The BioSig Client Server protocol

Version 0.11

(BSCS v0.1)

Alois Schlögl <alois.schloegl@tugraz.at>

3 Apr 2009

Introduction

This document describes the message protocol for storing and retrieving biomedical signal data at a dedicted server. The purpose is to provide a solution for data archiving of biomedical signals. The current protocol enables online recording and transmitting the data on a distant server without the need of a local storage. Moreover, a central server for biomedical signals is provided.

The protocol is currently a connection-based protocol using TCP. Each file is identified by a 64bit (8-byte) identity number (the ID), the number is randomly generated by the server. This prevents guessing the file identity. The uploading client receives the ID. Data can be retrieved from anywhere but only if the ID is known.

The transmitted data is based on the „General data format for biomedical signals" (GDF) [1]. The BSCS protocol is already implemented in BioSig4c++ v0.80 or later available from (http://biosig.sf.net). Tools for uploading and retrieving data to and from the server are available.

The specification of the BSCS protocol

If not specified otherwise, the client send a message and the server responds with a corresponding reply message. Each message consists of the following fields:

uint8_t VER	version number , always 0x01 (indicating version 0.1)
uint8_t CMD	command
uint8_t STATE	state of the server
uint8_t ERR	error code
uint32_t LEN	length of the buffer (network byte order, big endian)
void *buffer	data load (GDF byte order, little endian)

The command CMD byte supports the following commands:

0x01   BSCS_OPEN_READ,
open for reading is indicated by transmitting the ID number in the buffer, accordingly LEN = 8. In case of success, the server state is set to STATE_OPEN_READ, and the server may serve data requests BSCS_REQU_{HDR,DAT,EVT}

0x01   BSCS_OPEN_WRITE
open for writing is indicated by LEN=0; the ID number is returned in the reply message; in case of success, the server state is set to STATE_OPEN_WRITE; BSCS_OPEN_READ and BSCS_OPEN_WRITE share the same code, and can be distinguished whether LEN = 0 or LEN = 8;

0x02   BSCS_CLOSE
the data transmission is closed; also the connection is closed.

0x03   BSCS_SEND_MSG
buffer will contain any message string, typically using a null-terminated ascii string. This command does not trigger any reply message.

0x04   BSCS_SEND_HDR
The header information is send. The buffer contains the file header as defined by the GDF [1]. This command can be only sent once.

0x05   BSCS_SEND_DAT
one or more data blocks are send. The buffer contains the raw data of one or more GDF data blocks. This command can be repeated.

0x06   BSCS_SEND_EVT
one or more events are sent. The buffer contains an event table as defined by GDF. This command can be repeated.

0x07   BSCS_REQU_HDR
requests the header of the opened file. The reply message will contain the HDR information of the GDF file

0x08   BSCS_REQU_DAT
the buffer must contain two uint32_t numbers in little endian format, indicating the number of data blocks and the starting data block. Several consequtive requests are possible.

0x09   BSCS_REQU_EVT
the buffer is empty, the event table is requested. The reply message will contain the evenbt table according to the GDF [1].

0x0a   BSCS_PUT_FILE
send a whole file to the server

0x0b   BSCS_GET_FILE
retrieve a whole GDF [1] file from the server

0x8x   indicating a reply message is done by OR-ing the command with 0x80 (setting the highest bit).

The STATE field indicates the state of the connection to the server, and is reported in each reply message from the server to the client. The client must track the server state, and include the state in the message header. Otherwise, the server will ignore the message or reply with an error message.

0x00   STATE_INIT
In this state, only BSCS_OPEN_READ or BSCS_OPEN_WRITE commands can be performed

0x01   STATE_OPEN_READ
In this state BSCS_REQU_{HDR,DAT,EVT} commands can be performed.

0x02 STATE_OPEN_WRITE_HDR
   In this state BSCS_SEND_HDR command can be performed.
0x03 STATE_OPEN_READ
   In this state BSCS_SEND_{DAT,EVT} commands can be performed.

The ERR field is defined in the reply message. If some error occurred, a non-zero number is returned.

Server State Table

The first two columns of the state table define the valid combination of the current server state and the possible client commands. All other combinations are invalid. In case of success, the third column shows the resulting server state.

Client Command	SERVER STATE(t)	SERVER STATE(t+1)
BSCS_OPEN_READ	STATE_INIT	STATE_OPEN_READ
BSCS_OPEN_WRITE	STATE_INIT	STATE_OPEN_WRITE_HDR
BSCS_REQU_HDR BSCS_REQU_DAT BSCS_REQU_EVT	STATE_OPEN_READ	STATE_OPEN_READ
BSCS_SEND_HDR	STATE_OPEN_WRITE_HDR	STATE_OPEN_WRITE
BSCS_SEND_DAT BSCS_SEND_EVT	STATE_OPEN_WRITE	STATE_OPEN_WRITE
BSCS_CLOSE	<any>	STATE_INIT
BSCS_PUT_FILE	STATE_OPEN_WRITE_HDR	STATE_INIT
BSCS_GET_FILE	<any>	SERVER STATE(t) (no change)
BSCS_NOP (no operation)	<any>	SERVER STATE(t) (no change)

Further remarks

A server for testing is available at bscs://fstgam01.tu-graz.ac.at or bscs://129.27.3.99 at port 54321.

Once, the connection is established, the server will send a welcome message (CMD=0x03). However, this feature might be dropped in future.

References:

[1] A. Schlögl, GDF - A general dataformat for BIOSIGNALS v2.x, available from here
   http://arxiv.org/abs/cs.DB/0608052