Saint Vincent College

Department of Computing and Information Science

Security-Typed Languages
Jif Programming

Fr. Boniface Hicks, OSB

Special Lecture

"There is nothing covered up that will not be exposed and nothing secret that will not be made known." Luke 12:2

A criminal? or a crime?

A criminal? or a crime?

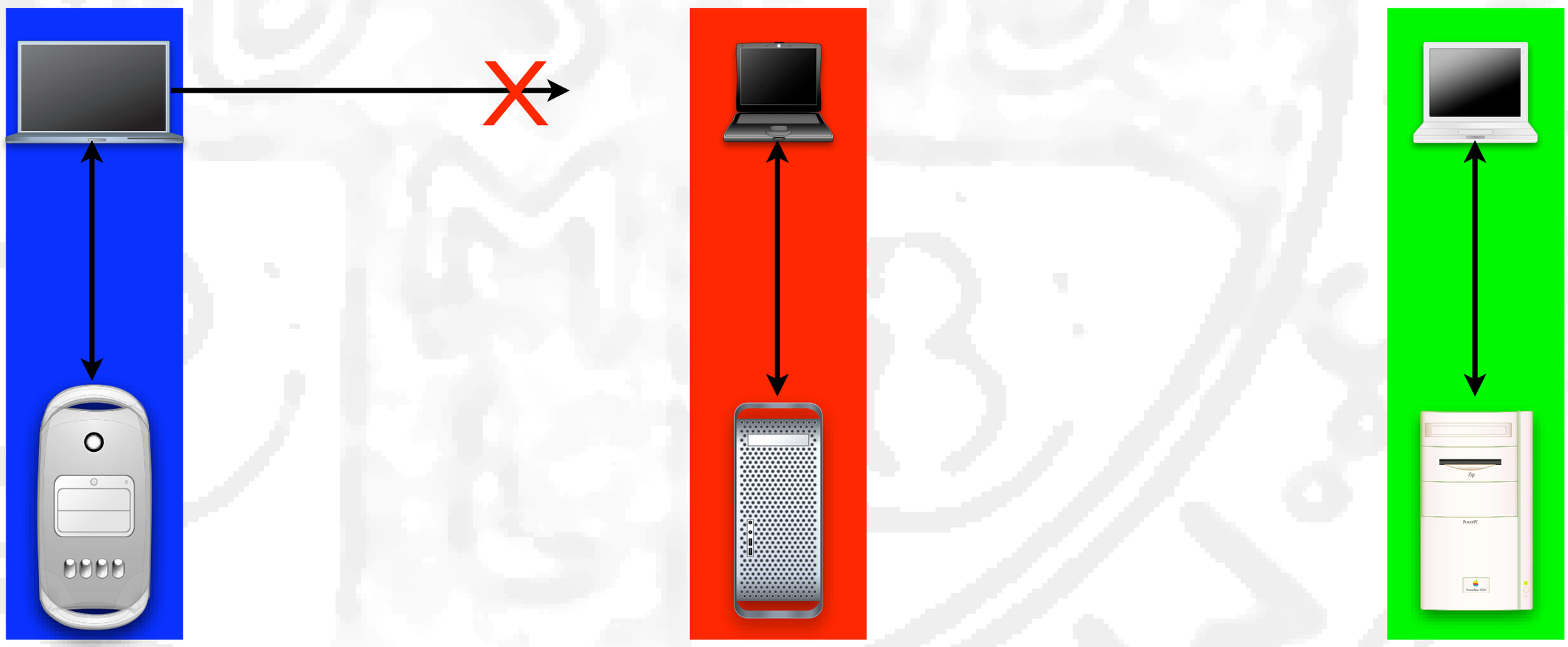
Inter-system Gap ("Air Gap")

Configuration

- separate computers
- separate networks

Protection

- no leakage between systems



Example: Military systems

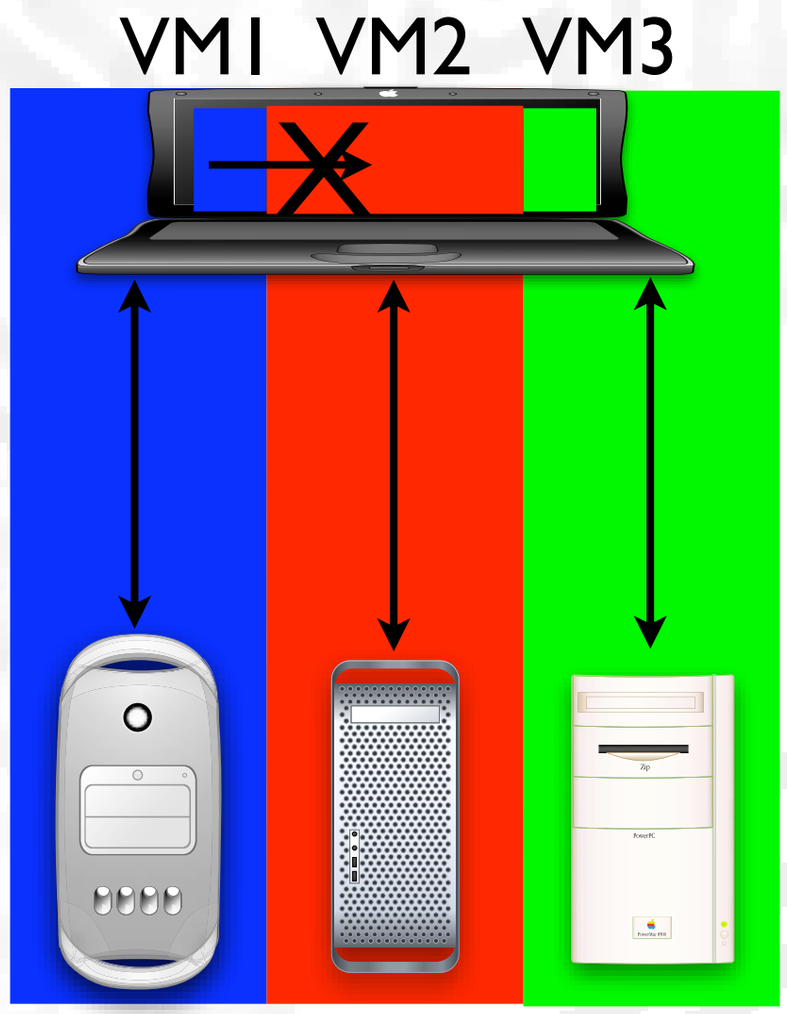
Inter-VM gap

Configuration

- one computer
- many single-level VMs

Protection

- no leakage between VMs
- VMM-level RefMon



Examples: NetTop, VMWare

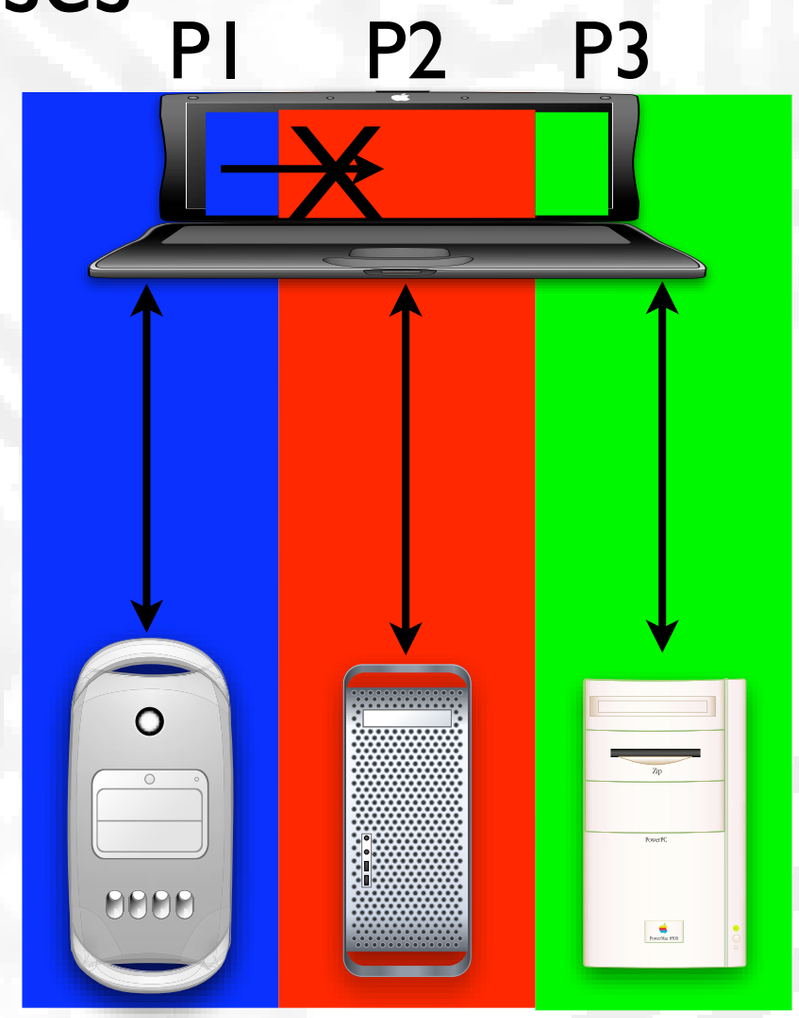
Interprocess Gap

Configuration

- one computer
- multi-level VMs
- single-level processes

Protection

- no leakage between processes
- OS-level RefMon



Examples: SELinux

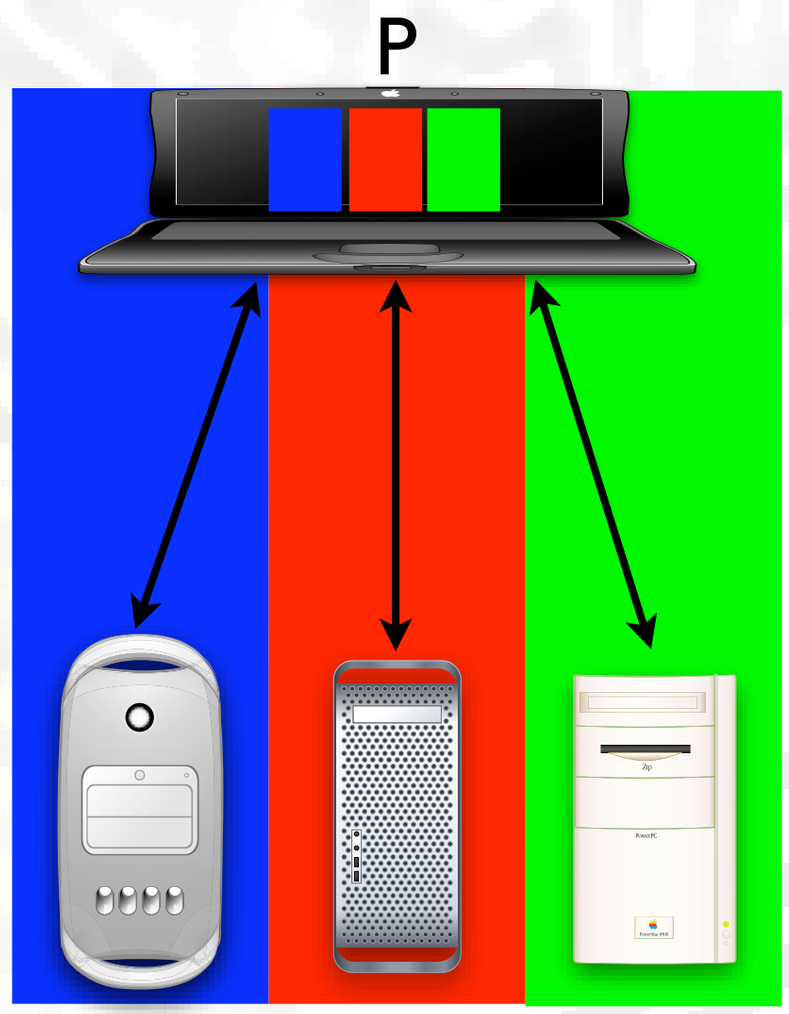
Inter-code Gap?

Configuration

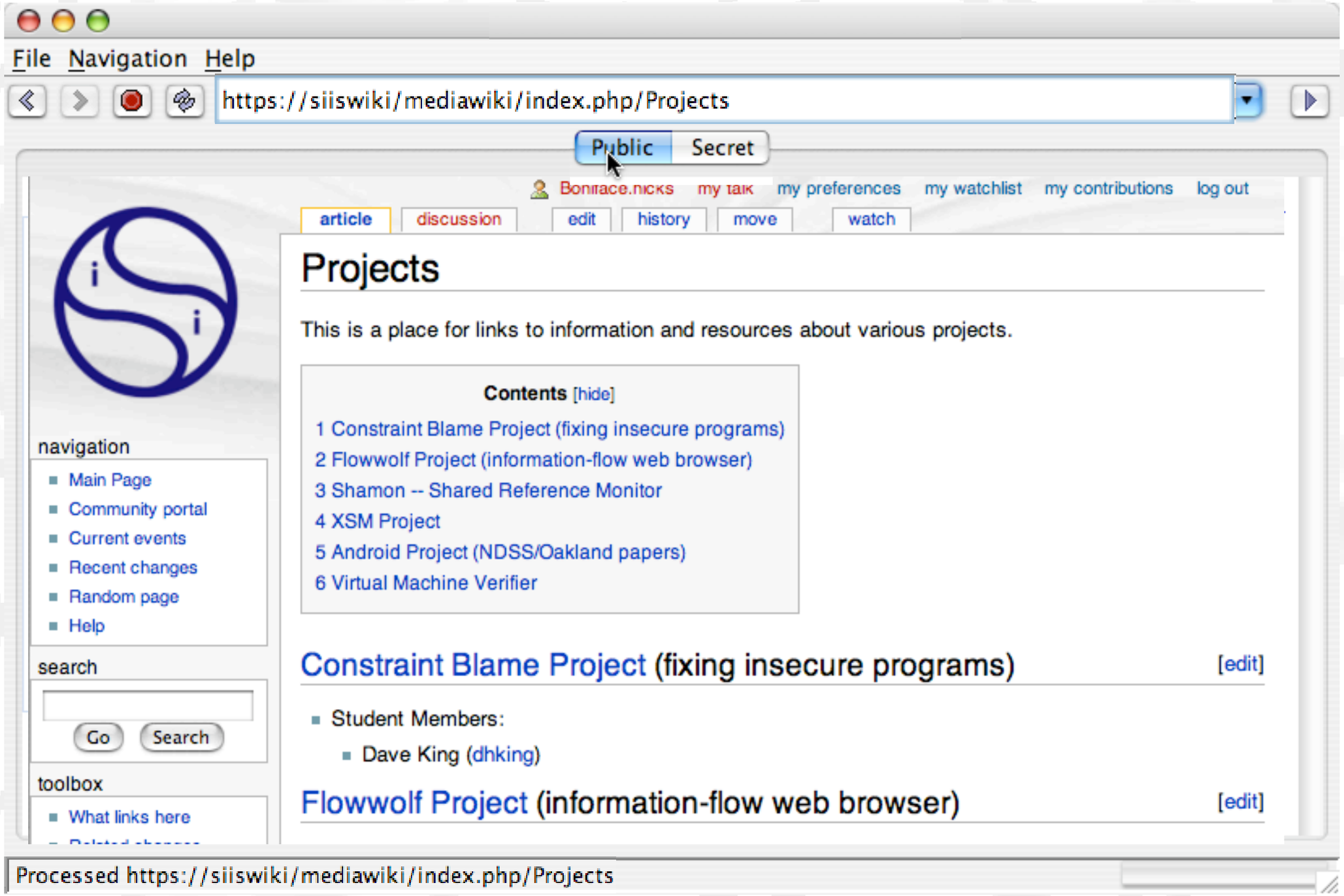
- one computer
- multi-level VMs
- multi-level processes
- single-level data

Protection

- no leakage between variables
- application-level RefMon



HTTP Mixes Data



HTTP Mixes Data

URL I

File Navigation Help

https://siiswiki/mediawiki/index.php/Projects

Public Secret

Boniface.nicks my talk my preferences my watchlist my contributions log out

article discussion edit history move watch

navigation

- Main Page
- Community portal
- Current events
- Recent changes
- Random page
- Help

search

Go Search

toolbox

- What links here

Projects

This is a place for links to information and resources about various projects.

Contents [hide]

Constraint Blame Project (fixing insecure programs) [edit]

- Student Members:
  - Dave King (dhking)

Flowwolf Project (information-flow web browser) [edit]

Processed https://siiswiki/mediawiki/index.php/Projects

HTTP Mixes Data

URL 1

URL 2

File Navigation Help

https://siiswiki/mediawiki/index.php/Projects

Public Secret

Boniface.nicks my talk my preferences my watchlist my contributions log out

article discussion edit history move watch

navigation

- Main Page
- Community portal
- Current events
- Recent changes
- Random page
- Help

search

Go Search

toolbox

- What links here

Projects

This is a place for links to information and resources about various projects.

Contents [hide]

Constraint Blame Project (fixing insecure programs) [edit]

- Student Members:
  - Dave King (dhking)

Flowwolf Project (information-flow web browser) [edit]

Processed https://siiswiki/mediawiki/index.php/Projects

HTTP Mixes Data

URL 1

URL 2

URL 3

File Navigation Help

https://siiswiki/mediawiki/index.php/Projects

Public Secret

Boniface.nicks my talk my preferences my watchlist my contributions log out

article discussion edit history move watch

navigation

- Main Page
- Community portal
- Current events
- Recent changes
- Random page
- Help

search

Go Search

toolbox

- What links here

Projects

This is a place for links to information and resources about various projects.

Contents [hide]

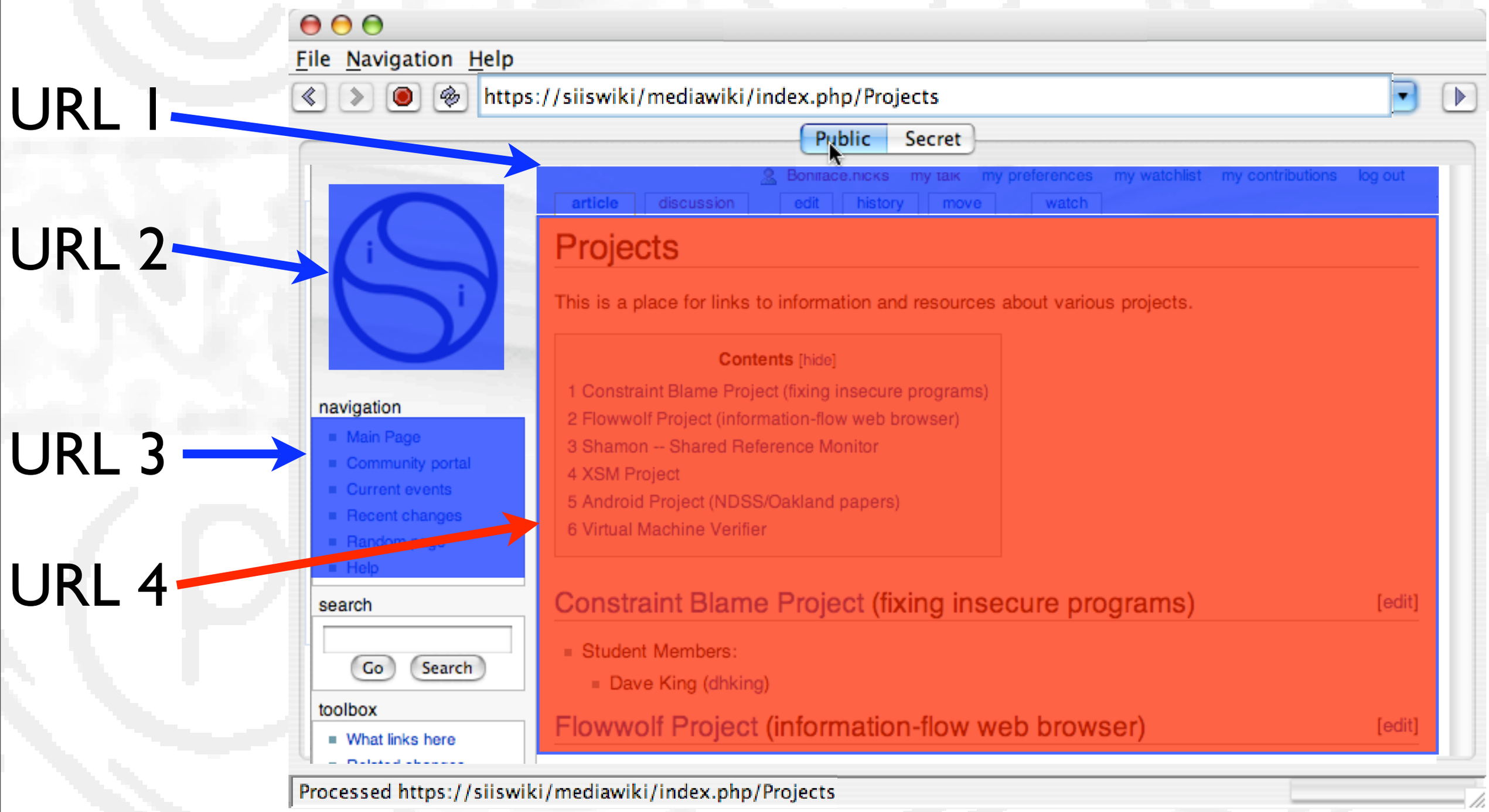
Constraint Blame Project (fixing insecure programs) [edit]

- Student Members:
  - Dave King (dhking)

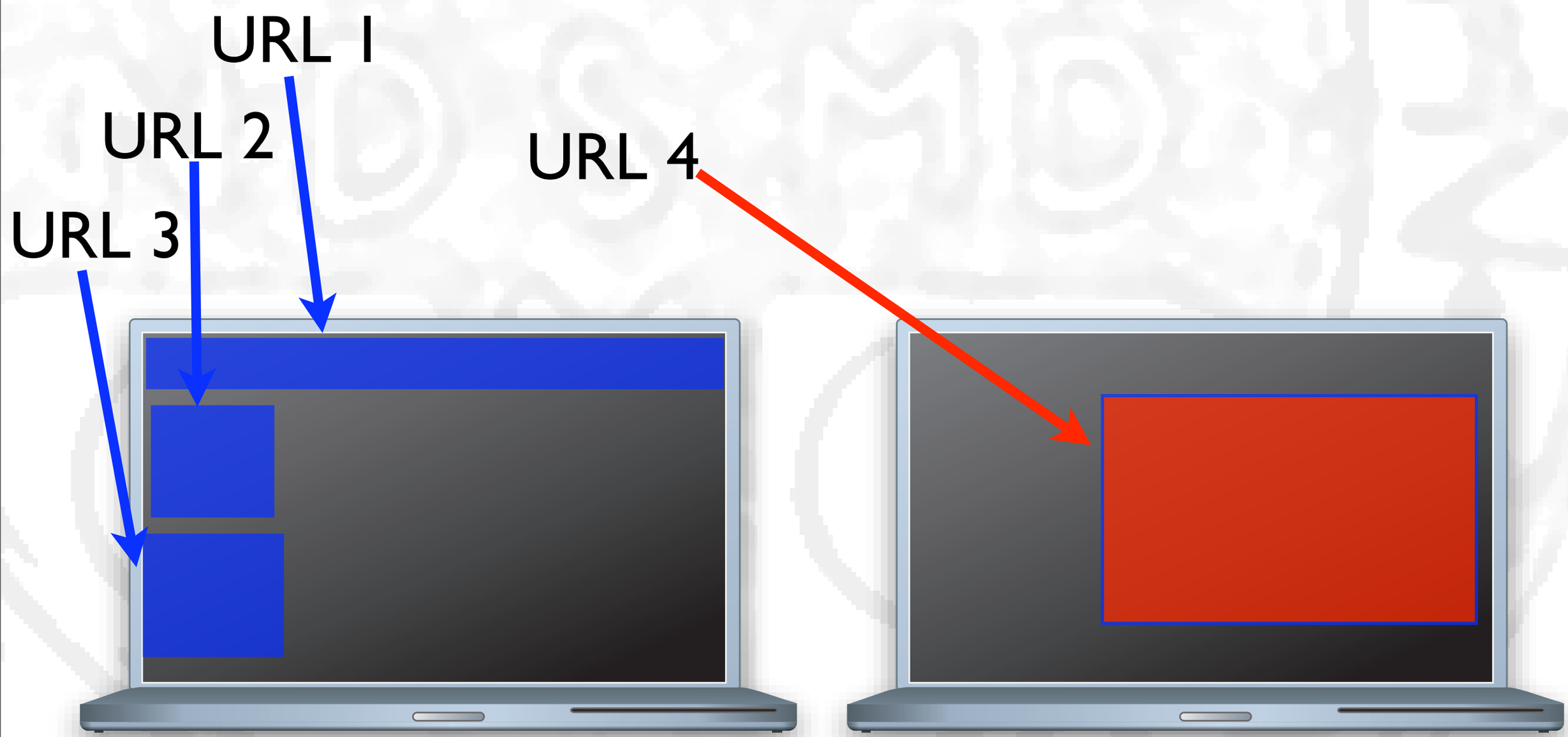
Flowwolf Project (information-flow web browser) [edit]

Processed https://siiswiki/mediawiki/index.php/Projects

HTTP Mixes Data



Air Gap Bad for HTTP



Multi-labeled Browsing

File Navigation Help

https://siiswiki/mediawiki/index.php/Projects

Public Secret

Boniface.nicks my talk my preferences my watchlist my contributions log out

article discussion edit history move watch

navigation
- Main Page
- Community portal
- Current events
- Recent changes
- Random page
- Help

search

Go Search

toolbox
- What links here

Projects

This is a place for links to information and resources about various projects.

Contents [hide]

Constraint Blame Project (fixing insecure programs) [edit]

- Student Members:
  - Dave King (dhking)

Flowwolf Project (information-flow web browser) [edit]

Processed https://siiswiki/mediawiki/index.php/Projects

Multi-labeled Browsing

URL I

File Navigation Help

https://siiswiki/mediawiki/index.php/Projects

Public Secret

Boniface.nicks my talk my preferences my watchlist my contributions log out

article discussion edit history move watch

navigation

- Main Page
- Community portal
- Current events
- Recent changes
- Random page
- Help

search

Go Search

toolbox

- What links here

Projects

This is a place for links to information and resources about various projects.

Contents [hide]

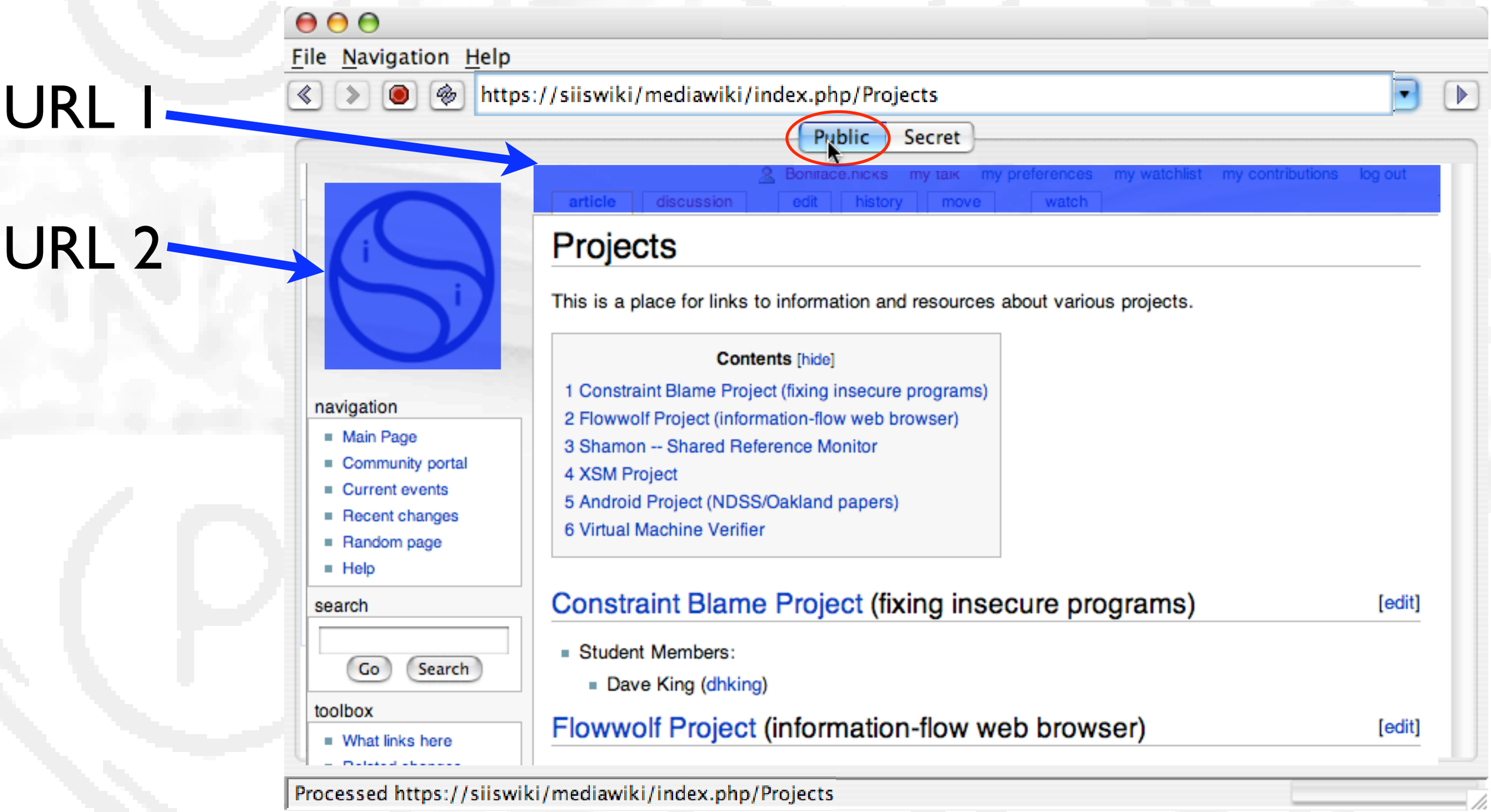
Constraint Blame Project (fixing insecure programs) [edit]

- Student Members:
  - Dave King (dhking)

Flowwolf Project (information-flow web browser) [edit]

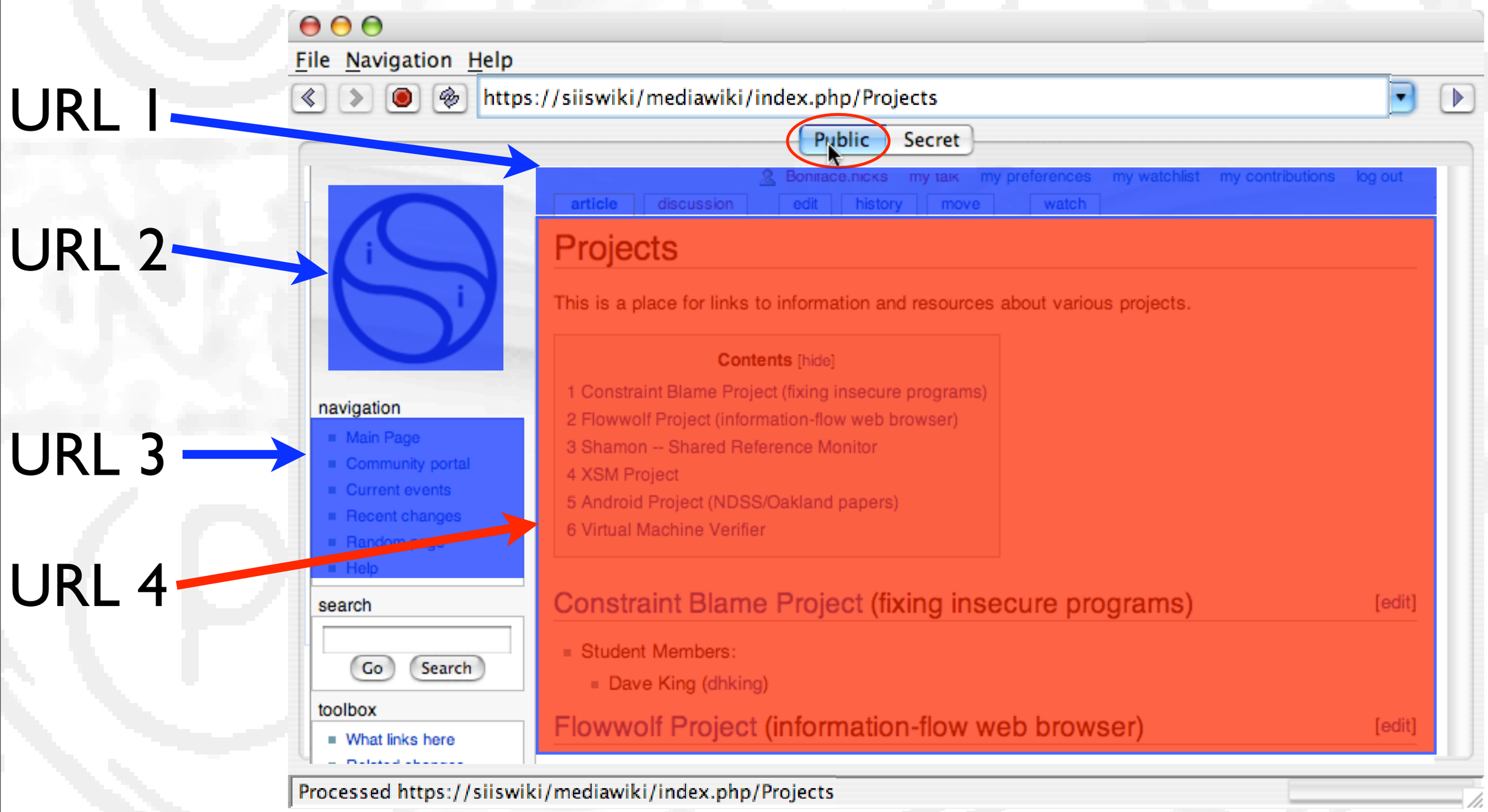
Processed https://siiswiki/mediawiki/index.php/Projects

Multi-labeled Browsing

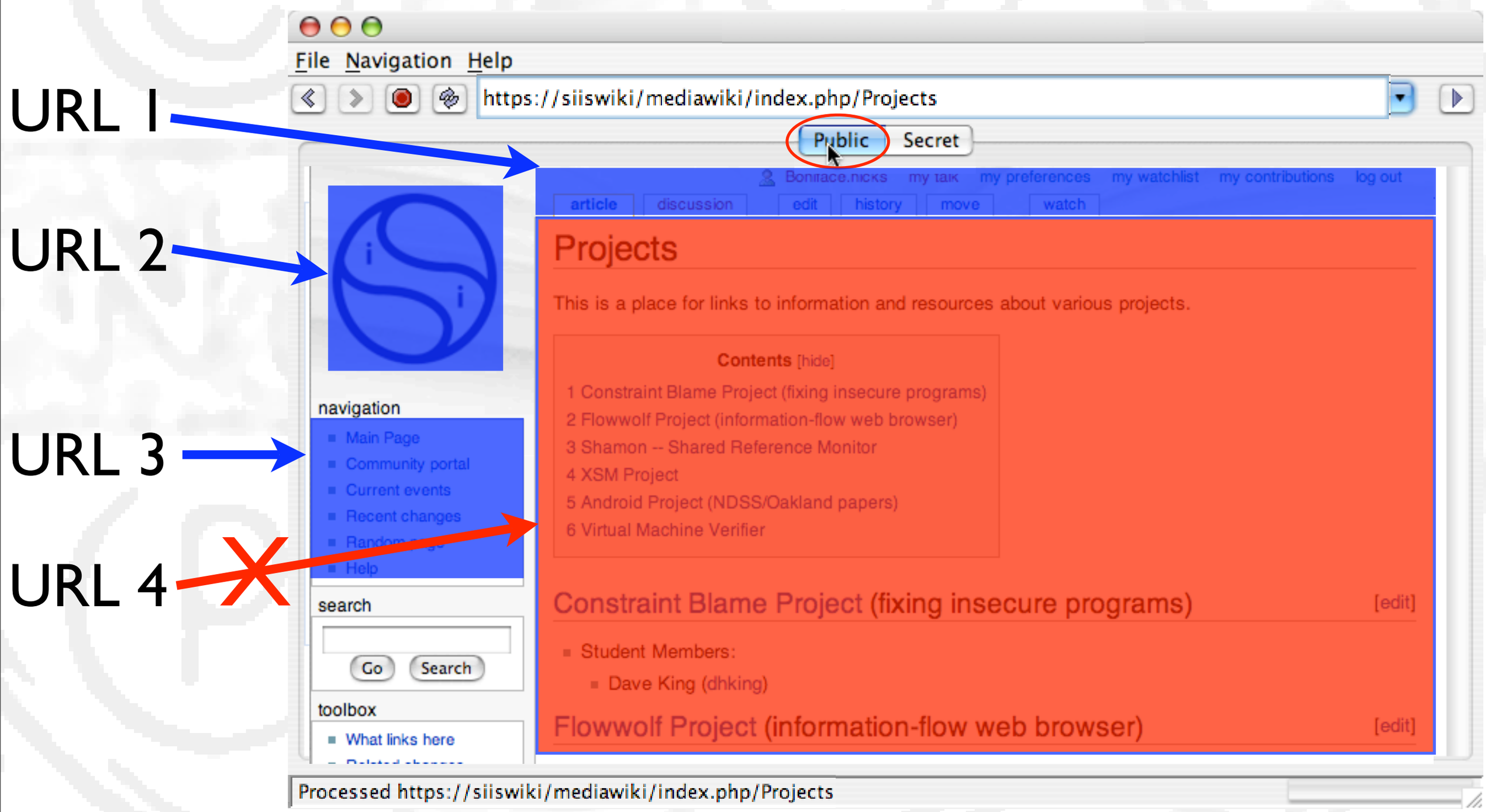


Multi-labeled Browsing

Multi-labeled Browsing



Multi-labeled Browsing



Multi-labeled Browsing

File Navigation Help

https://siiswiki/mediawiki/index.php/Projects

Public Secret

Boniface.hicks my talk my preferences my watchlist my contributions log out

article discussion edit history move watch

navigation

- Main Page
- Community portal
- Current events
- Recent changes
- Random page
- Help

search

Go Search

toolbox

- What links here

Projects

This is a place for links to information and resources about various projects.

Contents [hide]

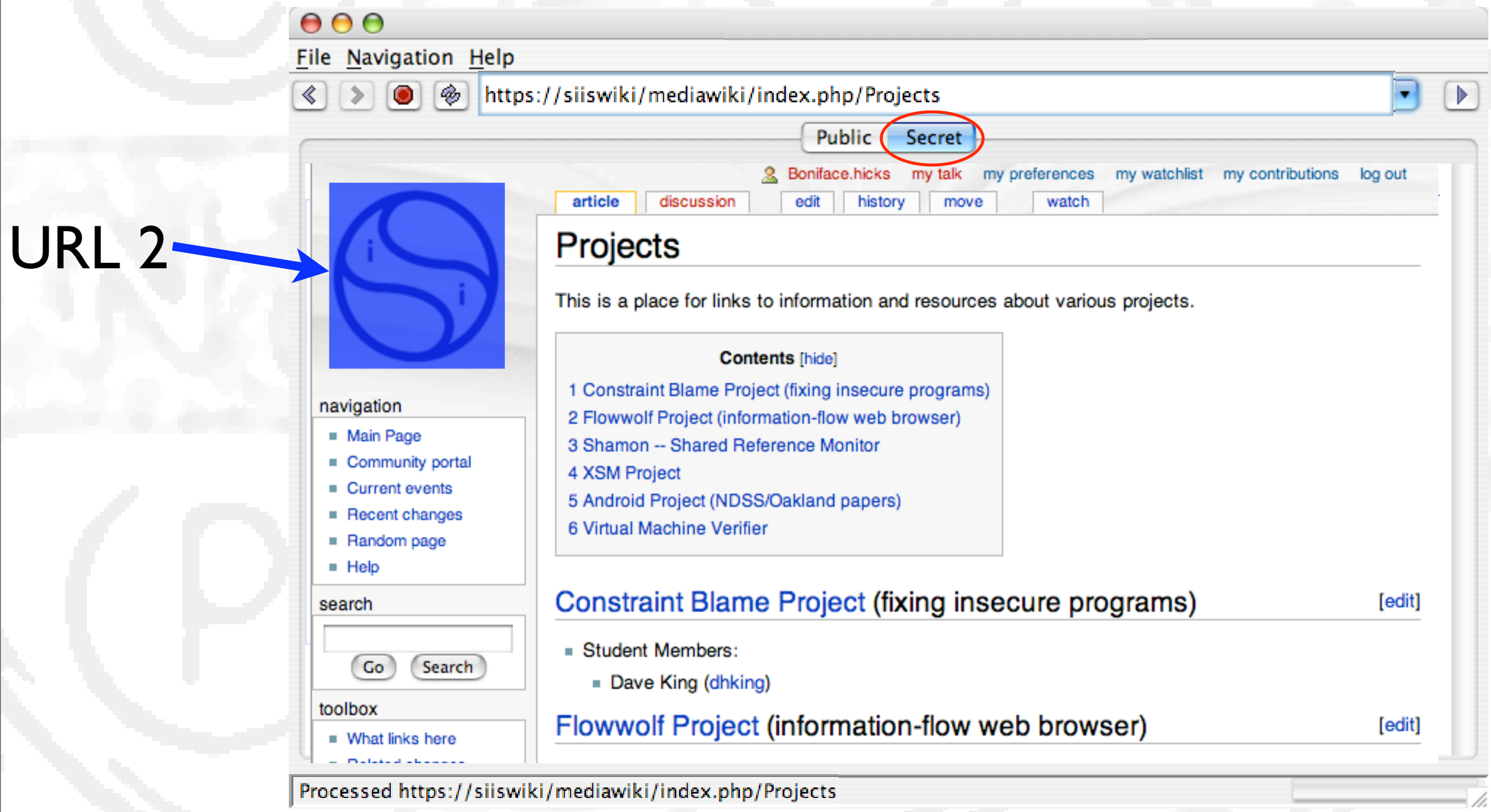
Constraint Blame Project (fixing insecure programs) [edit]

- Student Members:
  - Dave King (dhking)

Flowwolf Project (information-flow web browser) [edit]

Processed https://siiswiki/mediawiki/index.php/Projects

Multi-labeled Browsing



Multi-labeled Browsing

URL 2

URL 3

File Navigation Help

https://siiswiki/mediawiki/index.php/Projects

Public Secret

Boniface.hicks my talk my preferences my watchlist my contributions log out

article discussion edit history move watch

navigation

- Main Page
- Community portal
- Current events
- Recent changes
- Random page
- Help

search

Go Search

toolbox

- What links here

Projects

This is a place for links to information and resources about various projects.

Contents [hide]

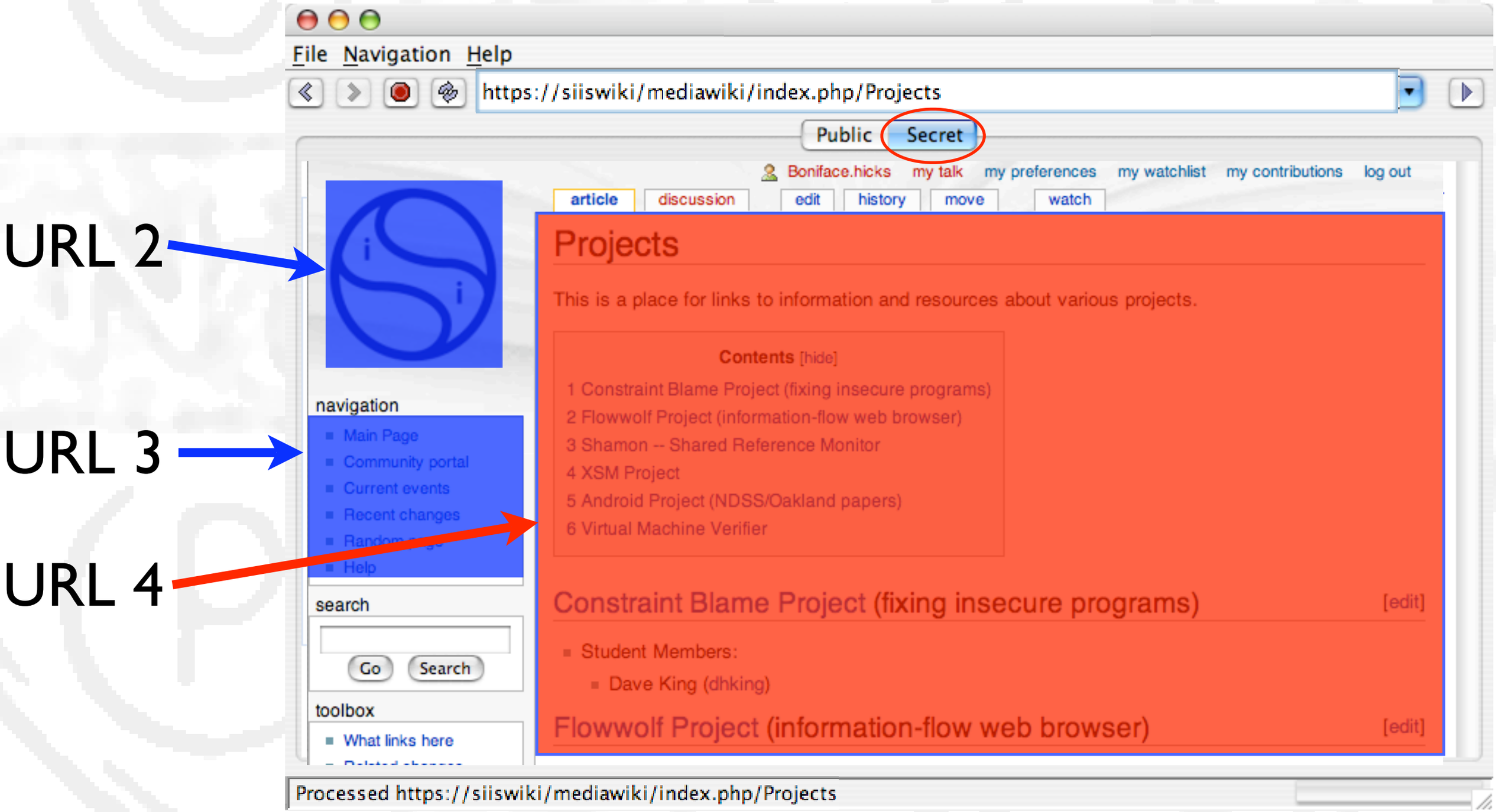
Constraint Blame Project (fixing insecure programs) [edit]

- Student Members:
  - Dave King (dhking)

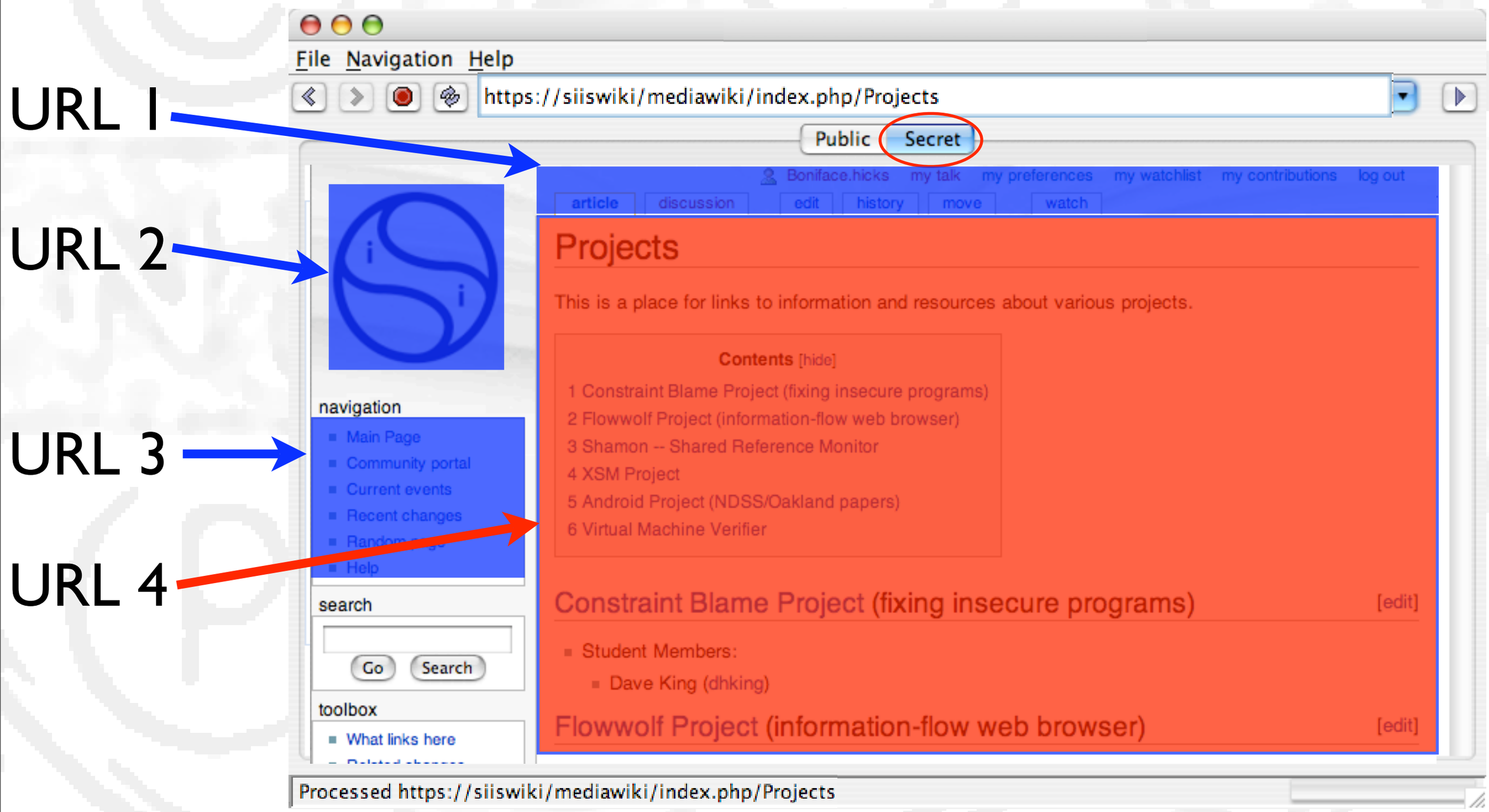
Flowwolf Project (information-flow web browser) [edit]

Processed https://siiswiki/mediawiki/index.php/Projects

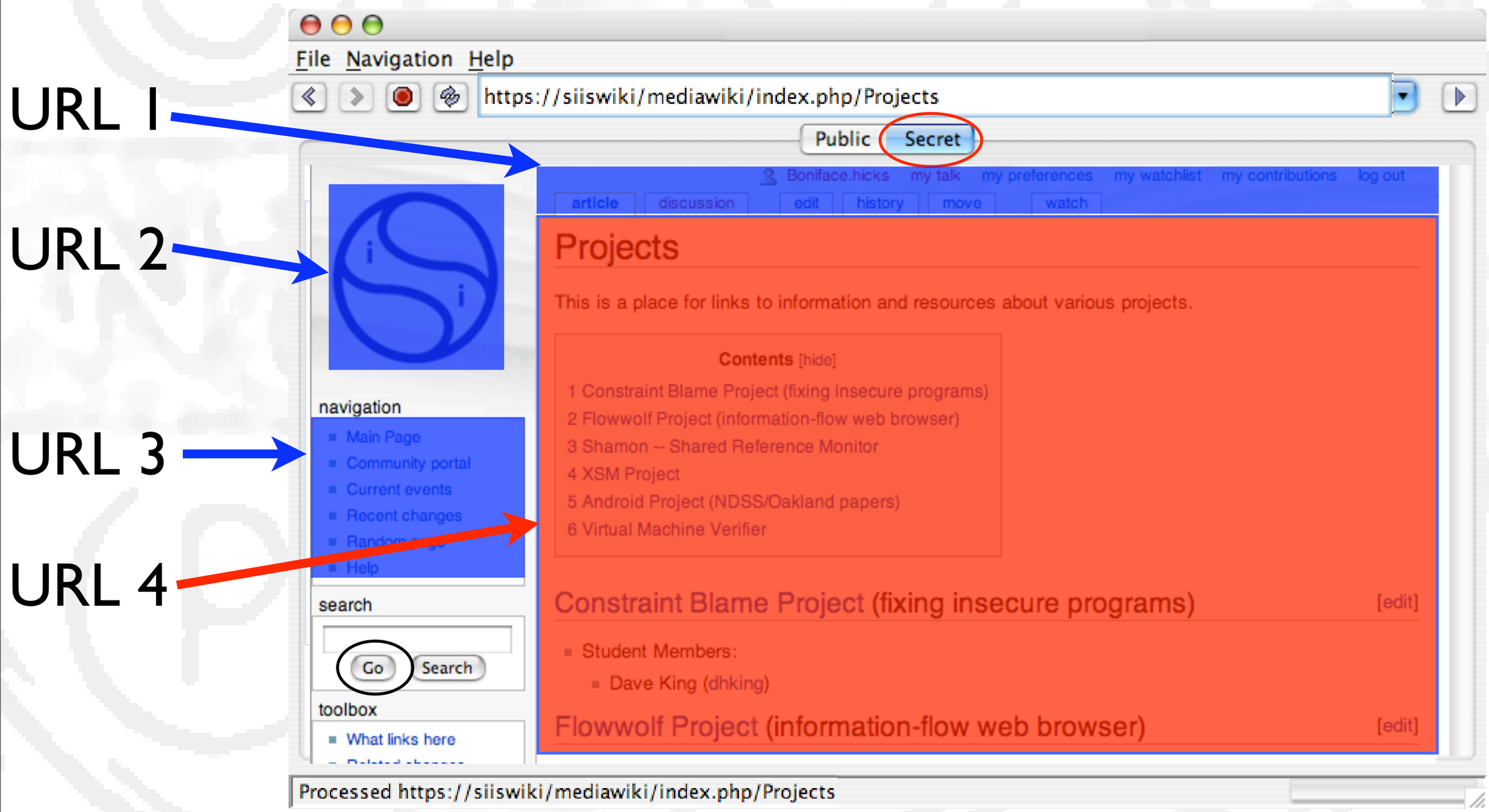
Multi-labeled Browsing



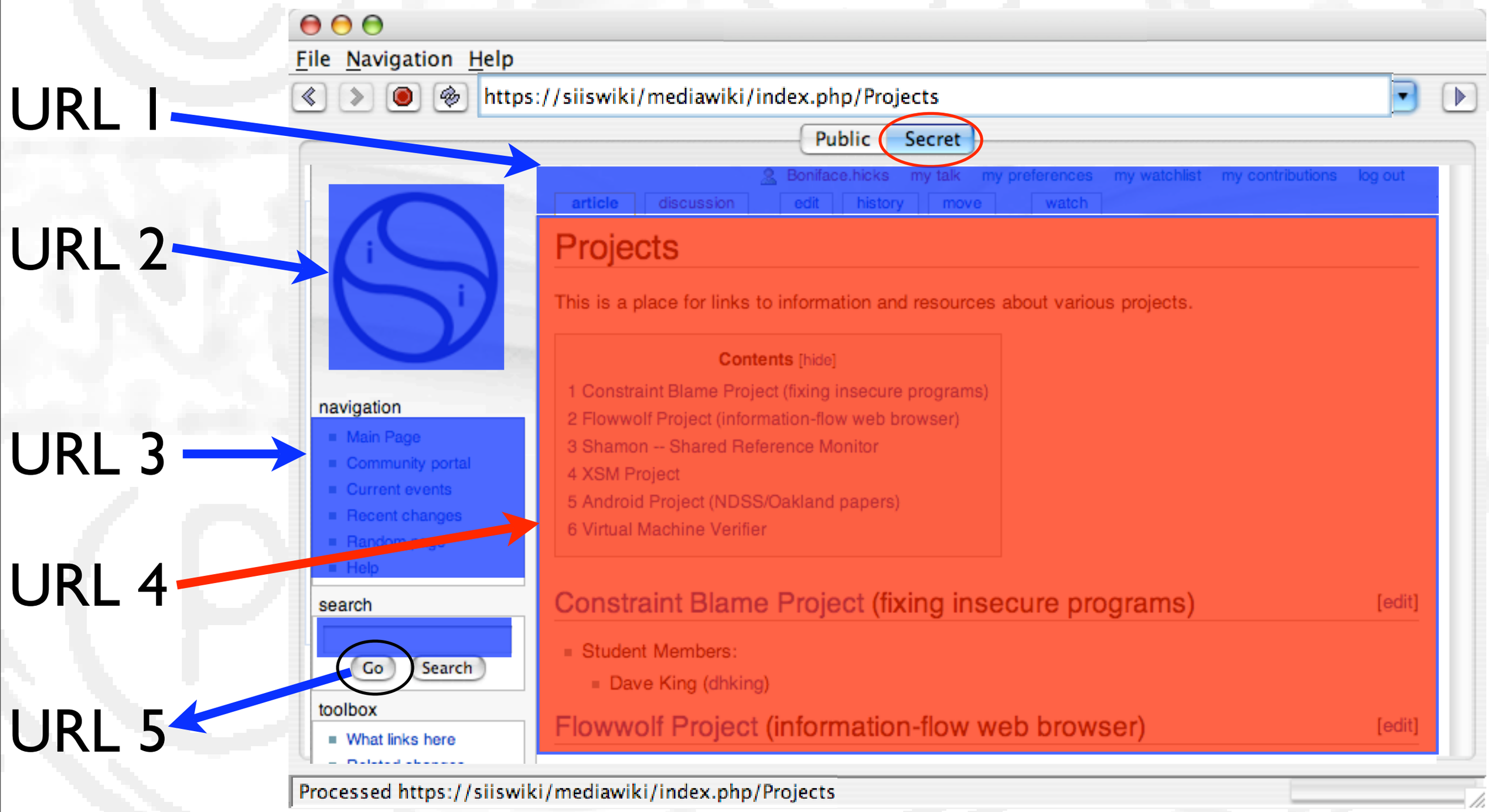
Multi-labeled Browsing



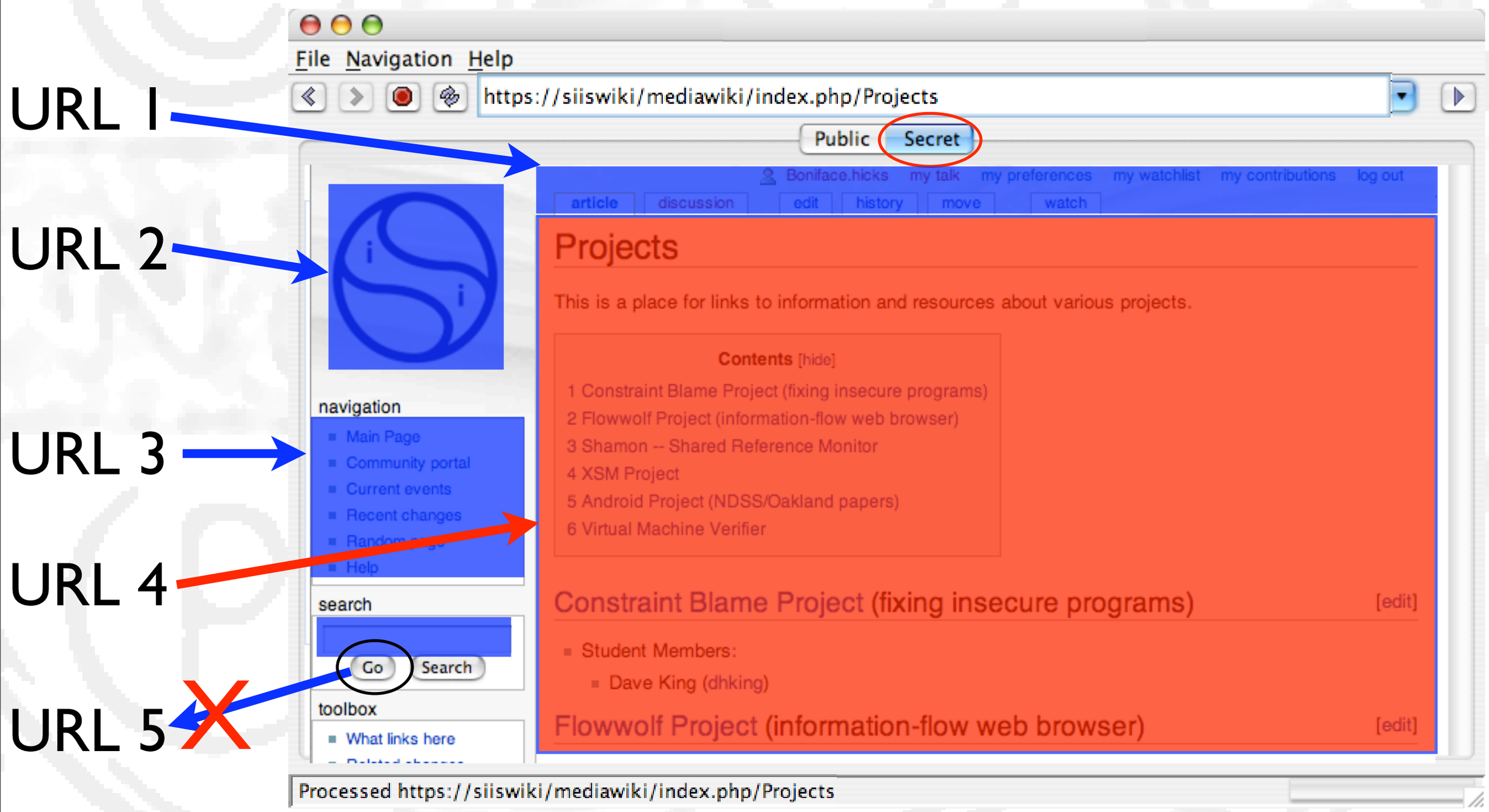
Multi-labeled Browsing



Multi-labeled Browsing



Multi-labeled Browsing



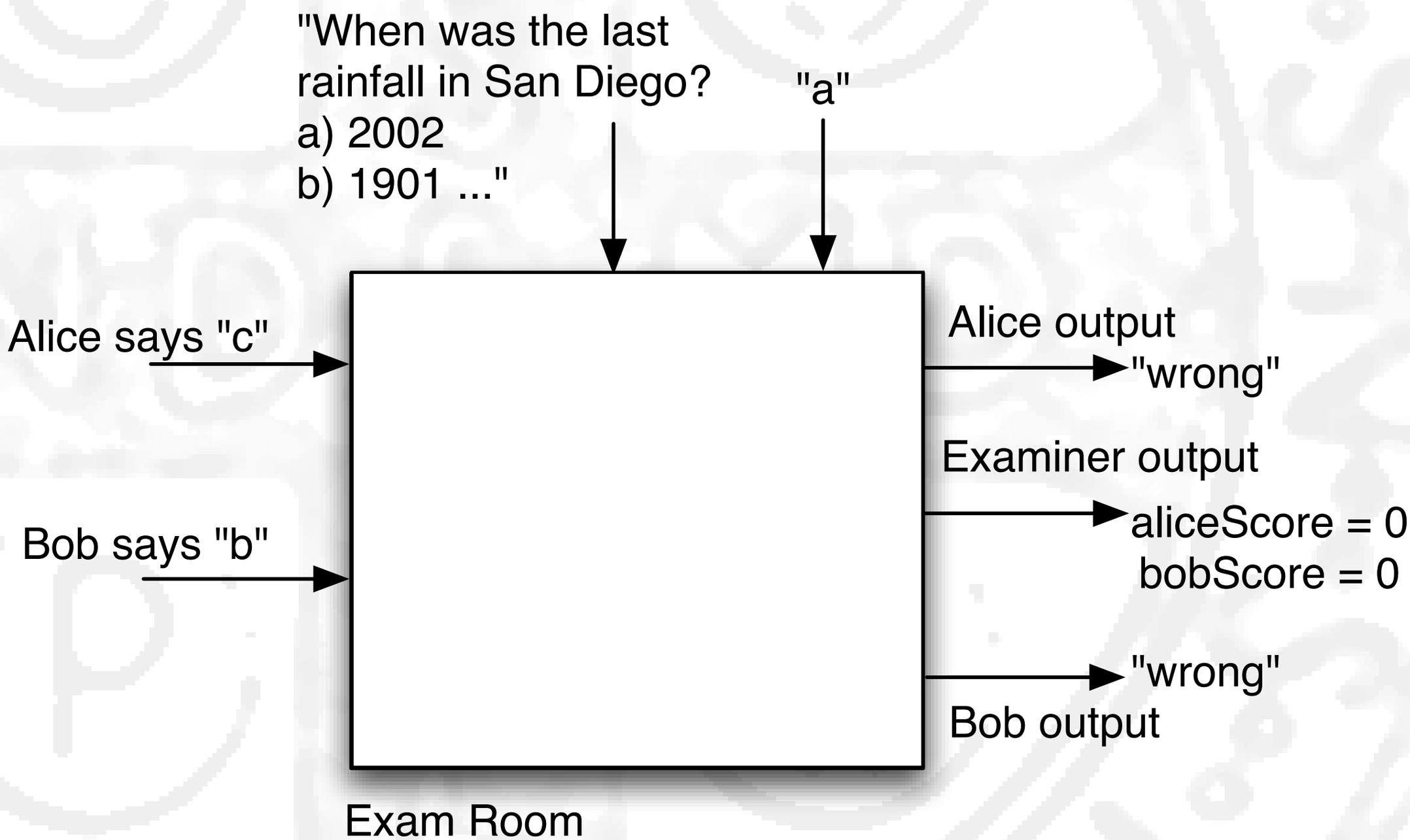
Application-Level RefMons

- Reference Monitor
  - Tamperproof
  - Verifiable
  - Complete Mediation
- Inline Reference Monitor (IRM)
  - Need to prove complete mediation
  - Much dynamic checking is needed at mediation points
- Type-based Approach
  - Strongly-typed languages enable complete mediation on types (*Important insight!*) compositionally
  - Much static checking is possible

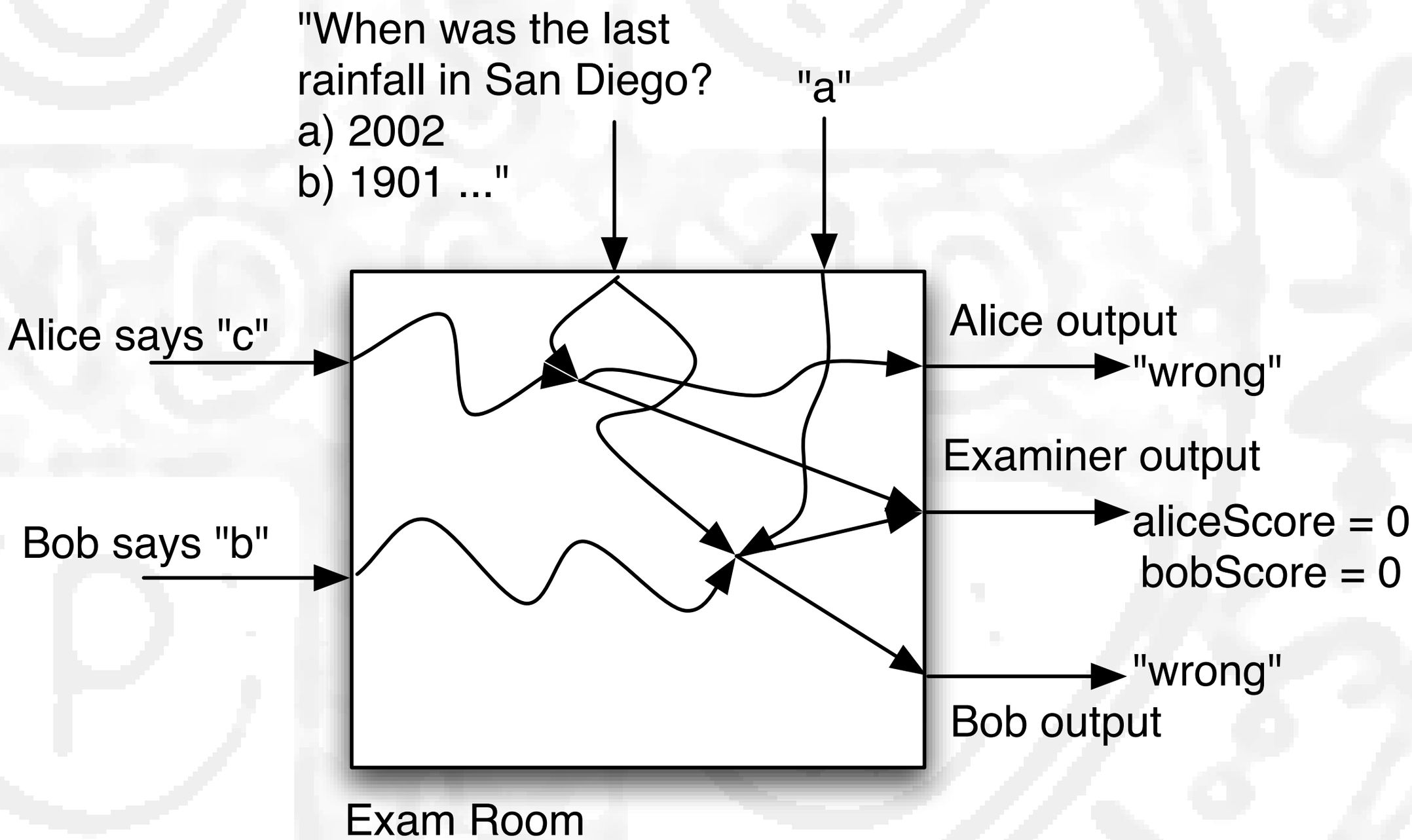
Security-Typed Languages

- Basic idea
  - add labels to types
  - add label comparison to type checking
  - richer flow models possible with subtyping
- Some STLs:
  - Jif
  - Fable (SELinks)
  - FlowCaml
  - Aura
  - others...

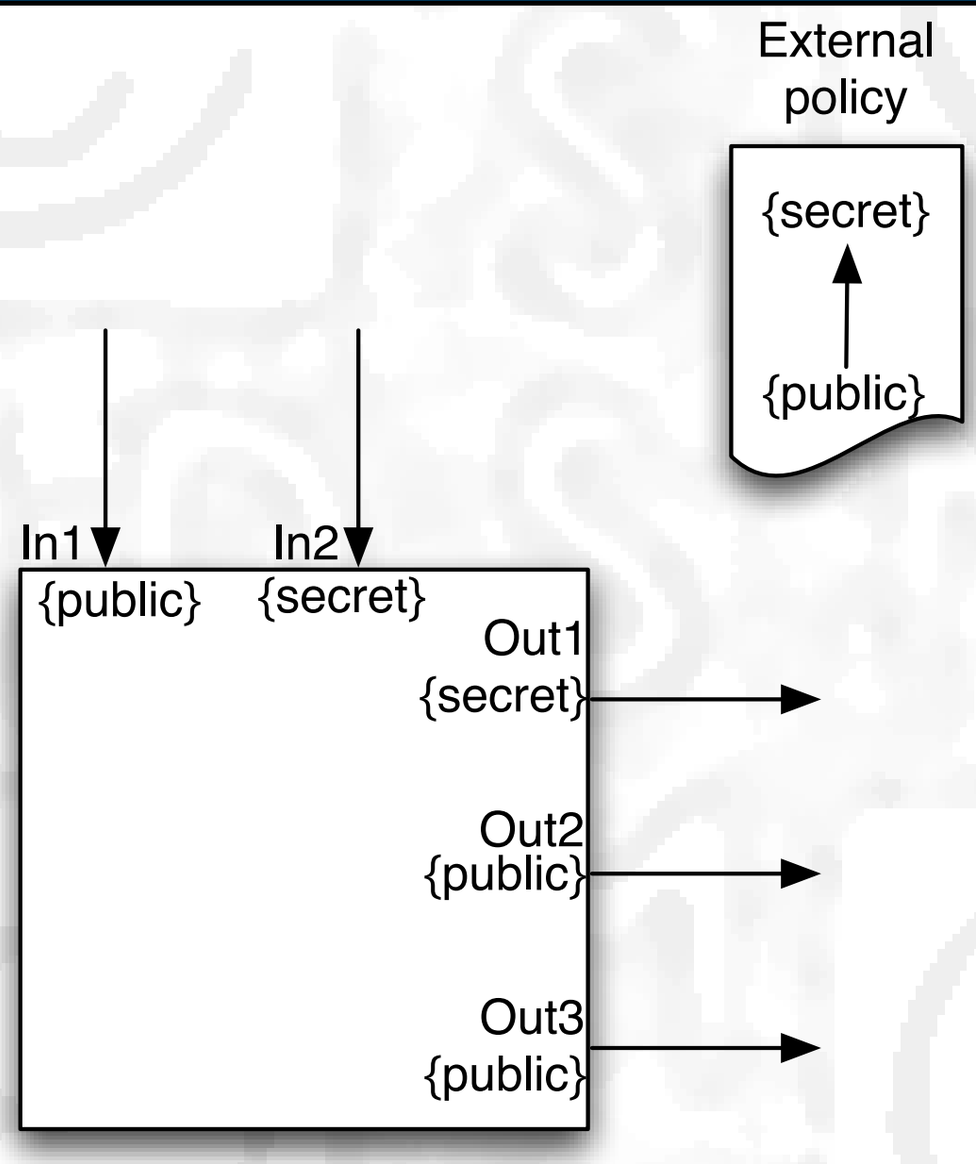
Exam Room Example



Exam Room Example

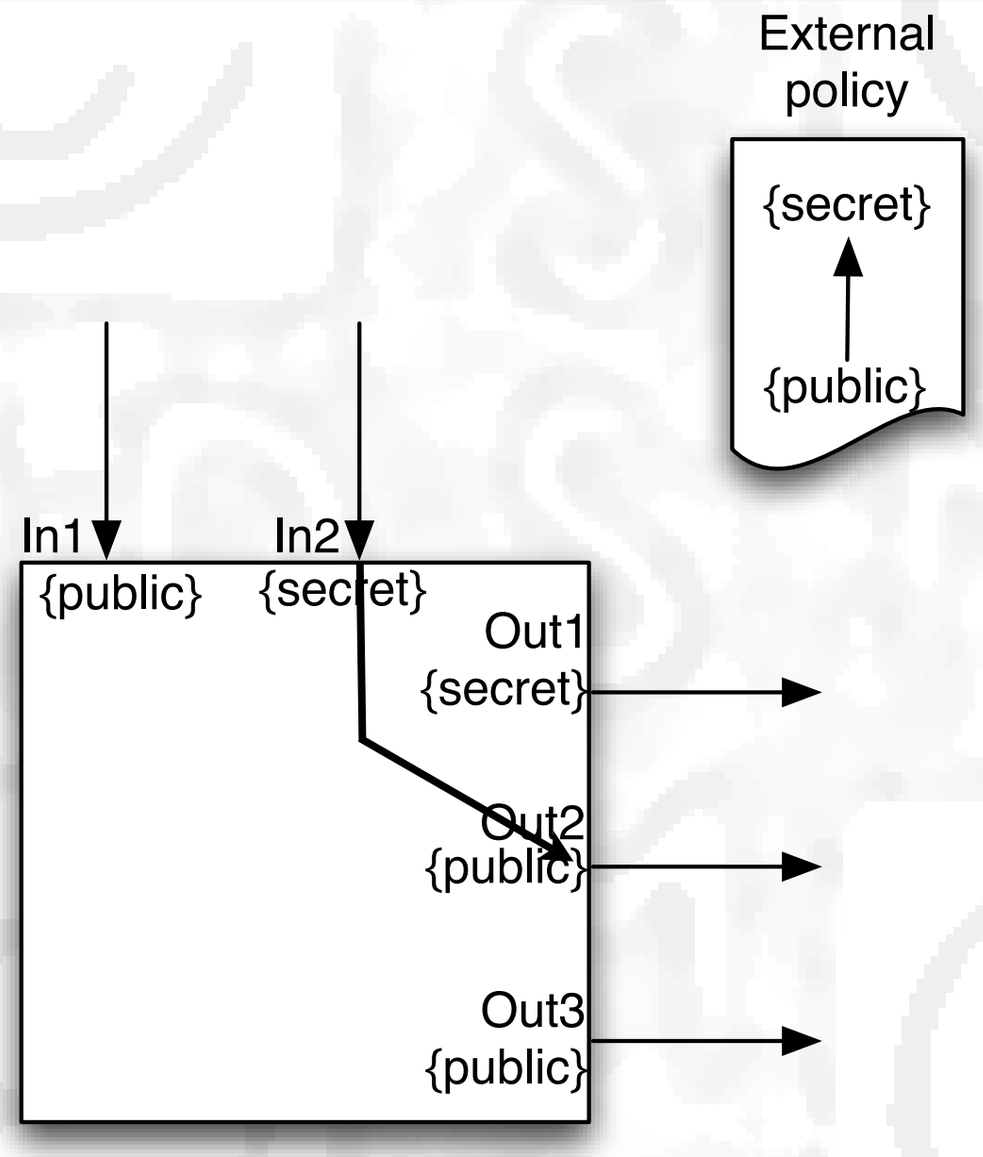


Add Labels for Infoflow Control



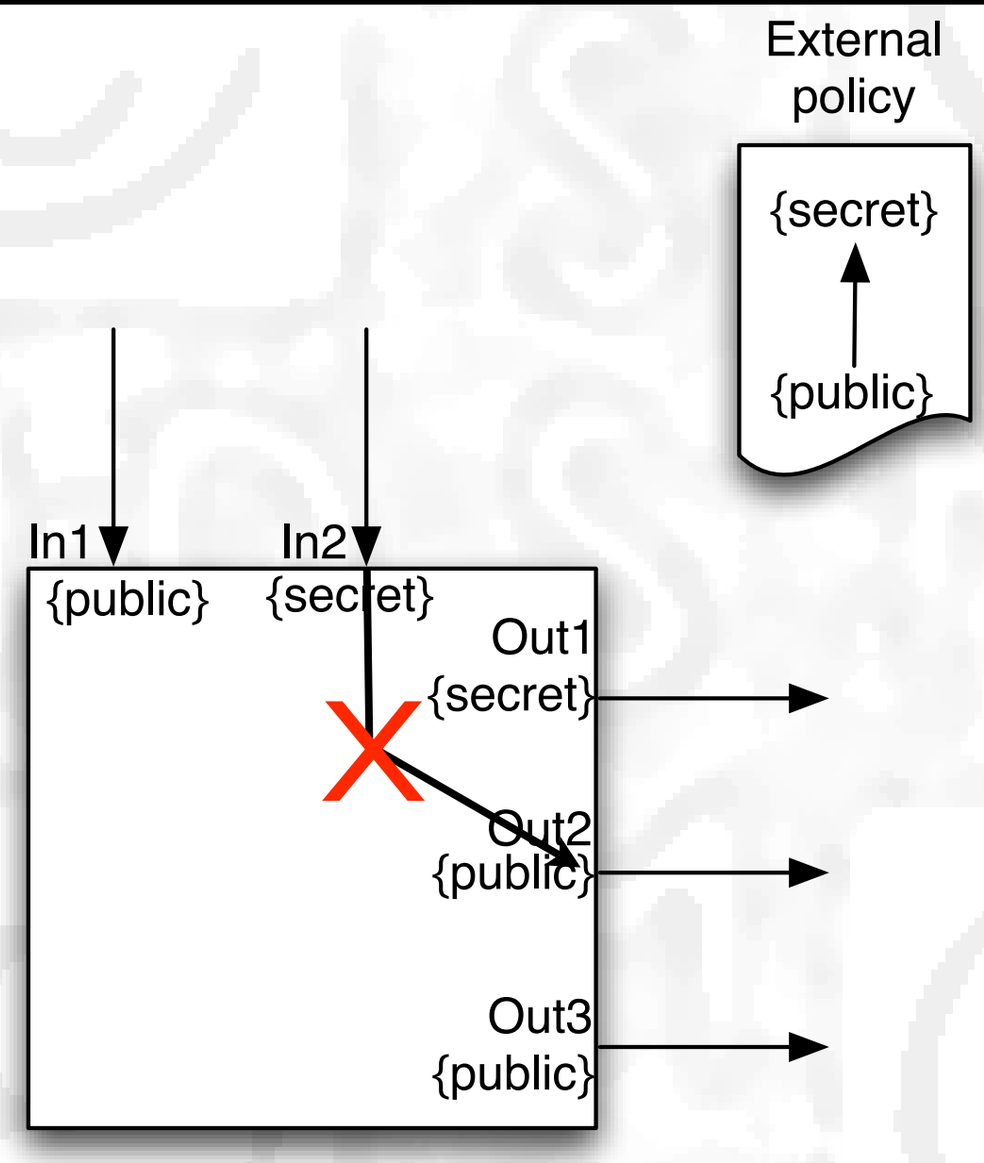
1. (static) type labels
2. label semantics

Add Labels for Infoflow Control



1. (static) type labels
2. label semantics

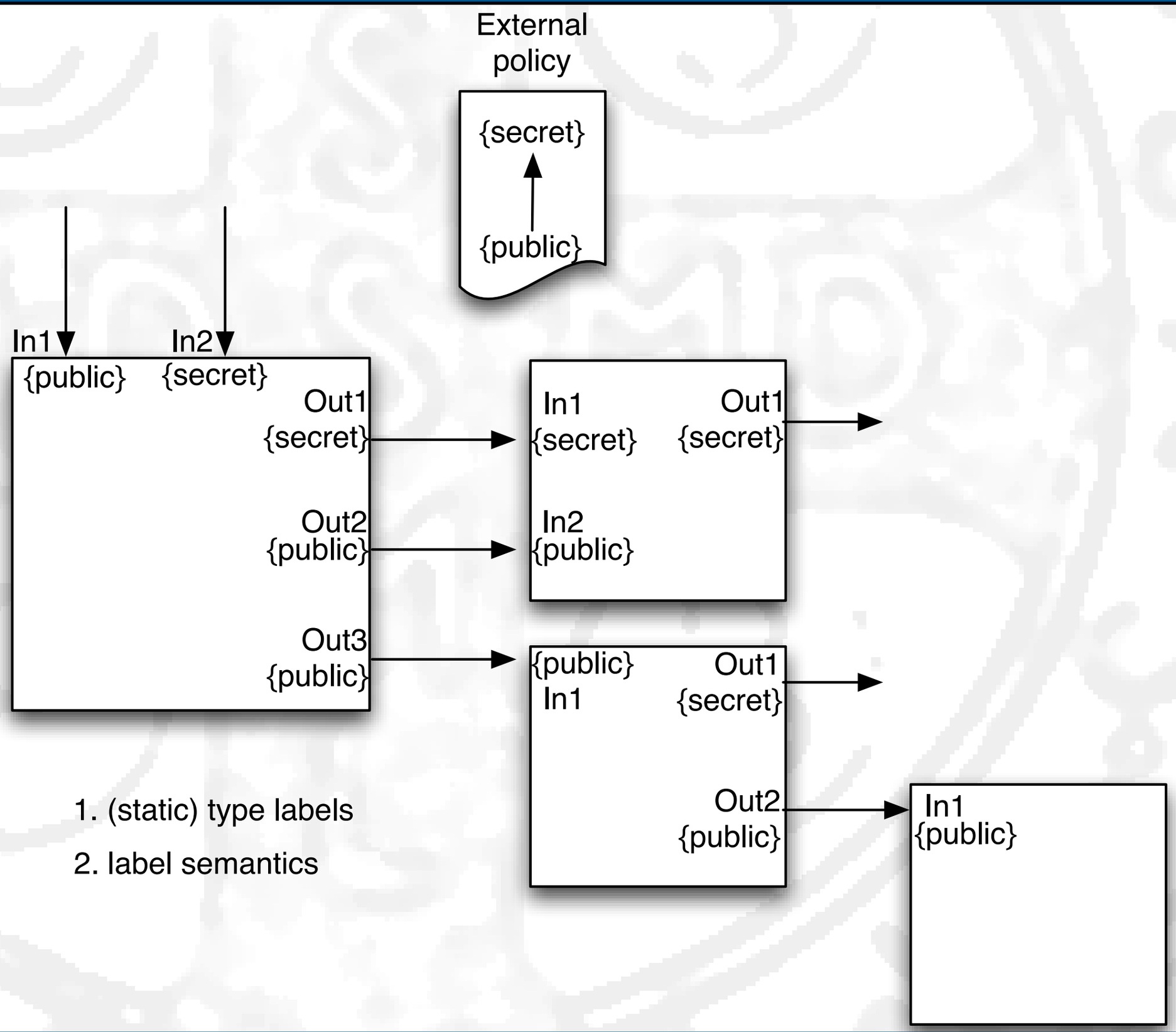
Add Labels for Infoflow Control



1. (static) type labels
2. label semantics

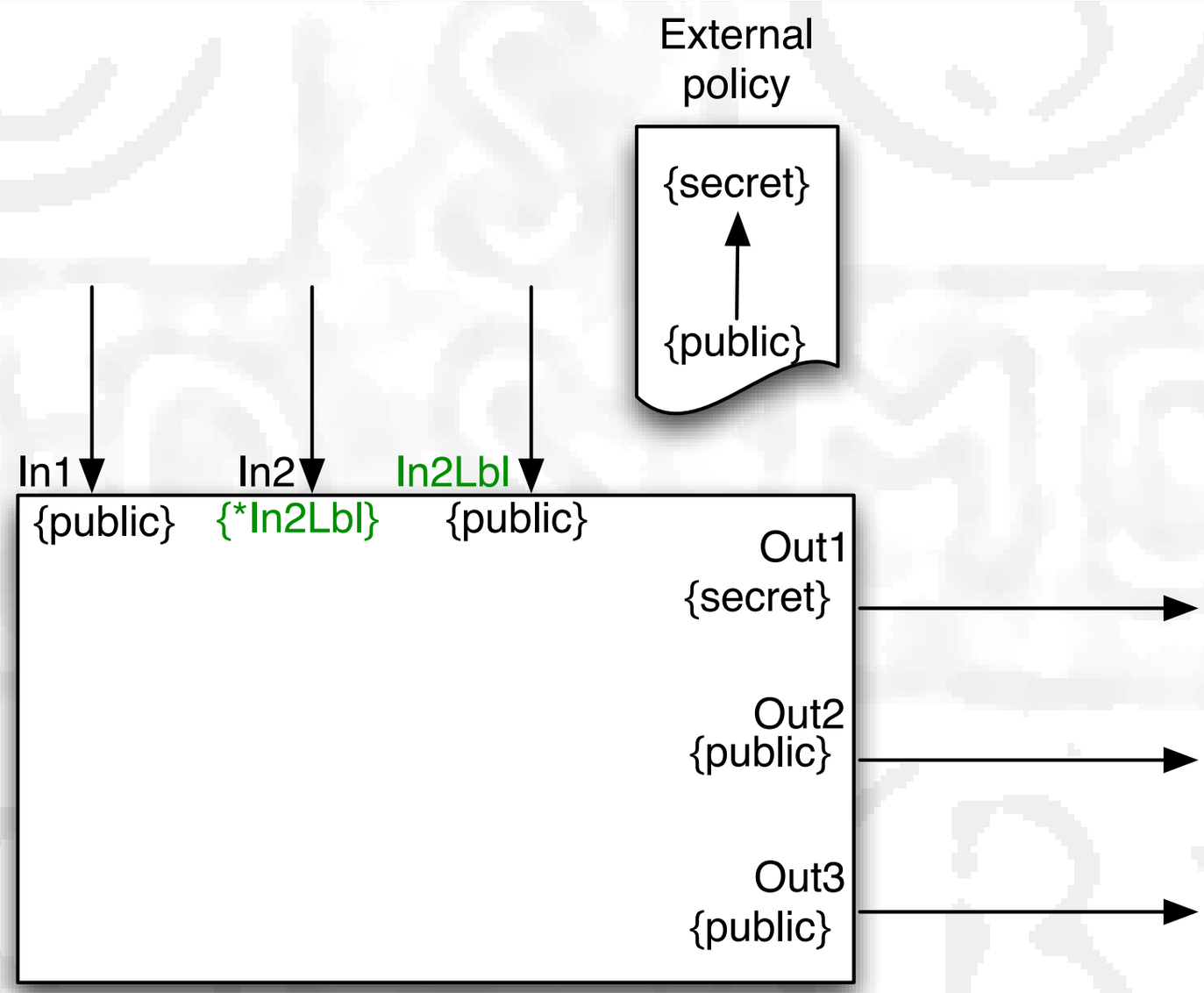
Compiler (typechecking) guarantees each module contains *no bad information flows*

Compositionality is Easy



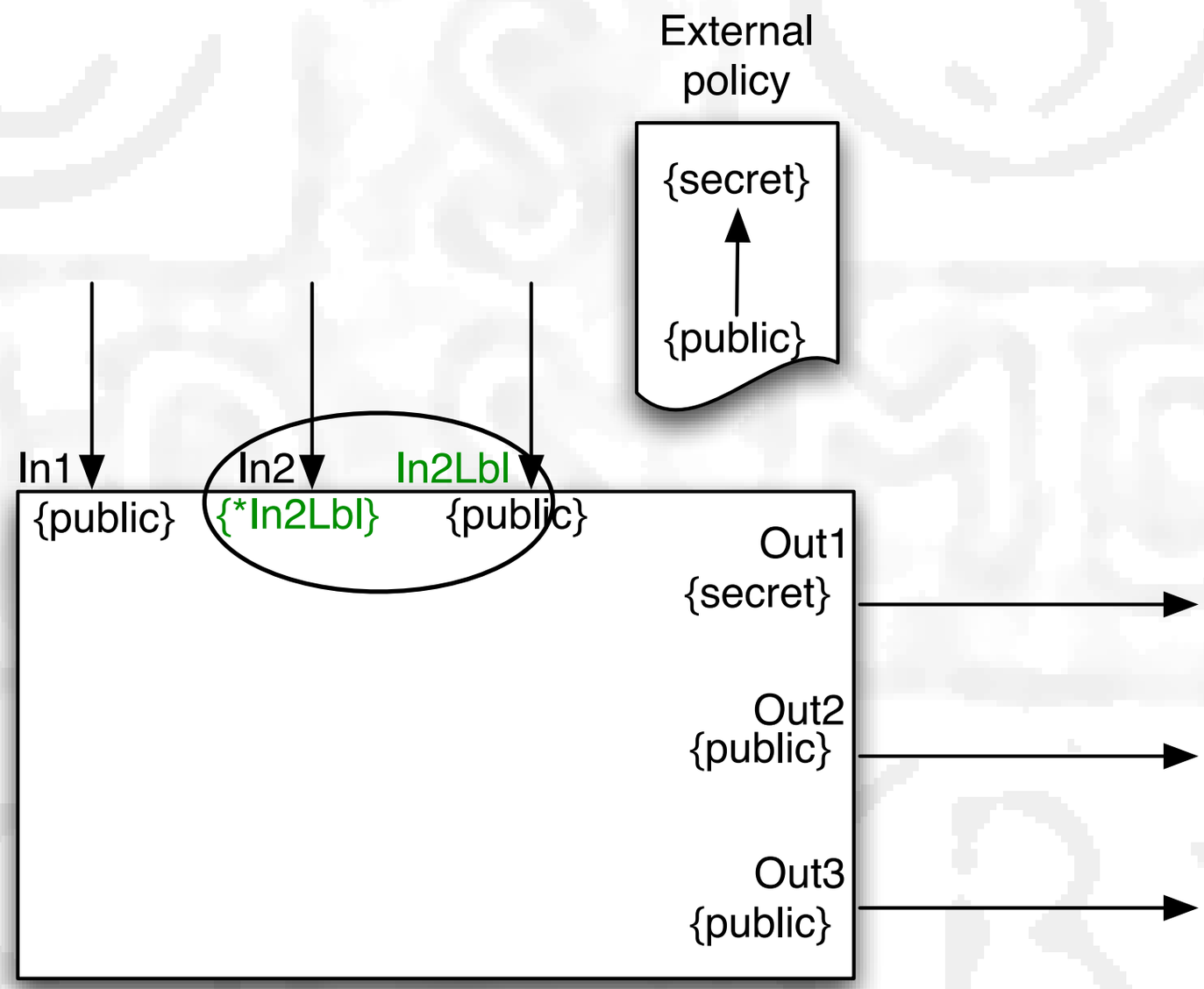
1. (static) type labels
2. label semantics

(Also dynamic type labels)



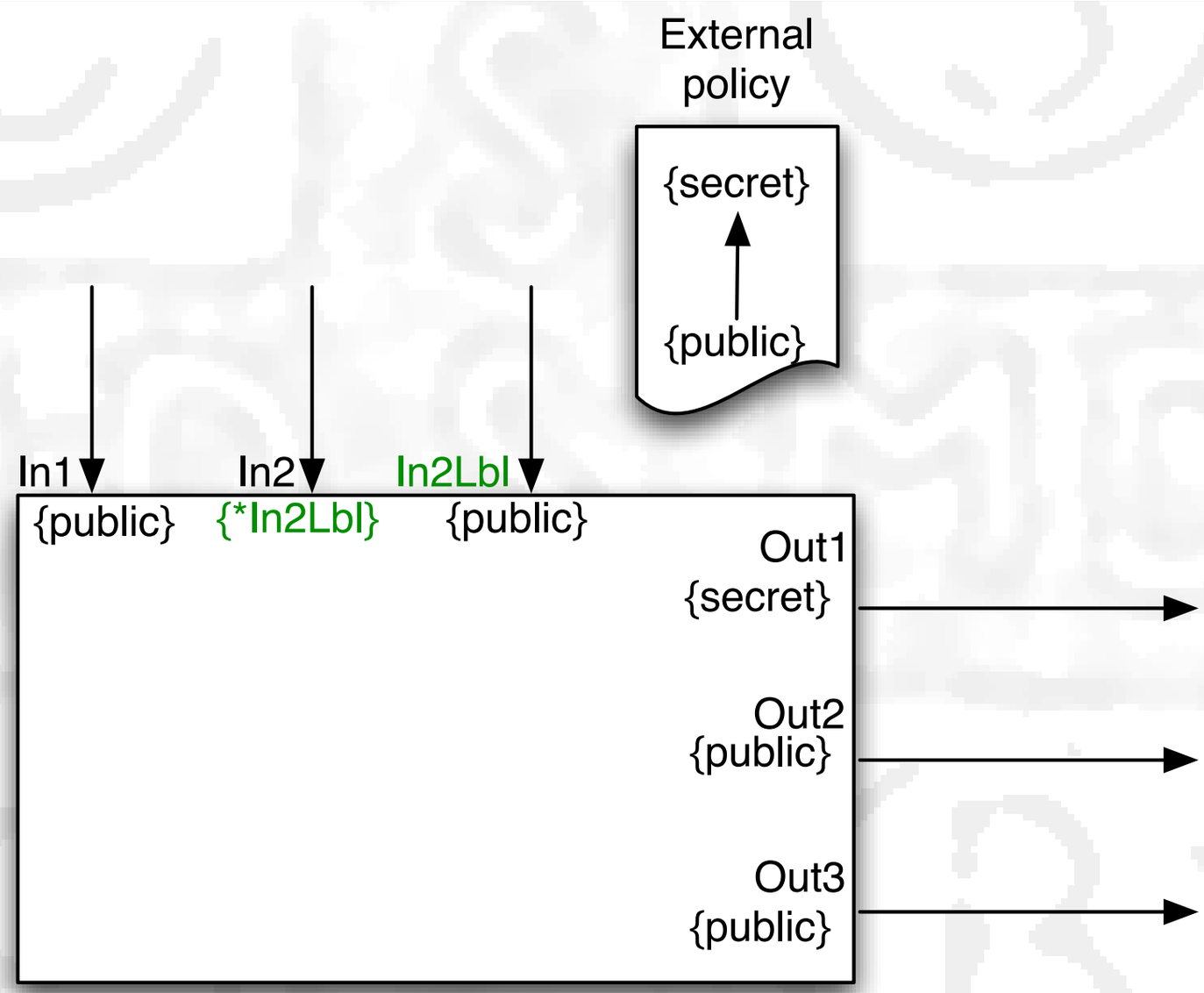
1. (static) type labels
2. label semantics

(Also dynamic type labels)



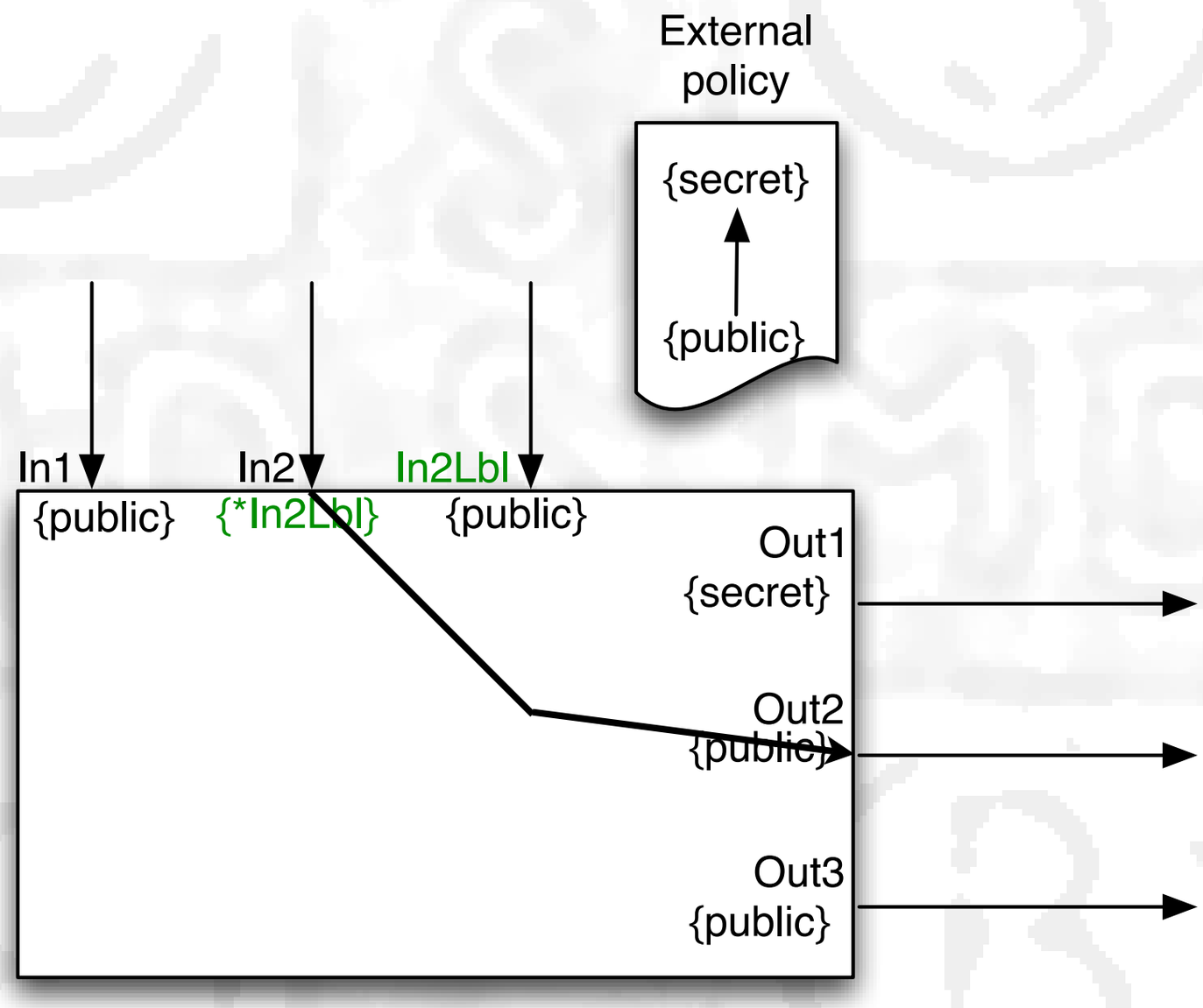
1. (static) type labels
2. label semantics

(Also dynamic type labels)



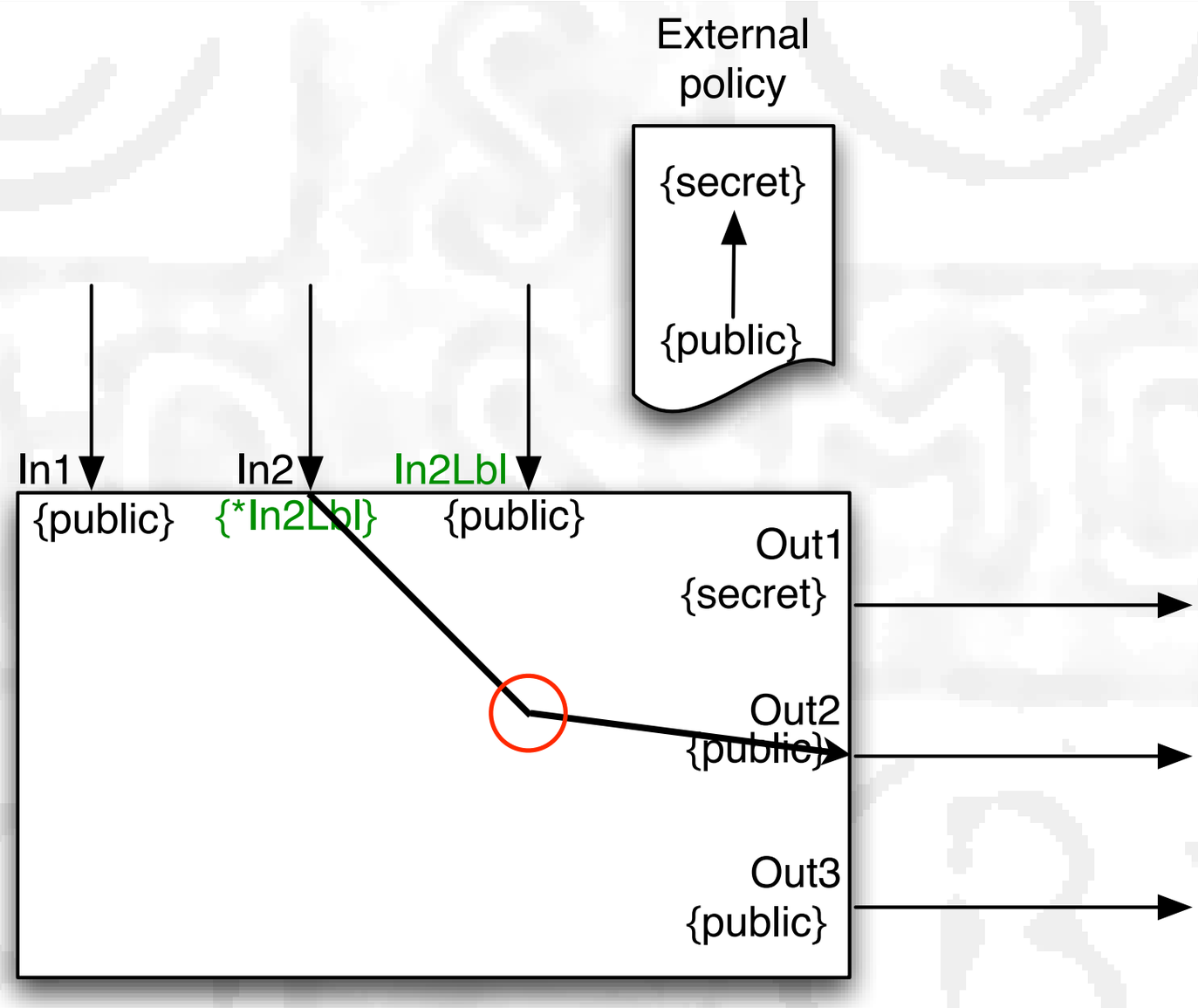
1. (static) type labels
2. label semantics

(Also dynamic type labels)



1. (static) type labels
2. label semantics

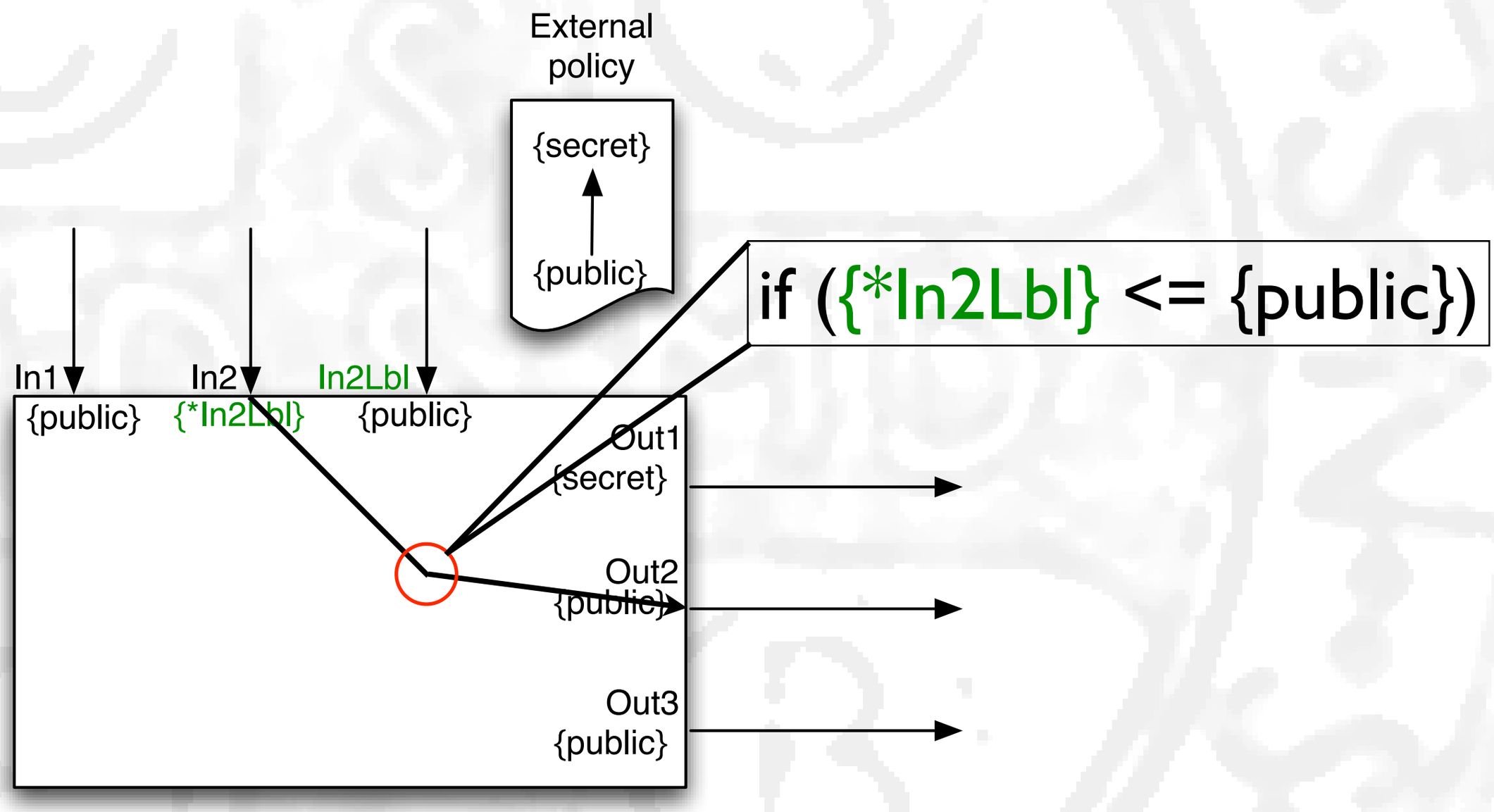
(Also dynamic type labels)



1. (static) type labels
2. label semantics

Compiler requires dynamic check (ensures complete mediation)

(Also dynamic type labels)



1. (static) type labels
2. label semantics

Compiler requires dynamic check (ensures complete mediation)

Why Jif?

Why Jif?

Salmonella outbreak still a sticky mystery

All Peter Pan peanut butter bought since May 2006 should be discarded

MSNBC News Services
Updated: 10:10 a.m. CT Feb 19, 2007

ATLANTA - All Peter Pan peanut butter bought since May 2006 should be discarded, the U.S. Food and Drug Administration said on Friday in a statement broadening its warning about salmonella-contaminated peanut butter.

More than 290 people from 39 states have become ill in the food poisoning outbreak since August, and 46 have been hospitalized, the U.S. Centers for Disease Control and Prevention reported.

As government scientists struggled to pinpoint the source of a salmonella outbreak linked to peanut butter, the first lawsuits were filed against ConAgra Foods Inc. on Friday.

Story continues below ↓

Federal health investigators said they strongly

Why Jif?

Salmonella outbreak still a sticky mystery

All Peter Pan peanut butter bought since May 2006 should be discarded

MSNBC News Services
Updated: 10:10 a.m. CT F

ATLANTA - All Peter
since May 2006 sho
Food and Drug Adm
a statement broade
salmonella-contami

More than 290 peop
become ill in the fo
August, and 46 hav
U.S. Centers for Di
Prevention reported

As government scie
the source of a salmonella outbreak linked to peanut butter, the first lawsuits were filed against ConAgra Foods Inc. on Friday.

Story continues below ↓

advertisement

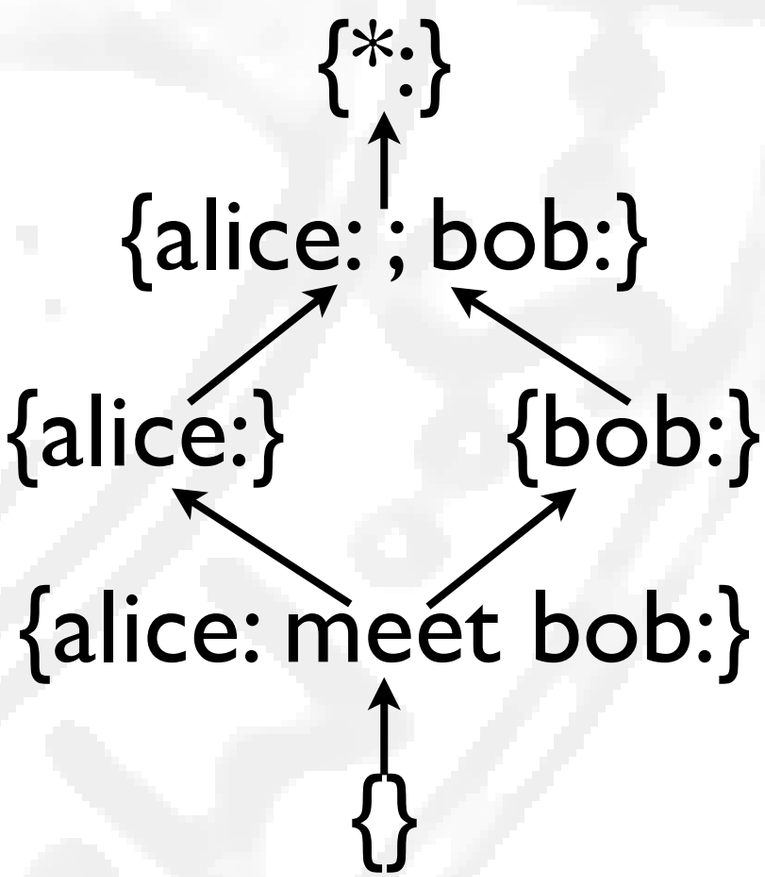
Federal health investigators said they strongly

Why Jif?

- Most mature security-typed language
- Full-featured variant of Java, has IDE (Jifclipse)
- Used to build some applications
  - battleship
  - mental poker (jifpoker)
  - email client (JPmail)
  - firewall (Flowwall)
  - servlet framework, servlets (SIF)
  - e-voting system (Civitas)
  - ...

Basic Jif

- Complete mediation - tracks all information flows
- Decentralized Label Model
  (could be parameterized out, see Fable)
  - partial ordering on principals
  - labels composed of principals
  - labels form a lattice
  - confidentiality ({alice:})
  - integrity ({alice!:})
  - both ({alice: ; alice!:})
  - join ({alice: ; bob:})
  - meet ({alice: meet bob:})



Exercise

- Look at the Java code and determine whether any of the SecretMessages.funX leak information from aliceInstructions into bobInstructions
- STL Goal: mark data with labels to ensure no leakage (and ensure complete mediation over dynamic labels)
- Jif Goal: A method's header should fully characterize all the flows in the method (facilitates compositionality)

Multi-level Data Structures


```
public class SecretMessages[label alice, label bob]
{
    String{*alice} aliceInstructions;
    String{*bob} bobInstructions;

    public SecretMessages(String{*alice} ai, String{*bob} bi) {
        aliceInstructions = ai;
        bobInstructions = bi;
    }
    public String{*alice} getAliceMsg() {
        return aliceInstructions;
    }
    public String{*bob} getBobMsg() {
        return bobInstructions;
    }
}
```

Explicit Flow Prevention


```
public class SecretMessages[principal alice, principal bob]
{
    String{alice:} aliceInstructions;
    String{bob:} bobInstructions;

    public SecretMessages(String{alice:} ai, String{bob:} bi) {
        aliceInstructions = ai;
        bobInstructions = bi;
    }
    ...

    public String{bob:} leak() {
        bobInstructions = aliceInstructions;
        return bobInstructions;
    }
}
```

- bob cannot read alice's data

Explicit Flow Prevention

```
public class SecretMessages[principal alice, principal bob]
{
    String{alice:} aliceInstructions;
    String{bob:} bobInstructions;

    public SecretMessages(String{alice:} ai, String{bob:} bi) {
```

```
Unsatisfiable constraint:
  rhs.nv <= label of field bobInstructions
    {alice: ; _!: _; this; caller_pc} <= {bob: }
in environment
  [{this} <= {caller_pc}]

Label Descriptions
------------------
 - rhs.nv = label of successful evaluation of right hand of assignment
 - rhs.nv = {alice: ; _!: _; this; caller_pc}
 - label of field bobInstructions = {bob: }
 - this = label of the special variable "this"
 - caller_pc = The pc at the call site of this method (bounded above by {bob: })

 More information is revealed by the successful evaluation of the right hand side of the
assignment than is allowed to flow to the field bobInstructions.
```

Implicit Flow Prevention


```
public class SecretMessages[label alice, label bob]
{
    String{*alice} aliceInstructions;
    String{*bob} bobInstructions;

    public SecretMessages(String{*alice} ai, String{*bob} bi) {
        aliceInstructions = ai;
        bobInstructions = bi;
    }
  ...
    public String{*bob} implicitLeak() {
        try {
            if (aliceInstructions.equals("Attack at dawn"))
                bobInstructions = "Attack at dawn";
        } catch (NullPointerException e) {}
        return bobInstructions;
    }
}
```

Implicit Flow Prevention 2


```
public String{*bob} implicitLeak2() {
    try {
        aliceInstructions.trim();
    } catch (NullPointerException e) {
        bobInstructions = null;
    }
    return bobInstructions;
}
```

```
public String{*bob} implicitLeak3() {
    try {
        aliceInstructions.trim();
        bobInstructions = null;
    } catch (NullPointerException e) {}
    return bobInstructions;
}
```

Implicit Flow Prevention 3?

```
public String{*bob} implicitLeak3() {
    try {
        aliceInstructions.trim();
        bobInstructions = null;
    } catch (NullPointerException e) {}
    return bobInstructions;
}
```

```
public String{*bob} implicitLeak3() {
    try {
        bobInstructions = null;
        aliceInstructions.trim();
    } catch (NullPointerException e) {}

    return bobInstructions;
}
```

Swapped order

Side-Effects Leakage


```
public class SecretMessages[label alice, label bob]
{
    String{*alice} aliceInstructions;
    String{*bob} bobInstructions;

    public String{*alice} getAliceMsg() {
        return aliceInstructions;
    }

    public void setAliceMsg() {
        aliceInstructions = "";
    }
}
```

- some information is stored at level {*alice}
- must be careful where this is called

Side-Effects Leakage

```
public class SecretMessages[label alice, label bob]
{
    String{*alice} aliceInstructions;
    String{*bob} bobInstructions;

    public String{*alice} getAliceMsg() {
```

```
Unsatisfiable constraint:
  rhs.nv <= label of field aliceInstructions
    {caller_pc; this} <= {alice}
in environment
    [{this} <= {caller_pc}]

Label Descriptions
------------------
 - rhs.nv = label of successful evaluation of right hand of assignment
 - rhs.nv = {caller_pc; this}
 - label of field aliceInstructions = {alice}
 - caller_pc = The pc at the call site of this method (bounded above by {*: })
 - this = label of the special variable "this"
 - alice = label parameter alice of class SecretMessages

 More information is revealed by the successful evaluation of the right hand side of the
assignment than is allowed to flow to the field aliceInstructions.
```

Side-Effects Leakage


```
public void sideEffectLeak() {
    try {
        if (bobInstructions.equals("Attack at Dawn"))
            setAliceMsg();
    } catch (NullPointerException e) {}
}
```

- To preserve compositionality, Jif requires method headers to cover all information flows

Side-Effects Leakage


```
public void sideEffectLeak() {
    try {
        if (bobInstructions.equals("Attack at Dawn"))
            setAliceMsg();
    } catch (NullPointerException e) {}
}

public class SecretMessages[label alice, label bob]
{
    String{*alice} aliceInstructions;
    String{*bob} bobInstructions;

    public String{*alice} getAliceMsg() {
        return aliceInstructions;
    }

    public void setAliceMsg{*alice}() {
        aliceInstructions = "";
    }
}
```

Side-Effects Leakage

```
public void sideEffectLeak() {
    try {
        if (bobInstructions.equals("Attack at Dawn"))
            setAliceMsg();
    } catch (NullPointerException e) {}
}

publ
{

}
```

Unsatisfiable constraint:
pc_call <= callee_PC_bound
{bob; this; caller_pc} <= {alice}
in environment
[{this} <= {caller_pc}]

Label Descriptions

- pc_call = label of the program counter at this call site
- pc_call = {bob; this; caller_pc}
- callee_PC_bound = lower bound on the side effects of the method public void setAliceMsg()
- callee_PC_bound = {alice}
- bob = label parameter bob of class SecretMessages
- this = label of the special variable "this"
- caller_pc = The pc at the call site of this method (bounded above by {*: })
- alice = label parameter alice of class SecretMessages

Calling the method at this program point may reveal too much information to the receiver of the method call. public void setAliceMsg() can only be invoked if the invocation will reveal no more information than the callee's begin label, callee_PC_bound. However, execution reaching this program point may depend on information up to the PC at this program point: pc_call.

Dynamic Labels


```
import java.io.PrintStream;

public class SpyMessage
{
    final public label{} lbl;
    String{*lbl} secret;

    public SpyMessage(principal{} user) {
        lbl = new label{user:};
        this.secret = "Attack at dawn";
    }

    public String{*lbl} getSecretOutput() {
        return secret;
    }
}
```

- Labels have labels!
- lbl refers to the label on the label
- *lbl refers to the label itself

Dynamic Mediation Point


```
public class Main {

    public static void main{}(principal{} user, String[] args) {
        final SpyMessage{user:} myMessage = new SpyMessage(user);

        PrintStream[{user:}] outS = null;
        jif.runtime.Runtime[user] rt = null;
        try {
            rt = jif.runtime.Runtime[user].getRuntime();
            outS = rt != null ? rt.stdout(new label{user:}) : null;
        }
        catch (java.lang.SecurityException e) {}

        if (myMessage.lbl <= new label{user:})
            if (outS != null)
                outS.println(myMessage.getSecretOutput());

        DEBUG.println("That's all.");
    }
}
```

Dynamic Mediation Point


```
public class Main {

    public static void main{}(principal{} user, String[] args) {
        final SpyMessage{user:} myMessage = new SpyMessage(user);

        PrintStream[{user:}] outS = null;
        jif.runtime.Runtime[user] rt = null;
        try {
            rt = jif.runtime.Runtime[user].getRuntime();
            outS = rt != null ? rt.stdout(new label{user:}) : null;
        }
        catch (java.lang.SecurityException e) {}

        if (myMessage.lbl <= new label{user:})
            if (outS != null)
                outS.println(myMessage.getSecretOutput());

        DEBUG.println("That's all.");
    }
}
```

Label required by outS

Label on myMessage

Some Challenges Remain

- Policy management
- Dynamic channels
- Compliance with system security
- Automated label inference
- Reduce false positives
- Clearer error messages, automated fixes
- Better declassification

Try It!

- Jif: http://www.cs.cornell.edu/jif
- Jifclipse: http://siis.cse.psu.edu/jifclipse

Try It!

- Jif: http://www.cs.cornell.edu/jif
- Jifclipse: http://siis.cse.psu.edu/jifclipse

QUESTIONS??