

# Upcoming developments and improvements in SeedLink

---

Andres Heinloo & Jan Becker



---

January 17, 2013

# Introduction

- When SeedLink was developed, our goal was to transfer data from a relatively small number of stations to the data center as fast as possible, but without any specific real-time requirements.
- In connection with large networks and real-time processing, some scalability issues and new requirements have arisen.

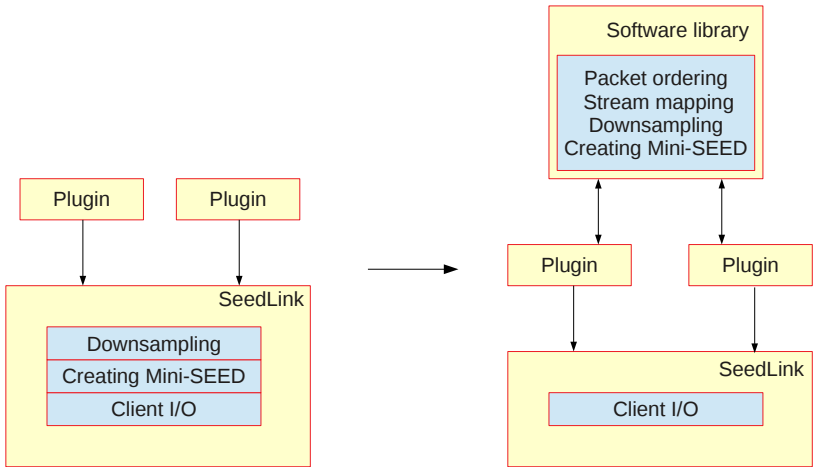
# Planned short-term improvements

In the first stage, it is planned to create a software library that implements the following for all plugins:

- generic, transparent buffering and reordering of out-of-order data (currently plugin-dependent);
- generic configuration of stream mapping (currently plugin-dependent);
- downsampling (multiple plugins→multiple CPU cores used);
- creating Mini-SEED.

This is illustrated on the following figure.

# Planned short-term improvements



# Long-term changes to be discussed

Probably a major rewrite and protocol change is needed to fully meet today's requirements of large-scale real-time networks. Below are some aspects that can be improved:

- Availability.
  - Better indexing/buffer structure to avoid long start-up time after a crash.
  - Support adding new stations in live system (without manual reconfiguration and restart), station wildcards.
  - Support multiple servers for redundancy.
- Performance.
  - Use multiple CPU cores if available (multi-threading).
  - Support arbitrarily large data buffer. No need to run slarchive→less disk I/O.
- Data completeness.
  - Support out-of-order data (backfilling).
- Real-time processing.
  - Let each client select maximum tolerable latency in case of gaps.
  - Support other formats besides Mini-SEED to avoid delay caused by fixed record size.