計數器設計：  
可以計數定量次數的事件發生後，由計數器產生結果  
   
單純的計數器  
一次攻擊一個動作(每發生一次就計數一次)  
frame 0  
bdy: ...  
frame 20 next: 21  
frame 21 wait: 0 next: 999 hit\_a: # hit\_d: 50  
frame 50  
...  
   
hit\_a: 值決定計數次數  
如果攻擊力為0 ，次數約為 = 500 / hit\_a: #  
   
一般的計數器  
此為考慮事件可能會同時發生的情形而做修正  
設hit\_a: 為1 ，wait: 為0  
取而代之，使用injury: 來代替計數值  
   
frame 0  
bdy: ...  
frame 20 next: 21  
bdy: ...  
frame 21 wait: 0 next: 999 hit\_a: 1 hit\_d: 50  
bdy: ...  
frame 50  
...  
   
動作0, 20, 21 均加入bdy: ，以防漏掉任何計數  
再來是考慮如何定injury: 的值…  
   
計數5 個事件發生時:  
先定最小攻擊力為99  
( 99 + 1 ) \* 5 = 500  
但如果5個事件同時發生  
( 99 \* 5 + 1 ) = 496 < 500 (不會使計數器有動作)  
故最低攻擊力要設為100  
( 100 \* 5 + 1 ) = 501 (5個同時發生)  
( 100 + 1 ) \* 5 = 505 (不同時發生)  
( 100 + 1 ) \* 4 = 404 (4次的時候不會有動作，5次滿才有反應)  
   
考慮20個事件發生時  
同時發生  
( 25 \* 20 + 1 ) = 501  
不同時發生  
( 25 + 1 ) \* 20 = 520  
( 25 + 1 ) \* 19 = 495 (19次時還不會有反應)  
   
考慮25個事件發生時  
同時發生  
( 20 \* 25 + 1 ) = 501  
不同時發生  
( 20 + 1 ) \* 25 = 525  
( 20 + 1 ) \* 24 = 504 (超過500 ，意即24次就有反應，有錯誤)  
故同時發生不可計數超過20個  
不過還是有解決方法，  
那就是一開始就先把計數器的血量加上去，就可以解決這個問題了  
(上式計算為hp 500 的預設情形下)  
   
動態的計數器  
此計數器為 為了節省id而設計，  
也因此捨棄了計數完畢能馬上發生效果的功能  
建議<bmp\_begin>中的撞擊聲刪掉，會很吵…

frame 0 state: 3006 next: 1  
frame 1 state: 3006 next: 2  
opoint: ...  
bdy: ...  
frame 2 state: 3006 wait: 0 next: 1 hit\_a: 1 hit\_d: 50  
bdy: ...  
frame 50  
...  
   
很明顯的，它不使用動作20  
取而代之，在動作1 的時候，造出物件，並攻擊自己-1 hp，與hit\_a: 1 相反作用，以隨時保持固定的hp

起點在1以及41 count( id: 253 type: 3 )

<frame> 1 start1

pic: 999 state: 15 wait: 1 next: 2

<frame\_end>

<frame> 2 s

pic: 999 state: 15 wait: 1 next: 3

opoint:

kind: 1 x: 0 y: 0 action: 25 oid: 253

opoint\_end:

<frame\_end>

<frame> 3 s

pic: 999 state: 3006 wait: 15 next: 4

opoint:

kind: 1 x: 0 y: 0 action: 6 oid: 253

opoint\_end:

itr:

kind: 8 x: -5 y: 3500 w: 10 h: 10 vrest: 10 dvx: 31

itr\_end:

bdy:

kind: 0 x: -5 y: 3200 w: 10 h: 10

bdy\_end:

bdy:

kind: 0 x: -5 y: 3220 w: 10 h: 10

bdy\_end:

<frame\_end>

<frame> 4 s

pic: 999 state: 3006 wait: 0 next: 5 hit\_a: 10 hit\_d: 100

itr:

kind: 8 x: -5 y: 3500 w: 10 h: 10 vrest: 10 dvx: 31

itr\_end:

bdy:

kind: 0 x: -5 y: 3220 w: 10 h: 10

bdy\_end:

<frame\_end>

<frame> 5 s

pic: 999 state: 3006 wait: 15 next: 3

itr:

kind: 8 x: -5 y: 3500 w: 10 h: 10 vrest: 10 dvx: 31

itr\_end:

bdy:

kind: 0 x: -5 y: 3220 w: 10 h: 10

bdy\_end:

<frame\_end>

<frame> 6 revive

pic: 999 state: 15 wait: 0 next: 7

<frame\_end>

<frame> 7 r

pic: 999 state: 18 wait: 0 next: 1000

itr:

kind: 0 x: -5 y: 3200 w: 10 h: 10 vrest: 10 injury: -10 fall: -1

itr\_end:

<frame\_end> #攻擊自己，回復因hit\_a所損的10hp

<frame> 31 dis

pic: 999 state: 9998 wait: 0 next: 1000

<frame\_end>

<frame> 41 start2

pic: 999 state: 15 wait: 1 next: 42

<frame\_end>

<frame> 42 s2

pic: 999 state: 15 wait: 1 next: 43

opoint:

kind: 1 x: 0 y: 0 action: 25 oid: 253

opoint\_end:

<frame\_end>

<frame> 43 s2

pic: 999 state: 3006 wait: 15 next: 44

opoint:

kind: 1 x: 0 y: 0 action: 6 oid: 253

opoint\_end:

itr:

kind: 8 x: -5 y: 3500 w: 10 h: 10 vrest: 10 dvx: 31

itr\_end:

bdy:

kind: 0 x: -5 y: 3200 w: 10 h: 10

bdy\_end:

bdy:

kind: 0 x: -5 y: 3240 w: 10 h: 10

bdy\_end:

<frame\_end>

<frame> 44 s2

pic: 999 state: 3006 wait: 0 next: 45 hit\_a: 10 hit\_d: 102

itr:

kind: 8 x: -5 y: 3500 w: 10 h: 10 vrest: 10 dvx: 31

itr\_end:

bdy:

kind: 0 x: -5 y: 3240 w: 10 h: 10

bdy\_end:

<frame\_end>

<frame> 45 s2

pic: 999 state: 3006 wait: 15 next: 43

itr:

kind: 8 x: -5 y: 3500 w: 10 h: 10 vrest: 10 dvx: 31

itr\_end:

bdy:

kind: 0 x: -5 y: 3240 w: 10 h: 10

bdy\_end:

<frame\_end>

<frame> 50 end

pic: 999 state: 15 wait: 5 next: 51

opoint:

kind: 2 x: 0 y: 3500 action: 396 oid: 30

opoint\_end:

wpoint:

kind: 1 x: 0 y: 3500 weaponact: 396

wpoint\_end:

<frame\_end> #結束，告訴本事件已進行完成，並將相關物件清除

<frame> 51 end

pic: 999 state: 15 wait: 5 next: 1000

wpoint:

kind: 1 x: 0 y: 3500 weaponact: 398

wpoint\_end:

<frame\_end>

<frame> 100 event1

pic: 999 state: 15 wait: 1 next: 101

<frame\_end>

<frame> 101 e1

pic: 999 state: 15 wait: 1 next: 50

opoint:

kind: 1 x: 0 y: 0 action: 0 oid: 998 facing: 2

opoint\_end:

<frame\_end> #come

<frame> 102 event2

pic: 999 state: 15 wait: 1 next: 103

<frame\_end>

<frame> 103 e2

pic: 999 state: 15 wait: 1 next: 50

opoint:

kind: 1 x: 0 y: 0 action: 2 oid: 998 facing: 2

opoint\_end:

<frame\_end> #stay

本來應該是放相應數量的道具(比如任務要2個，就放2個key...)  
本人懶，所以還是設計連續受攻擊來增加計數  
應該有發現到id使用一樣吧...程式碼可以放在一起沒關係 ( id: 253 type: 3 )

<frame> 20 be\_hit

pic: 1 state: 13 wait: 1 next: 21

itr:

kind: 8 x: -5 y: 3500 w: 10 h: 10 vrest: 10 dvx: 31

itr\_end:

<frame\_end>

<frame> 21 h

pic: 1 state: 13 wait: 10 next: 25

opoint:

kind: 1 x: 0 y: 0 action: 26 oid: 253

opoint\_end:

itr:

kind: 8 x: -5 y: 3500 w: 10 h: 10 vrest: 10 dvx: 31

itr\_end:

<frame\_end>

<frame> 25 wait\_be\_hit

pic: 0 state: 13 wait: 30 next: 0

bdy:

kind: 0 x: -20 y: -70 w: 40 h: 70

bdy\_end:

itr:

kind: 8 x: -5 y: 3500 w: 10 h: 10 vrest: 10 dvx: 31

itr\_end:

<frame\_end>

<frame> 26 count\_hit

pic: 999 state: 15 wait: 0 next: 27

<frame\_end>

<frame> 27 count\_hit

pic: 999 state: 18 wait: 0 next: 1000

itr:

kind: 0 x: -5 y: 3220 w: 10 h: 10 vrest: 10 injury: 250 fall: -1

itr\_end:

itr:

kind: 0 x: -5 y: 3240 w: 10 h: 10 vrest: 10 injury: 167 fall: -1

itr\_end:

<frame\_end> #計數2次及計數3次之事件可以以頻道區分

<frame> 30 h

pic: 1 state: 13 wait: 1 next: 21

itr:

kind: 8 x: -5 y: 3500 w: 10 h: 10 vrest: 10 dvx: 31

itr\_end:

<frame\_end>

<frame> 40 h

pic: 1 state: 13 wait: 1 next: 21

itr:

kind: 8 x: -5 y: 3500 w: 10 h: 10 vrest: 10 dvx: 31

itr\_end:

<frame\_end>

上述動作50的地方用了bandit  
因此再附上... bandit( id: 30 type: 0 )

<frame> 396 caught

pic: 999 state: 15 wait: 3 next: 0

bdy:

kind: 0 x: -5 y: 0 w: 10 h: 10

bdy\_end:

<frame\_end>

<frame> 398 dis

pic: 999 state: 9998 wait: 0 next: 1000

<frame\_end>

闖關的wpoint:使用：  
在stage中，除了玩家角色及電腦角色，到下一階段時都會從場上被清掉(非玩家控制且type不是0)  
另外一點就是手持武器也可以帶到下一階段，利用這一點就可在闖關裡做些特效  
   
假如一個物件id 400

frame 0 next: 1  
frame 1 next: 2  
wpoint: kind: 1  
opoint: kind: 2 oid: 400  
frame 2 next: 0  
wpoint: kind: 1 weaponact: 15  
...  
frame 15 next: 999  
wpoint: kind: 2  
   
一開始，一個動作2的物件持著動作15的物件  
紅色的next: 999 指 闖關到達下一階段時，就會跳回動作0 (因為一開始持著它的物件被清掉了)

那就拿我以前做的 假人質 為例吧  
也是說criminal被攻擊後變成敵人，而非我方 criminal( id: 300 type: 5 )

<frame> 70 bandit\_start

pic: 20 state: 0 wait: 2 next: 71 centerx: 38 centery: 78

<frame\_end>

<frame> 71 bandit

pic: 20 state: 0 wait: 2 next: 72 centerx: 38 centery: 78

opoint:

kind: 2 x: 0 y: 1 action: 398 oid: 300

opoint\_end:

wpoint:

kind: 1 x: 0 y: 1 weaponact: 398

wpoint\_end:

<frame\_end>

<frame> 72 bandit

pic: 20 state: 0 wait: 7 next: 73 centerx: 38 centery: 78

bdy:

kind: 1075 x: 21 y: 18 w: 43 h: 62

bdy\_end:

wpoint:

kind: 1 x: 0 y: 1 weaponact: 398

wpoint\_end:

<frame\_end>

<frame> 73 bandit

pic: 21 state: 0 wait: 5 next: 74 centerx: 38 centery: 78

bdy:

kind: 1075 x: 21 y: 18 w: 43 h: 62

bdy\_end:

wpoint:

kind: 1 x: 0 y: 1 weaponact: 398

wpoint\_end:

<frame\_end>

<frame> 74 bandit

pic: 22 state: 0 wait: 10 next: 72 centerx: 38 centery: 78

bdy:

kind: 1075 x: 21 y: 18 w: 43 h: 62

bdy\_end:

wpoint:

kind: 1 x: 0 y: 1 weaponact: 398

wpoint\_end:

<frame\_end> #此為id300，意指被打跳到動作75

<frame> 75 bandit

pic: 20 state: 0 wait: 2 next: 1000 centerx: 38 centery: 78

wpoint:

kind: 1 x: 0 y: 1 weaponact: 76

wpoint\_end:

<frame\_end>

<frame> 76 bandit

sound: data\013.wav

pic: 23 state: 0 wait: 2 next: 77 centerx: 38 centery: 78

<frame\_end>

<frame> 77 bandit

pic: 24 state: 0 wait: 2 next: 78 centerx: 38 centery: 78

<frame\_end>

<frame> 78 bandit

pic: 24 state: 8030 wait: 2 next: 999 centerx: 38 centery: 78

<frame\_end>

<frame> 398 dis

pic: 999 state: 15 wait: 1 next: 1000 centery: 78

<frame\_end>

動作71的地方先造出手持物件，weaponact指定398是因為若沒啟動，*"遺落"*的物件直接next1000消失  
若受攻擊跳至75，會**遺落**動作76的物件，之後8030變身  
因為手持物件並沒有**受到攻擊**，因此**隊伍**還是闖關的敵人