Pannasoft Ingenuity User Guide

Revision 1.10

For Pannasoft Ingenuity Version 1.0.2.x

September 2005

TABLE OF CONTENTS

SECTION 1: INTRODUCTION

Artificial neural networks (ANNs) are mathematical models of the human nervous system, with algorithms consisting of weighted interconnecting processing units. ANNs have been proven as a suitable artificial intelligence technology for pattern recognition with superior performance. Pattern recognition offers benefits of solving practical problems in the business, medical, engineering, as well as a wide range of application areas.

Proper data preprocessing, architecture selection and network training are required to reap the best from ANNs. The arduous task to optimize ANNs is alleviated with **Pannasoft Ingenuity**, the ANN engine that is simplified for use.

Pannasoft Ingenuity's component engine is based on the Autonomous Adaptive Algorithm™ (AAA), a hybrid ANN technology. AAA synergistically uses state-of-the-art machine learning algorithms encompassing neural computation and fuzzy computation to form a hybrid model.

Pannasoft Ingenuity comprises methods and algorithms specially configured and evaluated by Pannasoft's experts for optimized application based on ANN solutions. **Pannasoft Ingenuity** provides the best-of-breed ANN technology to be infused into real world software products to achieve optimal solutions of specific problems, yielding numerous returns in terms of best results, enhanced value, and competitive edge.

How This User Guide Is Organized

This user guide covers the steps required to get Pannasoft Ingenuity up and running inclusive of examples on how to complete the data normalization process, with description based on Microsoft SQL Server 2000 database. However, the user can use this system with other databases and change the database specific functions as specified in this document. The organization of this document is as follows.

Section 1 briefly introduces Pannasoft Ingenuity.

Section 2 highlights the features of Pannasoft Ingenuity.

Section 3 enlists and compares features available in various editions of Pannasoft Ingenuity.

Section 4 briefly introduces how to prepare and normalize the data before running Pannasoft Ingenuity.

Section 5 provides full details on installation and usage of the Training and Prediction modules. Configuration of the system is described in detail here.

SECTION 2: FEATURES

Automated / Manual Parameters Selection

Pannasoft Ingenuity gives users the flexibility to automatically or manually adjust the network parameters for analyzing data. In the auto mode, the efficient learning algorithm will automatically adjust network parameters to find a suitable architecture. In the manual mode, the user is able to manually fine-tune the parameters.

Rule Extraction and Pruning Strategy

Pannasoft Ingenuity provides a rule explanation capability to gain a wider degree of user acceptance on the results and to let the user understand the potential and ability of the trained engine in handling classification problems. The pruning strategy enables user to remove low confidence and unimportant prototypes (knowledge) from the system to reduce complexity and network size.

Retrieve Old Knowledge

Pannasoft Ingenuity enables user to retrieve old prototypes from the database and combine them with new input samples to form new prototypes. Thus, time needed to retrain is significantly reduced.

Multiple Classifier System

Multiple Classifier Systems enable Pannasoft Ingenuity to produce better results and higher accuracy in decision making using a voting mechanism.

Multiple Jobs

Pannasoft Ingenuity can accommodate different category of data sets (e.g. data on credit card spending, health diagnosis, manufacturing operation and etc) in one database.

Advantages of Pannasoft Ingenuity:

1. Ability to self-organize to an arbitrary sequence of sample patterns in stationary as well as non-stationary environments.
2. Ability to continuously learn new patterns without corrupting previously learned information.
3. Ability to reveal the embedded rule set in the network in linguistic format that users can clearly comprehend.
4. Ability to enhance the performance in decision making by using Multiple Classifier Systems

Pannasoft Ingenuity overcomes the limitation of Artificial Neural Networks

Neural networks have been confirmed to be useful in many applications. Nonetheless, most neural network models refuse to tell what they have learned due to their distributed knowledge representation. Neural networks are regarded as a "black box" technology because of the lack of comprehensibility. In general, comprehensibility is one of the required characteristics of reliable systems. Therefore many works have been carried out to address the issue of improving the comprehensibility of neural networks. Rule extraction is a technique to overcome this problem. The merits of including rule extraction techniques as a supplement to neural networks are as follows:

1. Provide an explanation component to neural networks
2. Overcome the problem of knowledge acquisition
3. Explore data and induce scientific theories
4. Improve the generalization of neural networks solutions

Pannasoft Ingenuity with AAA is a hybrid neural network-based system, integrated with a rule extraction technique that shows promising characteristic of building a genuinely autonomous system. The revelation of embedded rule set in Pannasoft Ingenuity can be accomplished in the IF-THEN format, which is straightforward and comprehensible.

SECTION 3: EDITIONS

The following table lists the various features that are available for different editions of Pannasoft Ingenuity.

Feature	Pannasoft Ingenuity Edition			
	Demo	Express	Standard	Professional
Number of Features	Per demo data	10	25	Unlimited
Multiple Classifier	✓	X	✓	✓
Retrieve Old Knowledge	✓	X	X	✓
Automated Parameters Selection	✓	✓	✓	✓
Rule Extraction	✓	X	X	✓
IP Address dependent	No			Yes
License Key	Pre-packaged	Email upon purchase		
Training data query	Pre-defined	Configurable		

Table 1: Different editions of Pannasoft Ingenuity

Note: We strongly recommend users who are using the Demo and Professional Edition to learn how to normalize the input data properly by using the normalization guide provided with Pannasoft Ingenuity. This is to ensure higher accuracy in the prediction results.

SECTION 4: DATA PREPARATION AND NORMALIZATION

Pannasoft Ingenuity combines fuzzy logic and neural network technologies to create learning and predicting capabilities. Therefore, data provided to Pannasoft Ingenuity must be a set of numbers (between 0 and 1). Here, the normalization process is performed with the database and Pannasoft Ingenuity will retrieve the normalized data from the database. The normalization process aims to convert all unprocessed data into fuzzy numbers which are later used by Pannasoft Ingenuity.

Example 1:

No	Feature 1	Feature 2
1	NULL	2.5
2	10	6.1
3	21	7.8
4	15	3.6
5	30	9.1

Table 2: Example of a numerical raw data

No	Feature 1	Feature 2
1	0	0.20
2	0.20	0.636364
3	0.64	0.842424
4	0.40	0.333333
5	1.00	1.00

Table 3: Normalized data from numerical raw data

Table 2 shows an example of a raw data that contains numerical value. NULL indicates that there is a missing feature in the first feature of first sample. Below equation is used to normalize the data in table 2:

$$f(x_i) = \left[\frac{x_i - \min(x)}{\max(x) - \min(x)} \right] \times (1 - 0.2) + 0.2 \qquad (1)$$

where x_i = i-th input of feature x
$\min(x)$ = minimum value in feature x, not included NULL
$\max(x)$ = maximum value in feature x

Table 3 shows the normalized data for the example in Table 2 using equation (1). Zero (0) is a reserved value to indicate NULL in the data.

Example 2:

No	Feature 1	Feature 2
1	NULL	cheese
2	car	bread
3	motorbike	milk
4	bicycle	milk
5	motorbike	Strawberry jam

Table 4: Example of a raw data that contains string

No	Feature 1	Feature 2
1	NULL	1
2	1	2
3	2	3
4	3	3
5	2	4

Table 5: A numerical representative for each string in Table 4

No	Feature 1	Feature 2
1	NULL	0.20
2	0.20	0.466667
3	0.60	0.733333
4	1.00	0.733333
5	0.60	1.00

Table 6: Normalized data for Table 5

Table 4 shows an example data which contains string. A middle step is needed in order to perform normalization. A set of numerical values have been used to represent each of the string/object in a feature as shown in Table 5. Using the equation (1), users are then able to convert the value into fuzzy number as shown in Table 6.

Note: Normalization is an important step to pre-process the input data to Pannasoft Ingenuity. **The "Normalization guide" document provided with Pannasoft Ingenuity shows an example of more detailed and automated approach to normalization. Users are recommended to read.**

SECTION 5: USING PANNASOFT INGENUITY

Pannasoft Ingenuity consists of 2 modules:

1. Training module
2. Prediction module

Pannasoft Ingenuity is available in Java Application Programming Interface (API). The target machine needs to have Java Development Kit (JDK)/Java Runtime Engine (JRE) version 1.4.2 and above. Pannasoft Ingenuity requires a RDBMS that is able to create database object View and Trigger for storing knowledge.

The following description explains some sample codes to invoke both modules mentioned earlier.

Pannasoft Ingenuity API consists of the following files:

1. com.pns.extract-x.x.x.x.jar
2. pannasoft.properties
3. pannasoft.lcf

There is 1 file for Apache Log4J implementation:

1. log4j-x.x.x.jar

There are 3 files for Microsoft SQL 2000 JDBC:

1. msbase.jar
2. mssqlserver.jar
3. msutil.jar

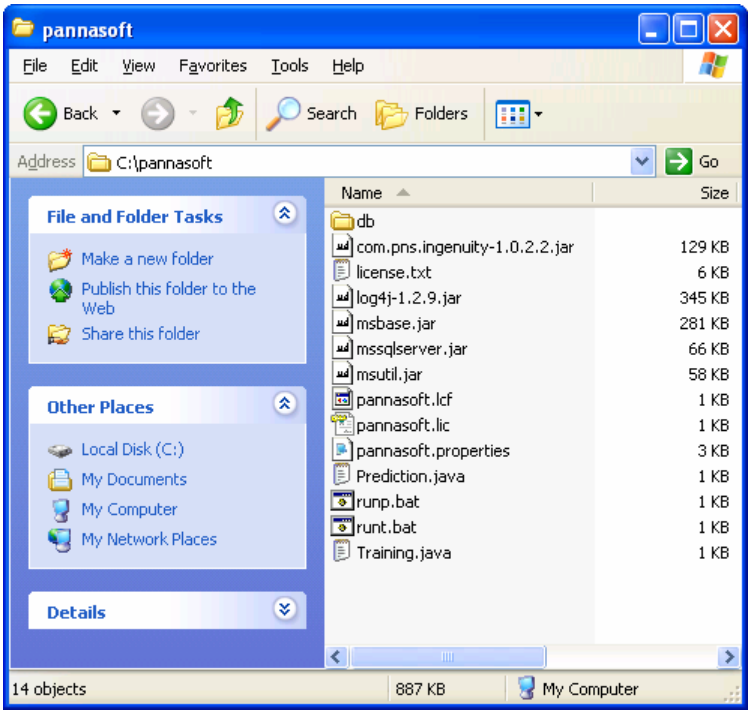
Sample Java application used to invoke Pannasoft Ingenuity:

1. To invoke Pannasoft Ingenuity's Training module: Training.java (binary file: Training.class)
2. To invoke Pannasoft Ingenuity's Prediction module: Prediction.java (binary file: Prediction.class)

Sample script used to compile and execute the sample Java application:

1. For Pannasoft Ingenuity's Training module: runt.bat
2. For Pannasoft Ingenuity's Prediction module: runp.bat

Extract all files into location C:\pannasoft as shown in the following page. To do a quick test, simply execute the batch files given to test both modules.



System Table

To create the required system tables, launch the given Pannasoft Ingenuity's database script and point it to your target database. If you are given database backup image[1], restore them according to your target database machine.

Training Module

After the normalization process, the user is ready to train Pannasoft Ingenuity to become an intelligent system by invoking the Training module. Pannasoft Ingenuity's Training module uses the normalized training data, i.e., View<Job_Code>NormTrainData, in the training process.

Pannasoft Ingenuity's training module checks all TRFLAG in <Job_Code>DATA table (example table GCDATA) before performing any training. For records that are needed for prediction, the user is required to set PRFLAG='Y' AND TRFLAG ='D' to avoid any training process on these records because prediction records contain no class (reference to column "class" in table "German") and Pannasoft Ingenuity's training module cannot process this type of records. Please refer to Appendix A for more detailed information on this "TRFLAG".

The user is able to configure network parameters to improve Pannasoft Ingenuity's accuracy in making prediction. The network parameters are discussed in the following section.

[1] Refer to Appendix B for steps of creating and restoring Microsoft SQL 2000 database.

To ensure Pannasoft Ingenuity's training module computes consistent knowledge, data manipulation (including the activities of inserting, updating, and deleting processes) in raw data table (example "German" table) and Pannasoft Ingenuity's reference table (example "GCDATA" table) are prohibited during the execution of the training module. Changes in both tables' data have a direct impact on the chances of all database View objects referred by Pannasoft Ingenuity's training module. Data inconsistencies will occur and Pannasoft Ingenuity's knowledge will be directly affected while readings on both tables are in progress.

Sample Code: Invoke Training module in Java

```
String szPropFile = "C:/pannasoft/pannasoft.properties";
com.pns.extract.Ingenuity myObj = com.pns.extract.Ingenuity.getInstance
(szPropFile);
myObj.doTraining ();
```

Note: Always remember to include the given jar file, i.e. file com.pns.extract-x.x.x.x.jar, into your classpath before compiling and executing Pannasoft Ingenuity modules.

In order to have trained record(s), Pannasoft Ingenuity's training module will search all records with TRFLAG='A', 'U' or 'T' in Pannasoft Ingenuity's reference table (example "CRSDATA" table) for prediction activities. Pannasoft Ingenuity always assumes that the records contain data in column "class", table "German".

After the training activities have ended for all the record(s) mentioned, TRFLAG will change to 'T' to indicate that they are trained by Pannasoft Ingenuity's training module.

Pannasoft Ingenuity always assumes that the record(s) in column "TRFLAG" of Pannasoft Ingenuity's reference table (example table "GCDATA") with value 'D' or 'P' are not used for training activities. Details of this flag are explained in Appendix A.

Prediction Module

The prediction module utilizes knowledge (prototypes) learned in the training module. It is capable of predicting unknown patterns based on the features given. The number of features presented must be the same with the number of attributes in the knowledge. However, missing values are allowed for these attributes presented to Pannasoft Ingenuity's prediction modules.

Refer to the previous normalization chapter to produce the required data before submitting to the prediction module. Pannasoft Ingenuity's prediction module will utilize the normalized data, i.e., View<Job_Code>NormPredictData, in the prediction process.

Pannasoft Ingenuity's prediction module checks the PRFLAG in Pannasoft Ingenuity's reference table (example "CRSDATA" table) before making any prediction. The user needs to set PRFLAG to 'Y' for those samples that are needed to make prediction and 'N' for those that are not in use for prediction. To perform the prediction process, Pannasoft Ingenuity's prediction module will select those records that contain TRFLAG='A', TRFLAG='U' or TRFLAG='T'. Please refer to Appendix A for more details about this PRFLAG.

To retrieve the predicted results, the user can refer to column RESULT in Pannasoft Ingenuity's reference table (example "CRSDATA" table). After the prediction process, PRFLAG is set to 'N' and prediction will not be repeated in future. The value in column RESULT is in normalized form. The database object view named Norm<Job_Code>ClassData (example view NormGCClassData) has the description value for column RESULT.

For example, when raw data contain only 2 classes (good and bad customers) in German Credit Data, the normalization process sets the good customers to 0 and bad customers to 1 (as defined in View named NormGCClassData). Then, column RESULT shows only 0 or 1 for prediction. Column CONFIDEN indicates the confidence factor of the prediction. Value 1.0 indicates full confidence while 0.0 indicates no confidence. This column only contains value if the "ruleOn" flag in pannasoft.properties is set to "true", otherwise, value=-1000 will be given to indicate no confidence value. This predicted result can be obtained by using the primary reference key in column "ID" of "GCDATA" table (it should be the same as the data in column "ID" of "GCDATA" table) to retrieve Pannasoft Ingenuity's prediction result.

Sample Code: Invoke Prediction module in Java

```
String szPropFile = "c:/pannasoft/pannasoft.properties";
com.pns.extract.Ingenuity myObj = com.pns.extract.Ingenuity.getInstance
(szPropFile);
myObj.doPrediction ();
```

Note: Remember to include the given jar file, i.e. file com.pns.extract-x.x.x.x.jar, into your classpath before compiling and executing Pannasoft Ingenuity modules.

In order to have predicted record(s), the records in Pannasoft Ingenuity's reference table (example "CRSDATA" table) need to have data PRFLAG='Y' and TRFLAG ='D'. Pannasoft Ingenuity's prediction module searches all records with PRFLAG='Y' in Pannasoft Ingenuity's reference table (example "CRSDATA" table) for prediction activities. Pannasoft Ingenuity always assumes record(s) in Pannasoft Ingenuity's reference table (example "CRSDATA" table) of column "PRFLAG" that has value 'N' which is not used for prediction activities.

Configuration in pannasoft.properties file

All configurations apply to both Pannasoft Ingenuity modules. The description for these network parameters is as follows.

The database configuration needs to be filled for Pannasoft Ingenuity to manipulate the target database data:

1. **db_authentication**: the database authentication approach. The possible values available are "server", "client".

2. **db_user_id**: the database authorized user accessing the target database.

3. **db_password**: the database authorized user password.

4. **db_url**: the JDBC driver dependent string allowing Pannasoft Ingenuity to access the target database, e.g., jdbc:microsoft:sqlserver://192.168.1.1:1433;DatabaseName=ingenuity ;SelectMethod=cursor.

5. **db_driver**: the JDBC driver name in use. It is database vendor specific components, e.g., com.microsoft.jdbc.sqlserver.SQLServerDriver.

6. **db_schema**: the database scheme for all target database objects.

The Apache Log4J configuration needs to be filled for logging purposes.

1. **log4j_lcf**: the Apache Log4J configuration (pannasoft.lcf) path.

In Pannasoft Ingenuity, there are a few network parameters that the user might be interested in and requested to modify the configuration file named pannasoft.properties.

1. **beta:** this is a learning parameter that will decide at which rate the recoding of the pattern should occur in the learning process. Note: 0 (more resistance to noisy data) < beta ≤ 1 (less resistance to noisy data). Default value is 1.0.

2. **rho/Test_rho:** A parameter that determines the degree of granularity of the learned patterns. Low values (→ 0) lead to rough categories with broad generalization while high values (→ 1) lead to fine categories with narrow generalization. **Note:** 0 ≤ rho < 1. Default value is 0.0.

3. **seed:** a number that controls the randomization for creating a training data and a testing data from raw data. Default value is 10.

4. **iNominator/iDenominator:** number of raw data that is distributed to become trained set. Default value = 2/3.

5. **noVoter:** number of base network in Pannasoft Ingenuity. Default value = 1.

6. **uthreshold:** A pruning threshold. Prototypes (knowledge) which are lower than this threshold will be deleted from the system to avoid complexity to form in the system. The default value is 0.4.

7. **reload_weights:** true = enable network to retrieve old prototype from the database, false = network will re-train all the available data (TRFLAG = 'A' and TRFLAG = 'T').

8. **ruleOn:** true = enable rule extraction (convert the prototype from database to linguistic format); false = disable rule extraction.

9. **autoTune:** true = enable network to auto adjust suitable network parameters in (2) and (3), false = user needs to manually insert the network parameter of beta and rho.

10. **job:** multiple jobs in a database. Symbols comprising semi colon (;), comma (,) and plus (+) are reserved characters to differentiate between databases, tables/views and features.

Example:
german_data;GCDATA;ViewGCNormData;ViewGCNormTrainData;ViewGCNormPredictData;rec_id,class,checking,duration,history,purpose,credAmt,saving,employ,installment,status,debtors,residence,property,age,otherPlan,housing,existCred,job,liability,tel,foreignWorker+wbc_data;WBCDATA;ViewWBCNormData;ViewWBCNormTrainData;ViewWBCNormPredictData;rec_id,class,Clump,CellSize,CellShap,Adhesion,EpiCell,Nuclei,Chroma,Nucleoli,Mitoses

Configuration in pannasoft.lcf file

All configurations in pannasoft.lcf originates from Apache[2]. To enable Pannasoft Ingenuity's logging, the following configuration is required.

1. **log4j.appender.xxx.File**: the log file path for Pannasoft Ingenuity's logging.

[2] Details of Apache Log4J configuration documentation can be obtained at url http://logging.apache.org/log4j/docs/manual.html

How to interpret results table

There are a few tables that keep the results produced by Pannasoft Ingenuity. DATAF2A, DATAF2B and DATAFAB are used to keep the weights created by Pannasoft Ingenuity. Table DATARES is used to keep the validation results of the test samples while DATARULE and DATACF are used to store the rules and the confidence factor respectively. Below are some of the descriptions for Table DATARES, DATARULE and DATACF.

DATARES:

Column Name	Descriptions
REC_ID	Test samples that have been used to validate the performance of Ingenuity after the learning process. Total records in this table for a particular data set are equivalent to 1 – iNominator/iDenominator.
WINNER	Selected weight that has been used to predict the test sample
OUTCLASS	Predicted class
STATUS	Status of the prediction, 0 indicates incorrect prediction while 1 indicates correct prediction
DATA_ID	Name of the data set used
VOTER_ID	i-th classifier. 0 indicates combined result from all results while the other value, i.e., 1,2,3,4,5,... indicate i-th classifier. This value is influenced by **noVoter**.

[State clearly that this section describes the results of rules extraction. Rules extraction feature is only available in Demo and Professional editions]

DATARULE:

Column Name	Descriptions
RULE_ID	Rule record ID
ATT_ID	Attribute/Feature ID, equivalent to number of feature given to the Pannasoft Ingenuity training module.
LBOUND	Lower bound value of the knowledge
UBOUND	Upper bound value of the knowledge
DATA_ID	Name of the data set used
VOTER_ID	i-th classifier

LBOUND and UBOUND indicate the covering value of a weight from minimum to maximum respectively.

DATACF:

Column Name	Descriptions
CF_ID	Confidence Factor ID, this ID are equivalent to RULE_ID
CF_VAL	Value of the Confidence Factor (CF)
CLASS	Belonging class
DATA_ID	Name of the data set used
VOTER_ID	i-th classifier

Figure 2 and 3 show two examples of rule and confidence factor table respectively after the Pannasoft Ingenuity training process. In Figure 2, rule_id = 1 indicates that this is the first rule in the table while att_id = 1 to att_id = 4 indicate that there are 4 features in this rule for data set called “iris_data”. Voter_id = 1 indicates that the rules are generated by first classifier. In Figure 3, cf_id = 1 belongs to rule_id = 1. cf_val and class are indicated as confidence factor and the belonging class of the rule respectively. Table 8 shows a simple rule interpretation for the first and second rule in Figure 2.

rule_id	att_id	lbound	ubound	data_id	voter_id
1	1	0.47916700000000001	0.8125	iris_data	1
1	2	0.34375	0.6875	iris_data	1
1	3	0.61864399999999997	0.72033899999999995	iris_data	1
1	4	0.53125	0.71875	iris_data	1
2	1	0.66666700000000001	0.85416700000000001	iris_data	1
2	2	0.5	0.75	iris_data	1
2	3	0.77118600000000004	0.89830500000000002	iris_data	1
2	4	0.6875	1.0	iris_data	1
3	1	0.25	0.5	iris_data	1
3	2	0.53125	0.9375	iris_data	1
3	3	0.25	0.36440699999999998	iris_data	1
3	4	0.25	0.40625	iris_data	1
4	1	0.5625	0.95833299999999999	iris_data	1
4	2	0.40625	0.5625	iris_data	1
4	3	0.74576299999999995	1.0	iris_data	1
4	4	0.78125	0.9375	iris_data	1
5	1	0.39583299999999999	0.75	iris_data	1
5	2	0.25	0.59375	iris_data	1
5	3	0.567797	0.682203	iris_data	1
5	4	0.53125	0.65625	iris_data	1

Figure 2: Data in DATARULE Table

cf_id	cf_val	class	data_id	voter_id
1	0.90000000000000002	0.5	iris_data	1
2	0.86666670000000001	1.0	iris_data	1
3	1.0	0.0	iris_data	1
4	0.89090913999999999	1.0	iris_data	1
5	0.64000005000000004	0.5	iris_data	1

Figure 3: Data in DATACF Table

Rule_ID = 1	Rule_ID = 2
IF	**IF**
ATT_ID = 1 from 0.4792 to 0.8125	ATT_ID = 1 from 0.6667 to 0.8542
ATT_ID = 2 from 0.34375 to 0.6875	ATT_ID = 2 from 0.5 to 0.75
ATT_ID = 3 from 0.6186 to 0.7203	ATT_ID = 3 from 0.7712 to 0.8983
ATT_ID = 4 from 0.53125 to 0.71875	ATT_ID = 4 from 0.6875 to 1.0
THEN	**THEN**
Class = 0.5 with CF = 0.90	Class = 1.0 with CF = 0.8667

Table 8: Interpretation in IF-THEN format

One will notice the rule values in Table 8 are all fuzzy numbers. User can convert the fuzzy values back to the original value using equation (2)

$$x_i = \left[\frac{f(x_i) - 0.2}{(1 - 0.2)} \right] \times \left(\max(x) - \min(x) \right) + \min(x) \qquad (2)$$

Where $f(x_i)$ = fuzzy number

APPENDIX A: DETAILS OF THE REFERENCE TABLE

Also known as <Job_Code>DATA (example GCDATA)

No	Column Name	Column Type	Nullable	Column Description	Sample Data and Prerequisite
1.	APPNO	varchar	No. No 0 to 999 values are allowed for this field.	Application / Agreement number; such as Applicant filled in form number, existing system generated running number or etc. This number is the table primary key or indexed data.	
2.	CRDATE	timestamp	No	Record creation timestamp.	Current timestamp (Default Value)
3.	TRFLAG	varchar	No	Training flag, it is the record status used for Pannasoft Ingenuity training.	**A**=Activate new record for training or re-training. **Example record**: historical/existing data in old system; transaction record completed (refer to all installment paid by customer). **U**= Activate existing record for training or re-training. **Example record**: existing record information is updated, and this flag will trigger Pannasoft Ingenuity to update Pannasoft Ingenuity repository. **D**=Deactivate record for training. **P**=Purged record, upon customer request, to avoid taking in this record for Pannasoft Ingenuity training.

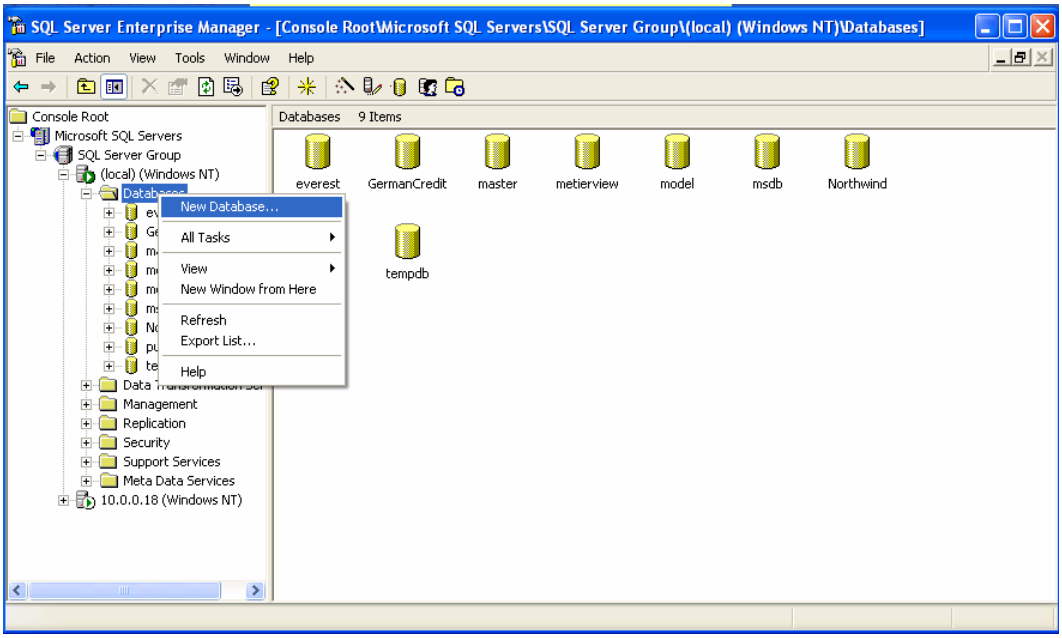
					Pannasoft possible flag: **T**= Pannasoft Ingenuity trained record. Pannasoft will turn this to T after the end of the training process.
4.	PRFLAG	varchar	No	Prediction flag, it is the record status used for Pannasoft Ingenuity prediction. Every new record in this table will trigger Pannasoft Ingenuity prediction in operation under the batch mode. **Note**: Initial data (historical data from existing system) should use default value (N) in this column.	**Y**=Prediction yes flag; It is waiting for Pannasoft Ingenuity prediction result flag. This flag instructs Pannasoft Ingenuity to perform prediction. Pannasoft Ingenuity will only retrieve this kind of record for performing prediction. Pannasoft possible flag: **N**=Prediction no flag; It is Pannasoft Ingenuity's complete prediction result flag. This flag shows Pannasoft Ingenuity has finished performing prediction. (Default value)
5.	RESULT	float	No	Pannasoft Ingenuity's prediction result. **Note**: Initial data (historical data from existing system) should use default value (N) in this column.	1000=Unknown/ no prediction has been made (Default value)
6.	MDDATE	timestamp	Yes	Record modification timestamp. Pannasoft Ingenuity will update this timestamp to trace the system response time.	NULL=(Default Value)

7.	CONFIDEN	double	Yes	Pannasoft Ingenuity's prediction confidence level. Pannasoft will update this value when the result is generated in column "result".	**Sample data:** 0.99, 0.8, 0.75 NULL=Column is not implemented yet (Default value)
8.	REC_ID	int	No	An auto increased reference number for Pannasoft Ingenuity to use. This auto number will be auto increased when a new key ID (original ID from raw data) is inserted (from trigger) into column APPNO.	

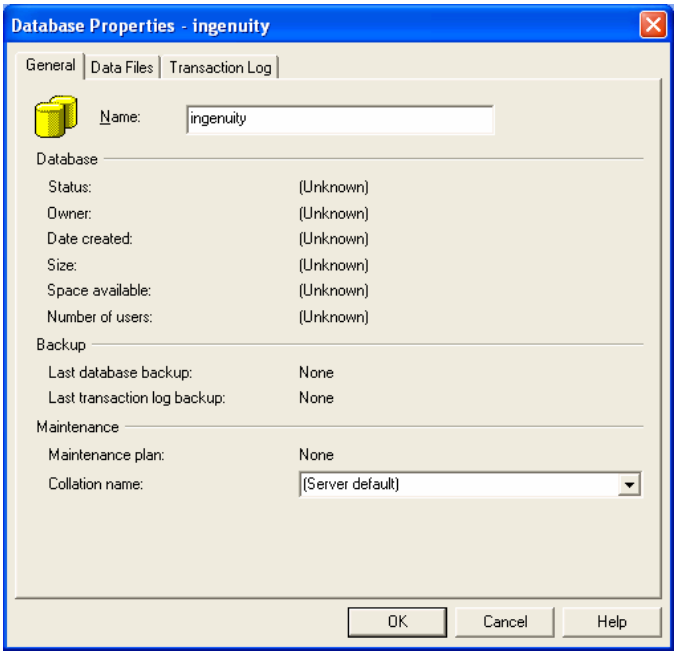
Note:
This table needs to be created as the specified name for Pannasoft Ingenuity to work.

APPENDIX B: CREATE & RESTORE MICROSOFT SQL 2000 DATABASE

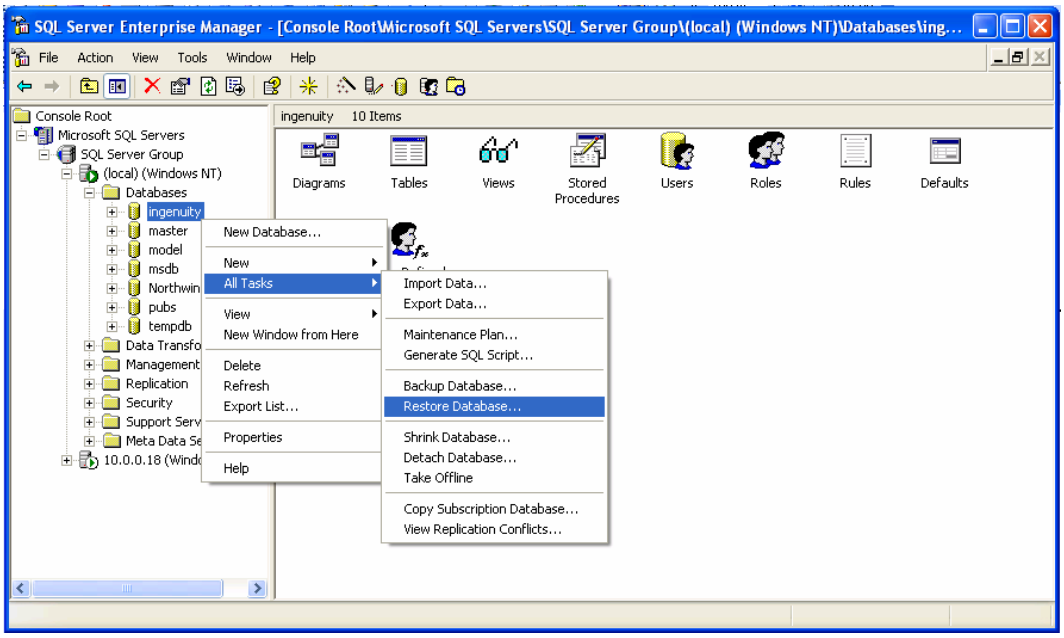
The user can use SQL Server Enterprise manager to perform the mentioned steps.



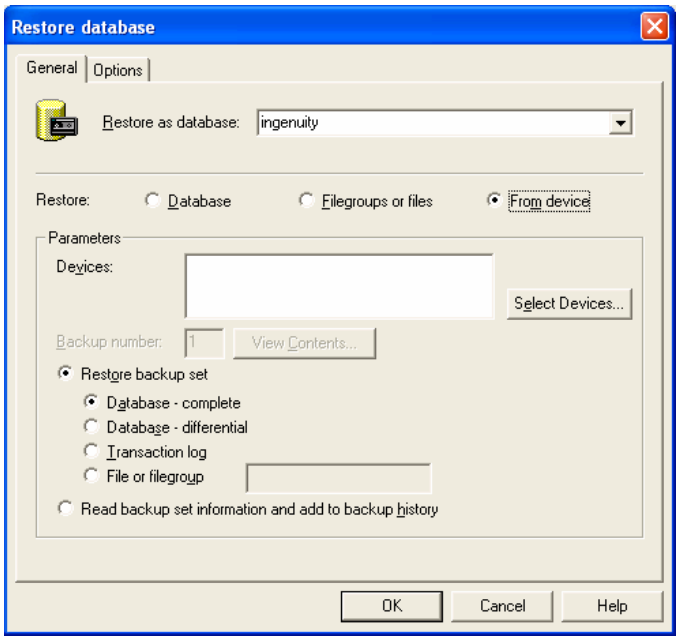
Right click on “Databases” and choose “New Database...” to create a new database.



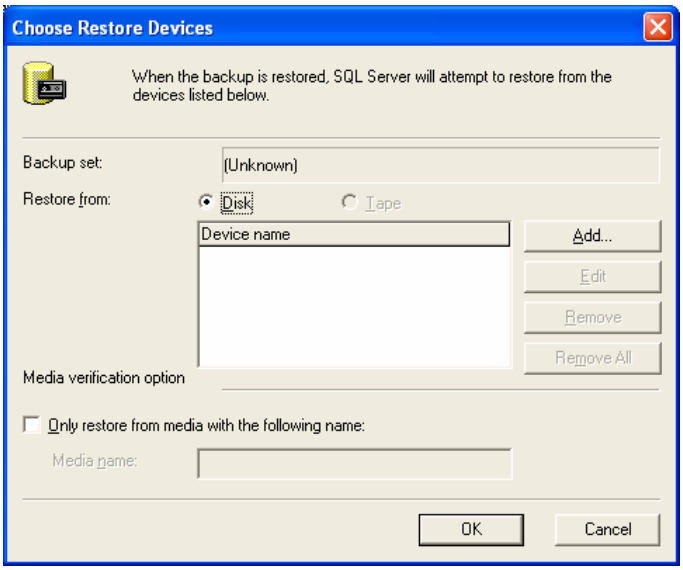
Name the new database as “ingenuity” as shown above.



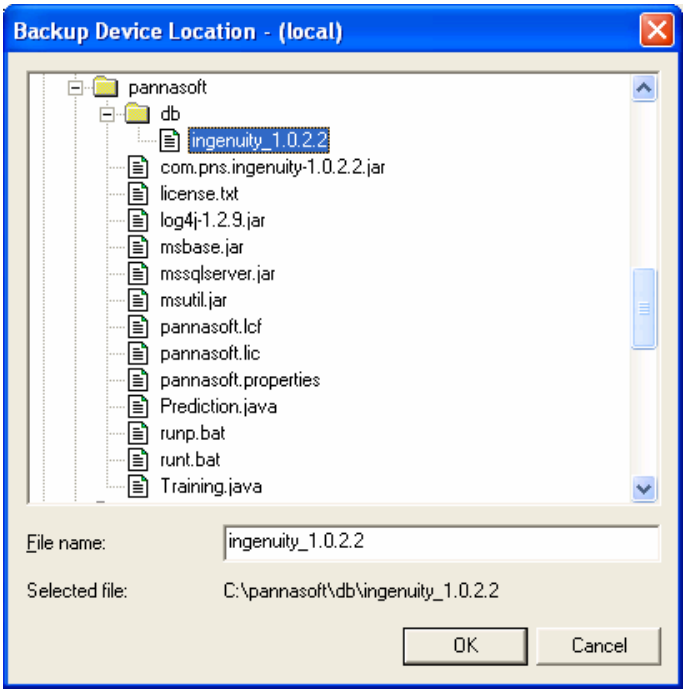
Right click on “ingenuity” to perform the backup step by choosing “Restore Database...”.



Choose the option “From device”, and click on the “Select devices...” button.

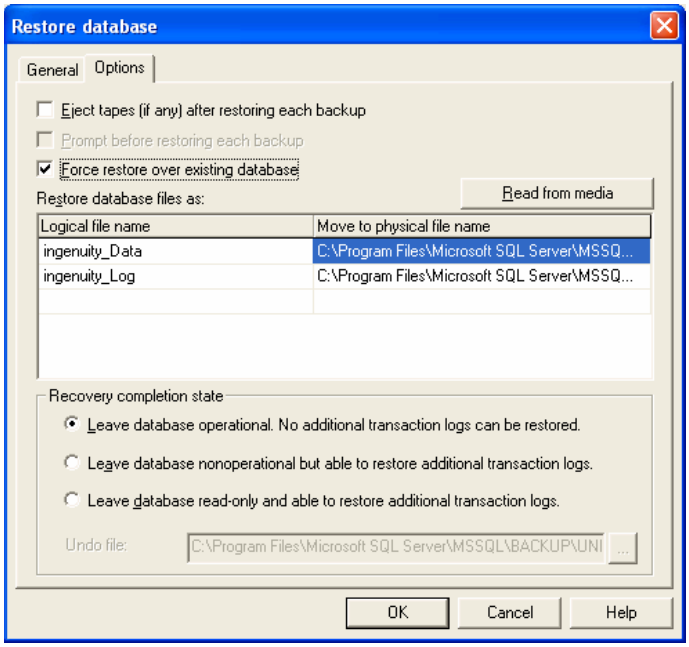


Click on the “Add...” button as shown on the screen above.

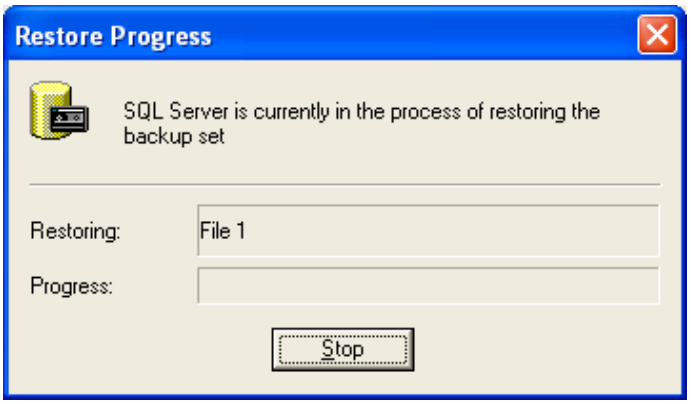


Browse to the location where the Microsoft SQL 2000 database backup file is residing and click on the “OK” button when done.

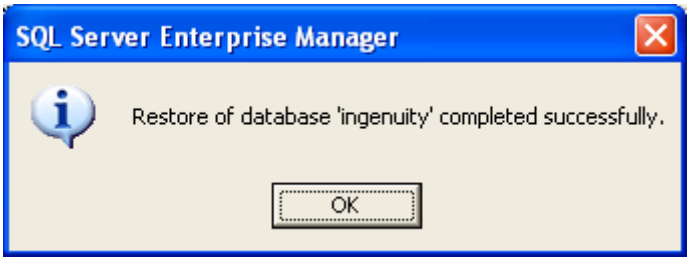
Click the “OK” button to proceed with the step.



Choose the option “Force restore over existing database” to force the backup process into the target database server.



The screen above shows the restoration of the “ingenuity” database is in progress.



When the “ingenuity” database is completely restored, a note is displayed as shown above.

GLOSSARY

Data mart: A repository of data gathered from operational data and other sources. It is designed to serve a particular community of knowledge workers

Features or attributes: The characteristics of the data

Classes or targets: Categories of a data sample

Normalization: A process to standardize the data

Fuzzy numbers: Existing numbers between 0 and 1, e.g., 0.125, 0.0026

Supervised learning system: A system that has a learning process based on data which are labeled by output targets

Weight: Prototype or Pannasoft Ingenuity's knowledge.

Linguistic rule: Statements that are understood by human, i.e., IF Body_Temperature > 37 C, THEN Fever.

Confidence factor: The confidence level of a rule provided by Pannasoft Ingenuity. The higher the value of the confidence factor, the higher the confidence of this rule in prediction

Missing data: Missing or null data value in a field. It indicates the absence of any meaningful or valid information in features.

About Pannasoft Technologies

Pannasoft Technologies was founded in 2003 with seed funding from a prominent venture capitalist. The company is a MSC (Multimedia Super Corridor) status company and an advanced partner for IBM Partnerworld. In 2004, Pannasoft Technologies was awarded the MGS (MSC Grant Scheme) to conduct R&D in data mining technology.

Pannasoft Technologies specializes in Soft Computing, a state-of-the-art technology that the company uses extensively to develop intelligent software solutions for a wide variety of industries including manufacturing, engineering, trading, education, healthcare and financial services. Our solutions enable monitoring and optimization of business and engineering processes to address their operational difficulties and to achieve optimal performance while minimizing costs and maximizing profits.

1-02-19, Jalan Mayang Pasir 1,
Mayang Mall,
11900 Bayan Lepas,
Penang,
Malaysia.

Tel : +604-645 5218
Fax : +604-645 2398

enquiry@pannasoft.com *email*
www.pannasoft.com *website*