

Re: socket communication: send & receive doesn't work right

Source:

<http://www.tech-archive.net/Archive/Development/microsoft.public.win32.programmer.networks/2007-04/msg00323>

- *From:* Ananya <Ananya@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx>
 - *Date:* Mon, 30 Apr 2007 08:32:02 -0700
-

How can I send the string as string (not bytes) in Java and how can I receive it in C++?

"Ananya" wrote:

Ok, but Michael suggested to send a string for getting around the bytes endian problem.

So I don't want to send a string as bytes. Can I send a string as a string?

"Scherbina Vladimir" wrote:

If I recall correctly you can convert "string" data type on Java side to a bytes array using `.getBytes(...)` or something like that.

--

--Vladimir, Windows SDK MVP

"Ananya" <Ananya@xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx> wrote in message news:21045337-B7CE-4FC2-86B0-4631DCBFF7D1@xxxxxxxxxxxxxxxxxxxx

That sounds great! But can I actually send and receive strings? I thought that I only can send and receive bytes. How do I send and receive strings?

Thanks in advance for your answer!

"Michael K. O'Neill" wrote:

Re: socket communication: send & receive doesn't work right

You can eliminate all issues of endian-ness by converting the doubles to strings (sprintf-style) and sending the string. On the receiving side, use an atof() function to convert back.

Note that this approach also resolves all issues concerning whether the sending and receiving side are both using the same implementation of IEEE-754 for storing double precision numbers. Since string equivalents are sent, the precise implementation simply doesn't matter.

Finally, it's often easier to debug, since you can "see" the number that's being sent.

Mike

"Scherbina Vladimir"

<v_scherbina@xxxxxxxxxxxxxxxx> wrote in message

news:EB69C218-6DAE-4776-ABC0-F601C7878116@xxxxxxxxxxxxxxxx

In Windows everything is little endian, <http://support.microsoft.com/kb/q102025/>. I suggest you to send simple

data

from java program, and analyze what you obtain on c++ side. For example, send an integer value 10, and tell us here what you receive in C++ application.

--

--Vladimir, Windows SDK

MVP

"Ananya"

<Ananya@xxxxxxxxxxxxxxxxxxxxxxxx> wrote in message

Re: socket communication: send & receive doesn't work right

news:7E0E8354-A53A-4B8F-A0B4-33A51785FEE4@xxxxxxxxxxxxxxxxxxxx

```
Well, I tried
not to
reverse
bytes by
saying:
for (i = 0; i
< j; i++)
{
ptr[i] =
result[i];
}
```

```
instead of:
for (i = 0; i
< num; i++)
{
for (j=0;
j<sizeof(double);
j++)
{
ptr[i*sizeof(double)+j]
=
(char)result[(i+1)*sizeof(double)-j-1];
}
}
in my C++
receiving
method.
```

```
But now the
doubles:
1.23 & 4.5
which I
send from
my Java
program
always
become:
1.8584604523406555e+038
&
5.910042899492e-318#DEN
in my C++
program.
```

```
Please help!
By the way,
how can I
test if my
server needs
```

Re: socket communication: send & receive doesn't work right

reversing
bytes
or not?

"Scherbina
Vladimir"
wrote:

I
am
saying
that
if
your
server
is
big
endian,
then
you
_do_not_
need

to

reverse
bytes.

--

--Vladimir,
Windows
SDK
MVP
"Ananya"

<Ananya@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>

wrote

in

message

news:8B29C2D6-060A-41BA-8180-F4AA146F49A6@xxx

Thanks!

Are

you

saying

that

in

my

C++

receiving

method

Re: socket communication: send & receive doesn't work right

Re: socket communication: send & receive doesn't work right

the
code
for
reversing
the
byte
order
is
incorrect?
Where
can
I
find
the
correct
code?

"Scherbina
Vladimir"
wrote:

The
problem
with
this
approach
is
that
Java
stores
the
binary
data

as

big
endians
only
(no
matter
what
CPU
architecture
is),
if
your
C++

Re: socket communication: send & receive doesn't work right

client

is
little
endian
then
unpredictable
results
may
be
obtained.
Check

this

issue.

--

--Vladimir,
Windows
SDK
MVP
"Ananya"

<Ananya@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>

wrote

in

message

<news:2951DED5-4DDF-40BF-A581-1A13>

I
am
trying
to
establish
socket
communication
between
my
Java
and

C++

program.

I
called
my
Java
program
from

Re: socket communication: send & receive doesn't work right

my
C++
program
with
ShellExecuteEx.

I
created
a
C++
Server
and
a
Java
Client,
which
is
accepted
by
the
Server.

I
did
a
test
of
sending
two
doubles:
1.23
&
4.5
from
my
Java
program
to
my
C++
program,
however
I
always
received
the
following
2
different
doubles:
1.1648250968930678e-302
&

Re: socket communication: send & receive doesn't work right

-6.4627233651951511e-086.

Here
is
my
Java
sending
method:
public
void
send_doubles(double
vals[],
int
len)
throws

IOException

```
{  
//  
convert  
our  
array  
of  
doubles  
into  
an  
array  
of  
bytes  
ByteArrayOutputStream  
bytestream;  
bytestream  
=  
new  
ByteArrayOutputStream(len*8);  
  
DataOutputStream  
out;  
out  
=  
new  
DataOutputStream(bytestream);  
  
for  
(int  
i=0;  
i<len;  
i++)  
{  
out.writeDouble(vals[i]);
```

Re: socket communication: send & receive doesn't work right

```
}  
  
output.write(byteStream.toByteArray()  
0,  
byteStream.size());  
output.flush();  
  
recv_ack();  
send_ack();  
}  
  
and  
my  
Java  
acknowledgement  
methods:  
//  
send  
a  
short  
acknowledgement  
to  
the  
server  
private  
void  
send_ack()  
throws  
IOException  
{  
int  
ack;  
  
ack  
=  
0;  
  
output.write(ack);  
output.flush();  
}  
  
//  
recv  
a  
short  
acknowledgment  
from  
the  
server  
private  
void
```

Re: socket communication: send & receive doesn't work right

```
recv_ack()  
throws  
IOException  
{  
int  
ack;  
  
ack  
=  
(int)input.read();  
}
```

```
And  
here  
is  
my  
C++  
receiving  
method:  
int  
Server::recv_doubles(double  
*val,  
int  
maxlen)  
throw  
(string)  
{  
int  
i,  
j;  
int  
numbytes  
=  
0;  
int  
end  
=  
0;  
int  
total_bytes  
=  
0;  
char  
*temp;  
char  
*result;  
  
temp  
=  
(char  
*)buffer;
```

Re: socket communication: send & receive doesn't work right

```
result
=
(char
*)buffer2;

j
=
0;

//
we
are
receiving
the
incoming
doubles
one
byte
at
a
time
while
(!end)
{
if
((numbytes=recv(new_fd,
temp,
BUFSIZE,
0))===-1)
{
throw
string("help!");
}

for
(i=0;
i<numbytes;
i++)
{
result[j]
=
temp[i];
j++;
}

total_bytes
=
total_bytes
+
numbytes;
if
```

Re: socket communication: send & receive doesn't work right

```
(total_bytes==maxlen*sizeof(double)
+
1)
{
end
=
1;
}
}

//
now
we
need
to
put
the
array
of
bytes
into
the
array
of
doubles
char
*ptr;
int
num
=
(j
-
1)/sizeof(double);

ptr
=
(char
*)val;

//
going
from
Java
to
C++,
we
need
to
reverse
the
order
```

Re: socket communication: send & receive doesn't work right

set

of
each

```
of
bytes
for
(i
=
0;
i
<
num;
i++)
{
for
(j=0;
j<sizeof(double);
j++)
{
ptr[i*sizeof(double)+j]
=
(char)result[(i+1)*sizeof(double)-j-1];
}
}
```

```
send_ack();
recv_ack();
```

```
return
num;
}
```

and
my
C++
acknowledgement
methods:

```
//
receive
a
short
acknowledgement
from
the
client
void
Server::recv_ack()
{
char
```

Re: socket communication: send & receive doesn't work right

```
temp[1];
int
total
=
0;

while
(total<1)
{
total
+=
recv(new_fd,
temp,
1,
0);
}

//
send
a
short
acknowledgement
to
the
client
void
Server::send_ack()
{
char
temp[1];
temp[0]
=
42;

send(new_fd,
temp,
1,
0);
}
```

Why
does
my
C++
program
receive
incorrect
doubles?

Thanks

Re: socket communication: send & receive doesn't work right

for
looking
at
my
code!