



圖 52 製造準備表單畫面(二)

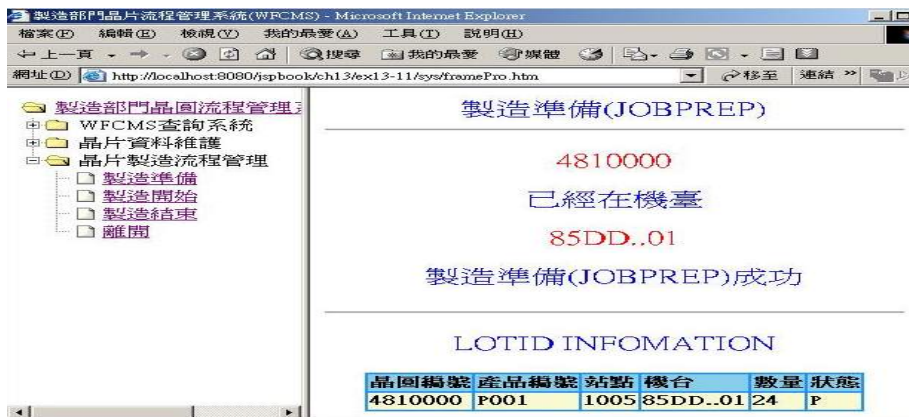


圖 53 製造準備結果

5.3.5 製造開始(JOB IN)

製造開始程式運作流程，如圖 54。JobIn.jsp 連結 LOT 資料庫，讀取晶圓狀態為 P 的晶圓清單，提供給使用者選擇畫面如圖 55。JobInByEqp.jsp 連結 LOT 資料庫讀取晶圓編號的機台編號，畫面如圖 56。JobIned.jsp 至 LOT 資料庫與 EQP 資料庫，更新相關資料成功，會顯示如圖 57 的成功訊息。

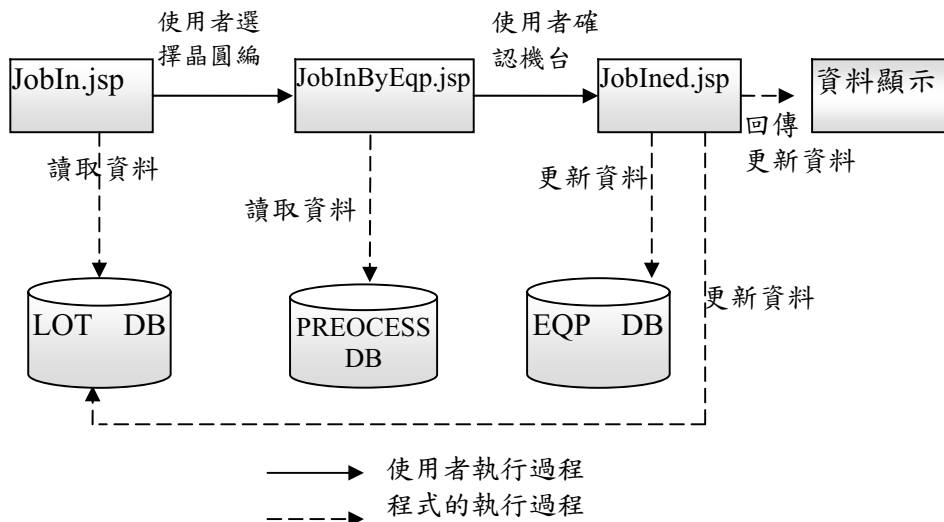


圖 54 製造開始運作流程圖



圖 55 製造開始表單畫面(一)



圖 56 製造開始表單畫面(二)



圖 57 製造開始結果

5.3.6 製造結束(JOB OUT)

製造結束程式運作流程，如圖 58；JobOut.jsp 連結 LOT 資料庫，讀取晶圓狀態為 R 的晶圓清單，提供給使用者選擇畫面如圖 59。

JobOutByEqp.jsp 連結 LOT 資料庫讀取晶圓編號的機台編號，畫面如圖 60。JobOuted.jsp 連結 PROCESS 資料庫，讀取下一製程站點編號，更新至 LOT 資料庫，再連結 EQP 資料庫，更新相關資料，顯示如圖 61 的成功訊息。

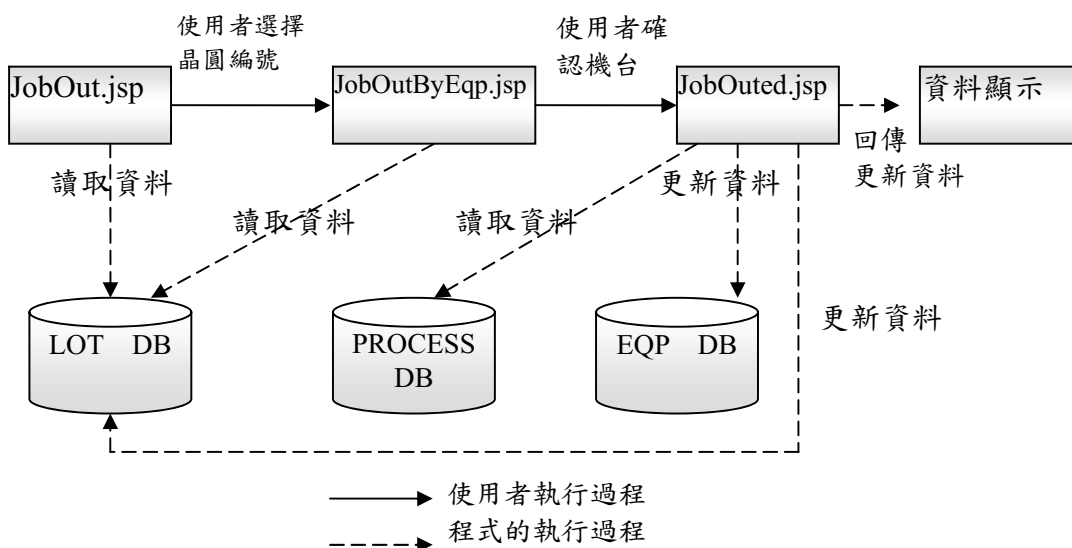


圖 58 製造結束運作流程圖



圖 59 製造結束表單畫面(一)

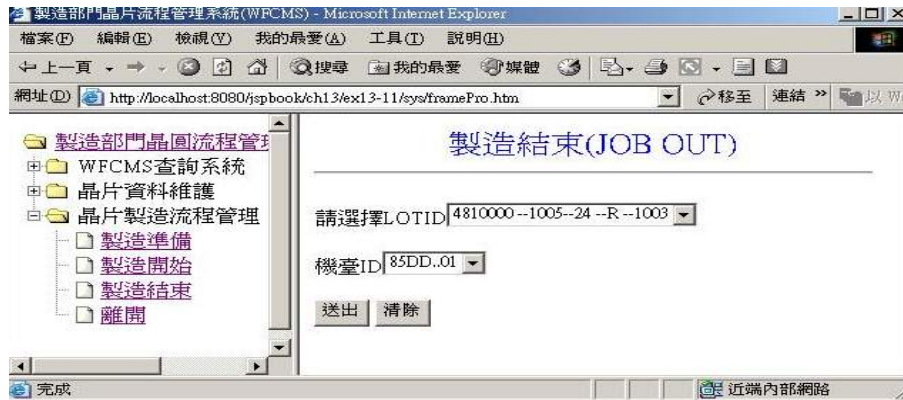


圖 60 製造結束表單畫面(二)



圖 61 製造結束結果

第六章 結論

企業因應資訊化的需求，各部門考量技術發展及策略聯盟的合作發展計畫等因素，產生各異質性資料庫系統。各部門為配合晶圓製造需求，開發許多資料庫維護介面。在每個晶圓製造流程，以人工方式透過各部門提供資料庫維護介面，存取與更新各部門異質資料庫的資料，非常沒有效率。

6.1 節總結設計一套整合各異質資料庫的晶圓流程管理系統(WFCMS)。提供一個整合的介面，改善製造流程，提升生產效率。6.2 節介紹系統效能評估分二部分。第一部份請各部門根據測試事件，測試晶圓流程管理系統(WFCMS)資料更新及讀取的正确性。第二部份以未整合的應用程式介面與透過晶圓流程管理系統(WFCMS)，更改相同資料的時間做比較，評估工作效率。6.3 節描述未來發展方向，晶圓流程管理系統(WFCMS)能提供更自動化的功能，期望達到無人工廠的理想。

6.1 總結

晶圓的製造流程非常繁雜，需要各部門的配合，尤其在資料共享上，希望能更有效率的整合。因此，設計一套整合各異質資料庫的晶圓流程管理系統(WFCMS)，提供一個整合的介面，改善製造流程，提升生產效率。此系統在應用程式開發上，使用 Java 的程式語言，運用網際網路技術。客戶端透過瀏覽器連結應用程式伺服器。在異質資料庫整合，使用 JSP 與 JDBC 與各異值資料庫連結，並採用 Connection pool 與 JNDI 的機制，避免程式與資料庫連結溝通頻繁，造成效能問題。

目前晶圓流程管理系統(WFCMS)，已在生產線上測試完成，各部門

的反應非常良好，同時提出功能改善建議與新功能的需求。

6.2 系統效能評估

晶圓流程管理系統(WFCMS)，主要整合各部門資料庫，各部門透過此系統，在更新或讀取資料時，除了要求速度及便利性外，更要求資料更新及讀取資料的正確性。因此在系統效能評估上，分兩部分。第一部份根據各部門所要求更新及讀取資料的邏輯，撰寫測試事件，並請各部門根據此測試事件，測試晶圓流程管理系統(WFCMS)，評估更新及讀取資料的正確性。第二部份以現行各部門使用未整合的應用程式介面，更改資料的時間，與透過晶圓流程管理系統(WFCMS)，更改相同資料的時間做比較，評估工作效率。



6.2.1 測試程式邏輯資料正確性

表列(表 11、表 12)的測試事件表，交由各部門的工程師，進行一個月的密集測試，並將測試結果記錄下來。由回收的各部門測試事件表統計，資料正確性的通過率達到 99%，可以說是完全滿足使用者的需求。至於 1%，是由於使用者在資料建置上，輸入不合法的字元(如空白及空字元)導致程式無法判讀而產生錯誤。針對不合法的字元問題，應用程式在輸入介面上，做資料欄位檢查的修改，修改完成後，資料邏輯的正確性已達到 100%。

表 11 登入機制測試事件

| 編號 | 使用功能 | 事件描述 | 是否通過 |
|----|------|----------------------|------|
| 1 | 登入系統 | 出現使用者帳號、密碼的文字方塊及登入按鈕 | |

| | | | |
|---|-----------|---|--|
| 2 | 輸入帳號 | 使用者帳號的文字方塊可以輸入文字與數字 | |
| 3 | 輸入密碼 | 使用者密碼的文字方塊可以輸入文字與數字 | |
| 4 | 按下登入 鈕 | 使用者帳號及密碼的文字方塊沒有輸入任何字元時，會出現提示的視窗 | |
| | | 使用者帳號及密碼的文字方塊輸入的帳號及密碼錯誤，導到提示錯誤訊息網頁 | |
| | | 輸入製造部門的使用者帳號及密碼，網頁正確導到有 WFCMS 查詢系統、晶圓資料維護、晶圓製造流程管理的網頁 | |
| | | 輸入製程部門的使用者帳號及密碼，網頁正確導到有 WFCMS 查詢系統、站點資料維護的網頁 | |
| | | 輸入設備部門的使用者帳號及密碼，網頁正確導到有 WFCMS 查詢系統、機台資料維護的網頁 | |
| 5 | 請重新登 入 | 提示錯誤訊息網頁，有請重新登入的超連結 | |
| | | 按下請重新登入的超連結，網頁會導到登入畫面 | |

共同功能測試事件

表 12 站點查詢測試事件

| 編號 | 使用功能 | 事件描述 | 是否通過 |
|----|---------|--|------|
| 1 | 站點查詢 | 出現站點名稱、站點種類的文字方塊及送出資料、重新設定按鈕 | |
| 2 | 輸入站點 | 站點名稱的文字方塊可以輸入文字與數字 | |
| 3 | 輸入站點種類 | 站點種類的文字方塊可以輸入文字與數字 | |
| 4 | 按下送出資料鈕 | 站點名稱及站點種類的文字方塊沒有輸入任何字元時，會出現提示的視窗 | |
| | | 站點名稱及站點種類的文字方塊輸入站點及種類時，會顯示正確的站點、站點種類、描述、參數欄位資料 | |
| 5 | 按下重新設定鈕 | 站點名稱及站點種類的文字方塊的資料會清空 | |

6.2.2 測試使用系統與介面切換的時間比較

現行製造部門在製造準備階段，首先切換至製造部門所提供的應用

程式介面，平均需要大約 0.5 秒。第二步驟輸入要生產的晶圓批號，確認查詢所得的資料，如製程類型、批號的狀態等，平均需要 2.3 秒。第三步驟，切換至設備部門所提供的應用程式介面，平均需要大約 0.5 秒。第四步驟輸入執行生產的機台編號，確認查詢的資料，如機台狀態、生產批號數量等，平均需要大約 2.3 秒。第五步驟，切換至製造部門提供的應用程式介面，平均需要大約 0.5 秒。第六步驟，輸入晶圓編號，更改為準備好(Prepare)的狀態，平均需要大約 1.5 秒。以上六個步驟，總計要花費 28.3 秒。

製造部門透過晶圓流程管理系統(WFCMS)，使用製造準備階段的應用程式介面，首先選擇生產晶圓編號(應用程式提供狀態為 Q 的晶圓編號清單)，平均需要大約 1 秒。第二步驟選擇生產機台(應用程式已整合各部門資料，提供可以使用機台編號清單)，平均需要大約 1 秒。第三步驟按下確定鈕，應用程式至相關資料庫更新資料的時間，平均需要大約 0.5 秒。第四步驟使用者檢視回傳的訊息是否正確，平均需要大約 1 秒，以上四個步驟，總計要花費 3.5 秒。

一個製造準備的階段，相差大約 24.8 秒。製造部門一天一位工程師大約要執行 100 次製造準備，如圖 61。使用各部門的應用程式大約需要 2830 秒，使用 WFCMS 大約只花 350 秒，相差約節省 $2830-350=2480$ 秒(41.3 分鐘)，全工廠大約有 120 位製造工程師，總共節省 $41.3 \times 120 = 4956$ 分鐘(82.6 小時)。

由以上測試反應時間比較結果，晶圓流程管理系統(WFCMS)確實替公司節省很多人力操作的時間。在更新各部門的資料庫方面，使用程式自動更新，確實避免許多人工輸入造成的錯誤。

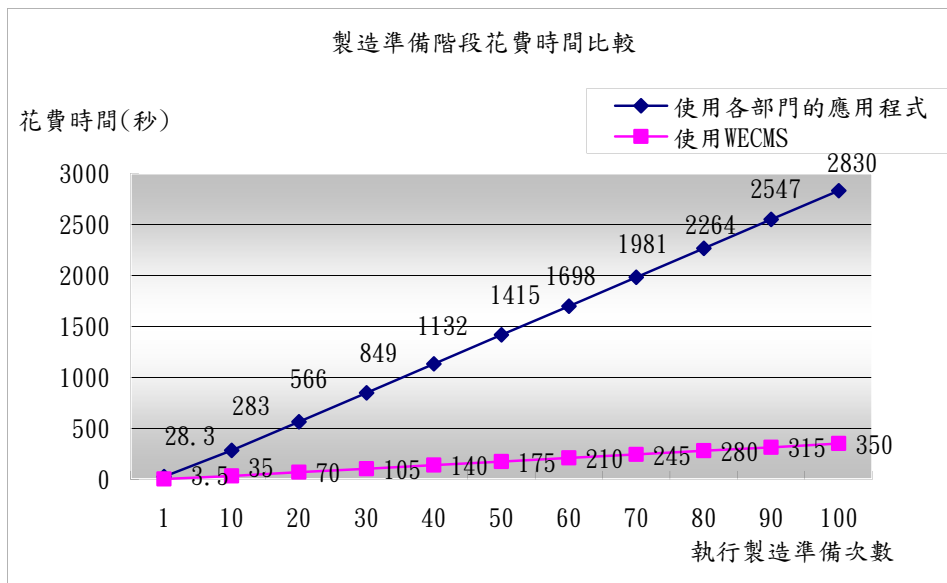


圖 62 製造準備階段花費時間比較圖

6.3 未來發展方向

目前晶圓流程管理系統(WFCMS)，針對製造部門、設備部門、製程部門的資料庫做整合，未來與製造流程相關的部門，如整合部門也希望能將部門的資料庫做整合，但相對需要部門資料庫的廠商，能提供網際網路架構的版本，或是提供 ODBC 與 JDBC 的資料庫驅動程式。

另外，設備部門進一步提出希望透過晶圓流程管理系統(WFCMS)與各設備的機台做資料交換的需求。直接傳遞機台參數給機台使用，讓機台自動生產，不需要製造工程師至機台的面板，選取製程參數，才能讓機台開始生產。如此，晶圓流程管理系統(WFCMS)能提供更自動化的功能，期望達到無人工廠的理想。

參考文獻

- [1] S. Vinoski, "CORBA: Integrating diverse applications within distributed heterogeneous environments", IEEE Communications, Vol.14, No.2, pp.7-9, Feb.1997.
- [2] 曾守正, "資料庫系統的回顧與未來研究發展", 中華民國資訊學會會刊(The Institute of Information and Computing Machinery, IICM), 第一卷, 第一期, 第 2-6 頁, 1996.
- [3] A Renner, "XML Data and Object Databases: The Perfect Couple", Proceedings of the 17th International Conference on Data Engineering, pp.143-148, April 02 - 06, 2001.
- [4] S.Vinoski, "CORBA: Integrating diverse applications within distributed heterogeneous environments", IEEE Communications, Vol. 14, No.2, pp.4-12, Feb. 1997.
- [5] P Kähkipuro, " Performance Modeling Framework for CORBA Based Distributed Systems", DEPARTMENT OF COMPUTER SCIENCE SERIES OF PUBLICATIONS A REPORT A-2000-3, University of Helsinki, pp.13-25, August 2000.
- [6] A.Hoffer, B.Prescott, R.McFadden, "Modern Database Management 6th Edition", Prentice Hall , 2002.
- [7] "Extensible Markup Language (XML) 1.0 (Fourth Edition) W3C Recommendation", August 2006, <http://www.w3.org/TR/REC-xml/>.
- [8] J.Chen and J. Xu, "XML and DB2", Proceedings of 16th IEEE International Conference on Data Engineering, 2000.
- [9] J.Pardi, "XML in Action", Microsoft Pree, Available onling at:<http://msdn.microsoft.com/xml/default.asp>.
- [10] "The Java tutorial,Third Edition", <http://java.sun.com/docs/books/tutorial/>.
- [11] "Java 2 Platform, Standard Edition, White Papers",

<http://java.sun.com/docs/white/index.html>.

[12] “JavaServer Pages Technology”, <http://java.sun.com/products/jsp/>.

[13] 郭尚君、李盈君, ” JSP 動態網頁入門實務”, 文魁資訊, December, 2003.

[14] “Java Naming and Directory Interface (JNDI)”,

<http://java.sun.com/products/jndi/>.

[15] “JNDI Tutorial”,

<http://java.sun.com/products/jndi/tutorial/information/download.html>.

[16] “Microsoft ODBC”, <http://www.microsoft.com/data/odbc/>.

[17] “The SQL standard”, http://www.jcc.com/SQLPages/jccs_sql.htm.

[18] “JDBC API”, <http://java.sun.com/products/jdk/1.2/docs/guide/jdbc/>.

[19] “JDBC Drivers”, <http://java.sun.com/products/jdbc/jdbc.drivers.html>.

[20] Pradhan, S. S. and Huang, W. V., “Virtual Manufacturing Information

System Using Java and JDBC”, Computers Industrial Engineering, Vol.35,

No.1-2, pp.255-258, 1998.

[21] “Connection pool reference”,

<http://java.sun.com/developer/onlineTraining/Programming/JDCBook/connpool.html>.

ool.html.

