Network connectivity diagram

Wire2waves DSC Coast Station

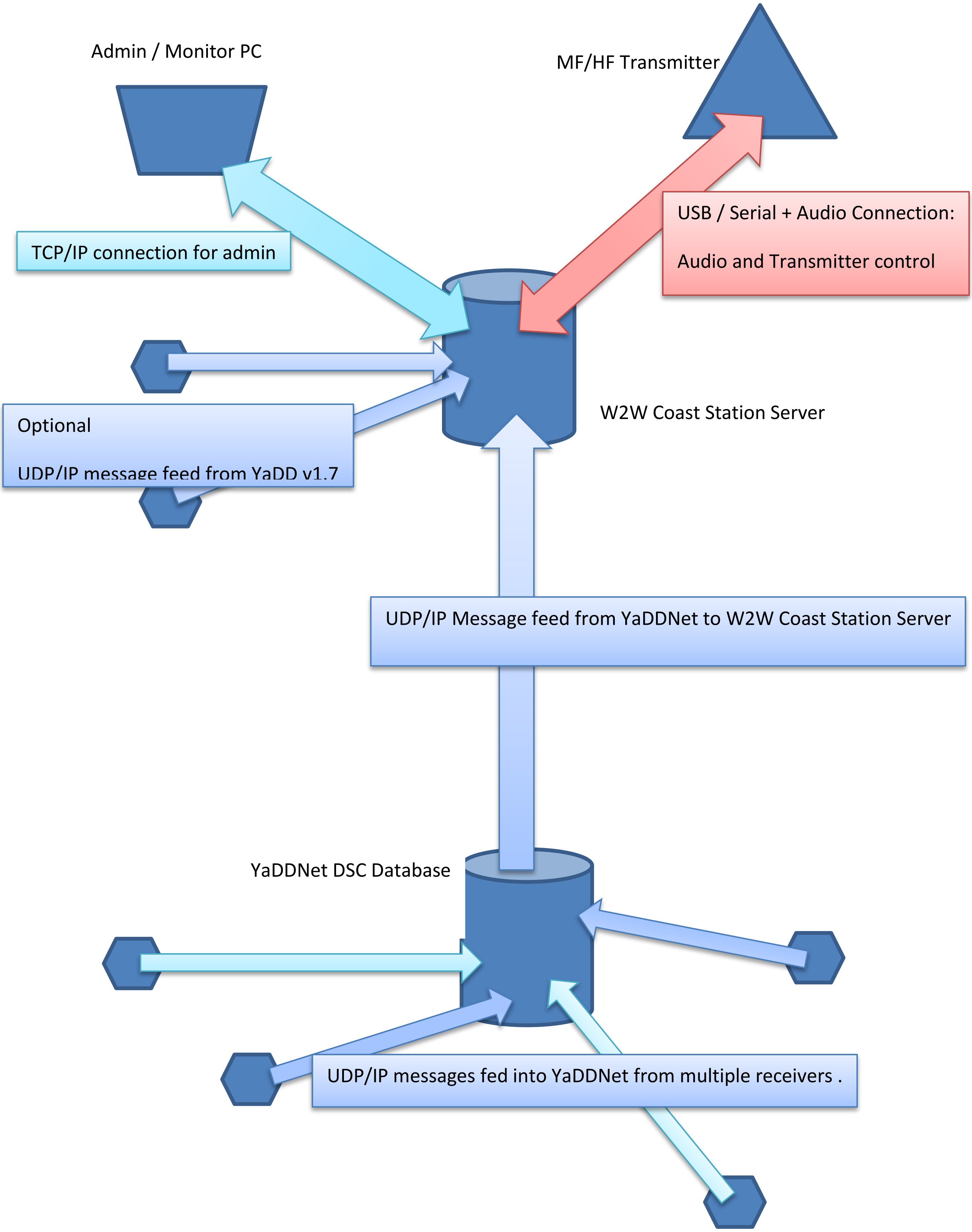
Seagate Radio, Dover (002320204)

Design overview

The system under development consists of the following components

- Coast Station Server software
- HF Transceiver and associated antenna
- Distributed network of DSC monitoring receivers
- Receive message collation server (YaDDNet)
- Coast Station administration client

Network connectivity diagram


Admin / Monitor PC
MF/HF Transmitter
TCP/IP connection for admin
USB / Serial + Audio Connection:
Audio and Transmitter control
Optional
UDP/IP message feed from YaDD v1.7
W2W Coast Station Server
UDP/IP Message feed from YaDDNet to W2W Coast Station Server
YaDDNet DSC Database
UDP/IP messages fed into YaDDNet from multiple receivers .

Network connectivity diagram

Server Software

The Server is: **w2w_coast_server_v2.py** and this uses the following additional local modules

- radio_functions.py
- dsc_functions.py
- resolve.py
- new_coast_dict.py
- coast_sql.py
- pass_verify.py

Other Python packages required are

- numpy
- pyaudio
- pyserial

Built in modules

- json
- hashlib
- random
- re
- datetime
- time
- SocketServer
- threading
- os
- sqlite3
- struct
- math

The Server provides TWO main functions:

- UDP/IP message interface, parsing and outgoing message generation, including Transmitter Control
- TCP/IP administration interface for monitoring and control

Network connectivity diagram

UDP/IP DSC Message Handling

In the standard implementation **w2w_coast_server** accepts a feed of DSC decoded messages from YaDDNet on UDP port 50669. These messages may or may not be pre-filtered at YaDDNet – to minimize bandwidth it is preferred to pre-filter to remove any messages not connected with Snargate from the outgoing UDP feed.

The **w2w_coast_server** parses each incoming DSC message fully, in order to

- discard messages from unwanted receiver sources (e.g. any outside of Europe)
- discard messages from unwanted DSC channel frequencies
- discard messages with CheckSum failures (ECC check error)
- discard messages with symbol parity errors
- discard messages with an incorrect number of MMSI digits

The messages that remain are deemed “acceptable” and are further inspected to

- identify those messages ***sent by*** Snargate Radio (MMSI 002320204) – for “Check Logging”
- identify those messages ***addressed to*** Snargate Radio 002320204 – for potential Test Calls

Messages ***addressed to*** Snargate Radio are then inspected to select only those that match

- Format = Selective Call, ***and***
- Telecommand 1 = Test, ***and***
- EOS = RQ (Request)

Messages that match all these criteria are then to be acted upon – a reply will be created and transmitted.

A reply is generated based on the MMSI of the calling station, with the following DSC symbols

- Format = Selective Call
- TO MMSI = MMSI of calling Station
- Category = Safety (or that of the originating message)
- FROM MMSI = 002320204
- Telecommand 1 = TEST
- Telecommand 2 = No Inf
- Data = No Inf
- EOS = BQ (Acknowledge)

The “reply” thus created is added to a “queue” – multiple requests can be received while a transmission is in progress. To manage the “contention” of access to a single transmitter each message must be queued and then transmitted in turn until the queue is empty. If an identical message is already in the queue then the new message will be discarded as a *“Duplicate“,* so as to ensure that only one reply is sent, despite multiple receivers reporting the same message. A duplicate is defined as a message with identical DSC Symbols and Transmit Frequency.

Network connectivity diagram

The DSC Messages arrive from YaDDNet in the following format:

```
Michael_AOR#1;16804.5;SEL;002320204;SAF;477534200;TEST;NOINF;--;--;REQ;OK
```

Each field is delimited (;) and the message contains

1. Receiver ID
2. Receiver Frequency
3. DSC Format
4. Addressee MMSI
5. DSC Category
6. Self-MMSI
7. Telecommand 1
8. Telecommand 2
9. Data (i.e. vessel position)
10. Data (i.e. requested frequency/channel)
11. EOS
12. ECC

The reported **Receiver Frequency** is used to control the Transmitter's frequency.

For the benefit of the user, and for logging, all numerical MMSIs are converted into Ship Name and Callsign via a "resolver" function which queries online resources (currently only www.aprs.fi) . Coast Station MMSIs are converted to Names using a local look-up table. All calls handled are logged in a SQLite database file locally.

Network connectivity diagram

TCP Administration Interface

For remote monitoring and control **w2w_coast_server** also has a TCP/IP network interface on port 50669 which allows multiple users to log in simultaneously using a graphical **w2w_client.** This gives remote access to all the server logging databases, allows remote shutdown of the server process, provides for the generation of outgoing DSC Test calls, provides a TX Disable/Enable function, allows individual DSC Frequencies to be Disabled/Enabled and allows individual Receiver IDs to be excluded from the system.

Users are provided with a username and password and all activity is logged with the username for auditing purposes. While the network traffic between Client and Server is not encrypted, the password is never sent in clear-text, but a SHA256 “salted/hashed” challenge/response protocol is used for authentication.

The **w2w_coast_server** is designed for unattended, automatic operation and user intervention is not normally required.