



FACULTAD DE INFORMÁTICA

TESINA DE LICENCIATURA

TÍTULO: Integrando los procesos de negocio con Internet de las Cosas

AUTORES: Luis Damian Candia (8173/1)

DIRECTOR: Dra. Patricia Bazán

CODIRECTOR: C.C Viviana Ambrosi

ASESOR PROFESIONAL: Lic. Anahi Rodriguez

CARRERA: Licenciatura en Sistemas

Resumen

En la actualidad, es necesario pensar la forma para modernizar la matriz industrial de la república argentina, sin que esto genere un fuerte impacto en los costos operativos. En el presente trabajo presentaremos una solución que va a permitir a una fábrica poder dar el primer salto hacia la revolución industrial 4.0. Para poder realizarlo la principal motivación de este trabajo es presentar un middleware que permita integrar tanto la tecnología de internet de las cosas (IoT) como las plataformas de gestión de procesos de negocio (BPMS) que ya existen y funcionan exitosamente en muchas empresas.

Palabras Clave

*Internet de las cosas
Procesos de negocio
Industria 4.0
Middleware*

Conclusiones

La solución propuesta permite a las empresas que, sin tener la necesidad de comprar maquinaria industrial nueva, puedan a un muy bajo costo, simplemente adaptando el equipamiento que ya cuentan, medir en tiempo real los valores de su producción.

Por otra parte, este trabajo permite contar con un sistema de gestión que pueda medir en tiempo real sus indicadores principales de producción. El enfoque de solución por procesos de negocio permite a las organizaciones mejorar la eficiencia de su gestión y la toma de decisiones, en el marco de la industria 4.0.

Trabajos Realizados

Estudio de ventajas y desventajas que cuentan con cada uno de los modelos de integración planteados hasta la actualidad.

Desarrollo de un mecanismo de integración entre Bonita Open Solution y una red dispositivos de IoT.

Implementación de un caso de estudio en una fábrica de bolsas de plástico de la ciudad de Buenos Aires, en el cual se modernizo su maquinaria industrial mediante la colocación de sensores y placas Raspberry para medir su producción en tiempo real y la posterior integración con Bonita Open Solution utilizando el mecanismo de integración propuesto anteriormente.

Trabajos Futuros

Implementación para otros protocolos de comunicación diferentes de API REST, como por ejemplo MQTT.

Extender la capa de middleware proveyendo distintos niveles de seguridad y confiabilidad.

Agregar a la solución diferentes tipos de sensores que midan indicadores de máquinas industriales.

Evaluar implementaciones con otros BPMS

Optimizar el despliegue de la solución con componentes de Docker.

Agradecimientos

Quisiera agradecer a Dios, quien me guía y me acompaña en cada paso de mi vida, a mi esposa Vanina quien con su amor y apoyo me dio fuerzas para poder terminar esta etapa que es muy importante para mí, también le dedico este logro a mis dos hijos Julia y Augusto.

A mi familia mis papas y mis hermanos que me ayudaron y estuvieron conmigo desde el primer día.

A Patricia Bazán, Viviana Ambrosi y Anahí Rodríguez, por su orientación, paciencia y buena predisposición que me ayudaron a realizar este trabajo.

Mis compañeros y amigos de trabajo, a los que conocí desde el primer día que pisé la facultad, a los que se fueron sumando con los años y fueron una parte muy importante de este camino.

Facultad de Informática y a la Universidad Nacional de La Plata que me dieron la oportunidad de estudiar esta carrera de forma completamente gratuita y ahora también de tener la oportunidad de desarrollarme como docente.

Índice General

Capítulo 1. Introducción.....	6
1.1 Contexto.....	6
1.2 Objetivo.....	6
1.3 IoT y su vínculo con la Industria 4.0	7
1.4 Procesos de negocio (BPM).....	8
1.5. Organización del Documento.....	9
Capítulo 2. BPM.....	10
2.1 Definiciones y generalidades.....	10
2.2 Ciclo de vida de un proceso de negocio.....	11
2.3 Diagrama de proceso de negocio (BPD).....	13
2.5 BPMS (Business Process Management System).....	19
2.6 Software Bonita Open Solution.....	20
Capítulo 3. Internet de las Cosas.....	22
3.1 Definiciones y generalidades.....	22
3.2 Elementos de IoT.....	24
3.3 Protocolos para soluciones IoT	26
Capítulo 4. Integración BPM e IoT.....	36
4.1 Soluciones basadas en procesos con IoT	37
4.2 Arquitectura de soluciones basadas en procesos con IoT	40
4.3 Modelado de soluciones basadas en procesos con IoT	42
4.4 Ejecución de soluciones basadas en procesos con IoT	47
4.5 Monitoreo de soluciones basadas en procesos con IoT	50
4.6 Otros aspectos a considerar en una solución BPM-IoT.....	50
Capítulo 5. Caso de Estudio.....	53
5.1 Arquitectura	54
5.2 Capa de aplicación. Modelado	58
5.3 Capa de comunicación	59
5.4 Pasos del proceso durante la capa de comunicación.....	61
5.4 Topología de red	63
5.6 Capa de middleware.....	64

5.7 Factores que podrían perjudicar la eficiencia del sistema	65
Capítulo 6. Conclusiones y Trabajos Futuros	66
6.1 - Trabajos futuros	66
6.2 - Contribuciones	67
Referencias.....	69
Anexo 1- Reportes del sistema.....	73

Índice de Figuras

Fig. 1 - Ciclo de vida clásico de procesos de negocio	11
Fig. 2 - Ciclo de vida de los procesos de negocio según M.Weske [5].	12
Fig. 3 - Flujo en la definición de un proceso de negocio	13
Fig. 4 – Posibles estados de un proceso de negocio	14
Fig. 5 – Elementos de un BPD.....	14
Fig. 6 – Evento de inicio de una instancia de proceso de negocio.....	17
Fig. 7 - Evento de inicio de una instancia ante la recepción de un mensaje.....	17
Fig. 8 - Evento de inicio de una instancia ante la ocurrencia de un determinado plazo temporal... ..	17
Fig. 9 - Subproceso de “Fabricación de autopartes” dentro del proceso de “Fabricación de un automóvil”	18
Fig. 10 – Arquitectura de las soluciones Bonita Soft.....	21
Fig. 11 – Comparación de arquitectura IoT con TCP/IP	24
Fig 12 – Arquitectura OPC	30
Fig 13 – Arquitectura MQTT	30
Fig 14 – Arquitectura DDS	31
Fig 15 – Arquitectura REST	34
Fig 16 – Arquitectura SOAP	34
Fig 17 – Arquitectura REST/JSON	35
Fig. 18 – Solución SOA.....	38
Fig 19 – Ciclo de vida de un proceso de IoT	39
Fig 20 – Composición de servicios.....	41
Fig 21 – Orquestación de procesos	41
Fig 22 – Coreografía de procesos	42
Fig 23 – Anotaciones BPM-IOT versión proyecto IoT-A	43
Fig 24 – Anotaciones versión del proyecto MakeSense.....	44

Fig 25 – Posibles modelos distribuidos	49
Fig 26 – Ciclo de producción de la fábrica.....	54
Fig 27 – Ejemplo de capa de middleware	56
Fig 28 – Arquitectura de capas planteada en el trabajo	57
Fig 29 – BPD – Tablero de control	59
Fig 30 – Raspberry instalada en una de las maquinas.....	60
Fig 31 – Servidor instalado en una de las computadoras de la fabrica	60
Fig 32 – Diagrama de conexión del sensor de efecto Hall	61
Fig 33 – Diagrama de pines GPIO - Raspberry.....	62
Fig 34 – Código fuente de lectura del sensor de efecto Hall.....	63
Fig 35 – Registro de tópicos	64
Fig 36 – Ejemplo de método de la API REST de la capa de middleware	65

Índice de Tablas

Tabla 1. Elementos de BPMN	15
Tabla 2. Protocolos IoT según el modelo OSI.....	27
Tabla 3- Ventajas y desventajas de los protocolos utilizados	33
Tabla 4. Propuestas de notaciones BPM-IoT [41]	46

Capítulo 1. Introducción

1.1 Contexto

La adopción de tecnología dentro de las organizaciones, muchas veces puede parecer un gasto innecesario, pero es una acción que puede aumentar considerablemente la productividad y competitividad de las mismas, haciendo más eficientes sus procesos, generando nuevos negocios para las empresas y favoreciendo la creación de nuevos puestos de trabajo.

Sin embargo, incluir nuevas tecnologías depende, en gran parte, de la posibilidad económica que cuente cada una de las empresas para poder invertir y que sean capaces de absorber en sus costos la modificación tecnológica de su matriz de producción.

Los gobiernos del mundo diseñan e implementan instrumentos e incentivos para potenciar la economía del conocimiento y acompañar a las empresas en la adopción de nuevas tecnologías, pero estas medidas muchas veces no son suficientes en países con economías frágiles como la Argentina.

1.2 Objetivo

El objetivo del presente trabajo es mostrar un caso de estudio en el cual se implementó una red de sensores de IoT con un *backend* - módulo privado de administración - basado en un BPMS (*Business Process Management System*) que permite gestionar una fábrica, adaptando con nuevas tecnologías toda la maquinaria de la planta para poder medir en tiempo real la producción de la misma, integrando ambas soluciones de un extremo al otro generando el concepto de BPMS-IoT.

Considerando este objetivo y la necesidad de integrar IoT con BPM, se desprenden distintas líneas de trabajo u objetivos específicos: 1- Analizar las debilidades y fortalezas de la integración de IoT en una solución concreta de BPM, 2- Utilización de herramientas de *hardware* y *software* libre para realizar el despliegue de soluciones de IoT e integrarlas con máquinas y sistemas ya existentes en las industrias y 3 - Adaptar los elementos con los que ya cuenta la industria a las nuevas tecnologías; esto permite una reutilización que no solamente favorece económicamente a las empresas, sino que también reduce el impacto de descarte de las mismas.

El escenario que plantea la Industria 4.0 amerita analizar una solución que considere distintas capas de abstracción y un modelo arquitectónico orientado a servicios [1]

Durante este trabajo, presentaremos una solución económica que permita a las empresas industriales poder integrar su línea de producción a sus sistemas de gestión de la información, sin necesidad de comprar equipos nuevos de producción, sino que adaptando e integrando su tecnología existente.

El marco teórico fue probado en una empresa de la industria de plástico de argentina, dichas pruebas serán explicadas en el capítulo de caso de estudio del presente trabajo.

En este sentido resulta apropiado interiorizarse de las ideas subyacentes en IoT (*Internet of Things*), con su correlato en la Industria 4.0 y analizarlas bajo la lupa de la gestión de procesos de negocio (BPM - *Business Process Management*) y una posible integración tecnológica entre ellas.

1.3 IoT y su vínculo con la Industria 4.0

En la era de Internet de las Cosas (IoT – *Internet of Things*) se calcula que para el año 2025 existirán más 26 billones de dispositivos conectados incluyendo casas, teléfonos, autos y fábricas. En este contexto es imposible imaginar que los procesos de las organizaciones no deban adaptarse a la nueva realidad, en la cual la mayoría de las actividades serán realizadas por autómatas que deberán tomar decisiones en base a la información obtenida por su red de sensores.

IoT es un fenómeno tecnológico bastante reciente. El término fue introducido por Kevin Ashton [2], investigador en el campo de la identificación por radiofrecuencia (RFID). Este autor plantea el debate sobre qué ocurriría si absolutamente todos los objetos que nos rodean nos brindarían información útil al instante. Esto cambiaría por completo nuestra forma de desenvolvernarnos con el mundo que nos rodea. Las computadoras podrían recopilar información sobre los objetos físicos sin intervención humana mediante el uso de tecnología de sensores y actuadores.

Los procesos industriales, por su parte, han atravesado tres grandes transformaciones tecnológicas a lo largo de la historia, las primeras revoluciones industriales como: 1- la creación de la máquina de vapor, 2- la electricidad y 3- la automatización, causaron un profundo impacto en la sociedad luego de su implementación. La cuarta revolución industrial llega de la mano de la tecnología [3].

La Industria 4.0 se refiere a un nuevo modelo de organización y de control de la cadena de valor a través del ciclo de vida del producto apoyado por las tecnologías de la información. En definitiva, se trata de la aplicación del modelo IoT, a la industria.

A diferencia de las tres revoluciones industriales anteriores, la cuarta revolución industrial se generó primero en la sociedad como un cambio cultural y en la actualidad la industria intenta alcanzar el mismo nivel de tecnología con el cual viven las personas en sus hogares. El concepto de “Industria 4.0” fue nombrado por primera vez en Alemania en el año 2010. Esta nueva ola de tecnologías, que hoy avanza en diversas regiones del mundo a distintas velocidades, impone transformación a partir de nuevas tecnologías industriales con foco digital, con potencial de crear fábricas con procesos productivos totalmente integrados y automatizados, permitiendo que diferentes sistemas interactúen analizando información en tiempo real para optimizar la producción,

predecir fallas e integrar las cadenas de suministros para volverlas más eficientes

El impacto en los procesos productivos será significativo, volviéndose más flexibles, eficientes y veloces, con importantes implicancias para los nuevos empleos que demandarán las empresas.

En esta transformación de los modelos organizacionales se identifican, como los más importantes aquellos relacionados con fábricas inteligentes, control de la producción y diseño de fábricas digitales, gestión del desempeño en tiempo real, tableros electrónicos de desempeño y mantenimiento predictivo.

Algunas de las tecnologías claves de esta transformación impuesta por la Industria 4.0 son:

- Robótica: robots industriales autónomos y cooperativos.
- Internet industrial: red de máquinas y productos con comunicación multidireccional entre objetos de la red.
- Simulación: simulación y optimización de redes de valor basada en información en tiempo real procesada por sistemas integrados.
- *Cloud* y ciberseguridad: administración de grandes volúmenes de datos en sistemas abiertos con alto nivel de interacción entre máquinas inteligentes, productos y sistemas que exige altos requisitos de seguridad.
- Manufactura aditiva: impresión 3D particularmente para prototipos y repuestos y para reducir envíos e inventario.
- Realidad aumentada: realidad aumentada para mantenimiento, logística y procesos operativos, Información de apoyo, ej.: mediante lentes.
- *Big Data* y analítica: análisis integral de la información disponible (ej.: ERP, SCM, CRM e información de robots productivos para apoyo a toma de decisiones en tiempo real.

La industria 4.0 es un concepto que forma parte de la agenda de las empresas de todo el mundo y las compañías argentinas no son una excepción. Así, en octubre de 2017 la Secretaría de la Transformación Productiva del Ministerio de Producción de Argentina, realizó una encuesta online a 78 ejecutivos de 66 grandes empresas industriales argentinas de diferentes sectores económicos. El objetivo era conocer el progreso en la implementación de nuevas tecnologías en sus procesos productivos y los desafíos más importantes que encuentran a la hora de ejecutar esta transformación y el resultado fue que solamente el 30 % de la industria de nuestro país había implementado alguna solución referida con la Industria 4.0, en la actualidad a pesar de no contar con una estadística oficial, seguramente ese número se redujo más producto de la pandemia de COVID 19 que sufrió el mundo [4].

1.4 Procesos de negocio (BPM)

La gestión de procesos de negocio (BPM - *Business Process Management*) es un campo de la gestión de operaciones que se centra en mejorar el desempeño

corporativo mediante la gestión y optimización de los procesos de negocio de una empresa. Por tanto, se puede conceptualizar como una forma de "optimización de procesos" [5].

Se argumenta que BPM permite a las organizaciones ser más eficientes, más efectivas y más capaces de cambiar que un enfoque funcional, enfoque tradicional de gestión jerárquica. Estos procesos pueden afectar el costo y la generación de ingresos de una organización.

Weske [5] define BPM, como la representación de las actividades de un proceso de negocio y sus restricciones de ejecución entre ellas, y que incluye conceptos, métodos y técnicas para apoyar su ciclo de vida

BPM facilita la gestión de los servicios que ofrece una organización. Tiene como objetivo describir la actividad empresarial como un conjunto de procesos, y ayudar a la ejecución de los mismos, tanto dentro de la organización como desde organizaciones externas (clientes o proveedores). El modelado de procesos de negocio se puede llevar adelante usando herramientas, describiendo la coordinación de servicios, las restricciones de los procedimientos y el flujo de trabajo que involucran las diferentes tareas, entrada y salida.

Este resultado demuestra la necesidad de pensar soluciones que sean económicas y de rápida implementación para que nuestra industria nacional no pierda competitividad.

En este sentido, integrar IoT en una solución BPM resulta un enfoque donde se puede poner en valor la aplicación de BPM, considerando que conlleva un ciclo de vida donde sus etapas integran un ciclo de mejora continua para modelado, despliegue, ejecución y monitoreo de los procesos de negocio.

1.5. Organización del Documento

EL documento se organiza de la siguiente manera:

- En el capítulo 2 estudiaremos que son y cuáles son los elementos principales de los procesos de negocio y cuál es su importancia dentro de las organizaciones.
- En el capítulo 3 estudiaremos el estado del arte de IoT y cuáles son sus protocolos principales.
- En el capítulo 4 estudiaremos posibles integraciones entre IoT y BPM analizando las ventajas y desventajas de cada una.
- En el capítulo 5 presentaremos un caso de estudio real en el cual implementaremos una integración de ambas tecnologías en una planta industrial.

Capítulo 2. BPM

En este capítulo se hará un repaso conceptual sobre que es un proceso de negocio y sus principios básicos, detallando el ciclo de vida que cumplen los mismos como así también una posible clasificación dependiendo de diversos criterios. El objetivo es brindar un marco teórico sobre este componente esencial para lograr el modelo de integración propuesto en este trabajo. Finalmente se veremos conceptos relacionados con la notación BPMN y los diagramas BPD que definen el flujo de ejecución de los BPM.

2.1 Definiciones y generalidades

En la evolución de las organizaciones hacia el paradigma orientado a procesos, se rompió la estructura de pensar al trabajo diario de una organización como un conjunto de tareas individuales, para pasar a imaginar a las actividades como un mapa de tareas que están completamente relacionadas entre sí, mediante la información que reciben y la que generan luego de procesarla [5].

Este cambio de enfoque en el modo de diseñar e implementar soluciones de sistemas de información, llevó a tener que tomar las siguientes consideraciones:

- Explicitar el conocimiento de un proceso de negocio ayudando a documentarlo, a definirlo y a implementarlo.
- Proveer la interoperabilidad de las soluciones de sistemas dentro de una organización.
- Resolver la dinámica de los problemas en términos declarativos y cubriendo todas las etapas del ciclo de vida del software.

BPM (*Business Process Management*) ha adquirido una atención considerable desde hace algunos años, tanto por las comunidades de administración de negocios como las de ciencia de la computación.

La gestión de procesos de negocio (BPM) incluye conceptos, métodos y técnicas para ayudar con el diseño, la administración, configuración, publicación y análisis de los procesos de negocio. La base de BPM es la representación explícita de los procesos de negocio con sus actividades y las restricciones de ejecución entre ellos.

Para poder implementar una solución orientada a procesos cada organización deberá definir claramente las áreas que la componen y las tareas que son responsabilidad de cada una, junto con sus restricciones. Luego de definir los puntos anteriores la organización tendrá una visión completa de todo el mapa de procesos de la organización y su trazabilidad.

Implementar una solución orientada a procesos nos permitirá mejorar nuestra efectividad y eficiencia concretando cada una de nuestras actividades de forma consistente, aumentando la productividad de nuestras tareas y generando ahorros de tiempo (dinero), además de permitirnos dar un salto de calidad significativo.

Para los analistas de software su tarea también se verá simplificada porque les va a permitir agilizar el proceso de sus desarrollos, identificar errores de forma más simple y contar con un mayor nivel de abstracción lo cual traerá mucha independencia de plataformas tecnológicas concretas.

BPM tiene principios básicos sobre los que se construye como disciplina, a continuación, veremos en detalle dichos principios:

1. El principio básico de BPM es que los procesos son bienes que producen valor para los clientes. Un proceso tiene clientes internos o externos y son los receptores de los resultados creados por los procesos.

2. El segundo principio es que los procesos deben ser administrados y mejorados continuamente. Como los procesos son bienes, los procesos principales y los procesos que generan mayor valor para los clientes, deben ser administrados cuidadosamente.

3. La administración de procesos implica la tarea de medir, monitorear, controlar y analizar los procesos de negocio.

Estas tres tareas van de la mano en un ciclo continuo, porque medir los procesos de negocio provee información, dicha información permite a las organizaciones predecir, reconocer y diagnosticar deficiencias del proceso y poder definir futuras mejoras.

2.2 Ciclo de vida de un proceso de negocio

Las Fig. 1 y 2, presentan dos variantes de ciclo de vida de los procesos: el clásico y el de Weske. La diferencia radica que el primero se centra en la mejora continua mientras que el segundo, pone el foco en la administración del ciclo de vida, lo cual adhiere mejor a las soluciones tecnológicas de BPM. El ciclo de vida clásico puede entenderse de manera conceptual, sin intervención de la tecnología.

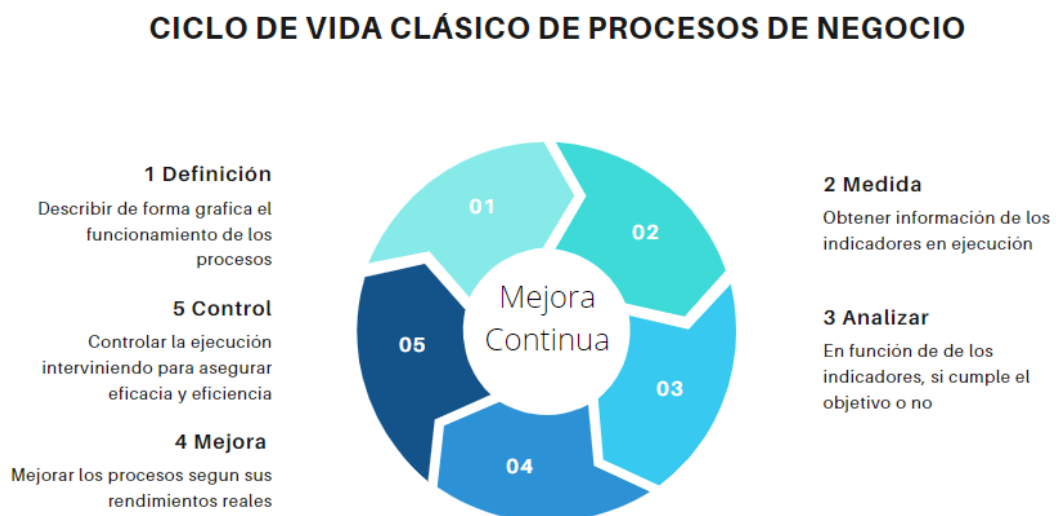


Fig. 1 - Ciclo de vida clásico de procesos de negocio

CICLO DE VIDA CLÁSICO DE PROCESOS DE NEGOCIO

Versión de Weske

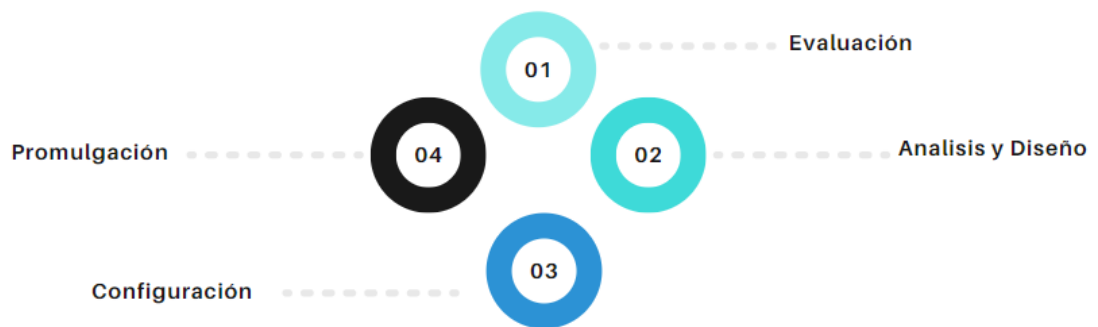


Fig. 2 - Ciclo de vida de los procesos de negocio según M.Weske [5].

Analizar información de un proceso es un paso esencial para identificar lo que los procesos necesitan mejorar, y cuáles son las mejoras que traerán mayor valor al proceso.

Los procesos de negocio son la clave para organizar estas actividades, pero ¿Qué son los procesos de negocio y por qué son tan importantes para las organizaciones?

Weske [5] define a un **proceso de negocio** como un conjunto de **actividades** que se realizan en coordinación en un ambiente organizacional y técnico. Estas actividades alcanzan el objetivo del negocio. Cada proceso de negocio representa una única organización, pudiendo interactuar con otras.

Básicamente un **proceso** es un conjunto de tareas que cuentan con un principio y un final bien definidos, que aceptan entradas de información que son procesadas por sus actividades internas y generan salidas que pueden ser consumidas por uno o más procesos.

Un **mapa de procesos**, representa una visión de los principales procesos de la organización y cómo se conectan atravesando absolutamente todas las áreas de nuestra organización e incluso en muchos casos como es su comunicación con agentes externos, como pueden ser otras empresas o incluso nuestros clientes. Es una representación gráfica de la organización en términos de su funcionamiento.

Un proceso de negocio cuenta con un **propietario** y respeta reglas de negocio. El propietario tiene la responsabilidad de realizar diseñar y ejecutar el proceso completo de extremo a extremo, siendo los actores o participantes quienes realizan las tareas interactuando entre sí.

Las **reglas de negocio** son definidas por los analistas y representan las condiciones para ejecutar con éxito cada tarea.

En una implementación orientada a procesos resulta clave monitorizar los indicadores que se definieron previamente para poder medir el grado de eficiencia y eficacia que cuentan.

En la Fig. 3 se muestra un resumen del flujo que debe cumplir un proceso.

FLUJO DE UN PROCESO DE NEGOCIO

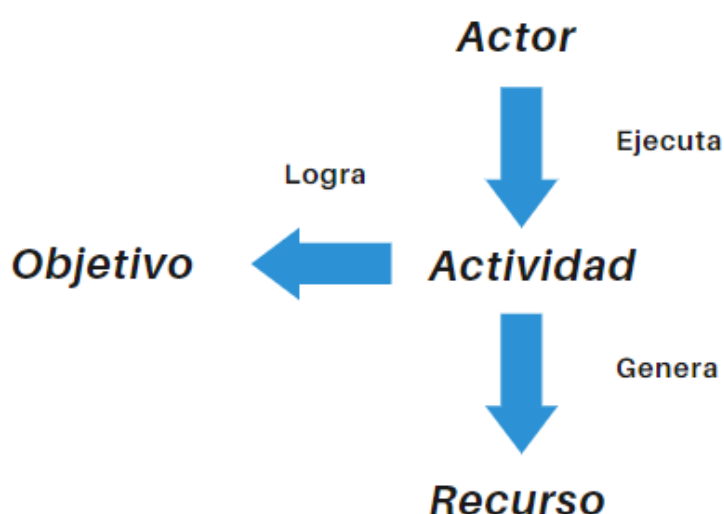


Fig. 3 - Flujo en la definición de un proceso de negocio

Un actor **ejecuta** una actividad, esa actividad está diagramada para **lograr** un **objetivo**, a su vez la actividad **genera** un recurso.

Cuando un proceso se ejecuta se crea una instancia de ese proceso, la cual cuenta con una identificación única para facilitar su monitoreo.

Cada instancia cuenta con diferentes estados posibles como se observa en la Figura 4.

2.3 Diagrama de proceso de negocio (BPD)

Para poder definir un proceso de negocios debemos realizar un diagrama que lo represente gráficamente, a ese diagrama comúnmente se lo llama BPD y cuenta los siguientes elementos

- Flujos de entrada y salida (eventos y resultados)
- Actividades
- Secuencias
- Puntos de decisión
- Participantes

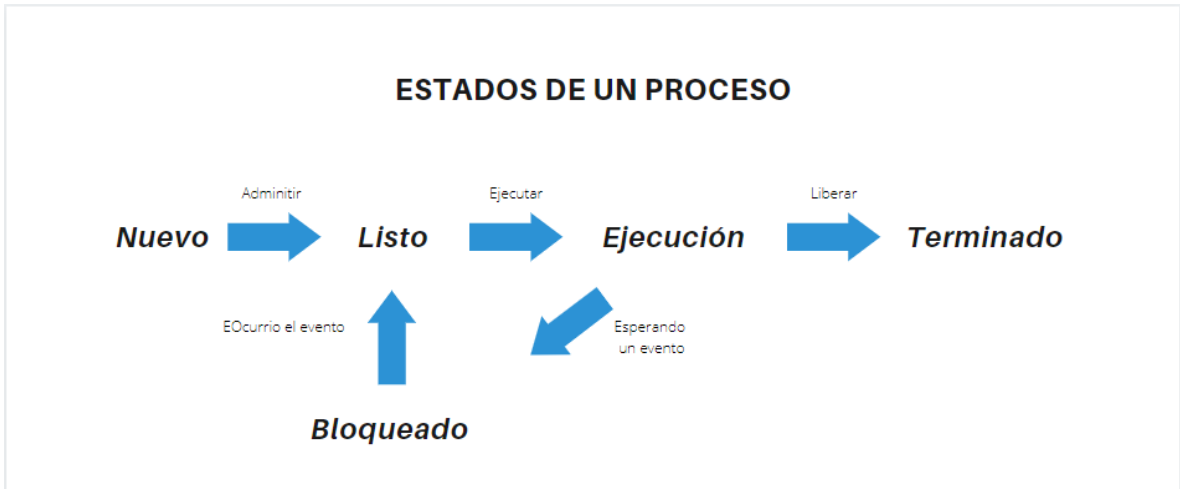


Fig. 4 – Posibles estados de un proceso de negocio

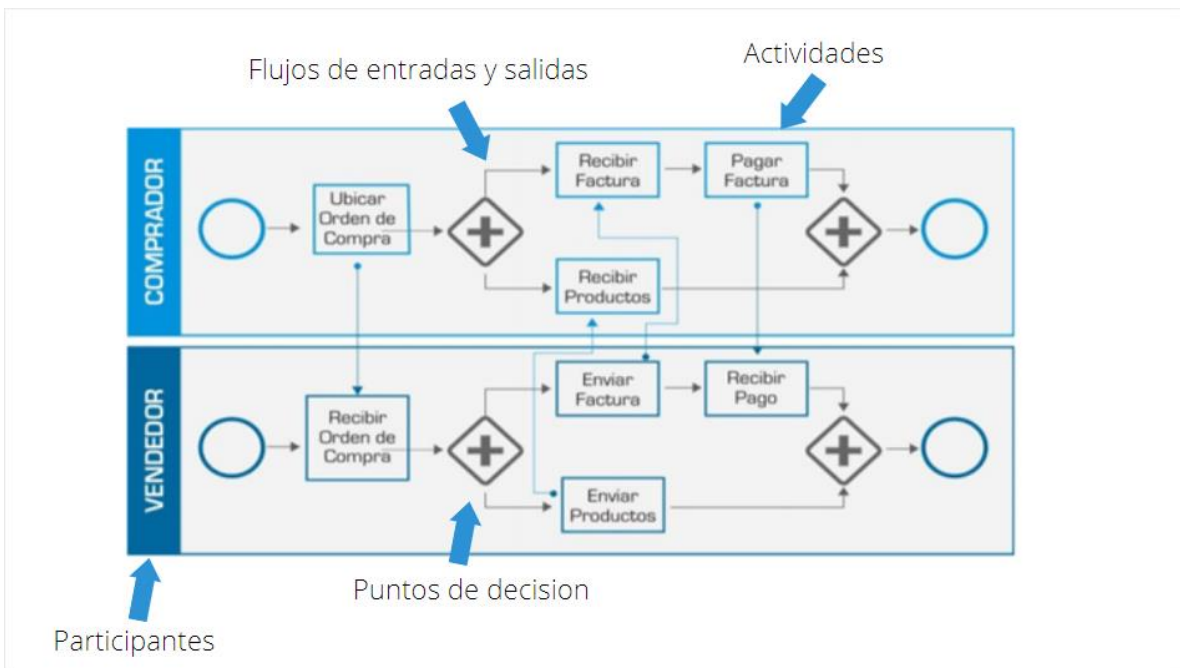


Fig. 5 – Elementos de un BPD

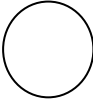

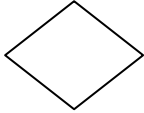

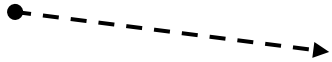
2.4 Elementos de la notación BPMN

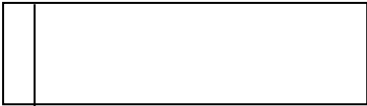
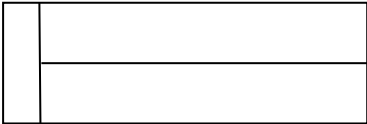



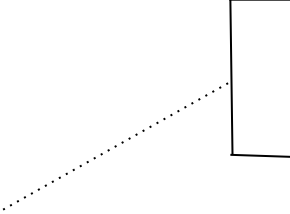
BPM como metodología posee una notación para expresar los diagramas que surgen luego de la actividad de análisis. Dicha notación (BPMN - *Business Process Management Notation*) [6] posee una serie de componentes con una semántica específica, permitiendo diagramar procesos de negocio (BPD). BPMN es una notación estándar desarrollada por *Object Management Group* (OMG) [6], el objetivo principal de BPMN fue proveer una notación estándar

legible para los usuarios de negocios, reduciendo la brecha entre el diseño y la implementación del proceso de negocio.

A través de la notación BPMN permite desarrollar diagramas de procesos con un conjunto de elementos gráficos, en la tabla 1 se observan los elementos básicos de BPMN [6]:

Tabla 1. Elementos de BPMN

Elemento	Descripción	Notación
Evento	Es algo que sucede durante la ejecución de un proceso. Afectan el flujo del proceso, suelen tener una causa o un impacto. Existen 3 tipos de eventos: comienzo, intermedio o fin.	
Actividad	Son las actividades que se realizan dentro del organismo. Pueden ser atómicas (tareas) o compuestas (subprocesos)	
Compuertas	Se utiliza para controlar la divergencia o convergencia de los flujos. Los marcadores internos indicarán el tipo de control de la conducta.	
Flujo de secuencia	Se utiliza para mostrar el orden en que las actividades se realizarán en un proceso y en una coreografía.	
Flujo de mensajes	Se utiliza para mostrar el flujo de mensajes entre dos participantes.	

Asociación	Se utiliza para vincular información y artefactos con elementos gráficos BPMN➔
Pool	Representa un participante en un proceso	
Lane	Son una subpartición dentro de un proceso. Los Lanes se utilizan para organizar y categorizar actividades.	
Objeto de datos	Se utilizan para mostrar cómo se utilizan los datos en las actividades (requeridos o producidos)	
Mensajes	Se utiliza para representar la comunicación entre dos Participantes	
Grupos	Se utiliza para agrupar por categorías elementos del diagrama.	
Anotaciones de texto	Se utilizan para agregar texto informativo al gráfico.	

Los **eventos** pueden ser de tres tipos diferentes de **inicio, intermedios y de fin**. A su vez los **eventos de inicio**, permiten la iniciación de una instancia de un proceso, se dividen en los siguientes sub-tipos de eventos inicio e inicio por recepción de mensajes.

También existen otros eventos de inicio que no todos los BPMS los implementan como, por ejemplo, por recepción de señal (el inicio de la instancia ocurre cuando el proceso recibe una señal de una instancia determinada) o evento de Inicio múltiple (la instancia puede iniciarse por confluencia de varias de las condiciones anteriores).

A continuación, observaremos algunos ejemplos de diagramas de proceso de negocio que cuentan con dichos elementos.

Tradicional: indica por default el comienzo de la instancia

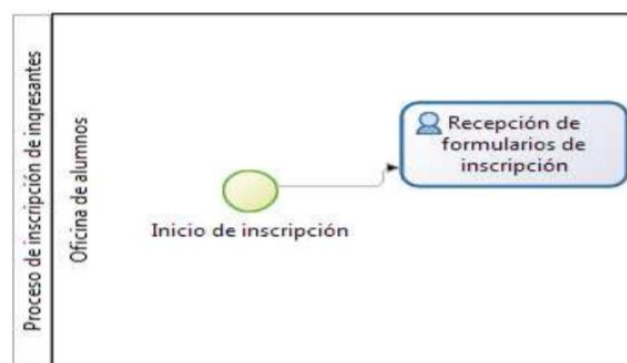


Fig. 6 – Evento de inicio de una instancia de proceso de negocio

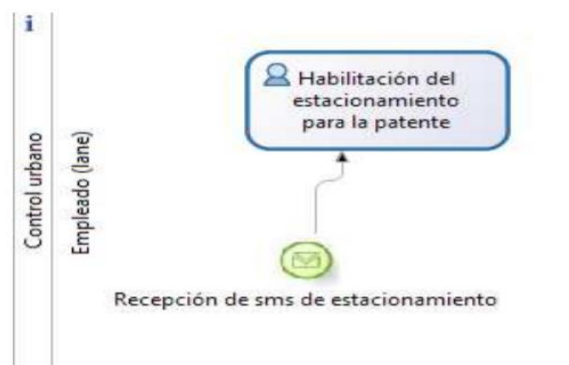


Fig. 7 - Evento de inicio de una instancia ante la recepción de un mensaje

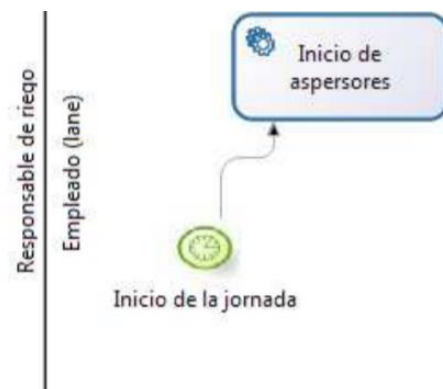


Fig. 8 - Evento de inicio de una instancia ante la ocurrencia de un determinado plazo temporal

Los **eventos intermedios** se producen dentro de la ejecución del proceso. Sus tipos son similares a los eventos de inicio. Por ejemplo, un evento intermedio por mensaje ocurre cuando se envía el mail a un destinatario, mientras que un evento por señal, espera el arribo de la misma o la emite hacia la próxima tarea.

Los eventos de fin permiten indicar la finalización del flujo del proceso. Algunos de los eventos de fin pueden ser: evento de fin básico, finaliza la ejecución de la instancia, evento de terminación que finaliza todas las instancias activas o evento por de fin por mensaje o señal, que los emite cuando finaliza la instancia.

Las **tareas** son las unidades de ejecución más pequeñas de un proceso, no es posible dividirlos en tareas más pequeñas. En caso que sean tareas de usuario deben tener un participante o actor asociado, quien realizará la tarea de forma manual, en caso de tratarse de una tarea automática se puede dividir en los siguientes tipos.

- Tarea de usuario: los responsables son actores del sistema. Se genera una interacción con dicho sistema.
- Tarea manual: es ejecutada completamente por mano de obra humana
- Tarea automática: se ejecuta por medio de un script o una unidad de software independiente del proceso

Los **subprocesos** son unidades de ejecución divisibles dentro del proceso. Se implementan mediante la sucesión de actividades y constituyen un detalle de actividades agrupadas juntas dentro del proceso principal. La Fig. 8, muestra cómo se modela el subproceso de *Fabricación de autopartes* dentro del proceso global de construcción de un automóvil.

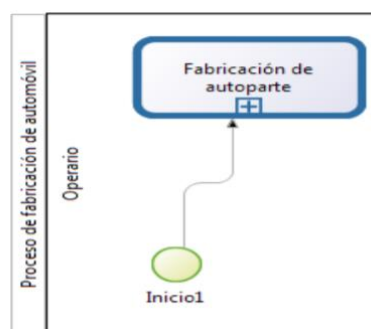


Fig. 9 - Subproceso de "Fabricación de autopartes" dentro del proceso de "Fabricación de un automóvil"

Las **compuertas** permiten evaluar condiciones que indican variaciones en el flujo dentro de la ejecución del proceso. Existen de distintos tipos:

- Compuerta exclusiva: permite seguir solo una de las ramificaciones indicadas. No puede haber más de un camino verdadero.

- Compuerta inclusiva: permite seguir más de una ramificación, en caso de que más de una sea verdadera
- Compuerta paralela: permite desencadenar la ejecución de varios flujos del proceso en forma simultánea.

2.5 BPMS (Business Process Management System)

La adopción de BPM y técnicas de administración de procesos por las organizaciones ha originado un fuerte interés en el mundo tecnológico para desarrollar productos y soluciones para soportar BPM. La tecnología BPM ha madurado al punto donde la administración de procesos en tiempo real es posible. Los BPMS representan un quiebre en el uso e implementación de sistemas de información. La metodología de implementación de los sistemas tradicionales se focaliza en funciones y objetos. Los procesos se relegan al flujo de trabajo, que por lo general no recibe mayor atención durante la implementación. Los BPMS rompen con la mentalidad de diseño de objetos. Pone al proceso en un foco central del diseño de soluciones. Esto alinea a las soluciones de IT a estar más en línea con la realidad de los procesos de las organizaciones.

Además de las funciones de diseño, un BPMS, permite que una vez implementado se puedan medir, monitorear, controlar y analizar el tiempo real la ejecución de los procesos. En pocas palabras, los BPMS sirven como el centro de control sobre las personas, aplicaciones de la empresa, y sus datos. Como centro de control los BPMS reciben datos en tiempo real de todas las tareas que se realizan en los procesos que controla, realizando un monitoreo en tiempo real.

Un BPMS puede detectar variaciones en el proceso y resolverlo. Las mediciones en el proceso de negocio proveen datos de proceso que se pueden analizar. El análisis de los datos de proceso es un paso esencial para identificar qué proceso debe mejorar y las mejoras a qué área pueden traer mayor valor al proceso.

Los sistemas de gestión de procesos de negocios (BPMS) son softwares utilizados como herramientas para representar y coordinar las actividades involucradas en un proceso de negocio. Convierten el modelo del proceso en un proceso ejecutable.

Las organizaciones que utilicen un BPMS para llevar a cabo el cambio de procesos de negocio habilitado por IT ganaran las siguientes capacidades:

1. Involucrarse mejor en el negocio en el diseño de soluciones de procesos de negocio habilitado por IT.
2. Habilidad de integrar personas y sistemas que participan en los procesos de negocio.
3. Habilidad de simular procesos de negocio para diseñar el proceso más óptimo para la implementación.
4. Habilidad de monitorear, controlar y mejorar procesos de negocio en tiempo real.

5. Habilidad de efectuar cambios en procesos de negocio existentes, en tiempo real sin mucho esfuerzo.

BPMS permite que los propietarios de los procesos de negocio se involucren directamente con la solución IT. El diseñador de procesos que provee un BPMS, es una herramienta que permite a los propietarios de los procesos de negocio o a los analistas de negocio diseñar los procesos de negocio en detalle.

Para ayudar a los propietarios y analistas de los procesos de negocio en el diseño de los procesos, Los BPMS provee la simulación de procesos y la posibilidad de modelado. El modo de simulación incluye distribuciones de probabilidad de tiempo para cada actividad. Se debe hacer un análisis previo para determinar qué algoritmo de simulación utilizar y que modelo de distribución probabilístico se adecua mejor para cada actividad en el proceso de negocio a probar.

Una vez que se colectan los datos, el simulador de procesos identifica qué pasos son los cuellos de botella y cualquier otra debilidad en el diseño de proceso. Basado en los resultados de simulación el diseño del proceso inicial se puede mejorar iterativamente.

Los BPMS dan a las organizaciones la capacidad de implementar mejoras en sus procesos casi en tiempo real sin demasiado esfuerzo o grandes modificaciones, ya que en su herramienta de diseño integra toda la información obtenida. Esto elimina la necesidad de reunir la información del proceso actual, luego de realizar la modificación el BPMS permite que cualquier instancia comenzada con el proceso original finalice utilizando la definición del proceso original, y que las nuevas instancias de proceso se realicen utilizando la definición del proceso mejorado [7].

2.6 Software Bonita Open Solution

Es una suite ofimática para la Gestión de procesos de negocio y realización de *Workflows*, creada en 2001. Es de código abierto y puede ser descargado bajo la licencia GPL v2. Características generales [8]:

- Está compuesto por dos ambientes, producción y desarrollo, este último por 3 elementos: *Bonita Studio*, *BonitaBPM engine*, *BonitaUserXP*, cuenta con más de 100 conectores existentes y también pueden ser creados utilizando *Bonita Users XP*, es muy flexible y potente.
- El despliegue de los procesos simplemente consta de la importación de los mismos en el ambiente de producción, el cual es igual al *Bonita User XP (MySQL, Postgre SQL, Oracle DB)* y la posibilidad de agregar otros conectores para ejecutar consultas H2 y otros gestores de BD.

La arquitectura actual de bonita se encuentra formada de los siguientes componentes: una API para administración de usuarios, tareas y procesos, la

interacción con el usuario mediante Bonita UserXP y el motor de ejecución – *Execution Engine* – al que se accede a través de la API.

Por medio de los conectores Bonita se comunica con otros sistemas externos, estos son la puerta de Bonita hacia el exterior, en la solución planteada en este caso de estudio utilizaremos la funcionalidad provista por los conectores para poder realizar las comunicaciones.

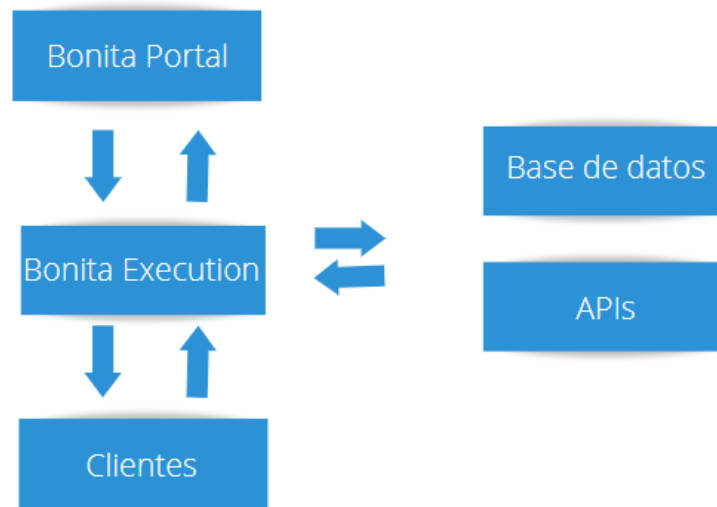


Fig. 10 – Arquitectura de las soluciones Bonita Soft

Capítulo 3. Internet de las Cosas

En este capítulo se hará un repaso sobre los principales conceptos de la internet de las cosas y cómo este nuevo concepto está cambiando al mundo y nuestra forma de comunicarnos con el entorno, detallando los elementos físicos que la componen y sus principales protocolos.

3.1 Definiciones y generalidades

Internet de las Cosas (IoT) es una tecnología emergente, que se estima va a traducirse en mayores ingresos económicos, mayor eficiencia y menores costos para las organizaciones, pero su despliegue puede ocasionar otros inconvenientes a la hora de su integración con tecnologías preexistentes. IoT es un fenómeno tecnológico bastante reciente.

El término fue introducido por Kevin Ashton [2] investigador en el campo de la identificación por radiofrecuencia (RFID). Este autor plantea el debate sobre qué ocurriría si absolutamente todos los objetos que nos rodean nos brindarán información útil al instante. Esto cambiaría por completo nuestra forma de desenvolvernos con el mundo que nos rodea. Las computadoras podrían recopilar información sobre los objetos físicos sin intervención humana mediante el uso de tecnología de sensores y actuadores.

IoT (“Internet of things”, Internet de las cosas en español) se define como una evolución de Internet en la cual los objetos que son utilizados en la vida cotidiana se encuentran conectados en red, enviando y recibiendo datos. También se puede definir como la interfaz entre el mundo físico y el digital, que permite obtener información de, y controlar, a los objetos mencionados anteriormente.

El concepto abarca desde sencillos sensores o actuadores para manejar electrodomésticos, hasta complejas infraestructuras públicas de alumbrado o sistemas industriales, pasando por coches y casas inteligentes [9].

A pesar de que la definición de “cosas” ha cambiado a medida que la tecnología fue evolucionando, el objetivo principal de realizar una lectura de la información sin la ayuda de la intervención humana, se mantiene. Esto llevó a una evolución radical de la Internet actual, transformándose en una red de objetos interconectados que no solo recolecta información del ambiente e interactúa con el mundo físico, sino que también utiliza los estándares existentes de internet para proveer servicios para la transferencia de información.

IoT conectará 50 billones de objetos inteligentes luego del 2021, cuando la población mundial alcance los 7.600 millones de habitantes [2].

Dentro del amplio rango de aplicaciones de IoT, podemos citar algunos ejemplos concretos:

- Una heladera que realice las compras al supermercado de forma automática y pueda avisar que productos se están por vencer.
- Que el jardín de una casa se riegue de forma automática, si se cumplen ciertos parámetros que controla en la tierra de las plantas como por ejemplo la humedad.
- Pulseras que cuando salgamos a correr informan nuestros signos vitales y nuestro rendimiento.

El ámbito de aplicación de IoT no está limitado solo a la vida personal o hogareña. Construir una red de sensores dentro de un ambiente en el cual se pretende sensar o medir un atributo, permite implementar IoT en diversos escenarios comerciales [10].

Ejemplo 1. Transporte/Logística

La sincronización continua de información sobre el flujo de materiales, seguimiento y rastreo en tiempo real de objetos permite mejorar no sólo la cadena de suministro, sino también el posicionamiento global y la identificación automática de la mercadería de la empresa.

Ejemplo 2. Casas Inteligentes

Cubre principalmente a tres aspectos: el uso de recursos (ahorro de agua y consumo de energía), seguridad (la detección de robo, incendio o la entrada no autorizada) y confort. El objetivo es lograr mejores niveles de confort mientras se disminuye el gasto total.

Ejemplo 3. Ciudades Inteligentes

Las aplicaciones IoT que podemos encontrar actualmente en las Ciudades Inteligentes o *Smart Cities* son variadas como control de tráfico, alumbrado inteligente, etc.

Ejemplo 4. Fábrica Inteligente

En una cadena de suministro global, las empresas pueden realizar un seguimiento de todos sus productos por medio de las etiquetas de identificación por radiofrecuencia (RFID). Como consecuencia, reducen sus gastos operativos y mejoran su productividad debido a una mayor integración con la planificación de recursos empresariales y de otros sistemas. Además, el mantenimiento de la maquinaria se ve facilitado por sensores conectados, lo que permite la monitorización en tiempo real del estado y el rendimiento de los equipos de la fábrica.

Entre los actores de IoT se pueden encontrar sensores, actuadores, dispositivos RFID y *gateways* usados para recolectar las mediciones de los sensores para un posterior procesamiento, mientras que la capa de red provee las capacidades de transporte y red necesarias para encaminar la información de IoT hacia diferentes lugares. La capa de soporte es una capa *middleware* que sirve para ocultar la complejidad de las capas más bajas a la capa de aplicación, y provee servicios genéricos y específicos, como almacenamiento en diferentes formas (sistemas de administración de bases de datos y/o sistemas *cloud computing*) y otros servicios.

Como muestra la Figura 11, existe una clara analogía entre la arquitectura de referencia para IoT y el modelo TCP/IP.

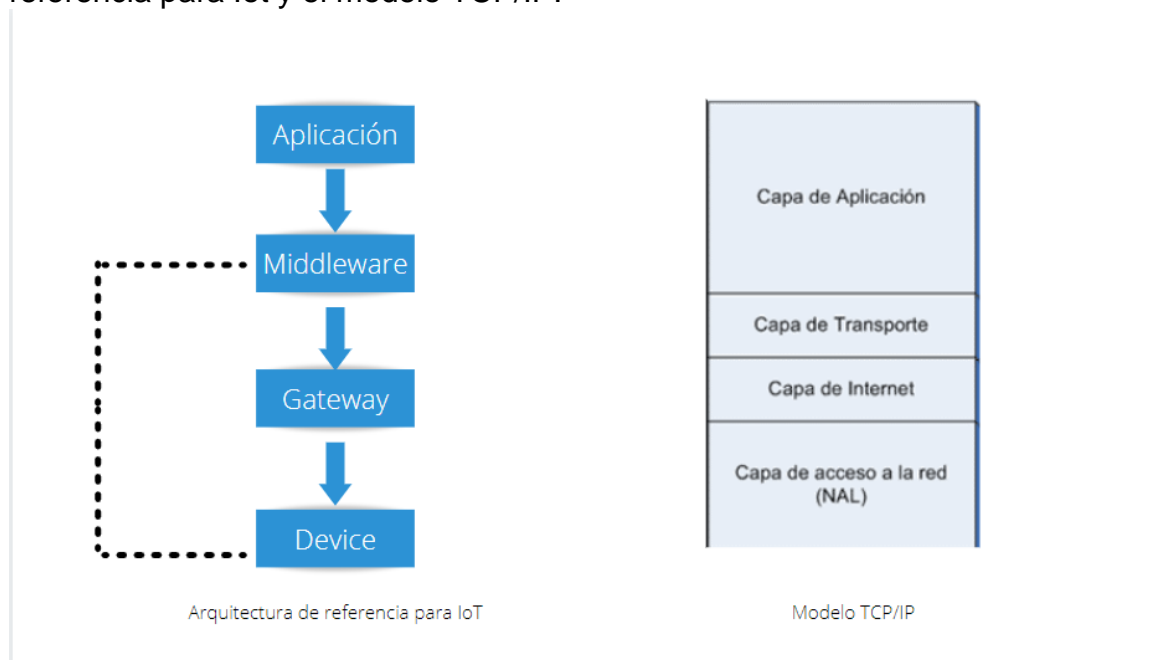


Fig. 11 – Comparación de arquitectura IoT con TCP/IP

3.2 Elementos de IoT

Usualmente los dispositivos de IoT se conectan en una red de sensores inalámbricos (WSN, Wireless Sensor Network) también denominadas redes personales inalámbricas o redes de área personal (PAN), es una red que en muchos casos es autoconfigurable de pequeños nodos sensores (llamados motas) que se comunican entre sí mediante señales de radio (habitualmente), y se despliegan una gran cantidad de ellos para medir datos del mundo real.

Las motas son pequeñas computadoras con una funcionalidad extremadamente básica, consisten de una unidad de procesamiento con poder computacional limitado y una memoria limitada, un dispositivo de comunicación por radio, una fuente de energía y uno o más sensores. Vienen en diferentes formas y tamaños, dependiendo del uso que se les vaya a dar. Pueden ser muy pequeñas, si por ejemplo se desplegarán varias y se necesita tener un pequeño impacto visual. Además, pueden tener una batería recargable como fuente de energía. La integración de estos dispositivos minúsculos y ubicuos en los escenarios más diversos asegura un amplio rango de aplicaciones. En una aplicación típica IoT, una WSN es diseminada en una región donde esté destinada a recolectar datos a través de sus nodos sensores. Estas redes proveen un puente entre el mundo físico y el virtual, y tienen habilidades sin precedentes para observar y entender fenómenos del mundo real de gran escala con una fina resolución espacio-temporal.

Todas las motas están compuestas de cinco elementos principales:

- **Procesador:** la tarea de esta unidad es procesar localmente información sensada por la mota a la que pertenece e información sensada por otras motas. En el presente, los procesadores son limitados en cuanto a poder computacional, pero dada la ley de Moore, estos serán más pequeños, más poderosos y consumirán menos energía. El procesador puede correr en diferentes modos: el modo sueño, utilizado la mayoría del tiempo para ahorrar energía, el modo ocioso cuando pueden llegar datos de otras motas, y el modo activo utilizado cuando se censan datos, o cuando se envían a, o reciben de otras motas.

- **Fuente de energía:** las motas pueden instalarse en varios ambientes, incluyendo regiones remotas y hostiles, por lo que deben utilizar poca energía.

Típicamente tienen poco almacenamiento energético, por lo que los protocolos de red deben enfatizar el ahorro de energía. También deben tener incorporados mecanismos para permitir al usuario final la opción de prolongar el tiempo de vida de la red. Además, las motas pueden estar equipadas de mecanismos de generación de energía, como celdas solares, por lo que pueden dejarse desatendidas por meses o años. Las fuentes de poder más usuales son las baterías recargables, los paneles solares y capacitores.

- **Memoria:** utilizada para almacenar tanto programas como datos.

- **Radio:** los dispositivos presentes en las WSN incluyen un radio inalámbrico de baja tasa y de corto alcance. Las tasas típicas están entre 10-100 kbps, y el rango es menor a 100 metros. La comunicación por radio usualmente es la tarea que más energía consume, por lo que es imperativo incorporar técnicas de ahorro de energía.

- **Sensores:** las redes de sensores consisten de varios tipos de sensores diferentes capaces de monitorear una amplia variedad de condiciones ambientales, como por ejemplo el medidor de temperatura, humedad, luz, presión, niveles de ruido, aceleración, humedad del suelo, etc. Algunas aplicaciones requieren un modo de multi sensado, por lo que cada dispositivo puede tener varios sensores incorporados [11].

Dentro de una red de WSN se pueden identificar los siguientes roles:

- **Nodos sensores:** son utilizados para sensar sus alrededores y transmitir los valores leídos a un nodo "sink", también llamado "estación base". Típicamente están equipados con diferentes tipos de sensores.
- **Nodos "sink" o "estaciones base":** Recolectan los valores leídos de los sensores de otros nodos, y los envían hacia un *gateway* con el que se encuentran directamente conectados, para un futuro procesamiento. Están equipados con capacidades mínimas de procesamiento y comunicación, pero no tienen capacidades para medir.
- **Nodos actuadores:** son dispositivos utilizados para controlar el ambiente, basándose en valores leídos por un sensor o por otras entradas. Pueden tener la misma configuración que una mota, pero equipados además con cualidades de control, por ejemplo, para apagar o prender una luz.

- **Nodos gateways:** conectados a los nodos sink están conectados a una fuente de alimentación estable, dado que consumen mucha energía. Estas entidades son dispositivos de cómputo normales como laptops, notebooks, desktops o teléfonos celulares. Sin embargo, no están equipados con capacidades para medir.

3.3 Protocolos para soluciones IoT

Analizados los conceptos básicos, cabe describir cómo comunicar el universo de dispositivos diferentes que pueden existir en una solución de IoT, es decir los protocolos.

Un gran desafío de IoT es lograr la interoperabilidad entre la amplia gama de dispositivos que existen. La conectividad de los dispositivos industriales con tecnologías de información y plataformas IoT es un tema amplio, dado que existen varios protocolos para cumplimentar esto, algunos que son privados y cerrados y otros que son estándares abiertos y que, además, coexisten.

Entre las redes podemos encontrar varias diferencias como por ejemplo su alcance, potencia y uso de la memoria.

Las conexiones a través de redes IP son complejas y consumen mucha memoria y potencia de los dispositivos IoT, mientras que el alcance no es un problema. Las redes que no son IP, como Bluetooth, requieren menos memoria y energía, pero tienen una limitación de alcance.

El tipo de conectividad que se utiliza como parte del protocolo de IoT depende del dispositivo, de su función y de los usuarios. Normalmente, la distancia a la que los datos deben viajar (corta o larga) determina el tipo de conectividad de IoT necesaria.

Una posible clasificación de las redes de IoT según alcance y consumo es [12]:

Redes de corto alcance y bajo consumo

Las redes de baja potencia y corto alcance están indicadas para hogares, oficinas y otros entornos de un tamaño reducido. Normalmente, necesitan baterías pequeñas y su uso suele resultar económico.

Ejemplos de este tipo de redes son Bluetooth, NFC, Wi-Fi/802.11, Z-Wave y Zigbee.

Redes de área extensa de bajo consumo (LPWAN)

Las redes LPWAN permiten la comunicación en un radio mínimo de 500 metros, tienen un consumo de energía mínimo y se usan para la mayoría de los dispositivos IoT. Los siguientes son algunos ejemplos comunes de redes LPWAN:

4G LTE y 5G para IoT, Cat-0, Cat-1, LoRaWAN, LTE Cat-M1, NB-IoT/CAT-M2 y Sigfox.

Protocolos utilizados en IoT según el modelo OSI

El modelo OSI constituye el modelo de referencia para protocolos de red que, sin dudas, continúa siéndolo en el escenario de IoT.

En la Tabla 2, se resumen los protocolos por capa del modelo OSI y se indican los que son específicos para IoT.

Tabla 2. Protocolos IoT según el modelo OSI

Capa	Protocolo	Descripción	IoT
Capa de aplicación	Advanced Message Queuing Protocol (AMQP)	Crea interoperabilidad entre el middleware de mensajería, permitiendo crear una mensajería normalizada a escala industrial.	Si
	Protocolo de aplicación restringida (CoAP)	Protocolo de transferencia de documentos que se ejecuta a través UDP, diseñado para ancho de banda limitado.	Si
	Servicio de distribución de datos (DDS)	Protocolo de comunicación punto a punto versátil que permite desde ejecutar pequeños dispositivos hasta conectar redes de alto rendimiento	Si
	Message Queue Telemetry Transport (MQTT)	Protocolo de mensajería diseñado para la comunicación ligera entre equipos utilizado en conexiones de poco ancho de banda con ubicaciones remotas.	Si
	OPC UA	Es un protocolo cliente/servidor, utilizado en la comunicación de máquinas industriales basadas en placas PLC.	Si
Capa de transporte	Protocolo de control de transmisión (TCP)	Protocolo orientado a la conexión que permite que dos nodos se conecten e intercambien información.	No
	Protocolo de datagramas de usuario (UDP)	Protocolo no orientado a la conexión que permite que dos nodos se conecten e intercambien información.	No
Capa de red	IP	Permite el intercambio de paquetes de información entre nodos de una red, sin garantía de recepción e identifica cada nodo.	No
	6LoWPAN	Protocolo de red apto para dispositivos de baja potencia que tienen una capacidad de	Si

		procesamiento limitada.	
Capa de enlace y Física	IEEE 802.15.4	Estándar de radio para una conexión inalámbrica de bajo consumo. Se usa con Zigbee, 6LoWPAN y otros estándares para crear redes inalámbricas insertadas.	No
	LPWAN (Low Power Wide Area Networks)	Permiten la comunicación a distancias desde 500 metros hasta más de 10 km en algunos lugares.	Si
	Bluetooth Low Energy (BLE)	BLE reduce el consumo de energía y el costo, y mantiene una distancia de conectividad similar a la del Bluetooth clásico.	No
	Ethernet	Esta conexión por cable es una opción menos costosa que proporciona conectividad rápida para datos con una latencia baja.	No
	Transmisión de datos en proximidad (NFC)	Conjunto de protocolos de comunicación que utilizan campos electromagnéticos y permiten que dos dispositivos se comuniquen si están a una distancia no superior a cuatro centímetros. Los dispositivos habilitados para NFC funcionan como tarjetas de identidad y suelen utilizarse para pagos móviles, vales y tarjetas inteligentes sin contacto.	Si
	Power Line Communication (PLC)	Tecnología de comunicación que permite enviar y recibir datos a través de los cables eléctricos. Permite alimentar y controlar un dispositivo IoT a través del mismo cable.	No
	Identificación por radiofrecuencia (RFID)	El protocolo RFID utiliza campos electromagnéticos para hacer un seguimiento de las etiquetas electrónicas no alimentadas de otro modo. El hardware compatible proporciona energía y se comunica con estas etiquetas para leer su información con fines de identificación y autenticación	Si
	Wi-Fi/802.11	Wi-Fi/802.11 es la conexión habitual en hogares y oficinas. Aunque es una opción económica, puede que no se	No

ajuste a todos los escenarios por su alcance limitado y el consumo energético ininterrumpido.

Z-Wave	Red en malla que usa ondas de radio de baja potencia para la comunicación de dispositivo a dispositivo.	Si
Zigbee	Especificación basada en IEEE 802.15.4 para un conjunto de protocolos de comunicación de alto nivel que se usan para crear redes de área personal con un radio digital pequeño de baja potencia.	Si

Protocolos IoT utilizados en la propuesta

Los protocolos utilizados de manera específica en la construcción de la solución propuesta son los siguientes:

OPC UA

Durante los últimos años la automatización de las máquinas industriales fue realizada utilizando placas PLC (*Programmable Logic Controller*), las cuales, para poder comunicarse entre sí, utilizan el protocolo OPC UA (*Open Productivity Collaboration Unified Architecture*), más conocido por “arquitectura unificada”.

Los clientes se conectan, navegan, leen y escriben al equipamiento industrial. UA define la comunicación desde la aplicación hacia la capa de transporte, lo que lo hace muy compatible entre los distintos fabricantes de PLC. También es muy seguro, y usa mensajes bidireccionales firmados y encriptación de transporte.

Es una buena solución para conectar información de sensores y PLC en aplicaciones industriales ya existentes, debido a que la mayoría de los equipos industriales de la actualidad cuentan con controladores de este tipo.

MQTT

MQTT (*Message Queuing Telemetry Transport*), es un protocolo que se maneja con suscripciones, diseñado para SCADA (*Supervisory Control And Data Acquisition*), permite controlar y supervisar procesos industriales a distancia) y redes remotas.

Se centra principalmente en el concepto de contar con un encabezado muy pequeño de ser muy estable y seguro, se utiliza principalmente para enviar mensajes en redes con mucho tráfico de paquetes.

Tal como HTTP, MQTT es muy simple de utilizar y la mayoría de los lenguajes de programación lo implementan mediante librerías y sus respuestas son en formato JSON.

MQTT es un protocolo que podemos utilizar principalmente cuando no conocemos la estructura de la red, ya que simplemente los nodos publican sus

mensajes en un canal, sin conocer quiénes son sus destinatarios finales y se registran para ser informados de ciertas acciones.

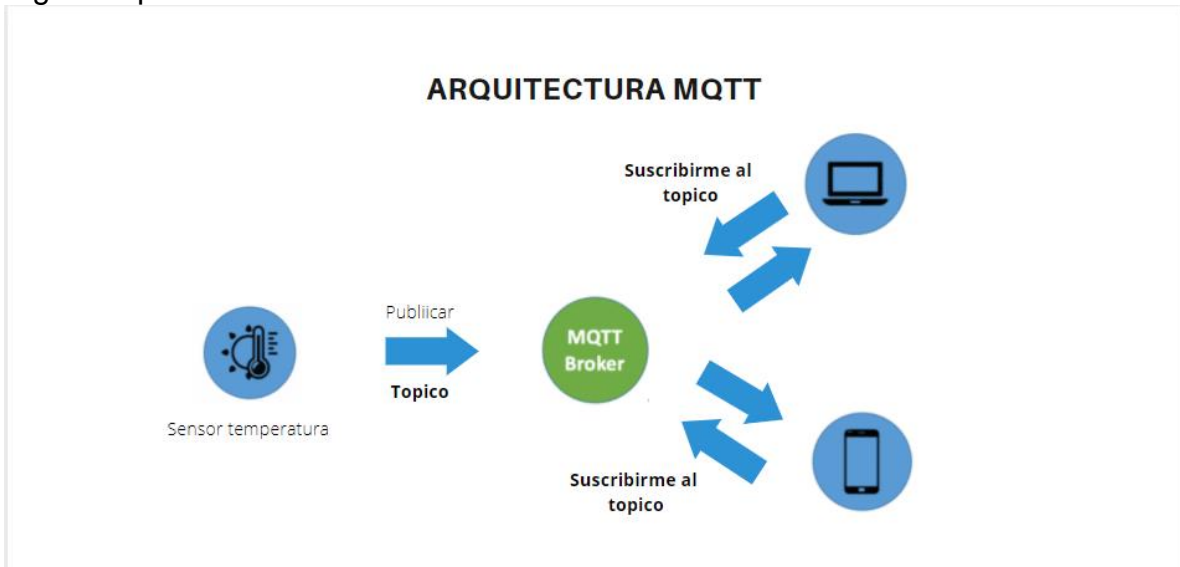


Fig 12 – Arquitectura OPC

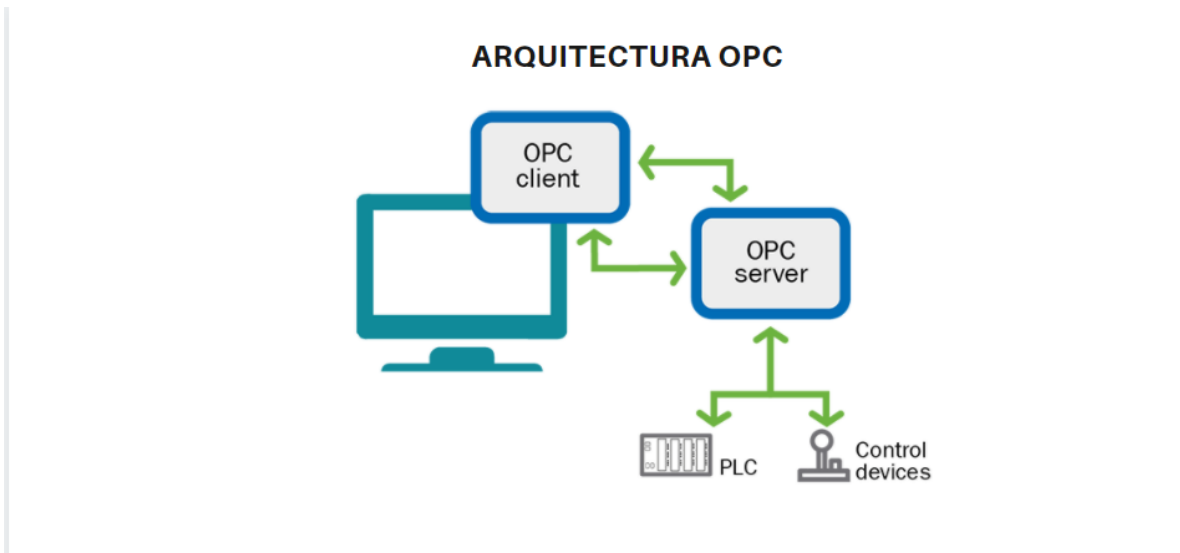


Fig 13 – Arquitectura MQTT

CoAP

CoAP fue creado como una versión liviana de HTTP cuya arquitectura ya vimos anteriormente, con una carga mínima para no saturar a la red, es similar a HTTP, pero usa UDP en lugar de TCP para realizar la comunicación a nivel de capa de transporte.

DDS

DDS (*Data Distribution Service*) es un protocolo de suscripción que a diferencia de MQTT no requiere de un agente centralizado, DDS está descentralizado. Los nodos de DDS se comunican directamente punto a punto a través de UDP. Esto hace que no sea necesaria una gestión centralizada de la red y que DDS sea un protocolo más rápido, con una resolución por debajo del milisegundo. DDS es una buena alternativa para las soluciones que requieren una respuesta en tiempo real.

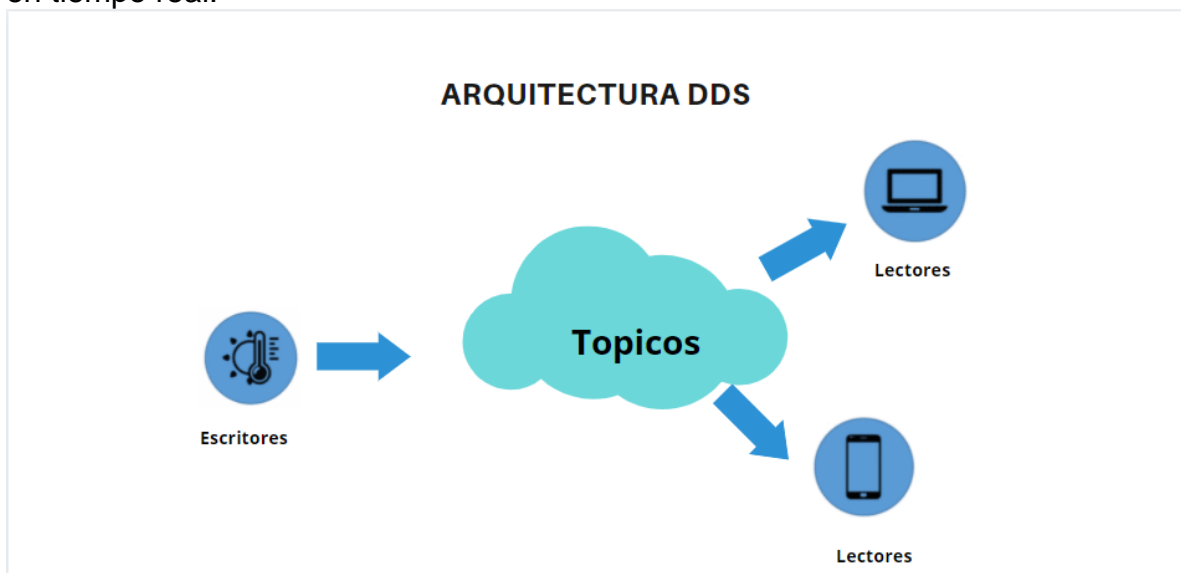


Fig 14 – Arquitectura DDS

En la siguiente tabla 3 se resume ventajas y desventajas de cada uno de los protocolos utilizados en la solución planteada.

Servicios web

A continuación, hablaremos sobre los servicios web como un paradigma de desarrollo de aplicaciones, introducir este tema nos permitirá comprender la arquitectura seleccionada para el caso de estudio. Permiten que las compañías vinculen su software propio al de terceros mediante un middleware que provee una interfaz entre clientes y servidores. Los clientes acceden a las operaciones publicadas en la interfaz del servicio web.

Son el bloque fundamental de una Arquitectura orientada a servicios, poseen interfaces bien definidas, independientes de su implementación.

Los clientes consumen los servicios sin la necesidad de entender cómo estos ejecutan sus pedidos, los servicios puede ser descubiertos dinámicamente, pueden componerse y proveen funcionalidad de grano grueso.

Los servicios cumplen con un objetivo de negocio predeterminado, no dependen del contexto o estado de otros servicios y tienen transparencia de localización.

Los servicios web pueden implementarse bajo los protocolos REST o SOAP.

Tabla 3- Ventajas y desventajas de los protocolos utilizados

Protocolo	Ventaja	Desventaja
OPC UA	Es un protocolo que actualmente se encuentra implementado en un alto porcentaje de la máquina industrial para comunicar los PLC entre sí.	La escritura de los datos en el PLC varía dependiendo del fabricante y en muchos casos son productos cerrados que no permiten su modificación.
MQTT	Es un protocolo muy liviano y muy útil cuando no conocemos la estructura de la red.	Cuenta con muy pocas desventajas, excepto que aún no es muy conocido entre los desarrolladores.
CoAP	Consumen menor energía que MQTT.	Actualmente todavía se encuentra en desarrollo por lo cual no está completamente estandarizado.
DDS	Es más rápido que MQTT al no necesitar un nodo central.	Puede tener una gran cantidad de pérdida de paquetes.

Las arquitecturas REST se construyen sobre recursos identificados por URIs. Cuentan con una interfaz común, por lo que las únicas operaciones permitidas son GET, POST, PUT y DELETE.

No tienen estado ni acceden a información de contexto. Toda la información para entender la petición debe encontrarse en la misma y utiliza códigos de respuesta nativos (404), permite transmitir cualquier tipo de datos, XML, JSON, Text.

Por su parte, los Servicios Web SOAP presentan una interface para invocar un método o función distribuido, el concepto es muy similar a RPC, cuenta con una especificación XML que indica como debe ser invocado, describiendo el formato de los mensajes que recibe y su tipo de respuesta

ARQUITECTURA REST

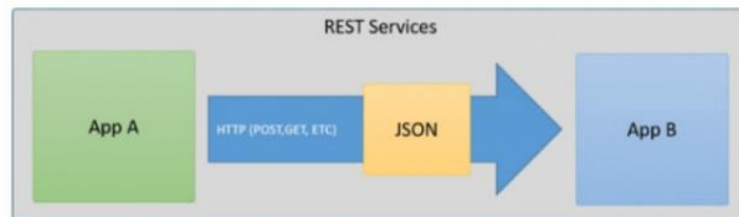


Fig 15 – Arquitectura REST

ARQUITECTURA SOAP



Fig 16 – Arquitectura SOAP

HTTP (REST/JSON)

HTTP (*Hypertext Transfer Protocol*, 'protocolo de transferencia de hipertexto') es un protocolo cliente/servidor sin estado.

Es uno de los protocolos más utilizados en el mundo para conexiones con páginas de internet, en IoT gira en torno a REST (*Representational State Transfer*, 'transferencia de estado representacional'), que es un modelo sin estados previos donde los clientes pueden acceder a recursos en el servidor a través de peticiones, solicitando recursos.

HTTP provee transporte, pero no define la presentación de la información. Así, un requerimiento HTTP puede contener diferentes formatos de presentación

(HTML, JavaScript, JSON (*JavaScript Object Notation*, 'notación de objeto JavaScript'), XML). En la gran mayoría de soluciones IoT están estandarizando para utilizar JSON para representar la información y HTTP para poder transportarla.

JSON es muy liviano y también es soportado por la mayoría de las herramientas y lenguajes de programación.

ARQUITECTURA REST / JSON

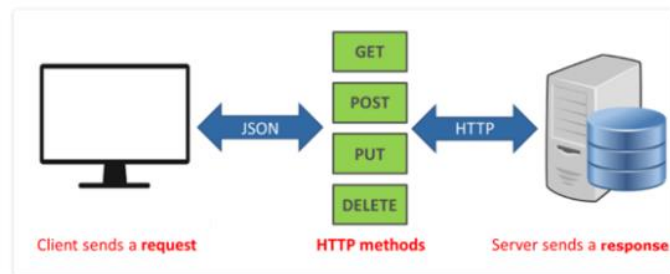


Fig 17 – Arquitectura REST/JSON

Capítulo 4. Integración BPM e IoT

Internet de las cosas (IoT) representa un entorno integral que consta de una gran cantidad de dispositivos inteligentes que interconectan objetos físicos que son muy diversos al mundo de Internet. Muchos dominios como la logística, la fabricación, la agricultura y la automatización del hogar han utilizado tecnologías de IoT.

Los sistemas de gestión de procesos de negocio (BPMS), por su parte, se han convertido en una solución eficaz y exitosa para la gestión coordinada y la utilización optimizada de los recursos. Sin embargo, para poder integrar ambas tecnologías debemos considerar muchos factores, ya que las entidades de IoT son muy heterogéneas y se encuentran conectadas a gran escala [7].

Existen muchas consideraciones que se deben tener en cuenta durante todo el ciclo de desarrollo, en este capítulo hablaremos de propuestas planteadas por diferentes autores y problemas que se plantean para realizar una integración entre IoT y BPM.

Como mencionamos anteriormente la tecnología actual permite conectar a elementos físicos de nuestra vida (alimentos, paquetes, electrodomésticos, vehículos, edificios, etc.) y lograr que se comunican entre ellos o con otros sistemas, con cierto grado de inteligencia sin intervención de las personas, logrando en muchos casos una comunicación que se conoce como máquina a máquina [13].

El objetivo de IoT es mejorar la vida de las personas y optimizar tareas tales como la agricultura, transporte, logística, educación, atención médica y más [14]. La industria predice que próximamente alrededor de 55 mil millones de dispositivos físicos se conectarán a Internet [15], y el valor de los ingresos de la economía aumentará hasta \$ 1.9- \$ 7.1 billones [16,17].

Hoy en día, IoT se ha convertido en uno de los temas más populares, tanto para la industria como para el mundo académico. Antes de que surgiera la visión de IoT, los Sistemas Ciberfísicos de ahora en adelante CPS, que interconectaban las entidades físicas con los sistemas de software, solían estar aislados. Cada subsistema tenía su propia topología y protocolos de comunicación.

Para integrar IOT en el universo aislado de CPS, uno de los principales enfoques es aplicar una solución de middleware basada en arquitectura orientada a servicios (SOA) como la que aplicaremos en el caso de estudio del presente trabajo.

Fundamentalmente SOA introduce el concepto de la interoperabilidad de sistemas aislados heterogéneos. Al aplicar SOA, CPS puede administrar dispositivos individuales como servicios atómicos o puede configurar un grupo de dispositivos para proporcionar servicios compuestos.

Por ejemplo, un dispositivo conectado a Internet puede brindar la temperatura del lugar físico en donde se encuentra ubicado a clientes de servicios web remotos. Además, un servicio compuesto en la nube puede explotar los datos

derivados de múltiples dispositivos en una zona geográfica específica para calcular y generar la información significativa para usuarios específicos.

En la última década, los sistemas de gestión de flujo de trabajo (WfMS) se han convertido en uno de los componentes principales de la composición de servicios.

Entre los diferentes WfMS, los sistemas de gestión de procesos de negocio (BPMS) han sido ampliamente aceptados [18], principalmente debido a la disponibilidad de herramientas y estándares internacionales como *Business Process Execution Language* (BPEL) [19], BPM y notación (BPMN) y Lenguaje de definición de procesos XML (XPDL) [20]. Como indicamos en capítulos anteriores del presente trabajo, los BPMS son “sistemas de software genéricos que sirven para promulgar y gestionar procesos comerciales operativos” [21].

4.1 Soluciones basadas en procesos con IoT

Se abordan en esta sección, diferentes problemas de la vida real en los cuales algunos autores consideraron posibles soluciones de integración IoT y BPM.

En [22] se presenta una solución que tiene que administrar una playa de estacionamiento de camiones de carga, algunos de ellos contienen una carga peligrosa que no pueden estacionarse cerca de otros camiones que contienen un tipo de carga similar, debido a que no lo permite la reglamentación vigente por normas de seguridad. Se podría implementar una solución mediante etiquetas de identificación por radiofrecuencia (RFID), sensores y una red ad hoc móvil en los vehículos, para indicarles en qué lugares libres es seguro que se estacionen.

Otro ejemplo es el que se describe en [23] donde se administra una red de suministros y se monitorea en tiempo real. La fábrica A tiene una línea de producción que requiere el suministro de la fábrica B con urgencia. Dado que IoT se ha implementado en todas las empresas asociadas a la fábrica B, A es capaz de identificar si B puede producir y enviar el suministro a tiempo o no, basándose en el análisis de los recursos en stock y las condiciones de envío. En el caso de que A descubra que B no puede cumplir con la tarea a tiempo, puede intentar encontrar una sustitución ya sea distribuyendo el pedido a varios proveedores o encontrando un proveedor alternativo que pueda manejar todo el pedido. Si A no puede encontrar una solución alternativa y tiene que posponer su producción, los trabajadores asignados originalmente y los vehículos para el envío pueden reasignarse a las otras tareas para reducir el desperdicio de recursos humanos.

En base a estos ejemplos, se observa que BPMS con IoT puede mejorar la calidad de la vida cotidiana de las personas, permitiendo que se activen acciones autónomas en función de ciertos eventos que se miden en función de los dispositivos IoT implementados.

Por ejemplo, los sistemas de calefacción, ventilación y aire acondicionado, utilizados en los edificios modernos se pueden monitorear y controlar desde el sistema de información remoto basado en BPM, precisamente a través de

dispositivos de red de actuadores y sensores inalámbricos conectados a Internet (WSAN).

Otro ejemplo, en el caso de un hotel, mejora la eficiencia de la gestión en la que el sistema puede automáticamente medir el uso del sistema de calefacción y electricidad y facturar al cliente en función del uso en lugar de cobrar la tarifa fija por cada habitación [24].

El sistema de atención médica basado en BPMS-IoT integra servicios en la nube con *Wireless Body Area Red (WBAN)*, presentado en [25], está formada por numerosos dispositivos sensores portátiles que pueden medir el azúcar en sangre, la temperatura corporal, los latidos del corazón, etc.

La Figura. 18 ilustra una arquitectura común BPMS basada en SOA que involucra entidades de IoT y el sistema de información basado en middleware.

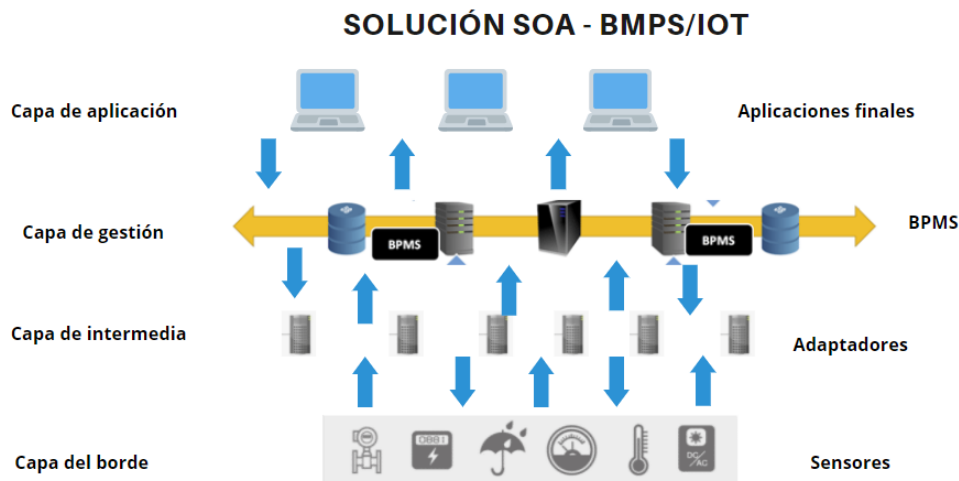


Fig. 18 – Solución SOA

Esta arquitectura, mostrada en la Fig. 18, se propone como una solución basada en SOA que involucra entidades de IoT y un sistema de información basado en middleware.

Sin embargo, esta arquitectura presenta las siguientes debilidades (que observaremos que no serán tales en el caso de estudio sobre el que se aplica).

Transparencia

Como muestra la Figura 18, las aplicaciones finales no tienen interacción directa con los nodos de IoT (es decir, nodos frontales inalámbricos conectados a Internet como teléfonos móviles, sensores inalámbricos, recolectores de datos, etc.).

El diseño oculta la topología detallada y la comunicación en la red externa, esto podría ocasionar que el diseñador de procesos no pueda definir adecuadamente los comportamientos de los mismos.

Agilidad

Los sistemas de TI basados en SOA a menudo utilizan tecnologías de interfaz y middleware para aprovechar diferentes entidades. Sin embargo, debido a la

falta de estandarización en el modelado de procesos y los niveles de ejecución, los sistemas generalmente resultan en un diseño complejo e inflexible.

Por ejemplo, según la arquitectura descrita en la Figura 18, si dos dispositivos, que están conectados a diferentes servidores *backend*, pero ubicados en una proximidad física, tienen la intención de interactuar entre sí, la comunicación debe pasar a través de múltiples capas. Como indica la visión [26], en los futuros entornos de IoT, los dispositivos cooperarán automáticamente para determinadas tareas con el fin de mejorar la eficiencia y la agilidad.

Ciclo de vida

Para poder integrar un BPMS con una solución de IoT, en general, podemos considerar tres fases en el ciclo de vida de BPMS: fase de diseño, fase de implementación/configuración y fase de ejecución/ajuste [27]. Dependiendo del enfoque de algunas disciplinas se puede clasificar cada fase en fases un poco más detalladas. Por ejemplo, los autores en sus trabajos anteriores [28, 29] han separado la fase de diseño a la fase de diagnóstico y la fase de diseño del proceso.

La fase de (re) diseño también puede implicar tres fases diferentes, incluido el descubrimiento de procesos, el análisis de procesos y el rediseño de procesos. La Figura 19, ilustra este ciclo de vida.

CICLO DE VIDA PROCESO - BPM/IOT

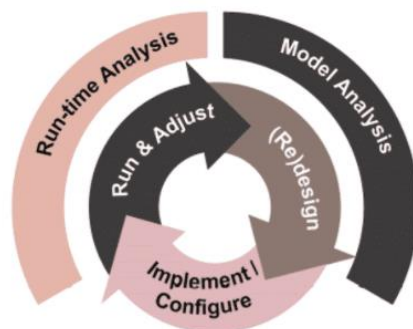


Fig 19 – Ciclo de vida de un proceso de IoT

La fase de diseño incluye cómo modelar los elementos de IoT y sus elementos relacionados dentro de un diagrama de un proceso de negocio.

La fase de implementación/configuración determina cómo implementar prácticamente el modelo de proceso como métodos ejecutables. Además, implica cómo implementar los métodos ejecutables en los motores de flujo de trabajo correspondientes para su ejecución.

La fase de ejecución y ajuste corresponde a cómo el BPMS supervisa y gestiona de forma autónoma el sistema en tiempo de ejecución, y cómo mejora y optimiza continuamente el proceso.

Si el diseñador de procesos reconoce la necesidad de mejorar los procesos, diseñará el sistema y continuará con el ciclo. Además de las tres fases principales del ciclo de vida, durante la fase de diseño, los desarrolladores del sistema aplicarán tecnologías para analizar el diseño del modelo (por ejemplo, mediante simuladores). El equipo de gestión también recopila los datos (por ejemplo, registros de eventos) en la fase de ejecución y ajuste para el diagnóstico del proceso. Las siguientes subsecciones proporcionan una revisión y un análisis basados en cada fase del ciclo de vida.

Construir una solución basada en procesos con IoT, implica revisar los siguientes aspectos, la arquitectura, el modelado, la ejecución y el monitoreo. La arquitectura plantea el desafío de decidir si se utilizará orquestación o coreografía de procesos. El modelado considera si es necesario diseñar nuevos elementos dentro de la notación para poder diagramar los elementos de IoT y sus actividades. La ejecución cómo modelado se instancia en una versión ejecutable y representa cada uno de los flujos implícitos en los modelos. El monitoreo, implica analizar el mejor método para registrar cada uno de los indicadores claves en una solución distribuida que va a contar con muchos actores variados.

4.2 Arquitectura de soluciones basadas en procesos con IoT

La decisión arquitectónica en torno a orquestación o coreografía cobra sentido en las soluciones con IoT, en tanto la orquestación requiere una coordinación centralizada mientras que la coreografía admite una coordinación distribuida.

La orquestación se basa principalmente en una arquitectura centralizada en la que un único sistema de gestión ejecuta todo el motor de procesos. Por otro lado, la coreografía representa un sistema que, en ciertas etapas, varios sistemas o dispositivos de información externos están manejando los procesos (o las partes del proceso).

Para comprender los conceptos de orquestación y coreografía nos referimos a los conceptos técnicos de una arquitectura SOA (*Service Oriented Architecture*). SOA que es una arquitectura basada en servicios la cual se puede implementar mediante orquestación o coreografía, la cual está formada por servicios de aplicación débilmente acoplados y altamente interoperables. Se basa en tres conceptos técnicos: servicios, interoperabilidad a través de un ESB y bajo acoplamiento [30].

La Figura 20 muestra la composición clásica de servicios SOA basada en los 3 vértices del triángulo formado por cliente, servidor y directorio o registro. Es una evolución del mecanismo de comunicación cliente/servidor hacia uno que incluye la publicación y búsqueda de servicios por parte de servidores y clientes, respectivamente.



Fig 20 – Composición de servicios

Los servicios Web, como una implementación particular de SOA, admiten orquestación o coreografía como se muestran en las Figuras 21 y 22.

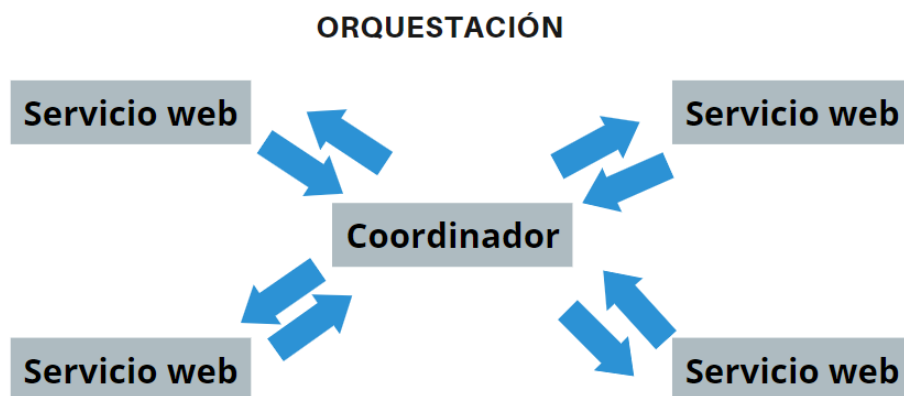


Fig 21 – Orquestación de procesos

En la coreografía no existe el rol del coordinador central, en su lugar cada Servicio Web conocerá exactamente cuándo ejecutar sus operaciones y con quien interactuar. Los participantes conocen la lógica del proceso, operaciones a ejecutar, mensajes a intercambiar y en qué momento ejecutarse.

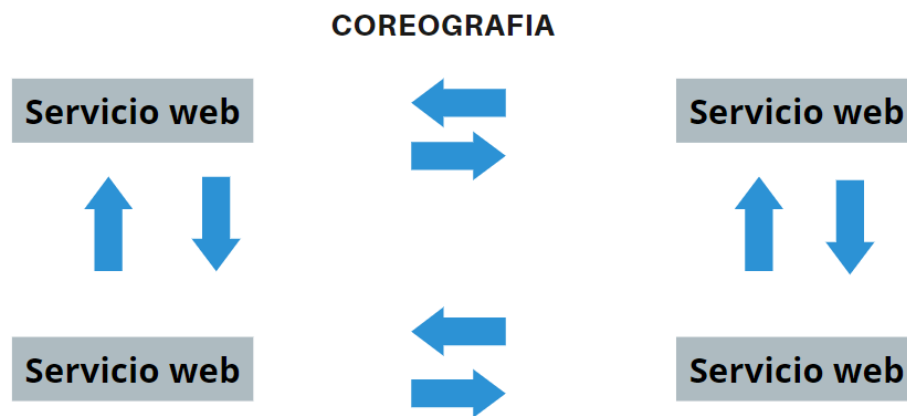


Fig 22 – Coreografía de procesos

En [31], [32] los autores consideran que la orquestación centralizada es insuficiente para reaccionar ágilmente a los eventos que ocurren en la red de IoT, cuando por ejemplo se trata de números dispositivos celulares conectados a una red de Internet móvil. En tales escenarios, existe la necesidad de aplicar una arquitectura basada en coreografías para permitir un cierto grado de distribución de procesos de negocios, en el cual los nodos de IoT necesitarán los mecanismos de ejecución del modelo de Procesos de Negocios (BP) y las capacidades de autogestión. Además, la distribución de la ejecución del proceso en los nodos puede mejorar aún más la flexibilidad, agilidad y adaptabilidad del BPMS-IoT, igualmente esta solución solamente sería viable en aquellos dispositivos de IoT que cuenten con una gran capacidad de procesamiento, para poder soportar la ejecución local de los procesos.

4.3 Modelado de soluciones basadas en procesos con IoT

Durante esta etapa la decisión a tomar, es entre mantener la notación de modelado existente (BPMN) o considerar extensiones a dicha notación, agregando nuevos elementos.

La introducción de los elementos de IoT en BPMN no es una tarea sencilla porque los dispositivos de IoT pueden ser heterogéneos en términos de protocolos de comunicación, topologías de red (conectividad), especificaciones de hardware (potencia informática y duración de la batería). En general, existen dos enfoques para modelar las entidades de IoT: 1- Expresar dispositivos de IoT como servicios o 2- Definir nuevos elementos notacionales para IoT.

En el primer caso, modelar los dispositivos IoT como servicios basados en URI puede simplificar la integración y también ser totalmente compatible con las herramientas existentes como BPMN o BPEL. En este enfoque, el sistema expresa los dispositivos de IoT como servicios de red regulares SOAP [33] basados en XML, que se puede comunicar a través del mecanismo de

solicitud-respuesta regular. En general, en este enfoque, los diseñadores de modelos de BPM asumieron que el sistema se puede conectar a dispositivos IoT directamente en función de la comunicación del servicio web. Sin embargo, en los sistemas del mundo real, muchos dispositivos de IoT no funcionan como entidades de servicios web normales. Por lo tanto, numerosos investigadores se están centrando en definir elementos nuevos para BPM-IoT.

En el segundo caso, al definir nuevos elementos a la sintaxis de BPMN que representen a los elementos de IoT, buscan abordar la realidad propia de las ventajas y desventajas de los sistemas IoT, que no se hacen presentes en los sistemas operativo/transaccionales clásicos [34] como, por ejemplo, la falta de conectividad y, en consecuencia, la interrupción del flujo del proceso, así como la existencia de servicios intermediarios para suplir estas contingencias.

Como se afirma en [35], cuando el proceso incluye tareas continuas como la transmisión de datos de sensores y eventos, las herramientas de modelado BPM basadas en estándares existentes no pueden definirlo explícitamente. En tales casos, se requiere incluir elementos de IoT específicos en el modelado de procesos para ampliar la potencia expresiva al modelo producido.

Los participantes en una solución IoT han sido analizados por diferentes autores como [36] y [37] que señalan el Proceso de IoT como un Pool.



Fig 23 – Anotaciones BPM-IOT versión proyecto IoT-A

Específicamente, el proyecto IoT-A ha modelado representado cada tipo de dispositivo IoT como un “Lane” y ubicándolos todos juntos dentro de un proceso, como se puede observar en la imagen anterior en el cual los dispositivos de IoT se encuentran dentro del proceso “IoT System”.

Por el contrario, makeSense propuso un diseño diferente, que separó específicamente los procesos relacionados con IoT por completo en diferentes Procesos, Por ejemplo, la próxima figura muestra un ejemplo en el que el modelo separa los procesos relacionados con IoT del Sistema Central.

Los distintos autores proponen diferentes elementos de modelado para IoT que resumiremos a continuación:

Entidad física.

Meyer en [34] presenta específicamente un análisis exhaustivo sobre cómo BPMN debería significar las entidades físicas del mundo real (un chocolate, una botella de leche, un animal) debemos tener en cuenta que el sistema puede interactuar con entidades físicas con protocolos muy heterogéneos, la anotación de texto y el objeto de datos son menos factibles para representar las entidades físicas. Por lo tanto, el resultado de su discusión muestra que prevalece la utilización de la notación de utilizar un “Lane”, definiendo al elemento como participante en BPMN para anotar entidades físicas.

FLUJO DE PROCESO BPM-IOT

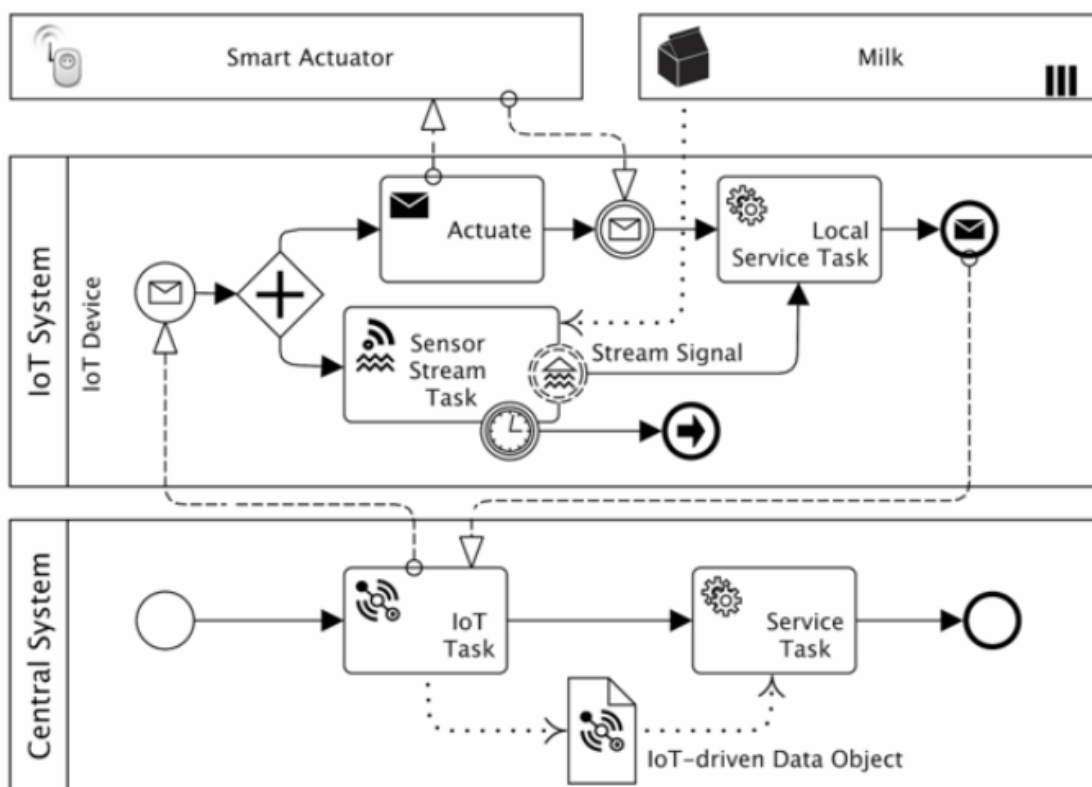


Fig 24 – Anotaciones versión del proyecto MakeSense

Actuador

Un actuador es un dispositivo que puede ejecutar comandos sobre objetos físicos. Por ejemplo, un sistema de IoT puede encender o apagar una luz de forma remota a través de un actuador de interruptor de luz interconectado. Comúnmente, los marcos existentes utilizan Tarea o Tarea de servicio para anotar los actuadores involucrados en actividades en BPMN. Sin embargo, en el trabajo de Yousfi [38, 39], introdujeron además el evento de inicio específico y las notaciones BPMN del evento intermedio para aclarar qué tipo de dispositivos actuadores se utilizan en los eventos.

En general, el enfoque para introducir en BPM tareas relacionadas con IoT es reemplazar los símbolos de notación (por ejemplo, el círculo de línea discontinua).

Sensor

Los sistemas de IoT utilizan sensores para adquirir datos específicos como el brillo, la temperatura, el movimiento de una entidad, la dirección del movimiento, etc. En general, los marcos existentes tienden a modelar los sensores de manera diferente. Las actividades de los sensores generalmente se diseñan como la extensión de la Tarea o la extensión de la Tarea de Servicio con un símbolo específico. Además, Yousfi introdujo los eventos de sensor con el mismo propósito que se aplicaba a los actuadores. descrito en el párrafo anterior.

Eventos

Appel [40] introdujo un elemento específico — *Stream Process Unit* (SPU) para integrar los procesos de transmisión continua de eventos. Las principales diferencias entre la tarea SPU y los elementos BPMN existentes, como la tarea de servicio, el bucle o la operación en paralelo son:

- (1) la SPU se basa en la publicación y suscripción del proceso de transmisión de eventos que opera en forma aislada del flujo de trabajo del proceso principal;
- (2) SPU realiza el proceso de forma continua para cumplir con un solo proceso, que es diferente al bucle que está repitiendo ciertos procesos (por ejemplo, tarea de flujo de sensor y señal de flujo en la Fig. 20).

Objeto de datos específico.

El objeto de datos en el BPMN clásico representa un dato o un archivo que se transmite de una actividad a otra. Por lo general, se trata de una transmisión única. Se introdujo el elemento de objeto inteligente, que es una subclase de objeto de datos en BPMN. El elemento Objeto inteligente consta del atributo de tipo para describir la fuente de los datos (por ejemplo, los datos fueron recopilados por Tarea de sensor o Tarea de lector).

En SPUs [40], los autores introdujeron el objeto de datos *Event Stream*, que es diferente al objeto de datos clásico en BPMN que representa un solo flujo de transmisión de datos.

El objeto de datos *Event Stream* representa los datos de transmisión independientes que se ingresen o emiten continuamente a través del sistema de flujo de trabajo. Este enfoque puede especificar la transmisión de información de los sensores en un escenario de IoT, que no se considera en el diseño del modelo de BP clásico.

La mayoría de los marcos introdujeron elementos de IoT para adaptarse a escenarios específicos. El elemento más conflictivo entre los autores es el elemento sensor. Algunos trabajos prefieren modelar estos dispositivos como sistemas de información individuales, pero algunos trabajos prefieren ocultar los detalles y solo consideran las tareas de los sensores correspondientes.

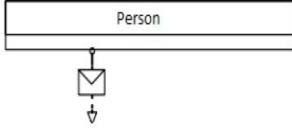


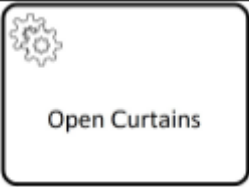
Desde la perspectiva de las implementaciones de IoT en el mundo real los dispositivos sensores están conectados a una red IP, e idealmente se puede

acceder a ellos como un servicio RESTfull basado en URI, el cual es completamente compatible con BPMN 2.0.

Por lo general, los enfoques de modelado existentes no han considerado los dispositivos móviles de IoT en la especificación de su modelo. Los dispositivos móviles de IoT (por ejemplo, dispositivos portátiles, dispositivos voladores, dispositivos portátiles, etc.) tienen estados dinámicos, incluida su ubicación, dirección de movimiento, conectividad, etc. Estos estados influyeron mucho en la operación del proceso. A efectos de transparencia y agilidad, este contexto debe considerarse en el modelo para reaccionar rápidamente a los eventos.

En la Tabla 4, se comparan las soluciones vistas anteriormente, indicando para cada elemento cual es utilizado en esta propuesta.

Tabla 4. Propuestas de notaciones BPM-IoT [41]

Entidad IoT	Representación BPM	Propuesta de autores	Implementación propuesta con Bonita Soft
Usuario		Meyer propone modelar el usuario como un proceso que envía un mensaje solicitando un recurso.	Proponemos implementarlo como un Lane del Pool del proceso
Entidad Física		Meyer propone modelar la entidad física como un proceso pasivo.	Lo modelaremos utilizando anotaciones de texto.
Dispositivo de IoT		Meyer propone modelar el dispositivo de IoT representándolo como un lane, debido a que es un participante más del modelo.	Proponemos implementarlo como un Lane del Pool del proceso
Recursos y Servicios		Sperner explica que el servicio es similar al elemento BPMN Service Task.	Utilizaremos el mismo elemento para representarlo en Bonita Soft.

4.4 Ejecución de soluciones basadas en procesos con IoT

La ejecución de una solución basada en procesos, consiste en transformar el modelo de BP abstracto en el programa de software ejecutable y legible por una máquina.

Los desafíos en esta fase para poder integrar BPM y IoT se relacionan principalmente con la falta de herramientas. Por lo general, los modeladores de BP diseñan los modelos de BP en herramientas gráficas, y las herramientas pueden generar los metadatos legibles por máquina para permitir que el motor de flujo de trabajo ejecute los procesos.

Sin embargo, las herramientas comunes como Activiti [42], Camunda [43], BonitaBPM [8], Apache ODE [44] no admiten muchos de los protocolos utilizados en los dispositivos de IoT, como CoAP y MQTT. Actualmente, no existe una herramienta que pueda abordar todos los protocolos utilizados por los dispositivos de IoT. Un enfoque común es introducir una capa de middleware para aprovechar el servicio integrado de los dispositivos de IoT con los servicios web SOAP o REST comunes. Sin embargo, sin la solución adecuada, el sistema puede sacrificar la transparencia (en la perspectiva de BPM), el rendimiento (gastos generales adicionales) y la agilidad (reacción de los eventos en tiempo de ejecución).

Una alternativa es modelar los procesos en un lenguaje ejecutable como BPEL [45]. El lenguaje de ejecución de procesos de negocio de servicios web (WSBPEL; o BPEL en breve) es un lenguaje de composición de servicios ejecutable e interoperable en el que BPEL es totalmente compatible con los Servicios Web. Idealmente, los metadatos BPEL se pueden ejecutar en cualquier motor compatible con el estándar siempre que los motores hayan obtenido las descripciones correspondientes requeridas (WSDL, esquema XML, etc.). Cabe destacar que la mayoría de los proyectos nombrados en el presente trabajo han planteado BPMS-IoT basado en BPEL.

Si bien las extensiones BPMN otorgan flexibilidad sumando nuevas entidades y elementos de IoT, no aporta interoperabilidad entre diferentes modelos de diseño porque están estrechamente ligados al motor de ejecución correspondiente y, por lo tanto, dan como resultado soluciones aisladas.

Otro aspecto ausente en los estándares existentes para la ejecución de los procesos con IoT es la distribución.

La ejecución del proceso representa cómo los motores de flujo de trabajo ejecutan el diagrama de modelo de proceso transformándolo en código legible por máquina. En general, los motores de flujo de trabajo son alojados en el servidor de administración central o en los dispositivos IoT.

Considerando el sistema centralizado clásico, donde el sistema no requiere realizar procesos complejos en los participantes, los motores de flujo de trabajo solo se alojan en el sistema central, en el que los dispositivos de IoT pueden operar como los servicios regulares basados en operaciones de solicitud/respuesta.

Por el contrario, si el sistema requiere que los procesos se ejecuten de forma distribuida, los dispositivos de IoT pueden necesitar incorporar motores de ejecución de procesos.

En general, hay tres tipos de escenarios posibles en BPMS-IoT para soluciones distribuidas, vamos a describir dichos escenarios a continuación.

Ejecución del motor de procesos únicamente en el servidor

Es el caso (a) de la figura 25, el nodo participativo solo realiza ciertas tareas basadas en el mecanismo de solicitud/respuesta o publicación/suscripción.

Por ejemplo, en el escenario de un hogar de ancianos descrito en [46], el BPMS del hospital puede asignar tareas de forma remota a las enfermeras a través de sus dispositivos móviles. Además, dado que los dispositivos móviles actúan como mediadores, pueden asignar tareas manuales a las enfermeras o recuperar los datos de los pacientes al control remoto BPMS hospitalario.

Ejecutando el motor de procesos únicamente en el servidor y enviando el código para su ejecución en el dispositivo de IoT

En este enfoque, el sistema traduce los metadatos del modelo BP generados por el editor del modelo al código binario ejecutable en un lenguaje de programación específico. En particular, este enfoque puede ser más adecuado para dispositivos que cuenten con recursos limitados para ejecutar los modelos de BP. Generalmente, en este enfoque, el sistema se basa en las tecnologías de middleware para traducir el modelo de BP al código ejecutable, luego envía el código a los dispositivos IoT para su posterior ejecución.

En particular, numerosos marcos BPMS-IoT han utilizado este enfoque para permitir que los dispositivos IoT participen en la ejecución de BP sin la necesidad de incorporar componentes de software intermedio complejos (por ejemplo, motor de flujo de trabajo) en los dispositivos.

Ejecución del motor de procesos en el dispositivo de IoT

La migración o implementación directa de un motor de flujo de trabajo en el dispositivo de IoT permite la mejor flexibilidad. Además, también permite una forma flexible de realizar la coreografía del proceso entre el servicio en la nube de back-end y los dispositivos de IoT de front-end o entre numerosos dispositivos de IoT, en los que los dispositivos de IoT situados en la red están ejecutando el flujo de trabajo [46].

Sin embargo, tal mecanismo requiere de dispositivos de mayor potencia computacional (por ejemplo, teléfonos inteligentes de alta gama). Mientras tanto, numerosos proyectos de investigación han introducido motores de ejecución de flujo de trabajo basados en estándares para sistemas operativos móviles.

POSIBLES INTEGRACIONES BPM-IOT

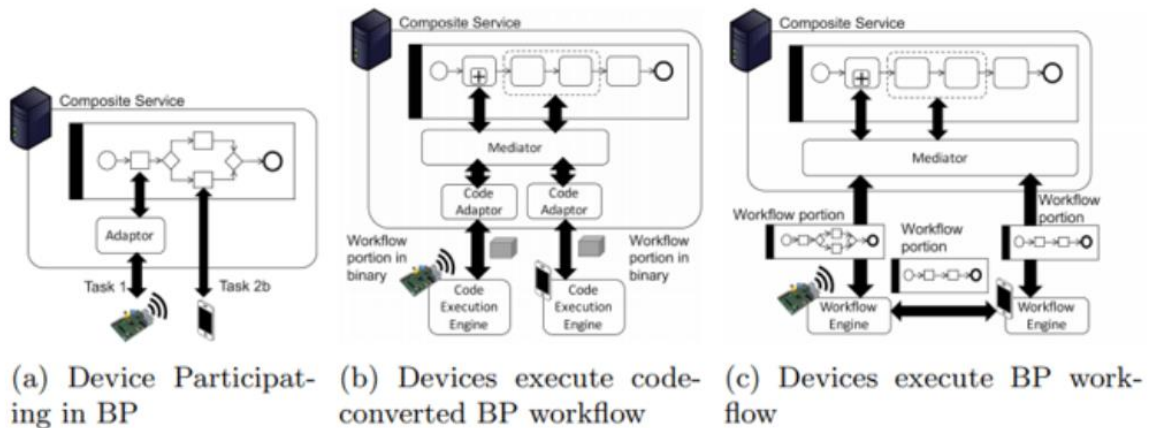


Fig 25 – Posibles modelos distribuidos

El inconveniente del enfoque de código compilado del modelo de BP en ejecución es que cada vez que cambia el modelo de proceso, el sistema necesita convertir nuevamente el modelo al código fuente ejecutable y desplegar el código en los dispositivos de IoT.

Por el contrario, el enfoque del motor de flujo de trabajo integrado puede realizar cambios rápidos en el modelo y la ejecución. Sin embargo, también puede consumir más recursos de hardware del dispositivo.

Los motores brindan la posibilidad de ejecución de flujo de trabajo de procesos de negocios distribuidos entre la nube a móvil y de móvil a móvil, en la que el sistema puede admitir una composición basada en coreografías para llevar la tecnología móvil altamente flexible y escalable basada en la nube. BPMS, que puede ser útil para muchos sistemas de IoT, como la logística y los casos de uso de AAL descritos anteriormente.

Fase de ejecución y ajuste en procesos con IoT

La fase de ejecución y ajuste representa el tiempo de ejecución del sistema después de la implementación del BPMS. En general, esta fase no implica ningún rediseño. Específicamente, dado que la fase de ejecución y ajuste solo realiza las actividades de gestión predefinidas, el BPMS debería proporcionar mecanismos de adaptación para la administración del sistema. Por ejemplo, el sistema debe registrar el historial del tiempo de ejecución, manejar los eventos y permitir los ajustes dinámicos.

Los siguientes son los principales mecanismos que debería adoptar un BPMS que quiera incluir dispositivos de IoT en su etapa de ejecución y ajuste.

4.5 Monitoreo de soluciones basadas en procesos con IoT

Comúnmente en un diseño como un entorno de IoT basado en servicios en la nube, el sistema requiere múltiples capas para permitir el monitoreo de la actividad de los nodos de IoT. En otras palabras, los dispositivos de IoT deben conectarse a los dispositivos que funcionan como intermediarios para permitir la comunicación entre ellos y los sistemas back-end mediante protocolos de publicación y suscripción (por ejemplo, MQTT).

El monitoreo en la fase de ejecución y ajuste involucra dos tipos: monitoreo del proceso y monitoreo del estado del dispositivo. El monitoreo de procesos involucra la observación acerca de cómo el sistema opera las actividades de IoT y cómo maneja los eventos en tiempo de ejecución. En esta etapa, el sistema recopila los archivos de registro de supervisión del proceso para su posterior análisis con el fin de mejorar el sistema en la fase de rediseño.

El monitoreo del estado del dispositivo, por su parte, juega un importante rol en las soluciones basadas en procesos con IoT, dado que las actividades del proceso involucran una gran cantidad de dispositivos de red inalámbrica, donde la falla del dispositivo en tiempo de ejecución puede comprometer el proceso completo.

Monitorear los dispositivos externos es una tarea crucial porque puede involucrar muchos factores, como el posible daño que puede ocurrir en el hardware del dispositivo final de IoT, o en si el clúster de la red perdió su conectividad con sus otros pares, el nodo del intermediario de datos perdió su conectividad con el sistema backend debido a una falla del hardware o del proveedor de Internet.

En general, la mayoría de los frameworks BPMS4IoT existentes no han estudiado ampliamente los problemas de fallas del dispositivo en tiempo de ejecución. Entre los marcos existentes, aplicó un enfoque genérico para manejar la falla del dispositivo en tiempo de ejecución. En su enfoque, el sistema asigna una tarea que envía informes periódicos a cada dispositivo del frontend. Si el servidor central no recibe el informe de un dispositivo en particular que excede el umbral de tiempo de espera, el sistema considerará que el dispositivo ha fallado. Por tanto, el sistema realizará la sustitución.

4.6 Otros aspectos a considerar en una solución BPM-IoT

Tolerancia a fallos

El paradigma de IoT implica una gran cantidad de dispositivos conectados a redes inalámbricas. Generalmente, estos dispositivos tienen menos estabilidad en conexión y con la capacidad de batería baja. Por lo tanto, un sistema adecuado debe considerar los dispositivos con recursos limitados y respaldar las soluciones correspondientes, como la sustitución reactiva de los dispositivos / actividades fallidas con sustituciones de forma inmediata y autónoma para retener los procesos.

Por ejemplo, considerando la conexión a Internet móvil poco confiable, los autores en [48] han propuesto un marco para distribuir y ejecutar tareas en los nodos móviles de la red de borde en modo fuera de línea.

Explícitamente, con tal mecanismo, cuando un nodo pierde su conexión a Internet con el sistema de gestión, el dispositivo de IoT aún podría continuar ejecutando sus procesos. Además, pueden enviar la salida del proceso y los registros monitoreados al sistema de administración distante una vez que estén nuevamente conectados.

Conocimiento del contexto

El contexto es cualquier información que se pueda utilizar para caracterizar la situación de una entidad. Una entidad es una persona, lugar u objeto que se considera relevante para la interacción entre un usuario y una aplicación, incluido el usuario y las aplicaciones mismas.

BPMS-IoT debe abordar la conciencia del contexto en términos de contingencias, personalización y eficiencia.

Las contingencias involucran la conectividad y la accesibilidad impredecibles de los servicios generalizados y los dispositivos de red inalámbrica.

Generalmente, hay dos esquemas para abordar las contingencias en BPMS-IoT: proactivo y reactivo.

El esquema proactivo tiene como objetivo prevenir la aparición de problemas de los dispositivos de IoT en tiempo de ejecución. Por ejemplo, en el proyecto MOPAL, los autores han definido ciertas reglas sensibles al contexto (por ejemplo, capacidad actual de la CPU, nivel de batería, ubicación geográfica, etc.) que limitan la ejecución de la tarea del flujo de trabajo en los dispositivos IoT.

El esquema reactivo comúnmente busca la sustitución de la ejecución de la tarea del flujo de trabajo. En el sistema de automatización del hogar BPMS basado en SOA propuesto por Chang y Ling [50], los dispositivos conectados pertenecen a categorías específicas en función de su contexto asociado. Por ejemplo, ambas tareas del flujo de trabajo, "sonido de alarma" y "encender TV → subir el volumen de TV", pueden generar el mismo tipo de contexto: "ruido fuerte" para despertar al usuario. Por lo tanto, cuando la configuración predeterminada, "alarma de sonido" falla por cualquier motivo, el sistema puede activar la sustitución, que es "encender la TV → subir el volumen de la TV" para lograr el mismo propósito. La personalización implica el suministro de servicios en función de las preferencias de los solicitantes. Por ejemplo, en el dominio de la informática ubicua inteligente, la conciencia del contexto generalmente ha considerado el contexto de la entidad.

Eficiencia energética

Es una de las principales preocupaciones en las soluciones IoT. Para conservar la energía de los dispositivos IoT, básicamente cada tarea de IoT debe asociarse con temporizadores para controlar el tiempo de suspensión/activación del dispositivo.

Despliegue

La implementación de dispositivos IoT individualmente puede ser costosa. Una posible estrategia es reutilizar el sistema ya implementado y proporcionar una plataforma de integración, como el servicio en la nube basado en SOA, que pueda habilitar la necesidad y también proporcionar la interoperabilidad entre los BPMS de diferentes organizaciones. Un BPMS-IoT puede comparar entre el costo de implementar sus propios dispositivos y el costo de utilizar los recursos de terceros.

Comunicación

El costo de la comunicación en tiempo de ejecución puede ser muy dinámico, no solo debido a los cambios de tráfico del proveedor de servicios de Internet (ISP), sino también influenciado por la cantidad de dispositivos IoT involucrados y la ubicación del solicitante. Por ejemplo, cuando una aplicación de IoT ubicada requiere el uso de WSN en su entorno, el proceso se basa principalmente en el centro de datos distante. Explícitamente, este enfoque puede causar una alta latencia cuando la conexión a Internet no está en buenas condiciones. Por lo tanto, muchos enfoques utilizan recursos basados en proximidad para aplicaciones intensivas de procesamiento y adquisición de datos.

Escalabilidad

La escalabilidad es uno de los principales requisitos para la gestión de dispositivos conectados a gran escala. En los marcos BPMS-IoT existentes, la escalabilidad involucra dos temas específicos: el volumen creciente de datos de flujo y el número creciente de dispositivos participativos de BP.

Capítulo 5. Caso de Estudio

Durante el presente capítulo explicaremos el caso de estudio seleccionado y cada una de las decisiones de implementación adoptadas teniendo en cuenta las consideraciones del marco teórico descrito en los capítulos anteriores.

Este trabajo además de plantear una solución arquitectónica para la integración de IoT con un sistema de gestión BPM ya existente en la institución, se enfoca en las ventajas económicas de adaptar maquinarias industriales que ya cuentan con muchos años de uso, para poder incluir nuevas tecnologías de sensores y actuadores, convirtiéndolas en piezas de una red de IoT con un costo mínimo para la organización.

Se propone la implementación de una red de sensores que puedan tomar en tiempo real información de las máquinas y enviarlas a una aplicación Web que se encarga de almacenarlas en su base de datos, para que luego los administradores de la planta puedan observar en tiempo real cuántas unidades llevan completas cada orden de trabajo y el tiempo en que estuvo productiva cada máquina para poder tomar mejores decisiones de negocio.

La fábrica utilizada para probar el caso de estudios se dedica a la producción de bolsas de plástico comerciales, sus principales clientes son bancos y correos, también realiza bobinas de plástico para ser utilizadas por otras industrias.

Durante la tarea de producción de la fábrica se ejecutan diferentes procesos de producción, primero se prepara el material líquido por las máquinas extrusoras, logrando un material sólido, que luego atraviesa un conjunto de rodillos, dispuestos en impresoras, para imprimir los gráficos de las bolsas.

Las máquinas denominadas confeccionadoras se encargan de cortar por unidad las bolsas que se generaron en los procesos anteriores, para eso cuenta con una guillotina que automáticamente, según las indicaciones de medida que le ingresa el operario, realiza el corte de cada una de las unidades. Las máquinas cuentan con distintos conjuntos de rodillos por los cuales va a pasar el material.

El sector de administración crea órdenes de trabajo indicando las unidades que deben fabricarse, el tiempo estimado de trabajo y la velocidad de funcionamiento de la confeccionadora. Este proceso cuenta con el inconveniente de que la información de la producción actual, junto con el tiempo de utilización y detención de la máquina son ingresados manualmente por un operario de la planta en una planilla de papel, la que luego al finalizar su turno, es dejada en administración para su posterior procesamiento.

Esto provoca que la disponibilidad de la información de producción de la planta sea lenta e insegura debido a los errores humanos que pueden existir al momento de completar las planillas.

La fábrica cuenta con 14 máquinas industriales entre confeccionadoras, extrusoras e impresoras para llevar a cabo la producción de las bolsas de plástico, en cada unidad se instaló una placa raspberry, con sus respectivos

sensores de efecto hall para medir la producción, en las confeccionadoras estos sensores se instalaron en las guillotinas de corte, mientras que en el resto de máquinas se instalaron en sus rodillos.

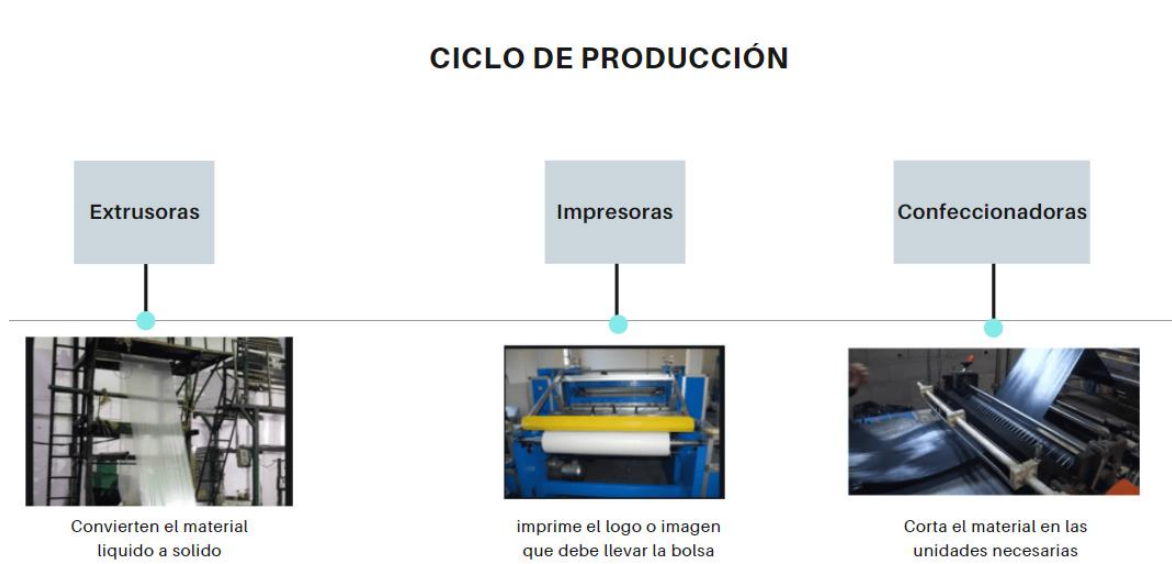


Fig 26 – Ciclo de producción de la fábrica

El caso de estudio propone utilizar una arquitectura de orquestación, mediante una pasarela para adaptar nuestra propia plataforma IoT a las necesidades del enfoque BPM, posibilitando de esta manera que la arquitectura y limitaciones anteriormente mencionadas sean transparentes para el proceso de negocio que se está ejecutando, centrándonos en construir una red de sensores (IoT) que realicen la lectura de la cantidad de cortes realizados por la confeccionadoras y las envíe a un sistema de gestión de procesos (BPMS) utilizando un middleware para favorecer su interacción.

A continuación, explicaremos cada una de las decisiones de diseño e implementación que tome en cada una de las etapas.

5.1 Arquitectura

Una solución arquitectónica proporciona una estructura completa con el objetivo de que los objetos que forman parte del universo del problema puedan interactuar entre ellos. El mundo de IoT tiene sus propias reglas en términos de tiempo de respuesta, unicidad del proveedor, confiabilidad de la red, persistencia del proveedor de servicios, rendimiento, carga útil, entre otros.

En este sentido, uno de los enfoques arquitectónicos de IoT se basa en datos: los sensores miden el mundo físico, los datos se envían a un punto central con potentes capacidades de procesamiento.

Antes de presentar la arquitectura propuesta y que utiliza la orquestación de servicios mediante un BPMS, se analizan algunos aspectos problemáticos de las arquitecturas de IoT como lo son: 1- Dispositivos heterogéneos, 2- Cantidad de dispositivos, 3- Uso de ancho de banda.

Las arquitecturas IoT brindan soluciones para la identificación de objetos y su localización y deben reaccionar ante la presencia de dispositivos heterogéneos, que difieren en términos de protocolos de red, sistemas operativos y software utilizados para resolver el mismo problema. A medida que las industrias brindan herramientas para acceder a estos dispositivos, los usuarios pueden encontrar problemas para hacerlos interactuar (mismo fabricante, mismo protocolo) o simplemente recopilar datos en el mismo formato (desde temperatura hasta movimiento, valor del acelerómetro, sensibilidad, etc).

Otro problema es el número de objetos que están en juego para un propósito definido. Por ejemplo, el valor de temperatura recopilado para una parte de un edificio puede ser el promedio de los datos recopilados de un número importante de dispositivos. La reacción a este valor agregado puede ser encender un conjunto de acondicionadores de aire. Esta especificidad de un valor / servicio único que se recopila / activa a partir de múltiples fuentes, es diferente del enfoque común de BPM en el que un servicio/evento es único en su formato y fuente. Esta rigidez de implementación en BPM funciona bien cuando la situación en cuestión no es dinámica y no evoluciona en un período de tiempo muy corto. Pero en el contexto de IoT, tal rigidez casi actúa como una desventaja.

Además, la multiplicidad conduce a otro inconveniente importante, ya que algunos elementos pueden fallar y pueden ser reemplazados por otros nuevos, que pueden diferir en términos de API (s), servicios y formato o una combinación de ellos. Como conclusión, la comunicación directa entre objetos de IoT y los BPMS parece difícil de lograr.

El último problema está relacionado con el ancho de banda y el rendimiento de la red. Estas características son fluctuantes y, a menudo, limitadas en comparación con los valores habituales que se encuentran en las redes cableadas. Debido a sus limitaciones específicas (especialmente la energía), las redes inalámbricas que utilizan los sensores y actuadores son lentas y poco fiables. Para evitar la pérdida de energía, el número de mensajes enviados (según la dirección) puede ser limitado en el tiempo. Por ejemplo, los protocolos de largo alcance (como Lora o Sigfox) que utilizan los dispositivos integrados pueden enviar datos con más frecuencia de la que reciben (solo unos pocos mensajes por día).

Por lo tanto, el enfoque distribuido puede no ser la mejor solución para integrar IoT en BPM, ya que su efectividad puede verse cuestionada, al no contar con dispositivos capaces de poder ejecutar de forma correcta los procesos de negocio, debido a sus limitaciones de recursos.

La solución que proponemos en el siguiente caso de estudio es utilizar un modelo de orquestación para definir la arquitectura que cuente con una capa de middleware que permita introducir un vínculo débilmente acoplado entre los dos dominios, proporcionando una integración más simple.

Este puente de software puede verse como un enlace entre la red de IoT, que es dinámica, y la arquitectura BPM, que es estática.

Realizando un primer acercamiento a la solución planteada podemos observar a todos los elementos del universo de BPM e IoT como servicios a los cuales podemos solicitar algo y nos brindan una respuesta en un formato particular. Hay similitudes entre los elementos de IoT y BPM cuando los imaginamos solamente como servicios a los cuales les pedimos algo y nos responden (*smart object*). Recordemos que el propósito de la aplicación de IoT es principalmente acceder a los datos recopilados por los dispositivos o utilizar la capacidad de los mismos para poder realizar una acción en el mundo físico. Para facilitar la integración con un paradigma de programación conocido, lo ideal es dar acceso a los datos y acciones proporcionados por los objetos (*Smart Objects*) a través de una API REST, aprovechando que es una metodología muy conocida entre los programadores.

ARQUITECTURA DE MIDDLEWARE

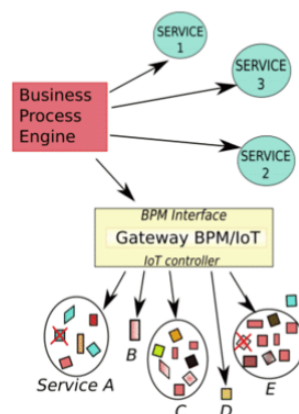


Fig 27 – Ejemplo de capa de middleware

La capa de middleware funciona como un traductor, ofreciendo una interfaz para que los procesos BPM puedan solicitar información a la red de dispositivos de IoT y otra interfaz para comunicarse con los dispositivos de IoT que pueden contar con múltiples protocolos de comunicación.

En el lado de IoT, la puerta de enlace conoce las diversas tecnologías utilizadas y sus diferentes APIs. Puede recopilar los datos, los calcula en caso de que el BPMS necesite un valor único hecho de múltiples fuentes y almacena los valores de las lecturas para proporcionar un tiempo de respuesta rápido, luego cuando existan peticiones se pueden transmitir al BPM a través de la puerta de enlace.

También la comunicación podría darse en sentido inverso y de esta forma BPM puede desencadenar acciones en el mundo real a través de IoT.

Podemos concluir que en nuestro modelo siempre los eventos enviados por los dispositivos de IoT son utilizados como entradas en el BPMS con el fin de iniciar un proceso asociado específico a nivel BPM y a su vez una salida de un

BP puede que desencadenar una respuesta física de algún dispositivo de IoT, logrando que ambos se comuniquen sin conocer la arquitectura, ni los protocolos que utilizan ya que la comunicación se realiza utilizando de intermediario a la capa de middleware.

A continuación, presentaremos la arquitectura planteada para probar el caso de estudio, y cada una de las capas que la contienen.

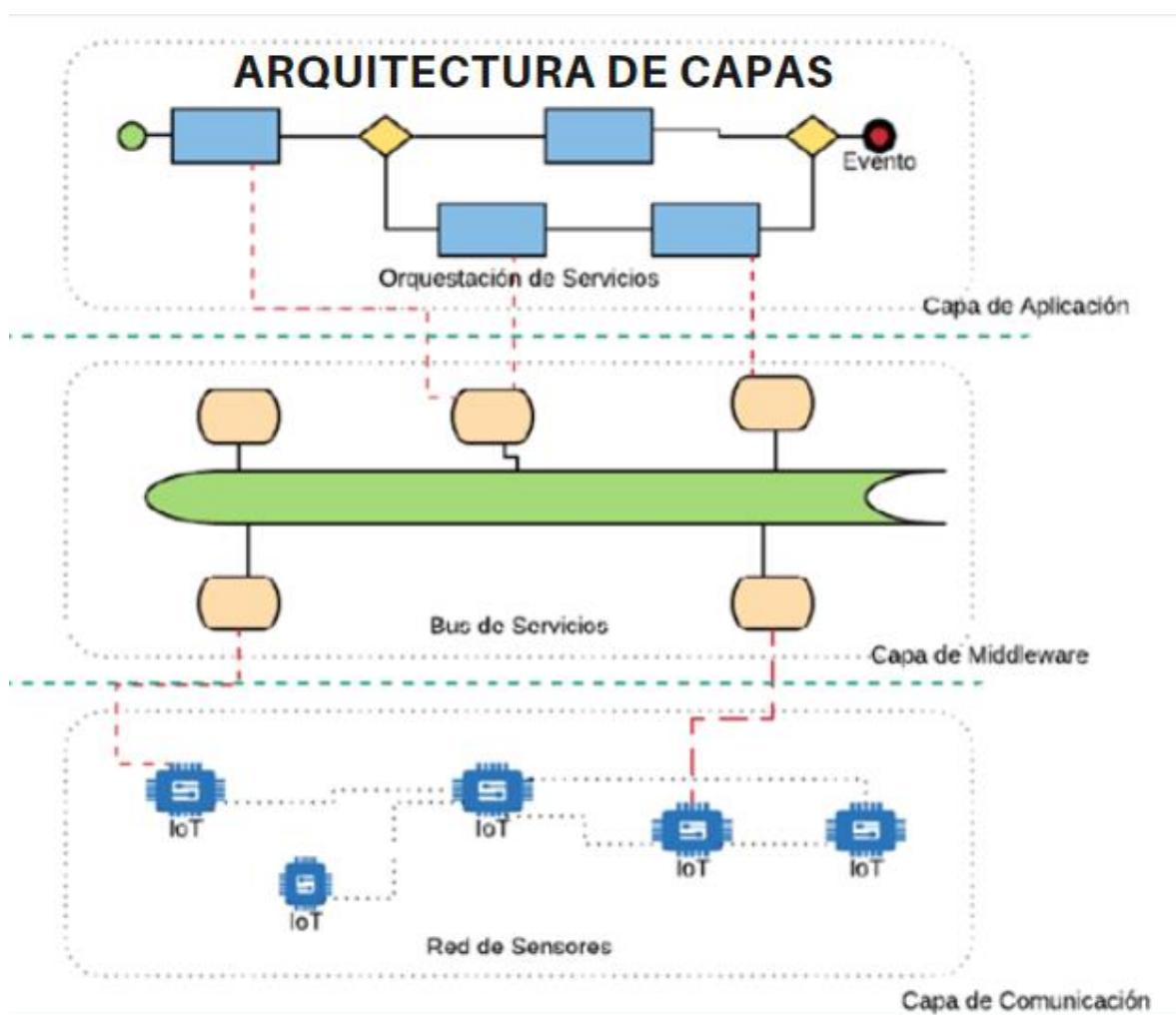


Fig 28 – Arquitectura de capas planteada en el trabajo

1. Capa de Comunicación: está conformada por una red de sensores que producen la información que alimenta a los sistemas de la organización. Esta capa posee sus propias reglas y protocolos de comunicación y establece un mecanismo uniforme

para propagar sus registros hacia una capa de servicios.

2. Capa Middleware: está conformada por servicios o componentes, que son las piezas de software que se comunican con la red de sensores y procesan los datos obtenidos para disponerlos al usuario.

3. Capa de Aplicación: realiza la orquestación de los servicios y resuelve los mecanismos y circuitos propios del dominio conformando el flujo de trabajo (workflow).

5.2 Capa de aplicación. Modelado

Luego de plantear una posible arquitectura, se definirá un posible modelo para el caso de estudio.

Entre los diferentes autores analizados para realizar este trabajo, existen dos ideas principales de cómo debe poder integrarse el modelado de un proceso de negocio BPD en una solución de IoT, algunos autores proponen agregar elementos nuevos a la notación BPM para poder representar a los dispositivos de IoT y otros proponen reutilizar los que ya existen, encapsulado y abstrayendo los elementos de IoT.

En nuestro modelo recurriremos a la solución de reutilizar los elementos existentes de la notación BPM, con el objetivo de brindar un modelo para las organizaciones que ya cuentan con un modelo de procesos implementada que les ocasione realizar los menores cambios posibles, para reducir sus costos de implementación.

Se presenta el modelo obtenido utilizando la notación propuesta para modelar un proceso de negocio de la Planta de producción de bolsas de plástico. El proceso seleccionado mide la cantidad de unidades producidas por cada una de las máquinas.

En la capa de aplicación del caso de estudio se encuentra el motor de procesos siendo ejecutado por el BPMS de Bonita. En el cual cada uno de los requerimientos funcionales de la gerencia de la fábrica son representados mediante un BPD (Diagrama de proceso de negocio).

A continuación, veremos el diagrama del proceso que permite observar en la pantalla del sistema un tablero de control que resume la información de la producción de las máquinas para un periodo determinado.

- 1- Se inicia una nueva instancia del proceso al acceder a la opción del menú que representa al tablero de control.
- 2- luego se ejecuta la actividad "Consultar listado de dispositivos", quien mediante una invocación a la capa de middleware por API REST, obtiene los identificadores de todos los dispositivos que se encuentran registrados en el canal "Medir producción".
- 3- Ejecuta la actividad "Consultar Unidades Producidas" quien consulta las unidades producidas por cada uno de los dispositivos del canal, para finalizar mostrando dicha información en la siguiente tarea, quien cuenta con un formulario para mostrar de forma clara los datos del tablero de control.

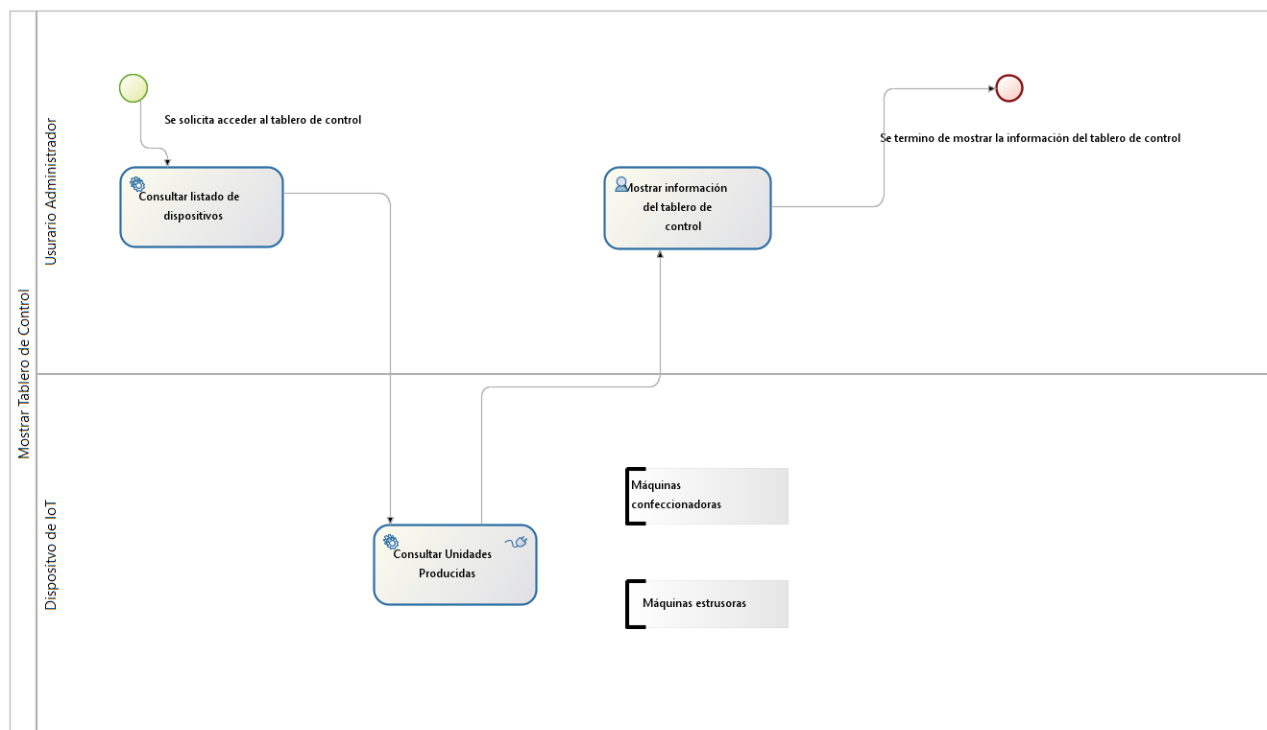


Fig 29 – BPD – Tablero de control

5.3 Capa de comunicación

A continuación, describiremos los elementos tecnológicos utilizados en la capa de comunicación.

- Raspberry PI 3 [52]: es una mini computadora del tamaño de una billetera, que a pesar de su tamaño reducido cuenta con una gran capacidad de procesamiento si la comparamos con otros dispositivos de IoT, cuenta con 1 GB de memoria RAM, un GPIO de 40 pines, 4 puertos USB, un puerto Ethernet y un módulo WIFI.

Los beneficios propuestos por este tipo de placas es su gran capacidad de procesamiento, ya que además de sensar las confeccionadoras, el staff de la empresa desea a futuro poder agregarle pantallas táctiles que permiten al operario mediante un aplicativo escrito en PYTHON poder observar información de la producción y poder interactuar con el sistema de IoT en la zona de producción de la planta.

- Sistema operativo Raspbedian [53]: es una versión del sistema operativo Debian optimizado para poder ejecutarse en las placas Raspberry PI. Cuenta con una interfaz de configuración completa para todos los parámetros configurables de la placa "raspicnf".
- Sensor de efecto Hall [54]: es un sensor con salida digital y retención de estado, que cambia de cero a uno ante la presencia de un campo

electromagnético. Cada vez que un imán esté dentro de su rango de sensado, el sensor cambiará su estado, de 0 a 1 o de 1 a 0 según su estado actual, teniéndolo hasta que de nuevo un imán se encuentre dentro de su rango de sensado, al contrario de otros sensores que solo cambian su estado cuando el imán está cerca.

- Python [55]: lenguaje de programación de propósito general, orientado a objetos, que también puede utilizarse para el desarrollo web.

PLACA RASPBERRY INSTALADA



Fig 30 – Raspeberry instalada en una de las maquinas

SERVIDOR INSTALADO EN LA FABRICA

```
10.0.1.2 - PuTTY
login as: root
root@10.0.1.2's password:
Linux IoT 4.9.0-9-amd64 #1 SMP Debian 4.9.168-1+deb9u5 (2019-08-11) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Mar  9 07:44:35 2021 from 10.0.1.254
root@IoT:~#
```

Fig 31 – Servidor instalado en una de las computadoras de la fabrica

5.4 Pasos del proceso durante la capa de comunicación

Cuando empieza el proceso de tomar lecturas de los sensores (Capa de Comunicación), las confeccionadoras cuentan con una cuchilla con la cual generan el corte de cada unidad, por lo tanto, se utilizó un sensor de efecto hall, que se activa al detectar un cambio en el campo magnético, en este caso sería cuando la cuchilla se acerca a la base de la máquina en donde está alojado el sensor para realizar un corte. En la siguiente figura se muestra el diagrama de conexión del sensor.

CONEXION DEL SENSOR

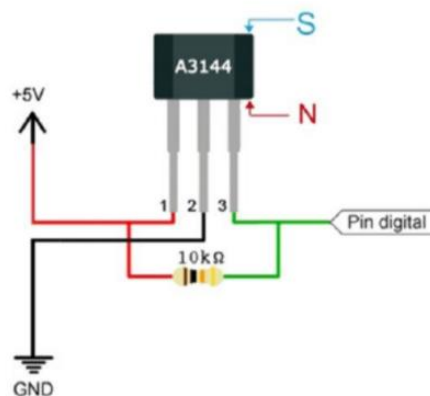


Fig 32 – Diagrama de conexión del sensor de efecto Hall

GPIO (Entrada Salida de propósito general)

Consta de una serie de pines o conexiones que se pueden usar como entradas o salidas para múltiples usos. Estos pines están incluidos en todos los modelos de Raspberry Pi aunque con cuentan con pequeñas diferencias entre ellos.

Los pines del GPIO cuentan con dos tipos de numeraciones para sus pines, nosotros dentro de nuestro código podemos definir si utilizamos la numeración física (BOARD) o la BCM

La opción se refiere a los pines por su número de "Broadcom SOC channel", estos no son correlativos como en el modo BOARD, en la imagen siguiente se muestran tanto los pines en BOARD como en BCM de las distintas versiones de Raspberry Pi, siendo los BCM los que su nombre comienza con GPIO y los centrales los pines BOARD. En Numeración GPIO, el número de pin se refiere al número en Broadcom SoC (*System on Chip*). Por lo tanto, siempre debemos considerar el mapeo de pines para usar el pin GPIO.

La placa raspberry realiza la lectura del sensor mediante código PYTHON en el GPIO correspondiente al pin digital del circuito que es por donde llegara la lectura del sensor. La información de la nueva lectura es enviada al servidor mediante un llamado a la API REST brindada por la capa de middleware.

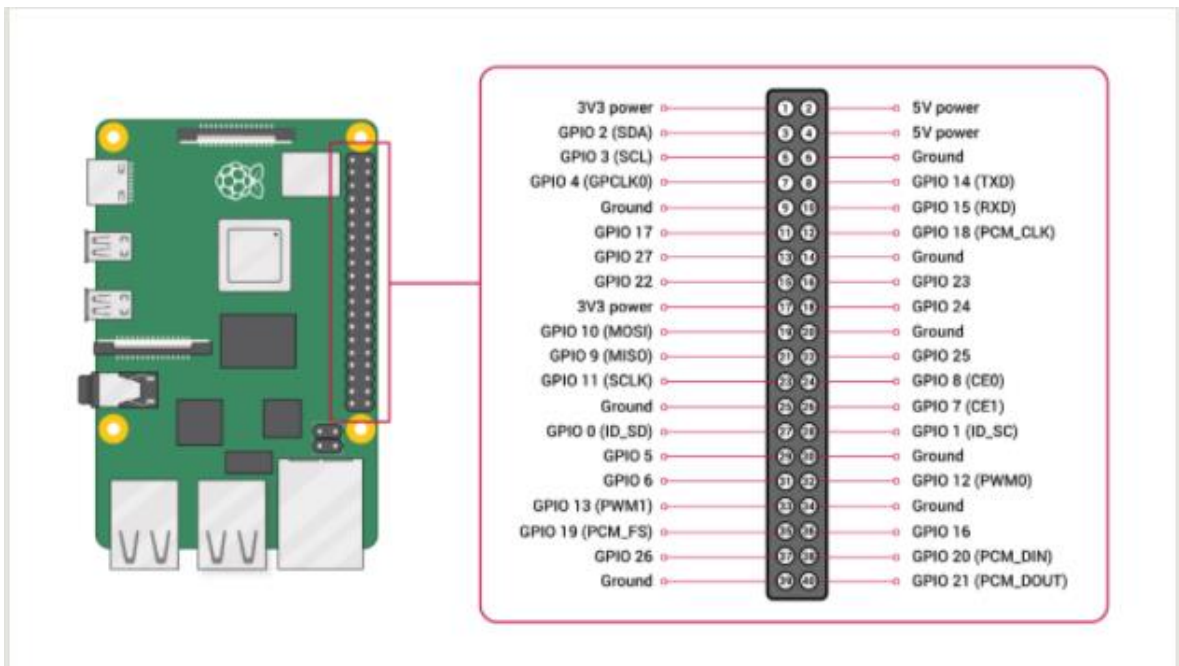


Fig 33 – Diagrama de pines GPIO - Raspberry

En la Figura 34, se muestra el código fuente para la lectura del sensor de efecto Hall. Durante la etapa de implementación del caso de estudio, se realizaron muchas versiones del código fuente que se ejecutaría en cada una de las maquinas industriales de la fábrica, en cada una de las versiones se buscó mejorar la calidad de la medición realizada por los sensores de efecto Hall, ya que una vez en la fábrica existían otros factores que no podíamos simular en nuestro laboratorio, como por ejemplo la interferencia generada por los motores de las maquinas, la velocidad de corte y movimiento diferente que existe entre cada una de las guillotinas de las confeccionadores.

La versión final del código que se observa en la Figura 34, inicia configurando la dirección del servidor remoto al que va enviar la información y cuál es el número de puerto del GPIO por el cual va ingresar la señal que indique que se generó una nueva lectura.

Luego inicia un bucle infinito en el cual cada 60 segundos transmite al servidor la cantidad de lecturas detectadas.

El sistema operativo está configurado para iniciar el programa cada vez que arranca, esto está pensado para lograr que ante un corte de luz por ejemplo el sistema se recupere de forma inmediata.

```

1 #Libreria para poder acceder a la información de las entradas y salidas de la placa raspberry
2 import RPi.GPIO as GPIO
3 #Libreria para poder utilizar las funciones de tiempo
4 import time
5 #Libreria para poder enviar las peticiones por API
6 import urllib
7 #Libreria para poder manipular los parametros de inicializacion del archivo
8 import sys
9 #El primer argumento representa el numero del GPIO en el cual debe buscar el pulso
10 port = sys.argv[1]
11 #El segundo argumento representa la URL de la API a la cual tiene que transmitir su información
12 server = sys.argv[2]
13 #Inicializaciones los contadores necesarios para la aplicacion
14 count = 0
15 last_time = 0
16
17 def send():
18     global count
19     url = server + '/' + str(count)
20     u = urllib.urlopen(url)
21     data = u.read()
22
23 def up_input(channel):
24     GPIO.remove_event_detect(port)
25     GPIO.add_event_detect(port, GPIO.RISING)
26     GPIO.add_event_callback(port, press_input)
27
28 def press_input(channel):
29     GPIO.remove_event_detect(port)
30     GPIO.add_event_detect(port, GPIO.FALLING)
31     GPIO.add_event_callback(port, up_input)
32     global count
33     count = count + 1
34     print count
35     global last_time
36     last_time = time.time()
37 #Configuro cual va a ser el mecanismo de numeracion de GPIO que voy a utilizar
38 GPIO.setmode(GPIO.BCM)
39 #Configuro
40 GPIO.setup(port, GPIO.IN, pull_up_down=GPIO.PUD_DOWN)
41 GPIO.add_event_detect(port, GPIO.RISING)
42 GPIO.add_event_callback(port, press_input)
43
44 print 'Sensores iniciado en el puerto ' + str(port)
45
46 while True:
47     time.sleep(60)
48     send()

```

Fig 34 – Código fuente de lectura del sensor de efecto Hall

5.4 Topología de red

Luego de definir la arquitectura a utilizar se procedió a escoger cual era el mejor medio de comunicación para el envío de paquetes entre las placas *raspberry* y el servidor, considerando el ambiente industrial en el que se implementa la solución, implementar una red wifi es la opción más económica, pero no era la más confiable ya podía ocurrir una gran pérdida de paquetes debido a la interferencia de los equipos industriales (motores, PLCs, etc), entonces se realizó un cableado estructurado desde el servidor a las *raspeberrys* utilizando diferentes *switch* intermedios.

Cada uno de los dispositivos de la red fue configurado con una IP estática, debido a la cantidad finita de dispositivos que deben tener acceso a ella, (En principio 15 contando los dispositivos RPI y el servidor).

Por seguridad se deshabilitó el servicio de DHCP para evitar que otros dispositivos pudieran acceder a la red, ya que el servidor asigna las direcciones IP utilizando un filtrado por número de MAC Address.

5.6 Capa de middleware

Como mencionamos anteriormente en nuestro caso de estudio contamos con una capa de software denominada middleware la cual fue desarrollada utilizando una API desarrollada utilizando el micro *framework* de PHP Slim [56] para atender y responder tanto a las peticiones de los dispositivos de IoT, como las peticiones realizadas por los procesos de negocio que se ejecutan en los BPMS.

Cada uno de los dispositivos de IoT que participan de la arquitectura de la fábrica, al momento de prender su sistema operativo, envía una petición mediante POST al middleware indicando su dirección IP, canal en el que va a transmitir e identificar.

La dirección IP la envía para que la capa de middleware puede retornar en el caso de ser necesario una respuesta individual al dispositivo, como por ejemplo podría ser que el BPMS solicite en alguna de sus tareas que todos los semáforos físicos de las máquinas de producción estén de color rojo.

Cada dispositivo de IoT puede registrarse a uno o más canales, cada canal representa el tipo de sensor se comunica por dicho canal, por ejemplo, un dispositivo que cuenta con un sensor de temperatura, para poder enviar la información recolectada se registrará en el canal "Medir Temperatura".

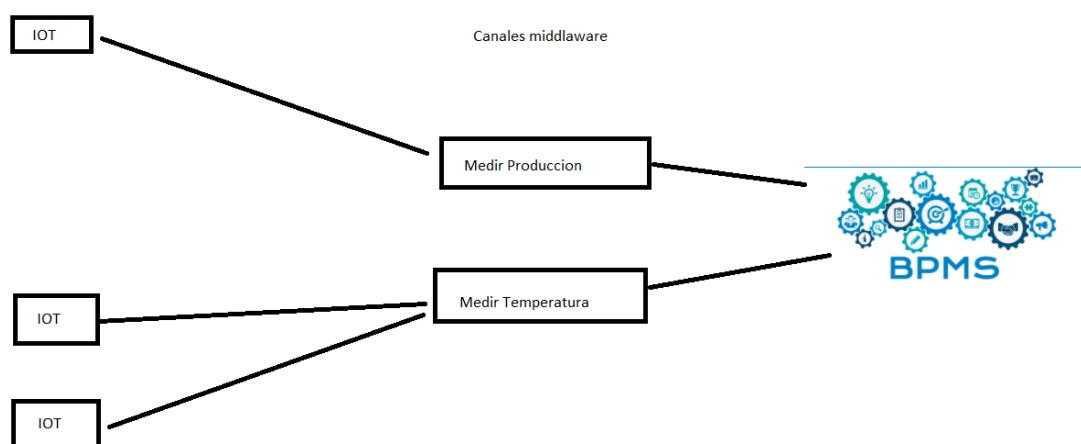


Fig 35 – Registro de tópicos

Los dispositivos de IoT envían sus datos en lapsos de 2 minutos a la capa de middleware, este tiempo que en algunas soluciones de IoT de tiempo real, puede parecer excesivo en mi solución no lo es, ya que de esta forma logré

evitar congestionar la red de paquetes continuamente, ya que las máquinas de producción de la fábrica en promedio realizan dos lecturas por segundo.

```
/* Retorna todas las lecturas de una maquina particular para un periodo de tiempo */
$app->get('/lecturas', function() use ($app) {
    $id = $app->request->get('id');
    $desde = $app->request->get('desde');
    $hasta = $app->request->get('hasta');
    $consulta = "SELECT * FROM lectura where `maquina` = :id and `fecha` BETWEEN :desde AND :hasta";
    $stmt = $mbd->prepare($consulta);
    $stmt->bindParam(':id', $id);
    $stmt->bindParam(':desde', $desde);
    $stmt->bindParam(':hasta', $hasta);
    $stmt->execute();
    $userData = $stmt->fetchAll(PDO::FETCH_ASSOC);
    $data['lecturas'] = $userData;
    echoResponse(200, $data);
});
```

Fig 36 – Ejemplo de método de la API REST de la capa de middleware

5.7 Factores que podrían perjudicar la eficiencia del sistema

Tráfico en la red

Al contar con 14 máquinas industriales al momento de implementar esta solución si transmitirán inmediatamente al middleware del servidor, tendríamos: (14×2) Lecturas por segundo * 120 (por dos minutos) = 3360 paquetes cada 2 minutos, dentro de la misma red de IoT industrial.

En cambio, con esta solución el número se reduce a (14×2) (cada dos minutos) = 28 paquetes cada 2 minutos, logrando una mejora considerable en la congestión de la red.

Tolerancia a fallos desde la capa de comunicación

Durante 2 minutos los dispositivos almacenan en variables internas de su código el contador de unidades producidas que van leyendo con sus sensores de efecto hall, al cumplirse con un tiempo preestablecido de 2 minutos, el dispositivo intenta enviar mediante por API REST la información al servidor, en caso de encontrarse con una respuesta con código 404, podría tratarse de una caída en la red o del servidor, en dicho caso el dispositivo de RPI almacena en una base de datos local la información, para ser enviada posteriormente y vuelve a iniciar su proceso de lectura de 2 minutos.

Tolerancia a fallos desde la capa de middleware

Continuamente el middleware se encuentra en la tarea de controlar que cada dispositivo registrado siga transmitiendo información, el middleware considera que un dispositivo se encuentra apagado o fuera de línea si en 3 minutos no recibió ninguna lectura de él.

Se considera que un canal está desconectado cuando los dispositivos que se encontraban registrados en él, se consideran todos apagados, en caso de que el BPMS solicite realizar alguna acción particular sobre dicho canal, recibirá la respuesta de “Canal desconectado momentáneamente” para que pueda tratar dicha situación como una excepción desde el lado de la capa de aplicación.

Capítulo 6. Conclusiones y Trabajos Futuros

Es necesario pensar soluciones que permitan la innovación tecnológica dentro de la industria a un bajo costo, esta innovación es indispensable para lograr que la industria nacional pueda ser competitiva con la de otros países más evolucionados tecnológicamente hablando.

Las empresas tienen que considerar a la tecnología de IoT como un gran aliado estratégico para poder fortalecer su crecimiento.

El futuro de IoT dependerá de cómo sus componentes puedan interactuar con los procesos de negocio o con las soluciones de IT que se ejecuten a su alrededor. Una costosa red de sensores carece de utilidad si no se construyen soluciones de software que consuman las señales. Si ese software está conformado por componentes orquestados como procesos de negocio, es natural pensar que los elementos de IoT deban ser tenidos en cuenta desde la fase de modelado de dichos procesos de negocio más adelante.

La solución planteada en cada una de las etapas del ciclo de desarrollo, durante el caso de estudio, fue pensada específicamente para el destinatario de la solución, seguramente si el cliente sería un productor agropecuario, la infraestructura de red en lugar de ser cableada y utilizar el protocolo REST, debería ser por frecuencia de radio y utilizando un protocolo más adecuado para la comunicación, una situación similar ocurre con las placas *raspeberry* utilizadas, en una implementación que no se necesite esa mucha capacidad de procesamiento en los dispositivos finales, o se necesite una gran cantidad de ellos, se puede optar por modelos de placas más económicos.

Lo importante es definir que sin importar los cambios que puedan ocurrir en la red de IoT los mismos seguirán siendo transparentes para el BPMS, ya que él se seguirá comunicando de la misma manera con la capa de middleware planteada.

6.1 - Trabajos futuros

Un posible trabajo futuro sería extender la implementación sobre otros protocolos diferentes del utilizado (API REST), por ejemplo, MQTT.

Asimismo, capa de middleware puede extenderse. Durante el presente trabajo se presenta una versión inicial de la misma, pero el objetivo es que pueda mejorarse su nivel de seguridad, funciones y confiabilidad. Esta mejora se ve favorecida por la arquitectura de la solución por capas de abstracción.

Otro aspecto para continuar investigando es el de la utilización de diferentes tipos de sensores, para medir otros indicadores de las máquinas industriales, como por ejemplo la velocidad de rotación de los rodillos que trasladan el material por las diferentes máquinas, o sensores de peso, para medir la cantidad de material procesado.

Finalmente, también otra línea de trabajo puede incluir la utilización de herramienta BPMS diferente a Bonita Soft como por ejemplo Bizagi, analizando

y comparando el tiempo de respuesta de ambas herramientas al comunicarse con la capa de middleware. Asimismo, también es importante abordar una solución mediante componentes de Docker para facilitar su despliegue.

6.2 - Contribuciones

Durante el transcurso del desarrollo de la tesina se presentaron, y se expusieron a evaluación dos artículos relacionados al tema de la tesina de grado. Los cuales han sido aceptados y luego expuestos por quien presenta esta tesina.:

1. Candia, D., Rodríguez, A. S., Bazán, P., Ambrosi, V. M., & Díaz, F. J. (2018). Integración de IoT en el modelado de procesos de negocio. In XX Workshop de Investigadores en Ciencias de la Computación (WICC 2018, Universidad Nacional del Nordeste).

Resumen

En la era de Internet de las Cosas (IoT – Internet of Things) se calcula que para el año 2025 existirán más 26 billones de dispositivos conectados incluyendo casas, teléfonos, autos y fábricas. En este contexto es imposible imaginar que los procesos de las organizaciones no deban adaptarse a la nueva realidad, en la cual la mayoría de las actividades serán realizadas por autómatas que deberán tomar decisiones en base a la información obtenida por su red de sensores.

Así como es necesario integrar IoT con soluciones IT tradicionales donde se procesan datos transaccionales, también las soluciones orientadas a procesos de negocio (PN– Procesos de Negocio) y su naturaleza estática, se ven desafiadas por el mundo de IoT donde se hace presente el dinamismo de las restricciones y la heterogeneidad de las entradas.

La línea de investigación propone presentar las debilidades y fortalezas de la integración de IoT en una solución de BPM, y presentarlas en un caso de estudio. Teniendo en cuenta que la aplicación de BPM conlleva un ciclo de vida donde sus etapas integran un ciclo de mejora continua para modelado, despliegue, ejecución y monitoreo de los procesos de negocio, en este trabajo se definirá la integración de BPM e IoT para la fase de modelado, quedando como trabajo futuro, abarcar todas las etapas del ciclo de vida de los procesos de negocio.

El caso de estudio a modelar estará basado en un proceso de negocio que incluya componentes de IoT, de la Planta Piloto Experimental de Residuos Electrónicos desarrollada en conjunto entre la UNLP, el programa E-Basura y la International Telecommunications Union (ITU).

Relación con la presente tesina de grado

En este trabajo se inició el análisis sobre la integración de IoT y BPM durante su etapa de modelado.

2. Candia, L. D., Rodríguez, A. S., Castro, N., Bazán, P., Ambrosi, V. M., & Díaz, F. J. (2018). Mejoras en maquinaria industrial con IoT: hacia la industria 4.0. In XXIV Congreso Argentino de Ciencias de la Computación (La Plata, 2018).

Resumen

Se calcula que en el año 2025 existirán más de 26 billones de dispositivos conectados incluyendo casas, teléfonos, autos y fábricas. Esta era de hiperconectividad se conoce como “La era de Internet de las Cosas (IoT – Internet of Things)”, la cual impone grandes desafíos y conduce a las industrias a adaptar sus procesos de producción a la nueva realidad. La aplicación de este nuevo concepto en la industria es conocida como Industria 4.0 o la nueva revolución industrial. La misma consiste en la digitalización de los procesos industriales por medio de la interacción de la inteligencia artificial con las máquinas y la optimización de recursos enfocada en la creación de efectivas metodologías comerciales. Esto implica cambios orientados a infraestructuras inteligentes y a la digitalización de metodologías. Este trabajo propone la actualización de una maquinaria industrial mediante la incorporación de sensores que alimentan sistemas informáticos de control, mediante una arquitectura por capas.

Relación con la presente tesina de grado

Este trabajo incluyó las primeras pruebas de concepto para integrar la red de sensores en la maquinaria industrial.

Referencias

- 1- Okano, M. T. (2017). IOT and Industry 4.0: The Industrial New Revolution. In International Conference on Management and Information Systems September (Vol. 25, p. 26)
- 2- Ashton, K. (2009). That 'internet of things' thing. RFID journal, 22(7), 97-114.
- 3- Candia, L. D., Rodríguez, A. S., Castro, N., Bazán, P., Ambrosi, V. M., & Díaz, F. J. (2018). Mejoras en maquinaria industrial con IoT: hacia la industria 4.0. In XXIV Congreso Argentino de Ciencias de la Computación (La Plata, 2018).
- 4- https://image-src.bcg.com/Images/Acelerando-el-Desarrollo-de-Industria-40-en-Argentina_tcm62-184622.pdf recuperado el 19/04/2021. Recuperado el 19/04/2021
- 5- Weske, M. (2012). Business process management architectures. In Business Process Management (pp. 333-371). Springer, Berlin, Heidelberg.
- 6- <https://www.omg.org/spec/BPMN/2.0/>
- 7- Karabogossian, L. D., Bazán, P., & Martínez Garro, J. N. (2014, October). Ejecución y monitoreo de procesos de negocios distribuidos entre diferentes motores de Bonita OS. In XVI Workshop de Investigadores en Ciencias de la Computación.
- 8- <https://es.bonitasoft.com/> Recuperado el 04/05/2021
- 9- Ines Robles, Alvaro Retana. IoT Roadmap, Iacnic25 (2/6 de mayo- La Habana, Cuba).
- 10- Pertini, B. (2017). Análisis y mejoras de seguridad a una aplicación prototipo en IoT (Doctoral dissertation, Universidad Nacional de La Plata).
- 11- Antonio Liñán Colina, Alvaro Vives, Marco Zennaro, Antoine Bagula, Ermanno Pietrosevoli. Internet of Things IN 5 DAYS.
- 12- <https://azure.microsoft.com/es-es/overview/internet-of-things-iot/iot-technology-protocols/> Recuperado el 04/05/2021.
- 13- J. Gubbi, R. Buyya, S. Marusic, M. Palaniswami, Internet of Things (IoT): A vision, architectural elements, and future directions, Future Generation Computer Systems 29 (7) (2013) 1645 – 1660.
- 14- L. Atzori, A. Iera, G. Morabito, The internet of things: A survey, Computer networks 54 (15) (2010) 2787–2805.
- 15- D. Evans, The internet of things: How the next evolution of the internet is changing everything, CISCO white paper 1 (2011).

- 16-P. Middleton, P. Kjeldsen, J. Tully, Forecast: The internet of things, worldwide, 2013, Tech. rep., Gartner (November 2013).
- 17-J. Manyika, M. Chui, J. Bughin, R. Dobbs, P. Bisson, A. Marrs, Disruptive technologies: Advances that will transform life, business, and the global economy, Tech. rep., Chrysalix (May 2013).
- 18-M. Dumas, M. La Rosa, J. Mendling, H. A. Reijers, Fundamentals of business process management, Springer-Verlag Berlin Heidelberg, Berlin, Germany, 2013.
- 19-D. Jordan, J. Evdemon, A. Alves, A. Arkin, S. Askary, C. Barreto, B. Bloch, F. Curbera, M. Ford, Y. Goland, et al., Web services business process execution language version 2.0, OASIS standard 11 (120) (2007) 5.
- 20-Workflow Management Coalition, Process Definition Interface – XML Process Definition Language (August 2012).
- 21-W. M. P. van der Aalst, A. H. ter Hofstede, M. Weske, Business process management: A survey, in: W. van der Aalst, M. Weske (Eds.), Business Process Management, Vol. 2678 of Lecture Notes in Computer Science, Springer, Berlin Heidelberg, 2003, pp. 1–12.
- 22-N. Glombitza, S. Ebers, D. Pfisterer, S. Fischer, Using bpm to realize business processes for an internet of things, in: H. Frey, X. Li, S. Ruehrup (Eds.), Ad-hoc, Mobile, and Wireless Networks, Vol. 6811 of Lecture Notes in Computer Science, Springer, Berlin Heidelberg, 2011, pp. 294–307.
- 23-S. Schulte, P. Hoenisch, C. Hochreiner, S. Dustdar, M. Klusch, D. Schuller, Towards process support for cloud manufacturing, in: Proceedings of the 18th. IEEE International Enterprise Distributed Object Computing Conference (EDOC '14), IEEE, Ulm, 2014, pp. 142–149.
- 24-S. Tranquillini, P. Spieß, F. Daniel, S. Karnouskos, F. Casati, N. Oertel, L. Mottola, F. J. Oppermann, G. P. Picco, K. Römer, et al., Process-based design and integration of wireless sensor network applications, in: Business Process Management, Springer, Berlin Heidelberg, 2012, pp. 134–149.
- 25-M. Quwaider, Y. Jararweh, A cloud supported model for efficient community health awareness, Pervasive and Mobile Computing 28 (2016) 35–50.
- 26-M. Conti, S. K. Das, C. Bisdikian, M. Kumar, L. M. Ni, A. Passarella, G. Roussos, G. Trster, G. Tsudik, F. Zambonelli, Looking ahead in pervasive computing: Challenges and opportunities in the era of cyberphysical convergence, Pervasive and Mobile Computing 8 (1) (2012) 2 – 21.
- 27-W. M. P. van der Aalst, Business process management: A comprehensive survey, ISRN Software Engineering 2013 (2013) 37.

- 28-W. M. P. van der Aalst, A. H. ter Hofstede, M. Weske, Business process management: A survey, in: W. van der Aalst, M. Weske (Eds.), Business Process Management, Vol. 2678 of Lecture Notes in Computer Science, Springer, Berlin Heidelberg, 2003, pp. 1–12.
- 29-M. Weske, W. M. van der Aalst, H. Verbeek, Advances in business process management, *Data & Knowledge Engineering* 50 (1) (2004) 1–8
- 30-Bazán, P. (2009). Un modelo de integrabilidad con SOA y BPM (Doctoral dissertation, Cap. 4, Facultad de Informática.
- 31-K. Dar, A. Taherkordi, R. Rouvoy, F. Eliassen, Adaptable service composition for very-large-scale internet of things systems, in: Proceedings of the 8th Middleware Doctoral Symposium, MDS '11, ACM, New York, NY, USA, 2011, pp. 2:1–2:6.
- 32-K. Dar, A. Taherkordi, H. Baraki, F. Eliassen, K. Geihs, A resource oriented integration architecture for the internet of things: A business process perspective, *Pervasive and Mobile Computing* 20 (2015) 145 – 159.
- 33-T. Peng, M. Ronchetti, J. Stevovic, A. Chiasera, G. Armellin, Business process assignment and execution from cloud to mobile, in: Proceedings of the 2014 Business Process Management Workshops, Springer International Publishing, Switzerland, 2014, pp. 264–276.
- 34- S. Meyer, A. Ruppen, L. Hilty, The things of the internet of things in bpmn, in: Proceedings of the 2015 Advanced Information Systems Engineering Workshops, Springer International Publishing, Switzerland, 2015, pp. 285–297.
- 35-S. Appel, P. Kleber, S. Frischbier, T. Freudenreich, A. Buchmann, Modeling and execution of event stream processing in business processes, *Information Systems* 46 (2014) 140–156.
- 36-S. Meyer, A. Ruppen, C. Magerkurth, Advanced Information Systems Engineering: 25th International Conference, CAiSE 2013, Valencia, Spain, June 17-21, 2013. Proceedings, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, Ch. Internet of Things-Aware Process Modeling: Integrating IoT Devices as Business Process Resources, pp. 84–98.
- 37-C. T. Sungur, P. Spiess, N. Oertel, O. Kopp, Extending bpmn for wireless sensor networks, in: Proceedings of the 15th. IEEE Conference on Business Informatics (CBI '13), IEEE, Vienna, 2013, pp. 109–116.
- 38-A. Yousfi, A. de Freitas, A. K. Dey, R. Saidi, The use of ubiquitous computing for business process improvement, *IEEE Transactions on Services Computing* 9 (4) (2015) 621–632.
- 39-A. Yousfi, C. Bauer, R. Saidi, A. K. Dey, ubpmn: A bpmn extension for modeling ubiquitous business processes, *Information and Software Technology* 74 (2016) 55– 68.
- 40-S. Appel, P. Kleber, S. Frischbier, T. Freudenreich, A. Buchmann, Modeling and execution of event stream processing in business processes, *Information Systems* 46 (2014) 140–156.
- 41-Candia, D., Rodríguez, A. S., Bazán, P., Ambrosi, V. M., & Díaz, F. J. (2018). Integración de IoT en el modelado de procesos de negocio. In XX Workshop de

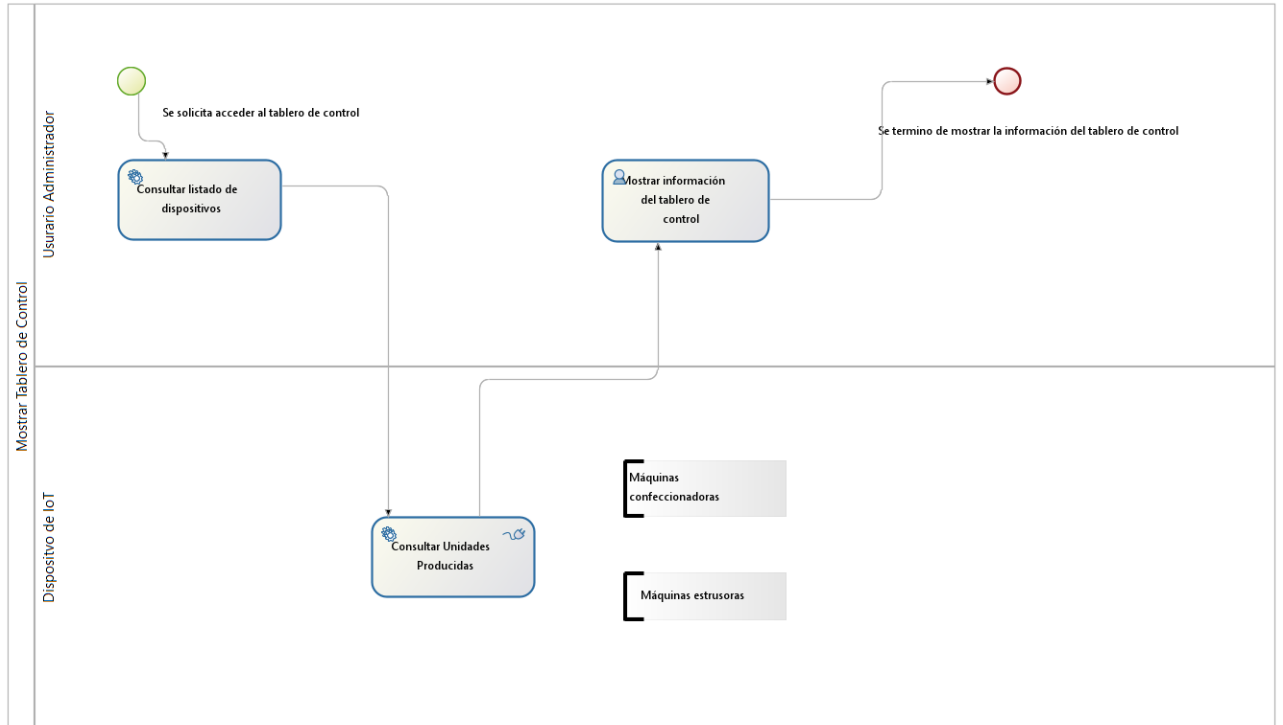
- Investigadores en Ciencias de la Computación (WICC 2018, Universidad Nacional del Nordeste).
- 42- <http://activiti.org>. Recuperado el 04/05/2021
 - 43- <https://camunda.com> Recuperado el 04/05/2021
 - 44- <http://ode.apache.org> Recuperado el 04/05/2021
 - 45- <https://www.ibm.com/docs/es/bpm/8.6.0?topic=types-bpel-process> Recuperado el 04/05/2021.
 - 46- P. J. I. Kaneshiro, P. D. Haghghi, S. Ling, Situation-aware adaptation to optimise energy consumption in intelligent buildings using coloured petri nets, in: Proceedings of the IEEE 9th Conference on Industrial Electronics and Applications (ICIEA), IEEE, Hangzhou, 2014, pp. 231–236.
 - 47- R. A. Thacker, K. R. Jones, C. J. Myers, H. Zheng, Automatic abstraction for verification of cyber-physical systems, in: Proceedings of the 1st. ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS '10), ACM, New York, NY, USA, 2010, pp. 12–21.
 - 48- R. Pryss, M. Reichert, A. Bachmeier, J. Albach, BPM to Go: Supporting Business Processes in a Mobile and Sensing World, in: L. Fischer (Ed.), BPM Everywhere: Internet of Things, Process of Everything, Future Strategies Inc., FL, USA, 2015, pp. 167–182.
 - 49- T. Peng, M. Ronchetti, J. Stevovic, A. Chiasera, G. Armellin, Business process assignment and execution from cloud to mobile, in: Proceedings of the 2014 Business Process Management Workshops, Springer International Publishing, Switzerland, 2014, pp. 264–276.
 - 50- C. Chang, S. Ling, Towards a context-aware solution for device failures in service-oriented workflow, in: Proceedings of the 10th International 43 Conference on Information Integration and Web-based Applications & Services, ACM, New York, NY, USA, 2008, pp. 77–83.
 - 51- Z. Wu, T. It"al"a, T. Tang, C. Zhang, Y. Ji, M. H"am"al"ainen, Y. Liu, Gateway as a service: A cloud computing framework for web of things, in: Proceedings of the 19th International Conference on Telecommunications (ICT '12), IEEE, Jounieh, 2012, pp. 1–6.
 - 52- <https://www.raspberrypi.org/>.
 - 53- <https://www.raspberrypi.org/software/operating-systems/>
 - 54- <https://www.ucm.es/data/cont/docs/76-2015-03-19-Efecto-Hall-Final.pdf>
 - 55- <https://www.python.org/>
 - 56- <https://www.slimframework.com/>

Anexo 1- Reportes del sistema

En este Anexo, se presentan las imágenes de los reportes del sistema de gestión desarrollado con Bonita Soft con una secuencia que muestra su diagrama BPD correspondiente, y sus conectores para comunicar con la capa de middleware que administra la conexión con los dispositivos de IoT ubicados en las máquinas.

Indice de figuras del Anexo

Tablero de control del sistema de gestión	76
Mapeo entre variables de proceso y métodos de la API REST	76
Ejemplo de Script Bonita BPMS.....	77
Ejemplo de editor de formulario Bonita Soft	77
Tablero de control de sistema de gestión.....	78
Producción de una máquina para un día determinado	78
Grafico de comparación de producción de las máquinas	79
Grafico productividad semanal.....	79
Estadísticas Semanales.....	80
Interface Operario de planta	80
Menú de opciones de Operario de planta	81



Tablero de control del sistema de gestión

GET (1.1.0)

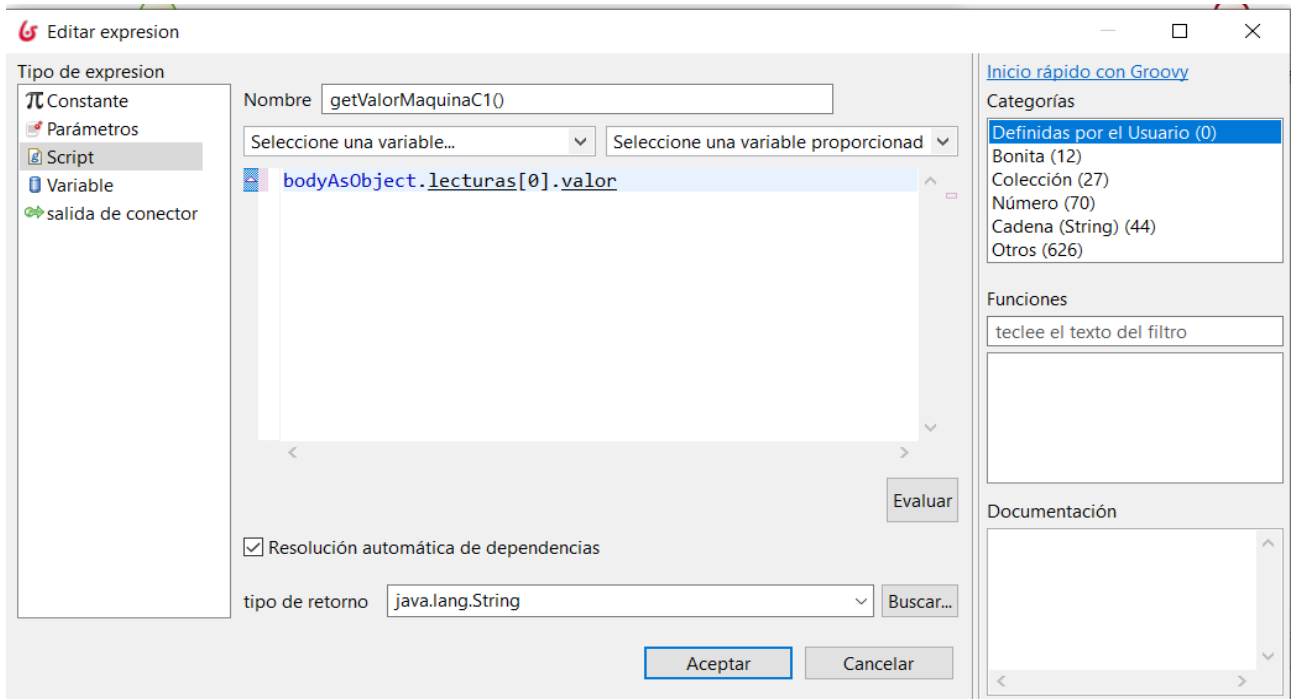
Mapea las salidas de este conector a las variables del proceso.

Obten las salidas del conector y almacenalas en las variables de proceso o de negocio

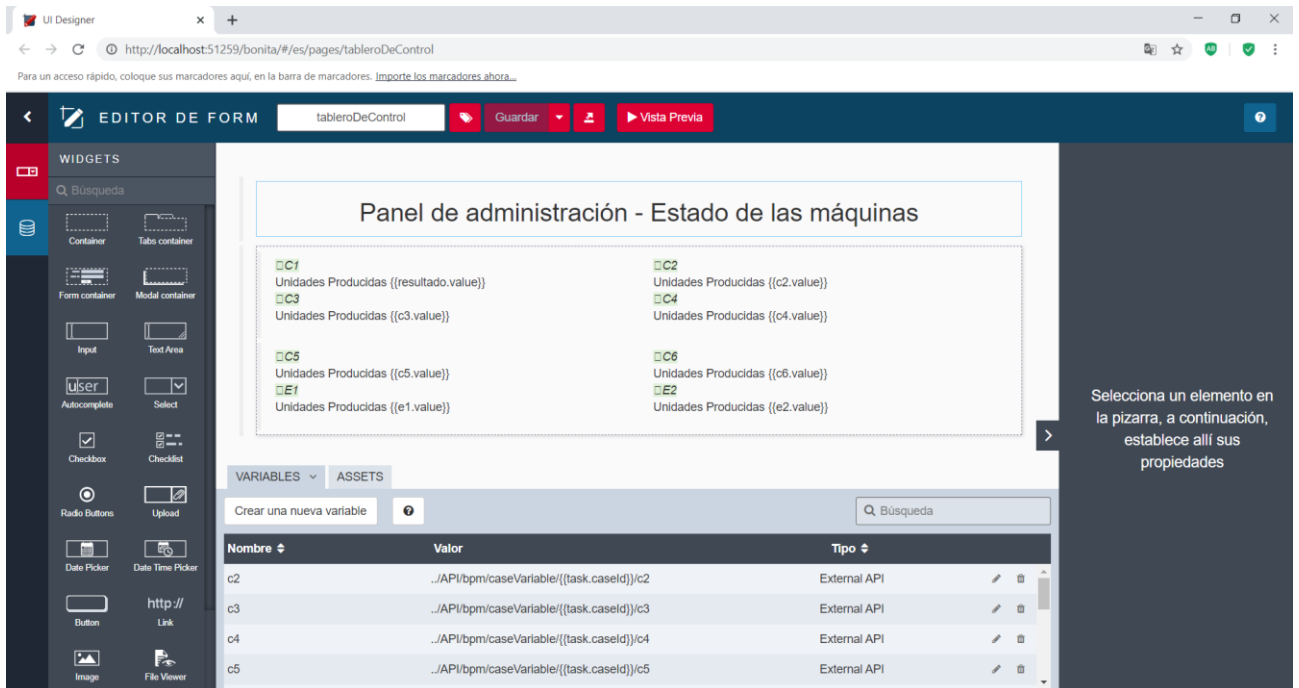
Si la respuesta Content-Type es compatible con Json, usa la salida 'bodyAsObject' para acceder rápidamente a los datos en un script (ejemplo: bodyAsObject.user.name o, en el caso de un Array, bodyAsObject[0].userName). Si no es compatible, usa la salida 'bodyAsString'.

c1	Toma valor de	getValorMaquinaC1()
c2	Toma valor de	getValorMaquinaC2()
c3	Toma valor de	getValorMaquinaC3()
c4	Toma valor de	getValorMaquinaC4()
c5	Toma valor de	getValorMaquinaC5()
c6	Toma valor de	getValorMaquinaC6()
e1	Toma valor de	getValorMaquinaE1()
e2	Toma valor de	getValorMaquinaE2()

Mapeo entre variables de proceso y métodos de la API REST



Ejemplo de Script Bonita BPMS



Ejemplo de editor de formulario Bonita Soft

Panel de administración - Estado de las máquinas

C1	Unidades Producidas 27	C2	Unidades Producidas 42
C3	Unidades Producidas 200	C4	Unidades Producidas 200
C5	Unidades Producidas 0	C6	Unidades Producidas 0
E1	Unidades Producidas 157	E2	Unidades Producidas 0

Tablero de control del sistema de gestión

Panel de administración - Productividad diaria

Maquina C1

Periodo	Horas Apagada	Horas Prendida	Horas Parada
Día completo	10 hs	12 hs	2 hs
Turno Mañana	0 hs	7 hs	1 hs
Turno tarde	2 hs	5 hs	1 hs
Turno Noche	8 hs	0 hs	0 hs

Producción de una máquina para un día determinado

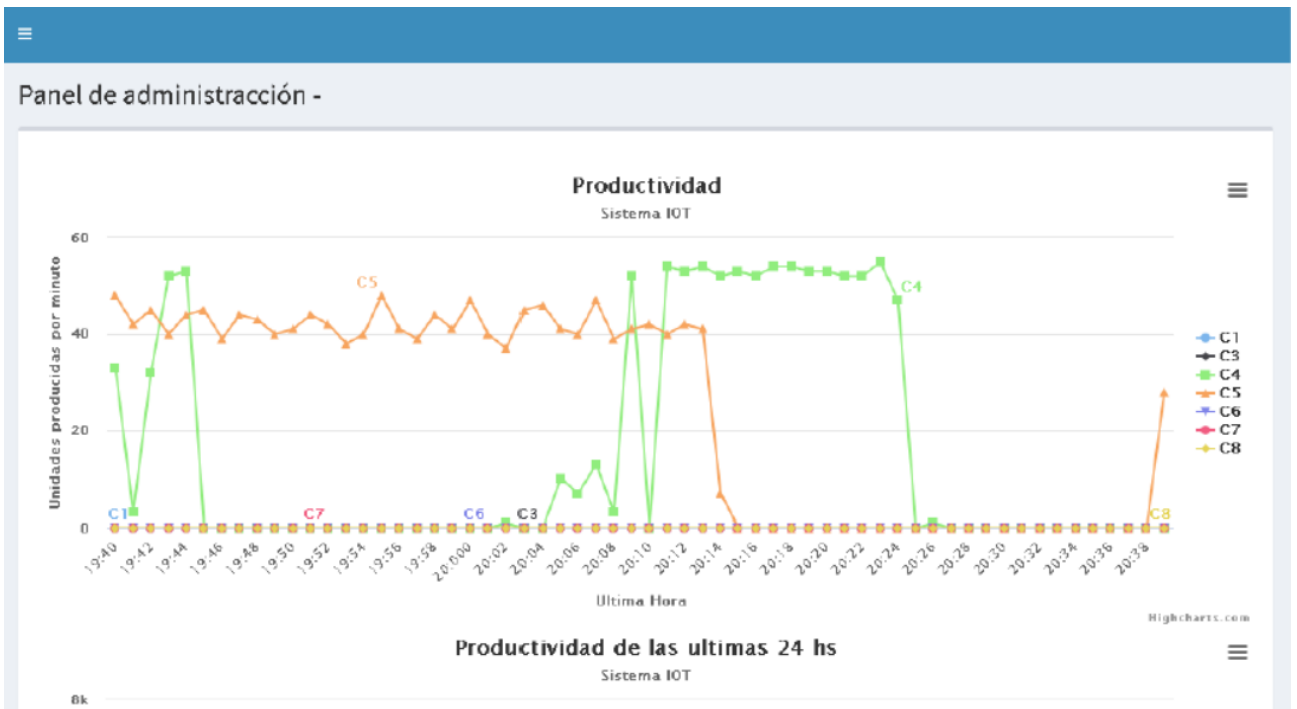


Gráfico de comparación de producción de las máquinas

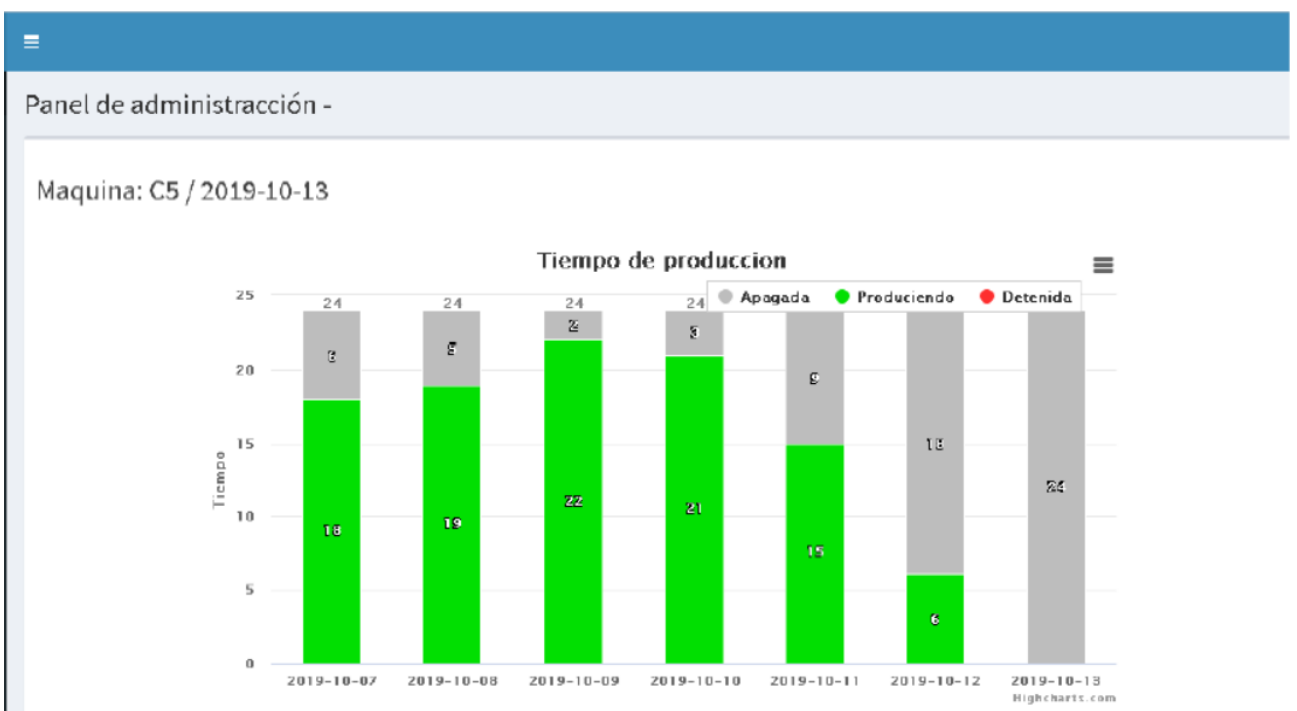


Gráfico productividad semanal

Maquina	Estado	Sem 1	Sem 2	Sem 3	Sem 4	Sem 5	Sem 6	Sem 7	Sem 8	Sem 9	Sem 10	Sem 11	Sem 12	Sem 13
---	OPERATIVA	00:00:00	104:12:00	3133:00	00:00:00	00:13:00	00:00:00	3:51:00	33:18:00	25:30:00	24:39:00	43:09:00	28:52:00	28:31:00
C5	HORAS PARADA	00:00:00	4:46:00	00:25:00	00:00:00	00:30:00	00:00:00	00:00:00	2:41:00	1:55:00	1:19:00	1:50:00	1:56:00	00:38:00
---	APAGADA	--	--	--	--	--	--	--	--	--	--	--	--	--
---	Unidades	0	294326	7767	0	1	0	8409	79379	75960	75131	230910	132625	103151

Volver

Estadísticas semanales

Sistema IoT

NUMERO DE OT

3

Progreso 0.00 %

Porcentaje de la tarea ejecutada



DETENER PROCESO

Debera seleccionar el motivo

Interface Operario de planta

Sistema IoT

	NUMERO DE OT 3 Progreso 0,00 %
	DETENER POR MANTENIMIENTO PREVENTIVO Se detendra la lectura de los sensores
	DETENER POR MANTENIMIENTO CORRECTIVO Se detendra la lectura de los sensores
	DETENER POR DESCANSO DEL OPERADOR Se detendra la lectura de los sensores
	VOLVER A ACTIVAR LA TAREA Se iniciara la lectura de los sensores

Menú de opciones de Operario de planta