

การออกแบบระบบสร้างเพิ่มข้อมูลพารามิเตอร์สำหรับการพัฒนาโปรแกรม

การสร้างเพิ่มข้อมูลพารามิเตอร์สำหรับการพัฒนาโปรแกรมนี้ประกอบด้วย 4 ส่วนหลัก คือ

1. การสร้างเพิ่มข้อมูลพารามิเตอร์สำหรับเพิ่มข้อมูล (File Parameter File Generating) (ถ้ามี) เป็นการกำหนดรายละเอียดของเพิ่มข้อมูล และรายละเอียดของเขตข้อมูลทั้งหมดในเพิ่มข้อมูลนั้น

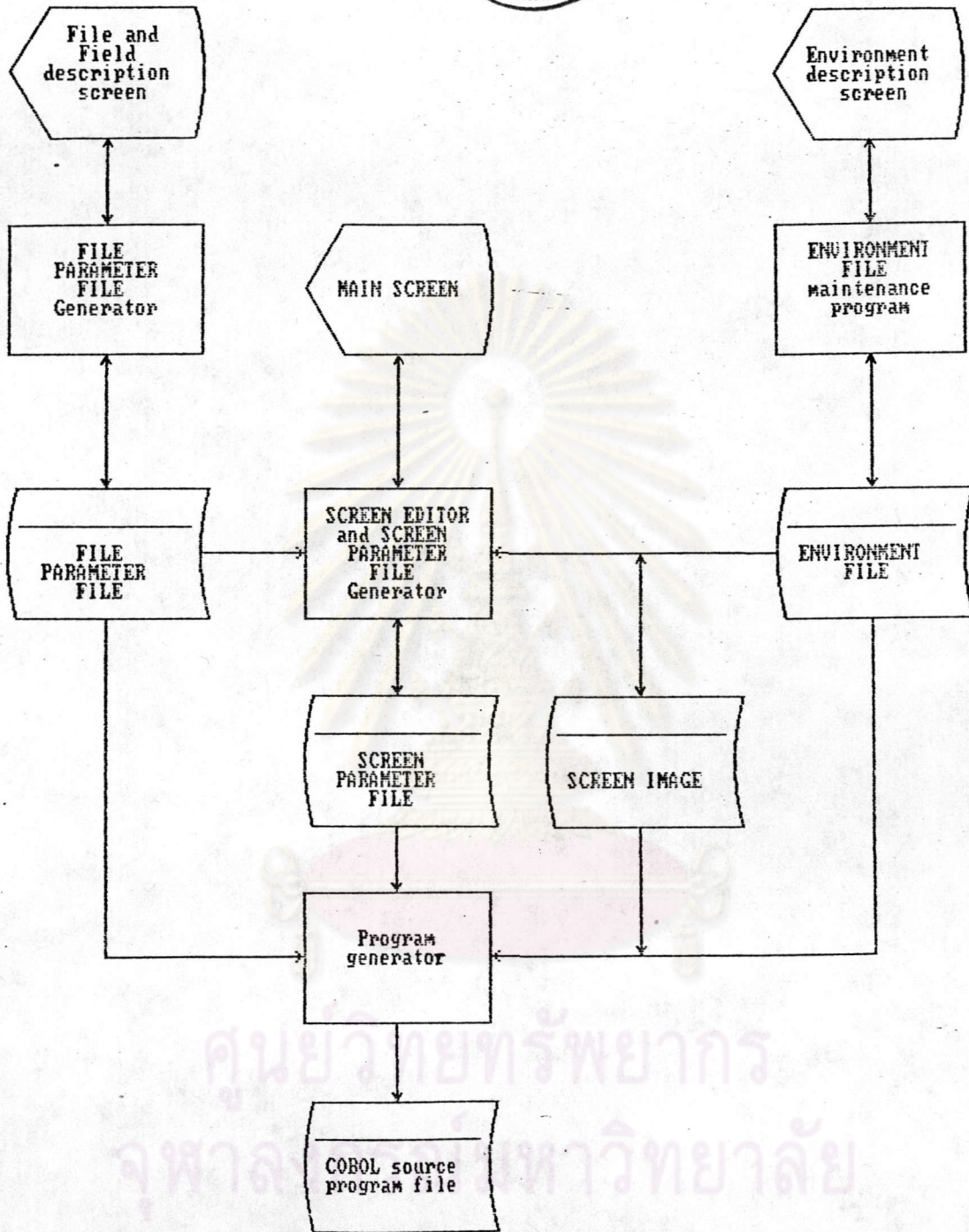
2. การสร้างเพิ่มข้อมูลของสภาพแวดล้อมของระบบ (Environment File Generating) เป็นการกำหนดสภาพแวดล้อมของระบบ เช่น จำนวนเขตข้อมูลสูงสุดที่มีได้ในแต่ละหน้าจอ จำนวนเพิ่มข้อมูลสูงสุดที่มีได้ในระบบ หรือไคเรคทอรีของเพิ่มข้อมูลต่าง ๆ เป็นต้น

3. การออกแบบหน้าจอ (Screen Designing) เป็นการกำหนดข้อความ (Text) ตำแหน่งของเขตข้อมูลและคุณสมบัติต่าง ๆ ของเขตข้อมูล การกำหนดเหล่านี้สามารถกระทำได้ทั้งกับหน้าจอ

นอกจากนี้ ในขั้นตอนนี้ยังมีส่วน การสร้างเพิ่มข้อมูลพารามิเตอร์สำหรับหน้าจอ (Screen Parameter File Generating) เป็นการเก็บข้อมูลที่กำหนดจากการออกแบบหน้าจอลงเพิ่มข้อมูล

4. การสร้างโปรแกรม (Program generating) เป็นการนำข้อมูลจากเพิ่มข้อมูลพารามิเตอร์สำหรับหน้าจอ และเพิ่มข้อมูลพารามิเตอร์สำหรับเพิ่มข้อมูล (ถ้ามี) และเพิ่มข้อมูลของสภาพแวดล้อมของระบบ มาสร้างโปรแกรมภาษาโคบอล ซึ่งเมื่อนำโปรแกรมนั้น ไปประมวลผลจะแสดงหน้าจอที่มีคุณสมบัติ ตามที่กำหนดในขั้นตอนออกแบบหน้าจอ และสามารถจัดการกับเพิ่มข้อมูลตามที่กำหนดในขั้นตอนแรก

ความสัมพันธ์ของขั้นตอนทั้งสี่ สามารถแสดงเป็นผังระบบงานได้ ดังรูป 4.1



รูป 4.1 ตั้งระบบงานแสดงความสัมพันธ์ของระบบสร้างแม่ข้อมูลกรามิเตอร์ สำหรับการพัฒนาโปรแกรม

โครงสร้างของข้อมูล (Data Structure)

โครงสร้างของข้อมูลที่ใช้ในระบบ ได้จัดแบ่งเป็นกลุ่มใหญ่ตามความเหมาะสมในการนำไปใช้ ได้แก่

1. โครงสร้างของข้อมูลสำหรับส่วนจัดการกับหน้าจอ
ส่วนจัดการกับหน้าจอนี้มี 2 ลักษณะ คือ

1.1 ส่วนที่มีลักษณะการทำงานที่เป็นการบรรณาธิกรทั้งหน้าจอ (Full Screen Editor) คือผู้ใช้สามารถเลื่อนเคอร์เซอร์ไปได้ทั่วทั้งจอภาพเพื่อกำหนด หรือเปลี่ยนแปลงแก้ไขตำแหน่งและเนื้อหาของข้อความ หรือเพื่อกำหนดตำแหน่งของเขตข้อมูล

โครงสร้างข้อมูลที่เหมาะสมกับงานลักษณะนี้ คือ โครงสร้างแบบ Array of Structure ดังต่อไปนี้

```

struct scr_line
{
    char data[82] ;          /* สำหรับเก็บข้อความที่จะปรากฏบนหน้าจอ */
    char attr[82] ;        /* สำหรับเก็บ Attribute ที่จะปรากฏบน
                           หน้าจอ */
    char fld_num[82] ;     /* สำหรับเก็บตำแหน่งและลำดับที่ของเขต
                           ข้อมูลบนจอภาพ */
};

struct scr_line scr_buff[25] ;
  
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

1.2 ส่วนที่มีลักษณะเป็นกรอบรายการย่อย (window) คือมีลักษณะเป็น ปุ่มขึ้นและปุ่มลง ผู้ใช้สามารถเลือกรายการโดยการเลื่อนแถบสว่าง โดยใช้เป็นลูกศร ขึ้นและลง แล้วกดแป้น ENTER หรือโดยการกดแป้นเสือกคีย์

ซึ่งโครงสร้างข้อมูลที่เหมาะสมกับงานลักษณะนี้ คือ โครงสร้างแบบ Structure ดังต่อไปนี้

```
struct window
{
    int row ;                /* แถวแรกของกรอบ */
    int col ;                /* คอลัมน์แรกของกรอบ */
    int row_size ;          /* ขนาดความยาวของกรอบ */
    int col_size ;          /* ขนาดความกว้างของกรอบ */

    struct frame *wscr ;    /* ตัวชี้ไปยังรูปแบบของกรอบ */
    char *clrscr ;          /* ตัวชี้ไปยังพื้นที่เก็บส่วนของ
                             หน้าจอที่ถูกกรอบซ้อนอยู่ */
}

struct frame
{
    int hotkey ;            /* ตำแหน่งตัวอักษรที่เป็น hotkey */

    char content [80] ;     /* เนื้อหา 1 บรรทัดของกรอบ */
}
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

2. โครงสร้างของข้อมูลสำหรับส่วนจัดการกับข้อมูลที่ได้รับจากกรอบรายการย่อย เนื่องจากข้อมูลที่ได้รับหรือแสดงในกรอบรายการย่อยโดยวิธีการป้อนข้อมูล หรือ บุคลากร มีเป็นจำนวนมาก บางครั้งต้องการข้อมูลทั้งหมดของเขตข้อมูลหนึ่ง ๆ แต่บางครั้งก็ต้องการข้อมูลในเรื่องใดเรื่องหนึ่งของเขตข้อมูลทั้งหมด นั่นคือมีการดึงข้อมูลทั้งในแนวดิ่ง (Vertical) และแนวนอน (Horizontal)

ดังนั้น โครงสร้างของข้อมูลที่เหมาะสมสำหรับงานลักษณะแบบนี้ คือ การเก็บเป็นลักษณะของตาราง (table) ของแต่ละเรื่อง เช่น ตารางของชื่อเขตข้อมูล ตารางของค่าแถวของเขตข้อมูล หรือตารางของค่าคอลัมน์ของเขตข้อมูล เป็นต้น ซึ่งสมาชิกที่ลำดับเดียวกันของแต่ละตารางจะเป็นข้อมูลในแต่ละเรื่องของเขตข้อมูล ณ ลำดับนั้น

เขตข้อมูลที่ 1	NAME	10	20	...	
เขตข้อมูลที่ 2					
เขตข้อมูลที่ 3					
.
.
เขตข้อมูลที่ n					

ตารางของ ชื่อเขตข้อมูล ตารางของ ค่าแถวของ เขตข้อมูล ตารางของ ค่าคอลัมน์ของ เขตข้อมูล

จากตัวอย่างข้างต้น จะได้ว่าเขตข้อมูลที่ 1 มีชื่อว่า NAME มีค่าแถวเป็น 10 และค่าคอลัมน์เป็น 20 เป็นต้น และเมื่อต้องการทราบชื่อเขตข้อมูลทั้งหมดก็ดูได้จากตารางของชื่อเขตข้อมูลได้โดยตรงเช่นกัน

โดยวิธีการเช่นเดียวกันนี้ ได้แปงตารางออกเป็น 2 กลุ่มใหญ่ คือ



2.1 กลุ่มของตารางที่เกี่ยวข้องกับหน้าจอ (Screen Variable Table) ซึ่งแบ่งย่อยได้อีกเป็น

2.1.1 ตารางที่เกี่ยวข้องกับ เขตข้อมูลและการรับข้อมูลบนหน้าจอ (Field Definition Table) ประกอบด้วย

- fld_row_tbl
เป็นตารางของตัวเลข เก็บค่าแถวของ เขตข้อมูลบนหน้าจอ
- fld_col_tbl
เป็นตารางของตัวเลข เก็บค่าคอลัมน์ของ เขตข้อมูลบนหน้าจอ
- fld_name_tbl
เป็นตารางของตัวอักษร ที่มีความยาวตามที่กำหนดใน FIELD_NAME_SIZE เก็บชื่อของเขตข้อมูลบนหน้าจอ
- file_flg_tbl
เป็นตารางของตัวเลข เก็บ flag ที่แสดงว่าชื่อของเขตข้อมูลบนหน้าจอ เป็นชื่อที่นำมาจากแก้ไขข้อมูล หรือเป็นชื่อที่ไม่ได้นำมาจากแก้ไขข้อมูล
- file_fld_no_tbl
เป็นตารางของตัวเลข เก็บลำดับที่ของเขตข้อมูลในแก้ไขข้อมูล (กรณี que เขตข้อมูลนั้นนำมาจากแก้ไขข้อมูล)
- file_data_tbl
เป็นตารางของตัวเลข เก็บลำดับที่ของแก้ไขข้อมูลในตารางชื่อแก้ไขข้อมูล
- i_o_tbl
เป็นตารางของตัวเลข เก็บประเภทของ เขตข้อมูลบนหน้าจอว่าเป็น เขตข้อมูลสำหรับรับข้อมูลเข้า แสดงข้อมูล หรือทั้งรับและแสดงข้อมูล
- aft_flg_tbl
เป็นตารางของตัวเลข เก็บเงื่อนไขของการแสดงข้อมูลของ เขตข้อมูลบนหน้าจอ

- aft_fld_tbl
เป็นตารางของตัวเลข เก็บลำดับที่ของเขตข้อมูลบนหน้าจอที่เขตข้อมูลนี้จะถูกแสดงหลังจากเขตข้อมูลนั้น
- aft_rtn_tbl
เป็นตารางของตัวอักษร ที่มีความยาวตามที่กำหนดใน USR_ROUTINE-SIZE เก็บชื่อของ routine ที่จะถูกประมวลผลก่อนที่จะแสดงเขตข้อมูลนี้
- fld_len_tbl
เป็นตารางของตัวเลข เก็บความยาวของเขตข้อมูลบนหน้าจอ
- edit_typ_tbl
เป็นตารางของตัวเลข เก็บรูปแบบการบรรณาธิกรณของเขตข้อมูลบนหน้าจอ
- edit_attr_tbl
เป็นตารางของตัวเลข เก็บลักษณะการรับข้อมูลของเขตข้อมูลบนหน้าจอ

2.1.2 ตารางที่เกี่ยวข้องกับการตรวจเช็คข้อมูลของเขตข้อมูลบนหน้าจอ (Check Condition Table) ประกอบด้วย

- chk_flg_tbl
เป็นตารางของตัวเลข เก็บ flag ที่แสดงว่าจะทำการตรวจเช็คข้อมูลในเขตข้อมูลโดยวิธีใด
- upper_flg_tbl
เป็นตารางของตัวเลข เก็บ flag ที่แสดงว่าเขตข้อมูลนี้มีการกำหนดขอบเขตบนสำหรับการตรวจเช็คช่วงหรือไม่
- upper_lit_tbl
เป็นตารางของตัวอักษรที่มีขนาดตามที่กำหนดใน CBL_LIT_SIZE ใช้เก็บค่าที่กำหนดให้เป็นขอบเขตบนสำหรับการตรวจเช็คช่วง
- lower_flg_tbl
เป็นตารางของตัวเลข เก็บ flag ที่แสดงว่าเขตข้อมูลนี้มีการกำหนดขอบเขตล่างสำหรับการตรวจเช็คช่วงหรือไม่

- lower_lit_tbl
เป็นตารางของตัวอักษร ที่มีขนาดตามที่กำหนดใน
CBL_LIT_SIZE ใช้เก็บค่าที่กำหนดให้เป็นขอบเขตล่างสำหรับการตรวจเช็คช่วง

- in_file_flg_tbl
เป็นตารางของตัวเลข เก็บ flag ที่แสดงว่าในกรณี
ที่ต้องการตรวจเช็คข้อมูลที่รับเข้ามา กับข้อมูลใหม่เพิ่มข้อมูล จะทำการตรวจเช็คว่ามีข้อมูลต้องมี
ใหม่เพิ่มข้อมูล หรือข้อมูลต้องไม่มีใหม่เพิ่มข้อมูล

- file_num_tbl
เป็นตารางของตัวเลข เก็บลำดับที่ในตารางเพิ่มข้อมูล
ของเพิ่มข้อมูลที่จะทำการตรวจเช็ค

- chk_cond_tbl
เป็นตารางของตัวเลข เก็บเงื่อนไขที่จะทำการตรวจ
เช็คข้อมูลที่รับเข้ามา กับข้อมูลใหม่เพิ่มข้อมูล

- usr_chk_rtn_tbl
เป็นตารางของตัวอักษร ที่มีขนาดตามที่กำหนดใน
USR_ROUTINE_SIZE ใช้เก็บชื่อของ routine ที่ทำการตรวจเช็คข้อมูลที่ใส่เข้ามา

2.1.3 ตารางที่เกี่ยวข้องกับสิ่งที่จะทำภายหลังการตรวจเช็คข้อมูล
ของเขตข้อมูลบนหน้าจอแล้ว (After check Condition Table) ประกอบด้วย

- aft_chk_flg_tbl
เป็นตารางของตัวเลข เก็บ flag ที่แสดงว่าสิ่งที่
จะทำ ภายหลังจากการตรวจเช็คข้อมูลแล้ว

- next_prog_tbl
เป็นตารางของตัวอักษร ที่มีความยาวตามที่กำหนดใน
SYS_NAME_LEN ใช้เก็บชื่อของ โปรแกรมต่อไปที่จะถูกเรียกหลังจากการตรวจเช็คข้อมูลแล้ว

- usr_chk_rtn_tbl
เป็นตารางของตัวอักษร ที่มีความยาวตามที่กำหนดใน
USR_ROUTINE_SIZE ใช้เก็บชื่อของ Routine ที่จะถูกทำหลังจากการตรวจเช็คข้อมูลแล้ว

2.2 กลุ่มของตารางที่เกี่ยวข้องกับแฟ้มของข้อมูล (Data File Variable Table) ประกอบด้วย

2.2.1 file_name_tbl

เป็นตารางของตัวอักษร ที่มีความยาวตามที่กำหนดใน FILE_NAME_SIZE ใช้เก็บรายชื่อของแฟ้มของข้อมูลทั้งหมดที่เกี่ยวข้องในระบบ

2.2.2 fheader_tbl

เป็นตารางของโครงสร้าง ที่เก็บข้อมูลของแต่ละแฟ้มของข้อมูล ดังต่อไปนี้

- ffile_type
คือ โครงสร้างของแฟ้มข้อมูล
- ffld_size
คือ จำนวนเขตข้อมูลที่มีได้สูงสุดใน 1 ระเบียบข้อมูล
- fkey_size
คือ จำนวน key สูงสุดที่มีได้สำหรับแฟ้มข้อมูล

2.2.3 fdetail_tbl

เป็นตารางของโครงสร้างที่เก็บข้อมูลของแต่ละแฟ้มของข้อมูล ดังต่อไปนี้

- #ffld_name
คือ ตัวชี้ (pointer) ไปยังตารางของตัวอักษร ที่มีความยาวตามที่กำหนดใน FIELD_NAME_SIZE ซึ่งเก็บชื่อของเขตข้อมูลทั้งหมดในระเบียบข้อมูลของแฟ้มของข้อมูล โดยเก็บเรียงลำดับตามลำดับของเขตข้อมูลในแฟ้มข้อมูล
- #ffld_type
คือ ตัวชี้ ไปยังตารางของตัวอักษร ซึ่งเก็บชนิดของเขตข้อมูลนั้น ๆ ว่าเป็นตัวเลข หรือตัวอักษรบนตัวอักษร
- #ffld_len
คือ ตัวชี้ ไปยังตารางของตัวเลข ซึ่งเก็บความยาวของแต่ละเขตข้อมูล
- #ffld_sign
คือ ตัวชี้ ไปยังตารางของตัวอักษรที่แสดงว่า เขตข้อมูลนั้น ๆ เก็บค่าโดยคำึงถึงเครื่องหมายด้วยหรือไม่

- *ffld_dec

คือ ตัวชี้ไปยังตารางของตัวเลขที่เก็บจำนวนตำแหน่งของทศนิยม ที่จะมีได้สำหรับเขตข้อมูลนั้น ๆ

- *ffld_key

คือ ตัวชี้ไปยังตารางของตัวเลขที่เก็บค่าที่แสดงว่าเขตข้อมูลนั้นเป็น key หรือไม่ และถ้าเป็น key จะเป็น key ตัวที่เท่าใด

- *ffld_key_dup

คือ ตัวชี้ไปยังตารางของตัวอักษรที่แสดงว่า เขตข้อมูลที่เป็น key นั้นเป็น key ที่ซ้ำ (duplicate) ได้หรือไม่



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

โครงสร้างแฟ้มข้อมูล (File Structure)

แฟ้มข้อมูลทั้งหมดของระบบ มีดังต่อไปนี้

1. แฟ้มข้อมูลพารามิเตอร์ (Parameter File) ประกอบด้วย

1.1 แฟ้มข้อมูลพารามิเตอร์สำหรับหน้าจอ (Screen Parameter File)

คือแฟ้มข้อมูลที่เก็บรายละเอียดทั้งหมดของหน้าจอ ซึ่งเกิดขึ้นใหม่เมื่อตอนการออกแบบหน้าจอ

1.1.1 ชื่อแฟ้มข้อมูล

กำหนดชื่อโดยยึดหลักมาตรฐานการตั้งชื่อ ของโปรแกรมควบคุมระบบ MS-DOS กล่าวคือ ชื่อแฟ้มข้อมูลมีความยาวได้สูงสุด 8 ตัวอักษร และระบบจะเพิ่ม extension ว่า .INF โดยอัตโนมัติ

1.1.2 องค์ประกอบของแฟ้มข้อมูล

1.1.2.1 จำนวนเขตข้อมูลของหน้าจอ (Number of screen fields)

1.1.2.2 ลักษณะของโปรแกรมที่จะถูกสร้างขึ้นเป็นการกำหนดว่าโปรแกรมที่จะถูกสร้างขึ้นนั้น เป็นโปรแกรมหลัก หรือ โปรแกรมย่อย

ถ้ามีค่าเป็น 0 (หรือ MAIN_PROGRAM_STYLE) แสดงว่าเป็นโปรแกรมหลัก

ถ้ามีค่าเป็น 1 (หรือ SUB_PROGRAM_STYLE) แสดงว่าเป็นโปรแกรมย่อย

1.1.2.3 ลักษณะที่ปรากฏและตำแหน่งของเขตข้อมูล (Attribute and Field Image) ในส่วนที่มีขนาด 25X80 ตัวอักษร เป็นเนื้อหาที่เก็บลักษณะที่ปรากฏของข้อความหรือเขตข้อมูล (Attribute Image)

1.1.2.4 รายละเอียดของเขตข้อมูล (Field Definition) เป็นส่วนที่เก็บรายละเอียดของเขตข้อมูลแต่ละเขต โดยมีความยาวเท่ากัน แต่มีจำนวนตามจำนวนเขตข้อมูลของหน้าจอ ประกอบด้วย

- ตำแหน่งแถวของเขตข้อมูลบนหน้าจอ
- ตำแหน่งคอลัมน์ของเขตข้อมูลบนหน้าจอ

จากเพิ่มข้อมูล

ได้มาจากเพิ่มข้อมูล

จากเพิ่มข้อมูลหรือไม่ได้มาจากเพิ่มข้อมูล

แสดงว่าเป็นเขตข้อมูลที่ได้มาจากเพิ่มข้อมูล

แสดงว่าเป็นเขตข้อมูลที่ไม่ได้มาจากเพิ่มข้อมูล

การกำหนดว่าเขตข้อมูลบนหน้าจอนี้เป็นเขตข้อมูลสำหรับรับข้อมูลเข้า (input) แสดงข้อมูล (output) หรือทั้งรับและแสดงข้อมูล (Input/Output)

แสดงว่าเป็นเขตข้อมูลสำหรับนำข้อมูลเข้า

แสดงว่าเป็นเขตข้อมูลสำหรับแสดงข้อมูล

แสดงว่าเป็นเขตข้อมูลสำหรับทั้งนำข้อมูลเข้าและแสดงข้อมูล

ถูกแสดงบนหน้าจอ หลังจากทำการประมวลผลเขตข้อมูลอื่น หรือทำการประมวลผล routine อื่นแล้วหรือไม่

แสดงว่า จะไม่มีการแสดงค่าเขตข้อมูลนี้

แสดงว่า จะทำการแสดงค่าเขตข้อมูลนี้หลังจากทำการประมวลผลเขตข้อมูลที่กำหนดแล้ว

แสดงว่า จะทำการแสดงค่าเขตข้อมูลนี้หลังจากทำการประมวลผล routine ที่กำหนดแล้ว

- ชื่อเขตข้อมูลบนหน้าจอ กรณีที่เป็นชื่อที่มา

- ชื่อเขตข้อมูลบนจอภาพ กรณีที่เป็นชื่อที่ไม่

- flag แสดงว่าชื่อเขตข้อมูลเป็นชื่อที่มา

ถ้ามีค่าเป็น 0 (หรือ DATA_FILE)

ถ้ามีค่าเป็น 1 (หรือ NON_FILE)

- ชื่อเพิ่มข้อมูลที่นำเขตข้อมูลมา

- ลำดับที่ของชื่อเขตข้อมูลในเพิ่มข้อมูล

- ประเภทของเขตข้อมูลบนหน้าจอ เป็น

ถ้ามีค่าเป็น 0 (หรือ INPUT_FLD)

ถ้ามีค่าเป็น 1 (หรือ OUTPUT_FLD)

ถ้ามีค่าเป็น 2 (หรือ IO_FLD)

- flag แสดงว่าเขตข้อมูลบนหน้าจอนี้จะ

ถ้ามีค่าเป็น 0 (หรือ NO_DSPLY)

ถ้ามีค่าเป็น 1 (หรือ AFTER_FLD)

ถ้ามีค่าเป็น 2 (หรือ AFTER_RTN)



ผู้ที่จะถูกแสดงหลังจาก

ที่จะทำการแสดงเขตข้อมูลนี้

จุลภาคด้วย

หน้าจอ เป็นแบบ Zero suppress and Comma หรือ Zero suppress หรือ Non zero suppress หรือ Alphanumeric

ถ้ามีค่าเป็น 0 (หรือ ZERO_SUP_COM) แสดงว่า จะทำการบรรณาธิกรณ เขตข้อมูลนี้แบบ Zero suppress โดยมีจุลภาคคั่นทุก ๆ ตัวเลข 3 หลัก

ถ้ามีค่าเป็น 1 (หรือ ZERO_SUP) แสดงว่า จะทำการบรรณาธิกรณ เขตข้อมูลนี้แบบ Zero suppress

ถ้ามีค่าเป็น 2 (หรือ NON_ZERO_SUP) แสดงว่า จะทำการบรรณาธิกรณ เขตข้อมูลนี้แบบ Non zero suppress

ถ้ามีค่าเป็น 3 (หรือ ALPHANUMERIC) แสดงว่า จะทำการบรรณาธิกรณ เขตข้อมูลนี้แบบ Alphanumeric

ได้แก่

REQUIRED เมื่อต้องการให้ เป็น เขตข้อมูลที่จะต้องถูกใส่ข้อมูล ถ้าไม่ใส่ข้อมูลจะไม่สามารถผ่านเขตข้อมูลนี้ไปได้

AUTO SKIP เมื่อต้องการให้ เป็น เขตข้อมูลทีเมื่อใส่ข้อมูลแล้วจะข้ามไป เขตข้อมูลต่อไปโดยอัตโนมัติ

SECURE เมื่อต้องการให้ เป็น เขตข้อมูลที่จะ ไม่แสดงข้อมูลที่ใส่เข้าไปให้เห็น

BELL เมื่อต้องการให้ เป็น เขตข้อมูลทีเมื่อใส่ข้อมูลแล้วจะมีเสียงดังขึ้น

- flag แสดงวิธีการตรวจเช็คข้อมูลว่าจะเป็นการตรวจเช็คช่วง (Range check) ตรวจเช็คกับแฟ้มข้อมูล (File check) หรือตรวจเช็คด้วย routine ที่สร้างขึ้นต่างหาก (User routine)

ถ้ามีค่าเป็น 0 (หรือ NO_CHECK)

แสดงว่า จะไม่มีการตรวจเช็คข้อมูลของเขตข้อมูลนี้

ถ้ามีค่าเป็น 1 (หรือ USER_CHECK_RTN)

แสดงว่าจะทำการตรวจเช็คข้อมูลของเขตข้อมูลนี้โดยใช้ routine ที่เขียนขึ้นต่างหาก

ถ้ามีค่าเป็น 2 (หรือ RANGE_CHECK)

แสดงว่าจะทำการตรวจเช็คข้อมูลของเขตข้อมูลนี้โดยวิธีเช็คช่วง

ถ้ามีค่าเป็น 3 (หรือ FILE_CHECK)

แสดงว่า จะทำการตรวจเช็คข้อมูลของเขตข้อมูลนี้โดยการตรวจเช็คกับข้อมูลในแฟ้มข้อมูล

- flag แสดงว่ามีการกำหนดขอบเขตบน

สำหรับการตรวจเช็คช่วงหรือไม่

- ค่าขอบเขตบนสำหรับการเช็คช่วง

- flag แสดงว่ามีการกำหนดขอบเขตล่าง

สำหรับการตรวจเช็คช่วงหรือไม่

- ค่าขอบเขตล่าง สำหรับการเช็คช่วง

- flag แสดงว่าจะทำการเช็คข้อมูลว่า

ต้องมีในแฟ้มข้อมูล หรือต้องไม่มีในแฟ้มข้อมูล

ถ้ามีค่าเป็น 0 (หรือ IN_FILE_CHECK)

แสดงว่า จะทำการตรวจเช็คข้อมูลของเขตข้อมูลนี้โดยการตรวจเช็คว่าต้องมีในแฟ้มข้อมูล

ถ้ามีค่าเป็น 1 (หรือ OUT_FILE_CHECK)

แสดงว่า จะทำการตรวจเช็คข้อมูลของเขตข้อมูลนี้ โดยการตรวจเช็คว่าต้องไม่มีในแฟ้มข้อมูล

- ลำดับของแฟ้มข้อมูลที่จะทำการตรวจเช็ค

ในตารางรายชื่อแฟ้มข้อมูล

- เงื่อนไขที่จะทำการตรวจเช็คข้อมูลที่ใส่

เข้ามากับข้อมูลในแฟ้มข้อมูล ซึ่งได้แก่เท่ากับ (Equal) มากกว่าหรือเท่ากับ (Greater equal) หรือมากกว่า (greater Than)

- ถ้ามีค่าเป็น 0 (หรือ EQ_CHK)
แสดงว่า ให้ เช็คว่า เท่ากับข้อมูลในแฟ้มข้อมูล
- ถ้ามีค่าเป็น 1 (หรือ GE_CHK)
แสดงว่า ให้ เช็คว่ามากกว่าหรือเท่ากับข้อมูลในแฟ้มข้อมูล
- ถ้ามีค่าเป็น 2 (หรือ GT_CHK)
แสดงว่า ให้ เช็คว่ามากกว่าข้อมูลในแฟ้มข้อมูล
- ชื่อ routine สำหรับตรวจเช็ค ที่สร้าง
ขึ้นต่างหาก
- flag บอกว่าหลังจากตรวจเช็คข้อมูลแล้ว
จะทำอะไรสามารถกำหนดได้ ดังต่อไปนี้
 - เป็น 0 (หรือ NOTHING) เมื่อต้องการ
ให้ผ่านไป เขตข้อมูลต่อไปเลย
 - เป็น 1 (หรือ DISPLAY_DATA) เมื่อ
ต้องการทำการแสดงข้อมูลใน เขตข้อมูลที่กำหนด ให้แสดงหลังจากเขตข้อมูลนี้
 - เป็น 2 (หรือ SAVE_DATA) เมื่อต้อง
ทำการเก็บข้อมูลลงแฟ้มข้อมูล
 - เป็น 3 (หรือ DELETE_DATA) เมื่อ
ต้องการทำการลบข้อมูลออกจากแฟ้มข้อมูล
 - เป็น 4 (หรือ NEXT_PROGRAM) เมื่อ
ต้องการทำการแสดงหน้าจอถัดไป
 - เป็น 5 (หรือ END_PROGRAM) เมื่อ
ต้องการจบโปรแกรม
 - เป็น 6 (หรือ USER_ROUTINE) เมื่อ
ต้องการทำการประมวลผล routine ที่สร้างขึ้นต่างหาก
- ชื่อ user routine ที่จะทำหลังจากทำ
การตรวจเช็คข้อมูลแล้ว
 - ชื่อ โปรแกรมของหน้าจอถัดไปที่จะทำหลัง
จากการตรวจเช็คข้อมูลแล้ว

1.2 แฟ้มข้อมูลพารามิเตอร์สำหรับแฟ้มข้อมูล (File Parameter File)

คือ แฟ้มข้อมูลที่เก็บรายละเอียดของแฟ้มข้อมูลที่เก็บข้อมูล (Data File)

1.2.1 ชื่อแฟ้มข้อมูล

กำหนดชื่อโดยยึดหลักมาตรฐานการตั้งชื่อของ โปรแกรม
ควบคุมระบบ MS-DOS กล่าวคือ ชื่อแฟ้มข้อมูลมีความยาวได้สูงสุด 8 ตัวอักษร และระบบ
จะเพิ่ม extension ว่า .FIL ให้โดยอัตโนมัติ

1.2.2 องค์ประกอบของแฟ้มข้อมูล

1.2.2.1 ระเบียบความคุม (File Header Record)

ประกอบด้วย

- โครงสร้างของแฟ้มข้อมูล เป็นแบบ
Sequential File, Relative File หรือ Indexed File

- จำนวนเขตข้อมูลในแฟ้มข้อมูล

- จำนวน key ในแฟ้มข้อมูล

ถ้าเป็นแฟ้มข้อมูลแบบเรียงลำดับ จำนวน
key ควรมีค่าเป็น 0

ถ้าเป็นแฟ้มข้อมูลแบบสัมพันธ์ จำนวน
key ควรมีค่าเป็น 1

ถ้าเป็นแฟ้มข้อมูลแบบเรียงลำดับที่มีตัวชี้
จำนวน key ควรมีค่ามากกว่า 0

1.2.2.2 รายละเอียดของแต่ละเขตข้อมูล (Field Detail Record) มีจำนวนตามจำนวนเขตข้อมูลที่ระบุในระเบียบความคุม ประกอบด้วย

- ชื่อเขตข้อมูล กำหนดชื่อโดยยึดหลัก
มาตรฐานการตั้งชื่อตัวแปรสำหรับภาษาโคบอล

- ชนิดของเขตข้อมูล เป็นชนิดตัวเลข
(Numeric) หรือตัวเลขปนตัวอักษร (Alphanumeric)

- ความยาวของเขตข้อมูล

- flag แสดงว่าเขตข้อมูลนี้มีการคิดเครื่อง
หมายด้วยหรือไม่

ข้อมูลที่เป็นตัวเลข

- จำนวนตำแหน่งของทศนิยม สำหรับเขต

ไม่ และถ้าเป็น key จะเป็น key ตัวที่เท่าใด

- ตัวเลขแสดงว่าเขตข้อมูลนี้เป็น key หรือ

key

ถ้ามีค่าเป็น 1 แสดงว่าเป็น Primary

Alternate key

- ถ้ามีค่าเป็น 2 ขึ้นไป แสดงว่าเป็น

จะสามารถมี key ซ้ำ (Duplicate key) ได้หรือไม่

- flag แสดงว่ากรณีที่เขตข้อมูลนี้เป็น Key

ไม่ได้

ถ้าเป็น Primary key จะมี key ซ้ำ

1.3 เพิ่มข้อมูลของข้อความที่จะปรากฏบนหน้าจอ (Screen Image File)

คือ เพิ่มข้อมูลที่เก็บส่วนที่เป็นข้อความที่ปรากฏบนหน้าจอ

1.3.1 ชื่อเพิ่มข้อมูล

กำหนดชื่อตามชื่อเพิ่มข้อมูลพารามิเตอร์สำหรับหน้าจอ โดยระบบจะเพิ่ม extension ว่า .SCR ให้โดยอัตโนมัติ

1.3.2 องค์ประกอบของเพิ่มข้อมูล

เป็นเพิ่มข้อมูลที่เก็บข้อความที่จะปรากฏบนจอภาพ มีขนาด 25x80 ตัวอักษร

เพิ่มข้อมูลนี้สามารถสร้างโดยโปรแกรมบรรณาธิการอื่น ๆ แล้วตั้งชื่อตามชื่อของเพิ่มข้อมูลพารามิเตอร์สำหรับหน้าจอ โดยมี extension เป็น .SCR

1.4 แฟ้มข้อมูลของสภาพแวดล้อมของระบบ (Environment File)

1.4.1 ชื่อแฟ้มข้อมูล

มีชื่อว่า ENVIRON.PG

1.4.2 องค์ประกอบของแฟ้มข้อมูล

เป็นแฟ้มข้อมูลที่เก็บรายละเอียด ดังต่อไปนี้

- จำนวนแฟ้มข้อมูลสูงสุดที่มีได้ในระบบ
- จำนวนเขตข้อมูลสูงสุดที่มีได้บนจอภาพ
- ไตเรคทอรีของแฟ้มข้อมูลพารามิเตอร์ของแฟ้มของข้อมูล
- ไตเรคทอรีของโปรแกรมภาษาโคบอลที่จะถูกสร้างขึ้น
- ไตเรคทอรีของแฟ้มข้อมูลของข้อมูลที่จะใช้ในระบบ

2. แฟ้มข้อมูลของโปรแกรมภาษาโคบอลที่สร้างขึ้น (Source Program File)

2.1 ชื่อแฟ้มข้อมูล

กำหนดชื่อตามชื่อแฟ้มข้อมูลพารามิเตอร์สำหรับหน้าจอดีระบบจะ
เพิ่ม extension ว่า .COB ให้โดยอัตโนมัติ

2.2 องค์ประกอบของแฟ้มข้อมูล

ประกอบด้วยโปรแกรมภาษาโคบอลที่สร้างขึ้น

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

องค์ประกอบของระบบสร้างแฟ้มข้อมูลพารามิเตอร์สำหรับการพัฒนาโปรแกรม

1. ระบบสร้างแฟ้มข้อมูลพารามิเตอร์สำหรับการพัฒนาโปรแกรม ได้จัดแบ่งโปรแกรมออกเป็นกลุ่มย่อย ๆ ดังต่อไปนี้

1.1 pg เป็นโปรแกรมหลักที่ทำหน้าที่

1.1.1 ควบคุมการจัดสรรแฟ้มข้อมูลพารามิเตอร์สำหรับหน้าจอ

1.1.2 ควบคุมการกำหนดเขตข้อมูล

1.1.3 ควบคุมการกำหนดและแสดงลักษณะที่ปรากฏบนหน้าจอ

(Attribute)

1.1.4 ควบคุมการสร้างโปรแกรมภาษาโคบอล

1.1.5 ควบคุมการกำหนดสภาพแวดล้อมของระบบ

ในการทำหน้าที่ต่าง ๆ เหล่านี้ จะมีการเรียกใช้ function ต่าง ๆ มากมาย จึงได้จัด function เป็นกลุ่ม ๆ แยกตามหน้าที่ ดังต่อไปนี้

- basicio

- wduty

- window

- editscr

- data

- gen

ซึ่งจะกล่าวถึงรายละเอียดต่อไป

1.2 basicio เป็น utility ที่ทำหน้าที่

1.2.1 จัดการเกี่ยวกับการรับตี๋บอร์ด

1.2.2 ควบคุมตำแหน่งของเคอร์เซอร์ เพื่อให้เคลื่อนที่ในทิศทางขึ้นลงและซ้ายขวาของจอภาพ

1.2.3 ควบคุมการแสดงผลบนหน้าจอ

1.2.4 ควบคุมการลบจอภาพทั้งหมด หรือการลบเพียงบางส่วน
ของจอภาพ

1.3 wduty เป็น utility ที่ทำหน้าที่

1.3.1 ความคุมการแสดงผลของกรอบรายการย่อยบนหน้าจอ

1.3.2 ความคุมการรับข้อมูลจากกรอบรายการย่อย

1.4 window เป็นกลุ่มของ function ที่ทำหน้าที่ควบคุมเกี่ยวกับการ
ป้อนข้อมูลหรือผลดำเนินการย่อยต่างๆ โดยการเรียกใช้ utility ใน wduty

1.5 editscr เป็นกลุ่มของ function ที่ทำหน้าที่ควบคุมเกี่ยวกับการ
บรรณาธิกรณกึ่งหน้าจอ โดยการเรียกใช้ utility ใน basicio

1.6 data เป็นกลุ่มของ function ที่ทำหน้าที่

1.6.1 นำข้อมูลที่รับจากหน้าจอ หรือจากกรอบรายการย่อยลงเก็บ
ในตารางต่าง ๆ หรือนำข้อมูลจากตารางขึ้นมาแสดงบนหน้าจอ

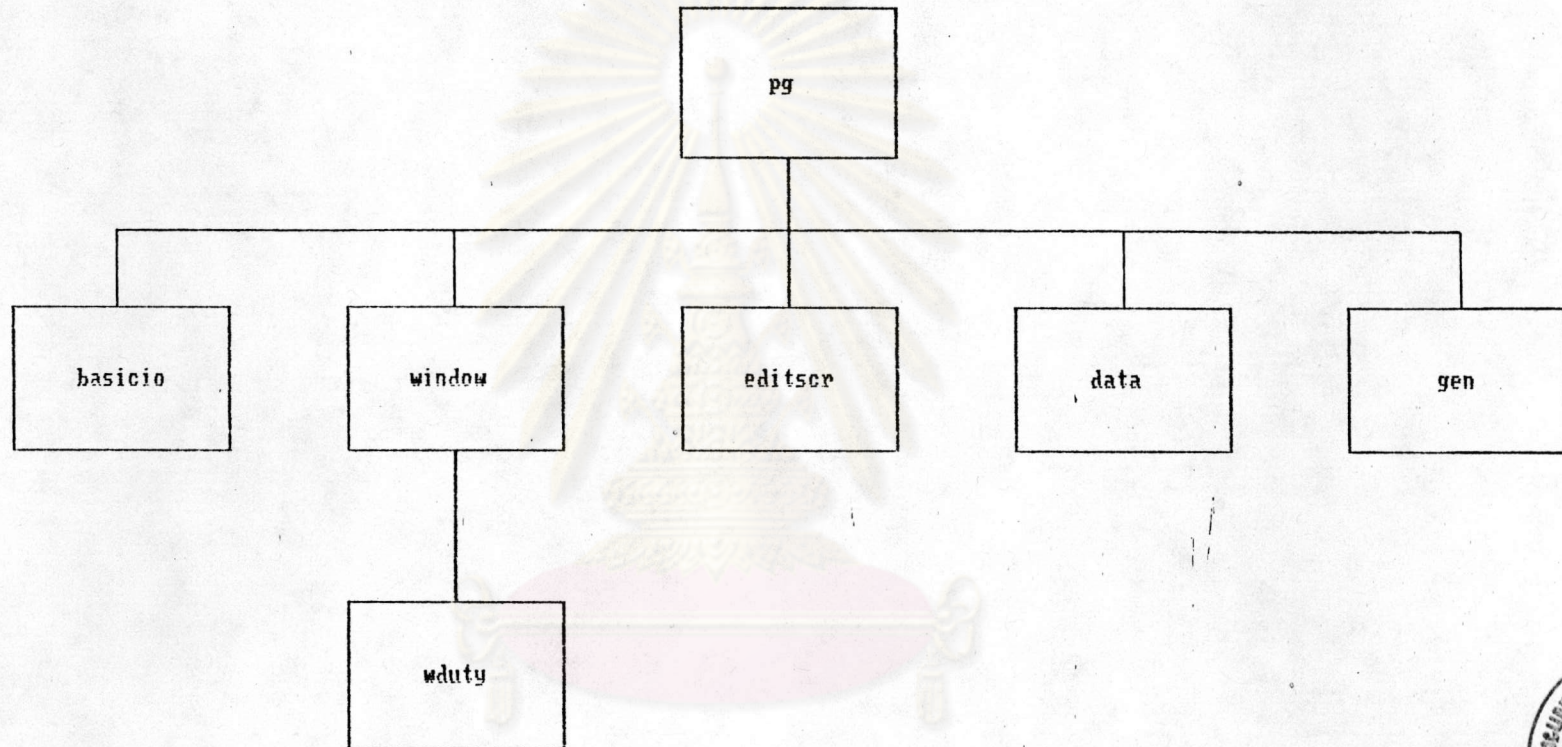
1.6.2 นำข้อมูลจากตารางลงเก็บในแฟ้มข้อมูลพารามิเตอร์ หรือ
นำข้อมูลจากแฟ้มข้อมูลพารามิเตอร์ขึ้นมาไว้ในตารางต่างๆ

1.7 gen เป็น routine ที่ทำหน้าที่

นำข้อมูลต่าง ๆ ที่มีอยู่ไปสร้างโปรแกรมภาษาโคบอล

ความสัมพันธ์ของกลุ่มของ routine สามารถแสดงเป็นผังระบบงานได้ดังรูป 4.2

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



4.2

ผังระบบงานแสดงความสัมพันธ์ของกลุ่มของ routine



2. ระบบสร้างแฟ้มข้อมูลพารามิเตอร์สำหรับแฟ้มข้อมูล file เป็นโปรแกรมที่ทำหน้าที่สร้างหรือแก้ไขแฟ้มข้อมูลพารามิเตอร์สำหรับแฟ้มของข้อมูล

3. ระบบสร้างแฟ้มข้อมูลของสภาพแวดล้อมของระบบ envi เป็นโปรแกรมที่ทำหน้าที่สร้างหรือแก้ไขข้อมูลในแฟ้มข้อมูลของสภาพแวดล้อมของระบบ



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย