IoTtalk 物聯網系統使用手冊

國立交通大學資訊工程學系林一平講座教授

liny@csie.nctu.edu.tw

1. 簡介

物聯網設備可以用設備功能或特徵來分類，為了描述方便，我們定義物聯網設備功能(特徵)(device feature，DF)為特殊的輸入或輸出功能(capability)。例如，擁有溫度感測器之穿戴式手環即具有溫度(Temperature)之輸入功能(input device feature, IDF)，擁有光學顯示之穿戴式眼鏡即具有顯示(Display)之輸出功能(output device feature, ODF)。一般而言，物聯網設備可以直接經由無線網路或間接經由智慧手機連接至網際網路，而在網路端開發/執行網路應用程式就可以傳送或接收來自物聯網設備之訊息。當輸入功能產生新的數值時，物聯網設備可以通知網路應用程式進行處理，並將結果傳送至物聯網設備之輸出功能，因此經由其功能，不同物聯網設備即可互相交談，也可以說，網路應用程式將輸入功能(IDF)對應至輸出功能(ODF)。

IoTtalk 系統是一個基於設備功能 (device feature) 特徵概念之物聯網設備平台，當物聯網設備連接至系統時，針對各種感測器，IoTtalk 會自動產生或使用應用軟體來處理，因此每一個輸入設備可以相當方便地連接至輸出設備。圖 1.1 說明 4 個物聯網設備 D1、D2、D3 以及 D4，圖之左側代表其輸入功能(IDFs)；而圖之右側則代表其輸出功能(ODFs)。範例中，D1 為一具有多種感測器之設備，稱為積木式感測器 MorSensor，MorSensor 之感測器可以很容易抽換成不同的感測器。如圖 1.1 左側所示，D1 具有包括加速器(Acceleration)以及溫度(Temperature)兩個輸入功能，而智慧手機 D2 則具有包括加速器(Acceleration)以及麥克風(Microphone)兩個輸入功能。另一方面，如圖 1.1 右側所示，D2 同時具有顯示(Display)之輸出功能；而燈泡(bulb) D3 則具有一個稱為流明(Luminance)之輸出功能。尾巴(Tail)D4 則具有一個稱為顫動(Vibration)之輸出功能，這個特殊設備叫做“Transparent Organ”，發

表於日內瓦國際發明展並獲得銀牌獎[Huang2014]，設備尾巴會根據顫動值強度來進行搖晃。

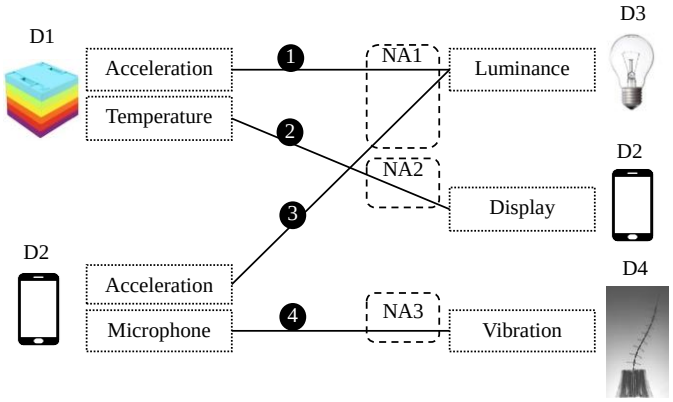

圖 1.1. 物聯網設備 D1, D2, D3, and D4 之連線

圖 1.1 中線段(1)到(4)分別代表這些物聯網設備間之交流關係，當線段連接一個輸入及輸出功能即表示此輸入與輸出功能存在某種對應關係，而這種對應關係是以 Python 程式實作而成之網路應用程式。虛線方塊 NA1 代表這個網路應用程式 NA1 實作 D1、 D2 與 D3 間之交流關係(1)與(3)。同樣地，NA2 實作 D1 與 D2 間之交流關係(2)，而 NA3 實作 D2 與 D4 間之交流關係(4)。舉線段(2)為例，線段(2)連接溫度感測器(Temperature)之輸入功能至顯示(Display)之輸出功能，也就是說，NA2 先處理來自 D1 之溫度值，再將處理結果顯示到智慧手機 D2。假如一支網路程式只處理一個獨立的設備功能，那麼我們就可以針對每一個設備功能分別撰寫程式。經由重複使用這些程式模組，即可輕易地建構網路應用程式。舉例而言，在圖 1.1 中，線段(4)之程式模組如圖 1.2 所示，網路應用程式 NA3 可經由麥克風模組處理 D2 麥克風(Microphone)之音量，先計算麥克風音量之大小，再將結果傳送至震動（Vibration）模組，震動模組將接收到的值轉化成震動之大小（vibration intensity）。之後 NA3 再輸出此震動值大小至震動輸出功能（ODF），進而驅動 D4 設備之尾巴震動。

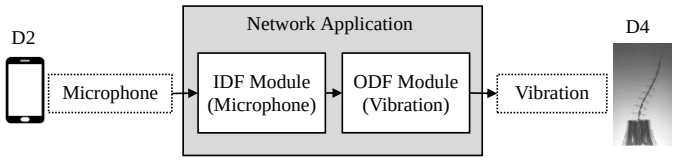

圖 1.2. 對應麥克風與震動之 NA3 軟體模組

既然輸入功能與輸出功能各自相互獨立，這些軟體模組即可重複使用來建立網路應用，也就能大幅加速物聯網應用開發之時間。圖 1.1 顯示不同的物聯網設備也可能具有相同之輸入或輸出功能（IDFs/ODFs），例如，D1 與 D2 都

具有加速器（Acceleration）這個輸入功能，所以，NA1 就可以重複使用相同模組來實作所需之工作，例如，線段(1)與(3)即代表類似之功能。

我們的中心概念即為設備特性（DF），這些設備功能可以根據傳統或非傳統之人類感官進行分類如下：

- 視覺（Sight）:輸入設備可以是一個參數（影片格式）之照相機或是紅綠藍（ＲＧＢ）三個參數之色彩感測器。而輸出設備可以是一個參數之文字顯示器或是圖形、視訊等等。其他種視覺輸出範例則是具有一個參數流明（Luminance）值之燈泡。
- 聽覺(Hearing):輸入設備可以是一個參數(聲音大小,sound pressure level，表示為 μPa 或 Pa）之麥克風。而輸出設備則可以是相同參數之麥克風或是音樂撥放器等等。
- 味覺（Taste)、嗅覺（smell)、與化學檢測（chemoreceptor）: 這些輸入設備可以是一個設備參數（CO-level）之碳氧化合物。
- 觸覺(Touch): 觸覺輸入設備可以是一個設備參數(每平方英寸幾磅,pound per square inch，PSI)之壓力感測器，例如大氣壓力(atmosphere，ATM)等等。此外觸覺輸入設備也可以是一個布林參數之開關按鍵;而觸覺輸出設備則可以是一個整數值參數之多階段開關（multi-stage switches）。
- 溫度(Temperature): 溫度感測器用一個參數值來代表攝氏或華式溫度數。溫度感測器可以正常運作之區間可以表示為[a, b]。
- 動作(Motion),包括動力(kinesthetic sense)與加速度(acceleration): 位置感測器可以是具有經度緯度時間（longitude, latitude and time）三項參數之室外 GPS 接收器或是室內 iBeacon 接收器。而動作感測器則具有三個維度，主要包括加速度計、陀螺儀等等，主要輸入輸出功能包括震動（vibration)、速度（speed)、流動(flow)等等。
- 回聲定位（Echolocation）、電覺（electroreception）、磁感應（magnetoception）: 輸入設備可以是一個聲音頻率（frequency）參數之超音波感測器、參數伏特之電壓感測器、以及強度與磁場方向兩個參數之磁力計。

根據設備特性,我們發展一套稱為 IoTtalk 之物聯網設備特性管理系統，IoTtalk 系統架構(如圖 1.3)包含網路領域(network domain，圖 1.3(a)) 與

設備領域（device domain，圖 1.3(b))兩個領域。其中網路領域由四個系統組成，建立、設定與管理系統（Creation, Configuration and Management system，縮寫 CCM; 圖 1.3 (c)) 系統性將物聯網設備特性分類，管理並自動設定輸入與輸出功能之連結，儲存所有相關資訊於資料庫系統(DB; 圖 1.3 (d))。執行與通訊系統（Execution and Communication system，縮寫 EC; 圖 1.3 (e)) 由兩個子系統組成， 通訊子模組系統（Communication SubModule system，縮寫 CSM;圖 1.3 (f))定義 HTTP based RESTful API (Application Programming Interface)，提供給設備應用（Device Application，縮寫 DA; 圖 1.3(g)) 用來傳送或取得輸入/輸出設備資訊。當一個物聯網設備註冊/取消註冊(registers/deregister) 到 EC 時，DA 會經由 HTTP API 要求 CSM 去改變資料庫中之設備狀態，當物聯網設備完成註冊至 DA 後，即可以經由 E C 彼此相互通訊。執行子模組系統（Execution SubModule system，縮寫 ESM;圖 1.3(h))代表執行網路應用之以連接相關輸入與輸出功能。圖形介面(GUI)(圖 1.3(i))提供一個友善的網頁使用者介面，可用來快速建立連線以及物聯網間有意義之交流（meaningful interactions)，經由此一圖形介面，使用者可以指揮 IoTtalk 去執行所需求工作 ，用以建立或設定設備特性，功能，以及連線設定。

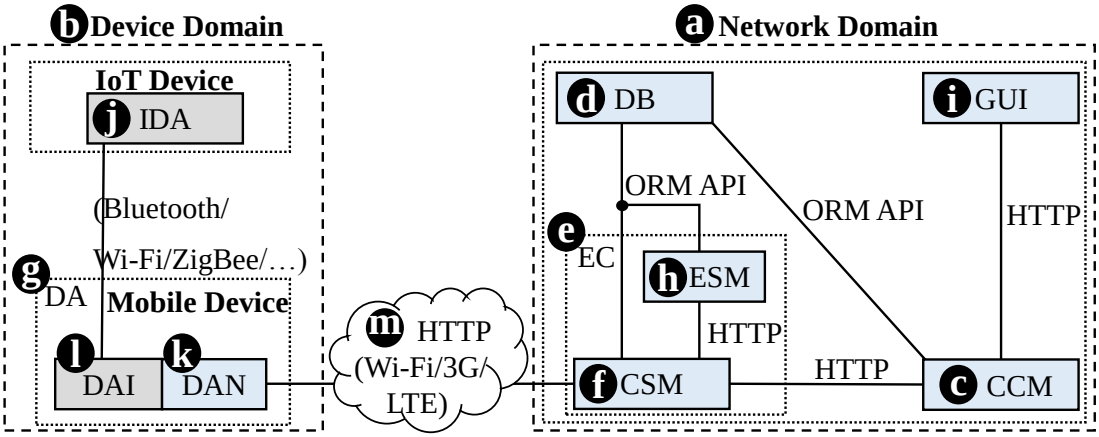

圖 1.3. IoTtalk 系統架構

在設備領域（device domain）中，DA（圖 1.3(g))被安裝於行動設備（例如，智慧手機），物聯網設備應用（IoT Device Application，縮寫 IDA;圖 1.3(j))會間接經由 DA 連接至 IoTtalk 系統。在行動設備中，DA 由兩個軟體部分組成。設備應用至網路（Device Application to the Network，縮寫 DAN; 圖 1.3 (k))經由 Wi-Fi, 3G, or LTE.與 IoTtalk 進行通訊，用來執行 IDA 註冊(registration)與資料交換。設備應用至物聯網設備(Device Application

to IoT Device，縮寫 DAI;圖 1.3 (l))依據 IDA 所指定之訊息格式與物聯網設備進行通訊，通常是經由藍芽（Bluetooth）、 Wi-Fi、ZigBee 等無線網路所傳送之字串。物聯網設備必須先註冊到 CSM，才能被 IoTtalk 所操作。IoTtalk 註冊之概念遵循行動通訊之註冊流程，詳細可參考[Lin2001]. 行動通訊中註冊包括認證(authentication)、加密設定(initialization of encryption)、以及手機位置更新。註冊後，行動設備即可連接到網路並開始進行資料連線。目前 IoTtalk 之版本並未採用應用層之認證與加密，由於物聯網設備相對簡單之特性，IoTtalk 之註冊流程僅包含連接（attachment）與資料連線設定。

我們已經開發包括網路版與單機版之兩個 IoTtalk 版本。在網路版中，IoTtalk 實作一個伺服器，可經由網際網路存取伺服器，網路版特別適合連接分散於各個不同網路之物聯網設備，同時網路版亦可支援多人同時分享各自之設備與功能。在單機版中，IoTtalk 可實作於個人電腦，目前所有物聯網設備全部需要透過智慧手機，使用 Wi-Fi 熱點(hotspot)服務方可連至 IoTtalk，也就是說，使用者只需打開手機之 Wi-Fi 熱點服務即可，並不需要 Wi-Fi AP or 3G/LTE 基地台來支援圖 1.3 連線(k)-(m)-(f)。圖 1.1 中之智慧手機 D2 即可當成熱點。所以 D2 除了是物聯網設備之角色外，亦是行動設備(圖 1.3 (g))以及在連線層（connectivity layer）中提供無線網路(圖 1.3 (m))，其中所有物聯網設備(D1、D2、D3 與 D4)均可經由 D2 連接到 IoTtalk 系統，本章節假設所有物聯網設備均可經由無線網路連接到 IoTtalk，若有不具無線網路功能之物聯網設備則不在本章考慮範圍。

2. 圖形介面 GUI 操作

圖 2.1 說明 IoTtalk 系統畫面專案視窗(Project Window)，主要可分成三部分： 選單(menu bar)(圖 2.1 (a))、圖形布局視窗(Graphical Layout Window)(圖 2.1 (b))、以及管理視窗(Management Window)(圖 2.1 (c))。選單部分有三個選項，選項"Project”項目是下拉式選單(圖 2.1 (d))，用來選定一個特定專案(project)。模型選項“Model” (圖 2.1(e))是另一個下拉式選單用來選擇在圖形布局視窗中的設備模型。連線開關“Connection ON/OFF”項目(圖 2.1(f))是一個開關按鍵，用來啟動或關閉連線。

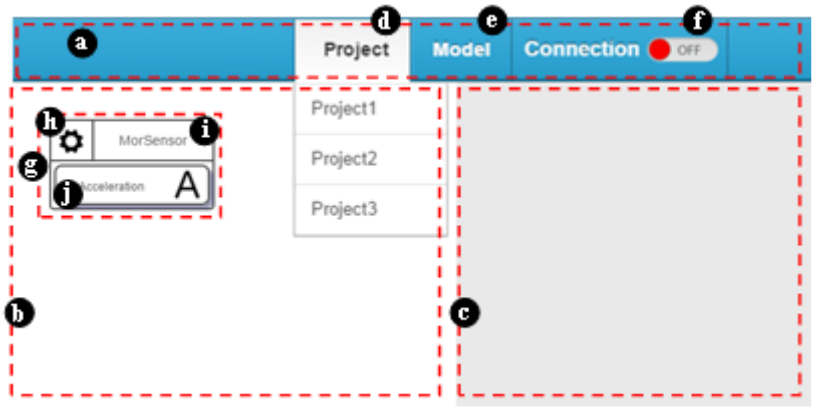

圖 2.1. 系統畫面專案視窗(Project Window): (a)選單(menu bar);(b)圖形布局視窗(Graphical Layout Window);(c)管理視窗(Management Window)

圖形布局視窗顯示物聯網設備與其連線情形。管理視窗允許使用者設定圖形布局視窗中之設備功能、連線、以及對應之函數(function)。物聯網設備之圖形化表示稱為設備物件(device object)，連線則稱為連線物件(connection object)。一個設備物件(圖 2.1(g))是一個設備模型(device model)的圖形化表示實例，可以利用圖形介面(GUI)產生並對應至真實的物聯網設備。設備物件包括一個設備設定物件(device setting object) (圖 2.1(h))、一個設備模型名稱物件(device model name object) (例如圖 2.1(i)中的一個 MorSensor 積木式感測器、以及好幾個設備功能(device feature，DF)物件(例如圖 2.1(j)中的加速器“Acceleration”)。藉由按壓(clicking)選擇這個設定物件，使用者可以設定相關參數、儲存或刪除這個設備物件。藉由按壓選擇這個名稱物件，使用者可以將設備物件連接至實際連接到 IoTtalk 系統的物聯網設備。另外藉由按壓選擇設備功能 DF 物件，使用者可以將這個 IDF 輸入功能設備連接至 ODF 輸出功能設備，當連線按鍵(Connection button)打開至“ON”時，所有的連線都將開啟，此時 IDF 輸入功能設備所產生之資料將經由這些連線被傳送至 ODF 輸出功能設備。

2.1. 設備物件建立(Device Object Creation)

當滑鼠移動到“Model”模型下拉式選單(圖 2.2 (e))時，IoTtalk 系統會傳送回該模型名稱並進行圖形顯示，而圖形介面則會利用這些設備模型名稱來產生下拉式選單(如圖 2.2 (e)與(f))。

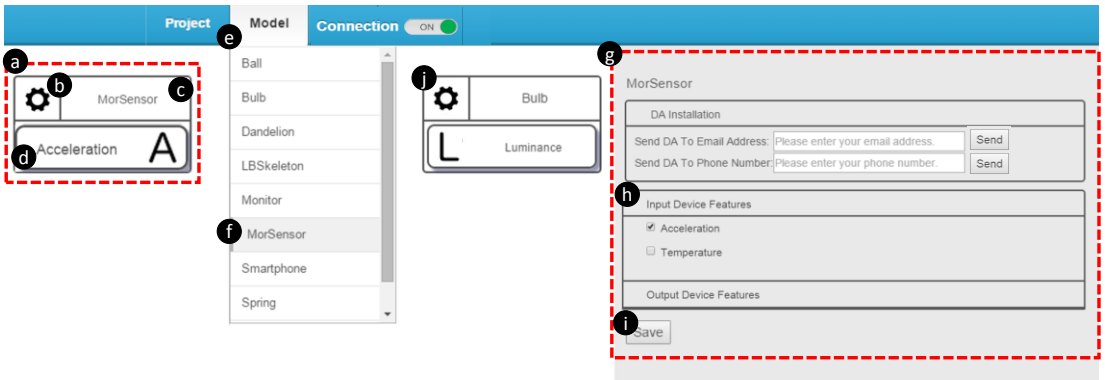

圖 2.2. 設備模型選擇(Device model selection)

當使用者選擇“Model”模型下拉式選單中的一個項目時(如圖 2.2(f)中的 MorSensor 感測器)，圖形介面 GUI 會將它們顯示在管理視窗中(圖 2.2(g))。參考圖 1.1 中 D1 (一個 MorSensor 感測器)有兩個 IDF 輸入設備功能(Acceleration 加速器與 Temperature 溫度計)，這個圖形介面會在管理視窗中將設備模型選擇(IDF/ODF Selection module)對應列出所有設備功能(圖 2.2(h)。值得一提的是，在這個範例中 MorSensor 感測器並無任何輸出設備功能。

對於包含在專案(project)中的物聯網設備而言，並非所有的功能都會被使用到，使用者可以從設備模型中只選擇連線所需之部分功能，當使用者選定所需功能時(如選定圖 2.2(h)中 Acceleration 加速器功能)，按下“Save”儲存按鈕(圖 2.2(i))後，就會產生一個設備物件(如圖 2.2(a)中圖形布局視窗所示)。就如前面所述，這個設備物件包括一個設備設定物件(圖 2.2(b)、一個設備模型名稱物件“MorSensor”(圖 2.2(c))、以及一個或多個設備功能物件(圖 2.2(d))

假如產生之設備物件只有一個輸入或輸出功能，例如燈泡(Bulb)只有一個流明亮度(Luminance)輸出功能，圖形介面會直接在圖形布局視窗中畫出該設備物件(圖 2.2 (j))。

2.2. 設備物件之圖形介面操作(GUI Operations on Device Objects)

首先本節將介紹如何下載 DA 程式至手機，設定設備物件，以及將物聯網設備連至這些設備物件。之後我們將展示如何在設備物件之間建立連線，並管理所對應之設備功能。

當使用者選定一個設備設定物件(圖 2.3 (a))，圖形介面會在管理視窗中顯示對應之設備功能選擇模組。具體而言，圖形介面會擷取該設備物件資訊，(例如 MorSensor 感測器可能之功能為加速器“Acceleration”與溫度“Temperature”，然而只有加速器 Acceleration 功能被選定在圖 2.3 中)，然後圖形介面會在管理視窗中顯示該設備模組(Device module)(圖 2.3(b))。

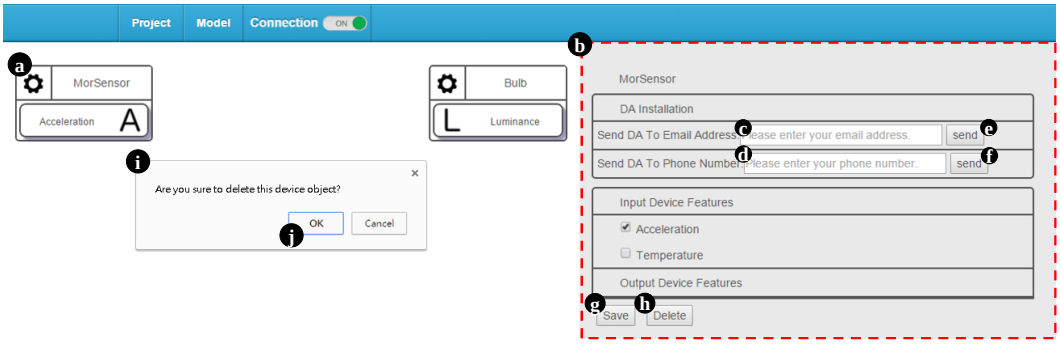

圖 2.3. 設備物件設定(Device object setting)

2.2.1. 設備應用安裝(Device Application Installation)

設備模組提供兩個輸入方塊(圖 2.3 (c)與(d))，分別用來設定經由電子郵件或簡訊傳送設備模組應用程式之下載超連結。在圖 1.3 中，一個物聯網設備可經由手機連接到 IoTtalk 系統，為了將物聯網設備正確地連接至

EasyConnet 系統，手機端必須先安裝相關之設備應用(DA)程式，假如使用者輸入選擇電子郵件位址(圖 2.3(c))， IoTtalk 將會傳送訊息至該電子郵件帳號，該訊息內含設備應用(DA)程式之下載超連結，當使用者經由手機端閱讀郵件並點選這個超連結時，系統將會自動下載並安裝這個設備應用(DA)程式至手機端。同樣地，上述設備應用(DA)程式下載程序亦可經由簡訊輸入方塊中所填之電話號碼(圖 2.3(d))之簡訊中執行。當使用者點選圖 2.3(e)or(f)中之傳送按鍵(“Send” button)，IoTtalk 系統會經由電子郵件或簡訊傳送下載 DA 程式之超連結訊息，而當使用者點選這個超連結時，這個設備應用(DA)程式將會自動下載並安裝至手機端。

2.2.2. 設備物件設定與連結(Device Object Setting and Binding)

經由設備模組的設定，使用者就可以操作該物聯網設備，例如變更所選之設備功能屬性或刪除該設備物件等操作。在使用者選擇設備功能並點選儲存按鍵“Save” button (圖 2.3(g))後，IoTtalk 系統將會更新設備物件資訊。然後圖形介面則會在圖形布局視窗中列出被修改之設備物件。當使用者按下刪除按鍵(“Delete” button) (圖 2.3 (h))後，將會跳出一個對話框來確認使用者是否要進行刪除動作(圖 2.3 (i))，當使用者點選“OK”按鍵(圖 2.3 (j))確認後，IoTtalk 才會從資料庫中刪除該設備物件，同時，圖形介面也將從圖形布局視窗中移除該設備物件。

當一個設備物件被產生時，它的名稱物件會顯示該設備模式名稱，例如圖 2.3 (a) 之 MorSensor，當使用者點選該名稱物件時，圖形介面會從資料庫中取得這個設備模式中所有已註冊之設備。我們假定相同的設備模式有好幾個設備已經註冊到 IoTtalk 系統，(例如 My MorSensor 與 My MorSensor2)，

對每一個註冊之物聯網設備，圖形介面都會在管理視窗中(圖 2.4 (a))產生一個以註冊設備列表。

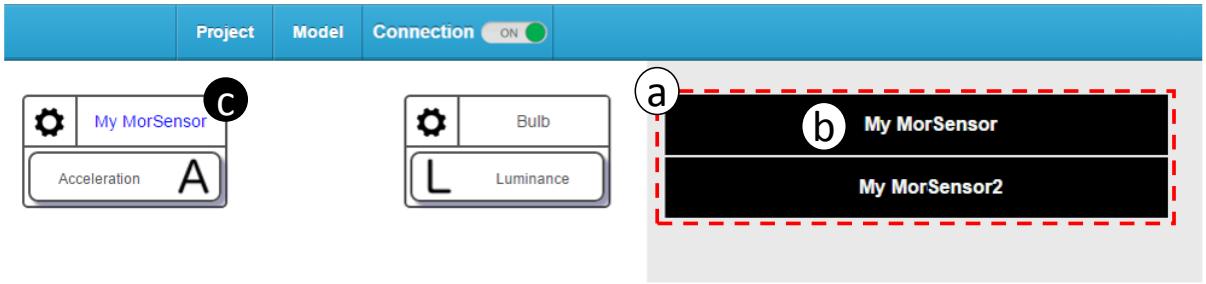

Figure 2.4. 設備對設備之物件連結

在使用者點選已完成註冊之設備(如圖 2.4 (b)之 My MorSensor)後，IoTtalk 系統會將真實設備連結至該設備物件，之後圖形介面將以設備名稱(如 My MorSensor)取代設備模式名稱如 MorSensor)，並以藍色顯示(圖 2.4(c))，用以代表該設備物件已被連結至真實設備。假如在已註冊設備列表中只有一個設備時，圖形介面將會直接取代，而不需要經過選擇的程序，也就是說，在此情形，圖 2.4(a)將不會被顯示。

如果要取消設備連結(unbind)，使用者可以在點選該設備名稱物件(如圖 2.4 (c)之 My MorSensor)，IoTtalk 系統將會從設備物件中取消連結該設備，舉圖 2.4 例而言，圖形介面將以“MorSensor”取代“My MorSensor”。

2.2.3. 設備功能連接操作 Operations on Device Feature Connection

在 IoTtalk 系統中，一個輸入設備功能(IDF)可以經由圖形布局視窗中一個被稱為連結物件(join object)的小圓圈連接至輸出設備功能(ODF)，在圖 2.5 中，Acceleration 加速器輸入功能物件(圖 2.5(a))被以線段連接至連結物件(圖 2.5 (b))，同時，Luminance 流明輸出功能物件已被另一線段

連結至該連結物件(圖 2.5 (c))。這兩個連線分別連結至輸入功能物件以及輸出功能物件，並經由連接圓點(join circle)定義了一條資料路徑(如圖 2.5 (a)-(b)-(c))。

假如一條連結只包括一個輸入功能設備稱為單一連結(single join)，如果一條連結包括多個輸入功能設備則稱為多重連結(multiple join)，對於多重輸入功能設備連結點，IoTtalk 系統可提供這些輸入功能設備間之連結函數，連結函數可用來設定這些輸入設備如何影響輸出設備。對於連接至同一個連接點的多個輸出設備功能(ODF)而言，輸入設備功能(IDF)對所有的輸出設備功能的影響都是相同的，所以即使有多個輸出設備功能(ODF)連接至同一個連接點，假如只有一個輸入設備功能(IDF)連至此一連接點，仍然可稱為單一連結。以下簡述單一連結，多重連結則會在下一節討論。

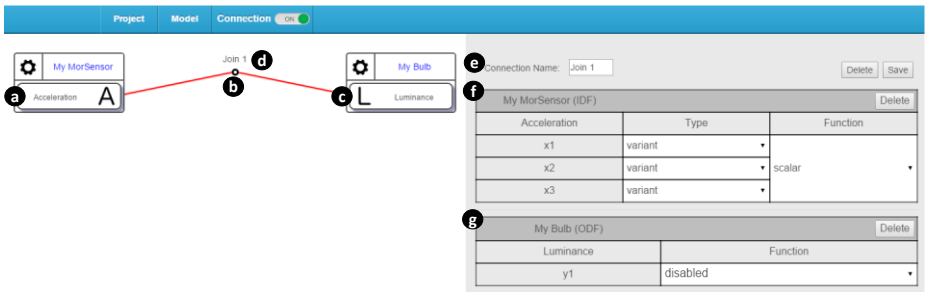

圖 2.5. 連線建立(Connection creation)

我們將以圖 2.5 中之範例介紹連線產生，藉由先後(反序亦可)點選加速器 Acceleration 輸入功能物件(圖 2.5(a))以及 Luminance 流明輸出功能物件(圖 2.5 (c))，IoTtalk 系統會建立一個連線，一個連線包括幾條線段以及線段連接點的連結物件(join object)，這連線會自動給定一個預設的名稱“Join 1”(圖 2.5(d))。值得說明的是，“Join K”僅代表計畫中第 K 條連線而已，稍後可以再更名。這連線是以 Python 所撰寫之網路應用程式實作完成。在 IoTtalk 將此連線儲存於資料庫之後，圖形介面將為此連線名稱(圖 2.5 (e))、輸入設備功能模組(圖 2.5 (f))、以及輸出設備功能(圖 2.5

(g))，在管理視窗中畫出輸入方塊，並以紅色顯示，藉以凸顯表示這個連線物件參數可以經由輸入設備功能模組與輸出設備功能模組在管理視窗中設定。

經由設定輸入設備功能模組(圖 2.6(a))與輸出設備功能模組(圖 2.6(b))，輸入以及輸出設備功能將被連結在一起，輸入設備功能模組為包括列出各項設備功能參數 (x_1, x_2, …, x_n)之表格，範例中類型“Type”之下拉式選單(圖 2.6 (c))允許使用者設定每一項參數 x_i 之類型， 假設 $x_i(j)$ 是加速器 Acceleration 輸入功能設備之第 j 項資料，當 IoTtalk 收到 $x_i(j)$, 參數 x_i 就可以取得其值(例如 $x_i(j)$值)或變形(例如 $x_i(j)-x_i(j-1)$)，IoTtalk 系統提供了一些可以對輸入參數進行處理的函數，例如最小值(min)、最大值(max)、純量(scalar)等等，使用者可以在函數“Function”下拉式選單(圖 2.6 (d))中選擇一個函數，同時，亦可增加新函數以“add new function” (圖 2.6(e))來建立新的函數，亦可選擇“disabled”來關閉所有處理函數。假如使用者關閉處理函數，所有設備功能參數(x_1, …, x_n)將會直接被傳送至下一階段(例如輸出設備功能模組)。如何建立一個設備功能函數將於第 2.2.5 節中描述。在圖 2.6 (c)中，參數 x_1, x_2, x_3 之變形類型被用來計算三軸加速器之變化，所以純量(scalar)函數(圖 2.6 (d))被選用來轉換輸入設備參數值(x_1, x_2, x_3)成為純量，同時這個純量也會自動地正規化於 0 到 1 之間之範圍[0,1]。

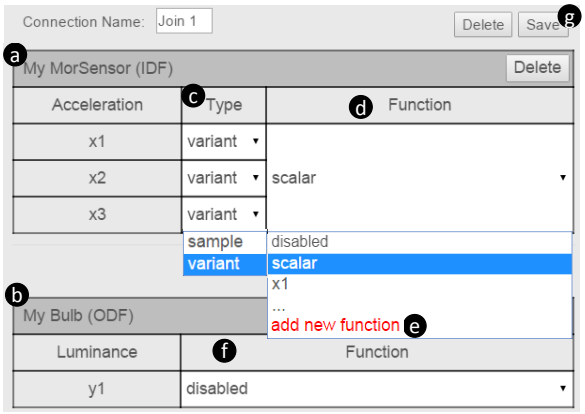

Figure 2.6. 管理視窗中之輸入以及輸出設備功能模組

在輸出設備功能模組中，函數“Function”下拉式選單(圖 2.6(f))允許使用者對輸出設備參數($y_1, \cdots, y_i, \cdots, y_k$)之每一個參數，分別選定一個處理函數。圖 2.6 (f)中關閉(disabled)了輸出功能函數處理，因此，從前階段(輸入設備功能模組)之輸入值將會直接指定給 y_i 。在 y_i 被真實之輸出設備收到之前，它可以先調整至一個範圍之內，藉由輸入與輸出設備功能模組之設定，積木感測器(MorSensor)之加速器 Acceleration 輸入設備功能可以對應至燈泡 Bulb 之流明 Luminance 輸出設備功能。

使用者可以點選儲存“Save”按鍵(圖 2.6 (g))進行連線設定存檔。在儲存過程中，IoTtalk 系統會收集輸入設備功能模組之設定(例如 sample type、IDF function)以及輸出設備功能模組之設定(例如 ODF function)，並儲存連線設定於資料庫中。舉例而言，在存檔後，圖 2.3 中的設備“My MorSensor”以及設備“My Bulb”即可以經由連線設定進行互動。

一條連線可以部份或完整地被刪除，假如連線之所有線段均被移除，就稱為被完全刪除，假如僅有部分線段被移除，即稱為部份刪除。所以，有兩種類別之刪除“Delete”按鍵，在輸入/輸出設備功能模組之外(圖 2.7(a))的刪除“Delete”按鍵即為完全刪除。相反地，在輸入/輸出設備功能模組之內(圖 2.7(b)(d))的刪除“Delete”按鍵即為部分刪除。當使用者按下完全刪除“Delete”按鍵，IoTtalk 系統將會從資料課中完整地刪除該連線資訊，同時也會移除此網路應用。最後，圖形介面將會從圖形布局視窗中移除這個連線物件。另外，如要刪除某些特定線段，只需選定對應至該線段之刪除鍵即可，舉例而言，當使用者按下在輸入設備功能模組(圖 2.7(b))之刪除鍵，IoTtalk 系統僅會刪除這個特定線段，而圖形介面亦僅從圖形布局視窗中刪除位於輸入設備功能物件(如 Acceleration)與連接物件(如 Join 1)之線段(圖 2.7(c))。

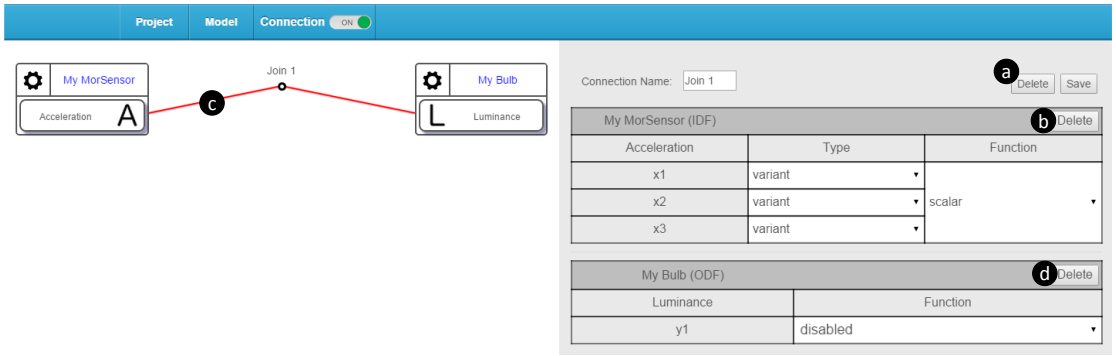

Figure 2.7. 連線刪除(Connection deletion)

2.2.4. 多重連結連線(Multi Join Connection)

如前所述,IoTtalk 系統可將多個物聯網輸入設備連接至一個輸出設備,我們提供的連結(join)函數可以支援多個輸入設備功能同時影響輸出設備功能。這些連結函數基本上與設備功能相同,唯一不同的是,這些設備參數可以從一個或多個物聯網輸入設備產生。

圖 2.8 擴充了圖 2.5 中單一連結成為多重連結。我們考慮一個叫做震動或聲音的遊戲(“Vibration or Voice”),其中,第一位玩家以積木感測器 MorSensor 產生大震動,而玩家 2 則以智慧型手機麥克風功能製造聲響。如果震動值較大則燈泡變亮;反之,如果麥克風值較大則燈泡變暗。以 IoTtalk 系統實作此遊戲,MorSensor 所量測之震動值即為第一位玩家之輸入,而麥克風所接收到的音量大小就是玩家 2 的輸入,同時燈泡的亮度流明 Luminance 即是這個遊戲的輸出。「震動值是否大於聲音」可以一個連結函數“大於(larger than)”($z_1>z_2$?),用來比較正規化後的加速器震動值(z_1)以及麥克風大小值(z_2),If $z_1>z_2$, then the result z_F=1. Otherwise ($z_1<z_2$), the result z_F=0. 在流明度 Luminance 的輸出設備功能模組選用增加函數 Increment(z_F),如果 z_F=1,就增加燈泡亮度;反之,如果 z_F=0,就降低燈泡亮度。

為了實作此一遊戲，我們擴充圖 2.5 中“Join 1”函數，增加一個新設備物件（圖 2.8(a))以及一個線段(圖 2.8(b))如下：當使用者點選麥克風設備功能物件 “Microphone”（圖 2.8(c)）以及連結物件“Join 1”（圖 2.8(d))來產生一條新線段。

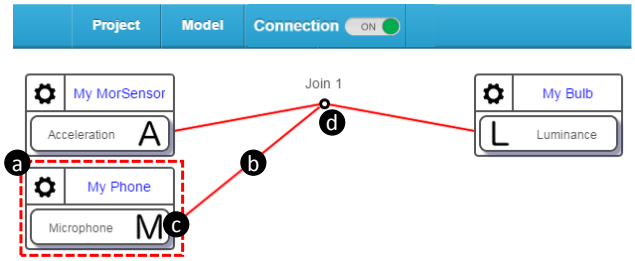

圖 2.8. 增加一個新設備物件以及新線段

假設 $x_{N,m}$ 代表第 m 條線段的輸出值，其中下註標 N 代表其值已被正規化成介於 0 到 1 之間的值，而連結同一物件的 M 個的輸入設備的輸出值則可表示為($x_{N,1}$, …, $x_{N,m}$, …, $x_{N,M}$)，而這些值則變成連結函數的輸入值 z_1, z_2, …, z_M。舉圖 2.8 為例，第一條線段($x_{N,1}$)將感測器“My MorSensor”的輸入設備加速器 Acceleration 與連結物件相連接，而第二條線段($x_{N,2}$)則將手機之感測器“My Phone”的麥克風 Microphone 也連接至同一連結物件。當使用者將多個輸入設備功能物件連接至同一個連結物件時，相對應之多重連結模組可參考圖 2.9 (a)之管理視窗所示，值得一提的是，假如只有連結一個輸入設備功能物件時，即使多個輸出設備功能物件連結至同一個物件，也不需要使用多重連結模組。在多重連接模組中，欄位“IDF (Line)”(如圖 2.9 (b))允許使用者重新安排各項參數之前後順序，也就是說，他允許交換輸入設備功能的輸出值 $x_{N,m}$. 在我們的範例中，$z_1=x_{N,1}$ 被設定為第一條線段(加速器 Acceleration; 圖 2.9 (c))，$z_2=x_{N,2}$被設定為第 2 條線段(麥克風Microphone; 圖 2.9 (d))，並未改變其順序。

當使用者點選“Join Function”選單，選擇了“larger than ($z_1>z_2$?)”(圖 2.9 (e))，如果 $z_1>z_2$ 為真，函數會回傳值 1；如果 $z_1>z_2$ 不成

立，函數則會回傳值 0。這個連結函數產生的結果 Z_F也就是輸出設備功能模組的輸入值(圖 2.9 (f))，如果 Z_F=1，增加函數 Increment(Z_F) (圖 2.9 (g)) 就會被選用來增加燈泡之流明亮度(luminance) ；相反地，如果 Z_F=0，則會減少亮度。最後，使用者必須點選儲存“Save”按鍵(圖 2.9 (h))來儲存這個設定，之後，感測器設備“My MorSensor”、“My Phone” 以及“My Bulb”就可以經由此連線設定進行互動，同時完成這個震動或聲音（“Vibration or Voice”)之遊戲。

圖 2.9. 多重連接模組(Multiple Join module)

2.2.5. 函數管理模組 Function Management Module

IoTtalk 系統採用設備功能函數(DF function)對不同設備功能間傳送資料值進行操作，一個功能函數可以將輸入設備所得之參數轉換成一個數值，例如，純量函數(scalar function)可以將三軸加速器之 X、Y、Z 三軸之值（X_1, X_2, X_3）變成一個數值如下：

$$\text{Scalar}(x_1, x_2, x_3) = \sqrt{{x_1}^2 + {x_2}^2 + {x_3}^2}$$

這些設備功能函數都儲存於資料庫之總體函數列表中(Global Function List)，相同函數可以被不同設備功能重複使用，例如，陀螺儀(Gyroscope)也可以使用相同函數將其 X、Y、Z 三軸之值轉換成一個純量數值。

當使用者從 IDF/ODF 模組之函數(“Function”)下拉式選單中，選擇增加新函數(“add new function”)時(圖 2.9 (i))，IoTtalk 系統會從資料庫中取得設備函數列表，圖形介面會先隱藏 IDF/ODF 模組，並在管理視窗中(圖 2.10)顯示函數管理模組(Function Management module)。如需結束函數管理，使用者可以點選關閉“Close”按鍵(圖 2.10(c))，之後圖形介面即會在管理視窗中重繪 IDF/ODF 模組。

在函數管理模組中，所有的設備功能函數均詳列於總體函數列表下(圖 2.10 (a))，並同時儲存於資料庫之函數表(Function Table)中，特定設備的功能函數(如加速器 Acceleration)則可見於特定功能函數列表(Specific DF Function List)中(例如，圖 2.10 (b)之 Acceleration Function List)，這個特定功能函數列表則儲存於資料庫之表格 FunctionSDF table 之中。

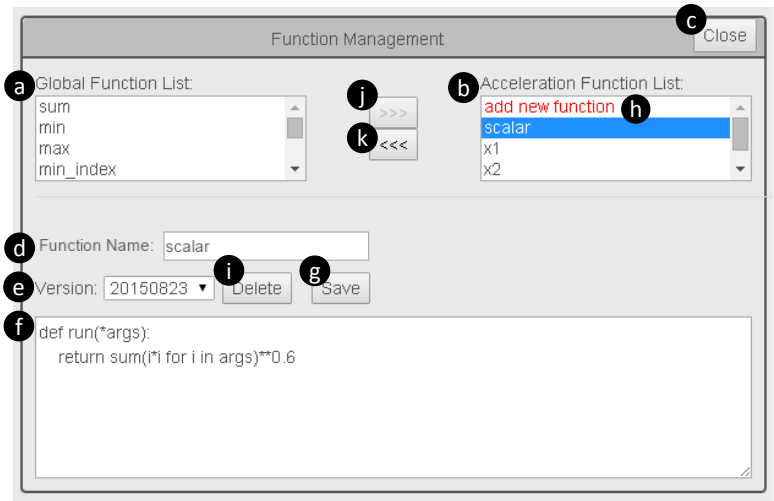

圖 2.10. 函數管理模組(Function Management module)

當使用者選擇一個設備功能函數(例如純量 scalar)，IoTtalk 系統將取得所選定之功能資訊，根據此資訊，圖形介面可顯示功能名稱（“Function Name”）輸入方塊（圖 2.10 (d))、版本“Version”下拉式選單(圖 2.10(e))、以及這個函數的 Python 程式碼(圖 2.10 (f)).當函數編輯完成，使用者點選儲存“Save”按鍵(圖 2.10 (g))後，IoTtalk 系統會儲存此一函數資訊，並在資料庫中產生一個新版本的設備功能函數。之後，圖形介面就會以這個新版本項目重繪版本(“Version”)下拉式選單。假如使用者在加速器函數列表(Acceleration Function List)中選擇增加新函數“add new function”(圖 2.10 (h))，函數名稱“Function Name”輸入方塊、版本“Version” 下拉式選單、以及函數內容都將從圖 2.10(d)、(e)、與(f)中被清除，圖形介面將提供空白欄位給使用者編輯這個新函數。當使用者儲存這個新函數，圖形介面不僅會重繪版本(“Version”)下拉式選單，也會同時以這個新的函數項目重繪加速器函數列表(Acceleration Function List)。當使用者選擇一個設備功能函數並且按下刪除“Delete”按鍵(圖 2.10(i))時，IoTtalk 系統會從資料庫中移除這個函數資訊。之後，圖形介面將清除這個函數名稱，並且重繪新的特定設備功能函數列表(Specific DF Function List)。

一個設備功能函數組織如下：使用者點選“>>>”按鍵(圖 2.10(j))，將選定的函數從總體函數列表(Global Function List)移至特定設備功能函數列表(Specific DF Function List)(例如，圖 2.10(b)之 Acceleration Function List)。這樣做，IoTtalk 系統會更新資料庫中的函數資訊，而圖形介面也會重繪特定設備功能函數列表(Specific DF Function List)，以便將這個選定的函數當成列表中的一個項目。相似於“>>>”操作，假如使用者點選“<<<” 按鍵(圖 2.10 (k))，IoTtalk 系統會將這個函數從特定設備功能函數列表移至總體函數列表。

2.3. 設備功能與模型管理(Device Feature and Model Management)

IoTtalk 系統提供設備功能與模型管理視窗(圖 2.11)，用以建立與管理設備功能與模型管理，此一視窗包含一個設備功能與模型“Device Feature/Device Model”(DF/DM)開關按鈕(toggle button)(圖 2.11(a))、一個設備功能分類條 category bar(圖 2.11(b))、設備功能視窗 Device Feature Window(圖 2.11(c))、以及設備模型視窗 Device Model Window(圖 2.11(d))。當在開關按鈕中的設備功能“Device Feature”模式被選用時(圖 2.11(a))，這個設備功能視窗(Device Feature Window)將會被啟動，此時使用者可以編輯一個新的或已經存在的設備功能。

在另一方面，如果在開關按鈕中的設備模型(“Device Model”)模式被選用時，將會跳出設備模型視窗(Device Model Window)，以便讓使用者產生或編輯這個設備模型。設備功能分類條會列出所有的設備功能分類，包括視覺(Sight)、觸覺(Touch)、回聲定位(Echolocation)、聽覺(Hearing)、溫度(Temperature)、動作(Motion)、化學感測器(Chemoreceptor)、動覺(Kinesthetic)以及磁力感知器(Magnetoception)等。新版本的 IoTtalk 系統將會減少功能分類數，只包括視覺(Sight)、聽覺(Hearing)、感覺(Feeling)、動作(Motion)、以及其他。

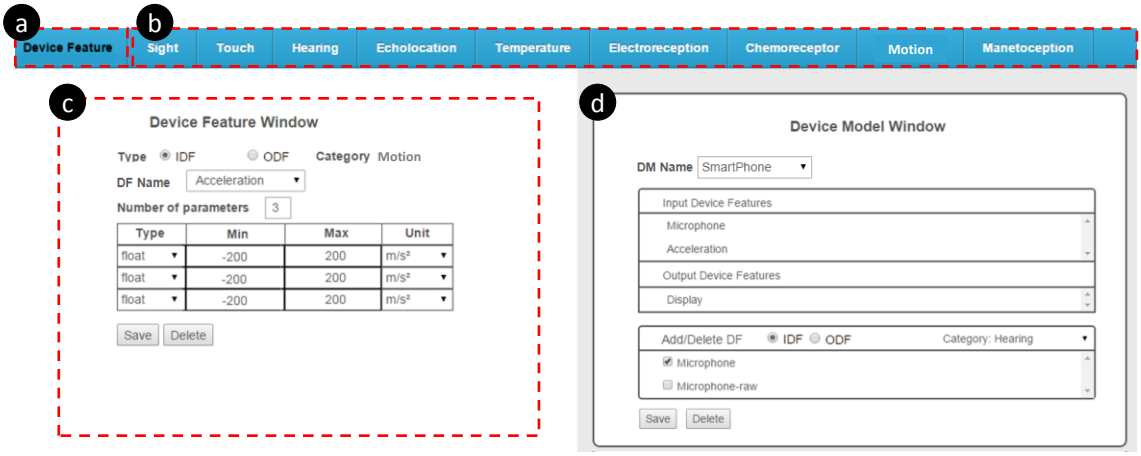

圖 2.11. 設備功能與模型管理視窗(Device Feature and Model Management Window)

2.3.1. 設備功能管理(Device Feature Management)

使用者可以按壓“DF/DM”按鍵開關至設備功能(“Device Feature”)模式，以便存取設備功能視窗。

在選定設備功能模式後，使用者可以在 DF 分類條下，點選 DF 分類(例如，視覺(Sight); 圖 2.12(a))，圖形介面會顯示兩個單選按鈕(radio buttons)分別提供 IDF/ODF 類型選擇(圖 2.12 (b))，並顯示所選定之 DF 分類(圖 2.12 (c))，其中， IDF 類型被自動選擇成預設型態，IoTtalk 系統會針對所選定之 DF 分類與 DF 類型，取得所有的設備功能(DF)(圖 2.12 (d))，這列表中第一個項目是可以用來產生新設備功能的增加新設備功能(“add new DF”)，其細節將會在下一子節中說明。假如使用者按壓 ODF 單選按鈕(圖 2.12 (b))，所有在“DF Name”列表中的 IDF 輸入設備功能 (圖 2.12 (d)) 都將被新選擇的輸出設備功能所取代。

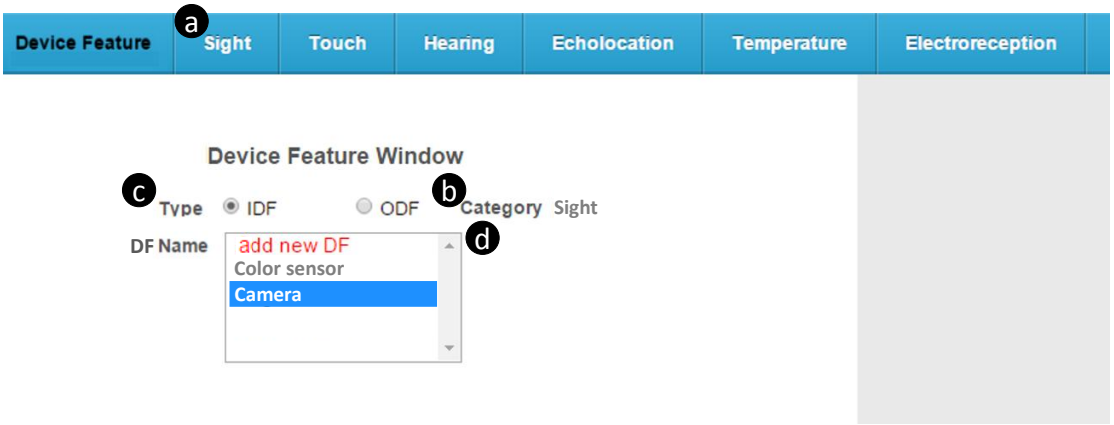

圖 2.12. 設備功能視窗(Device Feature Window)

為了產生新的設備功能，使用者可以在設備功能名稱“DF Name”列表中(圖 2.12 (d))，選擇“add new DF”模式，此時圖形介面將在設備功能視窗(Device Feature Window)中跳出設備功能參數模組(DF Parameter module)(圖 2.13(a)，設備功能參數模組的列數是根據參數的數目所產生(圖 2.13 (b))，其預設的參數為 1。對每一個 DF 參數，DF 參數模組包括資料型態(Type)(例如，浮點數 float, 字串 string 等;參考圖 2.13 (c))、最小值(Min)與最大值(Max)(圖 2.13(d))、以及單位(Unit)(例如, cm, m/s^2 等等; 圖 2.13 (e))。對於一個輸入設備功能而言，最小值/最大值都會自動指定，不需要使用者另行填入。但是，對於一個輸出設備功能而言，如果沒有填寫最小值/最大值，ODF 參數將可能是任意值，沒有範圍限制。使用者可以參考設備出廠之建議值規格填寫。舉例而言，圖 1.1 中 D3 的燈泡流明(luminance)值之範圍為[0, 500]。當使用者點選儲存“Save”按鍵(圖 2.13 (f))，圖形介面會跳出一個對話框讓使用者可以輸入新設備功能之名稱(圖 2.13 (g))。之後，IoTtalk 系統就儲存設備功能資訊到資料庫。值得注意的是，圖 2.13 (c)之“Type”欄位代表資料型態，而不是指圖 2.6 (c)管理視窗中，IDF 輸入設備功能模組的範例類型(“Type”)。

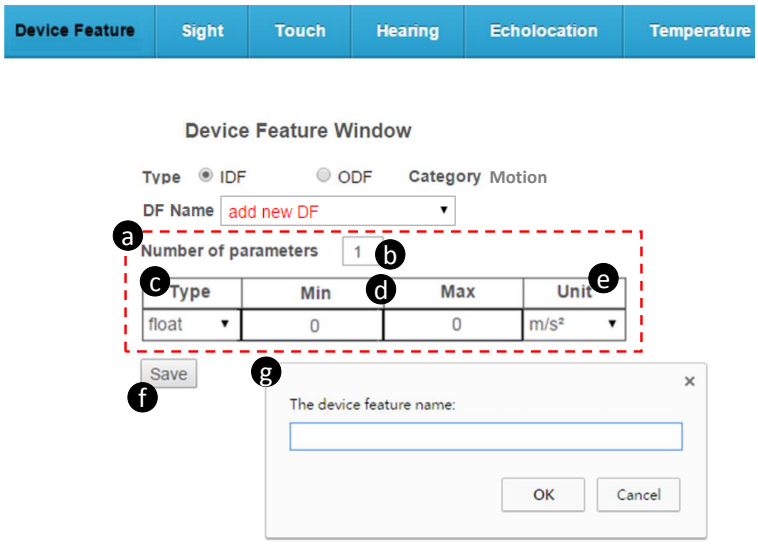

圖 2.13. 設備功能建立(Device feature creation)

從設備功能名稱(“DF Name”)列表中，使用者可以選擇並編輯一個已經存在的 DF (例如，圖 2.14 (a) 加速器 Acceleration)。假如使用者修改設備功能參數數目(圖 2.14 (b)), 設備功能參數模組將會根據新的數目被重新重繪。使用者可以編輯資料型態(data type)、最小值/最大值(Min/Max)、以及每一個參數的單位。在使用者完成編輯設備功能並且點選儲存“Save”按鍵(圖 2.14 (c))後，將會跳出一個對話框讓使用者可以確認這次修改 (圖 2.14 (d))。設備功能名稱將顯示於對話框中，允許使用者對已經修改的設備功能名稱進行變更，當使用者按下對話框之“OK” 按鍵(圖 2.14 (e))，IoTtalk 系統就會將此資訊儲存於資料庫中。但如果使用者按下對話框之“Delete”按鍵(圖 2.14(f))，IoTtalk 系統就會將此設備功能從資料庫中移除。

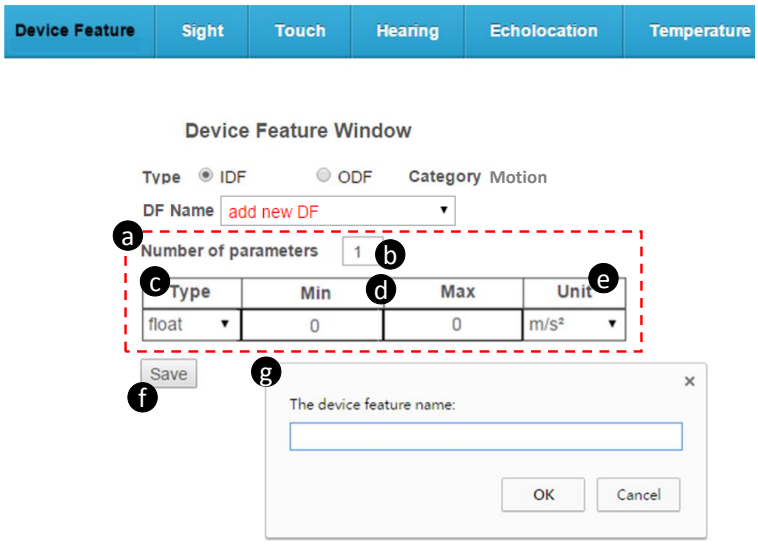

圖 2.14. 設備功能修改(Device feature modification) (*figure is replaced by figure 6.4 easyplug-In paper)

2.3.2. 設備模型管理(Device Model Management)

使用者可以按下“DF/DM”模式到設備模式(“Device Model”)來管理設備模型(圖 2.15 (a))。按下之後，就會跳出設備模型視窗(Device Model Window) (圖 2.15 (b)) 以及設備功能視窗(Device Feature Window) (圖 2.15 (c))。設備模型視窗顯示“DM Name”下拉式選單(圖 2.15(d))，由此使用者可以選擇一個設備模型，列表中第一個項目是增加新的設備模型(“add new DM”)，可以用來建立新的模型。設備功能視窗也同時顯示於設備模式中，以方便使用者，因為當使用者正在設定一個設備模型時，可能需要也知道某一個特殊設備功能之細節。

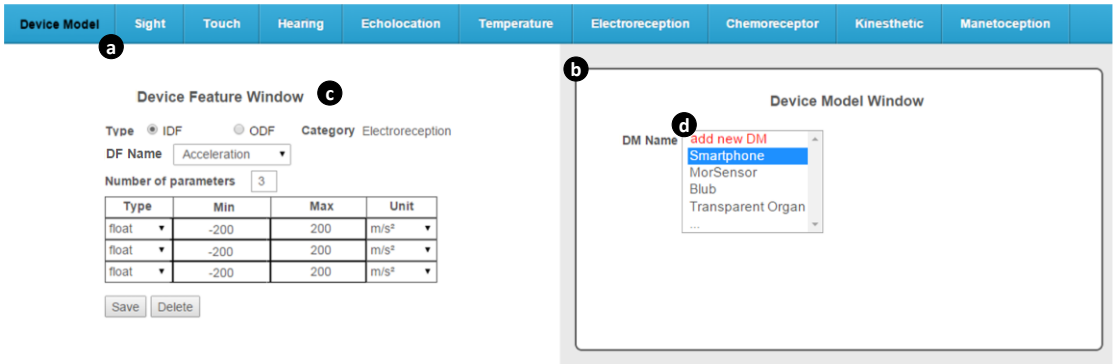

圖 2.15. 設備模型視窗(Device Model Window) (*Motion!)

為了建立一個新的設備模型，使用者可以在“DM Name”下拉式選單(圖 2.15 (d))中選擇增加一個新設備模型(“add new DM”)。之後圖形介面就會在設備模型視窗中跳出這個設備模型(圖 2.16(a))以及增刪設備功能(“Add/Delete DF”)模組(圖 2.16 (b))。這個設備模型會列出這個設備的所有功能。對一個新模型而言，設備功能模組初始值是空的，使用者可以經由增刪設備功能“Add/Delete DF”模組增加設備功能， 增加一個設備功能的操作如下:首先,使用者可以在設備功能分類條中選定一個設備功能類別，並經由增刪設備功能(“Add/Delete DF”)模組，從兩個選項按鈕中選定所需之設備功能(例如,圖 2.16 (c)中選定一個輸入設備功能 IDF),選完之後，增刪設備功能(“Add/Delete DF”)模組會顯示這個選定類別(圖 2.16(e))之所有的設備功能(圖 2.16(d))。當使用者在增刪設備功能模組中選定一個設備功能，這個設備功能就會自動地顯示於設備功能模組(圖 2.16 (a))，在使用者針對這個設備模型選完所有所需功能並且按下儲存“Save”按鍵(圖 2.16 (f))之後，圖形介面會跳出一個對話框讓使用者輸入這個新的設備模型名稱(圖 2.16(g))，同時 IoTtalk 系統也將在資料庫中建立這個新設備模型。

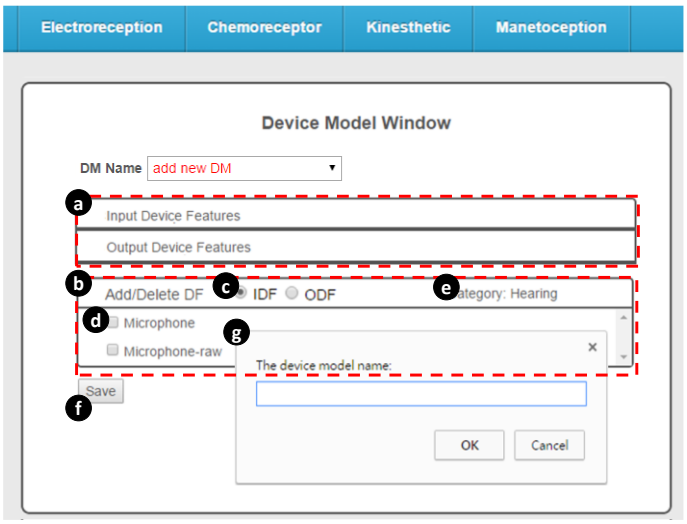

圖 2.16. 設備模型建立(Device model creation)

如果要設定一個已存在之設備模型，使用者可以在設備模型名稱(“DM Name”)下拉式選單中選定這個設備模型(例如,圖 2.17 (a)之 Smartphone)，選定之後，IoTtalk 系統會從資料庫中取出這個設備模型之設備功能，同時圖形介面也會跳出這個設備模型(圖 2.17(b))以及增刪設備模型(“Add/Delete DF”)模組(圖 2.17(c))。

如果要獲得一個特定設備功能之詳細資料,使用者可以將滑鼠指標移到

設備模組或增刪設備模組之設備功能名稱上(例如，圖 2.17 (d)之 Acceleration)，IoTtalk 系統就會取得所選擇的設備功能之細節資料，同時以圖形介面顯示在設備功能視窗(Device Feature Window)中(圖 2.17(e))，利用這個方式，使用者可以方便地檢視在設備模型視窗中所選定之設備功能。

假如使用者想要從設備模型中刪除一個設備功能(例如麥克風 Microphone)，他可以在設備功能模組中點選這個設備功能(圖 2.17(f))，此時這個設備功能也會自動地顯示在增刪設備功能(“Add/Delete DF”)模組中，之後使用者可以在增刪設備功能模組(圖 2.17(g))取消選定，並將他從設備模型中移除。

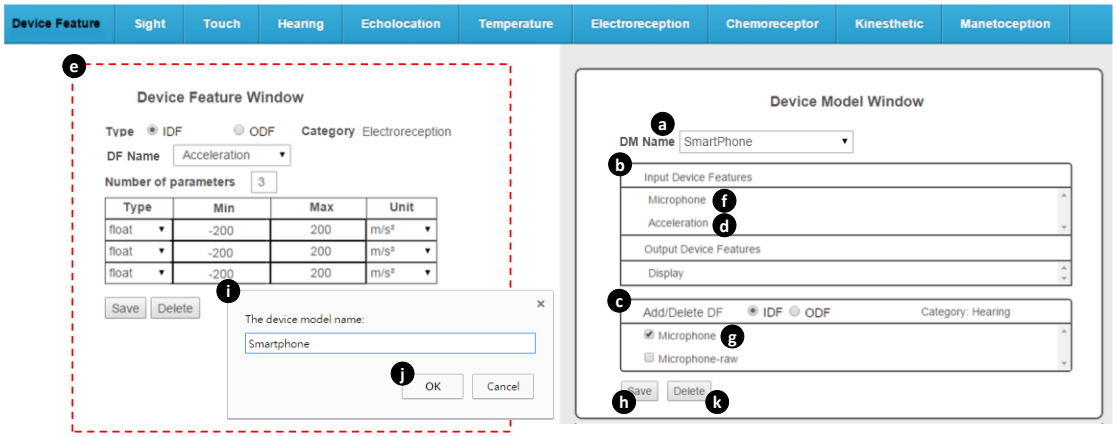

圖 2.17. 設備模型修改(Device model modification) (*Motion!)

在使用者完成設定設備模型並點選儲存(“Save”)按鍵(圖 2.17(h))之後，會跳出一個對話框以便使用者確定此修改(圖 2.17(i))，同時設備模型名稱也將出現於對話框中，以方便使用者將這個修改過之設備模型重新命名。當使用者按下對話框之確認(“OK”) 按鍵(圖 2.17 (j))，IoTtalk 系統會將這個特定的設備模型儲存於資料庫之中。反之，如果使用者按下刪除(“Delete”)按鍵(圖 2.17 (k))，IoTtalk 系統則會將此設備模型從資料庫中移除。