

# Recuperación de Información para la Web de las Cosas

## Information Retrieval for the Web of Things



**UNIVERSIDAD  
DE GRANADA**

**Héctor Cristyan Manta Caro**

**Director:** Prof. Dr. Juan M. Fernández-Luna

Programa de Doctorado en Tecnologías de la Información y la Comunicación  
Departamento de Ciencias de la Computación e Inteligencia Artificial  
Universidad de Granada

Tesis presentada para optar al grado de:  
*Doctor por la Universidad de Granada*

Granada, España. Marzo 2023

Editor: Universidad de Granada. Tesis Doctorales  
Autor: Hector Cristyan Manta caro  
ISBN: 978-84-1117-858-7  
URI: <https://hdl.handle.net/10481/82082>

A mi familia y a mis seres queridos siempre apoyándome en mis travesías . . .

A mi tutor siempre guiándome durante estos años . . .

## Declaración

El doctorando / *The doctoral candidate* Héctor Cristyan Manta-Caro y los directores de la tesis / *and the thesis supervisor/s*: Prof. Dr. Juan M. Fernández-Luna.

Garantizamos, al firmar esta tesis doctoral, que el trabajo ha sido realizado por el doctorando bajo la dirección de los directores de la tesis y hasta donde nuestro conocimiento alcanza, en la realización del trabajo, se han respetado los derechos de otros autores a ser citados, cuando se han utilizado sus resultados o publicaciones.

*Guarantee, by signing this doctoral thesis, that the work has been done by the doctoral candidate under the direction of the thesis supervisor/s and, as far as our knowledge reaches, in the performance of the work, the rights of other authors to be cited (when their results or publications have been used) have been respected.*

Lugar y Fecha / *Place and date*:

Granada, España. Marzo 2023

Director/es de la Tesis / *Thesis Supervisor/s*:

Prof. Dr. Juan M. Fernández-Luna

Doctorando / *Doctoral Candidate*:

Héctor Cristyan Manta Caro

## Resumen

La Internet y las tecnologías Web han evolucionado notablemente desde su conceptualización, cambiando nuestras vidas y la sociedad en numerosos aspectos. La creación de la Red de Redes y la *World Wide Web* (www) han además posibilitado una base tecnológica crucial para el progreso de la sociedad digital y la construcción de las ciudades inteligentes. Hoy en día, emergen nuevos paradigmas que enmarcan una nueva era, el de la Internet y la Web de las Cosas. Esto se debe a la posibilidad de interconectar a la Internet no solamente dispositivos tradicionales como lo son los teléfonos inteligentes, los ordenadores portátiles, la computación de tipo móvil y ubicua, sino también cualquier objeto en el mundo real. Esta hiperconexión de objetos animados o inanimados en el mundo real ofrece además la posibilidad de publicar servicios de tipo Web sobre estos objetos con la finalidad de proveer contenido altamente dinámico y datos en tiempo real a los usuarios finales acerca del estado o las acciones permitidas sobre estos. En la actualidad, la Internet de las Cosas (IoT) engloba alrededor de 12 mil millones de dispositivos interconectados, y se esperan unos 16.5 mil millones para 2025 y más de 25 mil millones de cosas para el 2030.

El IoT describe tecnologías y disciplinas de investigación que habilitan a la Internet adentrarse en el mundo real de objetos físicos que se interconectan transparentemente y adoptan cierta inteligencia. A través de IoT se obtiene información sobre propiedades, estados y características de las cosas para así accionar sus funcionalidades de forma telemática. La Web de las Cosas (WoT) propone la abstracción de entidades del mundo real en una especie de “*avatar*” Web virtual o “*gemelo*” Web inteligente, con el fin de adquirir, procesar y presentar información en tiempo real en relación con este *avatar* y con la capacidad de conectarse y controlar cosas en el mundo real a través de este. De estos paradigmas complementarios puede surgir una nueva generación de servicios, y muchos de estos también apoyados en la infraestructura de las nuevas redes 5G y 6G. Uno de ellos, crucial para nuestra interacción diaria con este nuevo ciber mundo inteligente, es la Recuperación de Información (RI), principalmente en la forma de motores de búsqueda, que también puede evolucionar hacia herramientas mucho más poderosas. Estos sistemas proporcionarán la capacidad de encontrar información relevante y pertinente sobre las cosas en el mundo real a través de sus abstracciones. Sobre esta base, debe definirse una nueva arquitectura de este tipo de servicios

con la sinergia y los desafíos que ambos IoT y WoT imponen, debido a su colosal tamaño e inaudito dinamismo.

Desde lo fundamental, los simuladores son una herramienta esencial para investigar y desarrollar nuevos sistemas, arquitecturas o protocolos. Los simuladores han jugado un papel clave en la historia, impulsando el desarrollo de Internet y la Web y sus componentes. Este documento destaca la opinión acerca que la investigación basada en la simulación continuará participando en la evolución de los paradigmas asociados con el IoT y WoT, así como de los sistemas de RI. La aparición de tales paradigmas implica una redefinición, reconstrucción hacia la evolución de los sistemas RI actuales, enfrentando así los nuevos desafíos. Por consiguiente, es esencial desarrollar modelos abstractos de representación Web por medio de la simulación para establecer nuevos enfoques en RI para WoT. Y así construir mecanismos de experimentación y validación de estos enfoques por medio de colecciones sintéticas dinámicas que mimeticen el comportamiento WoT por medio de simulación.

Existe aún un desafío latente debido a la heterogeneidad de propuestas que han surgido en el modelado y concepción misma en la arquitectura de WoT, en diferentes niveles, desde el mecanismo de representación Web, el lenguaje de descripción de cosas, hasta el enriquecimiento semántico alrededor de WoT. Todos estos niveles heterogéneos tienen un impacto directo en la construcción de los sistemas RI comúnmente presentados en la forma de motores de búsqueda para IoT o WoT, donde podrían tomarse diferentes direcciones para resolver los problemas de adaptabilidad a WoT para un campo particular de aplicación vía la especialización del sistema RI, su generalización. En cuanto a los principios y arquitectura RI no se evidencia un estudio a profundidad de las técnicas y estrategias de los sistemas RI en su idoneidad y adaptabilidad, dadas las características de dinamismo de la WoT, considerando los requerimientos que impone y los desafíos esperados.

En este trabajo doctoral se describe:

- Una propuesta para modelar WoT basada en una representación XML estructurada. Este modelo ha sido diseñado con flexibilidad y modularidad para permitir la representación de múltiples escenarios, siendo la fuente conceptual para el futuro desarrollo de sistemas RI.
- Un simulador de eventos discretos, nombrado SIM.WoT cuyo objetivo final es encapsular las dinámicas esperadas de la WoT para el desarrollo de sistemas de RI. El simulador genera una colección sintética de documentos XML en tiempo real que contienen contextos espacio-temporales e información textual con dimensiones altamente dinámicas. El simulador se caracteriza por su flexibilidad y versatilidad para representar escenarios del mundo real y ofrece una perspectiva única para la RI.

- Una propuesta de RI para la WoT que contempla las etapas claves de indexación, puntuación y presentación, denominada IR.WoT. Se describen en este trabajo las consideraciones de diseño, implementación en la nube y experimentación con base en una colección de documentos XML sintética a partir de simulación.
- Un estudio de adaptabilidad de los paradigmas y conceptos de la RI convencional al contexto de la WoT en la forma de una Revisión Sistemática de Literatura (RSL) y actualización del estado del arte a 2022. Construcción de un *Dataset* abierto que como resultado de la RSL contiene los datos y análisis de los motores de búsqueda y de los mecanismos de RI para la IoT y WoT en la literatura científica.
- Un reporte de experimentación de las etapas de indexación y recuperación en una propuesta de motor de búsqueda IR.WoT junto con una propuesta de evaluación. Finalizando con un análisis de resultados, recomendaciones con base en los resultados experimentales, conclusiones y trabajo futuro.

---

*English version*

---

The Internet and Web technologies have evolved remarkably since their conception, changing our lives and society in many ways. The creation of the *World Wide Web* (www) and the Network of Networks have also provided a crucial technological base for the progress of digital society and the construction of smart cities. Today, new paradigms are emerging that frame a new era, that of the Internet and the Web of Things. This new era is due to the possibility of interconnecting to the Internet, not only traditional devices such as smartphones, laptops, mobile and ubiquitous computing, but also any object in the real world. Hyperconnection of animate or inanimate objects in the real world also allows publishing Web-type services to provide highly dynamic content and real-time data to end users about the status or actions allowed on them. At the same time, these things (also referred to as objects or entities) are endowed with a certain intelligence. Currently, the Internet of Things encompasses around 12 billion interconnected devices, with an expected 16.5 billion by 2025 and more than 25 billion things by 2030.

The Internet of Things (IoT) describes technologies and research disciplines that enable the Internet to delve into the real world of physical objects, seamlessly connecting and taking on some intelligence. Through IoT, information is obtained on the properties, states and characteristics of things to activate their functionalities telematically. The Web of Things (WoT) proposes the abstraction of real-world entities into a kind of “*avatar*” Virtual Web or intelligent Web Agent in order to acquire, process and present information in real-time concerning this Avatar and with the ability to connect to and control things in the real world

through it. A new generation of services can emerge from these complementary paradigms, many of which are supported by the new infrastructure of 5G and 6G networks. One of them is crucial for our daily interaction with this new intelligent cyber world, Information Retrieval (IR), mainly in the form of search engines, which can also evolve into much more powerful tools. These systems will provide the ability to find relevant information about things in the real world through their abstractions. On this basis, a new architecture for this type of IR service must be defined. It must consider the synergy and challenges both IoT and WoT impose due to their colossal size and unprecedented dynamism.

Simulators are essential for researching and developing new systems, architectures or protocols. Simulators have played a key role in history, driving the development of the Internet and the Web and its components. This paper highlights the view that simulation-based research will continue to participate in the evolution of paradigms associated with the Internet and the Web of Things and information retrieval systems. The appearance of such paradigms implies a redefinition and reconstruction towards the evolution of current IR systems facing new challenges. Therefore, it is essential to develop abstract models of Web representation through simulation to establish new approaches in Information Retrieval for the Web of Things. Moreover, this builds mechanisms for experimentation and validation of these approaches through dynamic synthetic collections that mimic WoT behaviour through simulation.

A latent challenge exists due to the heterogeneity of WoT modelling proposals that have emerged in the conception of WoT architecture. At different levels, from the Web representation mechanism and the thing description language to the semantic enrichment around WoT. All these heterogeneous levels directly impact the construction of IR systems in the form of search engines for WoT. Different directions could be taken to solve the problems of adaptability to WoT for a particular field of application via i) specialization of the IR system or ii) generalization of the IR system. As for the IR principles and architecture, there is no evidence of an in-depth study of the techniques and strategies of the IR systems in their suitability and adaptability given the dynamic characteristics of the WoT, considering the requirements it imposes and the expected challenges.

This doctoral work describes:

- A proposal to model the Web of Things based on a structured XML representation. This model has been designed with flexibility and modularity to allow the representation of multiple scenarios, being the conceptual source for the future development of IR systems.



- A discrete event simulator named SIM.WoT, whose ultimate goal is to encapsulate the expected dynamics of the WoT for the development of IR systems. The simulator generates a synthetic collection of XML documents in real time containing spatiotemporal contexts and textual information with highly dynamic dimensions. The simulator is characterized by its flexibility and versatility to represent real-world scenarios and offers a unique perspective for IR.
- An IR proposal for the WoT that contemplates the critical stages of indexing, scoring and presentation, called IR.WoT. This paper describes design considerations, cloud implementation and experimentation based on a collection of synthetic XML documents from simulation.
- A study of adaptability of conventional IR paradigms and concepts to the WoT context in the form of a Systematic Literature Review (SLR) and update of state of the art to 2022. Construction of an open *Dataset* which, as a result of the SLR, contains the data and analysis of search engines and IR mechanisms for the Internet and the Web of Things in the scientific literature.
- A report of experimentation of the indexing and retrieval stages in an IR.WoT search engine proposal along with an evaluation proposal. We are finishing with an analysis of results, conclusions and future work.

# Índice general

<b>Índice de figuras</b>	<b>XVII</b>
<b>Índice de tablas</b>	<b>XXI</b>
<b>Nomenclatura</b>	<b>XXIII</b>
<b>1. Introducción</b>	<b>1</b>
1.1. De la Internet a la Web de las Cosas . . . . .	1
1.1.1. Definición 1. La Internet de las Cosas . . . . .	1
1.1.2. Definición 2. La Web de las Cosas . . . . .	3
1.1.3. Acerca de los Desafíos en RI para IoT WoT . . . . .	6
1.2. Objetivos del Trabajo Doctoral . . . . .	10
1.3. Contribuciones Originales de la Tesis . . . . .	10
<b>2. Estado del Arte</b>	<b>13</b>
2.1. Fundamentos de Búsqueda en la Web de las Cosas . . . . .	13
2.2. Revisión Sistemática de Literatura en IoTSE WoTSE . . . . .	15
2.3. Pioneros y Evolución de RI para IoT y WoT . . . . .	19
2.4. Taxonomías para Motores de Búsqueda en IoT WoT . . . . .	22
2.4.1. Identificación de Motores de Búsqueda IoTSE WoTSE . . . . .	25
2.4.2. Caracterización de Motores de Búsqueda IoTSE WoTSE . . . . .	25
2.4.3. Clasificación de Motores de Búsqueda IoTSE WoTSE . . . . .	26
2.4.4. Marco de Taxonomía Unificado y Holístico Propuesto . . . . .	28
2.5. Rastreo y Descubrimiento para IoT WoT . . . . .	30
2.6. Indexación, Estructura de Datos y Estrategias para IoT WoT . . . . .	36
2.6.1. Indexación en <i>Casi</i> Tiempo Real y Latencia Cero . . . . .	38
2.7. Lenguajes y Procesamiento de Consulta para IoT WoT . . . . .	40
2.8. Recuperación en IoT WoT . . . . .	42
2.9. Presentación (UI/UX) para IoT WoT . . . . .	44

2.10. Enriquecimiento semántico para IoT WoT . . . . .	45
2.11. Evaluación de Recuperación IoTSE WoTSE . . . . .	46
2.11.1. Evaluación RI Clásica por Colección de Prueba . . . . .	47
2.11.2. Marcos de Evaluación, Datasets y Colecciones de Prueba para IoT WoT	48
2.11.3. Creación de una Colección de Prueba RI-IoT RI-WoT . . . . .	52
2.11.4. Evaluación y Juicios de Relevancia . . . . .	53
<b>3. Simulador de Eventos Discretos para la Web de las Cosas (SIM.WoT)</b>	<b>55</b>
3.1. Arquitectura y Modelado de la Web de las Cosas . . . . .	55
3.1.1. Modelo de Descripción de la Web de las Cosas . . . . .	57
3.1.2. Representación de Web de las Cosas con base en XML . . . . .	61
3.2. Modelo y Arquitectura de Simulación de la Web de las Cosas . . . . .	65
3.2.1. Sub-Sistema de Modelos: Simulación de Entidades . . . . .	66
3.2.2. Sub-Sistema de Generación de Datos y Modelos de Comportamiento	67
3.2.3. Sub-Sistema de Control y Reportes de Simulación . . . . .	67
3.2.4. Código Fuente y Versiones . . . . .	69
3.3. Mecanismo de Simulación de Eventos Discretos . . . . .	69
3.3.1. Evento de Parada de Simulación . . . . .	70
3.3.2. Eventos en la Simulación de la Web de las Cosas . . . . .	71
3.3.3. Consideraciones de Desarrollo, Implementación y Ejecución de SIM.WoT . . . . .	74
3.4. Simulación de WoT: Escenarios Reales, Experimentación y Resultados . . .	78
3.4.1. Escenario A: Sistema Bicing . . . . .	78
3.4.2. Escenario B: Sistema Lübeck . . . . .	79
3.4.3. Escenario C: Sistema de Vivienda Asistida WoT . . . . .	80
3.5. Experimentación y Análisis de Resultados . . . . .	80
<b>4. Motor de Búsqueda para la Web de las Cosas (IR.WoT)</b>	<b>87</b>
4.1. Arquitectura RI para WoT con base en la Nube . . . . .	87
4.1.1. Dimensión I: Tipo de Información IoT WoT . . . . .	88
4.1.2. Dimensión II: Niveles de Abstracción y Alcance de Búsqueda . . .	90
4.1.3. Dimensión III: Computación, Almacenamiento, Visualización y Ana- lítica . . . . .	91
4.2. Arquitectura RI Modular para IoT WoT . . . . .	94
4.3. Consideraciones de Diseño en el Módulo de Consulta y Presentación . . . .	95
4.3.1. Interfaz de Usuario UI de Consulta . . . . .	96
4.3.2. Sub-Componente de Interpretación de Consultas para WoT . . . . .	98

---

4.3.3. Interfaz de Usuario UI de Resultados . . . . .	102
4.4. Consideraciones de Diseño en el Módulo de Indexación y Análisis de Información . . . . .	104
4.4.1. Sub-Componente de Indexación para WoT . . . . .	105
4.5. Consideraciones de Diseño en el Módulo de Puntuación y Recuperación . . . . .	113
4.5.1. Sub-Componente de Recuperación para WoT . . . . .	113
4.6. Consideraciones de Implementación y Despliegue en Nube (IR.WoT) . . . . .	115
4.6.1. Sub-Componente Especial para Integración IR.WoT y SIM.WoT . . . . .	115
<b>5. Experimentación, Evaluación de la Indexación y Búsqueda</b>	<b>119</b>
5.1. Diseño de Experimentos y Procedimientos . . . . .	119
5.1.1. Conjunto de Datos Experimentales: Escenario Simulado WoT . . . . .	120
5.1.2. Conjunto de Experimentos #1: Operaciones en Estructuras de Datos del Índice . . . . .	120
5.1.3. Conjunto de Experimentos #2: Creación del Índice . . . . .	124
5.1.4. Conjunto de Experimentos #3: Tiempo de Respuesta en Consulta . . . . .	126
5.1.5. Conjunto de Experimentos #4: Estudio de Usuarios Finales . . . . .	126
5.2. Reporte de Experimentación y Análisis de Resultados . . . . .	129
<b>6. Conclusiones y Trabajo Futuro</b>	<b>133</b>
6.1. Conclusiones y Consideraciones Finales . . . . .	133
6.2. Trabajo Futuro y Nuevas Líneas de Investigación . . . . .	137
6.2.1. Marcos de Modelado, Simulación y Prototipado para WoT . . . . .	137
6.2.2. Heterogeneidad conceptual en IoTSE WoTSE . . . . .	138
6.2.3. Seguridad, Privacidad y Confianza en RI para IoT WoT . . . . .	139
<b>Bibliografía</b>	<b>143</b>
<b>Apéndice A. Publicaciones</b>	<b>157</b>
<b>Apéndice B. Guía de Usuario para la Evaluación del Sistema IR.WoT</b>	<b>159</b>
B.1. Información acerca del Proyecto de Investigación . . . . .	159
B.2. Instrucciones y Guía de Usuario para la Búsqueda en la Web de las Cosas . . . . .	160



# Índice de figuras

1.1. Modelo de estructura WoT multi-nivel inspirado por Zhong et al. [151] y Sciullo et al. [114] . . . . .	3
2.1. Protocolo RSL para RI-IoT y RI-WoT. . . . .	17
2.2. Evolución de RI para IoT WoT. . . . .	21
3.1. Arquitecturas Conceptuales WoT a) Guinard [40], b) W3C y c) Propuesta. . . . .	57
3.2. Arquitectura Multi-Nivel Abstracta WoT (Propuesta). . . . .	58
3.3. Porcentaje de Estudios Analizados clasificados por Tipo de Modelo de Sensor. . . . .	59
3.4. Porcentaje de Estudios Analizados clasificados por Tipo de Modelo de Cosa. . . . .	59
3.5. Porcentaje de Estudios Analizados clasificados por Centricidad. . . . .	60
3.6. Porcentaje de Estudios Analizados clasificados por Tipo. . . . .	60
3.7. Modelo de Representación XML de WoT y Colección Dinámica de Prueba . . . . .	61
3.8. Representación Gráfica en Árbol del Esquema XSD de Sensor Virtual . . . . .	62
3.9. Representación Gráfica en Árbol del Esquema XSD de Cosa Virtual . . . . .	63
3.10. Representación Gráfica en Árbol del Esquema XSD de Espacio Inteligente . . . . .	64
3.11. Arquitectura Conceptual del Simulador y Descomposición Modular . . . . .	66
3.12. Interfaz Web Principal de SIM.WoT . . . . .	67
3.13. Interfaz Web Configuración de SIM.WoT . . . . .	68
3.14. Componentes macro y lógica del motor de simulación WoT . . . . .	70
3.15. Pre-calendización de eventos de <i>vThings</i> y <i>vSensors</i> . . . . .	72
3.16. Diagrama de Transición con Modelado de Cadena de Márkov para <i>vThing</i> . . . . .	74
3.17. Diagrama de Transición con Modelo de Cadena de Márkov para <i>vSensor</i> . . . . .	75
3.18. Resultados de Consumo de CPU Promedio (%). . . . .	83
3.19. Resultados de Consumo de Memoria (MB). . . . .	84
3.20. Resultados de Velocidad de Transferencia en Almacenamiento (B/s). . . . .	84
3.21. Resultados de Dinámica WoT en Ambiente A . . . . .	85
4.1. Arquitectura de RI como un Servicio (IRaaS) en Nube para IoT WoT . . . . .	89

4.2. Arquitectura WoT Modular y Capas (Propuesta) . . . . .	94
4.3. Arquitectura WoT Modular y sus Componentes (Propuesta) . . . . .	95
4.4. Interfaz de Consulta propuesta para IR.WoT . . . . .	96
4.5. Interfaz de Consulta Captura de Ubicación de Usuario . . . . .	98
4.6. Interfaz de Usuario UI de Resultados . . . . .	103
4.7. Esquema de Indexación propuesta para IR.WoT . . . . .	106
4.8. Estructura de Indexación propuesta para IR.WoT . . . . .	107
4.9. Documento XML simplificado de Prueba - Espacio Inteligente 1. . . . .	111
4.10. Documento XML simplificado de Prueba - Cosa Virtual 1. . . . .	111
4.11. Documento XML simplificado de Prueba - Sensor Virtual 1. . . . .	112
4.12. Ejemplo de Construcción del Índice: Inversión de Estructura. . . . .	113
4.13. Ejemplo de Construcción del Índice: Listas de Publicación. . . . .	114
4.14. Módulo de Integración IR.WoT y SIM.WoT . . . . .	117
5.1. Resultados de Tiempo de Operación de Inserción. . . . .	121
5.2. Resultados de Tiempo de Operación de Borrado. . . . .	122
5.3. Resultados de Tiempo de Operación de Actualización. . . . .	122
5.4. Resultados de Tiempo de Operación de Búsqueda. . . . .	123
5.5. Resultados de Creación del Índice en Tiempo. . . . .	124
5.6. Resultados de Creación del Índice en Tamaño. . . . .	125
5.7. Resultados de Velocidad de Indexado. . . . .	126
5.8. Resultados de Tiempo Promedio de Consulta. . . . .	127
5.9. Evaluación de Usuarios Finales a Dimensiones. . . . .	129
B.1. IR.WoT: Confirmación de Opciones de Ubicación de Usuario. . . . .	162
B.2. IR.WoT: Consulta con Restricción Entidad y Espacial. . . . .	163
B.3. IR.WoT: Respuesta a consulta con Restricción Entidad y Espacial. . . . .	163
B.4. IR.WoT: Verificación vía Postman de Respuesta. . . . .	164
B.5. IR.WoT: Consulta con Restricción Entidad, Espacial. Uso de Contexto de Propiedades. . . . .	164
B.6. IR.WoT: Respuesta a consulta con Restricción Entidad y Espacial. Uso de Contexto de Propiedades . . . . .	165
B.7. IR.WoT: Consulta con Restricción Entidad, Espacial-Temporal. Uso de Con- texto de Propiedades. . . . .	165
B.8. IR.WoT: Respuesta a consulta con Restricción Entidad y Espacial-Temporal. Uso de Contexto de Propiedades . . . . .	166

---

B.9. IR.WoT: Consulta con Restricción Entidad, Espacial-Temporal. Uso de Contexto de Eventos. . . . .	166
---	-----





# Índice de tablas

2.1. Marcos taxonómicos IoTSE WoTSE, dimensiones y enfoques. . . . .	24
2.2. Alcance de Búsqueda para IoTSE WoTSE; Niveles de Abstracción para Representaciones de Cosas. . . . .	31
2.3. Propuesta de Marco Taxonómico para IoTSE WoTSE. . . . .	32
2.4. Análisis de rastreo y descubrimiento para IoT WoT . . . . .	33
2.5. Análisis de Técnicas y Métodos de Indexación para IoT WoT . . . . .	33
2.6. Recopilación de trabajos clave en IoTSE WoTSE 2002-2022. . . . .	50
2.7. Marcos de Evaluación en Sistemas RI para IoT WoT . . . . .	51
2.8. Datasets Abiertos en relación a RI-IoT RI-WoT . . . . .	52
3.1. Tipo de eventos en la simulación WoT . . . . .	71
3.2. Límites y Características del Simulador (SIM.WoT) . . . . .	77
3.3. Ambientes de Experimentación usados con SIM.WoT . . . . .	81
3.4. Resultados de Experimentación con SIM.WoT: Escenarios A, B y C . . . . .	82
3.5. Resultados de Experimentación con SIM.WoT: Ambiente B . . . . .	82
3.6. Resultados de Experimentación con SIM.WoT: Ambiente C . . . . .	83
4.1. Límites y Características del Esquema de Indexación . . . . .	109
4.2. Análisis de Complejidad de Estructuras de Datos . . . . .	110
5.1. Colección XML RI Experimental . . . . .	121
5.2. Radio entre el Tamaño del Índice y el Tamaño Original de la Colección. . . . .	125
6.1. Temas de investigación a corto y largo plazo para RI-IoT e RI-WoT. . . . .	141



# Nomenclatura

## Acronyms / Abbreviations

API	Interfaz de Programación de Aplicaciones / <i>Application Programming Interface</i>
FEL	Lista de Eventos Futuros / <i>Future Event List</i>
HLD	Diseño de Alto Nivel / <i>High-Level Design</i>
IEEE	Instituto de Ingenieros Eléctricos y Electrónicos / <i>Institute of Electrical and Electronics Engineers</i>
IETF	Grupo de Trabajo en Ingeniería de Internet / <i>Internet Engineering Task Forces</i>
IoT	Internet de las Cosas / <i>Internet of Things</i>
IRaaS	Recuperación de Información como un Servicio / <i>Information Retrieval as a Service</i>
IRI	Identificador de Recurso Internacionalizado / <i>Internationalised Resource Identifier</i>
LLD	Diseño de Bajo Nivel / <i>Low-Level Design</i>
MAP	Precisión Promedio Media / <i>Mean Average Precision</i>
PBF	Marco de Unión de Protocolo / <i>Protocol Binding Framework</i>
RI	Recuperación de Información / <i>Information Retrieval</i>
SED	Simulación por Eventos Discretos / <i>Discrete-event Simulation</i>
SLR	Revisión Sistemática de Literatura / <i>Systematic Literature Review</i>
TDL	Lenguaje de Descripción de Cosas / <i>Thing Description Language</i>
TD	Descripción de Cosas / <i>Thing Description</i>
UI	Interfaz de Usuario / <i>User Interface</i>

UN Naciones Unidas / *United Nations*

W3C Consorcio WWW / *World Wide Web Consortium*

WoO Web de los Objetos / *Web of Objects*

WoS Web de Sensores / *Sensor Web or Web of Sensors*

WoT Web de las Cosas / *Web of Things*

XML Lenguaje de Etiquetado Extensible / *eXtensible Markup Language*

# Capítulo 1

## Introducción

### 1.1. De la Internet a la Web de las Cosas

En la actualidad, la cantidad de cosas conectadas a la Internet excede a la población mundial, que según datos de las Naciones Unidas llegó a los 8 mil millones de personas el 15 de noviembre de 2022 en su informe de Perspectivas Poblacionales<sup>1</sup>. Se estima que, en el año 2022, existieron más de 12 mil millones de dispositivos activos conectados y que serán unos 13 mil millones al cierre del año<sup>2</sup>, esto según reportan los portales especializados en estadísticas globales.

Si tenemos en cuenta que tan sólo el uno por ciento (1 %) de los objetos del mundo real están conectados a la Internet y que se espera que la cantidad de dispositivos conectados llegue a más de veinticinco mil millones para el 2030, las posibilidades de funcionalidades y servicios son inimaginables. Este nuevo paradigma se conoce como la Internet de las Cosas, *Internet of Things* (IoT), este comprende las tecnologías y las disciplinas de investigación que permiten a la red adoptar cierta inteligencia y adentrarse en el mundo real de los objetos físicos que están ahora están interconectados [32], y comparten datos entre sí.

#### 1.1.1. Definición 1. La Internet de las Cosas

IoT tiene múltiples definiciones, ya que este término identifica una colección de tecnologías que extienden la conectividad de Internet a objetos cotidianos y les permiten capacidades informáticas. IoT agrupa y conecta objetos físicos, o cosas, principalmente equipados con sensores y actuadores para adquirir datos en tiempo real sobre el objeto en sí, su estado y el entorno que lo rodea.

---

<sup>1</sup>Reporte UN - <https://www.un.org/es/desa-es/la-población-mundial-llegará-8000-millones-en-2022>

<sup>2</sup>Estadísticas para IoT en 2022 - <https://dataprot.net/statistics/iot-statistics/>

La Unión Internacional de Telecomunicaciones (*International Telecommunications Union*, ITU por sus siglas en inglés) lo ha definido como:

*“Una infraestructura global para la sociedad de la información, que permite servicios avanzados mediante la interconexión de cosas (físicas y virtuales) basadas en tecnologías de información y comunicación interoperables existentes y en evolución”* - Recomendación UIT-T Y.4000/Y .2020 (2012)<sup>3</sup>.

Según el Grupo de Trabajo en Ingeniería de Internet (*Internet Engineering Task Force*, IETF por sus siglas en inglés):

*“El concepto de IoT se refiere al uso de protocolos estándar de Internet para permitir la comunicación de persona a cosa y de cosa a cosa”. Por lo tanto, “IoT es una infraestructura de red de objetos físicos, denominados cosas”, integrados con inteligencia (electrónica, software, sensores, actuadores y conectividad) para permitir que las cosas intercambien datos de humano a máquina (H2M), de máquina a máquina (M2M).”* - IETF <sup>4</sup>

La iniciativa IoT del Instituto de Ingenieros Eléctricos y Electrónicos (IEEE por sus siglas en inglés) proporciona una discusión sobre una definición integral de IoT que abarca desde pequeños sistemas localizados restringidos a una ubicación específica hasta un sistema global extenso que se distribuye y se compone de sistemas complejos [84]. Esto desde una perspectiva de la estandarización.

Desde una perspectiva de categorización, IoT y WoT se han analizado con respecto a su enfoque, orientación y aplicabilidad. Pattar et al. [96] propone una categorización inicial con base en en datos o en objetos, reflejada también en los motores de búsqueda para IoT y WoT (*IoT Search Engine*, *WoT Search Engine*, *IoTSE/WoTSE* de sus siglas en inglés). IoT se puede considerar desde tres perspectivas: 1) enfoques orientados a datos para proporcionar flujos sin procesar y/o pre-, post-procesados en tiempo real sobre el mundo; 2) enfoques orientados a los protocolos, centrados en proporcionar piezas de comunicación funcionales entre las *Cosas* y las aplicaciones en la nube o IoTIWoT; 3) enfoques orientados en objetos, centrados en las cosas mismas que permiten la federación o estructuras más complejas del IoT.

La perspectiva con base en flujos de datos está orientada a proporcionar mecanismos únicos para manejar las características de velocidad, volumen y variedad de los datos en sí. Como ejemplo, se puede mencionar un predecesor IoT, el *Internet of Data* el cual se

<sup>3</sup>ITU-T Y.4000/Y.2020 - <https://www.itu.int/ITU-T/recommendations/rec.aspx?rec=y.2060>

<sup>4</sup>IoT en IETF <https://www.ietf.org/topics/iot/>

enfoca en proveer distintos tipos de datos en tiempo real y en tiempo cuasi-real, así como en permitir la transmisión de estos. Sin embargo, este carece del concepto de *cosa*. Se puede ampliar para crear interrelaciones y agregar semántica para convertirse en el enfoque de datos vinculados (ó *Linked Data* en inglés). Podríamos clasificar las diferentes soluciones IoT desde una concepción arquitectónica, dados los protocolos utilizados para las comunicaciones de nube a cosas. Algunas soluciones reutilizan tecnologías Web conectando cosas a través del protocolo HTTP convencional pero sin ningún tipo abstracción agregado. Por otra parte, el transporte de telemetría MQ (MQTT) se está convirtiendo en uno de los protocolos más utilizados, seguido del Protocolo avanzado de cola de mensajes (AMQP). La perspectiva orientada a objetos abarca enfoques más orientados a cosas, es decir, centrando el paradigma en la cosa en lugar de los datos crudos, con múltiples interpretaciones de lo que se puede considerar un objeto.

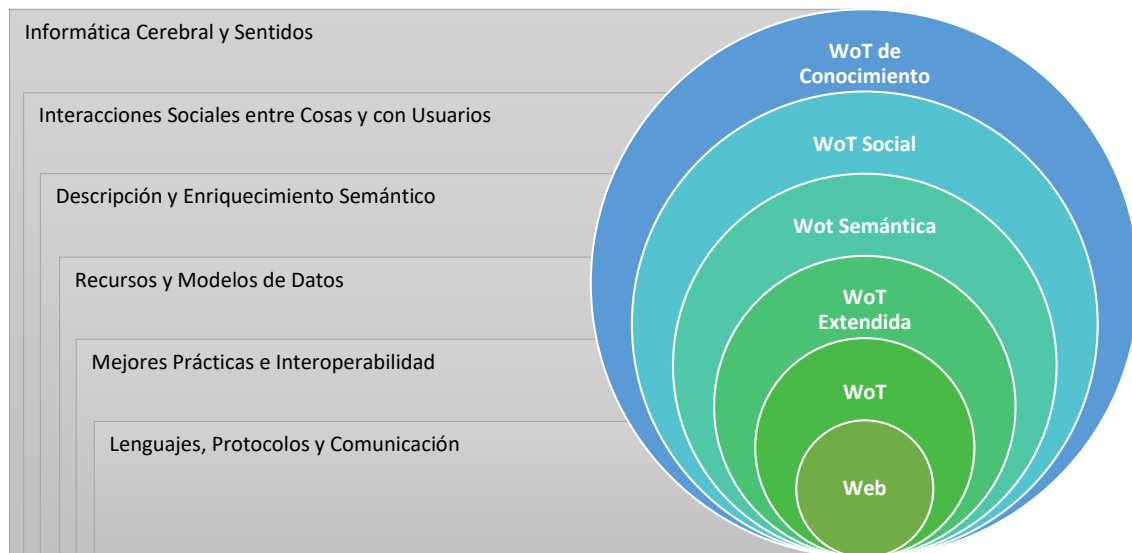


Figura 1.1 Modelo de estructura WoT multi-nivel inspirado por Zhong et al. [151] y Sciallo et al. [114]

### 1.1.2. Definición 2. La Web de las Cosas

El término *Web of Things* fue utilizado por primera vez en 2007 por David Raggett en la Conferencia de Desarrolladores Web de UWE en Bristol (Reino Unido) según varios medios especializados en la materia <sup>5</sup>, <sup>6</sup>. Los nuevos paradigmas de IoT y WoT están estrechamente

<sup>5</sup>WoT - <https://www.w3.org/People/Raggett/>

<sup>6</sup>WoT - <https://webofthings.org/>



relacionados y ambos imponen nuevos requisitos y limitaciones. Algunas investigaciones se han centrado en la arquitectura de WoT [40], [134] y las aplicaciones para integrar el mundo real en la Web. Se ha propuesto que la arquitectura WoT utilice principios RESTful como una forma de seguir los pilares de la escalabilidad y de la modularidad de la Web tradicional [51]. La WoT es vista como el uso de tecnologías Web para permitir el acceso a cosas conectadas a la IoT a través de una representación abstracta y virtual, recientemente también denominado *Avatar Web* [88]. En las primeras etapas de WoT, la Web de Sensores (WoS) permitió recopilar datos en tiempo real de cosas y variables físicas del mundo. A medida que el paradigma WoT evoluciona, las capas de abstracción se han vuelto más complejas y se han fusionado más tecnologías. Detrás del WoT existe una gran cantidad de tecnologías Web que aprovechan su evolución, como el protocolo CoAP, HTTP, Servicios Web WS-\*, API Restful, entre otros.

Es decir, si habilitamos acceso Web avanzado a través de elementos virtuales, que son representaciones abstractas de cosas en el mundo real, podemos crear espacios inteligentes que aparecen como el nuevo paradigma WoT [14]. La WoT puede referirse no sólo a aquellas partes de la Web que comprenden tecnologías Web especiales, sino también de sistemas de aplicaciones conectados al mundo real a través de sensores y actuadores [21], y a la combinación o evolución de paradigmas como la Web semántica, la Web Ubicua, y la Web 3.0 término acuñado para la centricidad en los datos y las maquinas. Recientemente, Sciullo et al. afirman que:

*La Web de las Cosas (WoT) denota una amplia gama de soluciones destinadas a lograr la interoperabilidad en escenarios de IoT gracias al poder unificador de tecnologías Web" [114].*

Y añaden, que la evolución de WoT ha sido compleja y no lineal [114], haciendo un repaso exhaustivo por su evolución partiendo desde la IoT y pasando por la concepción WoT original, la WoT Social y la Web Semántica y hasta abordar la WoT estándar. Es así como el paradigma WoT y su modelamiento siguen evolucionando. Es importante adicionar, que también se han introducido nuevos niveles de abstracción en el modelo WoT, que van desde la concepción WoT original, en nuestra notación WoT puro (con centricidad única en las cosas), pasando por un modelo WoT extendido (donde se incorporan recursos y modelos de datos), hasta el WoT semántico (donde se incorporan la descripción, anotaciones semánticas y se incorporan diferentes tipos de ontologías) [119]. En [151] se ha introducido un nivel más allá, el Wisdom WoT (W2T) la Web de las Cosas del Conocimiento, cuya idea central es realizar una inteligencia colectiva a nivel humano basada en tecnologías inteligentes inspiradas en el cerebro para proporcionar servicios de sabiduría en el W2T, como una gran mente de

conciencia de intercambio de datos. En la Figura 1.1 representamos la estructura en los niveles del modelo WoT.

La WoT proporciona una capa de abstracción y un modelo de descripción para el objeto en sí, lo que permite reutilizar y potenciar las tecnologías Web y crear una contraparte virtual del objeto físico. En términos de objetivo:

*“El objetivo final de WoT es permitir que las computadoras vean, escuchen y sientan el mundo real”. - World Wide Web Consortium (W3C).*

Desde el punto de vista de la estandarización:

*“WoT está destinado a permitir la interoperabilidad entre plataformas IoT y dominios de aplicaciones” - Recomendación W3C<sup>7</sup>.*

Principalmente con el objetivo de simplificar el uso de dispositivos IoT, la ingesta y conversión de datos, interrelaciones y conexiones locales y permitir la creación de nuevos servicios avanzados en forma de mezclas Web. Si bien WoT e IoT tienen como objetivo principal conectar cosas a través de Internet, existen algunas diferencias técnicas críticas entre ellos. Estas diferencias se definen principalmente por el propósito al que sirve cada uno, la abstracción de cosas que proporcionan y los matices en las implementaciones que implican.

Técnicamente hablando, desde su concepción, WoT se ha basado en tres elementos fundamentales <sup>8</sup>:

1. Un identificador de recurso internacionalizado (*Internationalised Resource Identifier*, IRI por sus siglas en inglés). Una identificación universal de alto nivel de las cosas es un requisito ampliamente utilizado y aceptado. Todo en WoT tiene que ser un recurso (URI/URL) con soporte para los mecanismos de descubrimiento.
2. Un marco de unión de protocolo (*Protocol Binding Framework*, PBF por sus siglas en inglés). La lógica del servicio de WoT está desacoplada de la comunicación subyacente, que se basa en enlaces a secuencias de comandos (API) y protocolos de enlace de datos para interactuar con las cosas en la red. La transferencia de estado representacional (REST) se ha utilizado en la mayoría de los enfoques y con otros protocolos en diversos modelos de comunicación.
3. Un lenguaje de descripción de cosas (*Thing Description Language*, TDL por sus siglas en inglés). El corazón de la WoT es un modelo de descripción de cosas. Desde el

<sup>7</sup>Arquitectura WoT - <https://www.w3.org/TR/architectura-wot/>

<sup>8</sup>Introducción al Framework WoT - <https://www.w3.org/2015/03/intro-wot.pdf>

punto de vista arquitectónico, Trifa et al. [139] ha propuesto un modelo de objeto Web dentro del W3C, basado en el trabajo de [40] y [134], para describir la contraparte virtual de los objetos físicos en WoT. Inspirándose en este trabajo, el W3C propuso una recomendación arquitectónica que aprovecha el trabajo de Trifa y Guinard. Además, Faheem et al. [30] exponen la necesidad de crear modelos estandarizados de descripción de cosas, denominados esquemas de cosas, en su convención.

En algunos estudios, el término avatar de WoT se ha acuñado [88] como *‘una abstracción general tanto para cosas individuales como para un grupo de cosas que definen lógicamente un concepto único’* mientras que en [58] los avatares se definen como entidades virtuales de objetos IoT heterogéneos. Los avatares Web facilitan la creación de aplicaciones y servicios que combinan información de diferentes fuentes y diferentes niveles de abstracción, dotados de capacidades de razonamiento para facilitar su coordinación [58]. Existen varios enfoques para proporcionar descripción de WoT y capacidades de enriquecimiento semántico. Por ejemplo, metadatos, microformatos<sup>9</sup>, descripciones de cosas basadas en XML, JSON o a través de nuevas modificaciones de HTML y ontologías Web [10], u otros estilos arquitectónicos.

### 1.1.3. Acerca de los Desafíos en RI para IoT|WoT

En la actualidad, los paradigmas IoT y WoT crean nuevos horizontes de investigación y desarrollo, pero al tiempo también crean nuevos desafíos a los sistemas con los que interactúan, incluidos los sistemas RI y los motores de búsqueda sobre estos paradigmas. Múltiples desafíos han sido documentados en la literatura durante los últimos años; algunos persisten, mientras que otros, a lo largo del tiempo, han sido mitigados por soluciones novedosas. Dentro de este contexto RI [30], los nuevos paradigmas WoT introducen factores altamente dinámicos que deben considerarse cuidadosamente, esto debido a que la WoT abstrae una gran cantidad de objetos del mundo real produciendo continuamente una gran cantidad de datos e información. Luego, su representación Web incorporará información de estado o variables críticas de interés que deben actualizarse en tiempo real y con frecuentes cambios de estado, lo que genera fuentes de datos e información de alto dinamismo y extensión.

Hasta ahora, sin embargo, ninguna investigación se ha centrado en el desarrollo de un simulador escalable de WoT. El dinamismo esperado del WoT implica miles de millones de cosas del mundo real que producirán constantemente una gran cantidad de información. Así mismo, el crecimiento exponencial en el número de sensores conectados a las cosas obtiene

<sup>9</sup>Formato de metadatos para contenido Web - <http://microformatos.org/>

una gran cantidad de datos que cambian frecuentemente debido a la naturaleza del IoT. El diseño, desarrollo, implementación y adaptación de nuevos motores de búsqueda como instancia de los sistemas RI que permiten encontrar cosas y proporcionar información sobre las variables de estas cosas (así como las características y servicios que ofrecen) representan un tema de investigación crucial [14], [45] dado el nuevo entorno WoT.

Este fuerte dinamismo no ha sido bien explorado o evaluado en sistemas RI convencionales. El WoT puede caracterizarse por una cantidad colosal de documentos en una colección altamente dinámica. Con el fin de abordar los nuevos requisitos y restricciones impuestos por el WoT en los sistemas RI, se requiere una recopilación de pruebas para fines de evaluación. Por el momento, sin embargo, no hay evidencia de que tal colección exista. Por lo tanto, una de las motivaciones de este trabajo de investigación es crear una colección sintética de pruebas de RI en tiempo real que imite el comportamiento complejo del WoT para evaluar aún más los sistemas RI. En consecuencia, la mejor forma de generar esta colección sintética es simular la dinámica IoT|WoT. La IoT|WoT cambiará por completo la forma en que interactuamos con el mundo y nuestro simulador se puede usar para mostrar escenarios reales que conducen a este ecosistema futuro con un enfoque flexible y versátil al tomar en cuenta fundamentos y perspectivas de recuperación de información. Estas colecciones de pruebas RI son empleadas para proponer nuevas líneas de investigación en el campo de RI evaluando primero diferentes estrategias de indexación XML, técnicas en tiempo real y las estructuras de datos más adecuadas. Esto último, en el marco de una propuesta RI para WoT, en forma de motor de búsqueda, que permite ampliar nuestro conocimiento acerca de la adaptabilidad, pertinencia y efectividad de los mecanismos RI actuales existentes.

### **Dinamicidad RI y Dinámica IoT|WoT**

El mayor desafío arquitectónico es el alto ritmo de generación de datos de IoT|WoT y de los cambios en los estados de las cosas y los datos proporcionados por los sensores. Recientemente, Faheem et al. [30] presenta la búsqueda dinámica como el principal desafío actual que enfrentan los sistemas IoTSE/WoTSE. Debido a eso, la mayoría de los enfoques de WoTSE se basan solo en la búsqueda con base en palabras clave o en la búsqueda de ubicaciones estáticas. De esa manera, insta a la construcción de mecanismos de indexación dinámica y la consideración de técnicas de recuperación con base en la intención de búsqueda del usuario, que pueden impactar positivamente la dinámica, adaptabilidad y escalabilidad de los sistemas WoTSE. Por otra parte, Aziez et al. [79] defiende la idea de que la gestión de la dinámica WoT es fundamental para encontrar el servicio relevante que cumpla con los requisitos del usuario final en tiempo real. Mientras que Ma-Liu [73] expresa el dinamismo de la búsqueda como resultados oportunos garantizados en términos de frescura. Tran et al.

[132] concuerda con Faheem et al. [30] y señalan que los problemas de investigación aún abiertos sobre la búsqueda dinámica en IoT|WoT resulta un problema de compensación entre el proceso de indexación y la frescura de los resultados.

### **Adaptabilidad de RI**

Se refiere a la capacidad de un sistema IoTSE|WoTSE para adaptar su comportamiento a diferentes escenarios de IoT|WoT, incluida la modificación de los componentes internos de RI para cumplir con las demandas de IoT|WoT. En investigaciones recientes, Aziez et al. [79], [1] abordan el problema del descubrimiento de IoT, y destacan la adaptabilidad como un desafío existente porque la mayoría de los enfoques no abordan los requisitos generalizados de IoT que deben cumplir las técnicas clásicas de RI. Tran et al. [132] amplía esta discusión argumentando que cada etapa del proceso de RI para un WoTSE ideal requerirá algún tipo de adaptación dadas las limitaciones de recursos en las cosas propiamente dichas, y las nuevas características en IoT|WoT en comparación con la Web tradicional.

### **Escalabilidad de RI**

Este desafío se refiere a la necesidad de ajustar los recursos informáticos de IoTSE|WoTSE para manejar la cantidad colosal de cosas y datos producidos por estas. [96], [146] ya han identificado que el diseño arquitectónico de IoTSE|WoTSE debe adaptarse con nuevas soluciones. Técnicamente hablando, la escalabilidad se considera la capacidad de un sistema relacionado con IoT|WoT para adaptarse a los cambios en el entorno del mundo real y satisfacer las necesidades cambiantes en el futuro. Un IoTSE|WoTSE debe tener la capacidad de manejar la creciente cantidad de carga de trabajo en términos de capacidades de procesamiento, almacenamiento y comunicaciones.

### **Interoperabilidad entre sistemas IoTSE|WoTSE e IoT|WoT**

La capacidad de IoT|WoT y los sistemas IoTSE|WoTSE de intercambiar información con otros sistemas, ser modular sin dependencias entre capas facilitando la innovación y evolución es impactada por la falta de estandarización en múltiples capas. Además, la falta de interoperabilidad [92] es un paraguas de problemas que ralentiza la evolución y evita la aparición de nuevos sistemas IoTSE|WoTSE. La interoperabilidad está presente en diferentes niveles: red, dispositivo, sintáctico, semántico y plataforma. Algunos desafíos abiertos impregnan la creación de aplicaciones compuestas entre dominios y entre plataformas. Es obligatorio madurar la estandarización del protocolo para las comunicaciones de dispositivo a dispositivo, la apertura de los enfoques API y la unificación de los marcos de prueba

y evaluación. Al final, la falta de interoperabilidad en la capa IoT/WoT indudablemente también afecta la evolución de IoTSE/WoTSE. La interoperabilidad sigue siendo un tema de investigación abierto [119]. Si bien la semántica promete resolver los problemas de interoperabilidad de múltiples enfoques, protocolos, integración y aplicaciones, implica definir el rol de cada entidad y elemento en IoT|WoT. Sin consenso sobre las ontologías y la representación del conocimiento, la ventaja de la interoperabilidad es borrosa y confusa [119].

Ahora es el momento de centrar la investigación RI en los servicios avanzados habilitados por los paradigmas IoT y WoT, que pueden cambiar nuestra perspectiva y construir nuevas formas de consumir información. Entre estos servicios avanzados en forma de aplicaciones, la RI es y será una de las formas esperadas y más valiosas de aplicaciones y herramientas abiertas basadas en WoT en el mundo cibernético, que pueden impulsar el paradigma completo de *ciudades inteligentes* proporcionando nuevas funcionalidades de búsqueda cibernética. La próxima generación de RI se considera una disciplina integral que agrega distintos aspectos de las herramientas teóricas modernas, como el aprendizaje de máquina, el análisis de inferencia causal y la toma de decisiones interactiva [141]. Se cree que es hoy en día la forma dominante de acceso a la información, generalmente de una naturaleza no estructurada, como es el texto. Sin embargo, la RI moderna ahora considera todo tipo de información, desde datos, texto, imágenes, sonidos y música hasta vídeos y transmisiones en vivo.

Los sistemas RI se convierten en herramientas esenciales dado que IoT y WoT ahora involucran fuentes de información dispares, que incluyen, entre otros, información de sensores representable por imágenes 2D/3D (geo-espaciales, infrarrojas, UV, imágenes médicas), por sonidos (acústicos, de sonar, infrasonidos/ultrasonidos), vídeo (vigilancia, medios y entretenimiento), y en general cualquier información multidimensional. En el contexto de las aplicaciones de RI para IoT|WoT. Tanto IoTSE como WoTSE han sido identificados como uno de los 10 principales temas de investigación en el espectro de IoT [149].

Como se ha mencionado anteriormente, los sistemas RI tradicionales no han sido probados ni evaluados en estos entornos altamente dinámicos y ricos, y se deben considerar nuevos riesgos y desafíos. Debido a las características notables de IoT y WoT, la comunidad de investigación en RI debe repensar los sistemas RI convencionales, el alcance, la arquitectura y las etapas internas. Este trabajo doctoral enfoca sus esfuerzos en atender algunos de estos desafíos y riesgos.

## 1.2. Objetivos del Trabajo Doctoral

En la investigación de esta tesis doctoral se ha planteado como objetivo general el desarrollar un sistema de RI para la WoT adaptado a las características especiales de este entorno tecnológico.

Y como objetivos específicos se han definido los siguientes:

- Estudiar los paradigmas y conceptos de la RI convencional, su idoneidad y su adaptabilidad para el contexto de la WoT.
- Diseñar y desarrollar un modelo y un simulador para la WoT que sirva como base para la investigación en el campo de la RI.
- Adaptar los algoritmos, métodos y paradigmas de la RI convencional y al contexto de la WoT, caracterizado por un alto dinamismo en la colección de documentos.
- Diseñar y desarrollar un sistema de RI para la WoT considerando las limitaciones y ventajas de los sistemas convencionales.
- Determinar los métodos de evaluación de desempeño en términos de eficiencia de los componentes principales de un sistema de RI para la WoT.

## 1.3. Contribuciones Originales de la Tesis

Son contribuciones originales de la investigación de esta tesis doctoral:

- Construcción de un estudio de adaptabilidad de los paradigmas y conceptos de la RI convencional al contexto de la WoT en la forma de una Revisión Sistemática de Literatura (RSL) y actualización del estado del arte a 2022. Construcción de un Dataset como resultado de la RSL de buscadores y mecanismos de RI para la Internet y la Web de las Cosas. Y una propuesta de taxonomía para los motores de búsqueda en WoT.
- Modelo de WoT con base en documentos XML y Simulador de eventos discretos para WoT desplegados en ambiente de nube para la realización de experimentos en RI. Ajuste de los mecanismos de simulación de eventos discretos para mimetizar el comportamiento WoT. Implementación del Simulador (SIM.WoT) describiendo las consideraciones de diseño, implementación y despliegue y su evaluación de desempeño.
- Adaptación e implementación de estructuras de datos y estrategias de indexación y recuperación en una arquitectura modular de RI para la experimentación usando la

generación automática de una colección WoT-RI sintética, empleando como base el algoritmo BM25 para la ponderación y reordenamiento RI de colección XML dinámica con base en relevancia.

- Diseño y desarrollo de un sistema RI, denominado (IR.WoT). Este considera las adaptaciones de sistemas convencionales con enfoque en la indexación RI con base en colecciones XML, empleando como base el algoritmo BM25. Integración del sistema IR.WoT y el simulador SIM.WoT. Adaptación del sistema de presentación de resultados. Construcción de métodos para la experimentación de variantes de indexación y recuperación en WoT.
- Diseño de experimentos para determinar el desempeño funcional y no funcional del sistema RI. Se compone de una serie batería de experimentos que fueron ejecutados para determinar el desempeño de diferentes esquemas de indexado en tiempo y espacio, así como el comportamiento de tiempo del sistema completo en términos de tiempo de respuesta ante diferentes tamaños de consulta y una evaluación desde la perspectiva del usuario final del sistema RI.

Los siguientes capítulos de esta tesis doctoral se estructuran de la siguiente forma:

El Capítulo 2 presenta el estado del arte con relación a la búsqueda en la WoT, partiendo de trabajos existentes, en especial de las taxonomías ya existentes. Contiene un estudio de adaptabilidad de los paradigmas y conceptos de la RI convencional al contexto de la WoT en la forma de una RSL y actualización del estado del arte a 2022. Se presenta un *Dataset* abierto que como resultado de la RLS contiene los datos y análisis de los motores de búsqueda y de los mecanismos de RI para la Internet y la Web de las Cosas en la literatura científica. Este recapitula la RSL, abordando cada una de las etapas en la recuperación de información para WoT. En este se presenta nuestra propuesta de taxonomía para motores de búsqueda para IoT|WoT. Y se discute un marco de evaluación a emplear para este tipo de sistemas RI.

Nuestra propuesta de simulación por eventos discretos para la WoT se describe en detalle en el Capítulo 3. Este capítulo presenta una propuesta para modelar la Web de las Cosas basada en una representación XML estructurada. Este modelo ha sido diseñado con flexibilidad y modularidad para permitir la representación de múltiples escenarios, siendo la fuente conceptual para el futuro desarrollo de sistemas RI. Del mismo modo, presenta un simulador de eventos discretos, nombrado SIM.WoT cuyo objetivo final es encapsular las dinámicas esperadas de la WoT para el desarrollo de sistemas de RI. El simulador genera una colección sintética de documentos XML en tiempo real que contienen contextos espacio-temporales e información textual con dimensiones altamente dinámicas. Se discuten las consideraciones de desarrollo e implementación. Por último, este capítulo presenta



la evaluación de desempeño del simulador en términos de recursos de procesamiento y almacenamiento.

El Capítulo 4 comienza con la discusión de alternativas arquitectónicas para sistemas RI para WoT con base en la nube. El capítulo describe propuesta de RI para la WoT que contempla las etapas claves de indexación, puntuación y presentación, denominada IR.WoT. En esta se incluye una descripción de la arquitectura modular del motor de búsqueda IR.WoT propuesto, posteriormente se discuten las consideraciones de diseño en los diferentes módulos que lo componen. Se realiza un especial énfasis en la investigación y análisis de mecanismos de indexación, estructura de datos y recuperación. El capítulo culmina con la presentación de la arquitectura RI implementada y sus consideraciones de despliegue e implementación y su interacción con base en una colección de documentos XML sintética a partir de simulación.

El diseño de experimentos y los procedimientos ejecutados se describe en el Capítulo 5. Este además contiene el reporte de experimentación y el análisis de los resultados. Un reporte de experimentación para la evaluación de desempeño de las etapas de indexación y recuperación en una propuesta de motor de búsqueda IR.WoT, se presentan diferentes baterías de experimentos para determinar el desempeño en términos de respuesta en tiempo y comportamiento espacio de almacenamiento. Incluye, una evaluación metodológica mediante uso desde la perspectiva de usuario final para determinar grados de satisfacción en la obtención de resultados relevantes a búsquedas sobre escenarios simulados WoT.

El Capítulo 6 concluye y presenta las consideraciones finales de este trabajo doctoral. Como complemento del capítulo, este contiene una discusión de las alternativas de trabajo futuro y de nuevas líneas de investigación en área de la recuperación de información para WoT.

# Capítulo 2

## Estado del Arte

### 2.1. Fundamentos de Búsqueda en la Web de las Cosas

Los sistemas RI en forma de motor de búsqueda son una de las formas esperadas de aplicaciones abiertas sobre los paradigmas IoT|WoT. A la vez que potencian nuestra interacción con sistemas inteligentes y nuevos enfoques cibernéticos en construcción. De esta manera, los sistemas RI pueden brindar información relevante acerca de los servicios y funcionalidades suministrados en las ciudades inteligentes a través de una visión aumentada del mundo.

*En general, la RI busca información relevante de naturaleza no estructurada, comúnmente texto dentro grandes colecciones de documentos, que satisface una necesidad de información dentro de grandes colecciones. El término "datos no estructurados, se refiere a datos que no tienen una estructura clara, semánticamente abierta y fácil de usar para una computadora". - Manning et al. (2008) [74].*

Una característica que distingue a la RI de otras disciplinas como la visión artificial, la minería de datos o el procesamiento del lenguaje natural es su interacción directa con los usuarios. Se requieren de decisiones complejas para satisfacer las necesidades de información de un usuario, no sólo la estimación de relevancia del contenido o la caracterización del contexto.

Los sistemas RI tradicionales no se han probado ni evaluado en los entornos altamente dinámicos de WoT, y es necesario considerar estos nuevos desafíos. Debido a las características especiales de WoT, se deben repensar los sistemas RI convencionales, no solo en su alcance, sino también en su arquitectura. Una forma de enfrentar los impactos de WoT es

utilizar los recursos de la nube y construir un paradigma de RI como un servicio (*Information Retrieval as a Service*, IRaaS por sus siglas en inglés) para desarrollar nuevos sistemas RI de próxima generación. Una característica importante de WoT es que, como consecuencia de la abstracción y visualización de conceptos del mundo real, WoT no se limita a dispositivos conectados, a diferencia de IoT. WoT también puede incluir cosas que no son ni pueden conectarse como personas y lugares, ideas abstractas, así como eventos y períodos de tiempo [58], de este modo estas entidades o cosas pueden clasificarse como tangibles e intangibles.

En adición, es evidente que la gran cantidad de datos que se recopilan a partir de múltiples fuentes de información en IoT y otros campos ha desafiado los campos de investigación en procesamiento de la información hasta la creación del concepto de *Big Data*. Los grandes datos se pueden ver como un fenómeno o efecto que puede ser producido por las tecnologías relacionadas con IoT|WoT. Particularmente hablando, WoT tiene la capacidad de producir datos de miles de millones de fuentes que son de velocidad sorprendente, volumen, volatilidad y variedad. Es así como, los servicios basados en la nube han impulsado el desarrollo de nuevas herramientas para procesar, analizar, almacenar y visualizar el *Big Data*. Una arquitectura de tipo RI puede aprovechar los recursos de la nube, mientras que distribuye algunas funcionalidades en los nodos de datos IoT que comúnmente tienen recursos de procesamiento limitados [124].

Los motores de búsqueda IoTSE|WoTSE se describen como:

*"Sistemas que permiten a los humanos y las máquinas recuperar el contenido relevante del IoT, como datos sensoriales, la existencia de servicios accionantes de funcionalidades y representaciones digitales de entidades físicas"* - [132], [131].

En la mayoría de los casos, las representaciones digitales se devuelven como resultado de una búsqueda de geo-localización por medio de los dispositivos adjuntos a las cosas. En una de las primeras definiciones, se pensaba que los motores de búsqueda WoTSE estaban reservados a *"buscar el estado en tiempo real de entidades del mundo real"* [112]. En algunos trabajos, los términos IoTSE y WoTSE se han usado indistintamente. Sin embargo, es posible reconocer que los IoTSE se enfocan en encontrar cosas entendidas como objetos, dispositivos o sensores (entidades físicas) y tratar con la búsqueda a nivel de datos (con ciertos matices, como datos sin procesar, datos vinculados o datos en *streaming*); o buscando a nivel de objeto (como, por ejemplo, un sensor, un conjunto de sensores o la ubicación de un objeto).

Por otro lado, cuando se logra un mayor nivel de abstracción de las cosas y se emplea una representación digital en forma de *Avatar Web* o gemelo digital (referido en inglés como *Digital Twin*), es posible enfocar la búsqueda en las cosas, empleando la riqueza descriptiva del modelo de cosas. Entonces, WoTSE puede encontrar servicios exhibidos por las cosas

o incluso acciones que las cosas pueden realizar en el mundo real. WoTSE puede ir más allá de encontrar cosas tangibles o intangibles, incluidas las relaciones sociales entre cosas y cosas e individuos. Además, WoTSE puede proporcionar una búsqueda universal, es decir, la capacidad de "*descubrir cualquier cosa, en cualquier lugar y en cualquier momento*" [99].

Denotamos como IoTSE aquellos enfoques capaces de buscar datos e información relacionada con sensores sin una representación Web formal. Denotamos como WoTSE aquellos enfoques capaces de buscar en todo el espectro de información, desde metadatos estáticos hasta todo, y específicamente para tratar con las representaciones digitales Web, sus estados y todos los diferentes niveles de abstracción. Denotamos como **Ámbito de búsqueda** la característica de un sistema RI-IoT o RI-WoT para responder a una consulta dada, con cualquiera de los representantes o abstracciones de información que van desde metadatos puros, datos de sensores, imágenes, sonidos, vídeos, *streaming*, a personas, eventos o todo.

## 2.2. Revisión Sistemática de Literatura en IoTSE|WoTSE

En esta sección se describe el enfoque utilizado para realizar una Revisión Sistemática de Literatura (RSL)<sup>10</sup>. Incluye la descripción del proceso seguido para la recopilación de los estudios primarios y la selección de las obras analizadas en este trabajo. El RSL se presenta sobre la recuperación de información para escenarios de IoT y WoT, incluida la definición de las preguntas de investigación y la estrategia de selección. Especificamos los criterios de inclusión y exclusión junto con los criterios de garantía de calidad. A continuación, se describe el procedimiento de extracción de datos. Hemos llevado a cabo el RSL siguiendo un enfoque iterativo de tres fases: revisión de planificación, ejecución e informes.

### Definición de preguntas de investigación

Definimos las siguientes Preguntas de Mapeo (*Map Questions*, (MQ) por sus siglas en inglés), que apoyan y guían el alcance de esta RSL:

- MQ1: ¿Cuántos estudios se han publicado a lo largo de los años?
- MQ2: ¿Quiénes son los investigadores más activos en el área?

---

<sup>10</sup>Nota: este trabajo doctoral incluye un conjunto de datos alojado en IEEE DataPort(TM), un depósito de datos creado por IEEE para facilitar la reproducibilidad de la investigación. La versión final de los datos extraídos y los resultados del análisis están disponibles como conjuntos de datos abiertos: *Conjunto de datos para la revisión sistemática de literatura sobre motores de búsqueda de IoT y WoT* en <https://iee-dataport.org/documents/dataset-revisión-literatura-sistemática-motores-de-búsqueda-iot-and-wot>. Incluye el protocolo RSL utilizado en este estudio y los datos de análisis de las preguntas de investigación de RSL.

- MQ3: ¿Cómo han evolucionado las subtareas de recuperación de información sobre IoT y WoT a lo largo de los años?
- MQ4: ¿Qué desafíos se han identificado y qué preguntas de investigación abiertas aún requieren nuevos avances?

Para perseguir los objetivos de este trabajo, definimos las siguientes Preguntas de Investigación (*Research Questions*, (RQ) por sus siglas en inglés):

- RQ1: ¿Existen otras RSL y taxonomías desarrolladas para RI-IoT e RI-WoT?
- RQ2: ¿Qué subtareas de RI se han utilizado, afectado o propuesto para los paradigmas IoT/WoT?
- RQ3: ¿Cuáles son los fundamentos básicos para IoT, WoT y la aplicabilidad de las subtareas de IR en esos paradigmas?
- RQ4: ¿Cómo se han modelado y estructurado algorítmica y técnicamente las soluciones propuestas para afrontar los retos de IoT/WoT?

### **Definición del Alcance RSL: PICOC**

Para ayudar a responder las preguntas de investigación, analizamos la literatura siguiendo el alcance de PICOC.

- Población (P): "*IoT*" O "*Internet of Things*" O "*WoT*" O "*Web of Things*".
- Intervención (I): "*Information Retrieval*" O "*Search Engine*" O "*Crawling*" O "*Indexing*" O "*Querying*" O "*Retrieval*" O "*Ranking*" O "*Discovery*" O "*Presenting*".
- Comparación (C): No aplicable.
- Resultados (O): Documento sobre el marco taxonómico de vanguardia y las direcciones futuras para RI en IoT y WoT.
- Contexto (C): Publicaciones revisadas por pares y de conferencias sobre subtareas, modelos, algoritmos y técnicas de RI en idioma inglés.

**Definición de los Criterios de Inclusión (IC) y Exclusión (EC):**

- IC1: Documento que proponga al menos un mecanismo de recuperación de información (subtarea, método, técnica, modelo, algoritmo, motor de búsqueda); Y
- IC2: La solución propuesta se aplica a IoT O WoT O precursor; Y
- IC3: El período de publicación del artículo debe estar comprendido entre 2000 y 2022.
- IC4: El estudio debe estar disponible en texto completo en bibliotecas digitales seleccionadas.
- EC1: El estudio es irrelevante para IoT|WoT o cualquiera de sus dominios precursores y el campo de RI.
- EC2: El estudio es una versión similar o reducida de una obra completa.
- EC3: El artículo es un estudio terciario o cualquier informe no científico.
- EC4: El estudio está escrito en un idioma diferente al inglés.

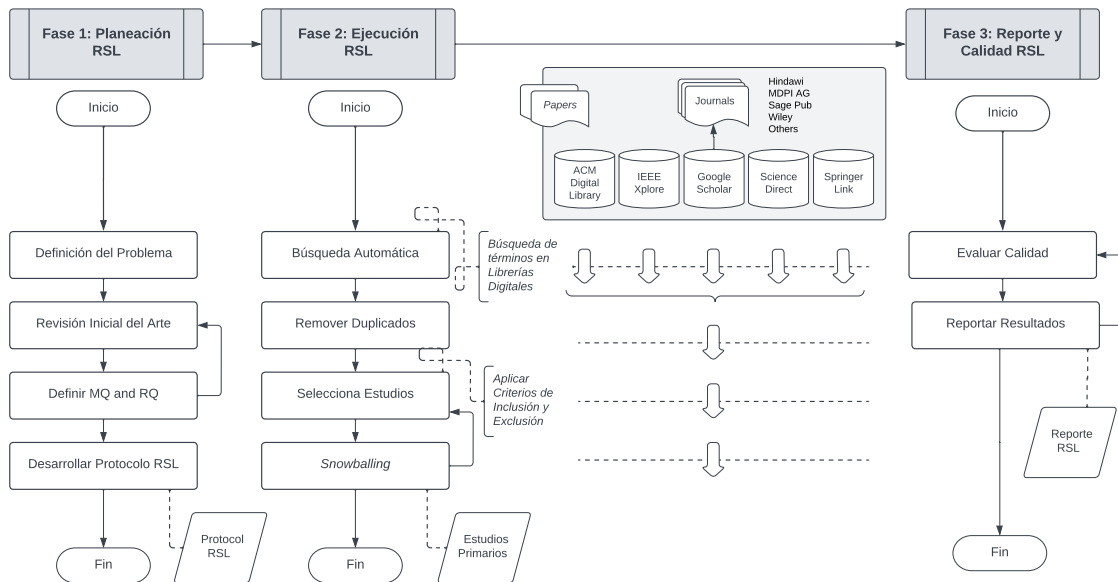


Figura 2.1 Protocolo RSL para RI-IoT y RI-WoT.

## Búsqueda y Extracción Iterativa de Datos

La búsqueda de estudios primarios deberá asegurar una cobertura integral. La estrategia de búsqueda en el RSL se muestra en la Figura 2.1, considera una compensación para incluir simultáneamente los estudios primarios más relevantes y excluir los irrelevantes. Bases de datos utilizadas: ACM Digital Library, IEEE Xplore, Google Scholar, ScienceDirect y Springer Link.

El proceso de extracción de datos en este estudio pasa por tres fases de la siguiente manera:

- Fase 1: El revisor principal extrae datos de los estudios principales, bibliometría, demografía, taxonomía (si corresponde), características e información de investigación. Los datos extraídos se tabulan para cada uno de los estudios. Los datos se generan para cada trabajo respondiendo preguntas de calidad y de autoevaluación.
- Fase 2: La estrategia de búsqueda realizada consta de cuatro etapas siguiendo la metodología descrita en [85]:
  1. se realizó una búsqueda automática sobre las bibliotecas digitales científicas más relevantes.
  2. se eliminan todos los documentos duplicados.
  3. siguiendo criterios predeterminados de inclusión, sólo se consideran trabajos relacionados con el tema.
  4. se buscan más estudios mediante bola de nieve hacia adelante **snowballing**.
- Fase 3: Evalúa la calidad de estos resultados e informes. Se recopilan y se estructuran los resultados más destacados para analizarlos y discutirlos. El investigador evalúa la calidad utilizando una lista de verificación para evaluar los aspectos relevantes para el RSL.

Dependiendo de la puntuación de la evaluación, cada trabajo se incluye o se excluye en la fase final para el informe.

El resultado del SLR consiste en una encuesta sobre sistemas RI para IoT y WoT. Esto comienza con la revisión del trabajo pionero y los estudios primarios recientes informados entre 2000 y 2022. Los resultados de preguntas de investigación del corte técnico se presentan en forma gráfica a través del Capítulo 3 y de las siguientes secciones. Los resultados en forma gráfica para las preguntas referentes a la bibliometría de los trabajos analizados puede ser consultado en línea en IEEE DataPort(TM), en el enlace proporcionado en la descripción inicial del RSL.

## 2.3. Pioneros y Evolución de RI para IoT y WoT

Es posible ubicar la investigación acerca de los mecanismos de búsqueda para los predecesores de IoT|WoT a principios de la década de 2000. Esta etapa puede ser nombrada como "los precursores de IoTSE basado en RI". A partir de 2006, existe un conjunto importante de esfuerzos llevados a cabo por Rukzio et al. [111], Ostermaier et al. [94], y Guinard-Trifa-Wilde [40], que desencadenó el término WoT. Sin embargo, se debe esperar hasta el final de la década de los 2000s antes de que se reporten algunos sistemas IoTSE|WoTSE completos. Estos enfoques exploraron diferentes lados del problema y contribuyeron al establecimiento de nuevas formas de ver el campo RI. Aquí se destacan las principales propuestas denominadas: DYSER [94], SNOOGLE [140], MICROSEARCH [124], OCH [34] y MAX [144].

Como se ilustra en la Figura 2.2, se identifican tres direcciones principales en estas líneas de investigación:

- Búsqueda de entidades: cosas, personas, lugares (con base en RI)
- Descubrimiento de recursos y servicios: datos, eventos, sensores/actuadores de dispositivos IoT (no basados en RI)
- Búsqueda de datos, secuencias y datos vinculados (basado en SPARQL)

Se centra el análisis en este trabajo sólo en aquellos sistemas RI completos o componentes que presentan algunos avances, adaptaciones o trabajos pioneros.

**Búsqueda multimodal y sensible al contexto.** Se emplea el término Búsqueda multimodal para referirnos a la capacidad de un sistema RI de usar varios medios distintos para la entrada y salida usando diferentes métodos para obtener los resultados relevantes. La conciencia del contexto es un caso particular de la búsqueda multimodal. En la década anterior, se pueden reconocer tres tendencias significativas:

- a) Recuperación en tiempo real (a nivel de sensor): MICROSEARCH, SNOOGLE, MAX;
- b) Búsqueda basada en ubicación (múltiples enfoques): SENSEWEB; y
- c) Estado y Búsqueda basada en valores: DYSER.

Todos utilizan un enfoque monomodo en un contexto particular, es decir, geoespacial, espaciotemporal o temático. Sin embargo, solo se presentaron unos pocos enfoques utilizando múltiples tipos de búsqueda de entrada/salida. Uno de los primeros trabajos de la década que utiliza el enfoque multimodal es IOT-SVK presentado en [23], descrito como un marco IoTSE híbrido basado en contexto espacio-temporal, valor y palabras clave. Si se examina en



detalle la WoT, es posible la recopilación de múltiples tipos de información, y de esta forma se pueden dar múltiples respuestas a los usuarios finales (humanos o máquinas). El cruce entre información espacial y temporal como I/O con información de sensores de cosas abstractas puede considerarse una búsqueda multimodal de primer nivel. Por ejemplo, el enfoque de Du et al. [25] utiliza especificaciones funcionales con el contexto espacio-temporal para proporcionar capacidades de búsqueda sobre IoT|WoT. Otro trabajo, [81] presenta GANDER, un motor de búsqueda para redes inalámbrica de sensores (*Wireless Sensor Networks*, WSN por sus siglas en inglés), cuyo objetivo es encontrar recursos próximos de IoT disponibles en los contextos *.Aquí y ahora"*.

**Búsqueda de ubicación y consciente de la distancia.** [60] presenta un robot autónomo para encontrar objetos del mundo real etiquetados en una habitación. A los objetos se les asigna su ubicación relativa de distancia y esta se registra en su etiqueta RFID en una base de datos. Si bien no es un sistema IoTSE|WoTSE completo, representa una dirección de investigación para implementar estrategias que permitan realizar búsquedas basadas en la ubicación sin usar puntos de referencia, y en su lugar usar la distancia relativa en espacios pequeños del mundo físico. [59] presenta DRAGON, un algoritmo de descubrimiento de datos basado en una tabla de datos distribuidos (DDT) que recopila metadatos estáticos de sensores circundantes y que utiliza un protocolo de enrutamiento punto a punto que puede realizar consultas a los nodos de IoT para responder acerca de la localización de las cosas.

Los enfoques basados en la ubicación y la conciencia de la distancia pueden ser vistos como un subconjunto de la categoría multimodal, donde estas estrategias especializadas buscan la posición de los recursos en el espacio con una variedad de mecanismos. Como se señaló anteriormente, el descubrimiento proximal basado en la ubicación sigue siendo un tema de investigación abierto. Los usuarios finales quieren buscar cosas inteligentes en su entorno inmediato y los IoTSE|WoTSE pueden resolver el problema con cualquiera de los tres enfoques: centralizando los datos, distribuyendo la consulta a los nodos o a través de un mecanismo híbrido.

Es importante señalar algunos trabajos y su contribución a desafíos no predominantes en la perspectiva RI. Por ejemplo, [107] introduce una mejora de seguridad en la esfera IoTSE mediante el uso de criptografía de curva elíptica, (*Elliptic Curve Cryptography*, ECC por sus siglas del inglés ) en la que se incluye el soporte de seguridad como una dimensión, respondiendo si el sistema IoTSE incluye protección de seguridad integrada o no [107]. Donde según las condiciones, el sistema IoTSE debe garantizar la privacidad de las identidades y los mensajes siguiendo un criptosistema basado en identidad.

De otra parte, los mecanismos de Inteligencia Artificial (IA) se incorporan a las arquitecturas IoT y WoT. Por ejemplo, [11] presenta MSNET y MSNAS, dos metodologías de

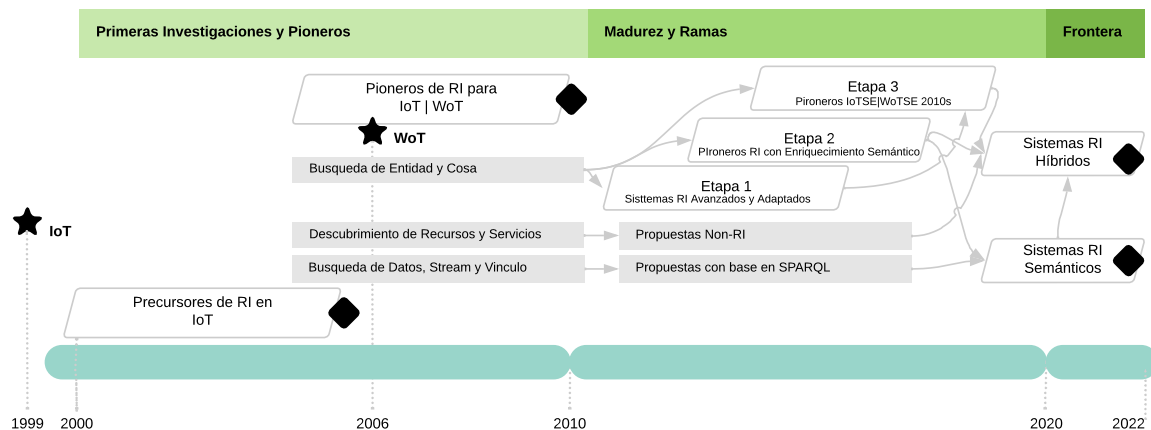


Figura 2.2 Evolución de RI para IoT|WoT.

búsqueda de arquitectura neuronal basadas en gráficos para IoT. Admiten un espacio de búsqueda flexible e implementan un mecanismo de búsqueda progresivo de grano a grueso, pero orientado a la búsqueda de arquitectura neuronal (NAS). El estudio muestra que los modelos actuales de redes neuronales profundas (DNN) aún no pueden implementarse completamente en dispositivos IoT. Si bien el enfoque principal de este trabajo no es RI, parte de esta nueva perspectiva podría emplearse en nuevos sistemas de clasificación/recuperación, especialmente para clasificar los resultados considerando las restricciones en las métricas de IoT. Qian et al. [106] presenta un motor de búsqueda de sensores heterogéneo basado en contenido y ubicación de IoT. La arquitectura prototipo considera algunas funciones esenciales: 1) un mecanismo de descubrimiento de recursos ejecutado por puertas de enlace regionales y locales; 2) anuncios de servicios, que deben entenderse como resultados de búsqueda dadas las necesidades de información de algunos usuarios y ordenados por calificación, comentarios y contexto; 3) propagación de información implementada como un enfoque de almacenamiento en caché para consultas frecuentes; 4) procesador de consultas basado en el registro de recursos, y 5) Servicios de nombres de IoT basados en RFC2141 (URN).

Aquí es importante notar que, si bien los enfoques con base en SPARQL y aquellos no relacionados con sistemas RI no son el enfoque de este documento, algunos de estos se han incluido al analizar los conjuntos de datos. Estos (enfoques SPARQ/Non-RI) se representan en la Figura 2.2, donde la evolución conceptual de IoTSE/WoTSE ilustra la madurez obtenida por las diferentes ramas agrupadas por las principales líneas de investigación; en la figura, también se destacan la situación actual y enfoques en la materia.

## 2.4. Taxonomías para Motores de Búsqueda en IoT|WoT

Las taxonomías son herramientas de categorización que permiten obtener una panorámica 360° sobre un tópico de investigación en particular, y en nuestro contexto de motores de búsqueda IoTSE|WoTSE, permite el contraste sobre sistemas de RI existentes [152], y pueden ayudar a inspirar nuevos sistemas y ayudar en la evolución de estos. Técnicamente hablando, existen cuatro etapas que guían el estudio taxonómico de especies en el mundo natural. Estas etapas son la identificación, la caracterización, la clasificación y por último la denominación. Entonces, una parte esencial de las taxonomías debe considerar aquellas dimensiones que faciliten estos cuatro elementos bajo evaluación para finalmente proponer convenciones de nomenclatura para las familias de RI. Desde una perspectiva tecnológica, la taxonomía debe proporcionar una visión holística de todos los enfoques, una caracterización precisa de las categorías o familias de especies, y conocimientos y futuras direcciones prospectivas para la evolución de esos sistemas RI.

En la Tabla 2.1, se proporciona una lista de taxonomías existentes asociadas con trabajos de IoTSE/WoTSE. También se presenta una categorización por *Tipo*, en la que se agrupa la taxonomía, su enfoque y algunos puntos a resaltar. Esta tabla contiene el número de propuestas estudiadas en cada taxonomía y las dimensiones allí evaluadas. Para este último punto, se destaca la taxonomía de Tran et al. [132] que contiene veinticuatro (24) dimensiones que incluyen la *meta-ruta*. En lugar de ampliar la granularidad en las dimensiones, nuestra propuesta apunta a agruparlas en tareas en cada etapa RI. Tran et al. adiciona un análisis fundamental de los enfoques desde el punto de vista del flujo de información. Una pieza fundamental del estudio es que la taxonomía [132] está estrechamente relacionada con la *meta-ruta*: una clasificación en forma de convención y nomenclatura de sistemas IoTSE/WoTSE similar a una huella dactilar, la cual sigue un enfoque que espera capturar el flujo de información en la recuperación.

Sin embargo, este acercamiento carece de un análisis en profundidad desde la perspectiva de la RI. La extensa granularidad en las dimensiones sesga de algún modo las características propias del sistema, en lugar de formular un criterio unificado y direcciones para la evolución RI. Por el contrario, Faheem et al. [30] consolida cuatro dimensiones: el caso de uso (considerado una dimensión en sí misma), el esquema de las cosas, la indexación y la recuperación. Aquí, la consolidación no permite una caracterización precisa de los diferentes trabajos. Las otras taxonomías se encuentran en algún punto medio de granularidad, con un conjunto de dimensiones que cubren las principales tareas de cada etapa RI y las características experimentales o del prototipo. Recientemente, Choudhury et al. [13] presenta una encuesta y una taxonomía para las técnicas de búsqueda de sensores. El análisis compara

técnicas e incluye como dimensiones el criterio del usuario, la detección como servicio, la escalabilidad, el aprendizaje automático, el acceso a la ubicación y la detección de amenazas. Si bien expone una nueva comparación entre técnicas, (solo se consideran cinco trabajos), carece de orientación RI y se limita a técnicas de búsqueda en la capa de sensores.

Taxonomía	Tipo	No. Propuestas Estudiadas	Dimensiones Evaluadas	Enfoque	Precauciones
Multi-Taxonomía (Propuesta)	Perspectiva RI	251	23	Holística	Diversidad
Tran et al [133]	Flujograma para WoT	214	24	Modelo Modular	No RI en detalle
Pattar et al [96]	Centrado en Datos/Objectos	60	7	Aplicabilidad	No RI en detalle
Suparna et al [18]	Modelo Contexto	55	6	Ontologías	No Caracterización
Noura et al [92]	Dispositivo-Plataforma	31	12	Interoperabilidad	No orientado a RI
Zhou et al [152]	Principio-Representación-Alcance	49	9	Técnicas	No Caracterización
Faheem et al [30]	Principio-Representación-Alcance	37	4	Búsqueda Dinámica	No Caracterización
Aziez et al [79]	Arquitectura-Representación	23	10	Descubrimiento	No orientado a RI
Romer et al [112]	Diseño Arquitectural	7	9	Tiempo Real Local	Revisión Limitada
Zhang et al [146]	Diseño Arquitectural	6	14	Tiempo Real Local	Revisión Limitada
Choudhury et al [13]	Diseño Arquitectural	5	6	Búsqueda de Sensores	Revisión Limitada
Evdokimov et al [29]	Alcance-Calidad	5	9	Descubrimiento	Revisión Limitada

Tabla 2.1 Marcos taxonómicos IoTSE|WoTSE, dimensiones y enfoques.

### 2.4.1. Identificación de Motores de Búsqueda IoTSE|WoTSE

Parte integral de una taxonomía IoTSE|WoTSE es su identificación y caracterización. Es importante notar que no todos los enfoques de investigación construyen un *organismo* completo. Por *organismo* completo este trabajo se refiere a un sistema RI de extremo a extremo, capaz de recibir información de los usuarios finales en forma de consultas de información y responder con información relevante, que incluye todos los procesos, subprocesos y tareas intermedios de RI. Por el contrario, la mayoría de los trabajos se centran en un bloque funcional o en una parte de todo el sistema RI. No obstante, también es necesaria esta visión por componente y los avances de un componente específico de IoTSE|WoTSE. En el mundo natural, los taxónomos presentan descripciones escritas organizadas de las características de especies similares.

### 2.4.2. Caracterización de Motores de Búsqueda IoTSE|WoTSE

En las siguientes secciones y subsecciones, se presenta un estado del arte de los sistemas IoTSE|WoTSE guiadas por las discusiones sobre el tema en taxonomías pre-existentes [96], [133], [132], [152]. El enfoque de estas encuestas y taxonomías es una parte importante de la discusión. Tran et al. [132] considera que se ha prestado gran atención a una forma específica de búsqueda en [146], [29], [112], algunas categorizaciones de IoTSE|WoTSE emplean la discusión guiada en los casos de uso como se presenta en Faheem et al. [30]. Faheem et al. [30] categoriza los sistemas reusando la clasificación anterior propuesta en [152] y [133], mientras identifica la aplicabilidad específica de estas investigaciones a través de los resúmenes de los trabajos más destacados. Es importante notar algunos esfuerzos para la construcción de taxonomías particulares que sirven para propósitos distintos a IoTSE y WoTSE, como por ejemplo para i) identificar las causas raíz de la interoperabilidad en IoT [92] y ii) clasificar la semántica y modelos de contexto orientados a WoT, según el uso de ontologías de dominio o de dominio cruzado, haciendo referencia a áreas de conocimiento y sus aplicaciones especializadas [18].

Este trabajo doctoral considera que existe una visión fragmentada en la investigación IoTSE|WoTSE. El marco de la taxonomía debe abarcar todas las *familias de especies* en lugar de un escenario específico. Además, algunas taxonomías han centrado la atención en los posibles problemas técnicos de IoTSE|WoTSE como en [152], en lugar de adoptar una visión más amplia y holística. Además, es posible reconocer que no existe una caracterización precisa de WoTSE de las taxonomías de IoTSE; ni existe una diferenciación de los mecanismos de búsqueda de las técnicas de descubrimiento. Los trabajos anteriores que construyen marcos de taxonomía para IoTSE|WoTSE son comparados en la tabla 2.1, que

es una versión actualizada, extendida y complementada de [133], donde se incluyen diez taxonomías con el número de enfoques estudiados, su tipo, dimensiones evaluadas, enfoque y advertencias. En cuanto a las advertencias, se encuentra que algunas revisiones tienen un alcance limitado; otros no presentan una caracterización precisa del trabajo de investigación dentro de las categorías; una minoría no está orientada a RI, por lo que se enfoca en otro tipo de discusión. Si bien es cierto que [133], [96] y [152] son las revisiones más exhaustivas de los últimos años, se requiere un estudio granular que cubra cada una de las etapas de los criterios de adaptación y homogeneización de RI al tiempo que se exponen los desafíos actuales en términos de mejora y evolución que enfrentan los sistemas IoTSE|WoTSE, así como las direcciones futuras.

### 2.4.3. Clasificación de Motores de Búsqueda IoTSE|WoTSE

Una clasificación binaria aceptada ha dividido los mecanismos de búsqueda para IoT|WoT en dos [118]:

- Basado en contexto, esto es, impulsado por los metadatos, o el entorno y las relaciones circundantes que permean la búsqueda.
- Basado en contenido, esto es, impulsado por los datos sin procesar y los valores medidos por los sensores que además podrían definir un cierto estado para las cosas.

Idealmente, los sistemas conscientes del contexto deben operar en entornos heterogéneos del mundo real que infieren automática o manualmente la intención del usuario, aumentando la precisión de los resultados [72].

Se entenderá por contexto cualquier información que pueda caracterizar y sea relevante para la interacción entre un usuario y un motor de búsqueda, es decir, la ubicación actual del usuario o el marco de tiempo esperado de las respuestas. La búsqueda basada en contenido, generalmente enmarcada en la búsqueda de sensores o sus datos, se ha definido formalmente de la siguiente forma: "*dada una gran cantidad de sensores, encuentre de manera eficiente un subconjunto de sensores que generen un valor buscado en un momento dado*" [28]. Esta definición se ha modificado posteriormente para incluir la búsqueda de un conjunto de entidades dado un estado deseado [112]. Sin embargo, aun con esta simple división se han encontrado inconsistencias entre los modelos de clasificación, principalmente debido a las diferentes perspectivas adoptadas para la creación de estas taxonomías.

Este trabajo ha identificado las siguientes perspectivas que se han empleado al construir los diferentes modelos de clasificación para IoTSE|WoTSE:

- Perspectiva con base en funciones y principios [152], [29]

- Perspectiva con base en el alcance de la búsqueda y los modelos de cosas [152], [96]
- Perspectiva con base en flujo de información y arquitectura [112], [146], [133]
- Perspectiva con base en casos de uso y su aplicación específica [30]

El impacto en el cambio de perspectiva de las taxonomías existentes puede resaltarse mediante ejemplificación. Tomando como base la taxonomía propuesta por Tran et al. [133], [131], la cual propone una convención de nomenclatura fundada en el concepto de *meta-ruta*. Es posible notar la divergencia en la clasificación de trabajos IoTSE|WoTSE, a la luz de distintas ópticas, por tanto diferente identificación, caracterización y clasificación. Este modelo descriptivo por *meta-ruta* [131], la idea es representar el flujo de información no solo como una entrada o salida (E/S), sino como una transición del tipo  $I \rightarrow X \rightarrow O$  a través de un intermediario, por lo que el modelo *meta-ruta* se representaría como:

$$(Consulta_{contenido}) \rightarrow Cosas^* \rightarrow (Resultado_{contenido}) \quad (2.1)$$

El modelo utiliza una notación especial en la que R representa al Representante virtual, es decir, el *Avatar Web* o el *Digital Twin*, T para cosa, F para Funcionalidad, D para contenido de IoT dinámico y S para contenido de IoT estático. Por medio de este modelo, [131] reconoce ocho familias de especies IoTSE|WoTSE:

- Flujo directo:
  - $R \rightarrow R$
  - $D \rightarrow D$
  - $F \rightarrow F$
  - $S \rightarrow S$
- Secuencia con cosa como intermediario:
  - $D + R \rightarrow T \rightarrow R$
  - $R \rightarrow T \rightarrow R + D$
  - $D \rightarrow T \rightarrow R$
  - $R \rightarrow T \rightarrow F$

No existe un lenguaje ni un entendimiento común para clasificar los “*organismos*” IoTSE|WoTSE y sus partes constitutivas. Mientras Tran et al. [133] clasifica algún tipo



de IoTSE como  $(R \rightarrow R)$ , en muchos casos, Pattar et al. [96] y Zhou et al. [152] no siguen estrictamente la misma clasificación según sus ópticas para el IoTSE en cuestión. Por ejemplo, DISCOWoT es desde una perspectiva basada en el flujo un sistema  $R \rightarrow R$  en [133], es un sistema con base en el contexto desde la perspectiva del alcance de la búsqueda en Pattar et al. [96], y un sistema con base en datos y entidad según el trabajo de Zhou et al. [152]. Y de forma similar para otros IoTSE|WoTSE. Tran et al. [133] clasifica FORWARDS-IOT como tipo  $(R \rightarrow R)$ , mientras que para Pattar et al. [96] es heterogéneo/semántico y basado en ontología y para Zhou et al. [152] es basado en datos enlazados/federados debido a su representación del conocimiento.

Si bien las dimensiones con las que se analizan los sistemas tienen un núcleo común, las perspectivas con las que finalmente se clasifican son muy distintas. Además, no existe una perspectiva orientada a RI que impulse la clasificación. Aunque trabajos anteriores han evaluado docenas de parámetros y dimensiones, no existe una visión holística que reúna todas las piezas.

#### 2.4.4. Marco de Taxonomía Unificado y Holístico Propuesto

El marco taxonómico unificado y holístico propuesto se describe en la Tabla 2.3. Se han incluido una comparativa con las principales taxonomías existentes. Se argumenta en este trabajo doctoral que los sistemas IoTSE deben diferenciarse de los sistemas WoTSE. Es probable que el primero se use directamente desde dispositivos autónomos de IoT a partir de datos agregados, servicios compuestos innovadores o dispositivos autónomos de IoT dentro de ciudades inteligentes. En este contexto, solo se espera la interacción M2M sin intervención humana. Este último debe ser consciente del contexto social y centrado en el ser humano como usuario final. En este sentido, un WoTSE proporcionará una abstracción Web (probablemente en forma de Avatar) que cumpla con el requisito al devolver la ubicación geográfica y realizar una acción predefinida (cuando sea posible). Esta propuesta describe una imagen precisa del amplio espectro de posibilidades desde una perspectiva RI cuyo objetivo final es brindar información relevante y precisa, independientemente de la ingeniería específica detrás de las técnicas y modelos de búsqueda.

En este trabajo se diferencian los sistemas IoTSE de los WoTSE, una distinción significativa de otras propuestas de taxonomía. IoTSE cubre todos los sistemas de búsqueda y descubrimiento de IoT, mientras que WoTSE cubre los sistemas RI-WoT. Nuestra propuesta en la Tabla 2.3 mapea una a una las familias de especies en Tran et al. [133], Pattar et al. [96] y Zhou et al. [152]. Si bien los sistemas IoTSE se han estudiado a fondo, existen preguntas de investigación abiertas para los sistemas WoTSE, que no se han identificado ni investigado adecuadamente. El enfoque *meta-ruta* en [131] fue la primera taxonomía que

propuso un panorama general que, al tiempo que encapsula el flujo de información para una futura arquitectura RI modular, define el proceso intermedio involucrado a nivel de objeto. Sin embargo, algunas obras disruptivas recientes no fueron clasificadas ni incluidas en las taxonomías anteriores debido al tiempo de publicación.

Además, algunas familias pueden representar múltiples comportamientos. En general, los datos de IoT, la transmisión y la búsqueda con base en contenido se incluyen universalmente en todas las taxonomías anteriores. La conciencia del contexto solo se estudia implícitamente, pero no se considera explícitamente en el modelo de *meta-ruta*. Zhou et al. [152] reconocer los recursos como sensores o entidades; sin embargo, no se incluye explícitamente la búsqueda de servicios IoT. Algunos trabajos usan los términos *capacidad* y *funcionalidad* indistintamente; otros significan un concepto ligeramente diferente. Se debe interpretar *funcionalidad* según el modelo de *meta-ruta* como un servicio físico y lógico provisto por cosas físicas y virtuales. Otro aspecto esencial es el reconocimiento explícito de la búsqueda predictiva de IoT como parte del marco de taxonomía. Si bien esto podría entenderse como una técnica para simplemente reducir el alcance de la búsqueda, creemos que el potencial de los enfoques en esta categoría puede dar forma a la evolución de la investigación de RI.

Por otro lado, Pattar et al. [96], y Zhou et al. [152] identifican la búsqueda de IoT con base en la ubicación como una categoría independiente. La el marco taxonómico propuesto concuerda con esto. Esta familia tiene sub-ramas que merecen un estudio y categorizaciones por separado. Además, si le asigna este estatus al contexto espacial, lo mismo debería hacerse con el contexto temporal. La búsqueda temporal de IoT aún no se ha estudiado a fondo; debe considerarse independientemente de la búsqueda predictiva debido a sus características contrastantes. Finalmente, la búsqueda semántica de IoT se reconoce como parte del enfoque heterogéneo en Pattar et al. [96] y como parte de la búsqueda de datos de *streaming* en Zhou et al. [152]. Se debe construir un lenguaje común para describir las cosas sobre ontologías genéricas o bien conocidas para mejorar la interoperabilidad de IoT|WoT y sus aplicaciones. Además de la búsqueda semántica y todas las familias IoTSE identificadas, se propone un conjunto de familias WoTSE. La búsqueda multi-modal WoT puede ayudar con la conciencia temática y múltiples ámbitos de búsqueda cuando se trata de WoT.

La búsqueda de WoT centrada en las cosas o en las redes sociales será la próxima generación de motores de búsqueda que ayuden a los usuarios finales a encontrar múltiples funciones, funcionalidades/acciones, eventos relacionados con las cosas, o que apoyen el intercambio y la orquestación de servicios de WoT complejos que busca el usuario final en su entorno social en el mundo real a través de sus avatares. No es reconocido por ninguna taxonomía anterior, se defiende en el marco taxonómico propuesta la idea de que, dadas las características ubicuas de WoT, la búsqueda segura sobre WoT constituye un pilar para el

futuro de WoT. La seguridad, la privacidad y la confianza deben impregnar todas las subtarear de RI.

Se incluye explícitamente la búsqueda de acciones sobre WoT, tanto acciones virtuales intangibles -realizadas por los representantes virtuales de las cosas, avatares o gemelos digitales en el mundo virtual, como acciones físicas -que pueden ser desencadenadas por las acciones virtuales intangibles en el mundo real-. La búsqueda WoT progresiva es una dirección de investigación prometedora que adopta un enfoque gradual de las dimensiones espacio-temporales, extensible a otras. En la parte superior del marco taxonómico propuesto, un ideal, probablemente una familia hipotética, es la búsqueda universal sobre WoT, una especie capaz de encontrar todo independientemente de los matices de WoT, donde quizás las emociones y sensaciones sintéticas podrían jugar algún papel.

## 2.5. Rastreo y Descubrimiento para IoT|WoT

Rastrear el WoT sigue siendo un campo abierto de investigación [133].

*"Rastreador" (a veces también llamado "robot" o "araña" o "crawler" en inglés) es un término genérico para cualquier programa que se utiliza para descubrir y escanear automáticamente sitios web siguiendo enlaces de una página web a otra". - Google Search Console <sup>11</sup>.*

Es probable que un rastreador de WoT contenga al menos un componente en la forma de un mecanismo de "descubrimiento". Nath et al. [90] defiende la idea de que un rastreador WoT podría ser una aplicación específica de dominio y en su lugar no debería ser genérica. Sin embargo, es debatible que ha de distinguirse un rastreador WoT de un descubridor IoT (enfocado principalmente a encontrar los recursos y servicios con base en sus identificadores o enlaces). Por tanto, en cambio, se considera que un rastreador WoT debe ser de tipo genérico como en la WWW, teniendo en cuenta los matices relacionados con los protocolos de comunicación de intercambio de datos utilizados por la infraestructura IoT|WoT, como CoAP RESTful o XML HTTP Request. Sin embargo, su arquitectura debe ser independiente del protocolo. Skarmeta et al. [120] indica que los mecanismos de rastreo y análisis de enlaces no son directamente aplicables a WoT debido a la falta de una definición formal de URI, similarmente en [132]. Esta declaración aplica solo para aquellos sistemas IoT, *Web of Objects* (WoO) y *Web of Sensors* WoS que no emplean el mecanismo de enlace URI/URL entre documentos. Debido a que la identificación URI/URL es ahora una parte integral de la arquitectura W3C WoT.

<sup>11</sup><https://developers.google.com/search/docs/advanced/crawling/overview-google-crawlers>

Our proposal		Tran et al.	Pattar et al.	Zhou et al.		
Denotation	Search Scope	Resources	(Inherent)	Content Being Searched		
Thing Search Scope	WoTSE (+)	Everything	NC	NC	NC	
		Live Event	NC	NC	NC	
		Person	NC	NC	NC	
		Social Relations	NC	Social Structure-based	NC	
		Space as Entity	NC	Location-based	NC	
		Service	Functionality	Resource and Service	NC	
Entity Search Scope	(-)	Set of Entities	Representative	Semantic and Ontology	Non-location Clustering	
		Entity		Semantic and Ontology	Entity Context-based	
		Set Sensors/Actuators		Resource and Service	Non-location Clustering	
		Actuator		Resource and Service	NC	
Data Search Scope	IoTSE Search Scope (+)	Sensor	Dynamic information	Semantic and Ontology	Sensor Context-based	
		Spatial		Location-based	Clustering Location-based	
		Future/Predictive		NC	NC	
		Historical		NC	O&M Data Historical	
		Linked		NC	Semantic and Ontology	Knowledge Representation
	(-)	Stream	Static information	Data-based Content	Range	Sensor Content-based
		Continuous			NC	
		Instantaneous			NC	
		Metadata			NC	
					NC	

Fully Considered  
 Partial Considered  
 Not Considered

Our proposal		Tran et al.	Pattar et al.	Zhou et al.	
Denotation	Thing Model	Representative	(Inherent)	Data/Knowledge	
Thing Description	(+)	Avatar / Consciosness	NC	NC	
		Extended TD	NC	NC	
		TD Thing Description	NC	NC	
Entity Representation	Abstraction Level (-)	Entity-Relations	NC	NC	
		Extended Entity (Twin)	NC	NC	
		Entity-Services (OWL-S)	Functionality	Heterogeneous / Semantic and Ontology-based	NC
		Semantic Entity (OWL-DL)	Representative		NC
		Entity (Generic + FOAF)			NC
Sensor (SSN / SSON)	NC				
Data Representation	(-)	Linked Data	Heterogeneous / Resource and Service Discovery	Linked Data	
		Stream Data		Streaming Data	
		Raw Data			

Fully Considered  
 Partial Considered  
 Not Considered

Tabla 2.2 Alcance de Búsqueda para IoTSE|WoTSE; Niveles de Abstracción para Representaciones de Cosas.

Our proposal		Tran et al.	Pattar et al.	Zhou et al.	
Denotation		Meta-path	Taxonomy	Taxonomy	
WoT IR Systems	WoTSE	WoT Everything Search	NC	NC	Text/Spatial Indexing
		WoT Progressive Search	NC	NC	NC
		WoT Actions Search	R → T → F	NC	NC
		WoT Secure Search	NC	NC	NC
		WoT Thing Search	R → T → R + D	Heter/Semantic	Entity search
		WoT Social-based Search	NC	Object/Social Structured	NC
		WoT Multi-modal Search	D + R → T → R	NC	NC
IoT Search & Discovery Systems	IoTSE	IoT Semantic Search	NC	Heter/Semantic Ontology	Stream/Semantic Model
		IoT Temporal Search	D + R → T → R	NC	O&M/Historial Data
		IoT Location Search	NC	Object/Location-based	Location/Non-location
		IoT Predictive Search	D → D		NC
		IoT Service Search	F → F	Heter/Resource & Service	NC
		IoT Resource Search	R → R		
		IoT Context-based Search	NC	Data-based/Context	Entity/Sensor search
		IoT Content-based Search	D (+ R) → T → R		
		IoT Data Stream Search	D → D	Data-based/Content	
IoT Data Search	S → S		Linked/Stream   O&M		




	Fully Considered
	Partial Considered
	Not Considered

Tabla 2.3 Propuesta de Marco Taxonómico para IoTSE/WoTSE.

Tipo	Esquema	Descubrimiento	Patrón	Mecanismo de Integración	Referencia	Comentarios
I	Activo	Rastreador	URI/URL	Modelo Semántico	WOTS2E [53]	Carece de Dinamismo para WoT.
I	Activo	Rastreador	API espacial	Map-Reduce	ThinkSeek [118]	Rastreador multi-pasos. Limitado para mapas.
II	Activo	Pub/Sub	Definición de Servicio	Federación (NGSI-LD)	IoTcrawler [45]	Carece de escalabilidad y el modelo es asumido.
III	Activo	Local	Def. de Recurso	Calendarización	EasiCrawl [65]	Optimización de tiempo de sueño.
III	Activo	Local	Def. de Servicio	Agrupación Social	- [58]	Útil para construir casos de uso en la localidad.
IV	Pasivo	Registro	Def. de Servicio	Marco <i>Arrowhead</i>	AHF-WoT [115]	Carece de escalabilidad en escala global.
IV	Pasivo	Registro	Definición de Servicio	QoS   Algoritmo GODRP	- [147]	Optimización de calidad pero complejo en métricas.
V	Pasivo	Registro	Def. de Recurso	Estándar OGC	- [105]	Se asume LD como modelo, empleado como registro.

Tabla 2.4 Análisis de rastreo y descubrimiento para IoT|WoT

Tipo	Esquema	Índice	Estructura de Datos	Estrategia	Referencia	Comentarios
I	Tradicional   Numérico	DB	Tuple-to-dDim	Skyline	Grid-PPPS [46]	Consultas numéricas.
I	Tradicional   Ontología	DB (RDF)	Grafo RDF	Predicción	- [82]	Modelo de predicción como grafo RDF.
I	Tradicional   Numérico	<i>Hashing</i>	B-Plus   DHT	Compresión	- [24]	Enfoque en Streaming IoT.
II	Distribuido	Agrupación	DP-means	Geo-Localización	- [31]	Índices jerárquicos.
II	Distribuido	Agrupación	BCCF-tree	K-Means	- [5]	Enfoque en trayectorias de GPS.
II	Distribuido	Agrupación	BCCF*-tree	Partición espacial	- [57]	Enfoque en trayectorias de GPS.
II	Distribuido	<i>Hashing</i>	DHT	Coordinado	- [129]	Enfoque en balanceo de nodos heterogéneos.
III	Híbrido	Múltiple	STK-tree	<i>Caching</i>	CESCE [125]	Consultas combinadas.

Tabla 2.5 Análisis de Técnicas y Métodos de Indexación para IoT|WoT

En WoT, el rastreador tendrá tres funciones principales. El primero es la identificación de fuentes de datos dado un patrón y el soporte de diferentes formatos de datos, por ejemplo, JSON, XML, RDF, RDFa, OWL, HTML. El segundo es encontrar y extraer metadatos útiles o elementos semánticos entre datos anteriores sin procesar, estáticos, pseudo-estáticos o dinámicos, homogeneizando los datos extraídos. La tercera función es integrarlos, vincularlos y correlacionarlos para construir un sistema de índice [120].

Según la definición de Igenia et al. [45], el término *Descubrimiento IoT* se refiere más a conectar nuevas fuentes de datos a los sistemas. Si bien se considera que el paso para extraer información adicional de otras fuentes, serán los algoritmos de rastreo. Los descubridores IoT pueden actuar como un mecanismo centralizado o distribuido, con algunas tendencias hacia la descentralización y la federación. La mayoría de los enfoques centralizados tienden a crear un registro de recursos y/o servicios de IoT con diversos métodos. Los enfoques distribuidos pueden basarse en la ubicación o la infraestructura subyacente, y el estilo de la arquitectura puede estar segmentada en capas o agruparse.

En los rastreadores web tradicionales, la fase inicial es averiguar qué páginas web existen. Debido al tamaño de WWW, no existe un registro central de todas las páginas web. Los motores de búsqueda deben buscar constantemente recursos nuevos, actualizados y agregarlos a su lista de páginas conocidas. Este proceso se denomina "*Descubrimiento de URL*". Sin embargo, si el objetivo de un rastreo automático es construir una colección o repositorio, entonces, de acuerdo con [132], un mecanismo de descubrimiento global para IoTWoT se puede asimilar a un mecanismo de rastreo de WoT".

Sin embargo, un repositorio central afectará la escalabilidad de cualquier solución. Desde una perspectiva a alto nivel, se pueden clasificar los enfoques como **Activos/Continuos** o **Pasivos/Activados** con base a su esquema de recopilación, de acuerdo con [132]. Y pueden almacenarse localmente en cosas o centralizar en la nube. Pueden utilizar independientemente mecanismos de tipo "empuje/tracción" *push/pull* o por publicación/suscripción. Alternativamente, se puede programar la frecuencia. Algunos ejemplos del mecanismo *push/pull* son las arañas web convencionales, o los rastreadores ajustados que son creados para reducir el alcance de la búsqueda. Por otro lado, el esquema *pasivo/activado* emplea algún mecanismo de predicción que alivia el tiempo entre rastreos, de forma que son programados para equilibrar la sobrecarga de procesamiento requerida versus la precisión esperada [145]. Además, el mecanismo pasivo también puede basarse en eventos y activarse justo cuando sea necesario.

Se destacan las principales propuestas de rastreo para WoT: IOTCRAWLER [45], [120], EASICRAWL [65] y THINGSEEK [117]. Ampliando la discusión sobre rastreo y descubrimiento de [132], donde Tran et al. categoriza el tipo de sistema WoTSE por el tipo de colector como a) rastreador web convencional, b) mecanismo de registro y c) descubrimiento local.

Además, Tran et al. evalúa el WoTSE según su tipo de colección, sea este de tipo virtual o real y su esquema de recolección como activo o pasivo. Se consolida el análisis de propuestas técnicas/algorítmicas relacionadas con el rastreo y el descubrimiento en la Tabla 2.4.

IOTCRAWLER [45] se compone de una capa de rastreo y procesamiento que se ejecuta constantemente en un modelo de publicación/suscripción. Se encarga de integrar nuevas fuentes de datos de diversa índole; sin embargo, se supone que existe un NGSI-LD (un estándar basado en ETSI) para registrar cosas en el repositorio, basado en un modelo de información propuesto llamado concepto *IoTStream*. Otros estilos de arquitectura incluyen [65], que presenta EASICRAWL implementando un mecanismo optimizado de latencia programado activo/continuo que extrae sensores de IoT. Rastrea eventos recientes de sensores de tipo sueño-vigilia periódicamente. En este se asume que se conoce el plan de sueño de los sensores. Shemshadi et al. [117] propone THINKSEEK que sigue un enfoque de varios pasos para leer el contenido, reestructurar los datos y fusionarlos mientras utiliza un algoritmo *MapReduce* y mejora los tiempos de procesamiento. El descubrimiento local se utilizará en el caso de uso de WoT/IoT en el que debe resolverse una consulta sobre un entorno circundante. Kadir et al. [58] propone un mecanismo distribuido para descubrir servicios similares a IoT que se basan en redes sociales con un método de agrupación; el caso de uso se ejemplifica en el contexto de los vehículos conectados, donde un conjunto de avatares trabajarán juntos para tomar una decisión. WoTSE, por Younan et al. [145], utiliza un rastreador web convencional para servicios WoT basado en un archivo raíz y arañas; propuestas similares se encuentran en otros trabajos como [99] y [53]. Sin embargo, los rastreadores tradicionales basados en la web podrían exhibir algunas limitaciones porque las representaciones dinámicas de WoT deberían apoyarse en las operaciones del lado del cliente, es decir, AJAX. Y en este caso, es posible que los rastreadores de WoT no recopilen ningún dato [145].

Finalmente, el más simple, pero menos escalable será un mecanismo descentralizado basado en registro estático utilizado por los usuarios finales para ingresar a las funcionalidades o servicios de WoT [79]. Sciullo et al. [113] creó el registro de servicios basado en el *Arrowhead Interoperability Framework*, lo que permite la interacción entre WoT W3C y las arquitecturas de servicios heredadas. Es posible diferenciar una subcategoría adicional destinada a construir un registro de recursos, por ejemplo, de sensores, por medio de la extensión de los estándares existentes, como se presenta en [105]. [68] presenta GEOCENS para monitorear el mundo físico a escala global; se añade una capa *Encontrabilidad* utilizando un catálogo de recursos a través del registro del sensor. GEOCENS descubre de acuerdo con OGC SOS al rastrear los servicios en cumplimiento del estándar OGC a través de Internet.



## 2.6. Indexación, Estructura de Datos y Estrategias para IoT|WoT

La indexación de WoT es una subtarea crucial en los sistemas RI. En términos generales, un sistema RI convencional realiza dos funciones principales [74]. En primer lugar, la indexación cuyo objetivo es crear un conjunto de estructuras de datos para recuperar de manera eficiente la información entregada a un recurso en forma de colección de documentos, principalmente texto. En segundo lugar, el procesamiento de consultas tiene como objetivo utilizar esas estructuras de datos de índice para producir una lista ordenada de documentos de forma eficiente para una consulta de usuario determinada. Varios trabajos han estudiado las funcionalidades, beneficios y desventajas de las estructuras de datos [76], [69] y [33], en el contexto de IoT y WoT. Las ventajas y desventajas del uso de estructuras de datos se presentan en [33], incluidos enfoques multidimensionales como R-tree, R\*-tree, SR-Tree, X-tree, kD-tree, VA-file y Pirámide. Además, los enfoques de tipo métrico incluyen M-tree, GNAT-tree, GH-tree, mVP-tree, PM-tree, MM-tree y oignon-tree.

Las soluciones de RI suelen basarse en índices estáticos debido a las características convencionales de la conocida Web, por lo que no son adecuadas para el dinamismo de WoT. La indexación dinámica para WoT requiere independencia de datos y escalabilidad asequible. Tran et al. [132] clasifica el tipo de índice como basado en texto, índices espaciales, índices de valores numéricos, mecanismos de agrupamiento, modelos de predicción e índices no especificados. Se pueden identificar diferentes direcciones en las técnicas de indexación dinámica para IoT y WoT tanto para estructuras de datos como para estrategias; algunas propuestas incluyen, (ver Tabla 2.5):

- (a) El índice se basa en un mecanismo centralizado. Tecnologías de bases de datos tradicionales o especializadas, como por ejemplo GeoDB, GraphDB.
- (b) El índice se basa en una especie de Registro, Directorio o Catálogo.
- (c) El índice se implementa como un archivo invertido similar a los motores de búsqueda tradicionales.
- (d) Índices de firma basados en enfoques *hash*.
- (e) Índices de agrupamiento.

Casi todos los índices se basan en algún árbol o modelo *hash*; la principal excepción son las estructuras de datos agrupados [76]. Los árboles definen el orden lexicográfico de los datos, mientras que el *hash* aleatoriza el orden de los datos. Cabe señalar que algunos

tipos especiales de datos requieren estructuras de datos especiales para mejorar la eficiencia y la eficacia de la indexación dinámica. Por ejemplo, se argumenta que la familia R-tree es adecuada para datos geo espaciales [76], [69], R+/MDR+-tree puede acelerar la recuperación de datos espacio-temporales [69], el árbol B clásico se puede usar para metadatos generales, el árbol RtGR para datos de sensores de observación. Liu et al. [69] argumentan que las series temporales y la dimensión temporal inherente de los datos observados no son adecuadas ni para los tipos de árbol KD ni de árbol cuádruple. Algunas estructuras de datos alternativas se estudian en [127], basadas en PKR-tree [126], para indexar datos masivos de IoT basados en partición recursiva con k-clustering en *Cloud-fog computing*, o utilizando STK-tree (palabra clave espacio-temporal) [125], basado en B\*-tree en el sistema CECSE. [95] propone un índice adaptativo que consta de índices espaciales (geo-hashes) y temáticos y temporales (M-GMM).

Ihm et al. [46] propone el uso de un método de horizonte extendido para resolver de manera eficiente las consultas Top-K en IoT. Los métodos de horizonte son métodos de representación para responder a las consultas top-k mediante la construcción del horizonte como un índice. GRID-PPPS se basa en métodos de representación donde los datos se expresan a través de tuplas y luego se convierten en objetos d-dimensionales para finalizar la construcción de un horizonte como índice. Es más probable que estos métodos se utilicen en la implementación de la base de datos SQL de los sistemas de búsqueda de IoT. Por lo tanto, pueden verse como métodos de base de datos avanzados para resolver consultas de tipo numérico en conjuntos de datos de sensores de IoT.

Fathy et al. [31] argumentan que es necesario un mecanismo dinámico, distribuido, eficiente y escalable. Sin embargo, también se están proponiendo algunos esquemas de índices centralizados. Una estrategia de indexación dinámica en línea debe actualizarse y reorganizarse regularmente mientras los datos y flujos vinculados se generan continuamente desde la capa de datos de IoT. Desde el punto de vista del diseño, el Indexador debe implementar una estructura de datos eficiente, un esquema de distribución adecuado y una estrategia de mantenimiento óptima. El esquema de distribución apunta a construir **Multi-Índices** o **Índices Híbridos** para mejorar la escalabilidad de las soluciones WoTSE e IoTSE o proporcionar riqueza y expresividad en las consultas, en consecuencia. Otro ejemplo es CESCE cuyo algoritmo central propone crear un índice de filtrado de palabras clave espacial-temporal, denominado SKIN-tree, y una salida predeterminada con STK-tree, basada en B+ tree. En contraposición, [152] y [30] proponen dos categorías para índices únicos: los que se basan en la indexación de texto convencional y los que se basan en la indexación espacial. Sin embargo, es esencial tener en cuenta que las necesidades de los usuarios sobre

la información de IoT y WoT cruzarán esos límites binarios en la información textual y espacial a la información espacio-temporal, temática y en cuasi tiempo real.

La indexación en tiempo real es un componente central en los sistemas RI y en los servicios de búsqueda de cualquier tipo. Los desafíos son considerables: la eficiencia, la escalabilidad y el rendimiento son factores a considerar al diseñar y/o construir un índice para sistemas WoT, dadas las características multidimensionales de los datos expuestos por las cosas inteligentes.

### 2.6.1. Indexación en *Casi* Tiempo Real y Latencia Cero

Múltiples enfoques han propuesto la denominada indexación en tiempo casi real (*Near Real-Time Indexing*, NRTI por sus siglas en inglés). Una colección de técnicas que mejora la frescura del índice y el tiempo de búsqueda. Sin embargo, la concepción de lo que significa tiempo real y casi tiempo real puede variar. Hoy en día existe también la denominación de indexación de latencia cero.

En algunos escenarios, NRTI reduce significativamente el tiempo de procesamiento en cola antes de la indexación, el momento en que el rastreador envía un segmento nuevo, modificado o eliminado. Algunas iniciativas incluyen la fragmentación al crear pequeñas porciones temporales para recibir documentos indexados, nuevos, modificados o eliminados. Se denomina a esta técnica *Segmentación de Documentos*. Por el contrario, en el enfoque de Apache Solr <sup>12</sup>, NRT se entiende como un tiempo mínimo para la capacidad de búsqueda después de que el documento ha sido indexado. El mecanismo utilizado por SOLR no bloquea las actualizaciones mientras hay una confirmación en curso. Además, se basa en la optimización del temporizador, utilizando una combinación de operaciones de escritura duras y suaves sobre disco. La optimización se realiza definiendo el número máximo de documentos en la cola y la cantidad de milisegundos a esperar antes de enviarlos al índice, denominadas como *Artefactos de Optimización*.

De manera similar, el enfoque de Elasticsearch <sup>13</sup> se basa en el desarrollo del llamado *Per-Segment Search Mechanism* y se basa en Lucene<sup>14</sup>. El objetivo es reducir el retardo entre la indexación de un documento y hacerlo visible para la búsqueda. Teniendo en cuenta que el principal cuello de botella son las operaciones del disco, la solución es utilizar un mecanismo más ligero para hacer visibles las nuevas actualizaciones. Se ejecuta en un segundo sin un comando de sincronización. Los documentos en el búfer de indexación se escriben en un nuevo segmento, que se puede buscar, pero aún no se ha confirmado. El nuevo segmento se

---

<sup>12</sup><http://lucene.apache.org/solr>

<sup>13</sup><https://www.elastic.co/>

<sup>14</sup><https://lucene.apache.org/>

escribe primero en un sistema de caché y solo después se envía al disco. Asimismo, Terrier<sup>15</sup> es un motor de búsqueda de código abierto muy reconocido, flexible, eficiente y efectivo. Está escrito en Java y fue desarrollado en la Universidad de Glasgow. Presenta un conjunto de técnicas: múltiples estructuras de índice en tiempo real con una estructura híbrida (un índice de caché o memoria diseñado para proporcionar una estructura actualizable rápidamente). El *Índice Incremental*, se implementa como un Índice Múltiple, donde un índice está en la memoria y los otros en el disco.

De la familia Apache, se puede destacar a FLUME SOLR SINK<sup>16</sup> como un sistema de tiempo casi real para procesar flujos continuos en índices de búsqueda en vivo. La latencia desde el momento de la llegada de los datos hasta el momento en que aparecen los datos en los resultados de la consulta de búsqueda es del orden de segundos. Los datos fluyen desde una o más fuentes a través de uno o más nodos FLUME SOLR SINKS dentro de un *Sistema Distribuido*. Una característica particular es que se puede indexar cualquier formato de datos. En el caso de SPHINX<sup>17</sup>, permite definiciones tanto de índices en tiempo real como de índices respaldados por SQL. El enfoque en tiempo real utiliza *SphinxQL* de SPHINX y se administra a través de Ruby, donde los campos y atributos hacen referencia a los métodos de Ruby. Por lo tanto, los registros en tiempo real pueden usar *Actualización directa con métodos*, por lo que los datos de SPHINX se pueden buscar casi de inmediato, lo que elimina el requisito de índices incrementales o delta.

Actualmente, la indexación de latencia cero ha sido acuñada por el sistema MARKLOGIC<sup>18</sup>. Este tipo de sistema propone la indexación de datos estructurados y no estructurados en el próximo ciclo de CPU. El sistema utiliza una base de datos no relacional (NoSQL), considerando la velocidad y los volúmenes de datos recopilados de una gran variedad. Tiene capacidades ricas que combinan búsqueda estructurada y no estructurada con aspectos geo-espaciales y temporales. Tiene una estructura híbrida, que indexa información textual y numérica de toda la colección mediante una técnica de *Triple Index*, que combina enriquecimiento semántico, visualización gráfica mediante SPARQL mientras utiliza documentos XML y JSON para el modelado de datos.

---

<sup>15</sup><http://terrier.org/>

<sup>16</sup><https://flume.apache.org/>

<sup>17</sup><http://sphinxsearch.com/>

<sup>18</sup><http://www.marklogic.com>

## 2.7. Lenguajes y Procesamiento de Consulta para IoT|WoT

### Lenguajes de consulta e intérpretes

Algunos de los precursores de los sistemas basados en RI han enfatizado los componentes para rastrear, descubrir, indexar, consultar y recuperar entidades de sensores y/o sus datos proporcionados (de tipo espontáneo, de flujo o en rango), establecer un middleware y/o proporcionar un enriquecimiento semántico. Estas capacidades de búsqueda de entidades de sensores permiten la estructura de consultas multidimensionales complejas que emplean un índice basado en árboles distribuidos sobre redes de sensores [66] o redes de sensores semánticos. El esfuerzo más importante de estos precursores radica en las capacidades de búsqueda de sensores y sus datos, en donde el procesamiento distribuido es visto como un elemento crítico en la mayoría de las propuestas junto con el desarrollo de SPARQL.

En la denominada Web de Datos, dentro de las iniciativas de la Web Semántica, integró el paradigma de los **Datos Vinculados** o *Linked Data* en la Web. La idea es estructurar, compartir e interconectar grandes volúmenes de datos para permitir razonamientos y consultas semánticas. Pronto, este mecanismo de datos vinculados se convierte también en parte de los enfoques de la Web de Sensores como en [105], [68] que presentan la propuesta de **Datos de Sensores Vinculados** para publicar y compartir datos ambientales en la nube de datos abiertos y vinculados. Similarment, en GEOCENS [68], se puede habilitar un mecanismo de consulta única para los datos de los sensores utilizando la representación de datos y objetos O&M basada en (SWE, *Sensor Web Enablement*). El sistema puede resolver múltiples consultas sobre los datos del sensor: de tipo rango, distancia y ventana de tiempo.

La estandarización ha sido una brújula en la mayoría de las propuestas en la literatura. SPARQL, sus extensiones y derivados juegan un papel en la WoT Semántica para habilitar un ecosistema IoT|WoT integrado. Sin embargo, no hay un consenso sobre las ontologías que se utilizarán, y todavía se necesita un mecanismo para proporcionar a los usuarios finales una interacción centrada en el ser humano con WoT. SPARQL permite consultas de bajo nivel, pero para consultas de alto nivel, se debe proporcionar el uso de lenguaje natural u otro mecanismo. A continuación se propone una clasificación simple de los intérpretes de consulta, según su lenguaje de consulta y sus capacidades, como:

- **De bajo nivel:** aquellos capaces de proporcionar un mecanismo de búsqueda coherente y sencillo para propuestas basadas en ontologías RDF en las capas de sensores y datos. En donde se agrupa SPARQL, sus derivados y extensiones, RDF<sup>19</sup> Data Query Language (RDQL), o adaptaciones, y Semantic Web Rule Language (SWRL)<sup>20</sup>.

<sup>19</sup><https://www.w3.org/Submission/RDQL/>

<sup>20</sup><https://www.w3.org/Submission/SWRL/>

- **De alto nivel:** proporcionan riqueza en la especificación de las expectativas del usuario y las necesidades de información. Algunos pueden recibir la consulta en lenguaje natural, mientras que otros brindan un lenguaje sintético alternativo. Se han propuesto nuevos enfoques durante la madurez de la investigación IoTSE/WoTSE. Como por ejemplo, [26] propone una consulta de IoT-WS como una tupla que consta de distancia, tiempo y funcionalidad. Proporciona un mecanismo simple, pero, por el contrario, puede restringir la expresividad y la riqueza de la consulta en sí.

### **Consultas de resolución múltiple**

Tang et al. [127], [126] afirman que la mayoría de trabajos anteriores se centran en la búsqueda con base en palabras clave con solo entidades WoT, dispositivos IoT y sensores como ámbito de búsqueda. De este modo, proponen SMPKR para dirigir consultas de tipo palabra clave y espacio-temporales sobre en una estructura de índice de árbol PKR. Sobre él, [125] propone CECSE un WoTSE colaborativo con base en un sistema de caché en la nube, de forma perimetral y para objetos móviles inteligentes. Sin embargo, todavía hay preguntas abiertas sobre la integración de datos y mecanismos para aprovechar una búsqueda universal en IoT/WoT. CECSE se basa en una arquitectura de búsqueda basada en caché de tres niveles con la finalidad de adaptarse a consultas frecuentes: caché en el nivel de dispositivo, nivel de borde y nivel de nube.

Diamantini et al. [22] propone un algoritmo basado en el contexto, multirresolución y multigranularidad para consultar el IoT. En este enfoque, los contextos se pueden ajustar o relajar en función de las necesidades de información del usuario o de los objetivos de la aplicación. Diamantini et al. [22] argumentan que la proximidad, la homogeneidad, la propiedad, el trabajo conjunto y la sociabilidad se utilizan para proporcionar características autónomas a IoT. Desde una perspectiva arquitectónica de IoTSE, su trabajo exhibe algunas características similares con el enfoque de múltiples estrategias en Ma-Liu et al. [73]. Sin embargo, Diamantini et al. se centra en los modelos de contexto (conjunto de dimensiones de búsqueda con diferente granularidad), los mecanismos de consulta de resolución múltiple y los modelos de ejecución de consulta múltiple. En cambio, el segundo, Ma-Liu se centra en la progresividad de la búsqueda.

### **Procesamiento de consultas y modelos de predicción**

IoT/WoT proporciona información adicional sobre el mundo real con tanta riqueza como nunca antes. A pesar de los avances en los modelos de predicción para buscar estados y datos, persisten algunas dificultades debido a la precisión limitada de la predicción y la sobrecarga

media relativa. Zhang et al. [148] presenta un mecanismo de búsqueda de sensor de modo dual que considera tanto el estado actual como el histórico de las entidades. Además, [148] propone un nuevo modelo de predicción basado en el aprendizaje profundo que mejora la precisión de los modelos anteriores en [82], [83]. Otros enfoques recientes apuntan a explorar la búsqueda de sensores de datos, como en [70] orientados a la combinatoria. Los modelos de predicciones tienen como objetivo reducir los gastos generales en términos de procesamiento de consultas y consumo de recursos de computación, almacenamiento y red. De manera similar, otros métodos de reducción de datos pueden aprovechar otros mecanismos eficientes para indexar y consultar.

Hatcher et al. [41] exploran técnicas de aprendizaje profundo utilizando la arquitectura de red neuronal de memoria a largo y corto plazo (LSTM) para predecir consultas agregadas. Proponen un marco genérico para el motor de búsqueda de IoT basado en un trabajo anterior en [67], combinándolo con un servicio de nombres basado en DNS para dispositivos y servicios de IoT. Este trabajo proporciona nuevos conocimientos para la evolución de la búsqueda de IoT mientras explora nuevas alternativas para realizar la predicción. Mietz et al. [83], [82] continúan el trabajo realizado con DYSER, brindando alternativas para modelos de predicción utilizando Redes Bayesianas y mecanismos de correlación para la búsqueda de sensores.

Los modelos de predicción tienen como objetivo encontrar una manera de reducir la cantidad de sensores a leer y, por lo tanto, reducir el consumo general de recursos, con el objetivo principal de optimizar los tiempos de consulta y búsqueda. Algunos sistemas intentan responder a la consulta directamente a partir de los datos del sensor previstos; otros usan los modelos de predicción para estimar estados futuros en lugar de datos de sensores. De esa forma, recuperan un subconjunto de sensores que pueden cumplir con los criterios ingresados por el usuario final y luego responden la consulta consultando los datos en tiempo real del subconjunto de sensores.

## 2.8. Recuperación en IoT|WoT

En este trabajo se propone una clasificación de enfoques basada en el tipo de ordenamiento. Cabe destacar un par de obras de la última década, desde las primeras investigaciones, como DYSER [94] se explora y propone mecanismos de clasificación de sensores utilizando modelos de predicción, con la idea de mejorar el rendimiento de búsqueda en lugar de la satisfacción del cliente. SNOOGLE [140] emplea un algoritmo de ponderación *tf-idf* convencional para calificar la relevancia de los objetos de IoT en el momento de la recuperación. De la misma manera, Microsearch [124] usa *tf-idf* para determinar puntajes y devolver una lista

ordenada de *top-k*. LIVEWEB [143] utiliza un modelo booleano clásico para la recuperación. Recientemente, Mili-Rodin [116] presentan WOTMAS2E un WOTSE basado en múltiples agentes, donde el proceso de ordenamiento se basa en la probabilidad de que un sensor se encuentre en ciertos estados, ponderados por el número de entidades y estados, donde el proceso se integra directamente al agente de indexación.

Sin embargo, en general, no existe un enfoque de investigación a los mecanismos de ordenamiento en la literatura en general. [31] define la recuperación en IoT como la priorización de recursos o servicios de IoT en función de criterios funcionales o no funcionales y presentan una revisión de las estrategias de ordenamiento donde la calidad de la información (QoI) y el valor de la información (VoI) son los principales criterios utilizados por la mayoría de los enfoques. Además, es importante destacar la diferencia entre el ordenamiento de tipo interno (comúnmente empleado para generar listas ordenadas de sensores) y el ordenamiento clásico con el objetivo de aumentar la relevancia de la información entregada al usuario final por parte de los motores de búsqueda. El primero intenta optimizar el rendimiento del motor de búsqueda, mientras que el segundo trata de optimizar la satisfacción del usuario. Perera et al. [102], [101] proponen una búsqueda basada en un método de selección de priorización para sensores semánticos donde los usuarios pueden expresar consultas por parámetros del sensor como confiabilidad, precisión, ubicación y duración de la batería. Luego, los sensores se clasifican de acuerdo con un índice ponderado basado en la prioridad comparativa (CPWI), creado a partir de una estimación de similitud de las preferencias del usuario.

Se han propuesto docenas de puntuaciones personalizadas o *multicriterio*. [26] calcula la *distancia de similitud funcional* entre los servicios de IoT y la consulta del usuario. WOTSF [145] propone un rango de estado de entidad. [101] usa la Distancia Euclidiana en *tf-idf*. [8] define una función de aptitud compuesta por la distancia espacial, la integridad y los parámetros técnicos y funcionales. [89] define un enfoque de similitud múltiple con medidas de contenido, tipo, semántica y similitudes globales. [137] utiliza una puntuación de similitud de sensor basada en lógica difusa. [19] utiliza la vida útil de los eventos correlacionados en el tiempo para ordenar los resultados. Recientemente, Perreira et al. [95] proponen una estrategia de recuperación de criterios múltiples para ayudar en la selección de recursos y servicios de IoT. Utiliza una definición de calidad de información multi-métrica que consta de integridad, puntualidad, plausibilidad, artificialidad y concordancia, que en conjunto producen una puntuación de calidad de experiencia (QoE) ponderada estimada por la aplicación de consumidor de IoT/WoT para ordenar los resultados. [137] describe un mecanismo de clasificación por asociación usando un algoritmo que explota conjuntos difusos para calcular una puntuación de similitud. En general, tampoco existe un consenso en la estimación de



relevancia, ni en las métricas a ser incorporadas dentro de las puntuaciones para ordenar sensores, cosas o entidades en IoT|WoT.

## 2.9. Presentación (UI/UX) para IoT|WoT

Dado el universo de investigación en WoTSE, se han considerado diferentes estrategias de presentación de resultados con diferentes formatos y alcances que podrían complementar las estrategias de presentación primarias de los sistemas RI tradicionales de cara a los usuarios finales: con interacciones humano-a-maquina (H2M *Human to Machine*) e interacciones maquina-a-maquina (M2M *Machine to Machine*). [132] evalúa la capa de presentación con las siguientes dimensiones: tipo de usuario, modalidad de interfaz (tipo de interacción), interfaz de consulta, interfaz de resultado. El primer enfoque es utilizar navegadores web populares para navegar por IoT/WoT. Las interfaces genéricas que provienen de los navegadores web estándar pueden funcionar en múltiples ecosistemas de IoT, pero, de hecho, son propietarias, lo que restringe la interoperabilidad entre ellos. [93] propone la Búsqueda Visual para IoT.

Un porcentaje significativo de los enfoques de IoTSE implementan una interfaz de usuario basada en mapas para navegar geográficamente. En esta etapa, se distingue entre la interfaz de usuario de consultas, la interfaz de usuario de resultados y las interfaces duales integradas. Como mecanismos de interfaz de usuario de entrada, la interfaz de usuario de consulta puede usar un formulario web clásico estructurado o una interfaz basada en mapas para consultas, así como con reconocimiento de ubicación, mientras que otros han agregado campos de formulario con reconocimiento de tiempo para capturar rangos o proporcionar un filtrado estricto con fecha y hora o incluir el tipo temático de parámetros. Según los resultados de salida, la interfaz de usuario utiliza convencionalmente una lista de resultados, por lo que se pueden mostrar en enfoques basados en mapas para localizar los resultados de la consulta.

Para las interacciones M2M, existe una intención común de utilizar interfaces programáticas API tipo RESTful como un estándar de facto en los sistemas WoTSE [52], [53], [16]. [87] utiliza interfaces RESTful, que brindan interacciones H2M/M2M y un mecanismo de descripción semántica basado en JSON para cosas inteligentes. Otros trabajos creen que el uso de API especializadas siguiendo el estilo arquitectónico como SOAP [72], es un mejor enfoque que la API RESTful. Finalmente, el enfoque basado en RDF API en [100] se puede destacar como una forma de manipular información semántica en gráficos RDF.

## 2.10. Enriquecimiento semántico para IoT|WoT

SPITFIRE [103] es uno de los pioneros en proporcionar un enriquecimiento semántico completo para el WoT semántico. Este motor de búsqueda proporciona una forma de encontrar tanto sensores como entidades abstractas dada una consulta basada en estados. Internamente, proporciona un vocabulario para integrar datos abiertos y vinculados, una ontología para describir entidades como cosas abstractas y sensores y, finalmente, un mecanismo para una descripción semiautomática de sensores. Una característica novedosa es el modelado semántico de las cosas como entidades y estados. Se pueden reconocer dos direcciones claras de mejora semántica de entidad y estado por separado. Por otro lado, [82] introduce un modelo semántico específico para la búsqueda en IoT; retoma algunos conceptos previos explotados en DYSER, como los modelos de predicción para datos de sensores (pero enriquecidos semánticamente empleando RDF), donde explota un modelo de sensor base, sus estados, su historia, el propio modelo de predicción del sensor y su ubicación. A partir de este punto, es posible reconocer diferentes modelos de vocabulario para mejorar los escenarios IoT/WoT y apoyar la búsqueda con sistemas IoTSE/WoTSE a través de relaciones usando RDF, RDFa, OWL, OWL-DL o OWL-S.

De manera similar a trabajos anteriores, [77] presenta múltiples estrategias para descubrir cosas inteligentes habilitadas semánticamente para la Web, construyendo el llamado DISCO-WoT. En [78] el mecanismo de búsqueda se extiende a través de una vista orientada a los recursos, un mecanismo de equilibrio de carga combinado con un módulo de almacenamiento en caché de consultas. Posteriormente, se encuentra el trabajo de [87] basado en la abstracción Web Avatar, que presenta un mecanismo de descubrimiento semántico exhibiendo e invocando funcionalidades de las cosas. Un enfoque inspirador y sólido es dividir las vistas del usuario y el sistema en torno al modelo de descripción de cosas propuesto. Denota *capacidades*, como las acciones físicas que los objetos IoT pueden realizar que se reconocen como relaciones semánticas Entidad-Capacidad, y *funcionalidades*, especificaciones de alto nivel que los avatares web pueden exponer en la Web que se reconocen como Relaciones semánticas entidad-funcionalidad, ambas relaciones son bloques de construcción para el desarrollo de aplicaciones WoT [87]. Christophe et al. [14] propone un proceso de búsqueda de tres pasos para encontrar cosas inteligentes habilitadas para la Web según algunos requisitos o expectativas del usuario. También ofrece un enriquecimiento a través de perfiles semánticos para establecer similitud y estimar un algoritmo de puntuación para seleccionar una consulta específica. [17] propone un modelo OWL-DL visto como un ecosistema IoT, que incluye conceptos como servicios, recursos, entidades y dispositivos, todos extendidos con sus relaciones. Cada concepto está minuciosamente construido e interrelacionado. El

llamado SENSE2WEB proporciona una interfaz H2M para realizar consultas utilizando SPARQL con un fuerte enfoque en la búsqueda basada en la ubicación complementada con una combinación de mapas de Google. [53] propone WOTS2E, un motor de búsqueda para la Web Semántica de las Cosas. Se basa en descubrir puntos finales de datos vinculados heterogéneos y mezclar las ontologías SSN y OWL-S para un mecanismo basado en el registro que registra los recursos y servicios de WoT.

El enriquecimiento semántico es una tendencia para superar los desafíos de interoperabilidad en IoT/WoT. [16] presenta EWOT, que se basa en el modelo W3C Thing Description (TD) pero se amplía a través del enriquecimiento semántico obtenido a través de la traducción a tripletas RDF. En [16], el TD estándar se describe como carente de expresividad para lidiar con la heterogeneidad de IoT. Cimino et al. [16] implementan un repositorio semántico centralizado mientras distribuye el acceso a los dispositivos de IoT, evitando la compatibilidad con SPARQL en los puntos finales de IoT. La literatura utiliza una mezcla de ontologías que van desde tesauros múltiples y genéricos [98] hasta FOAF<sup>21</sup>. FOAF también se puede utilizar para asociaciones sociales entre propietarios de datos/sensores/objetos y usuarios finales relacionados para reducir el espacio de búsqueda o el alcance al hacer frente a la considerable cantidad de datos producidos por los sensores. SSN Ontology es una ontología muy conocida y utilizada en [91], [55], [82], también mezclada con OntIoT Ontology-based Domain Specific Language, o la O&M Ontología de observación de sensores. Otras ontologías utilizadas en trabajos relacionados con IoTSE y WoTSE son: Dolce Ultralite<sup>22</sup>, GeoName<sup>23</sup>, Phenonet Open IoT Ontology<sup>24</sup>.

## 2.11. Evaluación de Recuperación IoTSE|WoTSE

La medición de la efectividad de un sistema de RI depende en su mayoría de las evaluaciones humanas de utilidad de la información encontrada y su relevancia. La RI como una disciplina altamente empírica requiere una evaluación cuidadosa y exhaustiva para demostrar el rendimiento de sus modelos [74]. Son varias las medidas de interés que se pueden relacionar con la calidad de las respuestas, en términos de la eficiencia y la eficacia de la RI para IoT|WoT. Tradicionalmente, a) la precisión y b) la exhaustividad (*Recall*) son métricas de calidad usadas en RI y otros campos afines. Desde un punto de vista más técnico, c) la velocidad de respuesta y d) el tamaño del índice son factores que pueden aumentar la calidad de experiencia de un usuario frente a un sistema RI.

<sup>21</sup><http://www.foaf-project.org/>

<sup>22</sup><http://ontologydesignpatterns.org/wiki/Ontology>

<sup>23</sup><https://www.geonames.org/ontology/documentation.html>

<sup>24</sup><https://www.w3.org/2005/Incubator/ssn/ssnx/meteo/phenonet>

La importancia de encontrar métodos de búsqueda eficaces y efectivos para las aplicaciones WoT ha sido identificada por [152] como un compromiso de todas las etapas de RI. Por ejemplo, [152] ha propuesto múltiples variables cualitativas como métricas para evaluar las técnicas de búsqueda en aplicaciones IoT|WoT: Tiempo de consulta, Precisión de consulta. En términos del tiempo de respuesta del sistema RI (que cuenta envío de una consulta y la recepción de los resultados), así como la precisión de los resultados obtenidos. Así es que en la aplicación de RI en IoT|WoT también es de vital importancia los métodos de evaluación considerando sus múltiples dimensiones.

### 2.11.1. Evaluación RI Clásica por Colección de Prueba

Para medir la efectividad de RI de forma ad-hoc tradicionalmente se tiene una colección de documentos, un conjunto de pruebas de necesidades de información, expresable como consultas, y un conjunto de juicios de relevancia, normalmente una evaluación binaria de *relevante* o *no relevante* para cada par consulta-documento. Ha sido ampliamente identificado como un desafío y una necesidad actual la disponibilidad de colecciones de prueba dinámicas enfocadas a los paradigmas IoT|WoT [131]. Ninguno de los principales foros de evaluación RI como TREC<sup>25</sup>, NTCIR<sup>26</sup>, o FIRE<sup>27</sup> contiene alguna colección específica para los paradigmas de IoT|WoT.

En RI, es posible determinar la efectividad de un sistema a un conjunto de tópicos empleando una colección de prueba en conjunto con su respectivo juicio especializado de relevancia [74]. A partir de los resultados y el juicio se determinan: Precisión (P) como la fracción de documentos recuperados que son relevantes, *Recall* (R) como la fracción de documentos relevantes que se recuperan y (F) como la media armónica ponderada entre precisión y recuperación:

$$P = \frac{TP}{TP + FP} \quad (2.2)$$

$$R = \frac{TP}{TP + FN} \quad (2.3)$$

$$F(1) = \frac{2 * TP}{2 * TP + FP + FN} \quad (2.4)$$

Donde TP representa los verdaderos positivos, FP los falsos positivos, FN los falsos negativos y TN los verdaderos negativos. Estos pueden ser ajustados al tipo de esquema de recuperación

<sup>25</sup>Tracks de Evaluación en TREC - <https://trec.nist.gov/data.html>

<sup>26</sup>Tracks de Evaluación en NTCIR - <http://research.nii.ac.jp/ntcir/data/data-en.html>

<sup>27</sup>Tracks de Evaluación en FIRE - <http://fire.irsi.res.in/fire/static/data>

y existen variaciones como la Precisión en K, R-Precisión y la Precisión Promedio Media, (*Mean Average Precision*, MAP de sus siglas en inglés).

### 2.11.2. Marcos de Evaluación, Datasets y Colecciones de Prueba para IoT|WoT

Como parte de una revisión sistemática de literatura (RSL) cuyos resultados se presentaran en la siguiente sección, se ha extendido este RSL para abarcar un quinto grupo de preguntas relacionadas con la evaluación de sistemas RI en el campo IoT|WoT.

- ¿Cómo las soluciones propuestas en los estudios analizados del RSL han evaluado el rendimiento de los sistemas IR-IoT e IR-WoT?
  - ¿Qué medidas de rendimiento en tiempo se han empleado?
  - ¿Qué medidas de tamaño de índice se han empleado?
  - ¿Qué medidas de evaluación clásica como precisión, exhaustividad y medida F se han empleado?
  - ¿Existen *datasets* o datos abiertos para la reproducibilidad de los experimentos o la reusabilidad de los datos?

La tabla 2.7 lista los principales marcos de evaluación analizados en el contexto de propuestas de sistemas RI o motores de búsqueda en IoT|WoT. La mayoría de *datasets* curados en la esfera IoT que se han encontrado tienen un enfoque hacia la aplicación a un campo específico como en la prueba de mecanismos de seguridad. Tal es el caso del *dataset* BoT-IoT<sup>28</sup>. Este contiene alrededor de 72 millones de registros con la respectiva categorización y/o relevancia automática de tipo de ataque y otros detalles técnicos. Otra fuente de información consultada es el sistema de búsqueda de datos de Google, disponible en <https://datasetsearch.research.google.com/>. Donde se pueden encontrar algunos pocos *datasets* creados por recientes investigaciones en el campo WoT y RI de forma independiente.

En la evaluación de IoTSE presentada por [130], los IoTSE se sometieron a una consulta de sensores habilitados para IoT que miden la "temperatura aparente" en "grados Celsius". Todas las instancias lograron una precisión y exhaustividad perfectas, debido a la utilización de una base de datos para almacenar y resolver las consultas sobre los metadatos del sensor. El diseño de métodos de evaluación de eficiencia en la implementación y en el uso de la búsqueda de IoT es un tema de vital importancia [41]. Hatcher et al. propone que la eficiencia del sistema de búsqueda en IoT se desarrolle desde dos perspectivas, aumentando la efectividad

<sup>28</sup>Dataset BoT-IoT - <https://research.unsw.edu.au/projects/toniot-datasets>

en la búsqueda de cada componente en tiempo de procesamiento y la escalabilidad en las interacciones entre etapas RI aumentando el "*Throughput*" o capacidad de procesamiento dada en consultas por segundo.

Una importante mayoría de propuestas IoTSE|WoTSE basan la evaluación de rendimiento y desempeño del sistema en términos de la complejidad real en tiempo y espacio. Las medidas de tiempo asociadas al procesamiento del índice, de consultas o del todo las etapas de recuperación hasta la entrega de resultados al usuario final. Y las medidas de espacio asociadas al tamaño del índice o al almacenamiento empleado.

Recientemente, [15] reporta el resultados de algunos experimentos llevados a cabo para el descubrimiento semántico de WoT, enmarcados en la iniciativa WoT W3C. Este emplea las definiciones formales TD y hace el uso de diez consultas de tipo SPARQL para analizar el tiempo de consulta en función del número de cosas (1-1000 TDs)<sup>29</sup>. Otro *dataset* abierto relativo a la iniciativa W3C es el liberado por Chapernay et al. en [9]. Este trabajo parte del modelo de descripción de cosas W3C TD, y propone la construcción de una ontología (W3C TD Ontology)<sup>30</sup> para el modelamiento del mundo real a partir de una colección de cosas virtuales. El *textitdataset* liberado comprende menos de una centena de cosas modeladas en JSON, con sesenta y cinco cosas en total.

---

<sup>29</sup>Dataset y Datos Abiertos de Experimentos - <https://doi.org/10.5281/zenodo.6674151>

<sup>30</sup>Draft Ontología de Descripción de Cosas - <https://www.w3.org/2019/wot/td>

Año	Autores	Referencia	Nombre del Sistema	WoTSE   IoTSE	Tipo de Datos	Tipo de Sensores	Tipo de Cosas	Contexto	Subtarea IR Predominante
2022	Hatcher et al.	[42]	SecureIoT	IoTSE	STD   DYD	PSN	ND	SEC	
2022	Mili-Rodin	[116]	WoTmAS2E	WoTSE	STD   DYD	PSN	PTH   AGT	CTX	Ranking and Retrieving
2022	Pavlopoulou-Curry	[98]	PoSSUM	Middleware	STD   DYD	ND	ND	ND	Semantic Enrichment
2021	Iggena et al.	[45]	IoT-Crawler	IoTSE	STR   LDA	VSN	ND	ND	Crawling and Discovery
2021	Nandan et al.	[109]	IoT-CBSE	IoTSE	STD   DYD	PSN	PHT	ND	Semantic Enrichment
2020	Sciullo et al.	[115]	AHF-WoT	WoTSE	STD   DYD	VSN	VTH   TD	CTX	Crawling and Discovery
2020	Tang et al.	[128]	CESCE	IoTSE	STD   DYD	PSN	PHT	ND	Query Processing
2020	Yang et al.	[142]	PSSPP	IoTSE	DYD	PSN	ND	ND	Security, Privacy and Trust
2019	Pattar et al.	[97]	ProSA	WoTSE	DYD	PSN	PHT	CTX   SRV	Crawling and Discovery
2019	Sciullo et al.	[113]	WoT Store	WoTSE	STD   DYD	VSN	VTH   TD	CTX   APP	Crawling and Discovery
2019	Silva et al.	[119]	SWoTPAD	WoTSE	DYD	ND	ND	ND	Semantic Enrichment
2019	Tang et al.	[127]	SMPKR	IoTSE	STD   DYD	PSN	PHT	ND	Query Processing
2018	Karim et al.	[56]	DESERT	Component	STR	ND	ND	ND	Query Processing
2018	Ma-Liu	[73]	PROVINS	Component	DYD   STA   IMG	PSN	ND	ND	Search Space
2018	Skarmeta et al.	[120]	IoT-Crawler	Component	DYD   STR	PSN	ND	ND	Crawling and Discovery
2017	Shemshadi et al.	[118]	ThinkSeek	WoTSE	DYD   STR	PSN	ND	ND	Crawling and Discovery
2016	Nunes et al.	[93]	VisIoT	Component	DYD	ND	ND	ND	Crawling and Discovery
2016	Qin et al.	[108]	CEIoT	Middleware	STD   DYD	PSN	PHT   AGT	CTX   SOC	Data Integration
2016	Shemshadi et al.	[117]	ThinkSeek	WoTSE	DYD   STR	PSN	ND	ND	Crawling and Discovery
2016	Younan et al.	[145]	WoTSE	WoTSE	STD   DYD	VSN	VTH	ND	Indexing
2015	Ebrahimi et al.	[27]	AntClust	IoTSE	STD   DYD	SSN	ND	ND	Search Space
2015	Gomes et al.	[38]	ForwardDS-IoT	IoTSE	SAT	SSN	ND	CTX   SRV	Middleware
2015	Li et al.	[65]	EasiCrawl	Component	DYD   STR	PSN	ND	ND	Crawling and Discovery
2015	Soldatos et al.	[121]	OpenIoT	IoTSE	STD	SSN	ND	ND	Middleware
2014	Kolcun-McCann	[59]	Dragon	Component	STD	PSN	ND	ND	Query Processing
2014	Michel et al.	[81]	Gander	IoTSE	STD   DYD	PSN	PHT	ND	Query Processing
2014	Perera et al.	[101]	CASSARAM	IoTSE	STD   STR	SSN	PHT	ND	Ranking and Retrieving
2013	Jara et al.	[48]	MobDigcovery	Component	DYD	PSN	ND	ND	Crawling and Discovery
2012	Ding et al.	[23]	IoT-SVK	IoTSE	STD   DYD	ND	ND	ND	Indexing
2012	Jara et al.	[49]	ImDNS-SD	Component	DRC	VSN	PHT	ND	Crawling and Discovery
2012	Le-Phuoc et al.	[63]	LSM	Component	DYD   SAT	ND	ND	ND	Middleware
2012	Pintus et al.	[104]	Paraimpu	IoTSE	DYD	PSN	ND	ND	Recommender
2012	Qian-Che	[107]	ISE	IoTSE	STD	ND	ND	ND	Security, Privacy and Trust
2011	Komatsuzaki et al.	[60]	IteMinder	IoTSE	STD	ND	ND	ND	Search Space
2011	Le-Phuoc et al.	[62]	CQELS	Middleware	STR   LDA   SAT	ND	ND	ND	Middleware
2011	Mayer-Guinard	[77]	DiscoWoT	WoTSE	STD	ND	VTH	ND	Crawling and Discovery
2011	Pfisterer et al.	[103]	SPITFIRE	WoTSE	DYD   SAT	PSN   VSN	ND	ND	Semantic Enrichment
2010	Ostermaier et al.	[94]	Dyser	WoTSE	DYD	VSN	VTH	ND	Indexing
2010	Tan et al.	[124]	Microsearch	IoTSE	STD	ND	ND	ND	Crawling and Discovery
2010	Wang et al.	[140]	Snoogle	IoTSE	STD	PSN	PHT	ND	Crawling and Discovery
2009	Elahi et al.	[28]	SensorRanking	Component	STD   STA	VSN	VTH	ND	Ranking and Retrieving
2009	Henson et al.	[43]	SemSOS	IoTSE	SAT	VSN   SML	ND	ND	Semantic Enrichment
2009	Jirka et al.	[50]	OSIRIS	Component	DYD	VSN   SML	ND	ND	Crawling and Discovery
2008	Frank et al.	[34]	OCH	IoTSE	STD	ND	PHT	ND	Query Processing
2007	Kansal et al.	[54]	SenseWeb	Middleware	STD	SWB	ND	ND	Middleware
2007	Nath et al.	[90]	SensorMap	Component	DYD	PSN	ND	ND	Crawling and Discovery
2005	Yap et al.	[144]	MAX	WoTSE	STD	ND	PHT	CTX	Query Processing
2003	Li et al.	[66]	DIM	Component	DYD	PSN	ND	ND	Query Processing

Tabla 2.6 Recopilación de trabajos clave en IoTSE|WoTSE 2002-2022.

Uso de acrónimos en la Tabla 2.6:

- Tipo de Datos: Datos estáticos denotados como (STD) por *Static Data*; Datos Dinámicos (DYD) por *Dynamic Data*; Datos Vinculados (LDA) por *Linked Data*; Imágenes (IMG); (STR) Datos en Streaming; Anotación Semántica (SAT) por *Semantic Annotation*; y Estados (STA).
- Tipo de Sensores: Sensores Físicos denotados como (PSN) por *Physical Sensor*; Sensor Virtual (VSN) por *Virtual Sensor*; Sensor Web (SWB); (SML) *Semántica SensorML*, (ND) No Descrito.
- Tipo de Cosas: Cosas Físicas denotadas como (PHT) por *Physical Thing*; Cosas Virtuales (VTH) por *Virtual Thing*; Cosas basadas en Agentes (AGT) por *Agent based or Avatar*; Descripción de Cosas W3C (TD) por *Thing Description Language W3C*; (ND) No Descrito.
- Contexto o Abstracción Extendida: Contexto de Seguridad (SEC) por *Security-awareness*; Consciente del Contexto (CTX) por *Context-Awareness*; Con base en Servicio (SRV); Con base en Aplicación (APP); Contexto Social (SOC) por *Social-awareness*.

Motor de Búsqueda (IoTSE   WoTSE)	Medidas de Complejidad		Evaluación RI	Escenario de Evaluación	Variable Principal	Dataset?
	Tiempo	Espacio				
- [42]	Velocidad del Sistema (QPS/t)	No	No	Consultas Genéricas	w/o Mecanismo DDoS	No
WoTmAS2E	Tiempo de Ejecución (t/step)	Tamaño del Índice	No	Asume consultas	Número de Dispositivos	No
- [2]	Tiempo Computacional (t)	Consumo de Espacio	No	CypherTexts	Número de Atributos	No
LUF	Velocidad del Sistema (QPS/t)	No	No	Consultas Genéricas	Tamaño del Clúster	No
- [41]	Volumen de Consultas (QV/t)	No	No	Consultas Específicas	Predicción de Slots	No
IoTcrawler	Ejecución de Consulta (t/Q)	No	No	Datasets de Sensores	Concurrencia	No
WoTHive	Ejecución de Consulta (t/Q)	No	No	Datasets de TDs	Carga de Consultas	Si

Tabla 2.7 Marcos de Evaluación en Sistemas RI para IoTWoT



Fuente de Información (Dataset)	# Datasets por Tipo			por Tamaño	por Juicio de Relevancia	Aplicabilidad a RI	Uso de Datasets
	IoT	WoT	RI	>10GB			
AWS Open Data	2	0	1	Indeterminado	0	Baja	Abierto
IEEE DataPort	200	0	4	Indeterminado	0	Baja	Abierto
Kaggle	143	0	23	30	0	Baja	Abierto
Google	43	4	100+	Indeterminado	Indeterminado	Media	Múltiples Tópicos
BoT-IoT	1	0	0	16GB	1	Baja	Seguridad en IoT
WoTHive	1	0	0	117KB	1	Media	Semántica
TD Ontology	0	1	0	491KB	0	Media	Semántica
TREC 2021	0	0	5	Indeterminado	5	Alta	Múltiples Tópicos
NTCIR 2022	0	0	6	Indeterminado	6	Alta	Múltiples Tópicos
NIST-15	0	0	6	Indeterminado	6	Alta	Múltiples Tópicos
FIRE 2021	0	0	6	Indeterminado	Indeterminado	Alta	Múltiples Tópicos

Tabla 2.8 Datasets Abiertos en relación a RI-IoT|RI-WoT

### 2.11.3. Creación de una Colección de Prueba RI-IoT|RI-WoT

Dadas las necesidades de evaluación sobre los sistemas de RI para los paradigmas IoT|WoT, y la carencia de *datasets* abiertos así como de colecciones de prueba enfocados a los paradigmas IoT|WoT, se tienen los siguientes caminos a fin de proveer de mecanismos de reproducibilidad de experimentos y reusabilidad de los datos en trabajos futuros en el estudio, análisis y desarrollo de sistemas RI en el campo. Y en últimas construir un marco de evaluación para sistemas RI-IoT y RI-WoT.

1. Creación de una Colección de Prueba RI-IoT|RI-WoT.
  - A partir de la modificación de Colecciones para la evaluación de desempeño RI con enfoque en tiempo-real.
  - A partir de Dataset IoT orientados a series de tiempo.
  - A partir de Dataset WoT orientados a la semántica.
2. Construcción via Aprendizaje Activo de una Colección de Prueba RI [110].
3. Evaluación RI como una simulación de búsqueda, una de las propuestas en el reciente NTCIR 2022<sup>31</sup> y en [150].

<sup>31</sup>Keynote en NTCIR 2022 - <http://research.nii.ac.jp/ntcir>

#### 2.11.4. Evaluación y Juicios de Relevancia

La creación de una colección RI de prueba clásica requiere de un gran esfuerzo en la obtención de las evaluaciones de usuarios y juicios de relevancia por parte de expertos. Metodológicamente hablando, se deben recopilar las evaluaciones de relevancia considerando:

- Necesidades de Información expresadas en la forma de Consultas. Lo que determina un número de consultas  $Q$ , sobre un número determinado de tópicos  $T$ .
- Un conjunto de documentos  $(N, K)$  recuperados por diferentes sistemas de RI (A|B|C)
- Un evaluación de relevancia dada por Expertos en el(los) tópico(s) sobre los documentos recuperados (juicios de relevancia).

Durante la evolución de la investigación RI, se han propuestos diferentes medidas de caracterización del acuerdo existente entre los jueces y su evaluación de relevancia sobre los documentos en la colección, mediante las denominada estadística Kappa [74].



# Capítulo 3

## Simulador de Eventos Discretos para la Web de las Cosas (SIM.WoT)

### 3.1. Arquitectura y Modelado de la Web de las Cosas

La arquitectura WoT, y particularmente el modelo de descripción de cosas, se constituyen en piedras angulares hacia la estandarización, la potencialización del paradigma y el desarrollo de nuevos sistemas. El consorcio W3C ha sido uno de los líderes en la estandarización de estos pilares, la arquitectura y modelo WoT, a través de uno de sus grupos de trabajo<sup>32</sup>. Actualmente, el W3C ha publicado recomendaciones en la forma de entregables normativos respecto a la arquitectura WoT<sup>33</sup>, al modelo de descripción de cosas en WoT<sup>34</sup>, y trabajos no normativos de tipo informativo y algunos borradores en tópicos descubrimiento de cosas, interfaces API, plantillas de protocolos y seguridad en WoT (estos borradores aún en construcción, no son recomendaciones de W3C).

Una primera normativa W3C describe la arquitectura abstracta para la WoT con un enfoque hacia un subconjunto de casos de uso y escenarios de despliegue concretos que van desde hogares inteligentes y fábricas inteligentes hasta vehículos interconectados. La arquitectura abstracta W3C WoT consiste en tres niveles: esta comienza por el nivel de dispositivo, luego el nivel de compuerta (o borde) y termina en el nivel de nube. A partir de esta arquitectura se construye una arquitectura conceptual también en tres niveles, comenzando en las plantillas de unión WoT, pasando por el modelo de interacción y terminando en la API de interacción. En la Figura 3.1 (a) se presenta la propuesta de arquitectura WoT en Guinard [40], esta comienza en la parte baja por una capa de accesibilidad para las cosas,

---

<sup>32</sup>Grupo de Trabajo WoT de W3C - <https://www.w3.org/WoT/WG/>

<sup>33</sup>Arquitectura W3C WoT - <https://www.w3.org/TR/wot-architecture/>

<sup>34</sup>Modelo W3C WoT de Cosas - <https://www.w3.org/TR/wot-thing-description/>

la cual es responsable de las comunicaciones de datos, incluidos metadatos, con la cosa en sí. Esta capa es similar a la capa de plantillas de unión de protocolo W3C que busca la integración de las cosas independientes del protocolo de comunicación subyacente. De modo que sea posible acceder a la información que las cosas monitorean, producen o exhiben sin dependencia a la tecnología de comunicación empleada en la red. El objetivo tras la capa de accesibilidad es construir una centricidad en las cosas y una orientación al recurso, de modo similar a lo que una página Web es a la WWW. A continuación, la capa de búsqueda proporciona capacidades para descubrir cosas, funcionalidades y servicios sobre estas cosas. La arquitectura contempla inherentemente los mecanismos para el etiquetado de las cosas y su posterior búsqueda en función de las necesidades de los usuarios. Las capas de uso compartido (o compartición), composición y aplicación proporcionan capacidades para la construcción de aplicaciones mucho más complejas en el ecosistema de WoT.

En paralelo, en la Figura 3.1 (b) se presenta la arquitectura W3C que propone en la parte baja las plantillas de unión o dispositivo WoT, una capa de modelado de la interacción con las cosas que emplea a su vez el modelo de descripción de cosas (TD W3C) y en la parte alta se vale de la capa de API *scripting* para la construcción de aplicaciones en el ecosistema de la WoT. En W3C además de la cosa en sí, componen la arquitectura WoT: el modelo de descripción de cosas, las plantillas de unión y el API Scripting. De forma general, la descripción de cosa de WoT es vista como información estructurada que se adhiere a un modelo formal y cierra la brecha entre los vocabularios de datos vinculados (LD Linked Data) y las API funcionales de los sistemas de IoT/WoT. W3C resalta que la TD puede ser visto como el HTML para las cosas en WoT. En segunda instancia, dada la heterogeneidad en las interacciones entre diversos sistemas IoT/WoT que siguen o no un estándar particular, W3C propone la construcción de una colección informal de metadatos de comunicación a los que denomina plantillas de unión WoT de modo que estas puedan explicar la forma en que interactúan las cosas en las respectivas plataformas o infraestructuras IoT. Por último, en la arquitectura existe un bloque opcional que facilita el desarrollo de aplicaciones mediante APIs como puntos de entrada al descubrimiento, consumo y exposición de las cosas, como objeto o interfaces tipo cliente y tipo servidor.

Por último, en la parte derecha de la Figura 3.1 (c) se presenta nuestra propuesta de arquitectura WoT híbrida. En comparación con las otras arquitecturas, concordamos con la visión de W3C en tres niveles. No obstante, proponemos, se complemente con la opción de incluir elementos que soporten la búsqueda WoT inherentemente en la arquitectura, como lo propone [40] y [134]. Aunque también es posible ver la búsqueda y recuperación de información en la WoT como una aplicación en la cima de la arquitectura W3C.

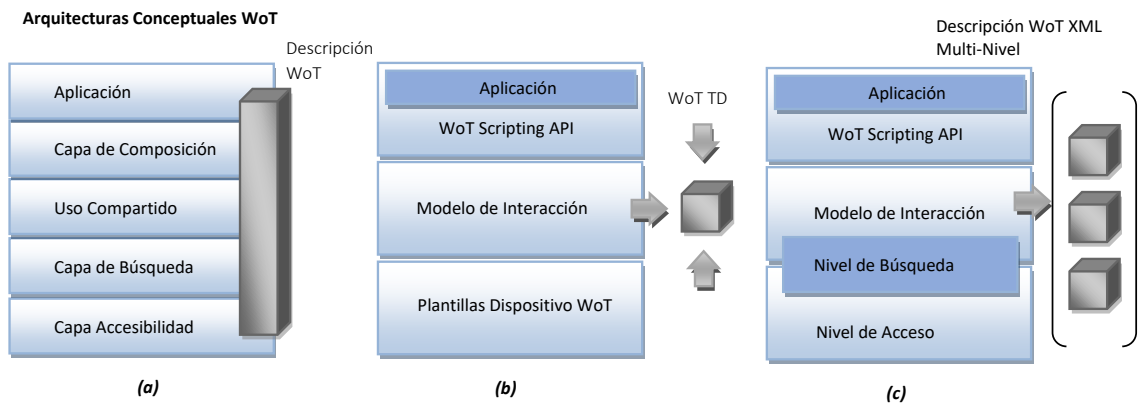


Figura 3.1 Arquitecturas Conceptuales WoT a) Guinard [40], b) W3C y c) Propuesta.

Todas las arquitecturas WoT tienen una centricidad en las cosas. Es importante resaltar que la W3C define:

*”Una Cosa es la abstracción de una entidad física o virtual que necesita ser representada en aplicaciones de IoT/WoT. Esta entidad puede ser un dispositivo, un componente lógico de un dispositivo, un componente de hardware local o incluso una entidad lógica, como una ubicación (por ejemplo, habitación o edificio)”.* - W3C.

Desde la perspectiva de W3C una cosa puede o no proveer una API de cara a la red (o al usuario final) que habilite su interacción o accesibilidad, empleando protocolos tales como CoAP. Una cosa puede exhibir únicamente metadatos sin una interfaz tipo Web con protocolos de infraestructura como MQTT o ZigBee entre otros. Sin embargo, una cosa debe poseer una descripción y de este modo cualquier entidad que tenga una descripción según este estándar es una cosa.

### 3.1.1. Modelo de Descripción de la Web de las Cosas

Como precursor de los modelos WoT, debemos destacar la investigación de [40]. Guinard define un mecanismo para describir cosas basadas en microformatos llamado modelo *Smart Things Metadata*, que se enfoca en el problema de describir los servicios provistos por las cosas y cubre información esencial para hacer búsquedas de cosas, funcionalidades y servicios. También, hoy en día, es bien conocida la contribución de la comunidad W3C a los esfuerzos de estandarización de los modelos de descripción de cosas. El modelo de descripción de cosas TD W3C consiste de un conjunto de metadatos de tipo semántico para

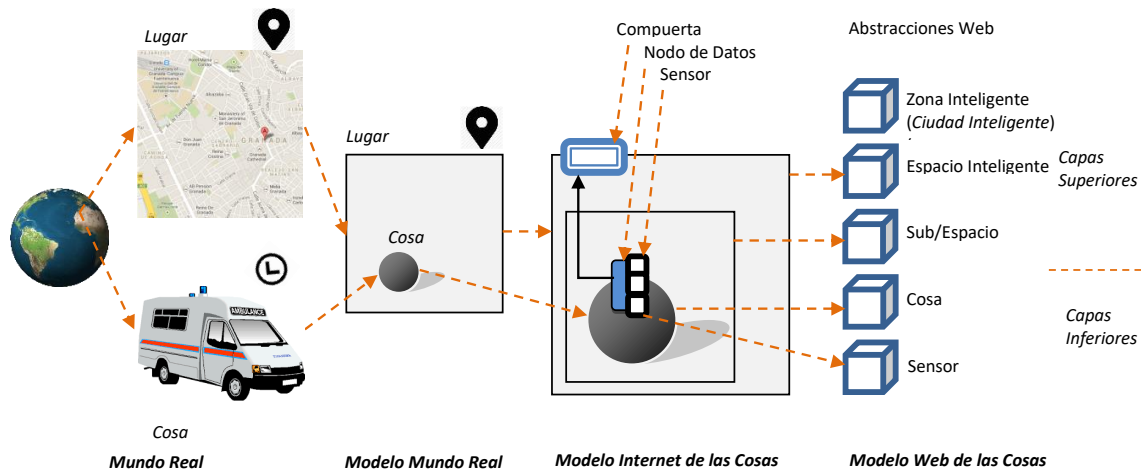


Figura 3.2 Arquitectura Multi-Nivel Abstracta WoT (Propuesta).

la cosa misma, de un modelo de interacción que reúne *Propiedades, Acciones y Eventos* de WoT para hacer que los modelos sean comprensibles por máquinas y características para la vinculación Web dotando de la capacidad de expresar relaciones entre cosas. El modelo TD W3C provee un mecanismo formal para describir las interfaces provistas por la infraestructura IoT y sus servicios, independiente de la implementación de protocolo.

Sin embargo, los modelos de descripción de cosas y los formatos de datos empleados en la actualidad son diversos y numerosos en las diferentes propuestas IoT/WoT. En las Figuras 3.3 y 3.4 se presenta una distribución de los principales modelos de descripción de sensores y cosas en las propuestas más significativas. Estos datos son parte de los resultados de la RSL en motores de búsqueda para IoT/WoT cuyo Dataset, preguntas de investigación, metodología y datos se encuentra disponible de forma abierta <sup>35</sup>. En el RSL también se incluye información de los formatos de datos empleados en cada trabajo, si así lo reporta, entre los que se destacan: COAP/HTML, Registros DNS, JSON o JSON-LD, NGSI-LDA, RDF, RFID/XML, WSML y XML.

Cabe resaltar que las diferentes propuestas de modelado en IoT/WoT pueden variar en centricidad, como por ejemplo en datos, en sensores o en las cosas. Como se observa en la distribución de modelos de cosas, la adopción del modelo estándar TD W3C aún es muy precario con una utilización de tan solo 2.7% sobre el conjunto de los trabajo analizados. De otro lado, es importante notar que las propuestas con centricidad en los datos o los sensores, no poseen una capa de abstracción del mundo real en modo de cosas, sino que en vez se

<sup>35</sup>Nota: Conjunto de datos alojado en IEEE DataPort(TM), un depósito de datos creado por IEEE para facilitar la reproducibilidad de la investigación. La versión final de los datos extraídos y los resultados del análisis están disponibles como conjuntos de datos abiertos: "Dataset for Systematic Literature Review on IoT and WoT Search Engines" - <https://iee-dataport.org/documents/dataset-systematic-literature-review-iot-and-wot-search-engines>

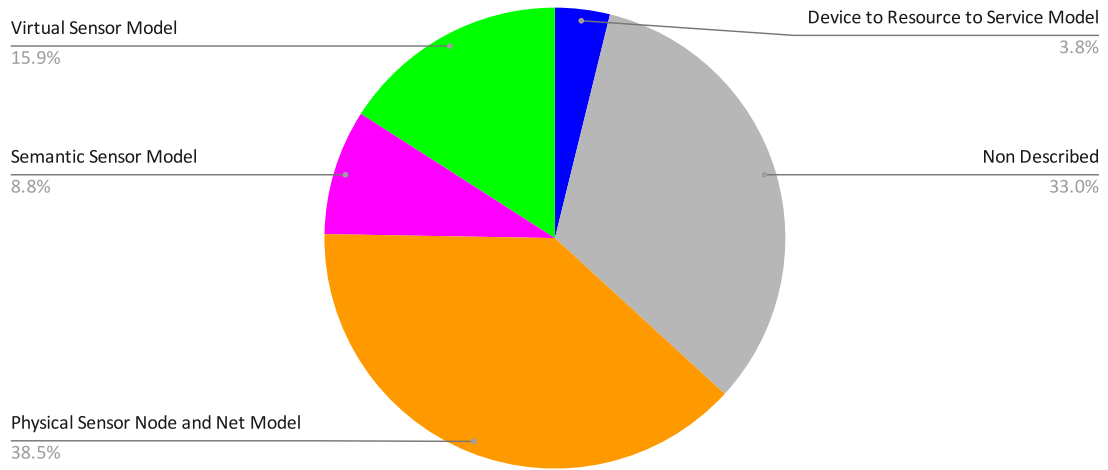


Figura 3.3 Porcentaje de Estudios Analizados clasificados por Tipo de Modelo de Sensor.

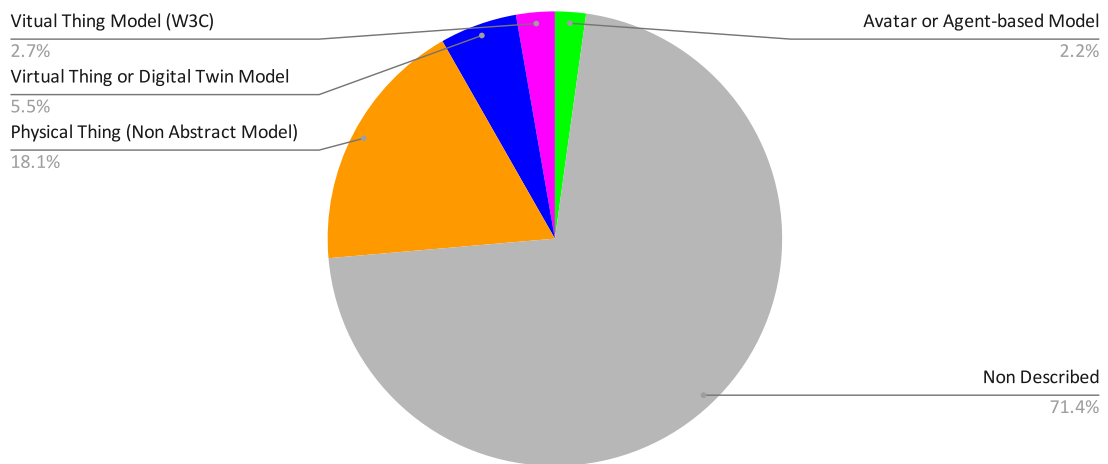


Figura 3.4 Porcentaje de Estudios Analizados clasificados por Tipo de Modelo de Cosa.



enfocan en la recolección de datos de variables de interés de fenómenos en el mundo real y su tratamiento. En las figura 3.5 y 3.6 se presentan un análisis comparativo de a) centricidad de los trabajos y b) el tipo de estudio presentado.

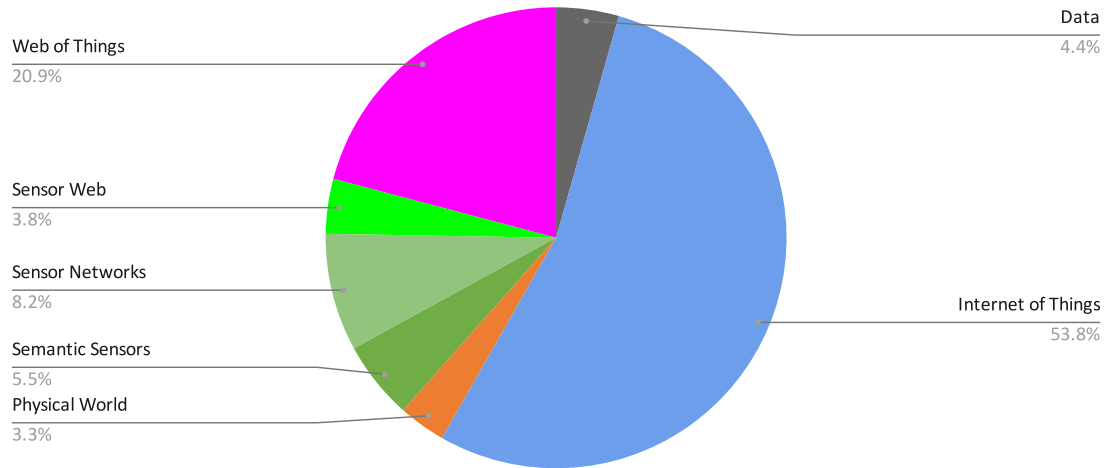


Figura 3.5 Porcentaje de Estudios Analizados clasificados por Centricidad.

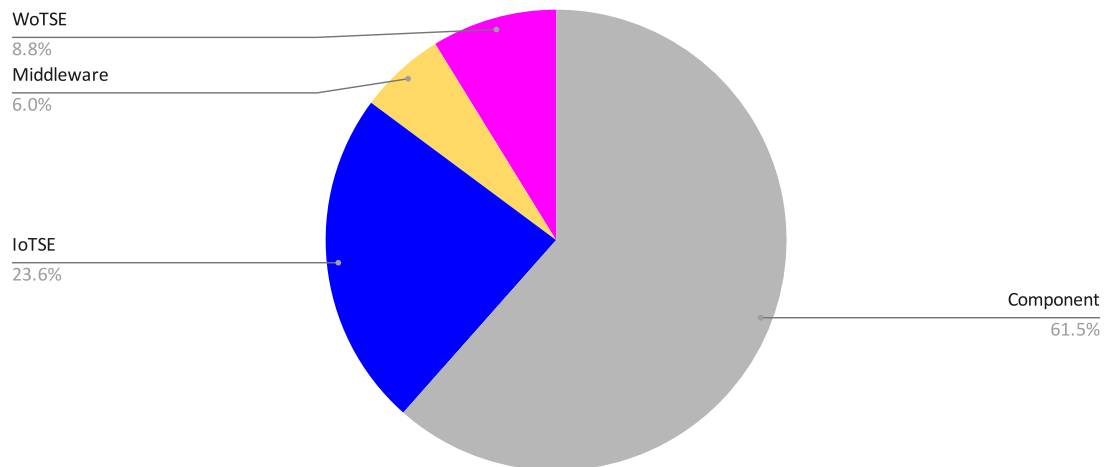


Figura 3.6 Porcentaje de Estudios Analizados clasificados por Tipo.

Estos dos paradigmas: centricidad en cosas y centricidad en datos/sensores son complementarios. Puesto que, las cosas como abstracción del mundo real podrían bien representar entidades físicas como lógicas. Por ejemplo, las cosas pueden representar condiciones ambientales capturadas a partir de una capa de sensorica, información de variables de interés asociadas a abstracciones virtuales del mundo como el clima. En [76] se propone un modelo de la WoT que presenta una conceptualización preliminar a través de la abstracción del

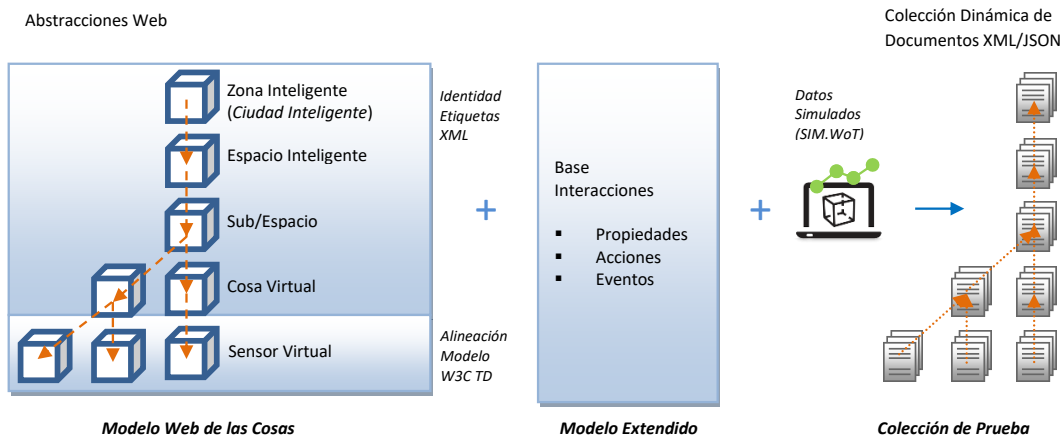


Figura 3.7 Modelo de Representación XML de WoT y Colección Dinámica de Prueba

mundo real, junto con una representación WoT que se enriqueció semánticamente utilizando elementos XML/URI. Nos referimos a éste como el modelo base. Este modelo considera el contexto espacial y establece un modelo que equilibra la representación del contexto espacial y las representaciones dinámicas de cosas y sensores sobre esquemas XSD como se ilustra en las Figuras 3.2 y 3.7.

### 3.1.2. Representación de Web de las Cosas con base en XML

Cada una de las entidades en las capas del modelo de descripción WoT propuesto se corresponde a un esquema XSD, que a su vez definen y ayudan en la generación de una colección dinámica de documentos XML para WoT, como se ilustra en la Figura 3.7. En esta se resaltan las características de la identificación Web y de las representaciones base y extendidas. A partir de las cuales fueron concebidas las abstracciones Web para el modelo WoT que dan origen a una colección dinámica XML/JSON a través de una simulación de eventos discretos.

El esquema XSD del sensor virtual propuesto tiene como base algunos de los componentes del modelo de Observaciones y Mediciones (O&M)<sup>36</sup> de OGC especificados dentro del lenguaje de modelado de sensores SensorML<sup>37</sup>. Este esquema de forma gráfica con representación en árbol se ilustra en la Figura 3.8 y se compone de:

- Etiquetas de información general con elementos de palabras clave, identificación y clasificación. Extensibles con un grupo de referencias con elementos de contacto, con la posibilidad de especificar elementos de documentación basada en roles especificados

<sup>36</sup>OGC O&M - <http://www.opengeospatial.org/standards/om>

<sup>37</sup>OGC SensorML - <http://www.opengeospatial.org/standards/sensorml>

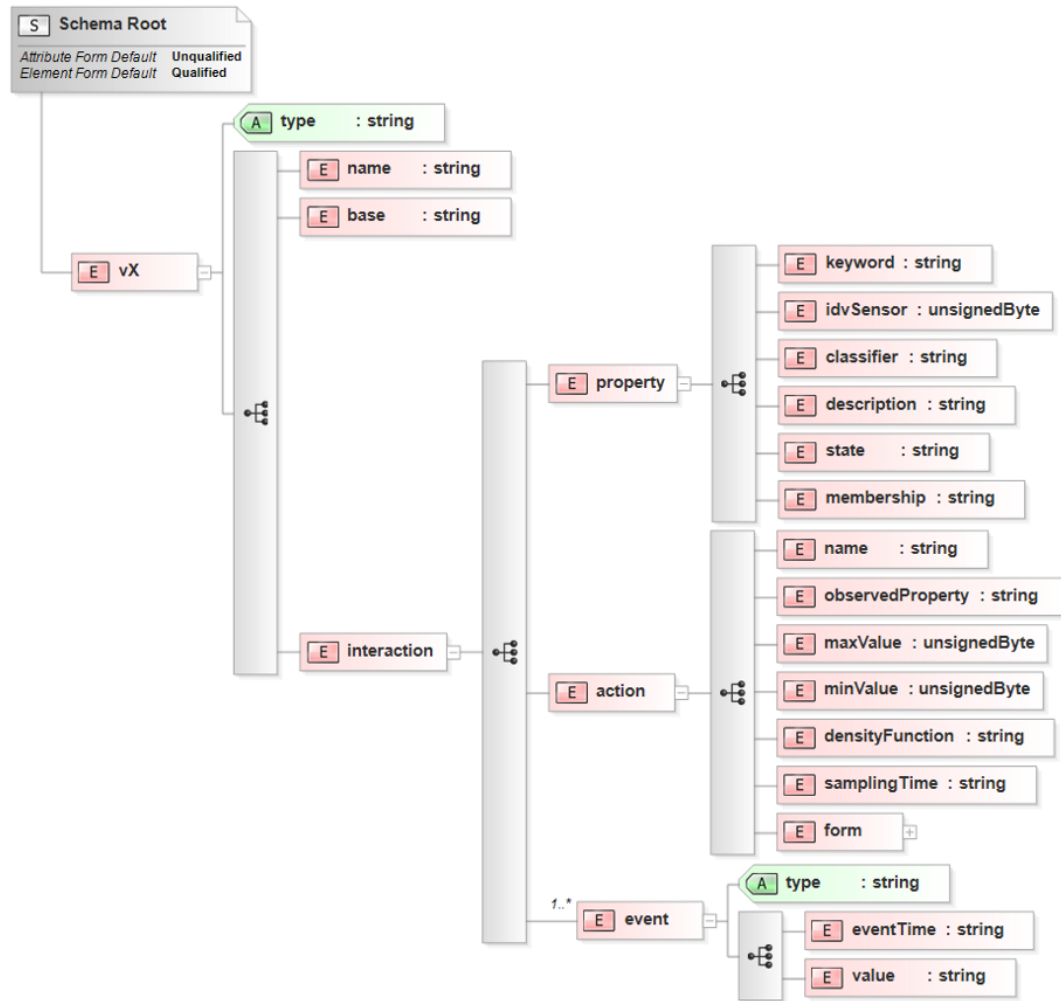


Figura 3.8 Representación Gráfica en Árbol del Esquema XSD de Sensor Virtual

en la implementación XML de SensorML. Se agrupan en conjunto con la siguiente sección del esquema XSD bajo el grupo de propiedades que permiten la caracterización del sensor virtual, la descripción de las capacidades del sensor, el estado de alto nivel, junto con un elemento de pertenencia para asociar el sensor virtual a la cosa virtual detectada o monitoreada.

- El siguiente grupo condensa los elementos de observación del esquema XML del sensor virtual y se basan en el esquema O&M, con un elemento de tiempo de muestreo, el tiempo del resultado, la característica de interés y el resultado agrupados bajo la etiqueta acción.
- Además, existe un grupo para almacenar el historial en forma de eventos.

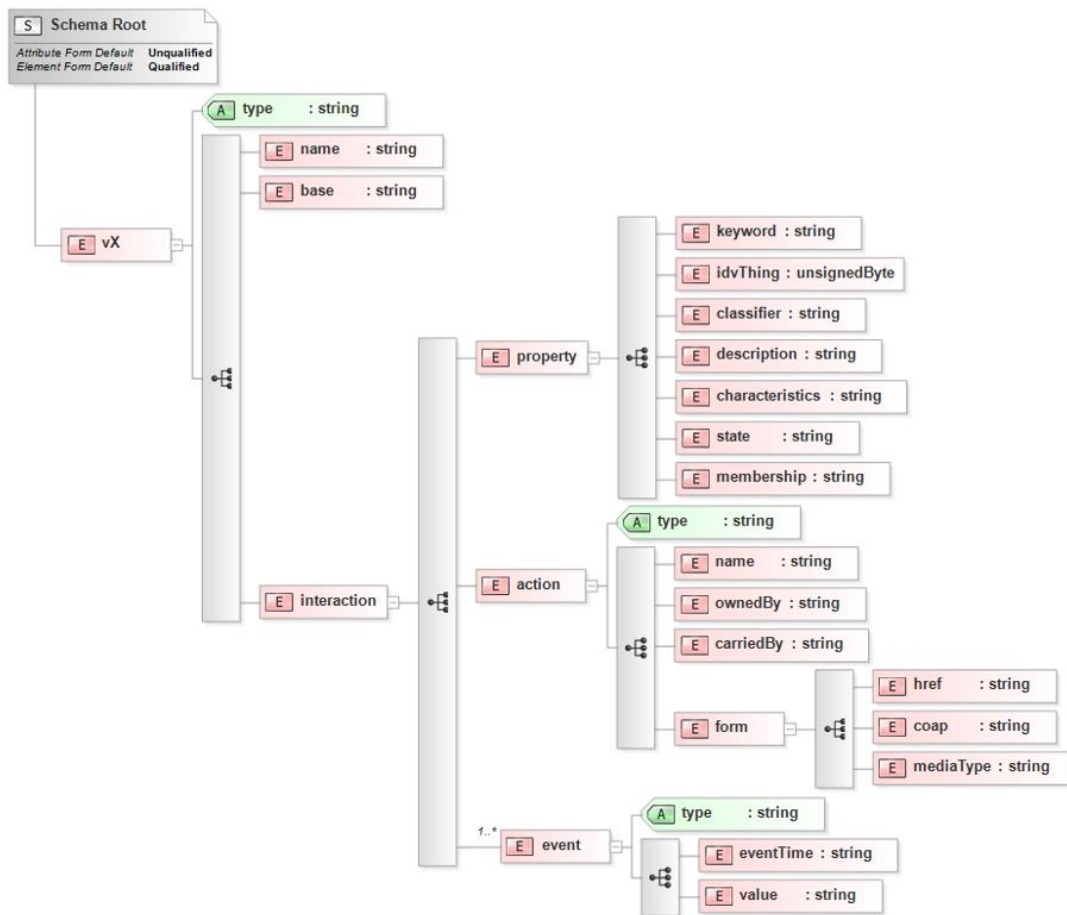


Figura 3.9 Representación Gráfica en Árbol del Esquema XSD de Cosa Virtual

El esquema XSD de cosa virtual debe capturar la información del fenómeno físico observado a la vez que describe cosas de carácter físico o virtual, por lo que se propone reutilizar algunos de los conceptos del esquema SWE<sup>38</sup> y Microformatos<sup>39</sup>. El esquema propuesto sigue la estructura de la representación XML del sensor virtual utilizando grupos y elementos: información general, referencias, propiedades e historial de eventos. El grupo de propiedades se ha enriquecido con un elemento de disponibilidad del objeto, y el campo de pertenencia está asociado con un nivel jerárquico superior, que indica el espacio inteligente donde se encuentra el objeto virtual. Se agrega una ubicación de elemento y una lista de sensores adjuntos. Organizado dentro de la etiqueta de interacciones considerando propiedades, acciones y eventos alineando el documento XML a la concepción del estándar TD para simplicidad de

<sup>38</sup>OGC SWE - <http://www.opengeospatial.org/standards/swes>

<sup>39</sup>Microformatos - <http://microformats.org/>

una posterior conversión a formato de documento JSON. El esquema se ilustra en la Figura 3.9.

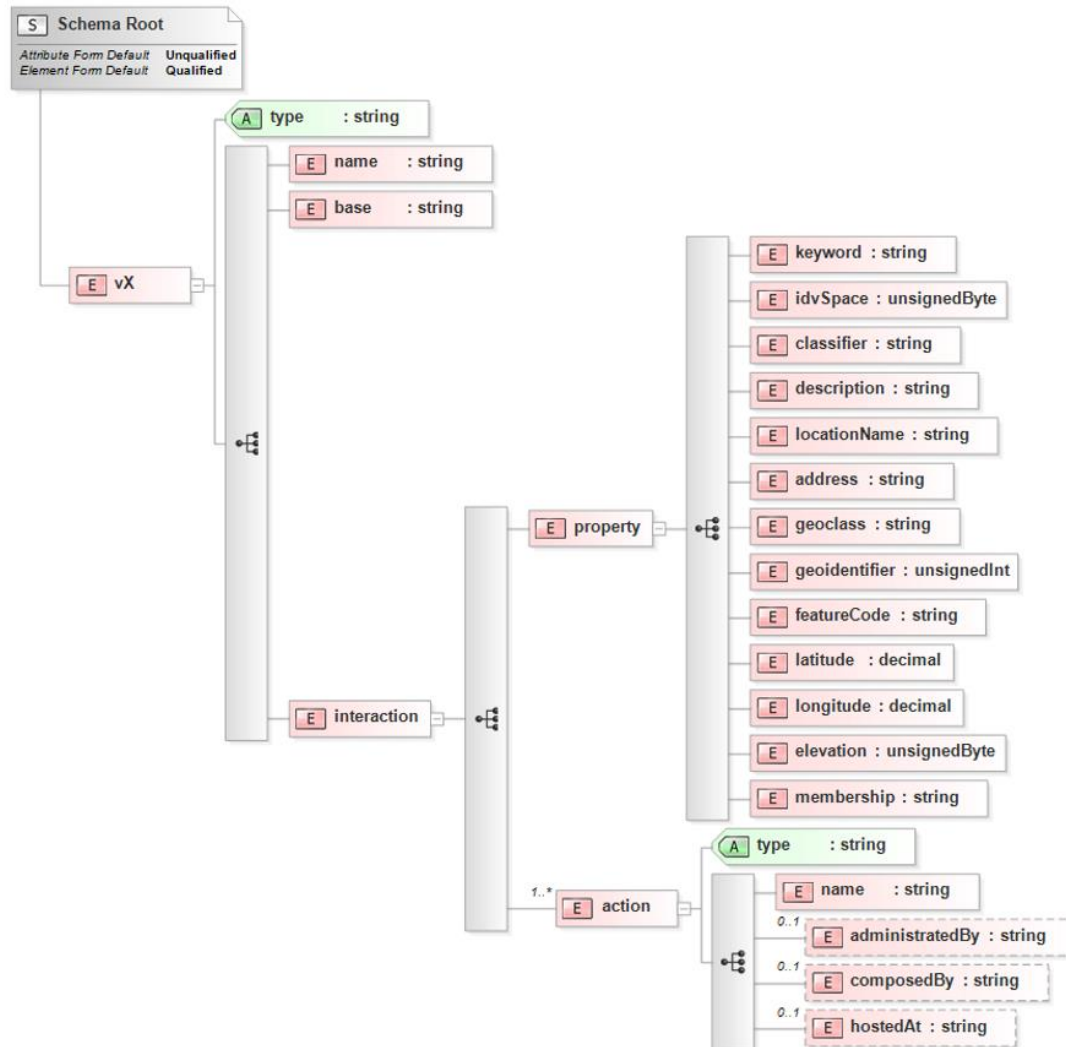


Figura 3.10 Representación Gráfica en Árbol del Esquema XSD de Espacio Inteligente

El esquema XSD del componente espacio inteligente se basa en el esquema de lugar de Microdatos en [schema.org](https://schema.org/Place)<sup>40</sup>. Este contiene elementos similares a los componentes anteriores con una lista de elementos virtuales y subespacios. La zona inteligente sigue un esquema similar con identificación de dominio web y la lista de espacios inteligentes que lo componen. Este esquema se ilustra en la Figura 3.10. La etiqueta acciones reúne información referente al organismo o persona encargada de temas administrativos del espacio inteligente, establecer

<sup>40</sup>Microdatos - <https://schema.org/Place>

los hipervínculos para establecer la jerarquía de espacio-subespacio, y listar las entidades virtuales alojadas en el subespacio correspondiente. La estructura XSD es igual para todos los componentes espaciales con la diferencia en que algunas etiquetas podrían asumir el valor Nulo significando que la etiqueta no es empleada para ese nivel jerárquico espacial.

Por último, como discusión final es importante notar que en esta propuesta la descripción de cosas se basa en tres elementos especiales: Eventos, Propiedades y Acciones. Pero también es posible ver estas desde las dimensiones de *Capacidades*, *Funcionalidades* y *Contexto*, como son presentadas en otras investigaciones. Además, la cosa se puede ver como un *Agente Inteligente* creado con una descripción y un comportamiento correspondiente. Enfatizamos el hecho de que nuestra propuesta se basa en el enriquecimiento de WoT con etiquetas XML en el nivel del modelo semántico WoT. También enfatizamos que el WoT impone factores dinámicos que afectan tanto la efectividad como la eficiencia de los sistemas de IR.

## 3.2. Modelo y Arquitectura de Simulación de la Web de las Cosas

La arquitectura de simulación de WoT propuesta considera el enfoque multi-capa descrito en la sección 3.1 al considerar las capas superiores del modelo WoT para encapsular el contexto espacial, y las capas inferiores para encapsular el contexto temporal y la dinámica de WoT. Un diseño preliminar de alto nivel (High-Level Design, HLD por sus siglas del inglés) para la arquitectura y el modelo de WoT se describe en [76]. Las siguientes subsecciones presentan una descripción más detallada de la arquitectura de cada subsistema. El diseño de bajo nivel (Low-Level Design, LLD por sus siglas del inglés) en términos del mecanismo de evento discreto detrás del simulador se describe en los siguientes capítulos. La arquitectura del simulador se ha diseñado en base a tres componentes modulares centrales que se ilustran en la Figura 3.11 y que, en conjunto, comprenden la lógica del proceso funcional del simulador. La arquitectura de tres niveles contiene una capa de persistencia (conexión a base de datos) y una capa de presentación (Interfaz de Usuario). Los módulos centrales del proceso funcional son:

- Sub-Sistema de Modelado de Simulación de Entidades
- Sub-Sistema de Generación de Datos de Simulación
- Sub-Sistema de Control e Informes de Simulación

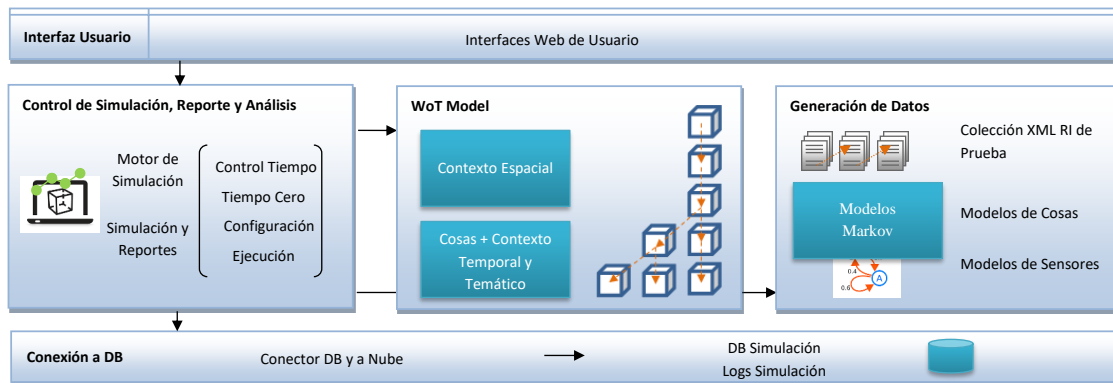


Figura 3.11 Arquitectura Conceptual del Simulador y Descomposición Modular

### 3.2.1. Sub-Sistema de Modelos: Simulación de Entidades

El subsistema de modelo de simulación es responsable de crear las abstracciones de las entidades de simulación. Se basa principalmente en el modelo extendido WoT. Las entidades de simulación para el contexto espacial provienen de las tres capas superiores que se construyen en base a esquemas XSD de zona inteligente, espacio inteligente y subespacio inteligente. Estas entidades de simulación son similares en estructura y tienen atributos espaciales similares. La entidad de zona inteligente se identifica mediante ID y un dominio Web, junto con la clasificación, descripción e información característica: nombre de ubicación, elevación, latitud, longitud, etc. En el caso de espacios inteligentes y subespacios, las etiquetas de información de pertenencia identifican la zona inteligente, en particular, y la estructura del árbol, en general, de estos espacios.

Las dos capas inferiores encapsulan el contexto temporal y la dinámica en términos de historia, eventos y marcas de tiempo dentro de los esquemas XSD. La entidad de cosa virtual contiene identificación, descripción y atributos de categorización. Con el propósito de seguir el paradigma WoT, las cosas virtuales incluyen etiquetas de propiedades e información sobre sus características. La construcción de su esquema XSD es una perspectiva simplista y se inspiró en los microdatos. El contexto temporal está encapsulado en el llamado atributo de eventos de la entidad, donde cada cambio de estado se registra con la marca de tiempo correspondiente. La cosa virtual puede estar en uno de varios estados exhibidos en su modelo de comportamiento siguiendo las reglas de eventos discretos que se describen en la sección 3.3. La entidad de simulación de sensor virtual usa el atributo de eventos de la misma manera que la cosa virtual. El sensor virtual está asociado a un modelo de medición, que sigue una función de densidad de probabilidad seleccionada por el usuario. También está asociado con un modelo de falla y los mecanismos para esto se describen en la sección 3.3. La

motivación detrás de la construcción del subsistema del modelo del simulador es encapsular los principales factores que producen un alto dinamismo en la WoT.

### 3.2.2. Sub-Sistema de Generación de Datos y Modelos de Comportamiento

El subsistema de generación de datos es responsable de construir lógicamente la colección de prueba IR en tiempo real de documentos XML (ver Figura 3.7). Los subsistemas de generación de datos, mientras tanto, usan los modelos de comportamiento de cosas virtuales y sensores virtuales para producir datos espaciotemporales. Por ejemplo, el modelo de movilidad proporciona la ubicación actual de una cosa virtual y su trayectoria, y el modelo de conectividad proporciona datos sobre su estado actual. En el caso del sensor virtual, el modelo de medición proporciona datos estocásticos siguiendo la función de densidad de probabilidad de la característica de intereses, y el modelo de falla proporciona el estado de operación de esta entidad.

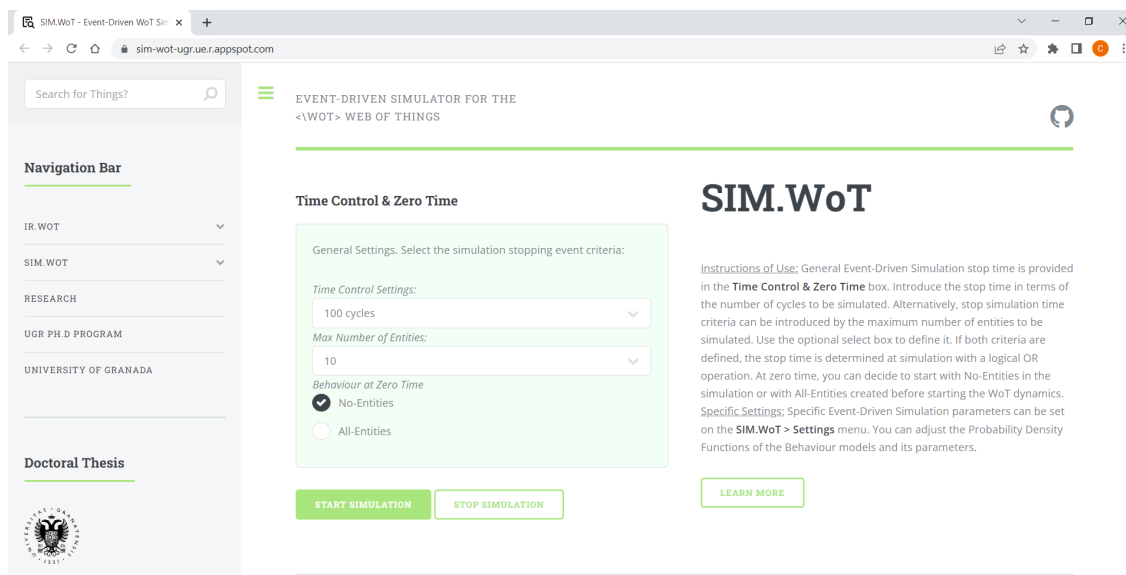


Figura 3.12 Interfaz Web Principal de SIM.WoT

### 3.2.3. Sub-Sistema de Control y Reportes de Simulación

El simulador proporciona una interfaz GUI basada en Web, como se ilustra en la Figura 3.12, para que un usuario ejecute una simulación WoT. En general, esto consta de los siguientes pasos:



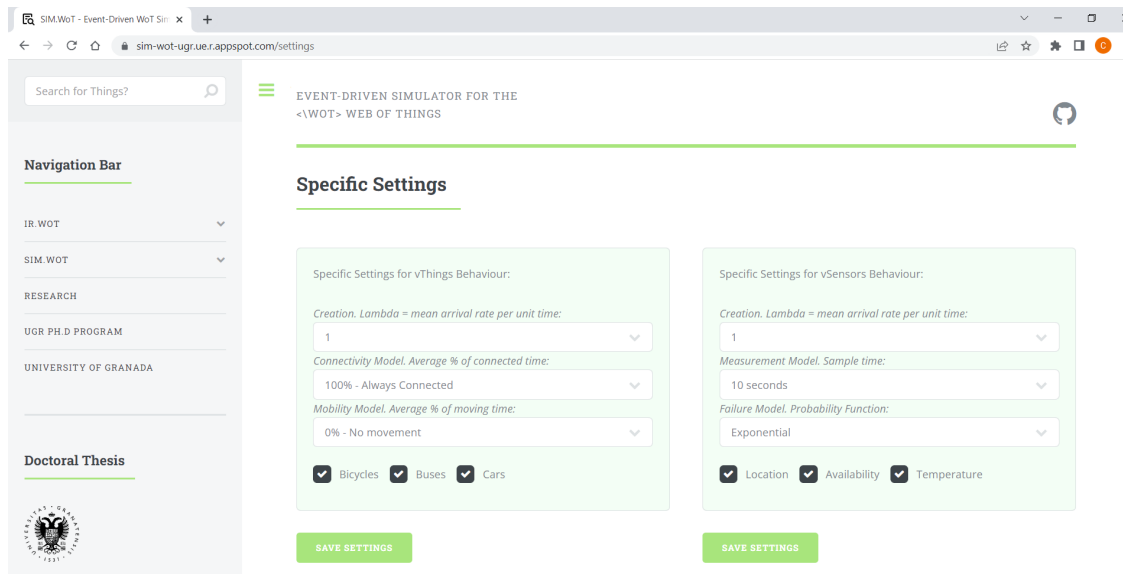


Figura 3.13 Interfaz Web Configuración de SIM.WoT

1. Crear las entidades de simulación: zonas inteligentes, espacios inteligentes, subespacios inteligentes, elementos virtuales y sensores virtuales. Este paso puede ser realizado de forma manual mediante formulario en la versión (V1), y a través de la base de datos o la API en la versión (V2).
2. Establecer los parámetros principales: contexto espacial y contexto temporal.
3. Configurar las observaciones a simular en términos de la función de densidad de probabilidad (PDF) de las variables de interés.
4. Configurar el sistema del motor de simulación.
  - Establecer los parámetros de parada de simulación.
  - Establece los modelos de generación estocástica: cosas virtuales y sensores virtuales.
5. Ejecutar la simulación.

Los subsistemas de control de simulación son responsables del propio motor de simulación, que a su vez administra el mecanismo de eventos discretos, establece el tiempo de simulación cero y controla el paso del tiempo por eventos. Finalmente, determina el final de la simulación dado el evento de detención. El subsistema de control se puede ver como la implementación del mecanismo de eventos discretos descrito en la siguiente sección 3.3. Además, recopila y registra información sobre algunas de las variables y estadísticas fundamentales del simulador.

### 3.2.4. Código Fuente y Versiones

El código fuente V1 puede modificarse fácilmente en el simulador para personalizar la disponibilidad de datos, el modelo de movilidad virtual y también es posible modificar el modelo de falla del sensor virtual. El objetivo de la simulación es representar el comportamiento dinámico de WoT en términos de generación de información, su arquitectura distribuida y la disponibilidad de información de las representaciones Web XML de objetos del mundo real y sus sensores asociados. En el momento de la simulación cero, se crean múltiples documentos XML basados en las entidades de simulación ingresadas por el usuario, siguiendo un directorio estructurado en árbol.

El código fuente de la versión (V1) (basado en Java) puede encontrarse en el siguiente repositorio GIT público: <https://github.com/cristyanmanta/sim-wot-ugr-v1.git>. El código fuente de la versión (V2) (basado en Python) puede encontrarse en el siguiente repositorio GIT público: <https://github.com/cristyanmanta/sim-wot-ugr-v2.git>. Este último, construido para ser desplegado en nube (Google Cloud Platform - App Engine).

## 3.3. Mecanismo de Simulación de Eventos Discretos

Esta sección se detalla el diseño de bajo nivel (LLD) de nuestro mecanismo de simulación de eventos discretos. La Figura 3.14 ilustra los componentes macro que componen el mecanismo de simulación por eventos discretos (DES) y la lógica de todo el motor de simulación propuesto. Comenzamos por enumerar los siguientes conceptos:

- Sistema: colección de entidades (por ejemplo, zonas inteligentes, espacios inteligentes y subespacios, cosas virtuales y sensores virtuales) que interactúan juntas a lo largo del tiempo para representar el mundo físico.
- Modelo WoT: una representación abstracta de un sistema, que contiene relaciones estructurales, lógicas o matemáticas en términos de estado, entidades y sus atributos, conjuntos, procesos, eventos, actividades y retrasos.
- Estado del sistema: colección de variables que contienen toda la información para describir el sistema en cualquier momento.
- Entidad: cualquier objeto o componente en el sistema que requiera representación explícita en el modelo.
- Atributos: las propiedades de una entidad dada.

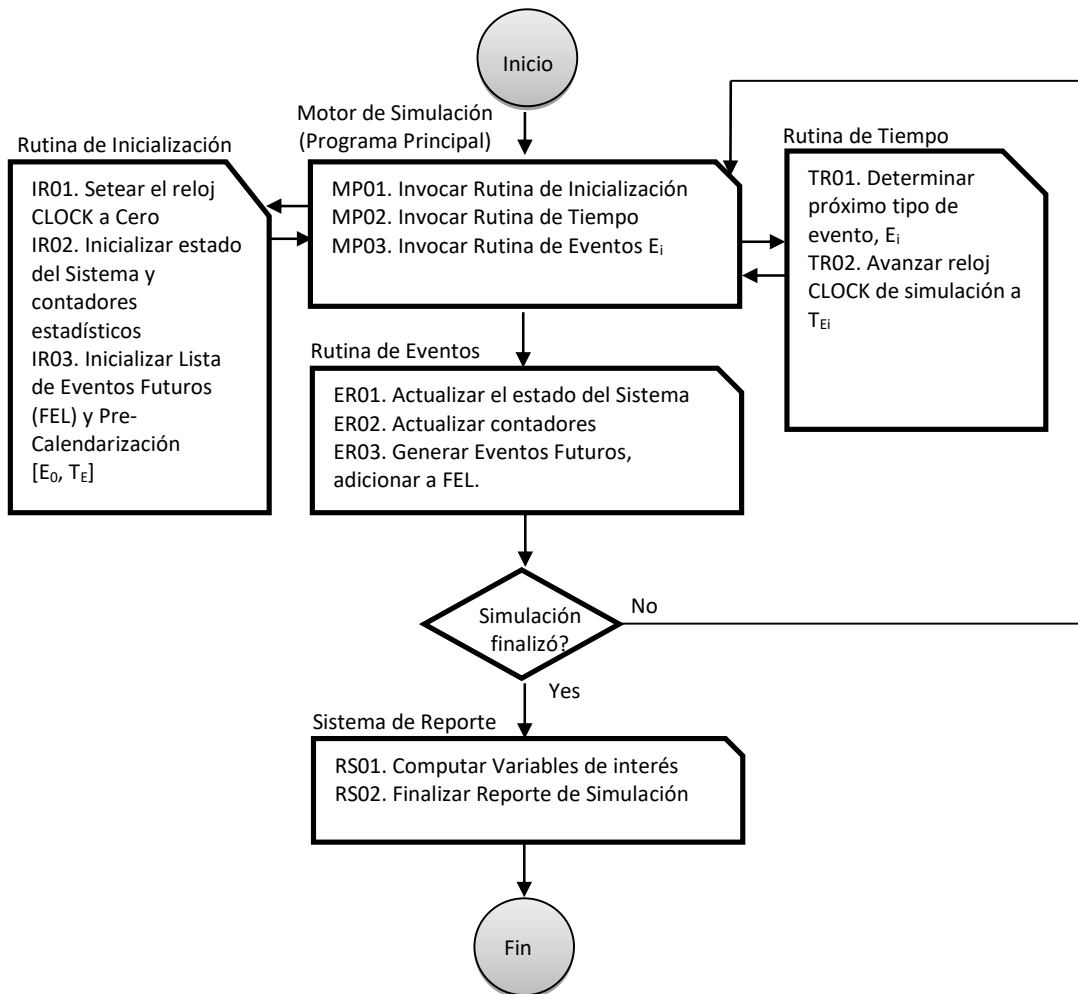


Figura 3.14 Componentes macro y lógica del motor de simulación WoT

- Lista de entidades: una colección de entidades asociadas.

La simulación de eventos discretos es estocástica, dinámica y discreta. Los tiempos entre llegadas son variables aleatorias y tienen funciones de densidad y distribución acumulativa.

### 3.3.1. Evento de Parada de Simulación

La simulación tiene un evento de parada (*Stop Event*, SE por sus siglas del inglés) que define cuánto tiempo se ejecutará la simulación. Hemos decidido utilizar los dos criterios siguientes para detener la simulación:

Tipo de Evento	Descripción del Tipo de Evento de Simulación
0	Evento de Parada de Simulación
1	Llegada de una nueva Cosa Virtual ( <i>vThing</i> )
2	Partida de una Cosa Virtual
3	Desconexión de una Cosa Virtual
4	Reconexión de una Cosa Virtual
5	Muestreo de un Sensor Virtual ( <i>vSensor</i> )
6	Falla de un Sensor Virtual
7	Reparación de un Sensor Virtual

Tabla 3.1 Tipo de eventos en la simulación WoT

- En el tiempo cero, programamos un evento de simulación de detención en un tiempo futuro específico (TE, Hora de finalización). Por lo tanto, sabemos que la simulación se ejecutará durante el intervalo de tiempo  $[0, TE]$ .
- TE se determina por simulación y es el momento de ocurrencia cuando el número de entidades *vThings* o *vSensors* alcanza el número máximo definido por el usuario.

### 3.3.2. Eventos en la Simulación de la Web de las Cosas

Consideramos que un evento es una ocurrencia instantánea que cambia el estado de un sistema. Se crea con un objeto donde los principales atributos son el tipo de evento y la hora del evento. Cada aviso de evento se registra junto con sus atributos de ocurrencia de tiempo actual o futuro y cualquier dato asociado necesario para la ejecución. Después de la inicialización del sistema de simulación, también se inicializa una lista de eventos futuros (FEL, *Future Event List*), que incluye avisos de eventos futuros ordenados según el tiempo de ocurrencia, y se realiza una primera pre-programación con el evento de detención. La Tabla 3.1 ilustra los tipos de eventos en la simulación WoT con una breve descripción.

El evento de parada se ha establecido como Tipo 0 y está programado en la rutina de inicialización en IR03. En el tiempo cero, también se pre-programan avisos de eventos de tipo 1 para cada *vThing* en la lista de entidades, y eventos de tipo 5 para cada *vSensor* asociado a estos *vThings*. El tiempo de llegada entre nuevas entidades sigue una distribución exponencial y se asume que los *vSensors* toman su primera unidad de muestreo después de que el *vThing* asociado alcanza la simulación. La Figura 3.15 muestra el resultado cronológico de IR03 en el FEL. El algoritmo de programación de eventos y avance de tiempo es implementado por la rutina de tiempo de la siguiente manera:

- Paso TR01. Se elimina el aviso de evento para el evento inminente en la FEL.

- Paso TR02. Se avanza el reloj al tiempo del evento inminente. Después de que el reloj avanza a  $T_i$ , se ejecuta un evento de Tipo 1 y se ejecuta la rutina de evento.
- Paso ER01. Se ejecuta el evento inminente enésimo: se actualiza el estado del sistema, se cambian los atributos de la entidad de simulación y se establece el próximo paso según sea necesario.
- Paso ER02. Se actualizan las estadísticas acumulativas y los contadores.
- Paso ER03. Se generan nuevos eventos futuros y se crean sus avisos de eventos en la FEL clasificados por tiempo de ejecución del evento.

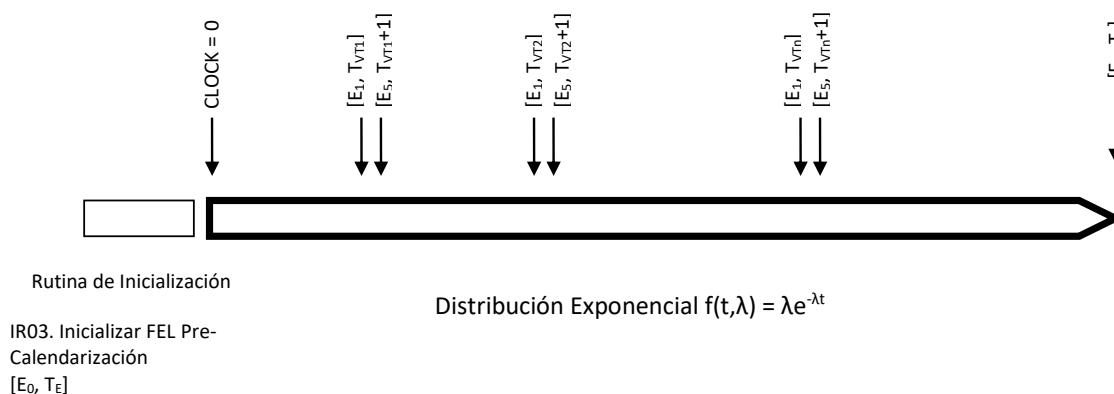


Figura 3.15 Pre-calendización de eventos de *vThings* y *vSensors*

### Evento Tipo 1: Llegada de un nuevo *vThing*

Se asigna un subespacio de llegada inteligente que se ha seleccionado aleatoriamente con una distribución uniforme entre todos los subespacios inteligentes en la simulación. La Figura 3.16 ilustra el diagrama de transición de eventos, en el cual, por ejemplo, un nuevo *vThing* pertenece a un subespacio inteligente que ha sido seleccionado con una probabilidad de  $1/(\text{cantidad total de subespacios})$ . Con respecto a la actualización del estado del sistema y las estadísticas y contadores acumulativos, suponemos que el nuevo *vThing* está en un estado estacionario en línea. Los asientos disponibles en el subespacio inteligente se reducen en uno. Este atributo se usa para limitar el número de *vThings* que pueden ser miembros o pueden estar contenidos en el subespacio inteligente. El contador de llegada *vThings* se incrementa en uno. El evento Tipo 1 puede generar los siguientes eventos futuros:

- Evento Tipo 2 dentro de un tiempo estacionario siguiendo una distribución exponencial desde el reloj actual.

- Evento tipo 1 en caso de que las entidades de simulación disponibles sean iguales a cero en el espacio, con un tiempo de reubicación después de una distribución exponencial del reloj actual.

#### **Evento Tipo 2: Salida de un *vThing***

En este caso, el estado de *vThing* cambia a movimiento en línea, lo que aumenta la cantidad de asientos disponibles en el subespacio inteligente. El contador de movimientos se incrementa en uno. El *vThing* podría desconectarse de la red debido a factores externos con una cierta probabilidad y, si esto ocurre, un evento Tipo 3 futuro se programará con un tiempo promedio de desconexión durante el movimiento. De lo contrario, un evento Tipo 1 se programa con un tiempo de movimiento promedio y ambos tiempos temporales se caracterizan por una distribución exponencial.

#### **Evento Tipo 3: Desconexión de un *vThing***

El estado *vThing* cambia a fuera de línea y, en consecuencia, la información en tiempo real sobre el *vThing* y los *vSensors* asociados no está disponible para leer en el WoT. El contador *vThing* fuera de línea se incrementa en uno. Al igual que los eventos Tipo 2, los eventos Tipo 3 se basan en una cadena de Markov de tres estados y el estado futuro de la *vThing* se calcula con una cierta probabilidad dada por la matriz de transición. Por lo tanto, el *vThing* podría reconectarse en algún punto futuro mientras se mueve o como resultado de alcanzar un subespacio inteligente cuando el *vThing* se conecta a Internet. Como resultado, programamos un evento Tipo 4 o Tipo 1 con un tiempo promedio de conexión desde el reloj.

#### **Evento Tipo 4: Reconexión de un *vThing***

En este caso, el estado *vThing* cambia a movimiento en línea. Un evento Tipo 1 futuro se programará.

#### **Evento Tipo 5: Muestreo de un *vSensor***

Como se mencionó anteriormente, se programa un evento Tipo 5 para cada intervalo de tiempo de simulación de *vSensor*, uno después de que llega un nuevo *vThing* a la simulación. Cuando un *vThing* alcanza un subespacio inteligente, sus *vSensors* asociados comienzan a medir y muestrear la variable de interés. Suponemos que el estado inicial de un *vSensor* es activo. El *vSensor* genera un valor de muestreo con una cierta función de distribución de probabilidad definida por el usuario. La variable aleatoria también se caracteriza por un valor

máximo y un valor mínimo. En el caso de que *vSensor* falle en el futuro con una probabilidad dada, se programa un evento Tipo 6 con un tiempo de falla y con distribución exponencial o Weibull. De lo contrario, el *vSensor* continúa funcionando normalmente y otro evento periódico tipo 5 se vuelve a programar a  $x$  segundos del reloj real.

### Evento Tipo 6: Fallo de un *vSensor*

Si esto ocurre, el *vSensor* ingresa al estado dañado. En consecuencia, no habrá información disponible de su variable de interés. Suponemos que el *vSensor* se puede reparar en el futuro, por lo que su reactivación está programada con un evento Tipo 7, dentro de un tiempo de reparación exponencial.

### Evento Tipo 7: Reparación de un *vSensor*

El *vSensor* vuelve al estado activo y se producirá un evento Tipo 5 en un tiempo de muestreo. La Figura 3.17 ilustra el diagrama de transición para un *vSensor*.

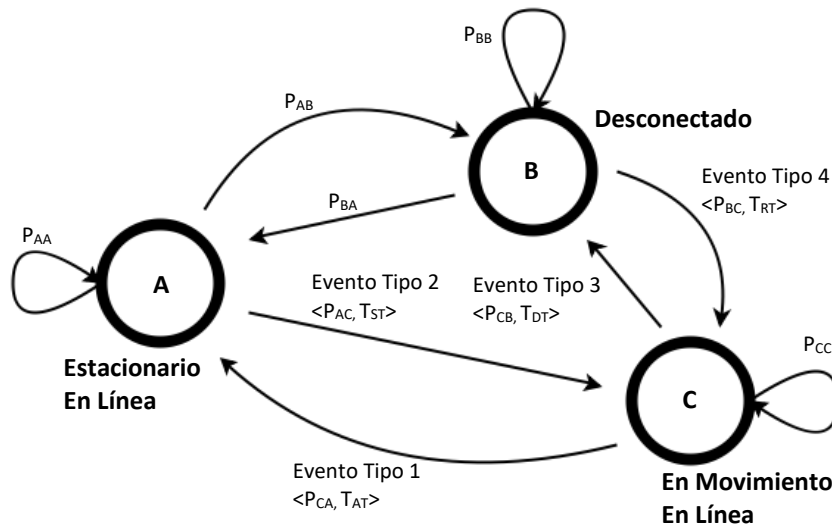


Figura 3.16 Diagrama de Transición con Modelado de Cadena de Márkov para *vThing*

### 3.3.3. Consideraciones de Desarrollo, Implementación y Ejecución de SIM.WoT

En términos generales, los investigadores podrían utilizar SIM.WoT para:

- Simular la dinámica WoT de un sistema, teniendo en cuenta los modelos de comportamiento de cosas y sensores.

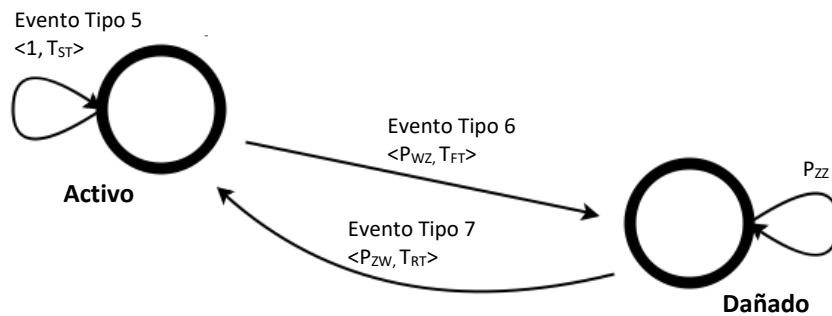


Figura 3.17 Diagrama de Transición con Modelo de Cadena de Márkov para *vSensor*

- Extender el comportamiento del modelo de cosas y sensores para incluir otras características y dinámicas de WoT.
- Soportar el diseño de sistemas basados en modelos, como los sistemas de RI, de recomendación para paradigmas IoTWoT, entre otros.
- Probar la extensión de modelos existentes para agregar nuevas funcionalidades y recopilar datos para optimizar los sistemas de IR.
- Explorar la aplicación industrial IoTWoT (esto requeriría la inclusión de aspectos relacionados con disponibilidad, confiabilidad y resistencia).
- Ayudar en la caracterización de sistemas dinámicos del mundo real para construir sistemas de control de próxima generación.

Desde la perspectiva IR, apoyar en el:

- Diseño y prueba de los módulos de indexación de un sistema IR en términos de tiempo de indexación, tiempo de respuesta de consulta.
- Diseño y clasificación de pruebas y módulos de puntuación y clasificación de un sistema IR en términos de precisión, exhaustividad y consumo de recursos.
- Evaluar diferentes estrategias de indexación XML, agregar técnicas en tiempo real y seleccionar las estructuras de datos más adecuadas para un sistema IR para el WoT.

Cuando se prepara una simulación, los siguientes pasos son importantes para recopilar y proporcionar los parámetros al simulador:

1. Mapee y adapte el modelo WoT a un escenario del mundo real



- (a) Contexto espacial preestablecido: los tres niveles de espacio contienen la misma información de etiqueta y permanecen constantes.
- (b) Recopilar información espacial: la información geográfica se puede extraer de <http://www.geonames.org/> a través de una consulta de base de datos o un servicio web. Por ejemplo, el Reino de España se caracteriza por el código geográfico PCLI Entidad política independiente, la clase geográfica A País y las coordenadas. Por otro lado, una zona inteligente o espacio inteligente requiere un clasificador interno y autorización de información de registro proporcionada por el usuario.
- (c) Recopilar información de la cosa: la información descriptiva sobre la cosa del mundo real se ingresa en campos de texto. Para escenarios reales como el Escenario A, la información se puede extraer de las fichas técnicas de los documentos de especificación, en este caso <https://www.bicing.cat/>.
- (d) Recopilar información del sensor: la información descriptiva sobre el objeto del mundo real se ingresa en campos de texto. Tres campos principales relacionados con la naturaleza intrínseca del sensor y la propiedad observada: la función de densidad de probabilidad, que podría basarse en estimaciones previas de la propiedad observada, por ejemplo distribuciones Gaussiana y Beta para propiedades tales como temperatura, humedad; los valores min-max o el rango de funcionamiento del sensor, por ejemplo 0°C - 200°C en un termopar; y la unidad de medida de las observaciones, por ejemplo Celsius en el caso del ejemplo.

## 2. Variables de entorno del mecanismo de eventos discretos:

- (a) Tiempo máximo: criterio de parada para detener la simulación, en función del tiempo del mecanismo de eventos discretos.
- (b) Máx  $vThings/vSensors$ : criterios de parada alternativos basados en máximo número de  $vThings/vSensors$  en la simulación.
- (c) Lambda  $vThings/vSensors$ : Lambda es la tasa de llegada media por unidad de tiempo que sigue una distribución exponencial para tiempos inter-llegadas.
- (d) Función de distribución de probabilidad (PDF)  $vThings/vSensor$ : Estas variables de entorno deben configurarse de acuerdo con el tamaño del sistema WoT a simular y la dinámica propia de los  $vThings/vSensor$ .

Característica	Soportado (Modelo Actual)	No Soportado (Posibles Extensiones)
Estructura	Modelo WoT anidado y extendido	vSensors conectados directamente a Espacios
Entidades	Distribución uniforme para la movilidad de vThing	Movilidad y trayectorias complejas
Capacidades	Estados markovianos de conectividad de vThing	Interconexión con protocolos M2M
Geo-Datos	Conciencia espacial con datos georreferenciados	SIG multicapa con sistemas vectoriales
Predicción	Datos estocásticos creados a partir de modelos	Reducción de datos basada en predicciones en vSensor
Recuperación	Perspectiva IR y colección sintética en disco	API de servicio web para consumir datos

Tabla 3.2 Límites y Características del Simulador (SIM. WoT)

### 3.4. Simulación de WoT: Escenarios Reales, Experimentación y Resultados

En esta sección, evaluamos el motor de simulación en términos de escalabilidad y diversidad de escenarios a simular. Elegimos los siguientes escenarios reales para ilustrar la flexibilidad de nuestro modelo y presentamos varios experimentos y resultados de simulación. Como mostramos en la sección anterior, hemos incluido un conjunto específico de modelos de comportamiento en el mecanismo de eventos discretos relacionados con la cosa virtual y las entidades sensores virtuales. Sin embargo, el mecanismo podría extenderse para incluir otros modelos no soportados en esta implementación. En la Tabla 3.2 enumeramos las características de nuestra propuesta y otras que no pueden realizarse actualmente pero que podrían incluirse en el futuro (ver Tabla 3.2).

Presentamos tres escenarios que fueron seleccionados para exhibir diferentes comportamientos WoT. En primer lugar, el Escenario A tiene un solo tipo de sensor, una sola cosa virtual y un solo espacio inteligente en una zona inteligente. En segundo lugar, el Escenario B tiene múltiples tipos de sensores, un único tipo de cosa virtual, dos espacios inteligentes en una zona inteligente. Finalmente, el Escenario C tiene múltiples tipos de sensores, múltiples tipos de cosas virtuales y múltiples espacios inteligentes en una zona inteligente.

#### 3.4.1. Escenario A: Sistema Bicing

Consideramos el sistema Bicing<sup>41</sup> como un escenario WoT. Bicing es un sistema de bicicletas compartidas en Barcelona y su objetivo principal es cubrir rutas diarias de corta y media distancia dentro de la ciudad. Nuestro primer paso en términos de simulación es mapear este escenario WoT real a nuestro modelo WoT. Para las capas superiores, se considera el Reino de España como nuestra zona inteligente con capacidad para federar espacios inteligentes españoles, y asociado directamente a los dominios .es y .cat WoT. En este sentido, la Ciudad de Barcelona se modela como un espacio inteligente que contiene más de 500 subespacios inteligentes, cada uno de los cuales representa una de las estaciones del sistema Bicing.

Creamos 7000 cosas virtuales correspondientes a las 7000 bicicletas en el sistema, cada una con un sensor virtual adjunto. Hay una diferencia a considerar en la infraestructura IoT entre el escenario real y nuestro modelo de WoT. Los sensores físicos, parte de la infraestructura IoT, están conectados físicamente a las estaciones del Bicing y recopilan la disponibilidad de una bicicleta en una ranura. Aunque consideramos que el contexto espacial

---

<sup>41</sup>Bicing Barcelona - <https://www.bicing.barcelona/>

es extremadamente importante, por definición, el WoT está centrado en las cosas. En nuestra propuesta, por lo tanto, la dinámica esperada puede escalar en tamaño agregando, por ejemplo, sensores de GPS, acelerómetros, sensores de monitor de frecuencia cardíaca de usuario a las bicicletas. Además, la visión de WoT vislumbra otorgar de inteligencia a las cosas a través de su representación en la Web. Como ya hemos comentado, las cosas virtuales están asociadas al espacio inteligente donde se ubican, y en nuestro modelo de movilidad simple las bicicletas se mueven de una estación a otra con una probabilidad uniforme. Suponemos que las bicicletas están conectadas a Internet. Cada vez que se crea una entidad, se genera una colección de documentos XML en el tiempo de simulación cero y hay un documento XML para cada entidad en la simulación. En otras palabras, construimos una colección de prueba de IR que comprende más de 14500 documentos XML. De esta colección, aproximadamente 3% son estáticos y el 97% son documentos XML dinámicos con información en tiempo real.

Mientras se ejecuta la simulación, cada movimiento de bicicleta se registra con una marca de tiempo en su documento XML con una actualización de su estación actual. La representación de bicicleta XML almacena cambios en la conexión a Internet y cambios generales en el estado de las cosas virtuales. En una capa inferior, la información sobre los sensores virtuales conectados a la bicicleta se guarda en documentos XML, registrando si la bicicleta está disponible o no para alquilar en ese subespacio y esta experimentación podría modelarse con una distribución de probabilidad binomial.

### 3.4.2. Escenario B: Sistema Lübeck

En el ejemplo anterior, mostramos un escenario con un tipo de sensor simple y el espacio inteligente único de Barcelona. El siguiente escenario es la instalación de Belgrado en el proyecto SmartSantander<sup>42</sup>. El sistema EkoBus se ha implementado en las ciudades de Belgrado y Pancevo y los vehículos de transporte público monitorean una serie de parámetros ambientales (CO, CO<sub>2</sub>, NO<sub>2</sub>, temperatura, humedad) en un área grande y brindan información adicional para el usuario final, como su ubicación y tiempos estimados de llegada a las paradas de autobús. Para simular este escenario, seleccionamos Serbia como zona inteligente para el modelo. Dos espacios inteligentes están contenidos en esta zona: las ciudades de Belgrado y Pancevo. Cada una de las rutas de ruta del sistema Ekobus se caracteriza como un subespacio inteligente. También modelamos cada parada de autobús como un subespacio inteligente. Cada Ekobus está representado por una cosa virtual y hay cinco en Belgrado y sesenta en Pancevo. Cada cosa virtual contiene seis tipos de sensores: NO<sub>2</sub>, CO, CO<sub>2</sub>, temperatura, humedad y ubicación GPS.

<sup>42</sup>SmartSantander - [www.smartsantander.eu/](http://www.smartsantander.eu/)

Durante la simulación de WoT, el movimiento de Ekobus se registra con una marca de tiempo en su documento XML y se actualizan su parada actual y su ruta. La representación XML de cada Ekobus almacena cambios en la conexión a Internet y cambios generales en el estado de las cosas virtuales. En una capa inferior, los documentos XML almacenan información sobre cada sensor virtual conectado al Ekobus, por ejemplo, si el Ekobus está en una parada de autobús en particular. En este escenario, en tiempo de simulación cero, hay una colección de prueba de IR que comprende 638 documentos XML. Esta colección representa una zona inteligente, dos ciudades inteligentes, sesenta subespacios de rutas inteligentes y ciento veinte subespacios de paradas de autobús inteligentes, sesenta y cinco cosas virtuales y trescientos noventa sensores virtuales.

### **3.4.3. Escenario C: Sistema de Vivienda Asistida WoT**

En los escenarios anteriores, describimos cómo se puede usar el modelo WoT propuesto para representar sistemas IoT/WoT del mundo real. Ambos emplean el modelo WoT en un enfoque similar y ambos tienen solo un tipo de cosa, aunque el segundo modelo usa diferentes tipos de sensores. En este ejemplo, presentamos un escenario con múltiples tipos de objetos y sensores virtuales.

Ambient Assisted Living (AAL) tiene como objetivo permitir que los adultos mayores vivan de forma activa e independiente en el hogar. Podemos modelar la casa con instalaciones AAL, como una zona inteligente que comprende espacios y subespacios inteligentes, por ejemplo, un estante. Además, el equipo médico se puede modelar como una cosa virtual, por ejemplo, un ECG portátil o un glucómetro. Vale la pena mencionar que se espera que la cantidad de dispositivos móviles médicos portátiles conectados a Internet se duplique en los próximos cinco años. Incluso medicamentos como la insulina pueden modelarse como algo virtual con algún tipo de conectividad o identificación inteligente para registrar su ubicación, cantidad o fecha de caducidad. En este escenario, un paciente o un familiar puede estar interesado en saber en tiempo real dónde encontrar el glucómetro o la insulina, o cuánta insulina le queda. Durante la simulación, construimos una colección de prueba RI que comprende documentos XML y estos representan una zona inteligente, diez espacios inteligentes, treinta subespacios inteligentes, veinte cosas virtuales y sesenta sensores virtuales.

## **3.5. Experimentación y Análisis de Resultados**

El diseño de la experimentación considera las siguientes variables fundamentales:

Para la colección sintética XML de prueba IR:

Variable	Ambiente A	Ambiente B	Ambiente C
Sistema Operativo:	Windows 8 Pro x64	Windows 10	Nube Google
Procesador:	AMD A6 2.10GHz	AMD Ryzen 5 2.00GHz	<i>Serverless</i>
Memoria RAM:	8,00 GB	8,00 GB	<i>Serverless</i>
Tipo de Recurso:	<i>Laptop</i>	<i>Laptop</i>	<i>App Engine</i>

Tabla 3.3 Ambientes de Experimentación usados con SIM.WoT

- Tamaño de la colección.
- Número de cambios en la colección.
- Tasa de crecimiento en la colección.

Para el rendimiento del simulador:

- Consumo de CPU.
- Consumo de memoria.
- Uso de disco / almacenamiento en nube.

Para la dinámica de WoT:

- Número de eventos en el tiempo para las cosas virtuales.
- Número de eventos en el tiempo para los sensores virtuales.

La Tabla 3.4 ilustra los resultados de un experimento realizado durante un período de 24 horas. En esta tabla se detalla el crecimiento del tamaño de la colección en relación con la cantidad de cosas virtuales y sensores, que también puede referirse a la cantidad de archivos y carpetas en la simulación. El tamaño inicial de la colección de prueba de IR se expresa en MB y depende de la cantidad de archivos y carpetas en el escenario de simulación. Este replica los resultados obtenidos sobre SIM.WoT V1, construido en Java [75]. En el cual los ficheros se organizaban en carpetas sobre el espacio de disco duro. En SIM.WoT V2, construido en Python, los ficheros se contienen en una instancia de Google Datastore en modo Firebase<sup>43</sup>, una solución de almacenamiento de documentos de tipo noSQL. Los ambientes de experimentación empleados tienen las especificaciones listadas en la Tabla 3.3.

Estimamos la tasa de crecimiento considerando el tamaño final después de que se detuvo la simulación. Calculamos un número promedio de cambios por segundo/minuto a partir de los archivos de registro que la simulación mantiene durante el período de tiempo de cada

<sup>43</sup><https://firebase.google.com/>

Ambiente A			
Variable	Escenario A	Escenario B	Escenario C
Tamaño Colección IR (Inicial en MB):	15.5	0.8	0.2
Tamaño Colección IR (Final en MB):	59.2	40.3	18.7
Radio de Crecimiento Colección en KB/m:	30.3	27.4	12.84
Cambios por Segundo:	107	14	9
Número de Ficheros:	12424	638	121
Número de Carpetas:	6424	248	61

Tabla 3.4 Resultados de Experimentación con SIM.WoT: Escenarios A, B y C

Ambiente B	
Variable	Escenario A
Tamaño Colección IR (Inicial en KB):	881.4
Tamaño Colección IR (Final en MB):	17.8
Radio de Crecimiento Colección en KB/m:	257.3
Cambios por Segundo:	221
Número de Ficheros:	14531
Número de Carpetas:	1

Tabla 3.5 Resultados de Experimentación con SIM.WoT: Ambiente B

evento. En el experimento sobre el ambiente A, se estima el valor de la colección a partir de los datos de almacenamiento de los ficheros XML/carpetas dadas por el sistema operativo. En los experimentos sobre los ambientes B y C, se estima el tamaño de colección a partir de la transmisión del fichero XML serializado en los *endpoints* creados para cada fichero sobre el dominio de SIM.WoT V2. Esto empleando la herramienta Postman<sup>44</sup>.

En la Figura 3.18 mostramos el resultado de la CPU y en la Figura 3.19 la memoria consumida para el experimento, junto con la velocidad de transferencia a disco duro en la Figura 3.20. Supervisamos los recursos de CPU y memoria asignados al proceso principal de Java en la computadora personal que alberga el motor de simulación.

Cuando está inactivo, estimamos un 0,18% del uso promedio de la CPU y 586 KB asignados al proceso de Java. Una vez que se ejecuta la simulación, la CPU promedio aumenta según la cantidad de entidades de simulación o la complejidad del escenario y también la memoria física utilizada por el proceso. Realizamos un seguimiento de un máximo del 4,25% del consumo de CPU en promedio por parte del motor de simulación con alrededor de 12500 entidades de simulación en ejecución, sobre el ambiente A.

<sup>44</sup><https://www.postman.com/>

Ambiente C	
Variable	Escenario A
Tamaño Colección IR (Inicial en KB):	0 (En Blanco)
Tamaño Colección IR (Final en MB):	270,7
Radio de Crecimiento Colección en KB/m:	17.8
Cambios por Segundo:	285
Número de Ficheros:	14531
Número de Carpetas:	1

Tabla 3.6 Resultados de Experimentación con SIM.WoT: Ambiente C

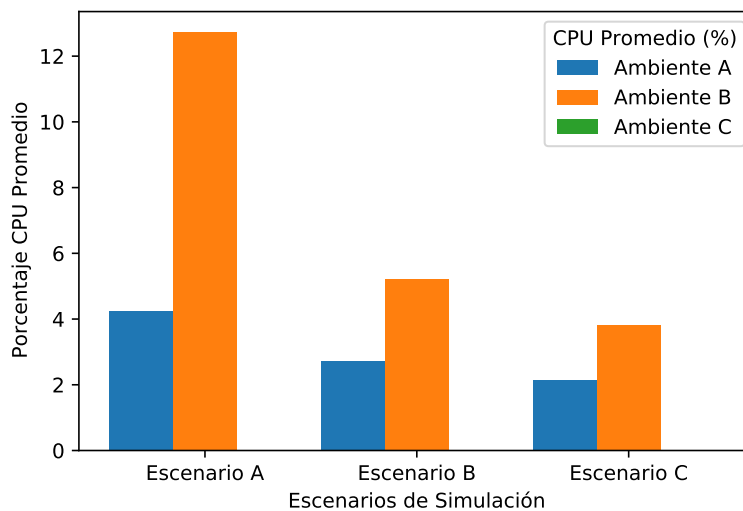


Figura 3.18 Resultados de Consumo de CPU Promedio (%).

*Nota:* El Ambiente C no es comparable con A y B. Aprox. 3.04 Mega Ciclos por Minuto.

Finalmente, la Figura 3.21 representa la dinámica del WoT en términos de la cantidad de eventos a lo largo del tiempo en las capas inferiores de nuestro modelo. En el Escenario A podemos ver que la dinámica del modelo es alta en ambos niveles debido a que la cantidad de entidades supera la cantidad de tipos de sensores. En este caso, podemos ver el impacto de un evento Tipo 5 (muestreo de un *vSensor*) ya que cada uno de los 7000 sensores en la simulación genera una medida por minuto en esta configuración de simulación. La dinámica del nivel *vThing* es bastante similar en todos los escenarios y hemos utilizado el mismo modelo de movilidad y conexión para cada uno de los tres escenarios, por lo que la dinámica es la misma. La dinámica de los eventos en las capas *vThing* y *vSensor* afecta directamente los cambios por segundo que asume la recopilación de pruebas de IR.



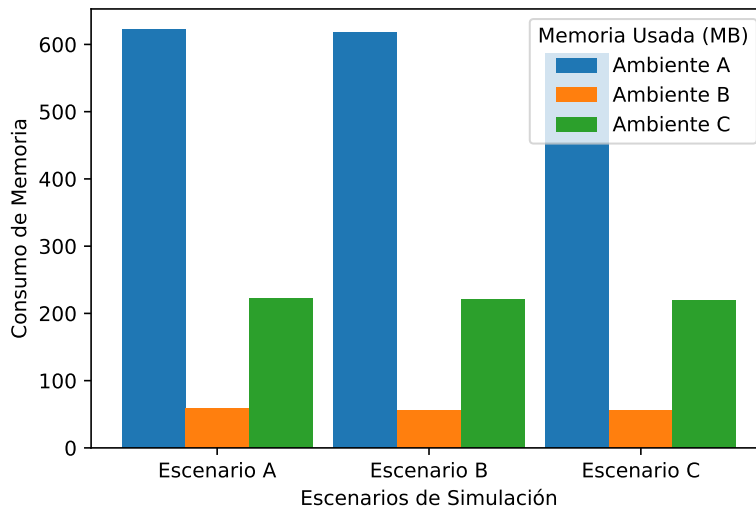


Figura 3.19 Resultados de Consumo de Memoria (MB).

Nota: El Ambiente C se compara con A y B mediante conversión de MiB a MB.

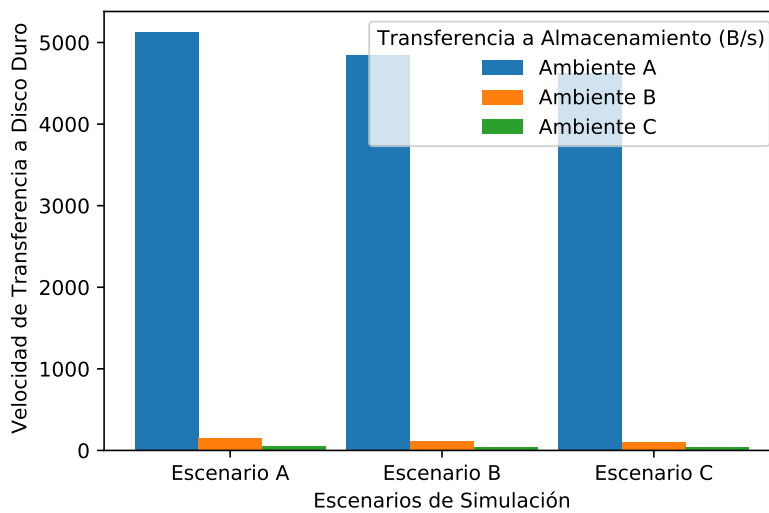


Figura 3.20 Resultados de Velocidad de Transferencia en Almacenamiento (B/s).

Nota: El Ambiente A a Disco Duro, se compara con B y C hacia *Firestore*.

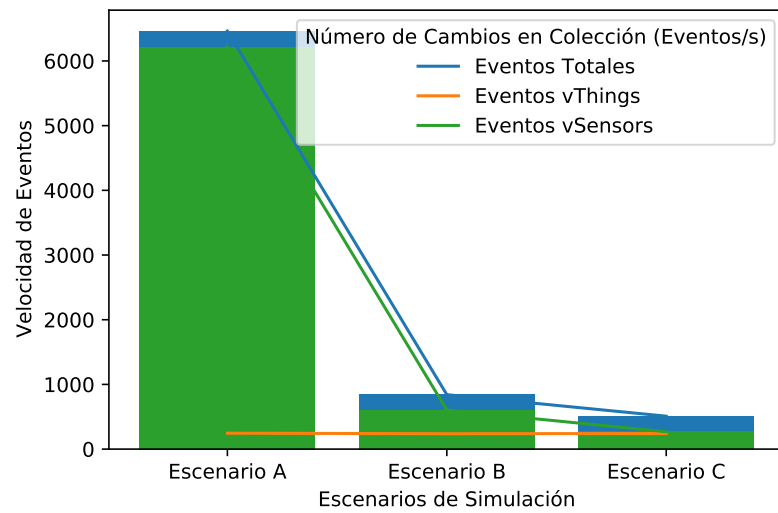


Figura 3.21 Resultados de Dinámica WoT en Ambiente A



# Capítulo 4

## Motor de Búsqueda para la Web de las Cosas (IR.WoT)

### 4.1. Arquitectura RI para WoT con base en la Nube

Una tendencia muy creciente es que el software integrado en micro-dispositivos equipados con sensores conectados a objetos en el mundo real se complemente por medio de servicios Web basados en la nube [44]. Esos servicios hacen uso de estándares web como HTTP, CoAP y XML. Los propósitos detrás de esta arquitectura mixta están dirigidos a ampliar las capacidades efectivas de procesamiento y almacenamiento de los dispositivos IoT, que por su naturaleza se basan en recursos limitados y de esta manera potenciar nuevas aplicaciones y servicios para WoT. De esta forma las cosas inteligentes pueden ser elementos de E/S federados, y ser constituidos de servicios y recursos distribuidos globalmente. Varios investigadores han señalado la creación de un nuevo paradigma, la *Nube de las Cosas* (CoT) [122], [86] exponiendo tanto problemas, retos y soluciones para la integración de este paradigma con otros. En [122], la arquitectura CoT propuesta es una plataforma IoT basada en la nube que se adapta a IaaS, PaaS y SaaS y esta basada en datos vinculados. [86] presenta en un *Survey*, retos, preguntas abiertas de investigación y herramientas hacia el uso de este tipo arquitectura como punto de encuentro entre IoT, WoT y Cloud, donde los servicios basados en la nube (XaaS) aceleran el desarrollo de aplicaciones de WoT e IoT y además permiten la composición dinámica de servicios. Vislumbramos un camino similar en el que los sistemas IR son un tipo de estas aplicaciones o servicios, que podrían ganar escalabilidad y enfrentar los nuevos desafíos apoyados en recursos en nube. Es así como pueden surgir otras nuevas perspectivas de la sinergia entre IoT, WoT, RI y Cloud.

Cabe citar que en términos generales, un sistema IR convencional realiza dos funciones principales [74]:

- **Indexación:** cuyo objetivo es establecer un conjunto de estructuras de datos con el fin de recuperar la información dada a un recurso en forma de una colección de documentos.
- **Consulta:** cuyo objetivo es usar estructuras de datos de índice para producir una lista ordenada de documentos para una consulta de usuario dada.

Consideramos el término documento como la granularidad de la información que se presentará al usuario de IR. El documento podría referirse no solo a ficheros de información textual sino también a tipos de información multimedia.

En esta sección, ofrecemos una propuesta para una arquitectura de servicio RI basada en la nube, que describe los componentes, las interfaces y las relaciones de alto nivel. Proponemos el siguiente modelo de arquitectura de alto nivel (ver Figura 4.1):

- **Bloque de indexación y análisis:**
  - Adquisición y recopilación de la información
  - Extracción y transformación de la información
  - Creación del índice
- **Bloque de consulta y recuperación:**
  - Interfaz de usuario y presentación
  - Interpretación de consultas
  - Recuperación y Modelos RI

En términos de las dimensiones de una Arquitectura de IRaaS, proponemos estructurar y discutir la arquitectura de RI a partir de tres (3) dimensiones principales:

#### **4.1.1. Dimensión I: Tipo de Información IoT|WoT**

Los datos proporcionados por WoT no son solo textuales. Además, se espera una gran cantidad de fuentes de datos a partir de características de interés del mundo real relacionadas con la cosa en sí o desde el entorno circundante. Va desde texto a imágenes, vídeo y datos de detección en tiempo real, entre otros. Proponemos estudiar la dinámica del WoT que impacta los sistemas RI en dos dimensiones principales: Tiempo y Datos.

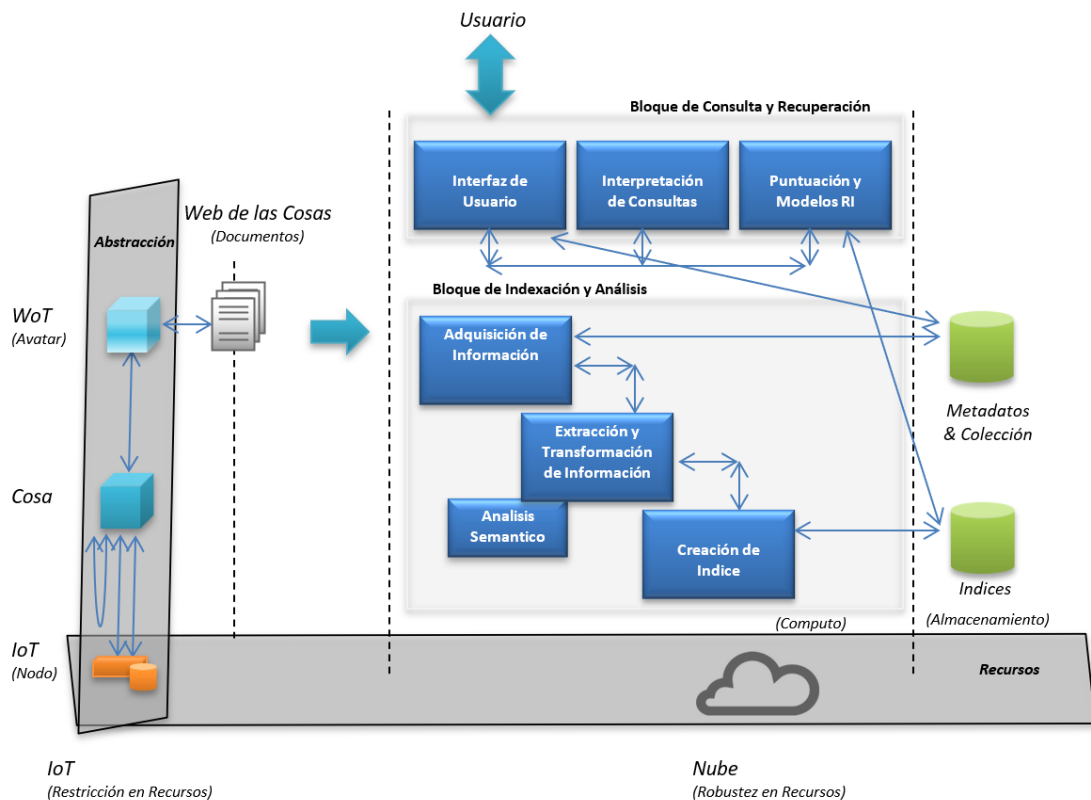


Figura 4.1 Arquitectura de RI como un Servicio (IRaaS) en Nube para IoT|WoT

### Desde la perspectiva del tiempo

Podemos evaluar el impacto de WoT en los componentes de un sistema IR en términos de velocidad de cambios en la colección de documentos y volatilidad de la información almacenada en la colección. Más allá de eso, en el mundo real, el contexto temporal adquiere relevancia y se convierte en parte de los eventos y la historia de las cosas. La arquitectura de WoT establece los eventos como un elemento fundamental. Es importante notar que el impacto de la dinámica WoT en el sistema IR, estaría estrechamente relacionado con los enfoques y recientes desarrollos para la denominada *RI temporal*. El carácter temporal de los datos está jugando un rol esencial en algunos campos de aplicación, por lo que la búsqueda de información temporal se ha convertido en un punto de interés en investigación actual en RI [64]. En este sentido, [80] argumenta que el alto dinamismo de la información en WoT es una razón para clasificar los datos provenientes por WoT como efímeros, que perderían validez o valor con el tiempo.

Los sistemas convencionales y existentes, entre ellos RI, que han habilitado el acceso a datos basados en ubicación, proporcionan solo capacidades de búsqueda de resultados relativamente estáticos, sin embargo, esperamos datos intrínsecamente efímeros de IoT|WoT,

y gran parte de estos datos no pueden indexarse fácilmente. Una arquitectura de RI basada en la nube IRaaS puede proporcionar un mecanismo para enfrentar la velocidad y la volatilidad de los datos. Una arquitectura IRaaS debe considerar los aspectos temporales y ser capaz de predecir o inferir información de eventos históricos y/o futuros mediante predeción.

### Desde la perspectiva de los Datos

Se espera un gran volumen y una rica variedad de datos. Además, el volumen y la diversidad de los datos generados en WoT crecen a un ritmo extraordinario [3]. Debido en parte a la dinámica, volatilidad y naturaleza *ad-hoc* de la mayoría de las redes y recursos subyacentes, en los que se construyen IoT y WoT. Esta dimensión presenta desafíos complejos para el procesamiento y la utilización de Big Data producidos por WoT [3]. Siguiendo estas ideas, los bloques de construcción básicos que componen un sistema RI pueden complementarse con paradigmas de nubes para realizar operaciones analíticas sobre *Big Data*.

#### 4.1.2. Dimensión II: Niveles de Abstracción y Alcance de Búsqueda

Desde el punto de vista de la arquitectura RI, se está recuperando información en tiempo cuasi-real de un mundo real mediante su avatar o gemelo digital WoT, que podría representarse con uno o varios documentos, ver Figura 4.1. En nuestra propuesta, hemos defendido la idea de que XML y JSON son los formatos de datos de representación más adecuado para IoT|WoT. Hay diferentes enfoques que enriquecen la WoT con anotaciones semánticas empleando diversos formatos como RDF, OWL o microdatos, pero XML proporciona simplicidad, mientras que al tiempo permite hacer anotaciones para así distinguir los atributos de las cosas.

Una característica distintiva de un sistema de IR para la WoT es su alcance de búsqueda, es decir, dada una consulta, qué tipo de información de resultado podría mostrar al usuario. Por ejemplo, una persona puede estar interesada en la temperatura promedio o las condiciones climáticas de un lugar en el momento actual o en el futuro. Entonces, en este caso, el resultado esperado podría ser una medida de temperatura promedio, como 20°C, estamos en el **nivel de búsqueda de datos**. En el siguiente nivel, supongamos que una persona está interesada en encontrar sensores de contaminación del aire dentro de una ciudad y en sus medidas específicas. En este caso, estamos en el nivel de **búsqueda de sensor y de actuador**. Un amplio número de investigaciones IR para el campo WoT pertenece a esta categoría. Además, una persona puede estar interesada en encontrar las cosas más cercanas que proporcionan un uso o servicio específico. Por ejemplo, una consulta de usuario podría ser, el objeto o cosa física más cercana que puede proporcionar capacidades de transmisión de video o

dónde está disponible la lavadora pública más cercana. En el último caso, estamos en el nivel de **búsqueda de nivel de cosas**. Una persona podría estar interesada en cualquier tipo de información en diferentes niveles de abstracción, desde datos brutos, información hasta conocimiento y sabiduría que involucra a todo el ecosistema WoT.

Resumimos esta dimensión, enfatizando que la arquitectura IRaaS debe proporcionar idealmente un servicio de búsqueda multi-modal o de cualquier cosa *Search for Anything, Everwhere, Anytime*. Entendiendo por "todo" como cualquier nivel de abstracción como sensores, grupo de sensores, cosas, grupos de cosas, tangibles o intangibles, lugares, eventos o servicios exhibidos por las cosas, todos ellos que coinciden con restricciones y limitaciones según las necesidades del usuario. Para construir un IRaaS de esta característica, el bloque de indexación y análisis debe ser capaz de crear un sistema de índice elástico y escalable. En ese sentido, proponemos usar mecanismos de almacenamiento en la nube para alojar el índice IRaaS. La creación del índice constituye un componente vital para ser evaluado considerando las condiciones de WoT.

### **4.1.3. Dimensión III: Computación, Almacenamiento, Visualización y Analítica**

La tercera dimensión está relacionada con los recursos. Es importante considerar que la escalabilidad requerida por la WoT y de un posible servicio IR embebido implica limitaciones y restricciones debido a las capacidades de computación, almacenamiento y comunicación de datos de los micro-dispositivos y nano-dispositivos conectados a las cosas. Los servicios basados en la nube pueden complementar las aplicaciones y servicios WoT entregando recursos cuasi-ilimitados. En las siguientes subsecciones se describen los bloques fundamentales de nuestra propuesta considerando la dimensión de los recursos.

#### **Recursos de Almacenamiento**

En primer lugar, proponemos mantener la colección de documentos en la nube, de esta forma sólo los documentos esenciales están en la memoria local de los dispositivos. Los elementos esenciales podrían ser documentos resumidos o comprimidos en la memoria de las cosas.

En segundo lugar, proponemos basar el índice principalmente en almacenamiento en la nube y crear algunos mecanismos para dividir o replicar la información esencial en la memoria de las cosas.

En tercer lugar, podrían existir otros repositorios de almacenamiento como vocabularios u ontologías, o la metadata asociada seleccionada para enriquecer con la anotación de los



documentos WoT. Es claro que existe un compromiso entre la distribución-descentralización de documentos descriptores en la nube y la validez de información en términos temporales, esto es que tan actualizados son los datos con respecto al estado en tiempo-real.

### **Recursos de Computación: Bloque de Indexación y Análisis de Información**

El tamaño (en términos reales de dispositivos) de IoT|WoT impone desafíos a todos los sistemas y componentes IR. Se sabe que las aplicaciones y contenido WoT necesitan estar bien organizadas para facilitar una recuperación altamente eficiente. En la propuesta BUDDYTHING de Giang et al. [37] debido a esa restricción de tamaño WoT se propone que los contenidos WoT se indexen por medio de un número de identificación de las cosas usando un esquema de números de serie basado en el Código Electrónico de Producto (EPC). Aunque indexar WoT usando un enfoque numérico ayuda a agrupar los contenidos del mismo modelo de cosas, como una forma de funcionalidad de agrupamiento, este carece de capacidades de enriquecimiento semántico y de rápida búsqueda de información. Los avances en los motores de búsqueda en tiempo cuasi-real no solo forman una frontera emocionante en la recuperación de información, sino que constituyen una base para un sistema de IR para el WoT. En nuestra propuesta, el bloque de indexación y análisis se compone de tres sub-componentes: i) el sub-bloque de adquisición información y recopilación, ii) el sub-bloque de extracción y transformación de información y iii) el sub-bloque de creación de índice. Las tareas que involucran la creación del índice para recuperar información de WoT de manera eficiente y efectiva son:

- Adquisición y recopilación de información. Existe una relación estrecha entre la descripción de las cosas para permitir que los motores de búsqueda indexen la WoT y para soportar consultas de búsqueda más ricas basadas en las relaciones entre las cosas. Por lo tanto, el rastreo (*crawling*) en WoT es parte del proceso de adquisición y es función del sub-bloque de adquisición y recopilación de información en nuestra propuesta. Se refiere al análisis recursivo de las interfaces (por ej., Recurso RESTful, RSS, WS \*) e hiperenlaces con otros recursos o servicios WoT. Se puede usar para extraer metadatos significativos para cosas que no proporcionan una descripción semántica explícita [78]. Durante el escaneo, este subcomponente IR puede usar solicitudes de tipo HTTP GET para obtener información de la cosa. Recientemente, aparte de HTTP, se han propuesto otros estándares, como XML, CoAP o MQTT. En este sentido, el bloque de adquisición y recopilación de información debe ser confiable para seguir diferentes tipos de interfaces de comunicación. En nuestra propuesta, HTTP REST puede verse como un mecanismo eficiente para intercambiar documentos y esquemas de tipo XML/JSON.

- **Extracción y Transformación de Información.** Este sub-componente tiene la función de analizar los documentos y utilizar la sintaxis propia del lenguaje de marcado XML para identificar la estructura de los documentos en la colección. En nuestra propuestas las etiquetas XML especifican la estructura de la información. Sin embargo, existen otros enfoques, como JSON-LD que se ha convertido en una propuesta atractiva como lenguaje de descripción de las cosas, en esta se combina la simplicidad de JSON con el poder de los datos vinculados y la Web semántica.
- **Creación de índice.** La creación del índice es el sub-componente principal en la etapa de indexación. Su objetivo es convertir la colección de documentos XML en una estructura de datos adecuada para recuperar la información de manera eficiente y efectiva. Una de las decisiones más importantes es la selección del tipo de índice.

### **Recursos de Computación: Bloque de Consulta y Recuperación**

En nuestra propuesta, el segundo grupo de sub-componentes está integrado por:

- **Interfaz de usuario y de presentación.** En términos de visualización, los sistemas de IR deben proporcionar una interfaz, donde los usuarios puedan realizar la entrada de consulta y posteriormente donde los resultados de IR se presenten.
- **Interpretación de consultas.** El lenguaje de consulta juega un papel importante en este sub-componente. La interpretación de consultas debe tratar las restricciones estructurales del modelo WoT, al tiempo que proporciona un mecanismo adecuado para que los usuarios realicen consultas significativas. Las interfaces basadas en el lenguaje natural son una opción en la que este sub-componente puede incrementar su complejidad para traducir a un lenguaje de consulta IR determinado.
- **Recuperación y Modelos RI.** El sub-componente de recuperación y modelos RI realiza la ordenación de documentos XML de acuerdo a la puntuación. Los sistemas RI no están siendo bien evaluados teniendo en cuenta el impacto de la dinámica de WoT. El sub-componente de clasificación y modelos IR es el núcleo de la etapa de consulta y recuperación en nuestro enfoque de arquitectura. Los modelos RI clásicos como los de correspondencia exacta, espacio vectorial o enfoques probabilísticos tienen sus propias características.

## 4.2. Arquitectura RI Modular para IoT|WoT

El trabajo doctoral de Guinard en [40] es considerado uno de los precursores de la arquitectura W3C WoT. Este presenta cinco capas de la Arquitectura WoT (ver Figura 3.1): accesibilidad, capacidad de búsqueda, uso compartido, composición y aplicación, donde la capa de búsqueda proporciona un mecanismo para encontrar información relevante relacionada con la cosa misma, los servicios expuestos por las cosas y aplicaciones. Nuestra contribución se enmarca en esta capa de capacidad de búsqueda, como se muestra en la Figura 4.2.

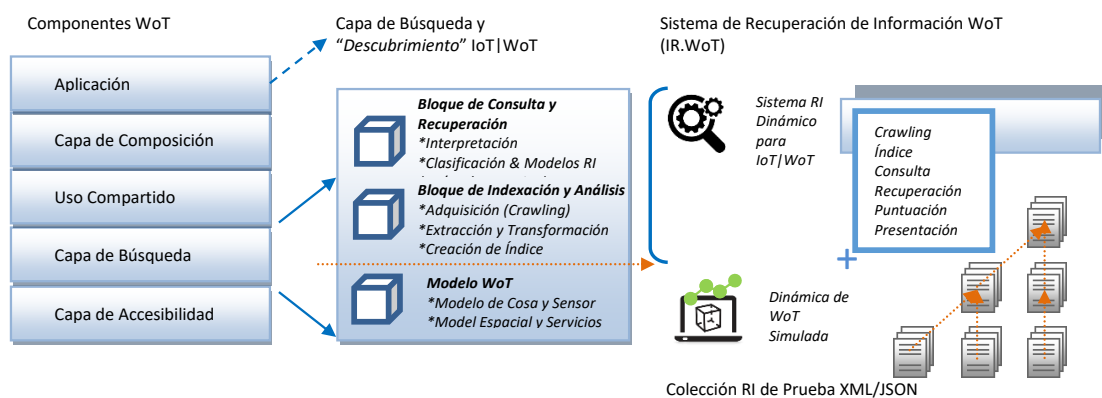


Figura 4.2 Arquitectura WoT Modular y Capas (Propuesta)

En [40], las capas de la Arquitectura WoT no están estrictamente definidas y literalmente no ocultan las capas anteriores. En términos de los principios de recuperación de información para WoT, utilizamos y basamos nuestro trabajo en la arquitectura de IR propuesta en la sección anterior, por lo que proponemos dos bloques de sub-componentes para la arquitectura de IR: el bloque de indexación y análisis y el bloque de consulta y recuperación.

Es un consenso entre los investigadores y los organismos de estandarización que los mecanismos de descubrimiento y búsqueda son funciones centrales para dentro de la arquitectura IoT|WoT. Por ejemplo, Shemshadi et. [118] proporciona una arquitectura de base modular para WoT. [118] destaca que los motores de búsqueda WoT en la literatura no se han implementado para datos a gran escala en entornos de producción. A pesar de los grandes esfuerzos de estandarización, todavía existen limitaciones como la disponibilidad pública de datasets de datos recopilados y de colecciones de prueba tipo IR [118], y además existe una gran barrera en la **Variedad** de los formatos de datos y de los modelos WoT. En adición, la **Volatilidad** de los datos impone una fuerte limitación en términos de la expiración rápida de estos debido a la naturaleza altamente dinámica de IoT|WoT impactando la relevancia

de la información recuperada. Las denominadas *Vs* de *Big Data* permean e impactan en el diseño eficaz y eficiente de los mecanismos de búsqueda [152] y se requieren nuevas técnicas y mecanismos para realizar sistemas IR para WoT a pesar de los sólidos principios fundamentales de búsqueda ya existentes [132].

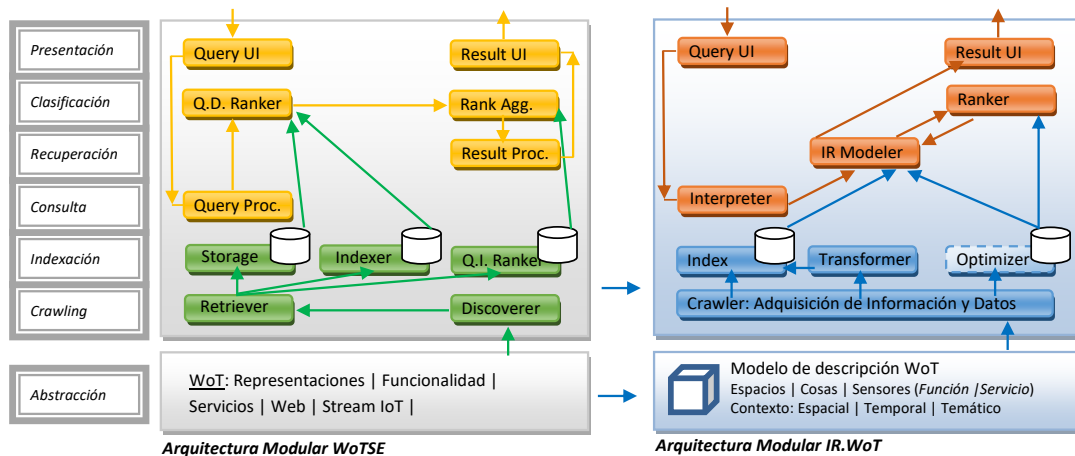


Figura 4.3 Arquitectura WoT Modular y sus Componentes (Propuesta)

### 4.3. Consideraciones de Diseño en el Módulo de Consulta y Presentación

El subsistema de presentación y visualización para WoT tiene como función considerar los resultados del proceso de ponderación y puntuación que son ordenados según su relevancia calculada a la consulta para renderizarlos en un documento HTML de vuelta al usuario final. Debido a las características propias de los documentos XML, JSON y de la información estructurada en general se conocen distintos mecanismos para la reorganización de la lista de resultados ordenados de forma que se eliminen los solapamientos de elementos XML y se unifique la información a ser presentada para facilidad de interpretación por parte del usuario. Dada la existencia de dependencia entre los resultados presentados y la satisfacción de usuario a una consulta es necesario considerar estrategias de presentación en el subsistema eliminando la redundancia de información, maximizando la respuesta a una consulta con documentos de similitud en su contexto al tiempo que se proporcione el mejor un punto de entrada dentro de los documentos. En el desarrollo de los escenarios de evaluación en INEX<sup>45</sup> se denominan estos problemas de investigación como la recuperación focalizada, la

<sup>45</sup>Iniciativa para la evaluación de recuperación XML - <https://inex.mmci.uni-saarland.de/>

relevancia en el contexto y el mejor en el contexto. Desde una perspectiva del estado del arte dado el universo de investigaciones en WoTSE se han considerado diferentes estrategias de presentación de los resultados con diferentes formatos y alcance que podrían complementar las estrategias primarias de presentación provenientes de la recuperación XML.

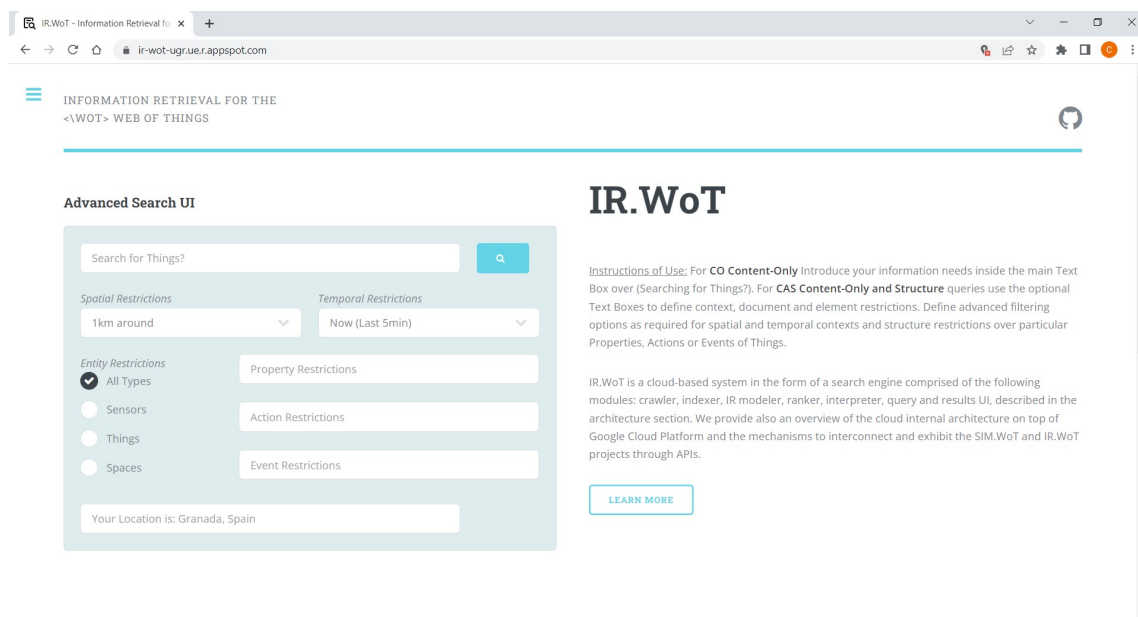


Figura 4.4 Interfaz de Consulta propuesta para IR.WoT

### 4.3.1. Interfaz de Usuario UI de Consulta

En la Figura se ilustra la propuesta de interfaz de consulta UI para el sistema RI para WoT propuesto (IR.WoT), el cual considera que un sistema WoTSE debería exhibir al menos un mecanismo de interacción H2M de forma que se capture la información de consulta que considere tanto consultas del tipo CO, como CAS. En nuestra propuesta hemos dispuesto de:

1. Un menú principal que contiene hiperenlaces a la página de inicio de consulta del sistema IR.WoT, página de inicio del simulador de eventos discretos al cual hemos renombrado como SIM.WoT, página para la configuración de variables de entorno propias del sistema de recuperación para la selección de mecanismos alternativos inherentes a su funcionamiento interno, página para visualización de resumen de variables de desempeño del sistema y por último una página con descripción de alto nivel de la propuesta.

2. Cuadro de texto (Text Box) para la captura de la consulta en lenguaje natural de solo contenido (CO) sobre toda la colección de documentos indexada.
3. Restricciones en la consulta referentes al contexto espacial y temporal. Con la posibilidad de seleccionar:
  - Rango circular de búsqueda = Aquí, a 1Km, a 10Km, a 100Km, . . . , en cualquier lugar.
  - Rango temporal histórico desde ahora = Ahora, hace una hora, . . . , en cualquier tiempo.
4. Restricciones de estructura referentes al tipo de documentos por ende al tipo de entidad buscada o también interpretada como restricciones en el alcance de la búsqueda. Con la posibilidad de buscar sobre toda la colección y todos los tipos de documentos y entidades, o filtrar los resultados de búsqueda sólo sensores, solo cosas o espacios. Opcionalmente, se contempla extender para desarrollo futuro la búsqueda sobre elementos XML relacionados con personas de contacto o personas propietarias o responsables de las cosas, espacios y sensores del sistema WoT.
5. Restricciones de contenido y estructura (CAS) sobre elementos XML que componen los documentos de la colección en lo referente a propiedades, acciones y eventos.

### **Descripción General de la Interfaz Principal**

El sistema IR.WoT le pedirá que comparta su ubicación actual, por lo que los resultados de la búsqueda se muestran centrados en su ubicación. De lo contrario, los resultados de la búsqueda se centrarán asumiendo que Granada, España es su ubicación geográfica actual, ver Figura 4.5.

La barra de navegación (panel izquierdo) contiene enlaces acerca de esta investigación científica e información acerca del sistema de recuperación de información. En los submenús se encuentran a) un enlace para retorno al /Home o interfaz de búsqueda, b) un resumen de las instrucciones de uso del buscador, c) información sobre la arquitectura general del sistema IR.WoT y d) algunos ajustes avanzados del funcionamiento interno del motor de búsqueda. Este menú puede ser replegado para maximizar el espacio de la interfaz de búsqueda.

El sistema IR.WoT, así como el simulador de eventos discretos para la Web de las Cosas SIM.WoT, que genera datos sintéticos en formato XML que alimenta los escenarios simulados en los cuales se buscará se encuentran publicados como código abierto bajo licencia GNU *General Public License v3.0* en el enlace indicado por el icono de *GitHub* en la esquina superior derecha.

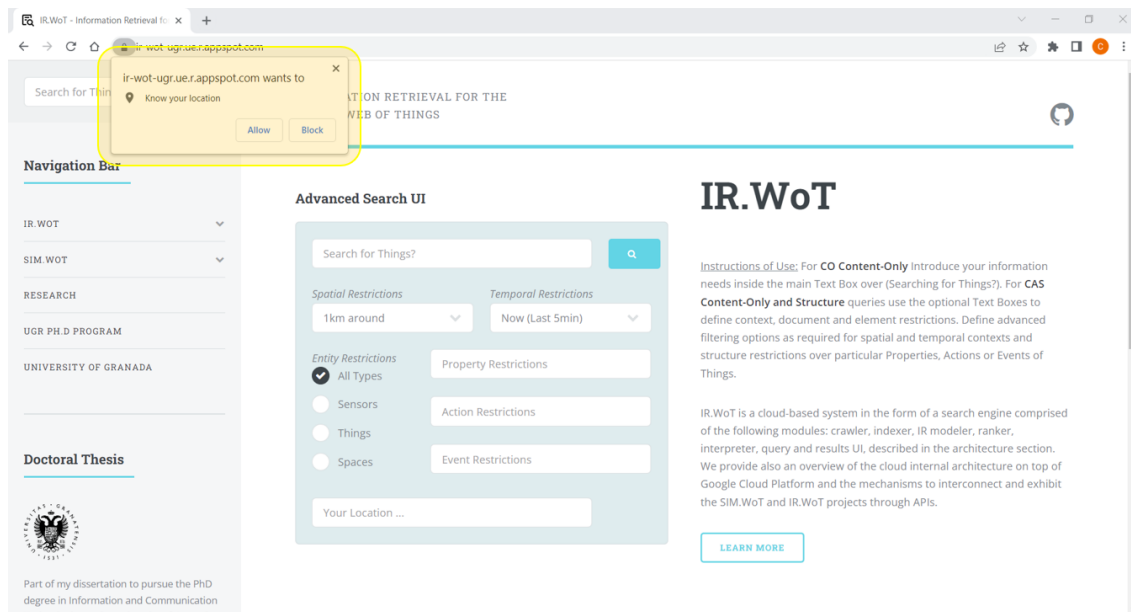


Figura 4.5 Interfaz de Consulta Captura de Ubicación de Usuario

### 4.3.2. Sub-Componente de Interpretación de Consultas para WoT

El procesamiento y los lenguajes de consulta en la recuperación XML han evolucionado desde sus tempranos orígenes empleando XML-QL<sup>46</sup>, XSLT<sup>47</sup>, XQL<sup>48</sup>, hasta llegar a lenguajes de facto aun empleados como XPath<sup>49</sup> y XQuery<sup>50</sup>. Estos fueron posteriormente extendidos para ser adaptados desde la centricidad en datos a la centricidad en texto con lenguajes como NEXI [135], ELIXIR [12], XIRQL [35] y XQuery-FT<sup>51</sup>. Convencionalmente, pueden identificarse dos categorías de consulta en contenido CO (*Content-Only* por sus siglas del inglés) y contenido y estructura CAS (*Content-and-Structure* por sus siglas del inglés). Es de nuestro interés el expresar consultas con restricciones en contenido y estructura con base en la ruta y con base en cláusulas. Los grandes exponentes de lenguaje de consulta XML de facto para estas categorías son NEXI y XQuery-FT.

El lenguaje de consulta NEXI (*Narrowed Extended XPath I* por sus siglas del inglés) desarrollado por la comunidad INEX [135] tiene como base las expresiones XPath para acceder y navegar dentro de los componentes y elementos de la colección de documentos XML. Debido a que la contención exacta de los elementos puede ser menos crítica en

<sup>46</sup>XML-QL - <https://www.w3.org/TR/1998/NOTE-xml-ql-19980819/>

<sup>47</sup>XSLT - <https://www.w3.org/TR/xslt-30/>

<sup>48</sup>XQL - <https://www.w3.org/TandS/QL/QL98/pp/flab.txt>

<sup>49</sup>XPath - <https://www.w3.org/TR/xpath/>

<sup>50</sup>XQuery - <https://www.w3.org/TR/xquery-31/>

<sup>51</sup>XQuery Full-Text - <https://www.w3.org/TR/xquery-full-text/>

las aplicaciones de IR, NEXI solo admite la notación de descendiente o auto (//) para las rutas. Para especificar la recuperación clasificada, NEXI reemplaza la función *contains* con *about*. En la propuesta IR.WOT se capturan a través de la interfaz de consulta los campos y restricciones de cara al usuario que son entonces traducidas de lenguaje natural a consultas en lenguaje NEXI. Convencionalmente, la traducción puede realizarse con varios métodos dada la estructura y contenido de los documentos XML de la colección. Para el modelo propuesto se tienen los siguientes tipos de consultas NEXI:

### Consultas Simples de la forma //A[B]

Donde A es el elemento XML objetivo de la consulta a ser recuperado y B un filtro o restricción de contenido (CO). De tal forma que una consulta puramente (CO) que recuperé cualquier tipo de documento y elemento XML dentro de la colección con el filtro proporcionado por el usuario en el campo principal de consulta sería traducido a:

$$// * [about(.,co\_query)] \quad (4.1)$$

Como ejemplo podríamos citar la necesidad general de información de una persona realizando turismo en una ciudad que aún no conoce:

$$// * [about(.,tourism\ in\ Barcelona) and about(.,bike\ riding)] \quad (4.2)$$

Se han considerado restricciones al tipo de documento cuya identificación se encuentra encapsulada en el atributo "type" del elemento raíz "vX" de los documentos XML. De forma que se establece un filtro de atributo con su correspondiente consulta NEXI:

$$//vX[@type = virtualSensor and about(.,co\_query)] \quad (4.3)$$

$$//vX[@type = virtualThing and about(.,co\_query)] \quad (4.4)$$

$$//vX[@type = intelligentZone or @type = smartSpace and about(.,co\_query)] \quad (4.5)$$

Para el primer caso se recuperarían documentos/elementos de tipo sensor virtual relacionados con la consulta de solo contenido, en el segundo caso solo documentos de tipo cosa virtual y por último la consulta NEXI consideraría cualquier tipo de documentos descriptor de zonas, espacios y subespacios inteligentes. Empleando el ejemplo anterior, si el usuario de la consulta quisiera únicamente lugares turísticos o de renta de bicicletas relacionados con su consulta emplearía la restricción de estructura encapsulada por el atributo de tipo de documentos que pertenece a elementos raíz.



### Consultas Compuestas de la forma //A[B]//C[D]

Proponemos adicionalmente restricciones de contexto tanto espaciales como temporales. De forma que a través del objeto de selección en la interfaz de consulta se establecen filtros de espacio y de tiempo. De forma convencional los motores de búsqueda actuales no establecen restricción, lo que podría ser traducido a resultados enmarcado en cualquier lugar “*anywhere*” y en cualquier tiempo “*anytime*”. Sin embargo, el paradigma WoT ha orientado a los acercamientos WoTSE hacia un enfoque en el aquí “*here*” y ahora “*now*” [80]. Considerando estos nuevos elementos se crea la posibilidad de incluir filtros aritméticos sobre espacio y tiempo. En nuestra propuesta consideramos que la localización actual en tiempo real del usuario podría ser capturada por el sistema IR.WoT si es compartida dada la información de sensor de sistema de posicionamiento global GPS en un dispositivo o es una variable de entorno en el menú de configuración del sistema IR.WoT automáticamente resulta dada la dirección IP Pública empleada por el dispositivo ejecutando el motor de búsqueda.

Consultas tipo para restricciones espaciales y temporales:

$$// * [about(.,co\_query) \text{ and } ./event/eventTime \leq \{time\}] \quad (4.6)$$

$$// * [about(.,co\_query) \text{ and } ./property/geocoordinate \leq \{distance\}] \quad (4.7)$$

En la interfaz de consulta existen campos para la captura de información asociada a la construcción de filtros de estructura sobre los elementos XML propiedades, acciones y eventos. Estas permiten indagar sobre la estructura de los documentos XML y estructurar consultas NEXI de mayor complejidad proporcionando al usuario de la posibilidad de opciones más enriquecidas en el proceso de búsqueda de información sobre un sistema WoT. De este modo, se presenta la posibilidad de encontrar cosas, sensores o espacios con características particulares, o la posibilidad de ofrecer servicios o funcionalidades en la forma de acciones o sobre estados en los que se encuentran dado su historial de eventos.

Consultas tipo de estructura y contenido:

$$//vX[@type = virtualSensor \text{ and } about(.,co\_query)]//events[about(.,cas\_query)] \quad (4.8)$$

$$//vX[@type = virtualThing \text{ and } about(.,co\_query)]//events[about(.,cas\_query)] \quad (4.9)$$

$$//vX[@type = smartSpace \text{ and } about(.,co\_query)]//events[about(.,cas\_query)] \quad (4.10)$$

### Interpretación de Consultas CAS: SCAS y VCAS

En nuestra propuesta, el módulo de interpretación de consultas emplea un formulario que agiliza la captura de consultas CAS y permite la obtención de consultas tipo NEXI acordes a la estructura de los documentos XML, sin que el usuario tenga que ser completamente consciente de la interpretación de las etiquetas de los elementos XML en la estructura. Este módulo implementa la siguiente estrategia:

- Si el elemento buscado satisface la ruta de recuperación, solo se devuelve el elemento en sí, evitando la redundancia al englobar todos los elementos bajo el documento de representación XML.
- De lo contrario, se devuelve el padre más cercano que cumple la condición de ruta de recuperación. En todos los casos, se recupera como máximo un elemento. Igualmente, evitando duplicación de resultados y realizando la ordenación de mayor a menor relevancia.
- Aquí solo se recuperarán los elementos de documento o el documento XML que contengan elementos sobre la consulta, luego de aplicar las restricciones.

Dado, que el procedimiento seguido para producir una recuperación para las consultas tipo NEXI es más complejo que una consulta pura de contenido. Una consulta CAS siempre contiene uno o dos filtros. Un filtro se especifica con corchetes ( [...] ) y puede contener una o más cláusulas *about* que definen los criterios de búsqueda basados en el contenido de la necesidad de información del usuario final [138]. Además, la consulta CAS tiene que terminar con un filtro, limitando las formas posibles de una consulta CAS a: C[D] o A[B]C[D] [135].

De acuerdo a la especificación original en [135], “*Las consultas pueden contener requisitos estructurales explícitos o implícitos. Tal consulta podría surgir si un usuario conoce la estructura del documento. Para responder a una consulta CAS, un motor de recuperación debe deducir la necesidad de información de la consulta, identificar elementos que coincidan con los requisitos estructurales y devolverlos ordenados de mayor a menor relevancia. Las consultas CAS se pueden interpretar de dos maneras, ya sea estrictamente (SCAS) o de forma flexible o vaga (VCAS).*”

Hemos implementado una combinación de interpretación estricta y vaga. De modo general cuando se utiliza una interpretación vaga de las restricciones estructurales, en lugar de una interpretación estricta, la lista ordenada simplemente se amplía con cada uno de los resultados intermedios que aún no están incluidos en la ponderación. Como resultado,

se ignoran algunas sugerencias estructurales. La lógica detrás de la interpretación de las consultas CAS es como sigue:

1. Restricción de Tipo de Entidad → SCAS.
2. Restricción Espacial → SCAS.
3. Restricción Temporal → SCAS.
4. Propiedades → VCAS.
5. Acciones → VCAS.
6. Eventos → VCAS.

En el Apéndice B se presentan algunos ejemplos de búsqueda y se describen brevemente algunos escenarios simulados sobre los cuales se realizan estos tipos de consultas de búsqueda, empleados con un conjunto de usuarios finales para los experimentos en el Capítulo 5.

### 4.3.3. Interfaz de Usuario UI de Resultados

Convencionalmente, los motores de búsqueda web exhiben una lista de resultados en orden decreciente de relevancia para la consulta. Sin embargo, dada la naturaleza intrínseca de los modelos de descripción de la web de las cosas con elementos anidados interdependientes en formatos XML/JSON, una lista de elementos ordenados no es la forma adecuada de mostrar los resultados para WoT. Es ampliamente aceptado, el hecho que la presentación de elementos individuales como resultado no es una de las buenas prácticas en la recuperación XML [61]. Esto puesto que los elementos XML dentro de un documento no son completamente independientes unos de los otros.

El módulo de la interfaz de usuario de resultados debe evitar la superposición de elementos, lo que se traduce en reducir o eliminar la redundancia de información. Como se indicó anteriormente, se pueden utilizar tres estrategias principales para lograr este objetivo: i) la recuperación enfocada, ii) la relevancia en el contexto y iii) lo mejor en el contexto.

- El primero se refiere al hecho de que considera que la relevancia de un documento/elemento para una consulta no es la única tarea a considerar, por lo que un enfoque en el resultado debe eliminar o disminuir la redundancia.
- El segundo aborda la necesidad del usuario de recibir los elementos recuperados. dentro de su contexto, esto es, la estructura de pertenencia de un elemento a su correspondiente documento.

- El último tiene como objetivo identificar puntos de entrada a la información en el documento o recuperar un elemento completo.

En general, estas tareas contribuyen a aumentar el índice de satisfacción del usuario en el momento en que podría aumentar el tiempo de recuperación y, en ese sentido, aplicar una forma óptima para que se realice de manera eficiente. La independencia o separación de la interfaz de usuario de consultas y resultados podría dar lugar a una divergencia en los enfoques conocidos. En el caso de nuestro modelo considera el contexto espacial y temporal, estas variables deben permear la lista de resultados. Es reconocido el hecho, que estas tareas podrían influenciarse unas a otras desde una perspectiva algorítmica [61]. En nuestra propuesta, hemos decidido asignar un valor de prioridad a cada estrategia de presentación de modo que la lista de resultados sea consistente entre todos los tipos de consultas.

### Lista Ordenada de Resultados y Visualización

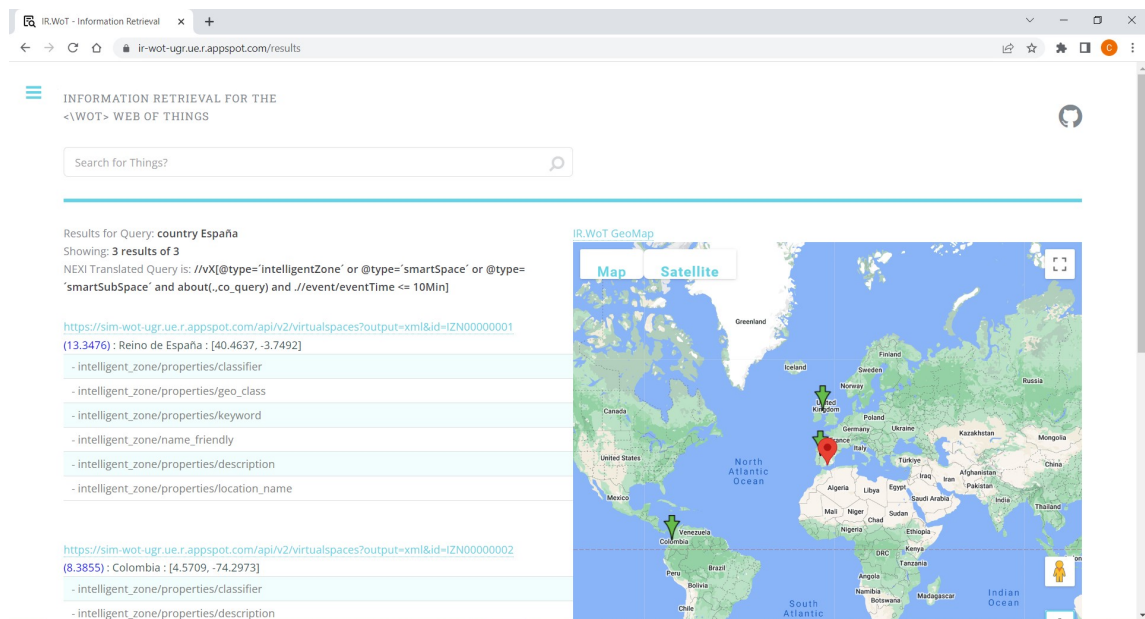


Figura 4.6 Interfaz de Usuario UI de Resultados

La Figura 4.6 ilustra un ejemplo de la interfaz de resultados del sistema IR.WoT. En esta se puede apreciar tres (3) áreas importantes:

- Información General de Resultados. Se entrega al usuario la siguiente información:
  - Se parafrasean los términos de consulta ingresados por el usuario.

- Número de Resultados presentados en la página y el número total de coincidencias. O en caso contrario, un mensaje si de ninguna coincidencia encontrada.
  - Una indicación de la traducción a NEXI de las restricciones ingresadas por el usuario.
- Lista Ordenada de Resultados. Cada ítem de resultado se presenta con la siguiente información de detalle.
- Hiper-enlace hacia el documento XML coincidente.
  - El valor de relevancia para cada documento, esto sólo para fines prácticos y de prototipo.
  - Nombre descriptivo corto de la entidad correspondiente.
  - Coordenadas geográficas de la entidad correspondiente a tiempo de consulta.
  - Estado general de la entidad (cosas y sensores virtuales) a tiempo de consulta.
  - Lista ordenada de mayor a menor relevancia de los elementos XML internos coincidentes a la consulta.
- Mapa de Ubicación de Resultados.
- Centro del mapa corresponde a la ubicación actual del usuario o de consulta.
  - Marcadores geo-localizados para cada uno de los resultados de consulta encontrados.
  - Radio de búsqueda seleccionado por el usuario en la consulta.

#### **4.4. Consideraciones de Diseño en el Módulo de Indexación y Análisis de Información**

Las soluciones de RI se basan normalmente en índices estáticos debido a las características convencionales de la conocida Web, por lo que no son adecuadas para el dinamismo de WoT. Por otro lado, la información estructurada en forma de documentos XML/JSON con jerarquías y relaciones implícitas ha impactado las investigaciones recientes de recuperación XML.

Dichos mecanismos XML no consideran la característica de WoT y se requiere una adaptación de sus componentes fundamentales. Nuestro modelo de descripción de cosas se basa en XML, por lo que el indexador se basará en estructuras de datos capaces de

proporcionar de manera eficiente operaciones de inserción, modificación, eliminación, entre otras.

Un estudio en términos de la evaluación de desempeño de las diferentes opciones de estructuras de datos para construir un esquema de índice para XML se presenta en [20]. En general, los sistemas de recuperación de información tienen como base un esquema de índices. Los índices invertidos pueden analizarse desde dos perspectivas: desde el punto de vista estructural con enfoque en los componentes del índice y desde el punto de vista operacional con enfoque en las fases en el ciclo de vida del esquema con sus operaciones esenciales en cada fase [7]. El esquema de índice invertido se compone de un diccionario de término vinculado con unas listas de publicación que en últimas contiene información de posicionamiento del término en la colección y en el documento en conjunto con estadísticas de ocurrencia. El diccionario permite la búsqueda de términos del vocabulario en la colección de manera eficiente como estructura superior a las listas de publicación. En adición al diccionario de términos y las listas de publicación es posible que un sistema de indexación para WoT considere índices adicionales para el almacenamiento eficiente de información de los recursos IoT, las cosas y en general el sistema WoT.

Un esquema de índice invertido puede variar según las características de dinamismo de la colección. En un índice invertido estático, se presume que la colección nunca tendrá cambios. Desde el punto de vista operacional una colección de documentos dinámica puede considerar las operaciones de inserción, borrado y modificación de documentos. A fin de mantener el índice actualizado se construyen estrategias de mantenimiento de índices invertidos.

Convencionalmente, las colecciones dinámicas se pueden tratar con una estrategia de mantenimiento semi-estático, por lo que la actualización del índice se ve como una operación de eliminación seguida de una inserción que puede demorar horas o días, excepto para los documentos de la colección prioritaria. La estrategia de mantenimiento incremental permite la inserción de nuevos documentos, pero sin la opción de modificar o eliminar documentos. Finalmente, la estrategia de mantenimiento dinámico considera la actualización arbitraria y frecuente de los documentos, sin embargo este campo no ha sido bien estudiado ni implementado en los mecanismos clásicos de indexación XML [7], [150].

#### **4.4.1. Sub-Componente de Indexación para WoT**

Como se muestra en la Figura 4.7, nuestra propuesta es analizar la combinación de tres estructuras de datos con la aplicación de tres estrategias dinámicas XML combinadas. En las siguientes subsecciones vamos a describir la decisión detrás de las combinaciones en estructura y operaciones desde las perspectivas teórica y pragmática. En [20] se presenta un estudio en términos de la evaluación del desempeño de las diferentes opciones de estructura

de datos para construir un esquema de índice para XML. Es un punto de partida para la discusión de las estructuras de datos como ladrillos para un esquema de indexación dinámica de WoT.

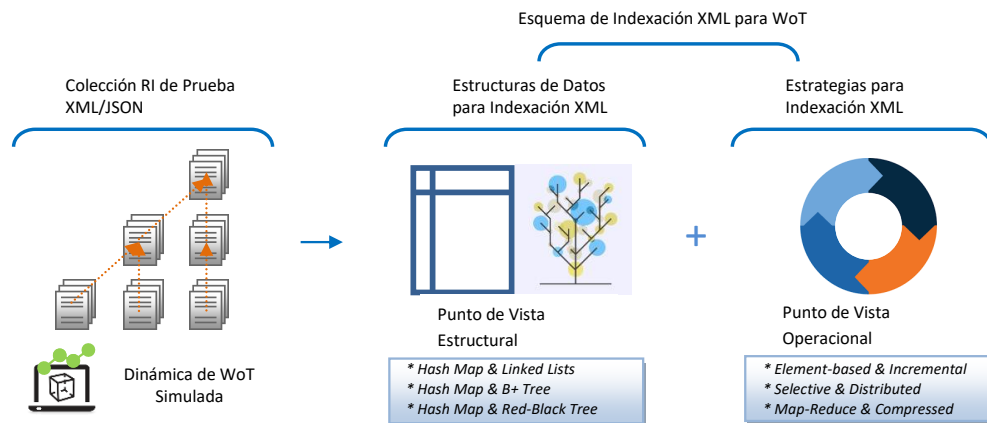


Figura 4.7 Esquema de Indexación propuesta para IR.WoT

### Estructuras de datos para diccionarios y publicaciones

Como se mencionó anteriormente, el diccionario que compone el índice invertido se puede almacenar en una tabla *hash* o una estructura similar, y la lista de publicaciones para cada término  $t$  se puede almacenar en una estructura de arreglo con longitud fija  $l_t$ . [20] presenta una evaluación dirigida por el tiempo de construcción del índice y el espacio de almacenamiento del índice donde hemos seleccionado las tres (3) principales estructuras combinadas con el mejor desempeño reportado:

- Mapa *hash* y Listas Vinculadas
- Mapa *hash* y Árbol B+
- Mapa *hash* y Árbol Negro-Rojo

Es importante señalar que en los experimentos de [20] se construye un índice directo sin su inversión. Por otro lado, los datos almacenados en las listas de publicaciones siempre constituyen la mayor parte de todos los datos del índice. En su totalidad, suelen ser demasiado grandes para almacenarse en la memoria principal y deben mantenerse en el disco. Habiendo dicho esto, en las siguientes subsecciones vamos a describir la decisión detrás del diseño interno de esas estructuras de datos para la indexación dinámica de XML.

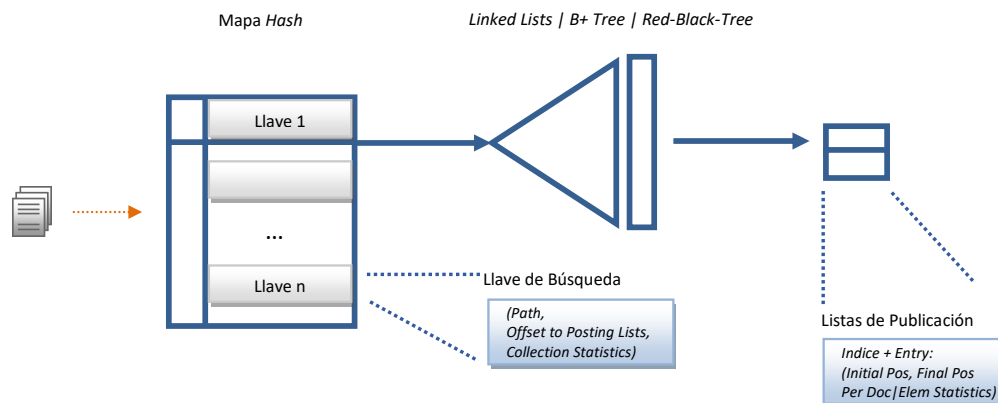


Figura 4.8 Estructura de Indexación propuesta para IR.WoT

### Estrategias de indexación dinámica de XML

La coexistencia de contenido y estructura en XML es un factor preponderante a considerar para decidir sobre las bases de la estrategia. Además, debido a que nuestro modelo de descripción se basa en una representación XML, la estrategia de indexación XML debe tener como objetivo definir de manera eficiente los elementos que se almacenarán en el índice y el cálculo de estadísticas [61]. La tarea principal del indexador es analizar el formato XML, elegir el elemento, construir el índice y finalmente calcular las estadísticas de términos  $tf$ ,  $idf$  a nivel de elemento. En los sistemas RI clásicos basados en XML se pueden identificar cinco estrategias principales de indexación [61]:

- Basado en elementos: estrategia altamente redundante que permite la recuperación en cualquier nivel, indexando todos los elementos compensados con la complejidad del espacio del Índice. Sin embargo, para los elementos anidados y los cálculos de  $ief$  no existe un consenso en la metodología de estimación. Por lo que se suele referir a la aplicación directa y por elemento del algoritmo BM25.
- *Leaf-Only*: estrategia no redundante que permite la recuperación a nivel de hoja solamente, compensando la complejidad temporal del cálculo de relevancia del elemento hoja más la propagación ascendente eficiente de la puntuación.
- Basado en agregación: indexación de elementos de hoja única mediante una compensación de representación agregada con el grado de influencia del elemento.
- Selectivo: Elementos de indexación con un número de palabras por encima de un umbral dado, o de un tipo, o fragmentación disjunta de compensación con combinación de estrategias para calcular estadísticas de términos.



- Distribuido: índice para cada tipo de elemento con estadísticas separadas, compromiso con la complejidad del espacio y paralelismo en el momento de la recuperación.
- Estructura: estadísticas de pares estructura/términos para captar la importancia de la estructura.
- *Map-Reduce*: marco de procesamiento de datos distribuidos a gran escala y modelo de programación utilizado para acelerar y administrar la creación y el mantenimiento de índices en los principales motores de búsqueda actuales.
- Comprimido: para almacenar y transferir de manera eficiente los datos del diccionario y el propio índice.

### **Perspectiva Teórica de Indexación para WoT**

Desde un punto de vista arquitectónico, las estructuras de indexación XML se clasifican en categorías en función del tipo de información almacenada en el diccionario, asumiendo el papel de una llave de búsqueda. Las categorías se resumen en la Tabla 4.1. El objetivo detrás de las estructuras de datos es reducir el espacio de búsqueda para aumentar el rendimiento del mecanismo de recuperación de información. Hemos seleccionado la arquitectura de unión estructural debido a que nuestro sistema IR.WoT maneja consultas de contenido directo CO y consultas condicionales en contenido y estructura CAS proporcionadas por el usuario de WoT en lenguaje natural. La llave de búsqueda almacena datos en forma de texto y/o datos numéricos, como se muestra en la Figura 4.8. El análisis de complejidad temporal de las estructuras de datos combinados se muestra en la Tabla 4.2.

Categoría	Llave de Búsqueda	Consulta	Limitaciones
Resumida	Camino Root-to-Leaf	Path P/C	Post-Procesamiento para caminos A/D
Unión Estructural	Elementos XML	Elementos Tokenizados	Operaciones de unión extra
Secuencial	Secuencias Codificadas	Por Coincidencia	Consultas Condicionales

Tabla 4.1 Límites y Características del Esquema de Indexación

P/C significa *Parent-Child relationship*.

A/C significa *Ancestor-Descendant relationship*.

Estructura del Índice	Inserción	Borrado	Actualización	Búsqueda
Mapa <i>hash</i> y Listas Vinculadas	O(1)	O(1)	O(1)*	O(n)
Mapa <i>hash</i> y Árbol B+	O(1)	O(1)	O(1)*	>O(Log n)**
Mapa <i>hash</i> y Árbol Negro-Rojo	O(1)	O(1)	O(1)*	O(Log n)

Tabla 4.2 Análisis de Complejidad de Estructuras de Datos

*Notas:* (\*) Operación de actualización vista como una operación de inserción + borrado. (\*\*) La operación de búsqueda en listas de publicación puede ser óptimamente O(Log n), sin embargo, si el árbol no está bien balanceado, la complejidad puede aumentar desde O(h), altura del árbol, hasta un máximo de n.

### Perspectiva Pragmática de Indexación para WoT

Hemos decidido basar la implementación de nuestro esquema de indexación en Python 3.7.4, dado que proporciona potentes herramientas de procesamiento mientras mantenemos un código simple y legible. En primer lugar, usamos el módulo *xml.etree.ElementTree*<sup>52</sup> que implementa una API simple y eficiente para analizar y crear datos XML con compatibilidad integrada con *XPath*. Luego, inicializamos estructuras combinadas que utilizan implementaciones de código abierto de listas enlazadas, árboles B+ y árboles negros y rojos disponibles bajo licencias MIT en Python Software Foundation<sup>53</sup>. Finalmente, para cada documento de la colección, el indexador construye y actualiza el índice invertido dependiente del esquema. Usamos el algoritmo de índice invertido de referencia en [39] con algunas adaptaciones:

---

#### Algorithm 1 Contrucción del índice en IR.WoT

---

**Require:**  $D \leftarrow XML\ Document$

```

1: Begin
2: Initalize complex structures  $I(s) = 0$ , for all  $s \in S$ 
3: repeat
4:   for each XML element  $xe \in D$  do
5:     if  $xe$  exists in  $I$  then
6:        $Posting\ lists \leftarrow xe_{id}$ 
7:     else
8:        $I[search\_key] \leftarrow xe$ 
9:     end if
10:  end for
11: until Collection is done (For each D in Collection)

```

**Ensure:** Stats Updation

---

<sup>52</sup>XML eTree module - <https://docs.python.org/3.7/library/xml.etree.elementtree.html>

<sup>53</sup>Python Software Foundation - <https://pypi.org/>

El documento XML *D* de entrada se representa como un árbol ordenado con raíz. Cada nodo en el árbol XML contiene principalmente elementos, pero también puede contener atributos en los elementos, y los bordes entre los nodos representan relaciones estructurales como padre-hijo o antepasado-descendiente.

```

1 <?xml version="1.0" ?>
2 <smart_space>
3   <!--Generated for SIM.WoT Project-->
4   <actions>
5     <admin>Municipality of Granada</admin>
6   </actions>
7   <base>/api/v2/virtualspaces/</base>
8   <name>SSP0000001</name>
9   <properties>
10    <description>Granada is the capital of the province of Granada, in
11      the autonomous community of Andalusia.</description>
12    <geo_class>City</geo_class>
13    <keyword>Europe, Spain, Andalusia</keyword>
14    <latitude>37.1881</latitude>
15    <location_name>Granada</location_name>
16    <longitude>-3.6066</longitude>
17    <membership>IZN0000001</membership>
18  </properties>
19 </smart_space>

```

Figura 4.9 Documento XML simplificado de Prueba - Espacio Inteligente 1.

```

1 <?xml version="1.0" ?>
2 <virtual_thing>
3   <!--Generated for SIM.WoT Project-->
4   <base>/api/v2/virtualthings/</base>
5   <name>VTH0000001</name>
6   <name_friendly>Ambulance 01</name_friendly>
7   <properties>
8     <description>This is a public ambulance</description>
9     <latitude>37.2036</latitude>
10    <longitude>-3.5750</longitude>
11    <status>Free</status>
12  </properties>
13  <actions>
14    <admin>Granada Hospital</admin>
15  </actions>
16  <events>
17    <creation>2023-01-23 16:32:34.026544+00:00</creation>
18    <last_modified>2023-01-23 16:32:34.026544+00:00</last_modified>
19  </events>
20 </virtual_thing>

```

Figura 4.10 Documento XML simplificado de Prueba - Cosa Virtual 1.

Para fines de ampliar la descripción del comportamiento interno del indexador, a continuación se detalla con un ejemplo práctico las estructuras de datos creadas a partir de una colección de prueba compuesta por documentos XML simplificados del modelo WoT, estos documentos XML sintéticos de prueba se ilustran en las Figuras 4.9, 4.10 y 4.11.

En la Figura 4.12 se muestra el resultado final de la inversión de índice para cada uno de los documentos. Esto involucra los siguientes pasos:

```

1 <?xml version="1.0" ?>
2 <virtual_sensor>
3   <!--Generated for SIM.WoT Project-->
4   <base>/api/v2/virtualsensors/</base>
5   <name>VSN00000001</name>
6   <name_friendly>Sensor 01</name_friendly>
7   <properties>
8     <description>02 Level Sensor</description>
9     <latitude>37.2036</latitude>
10    <longitude>-3.5750</longitude>
11    <status>Operative</status>
12    <value>75%</value>
13    <membership>VTH00000001</membership>
14  </properties>
15  <actions>
16    <admin>Granada Hospital</admin>
17  </actions>
18  <events>
19    <creation>2023-01-23 16:32:45.778535+00:00</creation>
20    <last_modified>2023-01-23 16:32:45.778535+00:00</last_modified>
21    <last_sample>2023-01-23 16:37:45.778535+00:00</last_sample>
22  </events>

```

Figura 4.11 Documento XML simplificado de Prueba - Sensor Virtual 1.

- Inicializar las estructuras de datos a emplear, esto es, por una parte el vocabulario (*hash*) que contiene las palabras (términos *tokens*).
- Al recorrer cada uno de los documento XML en la colección se crea para cada palabra una entrada en el vocabulario y se comienza el conteo del número de documentos en el cual se encuentra, que determinará también el número de listas de publicación asociadas a cada palabra.
- Cada lista de publicación contiene:
  - El identificador del documento en donde se encuentra la palabra.
  - La frecuencia del término *tf* en el documento.
  - El elemento XML contenedor de la palabra, representado por su ruta XML. Ejemplo: para la etiqueta XML *<description>* se tendrá: *intelligent<sub>2</sub>one/properties/description* en este tipo de entidades.
  - La frecuencia del término dentro del elemento XML *fte*. Estas se emplearán posteriormente en las estimaciones del valor de relevancia.

Cada lista de publicación se adiciona a la estructura complemento en cada una de las opciones de índice discutidas. La Figura 4.13 ilustra de manera gráfica y simplificada cada una de estas estructuras, en especial el de las listas de publicación.

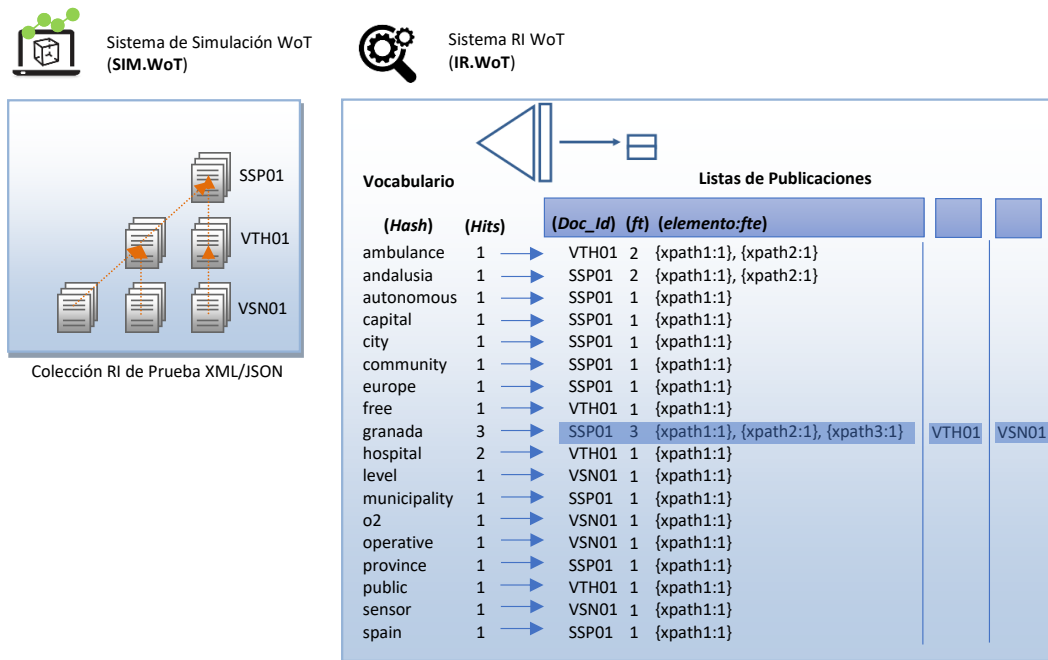


Figura 4.12 Ejemplo de Construcción del Índice: Inversión de Estructura.

## 4.5. Consideraciones de Diseño en el Módulo de Puntuación y Recuperación

### 4.5.1. Sub-Componente de Recuperación para WoT

El modelo de recuperación XML emplea una extensión del bien conocido algoritmo Okapi BM25. En el transcurso de la iniciativa de evaluación de recuperación XML (INEX, por sus siglas en inglés) y en otros foros de investigación se ha demostrado y evaluado un conjunto de aplicaciones del algoritmo base BM25 con sus modificaciones para recuperación focalizada, de elementos o *snippets* [36].

Si bien, recientemente se han propuesto modelos alternativos como el acercamiento de Belahyane et al. [4] empleando teoría de grafos o en Bessai et al. [6] usando algoritmos genéticos, el rendimiento de BM25 en los *tracks* INEX no ha sido superado considerablemente. Y existe un consenso generalizado al uso de variaciones de BM25 para ponderar características de la estructura XML dentro de las estadísticas de elementos y documento.

Comúnmente las tareas de búsqueda se centran en retornar una lista ordenada por relevancia de elementos XML de documentos en respuesta a una consulta de usuario. Se requiere adicionalmente que los elementos XML no presenten solapamiento, esto es, que no exista

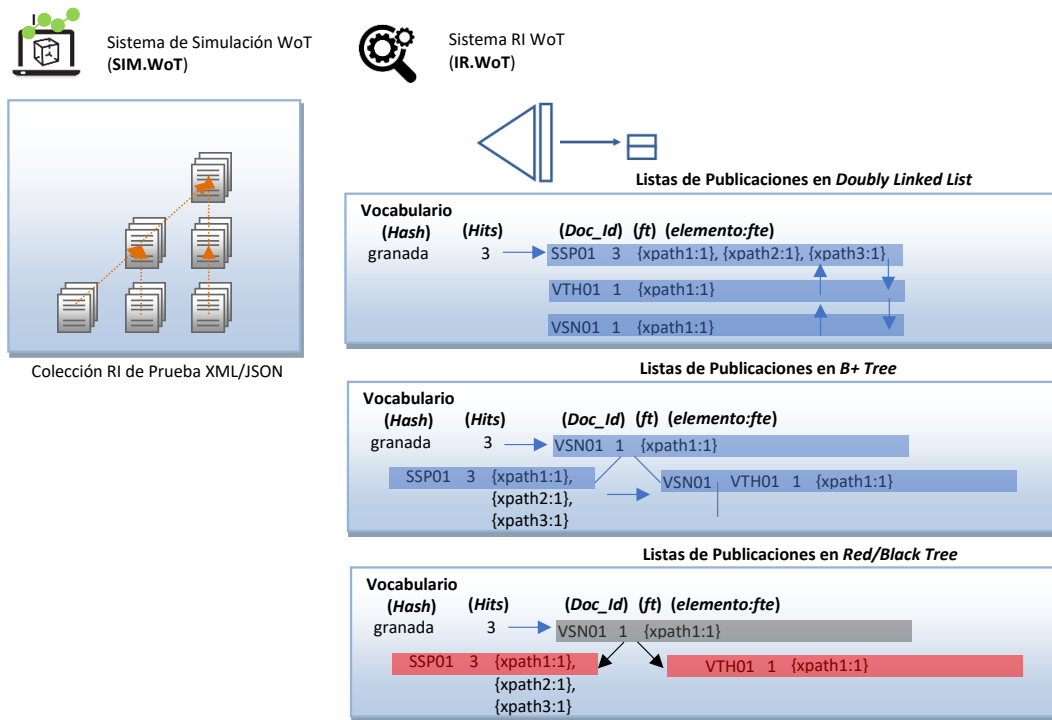


Figura 4.13 Ejemplo de Construcción del Índice: Listas de Publicación.

ninguna relación jerárquica entre el elementos, uno contenido en el otro [47]. La recuperación XML de elementos usa la siguiente versión de:

$$\text{score}(E, Q) = \sum_{i=1}^n \text{IDF}(q_i) \cdot \frac{f(q_i, E) \cdot (k_1 + 1)}{f(q_i, E) + k_1 \cdot (1 - b + b \cdot \frac{|e_E|}{\text{avgdl}})} \quad (4.11)$$

Donde  $Q$  es un conjunto de términos de consulta, IDF es el peso de termino  $i$ -ésimo,  $f(q_i, D)$  es la frecuencia de términos en un elemento XML,  $e_E$  es la longitud de un elemento XML de  $E$ , y  $\text{avgdl}$  es la longitud promedio de un documento en la colección. Al aplicar BM25 directamente se siguen empleando las estadísticas a nivel de documento para el cálculo IDF. BM25F es una extensión que explota información estructural del documento XML. El puntaje de relevancia para un elemento es calculado como:

$$\text{score}(E, Q) = \sum_{t \in q \cap e} \text{IDF}(q_i) \cdot \frac{\bar{x}_{e,t}}{K + \bar{x}_{e,t}} \quad (4.12)$$

Donde  $\bar{x}_{e,t}$  es la frecuencia de termino normalizada y ponderada y  $K$  es un parámetro ajustable. Para obtener  $\bar{x}_{e,t}$  la normalización de longitud se realiza primero por separado para cada

campo  $f$  asociado con un elemento  $e$ , produciendo frecuencias de términos normalizados específicos.

## 4.6. Consideraciones de Implementación y Despliegue en Nube (IR.WoT)

El código fuente de IR.WoT se encuentra disponible en repositorio público de GitHub publicado como código abierto bajo licencia GPL 3.0 en <https://github.com/cristyanmanta/ir-wot-ugr>. Este contiene el proyecto Python elaborado en el ambiente de desarrollo *PyCharm Professional Edition*<sup>54</sup> empleado bajo un paquete de licencia estudiantil. La propuesta de IR.WoT contempla las etapas críticas de indexación, interpretación de consulta, recuperación, puntuación y presentación. Este repositorio contiene algunas consideraciones de diseño, implementación en la nube y experimentación basadas en una colección de documentos XML sintéticos de simulación ejecutada en SIM.WoT.

Para la instalación en Google Cloud App Engine, se usan las pautas del ambiente estándar para Python, el código puede ser desplegado automáticamente mediante el SDK de Administración con el comando *gcloud app deploy*. Para el uso local se ejecuta el fichero Python principal por consola *python main.py*.

### 4.6.1. Sub-Componente Especial para Integración IR.WoT y SIM.WoT

El simulador por eventos discretos SIM.WoT proporciona una RESTful API y con ella, una serie de llamadas HTTP(S) para interactuar con el simulador y en especial con la colección RI de prueba, de la siguiente manera:

- Iniciar una simulación de WoT
- Detener una simulación de WoT
- Modificar los criterios de parada de simulación de WoT
- Rastrear (“*crawl*”) las entidades simuladas de WoT: espacios inteligentes, cosas y sensores virtuales.
- Obtener las estadísticas de simulación de WoT.

---

<sup>54</sup><https://www.jetbrains.com/pycharm/>



Cuya definición puede ser consultada en documento *Swagger*, ver pie de página <sup>55</sup>. Se tienen tres tipos de llamadas para cada un de los niveles del modelo WoT. Uno se encuentra disponible para rastrear los documentos XML de entidades de tipo: zona, espacio y subespacio inteligente. Y las otras dos llamadas para rastrear cosas y sensores virtuales respectivamente. A continuación se presentan tres ejemplos para cada tipo de llamada. Cada llamada emplea una operación de tipo HTTP GET con dos parámetros (clave:valor), el primero indica el tipo de formato de salida a XML (output:XML), y el segundo apunta al identificador de la entidad simulada y su representación XML a ser obtenida (id:IZN00000001). La instancia actual de SIM.WoT que sirve estas llamadas en la versión en nube se encuentra alojado en: <https://sim-wot-ugr.ue.r.appspot.com/>, la cual es la URL base para los siguientes puntos de llamada:

$$./api/v2/virtualspaces?output = xml&id = IZN00000001 \quad (4.13)$$

$$./api/v2/virtualthings?output = xml&id = VTH00000001 \quad (4.14)$$

$$./api/v2/virtualsensors?output = xml&id = VSN00000001 \quad (4.15)$$

De manera similar, la documentación perteneciente a la RESTful API del sistema IR.WoT contiene una colección de llamadas creadas para interactuar con el motor de búsqueda IR.WoT. Este proporciona llamadas API para:

- Búsqueda en la Web de las Cosas (WoT).
- Ejecutar consultas CO y CAS en WoT.
- Ejecutar marco experimental con IR.WoT.
- Recopilar estadísticas experimentales de IR.WoT.

Cuya definición puede ser consultada en documento *Swagger*, ver pie de página <sup>56</sup>. La Figura 4.14 ilustra la conexión entre el simulador de eventos discretos SIM.WoT y el indexador en el sistema de recuperación IR.WoT. Para esto se emplea un primer acercamiento clásico de rastreo, en donde a partir de un conjunto de URI/URL semilla se construye una pila de documentos XML a indexar. A partir de esta pila, el indexador comienza la creación de índice a partir de tiempo cero de simulación. Adicionalmente, para cada evento de simulación se notifica una modificación de los documentos XML, lo que concluye en un re-indexado programado del documento XML modificado.

<sup>55</sup>[https://app.swaggerhub.com/apis/cristyanmanta/sim\\_wot/v1\\_0](https://app.swaggerhub.com/apis/cristyanmanta/sim_wot/v1_0)

<sup>56</sup>[https://app.swaggerhub.com/apis/cristyanmanta/ir\\_wot/v1\\_0/](https://app.swaggerhub.com/apis/cristyanmanta/ir_wot/v1_0/)

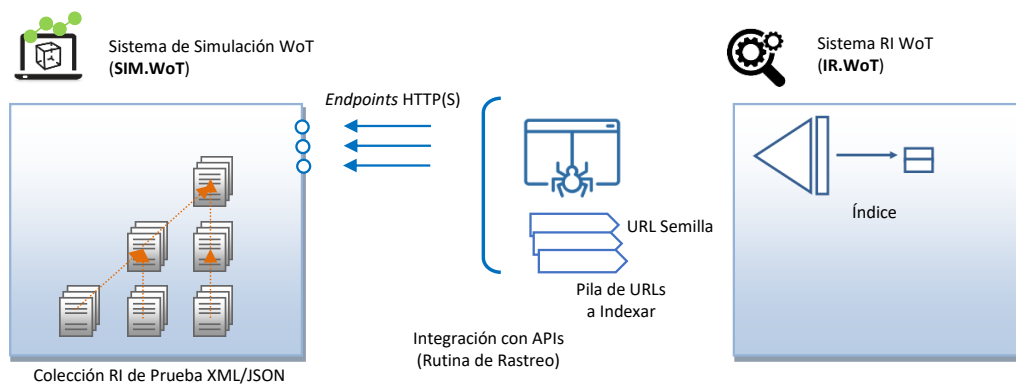


Figura 4.14 Módulo de Integración IR.WoT y SIM.WoT



# Capítulo 5

## Experimentación, Evaluación de la Indexación y Búsqueda

En este capítulo se describe el conjunto de experimentos diseñados para la evaluación de sistemas RI para WoT. Los experimentos se llevan a cabo empleando los servicios en nube de *Google Cloud Platform* para aislar el motor de búsqueda IR.WoT del motor de simulación SIM.WoT sobre proyectos diferentes. El objetivo de este aislamiento es evitar errores en la medición de las métricas de desempeño. Hemos decidido conectar nuestra arquitectura IR.WoT propuesta a través de una API RESTful, emulando así un escenario WoT realista donde el motor de búsqueda rastrea por HTTP(S) las cosas.

### 5.1. Diseño de Experimentos y Procedimientos

Técnicamente hablando, los esquemas de indexación se pueden evaluar en términos de desempeño considerando el costo de procesamiento en tiempo, espacio de memoria o disco y la complejidad del procesamiento. Proponemos realizar este conjunto de experimentos para evaluar el esquema de indexación. El módulo de indexación desempeña un papel clave en el rendimiento de IR, pero también tiene un impacto inherente en los recursos de tiempo y espacio. Por tanto, acelerar el mecanismo RI puede tener un compromiso implícito con el espacio de almacenamiento requerido en el momento de la creación. Por otra parte, las operaciones sobre el índice determinarán el desempeño general del sistema impactando el tiempo de respuesta a consulta. Si bien las medidas de desempeño determinarán e impactarán positivamente la experiencia de usuario, es importante evaluar la eficiencia del módulo de recuperación y el sistema en general. Como se menciona en la Capítulo 2, no existe un marco de evaluación para IR-WoT por lo que en esta serie de experimentos incluimos un modelo de

evaluación desde la perspectiva del usuario final, a fin de determinar medidas de eficiencia a la recuperación y de las características funcionales y no funcionales del sistema.

### 5.1.1. Conjunto de Datos Experimentales: Escenario Simulado WoT

De modo de proveer contexto en el conjunto de datos experimentales empleados, que tienen como base un escenario simulado WoT sobre el cual se se realizarán las consultas de información.

- Se han definido en la simulación tres (3) zonas inteligentes: España (ES), Colombia (CO) y Reino unido (UK) con base en datos geográficos *Geonames*<sup>57</sup>.
- Además, pertenecientes a estas zonas se han definido (500) espacios inteligentes con base en las ciudades principales de las zonas, incluyendo Granada, Barcelona, Bogotá y Londres.
- Usando la información de *GeoNames* se han definido como subespacios inteligentes 16.500 lugares alrededor de las ciudades.
- En las ciudades se han definido 16.500 cosas virtuales representando aparcamientos públicos de la ciudad, en los que se expone el estado general de ocupación como una propiedad. Generados de forma sintética por medio del simulador SIM.WoT.
- En las ciudades se han definido 16.500 cosas virtuales representando ambulancias públicas/privadas de la ciudad, en los que se expone el estado general de ocupación, así como su equipamiento como una propiedad, haciendo uso de algún tipo de sensor virtual. Generados de forma sintética por medio del simulador SIM.WoT.
- En las ciudades se han definido 16.500 sensores virtuales de tres (3) tipos representando variables ambientales de interés: concentración de CO<sub>2</sub>, temperatura y calidad del aire. Generados de forma sintética por medio del simulador SIM.WoT. Estos actualizan sus mediciones estocásticamente con tiempos entre medidas de 3 a 10 minutos.

### 5.1.2. Conjunto de Experimentos #1: Operaciones en Estructuras de Datos del Índice

Este conjunto tiene como objetivo la validación experimental del desempeño de las estructuras de datos propuestas con base en la complejidad de tiempo esperada por cada tipo

<sup>57</sup>BD Geográfica GeoNames - <https://www.geonames.org/>

Tipo de Entidad	Entidades en Colección	Tamaño por Documento (KB)	Tamaño por Tipo (KB)
IZN	4	2,02	8,08
SSP	500	2,02	1.010,00
SSS	16.500	1,27	20.955,00
VTH	16.500	1,13	18.645,00
VSN	16.500	1,13	18.645,00
<b>Total</b>	<b>50.004</b>	<b>1,19</b>	<b>59.263,08</b>

Tabla 5.1 Colección XML RI Experimental

de operación sobre las estructuras del índice. Se realiza la medición del tiempo promedio de las operaciones de inserción, modificación, borrado y consulta.

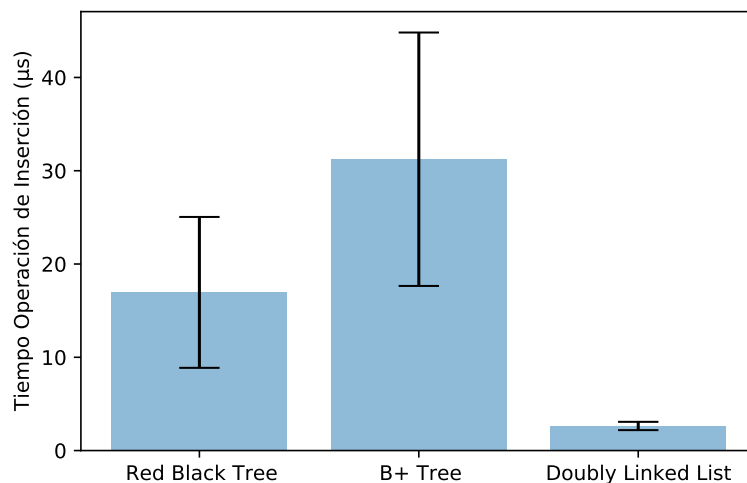


Figura 5.1 Resultados de Tiempo de Operación de Inserción.

- Tiempo de inserción en las estructuras de datos. Donde la función *perf\_counter()* devuelve un valor flotante del tiempo en segundos. Devuelve el valor (en fracciones de segundo) de un contador de rendimiento, es decir, un reloj con la resolución más alta disponible para medir una duración corta<sup>58</sup>. En este experimento se promedia el tiempo de las operaciones de inserción sobre una misma estructura y se estima su desviación estándar. La Figura 5.1 ilustra en diagrama de barras con error estos resultados. Si bien, algorítmicamente, la complejidad de tiempo de las estructuras es  $O(1)$  es conocido

<sup>58</sup>Librería *Time* de Python - <https://docs.python.org/es/3/library/time.html>

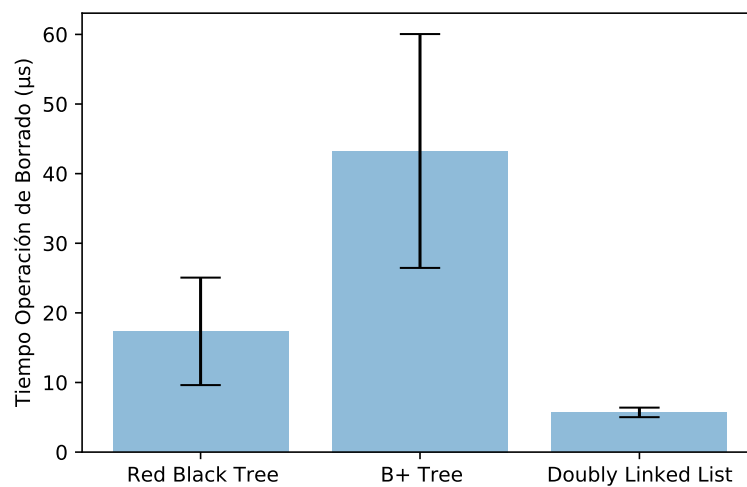


Figura 5.2 Resultados de Tiempo de Operación de Borrado.

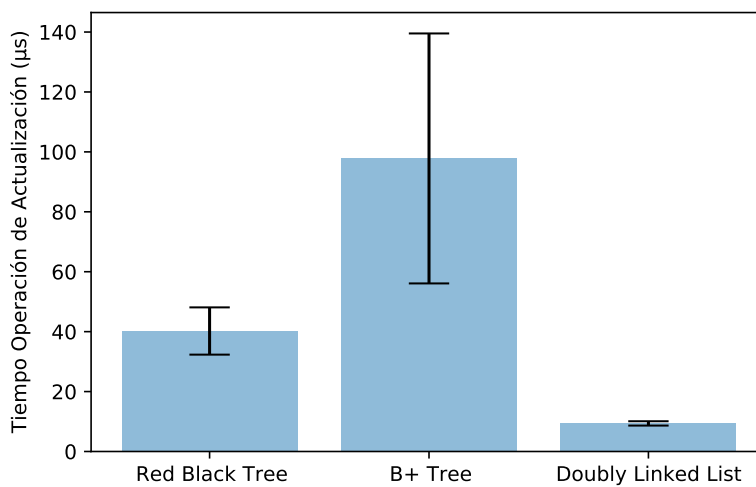


Figura 5.3 Resultados de Tiempo de Operación de Actualización.

que en la práctica esta puede llegar a  $O(\log(n))$  en el peor escenario para los árboles B+ y Rojo-Negro.

- Tiempo de borrado en las estructuras de datos. La Figura 5.2 ilustra tanto el tiempo promedio de las operaciones de borrado para cada estructura del índice, así como su desviación estándar. Se tiene un comportamiento similar para la operación de borrado en comparación con la operación de inserción. Sin embargo, es notorio una mayor

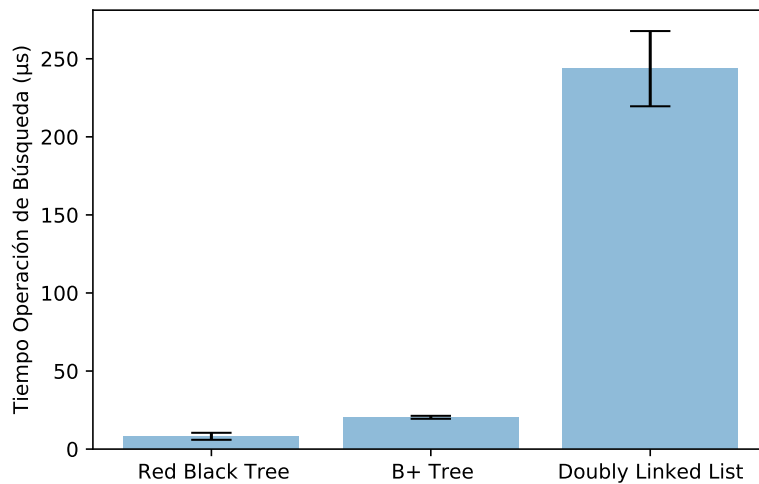


Figura 5.4 Resultados de Tiempo de Operación de Búsqueda.

desviación en el tiempo por operación especialmente para el árbol B+. Es conocido que el borrado de un elemento en un árbol B+ consta de tres eventos principalmente: buscar el nodo donde existe la llave, eliminar la llave y equilibrar el árbol si es necesario. Por lo que el peor comportamiento práctico de esta estructura puede deberse a estos eventos. Además la complejidad en tiempo en árboles puede llegar a  $O(\text{Log}(n))$ .

- Tiempo de operación de actualización en las estructuras de datos. La Figura 5.3 ilustra tanto el tiempo promedio de las operaciones de actualización para cada estructura del índice, así como su desviación estándar. Es importante recalcar que el comportamiento de estas operaciones se ve impactada por las operaciones elementales de borrado e inserción, puesto que es la forma como se están implementando en los esquemas de indexado.
- Tiempo de operación de búsqueda en la estructura de datos. En la Figura 5.4 se presenta el tiempo promedio de búsqueda sobre los diferentes estructuras propuestas y su desviación. Es considerablemente notorio, el cambio de comportamiento del esquema de indexado con base en listas doblemente enlazadas para este tipo de operaciones. Que refleja la complejidad en tiempo  $O(n)$  teórica, puesto que es necesario recorrer la estructura para una búsqueda. En contraposición, las estructuras de árbol presentan un comportamiento consistente en tiempo que en el peor de los casos seguirá una complejidad  $O(\text{Log}(n))$ .



La siguiente sub-sección presenta un conjunto de experimentos desde la perspectiva de la creación del índice. En términos de la complejidad en espacio, todas las estructuras presentan teóricamente un peor escenario de  $O(n)$ . Por lo que se incluye, un experimento para la medición del tamaño en memoria de los diferentes tipos de índice.

### 5.1.3. Conjunto de Experimentos #2: Creación del Índice

Este conjunto tiene como objetivo la validación experimental del desempeño de las estructuras de datos propuestas para la construcción de un índice desde cero.

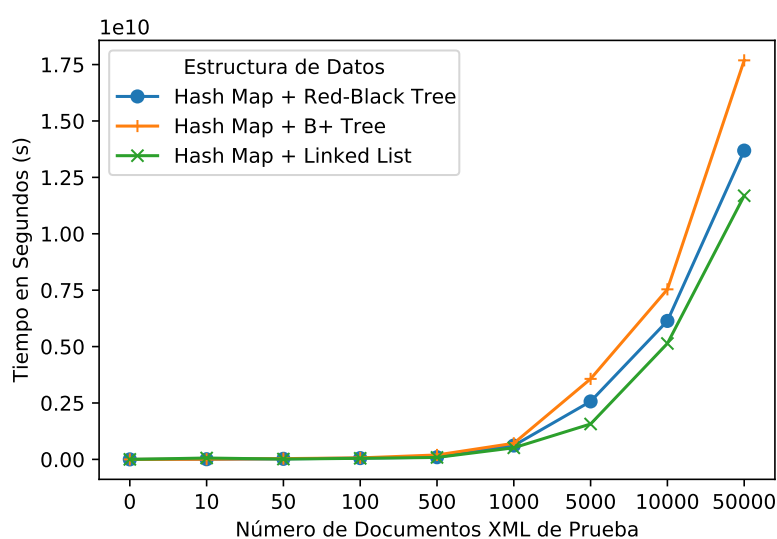


Figura 5.5 Resultados de Creación del Índice en Tiempo.

Se realiza la medición de:

- Tiempo de creación del índice. En este se emplea la librería estándar *time* de Python y la función *perf\_counter()*. La Figura 5.5 presenta el tiempo de creación de índice en función del número de documentos XML indexados para las distintas opciones de estructura de datos en memoria. (El tamaño promedio de documento es de 1.5KB). Es importante aclarar que este tiempo de creación incluye tiempo de procesamiento de los documentos XML, esto es, la recolección de la colección de prueba documento por documento vía HTTP, el análisis de la estructura, la inversión del índice y el cálculo de las estadísticas de documento, elemento y colección.
- Tamaño del índice en memoria. En este se emplea la librería pública *memObj* de Python<sup>59</sup> que proporciona un método para contabilizar el tamaño total en memoria de

<sup>59</sup>Librería *memObj* de Python - <https://pypi.org/project/memobj/>

Tipo de Índice	Tamaño de la Colección (KB)	Tamaño del Índice (KB)	Radio Índice a Colección
Hash Map + Red-Black Tree	56.494,83	6.514,10	0,115
Hash Map + B+ Tree	56.494,83	6.644,38	0,118
Hash Map + Doubly Linked List	56.494,83	6.123,25	0,108

Tabla 5.2 Radio entre el Tamaño del Índice y el Tamaño Original de la Colección.

un objeto. La Figura 5.6 presenta la evolución en tiempo del tamaño del índice (como instancia de un objeto) en memoria. Este presenta un comportamiento considerablemente congruente entre las tres alternativas evaluadas. En términos generales, el índice puede requerir un espacio en memoria y en un sistema de almacenamiento permanente con un espacio extra que la de la colección de prueba original.

- Radio del tamaño del índice a tamaño de la colección. Dado el tamaño de la colección en (KB), ver Tabla 2.7 y estimado el tamaño final del índice para cada estructura se calcula la relación entre estos. Tradicionalmente, el radio tamaño del índice a colección se encuentra en valores entre el 0.02 al 0.03 para algunas de las colecciones estándar del TREC [136]. En la Tabla 5.2 se presentan los valores obtenidos.
- Velocidad de Indexación. Por último, este conjunto de experimentos contabiliza el número de documentos XML indexados por unidad de tiempo (segundos).

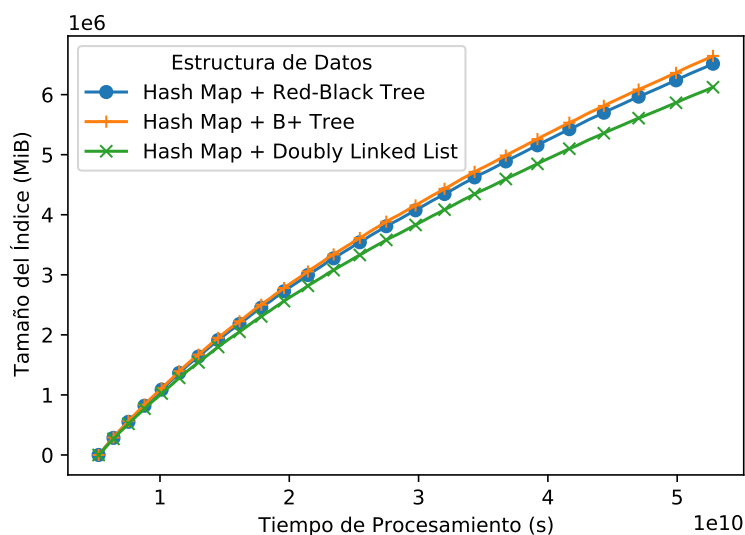


Figura 5.6 Resultados de Creación del Índice en Tamaño.

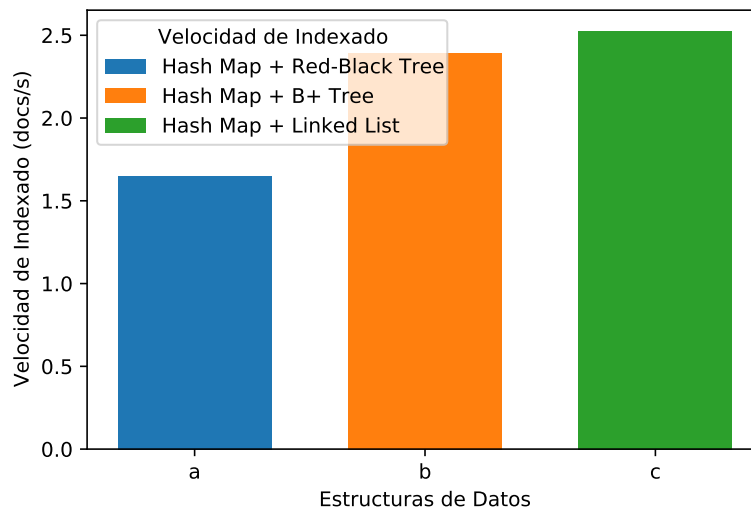


Figura 5.7 Resultados de Velocidad de Indexado.

#### 5.1.4. Conjunto de Experimentos #3: Tiempo de Respuesta en Consulta

Este conjunto de experimentos tiene como objetivo la validación experimental del rendimiento integral del sistema IR.WoT de extremo a extremo, esto es, la estimación del tiempo promedio de repuesta dada a una serie de consultas sintéticas empleando un número variable de palabras, ver Figura 5.8. Se realiza la medición de:

- Tiempo de Respuesta a Consulta. Se emplea para este experimento la librería estándar *time* capturando el tiempo promedio de procesamiento de consulta desde la emisión de esta por parte del usuario y la entrega de resultado por parte del sistema.

#### 5.1.5. Conjunto de Experimentos #4: Estudio de Usuarios Finales

Este conjunto de experimentos tiene como base un modelo sistemático de evaluación de motores de búsqueda Web orientado al usuario, que hemos adaptado a partir de [123]. Este divide los criterios de evaluación en:

- Criterios de Desempeño
  - Medidas de Eficiencia
  - Medidas de Utilidad
  - Medidas de Satisfacción de Usuario

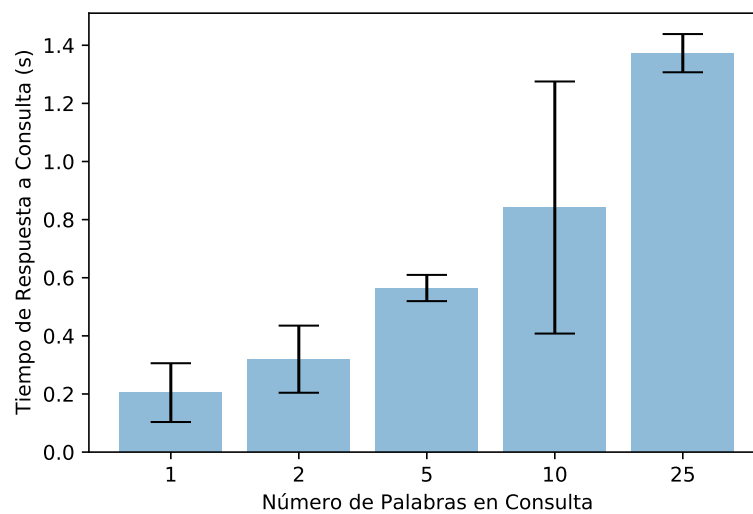


Figura 5.8 Resultados de Tiempo Promedio de Consulta.

- Medidas Funcionales
- Características No relacionadas con Desempeño
  - Antecedentes de los Participantes
  - Experiencia de los Participantes
  - Necesidades de Información del Usuario y Requerimientos de Búsqueda
  - Comportamiento de la Búsqueda

En este modelo, los usuarios finales deben realizar la evaluación de relevancia de los resultados de recuperación en función de la relevancia definida por el mismo usuario.

### Procedimientos de Prueba y Control

Para propósitos de aplicar el modelo de evaluación del motor de búsqueda IR.WoT por parte de un grupo de usuarios finales, se proporciona acceso directo a los usuarios a través de la instancia pública en: <https://ir-wot-ugr.ue.r.appspot.com/>. El conjunto de pruebas se conduce por medio de formularios y una guía de procedimiento descrita a continuación en esta sub-sección. Tanto la secuencia como el control de tiempo para las tareas y las mediciones son importantes con la finalidad de asegurar la consistencia de la evaluación. Se han seleccionado tres sitios de prueba (Universidad Distrital Francisco José de Caldas, Politécnico Grancolombiano) con la participación de cinco (5) participantes por cada sitio.

### **Sesión de Pruebas con Usuarios I**

1. Completar un formulario de consentimiento de participación y un cuestionario de antecedentes y experiencia del usuario. Se emplea como el modelo base de hoja de información y consentimiento informado para el participante en el estudio, ver Anexo 3.
2. Proporcionar información sobre la necesidad de información del usuario y los requisitos de búsqueda en un formulario de búsqueda.
3. Recibir sesión tutorial en el uso de IR.WoT.
4. Realizar búsquedas sobre el problema de información del propio usuario en el orden proporcionado.
5. Utilizar las instrucciones en línea haciendo clic en los menús correspondientes.
6. Obtener una copia digital de los resultados de búsqueda de un tamaño predeterminado;
7. Completar un cuestionario en línea sobre la satisfacción del usuario con respecto a las características del motor de búsqueda IR.WoT y la interacción. Ver cuestionario en línea a emplear <https://forms.office.com/r/4XdYWuCD6g> (Secciones 1, 2 y 3).
8. Repetir los pasos 4) a 7) hasta que se prueben todas las búsquedas.

### **Sesión de Pruebas con Usuarios II**

1. Evaluar la relevancia de los resultados de la búsqueda de acuerdo con un conjunto de pautas de juicio entregadas.
2. Seleccionar y clasificar los elementos de los resultados de búsqueda según las instrucciones proporcionadas.
3. Participar en una entrevista posterior a la búsqueda para proporcionar reacciones al proceso de búsqueda, el sistema RI y el rendimiento general del motor. Ver cuestionario en línea a emplear (Sección 4) <https://forms.office.com/r/4XdYWuCD6g>.

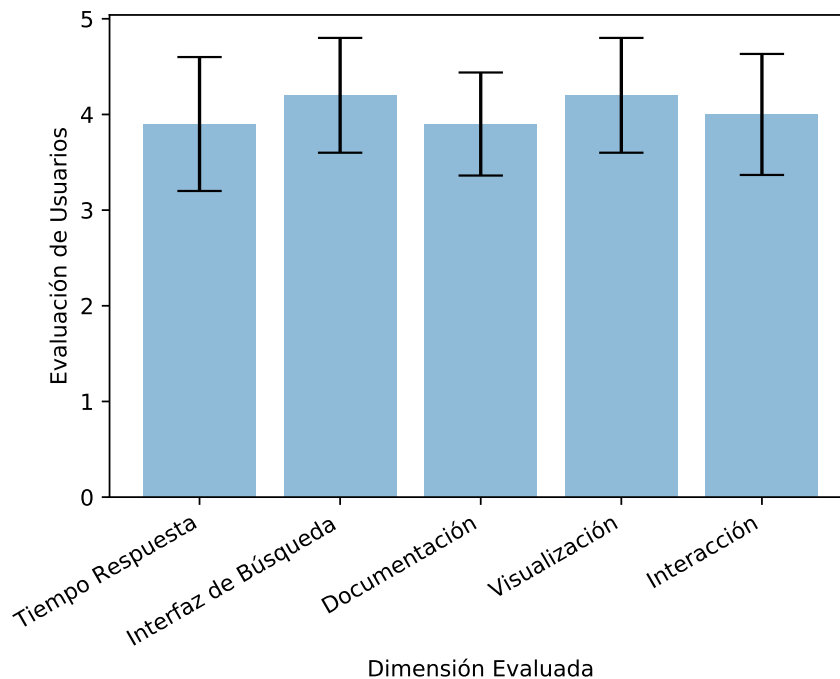


Figura 5.9 Evaluación de Usuarios Finales a Dimensiones.

## 5.2. Reporte de Experimentación y Análisis de Resultados

En términos fundamentales, el tiempo de creación del índice es directamente proporcional al tamaño de la colección de prueba, en específico al tamaño promedio de los documentos XML a indexar, y por ende del tamaño en Bytes o número de tokens/palabras que componen los elementos XML a procesar. Si bien, las estructuras de datos básicas tienen una complejidad algorítmica constante  $O(1)$  para las operaciones de inserción, las implementaciones propias del lenguaje Python/Java/C juegan un factor preponderante en el desempeño práctico de las mismas. Dado que las implementaciones evaluadas corresponden o con implementaciones siguiendo el paradigma de programación orientada a objetos (POO, por sus siglas en inglés) del árbol B+ y/o al uso de librerías públicas de *Pypi* para el árbol rojo-negro en <sup>60</sup>, el desempeño en tiempo y memoria de los mismos no es el ideal. En la práctica, la implementación simple de un mapa *Hash* y una doble lista enlazada tiene el mejor desempeño en tiempo. Esto debido a que la implementación de las estructuras de datos en base a estructuras bien conocidas del lenguaje optimizan ya su ejecución en el ambiente de ejecución (*runtime*).

Técnicamente hablando, se pueden establecer los siguientes criterios para el uso de estructuras combinadas para constituir el esquema del índice en diferentes escenarios de WoT:

<sup>60</sup>Librería *Árbol Rojo-Negro* de Python - <https://pypi.org/project/red-black-tree-mod/>

- WoT con Alta Velocidad de Cambios en Colección: La estructura compuesta de *Hash + Doubly Linked List* se constituye en la mejor opción para la indexación y re-indexación (actualización). La implementación de esta estructura permite operaciones de inserción por debajo de los 5 $\mu$ s (2.6 $\mu$ s promedio) y operaciones de actualización por debajo de los 10 $\mu$ s (9.3 $\mu$ s promedio) en los experimentos realizados, esto es, aproximadamente 6x más rápido que la estructura con base en *Hash + árbol rojo-negro* y 12x más rápido que la estructura con base en *Hash + árbol B+* en inserción. Es importante considerar los tiempos de procesamiento necesarios por los pasos intermedios en las estructuras de árbol para balancear o marcar las ramas/hojas, que en últimas aceleran en la práctica las operaciones de búsqueda.
- WoT con Moderada Velocidad de Cambios en Colección y Alto Número de Entidades (o opcionalmente Alta Profundidad de Documento): La estructura compuesta de *Hash + árbol rojo-negro* es la mejor opción de indexación de cara a la recuperación de información. Si bien se puede impactar la frescura de los resultados obtenidos debido al costo en tiempo de inserción, borrado y actualización, al exhibir un tiempo de búsqueda menor entre las estructuras comparadas, se constituye en la mejor opción al requerir operaciones de búsqueda sobre un número elevado de documentos en la colección. Dado que el número de listas de publicación es directamente proporcional al número de entidades y por ende al número de documentos XML descriptores el tiempo de recuperación se verá aumentado por las operaciones de búsqueda en el índice. En la estructura *Hash + árbol rojo-negro* se logran complejidades  $O(\log(n))$ , en los experimentos por debajo de 10 $\mu$ s (8.17 $\mu$ s promedio), lo que es 2.5x más rápido que el *Hash + árbol B+* y 30x más rápido que el *Hash + Doubly Linked List*.

El tamaño del índice depende directamente del tamaño del diccionario en número de términos. Por lo que una colección RI de prueba con poca variabilidad en términos contendrá un diccionario de tamaño moderado. Y por otra parte, del tamaño de las listas de publicación almacenadas por sus correspondientes estructuras en memoria, como instancias de objetos. El tamaño de las listas de publicación a su vez es factor de la estructura de los documentos XML, y por tanto de la profundidad de la jerarquía o relaciones de padre-hijo de los elementos XML. De nuevo, la implementación propia de las estructuras de datos juegan un factor decisivo en el rendimiento del esquema de indexación en tiempo de creación, puesto que cualquier ineficiencia en código afectará el rendimiento en tiempo de procesamiento del sistema. Existen mejores prácticas en el almacenamiento de las listas de publicación de cara a las operaciones de lectura/búsqueda que serán más costosas computacionalmente para los modelos de recuperación que accederán al índice para el cálculo de relevancia dada una consulta.

En algunos modelos de recuperación, es conveniente el almacenamiento en orden de identificación de documento. En otros, es importante almacenar las estadísticas de la colección, del documento, del elemento y del término ( $tf * idf$ ) de modo que estén disponibles en tiempo de ejecución del algoritmo del cálculo del puntaje de relevancia o de la similitud. En las construcciones tradicionales de las estructuras de datos, se toman decisiones en términos de los recursos de almacenamiento empleando la memoria RAM y/o el disco duro y/o almacenamiento en nube. Es posible que las limitaciones en memoria obliguen al uso de disco duro lo que impactará el rendimiento del sistema. En nuestra propuesta, los experimentos se realizan construyendo el índice completo en memoria, esto debido y permitido por el reducido tamaño de los documentos XML de los modelos WoT, sin embargo para afrontar la escalabilidad del sistema en un trabajo futuro será necesario abordar el uso de almacenamiento en nube (en sus diferentes formas: NoSQL, No relacional, de objetos, de documentos, de bloques) para la construcción de índices distribuidos. Es también posible el uso de tecnologías de memoria caché compartida en nube (hasta un máximo de 1GiB) entre múltiples instancias de un mismo micro-servicio de búsqueda para WoT.

La escalabilidad del sistema IR.WoT propuesto considera en el diseño la utilización de múltiples instancias de indexación, lo que requiere la segmentación del índice invertido:

- **Particionado por Documento:** En este las listas de publicación son más cortas lo que permite una mejor gestión y una mayor escalabilidad del índice en sí. Las inserciones deben distribuirse entre diferentes instancias del módulo de indexación, con las limitaciones propias de la infraestructura tecnológica subyacente. Las operaciones de actualización son independientes entre instancias debido a la partición por documentos lo que se traduce en aislamiento técnico entre instancias. Los experimentos realizados con IR.WoT y las conclusiones sobre el uso de las estructuras de datos se mantienen. Existen desventajas como contrapeso a esta elección puesto que al tiempo de consulta todas las instancias de índice deberán ser contactadas. Por consiguiente técnicamente hablando, la complejidad algorítmica para el cálculo de la ponderación y posterior re-organización se incrementa.
- **Particionado por Término:** En contraste con la metodología de segmentación anteriormente descrita, al tiempo de consulta sólo un subconjunto de instancias del índice deben ser contactadas. Lo anterior se traduce en una mejora en los tiempos de respuesta totales del sistema RI. Sin embargo, las operaciones de actualización sobre la estructura del índice requieren por ende contactar todas las instancias. Sobre este escenario, las conclusiones referentes a la estructura *Hash + Doubly Linked List* pueden ser empleadas como entrada al diseño de esquemas de índice distribuidos óptimos en complejidad de tiempo.



Ampliando lo ya mencionado, la complejidad en espacio no juega un factor preponderante en la elección de una estructura de índice específica. Si bien para la colección experimental de prueba se tienen radio tamaño índice a colección relativamente mayores que en conjuntos de datos tradicionales, es cierto que la estructura *Hash + Doubly Linked List* tiene a un menor radio, posicionándose alrededor de 0.1. Esto último puede posicionar esta estructura como la mejor elección en ambientes WoT con alto número de entidades. Puede ser parte de un trabajo futuro la exploración experimental de mecanismos de compresión del índice y/o su adaptación a los entornos IoTWoT. La elección dependerá del costo computacional en tiempo, espacio y conectividad (como también al costo tecno-económico de cada uno de estos factores en suma) y de la posibilidad de pérdida o no de información en la compresión que puede tomar ventaja de la corta validez de los datos y su característica efímera, especialmente en la capa de sensórica WoT.

Tomando como línea base el tiempo promedio de consulta para un número variable de términos (usando la implementación *Hash + árbol rojo-negro*), es importante recalcar que el comportamiento no crece linealmente. Las necesidades de información tradicionalmente por parte de usuarios finales no requieren de un número elevado de términos en consulta, más si del cumplimiento de intenciones de búsqueda con relación al contexto. En una dirección de investigación tangencial, es importante como trabajo futuro explorar las alternativas de implementación práctica de los interpretes de consulta que optimicen el tiempo de respuesta pero más importante aumenten la satisfacción del usuario final frente a la necesidad de información y/o aumenten la relevancia en recuperación de información IoTWoT frente a la intención del usuario final.

# Capítulo 6

## Conclusiones y Trabajo Futuro

### 6.1. Conclusiones y Consideraciones Finales

La WoT impone una dinámica diferente que debe y está siendo considerada en el diseño y desarrollo de nuevos sistemas RI. La WoT trae consigo no sólo la herencia de la Web de Datos con la incorporación de datos públicos y su interrelación y vinculación entre sí. Sino que además permite obtener datos de interés en tiempo real, como por ejemplo el cambio de estado de las cosas en el mundo físico (inclusive su ubicación geográfica, variables físico, químicas o de cualquier índole de interés). Este dinamismo se ve reflejado en la constante inserción, actualización y borrado sobre los documentos de descripción de las cosas y por ende impactan cualquier mecanismo de recuperación de esta información.

Es visible que el grado de idoneidad y adaptabilidad de los mecanismos de RI convencionales no es el adecuado al considerar el dinamismo IoT|WoT. En gran parte como resultado de los supuestos con los que se desarrollaron estos a partir de una Web predominantemente estática, que fue evolucionando para incorporar modos de interacción con los datos y luego con el mundo real. Es importante notar, que el movimiento de paradigma con WoT además tiene una dimensión adicional, y es el hecho que WoT permite no solo obtener información (vía sensores) sobre el estado, propiedades y eventos de las cosas sino que también en su concepción adiciona la posibilidad de interactuar a través de acciones (actuadores) sobre las cosas y su entorno.

La mayoría de los modelos de la WoT ubican a los sensores en la capa más baja, y desde allí una serie de capas superpuestas o interpuestas, según la visión y propósito de cada investigación. La capa central de los modelos es una capa de abstracción de entidades en el mundo real, que también expone una considerable cantidad de alternativas para la descripción y representación. Los modelos y/o lenguajes de descripción de cosas van desde la utilización de tecnologías web complementarias como metadatos, microformatos, microdatos

u ontologías. Dada la flexibilidad, sencillez y uso de los estándares JSON/XML, estos están siendo seleccionados para la construcción de esquemas de representación para la WoT en conjunto con extensiones y otras variedades de formatos de datos como JSON-LD.

El modelo de WoT y los lenguajes de descripción se convierten en una piedra angular y un impulsor en el desarrollo e investigación de nuevos sistemas circundantes, incluidos RI. Ha de reconocerse las limitaciones y desafíos que enfrenta la evolución de los sistemas RI, en lo fundamental por la escasez de sistemas IoT|WoT abiertos, públicos e interoperables y por la misma escasez de datos o colecciones IoT|WoT de igual forma abiertos, públicos e interoperables. De allí, que la simulación por eventos discretos se constituya en un mecanismo indispensable para sentar las bases y construya entornos para el desarrollo actual y futuro de nuevos sistemas RI.

Nuestro modelo WoT abstracto y la propuesta de representación de cosas vía una colección dinámica de documentos XML descriptores IoT|WoT apuntan a crear conjuntos de datos de prueba abiertos e interoperables. En todas las etapas de investigación de este trabajo doctoral, los organismos de estandarización y la interoperabilidad han sido hilos conductores. La estandarización y la interoperabilidad son estandartes necesarios dentro de la investigación científica y requeridos para aunar esfuerzos en la evolución de los nuevos paradigmas y el ecosistema de aplicaciones y servicios a su alrededor.

Una línea activa de desarrollo de WoT y parte de la hoja de ruta de estandarización consideran datos enlazados para modelar objetos virtuales como pasarelas de entidades físicas y abstractas. La variación JSON-LD, que combina la simplicidad de JSON y el poder de los datos vinculados se establece como un posible escenario para ser empleado como formato de datos base de la WoT actual y del futuro. Otras alternativas consideran conveniente introducir el avatar WoT en una representación abstracta de alto nivel de las cosas, o basar el modelo WoT en el paradigma de agentes inteligentes con funciones definidas y modelos de interacción novedosos. WoT implica un fuerte cambio de paradigma en comparación con la web tradicional, se espera que Web y WoT evolucionen constantemente y que las cosas permanezcan durante un ciclo de vida más largo. El control de este ciclo de vida se puede abordar por medio de la estandarización, de esta manera tecnologías como HTTP, CoAP, XML, EXI que son tecnologías web bien conocidas, están y estarán en continua evolución por W3C para afrontar estos mecanismos.

Son numerosos y prometedores los esfuerzos de organizaciones como W3C, IEEE, ISO, IETF, ITU-T y OpenGIS entre otros para estandarizar tecnologías que apuntan a la interconexión del mundo real y la interacción con este a través avatares o gemelos digitales. El modelo WoT propuesto considera y se caracteriza por una visión del mundo real que

presta importancia al contexto espacio-temporal, agregando relaciones entre cosas, sensores y espacios. El contexto temporal se agrega a través de eventos.

Como parte de la investigación se ha construido un simulador flexible y versátil a gran escala capaz de imitar un mundo WoT inteligente completo, contiene entidades que encapsulan contextos espacio-temporales. Como contribuciones de este modelo se tiene el tratamiento de la ubicación y el contexto espacial como factores principales en la simulación de WoT. En cuanto al enriquecimiento semántico, se alinearon los elementos XML de representación siguiendo propuestas vigentes. En adición la arquitectura de simulación por eventos discretos hace uso de recursos en nube con lo cual podría ser adaptada para encapsular la dinámica esperada en el WoT en términos de escala.

En este trabajo se propone una arquitectura del tipo IRaaS como una forma de aprovechar los recursos en nube para la implementación de mecanismos RI adaptados a WoT, y al mismo tiempo plantear la discusión sobre la investigación en la distribución o no de las funciones RI, donde la capacidad de cómputo, almacenamiento, red y seguridad se consideren factores fundamentales. La arquitectura ha de estar en torno a los nuevos paradigmas que impulsan nuevas alternativas de construir motores de búsqueda sobre los micro-dispositivos atados a las cosas, o sobre el borde de la nube empleando la computación de niebla, *Fog Computing*.

La investigación científica en RI para IoT|WoT en la últimos años y década ha dejado como resultado una amplia base teórica, práctica y metodológica para la construcción de los motores de búsqueda del futuro. También ha demostrado una posibilidad real para adaptar los paradigmas, estrategias y técnicas de los modelos de RI convencionales para el desarrollo de sistema(s) de recuperación para la IoT|WoT. El sistema IR.WoT propuesto considera la adaptación de etapas de indexación, puntuación, reorganización para abordar la dinámica de la WoT desde ambas perspectivas teórica y práctica.

Han surgido diferentes modelos de indexación para IoT|WoT, la mayoría de los enfoques utilizan el modelo de índice invertido. Varios enfoques híbridos combinan opciones basadas en árboles de búsqueda con alternativas basadas en diccionarios, mejorando las características generales de un sistema de múltiples índices y el tiempo de búsqueda. Por otro lado, los índices de agrupamiento son una minoría de investigaciones y su potencial aún está en desarrollo, pero prometen acelerar la etapa de indexación. Estos últimos son un claro camino para la investigación futura en mecanismos RI adaptados.

Las taxonomías de sistemas RI para IoT|WoT en la forma de IoTSE|WoTSE han identificado la posibilidad de construir mecanismos de ponderación dependientes e independientes de la consulta. Los primeros re-organizan la lista de resultados en función estadísticas bien conocidas como  $(tf * idf)$ , o acondicionadas con base en algoritmos existentes como variantes BM25, o con nuevas métricas que consideran la calidad de la información proporcionada

por las cosas o los sensores. En el segundo tipo de mecanismos de ponderación, se han propuesto múltiples acercamientos para la optimización de la recuperación de datos y/o entidades prediciendo el estado futuro de las cosas o prediciendo los datos de salida de los sensores para así optimizar los tiempos de procesamiento, y respuesta del sistema.

El sistema RI propuesto IR.WoT ha adaptado los mecanismos de indexación, ponderación y re-organización tradicionales al entorno dinámico IoT|WoT. Esto demuestra dos puntos importantes, por una parte la posibilidad de adaptación de mecanismos existentes mientras se construye sobre la base de conocimiento existente, al tiempo que, por otra parte, se explora la adición, evolución de nuevas aproximaciones que aborden las características propias de IoT|WoT.

La arquitectura de un sistema RI debe estar determinada por dos dimensiones principales, la efectividad, es decir, dada una consulta se muestre alta calidad con los resultados más relevantes, y la eficiencia en términos del tiempo de respuesta y rendimiento computacional para las consultas del usuario.

En este trabajo se ha incluido un análisis para los desafíos de IoTSE y WoTSE, señalando las preguntas abiertas restantes y los problemas que aún están presentes en la investigación de modelos y técnicas de RI para IoT|WoT. Técnicamente hablando, todas las etapas de RI se han visto profundamente afectadas por la nueva naturaleza de WoT. Por medio de una revisión exhaustiva de la literatura se han identificado nuevos enfoques para resolver los problemas en el descubrimiento, rastreo, indexación y recuperación de información para sistemas IoT|WoT. Los avances en la arquitectura WoT han sido acompañados por estandarización y recomendaciones técnicas en la W3C. Aún cuando, el número de propuestas WoT no emplean los estándares W3C propuestos, es indiscutible la madurez que se ha alcanzado en las tecnologías IoT|WoT. Desde una perspectiva netamente técnica, ha de declararse que existe aun un camino por recorrer. Por consiguiente, no hay escenarios completamente desarrollados para una evaluación tradicional de los desarrollos en los sistemas RI sobre los paradigmas IoT|WoT.

La heterogeneidad en las propuestas IoT|WoT y los acercamientos de búsqueda IoTSE tanto como WoTSE aun enfrentan desafíos que requieren de trabajo futuro e investigación. En este trabajo se han considerado una perspectiva integradora de las propuestas de evaluación de sistemas RI para IoT|WoT. Una parte de la evaluación presentada apunta al rendimiento del sistema en términos de uso eficiencia en tiempo. Por otra parte, se han discutidos los retos y se ha propuesto la reutilización de modelos de evaluación de usuario final para establecer las bases de creación de colecciones de prueba dinámica y determinar la efectividad del sistema RI en términos tradicionales de relevancia para el usuario final.

La realidad virtual y la aumentada juegan y jugarán un rol crucial en la próxima generación de motores de búsqueda. El dinamismo no sólo será un factor a considerar sino que la alta interactividad con el mundo real a través de sus representaciones digitales será el nuevo campo de investigación en presentación y visualización de resultados. En el sistema IR.WoT se ha propuesto mezclar las listas de resultados convencionales adaptadas al contenido y al contexto IoT|WoT siguiendo el modelo abstracto WoT propuesto. Este se ha complementado con una visualización geográfica simultánea. Nuevas líneas de investigación nacen desde las nuevas necesidades de los usuarios para interactuar con el mundo real, WoT y sus entidades.

## **6.2. Trabajo Futuro y Nuevas Líneas de Investigación**

Desde una perspectiva de usuario final, las encuestas aplicadas arrojan resultados generales satisfactorios. Es claro, que los motores de búsqueda actuales han ido incorporando paulatinamente con el tiempo nuevas características y su evolución a sentado una base de rendimiento y expectativas mínimas de interacción. Con base en las respuestas y entrevistas realizadas se sumarizan brevemente algunas de las líneas de investigación que pueden establecerse como trabajo futuro, complementadas con perspectivas de evolución de la revisión sistemática de literatura realizada, ver Tabla 6.1.

### **6.2.1. Marcos de Modelado, Simulación y Prototipado para WoT**

Como trabajo futuro, es importante abordar la representación del contexto espacial en simulación y en la arquitectura WoT propiamente dicha. El contexto espacio-temporal se puede mejorar por medio de sistemas de información geográfica (SIG) de varias capas y un enfoque de subdivisión espacial, donde los datos de representación SIG del mundo real incluyen por ejemplo capas de elevación y del uso del suelo y con capas vectoriales que se pueden descomponer y fusionar para generar Entornos geográficos virtuales (VGE). Tal como se mencionó anteriormente, la representación gráfica y el cruce de resultados con realidad virtual aumentada pueden proporcionar hallazgos fundamentales para proveer al usuario con información relevante, altamente dinámica y con alto grado de interactividad.

En futuras investigaciones, es importante estudiar y evaluar el impacto de los mecanismos de ponderación independientes de la consulta en la forma de modelos de predicción de datos en la capa sensorica. Estos han probado en la literatura tener impacto positivo en el diseño de componentes internos de los nuevos sistemas RI para IoT|WoT y su optimización. Estos son un componente fundamental para el desarrollo de nuevos mecanismos de fusión

de información y reducción de la redundancia incrementando la efectividad en tiempo y recursos.

En una etapa posterior, el uso de modelos semánticos basados en RDF, RDFa y ontologías OWL y sus variantes pueden permitir un enfoque homogéneo entre dominios de aplicación con la finalidad de intercambiar información en el ecosistema WoT. El enfoque propuesto por IR.WoT puede evolucionar a un sistema de RI en tiempo real basado en ontologías para WoT en escenarios de aplicaciones específicas. Una opción clara para lograr este objetivo de enriquecimiento semántico es mediante la conversión de modelos los sintácticos JSON/XML existentes a ontologías en formatos OWL/RDF.

### 6.2.2. Heterogeneidad conceptual en IoTSE|WoTSE

Cuatro obstáculos principales ensombrecen la evolución del campo RI para IoT|WoT debido a la heterogeneidad de los conceptos.

*Difícil de Reproducir* debido a la inexistencia de conjuntos de datos extendidos de IoT|WoT [133], que en la mayoría de los trabajos son propietarios o privados. Estos *datasets* se requieren para construir conjuntos de datos abiertos y públicos para la investigación de IoTSE|WoTSE. De hecho, en la actualidad se emplean algunos conjuntos de datos públicos y conocidos; sin embargo, estos *datasets* tiene una orientación hacia escenarios específicos de aplicación en lugar de proporcionar una perspectiva completa RI. Además, los experimentos completos conocidos no se pueden replicar ya sea por falta de conjuntos de datos o por problemas en la reproducibilidad de los experimentos. El problema de la reproducibilidad está relacionado con tres factores principales:

- Falta de conjunto de datos: uso de un conjunto de datos de propiedad o uso de una submuestra de un conjunto de datos disponible públicamente que no se ha publicado o para la cual no se ha divulgado la submuestra;
- Falta de detalles sobre la metodología de evaluación (procesamiento de datos) y protocolo (tareas);
- Falta de detalles sobre los parámetros del sistema.

*Difícil de Evaluar/Comparar* es una implicación del punto anterior, que ensombrece la evolución de los sistemas RI, dada la divergencia en los criterios de evaluación y las métricas de desempeño. Un conjunto importante de trabajos toma como imprescindibles métricas basadas en el tiempo para evaluar sus planteamientos. Sin embargo, debemos confiar en la versión ajustada o modificada de las métricas RI para comparar el rendimiento no solo en términos de eficiencia (complejidad de tiempo y espacio) sino también en términos de

efectividad, como métricas RI fuera de línea y en línea. Según nuestra revisión de la literatura, no existen esfuerzos documentados para estandarizar el proceso de evaluación mediante la creación de algunos protocolos de evaluación o la adopción de algunos enfoques ya existentes (mediante adaptación o extensión).

*Difícil de Acceder* Este es otro punto importante a mencionar. Es visto con optimismo, la creación de nuevas colecciones de conjuntos de datos de acceso público, como en Google Cloud Platform<sup>61</sup> con casi 200 conjuntos de datos. Algunos ejemplos de conjuntos de datos en vivo y dinámicos son Chicago Taxi Trips y NYC TLC Trips. Además, Amazon AWS proporciona acceso abierto a 175 conjuntos de datos<sup>62</sup> orientados principalmente a fines médicos y espaciales. Sin embargo, la disponibilidad de conjuntos de datos no es el único punto de bloqueo, ya que todavía hay muchos dominios y datos que no están abiertos a los investigadores.

*Difícil de Reutilizar.* [133] apunta la existencia de diferentes implementaciones para componentes RI modulares similares. La dificultad de reutilizar componentes proviene del uso de arquitecturas e interfaces no típicas entre componentes. La reutilización se puede lograr mediante la construcción de arquitecturas, descriptores y bibliotecas acordados. La estandarización se convierte en un pilar estructural para la próxima generación de IoTSE/-WoTSE.

### 6.2.3. Seguridad, Privacidad y Confianza en RI para IoT|WoT

La seguridad, privacidad y la confianza de motores de búsqueda para IoT|WoT se consideran desafíos críticos y requieren de nuevas líneas de investigación. Es mandatorio que los datos dentro de los sistemas IoTSE|WoTSE estén protegidos, ser confidenciales y privados en todas las tareas de RI. La seguridad, la privacidad y la confianza han sido temas relativamente poco explorados en los últimos años. Desde 2020, se pueden reconocer mayores esfuerzos para cubrir estos tres pilares de ciber-seguridad dentro de los sistemas IoTSE|WoTSE. Para ejemplificar sólo un caso de estudio, Yang et al. [142] proponen una estrategia de selección de las fuentes de información que participaran como originarios de datos IoT|WoT con protección de la privacidad para sistemas IoTSE. Este proporciona una nueva perspectiva, para la gestión del anonimato para el usuario final y para la solicitud en el momento de una consulta. Es también importante reconocer que los sistemas RI para IoT|WoT tienen que gestionar las necesidades de privacidad en entornos donde la legislación o la criticidad de los sistemas así lo requieran.

<sup>61</sup><https://cloud.google.com/public-datasets>

<sup>62</sup><https://registry.opendata.aws/>



Es una minoría, aquellos trabajos que prestan ahora atención a la investigación sobre la adhesión de las dimensiones de seguridad, privacidad y confianza en IoTSE|WoTSE. Otro claro ejemplo, es la prestación de servicios de confianza durante la búsqueda. Al igual que la necesidad de que existan mecanismos de autorización de acceso y uso compartido de recursos tanto IoT|WoT como IoTSE|WoTSE. Esto involucra por supuesto, nuevas líneas de investigación sobre esquemas de cifrado de búsqueda basado en atributos ligeros, con control de acceso detallado y de autorización que permitan una búsqueda bajo mecanismos de confianza. Por último punto, es importante anotar que nuevos estudios han identificado y caracterizado los problemas de seguridad, desafíos y vulnerabilidades de seguridad que enfrentan los sistemas IoTSE|WoTSE [42].

Tema de investigación	Plazo	Descripción de la Investigación
Marcos experimentales	Corto	Se han presentado marcos experimentales en la literatura. Se requiere un enfoque especial en la convergencia o en proporcionar accesibilidad, reutilización y reproducibilidad de los criterios de evaluación entre todos los enfoques. Estos son los pilares fundamentales necesarios para impulsar y estimular el desarrollo y la evolución de RI-IoT/WoT.
Criterios comunes de evaluación	Corto	Es concluyente, y en concordancia con lo expuesto en Tran et al. [133], [131] que la investigación de RI para IoT y WoT es un área compleja con desafíos técnicos y de estandarización que superar. La comparabilidad de los trabajos requiere no sólo la estandarización de conceptos y modelos, sino, lo que es más importante, un marco de evaluación común similar a la iniciativa TREC <a href="https://trec.nist.gov/trec.tracks.html">https://trec.nist.gov/trec.tracks.html</a> .
RI en el WoT Social	Largo	Khadir et al. [58] denominados avatares sociales, un tipo particular de avatares de WoT que pueden vincularse entre sí según compromisos y criterios de similitud, proporcionando inteligencia colectiva para respuestas más precisas y descubrimiento de servicios de WoT. Se requerirá un mecanismo de RI que se ocupe del tipo de consultas de contexto social con respecto a los avatares sociales. Es esperado por los usuarios finales una manera más social de interactuar y buscar sobre la WoT. Lo que supone la opción de compartir cosas en el mundo real y recuperar información sobre estas relaciones sociales cosas-usuarios.
Progresivo y Personalizado	Largo	Se argumenta que el enriquecimiento semántico se enfoca en encontrar similitudes pero no en brindar resultados personalizados centrados en el usuario [97]. El mecanismo progresivo tendrá como objetivo reducir el espacio de búsqueda al tiempo que reduce el tiempo de consulta de manera sistemática. Al mismo tiempo, algún enriquecimiento semántico debería personalizar los resultados del usuario, explorando más allá del clásico contexto espacio-temporal y de los mecanismos <i>Learn-to-Rank</i> .
Búsqueda basada en Emociones y Sentidos	Largo	Más allá de los límites tradicionales de la recuperación de información, Luis-Ferreira et al. [71] proponen una búsqueda multimodal que incluye sensaciones y emociones como conductores para buscar cosas, imitando la forma en que nuestros cerebros almacenan y encuentran cosas por experiencia. Se requiere proporcionar una contribución clara y válida a la descripción sensorial de las cosas y la búsqueda futurista multimodal requerida en futuras investigaciones.

Tabla 6.1 Temas de investigación a corto y largo plazo para RI-IoT e RI-WoT.



# Bibliografía

- [1] Aziez, M., Benharzallah, S., and Bennoui, H. (2017). Service discovery for the internet of things: Comparison study of the approaches. In *2017 4th International Conference on Control, Decision and Information Technologies (CoDIT)*, pages 0599–0604, New York, NY, USA. IEEE.
- [2] Bao, Y., Qiu, W., and Cheng, X. (2022). Secure and lightweight fine-grained searchable data sharing for iot-oriented and cloud-assisted smart healthcare system. *IEEE Internet of Things Journal*, 9(4):2513–2526.
- [3] Barnaghi, P., Wang, W., Dong, L., and Wang, C. (2013). A linked-data model for semantic sensor streams. In *2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing*, pages 468–475, New York, NY, USA. IEEE.
- [4] Belahyane, I., Mammass, M., Abioui, H., Moutaoukkil, A., and Idarrou, A. (2022). Structural information retrieval in xml documents: A graph-based approach. *International Journal of Advanced Computer Science and Applications*, 13(3).
- [5] Benrazek, A.-E., Kouahla, Z., Farou, B., Ferrag, M. A., Seridi, H., and Kurulay, M. (2020). An efficient indexing for internet of things massive data based on cloud-fog computing. *Transactions on Emerging Telecommunications Technologies*, 31(3):e3868.
- [6] Bessai-Mechmache, F. Z., Hammouche, K., and Alimazighi, Z. (2020). A genetic algorithm-based xml information retrieval model. In *2020 21st International Arab Conference on Information Technology (ACIT)*, pages 1–5.
- [7] Büttcher, S., Clarke, C. L. A., and Cormack, G. V. (2010). *Information retrieval : implementing and evaluating search engines*. MIT Press, Cambridge, Mass.
- [8] Cabral, L., Compton, M., and Müller, H. (2014). A use case in semantic modelling and ranking for the sensor web. In Mika, P., Tudorache, T., Bernstein, A., Welty, C., Knoblock, C., Vrandečić, D., Groth, P., Noy, N., Janowicz, K., and Goble, C., editors, *The Semantic Web – ISWC 2014*, pages 276–291, Cham. Springer International Publishing.
- [9] Charpenay, V. and Käbisch, S. (2020). On modeling the physical world as a collection of things: The w3c thing description ontology. In Harth, A., Kirrane, S., Ngonga Ngomo, A.-C., Paulheim, H., Rula, A., Gentile, A. L., Haase, P., and Cochez, M., editors, *The Semantic Web*, pages 599–615, Cham. Springer International Publishing.

- [10] Charpenay, V., Käbisch, S., and Kosch, H. (2018). Semantic data integration on the web of things. In *Proceedings of the 8th International Conference on the Internet of Things, IOT '18*, New York, NY, USA. Association for Computing Machinery.
- [11] Cheng, H., Zhang, T., Yang, Y., Yan, F., Teague, H., Chen, Y., and Li, H. (2019). Msnet: Structural wired neural architecture search for internet of things. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 2033–2036, New York, NY, USA. IEEE.
- [12] Chinenyanga, T. T. and Kushmerick, N. (2001). Expressive retrieval from xml documents. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '01*, page 163–171, New York, NY, USA. Association for Computing Machinery.
- [13] Choudhury, F., Rahman, F., Jamil, R., and Mansoor, N. (2020). Sensor searching techniques in internet of things: A survey, taxonomy, and challenges. *ICT Analysis and Applications*, 2(154):755–763.
- [14] Christophe, B., Verdot, V., and Toubiana, V. (2011). Searching the “web of things”. In *Proceedings of the 2011 IEEE Fifth International Conference on Semantic Computing, ICSC '11*, page 308–315, USA. IEEE Computer Society.
- [15] Cimmino, A. and García-Castro, R. (2022). Wothive: Enabling syntactic and semantic discovery in the web of things. *Open Journal of Internet Of Things (OJIOT)*, 8(1):54–65.
- [16] Cimmino, A., Poveda-Villalón, M., and García-Castro, R. (2020). ewot: A semantic interoperability approach for heterogeneous iot ecosystems based on the web of things. *Sensors*, 20(3):822.
- [17] De, S., Elsaleh, T., Barnaghi, P. M., and Meissner, S. (2012). An internet of things platform for real-world and digital objects. *Scalable Computing: Practice and Experience*, 13.
- [18] De, S., Zhou, Y., and Moessner, K. (2017). Chapter 1 - ontologies and context modeling for the web of things. In Sheng, Q. Z., Qin, Y., Yao, L., and Benatallah, B., editors, *Managing the Web of Things*, pages 3 – 36. Morgan Kaufmann, Boston.
- [19] Deng, Y., Zhang, Z., Liao, J., and Yang, L. T. (2017). Sim: A search engine by correlating scattered data sets for cyber, physical, and social systems. *IEEE Systems Journal*, 11(1):345–355.
- [20] Dhanalekshmi, G. and Asawa, K. (2017). Performance evaluation of various data structures in building efficient indexing schemes for xml documents. In *2017 Tenth International Conference on Contemporary Computing (IC3)*, pages 1–3.
- [21] Diaconescu, I.-M. and Wagner, G. (2014). Towards a general framework for modeling, simulating and building sensor/actuator systems and robots for the web of things. In *MORSE@ STAF*, pages 30–41. Citeseer.
- [22] Diamantini, C., Nocera, A., Potena, D., Storti, E., and Ursino, D. (2021). Querying the iot using multiresolution contexts. *IEEE Internet of Things Journal*, 8(7):6127–6139.

- [23] Ding, Z., Dai, J., Gao, X., and Yang, Q. (2012). A hybrid search engine framework for the internet of things. In *2012 Ninth Web Information Systems and Applications Conference*, pages 57–60, New York, NY, USA. IEEE.
- [24] Doan, Q.-T., Kayes, A. S. M., Rahayu, W., and Nguyen, K. (2022). A framework for iot streaming data indexing and query optimisation. *IEEE Sensors Journal*, Early Access(Early Access):1–1.
- [25] Du, C., Zhou, Z., Shu, L., Jia, X., and Wang, Q. (2013). An effective iot services indexing and query technique. In *2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing*, pages 777–782, New York, NY, USA. IEEE.
- [26] Du, C., Zhou, Z., Ying, S., Niu, J., and Wang, Q. (2015). An efficient indexing and query mechanism for ubiquitous iot services. *Int. J. Ad Hoc Ubiquitous Comput.*, 18(4):245–255.
- [27] Ebrahimi, M., Shafieibavani, E., Wong, R. K., and Chi, C. (2015). A new meta-heuristic approach for efficient search in the internet of things. In *2015 IEEE International Conference on Services Computing*, pages 264–270, New York, NY, USA. IEEE.
- [28] Elahi, B. M., Romer, K., Ostermaier, B., Fahrmaier, M., and Kellerer, W. (2009). Sensor ranking: A primitive for efficient content-based sensor search. In *2009 International Conference on Information Processing in Sensor Networks*, pages 217–228, New York, NY, USA. IEEE.
- [29] Evdokimov, S., Fabian, B., Kunz, S., and Schoenemann, N. (2010). Comparison of discovery service architectures for the internet of things. In *2010 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing*, pages 237–244, New York, NY, USA. IEEE.
- [30] Faheem, M. R., Anees, T., and Hussain, M. (2019). The web of things: Findability taxonomy and challenges. *IEEE Access*, 7:185028–185041.
- [31] Fathy, Y., Barnaghi, P., Enshaeifar, S., and Tafazolli, R. (2016). A distributed in-network indexing mechanism for the internet of things. In *2016 IEEE 3rd World Forum on Internet of Things (WF-IoT)*, pages 585–590, New York, NY, USA. IEEE.
- [32] Feki, M. A., Kawsar, F., Boussard, M., and Trappeniers, L. (2013). The internet of things: The next technological revolution. *Computer*, 46(2):24–25.
- [33] Ferrag, M. A., Kouahla, Z., Seridi, H., and Kurulay, M. (2019). Big iot data indexing: Architecture, techniques and open research challenges. In *2019 International Conference on Networking and Advanced Systems (ICNAS)*, pages 1–6, New York, NY, USA. IEEE.
- [34] Frank, C., Bolliger, P., Mattern, F., and Kellerer, W. (2008). The sensor internet at work: Locating everyday items using mobile phones. *Pervasive and Mobile Computing*, 4(3):421 – 447.
- [35] Fuhr, N. and Grossjohann, K. (2004). Xirql: An xml query language based on information retrieval concepts. *ACM Trans. Inf. Syst.*, 22(2):313–356.

- [36] Geva, S., Kamps, J., and Schenkel, R., editors (2012). *Focused Retrieval of Content and Structure, 10th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2011, Saarbrücken, Germany, December 12-14, 2011, Revised Selected Papers*, volume 7424 of *Lecture Notes in Computer Science*. Springer.
- [37] Giang, N. K., Ha, M., and Kim, D. (2015). Buddy thing: Browsing as a service for the internet of things. In *2015 IEEE International Conference on Services Computing*, pages 122–129.
- [38] Gomes, P., Cavalcante, E., Rodrigues, T., Batista, T., Delicato, F. C., and Pires, P. F. (2015). A federated discovery service for the internet of things. In *Proceedings of the 2nd Workshop on Middleware for Context-Aware Applications in the IoT, M4IoT 2015*, page 25–30, New York, NY, USA. Association for Computing Machinery.
- [39] Gopinathan, D. and Asawa, K. (2017). New path based index structure for processing cas queries over xml database. *Journal of computing and information technology*, 25(3):211–225.
- [40] Guinard, D. (2011). *A Web of Things Application Architecture – Integrating the Real-World into the Web*. PhD thesis, ETH Zurich, Zurich, Switzerland.
- [41] Hatcher, W. G., Qian, C., Gao, W., Liang, F., Hua, K., and Yu, W. (2021). Towards efficient and intelligent internet of things search engine. *IEEE Access*, 9:15778–15795.
- [42] Hatcher, W. G., Qian, C., Liang, F., Liao, W., Blasch, E., and Yu, W. (2022). Secure iot search engine: Survey, challenges issues, case study, and future research direction. *IEEE Internet of Things Journal*, Early Access(Early Access):1–1.
- [43] Henson, C. A., Pschorr, J. K., Sheth, A. P., and Thirunarayan, K. (2009). Semsos: Semantic sensor observation service. In *2009 International Symposium on Collaborative Technologies and Systems*, pages 44–53, New York, NY, USA. IEEE.
- [44] Hodges, S., Taylor, S., Villar, N., Scott, J., Bial, D., and Fischer, P. T. (2013). Prototyping connected devices for the internet of things. *Computer*, 46(2):26–34.
- [45] Iggena, T., Bin Ilyas, E., Fischer, M., Tönjes, R., Elsaleh, T., Rezvani, R., Pourshahrokhi, N., Bischof, S., Fernbach, A., Xavier Parreira, J., Schneider, P., Smirnov, P., Strohbach, M., Truong, H., González-Vidal, A., Skarmeta, A. F., Singh, P., Beliatis, M. J., Presser, M., Martinez, J. A., Gonzalez-Gil, P., Krogbæk, M., and Holmgård Christophersen, S. (2021). Iotcrawler: Challenges and solutions for searching the internet of things. *Sensors*, 21(5):1–32.
- [46] Ihm, S.-Y., Nasridinov, A., and Park, Y.-H. (2014). Grid-ppps: A skyline method for efficiently handling top-  $k$  queries in internet of things. *J. Appl. Math.*, 2014, Special Issue:10 pages.
- [47] Itakura, K. Y. and Clarke, C. L. (2010). A framework for bm25f-based xml retrieval. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '10*, page 843–844, New York, NY, USA. Association for Computing Machinery.

- [48] Jara, A. J., Lopez, P., Fernandez, D., Castillo, J. F., Zamora, M. A., and Skarmeta, A. F. (2013). Mobile digcovery: A global service discovery for the internet of things. In *2013 27th International Conference on Advanced Information Networking and Applications Workshops*, pages 1325–1330, New York, NY, USA. IEEE.
- [49] Jara, A. J., Martinez-Julia, P., and Skarmeta, A. (2012). Light-weight multicast dns and dns-sd (lmdns-sd): Ipv6-based resource and service discovery for the web of things. In *2012 Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, pages 731–738, New York, NY, USA. IEEE.
- [50] Jirka, S., Bröring, A., and Stasch, C. (2009). Discovery mechanisms for the sensor web. *Sensors*, 9(4):2661–2681.
- [51] Kajimoto, M. K. R. M. M. L. T. K. K. T. K. (2022). Web of things (wot) architecture.
- [52] Kamilaris, A., Papakonstantinou, K., and Pitsillides, A. (2014). Exploring the use of dns as a search engine for the web of things. In *2014 IEEE World Forum on Internet of Things (WF-IoT)*, pages 100–105, New York, NY, USA. IEEE.
- [53] Kamilaris, A., Yumusak, S., and Ali, M. I. (2016). Wots2e: A search engine for a semantic web of things. In *2016 IEEE 3rd World Forum on Internet of Things (WF-IoT)*, pages 436–441, New York, NY, USA. IEEE.
- [54] Kansal, A., Nath, S., Liu, J., and Zhao, F. (2007). Senseweb: An infrastructure for shared sensing. *IEEE MultiMedia*, 14:8–13.
- [55] Karim, F., Al Naameh, O., Lytra, I., Mader, C., Vidal, M., and Auer, S. (2018). Semantic enrichment of iot stream data on-demand. In *2018 IEEE 12th International Conference on Semantic Computing (ICSC)*, pages 33–40, New York, NY, USA. IEEE.
- [56] Karim, F., Lytra, I., Mader, C., Auer, S., and Vidal, M.-E. (2018). Desert: A continuous sparql query engine for on-demand query answering. *International Journal of Semantic Computing*, 12(03):373–397.
- [57] Kemouguette, I., Kouahla, Z., Benrazek, A.-E., Farou, B., and Seridi, H. (2021). Cost-effective space partitioning approach for iot data indexing and retrieval. In *2021 International Conference on Networking and Advanced Systems (ICNAS)*, pages 1–6, New York, NY, USA. IEEE.
- [58] Khadir, K., Guermouche, N., Monteil, T., and Guittoum, A. (2020). Towards avatar-based discovery for iot services using social networking and clustering mechanisms. In *2020 16th International Conference on Network and Service Management (CNSM)*, pages 1–7, New York, NY, USA. IEEE.
- [59] Kolcun, R. and McCann, J. A. (2014). Dragon: Data discovery and collection architecture for distributed iot. In *2014 International Conference on the Internet of Things (IOT)*, pages 91–96, New York, NY, USA. IEEE.
- [60] Komatsuzaki, M., Tsukada, K., Siio, I., Verronen, P., Luimula, M., and Pieskä, S. (2011). Iteminder: Finding items in a room using passive rfid tags and an autonomous



- robot (poster). In *Proceedings of the 13th International Conference on Ubiquitous Computing*, UbiComp '11, page 599–600, New York, NY, USA. Association for Computing Machinery.
- [61] Lalmas, M. and Trotman, A. (2009). *XML Retrieval*, pages 3616–3621. Springer US, Boston, MA.
- [62] Le-Phuoc, D., Dao-Tran, M., Xavier Parreira, J., and Hauswirth, M. (2011). A native and adaptive approach for unified processing of linked streams and linked data. In Aroyo, L., Welty, C., Alani, H., Taylor, J., Bernstein, A., Kagal, L., Noy, N., and Blomqvist, E., editors, *The Semantic Web – ISWC 2011*, pages 370–388, Berlin, Heidelberg. Springer Berlin Heidelberg.
- [63] Le-Phuoc, D., Nguyen-Mau, H. Q., Parreira, J. X., and Hauswirth, M. (2012). A middleware framework for scalable management of linked streams. *Journal of Web Semantics*, 16:42 – 51. The Semantic Web Challenge 2011.
- [64] Li, J. and Liu, L. (2018). Research on temporal information retrieval method based on data graph. In *2018 10th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, volume 02, pages 261–264.
- [65] Li, M., Chen, H., Huang, X., and Cui, L. (2015). Easicrawl: A sleep-aware schedule method for crawling iot sensors. In *2015 IEEE 21st International Conference on Parallel and Distributed Systems (ICPADS)*, pages 148–155, New York, NY, USA. IEEE.
- [66] Li, X., Kim, Y. J., Govindan, R., and Hong, W. (2003). Multi-dimensional range queries in sensor networks. In *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*, SenSys '03, page 63–75, New York, NY, USA. Association for Computing Machinery.
- [67] Liang, F., Qian, C., Hatcher, W. G., and Yu, W. (2019). Search engine for the internet of things: Lessons from web search, vision, and opportunities. *IEEE Access*, 7:104673–104691.
- [68] Liang, S. and Huang, C.-Y. (2013). Geocens: A geospatial cyberinfrastructure for the world-wide sensor web. *Sensors*, 13(10):13402–13424.
- [69] Liu, M., Li, D., Chen, Q., Zhou, J., Meng, K., and Zhang, S. (2018). Sensor information retrieval from internet of things: Representation and indexing. *IEEE Access*, 6:36509–36521.
- [70] Liu, M., Li, D., Zeng, Y., Huang, W., Meng, K., and Chen, H. (2020). Combinatorial-oriented feedback for sensor data search in internet of things. *IEEE Internet of Things Journal*, 7(1):284–297.
- [71] Luis-Ferreira, F. and Jardim-Gonçalves, R. (2013). Modelling of things on the internet for the search by the human brain. In Camarinha-Matos, L. M., Tomic, S., and Graça, P., editors, *Technological Innovation for the Internet of Things*, pages 71–79, Berlin, Heidelberg. Springer Berlin Heidelberg.

- [72] Lunardi, W. T., de Matos, E., Tiburski, R., Amaral, L. A., Marczak, S., and Hessel, F. (2015). Context-based search engine for industrial iot: Discovery, search, selection, and usage of devices. In *2015 IEEE 20th Conference on Emerging Technologies Factory Automation (ETFA)*, pages 1–8, New York, NY, USA. IEEE.
- [73] Ma, H. and Liu, W. (2018). A progressive search paradigm for the internet of things. *IEEE MultiMedia*, 25(1):76–86.
- [74] Manning, C., Raghavan, P., and Schütze, H. (2008). Introduction to information retrieval. *Natural Language Engineering*, 16(1):100–103.
- [75] Manta-Caro, C. and Fernández-Luna, J. M. (2017). Modeling and simulating the web of things from an information retrieval perspective. *ACM Trans. Web*, 12(1).
- [76] Manta-Caro, C. and Fernández-Luna, J. M. (2016). Advances in real-time indexing models and techniques for the web of things. In *2016 8th IEEE Latin-American Conference on Communications (LATINCOM)*, pages 1–6, New York, NY, USA. IEEE.
- [77] Mayer, S. and Guinard, D. (2011). An extensible discovery service for smart things. In *Proceedings of the Second International Workshop on Web of Things, WoT '11*, New York, NY, USA. Association for Computing Machinery.
- [78] Mayer, S., Guinard, D., and Trifa, V. (2012). Searching in a web-based infrastructure for smart things. In *2012 3rd IEEE International Conference on the Internet of Things*, pages 119–126, New York, NY, USA. IEEE.
- [79] Meriem, A., Benharzallah, S., and Bennoui, H. (2019). A full comparison study of service discovery approaches for internet of things. *Int. J. Pervasive Computing and Communications*, 15(1):30–56.
- [80] Michel, J. and Julien, C. (2014). A cloudlet-based proximal discovery service for machine-to-machine applications. In Memmi, G. and Blanke, U., editors, *Mobile Computing, Applications, and Services*, pages 215–232, Cham. Springer International Publishing.
- [81] Michel, J., Julien, C., and Payton, J. (2014). Gander: Mobile, pervasive search of the here and now in the here and now. *IEEE Internet of Things Journal*, 1(5):483–496.
- [82] Mietz, R., Groppe, S., Römer, K., and Pfisterer, D. (2013). Semantic models for scalable search in the internet of things. *Journal of Sensor and Actuator Networks*, 2(2):172–195.
- [83] Mietz, R. and Römer, K. (2011). Exploiting correlations for efficient content-based sensor search. In *2011 IEEE sensors proceedings*, pages 187–190, United States. Institute of Electrical and Electronics Engineers.
- [84] Minerva, R., Biru, A., and Rotondi, D. (2015). Towards a definition of the internet of things (iot). *IEEE Internet Initiative*, 1(1):1–86.
- [85] Mohamed, M. A., Kardas, G., and Challenger, M. (2021). Model-driven engineering tools and languages for cyber-physical systems—a systematic literature review. *IEEE Access*, 9:48605–48630.

- [86] Mohammed, A., Obaid, H., Pattnaik, P. K., and Pani, S. K. (2018). Towards on cloud of things: Survey, challenges, open research issues, and tools. In *2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*, pages 1458–1463.
- [87] Mrissa, M., Médini, L., and Jamont, J. (2014). Semantic discovery and invocation of functionalities for the web of things. In *2014 IEEE 23rd International WETICE Conference*, pages 281–286, New York, NY, USA. IEEE.
- [88] Mrissa, M., Médini, L., Jamont, J., Le Sommer, N., and Laplace, J. (2015). An avatar architecture for the web of things. *IEEE Internet Computing*, 19(2):30–38.
- [89] Nadim, I., Elghayam, Y., and Sadiq, A. (2018). Semantic discovery architecture for dynamic environments of web of things. In *2018 International Conference on Advanced Communication Technologies and Networking (CommNet)*, pages 1–6, New York, NY, USA. IEEE.
- [90] Nath, S., Liu, J., and Zhao, F. (2007). Sensormap for wide-area sensor webs. *IEEE Computer*, 40:90–93.
- [91] Negm, E., Makady, S., and Salah, A. (2020). Towards ontology-based domain specific language for internet of things. In *Proceedings of the 2020 9th International Conference on Software and Information Engineering (ICSIE)*, ICSIE 2020, page 146–151, New York, NY, USA. Association for Computing Machinery.
- [92] Noura, M., Atiquzzaman, M., and Gaedke, M. (2019). Interoperability in internet of things: Taxonomies and open challenges. *Mob. Netw. Appl.*, 24(3):796–809.
- [93] Nunes, L. H., Estrella, J. C., Nakamura, L. H. V., Libardi, R. M. D. O., Ferreira, C. H. G., Jorge, L., Perera, C., and Reiff-Marganiec, S. (2016). A distributed sensor data search platform for internet of things environments. *CoRR*, abs/1606.07932:1–12.
- [94] Ostermaier, B., Römer, K., Mattern, F., Fahrmaier, M., and Kellerer, W. (2010). A real-time search engine for the web of things. In *2010 Internet of Things (IOT)*, pages 1–8, New York, NY, USA. IEEE.
- [95] Parreira, J. S., Smirnov, P. A., Martinez, J. A. O., Rezvani, R. U., González Gil, P. U., and Pourshahrokhi, N. U. (2020). *D4.2 Large Scale IoT Crawling, Indexing and Ranking*. Siemens AG Austria, Austria.
- [96] Pattar, S., Buyya, R., Venugopal, K. R., Iyengar, S. S., and Patnaik, L. M. (2018). Searching for the iot resources: Fundamentals, requirements, comprehensive review, and future directions. *IEEE Communications Surveys Tutorials*, 20(3):2101–2132.
- [97] Pattar, S., Kulkarni, D. S., Vala, D., Buyya, R., K. R., V., Iyengar, S. S., and Patnaik, L. M. (2019). Progressive search algorithm for service discovery in an iot ecosystem. In *2019 International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, pages 1041–1048, New York, NY, USA. IEEE.

- [98] Pavlopoulou, N. and Curry, E. (2022). Possum: An entity-centric publish/subscribe system for diverse summarization in internet of things. *ACM Trans. Internet Technol.*, 22(3).
- [99] Penchala, S. K. (2019). Search engine on semantic web of things. *International Journal for Research in Applied Science and Engineering Technology*, 7:909–911.
- [100] Perera, C. and Vasilakos, A. V. (2016). A knowledge-based resource discovery for internet of things. *Knowledge-Based Systems*, 109:122 – 136.
- [101] Perera, C., Zaslavsky, A., Christen, P., Compton, M., and Georgakopoulos, D. (2013). Context-aware sensor search, selection and ranking model for internet of things middleware. In *2013 IEEE 14th International Conference on Mobile Data Management*, volume 1, pages 314–322, New York, NY, USA. IEEE.
- [102] Perera, C., Zaslavsky, A., Liu, C. H., Compton, M., Christen, P., and Georgakopoulos, D. (2014). Sensor search techniques for sensing as a service architecture for the internet of things. *IEEE Sensors Journal*, 14(2):406–420.
- [103] Pfisterer, D., Romer, K., Bimschas, D., Kleine, O., Mietz, R., Truong, C., Hasemann, H., Krölller, A., Pagel, M., Hauswirth, M., Karnstedt, M., Leggieri, M., Passant, A., and Richardson, R. (2011). Spitfire: toward a semantic web of things. *IEEE Communications Magazine*, 49(11):40–48.
- [104] Pintus, A., Carboni, D., and Piras, A. (2012). Paraimpu: A platform for a social web of things. In *Proceedings of the 21st International Conference on World Wide Web, WWW '12 Companion*, page 401–404, New York, NY, USA. Association for Computing Machinery.
- [105] Pschorr, J., Henson, C. A., Patni, H., and Sheth, A. P. (2010). Sensor discovery on linked data. In *Proceedings of the 7th Extended Semantic Web Conference, ESWC2010*, Heraklion, Greece. Springer Link.
- [106] Qian, C., Gao, W., Hatcher, W. G., Liao, W., Lu, C., and Yu, W. (2020). Search engine for heterogeneous internet of things systems and optimization. In *2020 IEEE Intl Conf on Dependable, Autonomic and Secure Computing*, pages 475–482, New York, NY, USA. IEEE.
- [107] Qian, X. and Che, X. (2012). Security-enhanced search engine design in internet of things. *j-jucs*, 18(9):1218–1235.
- [108] Qin, Y., Shemshadi, A., Sheng, Q., and Alzubaidi, A. (2016). Ceiot: A framework for interlinking smart things in the internet of things. In *Advanced Data Mining and Applications - 12th International Conference, ADMA 2016, Proceedings*, Lecture Notes in Computer Science, pages 203–218, Germany. Springer Verlag.
- [109] Raghu Nandan, R., Nalini, N., and Hamsavath, P. N. (2022). Iot-cbse: A search engine for semantic internet of things. In Shetty, N. R., Patnaik, L. M., Nagaraj, H. C., Hamsavath, P. N., and Nalini, N., editors, *Emerging Research in Computing, Information, Communication and Applications*, pages 265–271, Singapore. Springer Singapore.

- [110] Rahman, M. M., Kutlu, M., Elsayed, T., and Lease, M. (2020). Efficient test collection construction via active learning. In *Proceedings of the 2020 ACM SIGIR on International Conference on Theory of Information Retrieval, ICTIR '20*, page 177–184, New York, NY, USA. Association for Computing Machinery.
- [111] Rukzio, E., Paolucci, M., Wagner, M., Berndt, H., Hamard, J., and Schmidt, A. (2006). Mobile service interaction with the web of things. *13th International Conference on Telecommunications (ICT 2006), Funchal, Madeira island, Portugal, 2006c*, (-).
- [112] Römer, K., Ostermaier, B., Mattern, F., Fahrmaier, M., and Kellerer, W. (2010). Real-time search for real-world entities: A survey. *Proceedings of the IEEE*, 98(11):1887–1902.
- [113] Sciullo, L., Aguzzi, C., Di Felice, M., and Cinotti, T. S. (2019). Wot store: Enabling things and applications discovery for the w3c web of things. In *2019 16th IEEE Annual Consumer Communications Networking Conference (CCNC)*, pages 1–8, New York, NY, USA. IEEE.
- [114] Sciullo, L., Gigli, L., Montori, F., Trotta, A., and Felice, M. D. (2022). A survey on the web of things. *IEEE Access*, 10:47570–47596.
- [115] Sciullo, L., Montori, F., Trotta, A., Di Felice, M., and Salmon Cinotti, T. (2020). Discovering web things as services within the arrowhead framework. In *2020 IEEE Conference on Industrial Cyberphysical Systems (ICPS)*, volume 1, pages 571–576, New York, NY, USA. IEEE.
- [116] Seif Eddine, M. and Rodin, V. (2022). Wot search engine based on multi agent system: A conceptual framework. *International Journal of Interactive Mobile Technologies (iJIM)*, 16(05):pp. 49–61.
- [117] Shemshadi, A., Sheng, Q. Z., and Qin, Y. (2016). Thingseek: A crawler and search engine for the internet of things. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '16*, page 1149–1152, New York, NY, USA. Association for Computing Machinery.
- [118] Shemshadi, A., Sheng, Q. Z., Qin, Y., Sun, A., Zhang, W. E., and Yao, L. (2017). Searching for the internet of things: Where it is and what it looks like. *Personal Ubiquitous Comput.*, 21(6):1097–1112.
- [119] Silva, A. L. M., Pérez-Alcázar, J. d. J., and Kofuji, S. T. (2019). Interoperability in semantic web of things: Design issues and solutions. *International Journal of Communication Systems*, 32(6):e3911. e3911 dac.3911.
- [120] Skarmeta, A. F., Santa, J., Martínez, J. A., Parreira, J. X., Barnaghi, P., Enshaeifar, S., Beliatis, M. J., Presser, M. A., Iggena, T., Fischer, M., Tönjes, R., Strohbach, M., Sforzin, A., and Truong, H. (2018). Iotcrawler: Browsing the internet of things. In *2018 Global Internet of Things Summit (GloTS)*, pages 1–6, New York, NY, USA. IEEE.
- [121] Soldatos, J., Kefalakis, N., Hauswirth, M., Serrano, M., Calbimonte, J.-P., Riahi, M., Aberer, K., Jayaraman, P. P., Zaslavsky, A., Žarko, I. P., Skorin-Kapov, L., and Herzog, R. (2015). Openiot: Open source internet-of-things in the cloud. In Podnar Žarko, I., Pripužić, K., and Serrano, M., editors, *Interoperability and Open-Source Solutions for the Internet of Things*, pages 13–25, Cham. Springer International Publishing.

- [122] Son, Y.-H. and Lee, K.-C. (2018). Cloud of things based on linked data. In *2018 International Conference on Information Networking (ICOIN)*, pages 447–449.
- [123] Su, L. T. (2003). A comprehensive and systematic model of user evaluation of web search engines: I. theory and background. *Journal of the American society for information science and technology*, 54(13):1175–1192.
- [124] Tan, C. C., Sheng, B., Wang, H., and Li, Q. (2010). Microsearch: A search engine for embedded devices used in pervasive computing. *ACM Trans. Embed. Comput. Syst.*, 9(4).
- [125] Tang, J., Xue, X., Yangui, S., and Zhou, Z. (2020a). Efficient search for moving object devices in internet of things networks. In *2020 IEEE International Conference on Web Services (ICWS)*, pages 454–462, New York, NY, USA. IEEE.
- [126] Tang, J. and Zhou, Z. (2019). Searching the internet of things using coding enabled index technology. In Li, S., editor, *Green, Pervasive, and Cloud Computing*, pages 79–91, Cham. Springer International Publishing.
- [127] Tang, J., Zhou, Z., Shu, L., and Hancke, G. (2019). Smpkr: Search engine for internet of things. *IEEE Access*, 7:163615–163625.
- [128] Tang, J., Zhou, Z., Xue, X., and Wang, G. (2020b). Using collaborative edge-cloud cache for search in internet of things. *IEEE Internet of Things Journal*, 7(2):922–936.
- [129] Tang, S., Du, X., Lu, Z., Gai, K., Wu, J., Hung, P. C., and Choo, K.-K. R. (2022). Coordinate-based efficient indexing mechanism for intelligent iot systems in heterogeneous edge computing. *Journal of Parallel and Distributed Computing*, 166:45–56.
- [130] Tran, N. K. (2018). *A Reference Architecture and a Software Platform for Engineering Internet of Things Search Engines*. PhD thesis, School of Computer Science, The University of Adelaide, Adelaide, Australia.
- [131] Tran, N. K., Babar, M. A., Sheng, Q. Z., and Grundy, J. (2019). A framework for internet of things search engines engineering. In *2019 26th Asia-Pacific Software Engineering Conference (APSEC)*, pages 228–235, New York, NY, USA. IEEE.
- [132] Tran, N. K., Sheng, Q. Z., Babar, M. A., and Yao, L. (2017). Searching the web of things: State of the art, challenges, and solutions. *ACM Comput. Surv.*, 50(4).
- [133] Tran, N. K., Sheng, Q. Z., Babar, M. A., Yao, L., Zhang, W. E., and Dustdar, S. (2018). Internet of things search engine: Concepts, classification, and open issues. *CoRR*, abs/1812.02930.
- [134] Trifa, V. (2011). *Building Blocks for a Participatory Web of Things: Devices, Infrastructures, and Programming Frameworks*. PhD thesis, ETH Zurich, Zurich, Switzerland.
- [135] Trotman, A. and Sigurbjörnsson, B. (2005). Narrowed extended xpath i (nexi). In Fuhr, N., Lalmas, M., Malik, S., and Szilávik, Z., editors, *Advances in XML Information Retrieval*, pages 16–40, Berlin, Heidelberg. Springer Berlin Heidelberg.
- [136] Trotman, A. and Zhang, J. (2013). Future web growth and its consequences for web search architectures. *arXiv preprint arXiv:1307.1179*.

- [137] Truong, C., Römer, K., and Chen, K. (2012). Fuzzy-based sensor search in the web of things. In *2012 3rd IEEE International Conference on the Internet of Things*, pages 127–134, New York, NY, USA. IEEE.
- [138] van Zwol, R. (2006). B 3-sdr and effective use of structural hints. In *Advances in XML Information Retrieval and Evaluation: 4th International Workshop of the Initiative for the Evaluation of XML Retrieval, INEX 2005, Dagstuhl Castle, Germany, November 28-30, 2005. Revised Selected Papers 4*, pages 146–160. Springer.
- [139] Vlad Trifa, Dominique Guinard, D. C. (2022). Web thing model.
- [140] Wang, H., Tan, C. C., and Li, Q. (2010). Snoogle: A search engine for pervasive environments. *IEEE Transactions on Parallel and Distributed Systems*, 21(8):1188–1202.
- [141] Xu, D. and Ruan, C. (2022). Modern theoretical tools for understanding and designing next-generation information retrieval system. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, pages 1635–1637, New York, NY, USA. Association for Computing Machinery.
- [142] Yang, P., Kang, X., Wu, Q., Yang, B., and Zhang, P. (2020). Participant selection strategy with privacy protection for internet of things search. *IEEE Access*, 8:40966–40976.
- [143] Yang, X., Song, W., and De, D. (2011). Live web: A sensorweb portal for sensing the world in real-time. *Tsinghua Science and Technology*, 16(5):491–504.
- [144] Yap, K.-K., Srinivasan, V., and Motani, M. (2005). Max: Human-centric search of the physical world. In *Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems, SenSys '05*, page 166–179, New York, NY, USA. Association for Computing Machinery.
- [145] Younan, M., Khattab, S., and Bahgat, R. (2016). Wotsf: A framework for searching in the web of things. In *Proceedings of the 10th International Conference on Informatics and Systems, INFOS '16*, page 278–285, New York, NY, USA. Association for Computing Machinery.
- [146] Zhang, D., Yang, L. T., and Huang, H. (2011). Searching in internet of things: Vision and challenges. In *2011 IEEE Ninth International Symposium on Parallel and Distributed Processing with Applications*, pages 201–206, NY, USA. IEEE.
- [147] Zhang, N. (2021). Service Discovery and Selection Based on Dynamic QoS in the Internet of Things. *Complexity*, 2021:1–12.
- [148] Zhang, P., Li, X., Liu, Y., Kang, X., and Liu, Y. (2019). Sdu: State-based dual-mode sensor search mechanism toward internet of things. *IEEE Access*, 7:147962–147974.
- [149] Zhang, W. E., Sheng, Q. Z., Mahmood, A., Tran, D. H., Zaib, M., Hamad, S. A., Aljubairy, A., Alhazmi, A. A. F., Sagar, S., and Ma, C. (2020). The 10 research topics in the internet of things. In *2020 IEEE 6th International Conference on Collaboration and Internet Computing (CIC)*, volume 1, pages 34–43, New York, NY, USA. IEEE.

- 
- [150] Zhang, Y., Liu, X., and Zhai, C. (2017). Information retrieval evaluation as search simulation: A general formal framework for ir evaluation. In *Proceedings of the ACM SIGIR International Conference on Theory of Information Retrieval, ICTIR '17*, page 193–200, New York, NY, USA. Association for Computing Machinery.
- [151] Zhong, N., Ma, J., Liu, J., Huang, R., and Tao, X. (2018). *Wisdom Web of Things*. Springer Publishing Company, Incorporated, 1st edition.
- [152] Zhou, Y., De, S., Wang, W., and Moessner, K. (2016). Search techniques for the web of things: A taxonomy and survey. *Sensors*, 16(5):600.





# Apéndice A

## Publicaciones

### Artículos en Revistas Indexadas

- Cristyan Manta-Caro, Juan M. Fernández-Luna: Modeling and Simulating the Web of Things from an Information Retrieval Perspective. *ACM Trans. Web* 12(1): 6:1-6:27 (2018)  
<https://doi.org/10.1145/3132732>
- Cristyan Manta-Caro, Annalina Caputo, Juan M. Fernández-Luna: Information Retrieval for IoT and WoT: State-of-the-Art, Taxonomy Framework and Evolutionary Directions a someter a revista especializada en Recuperación de Información. Artículo elaborado en colaboración con la Dublin City University.

### Artículos en Conferencias

- Cristyan Manta-Caro, Juan M. Fernández-Luna, Wilmar Jaimes Fernández: Cybersecurity as Information Retrieval Dimension for Cloud-based Edge-powered IoT Search. *IEEE Latin-America Conference on Communications LATINCOM*, Santo Domingo, Dominican Republic, (2020): 1-6  
<https://doi.org/10.1109/LATINCOM50620.2020.9282336>
- Cristyan Manta-Caro, Juan M. Fernández-Luna: Information-Retrieval-as-a-Service for the Web of Things: A Survey and a Proposal of IRaaS Architecture. *IEEE 4th International Conference on Future Internet of Things and Cloud (FiCloud)*, Vienna, Austria, (2016): 325-333  
<https://doi.org/10.1109/FiCloud.2016.53>

- 
- Cristyan Manta-Caro, Juan M. Fernández-Luna: Advances in real-time indexing models and techniques for the web of things. IEEE Latin-America Conference on Communications LATINCOM, Medellin, Colombia, (2016): 1-6  
<https://doi.org/10.1109/LATINCOM.2016.7811572>
  
  - Cristyan Manta-Caro, Juan M. Fernández-Luna: A discrete-event simulator for the web of things from an information retrieval perspective. IEEE Latin-America Conference on Communications LATINCOM, Cartagena de Indias, (2014): 1-6.  
<https://doi.org/10.1109/LATINCOM.2014.7041851>

# Apéndice B

## Guía de Usuario para la Evaluación del Sistema IR.WoT

### B.1. Información acerca del Proyecto de Investigación

El propósito de esta ficha de consentimiento es proveer a los participantes en esta investigación con una clara explicación de la naturaleza de la misma, así como de su rol en ella como participantes. La presente investigación científica es conducida por Cristyan Manta-Caro, de la Universidad de Granada, España. La meta del estudio es evaluar un sistema de recuperación para la Web de las Cosas denominado, (IR.WoT) desde la perspectiva de usuario final. Este sistema en la forma de un motor de búsqueda publicado como un servicio Web en (<https://iro-wot-ugr.ue.r.appspot.com/>).

Si Usted accede a participar en este estudio de investigación, se le pedirá, sí es una primera sesión, seguir una serie de tareas de búsqueda empleando el sistema IR.WoT y completar una encuesta en línea en <https://forms.office.com/r/4XdYWuCD6g>. Posteriormente, en una segunda sesión se le pedirá responder preguntas en una entrevista acerca de la experiencia en el uso del sistema. Estas sesiones tomarán aproximadamente 90 minutos de su tiempo en total. Las respuestas a la segunda sesión se registrarán en un formulario en línea.

La participación en este estudio es estrictamente voluntaria. La información que se recoja será confidencial y no se usará para ningún otro propósito fuera de los de esta investigación. Sus respuestas al cuestionario y a la entrevista serán codificadas usando un número de participante y por lo tanto, serán anónimas. Desde ya le agradecemos su participación.

## B.2. Instrucciones y Guía de Usuario para la Búsqueda en la Web de las Cosas

1. Ingrese al sitio Web: <https://ir-wot-ugr.ue.r.appspot.com/> El sistema IR.WoT le pedirá que comparta su ubicación actual (ver Figura B.1), por lo que los resultados de la búsqueda se muestran centrados en su ubicación. De lo contrario, los resultados de la búsqueda se centrarán asumiendo que Granada, España es su ubicación geográfica actual.
2. Explore la barra de navegación (panel izquierdo) que contiene enlaces acerca de esta investigación científica e información acerca del sistema de recuperación de información. En los submenús se encuentran a) un enlace para retorno al /Home o interfaz de búsqueda, b) un resumen de las instrucciones de uso del buscador, c) información sobre la arquitectura general del sistema IR.WoT y d) algunos ajustes avanzados del funcionamiento interno del motor de búsqueda. Este menú puede ser replegado para maximizar el espacio de la interfaz de búsqueda.
3. El sistema IR.WoT, así como un simulador de eventos discretos para la Web de las Cosas SIM.WoT, que genera datos sintéticos en formato XML que alimenta los escenarios simulados en los cuales se buscará se encuentran publicados como código abierto bajo licencia GNU *General Public License v3.0* en el enlace indicado por el icono de GitHub en la esquina superior derecha.
4. Las opciones de búsqueda se pueden clasificar en dos grandes bloques:
  - CO Content-Only: Búsquedas de solo contenido, para esto introduzca sus necesidades de información dentro del cuadro de texto principal (*¿Search for Things?*).
  - CAS Content and Structure: Para consultas de estructura y contenido también conocidas como consultas CAS, defina opciones de filtrado avanzadas según sea necesario para contextos espaciales y temporales y restricciones de estructura sobre propiedades, acciones o eventos de cosas particulares. A continuación se mostrarán algunos ejemplos de búsqueda y se describirán brevemente los escenarios simulados sobre los cuales se pueden realizar consultas de búsqueda que se emplearon con un conjunto de usuarios finales.

A continuación se mostrarán algunos ejemplos de búsqueda y se describirán brevemente los escenarios simulados sobre los cuales se pueden realizar consultas de búsqueda que se emplearon con un conjunto de usuarios finales.

5. De modo de proveer contexto en los escenarios simulados sobre los cuales se realizarán las consultas de información. A continuación, se describirá de manera resumida el modelo de Web de las Cosas empleados, el cual tiene como inspiración las recomendaciones técnicas de W3C y el modelo de descripción de cosas (TD). El modelo de descripción de cosas TD W3C consiste en metadatos semánticos para la cosa misma, un modelo de interacción que reúne propiedades, acciones y eventos de las cosas, con un esquema semántico para hacer que los modelos de datos sean comprensibles por máquinas y características para vinculación Web dotando de la capacidad de expresar relaciones entre cosas. El modelo TD W3C provee un mecanismo formal para describir las interfaces provistas por la infraestructura IoT y sus servicios, independiente de la implementación de protocolo.
6. Se han definido en la simulación tres (3) zonas inteligentes: España (ES), Colombia (CO) y Reino unido (UK) empleando información en la base de datos geográficos Geonames Además, pertenecientes a estas zonas se han definido (4) espacios inteligentes con base en las ciudades de Granada, Barcelona, Bogotá y Londres. Usando la información de GeoNames se han definido como subespacios inteligentes lugares alrededor de las ciudades.
7. En las ciudades se han definido cosas virtuales representando aparcamientos públicos y ambulancias de la ciudad, en los que se expone el estado general de ocupación como una propiedad. Generados de forma sintética por medio del simulador SIM.WoT.
8. Adicionalmente se han definido sensores virtuales de tres (3) tipos representando variables ambientales de interés: concentración de CO<sub>2</sub>, temperatura y calidad del aire. Generados de forma sintética por medio del simulador SIM.WoT. Estos actualizan sus mediciones estocásticamente con tiempos entre medidas de 3 a 10 min.
9. Realice la búsqueda de las zonas, espacios, subespacios inteligentes y de cosas y sensores virtuales usando una consulta CO y variaciones de consultas CAS, ver Figuras B.2, B.5 y B.7.
10. Manipule los filtros CAS de entidades, espaciales y temporales a su conveniencia. Así como los filtros de contexto propiedades, acciones y eventos en forma textual (*el filtrado de rango numérico es una característica contemplada para futuras versiones*).
11. Verifique el(los) resultado(s) de la búsqueda en la lista y en el mapa. Algunos ejemplos de respuestas se ilustran en las Figuras B.3, B.6 y B.8.

12. Verifique el contenido de los enlaces de resultados. Para esto emplee en lo posible una herramienta alternativa al navegador que mantenga la estructura de documentos XML, como ejemplo Postman, ver Figura B.4. De lo contrario, el navegador empleara una conversión XML vía XSLT para renderizar el contenido obviando las etiquetas XML.

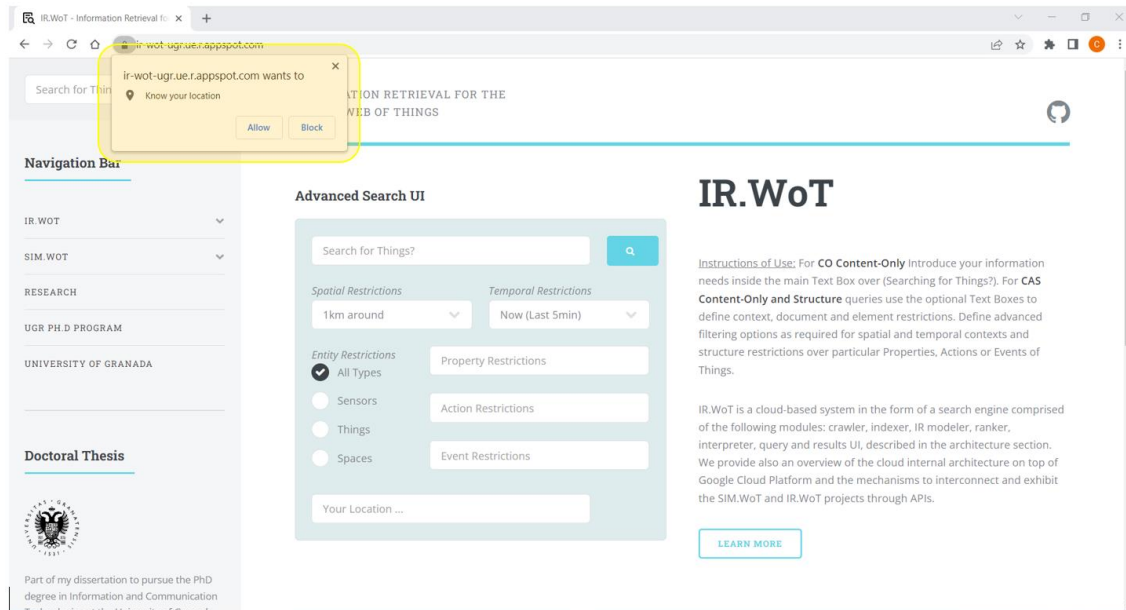


Figura B.1 IR.WoT: Confirmación de Opciones de Ubicación de Usuario.

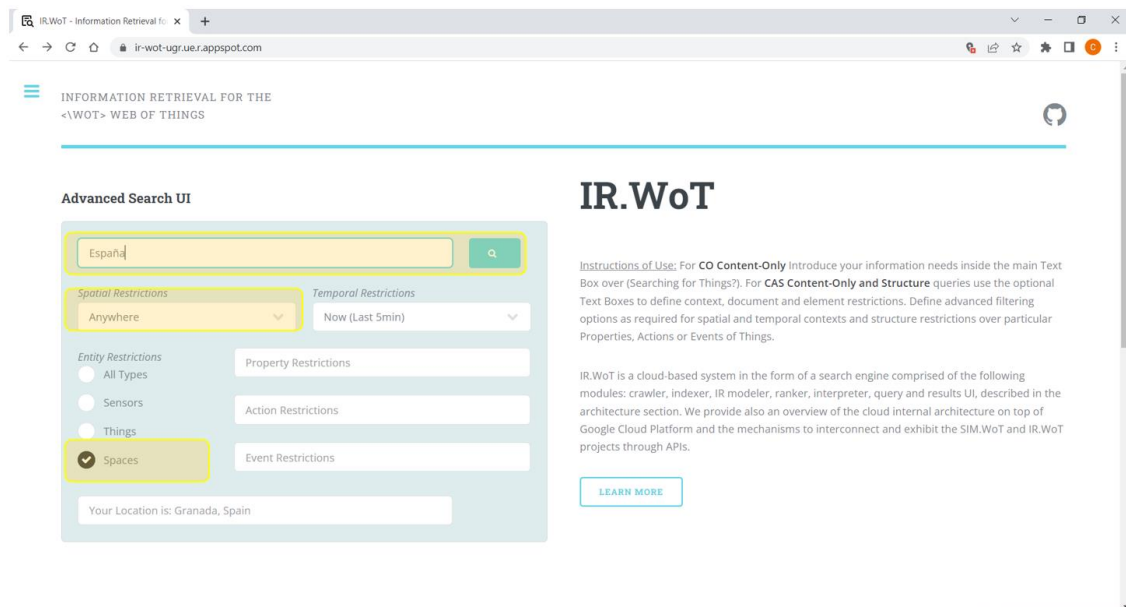


Figura B.2 IR.WoT: Consulta con Restricción Entidad y Espacial.

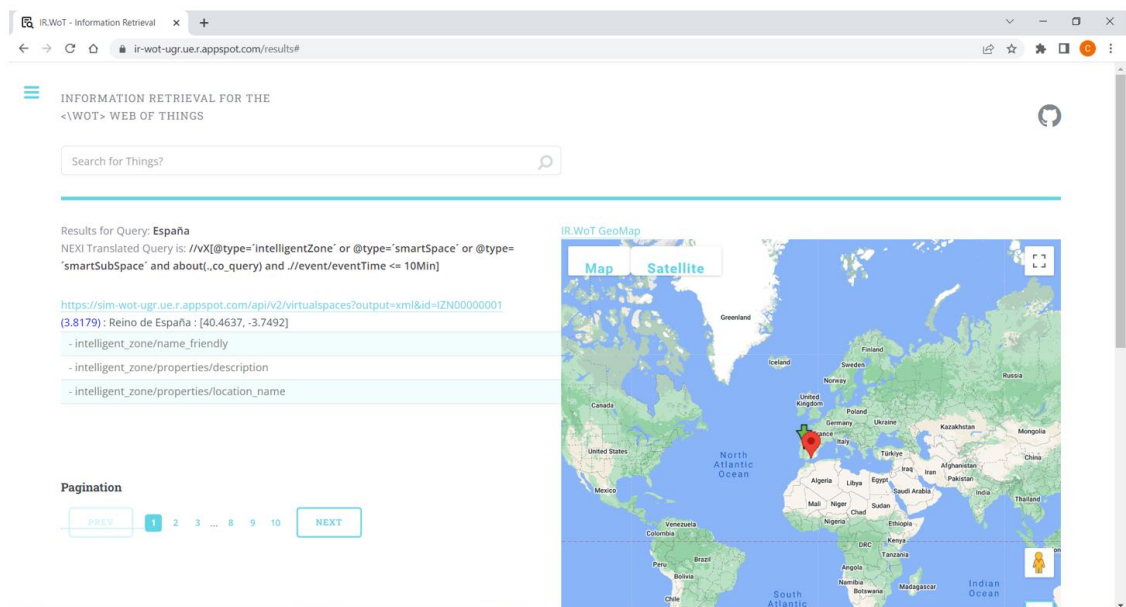
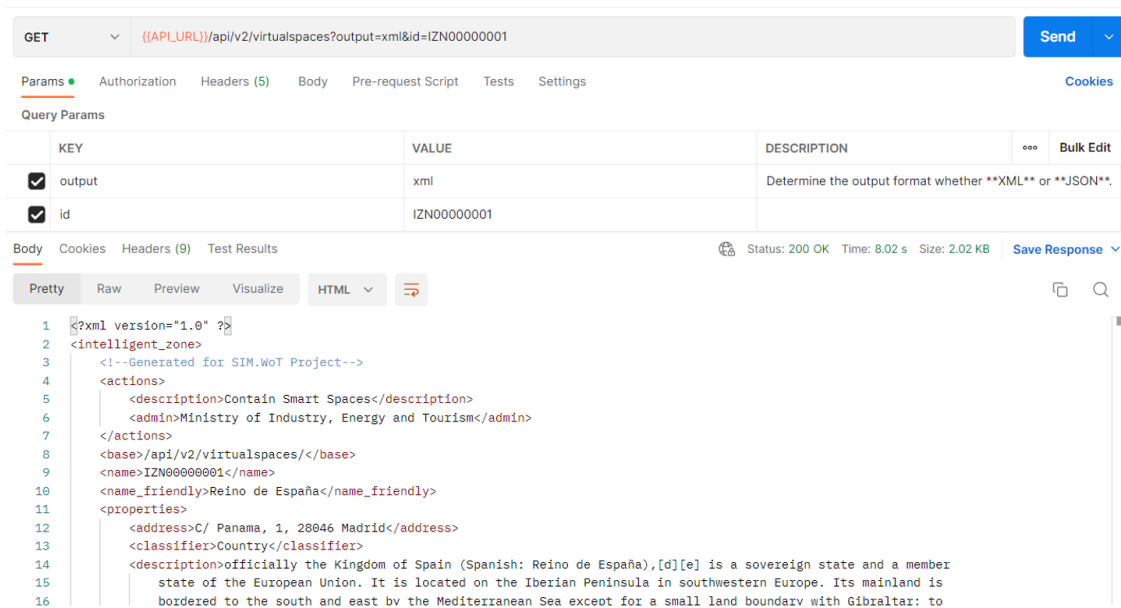


Figura B.3 IR.WoT: Respuesta a consulta con Restricción Entidad y Espacial.





GET `{{(API_URL)}}/api/v2/virtualspaces?output=xml&id=IZN00000001` Send

Params Authorization Headers (5) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	***	Bulk Edit
<input checked="" type="checkbox"/> output	xml	Determine the output format whether <b>**XML**</b> or <b>**JSON**</b> .		
<input checked="" type="checkbox"/> id	IZN00000001			

Body Cookies Headers (9) Test Results Status: 200 OK Time: 8.02 s Size: 2.02 KB Save Response

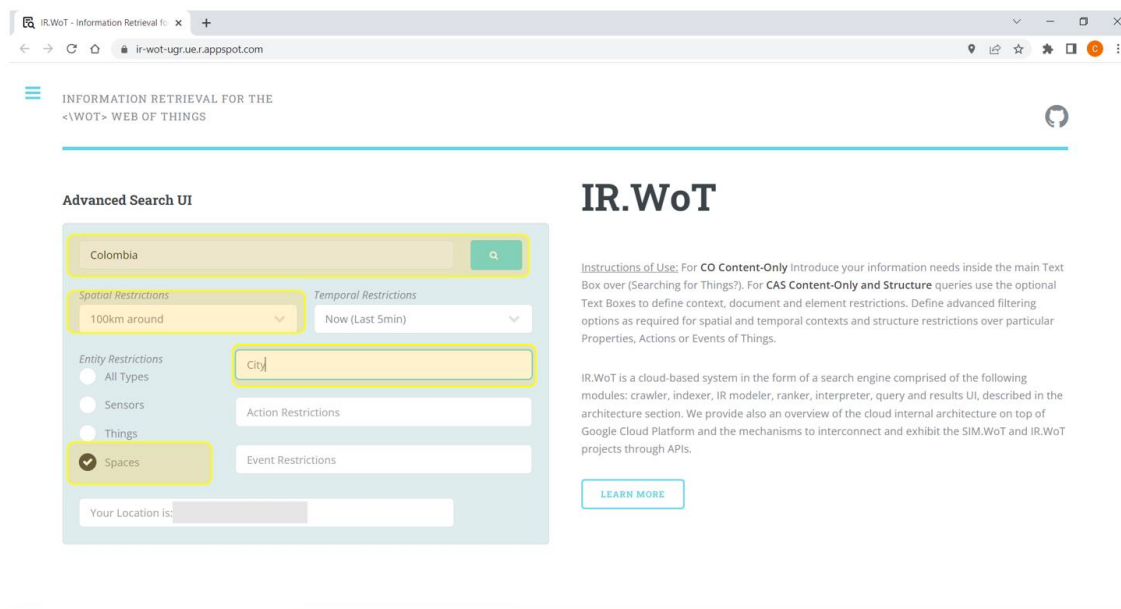
Pretty Raw Preview Visualize HTML

```

1 <?xml version="1.0" ?>
2 <intelligent_zone>
3   <!--Generated for SIM.WoT Project-->
4   <actions>
5     <description>Contain Smart Spaces</description>
6     <admin>Ministry of Industry, Energy and Tourism</admin>
7   </actions>
8   <base>/api/v2/virtualspaces/</base>
9   <name>IZN00000001</name>
10  <name_friendly>Reino de España</name_friendly>
11  <properties>
12    <address>C/ Panama, 1, 28046 Madrid</address>
13    <classifier>Country</classifier>
14    <description>officially the Kingdom of Spain (Spanish: Reino de España),[d][e] is a sovereign state and a member
15    state of the European Union. It is located on the Iberian Peninsula in southwestern Europe. Its mainland is
16    bordered to the south and east by the Mediterranean Sea except for a small land boundary with Gibraltar: to

```

Figura B.4 IR.WoT: Verificación vía Postman de Respuesta.



IR.WoT - Information Retrieval for the <WOT> Web of Things

Advanced Search UI

Colombia

Spatial Restrictions: 100km around

Temporal Restrictions: Now (Last 5min)

Entity Restrictions:

- All Types
- Sensors
- Things
- Spaces

City

Action Restrictions

Event Restrictions

Your Location is:

## IR.WoT

**Instructions of Use:** For **CO Content-Only** introduce your information needs inside the main Text Box over (Searching for Things?). For **CAS Content-Only and Structure** queries use the optional Text Boxes to define context, document and element restrictions. Define advanced filtering options as required for spatial and temporal contexts and structure restrictions over particular Properties, Actions or Events of Things.

IR.WoT is a cloud-based system in the form of a search engine comprised of the following modules: crawler, indexer, IR modeler, ranker, interpreter, query and results UI, described in the architecture section. We provide also an overview of the cloud internal architecture on top of Google Cloud Platform and the mechanisms to interconnect and exhibit the SIM.WoT and IR.WoT projects through APIs.

[LEARN MORE](#)

Figura B.5 IR.WoT: Consulta con Restricción Entidad, Espacial. Uso de Contexto de Propiedades.

INFORMATION RETRIEVAL FOR THE <WOT> WEB OF THINGS

Search for Things?

Results for Query: Colombia  
 NEXI Translated Query is: //vxl[@type="IntelligentZone" or @type="smartSpace" or @type="smartSubSpace" and about(.co\_query) and //property/geocoordinate <= 100Kms and //event/eventTime <= 10Min]//property[about(.cas\_query)]

<https://sim-wot-ugr.ue.r.appspot.com/api/v2/virtualspaces?output=xml&id=SSP00000003>  
 (5.3046) : Bogotá D.C. : [4.60971, -74.08175]

- smart\_space/properties/description
- smart\_space/properties/keyword

Pagination: 1 2 3 ... 8 9 10 NEXT

IR.WoT GeoMap

Map Satellite

Colombia

Figura B.6 IR.WoT: Respuesta a consulta con Restricción Entidad y Espacial. Uso de Contexto de Propiedades

INFORMATION RETRIEVAL FOR THE <WOT> WEB OF THINGS

Advanced Search UI

Parking

Spatial Restrictions: 10km around

Temporal Restrictions: Now (Last 5min)

Entity Restrictions: Things

Action Restrictions

Event Restrictions

Your Location is: Granada, Spain

## IR.WoT

Instructions of Use: For **CO Content-Only** Introduce your information needs inside the main Text Box over (Searching for Things?). For **CAS Content-Only and Structure** queries use the optional Text Boxes to define context, document and element restrictions. Define advanced filtering options as required for spatial and temporal contexts and structure restrictions over particular Properties, Actions or Events of Things.

IR.WoT is a cloud-based system in the form of a search engine comprised of the following modules: crawler, indexer, IR modeler, ranker, interpreter, query and results UI, described in the architecture section. We provide also an overview of the cloud internal architecture on top of Google Cloud Platform and the mechanisms to interconnect and exhibit the SIM.WoT and IR.WoT projects through APIs.

LEARN MORE

Figura B.7 IR.WoT: Consulta con Restricción Entidad, Espacial-Temporal. Uso de Contexto de Propiedades.

IR.WoT - Information Retrieval

INFORMATION RETRIEVAL FOR THE <WOT> WEB OF THINGS

Search for Things?

Results for Query: **Parking**  
Showing: **10 results of 54**  
NEXI Translated Query is: `//vX[@type='virtualThing' and about(.,co_query)] and //property/geocoordinate <= 10Km and //event/eventTime <= 10Min[/property[about(.,cas_query)]]`

<https://sim-wot-ugr.ue.r.appspot.com/api/v2/virtualthings?output=xml&id=VTH0000062>  
(16.4168) : Parking Slot 62 : [37.165403045943116, -3.541581460450677]  
- virtual\_thing/name\_friendly  
- virtual\_thing/properties/description  
- virtual\_thing/actions/description

<https://sim-wot-ugr.ue.r.appspot.com/api/v2/virtualthings?output=xml&id=VTH0000028>  
(16.4168) : Parking Slot 28 : [37.14026458804059, -3.6032503652323236]  
- virtual\_thing/name\_friendly  
- virtual\_thing/properties/description  
- virtual\_thing/actions/description

IR.WoT GeoMap

Figura B.8 IR.WoT: Respuesta a consulta con Restricción Entidad y Espacial-Temporal. Uso de Contexto de Propiedades

IR.WoT - Information Retrieval

INFORMATION RETRIEVAL FOR THE <WOT> WEB OF THINGS

**Advanced Search UI**

Oxygen

Spatial Restrictions: Here (100 mts) | Temporal Restrictions: Now (Last 5min)

Entity Restrictions:  Sensors,  Things,  Spaces

Property Restrictions: | Action Restrictions: | Sample

Your Location is: Granada, Spain

**IR.WoT**

Instructions of Use: For **CO Content-Only** introduce your information needs inside the main Text Box over (Searching for Things?). For **CAS Content-Only and Structure** queries use the optional Text Boxes to define context, document and element restrictions. Define advanced filtering options as required for spatial and temporal contexts and structure restrictions over particular Properties, Actions or Events of Things.

IR.WoT is a cloud-based system in the form of a search engine comprised of the following modules: crawler, indexer, IR modeler, ranker, interpreter, query and results UI, described in the architecture section. We provide also an overview of the cloud internal architecture on top of Google Cloud Platform and the mechanisms to interconnect and exhibit the SIM.WoT and IR.WoT projects through APIs.

[LEARN MORE](#)

Figura B.9 IR.WoT: Consulta con Restricción Entidad, Espacial-Temporal. Uso de Contexto de Eventos.