



工業技術研究院

Industrial Technology
Research Institute

資料名稱： 文字轉語音網頁應用服務安裝與開發指南

版別：第 1.4 版

文字轉語音網頁應用服務 安裝與開發指南

版 別：第 1.4 版

文件編號： 528A00104



1. 技術項目簡介	3
2. 應用範圍說明	3
3. ITRI TTS@Web 網頁介紹	3
3.1. 服務簡介	3
3.1.1. 開發流程	4
3.2. 註冊使用者帳號	4
3.3. 開發專區	4
3.4. 查詢合成狀態	4
4. Web Service API 程式開發指南	4
4.1. 基本開發環境需求	4
4.2. 申請合成帳號、密碼及 License Code	4
4.3. Web Service API 說明與範例	4
4.3.1. Web Service API 架構圖	4
4.3.2. Web Service API 傳回值說明	4
4.3.3. Web Service API 說明	4
5. TTS Web 應用程式快速開發指南	4
5.1. 說明	4
5.2. 使用 Web Service Tools	4
5.2.1. Web Service Tools 架構圖	4
5.2.2. 使用環境	4
5.2.3. 安裝說明	4
5.2.4. 設置說明	4
5.3. 使用 JAVASCRIPT API	4
5.3.1. JAVASCRIPT API 架構圖	4
5.3.2. 使用環境	4
5.3.3. 取得 license key	4
5.3.4. JavaScript API 說明	4
5.3.5. JavaScript API 範例	4



1. 技術項目簡介

經由簡單物件存取協定(SOAP，全寫為 Simple Object Access Protocol，可提供在不同應用程式、平台與硬體中交換資料的 XML 格式，而不需格式化介面)，可讓使用者在自己的網頁中提供文字轉語音體驗。

2. 應用範圍說明

本技術可應用於使用 PHP 或 ASP 的 Web 應用程式，亦可利用 JAVASCRIPT API 使用於純 html 網頁中，ITRI TTS@Web(ITRI TTS Web Service)架構圖如下圖所示：



■ 註：→ 使用 SOAP (Simple Object Access Protocol) 傳送

ITRI TTS@Web 情境示意圖

3. ITRI TTS@Web 網頁介紹

以下為 ITRI TTS@Web 的網頁介紹(<http://tts.itri.org.tw/>)。

3.1. 服務簡介

工研院文字轉語音 (TTS: Text To Speech) web 服務 (web service) 可讓您在自己的網頁提供 TTS 體驗。可使用 PHP、ASP、Javascript 等進行應用網頁撰寫。本服務使用 UTF-8 之格式，伺服



器提供 SOAP (Simple Object Access Protocol) 協議的 Web Service。

所謂的文字轉語音，是將所輸入的文字，轉換為語音進行輸出。早期的文字轉語音，具有機械音，韻律不流暢的缺點。目前所研發的高音質中文文字轉語音技術，所合成的聲音自然流暢近似真人發音。

工研院
文字轉語音 Web 服務
ITRI TTS @ Web

認識我們 | 開發專區 | 線上語音 | 會員登錄

讓網頁聲動起來!

服務簡介

帳號 密碼 登入

加入會員 | 會員權益 | 忘記密碼

廣告看板

- ITRI TTS@web特色
- 成功範例
- DM/EDM下載或列印
- 下載開發指南
- ITRI TTS@Web技術入選樂活百年搶鮮大賽
- 工研院語音技術推廣網

常見問答

Q: 有開發上的問題時，該如何聯絡？
A: 本站整理了綜整諮詢、技術移轉、網站內容的連絡代表窗口，詳細聯絡方式請參考連絡窗口。[MORE](#)

音量大小： 100

中英統合 Bruce (男) 中英切換 Bruce (男) 英文語音 Bob (男) 台語語音 A-Koan (女) 文讀台

輸入您想轉換為語音輸出的中文文字，可以混合簡體與繁體喔！(250字以內)
Please type any text which you wish to be spoken here. (under 250 words)

[試聽](#)

版權所有© 2011 工業技術研究院 資訊與通訊研究所 | 連絡窗口 | 建議瀏覽解析度1024x768以上

3.1.1. 開發流程

- 註冊帳號後，請至 E-mail 收取認證信。
- 閱讀開發專區-> Web service API，了解如何使用 TTS Web service API。
- 在自己的網頁中加入 api，呼叫 TTS Web service API，建立自己 TTS 服務。
- 可至線上語音->進階合成下載頁面，查詢合成狀態。所有資料與音檔保留一天即刪除

3.2. 註冊使用者帳號

網頁位置<http://tts.itri.org.tw/member/registration.php>，提供註冊合成帳號、密碼。



工業技術研究院

Industrial Technology
Research Institute

資料名稱： 文字轉語音網頁應用服務安裝與開發指南

版別：第 1.4 版

工研院
文字轉語音 Web 服務
ITRI TTS @ Web

認識我們 | 開發專區 | 線上語音 | **會員登錄**

首頁 > 會員登錄 > 註冊帳號

新增個人資料

謝謝您的參與和支持！若您對於會員權益或服務有任何疑問，歡迎連絡我們。

* 為必填欄位

* 帳號：

* 密碼：

* 確認密碼：

* 電子郵件信箱：

* 姓名：

* 單位：

* 驗證碼： 283293 [重新產生](#)

* 服務條款：

- 本試用服務產生之每一筆合成語音資料，於本站上只保留一天；亦即每一筆合成語音資料於合成產生24小時後，將從本站上刪除。
- 資料提供義務：會員應誠實登錄資料、告知應用內容、填寫試用問卷調查。
- 來源標示義務：本服務有權要求會員於其應用網頁或系統適當位置處，進行標示「工研院文字轉語音試用服務產出之合成語音」。

確認並同意以上條款

[送出](#) [清除](#)



版權所有© 2011 工業技術研究院 資訊與通訊研究所 | 連結窗口 | 建議瀏覽解析度1024x768以上

3.3. 開發專區

網頁位置 <http://tts.itri.org.tw/development/manual.php>，提供 TTS Web service API、TTS Web service Tool、TTS Web service javascript API 詳細說明。



The screenshot shows the homepage of the ITRI TTS @ Web service. At the top left is the ITRI logo and the text "文字轉語音 Web 服務" (Text-to-Speech Web Service) with "ITRI TTS @ Web". To the right are links for "認識我們" (About Us), "開發專區" (Development Zone), "線上語音" (Online Voice), and "會員登錄" (Member Login). Below these are several colorful speech bubble icons. The main content area is titled "Web Service API" and includes a table comparing "應用特色" (Application Features) for different service types.

使用SOAP通訊協定傳輸	直接	TTS download程式碼	不需要
操控自由度	高	Domain name	不需要
合成音檔下載處理	自行處理	License Code	不需要

首頁 > 開發專區 > Web Service API

Web Service API

應用特色

使用SOAP通訊協定傳輸	直接	TTS download程式碼	不需要
操控自由度	高	Domain name	不需要
合成音檔下載處理	自行處理	License Code	不需要

簡介

本服務使用UTF-8之格式，提供SOAP (Simple Object Access Protocol) 協議的TTS Web Service Servers，並提供合成音檔下載網址。



每個API提供函式(ConvertSimple, ConvertText, ConvertAdvancedText)的傳回值皆為String型態，其回傳值格式為resultCode&resultString&result，以`&`區隔開來，唯有resultCode等於0(Success)時，才會有result的值產生。下方表格（表一）為執行狀況代碼(resultCode)與說明(resultString)之對照值。

另外，在GetConvertStatus函式中，提供TTS合成的狀態，其格式為 resultCode&resultString&statusCode &status&result，以`&`區隔開來，當statusCode為2(completed)時，才會有 result的值產生。下方表格（表二）為狀態代碼(statusCode)與說明(status)之對照值。

(表一)

resultCode	resultString
0	Success
-1	No matched speaker
-2	Voice setting fail
-3	Output file setting fail
-4	invalid login
-5	invalid TTS Font Format
-6	out of service space
-7	account incative
-8	password incative
-9	incative TTSSpeaker

(表二)

statusCode	status
0	queued
1	processing
2	completed
3+	No matched speaker

TTS Web service API



工業技術研究院

Industrial Technology
Research Institute

資料名稱： 文字轉語音網頁應用服務安裝與開發指南

版別：第 1.4 版

The screenshot shows the homepage of the ITRI TTS @ Web service. At the top, there's a navigation bar with links for '認識我們' (About Us), '開發專區' (Development Zone), '線上語音' (Online Voice), and '會員登錄' (Member Login). Below the navigation is a breadcrumb trail: 首頁 > 開發專區 > Web Service Tools. The main content area is titled 'Web Service Tools' and includes a table comparing different features:

使用 SOAP 通訊協定傳輸	間接	TTS download 程式需	需要
操控自由度	中	Domain name	不需要
合成音檔下載處理	Tools 協助處理	License Code	不需要

Below the table is a section titled '簡介' (Introduction) with the text: '本服務提供程式開發者對於一般網頁訊息發佈時，可以網頁上聆聽合成效果並且將音檔下載至指定資料夾。'

The diagram titled 'Web Service Tools 架構示意圖' (Architecture Diagram) illustrates the system architecture. It shows a 'Web Server' (represented by a blue server icon) and a 'TTS Server' (represented by a black server icon). The 'Web Server' sends requests to the 'TTS Server' via double-headed arrows labeled with method names: 'TTS.ConverterIndex', 'TTS.PlayerSet.hidden', 'TTS.PlayerSet.width', 'TTS.playerSet.Height', 'TTS.Audiofilename', and 'TTS.ConvertInit'. The 'TTS Server' also has a double-headed arrow labeled 'DB' (Database) pointing to it.

Legend for the arrows:
—> 同時設定參數(已給預設值)及使用合成函式，並在網頁上直接呈現
—> 撥放控制列

成功範例

新聞類網頁應用（請點選網頁中最左下方的喇叭圖示 ，可立即聆聽網頁內容的撥放。）

應用教學

- Step1. 取得帳號/密碼

TTS Web service Tool



The screenshot shows the homepage of the ITRI TTS @ Web service. At the top, there's a navigation bar with links for '認識我們' (About Us), '開發專區' (Development Zone) in orange, '線上語音' (Online Voice), and '會員登錄' (Member Login). Below the navigation is a breadcrumb trail: '首頁 > 開發專區 > JavaScript API'. On the left, there's a sidebar with a 'JavaScript API' section and a '應用特色' (Application Features) table. The table compares three methods: SOAP, REST, and JavaScript. The JavaScript row shows: 操控自由度 (High, limited by browser), Domain name (Required), and License Code (Required). The main content area contains a diagram titled 'JavaScript API 架構示意圖' (Architecture Diagram) showing the interaction between a Web Server, TTS Server, and Database (DB).

JavaScript API

應用特色

使用SOAP通訊協定傳輸	間接	TTS download程式碼	不需要
操控自由度	低(限定Blog)	Domain name	需要
合成音檔下載處理	無	License Code	需要

簡介
本服務提供JavaScript範例程式供一般使用者使用於html環境(非php、asp等)下使用。

Web Server

TTS Server

DB

TTS.muteTag
TTS.PlayerSet.hidden
TTS.PlayerSet.width
TTS.playerSet.Height

TTS.ConvertInit
TTS.ConvertCustom

同時設定參數(已給預設值)及使用合成函式，並在網頁上直接呈現

撥放控制列

JavaScript API 架構示意圖

應用教學

- Step1. 取得帳號/密碼
- Step2. 做網址設定和取得License Key
- Step3. 複製代碼
- Step4. 設定參數(已給預設值)
- Step5. 使用合成函式(TTS.ConvertInit, TTS.ConvertCustom)
- Step6. 在網頁上直接呈現撥放控制列，點喇叭圖示後可以馬上做語音合成進行播放

TTS Web service javascript API

3.4. 查詢合成狀態

在使用合成函式過後，可至http://tts.itri.org.tw/online_tts/advance_n_download.php查詢合成狀態



工業技術研究院

Industrial Technology
Research Institute

資料名稱： 文字轉語音網頁應用服務安裝與開發指南

版別：第 1.4 版

工研院
文字轉語音 Web 服務
ITRI TTS @ Web

認識我們 | 開發專區 | 線上語音 | 會員登錄

首頁 > 瀏覽權益登錄

瀏覽權益登錄

很抱歉！本項目僅開放會員使用！瀏覽前，請先進行會員登錄程序，謝謝！

帳號：

密碼：

登入 清除

登入後即可查看

新增

ID	合成時間	語者	音量	速度/單位/震幅/特殊	文字內容	狀態
23813	2011-12-29 09:17:37	Bruce	100	0/0/5/正常	請輸入文字內容	132512 145863 61.wav
23812	2011-12-29 09:16:57	MCHEN_Bruce	100	0/0/5/正常	輸入您想轉換為語音輸出的中文文字，可以混合簡體與繁體喔！(250字以內) Please type any text which you wish to be spoken here. (under 250 words)	132512 141828 564.wav
23811	2011-12-29 09:16:45	MCHEN_Bruce	100	0/0/5/正常	輸入您想轉換為語音輸出的中文文字，可以混合簡體與繁體喔！(250字以內) Please type any text which you wish to be spoken here. (under 250 words)	132512 140618 213.wav
23810	2011-12-29 09:16:32	MCHEN_Bruce	100	0/0/5/正常	輸入您想轉換為語音輸出的中文文字，可以混合簡體與繁體喔！(250字以內) Please type any text which you wish to be spoken here. (under 250 words)	132512 139330 880.wav
23809	2011-12-29 09:13:23	MCHEN_Bruce	100	0/0/5/正常	輸入您想轉換為語音輸出的中文文字，可以混合簡體與繁體喔！(250字以內) Please type any text which you wish to be spoken here. (under 250 words)	132512 120516 751.wav
23808	2011-12-29 09:10:38	MCHEN_Bruce	100	0/0/5/正常	輸入您想轉換為語音輸出的中文文字，可以混合簡體與繁體喔！(250字以內) Please type any text which you wish to be spoken here. (under 250 words)	132512 104062 99.wav
23799	2011-12-29 08:53:50	MCHEN_Bruce	100	0/0/5/正常	輸入您想轉換為語音輸出的中文文字，可以混合簡體與繁體喔！(250字以內) Please type any text which you wish to be spoken here. (under 250 words)	132512 003120 371.wav



4. Web Service API 程式開發指南

4.1. 基本開發環境需求

需使用在支援 PHP 或 ASP 的 Web 伺服器。若伺服器使用 ASP 需先安裝 SOAP Toolkit 3.0 (<http://www.microsoft.com/downloads/en/details.aspx?FamilyID=c943c0dd-ceec-4088-9753-86f052ec8450>)

使用者在使用或開發者在測試時，可使用 IE、Google Chrome、Firefox、Opera 任一瀏覽器。

4.2. 申請合成帳號、密碼及 License Code

至網頁[註冊頁面](#)註冊，在[修改個人資料](#)取得 Lincense Code:

4.3. Web Service API 說明與範例

4.3.1. Web Service API 架構圖



-----> 該函式提供輸入預合成文字與合成參數設定，輸出合成編號(Convert ID)

-----> 該函式提供輸入合成編號，輸出目前合成的狀態，若狀態為合成完畢，將提供合成音檔下載網址

Web Service API 架構示意圖

4.3.2. Web Service API 傳回值說明

每個 API 的傳回值皆為 String 型態，其回傳值格式為 resultCode&resultString&result，以“&”區隔開來，當 resultCode 等於 0(Success)時，才會有 result 的值產生(ConvertID)。表一為執行狀況



代碼(resultCode)與說明表(resultString)，說明如下：

resultCode	resultString
0	Success
-1	No matched speaker
-2	Voice setting fail
-3	Output file setting fail
-4	invalid login
-5	invalid TTS Font Format
-6	out of service space
-7	account incative
-8	password incative
-9	incative TTSSpeaker
-10	incative Volume
-11	incative Speed
-12	invalid Text
-13	content size is too large
-14	inactive ConvertID
-15	inactive outputType
-16	inactive PitchModify

表一 執行狀況代碼與說明表

另外，在 **GetConvertStatus** 函式中，額外提供 TTS 合成的狀態，其回傳值格式為 resultCode&resultString&statusCode&status&result，以“&”區隔開來，當 statusCode 為 2(completed)時，才會有 result 的值產生。表二為 TTS 合成進度代碼(statusCode)與說明表(status)，說明如下：

statusCode	status
0	queued
1	processing
2	completed
3	No matched speaker

表二 TTS 合成進度代碼與說明表



TSSpeaker	explain
Bruce	中英切換男生語音(default)
Theresa	中英切換女生語音
Angela	中英切換小女孩語音
MCHEN_Bruce	中英統合男生語音
MCHEN_Joddess	中英統合女生語音
ENG_Bob	英文男生語音
ENG_Alice	英文女生語音
ENG_Tracy	英文小男孩語音
TW_LIT_AKoan	台語女生語音(文讀台)
TW_SPK_AKoan	台語女生語音(白話台)

表三 合成語者列表

4.3.3. Web Service API 說明

函式名稱	ConvertSimple
函式說明	不需額外選擇，直接提供固定語者(預設為 Bruce)、固定音量(預設為 100)與固定語音速度(預設為 0)
輸入參數	Account(String)—輸入您申請的帳號 Password(String)—輸入您申請的密碼 TTSText(String)—輸入欲合成的 TTS 文字
函式範例	ConvertSimple ("ITRI","ITRI Password","歡迎使用文字轉語音合成技術")
回傳參數	Result(String) – resultCode&resultString&result resultCode —回傳執行狀況代碼(參考表一) resultString —回傳執行狀況說明(參考表一) result —當 resultCode 為 0、resultString 為 Success 時，回傳檔案之 ConvertID
回傳範例	0&Success&20
PHP 範例	<?php //Setup Web Service \$client = new SoapClient("http://tts.itri.org.tw/TTSService/SOAP_1_3.php?wsdl"); // Invoke Call to ConvertSimple \$result=\$client-> ConvertSimple ("Account","Password","合成文字位置"); // Iterate through the returned string array \$resultArray=explode("&",\$result); list(\$resultCode, \$resultString, \$resultConvertID) = \$resultArray;



	<pre>echo "resultCode : ".\$resultCode."
"; echo "resultString : ".\$resultString."
"; echo "resultConvertID : ".\$resultConvertID."
"; ?></pre>
ASP 範例	<pre><% 'Setup Web Service Set objSoapClient = Server.CreateObject("MSSoap.SoapClient30") objSoapClient.ClientProperty("ServerHTTPRequest") = true objSoapClient.MSSoapInit ("http://tts.itri.org.tw/TTSService/SOAP_1_3.php?wsdl") 'Invoke Call to ConvertSimple dim result result = objSoapClient.ConvertSimple("Account","Password","合成文 字位置") 'Iterate through the returned string array resultArray = Split(result,"&") dim resultArray dim resultNameArray(3) resultNameArray(0) = "resultCode : " resultNameArray(1) = "resultString : " resultNameArray(2) = "resultConvertID : " for i=0 to UBound(resultArray) response.write resultNameArray(i)+resultArray(i)+"
" next Set objSoapClient = Nothing %></pre>
Android 範例	<pre>String methodName = "ConvertSimple"; String nameSpace = "http://tts.itri.org.tw/TTSService/"; String soapAction = nameSpace+methodName; SoapObject rpc = new SoapObject(nameSpace, methodName); rpc.addProperty("accountID", "Account"); rpc.addProperty("password", "Password"); rpc.addProperty("TTStext", "合成文字位置"); SoapSerializationEnvelope envelope = new SoapSerializationEnvelope(SoapEnvelope.VER11); envelope.bodyOut = rpc; envelope.dotNet = true; envelope.setOutputSoapObject(rpc); HttpTransportSE transport = new HttpTransportSE(endPoint); transport.debug = true; Object obj = null; try { transport.call(soapAction, envelope); obj = envelope.getResponse(); }</pre>



	<pre>result = obj.toString(); Log.i("shiaukai", "result:" + obj.toString()); //resultTV.setText(result); String[] retAry = result.split("&"); String status = retAry[0]; Log.i("shiaukai", status); if(status.equals("0")){ convertID = Integer.valueOf(retAry[2]); } Message msg = new Message(); msg.what=SoapExActivity.RESULT; SoapExActivity.this.handler.sendMessage(msg); //return obj.toString(); } catch (Exception e) { e.printStackTrace(); }</pre>
IOS 範例	<pre>NSString *accountID = @"Account"; NSString *password = @"Password"; NSString *TTStext = @"合成文字位置"; //建立 SOAP Message NSString *soapMessage = [NSString stringWithFormat: @"<?xml version=\"1.0\" encoding=\"utf-8\"?>\n" "<soap12:Envelope >\n" " xmlns:xsi=\"http://www.w3.org/2001/XMLSchema-instance\" " " xmlns:xsd=\"http://www.w3.org/2001/XMLSchema\" " " xmlns:soap12=\"http://www.w3.org/2003/05/soap-envelope\">\n" "<soap12:Body>\n" "<ConvertSimple xmlns=\"http://tts.itri.org.tw/\">\n" "<accountID>%@</accountID>\n" "<password>%@</password>\n" "<TTStext>%@</TTStext>\n" "</ConvertSimple>\n" "</soap12:Body>\n" "</soap12:Envelope>", accountID, password, TTStext]; //建立 URL Request 時需要的 NSERL 參數 NSURL *url = [NSURL URLWithString:@"http://tts.itri.org.tw/TTSService/SOAP_1_3.php?wsdl"]; //建立 URL Request NSMutableURLRequest *urlRequest=[NSMutableURLRequest requestWithURL:url]; //SOAP Message 長度， web service 需要 NSString *msgLength = [NSString stringWithFormat:@"%d", [soapMessage length]]; [urlRequest addValue: @"application/soap+xml; charset=utf-8" forHTTPHeaderField:@"Content-Type"]; //內容形態 [urlRequest addValue: msgLength forHTTPHeaderField:@"Content-Length"]; //SOAP message 長度</pre>



```
[urlRequest setHTTPMethod:@"POST"];
[urlRequest setHTTPBody: [soapMessage
dataUsingEncoding:NSUTF8StringEncoding]];
conn = [[NSURLConnection alloc] initWithRequest:urlRequest delegate:self];
if (conn) {
    webData = [NSMutableData data];
}
```

函式名稱	ConvertText
函式說明	提供選擇不同語者(speaker)之聲音、並可進行音量調整與語音速度調整
輸入參數	Account(String)—輸入您申請的帳號 Password(String)—輸入您申請的密碼 TTSText(String)—輸入欲合成的 TTS 文字 TTSSpeaker(String)—可選擇語者有 Bruce、Theresa 與 Angela Volume(Integer)—其音量大小可調整範圍為 0~100 Speed(Integer)—其語音速度可調整範圍為 -10~10 OutputType(String) —輸入聲音輸出之格式(wav、flv)
函式範例	ConvertText ("ITRI","ITRI Password","歡迎使用文字轉語音合成技術","Angela",50,5,"wav")
回傳參數	Result(String) – resultCode&resultString&result resultCode —回傳執行狀況代碼(參考表一) resultString —回傳執行狀況說明(參考表一) result —當 resultCode 為 0、resultString 為 Success 時，回傳檔案之 ConvertID
回傳範例	0&Success&21
PHP 範例	<?php //Setup Web Service \$client = new SoapClient("http://tts.itri.org.tw/TTSService/Soap_1_3.php?wsdl"); // Invoke Call to ConvertText \$result=\$client-> ConvertText ("Account","Password","合成文字位置 ","Theresa",100, 0, "flv"); // Iterate through the returned string array



	<pre>\$resultArray=explode("&",\$result); list(\$resultCode, \$resultString, \$resultConvertID) = \$resultArray; echo "resultCode : ".\$resultCode."
"; echo "resultString : ".\$resultString."
"; echo "resultConvertID : ".\$resultConvertID."
"; ?></pre>
ASP 範例	<pre><% 'Setup Web Service Set objSoapClient = Server.CreateObject("MSSoap.SoapClient30") objSoapClient.ClientProperty("ServerHTTPRequest") = true objSoapClient.MSSoapInit ("http://tts.itri.org.tw/TTSService/SOAP_1_3.php?wsdl") 'Invoke Call to ConvertText dim result result = objSoapClient.ConvertText("Account","Password","合成文字位置 ","Theresa",100,0, "flv") 'Iterate through the returned string array resultArray = Split(result,"&") dim resultArray dim resultNameArray(3) resultNameArray(0) = "resultCode : " resultNameArray(1) = "resultString : " resultNameArray(2) = "resultConvertID : " for i=0 to UBound(resultArray) response.write resultNameArray(i)+resultArray(i)+"
" next Set objSoapClient = Nothing %></pre>
Android 範例	<pre>String methodName = "ConvertText"; String nameSpace = "http://tts.itri.org.tw/TTSService/"; String soapAction = nameSpace+methodName; Log.i("shiaukai", soapAction); SoapObject rpc = new SoapObject(nameSpace, methodName); rpc.addProperty("accountID", "Account"); rpc.addProperty("password", "Password"); rpc.addProperty("TTStext", "TTSText"); rpc.addProperty("TTSSpeaker", "TTSSpeaker"); rpc.addProperty("volume", "Volume"); rpc.addProperty("speed", "Speed"); rpc.addProperty("outType", "OutputType"); SoapSerializationEnvelope envelope = new SoapSerializationEnvelope(SoapEnvelope.VER11); envelope.bodyOut = rpc; envelope.dotNet = true; envelope.setOutputSoapObject(rpc);</pre>



```
HttpTransportSE transport = new HttpTransportSE(endPoint);
transport.debug = true;
Object obj = null;
try {
    transport.call(soapAction, envelope);
    obj = envelope.getResponse();
    result = obj.toString();
    Log.i("shiaukai", "result:" + obj.toString());
    String[] retAry = result.split("&");
    String status = retAry[0];
    Log.i("shiaukai", status);
    if(status.equals("0")){
        convertID = Integer.valueOf(retAry[2]);
    }
    Message msg = new Message();
    msg.what = SoapExActivity.RESULT;
    SoapExActivity.this.handler.sendMessage(msg);
} catch (Exception e) {
    e.printStackTrace();
}
```

IOS 範例	<pre>NSString *accountID = @"Account"; NSString *password = @"Password"; NSString *TTStext = @"合成文字位置"; NSString *TTSSpeaker = @"Angela"; int volume = 100; //Range 0~100 default=100 int speed = 0; //Range -10~10 default=0 NSString *outType = @"wav";// or flv //建立 SOAP Message NSString *soapMessage = [NSString stringWithFormat: @"<?xml version=\"1.0\" encoding=\"utf-8\"?>\n" "<soap12:Envelope >\n" " xmlns:xsi=\"http://www.w3.org/2001/XMLSchema-instance\" " " xmlns:xsd=\"http://www.w3.org/2001/XMLSchema\" " " xmlns:soap12=\"http://www.w3.org/2003/05/soap-envelope\">\n" "<soap12:Body>\n" " <ConvertText xmlns=\"http://tts.itri.org.tw/\">\n" " <accountID>%@</accountID>\n" " <password>%@</password>\n" " <TTStext>%@</TTStext>\n" " <TTSSpeaker>%@</TTSSpeaker>\n" " <volume>%i</volume>\n" " <speed>%i</speed>\n" " <outType>%@</outType>\n" " </ConvertText>\n" "</soap12:Body>\n"]</pre>
---------------	---



```
"</soap12:Envelope>",accountID,password,TTStext,TTSSpeaker,volume,speed,outType];  
NSLog(@"%@",soapMessage);  
//建立 URL Request 時需要的 NSERL 參數  
NSURL *url = [NSURL  
URLWithString:@"http://tts.itri.org.tw/TTSService/SOAp_1_3.php?wsdl"];  
//建立 URL Request  
NSMutableURLRequest *urlRequest=[NSMutableURLRequest requestWithURL:url];  
//SOAP Message 長度， web service 需要  
NSString *msgLength = [NSString stringWithFormat:@"%d", [soapMessage length]];  
[urlRequest addValue: @"application/soap+xml; charset=utf-8"  
forHTTPHeaderField:@"Content-Type"];//內容形態  
[urlRequest addValue: msgLength forHTTPHeaderField:@"Content-Length"];//SOAP  
message 長度  
[urlRequest setHTTPMethod:@"POST"];  
[urlRequest setHTTPBody: [soapMessage dataUsingEncoding:NSUTF8StringEncoding]];  
conn = [[NSURLConnection alloc] initWithRequest:urlRequest delegate:self];  
if (conn) {  
    webData = [NSMutableData data];  
}
```

函式名稱	ConvertAdvancedText
函式說明	提供選擇不同語者(speaker)之聲音、並可進行音量調整、語音速度調整與韻律調整
輸入參數	Account(String)—輸入您申請的帳號 Password(String)—輸入您申請的密碼 TTSText(String)—輸入欲合成的 TTS 文字 TTSSpeaker(String)—可選擇語者有 Bruce、Theresa 與 Angela Volume(Integer)—其音量大小可調整範圍為 0~100，預設值為 100 Speed(Integer)—其語音速度可調整範圍為-10~10，預設值為 0 OutputType(String) –輸入聲音輸出之格式(wav、flv) PitchLevel(Integer) –韻律調整：值越大則音高越高；反之則音高越低，可調整範圍-10~10，預設值為 0 PitchSign(Integer) –韻律調整：0=正常、1=像機器人、2=像外國人說中文，預設值為 0 PitchScale(Integer) –韻律調整：值越大則抑揚頓挫越明顯；反之則越趨平版，可調整範圍 0~20
函式範例	ConvertAdvancedText("ITRI","ITRI Password","歡迎使用文字轉語音合成技術","Angela",50,5,"wav")



回傳參數	Result(String) – resultCode&resultString&result resultCode —回傳執行狀況代碼(參考表一) resultString —回傳執行狀況說明(參考表一) result —當 resultCode 為 0、resultString 為 Success 時，回傳檔案之 ConvertID
回傳範例	0&Success&20
PHP 範例	<?php //Setup Web Service \$client = new SoapClient("http://tts.itri.org.tw/TTSService/SOAP_1_3.php?wsdl"); // Invoke Call to ConvertText \$result=\$client-> ConvertAdvancedText ("Account","Password","合成文字位置","Theresa",10 "flv",5,1,10); // Iterate through the returned string array \$resultArray=split("&",\$result); list(\$resultCode, \$resultString, \$resultConvertID) = \$resultArray; echo "resultCode : ".\$resultCode." "; echo "resultString : ".\$resultString." "; echo "resultConvertID : ".\$resultConvertID." "; ?>
ASP 範例	<% 'Setup Web Service Set objSoapClient = Server.CreateObject("MSSoap.SoapClient30") objSoapClient.ClientProperty("ServerHTTPRequest") = true objSoapClient.MSSoapInit ("http://tts.itri.org.tw/TTSService/SOAP_1_3.php?wsdl") 'Invoke Call to ConvertText dim result result = objSoapClient. ConvertAdvancedText ("Account","Password","合成文字位置"," "flv",5,1,10) 'Iterate through the returned string array resultArray = Split(result,"&") dim resultArray dim resultNameArray(3) resultNameArray(0) = "resultCode : " resultNameArray(1) = "resultString : " resultNameArray(2) = "resultConvertID : " for i=0 to UBound(resultArray) response.write resultNameArray(i)+resultArray(i)+" " next Set objSoapClient = Nothing %>



Android 範例	<pre>String methodName = "ConvertAdvancedText"; String soapAction = nameSpace+methodName; SoapObject rpc = new SoapObject(nameSpace, methodName); rpc.addProperty("accountID", "Account"); rpc.addProperty("password", "Password"); rpc.addProperty("TTStext", "TTSText"); rpc.addProperty("TTSSpeaker", "TTSSpeaker"); rpc.addProperty("volume", "Volume"); rpc.addProperty("speed", "Speed"); rpc.addProperty("outType", "OutputType"); rpc.addProperty("PitchLevel", "PitchLevel"); rpc.addProperty("PitchSign", "PitchSign"); rpc.addProperty("PitchScale", "PitchScale"); SoapSerializationEnvelope envelope = new SoapSerializationEnvelope(SoapEnvelope.VER11); envelope.bodyOut = rpc; envelope.dotNet = true; envelope.setOutputSoapObject(rpc); HttpTransportSE transport = new HttpTransportSE(endPoint); transport.debug = true; Object obj = null; try { transport.call(soapAction, envelope); obj = envelope.getResponse(); result = obj.toString(); Log.i("shiaukai", "result:"+obj.toString()); String[] retAry = result.split("&"); String status = retAry[0]; Log.i("shiaukai", status); if(status.equals("0")){ convertID = Integer.valueOf(retAry[2]); } Message msg =new Message(); msg.what=SoapExActivity.RESULT; SoapExActivity.this.handler.sendMessage(msg); } catch (Exception e) { e.printStackTrace(); }</pre>
IOS 範例	<pre>NSString *accountID =@"Account"; NSString *password =@"Password"; NSString *TTStext =@"合成文字位置"; NSString *TTSSpeaker=@"Angela"; int volume =100; //Range 0~100 default=100 int speed =0; //Range -10~10 default=0 NSString *outType =@"wav";// or flv NSString *PitchLevel=@"0";//Range -10~10 default=0</pre>



```
NSString *PitchSign =@"1";//0,1,2 default=0
NSString *PitchScale=@"5";//Range 0~20 default=5
//建立 SOAP Message
NSString *soapMessage = [NSString stringWithFormat:
@"><?xml version=\"1.0\" encoding=\"utf-8\"?>
<soap12:Envelope "
"xmlns:xsi=\"http://www.w3.org/2001/XMLSchema-instance\" "
"xmlns:xsd=\"http://www.w3.org/2001/XMLSchema\" "
"xmlns:soap12=\"http://www.w3.org/2003/05/soap-envelope\">
<soap12:Body>
<ConvertAdvancedText xmlns=\"http://tts.itri.org.tw/\">
<accountID>%@</accountID>
<password>%@</password>
<TTStext>%@</TTStext>
<TTSSpeaker>%@</TTSSpeaker>
<volume>%i</volume>
<speed>%i</speed>
<outType>%@</outType>
<PitchLevel>%@</PitchLevel>
<PitchSign>%@</PitchSign>
<PitchScale>%@</PitchScale>
</ConvertAdvancedText>
</soap12:Body>
</soap12:Envelope>",accountID,password,TTStext,TTSSpeaker,volume,speed,outType,PitchLevel,Pitch
NSLog(@"%@",soapMessage);
//建立 URL Request 時需要的 NSERL 參數
NSURL *url = [NSURL URLWithString:@"http://tts.itri.org.tw/TTSService/SOAP_1_3.php?wsdl"];
//建立 URL Request
NSMutableURLRequest *urlRequest=[NSMutableURLRequest requestWithURL:url];
//SOAP Message 長度， web service 需要
NSString *msgLength = [NSString stringWithFormat:@"%d", [soapMessage length]];
[urlRequest addValue: @"application/soap+xml; charset=utf-8" forHTTPHeaderField:@"Content-Type"];
[urlRequest addValue: msgLength forHTTPHeaderField:@"Content-Length"];//SOAP message 長度
[urlRequest setHTTPMethod:@"POST"];
[urlRequest setHTTPBody: [soapMessage dataUsingEncoding:NSUTF8StringEncoding]];
conn=[[NSURLConnection alloc] initWithRequest:urlRequest delegate:self];
if (conn) {
    webData=[NSMutableData data];
}
```

函式名稱	GetConvertStatus
函式說明	提供詢問目前 TTS 合成之進度，其狀態表可參考 4.3.1 TTS API 回傳值說明之內容



輸入參數	Account(String)—輸入您申請的帳號 Password(String)—輸入您申請的密碼 ConvertID(int)—輸入欲查詢之 ConvertID
函式範例	GetConvertStatus("ITRI"," ITRI Password ",5523)
回傳參數	Result(String) – resultCode&resultString&statusCode&status&result resultCode —回傳執行狀況代碼(參考表一) resultString —回傳執行狀況說明(參考表一) statusCode —回傳 TTS 合成進度代碼(參考表二) status —回傳 TTS 合成進度說明(參考表二) result —當 resultCode 為 0(Success)時，且 statusCode 為 2(completed)，回傳檔案之下載位置。
回傳範例	0&Success&2&completed& http:// tts.itri.org.tw/TTSService/download/ITRI/1111.wav
PHP 範例	<?php //Setup Web Service \$client = new SoapClient("http:// tts.itri.org.tw/TTSService/SOAP_1_3.php?wsdl"); // Invoke Call to ConvertText \$result=\$client-> GetConvertStatus ("Account"," Password", ConvertID); // Iterate through the returned string array \$resultArray=explode("&",\$result); list(\$resultCode, \$resultString, \$statusCode, \$status, \$resultUrl) = \$resultArray; echo "resultCode : ".\$resultCode." "; echo "resultString : ".\$resultString." "; echo "statusCode : ".\$statusCode." "; echo "status : ".\$status." "; echo "resultUrl : ".\$resultUrl." "; ?>
ASP 範例	<% 'Setup Web Service Set objSoapClient = Server.CreateObject("MSSoap.SoapClient30") objSoapClient.ClientProperty("ServerHTTPRequest") = true objSoapClient.MSSoapInit ("http:// tts.itri.org.tw/TTSService/SOAP_1_3.php?wsdl") 'Invoke Call to ConvertText



	<pre>dim result result=objSoapClient.GetConvertStatus("Account","Password",Conve rtID) 'Iterate through the returned string array resultArray = Split(result,"&") dim resultArray dim resultNameArray(5) resultNameArray(0) = "resultCode : " resultNameArray(1) = "resultString : " resultNameArray(2) = "resultConvertID : " resultNameArray(3) = "statusCode : " resultNameArray(4) = "status : " resultNameArray(5) = "resultUrl : " for i=0 to UBound(resultArray) response.write resultNameArray(i)+resultArray(i)+"
" next Set objSoapClient = Nothing</pre>
Android 範例	String methodName = "GetConvertStatus"; String soapAction = nameSpace+methodName; SoapObject rpc = new SoapObject(nameSpace, methodName); rpc.addProperty("accountID", "Account"); rpc.addProperty("password", "Password"); rpc.addProperty("convertID", id); SoapSerializationEnvelope envelope = new SoapSerializationEnvelope(SoapEnvelope.VER11); envelope.bodyOut = rpc; envelope.dotNet = true; envelope.setOutputSoapObject(rpc); HttpTransportSE transport = new HttpTransportSE(endPoint); transport.debug = true; Object obj = null; try { transport.call(soapAction, envelope); obj = envelope.getResponse(); result = obj.toString(); Log.i("shiaukai","result:"+obj.toString()); Message msg =new Message(); msg.what=SoapExActivity.RESULT_STATUS; SoapExActivity.this.handler.sendMessage(msg); } catch (Exception e) { e.printStackTrace(); }
IOS 範例	NSString *accountID =@"Account"; NSString *password =@"Password"; int convertID =[txt_ConvertID.text intValue];



```
//建立 SOAP Message
NSString *soapMessage = [NSString stringWithFormat:
    @"<?xml version=\"1.0\" encoding=\"utf-8\"?>
    "<soap12:Envelope >
        xmlns:xsi=\"http://www.w3.org/2001/XMLSchema-instance\" "
        "xmlns:xsd=\"http://www.w3.org/2001/XMLSchema\" "
        "xmlns:soap12=\"http://www.w3.org/2003/05/soap-envelope\">
    "<soap12:Body>
        "<GetConvertStatus xmlns=\"http://tts.itri.org.tw/\">
            "<accountID>%@</accountID>
            "<password>%@</password>
            "<convertID>%i</convertID>
        "</GetConvertStatus>
    "</soap12:Body>
    "</soap12:Envelope>",accountID,password,convertID];
NSLog(@"%@",soapMessage);
//建立 URL Request 時需要的 NSERL 參數
NSURL *url = [NSURL URLWithString:@"http://tts.itri.org.tw/TSService/SOAP_1_3.php?wsdl"];
//建立 URL Request
NSMutableURLRequest *urlRequest=[NSMutableURLRequest requestWithURL:url];
//SOAP Message 長度， web service 需要
NSString *msgLength =[NSString stringWithFormat:@"%d", [soapMessage length]];
[urlRequest addValue: @"application/soap+xml; charset=utf-8" forHTTPHeaderField:@"Content-Type"]; //內容形態
[urlRequest addValue: msgLength forHTTPHeaderField:@"Content-Length"]; //SOAP message 長度
[urlRequest setHTTPMethod:@"POST"];
[urlRequest setHTTPBody: [soapMessage dataUsingEncoding:NSUTF8StringEncoding]];
conn=[[NSURLConnection alloc] initWithRequest:urlRequest delegate:self];
if(conn) {
    webData=[[NSMutableData data];
}
```

5. TTS Web 應用程式快速開發指南

本章節將說明如何將現有網頁快速有聲化方案。



5.1. 說明

提供兩種有聲化的方案：

1. 使用 Web Service Tools (需下載檔案，放在網頁目錄並且使用 JAVASCRIPT 調用)
2. 直接引入所提供之 JAVASCRIPT API(Text2SpeechJsApiV2)

兩種方案都需使用 JAVASCRIPT 調用。在使用時請確認網頁瀏覽器中 JAVASCRIPT 為開啟的狀態。

差別在於，第一種方案在音檔合成完後自動將檔案下載至指定目錄，且伺服器需支援 PHP 或 ASP；第二種方案不需下載任何東西，只要在網頁標頭引入 JAVASCRIPT 即可使用。



5.2. 使用 Web Service Tools

5.2.1. Web Service Tools 架構圖



Web Service Tools 架構示意圖

5.2.2. 使用環境

需使用在支援 PHP 或 ASP 的 Web 伺服器。若伺服器使用 ASP 需先安裝 SOAP Toolkit 3.0 (<http://www.microsoft.com/downloads/en/details.aspx?FamilyID=c943c0dd-ceec-4088-9753-86f052ec8450>)

設置安裝完成後使用 JAVASCRIPT 調用。

此插件使用必需是網站的設置者使用，若無控制網站權限，則無法使用。

使用者使用或開發者測試時可使用 IE、Google Chrome、Firefox、Opera 任一瀏覽器。



5.2.3. 安裝說明

將 TTSTool.zip(<http://tts.itri.org.tw/downloads/TTSTool.zip>)解壓縮至網頁目錄中(預設為該網頁目錄之根目錄)，解壓縮後，有四個資料夾，Audiofiles、config、func、images，至 config 資料夾中，修改 config.php/php 或 config.inc/asp 檔案，編輯檔案中的帳號、密碼(於 TTS WebService 主頁所申請之帳號、密碼)及音檔存放位置。

PHP 範例(config.php)：

```
<?php
/* 帳號*/
$cname = "你的帳號";
/* 密碼*/
$passwd = "你的密碼";
/* 音檔存放位置*/
$AudiofilesAddress = "../Audiofiles/";
?>
```

ASP 範例(config.inc)：

```
<%>
`帳號
cname = "你的帳號"
`密碼
passwd ="你的密碼"
`音檔存放位置
AudiofilesAddress = "../Audiofiles/"
%>
```

5.2.4. 設置說明

1. 在網頁的 HEAD 標籤中加入

```
<script language="javascript"src=".//TTSTool/TTSScript.js"></script>
```

以下為範例(紅色為新增部分)



```
<!DOCTYPE html>
<html>
    <head>
        <script language="javascript" src=".//TTSTool/TTSscript.js"></script>
    </head>
    <body>
        <div>測試句子</div>
    </body>
</html>
```

2. 在希望呈現 TTS 的圖示的位置，新增一個 DIV 標籤，並且給它一個 ID，

<div id="media"></div>

且將希望轉換的文字所在之標籤也給予一個 id

<div id="string">文字內容</div>

以下為範例(紅色為新增部分)

```
<!DOCTYPE html>
<html>
    <head>
        <script language="javascript" src=".//TTSTool/TTSscript.js"></script>
    </head>
    <body>
        <div id="string">測試句子</div>
        <div id="media"></div>
    </body>
</html>
```

3. 在該 DIV 標籤 之下插入一段 javascript 碼，



```
<script language="javascript" >
    TTS.ConverterIndex= "./TTSPluginEditor/";
    TTS.PlayerSet.width=100;
    TTS.PlayerSet.height=30;
    TTS.PlayerSet.hidden=true;
    TTS.serverType="php";
    TTS.Audiofilename=new Date().getTime();
    TTS.ConvertInit("string" , "media" , "Bruce" , "100" , "0" , "0" , "0" , "5");
</script>
```

4. 加入前述步驟之程式碼的完整範例如下：



```
<!DOCTYPE html>
<html>
<head>
<script language="javascript" src="./TTSPluginEditor/TTSscript.js"></script>
</head>
<body>
<div id="string">測試句子</div>
<div id="media"></div>
<script language="javascript">
TTS.ConverterIndex="./TTSPluginEditor/"; /*設定根目錄，預設為./TTSPluginEditor*/
          /*所設定的目錄位置為相對於本檔案(html、asp 或 php)的位置*/
TTS.PlayerSet.width=100; /*設定播放器寬度*/
TTS.PlayerSet.height=30; /*設定播放器高度*/
TTS.PlayerSet.hidden=true; /*設定是否隱藏播放器 true:隱藏，false:顯示*/
TTS.serverType="php"; /*設定伺服器應用程式語言類型 php(預設)、asp*/
TTS.Audiofilename=new Date().getTime(); /*設定下載音檔檔名，預設為時間戳記*/
TTS.ConvertInit("string" , "media" , "Bruce" , "100" , "0" , "0" , "0" , "5");
/* 參數依序為：
StringOrObj.：希望轉換的標籤 id，會過濾此標籤中所有的標籤，並且轉換文字為語音，  

若有不希望轉換的文字，則在該文字前後加上任意標籤並且加上 notts 屬性，值為 true (nottt="true")。
mediaID：設定最後的播放器及點擊圖示位置。為任意標籤。且一定要在此函數之上。
Speaker：設定語者。可選擇語者有 Bruce、Theresa 與 Angela。可設置空值(即使用預設值)，預設值為 Bruce。
Volume：音量大小，可調整範圍為 0~100。可設置空值(即使用預設值)，預設值為 100。
Speed：語音速度，可調整範圍為 -10~10。可設置空值(即使用預設值)預設值為 0。
PitchLevel：韻律調整：值越大則音高越高；反之則音高越低，可調整範圍 -10~10，可設置空值(即使用預設值)，預設值為 0。
PitchSign：韻律調整：0=正常、1=像機器人、2=像外國人說中文，可設置空值(即使用預設值)，預設值為 0。
PitchScale：韻律調整：值越大則抑揚頓挫越明顯；反之則越趨平版，可調整範圍 0~20，可設置空值(即使用預設值)，預設值為 0。*/
</script>
</body>
</html>
```



5.3. 使用 JAVASCRIPT API

5.3.1. JAVASCRIPT API 架構圖



5.3.2. 使用環境

有網路情況下，直接使用支援 JavaScript 的瀏覽器即可使用。

5.3.3. 取得 license key

註冊帳號之後，於修改個人資料取得 license key

並且設定(<http://tts.itri.org.tw/member/registration.php>)註冊使用網址。

在網頁的 head 中加入 script 其位置為

<http://tts.itri.org.tw/TTScript/Text2SpeechJsApiV2.php?key=yourLicense>

接著將網址中的 yourLicense 改為剛剛取得的 license key



以下為範例(紅色為新增部分)

```
<!DOCTYPE html>
<html>
    <head>
        <script language="javascript"
src="http://tts.itri.org.tw/TTS/TTScript/Text2SpeechJsApiV2.php?key=YourLicense"></script>
    </head>
    <body>
        <div>測試句子</div>
    </body>
</html>
```

並且將網址中 YourLicense 修改為上述取得的 License code，即可使用所提供之 TTS JavaScript API。

5.3.4. JavaScript API 說明

變數名稱	TTS.muteTag
變數說明	設定不合成的 tag，格式為"attribute:name attribute2:name..." (支援 attribute:id、name、class)。
變數範例	TTS.muteTag = "id:tagname class:classname";
使用時機	在原有網頁架構下，透過指定合成的標籤，在該標籤中，有可能會有多餘的、不希望合成的標籤，此時，可以設置 TTS.muteTag。

變數名稱	TTS.PlayerSet.hidden
變數說明	設定是否隱藏播放器控制欄 true:隱藏，false:顯示
變數範例	TTS.PlayerSet.hidden = true;
使用時機	若希望播放器控制欄顯示，則可將 TTS.PlayerSet.hidden 設為 false，反之設為 true(預設)。

變數名稱	TTS.PlayerSet.width , TTS.PlayerSet.height
------	--



變數說明	設定播放器寬度、高度。
變數範例	TTS.PlayerSet.width=100
使用時機	若播放器控制欄設定為顯示，即可設定控制欄的寬度、高度

函數名稱	TTS.ConvertInit, TTS.ConvertCustom
函數說明	合成函式。
參數說明	<p>StringOrObj(String)—希望轉換的標籤，支援(id、name、class)，格式為"id:idname name:name class:classname"，若有不希望轉換的文字，則在該文字前後加上任意標籤並且加上 notts 屬性，值為 true (nottt="true")。</p> <p>mediaID(String)—聲音控制 bar 產生位置</p> <p>Speaker(String)—可選擇語者有 Bruce、Theresa 與 Angela</p> <p>Volume(Integer)—其音量大小可調整範圍為 0~100，預設值為 100</p> <p>Speed(Integer)—其語音速度可調整範圍為-10~10，預設值為 0</p> <p>PitchLevel(Integer)—韻律調整：值越大則音高越高；反之則音高越低，可調整範圍-10~10，預設值為 0</p> <p>PitchSign(Integer) —韻律調整：0=正常、1=像機器人、2=像外國人說中文，預設值為 0</p> <p>PitchScale(Integer) -韻律調整：值越大則抑揚頓挫越明顯；反之則越趨平版，可調整範圍 0~20，預設值為 5</p>
函數範例	<pre>TTS.ConvertInit("id:x","media","","","","","","","","")</pre> <pre>TTS.ConvertCustom("id:x","media","","","","","","","","")</pre>
使用時機	TTS.ConvertInit、TTS.ConvertCustom 參數相同，差別在於，TTS.ConvertInit 已預設好 icon，TTS.ConvertCustom 需自定 event 去觸發



5.3.5. JavaScript API 範例

```
<!DOCTYPE html>
<html>
<head>
<script language="javascript"
src="http://tts.itri.org.tw/TTSCript/Text2SpeechJsApiV2.php?key=YourLicense"></script>
</head>
<body>
<div id="x" >文字內容
    <div class="notts" >不想合成的內容</div><!--說明 4-->
    <div name="notts2" >不想合成的內容</div>
    <div id="notts3" >不想合成的內容</div>
    <div notts="true" >不想合成的內容</div><!--說明 3-->
</div>
<div id="media"></div>
<script language="javascript">
    var x = new TTS(); /*說明 1*/
    x.PlayerSet.hidden=true; /*說明 2*/
    x.muteTag="class:notts|name:notts2|id:notts3";
    x.ConvertInit("id:x","media","","","","","","","");
</script>
</body>
</html>
```

以上述範例為例，首先在希望合成的文字的主標籤之下，首先宣告一個新的 TTS() ([說明 1](#))

`var x = new TTS();`

若不希望最後的播放器控制欄顯示，則設置(true:不顯示;false:顯示) ([說明 2](#))

`x.PlayerSet.hidden=true;`

設定不希望合成的標籤，若原始架構可改變，可在不希望合成的標籤，加入參數 `notts="true"`([說明 3](#))，或是原始架構有設置 `class`、`name` 或 `id` 則設定 `muteTag`([說明 4](#))

`x.muteTag="class:notts|name:notts2|id:notts3";`

最後再設置合成函式

`x.ConvertInit("id:x","media","","","","","","","","");`



第一個參數指定 id 為 x 的標籤內所有文字為轉換的文字。第二個參數指定 icon 與播放器所放置的位置。或自定 event 使用 ConvertCustom

```
<button onclick="new TTS. ConvertCustom("id:x","media","","","","","","","","")">開始合成</button>
```