
SIP Phone Client SDK

1.0

lxy

2014 09 06

1		1
2		3
2.1		3
2.2		3
2.3	API	5
3		13

SIP SDK Microsoft Windows SIP

-
-
-
-

SDKSIP

2.1

SIP SDK Microsoft Windows SIP

-
-
-
-

SDKSIP

2.2

Microsoft .NET C++/CLI .NET

Hesong.Sipua.dll

net/2.0/	.NET 2.0
net/4.0/	.NET 4.0 Full Pack
netcp/4.0/	.NET 4.0 Client Pack

2.2.1

- Windows XP SP3
- Windows Server 2003 SP2
- Windows Vista SP1
- Windows Server 2008
- Windows 7
- Windows Server 2008 R2
- Windows 7 SP1

- Windows Server 2008 R2 SP1

- x86
- **x64** Windows 64bit32bit

.Net Framework

- MMicrosoft .NET Framework 2.0 Service Pack 2
.NET Framework
- Microsoft .NET Framework 4 Client Profile
.NET Framework
- Microsoft .NET Framework 4
.NET Framework .NET Framework 4

Visual C++ Redistributable

- Microsoft Visual C++ 2005 Redistributable Package (x86)
Visual C++ Visual C++

: .NET 4.0
- Visual C++ Redistributable for Visual Studio 2012 Update 4
Visual Studio 2012 C++

: .NET 4.0

Apache log4net™

log4net.dll Hesong.Sipua.dll .NET

log4net 1.2.13

- Pentium 1 GHz 512 MB RAM
- 10 MB
-
-
-

2.2.2 Visual Studio

- Visual Studio 2005
- Visual Studio 2008
- Visual Studio 2010
- Visual Studio 2012
- Visual Studio 2013

2.3 API

C++/CLI API C++ C++/CLI C++

2.3.1 Hesong::Sipua::Lib

class Lib
SIP

Lib : Lib (unsigned int *sip_udp_port*, unsigned int *rtp_udp_port*)

- **sip_udp_port** – SIP UA SIP UDP 0 SIP 5060
- **rtp_udp_port** – SIP UA RTP UDP 0 1024 4000 RTCP RTP +1

SIP **Lib**

```
Lib : ~Lib ()  
    .NET
```

```
static Lib* Lib : :Instance
```

```
Lib::AccountList* Lib : :Accounts  
    SIP
```

AccountList

```
class Lib : :AccountList : public System::Collections::Generic::IList<Account>  
    Lib .NET System::Collections::Generic::IList List Account SIP
```

```
void Lib : :AccountList : :Add (Account* item)
```

```
    item
```

```
void Lib : :AccountList : :Clear ()
```

```
bool Lib : :AccountList : :Contains (Account* item)
```

```
    item
```

```
void Lib : :AccountList : :CopyTo (array<Account*>* array, int arrayIndex)
```

```
int Lib : :AccountList : :IndexOf (Account* item)
```

```
void Lib : :AccountList : :Insert (int index, Account* item)
```

```
bool Lib : :AccountList : :Remove (Account* item)
```

```
void Lib : :AccountList : :RemoveAt (int index)
```

```
int Lib : :AccountList : :Count
```

2.3.2 Hesong::Sipua::Account

```
class Account  
    SIP
```

```
Account : :Account (String* registerServer, String* project, String* userName, String* password, un-  
    signed int regTimeout, unsigned int regRetry, unsigned int regFirstRetry, String* la-  
    bel)
```

- **registerServer** – host[:port] 5060
- **project** –
- **userName** –
- **password** –
- **regTimeout** –
- **regRetry** –
- **regFirstRetry** –
- **label** –

Account : :**Account** (String* *registerServer*, String* *project*, String* *userName*, String* *password*, unsigned int *regTimeout*, unsigned int *regRetry*, unsigned int *regFirstRetry*)

Account : :**Account** (String* *registerServer*, String* *project*, String* *userName*, unsigned int *regTimeout*, unsigned int *retryInterval*, unsigned int *firstRetryInterval*, String* *label*)

Account : :**Account** (String* *registerServer*, String* *project*, String* *userName*, unsigned int *regTimeout*, unsigned int *retryInterval*, unsigned int *firstRetryInterval*)

Account : :**Account** (String* *registerServer*, String* *project*, String* *userName*, String* *password*, String* *label*)

Account : :**Account** (String* *registerServer*, String* *project*, String* *userName*, String* *label*)

Account : :**Account** (String* *registerServer*, String* *project*, String* *userName*)

~Account : :**Account** ()

int Account : :**Add** ()

: Lib::AccountList::Add ()

void Account : :**Del** ()

:

void Account : :**AnswerCall** ()

void Account : :**AnswerCall** (CallInfo* *callInfo*)

void Account : :**HangupCall** ()

void Account : :**HangupCall** (CallInfo* *callInfo*)

void Account : :**MakeCall** (String* *callee*)

callee

/ sip:123@host

void Account : :**SendDtmf** (String* *digits*)

digits DTFM

DTMF

`void Account::SendDtmf (CallInfo* callInfo, String* digits)`

- **callInfo** – DTMF
- **digits** – DTFM

DTMF

`void Account::Register ()`

`: Account::Add() Lib::AccountList::Add()`

`void Account::UnRegister ()`

`void Account::StartRereg ()`
488

`void Account::StopRereg ()`
`Account::StartRereg ()`

`RegInfo* Account::LastRegInfo`

`CallInfo* Account::CurrentCallInfo`

`int Account::Id`

ID

ID

`String* Account::UserId`

ID

:

"project_name" <sip:user_name@server_address>

`String* Account::RegUri`

URI

:

sip:host[:port]

`String* Account::Realm`

`String* Account::Project`

`String* Account::UserName`

`String* Account::Password`

`String* Account::Label`

`Boolean* Account::ReRegOn488`

status=488

false

`Object* Account::tag`

```

void Account::OnRegisterStateChanged (RegInfo* regInfo)
    regInfo
void Account::OnIncomingCall (CallInfo* info)
    info
    info CallInfo CallInfo::Answer() CallInfo::Hangup()
    Account::AnswerCall() Account::HangupCall()
void Account::OnCallStateChanged (CallInfo* regInfo)
    info

```

2.3.3 Hesong::Sipua::CallInfo

class CallInfo

```

CallInfo::CallInfo()
CallInfo::CallInfo (void* info)

```

```

~CallInfo::CallInfo()

```

```

void CallInfo::StartRecord()
    RecFile 256kB/s PCM
void CallInfo::StopRecord()

```

```

:
```

```

void CallInfo::Answer()
    Account::OnIncomingCall()
void CallInfo::Hangup()
    Account::OnIncomingCall()
    •
void CallInfo::SendDtmf (String* digits)
    digits DTFM
    DTMF

```

```
float CallInfo::InputVolume
    0
    2.0
: Windows Windows Windows
```

```
float CallInfo::OutputVolume
    0
    10.0
: Windows Windows Windows
```

```
int CallInfo::State
String* CallInfo::StateText
int CallInfo::LastStatus
    cause code
String* CallInfo::LastStatusText
String* CallInfo::RemoteInfo
String* CallInfo::RemoteInfoContact
    //
Boolean CallInfo::IsActive
String* CallInfo::RecFile
    CallInfo::StartRecord()
```

2.3.4 Hesong::Sipua::RegInfo

```
class RegInfo
```

```
RegInfo::RegInfo (void* info)
    info
```

```
~RegInfo::RegInfo (void* info)
    info
```

`int RegInfo::Code`

`String* RegInfo::Reason`

`int RegInfo::Expiration`

-
-
- *genindex*
 - *search*