

透過兩階層雲端動態路徑策略 以降低展場之擁塞機率

江茂綸 朝陽科技大學 資訊與通訊系 mlchiang@cyut.edu.tw	張立傑 朝陽科技大學 資訊與通訊系 clock@cc.hwh.edu.tws	鐘子智 華夏技術學院 數位媒體設計系 10130604@cyut.edu.tw	洪梓奇 朝陽科技大學 資訊與通訊系 s9930024@cyut.edu.tw
林昭宏 朝陽科技大學 資訊與通訊系 s9930046@cyut.edu.tw	陳中岳 朝陽科技大學 資訊與通訊系 s9930074@cyut.edu.tw	涂凱翔 朝陽科技大學 資訊與通訊系 s9930092@cyut.edu.tw	張智凱 朝陽科技大學 資訊與通訊系 s9930096@cyut.edu.tw

摘要

近年來，參觀展覽成為許多民眾的重要休閒活動。然而，在大多數的展覽中，眾多的人潮將會降低了參觀的品質及造成安全上的顧慮。因此，本篇提出二階段雲端動態路線配置架構(Cloud Dynamic Path strategy)來結合了物聯網及雲端運算的概念，透過即時感測現場資訊，如壅塞及火災等，再送至雲端環境上進行分散式運算，進而回傳建議路徑給參觀者，來降低壅塞機率以確保參觀品質及觀展者的安全。此外，本文的第一階段為壅塞機率避免，主要是用來計算各區域的壅塞機率並建議出壅塞機率較低的路徑，進而降低展場各展區的壅塞程度。而第二階段為動態權重調整階段，主要透過觀展率來動態調整權重以降低熱門展區的壅塞率，並提升冷門展區的觀展機率，以達到觀展人數的平衡。

關鍵字：物聯網、雲端運算、感測網路、導覽系統

Abstract

In recent years, visiting an exhibition becomes one of the leisure activities for many people. However, there are crowds of people to reduce the quality of visiting and increase the security problem in the popular exhibition. Therefore, two-phase Cloud Dynamic Path strategy (CDP) we proposed to combine concept of Internet of Things (IoT) and Cloud computing in this paper to collect real-time sensing information, such as congestion and fires. Subsequently, the collected information can be uploaded to the cloud distributed computing environment to recommend the visit path to reduce the probability of congestion in order to

ensure the quality of visit and security of visitors. Moreover, the first phase of this paper is Anti-Congestion Probability phase, we calculate probability of the regional congestion and suggest a path with low congestion in order to reduce the congestion for each exhibition regional. The second phase is Dynamic Weight Adjust phase, we through probability of visit to adjust the weights dynamically in order to reduce congestion probability of popular exhibition and balance the number of visitors.

Keywords: Cloud computing, Internet of Things, sensor network, Guide System

1. 前言

雲端運算(Cloud computing)[4]為透過分散式的概念將工作分配到多台主機來同時運算以達到高效能及高儲存容量的能力。特別是針對有大量資料及參觀人數的展場而言，將可快速提供即時及巨量資料分析能力。

物聯網(Internet of Things, IoT)[12][16]為近年來最熱門議題之一，其主要是需要在特定物體上貼上微型感測晶片來達到物物相連的網際網路型態，實現物品之間的溝通和對話，以提供管理和服務功能，如使用無線射頻辨識(RFID)、感測器(Sensor)、及無線通訊晶片(Wireless Communication Chip)等微型感測晶片。

因此，本展場導覽系統[7][9]為了提升參觀展覽的品質，我們結合目前最廣泛被使用的分散式儲存系統是 Google File System(GFS)[8]以及 Hadoop Distribution File System(HDFS) [1][3][15]來實作雲端運算架構。隨後，再增設感測設備將現在訊息收 HBase[15]中，結合壅塞機率(Congestion Probability; CP)[5][13]及動態權重值 (Dynamic weight value; DW)[6]來動態導引觀展者至最佳路徑，以降低展場壅塞程

度以達到展場各區域參觀人數的平衡。

此外，本文也結合感測模組與物聯網概念來針對現在溫溼度進行監控，若感測到展場溫度過高，則表示可能有火警發生，則本系統將會自動觸發警示燈，並依參觀者目前所在位置來顯示出現場逃生路線的標記，而參觀者也可透過手機來參考逃生路線的導覽，以利於快速疏散。因此，透過本系統不但可以透過事先規劃導覽路線來避開人潮壅擠的展區以提升觀展品質，並可透過物聯網來提升展場內觀展者的安全。

本文第二節為文獻探討，首先說明目前所使用的雲端技術及架構；而在第三節中，將說明本研究所提出之兩階段雲端動態路線配置；而在第四節中將進行二階段實例說明；最後，第五節為結論。

2. 文獻探討

2.1 HDFS

Hadoop Distributed File System (HDFS)是Hadoop[15]專案中的分散式存儲主要格式，可將分散的儲存資源整合成一個具容錯能力、高效率且超大容量的儲存環境，在Hadoop系統中大量的資料和運算時所產生的暫存檔案，都是存放在這個分散式的檔案系統上。

HDFS 是 master/slave 架構，由兩種角色組成，名稱節點(Name node)及資料節點(Data nodes)，Name node 負責檔案系統中各個檔案屬性權限等資訊 (metadata, namespace) 的管理及儲存；而 Data node 通常由數以百計的節點擔任，一個資料檔會被切割成數個較小的區塊，儲存在不同的 Data node 上，每一個區塊還會有數份副本存放在不同節點，可做到異地備份的功能，當其中一個節點損壞時，還能確保系統檔案中的資料保存完整。因此 Name node 還需要紀錄每一份檔案存放的位置，當有存取檔案的需求時，將會協調 Data node 來負責回應；此外，Name node 也會在節點損壞時自動進行資料的搬遷和複製。

2.2 HBase 資料庫

HBase[15]是由 BigTable[14]所衍生出來支援 Hadoop 的資料庫。其主要的作法是使用列(row)和行(column)來作為索引存取資料值。而HBase的另一個特點是每一筆資料都有一個時間戳記(timestamp)，因此同一個欄位可依不同

時間存在多筆資料。透過上述的特點可得知HBase 是被使用在即時隨機存取大型資料庫。

然而，HBase 底層則是使用了分散式的檔案系統HDFS (Hadoop Distributed File System)來進行儲存，因此除了資料可分散存放的特性之外，每一個分散的資料區塊都可各自擁有獨立的運算能力。當使用者對整個資料庫系統進行操作的時候，分散的資料則會各自獨立運算整理，並同步處理使用者的要求，來達到高效能分散式運算，進而加速整體系統的回應時間，以提升導覽效能。

2.3 Arduino

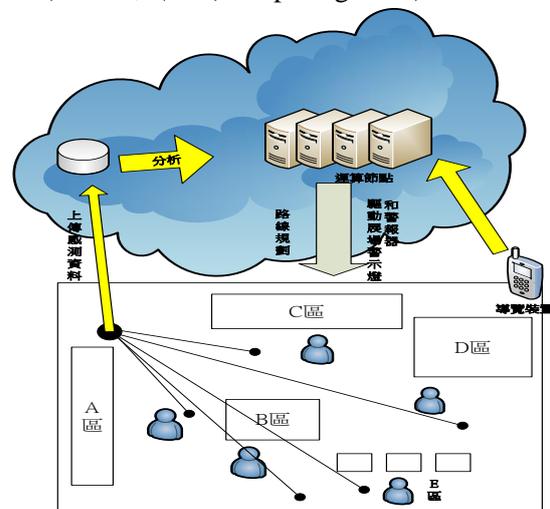
Arduino[2][16]提供了簡單好用的 I/O 介面，並具可使用類似於 Java、C 語言等高階語言的開發環境，因此我們透過 Arduino 來進行即時的溫濕度及壅塞程度的感測以實現物聯網概念。

Arduino 設計之初的目的是希望讓設計師和藝術家們能夠很快地透過它，學習電子和感測器的基礎知識，並應用到他們的設計當中。

Arduino 的出現,大大降低了互動設計的門檻，沒有學過電子知識的人也能夠使用它製作出各種充滿創意的作品。為了針對不同的應用領域，目前 Arduino 已設計出很多不同的型號以滿足不同使用者的需要。

3. 研究方法

本節主要架構於雲端運算儲存架構及物聯網環境上來探討壅塞程度及動態權重的議題，進而提出一個雲端展場導覽系統，其架構如圖一所示。而本架構主要分為三部分來進行探討，感測節點(Sensing node)、儲存節點(Storing node)、運算節點(Computing node)。

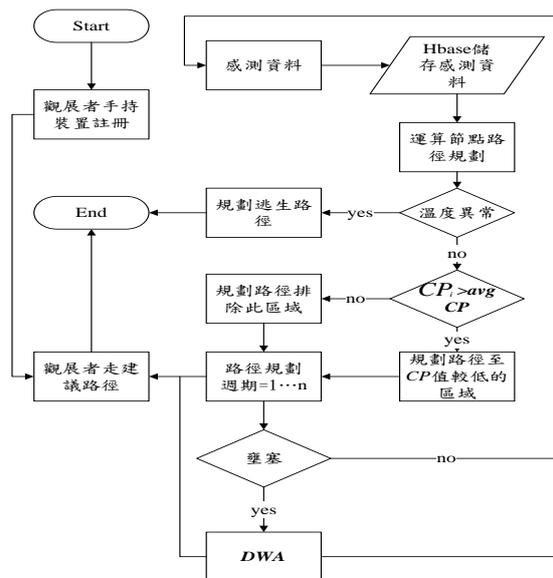


圖一、系統架構圖

感測節點(Sensing node)主要分為壅塞感測及溫溼度感測。感測節點會將感測資訊傳至叢集頭，再由叢集頭負責將收集的資訊上傳至儲存節點(Storing node)中。儲存節點(Storing node)，則負責將上傳的資訊分散存放至 Hbase 資料庫中。而運算節點(Computing node)會將存於 HBase 資料庫中的感測資訊透過 Map/Reduce 的方式[10][11]進行分散式運算後，再提供參觀者建議路徑。

隨後，我們針對系統流程進行說明。首先，觀展者在進入展場前必須進行註冊，系統會依據觀展者進入時間給予不同的建議路徑。而在觀展期間，感測節點會不斷的將感測資訊上傳至雲端資料庫 HBase 進行儲存，並經由雲端運算節點來進行分析判斷及監控。若本展場溫度發生異常，則運算節點將會替每一個觀展者來規劃逃生路徑，導引至各展場最近的出口，以確保觀展者的安全。此外，系統也將透過壅塞機率(Congestion Probability; CP)來判斷各展區的壅塞率，而當 CP_i 大於平均值的 CP 時，則表示此區域壅塞程度嚴重。因此，運算節點也將為觀展者進行規劃出較不擁塞的建議路線，以避免展區壅塞。隨後，本文透過第二階段 DWA 來動態調整各展區的權重並重新規劃建議路徑以降低熱門展區的壅塞率，並提升冷門展區的參觀人次，以達到參觀人數上的平衡。其相關流程圖如下圖二所示。

一般來說，當觀展者進入展場後，系統為了達到參觀人數的平衡，則會根據 HBase 收集的各區域壅塞程度及濕度資訊來快速進行運算，以提出的動態路線配置策略(Cloud Dynamic Path strategy; CDP)來動態配置路線。因此，本文提出 CDP 將會透過兩階段來處理及配置導覽路徑，進而維持整個展場配置的平衡以提升服務品質及觀展者安全。其第一階段為壅塞機率避免階段(Anti-Congestion Probability phase; ACP)及第二階段為動態權重調整階段(Dynamic Weight Adjust phase; DWA)將於下節詳細說明。



圖二、系統流程圖

3.1 壅塞機率避免階段 (Anti-Congestion Probability phase; ACP)

本階段透過先前學者[13]所使用的請求到達率及延遲時間來預先篩選出各區域中擁有較低壅塞機率(Congestion Probability; CP)的展場區域，以避免在單一區域中人潮過多而造成觀展品質降低，其相關公式如下所示。

壅塞機率(CP_i):

公式(1)中 M_i 表示 A_i 展區中擁有的展品數量， Pr_j 為 A_i 展區中擁有與其他展區相同展品的數量， p_j 為此展區中所有展品被參觀的熱門度。而 λ 為得知此展覽民眾的人數中真正到達此展區並進行觀展的數量，稱之為到達率 λ 。隨後，將可計算出每個展區的壅塞機率 CP_i 。而當某展區有較熱門的展品時，則會造成人潮壅擠，此時新進入的民眾就必須等待，此現象我們稱為壅塞。因此，我們透過 M/M/1[5] 來算出各個區域的壅塞機率(CP_i)。

$$CP_i = \sum_{j=1}^{M_i} \frac{p_j}{Pr_j} \lambda \quad (1)$$

3.2 動態權重調整階段 (Dynamic Weight Adjust phase; DWA)

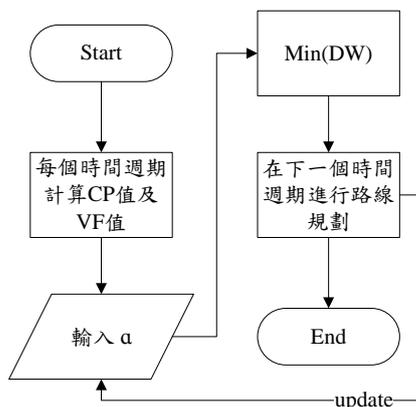
而當熱門展區的到達率增加而壅塞機率提高時，再透過第二階段的更新及動態權重調整階段(Dynamic Weight Adjust phase; DWA)來平衡各個展區的觀展人數。

由於第一階段針對提供 CP 值較低的建議

路徑，但是卻忽略展場中各個展區之間觀展率是否達到平衡。因此，本文透過第二階段來達到觀展人數的平衡。其主要作法是透過第一階段各週期的歷史紀錄，來計算及更新平均觀展率並分配權重值(α)，以降低熱門區域可能發生壅塞的可能，並提升冷門區域的觀展機率。隨後，並預測下一時間在各展區中觀展率，以期讓各展區之間的觀展人數更趨近於平衡。而本階段的權重係數(α)主要由 CP 值來訂定，主要是因為在整個選擇流程中，壅塞機率的高低會影響展區的壅塞程度。此外，壅塞機率(CP)和參觀次數(Visit Frequency; VF)也將是會相互影響的因子。其動態權重(DW)公式及詳細流程圖如公式二及圖三所示。

$$\text{Min}(DW) = (\alpha) CP_i + (1-\alpha) VF_i \quad (2)$$

公式(2)中動態權重值(DW)是指透過第一階段歷史紀錄得到各區域的平均觀展率($\text{avg}(VF)$)進行區域之間互相評估比較，隨後透過平均觀展率來進行預測及分配權重(α)，然而分配權重(α)較低則表示此區域較為熱門需減少阻塞發生的機率，若分配權重(α)較高則為較冷門區域，須提高觀展率，因此，我們透過公式(2)來計算出 $\text{Min}(DW)$ ，隨後我們將建議較熱門區域的觀展者轉移至較冷門區域以達到展場區域之間的觀展人數趨近於平衡。



圖三、動態權重調整流程圖

4. 實例說明

本節將提供一個實例來說明兩階段所提出的動態路線配置策略。首先我們在此畫展中擁有五個展區，假設每個展區擁有兩件展品且每幅畫都不相同。因此我們在此訂定展區中擁有

的展品數量 M_i 為 2 及展區中擁有與其他展區相同展品的數量 Pr_j 為 1，並進行收集各節點讀取次數。在此實例中，假設 5 個展場區域分別是 $N1$ 、 $N2$ 、 $N3$ 、 $N4$ 、 $N5$ ，而在時間點 $t1$ 下，表一展示出各展品在展區中的被參觀次數。

表一、時間點 $t1$ 下的各展區展品的被參觀次數

Area ID	Product 1 Visited times	Product 2 Visited times
A ₁	13	11
A ₂	23	25
A ₃	32	30
A ₄	28	35
A ₅	20	25

隨後，我們將針對公式(1)的 λ 、 p_j 、 Pr_j 參數進行定義：

λ :到達率。我們以 poisson[5]概念來進行對 λ 的定義，假設預計觀展的民眾有 200 位，而真正到達展場並進入參觀的民眾只有 126 位，然而其他 74 位民眾針對參觀此展覽的意願並不高，因此，計算後得到的到達率為 0.63。

Pr_j :展品。假設在 A_i 展區中擁有 3 件展品，我們會針對每件展品進行相同展品數的統計。

p_j :熱門度。如上述在 A_i 展區的各個展品參觀次數與展場中擁有相同展品的參觀次數加總進而得知熱門度。

隨後我們將進行分別算出各展區的熱門度，如下表二：

表二、節點熱門度 p_i

Area ID		p_i
A ₁	$A_{1_p1} = \frac{13}{242}$ $A_{1_p2} = \frac{11}{242}$	$p_1=0.054$ $p_2=0.045$
A ₂	$A_{2_p1} = \frac{23}{242}$ $A_{2_p2} = \frac{25}{242}$	$p_1=0.095$ $p_2=0.103$
A ₃	$A_{3_p1} = \frac{32}{242}$ $A_{3_p2} = \frac{30}{242}$	$p_1=0.132$ $p_2=0.124$
A ₄	$A_{4_p1} = \frac{28}{242}$ $A_{4_p2} = \frac{35}{242}$	$p_1=0.116$ $p_2=0.145$
A ₅	$A_{5_p1} = \frac{20}{242}$ $A_{5_p2} = \frac{25}{242}$	$p_1=0.083$ $p_2=0.103$

我們針對 CP 值進行計算，計算結果如下表三：

表三、區域壅塞機率 CP_i

Area ID	CP_i
A ₁	$CP_1 = (\frac{0.054 * 0.63}{1}) + (\frac{0.045 * 0.63}{1}) = 0.062$
A ₂	$CP_2 = 0.125$
A ₃	$CP_3 = 0.161$
A ₄	$CP_4 = 0.164$
A ₅	$CP_5 = 0.12$

計算完後我們必須做挑選的動作，所以接著我們計算表四的平均值 avg(CP)當作門檻值來進行篩選，平均值 avg(CP)為 0.126。

透過 avg(CP)門檻值的設定，我們可以在壅塞避免階段篩選出小於門檻值的區域，Area 1、Area 2 及 Area 5。隨後，我們透過運算節點進行路徑規劃，規劃的建議路線會避開壅塞機率較高的 Area 3 及 Area 4 以提供觀展者較佳的觀展品質。

隨後，本文將透 5 個時間階段來記錄各區域的參觀數來說明第二階段所提出的動態權重調整階段，其觀展數如下表四：

表四、展區觀展數

Time	Area 1	Area 2	Area 3	Area 4	Area 5
t-1	5	8	5	2	9
t-2	9	3	3	2	6
t-3	4	3	9	3	5
t-4	9	6	5	3	4
t-5	10	2	3	6	3

透過表四中觀展數的歷史紀錄，我們分別利用第一階段提出的 CP 公式分別計算出各節點參數值。

表五、展區參數值計算

	Area 1	Area 2	Area 3	Area 4	Area 5
VF	37	22	25	16	27
avg(VF)	7.4	4.4	5	3.2	5.4
CP	0.16	0.1	0.11	0.07	0.12

avg(CP)	0.112
---------	-------

透過表五中觀展數的歷史紀錄來計算出平均觀展率，進而分配 α 值並預測在下一個時間週期中的工作權重(DW)。因此，我們將建議觀展者可以從 DW 較大的展區轉移至 DW 較小的展區，並進行路徑的重新規劃。所以，本文可透過觀展率來調整 α 值，進而改善第一階段各展區之間觀展人數的不平衡問題。隨後，再透過第二階段 α 值的調整以提高展場品質及減少危害的發生。然而，在下次阻塞發生時，本系統也會進行歷史紀錄的評估，進而分配權重值 α 以達到參觀人數上的平衡，其詳細說明如下表六。

表六、下一時間工作預測

	Area 1	Area 2	Area 3	Area 4	Area 5
α	0.3*C P+0.7 *VF	0.6*C P+0.4 *VF	0.5*CP +0.5*V F	0.8*CP +0.2*V F	0.4*C P+0.6 *VF
t+1	5.23	1.82	2.55	0.7	3.28

透過表六，我們可以計算出下一週期時間中各區域間的權重值 α ，若發現平均觀展率較其他區域高則將 α 值設定較小的權重以降低阻塞，如表五中 Area 1 平均參展率為 7.4，高於其他展區，則在表六中可看到我們可依據使用者自訂或實驗結果來設定較小的 α 值為 0.3，而 Area 4 參展率為 3.2 較其他展區低，則設定較大的 α 值為 0.8。因此，以本例來說，位於 Area 1 的觀展者人數將會較多，所以對於即將進入的觀展者的規劃路線將會儘量的避開該區，並優先推薦 Area 4。

因此，根據上述結果，本文所提出之動態路線配置策略將可依每個週期中各區域的壅塞程度的不同來進行權重值的動態調整，並透過建議路徑來將降低熱門區域或提高冷門區域的觀展人數，以達到觀展人數的平衡，進而提高展場品質及減少危害的發生。

5. 結論

傳統大多數的展場中並未完整規劃觀展者的觀展動線，導致許多民眾因為某些特定區域較為熱門，因而造成壅塞，導致觀展品質不佳。此外，對於展場現場資訊的不充足，也將會造成相關展場的安全問題。因此，本文提出

雲端動態路線策略 (Cloud Dynamic Path strategy; CDP) 並結合物聯網物品之間的溝通概念來進行溫度感測及壅塞程度感測，進而快速規劃出觀展路徑，以期降低壅塞機率，並平衡各展覽品的參觀人數以提升展場服務品質，此外，本系統也透過即時溫度感測以提升展場安全性。

6. 誌謝

這篇論文是國科會計畫 (NSC 102-2221-E-324-014) 研究成果的一部份，我們在此感謝國科會經費支持這個計畫的研究。

7. 參考文獻

- [1] Apache Hadoop Wiki, http://en.wikipedia.org/wiki/Apache_Hadoop
- [2] Arduino: <http://www.Arduino.cc/>
- [3] The Apache Software Foundation Hadoop, <http://hadoop.apache.org/core/,2009.1000464>
- [4] M. Armbrust, et al., "A view of cloud computing," *Communications of the ACM*, Vol. 53, Issue 4, pp. 50-58, Apr. 2010.
- [5] A.W. Berger, et al., "Comparison of call gapping and percent blocking for overload control in distributed switching systems and telecommunications networks," *IEEE Transactions on Communications*, Vol. 39, Issue 4, pp. 574-580, Apr. 1991.
- [6] M. Bjorkqvist, et al., "Load-Balancing Dynamic Service Binding in Composition Execution Engines," *IEEE Asia-Pacific Services Computing Conference (APSCC)*, PP.67-74, Dec. 2010.
- [7] J. Choi, et al., "Software Engineering Issues in Developing a Context-aware Exhibition Guide System," *International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD)*, pp. 840-845, Aug. 2008.
- [8] S. Ghemawat, et al., "The Google file system," *Proceedings of the nineteenth ACM symposium on Operating systems principles*, New York, USA, pp. 29-43, Oct. 2003.
- [9] Y. P. Huang, et al., "Enhanced Interactivity in Learning-Guide Systems with RFID," *Annual RFID Eurasia*, PP. 1-6, Sept. 2007.
- [10] K.H. Lee, et al., "Parallel data processing with MapReduce: a survey," *ACM SIGMOD Record*, Vol. 40, Issue 4, pp.11-20, Dec. 2011.
- [11] N. Nurain, et al., "An In-depth Study of Map Reduce in Cloud Environment," *2012 International Conference on Advanced Computer Science Applications and Technologies(ACSAT)*, pp. 263-268, Nov. 2012.
- [12] C. Perera, et al. "Context Aware Computing for The Internet of Things: A survey," *IEEE Communications Surveys & Tutorials*, Vol. PP, Issue 99, pp. 1-41, May. 2013.
- [13] W. Qingsong, et al., "CDRM: A Cost-Effective Dynamic Replication Management Scheme for Cloud Storage Cluster," *IEEE International Conference on Cluster Computing (CLUSTER)*, pp. 188-196, Sept. 2010.
- [14] S. Ramanathan, et al., "Comparison of Cloud database: Amazon's SimpleDB and Google's Bigtable," *International Conference on Recent Trends in Information Systems (ReTIS)*, pp. 165-168, Dec. 2011.
- [15] M.N. Vora, "Hadoop-HBase for Large-Scale Data," *International Conference on Computer Science and Network Technology (ICCSNT)*, Vol. 1, pp. 601 – 605, Dec. 2011.
- [16] T. Yamanoue, et al., "A M2M System Using Arduino, Android and Wiki Software," *International Conference on Advanced Applied Informatics (IIAIAAI)*, PP. 123-128, Sept. 2012.