

# The use of the distributed object for the construction of control system: Study on the case of bioreactor

詹孟訓、張德明

E-mail: 9300038@mail.dyu.edu.tw

## ABSTRACT

As a result of variety of characteristic of organism products, bioprocess mostly produce small quantity of batch products, and it is vary different from continuous process. Therefore, the study use the object-oriented design to allow the program model design more flexibly in the bioprocess, and it is to match up multiple demands of market products. On analyses and designs of the monitor system, the study use united model language (UML) to express the relationship and condition between objects. The study use the distributed component object model (DCOM) of Microsoft to develop component. In the DCOM structure, users can make the application communicate with the component through the RPC (remote process control). For this reason, it can reach the aim of remote control. The study use embedded system to develop the station of real-time application, and use the thread priority to raise efficiency of application. The study compare with the system of default process scheduler, and the efficiency shows that the way of real-time design can make performance raise 49.25 times in the embedded system. And the study compare with the station of pc application, and the performance raise 18%. Therefore, by using the refined control component of design bio-control system can raise up the real-time efficiency of application.

Keywords : DCOM ; UML ; Embedded system

## Table of Contents

封面內頁 簽名頁 授權書.....	iii	中文摘要.....	iv	英文摘要.....	v
誌謝.....	vi	目錄.....	vii	圖目錄.....	viii
第一章 前言.....	1	第二章 文獻回顧.....	3	2.1. 監控系統的種類.....	3
2.2. 軟體工程的趨勢.....	4	2.3. 分散式物件架構.....	5	2.4. 分散式物件架構的即時功能.....	8
2.5. 物件導向設計之方法.....	12	第三章 技術背景.....	14	3.1. 開發平台.....	14
3.2. 同步技術.....	15	3.2.1. 執行緒的組成和處理函數.....	15	3.2.2. 關鍵區域.....	18
3.2.3. 互斥.....	18	3.2.4. 事件.....	18	3.3. 分析工具-UML.....	19
3.3.1. 類別圖.....	21	3.3.2. 元件圖.....	21	3.3.3. 部署圖.....	23
3.3.4. 循序圖.....	23	3.3.5. 合作圖.....	25	3.3.6. 狀態圖.....	25
3.3.7. 活動圖.....	27	3.4.1. COM元件.....	27	3.4.2. COM元件的呼叫.....	29
3.4.3. COM元件架構模型.....	31	第四章 系統規劃與設計.....	36	4.1. 環境描述.....	36
4.2. 元件的設計和關聯.....	40	4.2.1. RS232元件.....	40	4.2.2. ADAM元件.....	43
4.2.3. PID元件.....	54	第五章 結果與討論.....	62	5.1. 測試平台.....	62
5.2. 效能測試.....	66	5.2.1. RS232元件的效能改進.....	62	5.2.2. COM元件模型架構的效能.....	69
5.2.3. 即時效率.....	69	5.3. PID的控制.....	73	5.4. 應用實例.....	76
5.4.1. 系統架構.....	76	5.4.2. PC操作界面說明.....	81	5.4.3. PDA操作界面說明.....	89
第六章 總結.....	94	參考文獻.....	96	圖形表.....	100
圖目錄 圖 2-1 分散式物件架構的分類.....	7	圖 2-2 不可奪取的排程.....	10	圖 2-3 可奪取的排程.....	11
圖 2-4 deviceCOM的機制.....	13	圖 3-1 執行緒核心物件.....	16	圖 3-2 執行緒處理函.....	16

數.....	17	圖 3-3 結構性觀點分類.....	20	圖 3-4 觀點
圖.....	22	圖 3-5 行為觀點圖之一.....	24	圖 3-6 行為觀
點圖之二.....	26	圖 3-7 COM元件的設計.....	28	圖 3-8 IDispatch
介面.....	30	圖 3-9 單一執行緒模型.....	33	圖 3-10 公寓執行緒
模型.....	34	圖 3-11 自由執行緒模型.....	35	圖 4-1 分散式架
構.....	37	圖 4-2 發酵控制系統節點圖.....	38	圖 4-3 典型的發酵
系統.....	39	圖 4-4 類別關聯圖.....	41	圖 4-5 RS232類別
圖.....	42	圖 4-6 RS232控制循序圖.....	44	圖 4-7 RS232狀態
圖.....	45	圖 4-8 RS232活動圖.....	46	圖 4-9 ADAM裝
置.....	47	圖 4-10 單一點控制.....	49	圖 4-11 ADAM類
別圖.....	50	圖 4-12 ADAM控制循序圖.....	51	圖 4-13 ADAM狀態
圖.....	53	圖 4-14 PID類別圖.....	56	圖 4-15 PID控制循序
圖.....	57	圖 4-16 轉速PID控制活動圖.....	60	圖 4-17 溫度PID控制活
動圖.....	61	圖 5-1 迴圈活動圖.....	64	圖 5-2 迴圈和訊號事件的
時間比.....	65	圖 5-3 測試元件.....	67	圖 5-4 不同模型之比
較.....	68	圖 5-5 控制設計圖.....	70	圖 5-6 priority設定比較
圖.....	71	圖 5-7 效能比較.....	72	圖 5-8 PID控制.....
.....	74	圖 5-9 轉速開環路測試.....	75	圖 5-10 轉速控制
圖.....	77	圖 5-11 溫度開環路測試.....	78	圖 5-12 溫度控制
圖.....	79	圖 5-13 發酵槽的接線圖.....	80	圖 5-14 發酵控制
系統操作說明.....	82	圖 5-15 PC初使畫面說明.....	83	圖 5-16 監控畫
面.....	84	圖 5-17 確定周邊裝置的安置.....	85	圖 5-18 循環馬
達啟動畫面.....	86	圖 5-19 狀態設定.....	88	圖 5-20 PID設
定警示.....	90	圖 5-21 PID設定畫面.....	91	圖 5-22 PDA監
控畫面.....	92	圖 5-23 PDA狀態設定.....		

## REFERENCES

- 參考文獻 1. 方世榮編譯 (1991), 自動化生產系統及電腦整合製造, 曉園, 台北。 2. 王寧疆、黃裕彰編譯 (2002), Microsoft Windows CE API 程式開發指南, 華彩軟體, 台北, 513 - 570頁。 3. 吳宣諭、龍仁光 (2001), PDA無線遙控車, 機電整合, 4:81 - 88頁。 4. 周斯畏、張春霞、詹雅璟、簡杏玟 (2002), 物件導向系統分析與設計-使用UML與C++, 全華, 台北, 2 - 17頁。 5. 范逸之(2001), Visual Basic 與分散式監控系統:使用RS-232/485串列通訊, 文魁資訊, 台北。 6. 范逸之、陳立元(2001), Visual Basic與RS232串列通訊控制最新版, 文魁資訊, 台北, 1 - 22頁。 7. 高煥堂 (1997), Visual C++ 精通物件觀念與技術, 大隆, 台北, 1 - 51頁。 8. 許格超編譯 (1999), 循序漸進COM/DCOM, 碁峰資訊, 台北, 159 - 198頁。 9. 陳峰棋編譯 (2002), Win CE IA 平台超強建構:Win CE裝置驅動程式完全開發, 台灣培生教育出版股份有限公司, 台北。 10. 張裕益譯 (2001), UML使用手冊, 博碩文化, 台北。 11. 陶博文編譯 (1999), 精通DCOM程式技巧, 碁峰資訊, 台北, 2 - 68頁。 12. 葛湘達編譯 (1999), ATL COM 程式設計, 碁峰資訊, 台北50 - 165頁。 13. 滕至陽 (2002), 作業系統理論與實作, 博碩文化股份有限公司, 台北, 2 - 10頁。 14. 龍仁光、周宗毅、吳宣諭 (2001), PDA結合TCP/IP轉換器於工業控制之應用, 機電整合, 42:120 - 125頁。 15. 韓丹編譯 (2001), Windows CE開發寶典, 儒林圖書出版, 台北, 1 - 35頁。 16. 龔義評編譯 (2001), Microsoft Windows 應用程式設計開發指南, 華彩軟體股份有限公司, 台北, 321 - 522頁。 17. Adamopoulos, D. X., Pavlou, G., and Papandreou, C. A. (2002), Continuous media support in the distributed component object model, Computer Communications, 25:169-182. 18. Alho, K, Lassenius, C., and Sulonen, R. (1996), Process enactment support in a distributed environment, Computers in Industry, 29:5-13. 19. Boling, D. (1999), Updated with New Kernel Features, Windows CE 3.0 Really Packs a Punch, Microsoft Systems Journal. 4:23-38. 20. Chen, D., Mok, A. K., and Nixon, M. (1999), Providing real-time support through component object model, Microprocessors and Microsystems, 23:145-154. 21. Cortellessa, V., and Mirandola, R. (2002), PRIMA-UML: a performance validation incremental methodology on early UML diagrams, Science of Computer Programming, 44: 101-129. 22. Fernando, M. S., and Filipe, M. P. (2001), Application of Distributed Platforms in a Video Surveillance System, Real-Time Imaging, 7:447-455. 23. Grattan, N., and Brain, M. (2001), Windows CE3.0 Application Programming, Prentice-Hall(PH), London. 24. Grimble, M. J. (1988), Real-time Computer Control, Prentice Hall, UK, p.99-127. 25. Henderson-Sellers, B., and Firesmith, D. G. (1999), Comparing OPEN and UML: the two third-generation OO development approaches, Information and Software Technology, 41:139-156. 26. Intrinsic (1999), deviceCOM: A functional extension of COM/DCOM for specialized distributed embedded Windows systems, Intrinsic, <http://www.intrinsic.com>. 27. Mari?, I. (2001), Flow measurement system model with distributed resources, Flow Measurement and Instrumentation, 12:283-289. 28. Michaelis, M. (2000), COM+ Programming from the Ground Up, Berkeley, CA, U.S.A., p.40-77. 29. Microsoft (1996), DCOM Technical Overview, Microsoft, <http://msdn.microsoft.com/>. 30. Morari, M., and Zafiriou, E.(1989), Robust Process Control, Prentice-Hall(PH), London, p.116. 31. Muench, C. (2000), The Windows CE Technology Tutorial, Addison Wesley,

Arlington. 32. OMG (2001), Common Object Request Broker Architecture, Object Management Group, <http://www.omg.org/>. 33. Omstead, D. R. (1990), Computer Control of Fermentation Process, CRC Press, FL, p.165-190. 34. Potter, W. D., Liu, S., Deng, X., and Rauscher, H. M. (2000), Using DCOM to support interoperability in forest ecosystem management decision support systems, *Computers and Electronics in Agriculture*, 27: 335-354. 35. Rezayat, M. (2000), The Enterprise-Web portal for life-cycle support, *Computer-Aided Design*, 32:85-96. 36. Sang, J., Follen, G., Kim, C. and Lopez, I. (2002), Development of CORBA-based engineering applications from legacy Fortran programs, *Information and Software Technology*, 44:175-184. 37. Sheremetov, L. B., Smirnov, A. V. (1999), Component integration framework for manufacturing systems re-engineering: agent and object approach, *Robotics and Autonomous Systems*, 27:77-89. 38. Sun Microsystems (2002), System Architecture, JAVA, <http://java.sun.com/>. 39. Thomas G., Fisher, P.E. (1990), Batch control systems: Design, Application, and Implementation, ISA, North Carolina. 40. Vittorio, C., Raffaella, M. (2002), PRIMA-UML: a performance validation incremental methodology on early UML diagrams, *Science of Computer Programming*, 44:101-129. 41. Walter, T., Grabowski, J. (1999), A framework for the specification of test cases for real-time distributed systems. *Information and Software Technology*, 41:781-798. 42. Wagner, D. J., Brewer, F. E., and Aiken, A. (2000), A first step towards automated detection of buffer over-run vulnerabilities. In *Proceedings of the Year 2000 Network and Distributed System Security Symposium(NDSS)*. San Diego, CA. 43. Yao, P. (2000), Windows CE 3.0: Enhanced Real-Time Features Provide Sophisticated Thread Handling, *Microsoft Systems Journal*. 15:132-144.