



# **Anti Inference Hub 1.0 User's Guide**

**Sari Haj Hussein  
Chalmers University of Technology  
Lund University**

# Table of Contents

1. Introduction	1
1.1. What is Anti Inference Hub?	1
1.2. Some Intended Purposes	1
1.3. Features	1
1.4. Open Source Software	1
1.5. What Anti Inference Hub is Not	2
1.6. Supported Operating Systems	2
1.7. Where to Get Anti Inference Hub?	2
1.8. Development and Maintenance of Anti Inference Hub	2
1.9. Reporting Problems and Getting Help	2
2. Installing Anti Inference Hub	3
2.1. Introduction	3
2.2. Obtaining the Binary Package	3
2.3. Installing the Binary Package Under Windows	3
2.4. Installing the Binary Package Under Linux	3
2.5. Uninstalling Anti Inference Hub	3
3. User Interface	4
3.1. Introduction	4
3.2. Starting Anti Inference Hub	4
3.3. The Hub Window	4
3.3.1. The Menu for the Hub Window	4
3.3.1.1. The File Menu for the Hub Window	5
3.3.1.2. The Help Menu for the Hub Window	6
3.3.2. The Activity Log Panel	6
3.3.3. The Buttons Panel for the Hub Window	7
3.4. The Hub Client Window	7
3.4.1. The Menu for the Hub Client Window	8
3.4.1.1. The File Menu for the Hub Client Window	8
3.4.1.2. The Help Menu for the Hub Client Window	9
3.4.2. The Hub Host Name Panel	10
3.4.3. The Input to Hub Panel	10
3.4.4. The Output from Hub Panel	11
3.4.5. The Buttons Panel for the Hub Client Window	11
4. Connecting with the Database	12
4.1. Introduction	12
4.2. The Setup Database Connection Box	12
5. Setting Inference Channels	13
5.1. Introduction	13
5.2. The Setup Inference Channels Box	13
5.3. The Add Channel Box	14
6. Initializing Keys	15
6.1. Introduction	15
6.2. The Initialize Keys Box	15
7. Adjusting Super Clients	17
7.1. Introduction	17
7.2. The Adjust Super Clients Box	17
7.3. The Add Super Client Box	18
8. The Inference Problem: Demystified	19

8.1. Introduction	19
8.2. Specific Inference Problems	19
8.2.1. Inference from Queries Based on Sensitive Data	19
8.2.2. Statistical Databases	19
8.2.3. Inference from Data Combined with Metadata	20
8.2.3.1. Key Integrity	20
8.2.3.2. Functional and Multivalued Dependencies	21
8.2.3.3. Value Constraints	21
8.2.3.4. Classification Constraints	21
8.3. General Characterizations of the Inference Problem	21
8.4. Tools and Techniques for Dealing with Inference Channels	22
8.5. References	22
9. Technology behind Anti Inference Hub	24
10. The Sample Database	25
10.1. Introduction	25
10.2. Location of the Sample Database	25
10.3. Creating the Sample Database	26
10.4. Our Test Environment	26
11. Anti Inference Hub in Action	27
11.1. Introduction	27
11.2. The Inference Attempt to Address	27
11.3. Thwarting the Inference Attempt	27
A. Anti Inference Hub License	33
B. Anti Inference Hub Code Disclaimer	43

# Chapter 1. Introduction

## 1.1. What is Anti Inference Hub?

Anti Inference Hub is the first dynamic query processing engine that defends against the Inference Problem in Multilevel Databases by integrating smoothly with common DBMSs (Oracle, PostgreSQL, and MySQL), and monitoring queries submitted by end users.

## 1.2. Some Intended Purposes

Here are some example usages of Anti Inference Hub:

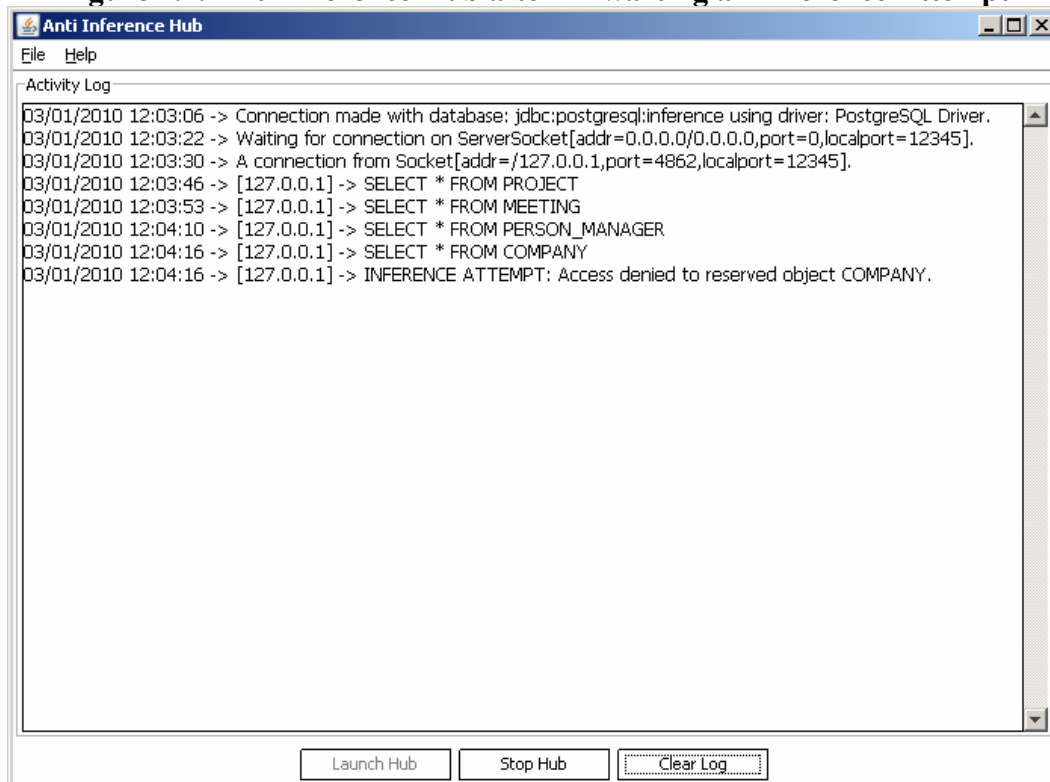
- Database designers use it to estimate the security of their database design.
- Database administrators use it to protect sensitive data stored in the database.
- Network security engineers use it to examine security problems, and identify attackers.
- People use it to learn more about the Inference Problem, and Database Security in whole.

## 1.3. Features

Following are some of the many features provided by Anti Inference Hub:

- Available for any operating system with Java installed.
- Integrates easily with Oracle, PostgreSQL, and MySQL.
- Fast query processing and analyzing.
- Results for safe queries are provided in a platform independent XML format.

**Figure 1.1. Anti Inference Hub after Thwarting an Inference Attempt**



## 1.4. Open Source Software

Anti Inference Hub is an open source software project, and is released under the GNU General Public License (GPL). You can freely use Anti Inference Hub on any number of computers you like, without

worrying about license keys or fees or such. In addition, all source code is freely available under the GPL. Because of that, it is very easy for people to contribute to the project.

### **1.5. What Anti Inference Hub is Not**

- Anti Inference Hub is not an intrusion prevention system (IPS). It only prevents inference attacks.
- Anti Inference Hub does not interfere with network traffic.

### **1.6. Supported Operating Systems**

Anti Inference Hub should run on any operating system with Java installed.

### **1.7. Where to Get Anti Inference Hub?**

You can get the latest copy of Anti Inference Hub from its website: <http://aih.sourceforge.net>.

### **1.8. Development and Maintenance of Anti Inference Hub**

Anti Inference Hub was developed by Sari Haj Hussein. Ongoing development and maintenance of Anti Inference Hub are also handled by Sari Haj Hussein. Anti Inference Hub is an open source software project, and is released under the GNU General Public License (GPL). All source code is freely available under the GPL. You are welcome to modify Anti Inference Hub to suit your own needs, and it would be appreciated if you contribute your improvements back to us. The Anti Inference Hub source code is available from Anti Inference Hub website: <http://aih.sourceforge.net>.

### **1.9. Reporting Problems and Getting Help**

If you have problems, or need help with Anti Inference Hub, please send an email describing the issue to [angyjoo@yahoo.com](mailto:angyjoo@yahoo.com).

## **Chapter 2. Installing Anti Inference Hub**

### **2.1. Introduction**

To use Anti Inference Hub, you must obtain a binary package for your operating system, then install it into its final destinations.

### **2.2. Obtaining the Binary Package**

You can obtain the binary package from Anti Inference Hub website: <http://aih.sourceforge.net>. It should be named something like: AntiInferenceHub<version>.zip.

### **2.3. Installing the Binary Package Under Windows**

Unpack Anti Inference Hub binary package into a directory of choice.

### **2.4. Installing the Binary Package Under Linux**

Same as in Windows, unpack Anti Inference Hub binary package into a directory of choice.

### **2.5. Uninstalling Anti Inference Hub**

Remove the directory where you unpacked Anti Inference Hub binary package.

## Chapter 3. User Interface

### 3.1. Introduction

By now you have installed Anti Inference Hub and are most likely keen to get started exploring the user interface.

### 3.2. Starting Anti Inference Hub

To start Anti Inference Hub under Windows, use:

- The file RunHub.bat to start the Hub.
- The file RunHubClient.bat to start the Hub Client.

And to start Anti Inference Hub under Linux, use:

- The file RunHub.sh to start the Hub.
- The file RunHubClient.sh to start the Hub Client.

All of these files are included in Anti Inference Hub binary package.

### 3.3. The Hub Window

Let's look at the Hub Window.

**Figure 3.1. The Hub Window**



The Hub Window consists of parts that are commonly known from many other GUI programs.

1. The Menu that is used to configure the Hub.
2. The Activity Log Panel that displays a summary of the activities done by the Hub.
3. The Buttons Panel that controls whether the Hub is on or off, and clears the log.

#### 3.3.1. The Menu for the Hub Window

The Menu for the Hub window sits on top of the window. Menu items will be grayed out if the corresponding feature is not available. For example, you cannot setup inference channels before setting up a database connection.

**Figure 3.2. The Menu for the Hub Window**

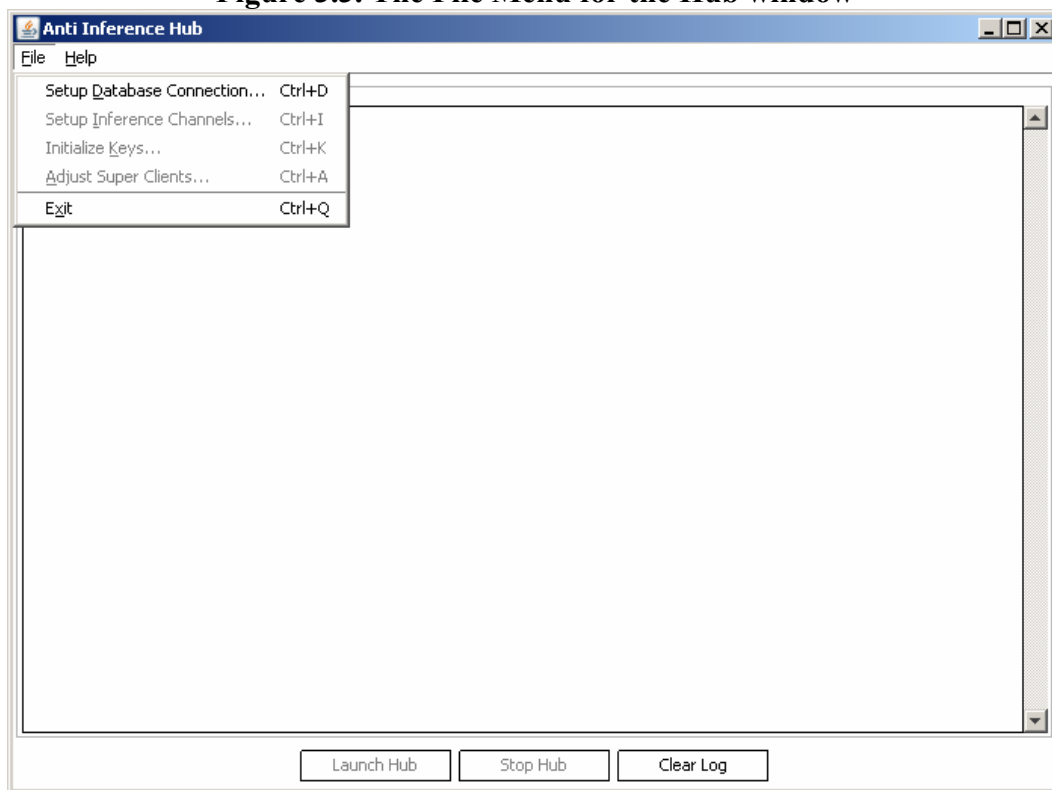


The Menu contains the following items:

- File This menu contains items to setup database connection, inference channels, initialize keys, adjust super clients, and exit the program.
- Help This menu contains items to help the user, and the usual about dialog.

### 3.3.1.1. The File Menu for the Hub Window

**Figure 3.3. The File Menu for the Hub window**



**Table 3.1. Items in the File Menu for the Hub window**

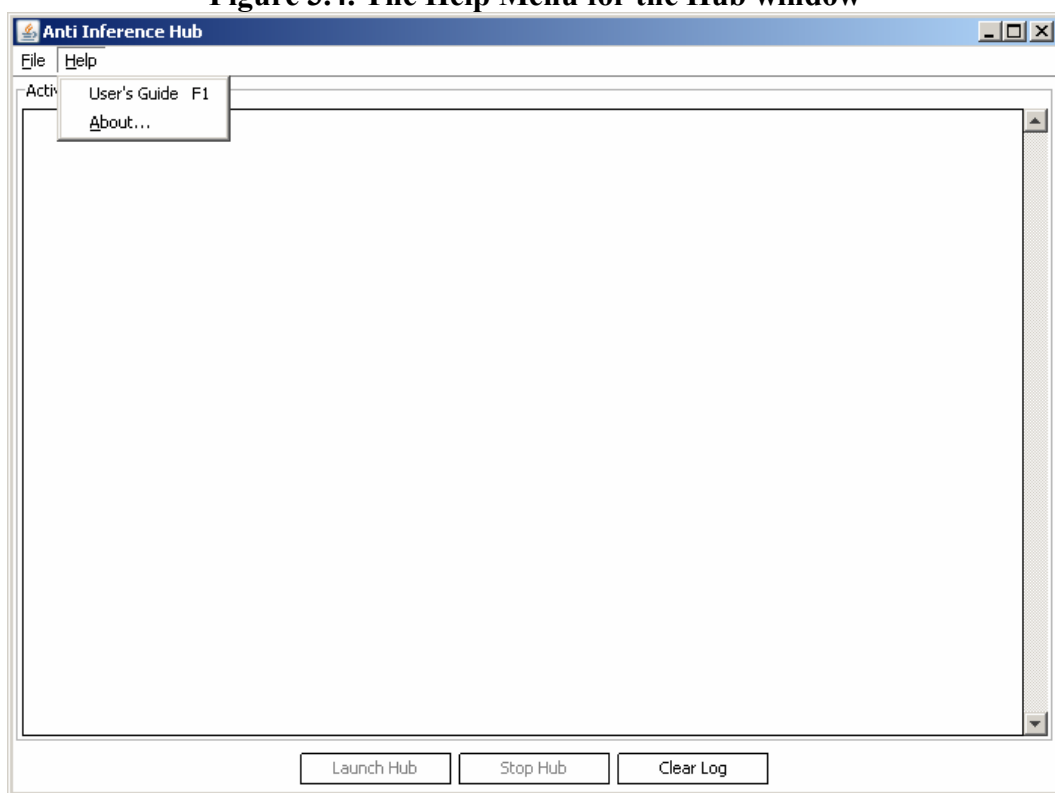
Menu Item	Accelerator	Description
Setup Database Connection...	Ctrl+D	This menu item brings up the Setup Database Connection Box that allows you to setup connection with supported DBMSs (Oracle, PostgreSQL, and MySQL).
Setup Inference Channels...	Ctrl+I	This menu item brings up the Setup Inference Channels Box that allows you to view, add, and remove inference channels for the database.



Initialize Keys...	Ctrl+K	This menu item brings up the Initialize Keys Box that allows you to view, initialize, and remove keys for inference channels objects.
Adjust Super Clients...	Ctrl+A	This menu item brings up the Adjust Super Clients Box that allows you to view, add, and remove super clients connecting with the Hub.
Exit	Ctrl+Q	This menu item quits the Hub.

### 3.3.1.2. The Help Menu for the Hub Window

**Figure 3.4. The Help Menu for the Hub window**



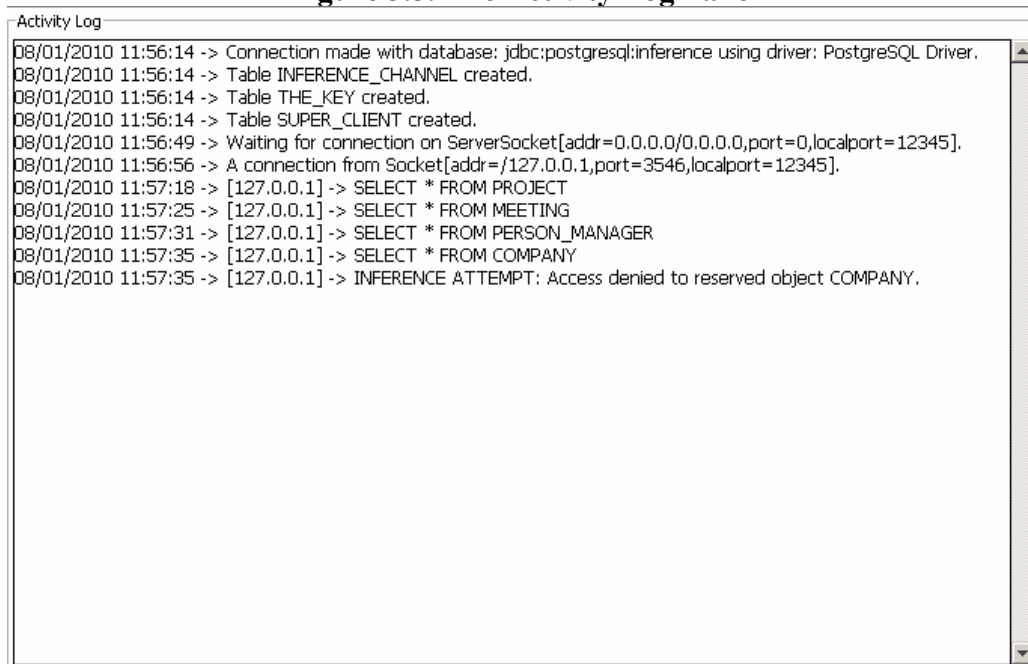
**Table 3.2. Items in the Help Menu for the Hub window**

Menu Item	Accelerator	Description
User's Guide	F1	This menu item opens the user's guide.
About...		This menu item brings up the About Box that provides some information on Anti Inference Hub, such as license, and third party packages used.

### 3.3.2. The Activity Log Panel

The Activity Log Panel displays a summary of the activities done by the Hub.

**Figure 3.5. The Activity Log Panel**

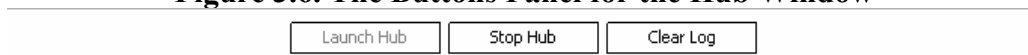


The activities logged in the Activity Log Panel are dated and timed, and they can be one of the following:

1. Establishing a successful connection with the DBMS.
2. Successful creation of statistical tables needed for query processing. The tables are INFERENCE\_CHANNEL, THE\_KEY, and SUPER\_CLIENT.
3. Listening for connection requests from clients on a default port (which is 12345).
4. Queries received by the Hub, and clients who sent them.
5. Inference attempts thwarted by the Hub, and clients who initiated the attacks.
6. Inference attempts permitted by the Hub, and super clients who were allowed to infer.

### 3.3.3. The Buttons Panel for the Hub Window

**Figure 3.6. The Buttons Panel for the Hub Window**



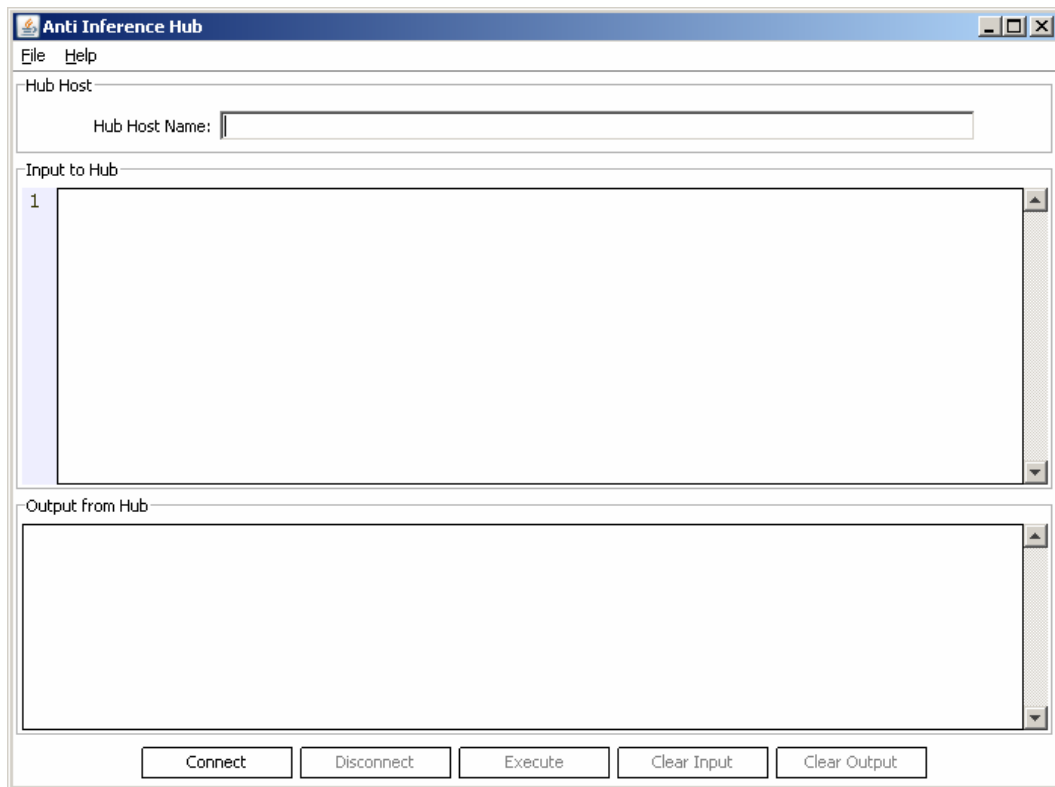
**Table 3.3. Buttons in the Buttons Panel for the Hub Window**

Button	Description
Launch Hub	This button is used to launch the Hub so that it listens for connection requests from clients.
Stop Hub	This button is used to stop the Hub so that no more connections are allowed.
Clear Log	This button is used to clear the content of the Activity Log Panel.

## 3.4. The Hub Client Window

Let's look at the Hub Client Window.

**Figure 3.7. The Hub Client Window**



The Hub Client Window consists of parts that are commonly known from many other GUI programs.

1. The Menu that has fewer functionalities than that of the Hub Window.
2. The Hub Host Name Panel that is used to enter the host name of the Hub the Hub Client will connect to.
3. The Input to Hub Panel that is used to enter SQL queries.
4. The Output from Hub Panel that is used to display Hub responses to SQL queries.
5. The Buttons Panel that controls whether the Hub Client is connected to the Hub or not, executes queries, and clears input/output.

### 3.4.1. The Menu for the Hub Client Window

The menu for the Hub Client Window sits on top of the window.

**Figure 3.8. The Menu for the Hub Client Window**

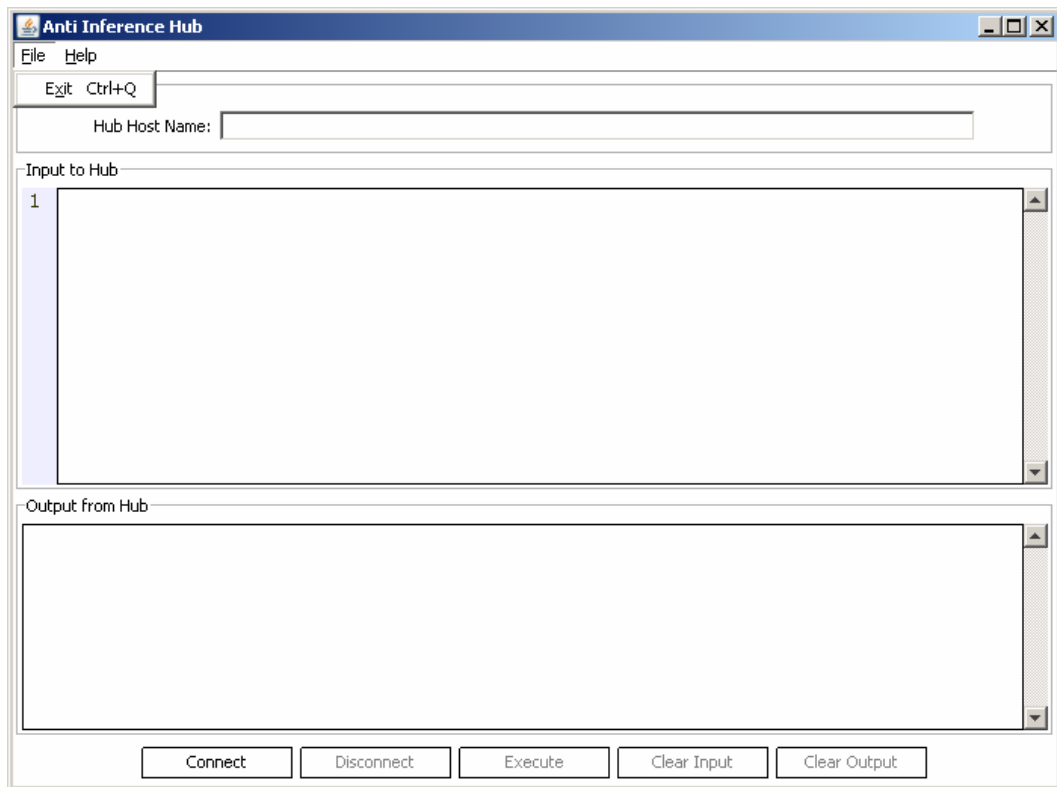


The Menu contains the following items:

- File     This menu contains an item to exit the program.
- Help    This menu contains items to help the user, and the usual about dialog.

#### 3.4.1.1. The File Menu for the Hub Client Window

**Figure 3.9. The File menu for the Hub Client Window**

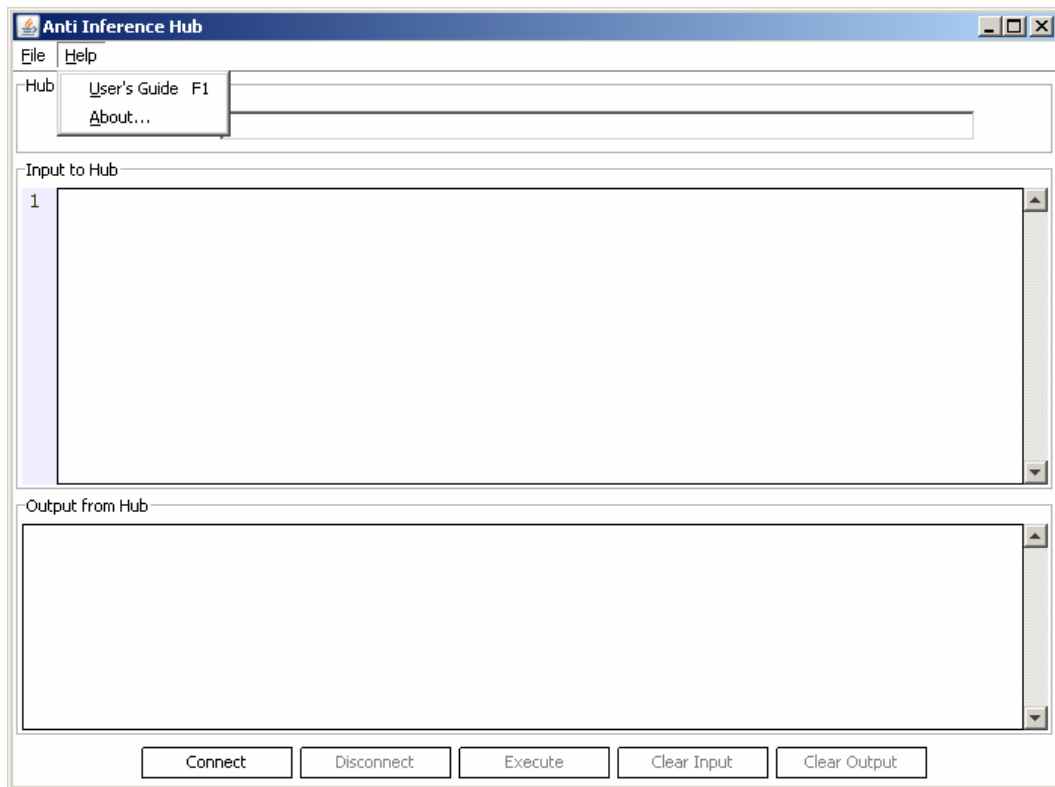


**Table 3.4. Items in the File menu for the Hub Client Window**

Menu Item	Accelerator	Description
Exit	Ctrl+Q	This menu item quits the Hub Client.

#### 3.4.1.2. The Help Menu for the Hub Client Window

**Figure 3.10. The Help Menu for the Hub Client window**



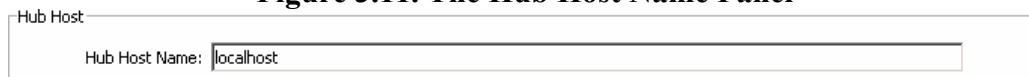
**Table 3.5. Items in the Help Menu for the Hub Client window**

Menu Item	Accelerator	Description
User's Guide	F1	This menu item opens the user's guide.
About...		This menu item brings up the About Box that provides some information on Anti Inference Hub, such as license, and third party packages used.

### 3.4.2. The Hub Host Name Panel

The Hub Host Name Panel is used to enter the host name of the Hub the Hub Client will connect to. The host name can be either a machine name or a textual representation of an IP address.

**Figure 3.11. The Hub Host Name Panel**



### 3.4.3. The Input to Hub Panel

The Input to Hub Panel is used to enter SQL queries. It comes with an out-of-the-box SQL syntax highlighter.

**Figure 3.12. The Input to Hub Panel**

```

1 SELECT PERSON_MANAGER_NAME, COMPANY_NAME
2 FROM PERSON_MANAGER, COMPANY
3 WHERE PERSON_MANAGER.WORK_FOR = COMPANY.COMPANY_ID
4 AND PERSON_MANAGER.PERSON_MANAGER_NAME = 'MANAGER 1';

```

### 3.4.4. The Output from Hub Panel

The Output from Hub Panel is used to display Hub responses to SQL queries. Results for safe SQL queries are provided in a platform independent XML format, whereas, notifications of blocked queries, and other error messages are provided in a text format.

**Figure 3.13. The Output from Hub Panel**

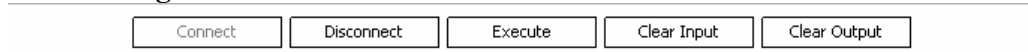
```

<?xml version="1.0" encoding="UTF-8"?>
<Results>
  <Row>
    <person_manager_name>MANAGER 1</person_manager_name>
    <company_name>COMPANY 1</company_name>
  </Row>
</Results>

```

### 3.4.5. The Buttons Panel for the Hub Client Window

**Figure 3.14. The Button Panel for the Hub Client Window**



**Table 3.6. Buttons in the Buttons Panel for the Hub Client Window**

Button	Description
Connect	This button is used to connect the Hub Client to the Hub specified in the Hub Host Name field.
Disconnect	This button is used to disconnect the Hub Client from the Hub.
Execute	This button is used to send the content (SQL query) of the Input to Hub Panel to the Hub to be executed.
Clear Input	This button is used to clear the content of the Input to Hub Panel.
Clear Output	This button is used to clear the content of the Output from Hub Panel.

## Chapter 4. Connecting with the Database

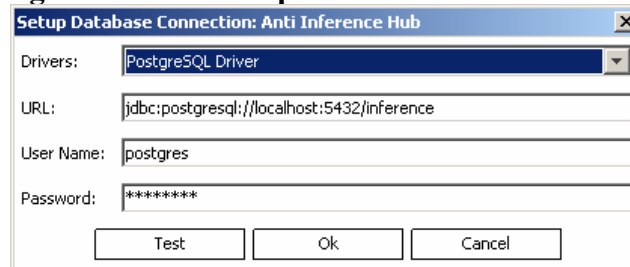
### 4.1. Introduction

Connecting the Hub with the database is the first step you should do to get it to work. Anti Inference Hub can protect databases designed using Oracle, PostgreSQL, and MySQL.

### 4.2. The Setup Database Connection Box

The Setup Database Connection Box is accessible through the Setup Database Connection menu item in the Hub Window, or the accelerator Ctrl+D.

**Figure 4.1. The Setup Database Connection Box**



**Table 4.1. Components in the Setup Database Connection Box**

Component	Description
Drivers Combo Box	This combo box is used to specify the JDBC driver the Hub will use to connect to the database. Depending on the DBMS you are using, this will be Oracle Thin Driver, or PostgreSQL Driver, or MySQL Driver.
URL Text Field	This field is used to specify the URL of the database the Hub will connect to. The URL depends on the driver specified in the Drivers Combo Box.
User Name Text Field	This field is used to specify the user name under which the Hub will connect to the database.
Password Text Field	This field is used to specify the password under which the Hub will connect to the database.
Test Button	This button is used to test the connection between the Hub and the database.
OK Button	This button is used to save the connection between the Hub and the database for ongoing usage by the Hub. It also closes the Setup Database Connection Box.
Cancel Button	This button is used to close the Setup Database Connection Box without saving any thing.

## Chapter 5. Setting Inference Channels

### 5.1. Introduction

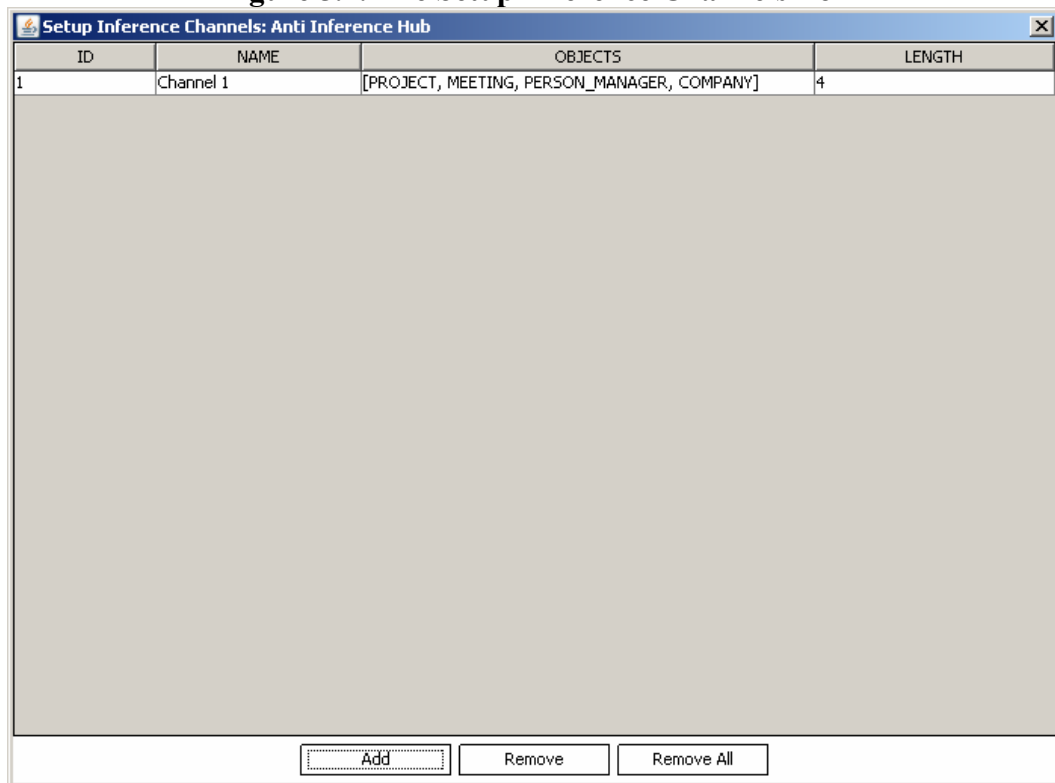
You should manually locate inference channels in the database so that the Hub can distinguish safe queries from unsafe ones. You should thoroughly consider your database design, and try to be as accurate as possible when doing this, otherwise, the Hub may block responses to safe queries!

Over time, many algorithms for locating inference channels were proposed by researches; however, all of them tend to generate inaccurate and unsatisfying results, therefore, Anti Inference Hub does not implement any of these algorithms, rather, it transfers the task of locating inference channels to the database designer (or the database security specialist). When a sound algorithm for locating inference channels has been proposed in the scientific community, we will make it available in Anti Inference Hub. For further information about this issue, please refer to Section 8.4. Tools and Techniques for Dealing with Inference Channels.

### 5.2. The Setup Inference Channels Box

The Setup Inference Channels Box is accessible through the Setup Inference Channels menu item in the Hub Window, or the accelerator Ctrl+I.

**Figure 5.1. The Setup Inference Channels Box**



**Table 5.1. Components in the Setup Inference Channels Box**

Component	Description
Inference Channels Table	This table lists currently located inference channels in the database. Every inference channel is characterized by its ID, name, objects contained in it, and its length.
Add Button	This button is used to locate a new inference channel in the database. When clicked, it brings



	up the Add Channel Box.
Remove Button	This button is used to remove the selected inference channel in the Inference Channels Table.
Remove All Button	This button is used to remove all inference channels in the Inference Channels Table whether selected or not.

### 5.3. The Add Channel Box

The Add Channel Box is used to locate a new inference channel in the database.

**Figure 5.2. The Add Channel Box**

The screenshot shows a dialog box titled "Add Channel". It has three main input areas: "Inference Channel Name:" with a text field containing "Channel 1"; "Database Objects:" with a dropdown menu showing "COMPANY"; and "Inference Channel Objects:" with a list box containing "PROJECT", "MEETING", and "PERSON\_MANAGER". At the bottom of the dialog are four buttons: "Include", "Exclude", "Clear", and "OK".

**Table 5.2. Components in the Add Channel Box**

Component	Description
Inference Channel Name Text Field	This field is used to specify the name of the new inference channel. A name must be specified.
Database Objects Combo Box	This combo box is used to specify a database object to include in the new inference channel. One object at least must be included.
Inference Channel Objects List	This list displays the current selection of database objects that will be included in the new inference channel.
Include Button	This button is used to include the selected database object from Database Objects Combo Box in the Inference Channel Objects List.
Exclude Button	This button is used to exclude the selected database objects in the Inference Channel Objects List. Note that more than one database object can be selected in the Inference Channel Objects List.
Clear Button	This button is used to clear the content of the Inference Channel Objects List.
OK Button	This button is used to save the new inference channel in the database. It also closes the Add Channel Box.

# Chapter 6. Initializing Keys

## 6.1. Introduction

Initializing keys for inference channels objects is an automatic process that you should consider right after locating inference channels in the database. For further information about the purpose of the keys, please refer to Chapter 9. Technology behind Anti Inference Hub.

## 6.2. The Initialize Keys Box

The Initialize Keys Box is accessible through the Initialize Keys menu item in the Hub Window, or the accelerator Ctrl+K.

Figure 6.1. The Initialize Keys Box

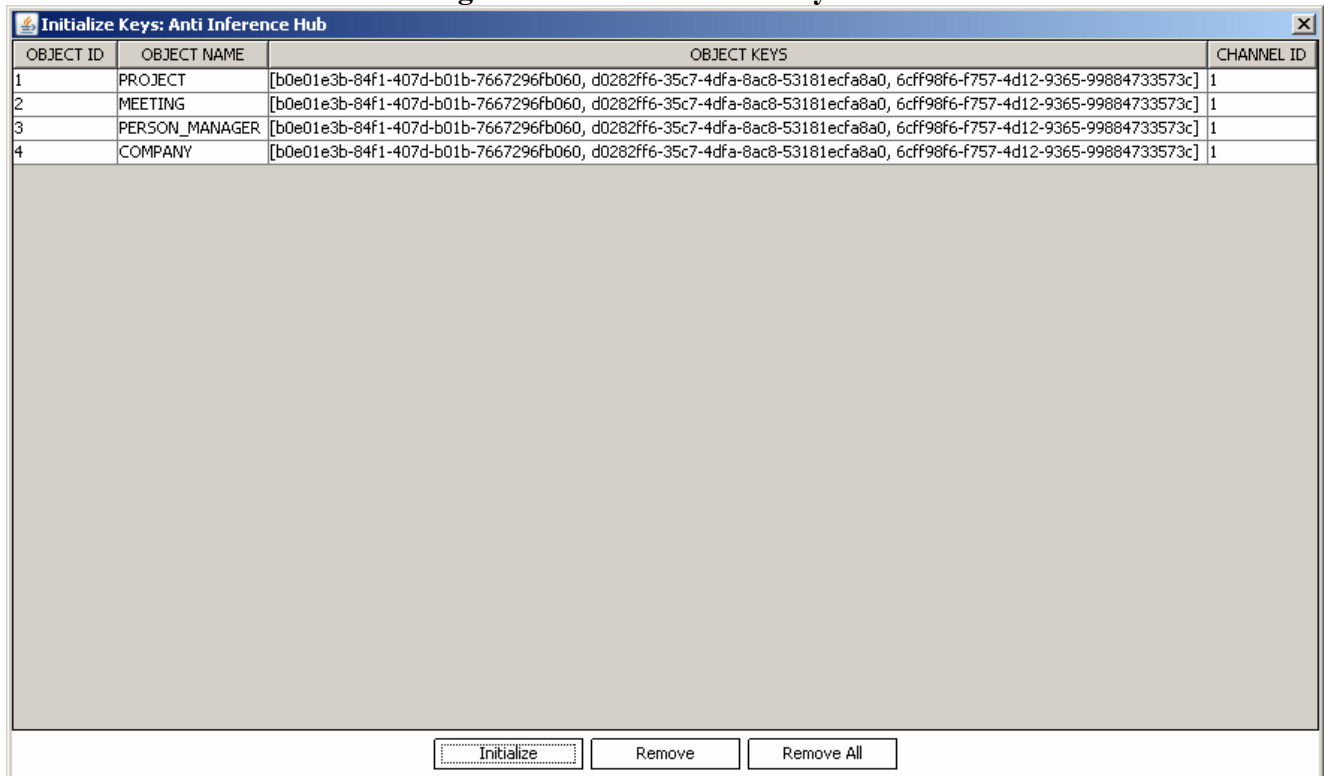


Table 6.1. Components in the Initialize Keys Box

Component	Description
Keys Table	This table lists all objects in all inference channels along with their keys. Information in this table is characterized by object ID, object name, object keys, and inference channel ID the object belongs to.
Initialize Button	This button is used to initialize keys for all objects in all inference channels in the database.
Remove Button	This button is used to remove the selected association between an object and its keys in the Keys Table. Note though that objects are not removed from inference channels.
Remove All Button	This button is used to remove all associations between objects and their keys in the Keys

	Table whether selected or not. Note again that objects are not removed from inference channels.
--	---

## Chapter 7. Adjusting Super Clients

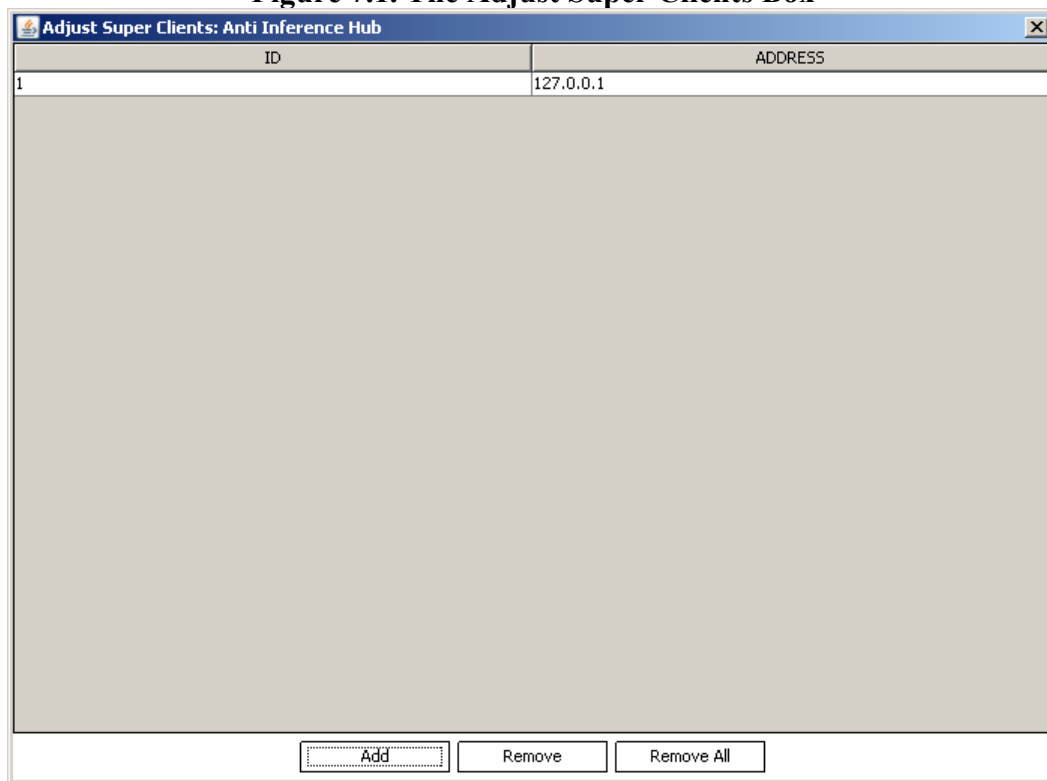
### 7.1. Introduction

Adjusting super clients is an optional step that you may consider before launching the Hub. A super client is a Hub Client that is allowed to infer, or in other words, a super client is a client that is considered database-friendly by the Hub! There is a very sound rationale behind allowing super clients in Anti Inference Hub. Read more about this by referring to Chapter 9. Technology Behind Anti Inference Hub.

### 7.2. The Adjust Super Clients Box

The Adjust Super Clients Box is accessible through the Adjust Super Clients menu item in the Hub Window, or the accelerator Ctrl+A.

**Figure 7.1. The Adjust Super Clients Box**



**Table 7.1. Components in the Adjust Super Clients Box**

Component	Description
Super Clients Table	This table lists all super clients. Every super client is characterized by its ID, and the textual representation of its IP address.
Add Button	This button is used to adjust a new super client. When clicked, it brings up the Add Super Client Box.
Remove Button	This button is used to remove the selected super client from the Super Clients Table.
Remove All Button	This button is used to remove all super clients in the Super Clients Table whether selected or not.

### 7.3. The Add Super Client Box

The Add Super Client Box is used to adjust a new super client.

**Figure 7.2. The Add Super Client Box**



**Table 7.2. Components in the Add Super Client Box**

Component	Description
Super Client Address Field	This field is used to specify the textual representation of the IP address of the new super client. This field has an installed IPv4 address filter, therefore, any attempt by the user to fill it with an invalid IP address will cause the field to revert to its default value which is 127.0.0.1.
OK Button	This button is used to save the new super client. It also closes the Add Super Client Box.

# Chapter 8. The Inference Problem: Demystified

## 8.1. Introduction

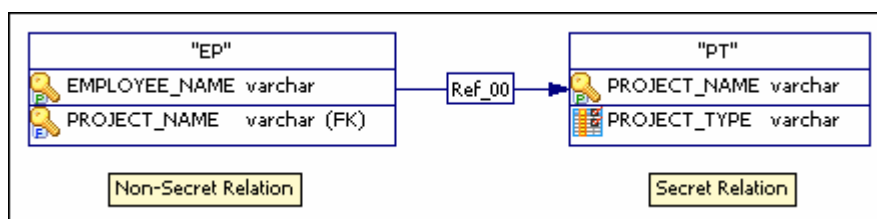
An inference channel in a database is a means by which one can infer data classified at a high level from data classified at a low level. The inference problem is the problem of detecting and removing inference channels. It is clear that inference problems are of vital interest to the designers and users of secure databases. Yet so far inference problems in multilevel databases have not been studied very deeply. This is partly due to the difficulty of the problem. We have no way of controlling what data is learned outside of the database, and our abilities to predict it will be limited. Thus even the best model can give us only an approximate idea of how safe a database is from illegal inferences. This fact should always be kept in mind when dealing with inference problems.

## 8.2. Specific Inference Problems

In this section we describe a number of inference channels that have been discovered in the course of database security research.

### 8.2.1. Inference from Queries Based on Sensitive Data

Suppose we have the following schema in a database:



And suppose a low user makes the following SQL query:

```
SELECT EP.EMPLOYEE-NAME
FROM EP, PT
WHERE EP.PROJECT_NAME = PT.PROJECT_NAME;
```

This query is evaluated by taking the natural join of the two relations EP and PT along PROJECT\_NAME, and then projecting along EMPLOYEE\_NAME, therefore, we have an inference channel, even though only the non-secret data (employee names) is being returned to the low user. The following SQL query represents a similar problem:

```
SELECT EP.EMPLOYEE_NAME
FROM EP, PT
WHERE EP.PROJECT_NAME = PT.PROJECT_NAME
AND PT.PROJECT-TYPE = 'SDI';
```

If we examine the above SQL queries carefully, we quickly observe that even though the data returned to the user has a low classification, the data that is required to evaluate the query has a higher classification. Thus, the inference channels arise from the fact that these queries are conditioned on data that are supposed to be invisible to the user.

Inferences of this type are easy to eliminate. The system can either modify the user query such that the query involves only the authorized data or simply abort the query.

### 8.2.2. Statistical Databases

The problem of statistical database security is the problem of answering queries about statistics on data, such as mean, median, standard deviation, and so on, without releasing the data itself. The threat against statistical database security is that an attacker may be able to find out statistical information

about individuals by posing queries on aggregate statistics over a period of time and performing arithmetic operations on the answers received, using his own information about the size and nature of the sets of individuals involved. For example, suppose the following sample database:

Name	Sex	Race	Aid	Fines	Drugs	Dorm
Adams	M	C	5000	45.	1	Holmes
Bailey	M	B	0	0.	0	Grey
Chin	F	A	3000	20.	0	West
Dewitt	M	B	1000	35.	3	Grey
Earhart	F	C	2000	95.	1	Holmes
Fein	F	C	1000	15.	0	West
Groff	M	C	4000	0.	3	West
Hill	F	B	5000	10.	2	Holmes
Koch	F	C	0	0.	1	West
Liu	F	A	0	10.	2	Grey
Majors	M	C	2000	0.	2	Grey

It might seem safe to report student aid total by sex and dorm. Such a report is shown in the following table:

	Holmes	Grey	West	Total
M	5000	3000	4000	12000
F	7000	0	4000	11000
Total	12000	3000	8000	23000

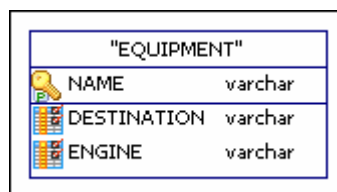
This seemingly innocent report reveals that no female living in Grey is receiving financial aid. Thus, we can infer that any female living in Grey (such as Liu) is certainly not receiving financial aid.

### 8.2.3. Inference from Data Combined with Metadata

We will consider some kinds of metadata used in databases, and show how it can be used to assist in making inferences.

#### 8.2.3.1. Key Integrity

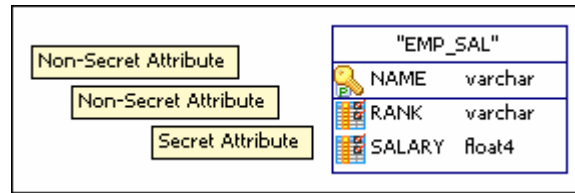
Suppose we have the following schema in a database:



And the following tuple in the relation (Wombat, Persian Gulf, Nuclear). Suppose a low user wants to insert the tuple (Wombat, Norfolk, Nuclear). If we choose to preserve key integrity, we must either delete the secret tuple or reject this insertion. We have an integrity problem if we delete the secret tuple (since it can cause data inserted by a high user to be deleted by a low user). If we reject the insertion, then the low user can derive an inference. It turns out that this problem can be eliminated using Polyinstantiation, in which case both tuples are allowed to exist.

### 8.2.3.2. Functional and Multivalued Dependencies

Suppose we have the following schema in a database:



Suppose every employee is aware of the constraint that all employees having identical ranks have the same salaries. Given this scenario, an employee who is not permitted to have access to secret data can easily determine employee salaries, which are secret. If we examine the above example carefully, we see that it contains an inference channel because the functional dependency  $RANK \rightarrow SALARY$  is not properly reflected in the classification levels of attributes RANK and SALARY. The way to avoid the problem in cases such as this is to raise the classification of the attribute RANK from non-secret to secret.

### 8.2.3.3. Value Constraints

Suppose that an attribute A is non-secret while attribute B is secret. Suppose the database enforces the constraint  $A + B \leq 20$ , which is made available to low users. The value of B does not affect the value of A directly, but it does determine the set of possible values A can take. Thus we have an inference channel. The usual solution to this problem is to allow such constraints to be defined only over a single security level. If a constraint is defined over several levels, then it is necessary to partition it into several single-level constraints. Thus, for example, the constraint  $A + B \leq 20$  can be partitioned into  $A \leq 10$  and  $B \leq 10$ .

### 8.2.3.4. Classification Constraints

Suppose the following integrity constraints apply to a database containing facts about ancient Iraqis: Every ancient Iraqi is a Sumerian, Elamite, Assyrian, or a Babylonian.

All Sumerian and Assyrian are peaceable.

All Elamite and Babylonian are violent.

Saddam does not wish it to be known that he is peaceable! Bush tries to find out about Saddam from a system that refuses to answer whenever the answer, together with the integrity constraints, would imply the secret:

Bush: Is Saddam a Sumerian?

System: I will not tell you.

Bush: Is he an Elamite?

System: No.

Bush: Is he an Assyrian?

System: No.

Bush: Is he a Babylonian, then?

System: I will not tell you.

Bush, knowing that it is his disposition that Saddam is trying to conceal, notes that if he were not a Sumerian, then the system could have answered "No" to the first question without revealing the secret. Thus, Saddam must be a peaceable Sumerian.

## 8.3. General Characterizations of the Inference Problem

Probably the earliest formal characterization of the inference problem in databases is that of Goguen and Meseguer [GOGU84]. Consider a database in which each data item is given an access class, and suppose that the set of access classes is partially ordered. Define the relation  $\rightarrow$  as follows: Given data



items  $x$  and  $y$ , we say  $x \rightarrow y$  if it is possible to infer  $y$  from  $x$ . The relation  $\rightarrow$  is reflexive and transitive. A set  $S$  is said to be inferentially closed if whenever  $x$  is in  $S$  and  $x \rightarrow y$  holds, then  $y$  belongs to  $S$  as well. Now, for an access class  $L$ , let  $E(L)$  denote the set consisting of all possible responses that are classified at access class less than or equal to  $L$ . There is an inference channel if  $E(L)$  is not inferentially closed.

In a refinement of Goguen and Meseguer's definition, Denning and Morgenstern [DENN86] derive the inference relation from classical information theory. Given two data items  $x$  and  $y$ , let  $H(y)$  denote the uncertainty of  $y$ , and let  $H_x(y)$  denote the uncertainty of  $y$  given  $x$  (where uncertainty is defined in the usual information-theoretic way). Then, the reduction in uncertainty of  $y$  given  $x$  is defined as follows:

$$INFER(x \rightarrow y) = \frac{H(y) - H_x(y)}{H(y)}$$

The value of  $INFER(x \rightarrow y)$  is between 0 and 1. If the value is 0, then it is impossible to infer any information about  $y$  from  $x$ . If the value is between 0 and 1, then  $y$  becomes somewhat more likely given  $x$ . If the value is 1, then  $y$  can be inferred given  $x$ . Denning and Morgenstern pointed out serious drawbacks of their definition:

1. In most cases it is difficult, if not impossible, to determine the value of  $H_x(y)$ .
2. The definition does not take into account the computational complexity that is required to draw the inference.

## 8.4. Tools and Techniques for Dealing with Inference Channels

In this section, we discuss the various techniques and tools that have been proposed for locating inference channels and preventing their exploitation once they have been found. Basically two kinds of techniques have been proposed for locating and eliminating inference channels. One is to use semantic data modeling techniques to detect inference channels in the database design, and then to redesign the database so that these channels no longer exist. The other is to evaluate database transactions (involving either reads or updates, or both) to determine whether they lead to illegal inferences. If they do, the query is either disallowed or reclassified at the higher level.

For the first technique, we refer the reader to [HINK88a], [SMIT90], [SMIT90a], and [LUNT89a]. And for the second technique, we refer the reader to [MAZU88], [STAC90], [THUR87], [STON74], and [HAIG90]. When you consider these research works, you soon realize that more work needs to be done to understand the relative advantages and disadvantages of the two techniques.

## 8.5. References

- [DENN86] Denning, Dorothy E., and Matthew Morgenstern, "Military Database Technology Study: AI Techniques for Security and Reliability," SRI tech. report, Aug. 1986.
- [DENN86a] Denning, Dorothy E., "A Preliminary Note on the Inference Problem in Multilevel Database Management Systems," Proc. Nat'l Computer Security Center Invitational Workshop on Database Security, June 1986.
- [GOGU84] Goguen, Joseph A., and José Meseguer, "Unwinding and Inference Control," Proc. Symp. Security and Privacy, Apr. 1984, pp. 75-86.
- [HAIG90] Haigh, J.T., R.C. O'Brien, P.D. Stachour, and D.L. Toups, "The LDV Approach to Database Security," in Database Security III: Status and Prospects, D.I. Spooner and C. Landwehr, eds., North-Holland, Amsterdam, 1990.
- [HINK88a] Hinke, Thomas H., "Inference Aggregation Detection in Database Management Systems," Proc. IEEE Symp. Research in Security and Privacy, Apr. 1988, pp. 96-106.
- [LUNT89a] Lunt, T.F., "Aggregation and Inference: Facts and Fallacies," Proc. IEEE Symp. Research in Security and Privacy, May 1989, pp. 102-109.

- [MAZU88] Mazumdar, S., D. Stemple, and T. Sheard, "Resolving the Tension between Integrity and Security Using a Theorem Prover," Proc. ACM Int'l Conf. Management of Data, ACM, New York, 1988, pp. 233-242.
- [MEAD88a] Meadows, Catherine, and Sushil Jajodia, "Integrity versus Security in Multi-Level Secure Databases," in Database Security: Status and Prospects, C. Landwehr, ed., North-Holland, Amsterdam, 1988, pp. 89-101.
- [SICH83] Sicherman, G.L., W. de Jonge, and R.P. van de Riet, "Answering Queries without Revealing Secrets," ACM Trans. Database Systems, Mar. 1983, Vol. 8, No. 1, pp. 41-59.
- [SMIT90] Smith, Gary W., "Modeling Security-Relevant Data Semantics," Proc. IEEE Symp. Research in Security and Privacy, May 1990, pp. 384-391.
- [SMIT90a] Smith, Gary W., The Modeling and Representation of Security Semantics for Database Applications, doctoral dissertation, George Mason Univ., Fairfax, Va., 1990.
- [STAC90] Stachour, Paul D., and Bhavani Thuraisingham, "Design of LDV: A Multilevel Secure Relational Database Management System," IEEE Trans. Knowledge and Data Eng., Vol. 2, No. 2, June 1990, pp. 190-209.
- [STON74] Stonebraker, M., "Implementation of Integrity Constraints and Views by Query Modification," ACM Nat'l Conf. Proc., 1974, pp. 180-186.
- [SU86] Su, Tzong-An, Inferences in Databases, doctoral dissertation, Case Western Reserve Univ., Cleveland, Ohio, 1986.
- [SU87] Su, Tzong-An, and Gultekin Ozsoyoglu, "Data Dependencies and Inference Control in Multilevel Relational Database Systems," Proc. Symp. Security and Privacy, Apr. 1987, pp. 202-211.
- [SU90] Su, Tzong-An, and Gultekin Ozsoyoglu, "Multivalued Dependency Inferences in Multilevel Relational Database Systems," Database Security III: Status and Prospects, D.L. Spooner and C. Landwehr, eds., North-Holland, Amsterdam, 1990, pp. 293-300.
- [THUR87] Thuraisingham, Bhavani M., "Security Checking in Relational Database Management Systems Augmented with Inference Engines," Computers and Security, Vol. 6, 1987, pp. 479-492.
- Sushil Jajodia, and Catherine Meadows, "Inference Problems in Multilevel Secure Database Management Systems," in "Information Security: An Integrated Collection of Essays," IEEE Computer Society Press, Los Alamitos, CA USA, 1995.
- Charles P. Pfleeger, and Shari Lawrence Pfleeger, "Security in Computing, Fourth Edition," Prentice Hall, October 13, 2006.

## **Chapter 9. Technology behind Anti Inference Hub**

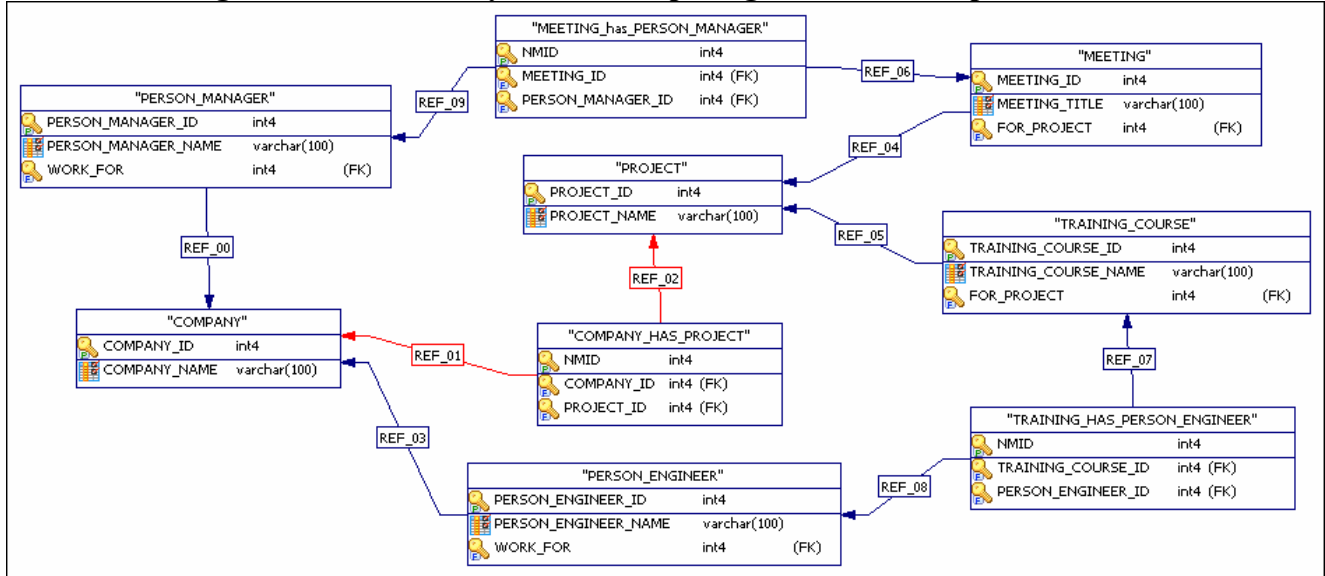
Anti Inference Hub is based on a dynamic query processing technology described in the following paper: X. Chen, R. Wei, "A Dynamic Method for Handling the Inference Problem in Multilevel Secure Databases," Proc. International Conference on Information Technology: Coding and Computing (ITCC'05) - Volume I - Volume 01, 2005, pp. 751-756.

# Chapter 10. The Sample Database

## 10.1. Introduction

Anti Inference Hub comes with a sample database that may be used for testing and understanding purposes.

**Figure 10.1. The Entity-Relationship Diagram of the Sample Database**



## 10.2. Location of the Sample Database

The sample database is included in the directory database in Anti Inference Hub binary package.

**Table 10.1. Content of the database Directory**

File	Description
inferenceoracle.sql	This file contains SQL script needed to create the sample database with an Oracle DBMS.
inferencepostgresqlmysql.sql	This file contains SQL script needed to create the sample database with a PostgreSQL DBMS, or a MySQL DBMS.
createoracledb.bat	This file executes SQL script in the file inferenceoracle.sql in accordance with an Oracle DBMS running under Windows.
createoracledb.sh	This file executes SQL script in the file inferenceoracle.sql in accordance with an Oracle DBMS running under Linux.
createpostgresqldb.bat	This file executes SQL script in the file inferencepostgresqlmysql.sql in accordance with a PostgreSQL DBMS running under Windows.
createpostgresqldb.sh	This file executes SQL script in the file inferencepostgresqlmysql.sql in accordance with a PostgreSQL DBMS running under Linux.
createmysqldb.bat	This file executes SQL script in the file inferencepostgresqlmysql.sql in accordance with a MySQL DBMS running under Windows.

createmySQLdb.sh	This file executes SQL script in the file inferencepostgresqlmysql.sql in accordance with a MySQL DBMS running under Linux.
entity-relationship diagram.pdd	This file is the entity-relationship diagram of the sample database built using MicroOLAP Database Designer for PostgreSQL.
entity-relationship diagram.bmp	This file is the entity-relationship diagram of the sample database exported as a bitmap image.

### 10.3. Creating the Sample Database

To create the sample database, first install a DBMS of choice under an operating system of choice, then navigate to the database directory, and execute the file corresponding to your platform.

### 10.4. Our Test Environment

The sample database was successfully tested with the following:

- Oracle Database 10g Release 2 (10.2.0.1.0) under Windows XP Service Pack 2 and under Linux Ubuntu.
- PostgreSQL 8.4.2-1 under Windows XP Service Pack 2 and under Linux Ubuntu.
- MySQL 5.1.42 under Windows XP Service Pack 2 and under Linux Ubuntu.

# Chapter 11. Anti Inference Hub in Action

## 11.1. Introduction

In this chapter, we put Anti Inference Hub in action using the sample database created in Chapter 10.

## 11.2. The Inference Attempt to Address

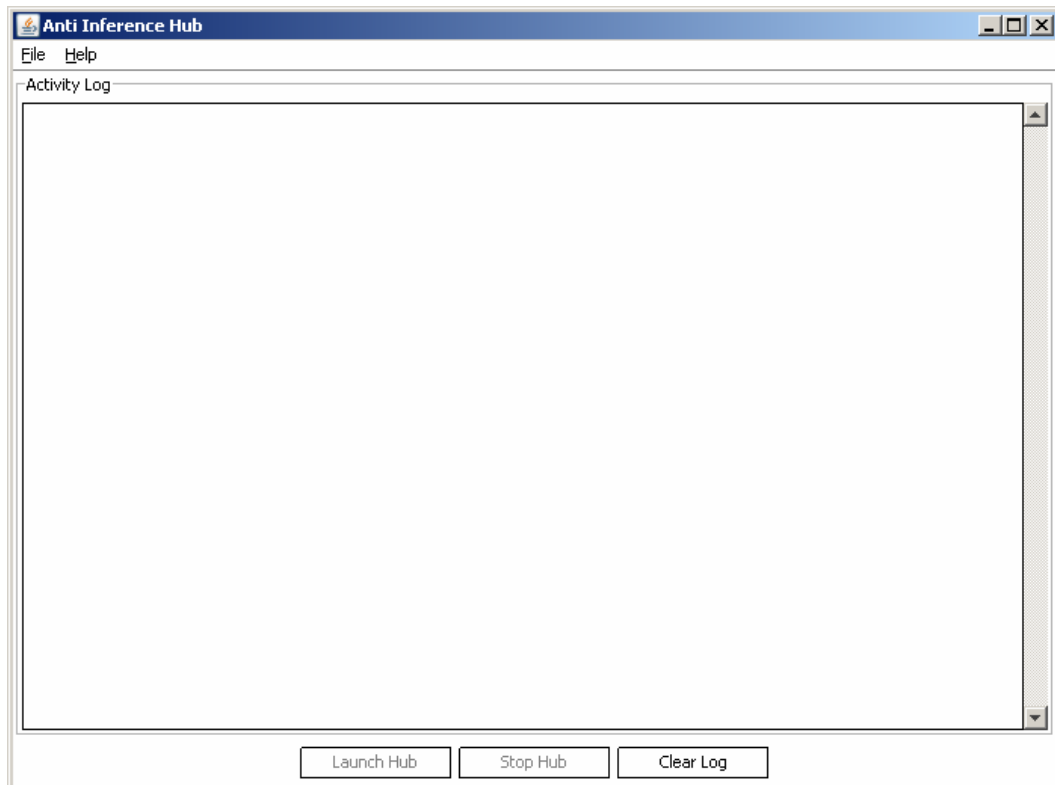
In the sample database, suppose that a low user is able to know the following by executing queries against the database:

- For which COMPANY a PERSON\_MANAGER works.
- PERSON\_MANAGER attending a MEETING.
- MEETING on a PROJECT.

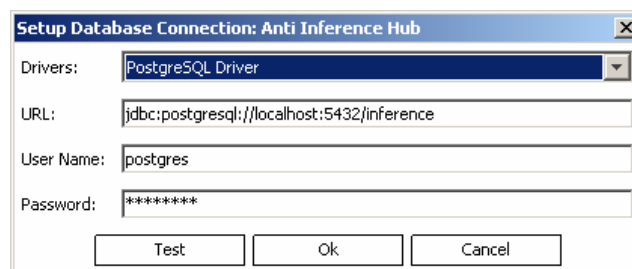
If that was true, then the low user can immediately infer the COMPANY supporting the PROJECT. In the following section, we show how Anti Inference Hub can be used to thwart this inference attempt.

## 11.3. Thwarting the Inference Attempt

Start the Hub as described in Section 3.2. Starting Anti Inference Hub.



Press Ctrl+D, and establish the connection with the DBMS you are using.



Press Ctrl+I, then press the button Add in the box that shows to setup an inference channel named "Channel 1", of length 4, containing the following objects: PROJECT, MEETING, PERSON\_MANAGER, and COMPANY.

Add Channel

Inference Channel Name:

Channel 1

Database Objects:

COMPANY

Inference Channel Objects:

PROJECT

MEETING

PERSON\_MANAGER

Include

Exclude

Clear

OK

Close the Setup Inference Channels Box, and press Ctrl+K, then press the button Initialize in the box that shows to initialize "Channel 1" objects keys.

Initialize Keys: Anti Inference Hub

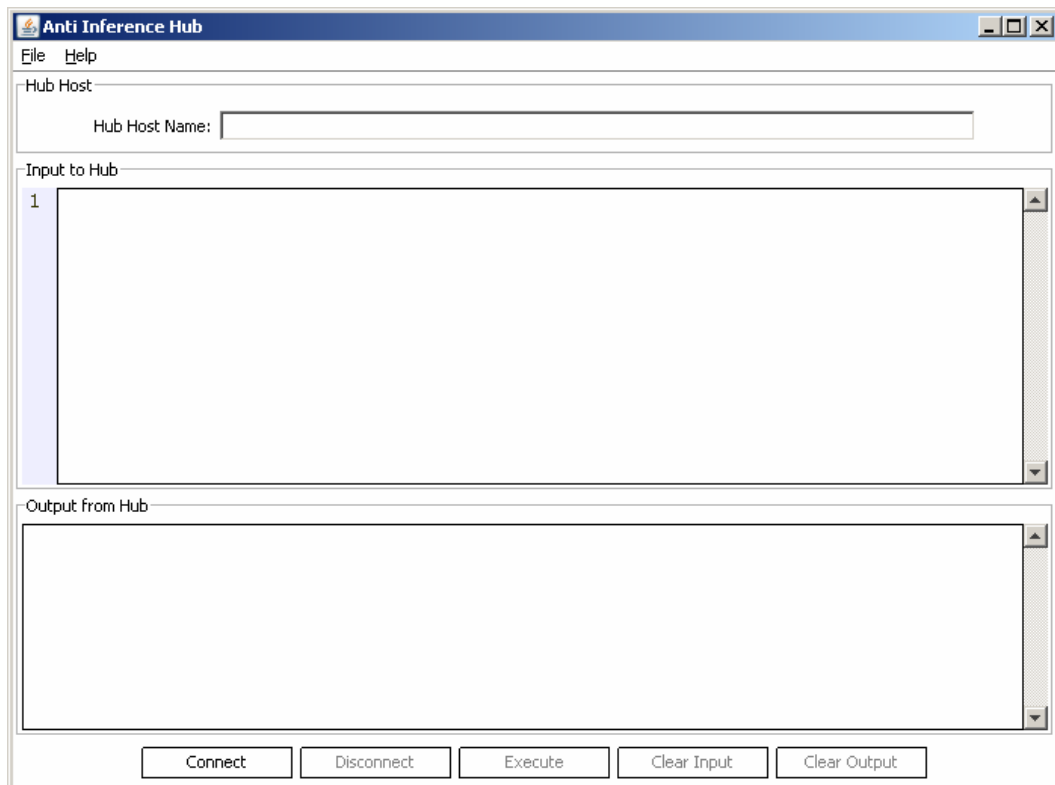
OBJECT ID	OBJECT NAME	OBJECT KEYS	CHANNEL ID
1	PROJECT	[341b70af-8ade-4dcb-8156-58341938c17f, 5f771833-e24e-486e-b0a6-4da0a1bdd628, 0bc0196c-913f-4aef-9877-aaef37f1cb18]	1
2	MEETING	[341b70af-8ade-4dcb-8156-58341938c17f, 5f771833-e24e-486e-b0a6-4da0a1bdd628, 0bc0196c-913f-4aef-9877-aaef37f1cb18]	1
3	PERSON_MANAGER	[341b70af-8ade-4dcb-8156-58341938c17f, 5f771833-e24e-486e-b0a6-4da0a1bdd628, 0bc0196c-913f-4aef-9877-aaef37f1cb18]	1
4	COMPANY	[341b70af-8ade-4dcb-8156-58341938c17f, 5f771833-e24e-486e-b0a6-4da0a1bdd628, 0bc0196c-913f-4aef-9877-aaef37f1cb18]	1

Initialize

Remove

Remove All

Close the Initialize Keys Box, then press the button "Launch Hub" to launch the Hub. Now, start the Hub Client as described in Section 3.2. Starting Anti Inference Hub.



Enter the appropriate Hub Host Name in its field, and connect to the Hub by pressing the Connect button. Act as a low user, and execute the following query in the Hub Client:

```
SELECT PERSON_MANAGER_NAME, COMPANY_NAME
FROM PERSON_MANAGER, COMPANY
WHERE PERSON_MANAGER.WORK_FOR = COMPANY.COMPANY_ID
AND PERSON_MANAGER.PERSON_MANAGER_NAME = 'MANAGER 1';
```

In this query, you are trying to know for which COMPANY a PERSON\_MANAGER works. You should receive the following XML output from the Hub:

```
<?xml version="1.0" encoding="UTF-8"?>
<Results>
  <Row>
    <person_manager_name>MANAGER 1</person_manager_name>
    <company_name>COMPANY 1</company_name>
  </Row>
</Results>
```

Which tells you that "MANAGER 1" works for "COMPANY 1". Note the change to "Channel 1" objects keys by opening the Initialize Keys Box in the Hub.



Initialize Keys: Anti Inference Hub			
OBJECT ID	OBJECT NAME	OBJECT KEYS	CHANNEL ID
1	PROJECT	[0bc0196c-913f-4aef-9877-aaef37f1cb18]	1
2	MEETING	[0bc0196c-913f-4aef-9877-aaef37f1cb18]	1
3	PERSON_MANAGER	[341b70af-8ade-4dc8-8156-58341938c17f]	1
4	COMPANY	[5f771833-e24e-486e-b0a6-4da0a1bdd628]	1

Now, execute the following query in the Hub Client:

```
SELECT MEETING_TITLE, PERSON_MANAGER_NAME
FROM MEETING, MEETING_HAS_PERSON_MANAGER, PERSON_MANAGER
WHERE MEETING.MEETING_ID = MEETING_HAS_PERSON_MANAGER.MEETING_ID
AND PERSON_MANAGER.PERSON_MANAGER_ID =
MEETING_HAS_PERSON_MANAGER.PERSON_MANAGER_ID
AND MEETING.MEETING_TITLE = 'MEETING 1';
```

In this query, you are trying to know the PERSON\_MANAGER attending a MEETING. You should receive the following XML output from the Hub:

```
<?xml version="1.0" encoding="UTF-8"?>
<Results>
  <Row>
    <meeting_title>MEETING 1</meeting_title>
    <person_manager_name>MANAGER 1</person_manager_name>
  </Row>
</Results>
```

Which tells you that "MANAGER 1" attends "MEETING 1". Note the change to "Channel 1" objects keys by opening the Initialize Keys Box in the Hub.

Initialize Keys: Anti Inference Hub			
OBJECT ID	OBJECT NAME	OBJECT KEYS	CHANNEL ID
1	PROJECT		1
2	MEETING	[0bc0196c-913f-4aef-9877-aaef37f1cb18]	1
3	PERSON_MANAGER	[341b70af-8ade-4dcb-8156-58341938c17f]	1
4	COMPANY	[5f771833-e24e-486e-b0a6-4da0a1bdd628]	1

Note that PROJECT object is now a reserved object in "Channel 1" because it has an empty key set. Now, execute the following query in the Hub Client:

```
SELECT MEETING_TITLE, PROJECT_NAME
FROM MEETING, PROJECT
WHERE MEETING.FOR_PROJECT = PROJECT.PROJECT_ID
AND PROJECT.PROJECT_NAME = 'PROJECT 1';
```

In this query, you are trying to know the MEETING on a PROJECT. If you received a response to this query (actually the response will tell you that "MEETING 1" is on project "PROJECT 1"), then you can immediately infer that "COMPANY 1" is supporting "PROJECT 1"; an inference you base on the results for queries you executed so far. Therefore, result for this query should be blocked. Indeed, it is! The output from the Hub will be:

```
INFERENCE ATTEMPT: Access denied to reserved object PROJECT.
```

Why don't you adjust yourself as a super client and re-execute the same query! Press Ctrl+A, then press the button Add in the box that shows to adjust yourself as a super client, then re-execute the same query. You should receive the following XML output from the Hub:

```
INFERENCE ATTEMPT FROM A SUPER CLIENT: Access permitted to reserved object PROJECT.
```

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<Results>
```

```
<Row>
```

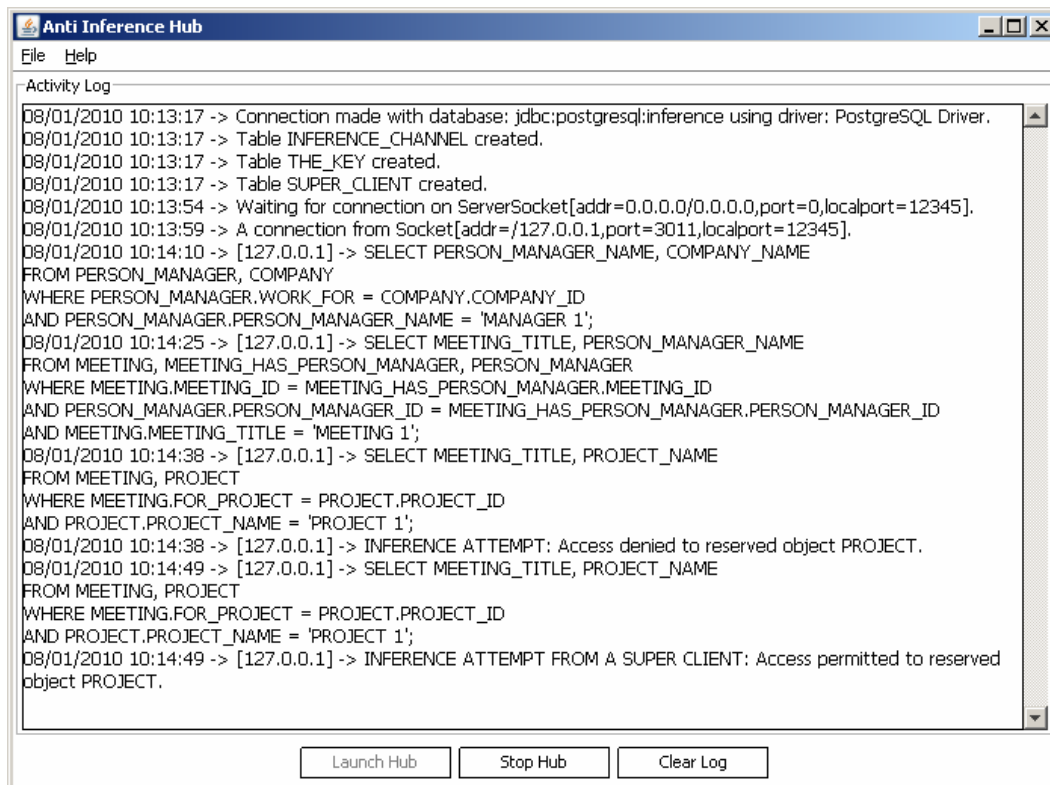
```
<meeting_title>MEETING 1</meeting_title>
```

```
<project_name>PROJECT 1</project_name>
```

```
</Row>
```

```
</Results>
```

Thus, as a super client, the Hub allows you to infer. Please note that Anti Inference Hub does not take any IP spoofing attacks into consideration. Securing a network against such attacks falls beyond the purpose of Anti Inference Hub. You may take a look at the Hub log which should be as follow:



When done, disconnect from the Hub, and stop it.

# Appendix A. Anti Inference Hub License

The GNU General Public License (GPL) explains all the things that you are allowed to do with Anti Inference Hub code and documentation.

GNU GENERAL PUBLIC LICENSE  
Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <<http://fsf.org/>>  
Everyone is permitted to copy and distribute verbatim copies  
of this license document, but changing it is not allowed.

## Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

## TERMS AND CONDITIONS

### 0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

### 1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically

linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

## 2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

## 3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

## 4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

## 5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
- b) The work must carry prominent notices stating that it is

released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

## 6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded

from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

## 7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal



Notices displayed by works containing it; or

c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or

d) Limiting the use for publicity purposes of names of licensors or authors of the material; or

e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or

f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

## 8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

## 9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

## 10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

## 11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

#### 12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

#### 13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

#### 14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

#### 15. Disclaimer of warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO,

THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

#### 16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

#### 17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

### END OF TERMS AND CONDITIONS

#### How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>
```

```
This program is free software: you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation, either version 3 of the License, or
(at your option) any later version.
```

```
This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.
```

```
You should have received a copy of the GNU General Public License
along with this program. If not, see <http://www.gnu.org/licenses/>.
```

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

```
<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <<http://www.gnu.org/licenses/>>.

The GNU General Public License does not permit incorporating your program

into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.

## **Appendix B. Anti Inference Hub Code Disclaimer**

The author of Anti Inference Hub code has used his best efforts in preparing the code. These efforts include the development, research, testing, and optimization of the theories and programs to determine their effectiveness.

Anti Inference Hub code is not designed or intended for use in the design, construction, operation or maintenance of any nuclear facility. Author disclaims any express or implied warranty of fitness for such uses.

The author makes no warranty of any kind, expressed or implied, with regard to Anti Inference Hub code or to the documentation accompanying it. In no event shall the author be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption whatsoever) arising out of, the furnishing, performance, or use of Anti Inference Hub code, even if advised of the possibilities of such damages.