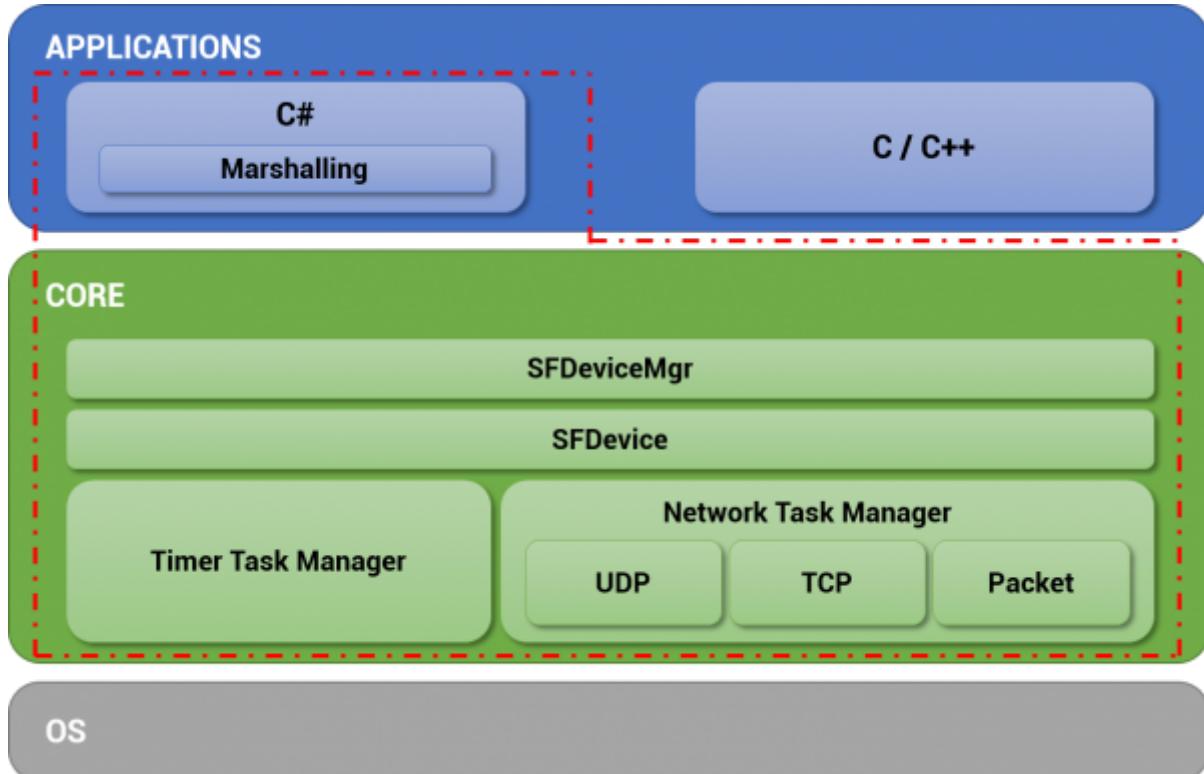
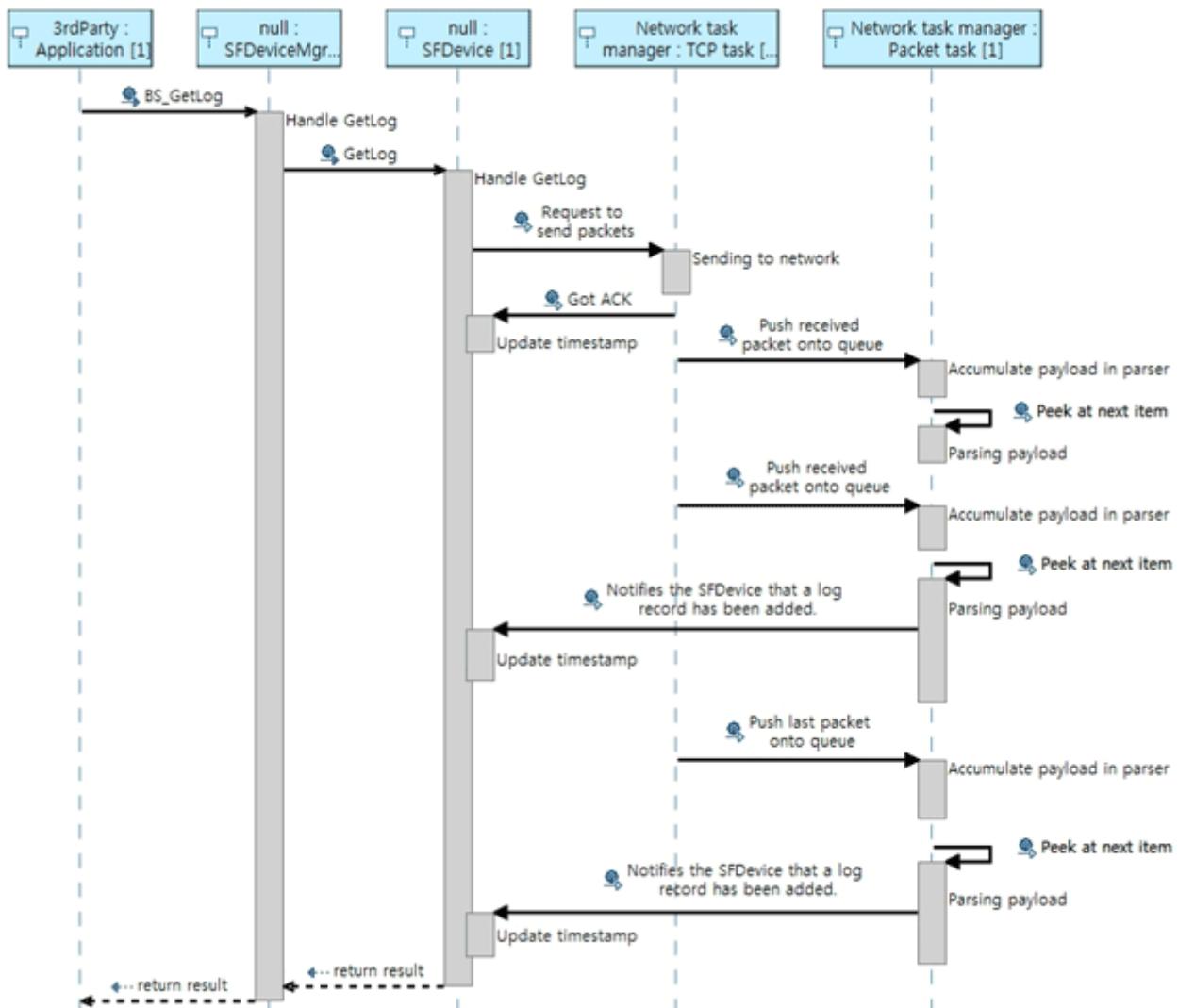


.....	1
.....	1
.....	1
.....	2
.....	3
BioStar 1.x SDK	3
- API	3
-	5
-	5
-	5
.....	7
Visual Studio	7

BioStar 2 Device SDK

SDK	Document ¹⁾					
	Include ²⁾					
	Lib	linux	lib	x86	BS_SDK_V2.so	
				x64	BS_SDK_V2.so	
	Lib	window	lib	x86	BS_SDK_V2.lib ³⁾ BS_SDK_V2.dll	
				x64	BS_SDK_V2.lib ⁴⁾ BS_SDK_V2.dll	
	Example ⁵⁾	C#				
		C++				





BioStar 2

BioStar 1.x SDK

API

BioStar 1.x SDK

API 가

```
if( m_DeviceType == BS_DEVICE_BIOENTRY_PLUS ||  
    m_DeviceType == BS_DEVICE_BIOENTRY_W ||  
    m_DeviceType == BS_DEVICE_BIOLITE ||  
    m_DeviceType == BS_DEVICE_XPASS ||  
    m_DeviceType == BS_DEVICE_XPASS_SLIM ||  
    m_DeviceType == BS_DEVICE_XPASS_SLIM2 )  
  
{  
    BEUserHdr userHdr;  
    // Retrieve a user from the device  
    BS_RET_CODE result = BS_GetUserBEPlus( m_Handle, m UserID, &userHdr,  
    m_TemplateData );  
    ...  
  
    // Transfer the user to the device  
    result = BS_EnrollUserBEPlus( m_Handle, &userHdr, m_TemplateData );  
    ...  
}  
else if( m_DeviceType == BS_DEVICE_BIOSTATION )  
{  
    BSUserHdrEx userHdr;  
  
    BS_RET_CODE result = BS_GetUserEx( m_Handle, m UserID, &userHdr,  
    m_TemplateData );  
    ...  
  
    result = BS_EnrollUserEx( m_Handle, &userHdr, m_TemplateData );  
    ...  
}  
else if( m_DeviceType == BS_DEVICE_DSTATION )  
{  
    DSUserHdr userHdr;
```

```
...
BS_RET_CODE result = BS_GetUserDStation( m_Handle, m UserID, &userHdr,
m_TemplateData, m_FaceTemplate_DST );
...

result = BS_EnrollUserDStation( m_Handle, &userHdr, m_TemplateData,
m_FaceTemplate_DST );
}
else if( m_DeviceType == BS_DEVICE_XSTATION )
{
    XSUserHdr userHdr;
    ...

BS_RET_CODE result = BS_GetUserXStation( m_Handle, m UserID, &userHdr );
...
result = BS_EnrollUserXStation( m_Handle, &userHdr );
}
else if( m_DeviceType == BS_DEVICE_BIOSTATION2 )
{
    BS2UserHdr userHdr;
    ...

BS_RET_CODE result = BS_GetUserBioStation2( m_Handle, m UserID,
&userHdr, m_TemplateData );
...
result = BS_EnrollUserBioStation2( m_Handle, &userHdr, m_TemplateData );
}
else if( m_DeviceType == BS_DEVICE_FSTATION )
{
    FSUserHdr userHdr;
    ...

BS_RET_CODE result = BS_GetUserFStation( m_Handle, m UserID, &userHdr,
faceTemplate );
...
result = BS_EnrollUserFStation( m_Handle, &userHdr, m_FaceTemplate_FST
);
}
```

BioStar 2.x SDK

API

```
BS2UserBlob userBlob =
(BS2UserBlob)Utils.AllocateStructure(typeof(BS2UserBlob));
...
```

```
int result = (BS2ErrorCode)API.BS2_EnrolUser(Program.sdkContext,
deviceHandle.info.id, ref userBlob);
...
```

BioStar 1.x SDK

() API

```
int handle = ;
uint deviceID = ;
int deviceType = ;

result = BS_OpenSocket( "192.168.0.5", 1471, &handle );
result = BS_GetDeviceID(handle, &deviceID, &deviceType);
```

BioStar 2.x SDK

BioStar 2.x SDK () 가

ID

```
const char* deviceAddress = "192.168.1.2";
uint16_t devicePort = 51211;
uint32_t deviceId = ;
BS2SimpleDeviceInfo deviceInfo;

int result = BS2_ConnectDeviceViaIP(context, deviceAddress, devicePort,
&deviceId);
int result = BS2_GetDeviceInfo(context, deviceId, &deviceInfo);
```

BioStar 1.x SDK

API가

가

BioStar 2.x SDK

API

가

BioStar 1.x SDK

가 가

UI/

가

, BioStar 2.x SDK

UI/

가

가 가

UI

UI/

가 가

Visual Studio

C/C++

Under construction

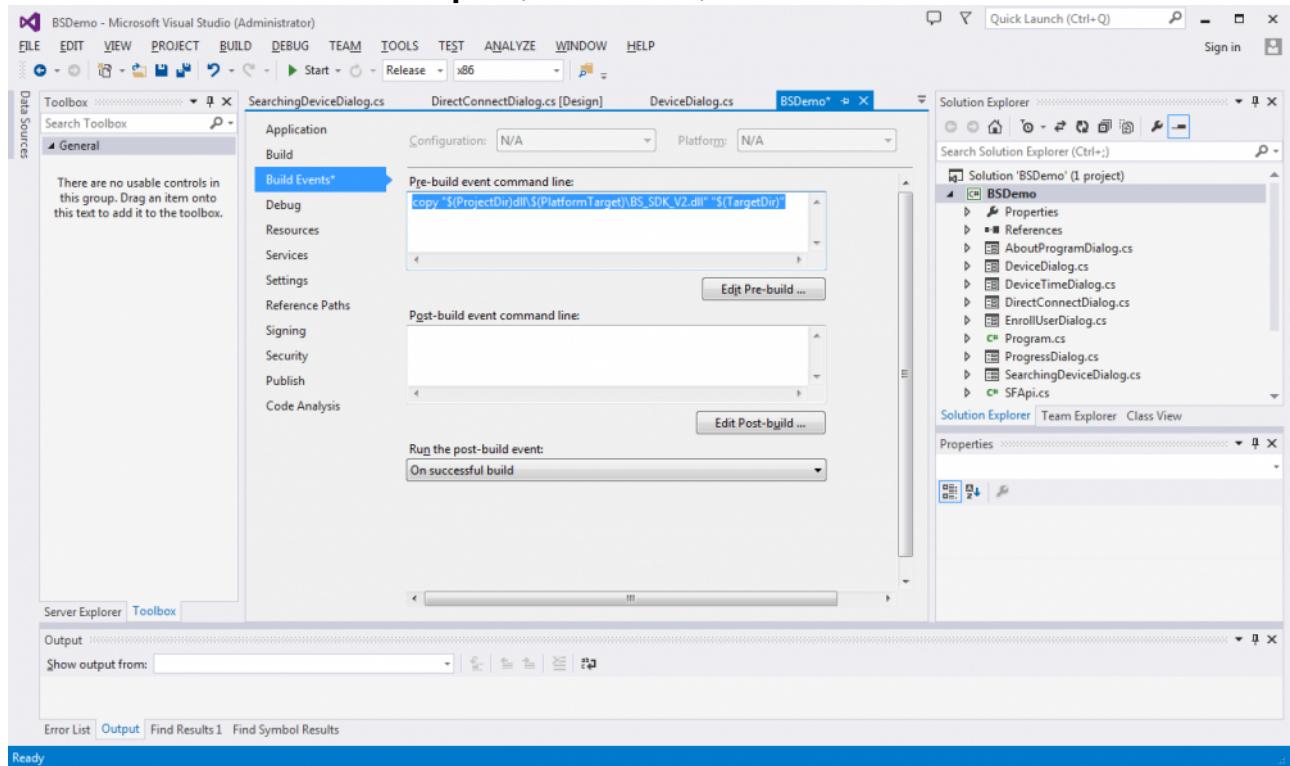
C#

1. SDK
2. DLL

```
copy "$(ProjectDir)lib\$(PlatformTarget)\BS_SDK_V2.dll" "$(TargetDir)"
```

3. SDK

SFApi.cs, SFEnum.cs, SFStruct.cs



1)

SDK

API

2)

API

3) 4)

C/C++

, C/C++

import

5)

SDK

가

.

From:

<https://kb.supremainc.com/kbtest/> - **BioStar 2 Device SDK**

Permanent link:

https://kb.supremainc.com/kbtest/doku.php?id=ko:getting_started&rev=1522913011

Last update: **2018/04/05 16:23**