Client

Log

Api

TCP Manager

UDP Manager

Network

Radio

Dynamic Models

Static Models

UDP Stream (Radio —> Client)

Panadapter	**Waterfall**	**IqStream**
AudioStream	**Meter**	**MicAudioStream**

UDP Stream (Client —> Radio)

TxAudioStream

UDP Stream (Client <—> Radio)

Opus

Without UDP Stream

Slice	**Amplifier**	**Tnf**
Equalizer	**UsbCable**	
Memory	**Xvtr**	

Atu

Gps

Profile

Wan

Cwx

Interlock

Transmit

Waveform

Dynamic Model Objects

With UDP stream

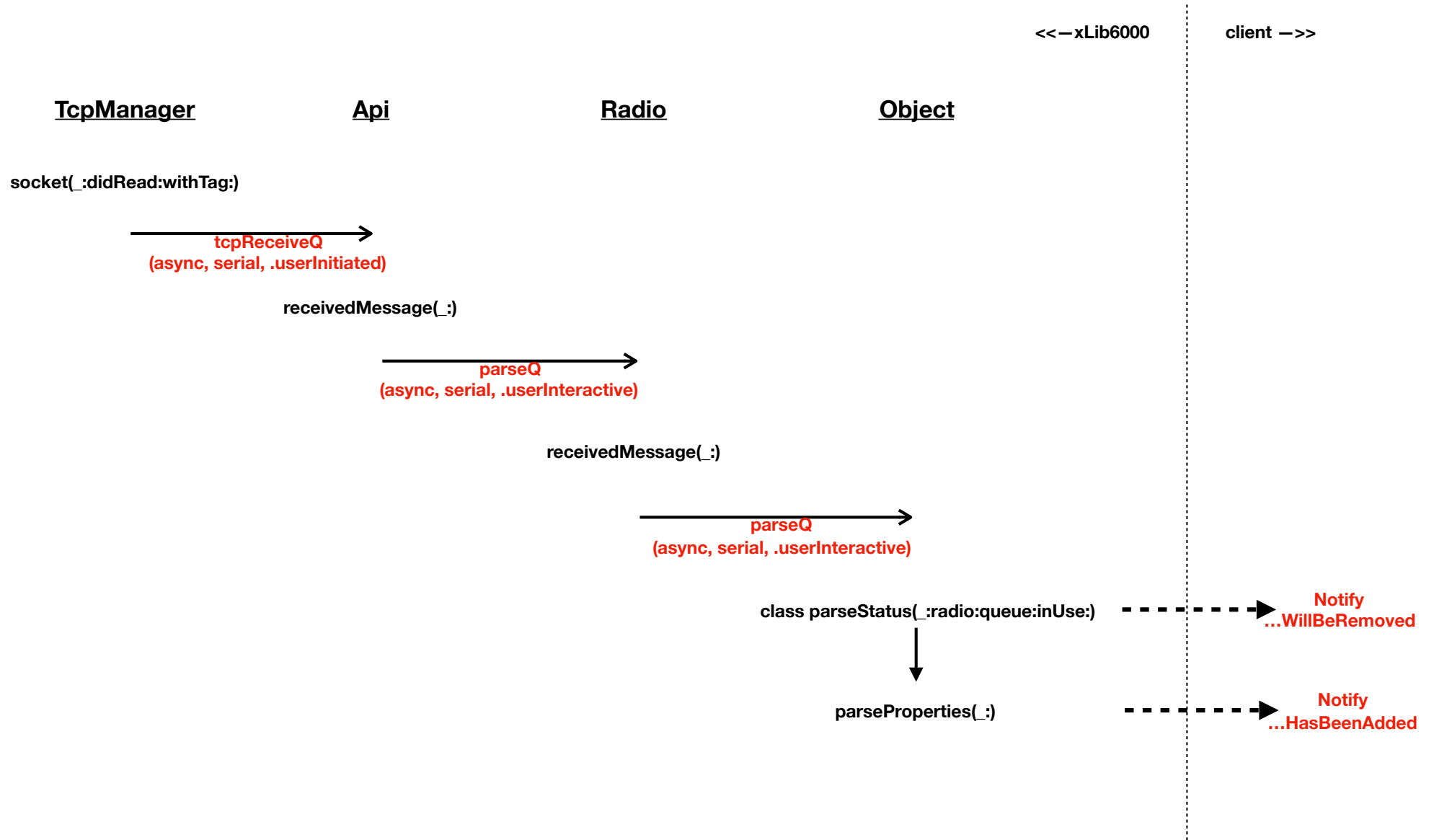| **Init(id: queue:)** |
| --- |
| Public properties:

    private(set) var id: |
| Custom methods:

    optional . . . |
| StatusParser protocol methods:

    class func parseStatus(_ keyValues: radio: queue: inUse:) |
| PropertiesParser protocol methods:

    func parseProperties(_ properties:) |
| VitaProcessor protocol methods:

    func vitaProcessor(_ vita:) |

Without UDP stream

Init(id: queue:)
Public properties: private(set) var id:
Custom methods: optional . . .
StatusParser protocol methods: class func parseStatus(_ keyValues: radio: queue: inUse:)
PropertiesParser protocol methods: func parseProperties(_ properties:)

Static Model Objects

Init(queue:)
Public properties: optional . . .
Custom methods: optional . . .
PropertiesParser protocol methods: func parseProperties(_ properties:)

TCP Stream path

TcpManager   **Api**   **Radio**   **Object**

socket(_:didRead:withTag:)

→ **tcpReceiveQ**
(async, serial, .userInitiated)

receivedMessage(_:)

→ **parseQ**
(async, serial, .userInteractive)

receivedMessage(_:)

→ **parseQ**
(async, serial, .userInteractive)

class parseStatus(_:radio:queue:inUse:) ⇢ **Notify**
...WillBeRemoved

↓

parseProperties(_:) ⇢ **Notify**
...HasBeenAdded

UDP Stream path

<<—xLib6000   client —>>

UdpManager   **Radio**   **Dynamic Model w/UDP**   **Client component**

udpSocket(_:didReceive: …)

udpReceiveQ
(async, serial, .userInitiated)

vitaParser(_:)

streamQ
(async, serial, .userInteractive)

vitaProcessor(_:)

streamQ
(async, serial, .userInteractive)

streamHandler(_:)