010 Editor Templates Cheatsheet

«Templates have a similar syntax to C/C++ structs but **they are run as a program**. Every time a variable is declared in the Template, the variable is mapped to a set of bytes in the current file.»

Structure with arguments

Bitfields
Padded
Unpadded

Endianness
BigEndian
LittleEndian

```
typedef struct (int size) {
   int id;
   int array[size];
} VAR_SIZED;

typedef struct {
   byte flag:1;
   byte version:7;
} FLAG_VERSION;

typedef struct {
   BigEndian();
   ushort  type;
   ushort  len;
   byte    value[len];
} TLV;

TLV tlv[5] <optimize=false>;
```

Optimization
Turn OFF the optimization for variable sized structs.
(by default, array size is based on the size of first element)

Built-in functions
- Interface Functions
- I/O Functions
- String Functions
- Math Functions
- Tool Functions

1 Define Types/Structures

```
typedef struct {
  char  chunkID[4];
  BigEndian();
  uint  length;
  char  type[4];
  local char tempID[4];
  while (FTell() < length) {
      if (FTell() + 4 < length)
        ReadBytes(tempID, FTell(), 4);
      else
          break;
      switch (tempID) {
         case "FORM": FORM_T chunk <comment=chunkInfo>;
                      break;
         case "DIRM": DIRM_T chunk <comment="Dir Chunk">;
                      break;
         default:
            Printf("Unknown chunk: %s\n", chunk.id);
            CHUNK_T chunk <comment=chunkID>;
      }
  }
} MAIN_FORM_T <bgcolor=cLtGreen>;
```

Local Variables
Not mapped to a file
Not displayed in the Template Results

Read* functions
Do not change file cursor

2 Implement Custom Functions

```
string chunkInfo (FORM_T &chunk) {
    string buf;
    SPrintf(buf, "%s:%s", chunk.id, chunk.type);
    return buf;
}
```

3 Declare Template Variables

```
ID header <comment="Header should be AT&T">;
if (header != "AT&T") {
    Warning("File is not a valid DjVu file.");
    return -1;
}
MAIN_FORM_T form <comment=mainFormID>;
```

Special keywords

sizeof
Returns the size in bytes of a type/variable.

startof
Returns the start address of the bytes the variable is mapped to in the file.

exists
Determines if a variable has been declared.

function_exists
Tests if a particular function is defined.

this
Accesses the variable representing the current structure being defined.

parentof
Accesses the struct or union that contains a given variable.

Built-in Types

Signed / Unsigned Integers

char, byte, CHAR, BYTE
uchar, ubyte, UCHAR, UBYTE

short, int16, SHORT, INT16
ushort, uint16, USHORT, UINT16, WORD

int, int32, long, INT, INT32, LONG
uint, uint32, ulong, UINT, UINT32, ULONG, DWORD

int64, quad, QUAD, INT64, __int64
uint64, uquad, UQUAD, UINT64, QWORD, __uint64

Floating Point Number

16-Bit: hfloat, HFLOAT
32-Bit: float, FLOAT
64-Bit: double, DOUBLE

Date types

DOSDATE, DOSTIME, FILETIME, OLETIME, time_t, time64_t

String types

string, wchar_t, wstring

GUID (Globally Unique Identifier)

GUID

Special Attributes

For types, fields and variables

< **format**=hex|decimal|octal|binary,
fgcolor=<color>,
bgcolor=<color>,
comment="<string>"|<function_name>,
name="<string>"|<function_name>,
open=true|false|suppress,
hidden=true|false,
read=<function_name>,
write=<function_name>
size=<number>|<function_name> >

Official documentation: https://www.sweetscape.com/010editor/manual/IntroTemplates.htm

Cheatsheets
by Pavel Rusanov