Typed Data

Apart from adding function calls the code, a programmer must also provide an XML (eXtensible Markup Language)

```
String config_filename /* the name of the config file*/
                       /* to be parsed */
```

Output parameter

```
TDTConfig tc /* a parsed copy of the configuration file */
```

Example:

```
tc = tdt_configure("config.xml");
```

2.2.2 tdt_open()

ose

`tdt_open()` **opens**

Output

2.3 Compiling and linking

2.3.1 Compiling the library

The TDT can be compiled as a static library, `libtdt.a`. This must be in the library path of the development environment, or a directory specified by the `-L` flag in gcc. A makefile is included with the source files. Running "`make lib`" from within the `tdt` source directory will rebuild the TDT library.

willdthe

2.3.2 Linking with your own programs

```
        /* ... and write
```

```
INTEGER tdtstate
INTEGER tdtconfig
```

3.2 User functions

tdt_configure()

Read and parse the specified configuration file.

Input parameter

```
C the name of the configuration file
configfilename
```

Output parameter

```
C a parsed
```

```
C a completed TDTState variable
tdtstate
```

Examples:

```
CALL tdt_fopen (tdtstate, tdtconf, "client_to_server")
```

3.2.3 tdt_fwrite()

Purpose

To write data to the connection given by the `TDTState` **parameter as per the XML identifer string (parameter**

3.3 Compiling and link

3.3.1 Compiling the

4the4Python4Prerequisites4Imp4orts4The4line4from4tdt4T D T4that4tdt.p4TH.4Declara

4.2.3 read

Purpose

To read the data specified by the givenT

4.2.6 end

Purpose

Closes connections or open files and releases TDT structures.

Input parameters

None

Example

```
tdt.end()
```

4.3 Example programs

```
from tdt import TDT

tdt = TDT ()
```

5.7 Example

The following XML

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<data_desc>

    <decl name="vardouble">double</decl>

    <decl name="astruct">
        <struct>
            <decl name="elem1">
                    <array size="2">int</array></decl>
            <decl name="elem2">double</decl>
        </struct>
    </decl>

    <decl name="varint">int</decl>

    <decl name="dyn">
        <array size="0">double</array>
    </decl>

</data_desc>
```

represents describes the data being transferred in the client program above. It describes the following C declarations:

```
double vardouble;

struct {
    int    elem1[2];
    double elem2;
} astruct;

int varint;
double *dynarray;
```

6 TDT Con gudouble les

The con gudouble les used by TDT are also written in XML.

6.1 <program> tag

The do

7.1 C Examples

The sample C programs (`testclnt.c` **and** `testserv.c` `"make clean"` **will remove old object (**`*.o`**) files and executable programs from**

Td (tkt)Tj /R16 y07 041.35Td (**the**) `wdc/tests` **and** `ddt` **directories (or** `dadc/tests`

f790 Td ((/R363