〔非公開〕

ＴＲ－Ｃ－０１５３

# オブジェクト指向データベースシステムにおける完全性のための機構と設計支援手法

沖　也寸志
Yasushi OKI

１９９６　３．１５

ＡＴＲ通信システム研究所

# オブジェクト指向データベースシステムにおける完全性のための機構と設計支援手法

沖　也寸志

１９９６年３月

目次

## あらまし

本報告では，オブジェクト指向データベース(OODB)におけるデータの完全性(Data Integrity)のための機構と設計支援手法について述べる．

通信技術およびネットワーク技術の進歩に伴い，ネットワークは高度かつ多様なサービスを提供できるようになりつつある．それに伴い，ネットワーク管理データベースはこれまで以上に多くのサービス情報やユーザ個人の情報を扱うことになるため，データベースセキュリティがますます重要になる．特にネットワーク管理にもオブジェクト指向パラダイムを導入することが主流になりつつあることを考えると，OODBのセキュリティは重要であると考える．

データベースセキュリティの主な課題として，機密性と完全性がある．これまでのOODBのセキュリティ研究では，機密性の研究がいくつか報告されている．しかしながら，OODBの完全性の研究は，その重要性にもかかわらずほとんど報告されていない．本報告では，まずデータの完全性のためにインテグリティレベルを用いた強制アクセス制御を導入について述べ，次にセキュリティ上の要求を満たす適切なインテグリティレベルを設定するための設計支援手法について述べる．最後に，まとめを行う．

## １．はじめに

通信技術およびネットワーク技術の進歩に伴い，ネットワークは高度かつ多様なサービスを提供できるようになりつつある．特にインテリジェントネットワークの研究により，将来的には，ユーザ自身が自分の必要とするサービスをカスタマイズできる機能が提供されるようになる．このようなサービスが実現されると，ネットワーク管理データベースはこれまで以上に多種多様なサービス情報を扱うことになるし，ユーザ個人の情報をも扱うことになるため，データベースセキュリティはますます重要になる．一方，データベース技術においても新しい動きがあり，新しい構造を持つデータベースとしてオブジェクト指向データベース(OODB)が注目されてきている．ネットワーク管理にオブジェクト指向パラダイムを導入する[CCITT 92]ことが主流になりつつあることを考えると，OODBのセキュリティに関する研究は重要である．

データベースセキュリティにおける主な課題としては，機密性と完全性がある．機密性は，知られてはならない情報の漏洩，すなわち機密漏洩に関係している．完全性は，不当な情報の書き込み，すなわち情報の変更に関係している．これまでにATRの荒木らはOODBの機密漏洩防止に関して，基本的にBLPモデル[BLP 76]に基づくアクセス制御機構[Araki 93a]を提案している．さらに，強制アクセス制御を行うためにはあらかじめ適切なセキュリティレベルを設定しておく必要があるので，適切なレベルを設定するためのセキュリティ設計支援手法[Araki 93b]を提案している．本報告では，上記の機密漏洩防止手法を適用して，アクセス制御の面からデータの完全性のための機構と設計支援手法について述べる．具体的には，インテグリティレベル[BLP 76][Biba 77]によるアクセス制御を導入し，OODBへの適用について述べる．さらに，セキュリティ上の要求を満たした適切なインテグリティレベルを求める手法として，機密漏洩防止のための設計支援手法が適用できることを示す[Oki 95].

本報告の構成は次のとおりである．２節では３節以降の議論の準備のため，OODBの機密漏洩防止手法について述べる．３節ではデータの完全性について述べる．４節では データの完全性のためのアクセス制御機構を述べる．５節ではセキュリティレベルとインテグリティレベルを求めるためのセキュリティ設計支援手法を述べる．最後に，６節でまとめを行い，到達点と限界，および今後の課題について述べる．

## ２．OODBの機密漏洩防止手法

この節では，OODBの機密漏洩防止手法について述べる．まず，機密漏洩防止の観点から，OODBのデータモデルについて述べ，そのモデルに適用するアクセス制御機構について述べる[Araki 93a]．次に，セキュリティレベルを決定するためのセキュリティ設計支援について述べる[Araki 93b]．

### ２．１　データモデル

現在のところ，OODBについての厳密な定義は存在しないが，セキュリティを考える上では以下の点が特徴になると考えられる．

(a)データとメソッドの一体化：個々のデータに対してそれを操作するためのメソッドが定義されており，データへのアクセスはメソッドを通じて行われる．

(b)クラスとインスタンスの関係：クラスはデータ構造の型やメソッドを定義しているオブジェクトであり，インスタンスはそれを具体化した個々のデータそのもののオブジェクトである．

(c)クラスの汎化階層：あるクラスで定義された型やメソッドを具体化，特殊化したクラスを定義した場合，そのクラスを元のクラスのサブクラスといい，元のクラスをスーパークラスという．サブクラスはスーパークラスの性質を基本的に継承する．このようなクラス間の階層関係を"is-a関係" という．

(d)複合オブジェクト：オブジェクトの構成要素が別のオブジェクトであるような入れ子構造をなすオブジェクトを複合オブジェクトという．このような関係を " part-of 関係"という．

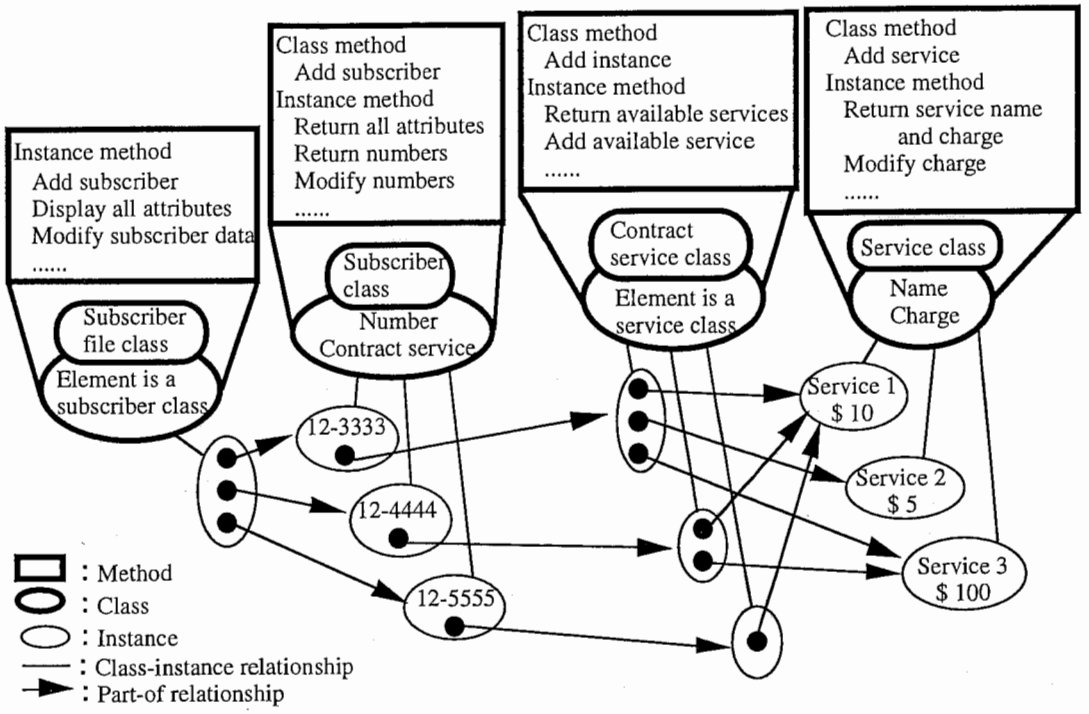


図１　加入者データベース

図１は電話の加入者情報を持つOODBシステムの例を表わしている．サービスクラス(Service class)は各サービスの名前(Name)と料金(Charge)のデータを持っており，加入サービスクラス(Contract service class)は各電話番号の契約サービス情報を持っている．加入者クラス(Subscriber class)は電話番号(Number)と加入サービスクラス内のデータとの対応関係(Contract service)，すなわちポインタを持っている．たとえば加入者"12-4444"は，"Service 1"と"Service 3"を契約しており，料金はそれぞれ"$10"と"$100"になっている．

## ２．２　アクセス制御機構

ATR teamは機密漏洩防止のためのアクセス制御に，基本的にBLPモデル[BLP 76]に基づく強制アクセス制御を採用している．BLPモデルはBell and LaPadulaがコンピュータシステムの機密性実現のために提案したモデルである．このモデルでは，ユーザーやシステム内のデータなどのシステムの構成要素を抽象的にエンティティと呼び，そのうちread/writeを行うエンティティをサブジェクト，その対象となるエンティティをオブジェクトと呼ぶ．（本報告では"オブジェクト"という言葉を，上記の意味か，あるいは”オブジェクト指向”のオブジェクトの意味で，特に断わらずに使用する．）アクセス制御の方針は次のとおりである．各エンティティにはセキュリティレベルを設定しておき，アクセスに伴って発生する情報フローが，セキュリティレベルの下位レベルから上位レベル，または同レベル間ならば許可する（図２参照）．

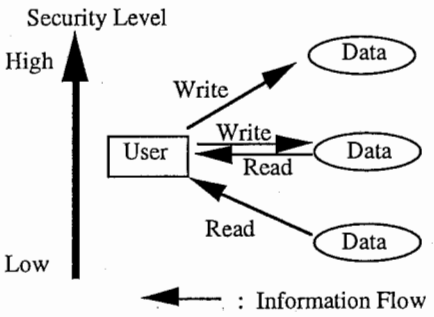


図２　セキュリティレベルによるアクセス制御

アクセス制御機構の提案では，OODBの機密評価のためのデータモデルと，OODB内の各オブジェクトが持つセキュリティレベルが機密漏洩防止のために満たすべき制約，“セキュリティレベル設定制約”を定めている．

提案されたデータモデルは，きめ細かい機密要求を設定できるという特長を持っている．たとえば，クラス中の各変数に異なるセキュリティレベルを設定できるし，メソッドにもセキュリティレベルを設定してメソッドの存在についても機密を考慮している．さらに，インスタンスごとに異なるセキュリティレベルを設定できる．

セキュリティレベル設定制約は，各エンティティの持つセキュリティレベルの関係が，機密漏洩防止の目的に矛盾しないために必要な制約である．たとえば，あるクラスＣのレベルはクラス中のエンティティのレベル以下になっているべきである．そうすることにより，そのクラスを知られてはならないユーザＵの場合には，レベル間の関係

SL(U) ＜ SL(C) ≦ SL(C中のエンティティ）

が成り立つので，クラス中のエンティティも知ることができないことが保証される．ここ

で，SL(e) はエンティティeのセキュリティレベルを表わすものとする．他にも様々なセキュリティレベル設定制約が存在しており，種類の総数は２９ほど存在する．

アクセス制御規則はアクセス実行に関してはBLPモデルとほぼ同様である．ただし，メソッドも含めてオブジェクトの存在を推測させないように，オブジェクトの表示に関して制限を追加している．ユーザのレベル以下のレベルを持つオブジェクトやメソッドはユーザに対して表示可能であるが，そうでない場合は表示されない．これにより，たとえばメソッドの名前が見えることにより，データの存在が推測されることは無い．

あるエンティティ E のセキュリティレベルを SL(E) で書くとき，アクセス制御規則は次のとおりである．

[アクセス制御規則]

ユーザを U, オブジェクトを O とすると

－アクセス許可条件：

read： $SL(U) \geqq SL(O)$

write： $SL(U) \leqq SL(O)$

（新オブジェクト追加の場合，新オブジェクトのレベルを SL(U) に設定）

－オブジェクトの表示条件：

$SL(U) \geqq SL(O)$

ただし，上記で述べたアクセス制御機構は，以下の仮定をしている．

－レベルの順序関係は半順序関係とする

－ユーザの権限の変更やデータ構造の変更（クラスの変更，新クラスの追加）の際は，レベルの再設定を行う

－クラスの多重継承（複数のクラスをスーパークラスとして持つ）を考慮しない

## 2．3　設計支援

セキュリティレベルによるアクセス制御機構を使って機密漏洩防止を行うためには，機密に関する要求を満足するセキュリティレベルを設定する必要がある．

### [セキュリティレベルの設定例]

図3は階層構造を持ったデータの例である．簡単化のため，セキュリティレベル設定制約はデータの階層構造だけにしている．データにアクセスするユーザA，B，Cに対して，以下の機密要求を実現する．

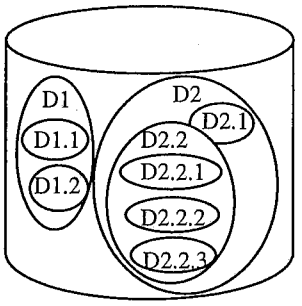


図3　階層構造

機密要求：

ユーザAにはデータD1.2とD2.2を機密にする．

ユーザBにはデータD2.2.3を機密にする．

ユーザCには制限を付けない．

これを実現するには，セキュリティレベルをたとえば以下のように設定すればよい．

SL (A) = 1,  SL(B) = 2, SL(C)=3

SL(D1) = 1, SL(D1.1) = 1, SL(D1.2) = 2,

SL(D2) = 1, SL(D2.1) = 1,

SL(D2.2) = 2, SL(D2.2.1) = 2, SL(D2.2.2) = 2, SL(D2.2.3) = 3

この例は，簡単にするために全順序のレベルが設定可能な例になっているが，ほとんどの場合には半順序のレベル設定が必要である．

□

実際には，セキュリティ設計者の求める機密要求，ユーザからのアクセス要求，データ構造に起因するセキュリティレベル設定制約が与えられたとき，機密要求を満足する一方で，できるだけアクセス要求を実現するセキュリティレベルを求める必要がある．これを間違いなく行うにはシステムの機械的な支援が必要であり，[Araki 93b]で提案している．

## 3．データの完全性

この節では，アクセス制御の観点から完全性について議論する．

完全性を損う原因は，Bibaが指摘しているように直接，間接の大きく2つに分類できる[Biba 77]．直接の完全性破壊は，不当な書き込みアクセスが行われることにより起きる．一方，間接の完全性破壊は，既に不当なデータが存在している場合に，それが読み込まれて別の不当なデータとして書き込まれることにより起きる．または，他を破壊する不当なプログラムが存在している場合に，そのプログラムが使用され，不当なデータの書き込みが行われることにより起きる．仮に，もし完全なシステムが存在しており，直接の完全性破壊を完全に防止できたなら，間接的な完全性破壊も防止できることになる．したがって，直接の完全性破壊防止が最も重要である．以降では，直接の完全性破壊の防止に注目する．

前節で，機密性のために情報フロー制御を行うアクセス制御を導入した．目的が機密性の場合には，情報フロー制御を行えば必ず目的を達することができる．一方，目的が完全性の場合には，アクセス制御は一つの手段であることは間違いないが，それだけでは十分ではない[Schell 86, Sandhu 93a]．書き込みアクセスが実行されると，ユーザに悪意がなくても誤って不当なデータが書き込まれる可能性がある．この問題に対して，たとえばデータの正当性の保護機構など，データベース構造の研究が必要である．しかしながら，適切なアクセス制御が重要であることには変わりない．不当なデータの書き込みが心配される場合にはもちろん，書き込みの必要のない場合にはアクセスを許可しないアクセス制御が必要である．

書き込み禁止のアクセス制御の実現方法として，前述のセキュリティレベルによるアクセス制御がWrite Downを禁止する性質を利用することが考えられる．たとえば，サブジェクトUがオブジェクトOにWriteできないように，セキュリティレベルを SL(U) > SL(O) の関係にする．これには2つの問題がある．一つは，SL(U) > SL(O) の関係にあるときに，絶対Writeができないならば，システムは可用性に関して重大な問題を持つことになる．セキュリティレベルの大きいサブジェクトは，機密を知る高い権限をもっているにもかかわらず，常にWriteできないことを意味する．サブジェクトが信頼できる人間ならば，自分のレベルを下げることにより(Down Grade)，サブジェクトのレベルとオブジェクトのレベルの関係が制御ルールに違反しないようにすれば，Writeすることは許されるべきである[Sandhu 93b]．2つめとして，機密性の条件から SL(U)≦SL(O) の関係にしたい場合には，完全性の要求と矛盾する．したがって，SL(U) > SL(O) の関係を，完全性のためのWrite禁止に使うことはできない．完全性のためには，セキュリティレベルとは独立したアクセス制御が必要である．

上記で述べたように，間接の完全性破壊の原因として他を破壊する不当なプログラムの問題がある．プログラムにはデータ以上に厳しい管理が必要なことは言うまでもないが，本報告では対象にしていない．

# ４．アクセス制御機構

## ４．１　完全性のためのアクセス制御追加

完全性のためのアクセス制御を実現する方法として，あらかじめサブジェクトとオブジェクトの組ごとの書き込みアクセス権をアクセスマトリクス中に設定しておいて，そのアクセス権により制御する discretionary access control も考えられるが，著者らはセキュリティレベルとは独立したインテグリティレベル[BLP 76]により制御する mandatory access control を採用した．なぜなら，OODBの複雑なデータ構造に対して書き込みアクセス権限をミスなく設定するために，ATR teamが提案した機密性のためのアクセス制御機構と設計支援手法が完全性にも有効なためである．

基本的に機密性のためのアクセス制御機構はそのままにしておき，完全性のためのアクセス制御を追加する．OODB内の各エンティティにあらかじめインテグリティレベルを設定しておき，主体（能動的なエンティティ）のインテグリティレベルが客体（受動的なエンティティ）のインテグリティレベル以上であれば書き込み可能とする．

あるエンティティＥのインテグリティレベルを IL(E) で表わすとき，改ざん防止も考慮したアクセス制御は以下のとおりである．下線部分が改ざん防止アクセス制御の追加分に相当する．

[制御ルール]

－各オブジェクトに対するアクセス許可条件：

read：　SL(U)≧SL(O)

write：SL(U)≦SL(O)　かつ　IL(U)≧　IL(O)

（新オブジェクト作成の場合，

新オブジェクトの SL(O) を SL(U) に設定，

さらに新オブジェクトの IL(O) を IL(U) に設定）

－メソッドの表示条件：

SL(U)≧SL(O)

－メソッドの実行条件：

write：　IL(U)≧ IL(m)

（これが満たされれば，新オブジェクトはインテグリティレベル設定制約（後述）に違反しない）

## ４．２　データモデル

インテグリティレベルのアクセス制御機構においてもきめ細かい書き込み禁止要求を可能にするために，ATR teamが提案したOODBのデータモデルを採用する．このデータモデルにより，OODBのクラス中の変数のそれぞれに異なるインテグリティレベルを設定できること，インスタンスごとに異なるインテグリティレベルを設定できること，書き込みメソッドにもインテグリティレベルを設定することにより

新規データの追加を制御できることなど，きめ細かい書き込み禁止要求を実現できるという特長がある．

## 4．3　インテグリティレベル設定制約

機密性のためにセキュリティレベル設定制約が必要だったように，完全性の場合もインテグリティレベル設定制約が必要である．

制約が必要になる理由は2つある．一つはエンティティ間の意味的な包含関係から生じる制約である．たとえば，あるクラスに書き込み禁止なら，そのクラス内のインスタンスも書き込み禁止であるべきである．もう一つは，メソッドの実行許可判断を正しく行うための制約である．たとえば，あるメソッドがいくつかのインスタンス変数に関連するとき，メソッドのレベルはそれがアクセスするインスタンス変数のレベル以上にすべきである．もしそうでないなら，メソッドを実行したけれどもwriteが全く許可されないことがありうる．また，新オブジェクト（新インスタンス）を追加するメソッドを実行する場合には，追加されたインスタンスがレベル設定制約に矛盾するかもしれない．

ここでは制約のうち，重要な関係のみをインフォーマルに説明するにとどめる．基本的には，各セキュリティレベル設定制約をインテグリティレベルに対応させて考えることができる．

（1）意味的な包含関係から生じる制約

－クラスとそのクラス内のエンティティ

IL(クラス) ≦IL(クラス内のエンティティ)

エンティティは，サブクラス，インスタンス，インスタンス変数，クラス変数，メソッドなど．

－インスタンス変数とインスタンス変数値

IL(インスタンス変数) ≦IL(インスタンス変数値)

（2）メソッドの実行許可判断，および新オブジェクトのレベルを適正に保つための制約

－メソッドとそのメソッドが書き込みするインスタンス変数

IL(メソッド) ≧IL(インスタンス変数)

－メソッドと，それが二次的に起動するメソッド

IL(メソッド) ≧IL(二次的に起動するメソッド)

# 5．　設計支援

## 5．1　セキュリティ設計の全体像

OODBの各エンティティのインテグリティレベルを決める作業を完全性設計と呼ぶ．このとき，セキュリティ設計者が考える完全性のためのデータのインテグリティ要求に対して，ユーザのアクセス要求をできるだけ満足するレベルを設定すべきである．インテグリティ要求は，あるユーザにデータの変更や追加を禁止するものであり，たとえば「ユーザ A はデータ 1 を変更してはならない」，「ユーザ B はクラス 1 内に書き込んではならない（データの変更も追加も禁止）」というような要求である．一方，アクセス要求は，ユーザが実行を希望するメソッドのことであり，たとえば「ユーザ A はメソッド 1 を実行したい」というものである．

完全性設計を人手で行うことは，データ量の増加，ユーザ数の増加，データ構造の複雑化，および要求の複雑化を考えると，非常に困難である．したがって，要求に応じて機械的にレベルを決定する設計支援手法が必要である．

ATR teamは，OODBの各エンティティのセキュリティレベルを決めるために，入力されたアクセス要求，機密要求に対してできるだけアクセス要求を満足するセキュリティレベルを設定する支援手法を提案している[Araki 93b]．この設計支援手法はインテグリティレベルの設計支援にも適用可能である．この手法によるセキュリティレベルとインテグリティレベルの両方を求めるセキュリティ設計の全体の手順は，以下のとおりである．

ステップ 1　要求入力
- (1.1)　アクセス要求入力
- (1.2)　機密要求入力
- (1.3)　インテグリティ要求入力

ステップ 2　要求の矛盾検出と矛盾解消
- (2.1)　アクセス要求と機密要求の矛盾検出と解消
- (2.2)　アクセス要求とインテグリティ要求の矛盾判定と矛盾解消

ステップ 3　レベルの計算
- (3.1)　セキュリティレベルの計算
- (3.2)　インテグリティレベルの計算

## 5．2　インテグリティレベルの設計支援

セキュリティ設計のうち，インテグリティレベルの設計支援に関する部分について述べる．

### （ステップ　1　(1.1, 1.3)）要求入力

アクセス要求とインテグリティ要求を設計支援システムに入力する．入力されたアクセス要求は，ユーザとそのユーザが実行を希望するメソッドの組 (U, m) で表わされる．インテグリティ要求は，ユーザとそのユーザに書き込みを禁止したいオブジェクトの組 (U, O) で表わされる．

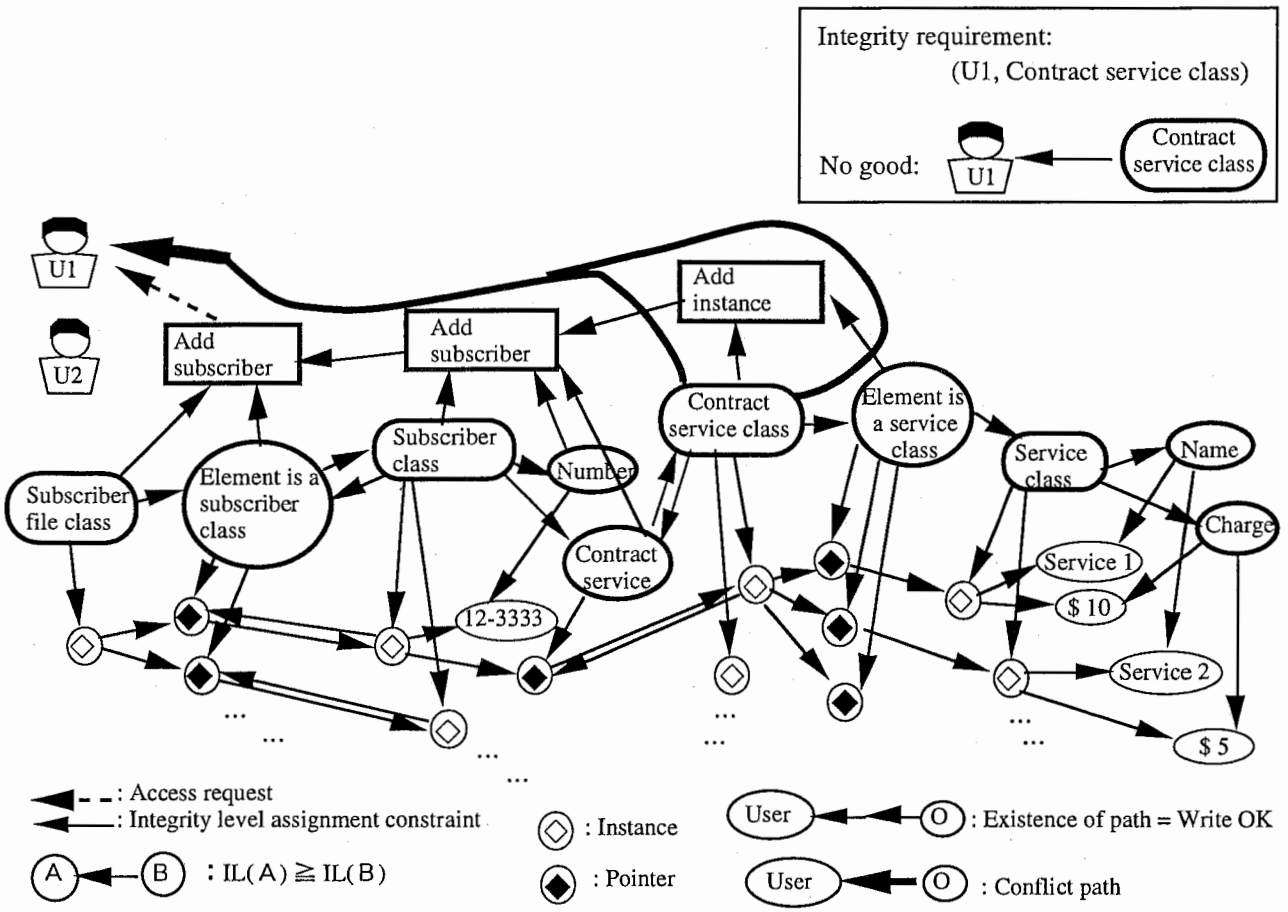


図４　矛盾検出および矛盾解消の例

### （ステップ　2 (2.2)）要求の矛盾検出と解消

アクセス要求のうち書き込みを伴うアクセス要求とインテグリティ要求間の矛盾を検出する．最初に，ユーザとOODBのエンティティを頂点に，アクセス要求とインテグリティレベル設定制約を有向辺とした有向グラフを作成する．有向辺の向きは，アクセス要求の場合はメソッドからユーザへの向きであり，インテグリティレ

ベル設定制約の場合は，たとえばエンティティE1, E2 に E1 ≦ E2 の制約があった場合，E1 から E2 の向きとする．図４は図１の加入者データベースの一部分とユーザ１のアクセス要求を有向グラフとして表わした例である．

作成された有向グラフ上で，オブジェクトからユーザに至るパスが存在するならば，インテグリティレベルに関する制御ルールによりユーザはオブジェクトに対する書き込みが許可されることになる．図４の例では，たとえば，加入サービスクラス(Contract service class)からユーザ１(U1)にパスが存在するので，ユーザ１は加入サービスクラスに書き込みできることになる．ところが，インテグリティ要求として(U1, Contract service class) が設定されているので，インテグリティ要求とアクセス要求は矛盾を含んでいる．逆に言えば，あるインテグリティ要求の組 (U, O) があるにもかかわらず，有向グラフ中にオブジェクト O からユーザ U に至るパスがある場合を矛盾という．図４の例では，加入サービスクラスからユーザ１に至る太い矢印が矛盾パス(Conflict path)である．

セキュリティ設計支援システムは上記の矛盾検出を行い，矛盾の有無をセキュリティ設計者に報告する．矛盾があった場合は，セキュリティ設計者は設計支援システムとの対話処理により，「アクセス要求をあきらめる」，「部分的な実行を行うメソッドでかまわない」，または「部分的な要求を実現する新メソッドを作成する」という選択肢から処置を選び，矛盾パスを削除する．矛盾が複数存在するときは，この処理を繰り返し，すべての矛盾を解消する．

### （ステップ３ (3.2)）インテグリティレベルを計算

ステップ２で得られた有向グラフの有向辺のみを許可し，有向辺の設定のないエンティティ間には関係が定義されない半順序レベルを計算により求める．得られた半順序レベルがインテグリティレベルになる．半順序レベルの計算方法は，セキュリティレベルの計算方法と同じである．

## 6．まとめ

### （到達点：何が得られたのか？）

本報告では，OODBシステムの完全性を実現するためのアクセス制御機構と，個々のOODBに関する要求を考慮したアクセス制御設計支援手法について述べた．

システムのセキュリティに関して機密性と完全性が注目されているが，特に機密性については盛んに研究されてきている．これまでに機密性のためにセキュリティレベルを用いたアクセス制御機構とアクセス制御設計支援手法が提案されており，同様にアクセス制御というアプローチによる完全性実現の研究が望まれていた．アクセス制御の面からシステムの完全性を実現するためには，信頼できる書き込みアクセスのみを許可するためのしくみが必要であり，さらにそのしくみに基づいて動作する個々のOODBシステムに関する要求を満たす適切な権限を決定する必要がある．本報告では，アクセス制御機構としてインテグリティレベルを用いたアクセス制御規則を追加し，OODBの各エンティティのレベル間の矛盾を防止するためのインテグリティレベル設定制約について述べた．さらに，このアクセス制御機構が動作するシステムに対して，ユーザのシステム利用要求とセキュリティ管理者のセキュリティ上の要求を与えられたとき，適切なインテグリティレベルを誤りなく機械的に決定するために，セキュリティレベルを決定する設計支援手法が適用できることを述べた．設計支援システムに要求を入力し，矛盾が検出されたときはシステムとの対話処理により矛盾解消をすれば，できるだけ要求を満足するインテグリティレベルを求めることができる．

### （限界）

完全性実現のしくみとしてアクセス制御の重要性は明らかであるが，残念ながらアクセス制御面からのアプローチだけでは十分でない．なぜなら，書き込みアクセスを行うユーザに全く悪意がない場合でも，データの変更間違いの可能性は残るためである．この問題を防ぐには，システムが誤った入力データを許容しないしくみ，すなわちデータの正当性の維持機構が不可欠である．これについてはデータベース構造の研究がなされており，これらの研究結果を利用すべきである．

一方，機密性に関しては，現時点まで注目しなかった問題に，属性結合問題と推論問題がある．属性結合問題には２つのタイプがある．一つは公開されている属性ＡとＢのデータ，属性ＢとＣのデータから機密情報の属性ＡとＣのデータがわかってしまう問題である．たとえば，名前と仕事の関係情報，仕事と給与の関係情報から名前と給与の関係がわかってしまう．もう一つは，ある情報が一定量集まることにより機密がわかってしまう問題である．たとえば，軍の部隊の配置情報を一定量集めると，何らかの作戦がわかってしまう．また，推論問題とはデータベース内の公開されている情報と，データベース外の情報から機密情報が推論できてしまう問題である．たとえば，データベース情報では，ある戦闘機に800ポンドの爆弾を積

むことになっており，一般に爆弾１個は100ポンドであることが知られているなら，その戦闘機は８個の爆弾を積むことがわかってしまう[White 95]．これらの属性結合問題と推論問題は，「機密漏洩の本質は情報フローにある」という方針だけでは対処できないので，新たな視点からのアプローチが必要である．

## （今後の課題：何が残っているのか？）

応用研究（実用化研究）として，以下の研究課題がある．

１．OODBシステムと設計支援システム間のインタフェース

設計支援システムを使ってセキュリティ設計を行うためには，設計支援システムは対象となるOODBシステム内のデータ構造とデータを把握していなければならない．個別のOODBごとにインタフェースの設計をすることも可能であるが，情報交換を行うためのインタフェースの研究が望まれる．

２．レベル設定アルゴリズムの速度評価と高速化

現試作システム上では，小規模なデータに対してさえレベルを求めるための計算時間がずいぶんかかっており，このままでは大規模データになると使えないことになるが，次の２つの理由により大きな問題とは考えていない．

現実的な使われ方を考えると，データもアクセスする側（ユーザ）もかなり大きな単位をひとつのエンティティとして扱われると予想される．したがって，レベル設定の必要なエンティティ数はあまり大きくならないと考えている

仮に多数のエンティティに対するレベル設定が必要だとしても，現試作システムでの問題は，試作システムが高速計算の得意でない言語（オブジェクト指向言語）を使ってレベル設定アルゴリズムを実現していることが大きな原因であると著者は考えている．高速計算の得意な言語を使ってアルゴリズムを実現し見積りを行う必要がある．それでもまだ時間計算量の問題が深刻な場合に限り，計算高速化の研究が必要である．

３．適用対象システムの拡大

ネットワーク管理にオブジェクト指向の考え方を導入することが検討されているなど，各方面でオブジェクト指向システムの研究がなされていることを考えると，提案した手法はOODBだけでなく多方面への応用が期待できる．

さらに，これまでオブジェクト指向システムを対象としたが，オブジェクト指向概念に基づかないシステム（既存のシステム）への適用も検討する価値がある．対象システムのデータ構造を分析して新たなアクセス制御機構を定めれば，設計支援手法は今のまま適用可能である．

次に，（限界）のところで述べたように，さらに困難な問題への挑戦として以下

の研究がある．

1．機密性：情報の結合問題，推論問題

2．完全性：誤りによる不当な書き込み問題

ただし，上記1，2に取り組む場合，同時に可用性という面からの考察が不可欠である．

機密性の結合問題については，属性の組み合わせにもアクセス権限を設定するアイデアがあるが，可用性に対する副作用が大きいので簡単ではない．あまり厳しく制限すると使いにくい（使えない）システムになってしまう．推論問題の方はさらに困難な問題である．推論に使われる知識の方は管理できないからである．

完全性に関しては，データベース中のデータの正当性維持機構の研究を進める必要がある．

# 参考文献


[BLP 76] Bell, D.E. and LaPadula, L.J., "Secure Computer System: Unified Exposition and Multics Interpretation", MITRE Corp., MTR-2997, 1976 (Available as NTIS AD-A023588).

[CCITT 92] CCITT Recommendation M.3010, "Principles for a Telecommunications Management Network", 1992.

[Araki 93a] Araki, T., Chikaraishi, T., Hardjono, T., Ohta, T. and Terashima, N., "An Access Control Mechanism for Object-Oriented Database Systems", IEICE Trans. Fundamentals, vol.E76-A, no.1, Jan. 1993.

[Araki 93b] 荒木禎史，力石徹也，Hardjono, T.，太田理，”オブジェクト指向データベースのセキュリティ設計支援手法”，信学会 1993年暗号と情報セキュリティシンポジウム SCIS93-14A, Jan. 1993.

[Biba 77] Biba, K.J., "Integrity Considerations for Secure Computer Systems", Mitre TR-3153, Mitre Corporation, Bedford, Massachusetts, (1977). (Also available through National Technical Information Service, Springfield, Va., NTIS AD-A039324)

[Oki 95] Oki, Y., Chikaraishi, T., Shimomura, T. and Ohta, T. , "A Design Method for Data Integrity in Object-Oriented Database Systems", IEEE SICON/ICIE '95, pp.204-209, Jul. 1995.

[Schell 86] Schell, R.R. and Denning, D.E., "Integrity in Trusted Database Systems", the 9th National Computer Security Conference, pp.30-36, 1986.

[Sandhu 93a] Sandhu, Ravi S., "On Five Definitions of Data Integrity", Database Security VII(A-47) Status and Prospects, IFIP WG11.3 Workshop on Database Security, pp.257-267, Sep., 1993.

[Sandhu 93b] Sandhu, Ravi S., "Lattice-Based Access Control Models," IEEE Computer, pp.9-19, Vol.26, Nov., 1993.

[White 95] White, Gregory B., Fisch, Eric A. and Pooch, Udo W., "Computer System and Network Security", Computer Engineering Series, CRC Press, 1995.