

## General

The four methods in the IDL (Interface Definition Language) code belong to a COM object defined by an unknown class name or interface. The specification is incomplete as the information provided by the given code is not sufficient.

## Interfaces

### 1 Unknown

This Unknown interface allows ...

The following COM Method declarations are in this interface.

- 1.1 IsCallActive
- 1.2 PlayFile
- 1.3 Recognize
- 1.4 GetSlotVal

## COM Methods

### 1.1 Unknown::IsCallActive method

#### Description

This method will check the channel direction and will return the channel activity status.

#### C/C++ syntax

```
STDMETHOD (IsCallActive)(  
    HA_DIRECTION eDirection,  
    HA_CHANNEL_STATUS* peCallIsActive);
```

#### Parameters

HA_DIRECTION	eDirection
HA_CHANNEL_STATUS*	peCallIsActive

#### C/C++ return value

ACTIVE	This value is returned if the method call is successful
INACTIVE	This value is returned if the method call fails
SHUTDOWN	This value is returned if the method call is aborted

### 1.2 Unknown::PlayFile method

#### Description

This method plays the default message on an active incoming channel. This also checks for the pre- and post-message silence in the incoming channel.

#### C/C++ syntax

```
STDMETHOD (PlayFile)(  
    BSTR bstrFullName,
```

```
long lPreSilence,  
long lPostSilence,  
HA_PROMPT_FLAGS eFlags);
```

## Parameters

BSTR	bstrFullName
long	lPreSilence
long	lPostSilence
HA_PROMPT_FLAGS	eFlags

## Remarks

The pre- and post-silence of the incoming channel can be defined by the programmer.

## C/C++ return value

None

## 1.3 Unknown::Recognize method

### Description

This method tries to recognize the speech given as input in the incoming channel. The method checks for the user inactivity by prolonged silence. If the channel status remains active, the method flags the speech and checks the confidence level based on the GrammarNames (built-in dictionary). The method accordingly flags the recognition status and returns the same.

### C/C++ syntax

```
STDMETHOD (Recognize)(  
    long lTimeout,  
    HA_GETSTR_FLAGS eFlags,  
    VARIANT* pvarGrammarNames,  
    HA_GETSTR_STATUS* peStatus);
```

## Parameters

long	lTimeout
HA_GETSTR_FLAGS	eFlags
VARIANT*	pvarGrammarNames
HA_GETSTR_STATUS*	peStatus

## Remarks

This method is based on IVR (Interactive Voice Recognition)

## C/C++ return value

TIMEOUT	This value is returned on no speech activity for the predefined period
OK	This value is returned if there is speech activity and it is recognized
DTMF	This value is returned to convert the incoming channel into DTMF (Dial Tone Multi Frequency) mode for user menu interaction
NO CONFIDENCE	This value is returned if the speech recognition fails and does not match GrammarNames

### 1.4 Unknown::GetSlotVal method

#### Description

This method identifies the slot value and assigns confidence level to it based on speech recognition. A level 0 is considered 100% confident.

#### C/C++ syntax

```
STDMETHOD (GetSlotVal)(  
    BSTR bstrSlotName,  
    long lPosition,  
    long* plSlotConfidence,  
    BSTR* pbstrSlotValue);
```

#### Parameters

BSTR	bstrSlotName
long	lPosition
long*	plSlotConfidence
BSTR*	pbstrSlotValue

#### C/C++ return value

PERCENTAGE	This returns a percentage value of the speech recognition confidence level
REAL_VALUE	This returns the actual word of the speech from the GrammarNames