



**DIMENSION4™**

**DIMENSION4 Sensor Networking**

From version 3.5

Copyright © 2023, Ubisense Limited 2014 - 2023. All Rights Reserved. You may not reproduce this document in whole or in part without permission in writing from Ubisense at the following address:

Ubisense Limited  
St Andrew's House  
St Andrew's Road  
Cambridge CB4 1DL  
United Kingdom

Tel: +44 (0)1223 535170

WWW: <https://www.ubisense.com>

All contents of this document are subject to change without notice and do not represent a commitment on the part of Ubisense. Reasonable effort is made to ensure the accuracy of the information contained in the document. However, due to on-going product improvements and revisions, Ubisense and its subsidiaries do not warrant the accuracy of this information and cannot accept responsibility for errors or omissions that may be contained in this document.

Information in this document is provided in connection with Ubisense products. No license, express or implied to any intellectual property rights is granted by this document.

Ubisense encourages all users of its products to procure all necessary intellectual property licenses required to implement any concepts or applications and does not condone or encourage any intellectual property infringement and disclaims any responsibility related thereto. These intellectual property licenses may differ from country to country and it is the responsibility of those who develop the concepts or applications to be aware of and comply with different national license requirements.

UBISENSE®, the Ubisense motif, SmartSpace® and AngleID® are registered trademarks of Ubisense Ltd. DIMENSION4™ and UB-Tag™ are trademarks of Ubisense Ltd.

Windows® is a registered trademark of Microsoft Corporation in the United States and/or other countries. The other names of actual companies and products mentioned herein are the trademarks of their respective owners.

# Contents

---

<b>Introduction</b> .....	<b>1</b>
<b>Running sensor protocols and ports</b> .....	<b>2</b>
<b>Sensor boot sequence</b> .....	<b>4</b>
<b>nmap scan report</b> .....	<b>7</b>



# Introduction

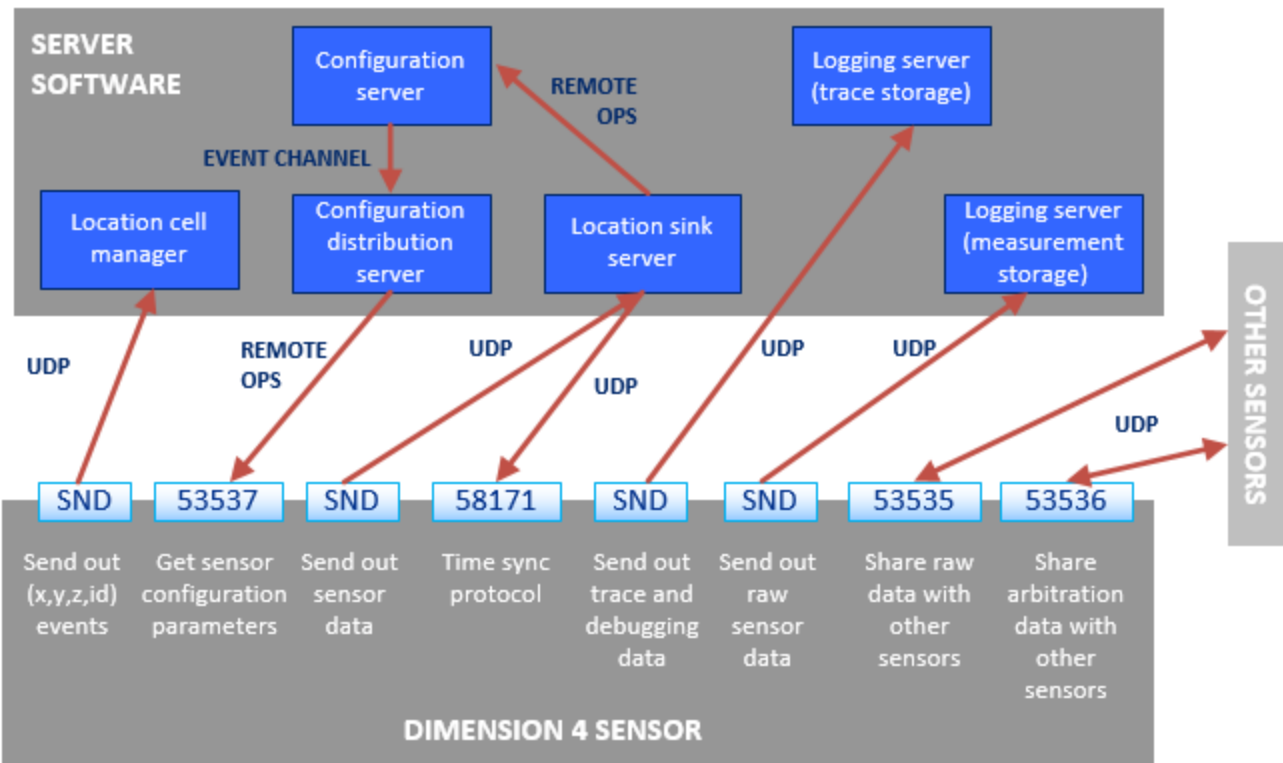
---

This document presents an overview of the DIMENSION4 sensors' use of the network and describes the interactions they have with the DIMENSION4 location platform services.

It describes the network behavior of the sensor when it is in its running state, and then goes on to describe the network behavior of the sensor when booting up. Finally it gives an example nmap scan of the sensor software (using a sensor running DIMENSION4 3.2 kernel/filesystem 1004/2412).

## Running sensor protocols and ports

The diagram below gives an overview of how a UbisenseDIMENSION4 sensor communicates with the DIMENSION4 (and SmartSpace) server software and with other DIMENSION4 sensors.



The light blue rectangles correspond to open ports on the sensor. Where the sensor is listening, the port number is labeled (these port numbers are valid at the time of this guide's publication and are subject to change in subsequent releases); where the sensor is sending the port is just labeled SND (it is bound using the standard bind(0) system call so the specific port will vary depending on execution).

Arrows indicate direction of communication, and the protocols used are labeled next to the arrows. The label UDP indicates raw UDP messages. The label REMOTE OPS indicates the Ubisense remote operations protocol and the label EVENT CHANNEL indicates the Ubisense event channel protocol. As described in the document *Ubisense Architecture: Schema Security*,

the remote operations and event channel protocols may be encrypted. It is therefore possible to configure DIMENSION4 such that sensor configuration is protected throughout the signal path from configuration tool to each sensor.

# Sensor boot sequence

---

The sensor boot sequence is described in detail in the documentation that ships with the DIMENSION4 location platform software. This document provides a brief overview to outline the implication for sensor network behavior.

1. Obtaining an IP address
  - a. The sensor loads the firmware from non-volatile memory.
  - b. The firmware reads the sensor's network configuration from non-volatile memory, and decides whether to obtain its IP using DHCP or to use statically-assigned IP settings.
  - c. The firmware performs DHCP if required.
  - d. The firmware sends ARP probes and announcements in accordance with RFC-5227. If the firmware detects an IP address conflict at any point (e.g. by receiving a conflicting ARP probe) then it will reboot. If a CNC is set for the sensor, it will stay in Pending validation state.
2. Firmware search for the Configuration Server
  - a. The firmware reads the search order from non-volatile memory. If the search order has not been configured or has been marked as invalid then a default search order is used.
  - b. For each search method, the firmware tries to contact the configuration server, performing needed DNS requests. If the firmware receives a valid response, it goes to the next step, marking the network configuration as valid.
3. Obtaining boot configuration and software files
  - a. The firmware reads the boot configuration server's address from the configuration server's response.
  - b. The firmware sends a request to the boot configuration server, which replies with the version of the software to run, and the address of a boot file server to obtain the software from. If the boot configuration server has the **boot** trace enabled, it will report **Sensor <MAC> requested configuration...** Currently the boot configuration server will always send its own address in response to the configuration request,



because the boot configuration server also serves as the single boot file server for the system.

- c. If the correct version of the software is already “flashed” into non-volatile memory, the firmware loads it.
- d. Otherwise the firmware requests the software from the boot file server. If the boot file server has the **boot** trace enabled, it will report messages of the form **Sensor <MAC> requested ...** and then **Sensor <MAC> completed request ...** if the download is successful.

#### 4. Starting the software and software networking

- a. The firmware runs the software, telling it whether to use DHCP or the existing static IP information.
- b. The software performs DHCP if requested and if so it also sends ARP probes and ARP announcements in accordance with RFC-5227.



If the sensor is using static IP, then no ARP probes or announcements are needed because the IP settings have not changed.

#### 5. Software configuration

- a. The software performs a search for the Configuration Server in the same way as the firmware does. It should not time out because it is repeating the task that succeeded when run from firmware
- b. The software reads the address of the location system configuration server from the Core Server’s response, and then registers its MAC address, IP address and the location system parameters it needs to run.
- c. The appropriate cellular location system configuration distribution server responds to the sensor with the required location system parameters, including the IP address of the cellular location sink service to send status and location information to.

#### 6. Firmware/software/network configuration

- a. The software checks if various diagnostic parameters have been written to the sensor’s non-volatile memory. If not, then the sensor calculates these, writes them to the non-volatile memory and reboots.

- b. The software checks if there are any pending firmware upgrades. If so, the sensor requests its boot configuration from boot configuration server and verifies that the requested version of firmware is available. The sensor then downloads the firmware from the boot file server, writes it to the non-volatile memory and reboots.
- c. The sensor performs the same checks as above with the software.
- d. The software checks if there are any pending changes to the sensor's network configuration. If so, the sensor writes the updated network configuration to the non-volatile memory and reboots.

7. Sensor hardware and timing initialization

- a. The software starts to send periodic status updates. At this stage its status appears as **Initializing**.
- b. The software initializes the rest of the hardware, and waits until it gets a stable timing signal, or until it is generating a stable timing signal if it is a timing source.
- c. The sensor's status is now **Running**. If the `sensor_init` trace is enabled, the sensor will report **Initialisation complete**.

Using this boot sequence it should be possible to ensure that there is appropriate networking provision for the sensors and that the sensors meet the relevant site-level standards for networked devices.

## nmap scan report

---

Here is the output of an nmap scan of a sensor running software kernel/filesystem 1004/2412 (from DIMENSION4 3.2). This demonstrates that there are no open TCP ports, and the scanner software was unable to uncover any identification details for the device operating system.

```
$ # Scan for any open TCP ports
$ nmap -A -p1-65535 10.42.5.50
```

```
Starting Nmap 7.40 ( https://nmap.org ) at 2017-07-21 13:33 ope
Nmap scan report for 10.42.5.50
Host is up (0.0010s latency).
All 65535 scanned ports on 10.42.5.50 are closed
MAC Address: 00:11:CE:D4:07:95 (Ubisense Limited)
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop
```

### TRACEROUTE

```
HOP RTT      ADDRESS
1   1.01 ms  10.42.5.50
```

OS and Service detection performed. Please report any incorrect results at <https://nmap.org/submit/> .

Nmap done: 1 IP address (1 host up) scanned in 33.27 seconds