

Manually declaring transactional/persistent components

```
1 void TMhashtable_update(TM_ARGDECL hashtable_t* hashtablePtr,  
2                          void* keyPtr, void* dataPtr) {  
3     list_t* chainPtr = get_chain(hashtablePtr, keyPtr);  
4     pair_t* pairPtr;  
5     pair_t updatePair;  
6     updatePair.firstPtr = keyPtr;  
7     pairPtr = (pair_t*)TMLIST_FIND(chainPtr, &updatePair);  
8     pairPtr->secondPtr = dataPtr;  
9 }
```

*Prohibited operation,
will cause a runtime error*

(a)

*Transactional interface
of third-party libraries*

Unmodified syntax and semantics

```
1 void hashtable_update(hashtable_t* hashtablePtr,  
2                       void* keyPtr, void* dataPtr) {  
3     list_t* chainPtr = get_chain(hashtablePtr, keyPtr);  
4     pair_t* pairPtr;  
5     pair_t updatePair;  
6     updatePair.firstPtr = keyPtr;  
7     pairPtr = (pair_t*)list_find(chainPtr, &updatePair);  
8     pairPtr->secondPtr = dataPtr;  
9 }
```

*Valid operation, persistent
memory will ensure crash consistency*

(b)