บทที่ 4

BREW

4.1 BREW (Binary Runtime Environment for Wireless)

หลายล้านคนทั่วโลกเห็นว่าโทรศัพท์เป็นเครื่องมือสื่อสารที่จำเป็นอีกอันหนึ่ง เทคโนโลยี BREW (Binary Runtime Environment for Wireless) ถือเป็น Hardware platform ใหม่โดยเฉพาะประเทศไทยและกำลังเริ่มนิยมในการเขียนโปรแกรมบน mobile phones แบบใหม่ซึ่งจะ กลายเป็นที่นิยมในโลกของการสื่อสารไร้สายและโทรศัพท์มือถือ มีผู้ผลิตมือถือรายใหญ่หลายรายสนใจที่จะผลิตอุปกรณ์มือถือที่มีความสามารถในการรองรับสนับสนุนเทคโนโลยี BREW นี้ ปัจจุบัน BREW มีการพัฒนามาแล้ว 4 version หลักได้แก่ 1.0 1.1 2.0 3.0 แต่เครื่องมือถือในประเทศไทยปัจจุบันรองรับแค่เวอร์ชัน 2.0 เท่านั้นเอง

อีกตัวหนึ่งที่เกี่ยวข้องกับเทคโนโลยี BREW คือ SDK ตัว BREW SDK นี้เป็น library ที่โปรแกรมเมอร์สามารถใช้ในการสร้าง application สำหรับ BREW Platform โดยใช้ภาษา C หรือ C++ แต่ส่วนมากแล้วและตัวอย่างส่วนมากเป็นภาษา C ซึ่ง SDK ได้มีฟังก์ชันที่ใช้ส่วนมากประกอบด้วยฟังก์ชันทางด้าน Graphics, Networking, Sound, Text Display, GUIs และอื่นๆอีกมากมายและจุดเด่นของมันคือมี Emulator ให้ใช้ในการทดลองโดยที่สามารถนำมาทดลองโปรแกรมที่เราเขียนขึ้นมาบนเครื่องคอมพิวเตอร์ที่มีระบบปฏิบัติการ Windows แทนที่จะต้องทดลองไปทดลองโปรแกรมกับอุปกรณ์ของจริงทำให้ง่ายต่อการทดลองและตรวจแก้ไขโปรแกรม มันสามารถที่จะกำหนดคุณสมบัติของ Emulator ได้เช่นกำหนดขนาดของหน่วยความจำ ความละเอียดของหน้าจอเป็นต้น แต่ Emulator ตัวนี้ก็ไม่สามารถที่จะทดลองได้เหมือนอุปกรณ์จริงทุกประการ

และสิ่งที่ BREW แตกต่างไปจากเทคโนโลยีอื่น เช่น J2ME คือการเผยแพร่และการกำหนดมาตรฐานของ Application ที่จะทำงานบนเทคโนโลยี BREW เป็นลักษณะที่จะมีองค์กรกลางที่คอยให้มาตรฐานของ Application นั้นที่จะเอาขึ้นไปทำงานบนอุปกรณ์จริงซึ่งจะระบบระเบียบของการขอมาตรฐานนี้เช่นต้องเป็นสมาชิกหรือต้องมีการทดสอบ Application กับทางองค์กรก่อน

4.1.1 หลักการพื้นฐานของ BREW Platform

ลักษณะของ BREW ในมุมมองของผู้ที่พัฒนา Application หลักๆ มีดังนี้

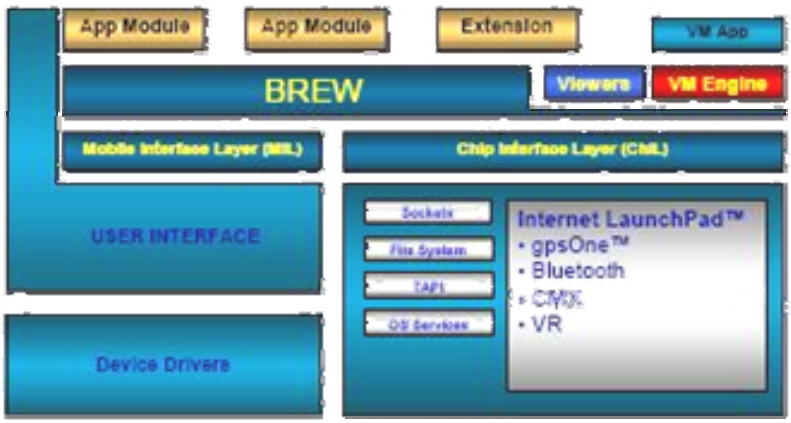
1. BREW เป็นกลุ่มของฟังก์ชันการทำงาน(APIs)ที่ใช้ในการเขียนโปรแกรมสำหรับ Mobile devices

2. มีฟังก์ชันการทำงานมากมายที่มีให้ใช้ทั้งเข้า access อุปกรณ์เช่น Display , การจัดการเกี่ยวกับ Memory , การจัดการเกี่ยวกับ File เป็นต้น
3. เป็นตัวกลางไม่ให้โปรแกรมเมอร์ต้องไปจัดการกับอุปกรณ์ภายในของ Mobile devices

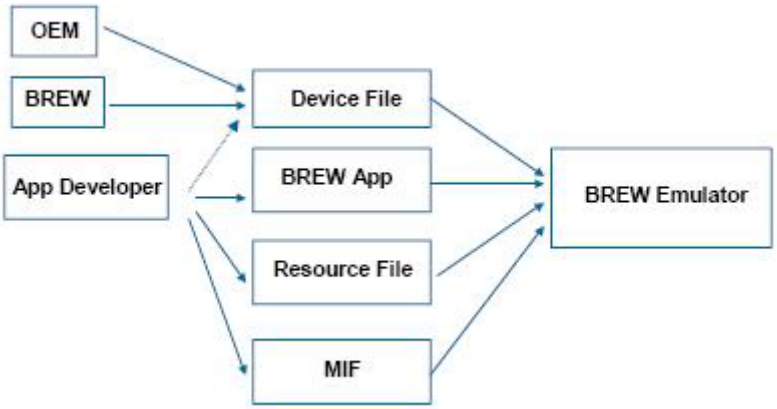
4.1.2 ระบบ BREW โดยรวม

1. BDS for Carriers
2. Open platform for device OEMs
3. BREW SDK application developers
4. Application for end-user download

4.1.3 สถาปัตยกรรมและระบบการทำงานของ BREW



รูปที่ 4-1 แสดง Device Architecture (Layer) ของ BREW



รูปที่ 4-2 แสดง ระบบการทำงานของ BREW

4.1.4 BREW SDK COMPONENTS

1. BREW Emulator ใช้ในการทดลองโปรแกรมที่เขียนบนเครื่องคอมพิวเตอร์
2. BREW Resource Editor เป็นโปรแกรมช่วยสร้าง Resource ที่จะใช้ใน Application ที่สร้าง
3. BREW MIF Editor เป็นโปรแกรมที่ช่วยสร้าง module information files(MIF) ที่จะเป็นตัวบอกว่า Application ที่จะสร้างขึ้นนั้นจะมีลักษณะแบบไหน
4. BREW Compressed Image Authoring Tool เป็นโปรแกรมที่ช่วยในการสร้างภาพที่เป็นภาพเคลื่อนไหวที่จะใช้ใน BREW เทคโนโลยี

4.1.5 MODULE INFORMATION FILE (MIF)

เป็นไฟล์ที่บอกถึงลักษณะของ Application ที่โปรแกรมเมอร์ที่จะสร้างและบอกว่าจะใช้งาน Module ไหนในการสร้าง Application แต่ละ Module มีการร้องขอสิ่งต่างๆ ดังนี้

1. หมายเลขของ Application
2. ClassID ของแต่ละ Module ที่มีใช้ Application
3. รวมทั้ง Title, Icons ที่จะใช้ในแต่ละ Application

4.1.6 ความแตกต่างระหว่าง Handset กับ Emulator

1. Speed แน่นอนเรื่องความเร็ว Emulator จะทำงานเร็วกว่าอุปกรณ์จริง (Handset)
2. Appearance ความแตกต่างในการแสดงผลของ Emulator และ Handset
3. Memory เรื่องของหน่วยความจำ Emulator มีมากกว่า Handset ซึ่ง Emulator สามารถกำหนดได้

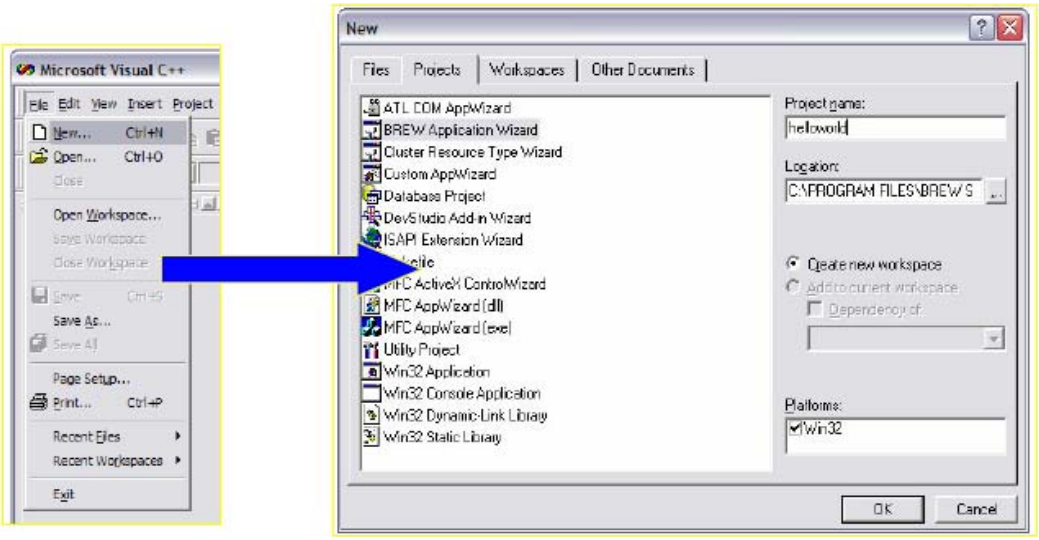
ดังนั้นในขั้นตอนการพัฒนาโปรแกรมต้องคำนึกถึงเรื่องนี้ด้วยเพราะอาจทดลองบน Emulator สามารถทำงานได้แต่พอเอาโปรแกรมมาลงบนอุปกรณ์จริงแล้วอาจไม่สามารถทำงานเนื่องจากเงื่อนไขต่างๆ ที่ Emulator มีความแต่ต่างไปจากอุปกรณ์จริงเช่นเรื่องหน่วยความจำหรือจะเป็นเรื่องความเร็วของทั้งสองแบบที่มีความแตกต่างกัน

4.2 การเขียน BREW Application

การเขียน BREW Application ที่ถูกใช้ในการพัฒนาครั้งนี้ เราจะกระทำบน Microsoft Visual Studio 6.0 environment. ในส่วนนี้จะแสดงการแนะนำคร่าวๆ ของการพัฒนาและประมวลผลของ BREW รวมถึงขั้นตอนพื้นฐานที่จำเป็นถึงการเขียนเกี่ยวกับการทำหน้าที่ของ BREW Application

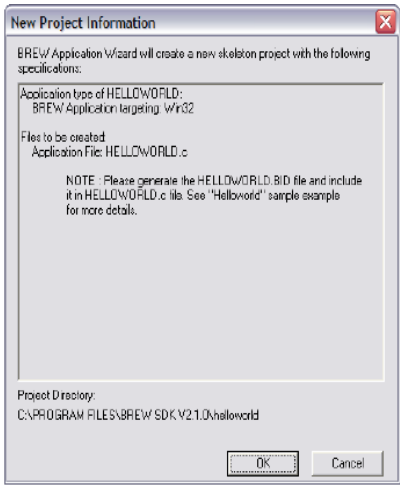
4.2.1 การสร้าง Application ใหม่ด้วย BREW Application Wizard

การใช้ BREW Application Wizard ทำให้สะดวกในการสร้าง Application ใหม่ กรเริ่มต้นใช้ BREW Application Wizard เข้าไปที่ Microsoft Visual Studio แล้วเลือกที่ File>New เพื่อสร้าง Project ใหม่ BREW Application Wizard จะแสดงให้ภายในชนิดเมนูโครงงาน จากนั้นก็ใส่ชื่อของโครงงานที่จะทำการพัฒนาแล้วก็เลือกที่ OK เพื่อเริ่มต้นการเขียน Application ดังรูปที่ 4-4



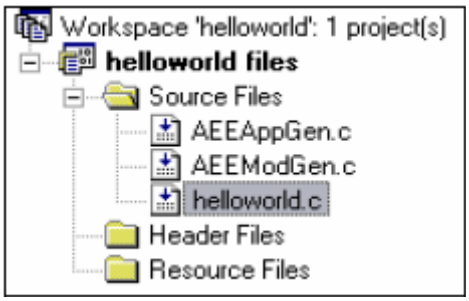
รูปที่ 4-3 แสดง BREW Application Wizard

จากนั้นเมื่อการสร้าง Project ใหม่ก็จะมีการรายงานสรุปการกระทำต่างๆที่ได้เลือกไป



รูปที่ 4-4 สรุป Information

จากนั้นก็เปิดไฟล์ที่ได้ตั้งชื่อไว้ในตัวอย่างนี้มีไฟล์ที่ชื่อว่า helloworld.c (ชื่อไฟล์ยกตัวอย่าง) เป็นไฟล์หลักๆที่ใช้ในการพัฒนา Application



รูปที่ 4-5 แสดง Workspace

ส่วนไฟล์ AEEAppGen.c, AEEModGen.c และ library ไฟล์เหล่านี้เป็นไฟล์ที่ใช้ในการสนับสนุนการพัฒนา BREW Application

ตัวอย่างโครงสร้างของ Code ที่ถูก Generated โดย BREW Application Wizard

```
#include "AEEModGen.h"   // Module interface definitions      (1)
#include "AEEAppGen.h"   // Applet interface definitions
#include "AEEShell.h"    // Shell interface definitions

#include "helloworld.bid"                                      (2)

typedef struct _helloworld {                                   (3)
    // add your own variables here...
                ⋮
} helloworld;
int AEEClsCreateInstance(AEECLSID ClsId, IShell *pIShell,      (4)
    IModule *po,  void **ppObj) {
                            ⋮
}
static boolean helloworld HandleEvent(helloworld* pMe,         (5)
 AEEEvent eCode, uint16 wParam, uint32 dwParam) {
                        ⋮
 }
boolean helloworld_InitAppData(helloworld* pMe) {              (6)
                        ⋮
 }
void helloworld_FreeAppData(helloworld* pMe) {                 (7)
                        ⋮
 }
```

รูปที่ 4-6 แสดง Code ที่ถูก Generated โดย BREW Application Wizard

ความหมายและความสำคัญในแต่ละส่วนของ Code ที่ถูก Generated โดย BREW Application Wizard ซึ่งโดยทั่วไปแล้วจะมีอยู่ประมาณ 7 ส่วน ตามสัญลักษณ์หมายเลขที่แสดงอยู่ด้านหลัง code ดังนี้

1). API Library Includes

เป็นส่วนที่ประกาศว่าไฟล์ที่ถูกรวมเข้ามาซึ่งเป็นไฟล์ที่ใช้เฉพาะเจาะจงของ BREW API และไฟล์ Header (*.h) จำเป็นต้องถูกประกอบด้วยในโปรแกรม ซึ่งส่วนของข้อมูลบนไฟล์ Header นั้นจะถูกบรรจุคำอธิบายใน BREW API Reference. ข้อแนะนำในการป้องกันการใช้ข้อมูลที่ทีตัวแปร Global และ ตัวแปร Static ไม่ควรใช้ C Standard Library ใน BREW Application และส่วยเพิ่มเติมคือ การคำนวณเกี่ยวกับ floating-point นั้นควรใช้ floating-point helper function.

2). Class ID file

ก่อนการ Compile Application เรานั้นจะต้องมีการ Generate ไฟล์ Class ID. ไฟล์นี้จะถูกบรรจุไปด้วยเลขฐาน 16 จำนวน 32 บิต ที่เป็นตัวระบุ Application ที่จะต้องไม่ซ้ำกัน ซึ่งถูกใช้สร้างเป็นกลไกในการติดต่อของ BREW สำหรับการทดสอบบนระบบคอมพิวเตอร์ ซึ่งมันพอเพียงต่อการ Generate Local Class ID ผ่านทาง MIF Editor.

3). Applet Structure

Objects ใดๆ ที่ถูกใช้ในระหว่างการทำงานของโปรแกรม ควรจะถูกวางในที่ applet structure. เช่นตัวแปร Static และตัวแปร global จะไม่ถูกยอมรับภายใน BREW, applet structure จัดเตรียมกลไกสำหรับสถานที่เก็บเอกสารสำคัญๆ Element ของ Applet ทั้งหมดจะถูก initialized โดย Method Initial data ของ Application นั้นๆ

4). AEEClsCreateInstance() Method

Method นี้เป็นที่ต้องการสำหรับ Application ทั้งหมด. BREW ถูกร้องขอ routine นี้โดยอัตโนมัติในการสร้าง Instance. ภายใน Method AEEAppletNew() นี้คือ เรียกว่าหน้าที่การ register event handler, กระบวนการ initialization ข้อมูลและการ deallocation ข้อมูล. กระบวนการ Initialization ทั้งหมด คือถูกกระทำเจาะจงโปรแกรมในการ initialization data. Method AEEClsCreateInstance() มันควรจะไม่ถูกแก้ไข.

5). Event Handler

Application ทั้งหมดของ BREW จะต้องบรรจุ event handler function ที่ใช้ในการเรียกใช้ AEEAppletNew()ภายใน method AEEClsCreateInstance(). Event จะถูกส่งมายังfunction นี้โดย BREW Layer. หลังจากได้รับ Event ควรจัดการให้มีการ return TRUE เพื่อที่เป็นการบอกว่าเป็นการจัดการที่สมบูรณ์หรือ return FALSE ถ้า event ไม่มีการถูกจัดการ. Event codes supports โดย BREW (AEE Event) ที่ถูกแสดงในส่วน data type ของ BREW API Reference.

6). Applet data init method

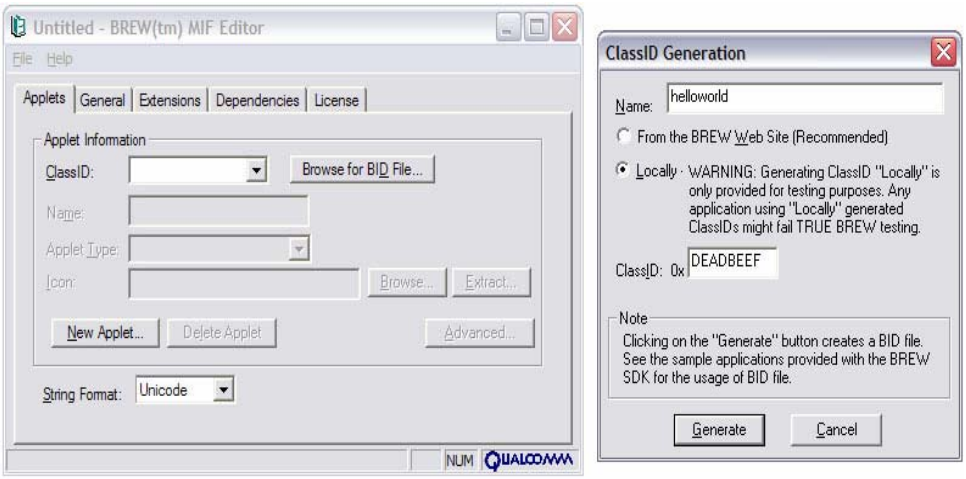
Method นี้ถูกใช้ในการ Initialized code ที่ถูกร้องขอโดย applet. โดยปกติแล้ว Function นี้จะถูกใช้ในการยกตัวอย่าง member applet structure หรือสำหรับการ allocate memory สำหรับ buffer. การ Return ค่าซึ่งเป็นค่า Boolean นั้นถูกใช้ในการประมวลผลไม่ว่าจะเสร็จสมบูรณ์หรือไม่สมบูรณ์. โดยที่ Place code การจัดสรรของคุณใน method เดียวจะลดการละเลยที่อาจจะมีผลต่อสมรรถภาพ memberของ applet structure.

7). Applet data free method

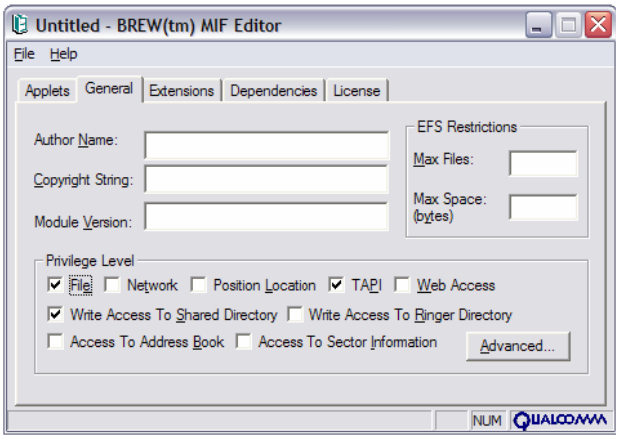
Applet data free method นี้ถูกใช้ในการ deallocate ของทรัพยากรทั้งหมดและการเพิ่มขั้นตอนการถูกร้องขอสำหรับการ safety terminate application. ทรัพยากรทั้งหมดที่ Allocate มาจาก method initialization data ควรจะถูก deallocated ภายใน method นี้.

4.2.2 การใช้ BREW MIF Editor

จะต้องมีการสร้าง MIF ก่อนที่จะวาง Application ไปไว้ใน BREW Device หรือ BREW Simulator. MIF จะบรรจุสิ่งที่มีความสำคัญคือ Information ที่เกี่ยวกับ Application ดังเช่น ชื่อ, Icon, Class ID และ privilege levels. การวางไฟล์ต่างๆในโฟลเดอร์ ที่เป็นโฟลเดอร์ของ Application คือต้องวาง MIF ไฟล์ไว้ระดับเดียวกันกับโฟลเดอร์ที่เก็บไฟล์ที่เป็นไฟล์รันโปรแกรม BREW MIF Editor ถูกบรรยายลักษณะการใช้ในส่วนนี้เป็นเครื่องมือที่ช่วยให้ง่ายต่อการสร้างไฟล์ MIF. ดังแสดงเป็นขั้นตอนดังรูปต่อไปนี้



รูปที่ 4-7 แสดงการ Generate Local class ID

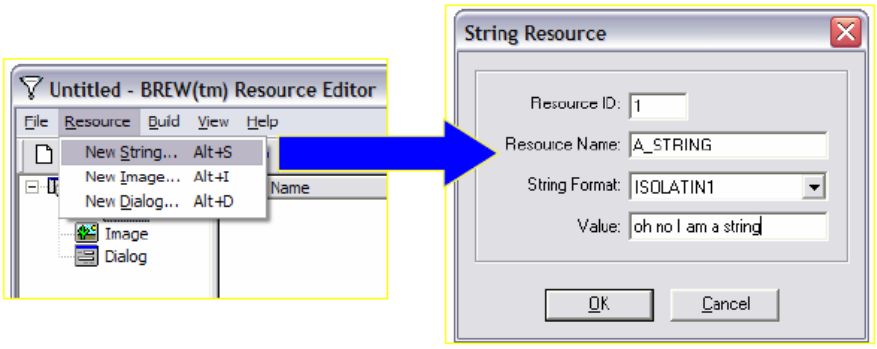


รูปที่ 4-8 แสดง Specify setting privileges

4.2.3 การใช้ BREW Resource Editor

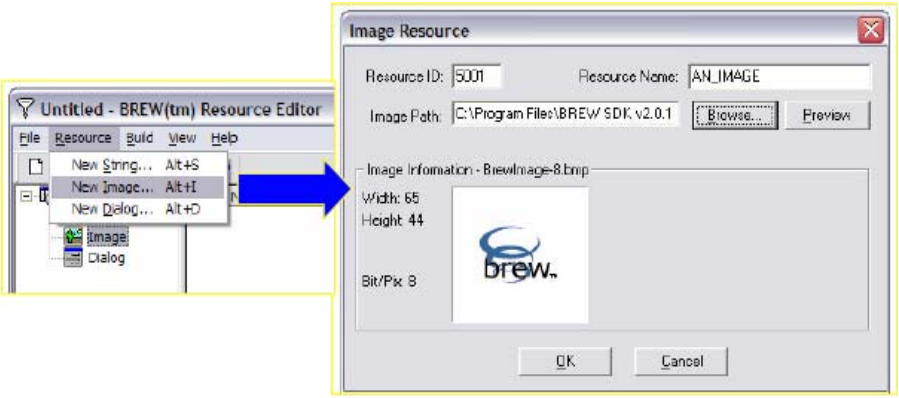
BREW Resource Editor เป็นเครื่องมือที่ถูกใช้ในการสร้างไฟล์ที่เอาไว้เก็บ resource ที่เพื่อจะนำมาใช้ในโปรแกรม โดยจะประกอบไปด้วย resource 3 ชนิดคือ string resource , image resource และ dialog box.

- การสร้าง String resource



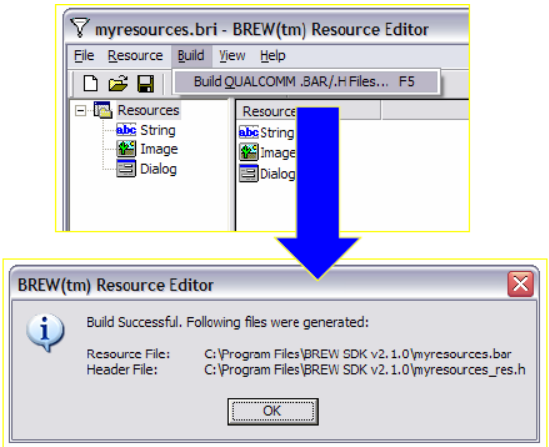
รูปที่ 4-9 แสดงการเพิ่ม string resource

- การสร้าง Image resource



รูปที่ 4-10 แสดงการเพิ่ม Image resource

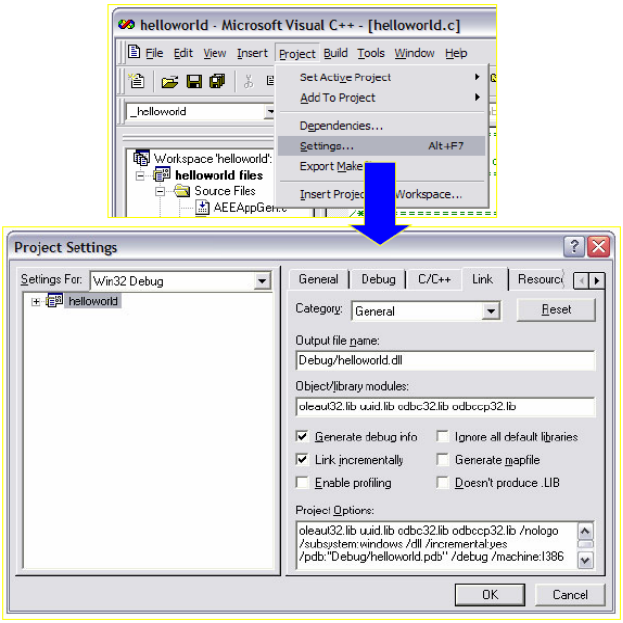
จากนั้นเมื่อสร้าง resource ทั้งหมดตามที่ต้องการแล้วจะต้องมีการ compile ไฟล์ที่ใช้ในการเก็บ resource นี้ด้วย



รูปที่ 4-11 แสดงการ Compile resource file

4.2.4 การ Compile BREW Application เพื่อใช้ในการจำลองการใช้งาน (Simulation)

หลังจากที่เราได้เขียน BREW Application แล้วจะต้องมีการสร้างไฟล์ DLL เพื่อเอามาใช้สำหรับ Simulator และอย่างแรกต้องแน่ใจก่อนว่าได้มีการสร้าง MIF ไฟล์แล้วเก็บไว้ในระดับพาร์ทเดียวกันของ Application และได้สร้างไฟล์ class ID แล้ว ซึ่งโดย default ของไฟล์ DLL นั้นจะถูกระบุอยู่ใน directory ที่มีชื่อว่า Debug (Debug/<projectname.dll>) ซึ่งจะทำให้ตัว Simulator นั้นไม่สามารถ execute application ได้ ดังนั้นจะต้องมีการระบุให้มีการสร้างไฟล์ DLL ใน directory ของ project นั้น โดยถูกแสดงไว้ดังนี้

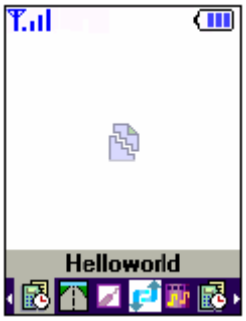


รูปที่ 4-12 แสดงการ Setting output file .dll

ในช่องของ Output file name: นั้นให้ทำการลบคำว่า Debug/ ออก เพื่อทำให้การ compile program แล้วสร้างไฟล์ *.dll ให้อยู่ใน directory project นั้นๆ ซึ่งไฟล์ *.dll นี้ เป็นไฟล์ที่ซึ่งถูกเรียกใช้โดย Simulation ในการทดสอบการทำงานของ Application

4.2.5 การ Run BREW Application เพื่อใช้ในการจำลองการใช้งาน (Simulation)

หลังจากที่มีการ Configure ค่าต่างๆ ที่จะถูกใช้โดย Simulator แล้วและแน่ใจว่าได้สร้าง MIF file ที่เก็บไว้ใน directory ถูกต้องแล้วก็ทำการรัน Application ได้ทันที Build > Start debug > go หรือ กด F5



รูปที่ 4-13 แสดง Application Menu

การวางไฟล์ต่างๆในโฟลเดอร์ ที่เป็นโฟลเดอร์ของ Application คือต้องวาง MIF ไฟล์ไว้ระดับเดียวกันกับโฟลเดอร์ที่เก็บไฟล์ที่เป็นไฟล์รันโปรแกรม