

---

# Berkeley DB for TinyloT

---

Sejong Univ.

Name : Park Minji

E-mail : [iorw0224@gmail.com](mailto:iorw0224@gmail.com)



## 함수 수정

C : Store\_CSE, Store\_AE, Store\_CNT, Store\_CIN, Store\_Sub  
R : Get\_CSE, Get\_AE, Get\_CNT, Get\_CIN, Get\_Sub  
U : Update\_CSE, Update\_AE, Update\_CNT, Update\_CIN, Update\_Sub  
D : Delete\_CSE, Delete\_AE, Delete\_CNT, Delete\_CIN, Delete\_Sub

### 1. 구조체명 변경

SUB -> Sub

### 2. 함수 이름 변경

Subscription -> Store\_Sub

Sub\_Update -> Update\_Sub

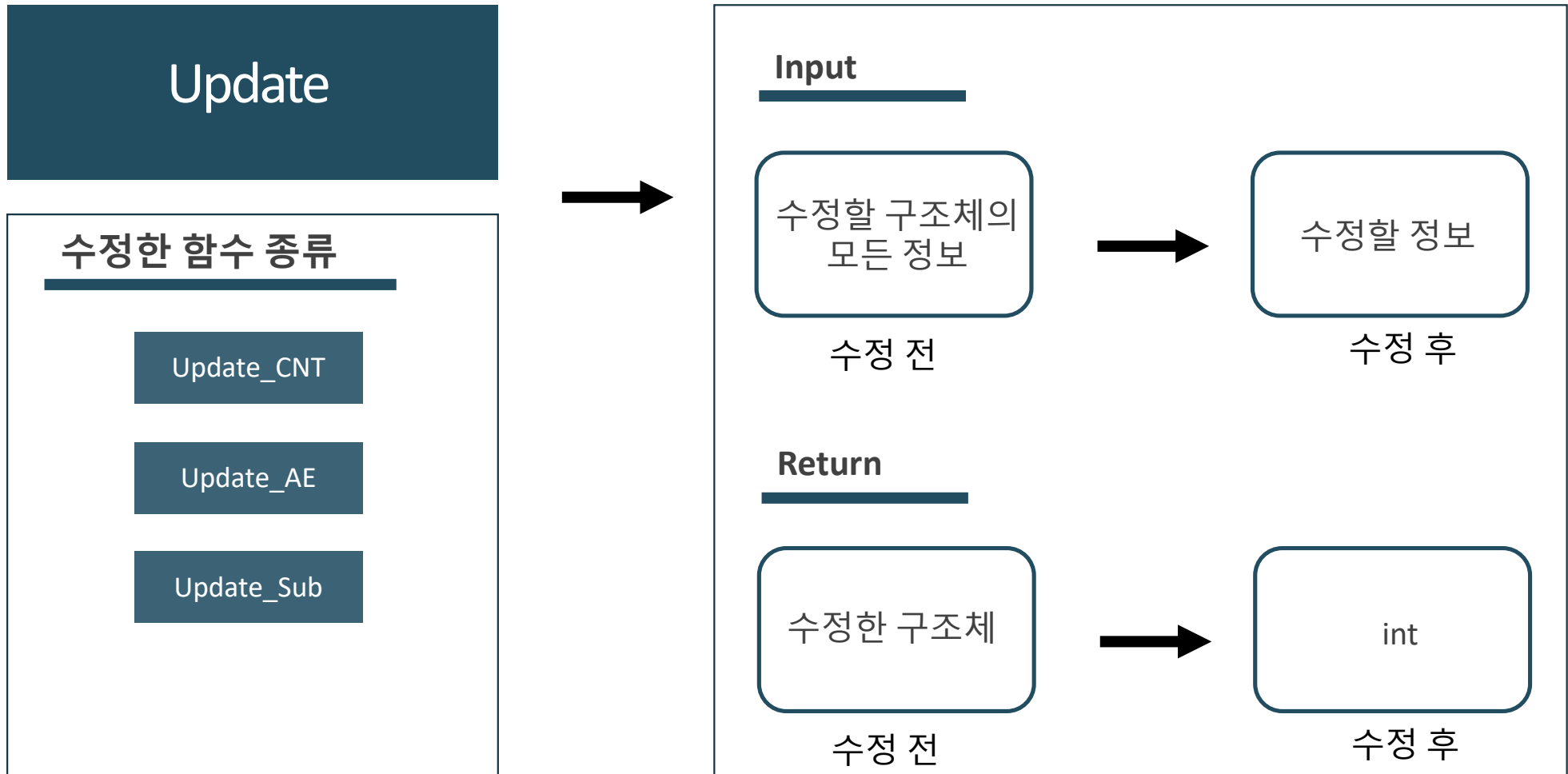
Sub\_Delete -> Delete\_Sub

### 3. Update 함수 인자로 들어오는 오브젝트의 ri 가 NULL이면 에러 처리

**ri NULL ERROR**

### 4. Update 함수 NULL 값 처리

# Update 함수 수정



# AE, CNT DB Type Update

```
/* Not NULL:0, NULL:1 */
int rn_f=0, pi_f=0, ct_f=0,

if(ae->rn==NULL) rn_f=1;
if(ae->pi==NULL) pi_f=1;
if(ae->ct==NULL) ct_f=1;
if(ae->lt==NULL) lt_f=1;
if(ae->et==NULL) et_f=1;
if(ae->api==NULL) api_f=1;
if(ae->aei==NULL) aei_f=1;
if(ae->ty==0) ty_f=1;
if(ae->rr==false) rr_f=1;
```

수정할 구조체 멤버-> 0  
수정하지 않을 구조체 멤버 ->1

```
if (strncmp(key.data, "rn", key.size) == 0 && rn_f==0) {
    cnt_rn++;
    if (cnt_rn == idx) {
        data.size = strlen(cnt_object->rn) + 1;
        strcpy(data.data, cnt_object->rn);
        dbcp->put(dbcp, &key, &data, DB_CURRENT);
    }
}

if (strncmp(key.data, "pi", key.size) == 0 && pi_f==0) {
    cnt_pi++;
    if (cnt_pi == idx) {
        data.size = strlen(cnt_object->pi) + 1;
        strcpy(data.data, cnt_object->pi);
        dbcp->put(dbcp, &key, &data, DB_CURRENT);
    }
}

if (strncmp(key.data, "et", key.size) == 0 && et_f==0) {
    cnt_et++;
    if (cnt_et == idx) {
        data.size = strlen(cnt_object->et) + 1;
        strcpy(data.data, cnt_object->et);
        dbcp->put(dbcp, &key, &data, DB_CURRENT);
    }
}

if (strncmp(key.data, "lt", key.size) == 0 && lt_f==0) {
    ...
}
```

인자에 값이  
들어온 멤버만  
수정

기존 코드에 []\_f 검사하는 부분 추가

# AE, CNT DB Type Update

<UPDATE 전> AE.db

Key	Value
aei	TAE1
aei	TAE3
aei	TAE2
api	tinyProject1
api	tinyProject3
api	tinyProject2
ct	20220513T083900
ct	20220513T083900
ct	20220513T083900
et	20240513T083900
et	20240513T083900
et	20240513T083900
lt	20220513T083900
lt	20220513T083900
lt	20220513T083900
pi	5-20191210093452845
pi	5-20191210093452845
pi	5-20191210093452845
ri	TAE1
ri	TAE3
ri	TAE2
rn	Sensor1
rn	Sensor3
rn	Sensor2
rr	true
rr	true
rr	true
ty	2
ty	2
ty	2

Cursor →  
Find "TAE2"  
세 번째 index

```
rn_f=0,  
pi_f=1,  
ct_f=1,  
lt_f=1,  
et_f=1,  
api_f=1,  
aei_f=1,  
rr_f=1,  
ty_f=1;
```

AE2 수정

ri : TAE2  
rn : Sensor2\_update

<UPDATE 후> AE.db

Key	Value
aei	TAE1
aei	TAE3
aei	TAE2
api	tinyProject1
api	tinyProject3
api	tinyProject2
ct	20220513T083900
ct	20220513T083900
ct	20220513T083900
et	20240513T083900
et	20240513T083900
et	20220513T083900
lt	20220513T083900
lt	20220513T083900
lt	20220513T083900
pi	5-20191210093452845
pi	5-20191210093452845
pi	5-20191210093452845
ri	TAE1
ri	TAE3
ri	TAE2
rn	Sensor1
rn	Sensor3
rn	Sensor2_update
rr	true
rr	true
rr	true
ty	2
ty	2
ty	2

Cursor →

# SUB DB Type Update

```
/* Not NULL:0, NULL:1 */
int rn_f=0, pi_f=0, ct_f=0, lt_f=0,

if(sub_object->rn==NULL) rn_f=1;
if(sub_object->pi==NULL) pi_f=1;
if(sub_object->ct==NULL) ct_f=1;
if(sub_object->lt==NULL) lt_f=1;
if(sub_object->et==NULL) et_f=1;
if(sub_object->nu==NULL) nu_f=1;
if(sub_object->net==NULL) net_f=1;
if(sub_object->sur==NULL) sur_f=1;
if(sub_object->ty==0) ty_f=1;
if(sub_object->nct==0) nct_f=1;
```

수정할 구조체 멤버 -> 0  
수정하지 않을 구조체 멤버 -> 1

```
while ((ret = dbcp->get(dbcp, &key, &data, DB_NEXT)) == 0) {
    if (strcmp(data.data, sub_object->ri, data.size) == 0) {
        idx=0;
    }
    if(idx!=-1 && idx < struct_size){
        // idx==0 -> ri
        if(idx==1 && rn_f==0){
            data.size = strlen(sub_object->rn) + 1;
            strcpy(data.data, sub_object->rn);
            dbcp->put(dbcp, &key, &data, DB_CURRENT);
        }
        if(idx==2 && nu_f==0){
            data.size = strlen(sub_object->nu) + 1;
            strcpy(data.data, sub_object->nu);
            dbcp->put(dbcp, &key, &data, DB_CURRENT);
        }
        if(idx==3 && net_f==0){

```

인자에 값이  
들어온 멤버만  
수정

...

DB상의 정해진 순서대로 순회하며 수정

# SUB DB Type Update

SUB.db

ri : 23-2022040684653299304  
rn : Sub1\_update

Update할 Sub의 ri

```
rn_f=0,
pi_f=1,
ct_f=1,
lt_f=1,
et_f=1,
net_f=1,
sur_f=1,
nct_f=1,
nu_f=1,
ty_f=1;
```

Key(label)	Value(uri)
3-2022040684023203796	23-202204068465329930
3-2021040684023203796	Sub1
3-2022040684023203796	<a href="http://223.131.176.101:3000/ct=json">http://223.131.176.101:3000/ct=json</a>
3-2021040684023203796	1 (net)
3-2022040684023203796	test/test/test/test1
3-2021040684023203796	20220406T084653
3-2022040684023203796	20260406T084653
3-2021040684023203796	20220406T084653
3-2022040684023203796	23
3-2022040684023203796	1
3-2021040684023203796	23-202104068465329930
3-2022040684023203796	Sub2
3-2021040684023203796	<a href="http://223.131.176.101:3000/ct=json">http://223.131.176.101:3000/ct=json</a>
3-2022040684023203796	4 (net)
3-2021040684023203796	test/test/test/test2
3-2022040684023203796	20210406T084653
3-2021040684023203796	20250406T084653
3-2022040684023203796	20210406T084653
3-2021040684023203796	23
3-2021040684023203796	1

Sub1\_update

# Delete 함수 수정

