

บทที่ 5

การพัฒนาโปรแกรม

ระบบสารสนเทศเพื่อควบคุมการผลิตของบริษัทลำพูนชิงเคนแก่น จำกัด เป็นระบบที่มีการทำงานใน 2 ลักษณะ คือการทำงานแบบ Client-Server ซึ่งใช้โปรแกรม Sybase Power Builder 8.0 เป็นเครื่องมือในการพัฒนาโปรแกรม และ Web Application ซึ่งใช้โปรแกรม Macromedia Dreamweaver8 เป็นเครื่องมือในการพัฒนา และได้มีการใช้โปรแกรม Chart Director เป็นเครื่องมือในการทำกราฟแสดงผลบน We Application โดยในส่วนของการพัฒนาโปรแกรมที่มีส่วนสำคัญคือการแสดงรายงานในรูปแบบกราฟฟิค

5.1 การพัฒนาโปรแกรมโดยโปรแกรม Sybase PowerBuilder 8.0

โปรแกรม PowerBuilder เป็นเครื่องมือในการพัฒนาโปรแกรมซึ่งพัฒนาโดยบริษัท Sybase โดยในการพัฒนาโปรแกรมด้วยเครื่องมือนี้ จะประกอบไปด้วย Object ที่สำคัญและเกี่ยวข้องกับระบบสารสนเทศเพื่อควบคุมการผลิตของบริษัทลำพูนชิงเคนแก่น จำกัดที่สำคัญดังนี้

ตาราง 5.1 ตารางแสดง Object ที่สำคัญของโปรแกรม PowerBuilder

ออบเจกต์	คำอธิบาย
workspace	เป็น Object ที่ใช้ในการกำหนดพื้นที่ที่ใช้ในการทำการพัฒนาระบบ
target	เป็นพื้นที่ที่ใช้ในการสร้าง Application หรือระบบ โดยใน 1 workspace สามารถเก็บ application ได้มากกว่า 1 application
library	เป็นที่เก็บ PowerBuilder Object ต่างๆของโปรแกรม PowerBuilder ไม่ว่าจะเป็น window, datawindow, menu, function เป็นต้น
application	เป็นหัวใจหลักหรือแกนกลางในการพัฒนาโปรแกรมโดยโปรแกรม PowerBuilder ซึ่งใน application จะเป็น Object ที่ใช้ในการควบคุมการเริ่มต้นการทำงานของโปรแกรมที่ต้องการพัฒนา
window	เป็น Object ที่ใช้ในการติดต่อกับผู้ใช้งาน
datawindow	เป็น Object ที่ใช้ในการเชื่อมโยงหรือติดต่อกับฐานข้อมูล
menu	เป็น Object ที่ใช้ในการสร้างเมนูบนจอภาพ

ตาราง 5.1 ตารางแสดง Object ที่สำคัญของโปรแกรม PowerBuilder (ต่อ)

ออบเจกต์	คำอธิบาย
structure	เป็นโครงสร้างของข้อมูลที่ใช้ในการประกาศค่าตัวแปรต่างๆที่ใช้ในระบบ
function	เป็น Object ที่ใช้ในการสร้างฟังก์ชันต่างๆ เพื่อให้หน้าจอหรือ window ต่างๆเรียกใช้งานได้

การพัฒนาโดยโปรแกรม PowerBuilder8.0 จะเริ่มจากการสร้าง workspace หรือพื้นที่ทำงานก่อน จากนั้นจึงทำการสร้าง target โดยขณะที่เราสร้าง target นั้น โปรแกรมจะทำการสร้าง application ให้โดยอัตโนมัติ จากนั้นจึงทำการสร้าง library เพื่อรองรับการสร้าง Power Builder Object ไม่ว่าจะเป็น window datawindow menu และอื่นต่อไป หลังจากทำการพัฒนาโปรแกรมเรียบร้อยแล้ว เมื่อทำการ Compile โปรแกรม File ที่เป็น library จะถูกสร้างให้เป็น Object ที่เป็นส่วนที่สำคัญในการใช้งานโปรแกรมต่อไป โดยโปรแกรมจะสร้าง file ที่มีส่วนขยายเป็น .pbd ให้โดยอัตโนมัติ ทั้งนี้เพื่อให้ง่ายต่อการปรับปรุงแก้ไขโปรแกรมในอนาคตต่อไปนั่นเอง

ในระบบสารสนเทศเพื่อควบคุมการผลิตของบริษัทลำพูนชิงเคนเกิน จำกัดจะมีการสร้างlibrary ที่ใช้ในการพัฒนาระบบดังนี้

ตาราง 5.2 แสดง library ของระบบสารสนเทศเพื่อควบคุมการผลิตของบริษัทลำพูนชิงเคนเกิน จำกัด

Library	คำอธิบาย
lsdstde.pbl	เป็น library ที่ใช้ในการเก็บข้อมูล Application และ เมนูหลักของระบบ
lsdstd1.pbl	เป็น library ที่ใช้ในการเก็บ Object ที่สร้างขึ้นเป็นกรอบมาตรฐานของการพัฒนาระบบ(Frame Work) เพื่อใช้คุณสมบัติทางด้านการถ่ายทอดคุณลักษณะ(Inherit) ไปยัง Object หลักของระบบสารสนเทศเพื่อควบคุมการผลิตของบริษัทลำพูนชิงเคนเกิน จำกัด ต่อไป
pcs01.pbl	เป็น library ที่ใช้ในการเก็บ Object หลักของระบบสารสนเทศเพื่อควบคุมการผลิตของบริษัทลำพูนชิงเคนเกิน จำกัด
managedb.pbl	เป็น library ที่ใช้ในการเก็บ Application ที่ใช้ในการ Restore Database

จากตาราง 5.2 สามารถแสดงและอธิบาย Object ที่สำคัญในแต่ละ library ของระบบสารสนเทศเพื่อควบคุมการผลิตของบริษัทลำพูนชิงเดินเกิน จำกัด ได้ดังนี้

1) **Isdstde.pbl** ประกอบไปด้วย Object ที่สำคัญดังนี้

ตาราง 5.3 แสดง Object ต่างๆใน library Isdstde.pbl

ลำดับ	ออบเจกต์	ประเภทออบเจกต์	คำอธิบาย
1	pcs	application	application หลักของระบบ
2	m_main_mdi	menu	เมนูหลักของระบบ

2) **Isdstd1.pbl** ประกอบไปด้วย Object ที่สำคัญดังนี้

ตาราง 5.4 แสดง Object ต่างๆใน library Isdstd1.pbl

ลำดับ	ออบเจกต์	ประเภทออบเจกต์	คำอธิบาย
1	d_system_error	datawindow	ใช้ในการแสดงรายละเอียดของข้อผิดพลาดต่างๆของระบบที่เกิดขึ้น
2	f_dwdelete	function	เป็น function ที่ใช้ในการลบข้อมูลใน datawindow ใช้ในกรณีที่ datawindow แสดงข้อมูลในลักษณะข้อมูลที่ละ record
3	f_dwdeleterow	function	เป็น function ที่ใช้ในการลบข้อมูลใน datawindow ทีละแถว ใช้ในกรณีที่ datawindow แสดงข้อมูลในลักษณะที่เป็นแถว
4	f_dwupdate	function	เป็น function ที่ใช้ในการบันทึกข้อมูลใน datawindow ใช้ในกรณีที่ datawindow แสดงข้อมูลในลักษณะข้อมูลที่ละ record
5	f_dwupdate_mutirow	function	เป็น function ที่ใช้ในการบันทึกข้อมูลใน datawindow ใช้ในกรณีที่ datawindow แสดงข้อมูลในลักษณะที่เป็นแถว
6	m_sheet_dw_multi_row	menu	เมนู Operation ต่างๆ เช่น เพิ่มข้อมูล ลบข้อมูล บันทึกข้อมูล เป็นต้น ที่ใช้กับ datawindow ที่มีลักษณะเป็น Multi Row

ตาราง 5.4 แสดง Object ต่างๆ ใน library lsdstd1.pbl (ต่อ)

ลำดับ	ออบเจกต์	ประเภทออบเจกต์	คำอธิบาย
7	m_sheet_dw_single_ row	menu	เมนู Operation ต่างๆ เช่น เพิ่มข้อมูล ลบ ข้อมูล บันทึกข้อมูล เป็นต้น ที่ใช้กับ datawindow ที่มีลักษณะเป็น Single Row หรือข้อมูลที่แสดงทีละ record
8	m_sheet_report_ condition	menu	เมนู Operation ต่างๆ ที่ใช้ในหน้าจอที่ใช้ ในการออกรายงาน เช่น condition, print, First page, Last Page เป็นต้น
9	w_about	window	หน้าจอที่ใช้ในการแสดงข้อมูลผู้จัดทำ ระบบ
10	w_ancestor_report	window	หน้าจอที่ใช้เป็นต้นแบบของหน้าจอที่ใช้ ในการออกรายงาน ที่ใช้ในการ Inherit ต่อไป
11	w_chg_password	window	หน้าจอการเปลี่ยนรหัสผ่าน
12	w_login	window	หน้าจอ Login
13	w_mdiframe	window	หน้าจอหลักของระบบที่ใช้ในการ เชื่อมต่อกับเมนูหลักของระบบ
14	w_popup	window	หน้าจอที่ใช้แสดงปุ่มทางลัดเพื่อเข้าสู่ หน้าจอต่างๆ
15	w_sheet_recording_ ancestor	window	หน้าจอที่ใช้เป็นต้นแบบของหน้าจอที่ใช้ ในการรับข้อมูล ที่ใช้ในการ Inherit ต่อไป
16	w_system_error	window	หน้าจอแสดงข้อผิดพลาดของการทำงาน ของหน้าจอต่างๆ

3) pcs01.pbl ประกอบไปด้วย Object ที่สำคัญดังนี้

ตาราง 5.5 แสดง Object ต่างๆ ใน library pcs01.pbl

ลำดับ	ออบเจกต์	ประเภทออบเจกต์	คำอธิบาย
1	d_gproduct_input	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้รับข้อมูลกลุ่มผลิตภัณฑ์
2	d_gproduct_list	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้แสดงข้อมูลกลุ่มผลิตภัณฑ์
3	d_input_d	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้รับข้อมูลรายละเอียดการผลิต
4	d_input_h	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้รับข้อมูลการผลิต
5	d_input_ng	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้รับข้อมูลงานเสียที่เกิดขึ้นระหว่างการ ผลิตแต่ละกระบวนการ
6	d_input_problem	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้รับข้อมูลปัญหาที่เกิดขึ้นระหว่างการ ผลิตแต่ละกระบวนการ
7	d_input_process_ select	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้แสดงข้อมูลกระบวนการที่ต้องการ เลือกเพื่อบันทึกข้อมูล
8	d_input_search	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้แสดงข้อมูลการผลิตที่ต้องการค้นหา
9	d_lotcontrolsheet_list	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้แสดงข้อมูลการผลิตที่ต้องการเลือก ออกรายงาน Lot Control Sheet
10	d_ng_import	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้รับข้อมูลงานเสียที่นำเข้ามา
11	d_ng_import_update	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้บันทึกข้อมูลงานเสียที่นำเข้ามา

ตาราง 5.5 แสดง Object ต่างๆใน library pcs01.pbl (ต่อ)

ลำดับ	ออบเจกต์	ประเภทออบเจกต์	คำอธิบาย
12	d_ng_input	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้รับข้อมูลงานเสีย
13	d_ng_list	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้แสดงรายการงานเสียที่บันทึกไปแล้ว
14	d_plan_inf	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้รับข้อมูลแผนการผลิต
15	d_plan_lot	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้รับข้อมูลแผนการผลิตที่ผ่านการ คำนวณแล้ว
16	d_plan_product_list	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้แสดงรายการผลิตภัณฑ์เพื่อเลือกทำ รายการแผนการผลิตประจำเดือน
17	d_process_input	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้รับข้อมูลกระบวนการผลิต
18	d_process_list	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้แสดงรายการกระบวนการผลิต ที่บันทึกไปแล้ว
19	d_product_import	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้รับข้อมูลผลิตภัณฑ์ที่นำเข้ามา
20	d_product_import_ update	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้บันทึกข้อมูลผลิตภัณฑ์ที่นำเข้ามา
21	d_product_input	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้รับข้อมูลผลิตภัณฑ์
22	d_product_list	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้แสดงรายการผลิตภัณฑ์ที่บันทึก ไปแล้ว

ตาราง 5.5 แสดง Object ต่างๆใน library pcs01.pbl (ต่อ)

ลำดับ	ออบเจกต์	ประเภทออบเจกต์	คำอธิบาย
23	d_stdprocess_d	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้รับข้อมูลรายละเอียดมาตรฐาน กระบวนการผลิต
24	d_stdprocess_h	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้รับข้อมูลมาตรฐานกระบวนการผลิต
25	d_stdprocess_list	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้แสดงรายการมาตรฐานกระบวนการ ผลิตที่บันทึกไปแล้ว
26	d_stdprocess_ng	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้รับข้อมูลงานเสียของมาตรฐาน กระบวนการผลิต
27	d_user_import	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้รับข้อมูลผู้ใช้งานระบบที่นำเข้ามา
28	d_user_import_update	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้บันทึกข้อมูลผู้ใช้งานระบบที่นำเข้ามา
29	d_user_input	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้รับข้อมูลผู้ใช้งานระบบ
30	d_user_list	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้แสดงรายการผู้ใช้งานระบบที่บันทึก ไปแล้ว
31	d_userstatus_list	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้แสดงรายชื่อผู้ที่กำลังใช้งานระบบ
32	dddw_gproduct	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้แสดงชื่อกลุ่มผลิตภัณฑ์ใน Drop Down List box
33	dddw_gproduct_ex	datawindow	แสดงชื่อกลุ่มผลิตภัณฑ์ใน Drop Down List box

ตาราง 5.5 แสดง Object ต่างๆใน library pcs01.pbl (ต่อ)

ลำดับ	ออบเจกต์	ประเภทออบเจกต์	คำอธิบาย
34	dddw_ng	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้แสดงชื่องานเสียใน Drop Down List box
35	dddw_process	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้แสดงชื่อกระบวนการผลิตใน Drop Down List box
36	dddw_product	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้แสดงชื่อผลิตภัณฑ์ใน Drop Down List box
37	dddw_stdprocess	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้แสดงชื่อมาตรฐานกระบวนการผลิต ใน Drop Down List box
38	dddw_stdprocess_ex	datawindow	แสดงชื่อมาตรฐานกระบวนการผลิตใน Drop Down List box
39	r_lotcontrolsheet	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้แสดงรายงาน Lot Control sheet
40	r_lotcontrolsheet_ng	datawindow	ส่วนที่ใช้ในการเชื่อมต่อกับฐานข้อมูล ใช้แสดงรายงาน Lot Control sheet ใน หน้าแสดงงานเสีย
41	f_decrypt	function	ฟังก์ชันที่ใช้ในการถอดรหัสผ่าน ผู้ใช้งานระบบ
42	f_encrypt	function	ฟังก์ชันที่ใช้ในการเข้ารหัสผ่าน ผู้ใช้งานระบบ
43	str_pcs	structure	Structure ที่ใช้ในการเก็บตัวแปรข้อมูล การผลิต
44	w_database_backup	window	หน้าจอที่ใช้ในการสำรองข้อมูล
45	w_gproduct	window	หน้าจอข้อมูลกลุ่มผลิตภัณฑ์

ตาราง 5.5 แสดง Object ต่างๆใน library pcs01.pbl (ต่อ)

ลำดับ	ออบเจกต์	ประเภทออบเจกต์	คำอธิบาย
46	w_input	window	หน้าจอรับข้อมูลการผลิต
47	w_input_search	window	หน้าจอค้นหาข้อมูลการผลิต
48	w_input_select	window	หน้าจอเลือกกระบวนการผลิตที่ต้องการบันทึกข้อมูลการผลิต
49	w_ng	window	หน้าจอข้อมูลงานเสีย
50	w_ng_import	window	หน้าจอนำเข้าข้อมูลงานเสีย
51	w_plan	window	หน้าจอข้อมูลแผนการผลิต
52	w_process	window	หน้าจอข้อมูลกระบวนการผลิต
53	w_product	window	หน้าจอข้อมูลผลิตภัณฑ์
54	w_product_import	window	หน้าจอนำเข้าข้อมูลผลิตภัณฑ์
55	w_r_locontrol_sheet	window	หน้าจอออกรายงาน Lot Control Sheet
56	w_r_locontrol_sheet_condition	window	หน้าจอระบุเงื่อนไขที่ต้องการออกรายงาน Lot Control Sheet
57	w_select_ng	window	หน้าจอเลือกข้อมูลงานเสีย
58	w_select_process	window	หน้าจอเลือกข้อมูลกระบวนการ
59	w_stdprocess	window	หน้าจอข้อมูลมาตรฐานกระบวนการผลิต
60	w_stdprocess_ng	window	หน้าจอข้อมูลงานเสียในมาตรฐานกระบวนการผลิตแต่ละกระบวนการ
61	w_stdprocess_search	window	หน้าจอค้นหากระบวนการผลิต
62	w_user	window	หน้าจอข้อมูลผู้ใช้งานระบบ
63	w_user_import	window	หน้าจอนำเข้าข้อมูลผู้ใช้งานระบบ
64	w_userstatus	window	หน้าจอข้อมูลสถานะการใช้งานระบบ

4) managedb.pbl ประกอบไปด้วย Object ที่สำคัญดังนี้

ตาราง 5.6 แสดง Object ต่างๆใน library managedb.pbl

ลำดับ	ออบเจกต์	ประเภท ออบเจกต์	คำอธิบาย
1	managedb	application	Application ที่ใช้ในการ Restore Database
2	w_database_restore	window	เมนูหลักของระบบ

5.2 การพัฒนาโปรแกรมโดยโปรแกรม Chart Director

Chart Director เป็นโปรแกรมที่พัฒนาโดยบริษัท Advanced Software Engineering การทำงานของโปรแกรมจะอยู่ในรูปแบบของ Library ซึ่งมีวัตถุประสงค์หลักในการสร้างกราฟในรูปแบบต่างๆ ไม่ว่าจะเป็นกราฟแท่ง กราฟเส้น กราฟวงกลม ฯลฯ ในรูปแบบของ Web Application ในระบบสารสนเทศเพื่อควบคุมการผลิตของบริษัทลำพูนชิงเด็นเกิน จำกัด จะมีการเรียกใช้ library ของ Chart Director ผ่านหน้าจอที่เป็น Active Server Page หรือ ASP โดยการแสดงผลของ Chart Director จะทำการแปลง Script หรือ code จากภาษาคอมพิวเตอร์ เปลี่ยนให้อยู่ในรูปแบบของรูปภาพ โดยมีตัวอย่างของการเขียนโปรแกรมแสดงดังนี้

```
<%
Set cd = CreateObject("ChartDirector.API")

'The data for the bar chart
data1 = Array(1,2,3)

'The labels for the bar chart. The labels contains embedded images as icons.
labels = Array("1","2","3")

'Create a XYChart object of size 600 x 350 pixels, using 0xe0e0ff as the
'background color, 0xccccff as the border color, with 1 pixel 3D border effect.
Set c = cd.XYChart(600+70, 350+110, &He0e0ff, &Hccccff, 1)

'Set default directory for loading images from current script directory
Call c.setSearchPath(Server.MapPath("."))

'Add a title to the chart using 14 points Times Bold Italic font and light blue
'(0x9999ff) as the background color
if Lot = "" or Lot = "ALL" then
    Call c.addTitle("Average Production Lead Time <br> Product Name :: " & PName & " ("
    &PCODE&") <br>Plan :: " & Plan, "timesbi.ttf", 14 _
    ).setBackground(&H9999ff)
else
    Call c.addTitle("Production Lead Time <br> Product Name :: " & PName & "(" &PCODE&")
    <br>Plan :: " & Plan & " Lot No. :: " & Lot, "timesbi.ttf", 14 _
    ).setBackground(&H9999ff)
end if
```

```

'Set the plotarea at (60, 45) and of size 500 x 210 pixels, using white
'(0xffffffff) as the background
Call c.setPlotArea(60+40, 45+50, 500+20, 210+80, &Hffffff)

'Swap the x and y axes to create a horizontal bar chart
Call c.swapXY()

'Add a title to the y axis using 11 pt Times Bold Italic as font
Call c.yAxis().setTitle("Process L/T (Min) <br> Total Lead Time = "& TotalLeadtime&" Mins",
"timesbi.ttf", 11)

'Set the labels on the x axis
Call c.xAxis().setLabels(labels)

'Disable x-axis ticks by setting the tick length to 0
Call c.xAxis().setTickLength(0)

'Add a stacked bar layer to the chart
Set layer = c.addBarLayer2(cd.Stack)

'Add the first two data sets to the chart as a stacked bar group
Call layer.addDataSet(data1, &H6666ff, "")

'Set the sub-bar gap to 0, so there is no gap between stacked bars with a group
Call layer.setBarGap(0.2, 0)

'Set the bar border to transparent
Call layer.setBorderColor(cd.Transparent)

'Set the aggregate label format
Call layer.setAggregateLabelFormat( _"{value} Min.")

'Set the aggregate label font to 8 point Arial Bold Italic
Call layer.setAggregateLabelStyle("arialbi.ttf", 8)

'Reverse 20% space at the right during auto-scaling to allow space for the
'aggregate bar labels
Call c.yAxis().setAutoScale(0.2)

'Add a legend box at (310, 300) using TopCenter alignment, with 2 column grid
'layout, and use 8 pts Arial Bold Italic as font
Set legendBox = c.addLegend2(310, 300, 2, "arialbi.ttf", 8)
Call legendBox.setAlignment(cd.TopCenter)

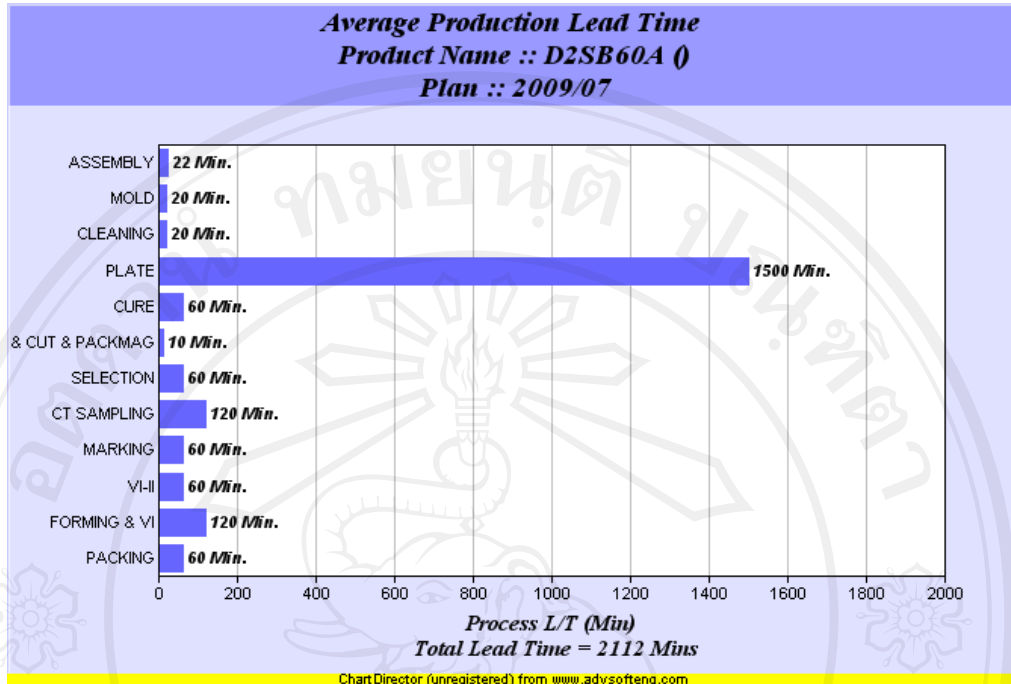
'Set the format of the text displayed in the legend box
Call legendBox.setText("Year {dataGroupName} {dataSetName} Revenue")

'Set the background and border of the legend box to transparent
Call legendBox.setBackground(cd.Transparent, cd.Transparent)

'output the chart
Response.ContentType = "image/png"
Response.BinaryWrite c.makeChart2(cd.PNG)
Response.End
%>

```

โดยตัวอย่างผลที่ได้จากการใช้งาน Chart director สามารถแสดงได้ดังรูป



รูป 5.2 แสดงกราฟที่ได้จากการใช้โปรแกรม Chart Director ผ่าน ASP

5.3 การเข้าและถอดรหัสข้อมูลผู้ใช้งานระบบ

การพัฒนาาระบบสารสนเทศเพื่อควบคุมการผลิตของบริษัทลำพูนชิงเดินเกิน จำกัด ได้ทำการเข้าและถอดรหัสในส่วนของการรหัสผ่านของผู้ใช้งานระบบ โดยใช้หลักการแทนที่ตัวอักษรด้วยรหัส Ascii (Ascii Code) โดยมีรายละเอียดดังนี้

Dec	Char	Dec	Char	Dec	Char	Dec	Char	Dec	Char	Dec	Char	Dec	Char	Dec	Char	Dec	Char	Dec	Char	Dec	Char
1	0	26	0	51	3	76	L	101	e	126	~	151	—	176	ร	201	๕	226	โ	251	๐๘
2	0	27		52	4	77	M	102	f	127	0	152	0	177	๗	202	๖	227	ใ	252	๐
3	0	28		53	5	78	N	103	g	128	€	153	0	178	๘	203	๗	228	ใ	253	๐
4	0	29		54	6	79	O	104	h	129	0	154	0	179	๙	204	๘	229	๗	254	๐
5	0	30		55	7	80	P	105	i	130	0	155	0	180	๐	205	๙	230	๗	255	๐
6	0	31		56	8	81	Q	106	j	131	0	156	0	181	๑	206	๐	231	๘		
7	0	32		57	9	82	R	107	k	132	0	157	0	182	๑	207	๑	232	๘		
8	0	33	!	58	:	83	S	108	l	133	...	158	0	183	๒	208	๑	233	๘		
9		34	"	59	;	84	T	109	m	134	0	159	0	184	๓	209	๑	234	๘		
10		35	#	60	<	85	U	110	n	135	0	160	0	185	๓	210	๑	235	๘		
11	0	36	\$	61	=	86	V	111	o	136	0	161	๐	186	๓	211	๑	236	๘		
12	0	37	%	62	>	87	W	112	p	137	0	162	๐	187	๓	212	๑	237	๘		
13		38	&	63	?	88	X	113	q	138	0	163	๐	188	๓	213	๑	238	๘		
14	0	39	'	64	@	89	Y	114	r	139	0	164	๐	189	๓	214	๑	239	๘		
15	0	40	(65	A	90	Z	115	s	140	0	165	๐	190	๓	215	๑	240	๘		
16	0	41)	66	B	91	[116	t	141	0	166	๐	191	๓	216	๑	241	๘		
17	0	42	*	67	C	92	\	117	u	142	0	167	๐	192	๓	217	๑	242	๘		
18	0	43	+	68	D	93]	118	v	143	0	168	๐	193	๓	218	๑	243	๘		
19	0	44	,	69	E	94	^	119	w	144	0	169	๐	194	๓	219	๑	244	๘		
20	0	45	-	70	F	95	_	120	x	145	'	170	๐	195	๓	220	๑	245	๘		
21	0	46	.	71	G	96	`	121	y	146	'	171	๐	196	๓	221	๑	246	๘		
22	0	47	/	72	H	97	a	122	z	147	"	172	๐	197	๓	222	๑	247	๘		
23	0	48	0	73	I	98	b	123	{	148	"	173	๐	198	๓	223	๑	248	๘		
24	0	49	1	74	J	99	c	124		149	"	174	๐	199	๓	224	๑	249	๘		
25	0	50	2	75	K	100	d	125	}	150	-	175	๐	200	๓	225	๑	250	๘		

รูป 5.3 แสดงตาราง Ascii Code ที่ใช้ในระบบ

1) การเข้ารหัส มีวิธีการในการเข้ารหัสดังนี้

- (1) นำรหัสผ่านของผู้ใช้งานระบบแต่ละคนมาแยกเป็นอักขระแต่ละตัว
- (2) นำอักขระแต่ละตัวมาแทนที่ตามตารางข้างต้น โดยอักขระใดที่แทนค่าเป็นตัวเลขแล้วมีความยาวไม่ถึง 3 หลักให้เติมเลข 0 ด้านหน้าอักขระนั้นๆ

เช่น

A แทนค่าด้วย	65	ดังนั้น แทนที่เป็น	065
q แทนค่าด้วย	133	ดังนั้น แทนที่เป็น	133 เป็นต้น

- (3) ใช้เครื่องหมาย “%” นำหน้าการแทนที่ที่ได้จากข้อ (2) เพื่อเป็นการสิ้นสุดการเข้ารหัสแต่ละอักขระ

เช่น

A เข้ารหัสได้เป็น %065
q เข้ารหัสได้เป็น %133 เป็นต้น

ในการเขียนโปรแกรมเพื่อเข้ารหัสโดยใช้โปรแกรม Power Builder 8.0 นั้นจะใช้คำสั่ง ASC() ในการเปลี่ยนค่าตัวอักษรให้เป็น Ascii Code โดยสามารถแสดงคำสั่งที่ใช้ในการเขียนโปรแกรมได้ดังนี้

```
String ls_convert,ls_encrypt
Integer li_char,i

For i = 1 to len(trim(as_pass))
  li_char = ASC(mid(trim(as_pass),i,1))
  if len(string(li_char)) < 3 then
    ls_convert = ls_convert + '%' + Fill('0',3 - len(string(li_char))) + String(li_char)
  else
    ls_convert = ls_convert + '%' + String(li_char)
  end if
Next

Return ls_convert
```

2) การถอดรหัส มีวิธีการในการถอดรหัสดังนี้

- (1) นำรหัสผ่านของผู้ใช้งานระบบที่เข้ารหัสผ่านแล้วมาทำการตัดอักขระทีละ 4 ตัวอักษร
- (2) ใช้คำสั่งตัดข้อความ Right(text,3) เพื่อตัดเครื่องหมาย “%” ออกจากการแทนที่ที่ได้จากข้อ (1)

เช่น

%065 เมื่อใช้คำสั่ง Right(text,3) จะได้เป็น 065 เป็นต้น

- (3) นำข้อความจากข้อ (2) มาทำการค้นหว่าตรงกับอักขระใดตารางในรูป 5.3

(4) นำอักขระที่ได้จากการถอดรหัสในข้อ (3) มาเรียงต่อกันจึงจะสามารถ
ถอดรหัสข้อมูลได้

ในการเขียนโปรแกรมเพื่อถอดรหัสโดยใช้โปรแกรม Power Builder 8.0 นั้น
สามารถแสดงคำสั่งที่ใช้ในการเขียนโปรแกรมได้ดังตัวอย่างต่อไปนี้

```

long i
string ls_code,ls_decrypt,ls_return,ls_char

For i = 1 to len(as_pass) Step 4

    ls_code = Right(mid(as_pass,i,4),3)

    if ls_code >= '051' AND ls_code <= '100' then
        CHOOSE CASE ls_code
            CASE '051'
                ls_char = '3'
            CASE '052'
                ls_char = '4'
            CASE '053'
                ls_char = '5'
            CASE '054'
                ls_char = '6'
            CASE '055'
                ls_char = '7'
            CASE '056'
                ls_char = '8'
            CASE '057'
                ls_char = '9'
            CASE '058'
                ls_char = ':'
            CASE '059'
                ls_char = ';'
            CASE '060'
                ls_char = '<'
            CASE '061'
                ls_char = '='
            CASE '062'
                ls_char = '>'
            CASE '063'
                ls_char = '?'
            CASE '064'
                ls_char = '@'
            CASE '065'
                ls_char = 'A'
            CASE '066'
                ls_char = 'B'
            CASE '067'
                ls_char = 'C'
            CASE '068'
                ls_char = 'D'
            CASE '069'
                ls_char = 'E'
            CASE '070'
                ls_char = 'F'

            CASE '071'
                ls_char = 'G'
        
```

```

CASE '072'
    Is_char = 'H'
CASE '073'
    Is_char = 'I'
CASE '074'
    Is_char = 'J'
CASE '075'
    Is_char = 'K'
CASE '076'
    Is_char = 'L'
CASE '077'
    Is_char = 'M'
CASE '078'
    Is_char = 'N'
CASE '079'
    Is_char = 'O'
CASE '080'
    Is_char = 'P'
CASE '081'
    Is_char = 'Q'
CASE '082'
    Is_char = 'R'
CASE '083'
    Is_char = 'S'
CASE '084'
    Is_char = 'T'
CASE '085'
    Is_char = 'U'
CASE '086'
    Is_char = 'V'
CASE '087'
    Is_char = 'W'
CASE '088'
    Is_char = 'X'
CASE '089'
    Is_char = 'Y'
CASE '090'
    Is_char = 'Z'
CASE '091'
    Is_char = '['
CASE '092'
    Is_char = '\'
CASE '093'
    Is_char = ']'
CASE '094'
    Is_char = '^'
CASE '095'
    Is_char = '_'
CASE '096'
    Is_char = '`'
CASE '097'
    Is_char = 'a'
CASE '098'
    Is_char = 'b'
CASE '099'
    Is_char = 'c'
CASE '100'
    Is_char = 'd'

```

END CHOOSE

```
Elseif ls_code >= '101' AND ls_code <= '150' then
```

```
  CHOOSE CASE ls_code
```

```
    CASE '101'
```

```
      ls_char = 'e'
```

```
    CASE '102'
```

```
      ls_char = 'f'
```

```
    CASE '103'
```

```
      ls_char = 'g'
```

```
    CASE '104'
```

```
      ls_char = 'h'
```

```
    CASE '105'
```

```
      ls_char = 'i'
```

```
    CASE '106'
```

```
      ls_char = 'j'
```

```
    CASE '107'
```

```
      ls_char = 'k'
```

```
    CASE '108'
```

```
      ls_char = 'l'
```

```
    CASE '109'
```

```
      ls_char = 'm'
```

```
    CASE '110'
```

```
      ls_char = 'n'
```

```
    CASE '111'
```

```
      ls_char = 'o'
```

```
    CASE '112'
```

```
      ls_char = 'p'
```

```
    CASE '113'
```

```
      ls_char = 'q'
```

```
    CASE '114'
```

```
      ls_char = 'r'
```

```
    CASE '115'
```

```
      ls_char = 's'
```

```
    CASE '116'
```

```
      ls_char = 't'
```

```
    CASE '117'
```

```
      ls_char = 'u'
```

```
    CASE '118'
```

```
      ls_char = 'v'
```

```
    CASE '119'
```

```
      ls_char = 'w'
```

```
    CASE '120'
```

```
      ls_char = 'x'
```

```
  END CHOOSE
```

```
Else
```

```
  CHOOSE CASE ls_code
```

```
    CASE '251'
```

```
      ls_char = ''
```

```
    CASE '252'
```

```
      ls_char = ''
```

```
    CASE '253'
```

```
      ls_char = ''
```

```
  END CHOOSE
```

```
End if
```

```
ls_decrypt = ls_decrypt + ls_char
```

```
Next
```

```
Return ls_decrypt
```