SNSにおける
情報漏洩を防止するための
情報フィルタの適用

木下研究室　　200402735　　内野雄策

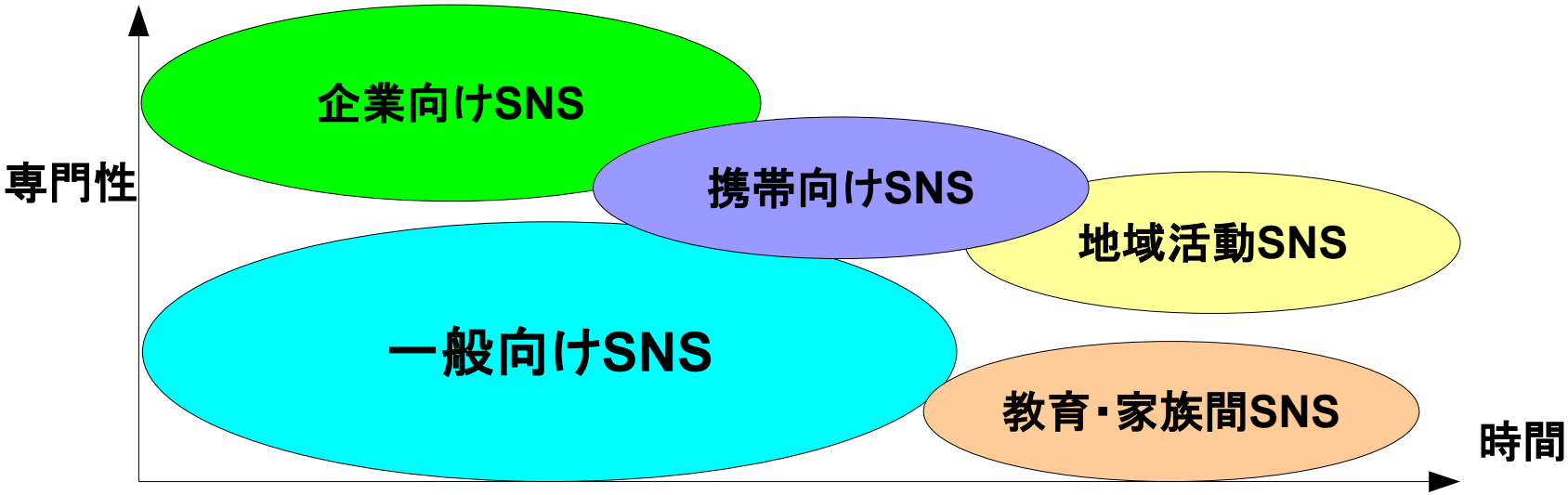
作成日:2009年2月13 日

背景と目的

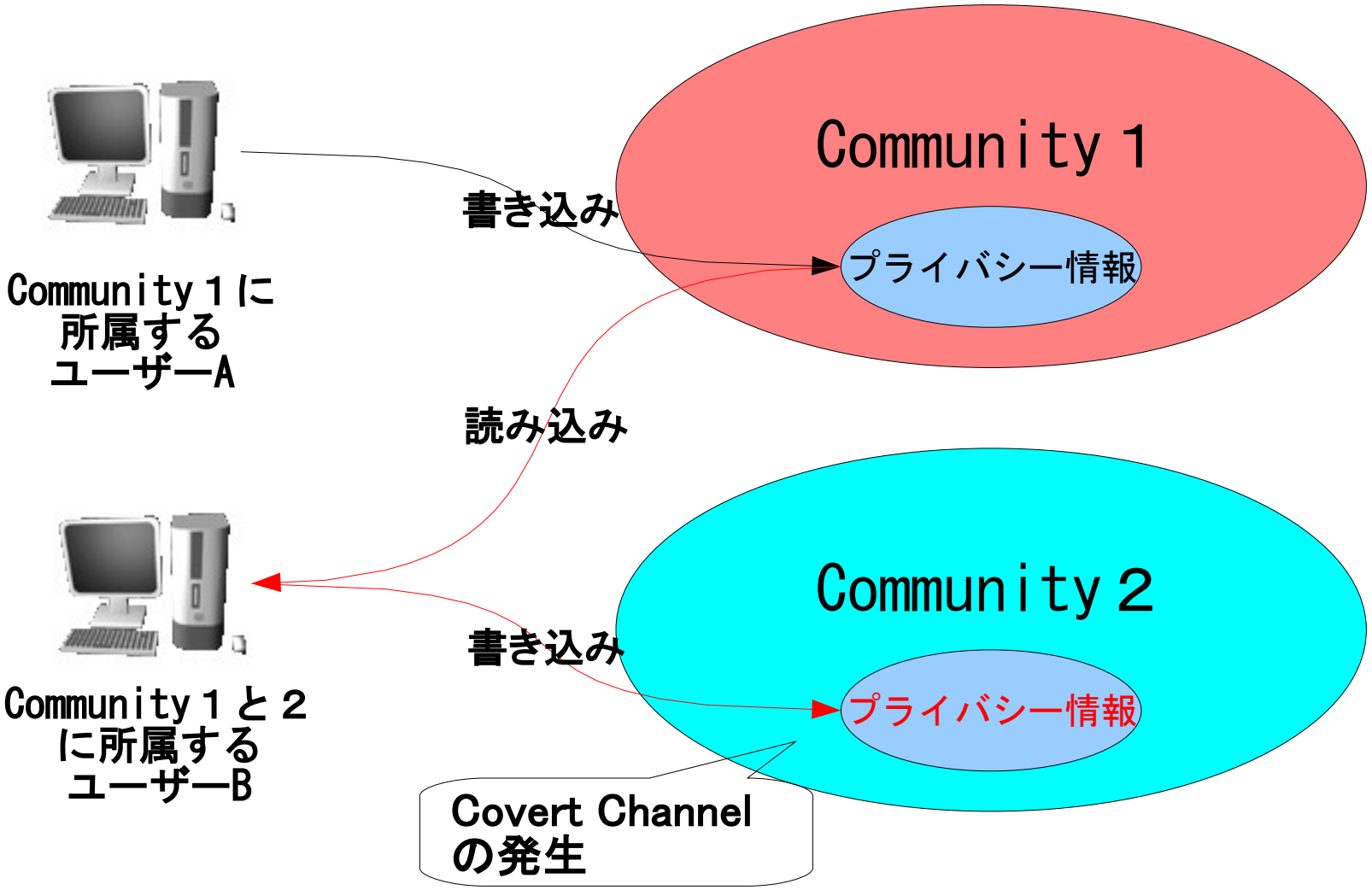
- SNSが様々な分野で利用され始めている。
- コミュニティ間で、意図しない情報漏洩が発生する危険が増す。
- 情報漏洩を防止するための
  - 情報フィルタの適用法を考案し、
  - オープンソースのSNSエンジンのOpenPNEに実装する。

SNSの推移

- 近年、ネットワークの拡大と共に、mixi を始めとしたSNSが浸透している。
- SNSユーザーの増加に伴い、このSNSには趣味から、政経済、教育など、様々な遷移を遂げる。
- Web上での個人情報などの重要なやり取りは、利便性の向上と，それと同時に情報改竄や情報漏洩、および不正アクセスという問題が発生する。
- それらに対する安全性の確保が重要視されるようになる。



コミュニティ間における間接情報フロー（Covert Channel）



アクセス行列におけるCovert Channelの流れ

アクセス権と矛盾する間接情報フロー（Covert Channel）が発生する場合

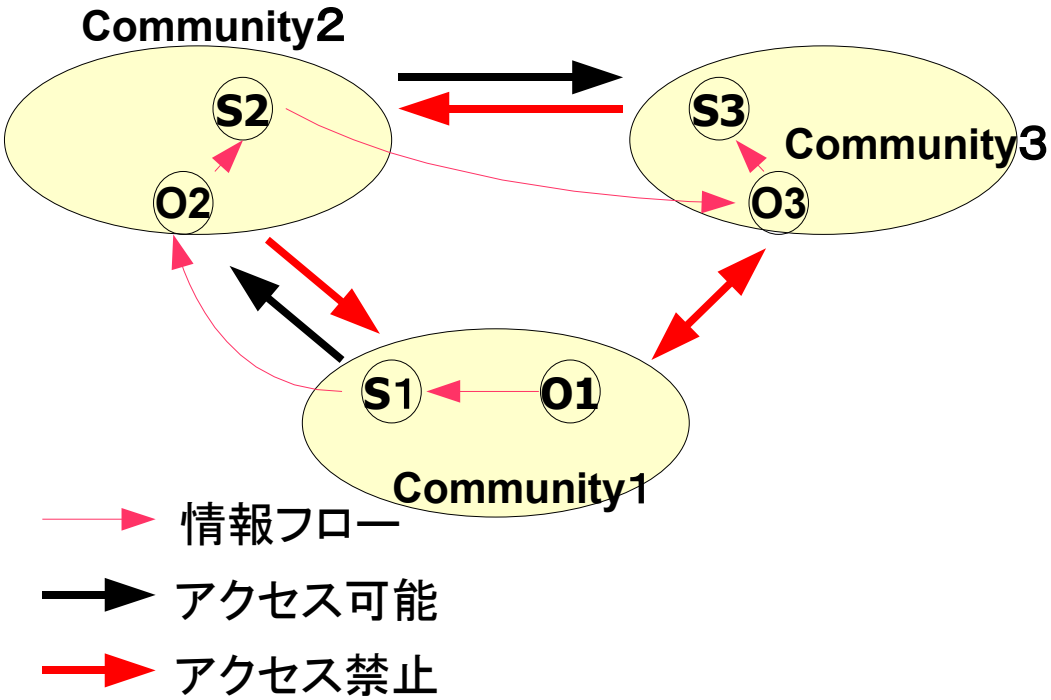
	S_1	S_2
O_1	R	φ
O_2	φ	R

情報フロー
が起こる場合

	S_1	S_2
O_1	R	φ
O_2	W → R	

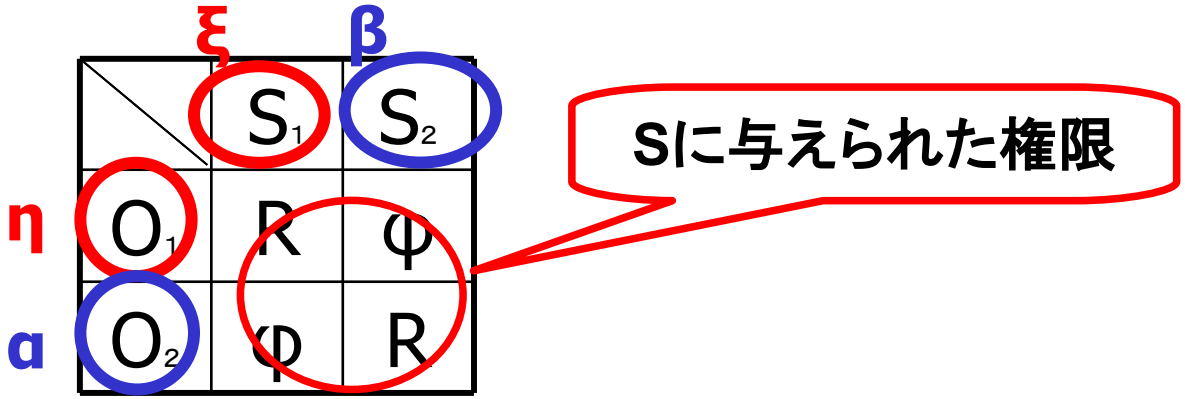
S：主体（ユーザー） O：客体（情報） R：Read権 W：Write権 φ：アクセス禁止

	S_1	S_2	S_3
O_1	R	φ	φ
O_2	W →	R	φ
O_3	φ	W →	R



情報フィルタ

S(subject) ： アクセスしようとするプログラム、情報にアクセスする主体。
O(object) ： 操作の対象となっているデータ、プログラムが扱う客体（対象）
Community ： Communityに属する主体及びObjectの集合、もしくは属性。



S:主体　O:客体　R:Read権　W:Write権　φ:アクセス禁止

Covert Channelを防ぐには、主体に与えられている権限を変更することで、不正な情報経路を制限できる。

Covert Channelを制御するため、
主体と客体に属性を与え定義する。

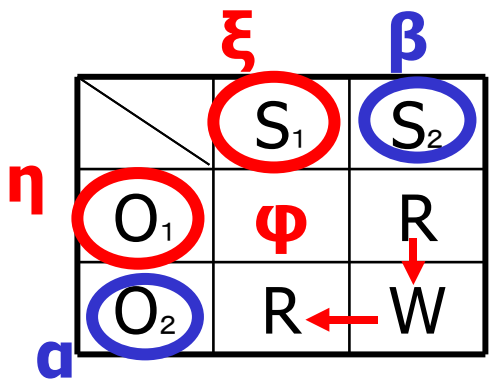
情報フィルタルール(1)

SとOに与える5つの属性

- **競合属性**
  コミュニティ間において、競合属性であるとき、競合属性間のSubjectはObjectに対してアクセス禁止となる。
- **階層属性**
  セキュリティレベルによって、Subjectの情報取得を制御する。
- **所有属性**
  所有属性であるObjectには書き込み禁止となる。但し、同じ所有属性ならば可能となる。
- **役割属性**
  役割によって，Subjectの権限管理する事が可能となる。
- **プライベート属性**
  プライベート属性のObjectにはアクセス禁止となる。但し、同じプライベート属性のSubject、もしくはアクセス許可により可能となる。

情報フィルタルール(2)

5つの属性の組み合わせによる4つの情報フィルタ



O1の情報漏えいを防止するため、
η、ξ、α、βの属性の関係から
(1)～(4)のフィルタを選択し、
Sに与える権限を制限する。

(1)

	S1	S2
O1	φ	φ
O2	R	W

S2のO1へ対するRead権限を削除

(2)

	S1	S2
O1	φ	R
O2	R	φ

S2のO2へ対するWrite権限を削除

(3)

	S1	S2
O1	φ	R
O2	φ	W

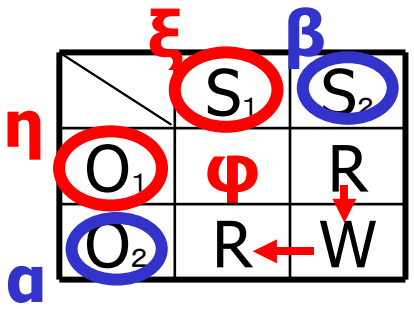
S1のO2へ対するRead権限を削除

(4)

	S1	S2
O1	R	R
O2	R	W

S1のO2へ対するRead権限を追加

情報フィルタルール(3)



(1)

	S1	S2
O1	φ	φ
O2	R	W

(2)

	S1	S2
O1	φ	R
O2	R	φ

(3)

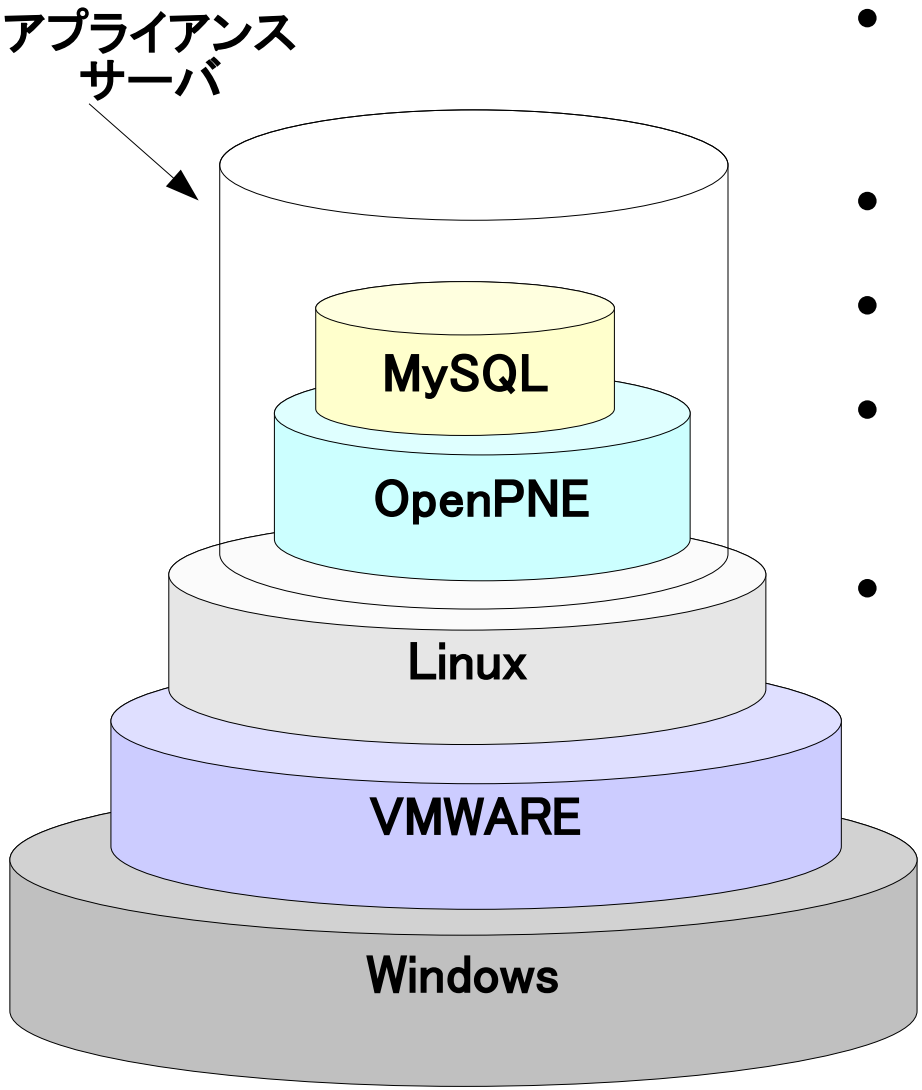
	S1	S2
O1	φ	R
O2	φ	W

(4)

	S1	S2
O1	R	R
O2	R	W

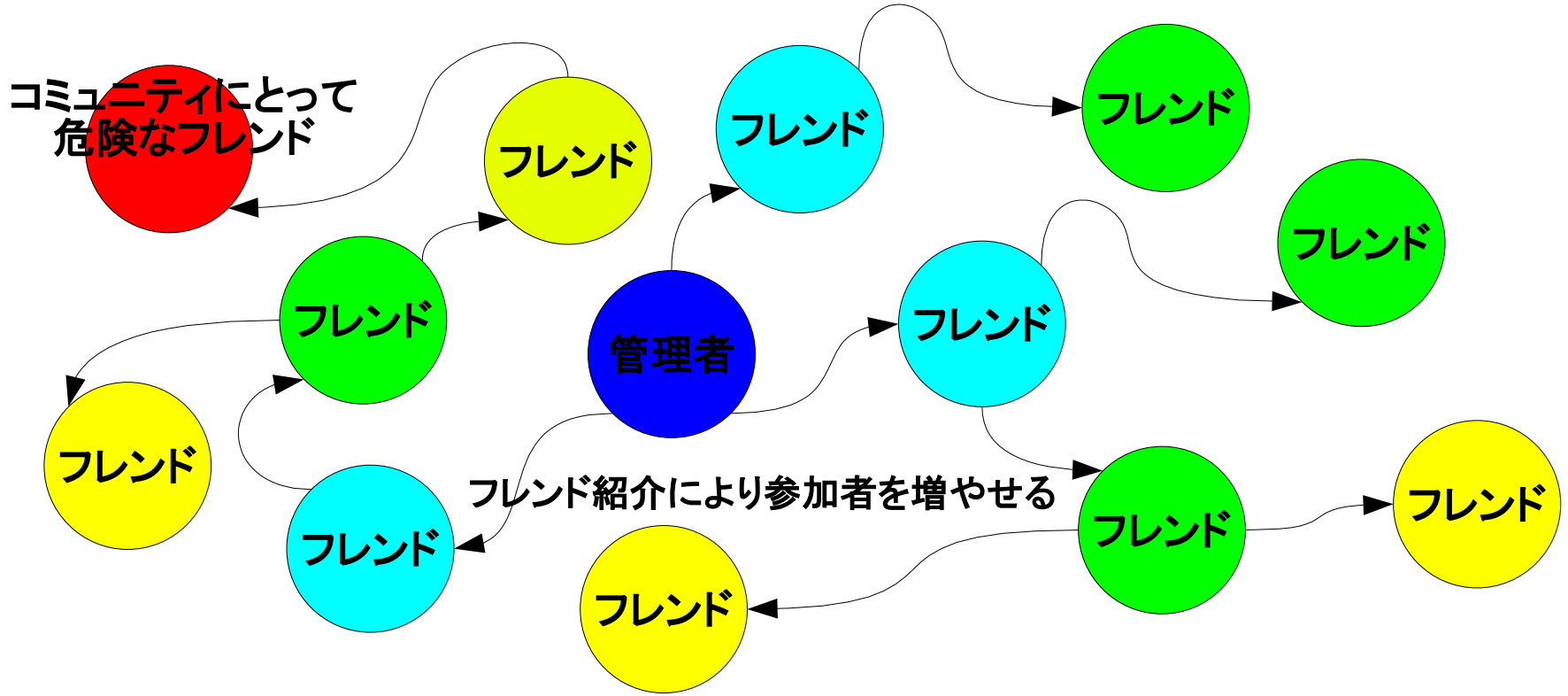
- **S1**と**S2**が競合の場合（3）を選択する。
- **S**と**O**にセキュリティレベルが設定されていた場合、（2）～（4）を選択する。
- **S1**、**S2**の役割によるアクセス制限のある場合、I
  - 役割関係から**O1**、**O2**が**S2**によって管理される場合、（3）を選択する。
  - 役割関係から**O1**、**O2**が**S2**によって管理され、かつ、**S1**が**O2**を**Read**する必要がある場合、仮想的に**S2'**を付加し、（1）を選択する。**S2'**は**O1**を**Read**、**S2**は**O1**をφとする。但し、I**S2**と**S2'**は同一主体であるとする。
- **O1**が**S2**のプライベート情報の場合、（3）を選択する。
- **O1**が**S2**の所有の刻印がしてあるならば、他の**Object**に書き込んではならない、かつその**Object**に書き込んではならない。
- 該当するものが無ければ、（4）を選択する。

実験環境



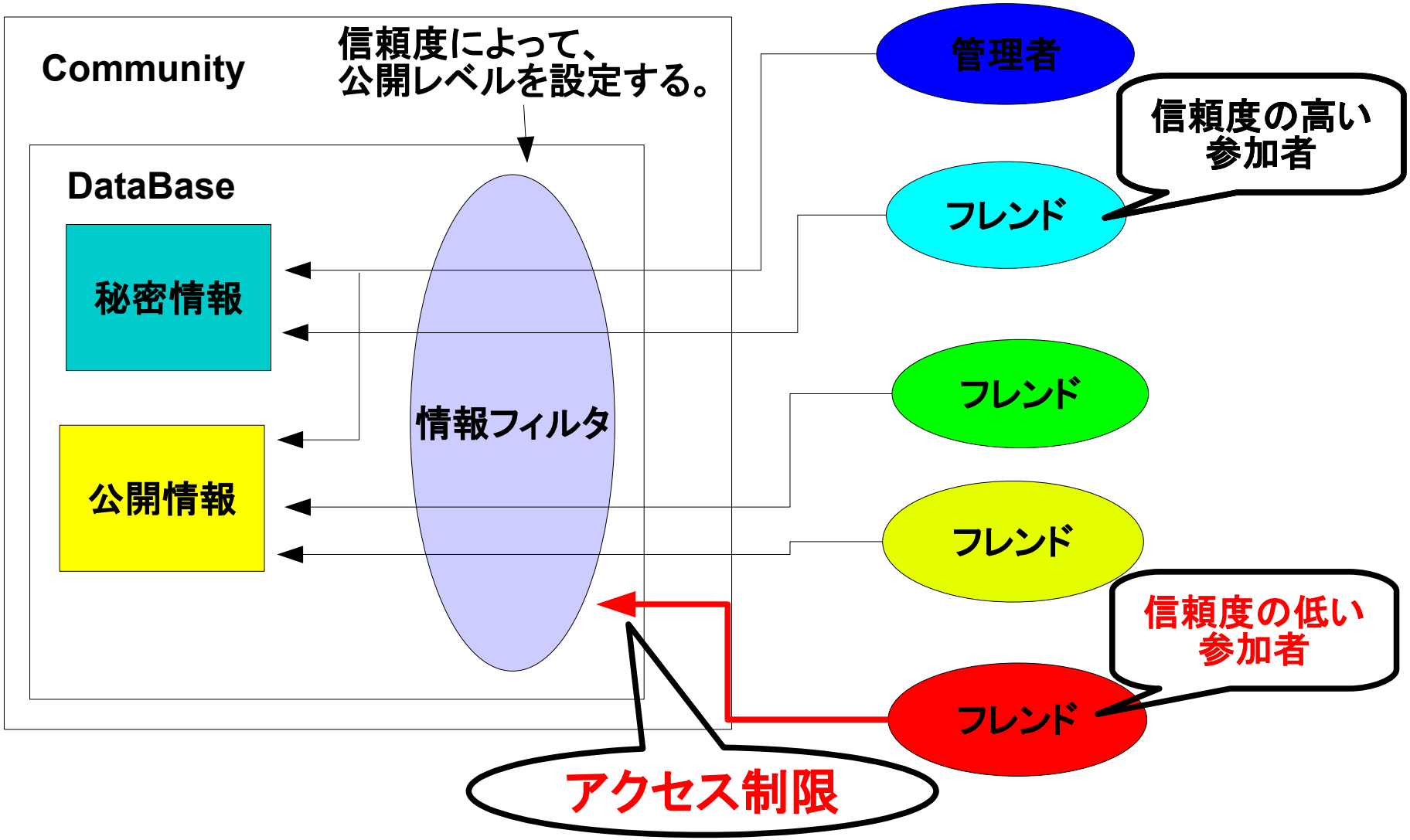
- ホストOS(Windows)
- 仮想マシンソフトウェアであるVMware
- ゲストOS(Linux)
- SNSエンジンであるOpenPNE
- OpenPNE用のアプライアンスサーバ
- データベースであるMySQL
  使用言語：PHP

OpenPNEにおけるフレンド紹介における信頼度の低下



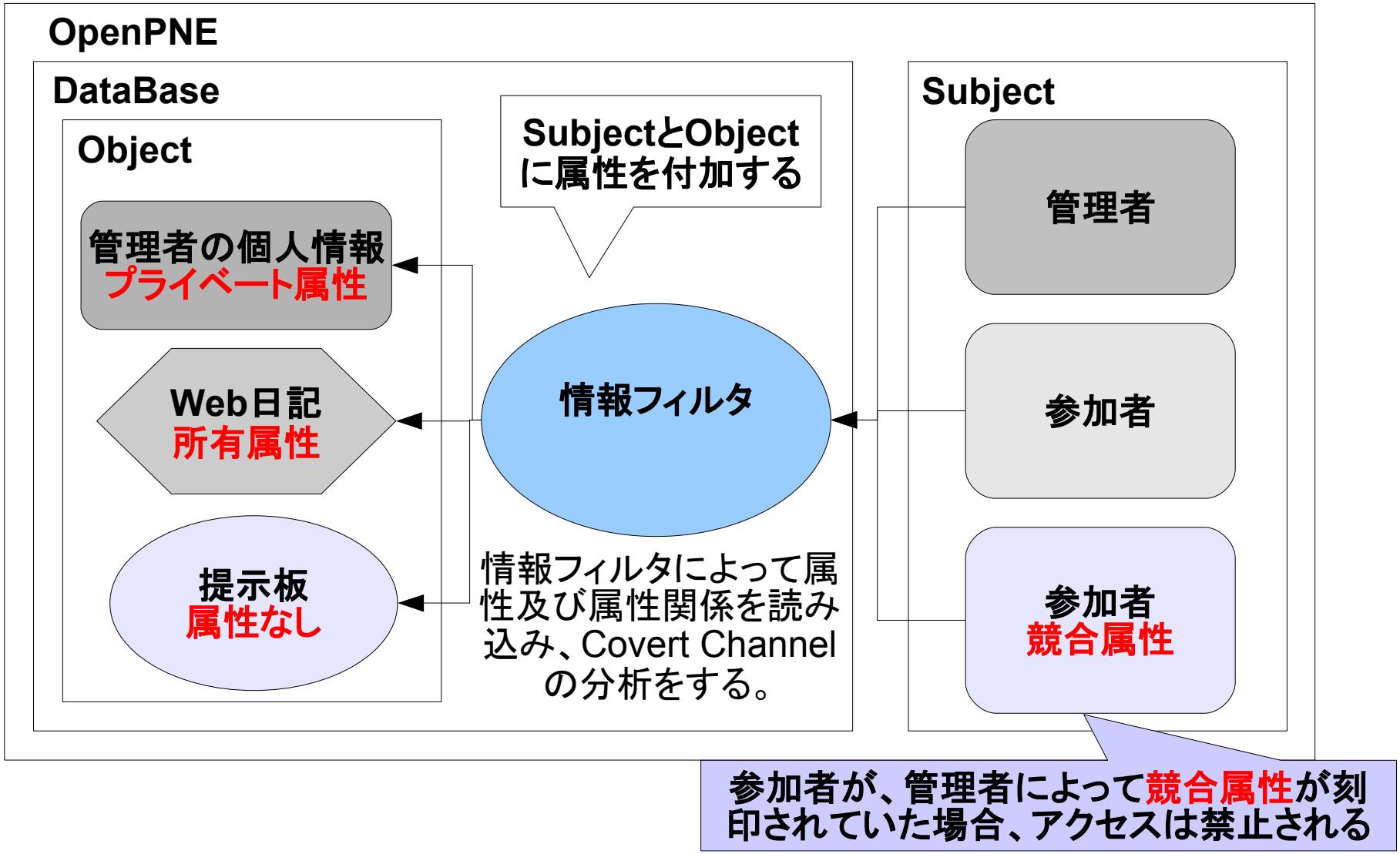
- OpenPNEにある「友人の招待」機能はフレンドに紹介し、そのSNSの参加者を増やすことができる。
- フレンド紹介によって、コミュニティへの参加人数は増えていくが、そのコミュニティにとって危険な人物に紹介される可能性がある。

信頼度による情報フィルタの適用



属性関係に基づいた情報フィルタの適用

管理者（subject）によって刻印された情報（object）は、
情報フィルタによって属性を付加される



結果

- OpenPNEにおけるCovert Channelのパターンを解析した。
- 信頼度におけるフィルタリングを実装した。
- 主体と客体の属性に基づくフィルタリングの提案をした。

今後の課題

- 提案したフィルタリングの実装。
- SNSで発生すると思われるCovert Channelのパターンを、情報フィルタに反映させ、主体と客体の属性による高度な情報フィルタとしたい。