

Centro de Investigación y de Estudios Avanzados del I.P.N.  
Unidad Tamaulipas  
**Protocolo de tesis**

Tesista: José Carlos Morín García  
Director de tesis: Dr. José Luis González Campeán  
Co-Director de tesis: Dr. Iván López Arévalo

19 de agosto de 2021

**Resumen**

Los procesos de toma de decisiones se basan en información obtenida a partir de cúmulos de datos. Recientemente, estos procesos comienzan a incorporar herramientas de fusión de datos (*FD*) para confrontar información proveniente de múltiples fuentes, con el fin de tener un panorama amplio del problema antes de tomar decisiones enfocadas a resolver un problema dado. Sin embargo, crear un servicio *FD* no es una tarea trivial porque estos servicios generalmente están asociados a una dependencia con el proveedor de servicios *FD* (dependencia *usuario-proveedor*), los cuales a su vez dependen del método de orquestación de datos (dependencia *fusión-orquestación*).

En este proyecto de tesis se propone un método de orquestación de datos, independiente de la infraestructura y determinado por esquemas basados en variables espacio-temporales para crear servicios dinámicos de *FD*. Este método incluirá un esquema de despliegue y acoplamiento de servicios agnósticos de la infraestructura de cómputo en la nube para crear servicios de *FD* que eviten la dependencia *Usuario-proveedor*. Un modelo de construcción basado en cubos espacio-temporales permitirá desacoplar la fusión de la orquestación, lo cual habilitará a los usuarios a crear servicios dinámicos de *FD*, lo cual evitará la dependencia *Fusión-Orquestación*. Se definirá un esquema declarativo para definir el modelo de construcción que permita a los usuarios crear servicios dinámicos de *FD* a partir de parámetros espacio-temporales. Se desarrollará un prototipo basado en este modelo para crear servicios dinámicos de *FD*.

**Palabras clave:** *Big Data Analytics*, Fusión de Datos, Orquestación de datos, *vendor lock-in*.

## 1. Datos generales

### 1.1. Título de proyecto

Método de orquestación para servicios de fusión de datos definidos por variables espacio-temporales.

### 1.2. Datos del alumno

Nombre: José Carlos Morín García.  
Matrícula: 209860006.  
Dirección: Calle Heroes No. 1513 Fracc. Mi Ranchito  
Cd. Victoria, Tamaulipas, México.  
Teléfono (personal): +52(834) 3116544.  
Dirección electrónica: jose.morin@cinvestav.mx

### 1.3. Institución

Nombre: CINVESTAV-IPN  
Departamento: Unidad Tamaulipas.  
Dirección: Km. 5.5 carretera Cd. Victoria - Soto la Marina.  
Parque Científico y Tecnológico TECNOTAM.  
Ciudad Victoria, Tamaulipas, C.P. 87130.  
Teléfono: +52 (834) 107 0220.

### 1.4. Beca de tesis

Institución otorgante: Consejo Nacional de Ciencia y Tecnología.  
Tipo de beca: Beca de estudios de maestría.  
Vigencia: 01 de septiembre de 2020 hasta el 31 de agosto de 2022.

### 1.5. Datos del asesor

Nombre: Dr. José Luis González Campeán.  
Dirección: Km. 5.5 carretera Cd. Victoria - Soto la Marina.  
Parque Científico y Tecnológico TECNOTAM.  
Ciudad Victoria, Tamaulipas, C.P. 87130.  
Teléfono (oficina): +52 (834) 107 0220 - Extensión . 1138.  
Institución: CINVESTAV-IPN.  
Departamento adscripción: Unidad Tamaulipas.  
Grado académico: Doctorado

## 1.6. Datos del coasesor<sup>1</sup>

Nombre:	Dr. Iván López Arévalo
Dirección:	Km. 5.5 carretera Cd. Victoria - Soto la Marina. Parque Científico y Tecnológico TECNOTAM. Ciudad Victoria, Tamaulipas, C.P. 87130.
Teléfono (oficina):	834 107 0258
Institución:	CINVESTAV-IPN
Departamento adscripción:	Unidad Tamaulipas
Grado académico:	Doctorado

## 2. Descripción del proyecto

Este trabajo de tesis se contextualiza en el área de Sistemas Distribuidos, dentro de Cómputo en la Nube, específicamente en el proceso de *orquestración para servicios de fusión de datos*, abordando el problema de *vendor lock-in*.

Con esto se espera crear un método de orquestración para fusión de datos que incluya, en una primera fase, un modelo de manejo declarativo para desplegar servicios de fusión de datos sobre múltiples infraestructuras y plataformas. En una segunda fase se espera obtener un modelo de fusión de datos basado en parámetros espacio-temporales para desacoplar los procesos de fusión de la orquestración que se realizará mediante el modelo de la primera fase.

### 2.1. Antecedentes

En esta sección se presenta, brevemente, una introducción a los conceptos más relevantes para describir el objeto de estudio del presente trabajo de tesis: *Big Data (BD)*, *Big Data Analytics (BDA)*, fusión de datos y orquestración.

#### 2.1.1. Manejo de grandes volúmenes de datos

En la actualidad se están generando, registrando, analizando, compartiendo y consumiendo cantidades ingentes de datos e información [1] [2] [3], se estima que diariamente en el mundo se generan más de 2.5 quintillones de bytes en el mundo<sup>1</sup>. Un gran porcentaje de los cúmulos de datos (volúmenes de datos que crecen a tasas constantes y que acumulan el volumen de crecimiento al volumen de datos original) [4]<sup>2</sup> actualmente se encuentran disponibles en la nube [5] y son susceptibles de ser usados para extraer información, la cual resulta crucial para procesos de toma de decisiones, tales como la observación de la tierra/espacio [6], manejo de territorio [7], estudio/diagnóstico de enfermedades [8], manejo de efectos contaminantes/demografía [9], por nombrar algunos.

Para gestionar tal cantidad de datos con el fin de analizarlos para extraer información, se aplican técnicas de procesamiento de grandes volúmenes de datos (*Big Data*). Estas técnicas incluyen conjuntos de herramientas, algoritmos de analítica y procedimientos de manejo de información para

---

<sup>1</sup>Cifras obtenidas por IBM - <https://developer.ibm.com/es/articles/que-es-big-data/>

<sup>2</sup>Estudios recientes indican que el 40% de los datos son adecuados para producir información mediante procesos de analítica de datos

organizar la creación, manipulación y tratamiento de cúmulos de conjuntos de datos [10].

El modelo de procesamiento tradicional utilizado para integrar las herramientas consideradas para el procesamiento de datos es el llamado *Extracción-Transformación-Carga* o *ETL* [11] por sus siglas en inglés. La Figura 1 muestra un ejemplo de un procesamiento de grandes volúmenes de datos modelado como un esquema *ETL*. De esta forma, los datos son *Extraídos* de la nube, *Transformados* por un servicio de Big Data, cuyo resultado es entregado (*L*) a un proceso de toma de decisiones [12].

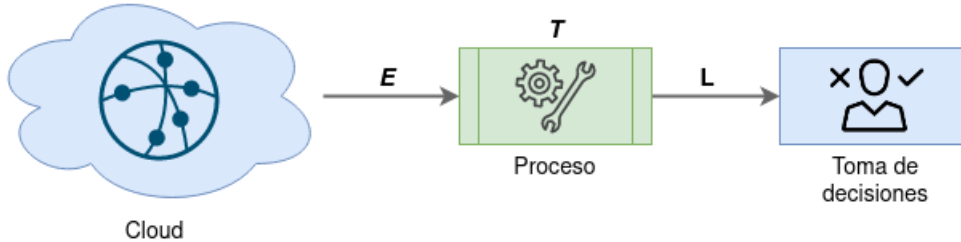


Figura 1: *ETL* en Big Data

Los procesos de *transformación* básicamente son algoritmos de preprocesamiento y procesamiento de datos. Los algoritmos de preprocesamiento generalmente incluyen herramientas tales como minado de texto, mapeo, preparación de datos (series de tiempo, muestreo, reducción de dimensionalidad o enriquecimiento por interpolación/extrapolación de datos faltantes, etc.) [13]. Los algoritmos de procesamiento consideran servicios tales como:

- *Analítica de grandes volúmenes de datos (BDA)*, los cuales están basados en modelos estadísticos/probabilísticos (agrupamiento, clasificación, por nombrar algunos) cuya función es reducir cúmulos de datos a información útil [14].
- *Creación de contenidos (BDC)*, el procesamiento en ráfagas (*streaming processing*) común en escenarios de Internet de las Cosas (IoT) o Industria 4.0 [15] o distribución y entrega de contenidos [16] son ejemplos de este tipo de procesamiento.

### 2.1.2. Manejo de servicios de fusión de datos

Los servicios *BDA*, específicamente en un servicio llamado *fusión de datos*, resulta especialmente desafiante, ya que el modelo tradicional de *ETL* no necesariamente captura su comportamiento [17].

Los servicios de fusión de datos consideran aspectos tales como:

- La integración o concatenación de las variables de múltiples fuentes de datos, es decir, juntar una cantidad de variables de múltiples fuentes y tener un conjunto de datos más completo.
- El enriquecimiento mediante regresiones, interpolaciones, extrapolaciones, o algún método de análisis aplicado a dichas fuentes con nuevos datos/información.
- La reducción de datos mediante intersecciones o uniones de variables. Por ejemplo, eliminar aquellas variables que no se ocuparán o hay pérdida de datos en sus registros.

Dentro de la fusión de datos existen varios modelos y técnicas empleadas, las cuales se pueden dividir de acuerdo con los siguientes criterios:

- Atendiendo la relación entre las entradas y salida propuesta por Durrant-Whyte [18].
- De acuerdo a las entradas/salidas de los tipos de datos y su naturaleza definida por Dasarathy [19].
- Basado en los diferentes niveles de fusión de datos definidos por el Departamento de Defensa de los Estados Unidos [20].
- Dependiendo de la arquitectura en donde se despliegue el servicio de fusión de datos (centralizada, descentralizada y distribuida) [17].

Todos los modelos de fusión de datos incluyen la entrada de  $i$  puntos de datos (pre-establecidos) a un servicio de *BDA* (transformador) y el resultado de la fusión es usado como insumo de otros procesos ya sea de *FD*, *BDA* o de resumen de datos (usado por los procesos de toma de decisiones).

### 2.1.3. Manejo de servicios de fusión de datos en la nube

Los servicios de *fusión de datos* son generalmente ejecutados en la nube. En los últimos años han resultado determinantes para que las organizaciones y/o comunidad científica lleven a cabo estudios de fusión de información en dominios tales como medicina [21] (donde se interconecta información ambiental con historiales clínicos para descubrir prevalencia de agentes de enfermedades), clima [22] (donde se interconecta información de temperaturas con contaminantes para descubrir correlaciones), observación de la tierra [23] (donde se interconecta información satelital con información de monitoreo ambiental), por nombrar algunos.

La Figura 2 se muestran los diferentes componentes requeridos para crear un servicio de fusión de datos en la nube [24] [25] [26]. Como se puede observar, se requieren algoritmos de análisis, manejo de datos, herramientas para acceso a la infraestructura en donde se estará desplegando el servicio y, por último las fuentes de datos que se van a analizar.

Para la unión de los algoritmos de análisis y manejo de datos se genera software en *BDA*, que cuando son desplegados en una infraestructura se obtienen plataformas para *BDA*. Dichas plataformas pueden trabajar con datos que son entregados por diferentes fuentes. Una vez que se dispongan estas plataformas diseñadas y desplegadas, es posible realizar la fusión de datos o cualquier otro tipo de proceso de *BDA*.

El modelo de procesamiento tradicional *ETL* considera una única entrada de extracción para una fase de transformación. Por tanto, este modelo de procesamiento no puede, por definición, producir un modelo de fusión de datos. En sentido, se debe adecuar el modelo *ETL* para poder materializar un modelo de fusión de datos.

La construcción de un servicio *FD* se requiere de un *orquestador de datos* que gestione las múltiples entradas y/o posibles potenciales interconexiones recursivas de las salidas a entradas de otros procesos de fusión [24] [10]. La *orquestación de datos* por tanto se puede entender como un

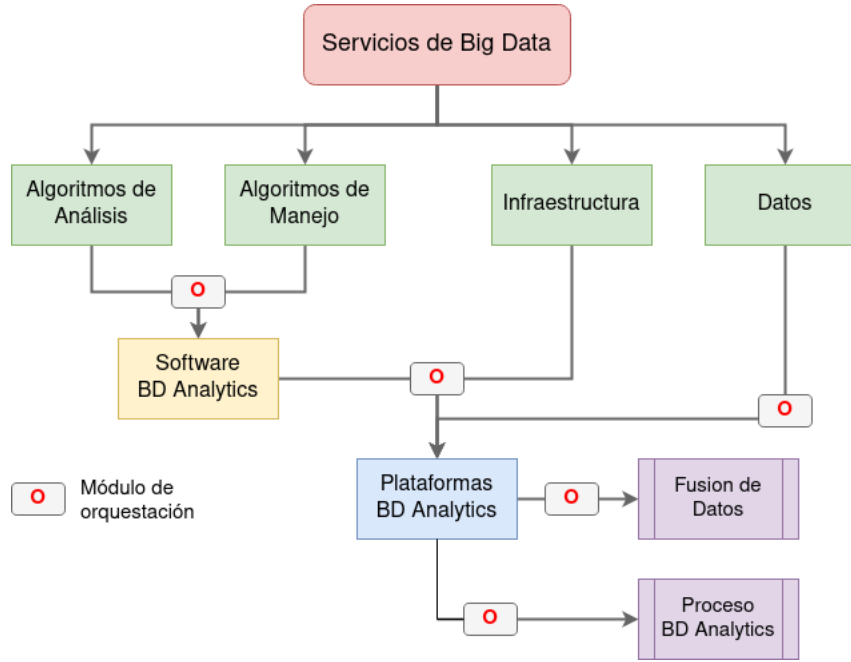


Figura 2: Componentes tradicionales de un *BDA* para crear un servicio de fusión de datos.

proceso automatizado en el que un servicio de software crea las estructuras de software requeridas para materializar un modelo de *FD* [27]. Este servicio crea estructuras de software que se ejecutan en la nube para manejar la extracción desde múltiples fuentes de datos, entregar los datos extraídos a las múltiples fases de transformación y finalmente consolidar los datos en una sola respuesta para los procesos de toma de decisiones [28]. Los procesos de orquestación son generalmente creados *ad hoc* y esta creación depende de las particularidades de los modelos de función (sus múltiples entradas y/o múltiples interconexiones).

## 2.2. Motivación

Los procesos de toma de decisiones se basan en información obtenida a partir de cúmulos de datos [29]. Recientemente, estos procesos comienzan a incorporar herramientas para confrontar información proveniente de múltiples fuentes, con el fin de tener un panorama amplio del problema antes de tomar decisiones enfocadas a resolver un problema dado [30].

Los procesos *FD*, por tanto representan una herramienta clave para los procesos de toma de decisiones. El principal desafío para incorporar un proceso *FD* a un servicio existente (e.g. un proceso de toma de decisiones) así como producir servicios *FD* dinámicos resulta un desafío tanto tecnológico como investigación. Crear sistemas de *FD* dinámicos implica que la *FD* no esté condicionada ni por el proveedor de servicio (implica por tanto evitar dependencia *usuario-proveedor*) ni por el método de orquestación de datos (dependencia *usuario-proveedor* lo cual implica que la orquestación de datos no esté definida por el modelo *FD*). Crear este tipo de servicio no es una tarea trivial porque se deberían afrontar dos retos: el primero crear servicios agnósticos que permitan a los usuarios declarar los lugares desde los cuales se obtendrán las fuentes de datos, los lugares donde se ejecutarán los procesos de *FD* y los lugares en donde se depositará/entregará la informa-

ción resultante. El segundo es crear esquemas de acoplamiento para la orquestación de datos para servicios *FD* sin depender del modelo fusión, lo cuales permitan a los diseñadores crear servicios *FD* basados en diferentes modelos de entradas.

### 3. Planteamiento del problema

Los servicios *FD* y *BDA* son primordialmente ejecutados en la nube, pero adolecen del problema de dependencia que se expresa en dos direcciones: la dependencia *Usuario-Proveedor* y la dependencia *Fusión-Orquestación*:

- *Dependencia usuario-proveedor*: La dependencia del usuario de servicios de *FD* con el prestador de dichos servicios (*vendor lock-in* <sup>3</sup> En este trabajo el servicio es *FD* y el prestador es un proveedor de servicios de *FD* en la nube, por su acepción en inglés) se produce cuando el usuario delega tanto las fuentes de datos, como los servicios de *FD* o *BDA* y orquestación a los proveedores de servicios en la nube (ver flecha azul en Figura 3) [32]. En tal caso, la factibilidad de ejecución de un *FD* depende de los recursos y disponibilidad del proveedor. De la misma forma esta dependencia se extiende a la elección del método *FD*, los servicios *BDA* incluidos en el servicio de *FD* y la recuperación de información. Esta dependencia produce efectos tales como acumulación de datos (que evitan la viabilidad de migración de datos/servicios a otros proveedores o incrementen los costos de los servicios) y no la disponibilidad del *FD* o de los servicios del mismo proveedor de servicios en casos de falla (principalmente apagones [33]).
- *Dependencia fusión-orquestación*: Se produce cuando el usuario solo crea procesos de *FD* dependiendo de los servicios de *BDA* que contiene el catálogo del proveedor y datos disponibles en su nube, los cuales están definidos por un esquema de orquestación propio y predefinido con parámetros estáticos (que el usuario no puede modificar). Estos parámetros están definidos por los proveedores a quienes conviene que estos parámetros sean estáticos para garantizar una calidad de servicio previamente pactada (ver flecha roja de la Figura 3).

Al estudiar el problema del *vendor lock-in* de los servicios *FD* se pueden identificar las siguientes preguntas de investigación:

- ¿Qué método de orquestación permitiría crear servicios de *FD* que mitiguen situaciones de dependencia usuario-proveedor?
- ¿Qué etapas del método de *FD* permitiría disminuir la dependencia de fusión-orquestador?

Atendiendo las preguntas de investigación y tomando en cuenta la literatura en el estado del arte, vemos conveniente establecer las siguientes premisas divididas en dos grupos:

---

<sup>3</sup>Vendor lock-in es la acepción en inglés que describe la dependencia de un usuario de un servicio dado con el prestador de dicho servicio [31].

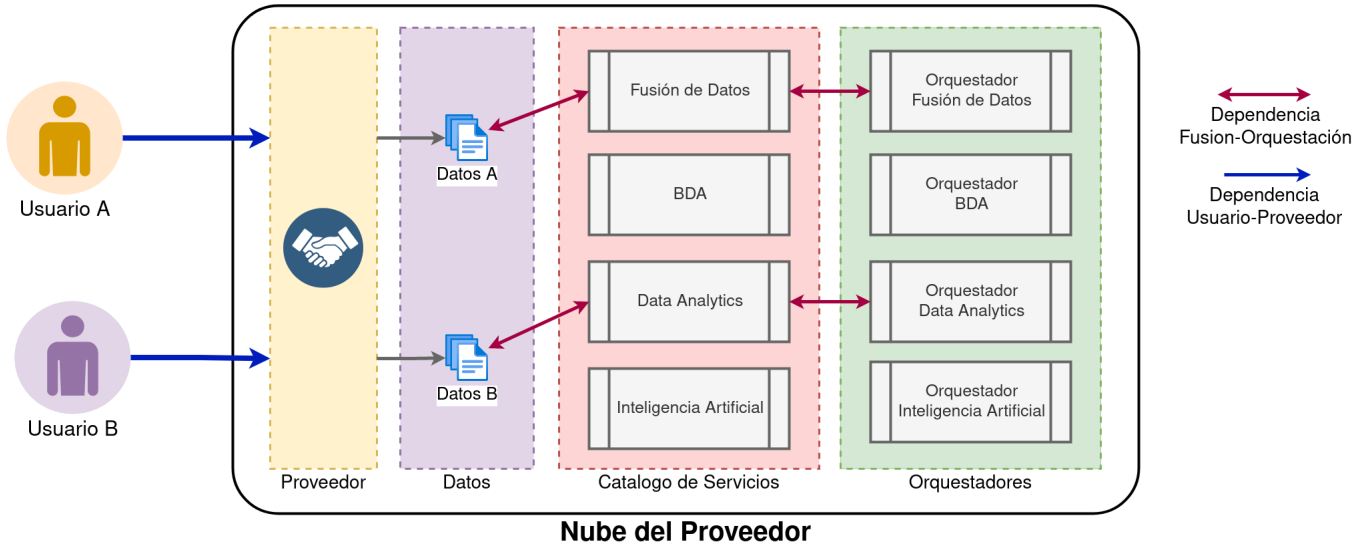


Figura 3: Dependencias entre el usuario y los servicios de BD.

### Dependencia usuario-proveedor

1. Los usuarios deben crear sus servicios  $FD$  asumiendo como límites los recursos ofrecidos por el proveedor.
2. El proveedor determina los factores no-funcionales (seguridad, confiabilidad, eficiencia, costos, etc).

### Dependencia fusión-orquestador

1. Las variables espaciales y/o temporales son comúnmente utilizadas para realizar procesos de  $FD$  debido a que este tipo de variables son generalmente creadas como metadata descriptora de contexto de los eventos (e.g. registros de una base de datos o acciones en contenidos). En este sentido, el espacio/tiempo de eventos descritos en una fuente de datos se puede equiparar con el espacio/tiempo de eventos de otra fuente, aunque dichas fuentes sean de naturaleza heterogénea. Razón por la cual, estas variables son comúnmente utilizadas como denominador común en procesos de  $FD$ .
2. La orquestación de datos está ligada funcionalmente (por las tareas que se realizan) a la infraestructura, plataforma y/o software empleados.
3. Los métodos de  $FD$  dependen funcionalmente de la orquestación definida por el proveedor.

## 4. Hipótesis

Con base en las preguntas de investigación y las premisas establecidas, es posible definir la hipótesis de este trabajo de tesis:

*Un método de orquestación, independiente de la infraestructura y determinado por esquemas basados en variables espacio-temporales, puede crear servicios dinámicos de  $FD$  que mitiguen los efectos de las dependencias usuario-proveedor y fusión-orquestador.*



## 5. Objetivos generales y particulares del proyecto

### General

Crear un método de orquestación agnóstico para servicios dinámicos de *FD* sobre fuentes de datos incluyendo variables espacio-temporales.

### Particulares

1. Definir un modelo de procesamiento agnóstico (de la infraestructura y/o plataforma) para crear servicios dinámicos de *FD* sobre múltiples entornos de nube mediante cláusulas declarativas.
2. Crear un esquema de despliegue y acoplamiento de servicios de *FD* definido por variables espacio-temporales empleando el modelo anterior.
3. Diseñar un esquema de validación del método propuesto empleando un conjunto de datos climatológicos (variables espacio-temporales).

## 6. Introducción a la primera aproximación conceptual de la solución propuesta

En esta sección se describe, de forma resumida, una primera aproximación conceptual de la solución propuesta para enfrentar al problema mencionado previamente. La solución propuesta considera el diseño, desarrollo e implantación de un método de orquestación, independiente de la infraestructura y determinando por esquemas basados en variables espacio-temporales para crear servicios dinámicos de *FD*.

En esta sección, sólo se esbozan las posibles fases del método a partir de una representación conceptual de la fase de diseño de dicho método, así como una primera aproximación del modelo de procesamiento agnóstico para crear servicios dinámicos de *FD* sobre múltiples entornos de nube. A partir de este modelo también se describen algunos posibles esquemas de despliegue y acoplamiento de servicios de *FD* que se podrían obtener.

Se describe entonces el avance conseguido hasta el momento sobre una aproximación incipiente de la fase de diseño, con el fin de brindar un panorama que permita identificar y visibilizar el trabajo de investigación que se propone realizar para llevar a cabo la formalización y materialización de las fases del método y su correspondiente secuencia.

### 6.1. Premisa conceptual del método de orquestación para servicios dinámicos de *FD*

Como se ha descrito previamente, un modelo de *FD* considera múltiples entradas de tipos heterogéneas (datos, características y decisiones); que son integrados mediante intersecciones o uniones por un ente transformador (proceso de *BDA*) en un solo resultado que es entregado a un servicio consumidor (ya sea otro *FD*, otro *BDA* o *datawarehouse/datalake*). Esto crea relaciones recursivas en un servicio *FD* que no necesariamente pueden ser modeladas por un sistema *ETL* tradicional, pues este modelo tradicional considera un solo punto de obtención y un solo punto de entrega.

En cambio, en este trabajo de tesis se propone modelar un servicio de  $FD$  como un sistema de compuertas que sirva de guía para realizar la orquestación de los datos. Al emular un sistema de compuertas el modelo propuesto puede absorber todas las múltiples entradas esperadas en un  $FD$  así como las posibles interconexiones recursivas que se podrían presentar dependiendo de las necesidades de los usuarios finales.

Con base en las dos dependencias que se tienen del proveedor para  $FD$ , a continuación se describen dos maneras de mitigar tales dependencias.

## 6.2. Mitigación de la dependencia usuario-proveedor

Para la dependencia usuario-proveedor se creará un método de despliegue y acoplamiento de servicios  $FD$  que permitirá a los usuarios finales crear estos servicios mediante un esquema de cláusulas declarativas (por definir). En este sistema, el usuario final manejará conjuntos de componentes de un  $FD$  como se ilustra a continuación:

$$FD = \{MDS_{i=1}^n \xrightarrow{D_i} BDA \xrightarrow{R_i} Sk_l\}$$

donde:

$$\begin{aligned} MDS &= \{DS_i, DS_{i+1}, \dots, DS_n\}, (n > 1) \\ DS_i &= (Id, Path), DS_i \neq \emptyset \\ BDA &= \{DA_j, DA_{j+1}, \dots, DA_m\}, (m > 1) \\ DA_j &= (Id, Path), DA_j \neq \emptyset \\ MDSk &= \{DSk_k, DSk_{k+1}, \dots, DSo\}, (o > 1) \\ DSk_k &= (Id, Path), DSk_k \neq \emptyset \\ Sk_l &\in SinkPool, SinkPool = \{BDA \vee FD \vee MDSk_k\} \end{aligned}$$

El significado de las variables es el siguiente:

- $MDS$  = Múltiples fuentes de datos (*Multiple Data Source*)
- $DS$  = Fuente de datos (*Data Source*)
- $BDA$  = Conjunto de procesos de Big Data Analytics (*Big Data Analytics*)
- $DA$  = Análisis de Datos (*Data Analytics*)
- $MDSk$  = Múltiples depósitos de datos (*Multiple Data Sink*)
- $DSk_k$  = Fuente para depósito de datos (*Data Sink*)
- $SinkPool$  = Conjunto de depósitos de datos (*SinkPool*)
- $Sk_l$  = Depósito de datos

De esta forma, mediante las cláusulas declarativas anteriores, los usuarios finales pueden definir las variables del modelo de despliegue y acoplamiento. Por ejemplo, la cláusula

$$FD = \{\{DS_1, DS_2\} \xrightarrow{spatioPatt} SpatioIntersection\}$$

significa que la  $FD$  se realizará mediante una intersección de la variable espacial (e.g. altitud-longitud), que en este ejemplo se establece como  $spatioPatt = (altitudX, longitudY)$ . De la misma forma se declara que el conjunto de las fuentes de datos  $MDS$  es  $\{DS_1, DS_2\}$ , donde  $DS_1$  podría tomar el siguiente valor  $DS_1 = (Merra, path = /Amazon/MERRA^4)$ , y  $DS_2$  podría tomar el valor  $DS_2 = (EMAS, path = /google/EMAS^5)$ .

El mismo procedimiento aplicaría para el resto de cláusulas, tales como  $DA$ s o  $BDA$  y los  $Sk_l$  finales. De esta forma la ubicación de los componentes son definidos por el usuario final mediante este esquema declarativo, con lo cual se podría quitar la dependencia usuario-proveedor (Figura 4). Las fuentes  $DS$ s, los  $BDA$  y los  $SK_l$  podrían estar en infraestructuras distintas, compartiendo infraestructura o centralizadas en una misma infraestructura.

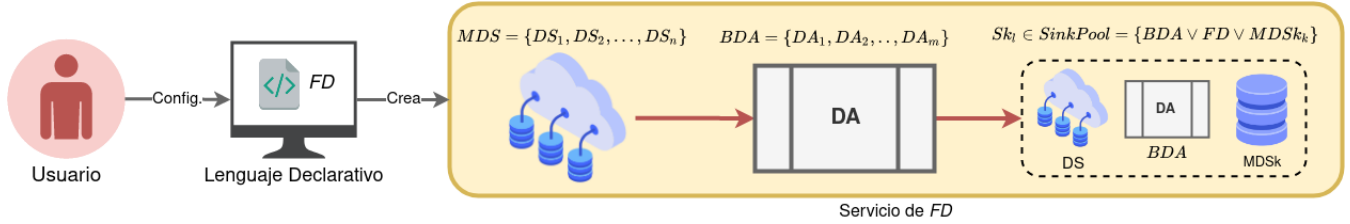


Figura 4: Creación de Servicios de  $FD$ .

### 6.3. Mitigación de la dependencia de fusión-orquestador

Para esta dependencia se planea, a partir del servicio de  $FD$  creado por el usuario con el esquema declarativo, crear un servicio de orquestación que pueda desplegar múltiples versiones del servicio de  $FD$  en diferentes infraestructuras. Estas versiones dependerían de las variables elegidas por el usuario. En este trabajo de tesis se trabajará con dos tipos de variables que suelen ser comunes en los servicios de  $FD$ , variables *espaciales* y variables *temporales* [34].

La idea es que el usuario final no solamente defina las cláusulas de las fuentes de datos, sino que también indique qué partes de esas fuentes se deben usar en cada caso. Esto puede generar tantas versiones de la  $FD$  como variables espacio-temporales elija el usuario. En este trabajo, para manejar estas variables, se propone una estructura de *cubo de datos* (como ilustra la Figura 5), con sus conocidas 3 dimensiones ( $X, Y, Z$ ) de entrada. Donde el cubo representa una fuente de datos, donde sus ejes  $X$  y  $Y$  representarán, dependiendo de la elección del usuario final, ya sea una variable temporal o una variable espacial. El eje  $Z$  es una variable de  $FD$  que indique ya sea parámetros de ejecución de un  $BDA$  o parámetros  $X, Y, Z$  de otro cubo (fuente de datos).

Asumamos un cubo creado a partir de una fuente por ejemplo  $DS_i = EMAS$ , la cual en efecto tiene variables *espaciales* (latitud-longitud de los sensores de temperatura desplegados en campo por la CONAGUA -Comisión Nacional del Agua-) que se pueden representar en la dimensión

<sup>4</sup>MERRA (Modern-Era Retrospective Analysis for Research and Applications) es un proyecto satelital de la NASA que proporciona datos meteorológicos de distintas localidades desde 1980 con una semana de retraso a partir del tiempo actual. - <https://gmao.gsfc.nasa.gov/reanalysis/MERRA/>

<sup>5</sup>EMAS es un sistema que produce datos meteorológicos sobre el territorio mexicano a través de antenas. - <https://smn.conagua.gob.mx/es/observando-el-tiempo/estaciones-meteorologicas-automaticas-ema-s>

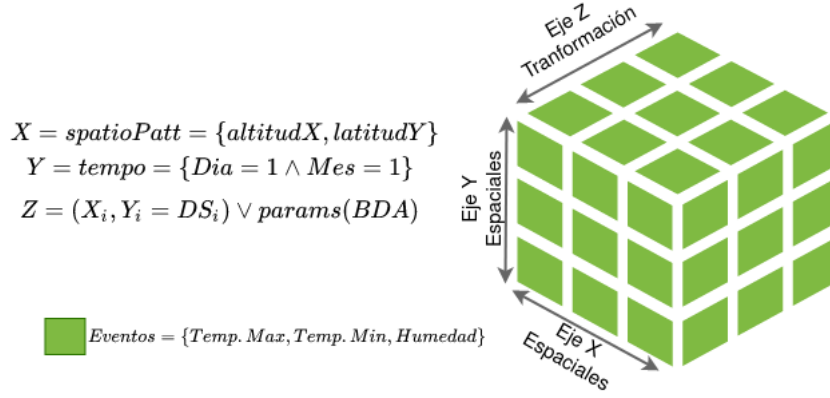


Figura 5: Cubo espacio-temporal.

$X$  del cubo; y variables *temporales* (expresadas en día, mes y año) que pueden representarse en la dimensión  $Y$ . Asumamos también que la dimensión  $Z$  puede representar la respuesta de una función de búsqueda en otro cubo (supongamos la fuente *MERRA*, la cual también posee variables espacio-temporales). Como resultado ( $Z = (X_1, Y_i \in DS = MERRA)$ ) retornará los valores  $X, Y \in MERRA$  para los valores  $X, Y \in EMAS$ .

En este contexto, un usuario final puede definir un *FD* de la siguiente forma:

$$\begin{aligned}
 FD_{v1} = Cube = (X, Y, Z) = \\
 (X = spatioPatt = (latitud_X, longitudud_Y \in EMAS), \\
 Y = tempo = (dia = 1 \wedge mes = 1) \in EMAS, \\
 Z \in MERRA = (X, Y \in EMAS)
 \end{aligned}$$

Este servicio ( $FD_{v1}$ ) es un clon de un *FD* que ejecuta un *BDA* (p.ej. clustering, fusión, etc), el cual ha sido definido en la fase declarativa de *FD* y al que se le pasa como parámetro de ejecución un cubo (*Cube*). El cubo representa un entorno, la selección de parámetros  $X, Y, Z$  representa un espacio de direcciones acotado de ese entorno. La dimensión  $Z$  indica el mismo espacio de direcciones (entrada/salida), pero de otro entorno  $[(Z \in MERRA), (X, Y \in EMAS)]$ . De esta forma se pueden crear otras versiones del *FD* mediante el cambio de parámetros  $X, Y, Z$ . Por ejemplo:

$$\begin{aligned}
 FD_{v2} = Cube1 = (X, Y, Z) = \\
 (X = spatioPatt = (latitud_X, longitudud_Y \in EMAS), \\
 Y = tempo = (dia = 2 \wedge mes = 1) \in EMAS, \\
 Z \in MERRA = (X, Y \in EMAS)
 \end{aligned}$$

Este servicio ( $FD_{v2}$ ) es un clon de del servicio  $FD_{v1}$  pero procesará el día 2 de las fuentes de datos, mientras que  $FD_{v1}$  procesará el día 1. Como resultado, ambos sistemas se pueden ejecutar en paralelo y/o en forma distribuida incluso si estos están desplegados en distintas infraestructuras. Básicamente este modelo crea patrones *map-reduce* y *divide&conquer* en armonía con el esquema de orquestación, el cual es definido por el esquema declarativo.

## 6.4. Esquema conceptual propuesto

En la Figura 6 se muestra un ejemplo de cómo podría ser el servicio de *FD* orquestado propuesto en este trabajo de tesis. Se observa un escenario en donde existen múltiples fuentes de datos (*MDS*), las cuales contienen datos en crudo que son adquiridas por una entidad llamada *Orquestador*. El orquestador organiza los procesos que se llevarán a cabo dentro del ciclo de vida del servicio de *FD*, por ejemplo las funciones dedicadas al preprocesamiento, pre-fusión (p.ej. algoritmos de manejo *AM*<sup>6</sup> y algoritmos de análisis *AA*<sup>7</sup>) y funciones *BDA* de los datos obtenidos de las *MDS*. Durante cada una de estas etapas de procesamiento se producen como resultado cubos de datos (*Cube*) con el formato presentado en la Figura 5. Los resultados entregados por las funciones *BDA* pueden ser transmitidas a un conjunto de depósitos de datos (*DSK*), un conjunto de procesos *BDA* u otro servicio de *FD*.

## 7. Metodología de trabajo

Como guía para el desarrollo de la propuesta de este trabajo de tesis se estableció una metodología de trabajo, la cual consta de las siguientes etapas:

- **Fase 1:** Diseño e implementación de la solución propuesta.  
En esta etapa se definirá cada tarea en donde se realizara la solución antes planteada así como el estado del arte sobre esta tesis y un modelo declarativo.
  - **1.1** Investigación en la literatura de los trabajos reportados.
    - **1.1.1** Conceptos relevantes. Revisión y redacción de los conceptos relevantes y como han sido aplicados dentro del trabajo de tesis.
    - **1.1.2** Trabajos relacionados. Investigación de los diferentes trabajos relacionados y sus semejanzas con el trabajo de tesis.
    - **1.1.3** Revisión y actualización de los trabajos relacionados. Redacción de los trabajos relacionados.
  - **1.2** Definición de un modelo declarativo.
    - **1.2.1** Diseño del modelo. Diseñar un modelo de orquestación de datos para *FD*.
    - **1.2.2** Implementación del modelo. Conformar al modelo diseñado, implementar un lenguaje declarativo que interprete este modelo.
  - **1.3** Implementación de un prototipo.
    - **1.3.1** Diseño de la arquitectura. Desarrollar una arquitectura de sistemas distribuidos que mejor se adapte al modelo de orquestación de datos para *FD*.
    - **1.3.2** Desarrollo del prototipo. A partir del diseño, crear un prototipo funcional que nos permita realizar los objetivos planteados.
    - **1.3.3** Despliegue, pruebas de funcionalidad y corrección de defectos. Generar pruebas sobre el comportamiento del prototipo y solucionar errores encontrados.
- **Fase 2:** Experimentación.  
En esta etapa a partir del modelo y el prototipo creado en la fase anterior se realizará la experimentación correspondiente, para conocer el comportamiento del proyecto.

---

<sup>6</sup>*AM* es un tipo de algoritmo encargado de realizar el manejo de cubos de datos.

<sup>7</sup>*AA* es un tipo de algoritmo encargado de ejecutar técnicas de analítica basado en cubos de datos.

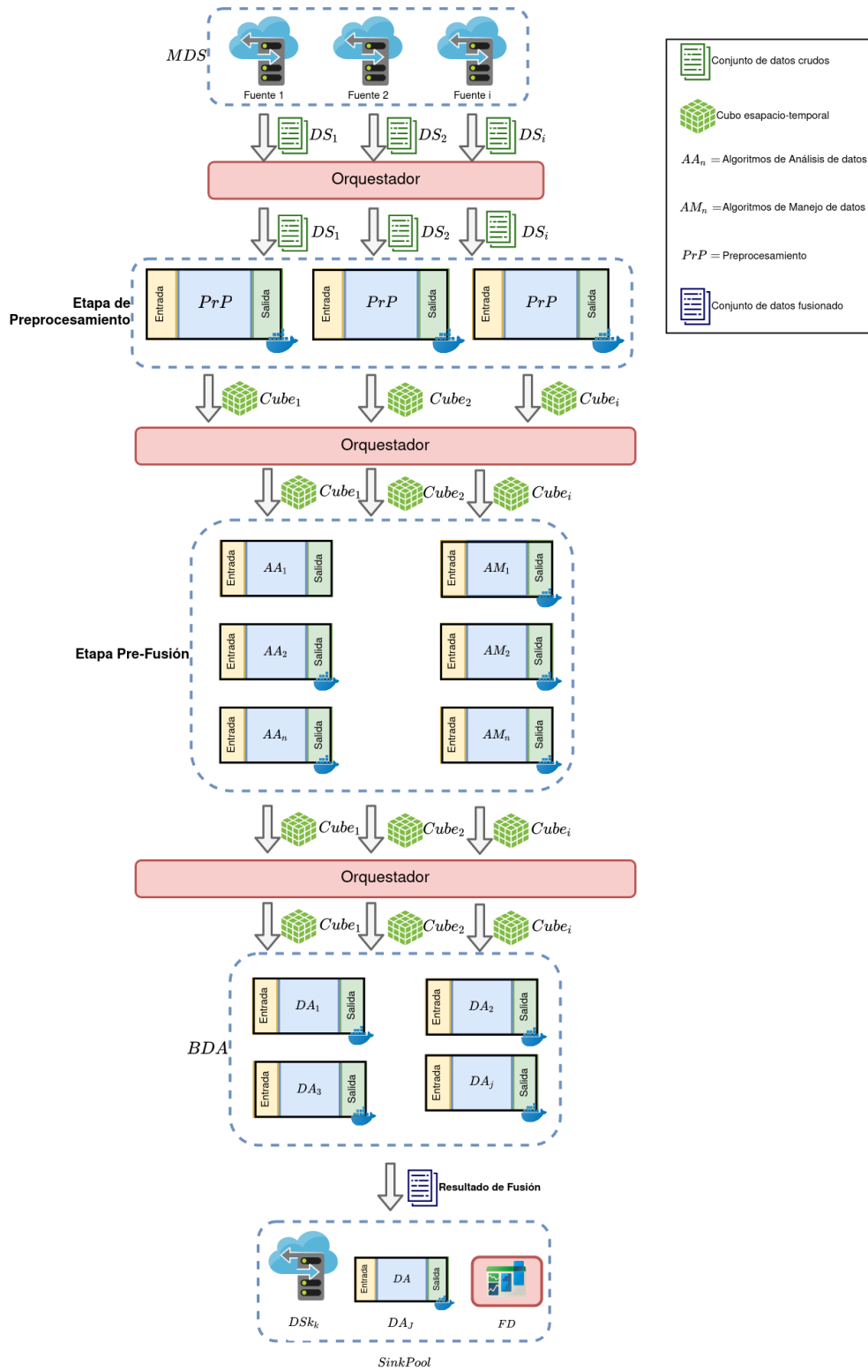


Figura 6: Ciclo de vida del servicio de  $FD$  Orquestado propuesto.

- **2.1** Despliegue del prototipo en una infraestructura. Colocar el prototipo funcional en una infraestructura para obtener el mejor desempeño.
  - **2.2** Definición de métricas de evaluación. Definir aquellas mediciones que se presenten en el prototipo y que funcionen para conocer el desempeño del sistema.
  - **2.3** Definición de experimentos. Una vez teniendo las métricas a obtener, enumerar los diferentes experimentos que probaran el prototipo generado.
    - **2.3.1** Definición de variación de experimentos. Definir los diferentes comportamientos de los experimentos
    - **2.3.2** Definición de configuraciones. Definir las configuraciones se presenten las métricas establecidas anteriormente.
    - **2.4.3** Pruebas. Realización de las pruebas conforme a las configuraciones establecidas.
  - **2.4** Estudio de caso.
    - **2.4.1** Definición del caso de estudio. Relación de la *FD* dentro de servicios de orquestación de datos.
    - **2.4.2** Definición del ambiente experimental. Definición de configuraciones donde este presente el caso de estudio.
    - **2.4.3** Pruebas. Realización de las pruebas conforme a las configuraciones establecidas.
- **Fase 3:** Análisis de Resultados.
- En esta fase, expondremos los resultados y su interpretación de acuerdo a los datos que se obtuvieron en los experimentos de la fase anterior.
- **3.1** Análisis de resultados de la evaluación de experimentos controlados.
    - **3.1.1** Análisis de rendimiento.
      - ◊ Analizar las diferentes pruebas conforme a las métricas establecidas.
      - ◊ Análisis de sobrecarga (overhead) adicional, en términos de cómputo y latencia, de la inclusión del orquestador de datos en el ciclo de vida de la fusión de datos.
    - **3.1.2** Análisis de variabilidad estadística. Análisis de las diferentes medidas de variabilidad con respecto a las métricas establecidas.
  - **3.2** Análisis de resultados del estudio de caso sobre datos climatológicos.
    - **3.2.1** Análisis de Rendimiento. Analizar las diferentes pruebas conforme a las métricas establecidas en el caso de estudio.
    - **3.2.2** Análisis de variabilidad estadística. Análisis de las diferentes medidas de variabilidad con respecto a las métricas establecidas.
- **Fase 4:** Redacción de documentación final.
- En esta fase se describirá la escritura de la tesis así como la generación de un artículo científico.
- **4.1** Escritura de tesis: Escritura de los siguientes capítulos de manuscrito:
    - Introducción: Se describirá el objeto de estudio, antecedentes, motivación, objetivos, hipótesis y establecimiento del problema.

- Marco Teórico: Se definirá los conceptos básicos asociados al objeto de estudio, así como técnicas, algoritmos y modelos sobre los que se fundamenta la solución propuesta al problema establecido.
  - Trabajos relacionados: Se describirán soluciones estrechamente asociadas a la solución propuesta y realizando un estudio cualitativo comparativo entre dichas soluciones y la solución propuesta.
  - Formalización del método propuesto: Se describirán en detalle cada componente de la solución propuesta que incluye los modelos, esquemas y mecanismos creados para materializar la solución propuesta.
  - Resultados y evaluación experimental: Se describirán la metodología para realizar la evaluación experimental, así como métricas y características de la infraestructura utilizada, así como las baterías de experimentación para pruebas de concepto y estudios de caso.
  - Conclusiones: Se interpretarán los resultados obtenidos.
  - Identificación de limitantes: Expresar aquellas áreas en donde el tema de tesis no está involucrado.
  - Definición de trabajo futuro. Redacción de las posibles mejoras dentro del trabajo realizado.
- **4.2** Generación de reporte técnico o artículo científico.



## 8. Cronograma de actividades

A continuación se presenta el cronograma de las actividades a realizar durante el desarrollo de este trabajo de investigación.

Actividades	2021				2022							
	Septiembre	Octubre	Noviembre	Diciembre	Enero	Febrero	Marzo	Abril	Mayo	Junio	Julio	Agosto
<b>1. Diseño e implementación de la solución propuesta</b>												
<b>1.1 Investigación en la literatura de los conceptos...</b>												
1.1.1 Conceptos relevantes												
1.1.2 Trabajos relacionados												
1.1.3 Revisión y actualización de los trabajos relacionados												
<b>1.2 Definición del modelo declarativo</b>												
1.2.1 Diseño del modelo												
1.2.1 Implementación del modelo												
<b>1.3 Implementación del prototipo</b>												
1.3.1 Diseño de la arquitectura												
1.3.2 Desarrollo del prototipo												
<b>2. Evaluación experimental.</b>												
<b>2.1 Despliegue del prototipo en una infraestructura</b>												
<b>2.2 Definición de métricas</b>												
<b>2.3 Definición de experimentos</b>												
2.3.1 Definición de variación experimental												
2.3.2 Definición de configuraciones												
2.3.3 Pruebas												
<b>2.4 Estudio de caso</b>												
2.4.1 Definición de caso												
2.4.2 Definición de ambiente experimental												
2.4.3 Pruebas												
<b>3. Análisis de resultados</b>												
<b>3.1 Análisis de resultados de la evaluación...</b>												
3.1.1 Análisis de rendimiento												
3.1.2 Análisis de variabilidad de estadística												
<b>3.2 Análisis de resultados del estudio de caso</b>												
3.2.1 Análisis de rendimiento												
3.2.2 Análisis de variabilidad de estadística												
<b>4. Redacción de documentación</b>												
<b>4.1 Escritura del documento de tesis</b>												
<b>4.2 Generación de reporte técnico o artículo científico</b>												

Figura 7: Plan de trabajo.

## 9. Infraestructura

La infraestructura que se piensa utilizar para el desarrollo de este trabajo de tesis, así como la infraestructura de experimentación se presentan a continuación.

### Equipo para desarrollo

- Procesador: Intel Core i5-6200U
- Memoria: 8GB
- Disco Duro: 1 TB
- Cores: 2 físicos y 2 virtuales

### Equipo para experimentación

Hostname	Sockets	Cores por Socket	Threads por Socket	RAM
Compute6	1	6	2	64GB
Compute8	2	8	1	64GB
Compute9	1	12	1	64GB
Compute10	1	12	1	64GB
Compute11	1	12	1	64GB
Compute12	1	12	1	64GB

Tabla 1: Equipo de cómputo que será utilizado para desplegar y evaluar los experimentos.

## 10. Estado del arte

El tema central de este proyecto de tesis es orquestación para crear servicios de fusión de datos. En esta sección se presenta un repaso de los conceptos abordados dentro de esta tesis y la comparación entre la fusión de datos con la integración de los mismos. También se describen algunos trabajos relacionados con la propuesta de tesis. La principal intención de esta sección es mostrar el estado del arte y dar a conocer la forma cómo se han abordado la orquestación de datos para fusión de datos en algunos dominios.

### 10.1. Conceptos Relevantes

En este apartado se muestran los conceptos fundamentales asociados con el desarrollo de esta tesis y cómo han sido involucrados dentro del tema y su relevancia dentro de las investigaciones y desarrollo de servicios.

#### Orquestación de datos

La orquestación en *Big Data* (BD) se refiere al control centralizado de los procesos que administran datos, sistemas, centros de datos o fuentes masivas de datos (también conocidas como *datalakes*).

Las herramientas de orquestación de *BD* permiten a los equipos de Tecnologías de la Información (*IT*) diseñar y automatizar procesos de un extremo a otro. Estos procesos pueden incorporar datos, archivos y dependencias completas de una organización, sin tener que escribir scripts personalizados [27].

De acuerdo con la empresa Advanced Systems Concepts, Inc.<sup>8</sup>, las plataformas de orquestación de datos permiten a los equipos de *IT* integrar rápidamente nuevas fuentes de datos existentes utilizando el modelo de procesamiento de datos *ETL* y procesos de *BD* [35].

La orquestación de datos permiten que cada punto de procesamiento de datos del usuario funcione en armonía dentro del flujo de trabajo que configure el mismo usuario. Las herramientas de orquestación de datos han tenido aplicación en áreas recientes, como en las redes telefónicas 5G [36] y sistemas de *blockchain* [37].

### Fusión de Datos

El Departamento de Defensa de Estados Unidos define la fusión de datos como “un proceso multinivel y multifacético que se encarga de la detección, asociación, correlación, estimación y combinación automáticas de datos e información de múltiples fuentes” [38]. Entre las áreas de aplicación se encuentra la combinación de datos espacio-temporales [6], médicos [8] y terrestres [7], esta última es un área fuertemente heterogénea para el desarrollo de sistemas de observación de la Tierra, en donde se requieren múltiples vistas sobre un objeto en común. En la fusión de datos se pueden presentar diversos modelos y/o técnicas que ayudan a realizar este proceso, por ejemplo el modelo propuesto por la Junta de Directores de Laboratorios (JDL<sup>9</sup> por sus siglas en inglés) concibe un modelo de procesos multinivel y fases que trabajan con la asociación, correlación y estimación de los datos [20] o bien la propuesta de Dasarathy [19], quien expone un método de clasificación de acuerdo a la naturaleza de los datos así como sus entradas y salidas (datos, características y decisiones).

Existen algunos trabajos enfocados en realizar el proceso de fusión de datos para observaciones médicas [39], observaciones dentro de bosques [40] y datos de variables espacio-temporales [41]. La fusión de datos puede ser utilizada para mejorar la calidad de los datos de entrada y los procesos de toma de decisiones, cuando se encuentre alguna relación entre las diferentes fuentes. Si las fuentes de datos por procesar son totalmente disjuntas, no es posible realizar el proceso de fusión.

### Flujos de Trabajos

Cuando se requiere del manejo especializado de las actividades dentro de un proceso es conveniente utilizar flujos de trabajo (*workflow*), los cuales son utilizados para la automatización de procedimientos en los que los datos, información o tareas se pasan entre los participantes del flujo, de acuerdo con un conjunto definido de reglas para lograr un objetivo [42].

Los flujos de trabajo pueden ser desplegados en escenarios de nube, por ejemplo Xu *et al.* [43] describen una estrategia para generar múltiples flujos de trabajo con el fin de tener una calidad de servicios eficiente dentro de una nube. Zulfiqar *et al.* [44] presentan un sistema de manejo de flujos de trabajo tolerante a fallos en la nube. Existen además otros trabajos en los cuales se puede ver

---

<sup>8</sup>ActiveBatch, Advanced Systems Concepts, Inc. - <https://www.advsyscon.com/>

<sup>9</sup>Joint Directors of Laboratories - Grupo de trabajo de fusión de datos, establecido en 1986

que los flujos de trabajo ayudan a tener el control y monitoreo de los diferentes puntos en los que se encuentra un proceso de interés.

### Modelo de procesamiento ETL

Extraer, Transformar y Cargar (*ETL -Extract, Transform, Load-*) es un modelo de procesamiento que permite realizar la adquisición de datos a partir de una fuente, su posterior transformación ejecutando alguna operación o modificación a los datos y su transferencia a un repositorio destino [11], como ilustra la Figura 1.

Las tareas que se realizan son:

- *Extraer*: consiste en extraer datos de una aplicación fuente o una fuente de datos.
- *Transformar*: los datos extraídos pasan a esta etapa y se transforman aplicando una serie de funciones o tareas de procesamiento.
- *Cargar*: los datos transformados finalmente se cargan en una fuente destino.

Los procesos que basados en este modelo *ETL* tienen aplicación dentro de *BigData*, por ejemplo, Bansal [45] propuso un *framework* semántico utilizando tecnologías semánticas para la integración y publicación de múltiples fuentes basándose en el modelo *ETL*. El modelo *ETL* también tiene cabida dentro del área médica, por ejemplo el *framework* para la conversión de bases de datos de salud a OMOP<sup>10</sup> [46] [47]. Con base en el proceso *ETL* se pueden construir procesos más grandes, incluso de un proceso *ETL* se puede pasar a otro proceso *ETL* y así sucesivamente para generar flujos de trabajo.

## 10.2. Trabajos relacionados

En este apartado se muestran algunos de los trabajos relacionados con el tema estudiado, que si bien no se generalizan explícitamente ni afrontan totalmente la problemática descrita, giran en torno a este trabajo de investigación. La búsqueda en la literatura se dividió en dos áreas: (a) trabajos relacionados con orquestación para fusión de datos y (b) herramientas que permiten realizar fusión de datos.

El tipo de dato en el que se focaliza este trabajo de tesis son los relacionados a datos climáticos. La información climática debe ser representativa y precisa del lugar donde las estaciones climatológicas se encuentran ubicadas, de esta manera los analistas puede realizar estudios relacionados con la predicción meteorológica y climática. Permitiendo tomar decisiones futuras sobre agricultura, desastres naturales y predicción del clima en una zona específica [48]. Con base en ello se desea ver la factibilidad de crear un método armonizado de fusión de datos agnóstico para la nube. El manejo de distintas fuentes de datos son un desafío para las aplicaciones actuales dado que requieren mejorar el manejo de los procesos y la calidad de información para la toma de decisiones en variables espacio-temporales dentro del análisis del clima [41].

---

<sup>10</sup>El modelo de datos común de OMOP permite transformar los datos de observación dispares en un formato común - <https://www.ohdsi.org/data-standardization/the-common-data-model>

### 10.2.1. Orquestación para fusión de datos

El análisis de los datos meteorológicos es un área propicia para la fusión de datos de distintas fuentes [49]. Esto se debe a la cantidad de fuentes de datos y la cantidad de datos que se generan diariamente. Es importante mencionar que no todas las fuentes de datos son candidatas para realizar el proceso de toma de decisiones debido a la poca información que contienen (carencias en los datos) y por ende es necesario complementarlas con variables de una o más fuentes de datos externas que tengan información sobre el área analizada.

En esta sección se presentan aquellos trabajos, que hemos identificado hasta el momento en la literatura, y que están relacionados los modelos de orquestación de datos. Estos trabajos tienen cierta relación con el objeto de este trabajo de tesis y la fusión de datos para análisis de comportamientos climáticos. Se seguirá revisando la literatura en los próximos meses.

#### A) Fusión de datos espaciales en infraestructuras de datos espaciales utilizando *Linked Data*

Este trabajo fue realizado por Wiemann y Bernard en 2015 [41], en éste describen enfoques, requisitos y factores limitantes para la fusión de datos espaciales basada en servicios, con un enfoque particular en la interacción de *SDI* (*Spatial Data Infrastructures*) y estándares de Web Semántica. Las SDI establecidas proporcionan un medio para publicar, buscar, acceder y procesar datos espacio-temporales en la Web [50]. Se aprovechan las tecnologías de la Web Semántica para permitir acceso ubicuo a datos interconectados en la Web [35].

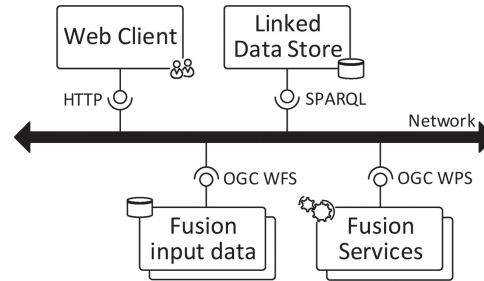


Figura 8: Infraestructura de servicios implementada para la fusión de datos espaciales [41].

En la Figura 8, se muestra la infraestructura que proponen los autores, la cual contiene 4 componentes: a) *Web Client* admite la creación y definición de flujos de trabajo de fusión, donde los flujos de trabajo creados comienzan con la selección de las fuentes. Una vez que se han seleccionado el conjunto de datos de referencia y el conjunto de datos de destino, el sistema proporciona información sobre la posible comparabilidad de los conjuntos de datos mediante el análisis de los sistemas de referencia espacial utilizados y sus extensiones espaciales. Una vez establecidos los parámetros de fusión y haber seleccionado los datos, el componente b) *Fusion input data* recibirá y almacenará temporalmente los datos a fusionar. Si un usuario desea comparar y combinar los conjuntos de datos mediante procesos de fusión personalizada, se debe proporcionar una instancia de OGC WPS (*OGC Web Processing Service* [51]) adecuada dentro del componente c) *Fusion Services*, que será el encargado de aplicar la técnica de fusión establecida. Para finalizar, los datos son alojados dentro del servicio de d) *Linked Data Store* y manejados por *SPARQL*. En la Figura 9 se describen los pasos base establecidos en el flujo de *FD*, los autores indican que no deben considerarse como

una secuencia estricta, ya que los procesos son independientes y es posible omitirlos, reiterarlos o combinarlos de manera diferente.



Figura 9: Proceso de fusión implementado en [41].

### B) Fusión y presentación de datos sanitarios mediante el mecanismo de orquestación de arquitectura orientada a servicios (SOA)

Este trabajo, publicado en 2011, no es especialmente dirigido al tratamiento de datos espacio-temporales, pero cuenta con algunos aspectos que se relacionan dentro del tema de tesis propuesto. Este proyecto desarrollado por Universidad de Thanjavur [52] propone fusión de datos en el dominio de *Smart Healthcare*, esto debido a que los sensores utilizados por los médicos producen datos que pueden ayudar a realizar una mejor atención médica y reducir el desencadenamiento de otras enfermedades en los pacientes.

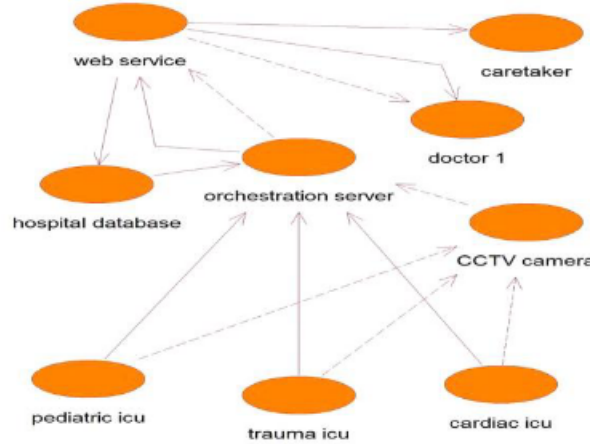


Figura 10: Arquitectura implementada en [52].

En este proyecto se propone la arquitectura que se muestra en la Figura 10, en la cual se consideran tres tipos de unidades de cuidados intensivos (*ICU* por sus siglas en inglés): pediatría, traumatología y cardíaca, cada una de estas contiene su propio conjunto de datos y tiene asignado un médico especializado. La fusión es necesaria cuando el médico envía información a un servicio web y todos los datos de las *ICU* se envían al servidor de orquestación, donde se almacenan en la base de datos junto con la transmisión de las cámaras CCTV de la *ICU* correspondiente. El orquestador es el encargado de organizar todos estos datos dependiendo del médico que haya realizado la carga de datos y así fusionarlos con el tipo de *ICU*.

### C) Formalización e implementación basada en web de fusión de datos espaciales.

Este trabajo, publicado en el año 2016 [53], está relacionado con la fusión de datos orquestada mediante un servidor web. Se basa en el trabajo de Wiemann y Bernard [41]. Los autores proponen una implementación sobre un servicio de fusión de datos dentro del área de datos espaciales. La arquitectura propuesta se muestra en la Figura 11. A partir de la interacción del usuario con la plataforma web, éste indica aquellas variables que desea fusionar. Esta plataforma tiene unos servidores a su disposición, los cuales están dedicados al almacenamiento de los datos, proceso de pre-fusión y de fusión de los datos. A partir de esto, siguiendo las configuraciones especificadas por el usuario, se pueda generar una fusión de datos espaciales personalizada con las variables que el usuario desee, pero teniendo en cuenta las fuentes que tenga la plataforma.

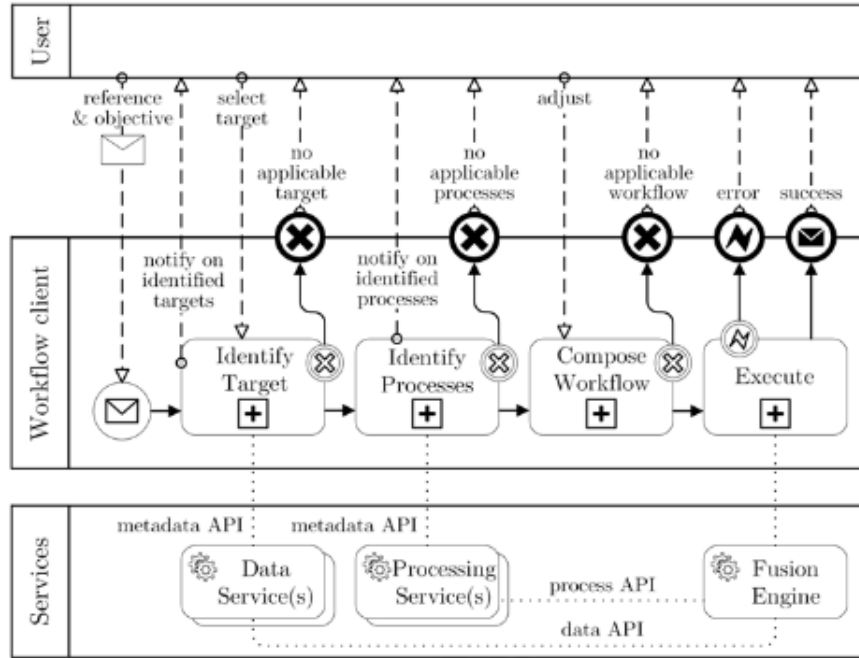


Figura 11: Flujo de trabajo y ejecución a través de la notación BPMN.

### D) Una plataforma de fusión de datos de autoevolución para modelos de agua a gran escala.

Li *et al.* [54] propusieron una plataforma para la adquisición y asimilación de datos de aspectos científicos sobre los factores humanos y climáticos que interactúan y están relacionados con la escasez mundial de agua. Este proyecto contiene un método para abordar la fusión automatizada de datos transferibles como datos para simulación y de observación del ciclo del agua a gran escala y de alta dimensión. Este enfoque permite la adaptabilidad de los datos a varios formatos, tiene la capacidad de caracterizar la heterogeneidad temporal y espacial; y puede representar estructuras complejas para distintos fines científicos. En la Figura 12 se muestra el flujo de trabajo desde

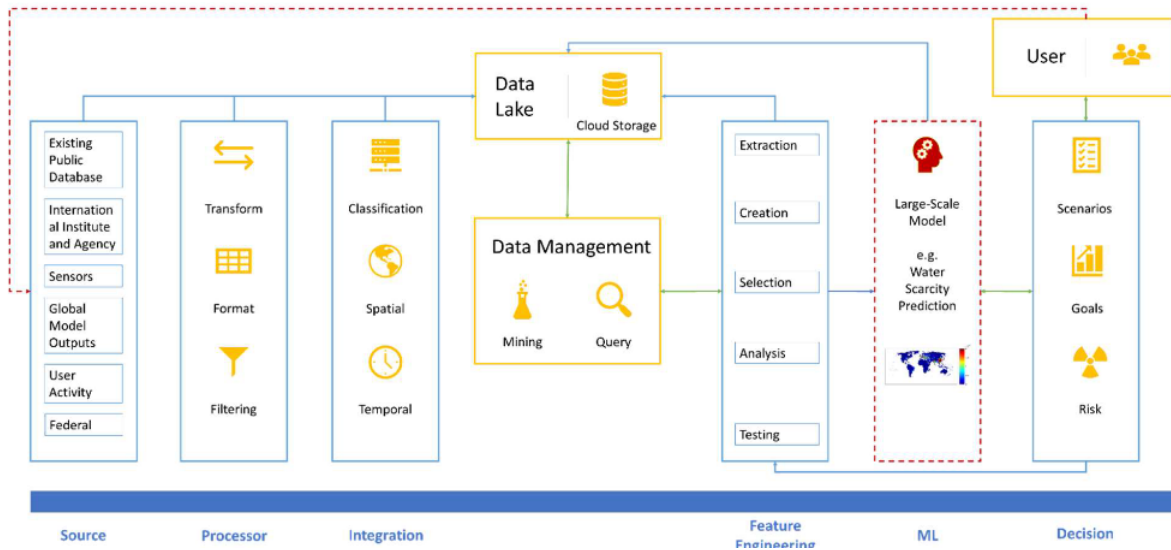


Figura 12: Arquitectura implementada para la fusión de datos automatizada en [54].

la recopilación de datos sin procesar hasta la decisión final de aplicar alguna técnica de *machine learning*. Los conjuntos de datos de diversas fuentes se recopilan, procesan y fusionan con la intersección de variables espacio-temporales, antes del almacenamiento en un *datalake* basado en la nube. De acuerdo a los autores, la orquestación de datos es realizada en el módulo llamado *Data Management*, donde internamente se aplican técnicas de minería de datos y consulta para ejecutar los procesos configurados por el usuario. El usuario es responsable de controlar en la plataforma la preparación de los conjuntos de datos de acuerdo con los objetivos y escenarios de que desee con la técnica de *machine learning* a emplear.

### 10.2.2. Servicios para fusión de datos

Existen algunos servicios en la nube que te proveen servicios de fusión de datos, creando flujos de trabajo donde el usuario puede personalizarlos a su manera y así generar sus propios procesos de fusión de datos. Al usar estos servicios el usuario da por hecho que utilizará las orquestaciones de datos fijas del proveedor para poder obtener un rendimiento aceptable. Los servicios más relevantes se describen brevemente a continuación.

#### *Google Cloud Data Fusion*

*Cloud Data Fusion*<sup>11</sup> es una plataforma propuesta por Google [25], la cual ayuda a crear servicios de flujos de procesamiento de datos con el fin de realizar integración de éstos. Esta plataforma ofrece integraciones predefinidas que ayudan a los usuarios a agilizar sus procesos del flujo de trabajo. Para mitigar los escenarios de *vendor lock-in* utiliza la tecnología de código abierto de CDAP<sup>12</sup> [55]. Propone la tecnología sin servidores (*serverless*<sup>13</sup>), lo que ayuda a los usuarios a evitar los cuellos

<sup>11</sup>Cloud Data Fusion - <https://cloud.google.com/data-fusion>

<sup>12</sup>CDAP es una aplicación de código abierto que funciona como integrador para la creación y manejo de aplicaciones en múltiples nubes

<sup>13</sup>En este modelo de servicio el usuario no requiere infraestructura propia, todo ejecuta en la infraestructura del proveedor de la nube, donde se realizan todas las tareas, el usuario sólo se encarga de definir los que desea usar.



de botella en los procesos mediante intuitiva interfaz.

### ***AWS Glue***

Amazon puso a disposición de los usuarios *AWS Glue*<sup>14</sup> [24] con el fin de crear servicios de integración de datos sin necesidad de manejar servidores (*serverless*). Con esto se facilita descubrir, preparar y combinar datos para análisis, dado que se cuentan con más recursos para el procesamiento de los datos, aprendizaje automático y desarrollo de aplicaciones. Esta plataforma cuenta con todos elementos básicos para la creación de servicios de fusión de datos y que los usuarios puedan acceder fácilmente a sus datos. El modelo de *AWS Glue* está basado con el modelo tradicional *ETL* [11], con el que los ingenieros de datos y los desarrolladores pueden utilizar *AWS Glue Studio* para crear, ejecutar y supervisar visualmente flujos de trabajo.

### ***Apache Airflow***

*Apache Airflow*<sup>15</sup> [26] es una plataforma para crear, programar y monitorear flujos de trabajo como grafos acíclicos dirigidos (DAG) de tareas [56]. El usuario que utiliza *Airflow* puede ejecutar sus tareas en una serie de trabajadores (procesos de ejecución) mientras siga las dependencias especificadas. Es decir, el usuario se debe de apegar a las reglas dadas por *Apache Airflow*. La idea de un lenguaje declarativo en la plataforma facilita a los usuarios la realización de flujos de trabajo mediante procesos independientes. La interfaz de usuario facilita también la visualización de las ejecuciones en producción, con lo cual el usuario podrá monitorear el progreso y solucionar problemas cuando sea necesario. *Apache Airflow* es un proyecto de código abierto que permite a los desarrolladores orquestar flujos de trabajo para extraer, transformar, cargar y almacenar datos.

### **10.2.3. Resumen**

En esta sección se realiza una Comparación cualitativa de características funcionales de los métodos de orquestación de datos disponibles en el estado del arte cercanos al método propuesto. Esta comparativa sigue las características comúnmente evaluadas entre los servicios de orquestación actualmente disponibles [57].

En la Tabla 2 se muestra una comparativa entre los proveedores de servicios de *FD* y el trabajo relacionado descrito anteriormente. Se puede observar en la tabla que estos diferentes trabajos tienen cierta relación con la solución propuesta en este trabajo de tesis. En la tabla se muestran las características identificadas como comunes entre estos trabajos:

1. *Enfoque de desarrollo*: objetivo con el cual fue desarrollado el trabajo.
2. *Posibilidad de añadir nuevas fuentes de datos*: capacidad de que los usuarios añadan nuevas instancias de fuentes de datos de origen.
3. *Reducen los escenarios de dependencias*: reducir las dependencias encontradas dentro de la problemática.
4. *Estandarización de datos personalizada*: establecer una adaptación personalizada para normalizar los datos.

---

<sup>14</sup>AWS Glue - <https://aws.amazon.com/es/glue/>

<sup>15</sup>Apache Airflow - <https://airflow.apache.org/>

5. *Permite datos no estructurados*: lectura y manejo de los datos no estructurados.
6. *Monitoreo*: permite la visualización de comportamiento en las diferentes etapas de los flujos de trabajo.

	<i>Google Cloud</i> [25]	<i>AWS Glue</i> [24]	<i>Airflow</i> [26]	A [41]	B [52]	C [53]	D [54]
<b>1</b>	Integración de Datos	ETL, Integración de Datos	Orquestación, <i>Workflows</i>	Fusión de datos, Orquestación de <i>Workflows</i>	Fusión de datos, Orquestación de <i>Workflows</i>	Fusión de datos	Fusión de datos
<b>2</b>	Posible	Posible solo con códigos en Scala o Python	Posible	No Posible	No Posible	No Posible	No especificado
<b>3</b>	Reduce con CDAP [55]	Medianamente Posible	No especificado	No Posible	No Posible	No Posible	Medianamente Posible
<b>4</b>	Posible	Posible	Posible	No posible	No Posible	No Posible	No Posible
<b>5</b>	Posible	Posible	Posible	No posible	No Posible	No Posible	Posible
<b>6</b>	Posible	No, pero posible	Posible	Posible	No Posible	Posible	Medianamente Posible

Tabla 2: Comparación cualitativa de características funcionales de los métodos de orquestación de datos disponibles en el estado del arte.

Se espera que el método propuesto permita la orquestación de datos para realizar fusión de datos entre las distintas fuentes de origen, teniendo la capacidad permitir la adición de nuevas fuentes de datos. Con este método de orquestación se pretende reducir los efectos de las dependencias *usuario-proveedor* y *fusión-orquestador*, para que el usuario final defina las tareas de procesamiento/análisis de los datos que crea convenientes con base en sus fuentes de datos. El método propuesto contemplaría el manejo de datos de tipo no estructurados y permitiría el monitoreo de las diferentes fases de *FD* o *BDA* que configure el usuario final.

## 11. Contribuciones o resultados esperados

- Un nuevo método para la orquestación de datos basado en variables espacio-temporales configurables para crear servicios de fusión de datos independiente de la infraestructura y/o plataforma.
- Un prototipo funcional que implemente el método propuesto.
- Un informe técnico o publicación en congreso que reporte el trabajo realizado.

## Referencias

- [1] Youssra Riahi. Big data and big data analytics: Concepts, types and technologies. International Journal of Research and Engineering, 5:524–528, 11 2018.
- [2] V Rajaraman. Big data analytics. Resonance, 21(8):695–716, 2016.
- [3] Charles F Hofacker, Edward Carl Malthouse, and Fareena Sultan. Big data and consumer behavior: Imminent opportunities. Journal of consumer marketing, 2016.
- [4] Ying Chen, JD Elenee Argentinis, and Griff Weber. Ibm watson: how cognitive computing can be applied to big data challenges in life sciences research. Clinical therapeutics, 38(4):688–701, 2016.
- [5] Samir El-Seoud, Hosam El-Sofany, Mohamed Abdelfattah, and Reham Mohamed. Big data and cloud computing: Trends and challenges. International Journal of Interactive Mobile Technologies (IJIM), 11:34, 04 2017.
- [6] MG Schultz, Clara Betancourt, Bing Gong, Felix Kleinert, Michael Langguth, LH Leufen, Amirpasha Mozaffari, and Scarlet Stadtler. Can deep learning beat numerical weather prediction? Philosophical Transactions of the Royal Society A, 379(2194):20200097, 2021.
- [7] Marzieh Fathi, Mostafa Haghi Kashani, Seyed Mahdi Jameii, and Ebrahim Mahdipour. Big data analytics in weather forecasting: a systematic review. Archives of Computational Methods in Engineering, pages 1–29, 2021.
- [8] Mihoko Okada. Big data and real-world data-based medicine in the management of hypertension. Hypertension Research, 44(2):147–153, 2021.
- [9] Farhad F Yusifov, Narmina E Axundova, et al. Analysis of demographic indicators based on e-demography data system. İTP Jurnalı, 2021.
- [10] Juan José Camargo-Vega, Jonathan Felipe Camargo-Ortega, and Luis Joyanes-Aguilar. Conociendo Big Data. Revista Facultad de Ingeniería, 24:63 – 77, 01 2015.
- [11] ¿en qué consiste un proceso de etl (extraer, transformar y cargar)? - talend, Jan 2021.
- [12] Danny Grasse and Greg Nelson. Base sas® vs. sas® data integration studio: Understanding etl and the sas tools used to support it. SAS Users Group International, 2006.
- [13] Ashutosh Kumar Dubey, Abhishek Kumar, and Rashmi Agrawal. An efficient aco-pso-based framework for data classification and preprocessing in big data. Evolutionary Intelligence, 14(2):909–922, 2021.
- [14] Abhishek Behl, Pankaj Dutta, Stefan Lessmann, Yogesh K Dwivedi, and Samarjit Kar. A conceptual framework for the adoption of big data analytics by e-commerce startups: a case-based approach. Information systems and e-business management, 17(2):285–318, 2019.
- [15] Gang Li, Jianlong Tan, and Sohail S Chaudhry. Industry 4.0 and big data innovations, 2019.
- [16] Ian Gorton and John Klein. Distribution, data, deployment: Software architecture convergence in big data systems. IEEE Software, 32(3):78–85, 2015.

- [17] Federico Castanedo. A review of data fusion techniques. The Scientific World Journal, 2013:1–2, 2013.
- [18] Hugh F Durrant-Whyte. Sensor models and multisensor integration. In Autonomous robot vehicles, pages 73–89. Springer, 1990.
- [19] Belur V Dasarathy. Sensor fusion potential exploitation-innovative architectures and illustrative applications. Proceedings of the IEEE, 85(1):24–38, 1997.
- [20] Erik Blasch, Alan Steinberg, Subrata Das, James Llinas, Chee Chong, Otto Kessler, Ed Waltz, and Frank White. Revisiting the jdl model for information exploitation. In Proceedings of the 16th International Conference on Information Fusion, pages 129–136, 2013.
- [21] Vladimir Gajić. Historical review of study on weather influence on people and development of medical meteorology. ABC-časopis urgentne medicine, 13(2-3):65–69, 2013.
- [22] V Epitropou, K Karatzas, A Karppinen, J Kukkonen, and A Bassoukos. Orchestration services for chemical weather forecasting models in the frame of the pescado project. In 8th International Conference on Air Quality—Science and Application; Athens, pages 19–23, 2012.
- [23] Ashit Talukder, Mohammed Elshambakey, Sameer Wadkar, Huikyo Lee, Luca Cinquini, Shannon Schlueter, Isaac Cho, Wenwen Dou, and Daniel J Crichton. Vifi: Virtual information fabric infrastructure for data-driven discoveries from distributed earth science data. In 2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCCom/IOP/SCI), pages 1–8. IEEE, 2017.
- [24] Kalyan Sudhakar. Amazon web services (aws) glue. International Journal of Management, IT and Engineering, 8(9):108–122, 2018.
- [25] Nitin Naik. Connecting google cloud system with organizational systems for effortless data analysis by anyone, anytime, anywhere. In 2016 IEEE International Symposium on Systems Engineering (ISSE), pages 1–6. IEEE, 2016.
- [26] Pramod Singh. Airflow. In Learn PySpark, pages 67–84. Springer, 2019.
- [27] Yulei Wu. Cloud-edge orchestration for the internet-of-things: Architecture and ai-powered data processing. IEEE Internet of Things Journal, 2020.
- [28] Uroš Paščinski, Jernej Trnkoczy, Vlado Stankovski, Matej Cigale, and Sandi Gec. Qos-aware orchestration of network intensive software utilities within software defined data centres, 2018.
- [29] Marijn Janssen, Haiko van der Voort, and Agung Wahyudi. Factors influencing big data decision-making quality. Journal of business research, 70:338–345, 2017.
- [30] Said Elbanna. Strategic decision-making: Process perspectives. International Journal of Management Reviews, 8(1):1–20, 2006.
- [31] Justice Opara-Martins, Reza Sahandi, and Feng Tian. Critical analysis of vendor lock-in and its impact on cloud computing migration: a business perspective. Journal of Cloud Computing, 5(1):1–18, 2016.

- [32] Justice Opara-Martins, Reza Sahandi, and Feng Tian. Critical analysis of vendor lock-in and its impact on cloud computing migration: A business perspective. Journal of Cloud Computing, 5(1), 2016.
- [33] Shutdowns, turnaround y outages: Emerson us.
- [34] Hai Nguyen, Matthias Katzfuss, Noel Cressie, and Amy Braverman. Spatio-temporal data fusion for very large remote sensing datasets. Technometrics, 56(2):174–185, 2014.
- [35] Simplifying big data with data orchestration, Jan 2021.
- [36] Hind Fadhil Abbas. Management of network service orchestration and 5g networks. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 12(10):1109–1114, 2021.
- [37] Konstantinos Papadakis-Vlachopapadopoulos, Ioannis Dimolitsas, Dimitrios Dechouniotis, Eirini Eleni Tsiropoulou, Ioanna Roussaki, and Symeon Papavassiliou. On blockchain-based cross-service communication and resource orchestration on edge clouds. Informatics, 8(1), 2021.
- [38] Erik Blasch, Alan Steinberg, Subrata Das, James Llinas, Chee Chong, Otto Kessler, Ed Waltz, and Frank White. Revisiting the jdl model for information exploitation. In Proceedings of the 16th International Conference on Information Fusion, pages 129–136, 2013.
- [39] Rachel C King, Emma Villeneuve, Ruth J White, R Simon Sherratt, William Holderbaum, and William S Harwin. Application of data fusion techniques and technologies for wearable health monitoring. Medical engineering & physics, 42:1–12, 2017.
- [40] Thomas Hilker, Michael A Wulder, Nicholas C Coops, Julia Linke, Greg McDermid, Jeffrey G Masek, Feng Gao, and Joanne C White. A new data fusion model for high spatial-and temporal-resolution mapping of forest disturbance based on landsat and modis. Remote Sensing of Environment, 113(8):1613–1627, 2009.
- [41] Stefan Wiemann and Lars Bernard. Spatial data fusion in spatial data infrastructures using linked data. International Journal of Geographical Information Science, 30(4):613–636, 2016.
- [42] David Hollingsworth. Workflow management coalition the workflow management coalition specification workflow management coalition the workflow reference model.
- [43] Meng Xu, Lizhen Cui, Haiyang Wang, and Yanbing Bi. A multiple qos constrained scheduling strategy of multiple workflows for cloud computing. In 2009 IEEE International Symposium on Parallel and Distributed Processing with Applications, pages 629–634, 2009.
- [44] Zulfiqar Ahmad, Babar Nazir, and Asif Umer. A fault-tolerant workflow management system with quality-of-service-aware scheduling for scientific workflows in cloud computing. International Journal of Communication Systems, 34(1):e4649, 2021.
- [45] Srividya K. Bansal. Towards a semantic extract-transform-load (etl) framework for big data integration. In 2014 IEEE International Congress on Big Data, pages 522–529, 2014.
- [46] OHDSI. Omop common data model, 2021.

- [47] Juan C Quiroz, Tim Chard, Zhisheng Sa, Angus G Ritchie, Louisa Jorm, and Blanca Gallego. Extract, transform, load framework for the conversion of health databases to omop. medRxiv, 2021.
- [48] Rafael Guajardo P., Guadalupe Granados R., Ignacio Sánchez C., Gabriel Díaz P., and Finaaldina Borbosa M. Validacion espacial de datos climatologicos y pruebas de homogeneidad: Caso, veracruz, México, Jul 2017.
- [49] Andreas Pfeuffer and Klaus Dietmayer. Optimal sensor data fusion architecture for object detection in adverse weather conditions. In 2018 21st International Conference on Information Fusion (FUSION), pages 1–8, 2018.
- [50] Lars Bernard, Ioannis Kanellopoulos, Alessandro Annoni, and Paul Smits. The european geoportal—one step towards the establishment of a european spatial data infrastructure. Computers, Environment and Urban Systems, 29(1):15–31, 2005. Geoportals.
- [51] Anthony M Castronova, Jonathan L Goodall, and Mostafa M Elag. Models as web services using the open geospatial consortium (ogc) web processing service (wps) standard. Environmental Modelling & Software, 41:72–83, 2013.
- [52] Veeramuthu Venkatesh, Pethuru Raj, Kaushik Gopalan, and T Rajeev. Healthcare data fusion and presentation using service-oriented architecture (soa) orchestration mechanism. IJCA Special Issue on Artificial Intelligence Techniques-Novel Approaches & Practical Applications, 2:17–23, 2011.
- [53] Stefan Wiemann. Formalization and web-based implementation of spatial data fusion. Computers & Geosciences, 99:107–115, 2017.
- [54] Xinya Li, Chris Vernon, Min Chen, Heng Wang, and Zhangshuan Hou. A self-evolution data fusion platform for large-scale water models. 4 2021.
- [55] Randall C Gowat. Blockchain traceability for the counterfeit detection and avoidance program (cdap) final report. Technical report, Accenture Federal Services Arlington United States, 2020.
- [56] Patrick Healy and Nikola S Nikolov. How to layer a directed acyclic graph. In International Symposium on Graph Drawing, pages 16–30. Springer, 2001.
- [57] Stich. Apache airflow vs. google cloud dataflow vs. stitch - compare features, pricing, services, and more., Jan 2021.

# Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional.

## Unidad Tamaulipas

### Carta de Aprobación de Protocolo de Tesis

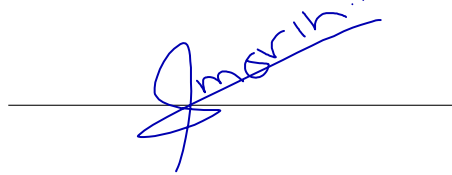
**Método de orquestación para servicios de fusión de datos definidos por  
variables espacio-temporales.**

Fecha de inicio: Septiembre de 2020

Fecha de terminación: Agosto de 2022

Nombre del alumno: José Carlos Morín García

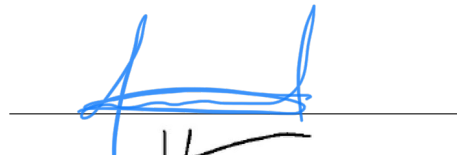
Firma del alumno:



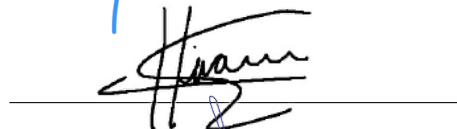
#### Comité de Revisión de Protocolo de Tesis

He leído el protocolo de tesis y estoy de acuerdo con su contenido para el desarrollo de una tesis de maestría.

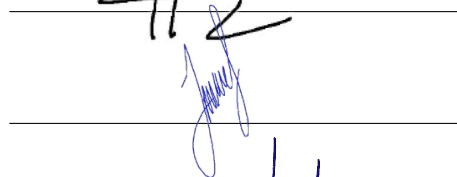
Dr. José Juan García Hernández



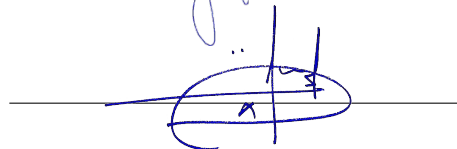
Dr. Hiram Galeana Zapien



Dr. José Luis González Compeán



Dr. Iván López Arévalo



Ciudad Victoria, Tamaulipas, a 19 de agosto de 2021