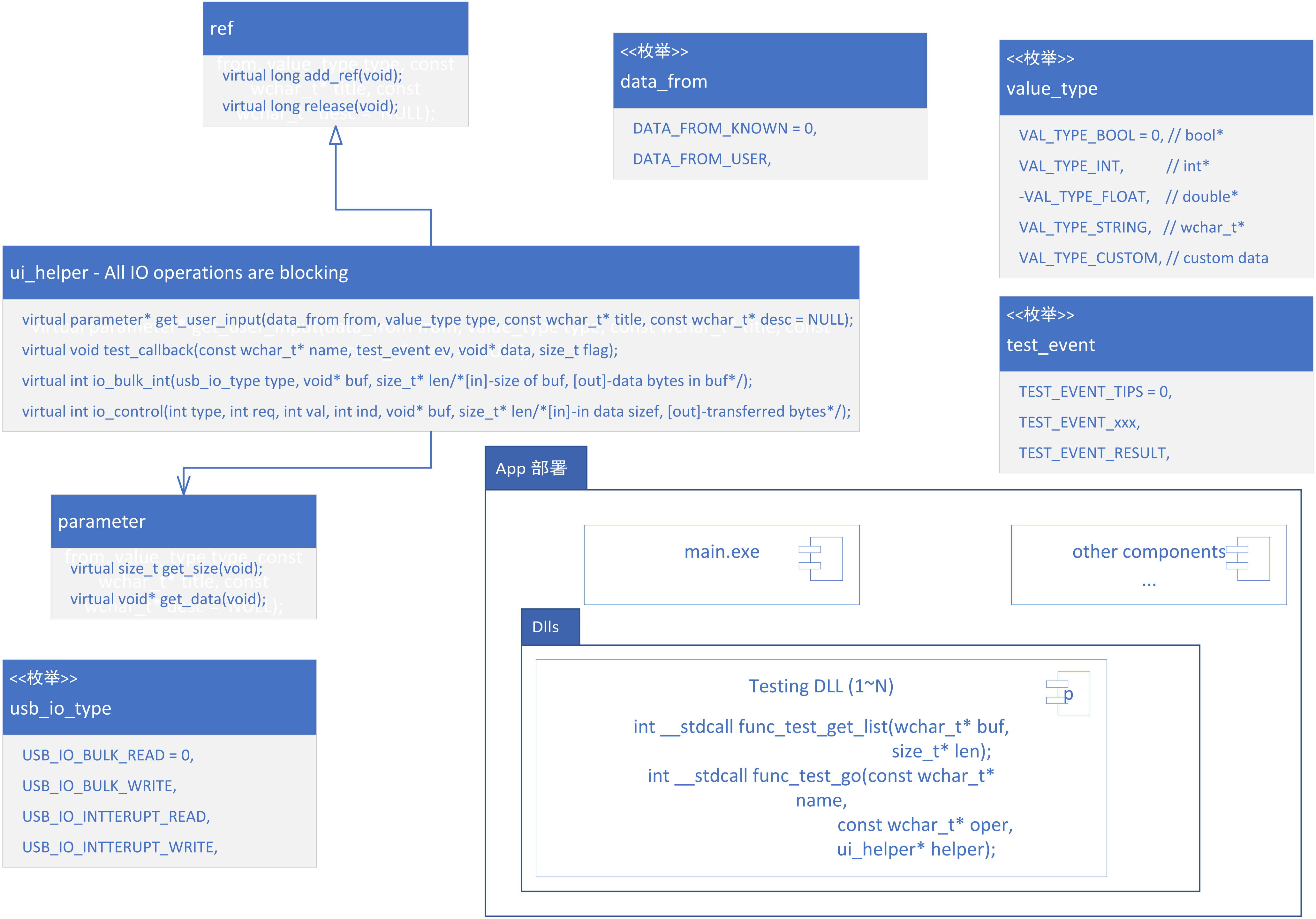

ref

virtual long add_ref(void);

virtual long release(void);

<<枚举>>

data_from

DATA_FROM_KNOWN = 0,

DATA_FROM_USER,

<<枚举>>

value_type

VAL_TYPE_BOOL = 0, // bool*

VAL_TYPE_INT, // int*

-VAL_TYPE_FLOAT, // double*

VAL_TYPE_STRING, // wchar_t*

VAL_TYPE_CUSTOM, // custom data

ui_helper - All IO operations are blocking

```
virtual parameter* get_user_input(data_from from, value_type type, const wchar_t* title, const wchar_t* desc = NULL);
virtual void test_callback(const wchar_t* name, test_event ev, void* data, size_t flag);
virtual int io_bulk_int(usb_io_type type, void* buf, size_t* len/*[in]-size of buf, [out]-data bytes in buf*/);
virtual int io_control(int type, int req, int val, int ind, void* buf, size_t* len/*[in]-in data sizef, [out]-transferred bytes*/);
```

<<枚举>>

test_event

TEST_EVENT_TIPS = 0,

TEST_EVENT_xxx,

TEST_EVENT_RESULT,

parameter

virtual size_t get_size(void);

virtual void* get_data(void);

App 部署

main.exe

other components

...

Dlls

Testing DLL (1~N)

p

```
int __stdcall func_test_get_list(wchar_t* buf,
                                 size_t* len);
int __stdcall func_test_go(const wchar_t*
name,
                           const wchar_t* oper,
                           ui_helper* helper);
```

<<枚举>>

usb_io_type

USB_IO_BULK_READ = 0,

USB_IO_BULK_WRITE,

USB_IO_INTTERUPT_READ,

USB_IO_INTTERUPT_WRITE,

测试列表

```
// 存储在服务器或者由管理人员提前配置的指定机型测试项配置（需要测试哪些项目的列表）—cfg-master
{
        "global":
        {
                "vid-org": 0x3072,                                  // 初始状态VID
                "pid-org": 0x200,                                   // 初始状态PID
                "vid-to": 0x3072,                                   // 要写入的目标VID
                "pid-to": 0x239,                                    // 要写入的目标PID
                "report-url": "user@factory-quality-center"         // 数据上报/数据库连接字符串参数
        },
        "1":
        {
                "name": "test-1",        // 全局唯一名字
                "title": "歪斜检测",      // 测试项目标题
                "man": false,            // false － 程序联机检测项目；true - 全人工手动检测项目
                "err-level": "fatal"     // "fatal" - stop all testing works when error, "warning" - warning/ignore this item and continue testing
        }
}
```

测试项

```
// 各专用测试DLL 返回支持的测试项 配置—cfg-performer
{
        "1":
        {
                "name": "test-1",        // 全局唯一，与cfg-master中name对应
                "ver":      1            // 如果有多个DLL同时支持某一个测试项，则取版本号高的测试
        }
}
```