

行政院國家科學委員會專題研究計畫 成果報告

動態網路服務之整合性階層式部署之研究與發展 研究成果報告(精簡版)

計畫類別：個別型
計畫編號：NSC 100-2221-E-029-020-
執行期間：100年08月01日至101年07月31日
執行單位：東海大學資訊工程學系

計畫主持人：陳隆彬

計畫參與人員：碩士班研究生-兼任助理人員：陳志賢
碩士班研究生-兼任助理人員：李宸光

公開資訊：本計畫涉及專利或其他智慧財產權，2年後可公開查詢

中華民國 101 年 12 月 07 日

中文摘要：一個雲端服務應用程式可能包含多個服務元件，這些元件的運作成本包含了計算成本以及資料傳輸成本。客戶通常會將元件複本佈署在對計費方式較有利的伺服器節點中。研究顯示，只傳送所需要的物件內容，而不更新所有的服務物件可以減少雲端服務的運作成本，並且提高整體效能。找出佈署策略之最低成本是一個困難的任務，因此很多研究將重點放在求較佳解的方法。本計畫探討階層式環境的服務元件的佈署策略的較佳解，我們先求出一組近似解，接著分析服務元件之間的流程相依關係，微調原先之近似解並進一步精簡。研究結果顯示靠著消除多餘的物件，部署成本能夠降低並且能夠提高網路服務效能。

中文關鍵詞：資料複本、內容遞送、雲端計算、Web 服務

英文摘要：Service-oriented architecture (SOA) is a computing paradigm uses services as basic building blocks to enable rapid and flexible development of software. This project aims to develop a Content Service System integrating the functionalities of digital document representation, In this project, a service application, comprising of many services, is represented by using the directed graph called SDG in which each vertex refers a service and each edge (a, b) refers to the dependency relation from service a to service b . In order to enhance the scalability and efficiency of the system, this project develops an efficient algorithm for deploying the nodes in SDG over the hierarchical network such that the total service operating cost is minimized.

英文關鍵詞：Web service, Service-Oriented Architecture, Content delivery, Cloud computing

動態網路服務之整合性階層式部署之研究與發展

計畫類別：個別型計畫 整合型計畫

計畫編號：NSC 100-2221-E-029-020-

執行期間：100 年 08 月 01 日至 101 年 07 月 31 日

執行機構及系所：東海大學

計畫主持人：陳隆彬 副教授(資訊工程學系)

共同主持人：

計畫參與人員：陳志賢，李宸光

本計畫除繳交成果報告外，另須繳交以下出國報告：

赴國外移地研究心得報告

赴大陸地區移地研究心得報告

出席國際學術會議心得報告及發表之論文

國際合作研究計畫國外研究報告

處理方式：除列管計畫及下列情形者外，得立即公開查詢

涉及專利或其他智慧財產權，一年二年後可公開查詢

中 華 民 國 101 年 10 月 1 日

摘要

一個雲端服務應用程式可能包含多個服務元件，這些元件的運作成本包含了計算成本以及資料傳輸成本。客戶通常會將元件複本佈署在對計費方式較有利的伺服器節點中。研究顯示，只傳送所需要的物件內容，而不更新所有的服務物件可以減少雲端服務的運作成本，並且提高整體效能。找出佈署策略之最低成本是一個困難的任務，因此很多研究將重點放在求較佳解的方法。本計畫探討階層式環境的服務元件的佈署策略的較佳解，我們先求出一組近似解，接著分析服務元件之間的流程相依關係，微調原先之近似解並進一步精簡。研究結果顯示靠著消除多餘的物件，部署成本能夠降低並且能夠提高網路服務效能。

關鍵字：資料複本、動態網頁、雲端計算、Web service

CHAPTER 1 簡介

Web 應用是由 server 端執行複雜的商業邏輯運算，提供應用程式資料給 client 端的程序。透過雲端服務網路，web 服務應用程式已成為全面性的軟體。為增加應用程式的可用性，系統會透過複製內容元件並散布於多個網路節點 [1][2][3]。我們將討論在樹狀的階層式網路中，由根節點擔當 server 主機。階層式網路組織是一種常見的高擴展性的分散式計算基礎設施配套。網路的內容服務佈署一直是一個基本的應用問題，這吸引了許多研究工作來自學術界和各個行業[4]的投入。

如何在網路頻寬與主機計算能力下做一個平衡的取捨，這是個需要來探討的問題。如果當網路頻寬變小而嚴重延遲時，這時候就不適合做大量的資料傳輸；相反的，如果你的電腦運算速度是老舊較慢的，那麼寧可由網路取得資料一定會比自行計算還要有效率的多。在[3]作者研究發展有效的演算法，找出最小成本內容複本策略。

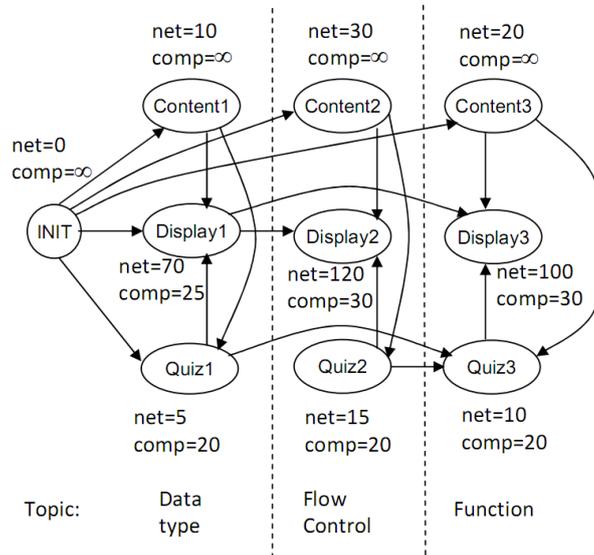
CHAPTER 2 問題定義

2.1 Object Dependency Graph (ODG)

動態web應用程序採用非循環的有向圖ODG (object dependency graph) 來代表內容應用程式，我們假設 $G = (V, E)$ 是ODG G 表示式， $V = \{v_0, v_1, \dots, v_n\}$ ， $v_i \in V$ 表示web上的服務物件，鏈結 $(v_i, v_j) \in E$ 表示物件 v_i 連向 v_j 的相依關係，其中 $0 \leq i, j \leq n$ 。

圖ODG G ，在一段edge上 $(v_i, v_j) \in E$ ，有一條途徑可以從 v_i 連向 v_j ，表示成 $v_i \rightarrow v_j$ 。 v_i 相對於 v_j 來講是前行節點 $\text{PRED}(v_i) = \{v_j | (v_j, v_i)\}$ ， v_j 相對於 v_i 也就被稱為後繼節點 $\text{SUCC}(v_i) = \{v_j | (v_i, v_j)\}$ 。如果 $\text{PRED}(v_i) = \{\emptyset\}$ 表示 v_i 是第一個節點，則稱為initial節點(I)表示；如果 $\text{SUCC}(v_i) = \{\emptyset\}$ 則代表 v_i 是最後一個終端節點，則稱為target節點(J)。假設集合 $W \subseteq V$ ，則 $\text{PRED}(W) = \{v_j | v_j \rightarrow v_i, v_i \in W, v_j \in V \setminus W\}$ 且 $\text{SUCC}(W) = \{v_j | v_i \rightarrow v_j, v_i \in W, v_j \in V \setminus W\}$ (我們使用簡易的數學表示方法來表達 $X \setminus Y$ ，即是將 $X - X \cap Y$ 的集合)。正因為ODG G 是一個非循環的有向圖，所以 $\text{PRED}(W) \cap \text{SUCC}(W) = \{\emptyset\}$ 。

以圖表 1 為例，ODG圖代表網路上提供的web版程式語言學習教材，每個大主題又包含內容、網頁的呈現、與測驗等三類物件。根據ODG問題模型，一個目標物件 v_i 可經由server直接傳送資料到client端，如果網路變小延遲過高(也就是成本較高)，目標物件 v_i 也可由client端自行計算，以減少傳輸上的成本。



圖表 1：web 版程式語言電子書 ODG。

2.2 佈署策略

階層式網路主機，假設由 $M + 1$ 個主機 (s_0, s_1, \dots, s_M) 組成的樹狀結構。對這 $M + 1$ 個主機就有 $M + 1$ 種佈署項目，分別對應到所有策略的集合 $((N_0, C_0), (N_1, C_1), \dots, (N_M, C_M))$ 。樹狀結構的網路會形成一個供需鏈，上游伺服器會提供下游伺服器物件所需的計算結果，網路廣播方向是由根結點主機往葉節點終端主機分散計算傳輸新的內容服務。考量物件在網路上的關聯性，並非所有佈署項目都是可行的(或稱合法的)，可行的佈署策略定義如下。

定義 2.1 (可行的佈署策略)：ODG $G = (V, E)$ ，階層式主機 (s_0, s_1, \dots, s_M)，對應的複本策略 $((N_0, C_0), (N_1, C_1), \dots, (N_M, C_M))$ 是一“可行的”佈署項目：

- (1) (Computable)：在同一主機上，一個物件只能有一種佈署方式 (O, N, C) ， $N_x \cap C_x = \{\emptyset\}$ ，其中 $0 \leq x \leq M$ 。其前行節點不可為 O -node， $I \cap C_x = \{\emptyset\}$ 。
- (2) (Supply-Demand)：根主機 s_0 ， $N_0 = I$ 。若主機 s_x 是根主機以外的節點 $s_q = \text{parent}(s_x)$ ，則 s_q 的 N -node N_x 由 s_x 透過網路提供， $N_x \subseteq (N_q \cup C_q)$ 。
- (3) (Fulfillment)：起始節點必須佈署在根主機 s_0 (root 主機)，終端節點必須佈署在終端主機 s_M (最底層葉節點終端主機)。 $T_x \subseteq (N_x \cup C_x)$ 。 ■

圖表 2 表示一個合法的階層式主機 (s_0, s_1, s_2) 佈署策略。圖中的 s_2 主機對應的佈署項目 (N_2, C_2) ，

- $\text{cap}(v_{i,e}, v_{i,b}) = \infty$

為方便起見，無限容量的edge又稱為 ∞ -edge。

以下特性 P1 證明H-cut不可順向切過任何 ∞ -edge否則成本會變成 ∞ ：

P1 對H-cut (X, Y) 與任何 ∞ -edge (v, w) ，條件 $v \in X \wedge w \in Y$ 不成立。反過來說， $(v \in X \wedge w \in X) \vee (v \in Y \wedge w \in Y) \vee (v \in Y \wedge w \in X)$ 成立。

令 $(X, Y) = \{v | (v_{i,b}, v_{i,e}) \in E, v_{i,b} \in X, v_{i,e} \in Y\}$ 為 N_x 的集合。上述H-cut與傳統切割(graph cut)定義有部分不同，根據[7]，最小H-cut可用現有的最大流量演算法解出。H-cut的成本公式定義：

$$\text{cost}(X, Y) = \sum_{v \in H(X, Y)} \text{net}(v_{i,b}, v_{i,e}) + \sum_{v \in Y} \text{comp}(s, v_{i,b}) \quad (2)$$

圖形切割與佈署策略問題之轉換是產生整合圖 $H = (U, F)$ 。假設有一圖形 $G = (V, E)$ 代表ODG G 關係，如圖表 3(a)。階層式樹狀網路結構，如圖表 3(b)。整合圖 H 的演算法定義如下：

定義 2.2： 給定一個網路 s 有 $M+1$ 個樹狀結構主機 s_0, s_1, \dots, s_M ，ODG $G = (V, E)$ 。整合圖 $H = (U, F)$ 建構如下：

- (1) 對所有主機 $s_k (0 \leq k \leq M)$ ，將所有點和線加上標籤 (k) ，建立子圖 $G^{(k)} = (V^{(k)}, E^{(k)})$ ， $G^{(k)}$ 的建立方式如下：
 - For each object v_i in the ODG
 - Add two vertices $v_{i,e}^{(k)}, v_{i,b}^{(k)}$ to graph H .
 - Add two edge $(v_{i,e}^{(k)}, v_{i,b}^{(k)})$ to graph H .
 - For each edge (v_i, v_j) in the ODG
 - Add edge $(v_{i,e}^{(k)}, v_{j,b}^{(k)})$ to graph H .
- (2) 建立整合圖 H 的點集合 $U = (V^{(0)} \cup V^{(1)} \dots \cup V^{(M)} \cup \{T\})$ ，整合圖 H 結束點為 T 。
- (3) 建立以下三個 ∞ -edges：
 - $E_{DOWN} = \{(v_{i,b}^{(p)}, v_{i,b}^{(k)}) | v_i \in V, s_p \text{ 是 } s_k \text{ 的 parent 主機}\}$ 。
 - $E_{SOURCE} = \{(s, v_{i,b}^{(k)}) | v_i \in I \text{ 是 initial 節點}, 0 \leq k \leq M\}$ 。
 - $E_{TARGET} = \{(v_{i,e}^{(k)}, T) | v_i \in V, v_i \text{ 是 } G \text{ 的結束點}, 0 \leq k \leq M\}$ 。
- (4) edges 成本： $E_{COMP} = \{(s, v_{i,b}^{(k)}) | v_i \in V \setminus I, 0 \leq k \leq M\}$ 。
 - 對edges $(s, v_{i,b}^{(k)})$ ， $\text{cap}((s, v_{i,b}^{(k)})) = \text{comp}(v_i)$ 。
 - 對edges $(v_{i,b}^{(k)}, v_{i,e}^{(k)})$ ， $\text{cap}((v_{i,b}^{(k)}, v_{i,e}^{(k)})) = \text{net}(v_i)$ 。
- (5) 建立整合圖 H 的線集合： $F = (E^{(0)} \cup E^{(1)} \dots \cup E^{(M)}) \cup E_{DOWN} \cup E_{SOURCE} \cup E_{TARGET}$ 。
- (6) 假設 (X, Y) 在整合圖 H 是最小切割。
- (7) $\mathbf{R} = ((X \cap V^{(0)}, Y \cap V^{(0)}), (X \cap V^{(1)}, Y \cap V^{(1)}) \dots, (X \cap V^{(M)}, Y \cap V^{(M)}))$ ，回傳 \mathbf{R} 。 ■

由 (X, Y) 導出 $G^{(x)}$ 的一組切割項目 $(X \cap V^{(x)}, Y \cap V^{(x)})$ ，複本策略演算法回傳 $((X \cap V^{(0)}, Y \cap V^{(0)}), (X \cap V^{(1)}, Y \cap V^{(1)}) \dots, (X \cap V^{(M)}, Y \cap V^{(M)}))$ 。

(X, Y) 切割跟整合圖 H 之間的相對應關係與複本策略如下：

- 如果 $\text{edge}(v_{i,b}^{(x)}, v_{i,e}^{(x)})$ 在切割線右邊，物件 v_i 就是一個 $C\text{-node}$ 。 F_{COMP} 中 $\text{edge}(s, v_{i,b}^{(x)})$ 一定會被切割到，且成本 $\text{cap}((s, v_{i,b}^{(k)})) = \text{comp}(v_i)$ 。
- 如果 $\text{edge}(v_{i,b}^{(x)}, v_{i,e}^{(x)})$ 橫跨切割線，物件 v_i 就是一個 $N\text{-node}$ 。切割線必定會切割到 $\text{edge}(v_{i,b}^{(x)}, v_{i,e}^{(x)})$ ，其成本為 $\text{cap}((v_{i,b}^{(k)}, v_{i,e}^{(k)})) = \text{net}(v_i)$ 。
- 如果 $\text{edge}(v_{i,b}^{(x)}, v_{i,e}^{(x)})$ 在切割線左邊，物件 v_i 就是一個 $O\text{-node}$ 。沒有 edge 與 $v_i^{(x)}$ 有相關聯，也就是成本為 0。

令 $(X^{(x)}, Y^{(x)})$ 為 $H\text{-cut}(X, Y)$ 在子圖 $G^{(x)}$ 上的子切割。根據[8]，圖表 4 的佈署策略是合法的。 $H\text{-cut}(X, Y)$ 對應到以下佈署策略：

$$N_0 = \{v | v_{i,b}^{(0)} \in X^{(0)}\}, C_0 = \{v | v_{i,e}^{(0)} \in Y^{(0)}\} \quad (3)$$

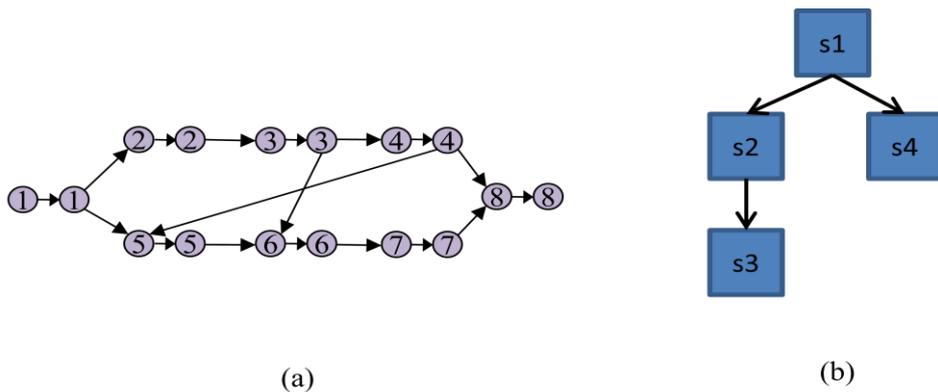
...

$$N_M = \{v | v_{i,b}^{(M)} \in X^{(M)}\}, C_M = \{v | v_{i,e}^{(M)} \in Y^{(M)}\}$$

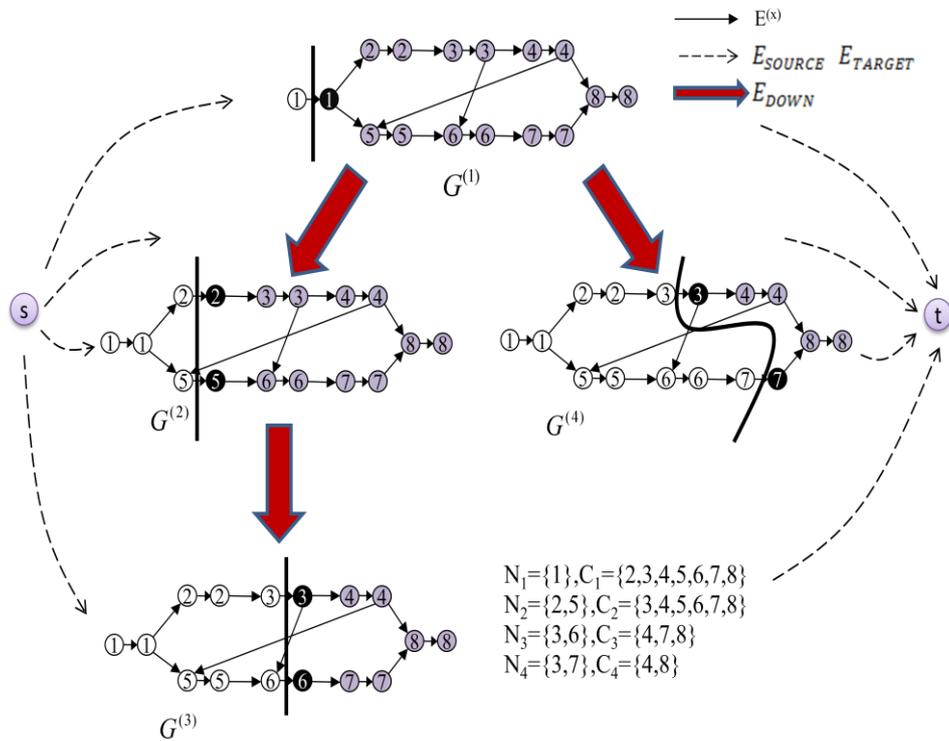
較佳解演算法步驟如下：

1. 將佈署策略問題轉換成整合圖 H 。
2. 找出整合圖 H 上最小切割 $C_{min} \in H\text{-cut}$ 。
3. C_{min} 對應到一組合法的佈署策略(定義 2.2)。
4. 將利用定義 2.2 佈署的結果精簡，消除多餘的節點成本。

本篇計畫探討是在階層式的架構下應用，而另外線性架構的網路已經有先前的學者研究[10]。首先讀入 $ODG G = (V, E)$ ， $V = (v_1, v_2 \dots, v_n)$ ，成本函數 net 與 comp 還有目標物件集合 $(T_0, T_1 \dots, T_M)$ 。接下來將成本與 T 依照順序佈署到主機的 ODG 物件上，接下來執行最大流量演算法找出最小切割線佈署 O, N, C 的結果，假設如圖表 4，依照[8]這是個合法的佈署圖。



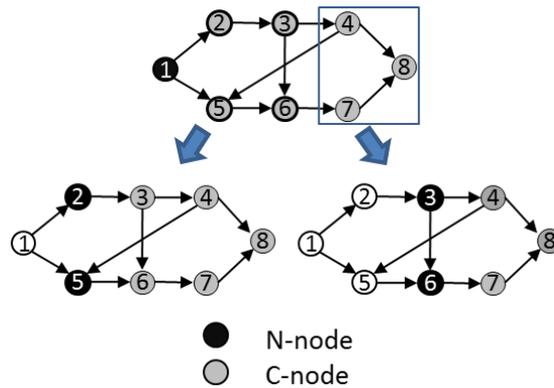
圖表 3：(a) $ODG G$ ，(b) $network S$ 。



圖表 4：整合圖 H 的切割線。

3.2 Direct 複本策略精簡方法

在滿足策略定義 2.1 且是個可行的佈署情況下移除多餘的物件，可進一步的精簡減少總成本。



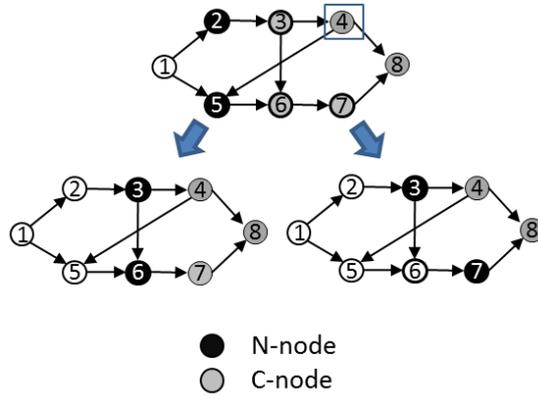
圖表 5 parent 主機 s_x , child 主機 $s_y, s_z, N_x = \{1\}, N_y = \{2, 5\}, N_z = \{3, 6\}, Dir(N_x, C_x) = \{1, 2, 3, 4, 5, 6\}$ 。
 如圖表 5, direct 相依物件的路徑, $N_x = \{1\}, N_y = \{2, 5\}, N_z = \{3, 6\}, Dir(N_x, C_x) = \{1, 2, 3, 4, 5, 6\}$ 。

定義 3.1：根據公式(3)佈署對應項目：

1. $Dir(N_x, C_x) = PRED(N_y) \cup N_y \cup PRED(N_z) \cup N_z$ 。
2. Set of nodes in $V \setminus Dir(N_x, C_x)$ can be removed from (N_x, C_x) 。■

3.3 InDirect 複本策略精簡方法

延伸 3.2 小節的 $Dir(N_x, C_x)$, Dir 以扣除一部分多於成本，這章節將利用更進一步方法 $InDir(N_x, C_x)$ 的物件相依關係，來更加的精簡成本大小。



圖表 6 parent 主機 s_x , child 主機 $s_y, s_z, N_x = \{2,5\}, N_y = \{3,6\}, N_z = \{3,7\}, Dir(N_x, C_x) = \{2,3,4,5,6,7\}, InDir = \{4\}$ 。

由 N_x 後繼節點 $C-node \setminus InDir$ 佈署 N_y, N_z 即可。圖表 6 中, $InDir$ 節點包含在 $Dir(N_x, C_x) = \{2,3,4,5,6,7\}$ 其中 $C_x = \{3,4,6,7\}$ 。 N_x 節點 5 並不需要前行節點 4 的計算結果, 因此 $InDir = \{4\}$ 。

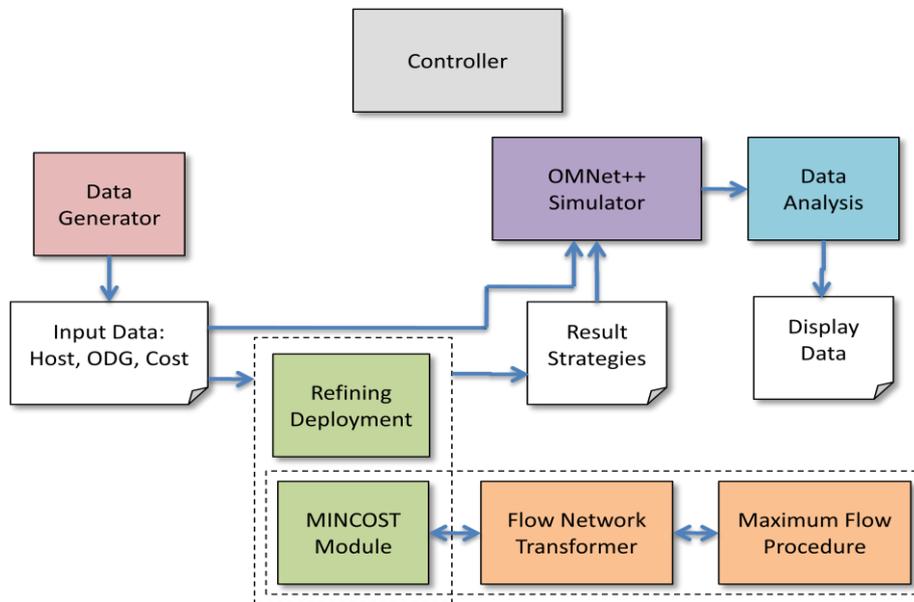
定義 3.2: 根據公式(3)佈署對應項目

1. $InDir(N_x, C_x) = \{v_i | v_i \in Dir(N_x, C_x) \text{ and } \forall (v_i, v_j) \in E, v_j \in PRED(N_y) \cup PRED(N_z) \Rightarrow v_i \in PRED(N_x) \setminus N_y \cup N_z\}$ 。
2. Nodes in set $InDir(N_x, C_x)$ can also removed from (N_x, C_x) 。 ■

CHAPTER 4 實驗結果

4.1 實驗系統架構

在執行最大流量程式之前必須讀入 `cost.txt` 檔, 依序讓每個物件附上一組 `net` 與 `comp` 數值。程式中 `odg1` 與 `odg2` 分別表示 network 拓撲與 `ODG` 圖, `odg1` 與 `odg2` 參數可供修改。

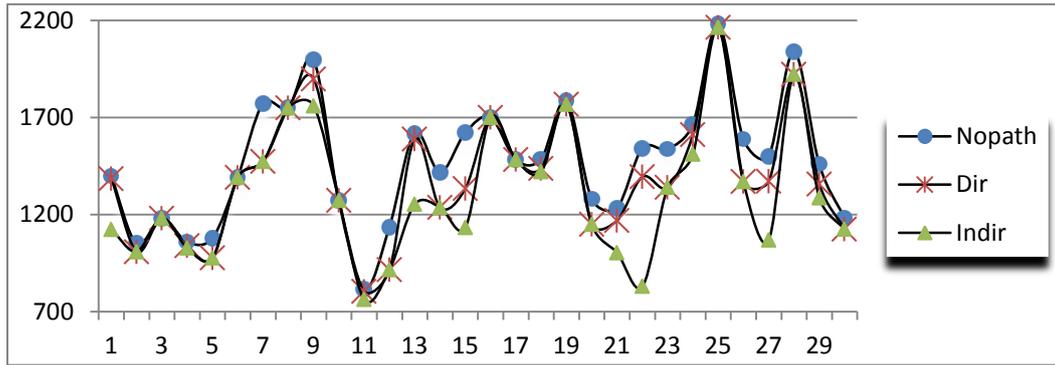


圖表 7: 實驗系統架構。

`odg1` 代表階層式主機 (s_0, s_1, \dots, s_p) 之間的相依關係圖, 而 `odg2` 代表每個主機裡 `ODG` 中物件與物件的執行先後順序。程式執行結果是以文字來記錄物件的佈署與流量, 為了方便我們驗證實驗結果與否, 使用 `Graphviz` (`Graph Visualization Software`) [13] 軟體執行文字轉圖的動作。待確認網路關係與流量正確後, 再將最大流量演算法產生的佈署策略結果 `OMNET-algorithm_ID.txt` 複製到 `OMNet++` 模擬器, 執行網路模擬的成本數據分析。

4.2 實驗討論

這個章節我們來討論實驗結果，我們提出在階層式網路下增強系統效能的方法。在 *OMNet++* 模擬的環境下執行所記錄的結果。



圖表 8：模擬效能的三種方法：*Nopath*、*Dir* 與 *InDir*。

圖表 8 是實驗的結果，Y 軸代表程式執行時間，然而 X 軸代表 *ODGs* 大小。Y 軸上的每個數據都是由 200 個執行時間取平均而得的。由這個實驗結果顯示，*InDirect* 路徑的精簡方法比另外兩種方法還要好，以 *Nopath* 方法為最差。

CHAPTER 5 結論與未來發展

本計畫提出了一種精簡的演算法，透過消除 *direct* 與 *InDirect* 的相依關係，在樹狀網路裡可以消除多餘佈署的元件。在實驗中，*InDirect* 如果符合條件的元件不多，又或者條件達成但元件的成本很低，那麼實質上能減少的總成本相對的也減少，所以在圖表 8 中有些 *InDirect* 減少的成本並不明顯。但實驗顯示，比較先前的演算法，我們新的演算法佈署策略比非精簡的方法更有效。實際將本報告所提出的方法實作到網路服務系統，將是未來工作。

參考文獻

1. B. Krishnamurthy F. Douglis, A. Feldmann and J. Mogul. Rate of change and other metrics: A live study of the world wide web. In Proceedings of the USENIX Symposium on Internet Technologies and Systems, December 1997.
2. A. Labrinidis and N. Roussopoulos. Webview materialization. In Proceedings of the 2000 ACM SIGMOD international conference on Management of data table of contents, May 2000.
3. X. Tang and S.T. Chanson. Minimal cost replication of dynamic web contents under at update delivery. IEEE Trans. Parallel and Distributed Systems, 15(5): 431-441, May 2004.
4. Z. Yu J. Ravi and W. Shi. A survey on dynamic web content generation and delivery techniques. Journal of Network and Computer Applications, 32(5):943-960, 2009.
5. Lung-Pin Chen, "Optimal Update Delivery Strategy for Hierarchical Content service Environment", Private communications, Dec 2009.
6. A. Goldberg and R. Tarjan. A new approach to the maximum flow problem. Journal of the ACM, 35(4):921-940, October 1988.

7. Lung-Pin Chen, Chung-Yuan Lai , “A Simplified Deployment Strategy Management Service for Workflow Components”, June, 2012
8. Lung-Pin Chen, Yu-Hsiang Huang,Ke-Chiang Chih, “Efficient Reconstruction of Dynamic Web Contents in Hierarchical Environment”, November 18-November 20.
9. W. C. J. H. L.P. Chen, I.C. Wu and M. Ho. Incremental digital content object delivering in distributed systems. IEICE Transactions on Information and Systems, E93-D(6), June 2010.
10. Lung-Pin Chen, Yu-Hsiang Huang,Ke-Chiang Chih, “Efficient Reconstruction of Dynamic Web Contents in Hierarchical Environment”, November 18-November 20.
11. Andras Varga. Using the omnet++ discrete event simulation system in education. IEEE Transactions on Education, 42(4), 1999.
12. <http://www.wretch.cc/blog/wulu2005/342870> July 2012.
13. <http://zh.wikipedia.org/wiki/Graphviz> June, 2010.

國科會補助專題研究計畫成果報告自評表

請就研究內容與原計畫相符程度、達成預期目標情況、研究成果之學術或應用價值（簡要敘述成果所代表之意義、價值、影響或進一步發展之可能性）、是否適合在學術期刊發表或申請專利、主要發現或其他有關價值等，作一綜合評估。

1. 請就研究內容與原計畫相符程度、達成預期目標情況作一綜合評估

- 達成目標
 未達成目標（請說明，以 100 字為限）
 實驗失敗
 因故實驗中斷
 其他原因

說明：

本報告說明第一年的執行成果，我們設計出新的方法並以統計模擬來做驗證。第一年的執行成果可以說已達成預期目標。原先計畫報告是要研究單一應用程式(包含多個服務元件)的部署策略，並在第二年將新的方法擴展到多個應用程式，也就是某些子部署策略能夠組成更大的策略，這部分問題需要更大 scale 的演算法設計與模擬驗證。由於本計畫為一年期，這部分問題我們列為未來工作。

2. 研究成果在學術期刊發表或申請專利等情形：

- 論文：已發表 未發表之文稿 撰寫中 無
專利：已獲得 申請中 無
技轉：已技轉 洽談中 無
其他：(以 100 字為限)

本計畫成果發表在國內會議，得到 Best Paper Award。相關的雲端服務與應用的成果發表在以下 SCI 期刊，正在 revise 階段。

Lung-Pin Chen, Chien-An Lin, Kuan-Ching Li, Ching-Hsien (Robert) and Hsu, Zhi-Xian Chen, A Scalable Blackbox-Oriented e-Learning System based on Desktop Grid over Private Cloud, Future Generation Computer Systems, in revision, July, 2012, (Impact factor: 1.978)

3. 請依學術成就、技術創新、社會影響等方面，評估研究成果之學術或應用價值（簡

要敘述成果所代表之意義、價值、影響或進一步發展之可能性) (以 500 字為限)

本計劃為技術發展類型之研究。由於手持式裝置與雲端服務為將來趨勢，發展相當快速，雲端服務的經濟成本的議題一直受到客戶與業者的注意。目前大多數雲端服務平台都是以計算量與網路流量來計算服務建置與運作費用。我們的研究是探討如何部署雲端服務元件，並嘗試降低計算量與網路流量這兩個成本，這個成果具有相當實用性。

國科會補助計畫衍生研發成果推廣資料表

日期:2012/12/07

國科會補助計畫	計畫名稱: 動態網路服務之整合性階層式部署之研究與發展
	計畫主持人: 陳隆彬
	計畫編號: 100-2221-E-029-020- 學門領域: 計算機網路與網際網路
無研發成果推廣資料	

100 年度專題研究計畫研究成果彙整表

計畫主持人：陳隆彬		計畫編號：100-2221-E-029-020-				計畫名稱：動態網路服務之整合性階層式部署之研究與發展	
成果項目		量化			單位	備註（質化說明：如數個計畫共同成果、成果列為該期刊之封面故事...等）	
		實際已達成數（被接受或已發表）	預期總達成數（含實際已達成數）	本計畫實際貢獻百分比			
國內	論文著作	期刊論文	0	0	100%	篇	
		研究報告/技術報告	0	0	100%		
		研討會論文	2	2	100%		
		專書	0	0	100%		
	專利	申請中件數	0	0	100%	件	
		已獲得件數	0	0	100%		
	技術移轉	件數	0	0	100%	件	
		權利金	0	0	100%	千元	
	參與計畫人力（本國籍）	碩士生	2	2	100%	人次	
		博士生	0	0	100%		
		博士後研究員	0	0	100%		
		專任助理	0	0	100%		
國外	論文著作	期刊論文	1	1	100%	篇	以投稿但正在 revise 階段
		研究報告/技術報告	0	0	100%		
		研討會論文	0	0	100%		
		專書	0	0	100%	章/本	
	專利	申請中件數	0	0	100%	件	
		已獲得件數	0	0	100%		
	技術移轉	件數	0	0	100%	件	
		權利金	0	0	100%	千元	
	參與計畫人力（外國籍）	碩士生	0	0	100%	人次	
		博士生	0	0	100%		
		博士後研究員	0	0	100%		
		專任助理	0	0	100%		

<p>其他成果 (無法以量化表達之成果如辦理學術活動、獲得獎項、重要國際合作、研究成果國際影響力及其他協助產業技術發展之具體效益事項等，請以文字敘述填列。)</p>	<p>無</p>
--	----------

	成果項目	量化	名稱或內容性質簡述
科 教 處 計 畫 加 填 項 目	測驗工具(含質性與量性)	0	
	課程/模組	0	
	電腦及網路系統或工具	0	
	教材	0	
	舉辦之活動/競賽	0	
	研討會/工作坊	0	
	電子報、網站	0	
	計畫成果推廣之參與(閱聽)人數	0	

國科會補助專題研究計畫成果報告自評表

請就研究內容與原計畫相符程度、達成預期目標情況、研究成果之學術或應用價值（簡要敘述成果所代表之意義、價值、影響或進一步發展之可能性）、是否適合在學術期刊發表或申請專利、主要發現或其他有關價值等，作一綜合評估。

1. 請就研究內容與原計畫相符程度、達成預期目標情況作一綜合評估

達成目標

未達成目標（請說明，以 100 字為限）

實驗失敗

因故實驗中斷

其他原因

說明：

2. 研究成果在學術期刊發表或申請專利等情形：

論文： 已發表 未發表之文稿 撰寫中 無

專利： 已獲得 申請中 無

技轉： 已技轉 洽談中 無

其他：（以 100 字為限）

本計畫成果發表在國內會議，得到 Best Paper Award。相關的雲端服務與應用的成果發表在以下 SCI 期刊，正在 revise 階段。

Lung-Pin Chen, Chien-An Lin, Kuan-Ching Li, Ching-Hsien (Robert) and Hsu, Zhi-Xian Chen, A Scalable Blackbox-Oriented e-Learning System based on Desktop Grid over Private Cloud, Future Generation Computer Systems, in revision, July, 2012, (Impact factor: 1.978)

3. 請依學術成就、技術創新、社會影響等方面，評估研究成果之學術或應用價值（簡要敘述成果所代表之意義、價值、影響或進一步發展之可能性）（以 500 字為限）

本計劃為技術發展類型之研究。由於手持式裝置與雲端服務為將來趨勢，發展相當快速，雲端服務的經濟成本的議題一直受到客戶與業者的注意。目前大多數雲端服務平台都是以計算量與網路流量來計算服務建置與運作費用。我們的研究是探討如何部署雲端服務元件，並嘗試降低計算量與網路流量這兩個成本，這個成果具有相當實用性。