SmartBridge Professional

Version 4.0

SmartBridge plug-ins

How to write your own plug-in for your SmartBridge

issamsoft.com

SmartBridge plug-ins

SmartBridge supports plug-ins, you can write your own plug-in using one of the native programming language like (C/C++, Delphi, Rust …etc) that can compile dynamic link libraries. You can use these extensions to connect SmartBridge with other software/system like (Billing, ERP…etc). You may need to have the source code of the other software to be able to modify it and process the data saved by your plug-in.

Plug-in structure

SmartBridge plug-in is a dll file that contains callback functions to be called by SmartBridge. Currently one plug-in is available which means you can write one plug-in only. This plug-in will contains two callbacks to called when (Save, Delete) weight ticket in SmartBridge. For this plug-in the dll file should be named **wmodule.dll** and should be saved in **modules** directory in SmartBridge path.

The save ticket callback should be named **sb_save_ticket_callback** and has the next declaration:

```
void  sb_save_ticket_callback(
                        int ticket_number,
                        const wchar_t *tare_weight,
                        time_t tare_weight_datetime,
                        const wchar_t *gross_weight,
                        time_t gross_weight_datetime,
                        const wchar_t *net_weight,
                        const wchar_t *unit,
                        double cargo_count,
                        double product_unit_price,
                        double product_total_price,
                        int record_type, // -0 auto  -1 manual -2 edit
                        const wchar_t *vehicle_number,
                        const wchar_t *driver_name,
                        const wchar_t *product,
                        const wchar_t *warehouse,
                        const wchar_t *client,
                        const wchar_t *custom_field1,
                        const wchar_t *custom_field2,
                        const wchar_t *custom_field3,
                        const wchar_t *custom_field4,
                        const wchar_t *custom_field5,
                        const wchar_t *custom_field6,
```

```
                        const wchar_t *custom_field7,
                        const wchar_t *custom_field8,
                        const wchar_t *custom_field9,
                        const wchar_t *custom_field10,
                        const wchar_t *custom_field11,
                        const wchar_t *custom_field12,
                        const wchar_t *custom_field13,
                        const wchar_t *custom_field14,
                        const wchar_t *custom_field15,
                        const wchar_t *custom_field16
                        );
```

SmartBridge will call this function from your plug-in each time ticket saved.

The delete ticket callback should be named **sb_delete_ticket_callback** and has the next declaration:

```
void sb_delete_ticket_callback(int ticket_number)
```

SmartBrigde will call this function from your plug-in each time ticket deleted.

You should handle exceptions inside the callbacks functions. Undefined behaviors inside the callbacks may cause SmartBridge crashing.

Plug-in sample in C

In this sample I will show you how to write SmartBridge plug-in in C. this plug-in will save the ticket data in a separate MySql database.

First let create the MySql database.

```
Create Database weightdb default character set utf8;
```

Create simple table to save needed ticket data.

```
create table tblWeights
(
    wID integer UNSIGNED not null AUTO_INCREMENT,
    ticket_number integer UNSIGNED not null,
    plateNumber  varchar(256),
    TareWeight double,
    TareWeightDateTime DateTime,
    GrossWeight double,
    GrossWeightDateTime DateTime,
    netWeight  DOUBLE,
    PRIMARY KEY (`wID`)
```

```
)ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci
AUTO_INCREMENT=1;
```

Now let us write the C code for this plug-in. this code will be compiled using MS C compiler.

```
#include <stdlib.h>
#include <stdio.h>
#include <mysql.h>
#include <time.h>
#include <string.h>
#include <wchar.h>

__declspec(dllexport) void  sb_save_ticket_callback(
                int ticket_number,
                const wchar_t *tare_weight,
                time_t tare_weight_datetime,
                const wchar_t *gross_weight,
                time_t gross_weight_datetime,
                const wchar_t *net_weight,
                const wchar_t *unit,
                double cargo_count,
                double product_unit_price,
                double product_total_price,
                int record_type, // -0 auto  -1 manual -2 edit
                const wchar_t *vehicle_number,
                const wchar_t *driver_name,
                const wchar_t *product,
                const wchar_t *warehouse,
                const wchar_t *client,
                const wchar_t *custom_field1,
                const wchar_t *custom_field2,
                const wchar_t *custom_field3,
                const wchar_t *custom_field4,
                const wchar_t *custom_field5,
                const wchar_t *custom_field6,
                const wchar_t *custom_field7,
                const wchar_t *custom_field8,
                const wchar_t *custom_field9,
                const wchar_t *custom_field10,
                const wchar_t *custom_field11,
                const wchar_t *custom_field12,
                const wchar_t *custom_field13,
```

```
                const wchar_t *custom_field14,
                const wchar_t *custom_field15,
                const wchar_t *custom_field16
                            ){
  MYSQL *con = mysql_init(NULL);
  if(NULL == con)
    return;
  mysql_options(con, MYSQL_SET_CHARSET_NAME, "utf8");
  if(mysql_real_connect(con, "localhost", "root", "root_pass",
"weightdb", 3306, NULL, 0) == 0)
    return;
  //tare_weight
  wchar_t tr_w[32] = {0};
  if(wcslen(tare_weight) == 0)
    swprintf(tr_w, sizeof tr_w / sizeof *tr_w, L"NULL");
  else
    swprintf(tr_w, sizeof tr_w / sizeof *tr_w, tare_weight);
  //tare_weight_datetime
  wchar_t tr_dt[24] = {0};
  if(0 != tare_weight_datetime){
    wcsftime(tr_dt, sizeof tr_dt / sizeof *tr_dt, L"'%F %T'",
gmtime(&tare_weight_datetime));
  }
  else
    swprintf(tr_dt, sizeof tr_dt / sizeof *tr_dt, L"NULL");
  //gross_weight
  wchar_t gr_w[32] = {0};
  if(wcslen(gross_weight) == 0)
    swprintf(gr_w, sizeof gr_w / sizeof *gr_w, L"NULL");
  else
    swprintf(gr_w, sizeof gr_w / sizeof *gr_w, gross_weight);
  //gross_weight_datetime
  wchar_t gr_dt[24] = {0};
  if(0 != gross_weight_datetime)
    wcsftime(gr_dt, sizeof gr_dt / sizeof *gr_dt, L"'%F %T'",
gmtime(&gross_weight_datetime));
  else
    swprintf(gr_dt, sizeof gr_dt / sizeof *gr_dt, L"NULL");
  //net_weight
  wchar_t nt_w[32] = {0};
```

```
  if(wcslen(net_weight) == 0)
    swprintf(nt_w, sizeof nt_w / sizeof *nt_w, L"NULL");
  else
    swprintf(nt_w, sizeof nt_w / sizeof *nt_w, net_weight);
  //prepare query
  char qrfmt[] = "Insert into tblWeights (ticket_number, plateNumber,
TareWeight, TareWeightDateTime, GrossWeight, GrossWeightDateTime,
NetWeight) values (%d, '%ls', %ls, %ls, %ls, %ls, %ls)";
  char *qr;
  int sz = snprintf(NULL, 0, qrfmt, ticket_number, vehicle_number,
tr_w, tr_dt, gr_w, gr_dt, nt_w);
  if(sz < 0)
    return;
  qr = (char*)malloc((sz+1) * sizeof(char));
  if(!qr)
    return;
  snprintf(qr, sz+1, qrfmt, ticket_number, vehicle_number, tr_w, tr_dt,
gr_w, gr_dt, nt_w);
  //execute query
  mysql_query(con, qr);
  //cleanup
  free(qr);
  mysql_close(con);
}

__declspec(dllexport) void sb_delete_ticket_callback(int
ticket_number){
  MYSQL *con = mysql_init(NULL);
  if(NULL == con)
    return;
  if(mysql_real_connect(con, "localhost", "root", "root_pass",
"weightdb", 3306, NULL, 0) == 0)
    return;
  char *qrfmt = "delete from tblWeights where ticket_number = %d";
  char *qr;
  int sz = snprintf(NULL, 0, qrfmt, ticket_number);
  if(sz < 0)
    return;
  qr = (char*)malloc((sz+1)*sizeof(char));
  if(!qr)
```

```
    return;
  snprintf(qr, sz+1, qrfmt, ticket_number);
  mysql_query(con, qr);
  free(qr);
  mysql_close(con);
}
```

Compile this code using MS compiler into **wmodule.dll**. Create **modules** directory inside SmartBridge path. Copy wmodule.dll into the modules directory.