

WCFインタフェース定義

ソースコード

IScAPIs.cs

WCFサービスで公開しているインタフェース定義です[*20220118/u 以降

IScAPIs.cs

```
using System.Collections.Generic;
using System.ServiceModel;

namespace AssistantSeika
{
    [ServiceContract(SessionMode = SessionMode.Required)]
    public interface IScAPIs
    {
        /// <summary>
        /// AssistantSeikaのバージョンを返す
        /// </summary>
        /// <returns>バージョン文字列</returns>
        [OperationContract]
        string Verson();

        /// <summary>
        /// AssistantSeikaの製品スキャン実行
        /// </summary>
        [OperationContract]
        void ProductScan();

        /// <summary>
        /// PetitGateHttpwの起動実行
        /// </summary>
        [OperationContract]
        void BootHttpService();

        /// <summary>
        /// 利用可能話者の一覧を返す
        /// </summary>
        /// <returns>話者一覧</returns>
        [OperationContract]
        Dictionary<int, string> AvatorList();

        /// <summary>
        /// 少し詳しい情報を持った話者一覧を返す
        /// </summary>
    }
}
```

```
/// <returns>話者一覧</returns>
[OperationContract]
Dictionary<int, Dictionary<string, string>> AvatorList2();

/// <summary>
/// 更に詳しい情報を持った話者一覧を返す
/// </summary>
/// <returns>話者一覧</returns>
[OperationContract]
Dictionary<int, Dictionary<string, string>>
AvatorListDetail2();

/// <summary>
/// デフォルトのパラメタ情報を得る
/// </summary>
/// <param name="cid">話者</param>
/// <returns>デフォルトのパラメタ情報</returns>
[OperationContract]
Dictionary<string, Dictionary<string, Dictionary<string,
decimal>>> GetDefaultParams2(int cid);

/// <summary>
/// 現在のパラメタ情報を得る
/// </summary>
/// <param name="cid">話者</param>
/// <returns>現在のパラメタ情報</returns>
[OperationContract]
Dictionary<string, Dictionary<string, Dictionary<string,
decimal>>> GetCurrentParams2(int cid);

/// <summary>
/// パラメタを初期化する
/// </summary>
/// <param name="cid">話者</param>
[OperationContract]
void ResetParams2(int cid);

/// <summary>
/// 発声する
/// </summary>
/// <param name="cid">話者</param>
/// <param name="talktext">発声させるテキスト</param>
/// <param name="filepath">音声を保存するファイル名 (フルパス)
ス</param>
/// <param name="effects">適用する音声効果情報</param>
/// <param name="emotions">適用する感情情報</param>
/// <returns>再生時間 (ミリ秒) </returns>
[OperationContract]
double Talk(int cid, string talktext, string filepath,
```

```
Dictionary<string, decimal> effects, Dictionary<string, decimal>
emotions);

    /// <summary>
    /// 発声する
    /// </summary>
    /// <param name="cid">話者</param>
    /// <param name="talktexts">発声テキストの配列</param>
    /// <param name="filepath">音声を保存するファイル名 (フルパ
ス</param>
    /// <param name="effects">適用する音声効果情報</param>
    /// <param name="emotions">適用する感情情報</param>
    /// <returns>再生時間 (ミリ秒) </returns>
    [OperationContract]
    double Talk2(int cid, string[] talktexts, string filepath,
Dictionary<string, decimal> effects, Dictionary<string, decimal>
emotions);

    /// <summary>
    /// 非同期に発声する
    /// </summary>
    /// <param name="cid">話者</param>
    /// <param name="talktext">発声させるテキスト</param>
    /// <param name="effects">適用する音声効果情報</param>
    /// <param name="emotions">適用する感情情報</param>
    [OperationContract]
    void TalkAsync(int cid, string talktext, Dictionary<string,
decimal> effects, Dictionary<string, decimal> emotions);

    /// <summary>
    /// 非同期に発声する
    /// </summary>
    /// <param name="cid">話者</param>
    /// <param name="talktexts">発声テキストの配列</param>
    /// <param name="effects">適用する音声効果情報</param>
    /// <param name="emotions">適用する感情情報</param>
    [OperationContract]
    void TalkAsync2(int cid, string[] talktexts, Dictionary<string,
decimal> effects, Dictionary<string, decimal> emotions);
    }
}
```

技術資料, Windows, AssistantSeika, WCF, DLL

From:
<https://wiki.hgotoh.jp/> - 努力したWiki

Permanent link:
<https://wiki.hgotoh.jp/documents/tools/assistantseika/interface/wcf/wcf-001>

Last update: 2023/11/05 07:38



