WEAKENING NON-INTERFERENCE ON DATABASES

ISABELLA MASTROENI AND ROSALBA ROSSATO

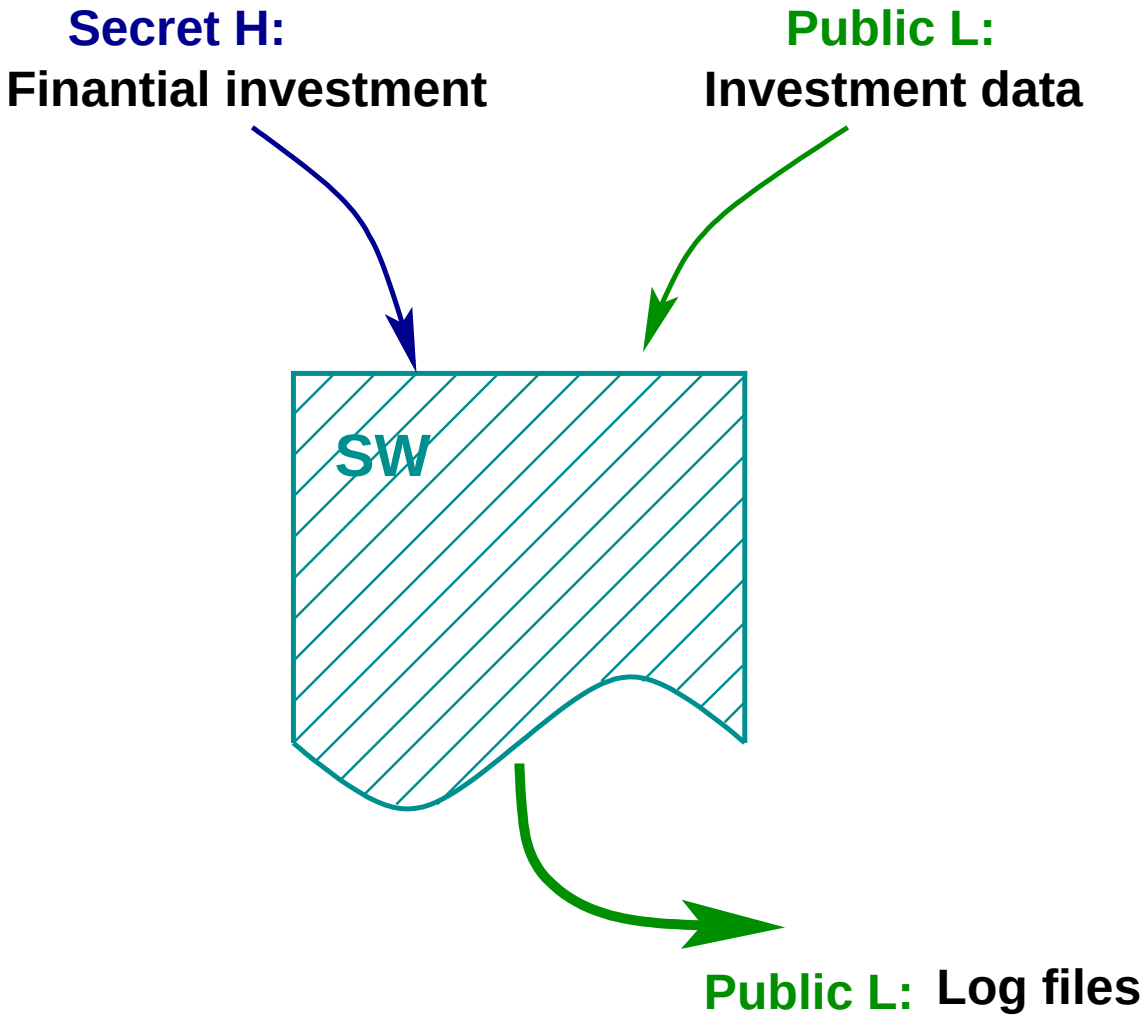
Dip. di Informatica
Università di Verona

Dip. di Elettronica e Informazione
Politecnico di Milano

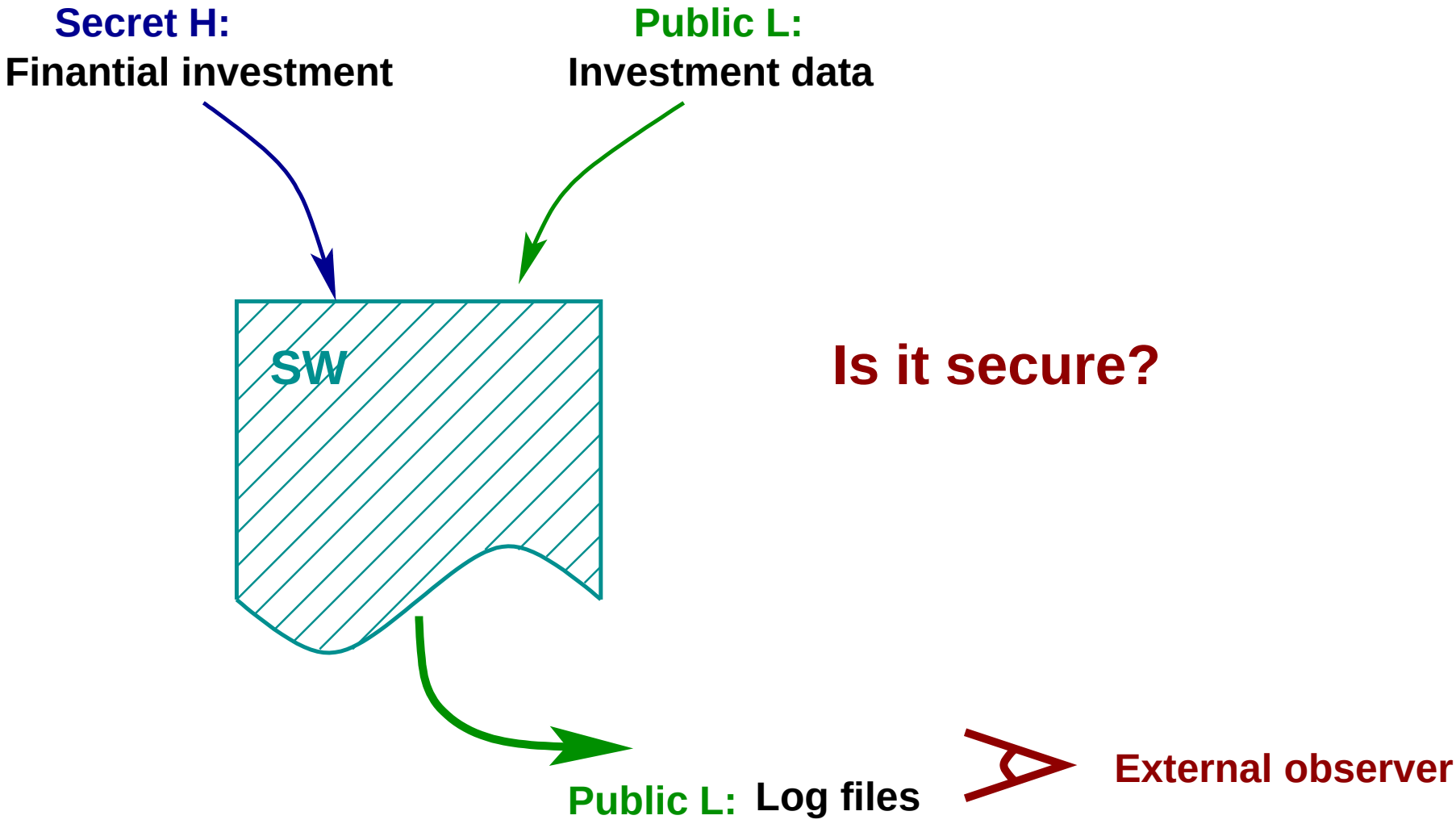
CERICS 2006

London, July 20, 2006

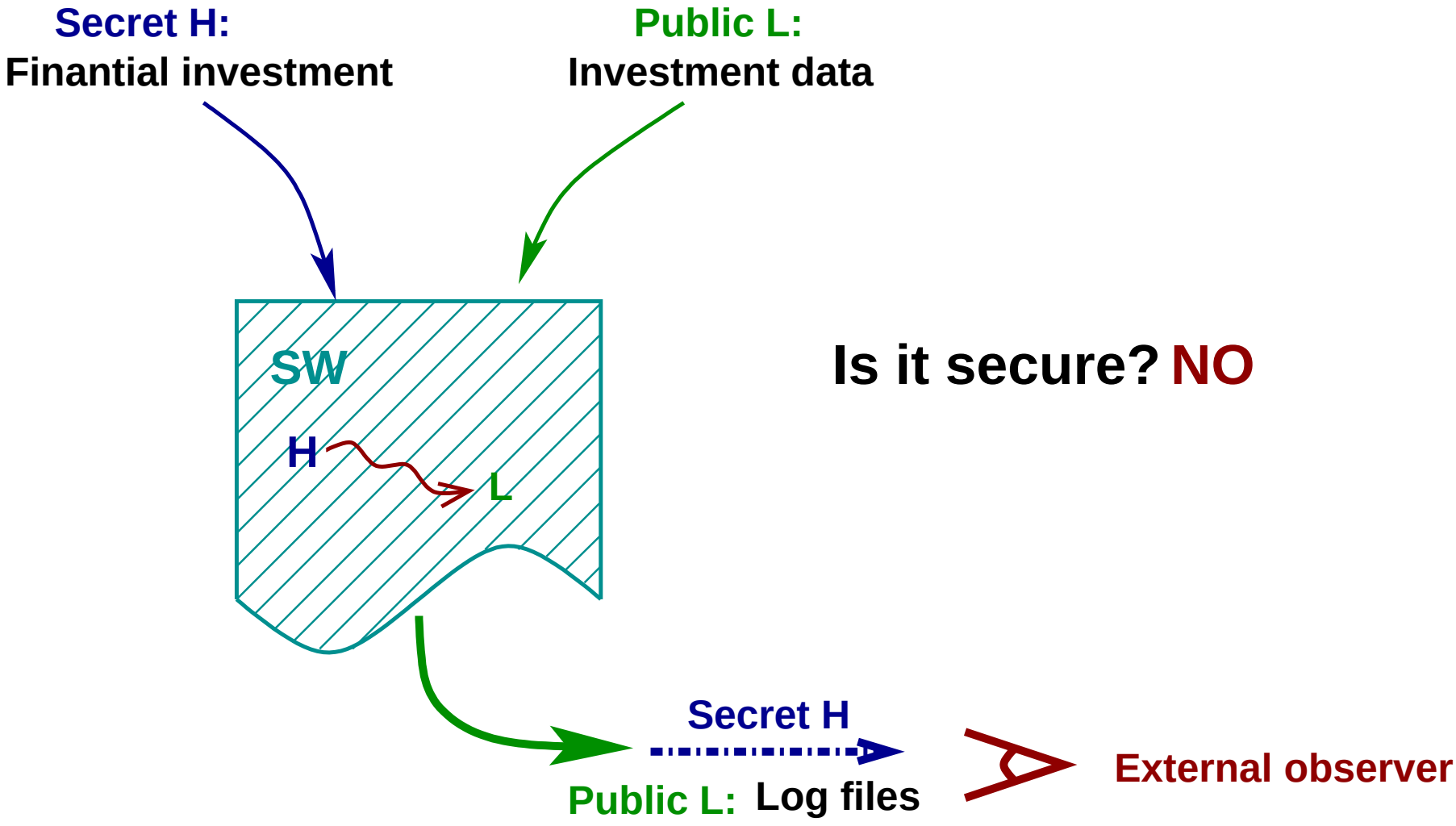
The Problem: Non-Interference



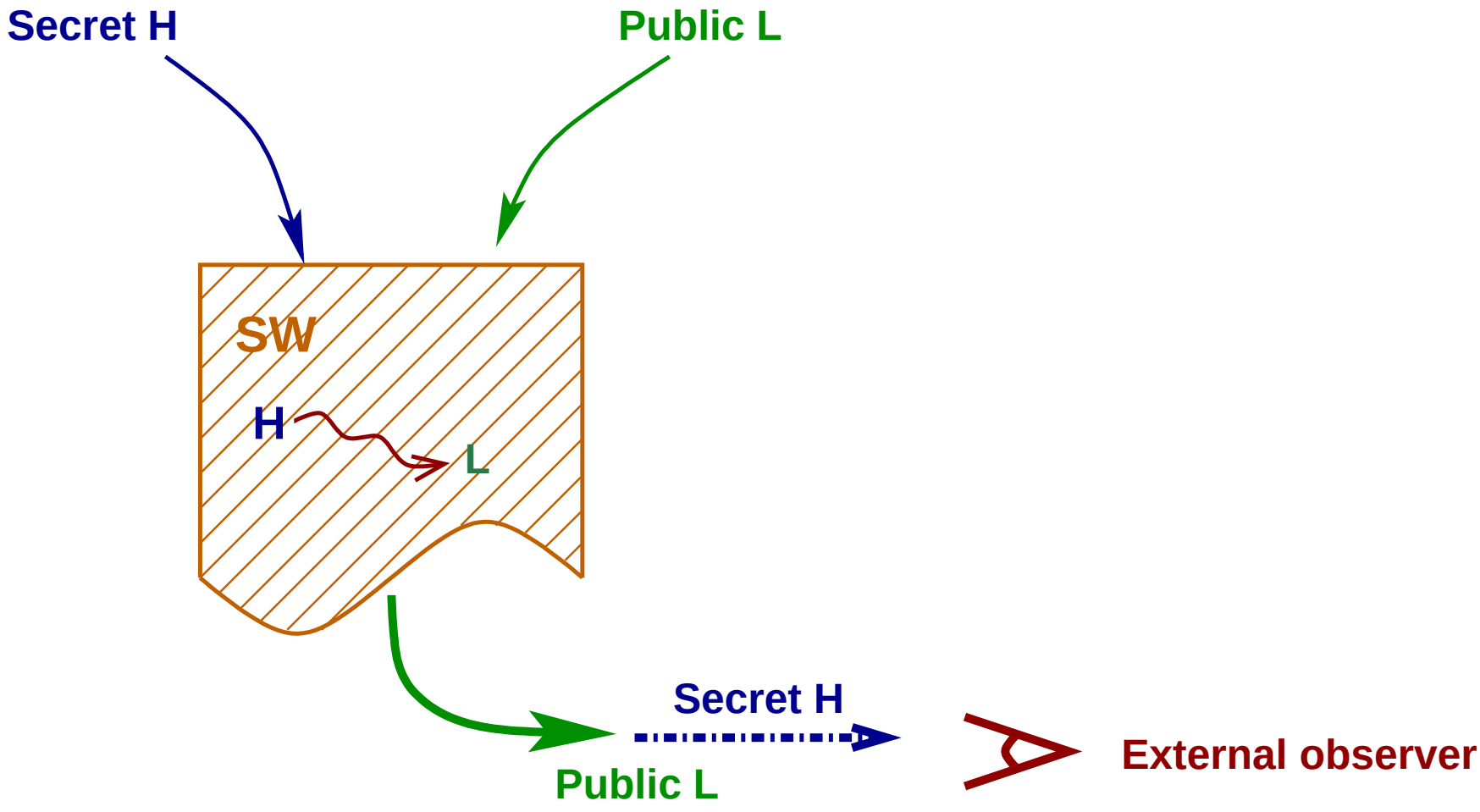
The Problem: Non-Interference



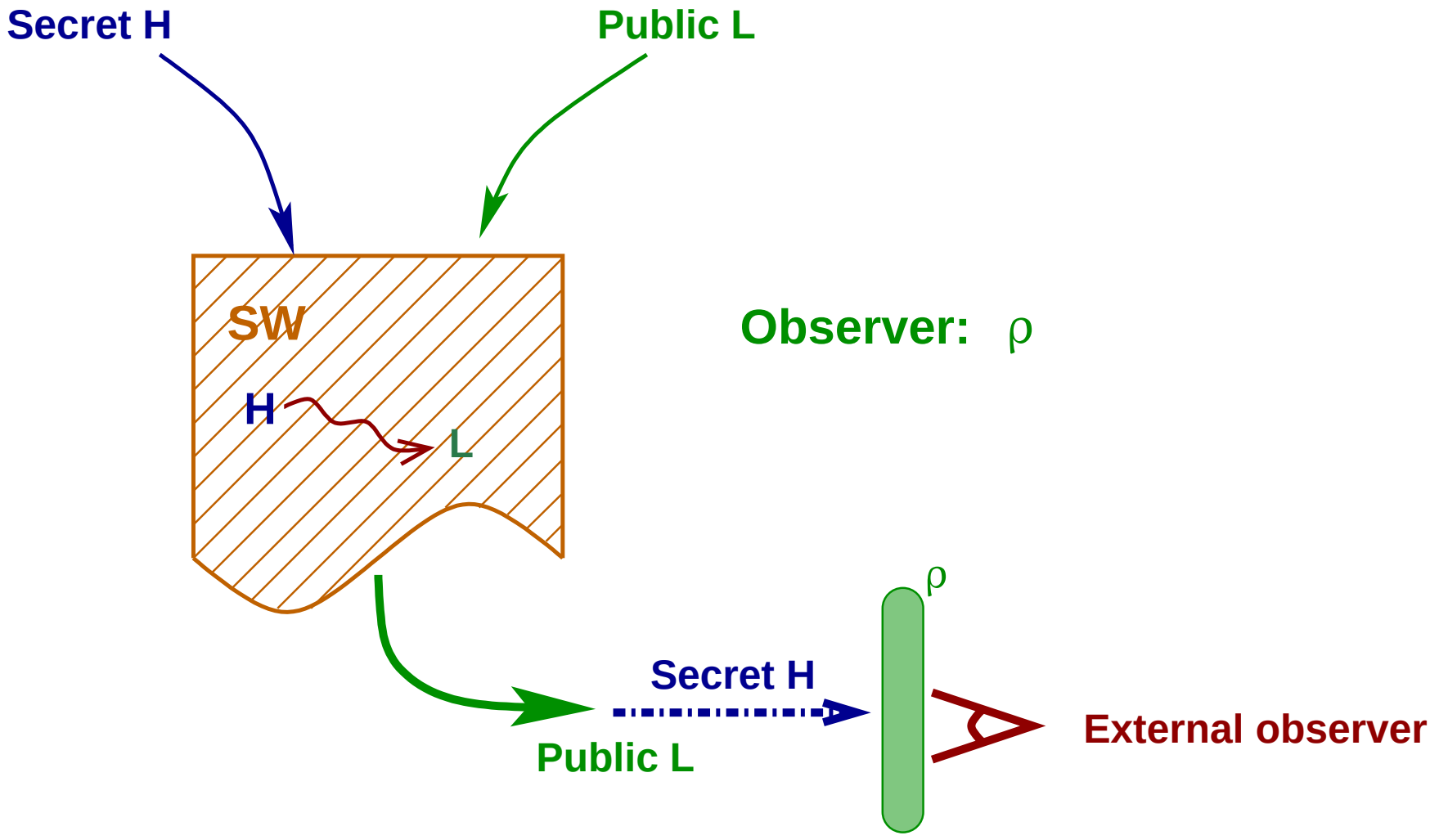
The Problem: Non-Interference



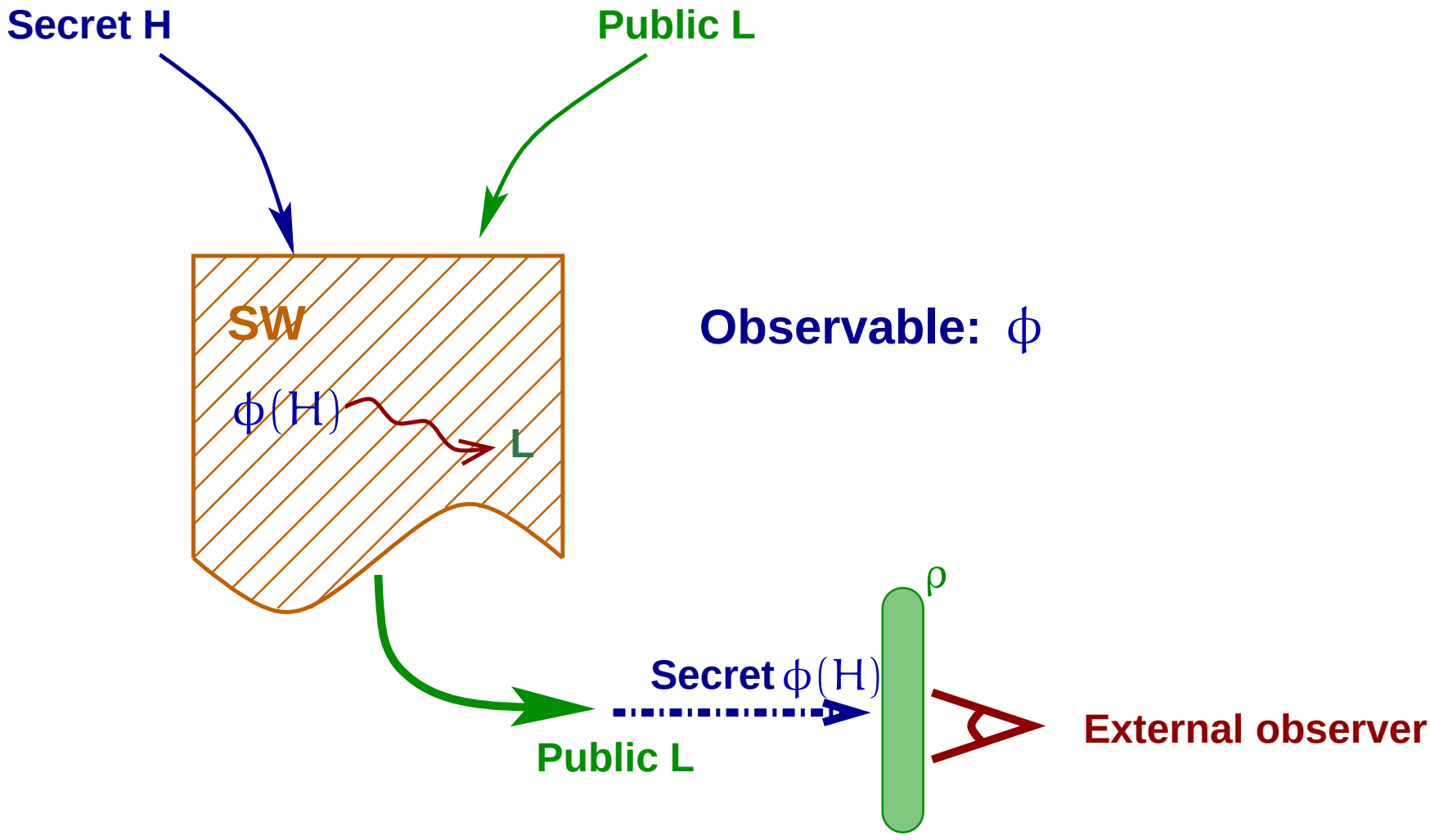
Abstracting Non-Interference



Abstracting Non-Interference



Abstracting Non-Interference



Abstract Interpretation: The idea

Abstract interpretation formalizes the notion of abstraction:

Abstract Interpretation: The idea

Abstract interpretation formalizes the notion of abstraction:

- Given a concrete domain C:
  Lattice of abstractions of $C \equiv$ Lattice of *uco* on $\wp(C)$

Abstract Interpretation: The idea

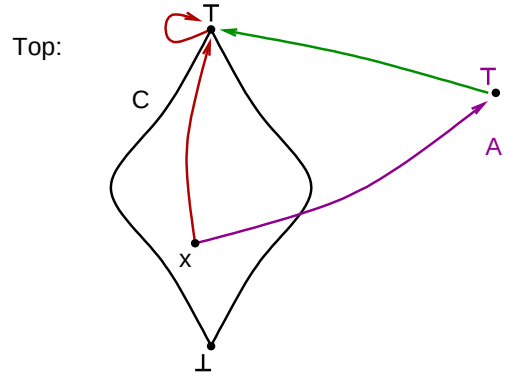
Abstract interpretation formalizes the notion of abstraction:

- Given a concrete domain C:
  Lattice of abstractions of $C \equiv$ Lattice of *uco* on $\wp(C)$
- Order: Relative precision;

Abstract Interpretation: The idea

Abstract interpretation formalizes the notion of abstraction:

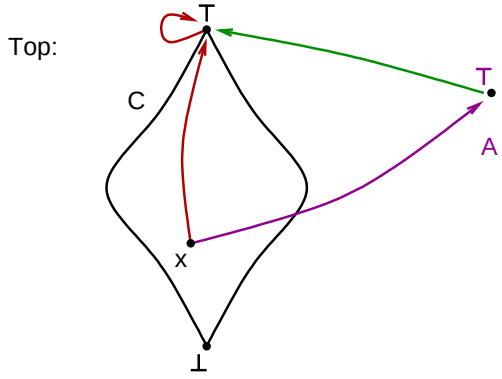
- Given a concrete domain C:
  Lattice of abstractions of $C \equiv$ Lattice of *uco* on $\wp(C)$
- Order: Relative precision;

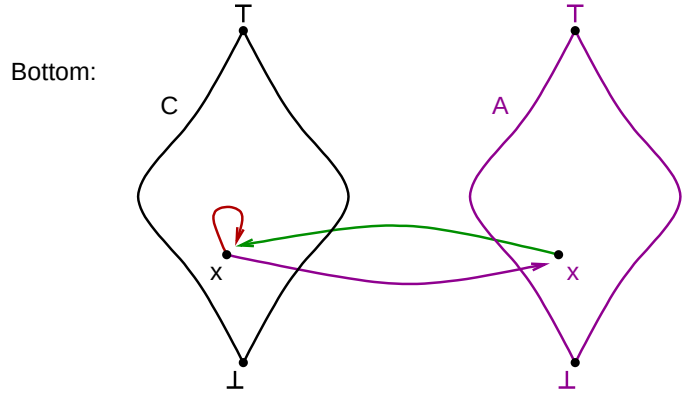


Abstract Interpretation: The idea

Abstract interpretation formalizes the notion of abstraction:

- Given a concrete domain C:
  Lattice of abstractions of $C \equiv$ Lattice of *uco* on $\wp(C)$
- Order: Relative precision;

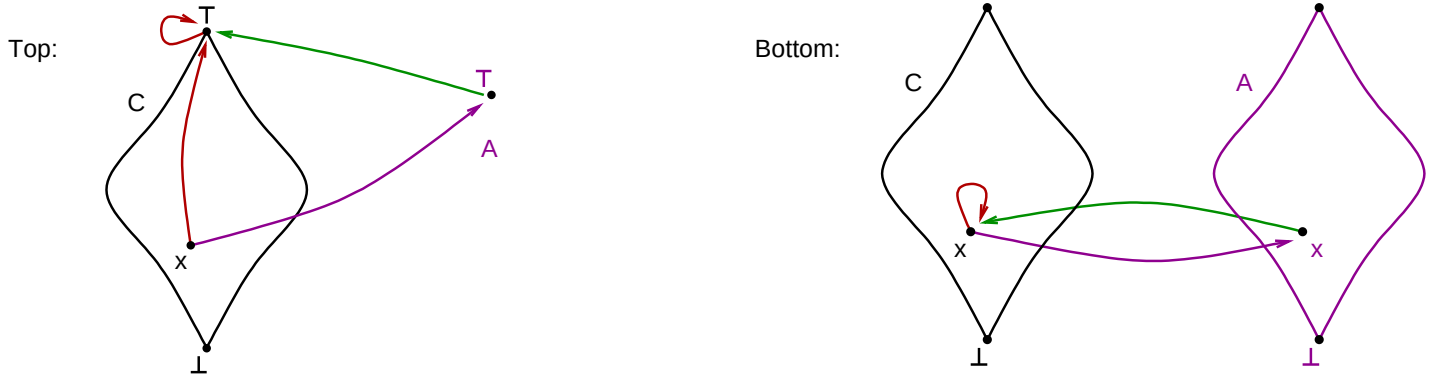




Abstract Interpretation: The idea

Abstract interpretation formalizes the notion of abstraction:

- Given a concrete domain C:
  Lattice of abstractions of $C \equiv$ Lattice of *uco* on $\wp(C)$
- Order: Relative precision;



MATHEMATICAL STRUCTURE

⇓

EACH CONCRETE ELEMENT HAS ITS *best* ABSTRACTION!

ANI for imperative programs

Results about abstract non-interference for imperative programs:

ANI for imperative programs

Results about abstract non-interference for imperative programs:

- Characterization of the strongest harmless attacker [Giacobazzi & Mastroeni ’04];

ANI for imperative programs

Results about abstract non-interference for imperative programs:

- Characterization of the strongest harmless attacker [Giacobazzi & Mastroeni ’04];
- Characterization of the maximal amount of information disclosed by a fixed attacker [Giacobazzi & Mastroeni ’04];

ANI for imperative programs

Results about abstract non-interference for imperative programs:

- Characterization of the strongest harmless attacker [Giacobazzi & Mastroeni '04];
- Characterization of the maximal amount of information disclosed by a fixed attacker [Giacobazzi & Mastroeni '04];
- Derivation of a logic for ANI [Giacobazzi & Mastroeni '04];

ANI for imperative programs

Results about abstract non-interference for imperative programs:

- Characterization of the strongest harmless attacker [Giacobazzi & Mastroeni '04];
- Characterization of the maximal amount of information disclosed by a fixed attacker [Giacobazzi & Mastroeni '04];
- Derivation of a logic for ANI [Giacobazzi & Mastroeni '04];
- Characterization of ANI as a problem of completeness in the abstract interpretation framework [Giacobazzi & Mastroeni '05];

ANI for imperative programs

Results about abstract non-interference for imperative programs:

- Characterization of the strongest harmless attacker [Giacobazzi & Mastroeni ’04];
- Characterization of the maximal amount of information disclosed by a fixed attacker [Giacobazzi & Mastroeni ’04];
- Derivation of a logic for ANI [Giacobazzi & Mastroeni ’04];
- Characterization of ANI as a problem of completeness in the abstract interpretation framework [Giacobazzi & Mastroeni ’05];
- Characterization of ANI in terms of PERs instead of closure operators [Hunt & Mastroeni ’05];

Database Security

- One of the main task of information security policies is to enforce confidentiality of data
  - Protection of data against unauthorized disclosure of sensitive information

Database Security

- One of the main task of information security policies is to enforce confidentiality of data
  - Protection of data against unauthorized disclosure of sensitive information
- The standard approach to protect multilevel secure databases from violations of confidentiality is mechanism of *mandatory access control*
  - Data and users are classified in term of security classes

Database Security

- One of the main task of information security policies is to enforce confidentiality of data
  - Protection of data against unauthorized disclosure of sensitive information
- The standard approach to protect multilevel secure databases from violations of confidentiality is mechanism of *mandatory access control*
  - Data and users are classified in term of security classes
- Unauthorized releases of information may occur due to “implicit dependencies” between private and public information, i.e. *inference channels*

Relational Databases: Background

- RELATIONAL SCHEMA $R(U)$: R is the relation name and $U = \{A_1, A_2, , A_n\}$ is the set of attributes
- DOMAIN OF VALUES: $dom(A_i) = D_i \ \ \forall A_i \in U$
- TUPLE OVER X: $X \subseteq U$, function from X into $dom(X) = \cup_{A \in X} dom(A)$
- RELATION r OVER U: set of tuples over U
- $t[A_i]$ represents the value of the attribute A_i on the tuple t
- DATABASE SCHEMA $R = \{R_1(U_1), \ldots, R_k(U_k)\}$: $R_i(U_i)$ is the relational schema of the i^{th} relation of the database

Relational Databases: Example

EMPLOYEE

SSN	Name	BDate	Address	Sex	Salary	DNO
104	Smith	25/05/70	5th Avenue	M	1.500	4
124	Benson	10/04/68	Castle Spring	F	1.800	5
345	Alicia	19/07/68	980 Dallas	F	2.000	5
555	Borg	10/11/37	450 Stone	M	2.500	1

DEPARTMENT

DNumber	DName
1	Headquarters
4	Administration
5	Research

Relational Databases: Views and Queries

- Select the name and the code of employees belonging to the research department:

```
SELECT Name, SSN
FROM EMPLOYEE, DEPARTMENT
WHERE DNO=DNumber AND DName='Research'
```

Relational Databases: Views and Queries

- Select the name and the code of employees belonging to the research department:

```
SELECT Name, SSN
FROM EMPLOYEE, DEPARTMENT
WHERE DNO=DNumber AND DName='Research'
```

Result:

Name	SSN
Benson	124
Alicia	345

Relational Databases: Views and Queries

- Select the name and the code of employees belonging to the research department:

```
SELECT Name, SSN
FROM EMPLOYEE, DEPARTMENT
WHERE DNO=DNumber AND DName='Research'
```

- An SQL view is a single table that is derived from other tables; it does not necessarily exists in physical form

Relational Databases: Views and Queries

- Select the name and the code of employees belonging to the research department:

```
SELECT Name, SSN
FROM EMPLOYEE, DEPARTMENT
WHERE DNO=DNumber AND DName='Research'
```

- An SQL view is a single table that is derived from other tables; it does not necessarily exists in physical form

```
CREATE VIEW WORKS_ON
AS SELECT Name, SSN, DName
   FROM EMPLOYEE,DEPARTMENT
   WHERE DNO=DNumber
```

Relational Databases: Views and Queries

- Select the name and the code of employees belonging to the research department:

```
SELECT Name, SSN
FROM EMPLOYEE, DEPARTMENT
WHERE DNO=DNumber AND DName='Research'
```

- An SQL view is a single table that is derived from other tables; it does not necessarily exists in physical form

```
CREATE VIEW WORKS_ON
AS SELECT Name, SSN, DName
   FROM EMPLOYEE,DEPARTMENT
   WHERE DNO=DNumber
```

WORKS_ON

Name	SSN	DName
Smith	104	Administration
Benson	124	Research
Alicia	345	Research
Borg	555	Headquarters

Standard Security Approaches on DB

- Access control policy: defines the collection of access privileges and access rules
  - Discretionary access control policy: specifies users' privileges relating to different system resources

Standard Security Approaches on DB

- Access control policy: defines the collection of access privileges and access rules
  - Discretionary access control policy: specifies users' privileges relating to different system resources
    - Account level: privileges of each account are specified
    - Relation level: it is possible to control the privilege to access each individual relation or view in the DB

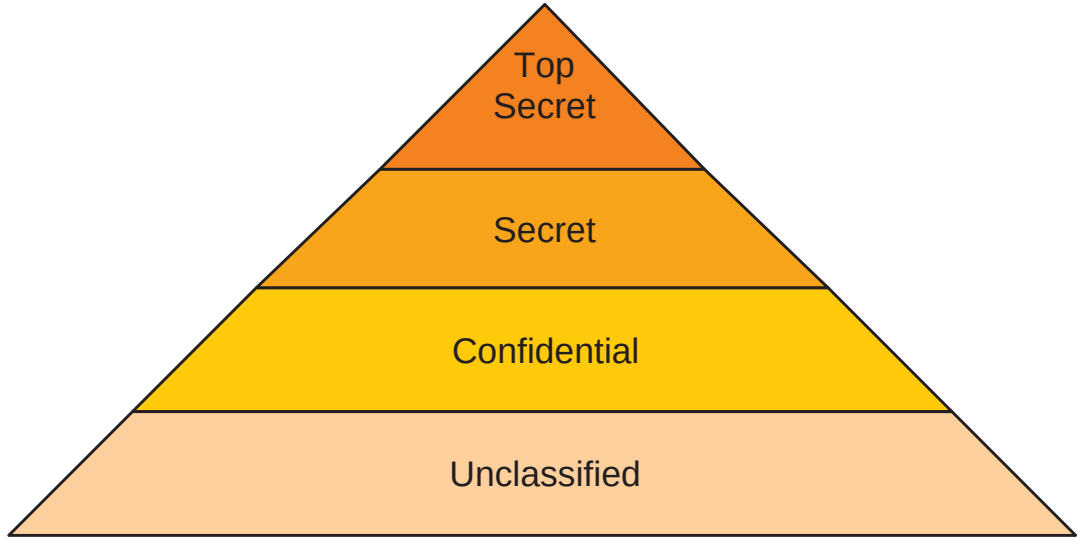
Standard Security Approaches on DB

- Access control policy: defines the collection of access privileges and access rules
  - Discretionary access control policy: specifies users' privileges relating to different system resources
    - Account level: privileges of each account are specified
    - Relation level: it is possible to control the privilege to access each individual relation or view in the DB
  - Views are an important discretionary authorization mechanism

All-or-nothing method

Standard Security Approaches on DB

- Access control policy: defines the collection of access privileges and access rules
  - Discretionary access control policy: specifies users' privileges relating to different system resources
  - Mandatory access control policy: defines user access to system resources using the user *security clearance* and the *security classification* of the resource



The idea

- NON INTERFERENCE: The results of *admissible* queries have not to depend on confidential data in the DB!

The idea

- NON INTERFERENCE: The results of *admissible* queries have not to depend on confidential data in the DB!
- ABSTRACT NON INTERFERENCE: *Properties* of admissible queries have not to depend on particular *properties* of confidential data in the DB!

The idea

- NON INTERFERENCE: The results of *admissible* queries have not to depend on confidential data in the DB!
- ABSTRACT NON INTERFERENCE: *Properties* of admissible queries have not to depend on particular *properties* of confidential data in the DB!
- HOW CAN WE APPLY ANI TO DB WHERE DATA ARE CLASSIFIED AS PUBLIC AND PRIVATE?

The idea

- NON INTERFERENCE: The results of *admissible* queries have not to depend on confidential data in the DB!
- ABSTRACT NON INTERFERENCE: *Properties* of admissible queries have not to depend on particular *properties* of confidential data in the DB!
- HOW CAN WE APPLY ANI TO DB WHERE DATA ARE CLASSIFIED AS PUBLIC AND PRIVATE?

VARIABLES $\rightsquigarrow$ ATTRIBUTES

STATES $\rightsquigarrow$ TUPLES

PROGRAMS $\rightsquigarrow$ QUERIES

Non-Interference on DB

- Salary is a private data
  - It is not possible to retrieve (and use) information about a salary
  - Given the name of an employee, it is not possible to retrieve his/her salary

Name	SSN	BDate	Address	Sex	Salary	DNo
Smith	104	25/05/70	5th Avenue	M	1.500	4
Benson	124	10/04/68	Castle Spring	F	1.800	5
Alicia	345	19/07/68	980 Dallas	F	2.000	5
Borg	555	10/11/37	450 Stone	M	2.500	1

Non-Interference on DB

- Salary is a private data
- View on the relation EMPLOYEE on its public data corresponds to an abstraction over the Salary attribute

Non-Interference on DB

- Salary is a private data
- View on the relation EMPLOYEE on its public data corresponds to an abstraction over the Salary attribute

```
CREATE VIEW PUBLIC_DATA
AS SELECT Name, SSN, DName, Address, Sex, DNO
   FROM EMPLOYEE
```

Name	SSN	BDate	Address	Sex	DNo
Smith	104	25/05/70	5th Avenue	M	4
Benson	124	10/04/68	Castle Spring	F	5
Alicia	345	19/07/68	980 Dallas	F	5
Borg	555	10/11/37	450 Stone	M	1

Non-Interference on DB

- Salary is a private data
- View on the relation EMPLOYEE on its public data corresponds to an abstraction over the Salary attribute

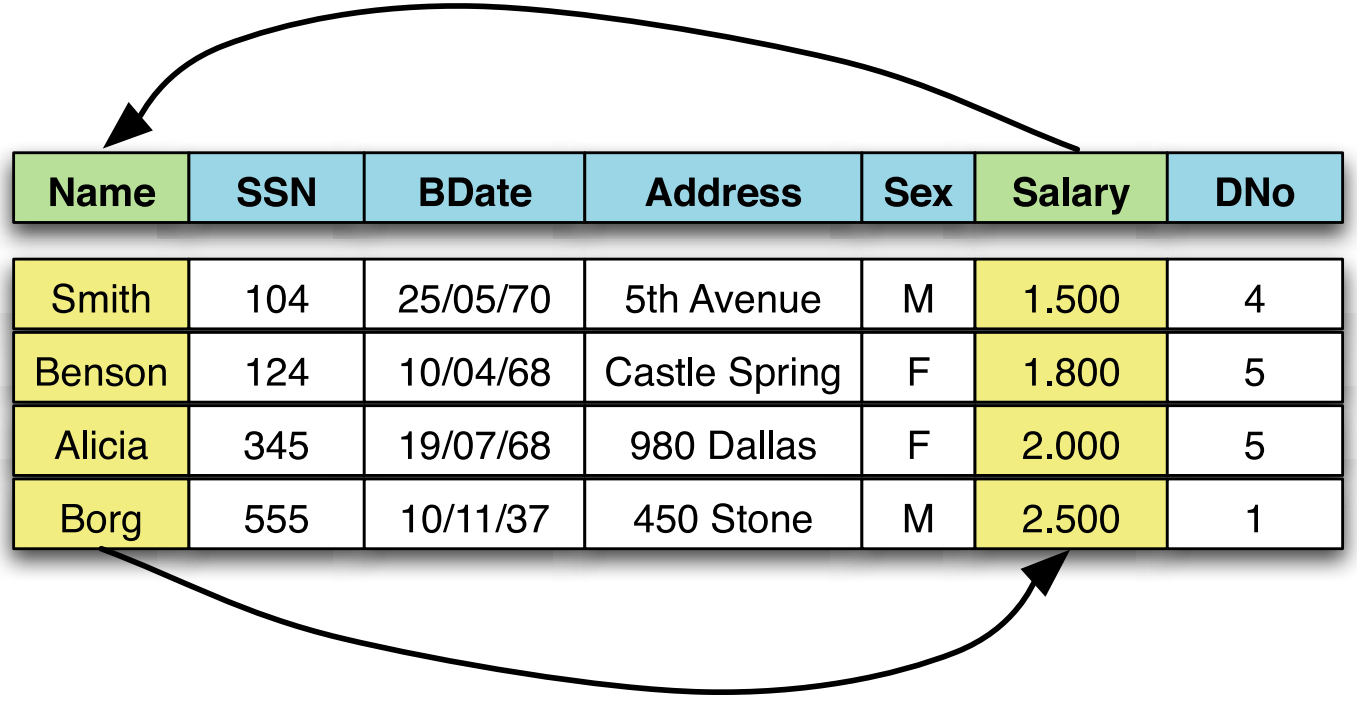
Name	SSN	BDate	Address	Sex	Salary	DNo
Smith	104	25/05/70	5th Avenue	M	**T**	4
Benson	124	10/04/68	Castle Spring	F	**T**	5
Alicia	345	19/07/68	980 Dallas	F	**T**	5
Borg	555	10/11/37	450 Stone	M	**T**	1

Abstract Non-Interference on DB (1)

- Salary is observable, but not the relation with Name.
  - It is possible to retrieve (e.g. statistics...) any information about a salary
  - Given the name of an employee, it is not possible to retrieve his/her salary
- For statistics, the name of employees is abstracted
- The corresponding view removes the attribute Name

Abstract Non-Interference on DB (1)

- Salary is observable, but not the relation with Name.



Name	SSN	BDate	Address	Sex	Salary	DNo
Smith	104	25/05/70	5th Avenue	M	1.500	4
Benson	124	10/04/68	Castle Spring	F	1.800	5
Alicia	345	19/07/68	980 Dallas	F	2.000	5
Borg	555	10/11/37	450 Stone	M	2.500	1

Abstract Non-Interference on DB (1)

- Salary is observable, but not the relation with Name.

```
CREATE VIEW PUBLIC_DATA2
AS SELECT SSN, BDame, Address, Sex, Salary, DNO
   FROM EMPLOYEE
```

SSN	BDate	Address	Sex	Salary	DNo
104	25/05/70	5th Avenue	M	1.500	4
124	10/04/68	Castle Spring	F	1.800	5
345	19/07/68	980 Dallas	F	2.000	5
555	10/11/37	450 Stone	M	2.500	1

Abstract Non-Interference on DB (1)

- Salary is observable, but not the relation with Name.

Name	SSN	BDate	Address	Sex	Salary	DNo
T	104	25/05/70	5th Avenue	M	1.500	4
T	124	10/04/68	Castle Spring	F	1.800	5
T	345	19/07/68	980 Dallas	F	2.000	5
T	555	10/11/37	450 Stone	M	2.500	1

Abstract Non-Interference on DB (2)

- Salary is *partially* observable.
  - It is possible to retrieve some properties about, e.g. intervals, signs, parity etc.
  - Given the name of an employee, it is possible to retrieve some information about his/her salary
- Transform DB by abstracting values of Salary

Abstract Non-Interference on DB (2)

- Salary is *partially* observable.
- ABSTRACT SQL:

```
CREATE VIEW PUBLIC_DATA3
AS SELECT Name, SSN, BDame, Address, Sex,
          Int(Salary), DNO
   FROM EMPLOYEE
```

Name	SSN	BDate	Address	Sex	Salary	DNo
Smith	104	25/05/70	5th Avenue	M	[1.500,2.000[	4
Benson	124	10/04/68	Castle Spring	F	[1.500,2.000[	5
Alicia	345	19/07/68	980 Dallas	F	[2.000,2.500[	5
Borg	555	10/11/37	450 Stone	M	[2.500,3.000[	1

Future directions

We aim to develop these ideas in the following directions:

Future directions

We aim to develop these ideas in the following directions:

- Characterization of the *strongest observation* that makes a view secure [The strongest harmless attacker in ANI];

Future directions

We aim to develop these ideas in the following directions:

- Characterization of the *strongest observation* that makes a view secure [The strongest harmless attacker in ANI];
- Characterization of the *maximal amount* of information disclosed about a private attribute [The maximal amount of information disclosed in ANI];

Future directions

We aim to develop these ideas in the following directions:

- Characterization of the *strongest observation* that makes a view secure [The strongest harmless attacker in ANI];
- Characterization of the *maximal amount* of information disclosed about a private attribute [The maximal amount of information disclosed in ANI];
- Characterization of the views that can brake a confidentiality policy on a fixed DB [The closest program that satisfies a fixed policy].

Future directions

We aim to develop these ideas in the following directions:

- Characterization of the *strongest observation* that makes a view secure [The strongest harmless attacker in ANI];
- Characterization of the *maximal amount* of information disclosed about a private attribute [The maximal amount of information disclosed in ANI];
- Characterization of the views that can brake a confidentiality policy on a fixed DB [The closest program that satisfies a fixed policy].

ANY IDEA/SUGGESTION/COMMENT IS REALLY WELCOME!