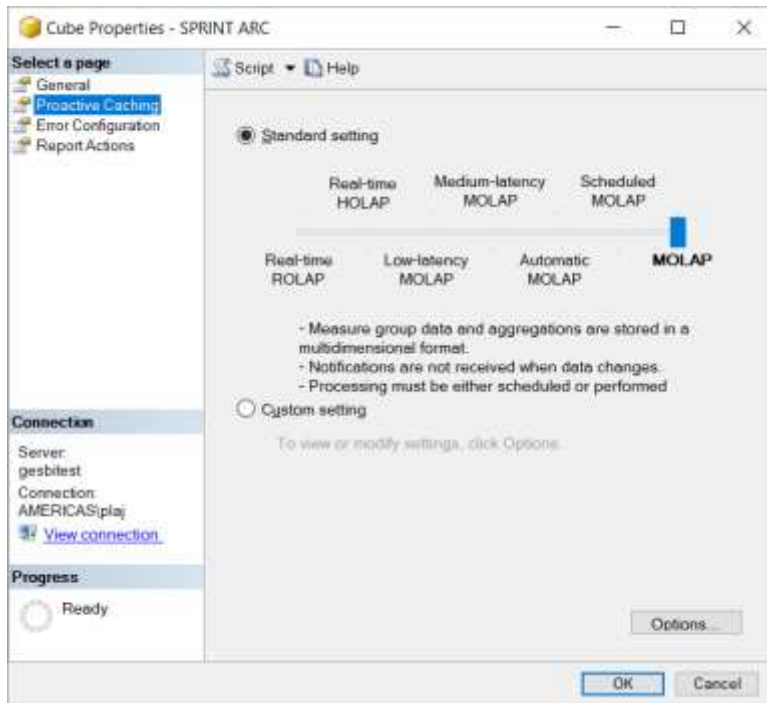


Ilustración 59 - Cubo SPRINT_ARC cache MOLAP

El procesamiento requiere de otro paquete de SSIS solo para procesamiento y que además esté calendarizado para garantizar una actualización en los datos del cubo.

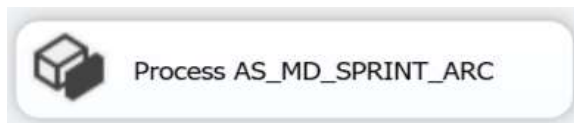


Ilustración 60 - SSAS tarea de procesamiento en SSIS

El paquete es muy simple, pues solo consta de una tarea cuya función es procesar el cubo en SSAS. Para ello, es necesario indicarle cuál es el servidor donde se encuentra corriendo SSAS y también cuál es la base de datos multidimensional.

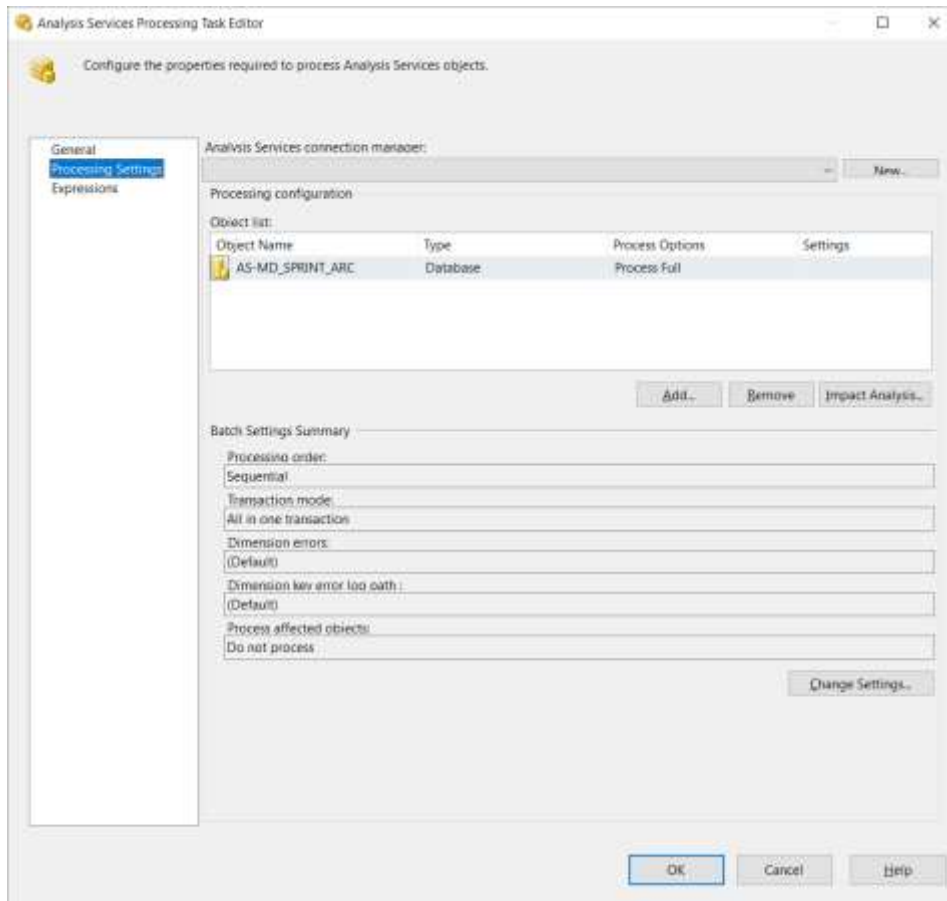


Ilustración 61 - Propiedades tarea de procesamiento del cubo

Posterior al paquete de procesamiento, viene la necesidad de procesarlo de manera calendarizada mediante un *JOB* en *SQL SERVER*.

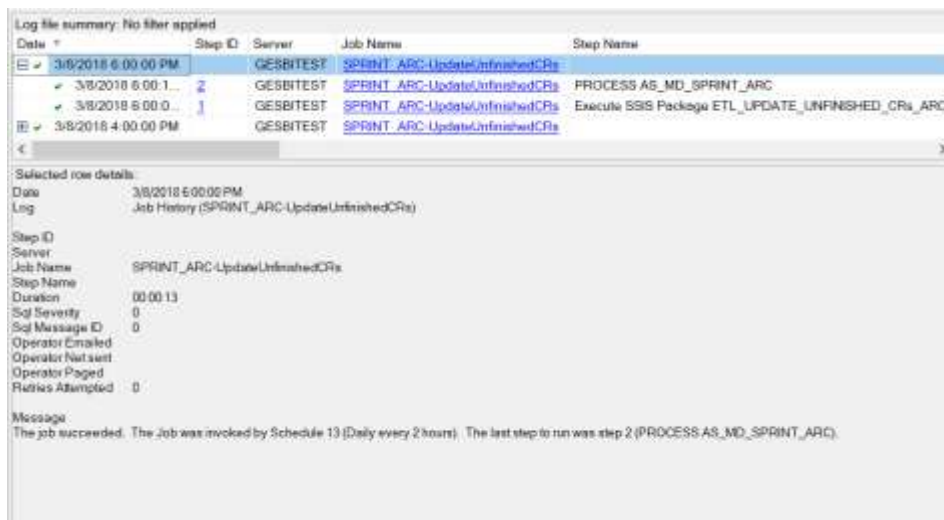


Ilustración 62 - Ejecución del JOB eliminar duplicados y actualizar SPRINT_ARC

Ap 14. Consulta SQL en Tableau para CR's completados

```

Edit Custom SQL
select
convert(date, convert(char(4), datepart(yy, closedate)) + '-' + convert(varchar(2), datepart(mm, closedate)) + '-01') Month
, STATE
, SUBMITDATE
, CLOSEDDATE
, {CLOSED_NO_CHANGE_IN_DATE}
, ESTIMATED_FIX_DATE
, DATEDIFF(dd, SUBMITDATE, CLOSEDDATE) Days2Complete
, ISNULL(DATEDIFF(dd,
CASE WHEN SUBMITDATE > ESTIMATED_FIX_DATE
then NULL
ELSE ESTIMATED_FIX_DATE
END
, CLOSEDDATE), 0) Commitment2EFD
, PROJECT_NAME
, PRODUCT
, ISSUEID
, TITLE
, FOUND_IN_VERSION
, HOW_FOUND
, CASE WHEN left(REPORTER_TYPE, 3) = 'SGI' THEN 'SGI' ELSE 'Non-SGI' END REPORTER_TYPE
, COMPONENT
, TARGET_VERSION
, LAB_REVIEW_ENGINEER LAB_REVIEWER
from BO_QX_CR bo
where
PROJECT_NAME = 'SGI TEST'
and PRODUCT in ('Sustaining Servers', 'Sustaining Storage')
and STATE in ('Completed', 'Closed-No Change')
and closedate > DATEADD(m, DATEDIFF(m, 0, GETDATE()) - 12, 0)
--order by Month, TARGET_VERSION, ISSUEID
Preview Results...  Insert Parameter
OK Cancel

```

Ilustración 63 - Consulta SQL en Tableau para CR's completados

Ap 15. Consulta seguimiento de CR's abiertos

```
select
STATE
, SUBMITDATE
, CLOSEDATE
, ESTIMATED_FIX_DATE
, ISNULL(DATEDIFF(dd,
CASE WHEN SUBMITDATE > ESTIMATED_FIX_DATE
then NULL
ELSE ESTIMATED_FIX_DATE
END
, GETDATE()),0) Commitment2EFD
, DATEDIFF(dd, SUBMITDATE, GETDATE()) DaysOpen
, PROJECT_NAME
, PRODUCT
, ISSUEID
, TITLE
, FOUND_IN_VERSION
, HOW_FOUND
, CASE WHEN left(REPORTER_TYPE,3) = 'SGI' THEN 'SGI' ELSE 'Non-SGI' END REPORTER_TYPE
, COMPONENT
, TARGET_VERSION
, LAB_REVIEW_ENGINEER LAB_REVIEWER
from BO_QX_CR bo
where
PROJECT_NAME = 'SGI TEST'
and PRODUCT in('Sustaining Servers', 'Sustaining Storage')
and STATE not in ('Completed', 'Closed-No Change', 'Duplicate')
```

Glosario

Administración de Datos: la definición oficial según DAMA (Data Management International Organization) es “La Administración de Datos es el desarrollo, ejecución y supervisión de planes, arquitecturas, políticas, programas y prácticas que controlen, protejan, entreguen y amplíen el valor de los datos y activos de información”.

BI: Business Intelligence en inglés, traducido a Inteligencia de Negocios en español. No hay que confundirlo con Minería de Datos, aunque ambos tienen un propósito de ayudar al negocio. La inteligencia de negocios está más relacionada con definir cuáles son los datos que se necesitan medir, después una limpieza y ETL, para seguir en un almacenamiento que facilite reportes del negocio. Una explicación más amplia se puede encontrar en el marco teórico “Descripción teórica de BI”.

Buffer: o ‘colchón’ en español, la idea es que el SM (Scrum Master) no comprometa desde la planificación del Sprint, la capacidad total del equipo. Los obstáculos no se perciben en un inicio y para afrontar este se necesita ese colchón, además para poder hacer frente a la volatilidad del Sprint.

Burndown: trata de un gráfico donde el eje X marca los días del Sprint y el eje Y, marca la cantidad de puntos que faltan por completar. Tiene además una línea de cuál sería el comportamiento idóneo el cual es la totalidad que falta por hacer el día 1 del Sprint y va disminuyendo día a día hasta llegar al último día sin ningún faltante.

CR: Change Record en inglés o registro de cambio en español, son parte de la fase de transición en ITIL, en este documento debido a interpretación por el negocio donde se le refiere como Change Request o solicitud de cambio en español (AXELOS, 2011). Para efectos de los descrito en este proyecto el control de CR’s se da por medio de la herramienta QUIX.

Data Mart(DM): el Mercado de Datos, proviene de la misma formación que el Data Warehouse, solo que su objetivo es cubrir áreas específicas, dos de las características principales que lo difieren están que los usuarios son limitados, y que está enfocado en manejar datos de solo cierta área operativa.

Data Warehouse(DW): el Almacén de Datos, consta, sobre todo, de un expediente completo de una organización, más allá de la información transaccional y operacional, almacenada en una base de datos diseñada para favorecer el análisis y la divulgación

eficiente de datos. El almacenamiento de los datos no debe usarse con datos de uso actual. La parte de DW 2.0 es una metodología diferente de implementación de BI, su autor es Bill Inmon, una explicación más gráfica se puede ver en Diagrama de Inmon DW 2.0. Además, existen dos versiones de DW 1.0, una de Ralph Kimball descrita en *Apéndice Ap 1* y otra con una variante en el orden, también de Bill Inmon en el *Apéndice Ap 2*.

ETL: Extraer, Transformar o modificar y Cargar (Extract, Transform, Load, en inglés); estos son los pasos claves para la unificación de datos. Un claro ejemplo, es el de la cédula de identidad; donde dependiendo de la fuente, esta se encuentra con guiones, o con ceros de más; al igual otro ejemplo común son los diferentes formatos para las fechas; de ahí el proceso de extraer, transformar en un formato estándar y cargar en una fuente de datos única.

GPE: Global Product Engineering en inglés, es Ingeniería Global de Producto, refiere a la organización que asigna responsables de productos dentro de la cadena de suministro, tienen relación con los equipos de investigación y desarrollo para los lanzamientos al mercado de nuevos productos, como también con la sostenibilidad de la calidad en productos que ya estén en el mercado y antes del fin del ciclo de vida de cada producto; aunque no están encargados del inventario de partes de su producto en cada fábrica, sí tienen una ligera relación con las órdenes de compra.

ITIL: Information Technology Infrastructure Library en inglés, es un conjunto de publicaciones de mejores prácticas para la gestión de servicios de TI. Divide el ciclo de vida del sector servicios en 5 fases: estrategia, diseño, transición, operación y mejora continua. Cada fase tiene una publicación de apoyo (en forma de libro). ITIL además ofrece diversas certificaciones para individuos, así como el ISO 20000-1:2011 para empresas. Actualmente la empresa inglesa AXELOS es la dueña de la metodología ITIL. (ISO, 2015)

OLAP: Procesamiento Analítico En Línea (On-Line Analytical Processing, en inglés), es una manera de agilizar el análisis de datos. Provee una manera distinta de ordenar los datos, de forma que facilite y vuelva menos pesada la consulta entre diferentes áreas, lo que se conoce como estructuras multidimensionales (o Cubos OLAP).

Product Backlog: cola del producto como se conoce en español, es la lista de requerimientos que el cliente ha solicitado y el Product Owner ha aceptado, tal lista se prioriza y según su importancia, necesidad y urgencia. El orden de esta lista ayuda a clasificar que es lo que irá en cada lanzamiento de producto (Release en inglés).

Product Owner(PO): en español es el Dueño del Producto, es el que le dice NO al cliente, es quien lleva el control de que entra en el Product Backlog, además es quien propone la primera versión de los requisitos que entrarán en el siguiente Sprint.

QUIX: QUality Information eXchange en inglés, es la herramienta usada para CR's donde se les da seguimiento a los cambios en productos. Es la parte de gestión de cambios de la herramienta de gestión de negocios de Serena, quienes ahora pertenecen a MicroFocus.

SAFe: Scaled Agile Framework por sus siglas en inglés, lo que se traduce en Marco Ágil Escalable, es una combinación de varias metodologías ágiles en conjunto con un poco de manejo de proyectos según el PMi (Project Management Institute, en inglés). La última versión es de Enero del 2016 y la mantiene la compañía Scaled Agile Inc.

Scrum: es la metodología ágil de ciclos de desarrollo con productos que evolucionan según cada ciclo, hasta llegar a la salida al mercado con el producto acordado. Los grupos de desarrollo se componen de los desarrolladores, el Scrum Master y el Product Owner que audite que lo que se hace concuerde con lo solicitado. Más detalle sobre esta metodología en la sección *Resumen de 'Guía de Scrum'*.

Scrum Master(SM): es el intermediario entre los desarrolladores y los otros interesados en el producto, sus tareas son llevar un control del progreso en la elaboración de los requerimientos acordados, saber cuál es el ritmo o velocidad de producción de su equipo, ayudar en la resolución de obstáculos que puede presentar cada requerimiento y además es el que comunica al PO (Product Owner) que es lo que se puede cumplir para cada Sprint. En otras palabras, es como el líder del equipo y procura que los desarrolladores se preocupen solo por desarrollar, procurando que existan el menor número de trabas posibles y comprometerse solo a lo que sabe se puede cumplir.

SGI: Servicios Globales de Ingeniería, es la organización dentro del ala de Cadena de Suministro que se encarga de la parte de calidad, desarrollan tanto las herramientas ejecutoras de las pruebas, como también las pruebas de calidad en sí. Fallos en las

herramientas o en las pruebas de calidad, repercuten en costos en la línea de producción, como también en la calidad del producto que llega al cliente.

Sprint: o ciclos en español, es un término usado en grupos de desarrollo de software, referente a un lapso corto de tiempo en donde se procura desarrollar los requerimientos acordados a un principio del ciclo. Cada ciclo puede durar entre 1 a 4 semanas.

Sprint Backlog: es la lista de requerimientos acordada para el Sprint.

User Story (User Stories, en plural): en documentos técnicos en español se sigue haciendo referencia al término en inglés. Son una forma simple y clara de documentar requerimientos, la idea es que sean cortos y entendibles tanto para el usuario a nivel de negocio, como para el desarrollador técnico.

Volatilidad: en una metodología ágil, nada está escrito en piedra, durante el Sprint algunos requerimientos podrían salir y otros podrían entrar. Entre más al inicio del Sprint el cambio tendrá menor impacto, entre más al final del Sprint es menos probable poder cumplirlo. La métrica de volatilidad procura mantener un nivel de aceptable, tolerable y riesgosa.

Validación: tomado del glosario de ITIL por AXELOS “(ITIL Transición del Servicio) Es una actividad que garantiza que un servicio de TI - nuevo o modificado, proceso, plan o cualquier otro entregable satisfaga las necesidades del negocio. La validación garantiza que se cumplan los requerimientos del negocio aunque estos pueden haber cambiado desde el diseño original. Véase también aceptación, calificación, validación y pruebas de servicio, verificación.” (AXELOS, 2011)

XP: eXtreme Programming en inglés, es una metodología ágil, conocida popularmente por sugerir la controversial programación en parejas. Como metodología ágil tiene algunas similitudes con Scrum. XP le gana a Scrum en diseño y calidad, pero Scrum en planificación. Más detalle sobre esta metodología en la sección *Resumen de 'El nuevo XP'*.

NOTA: Algunos de los términos relacionados a BI usados anteriormente se tomaron de manera directa de la investigación realizada en un anterior proyecto final de graduación (Plá Rojas, 2013, págs. 19-25).

Acrónimos

Tabla 6 - Lista de acrónimos

Acrónimo	Significado	
BI	Business Intelligence	Inglés
	Inteligencia de Negocios	Español
CR	Change Record	En inglés según ITIL
	Registro de Cambio	En español según ITIL
	Change Request	En inglés término dado en el negocio
	Solicitud de Cambio	En español término dado en el negocio
DM	Data Mart	Inglés
	Mercado de Datos	Español
DW	Data Warehouse	Inglés
	Almacén de Datos	Español
ETL	Extract, Transform, Load	Inglés
	Extraer, Transformar, Cargar	Español
GPE	Global Product Engineering	Inglés
	Ingeniería Global de Productos	Español
ITIL	Information Technology Infrastructure Library	Inglés
	Biblioteca de infraestructura de tecnología de la información	Español
OLAP	On-Line Analytical Processing	Inglés
	Procesamiento Analítico En Línea	Español
PE	Product Engineer	Inglés
	Ingeniero del Producto	Español
PO	Product Owner	Inglés
	Dueño del Producto	Español
QUIX	QUality Information eXchange	Inglés
	Intercambio de información de calidad	Español
SAFe	Scaled Agile Framework	Inglés
	Marco Ágil Escalable	Español
SM	Scrum Master	Inglés
	Maestro Scrum	Español
SGI	Servicios Globales de Ingeniería	Español
XP	eXtreme Programming	Inglés
	Programación eXtrema	Español