

Interview

XLX Multiprotocol Reflector Gateway für D-Star

Die Vielfalt moderner Digital Voice Betriebsarten – kurz DV – im Amateurfunkdienst hat in den vergangenen Jahren zugenommen. Hinter ihnen stehen Datennetze, die meist nur die Kommunikation im jeweiligen Standard erlauben. Luc Engelmann, LX1IQ, arbeitet mit seinem Team an einem System, damit alle Funkamateure ungeachtet des DV-Standards miteinander kommunizieren können.

Wie lange hat die Entwicklung des XLX-Systems bisher gedauert?

Erste Überlegungen haben wir im Oktober 2015 angestellt, der XLX Multiprotocol Gateway Reflektor ist somit noch recht jung. Die ersten Zeilen Code entstanden bereits Anfang November. Am 7. des Monats ging die erste Beta-Version online, damals aber noch ohne DCS-Protokoll-Unterstützung.

Wie viele Entwickler waren an der Entwicklung beteiligt?

Wir sind eine Gruppe von ambitionierten Funkamateuren, die Spaß an der digitalen Kommunikation haben. Aus diesen Leuten setzt sich die „Digital Voice Communications Working Group“ des RLs (Radio Amateurs du Luxembourg) zusammen (www.rlx.lu). Jeder hat auf seine Art und im Rahmen seiner Möglichkeiten zum Projekt beigetragen. Rein programmiertechnisch sind zwei Personen eingebunden.

Was war das ursprüngliche Ziel/die Motivation, ein eigenes System zu entwickeln?

Nachdem Luxemburg nun eine quasi vollständige D-Star-Abdeckung hat, war es die logische Konsequenz, unsere Relais untereinander und auch weltweit zu vernetzen. Keine der bestehenden Systeme decken jedoch unsere Bedürfnisse momentan ab. Diese Systeme spalten die Funkamateure in verschiedenen Lager wie D-Star, DMR+, DMR-MARC, C4FM, um nur einige zu nennen. Wir sind aber der Meinung, dass im Zeitalter der digitalen Kommunikation die verschieden Betriebsarten zusammen geführt werden sollten.

Sehr schnell merkten wir aber, dass die Entwickler der bestehenden Systeme kein Interesse an uns sowie unseren Ideen hatten, was uns dazu bewegte, unser eigenes System zu entwickeln.

Welche Protokolle beherrscht XLX aktuell?

Der XLX beherrscht aktuell alle gängigen D-Star-Protokolle wie DCS, DExtra und DPlus.

Welche Betriebs-Modi (D-Star, DMR, APCO ...) werden aktuell unterstützt, welche sind in Planung?

Wenn man über die verschiedenen Betriebsarten spricht, muss man erst die Philosophie verstehen, die hinter dem XLX steckt. Es wäre ein Leichtes, jetzt schon DMR-Relais an den XLX zu koppeln. Das Internet Protokoll sowie den Verbindungsmechanismus beherrschen wir bereits. Die Idee, die hinter dem XLX steckt, ist jedoch eine ganz andere. Sämtliche digitale Betriebsarten sollen auf dem XLX zusammen geführt werden. Nur so kann der Benutzer frei entscheiden, ob er den DMR-, D-Star oder C4FM-Standard benutzen möchte und kann trotzdem unabhängig von der Betriebsart mit den anderen Funkamateuren QSOs fahren. Im Moment funktioniert das schon wunderbar zwischen DCS, DExtra und DPlus. Der nächste Schritt wird die Implementierung von DMR sein. Ob uns das gelingen wird, wird sich zeigen. Auch C4FM steht auf unserer Roadmap.

Ist eine Cross-Verlinkung/Bridging zwischen den einzelnen Modi möglich bzw. geplant?

Wir möchten das analoge Audiosignal als „Common-Interface“ vermeiden. Einmal digitalisiert, sollte man bis zum „End-User“ digital bleiben. Aus diesem Grund wird es keine Bridge geben, so wie man sie von anderen Systemen her kennt. Die Audioqualität durch die mehrfache Umwandlung ist uns einfach zu schlecht. Alle Protokolle werden über direktes Transcoding eingebunden werden. Momentan kann man schon über das DCS-, REF- oder XRF-Protokoll einsteigen, später sollte dies auch mit DMR, C4FM oder weiteren Protokollen möglich sein. Auch lassen sich die XLX-Server unter einander vernetzen. Als Verbindung wird eine „Peer to Peer“-Verbindung über das XLX-Protokoll benutzt.

Wie „offen“ ist das System nach außen? Gibt es definierte Schnittstellen für eigene Entwicklungen?

Der XLX besteht aus zwei Teilen, dem XLX Core Server und dem Dashboard. Im Moment gibt es drei Schnittstellen, eine JSON, eine XML und einen Interlink. Der XLX ist seit Januarbeginn 2016 unter GPL Lizenz als Open Source frei verfügbar.

Welche Programmiersprache(n) werden verwendet?

Der Core Server ist komplett in C++ geschrieben, das Dashboard in PHP.

Welche Mindestanforderungen werden an einen Server gestellt?

Der XLX ist sehr sparsam an Ressourcen. Obschon er auf einem Raspberry Pi laufen kann, empfehlen wir dennoch eine Server-Infrastruktur. Wie viel Leistung schlussendlich benötigt wird, ist aber im Endeffekt davon abhängig, wie viele Module und Protokolle gleichzeitig genutzt werden.

Wie stellt ihr euch euren bevorzugten Weg für Supportanfragen vor?

Der Support wird ausschließlich über die Yahoo Gruppe [xlxd-star](https://groups.yahoo.com/neo/groups/xlxd-star/info) abgewickelt, zu finden unter: <https://groups.yahoo.com/neo/groups/xlxd-star/info>.

(Die Fragen stellte Kim Hübel, DG9VH)



Luc Engelmann, LX1IQ