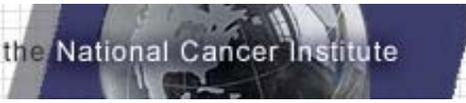




caBIG™ cancer Biomedical
Informatics Grid™

an initiative of the National Cancer Institute



VISDA

Installation Guide



Table of Contents

Introduction.....	1
Overview of VISDA	1
VISDA Minimal System Requirements	1
Minimal System Requirements	1
VISDA Software and Technology Requirements.....	2
Software Requirements	2
Required Software—Not Included in the VISDA	2
Documentation and Source Code Styling Tools	2
Useful tools included in the VISDA.....	3
Upgrading to VISDA from Previous VISDA Versions.....	4
Installing the VISDA.....	4
Preliminary Considerations.....	4
Downloading the VISDA.....	4
Compiling the Code	5
Running the Example.....	5
Perform Gene Clustering	6
Load Local Mage-ml Data	6
Connect to Remote Site (caArray Database)	7
APPENDIX I VISDA COMPONENTS.....	8
Contacting Technical Support	8
VISDA Application Support.....	8

Introduction

This installation guide outlines the supported configurations and technical installation instructions for the VISDA. Directions for testing the VISDA are also included here.

All of the examples and screenshots included in this chapter are Windows specific. If you are using a different platform, then modify the information as appropriate for your system.

Overview of VISDA

VISDA (Visual Statistical Data Analyzer) is a caBIG™ analytical tool for cluster modeling, visualization, and discovery. Being statistically-principled and visually-insightful, VISDA exploits human gift for pattern recognition and allows users to discover hidden clustered data structure within high dimensional and complex biomedical data sets. The unique features of VISDA include its hybrid algorithm, robust performance, and “tree of phenotype”. With global and local biomarker identification and prediction functionalities, VISDA allows users across the cancer research community to analyze their genomic/proteomic data to define new cancer subtypes based on the gene expression patterns, construct hierarchical trees of multiclass cancer phenotypic composites, or to discover the correlation between cancer statistics and risk factors. Input is a microarray data loaded from local pre-defined tab-delimited format file in Stanford text format, local MAGE-ML file, or from caArray database, with common gene identifiers and sample names. Outputs are the “soft-clustering” probabilities of the samples/genes in each cluster and a hierarchical “tree of phenotype” or “tree of gene module”. This release of VISDA supports caBIG compatibility level silver.

VISDA Minimal System Requirements

Minimal System Requirements

- Internet connection
- Tested Platforms

The current VISDA v1.0 supports Windows and Linux platforms, as shown in Table 1. It does not support Mac OS X platform since the C++ source code of VISDA v1.0 was not compiled under Mac OS X.

	Linux Server	Windows
Model	Dell GX 270	Dell GX 270
CPU	1 x Intel® Pentium™ Processor IV 1.50GHz	1 x Intel® Pentium™ Processor IV 3.20GHz
Memory	1 GB	2 GB
Local Disk	System 2 x 60GB	System 1 x 60GB
OS	Red Hat Linux ES 3 (RPM 2.4.21-20.0.1)	Windows XP/2000 Professional

Table 1 Platform Testing Environment

VISDA Software and Technology Requirements

Software Requirements

Required Software—Not Included in the VISDA You must download and install the required software that is not included with the VISDA (listed in Table 2) The software name, version, description, and URL hyperlinks (for download) are indicated in the table.

(Required software that *is* included with the VISDA is listed in Appendix I on page 8.)

Software Name	Version	Description	URL
The Runtime Environment (JRE):Java 2 Standard Edition (J2SE)	JRE5.0 or higher is needed for running VISDA	The J2SE Runtime Environment (JRE) allows end-users to run Java applications	http://java.sun.com/j2se/1.5.0/download.jsp

Table 2 Required software and technology for VISDA

Documentation and Source Code Styling Tools

The following tools, useful for documentation and source code styling, are part of the VISDA framework.

**Useful tools
included in the
VISDA**

- **Javadoc** – Execute the Ant task doc “ant javadoc” to generate Javadocs for your beans. Your javadocs will be generated to the {home_directory}\output\javadoc directory. For more information on Javadoc see <http://java.sun.com/j2se/javadoc/>.

NOTE:



VISDA Components are listed in Appendix I on page 8.

Upgrading to VISDA from Previous VISDA Versions

This section includes information on upgrading your product from previous version.

Note the following points in downloading and installing the VISDA upgrade:

Step	Action
1	Follow the directions for downloading the VISDA deployment zip file on page 4. If you have previously installed VISDA v0.x, to upgrade to VISDA v1.0, unzip the VISDA deployment zip file to a new directory.
2	Follow installation instructions for the new VISDA beginning on page 4.

Installing the VISDA

Preliminary Considerations

 BEFORE YOU BEGIN	<p>The VISDA has been tested with the operating systems and hardware specified on pages 1 through 3 of this guide. We cannot guarantee the VISDA will work if you are using variations of these operating systems and/or hardware.</p>
---	--

Downloading the VISDA

Complete the following steps to download the VISDA:

Step	Action
1	Go to the VISDA download web site (https://cabig.nci.nih.gov/tools/VISDA) to download visda-Deploy-V1.0.rar to a temporary directory on your computer (for example, C:\temp in Windows). This may take several minutes.

Step	Action																																				
2	<p>Extract the contents of VISDA distribution zip file to your root directory. For example, if your root directory in Windows is C:\, then your VISDA home directory is C:\VISDA</p> <p>Include appropriate information and screen shots.</p> <table border="1"> <thead> <tr> <th>Name ^</th> <th>Size</th> <th>Type</th> </tr> </thead> <tbody> <tr> <td>data</td> <td></td> <td>File Folder</td> </tr> <tr> <td>doc</td> <td></td> <td>File Folder</td> </tr> <tr> <td>lib</td> <td></td> <td>File Folder</td> </tr> <tr> <td>java.policy</td> <td>1 KB</td> <td>POLICY File</td> </tr> <tr> <td>libVISDA_CJavaInterface.so</td> <td>129 KB</td> <td>SO File</td> </tr> <tr> <td>MAGE-ML.dtd</td> <td>156 KB</td> <td>DTD File</td> </tr> <tr> <td>RunVISDA.bat</td> <td>1 KB</td> <td>MS-DOS Batch File</td> </tr> <tr> <td>RunVISDA.sh</td> <td>1 KB</td> <td>SH File</td> </tr> <tr> <td>VISDA_CJavaInterface.dll</td> <td>396 KB</td> <td>Application Extension</td> </tr> <tr> <td>VISDA_Readme_V1.0.txt</td> <td>5 KB</td> <td>Text Document</td> </tr> <tr> <td>VISDA_ReleaseNotes_V1.0.txt</td> <td>5 KB</td> <td>Text Document</td> </tr> </tbody> </table>	Name ^	Size	Type	data		File Folder	doc		File Folder	lib		File Folder	java.policy	1 KB	POLICY File	libVISDA_CJavaInterface.so	129 KB	SO File	MAGE-ML.dtd	156 KB	DTD File	RunVISDA.bat	1 KB	MS-DOS Batch File	RunVISDA.sh	1 KB	SH File	VISDA_CJavaInterface.dll	396 KB	Application Extension	VISDA_Readme_V1.0.txt	5 KB	Text Document	VISDA_ReleaseNotes_V1.0.txt	5 KB	Text Document
Name ^	Size	Type																																			
data		File Folder																																			
doc		File Folder																																			
lib		File Folder																																			
java.policy	1 KB	POLICY File																																			
libVISDA_CJavaInterface.so	129 KB	SO File																																			
MAGE-ML.dtd	156 KB	DTD File																																			
RunVISDA.bat	1 KB	MS-DOS Batch File																																			
RunVISDA.sh	1 KB	SH File																																			
VISDA_CJavaInterface.dll	396 KB	Application Extension																																			
VISDA_Readme_V1.0.txt	5 KB	Text Document																																			
VISDA_ReleaseNotes_V1.0.txt	5 KB	Text Document																																			

Compiling the Code

To compile C++ source code, C/C++ compiler is needed (e.g. gcc 3.4 and glibc 2.1.3 under Linux). Under 'visda-C-V1.0' directory, run 'CompileVisdaC.bat' for windows or 'CompileVisdaC.sh' for Linux. The VISDA_CJavaInterface.dll or libVISDA_CJavaInterface.so file will be created under 'visda-C/out' directory.

To open and compile Java source code, Eclipse 3.1 is required. To compile the source code directly, Apache ANT (<http://ant.apache.org>) is needed. Under 'visdaDev-V1.0\src' directory, run 'ant build' to compile java codes; run 'ant jar' to generate jar file; run 'ant javadoc' to generate API docs. The jar file will be created under 'visdaDev' directory

Running the Example

The simplest way to run VISDA is to just double-click "RunVISDA.bat" under Windows or type in "RunVISDA.sh" from command line under Linux/Unix. This will run VISDA with the default memory allocation for Java, typically 512Mb. To run VISDA with additional memory, you need to modify the parameters: -Xms512m -Xmx512m in the "RunVISDA.bat" or "RunVISDA.sh" file. In this variant, the -Xms parameter is the memory allocated to VISDA at startup, and -Xmx is the maximum memory allocated to VISDA.

The user can run VISDA with an example data file "NCI_63Test_2label.txt" located under "/visda-Deploy-V1.0/data" in the following steps.

Step 1: File -> Open a Tab-Delimited DataSet

After loading the data, one 'Dataset' node will be added under view tree. The heat map of the original data will be displayed.

- Step 2: Config -> Configure Parameters
Choose the parameters, including the clustering type, feature number, supervised/unsupervised, label, Posterior Probability threshold, HC-DCA, and MDL options. Click 'Set' button. Then selected data heat map and features' performance table can be viewed under the 'Analysis' node.
- Step 3: Analysis -> Start Top-level Analysis
Follow the instructions in dialogs. The top level result will be put under the 'Levels/Level_1'. The top cluster result will be put under the 'Clusters/Cluster 1'.
- Step 4: If lower level clustering is needed, then go to: Analysis -> Start Deeper-level Analysis
The next lower level results will be put under the 'Levels/Level_2'. The new-generated clusters will be put under their parent-cluster node.
- Step 5: Repeat step 4 to perform deeper level analysis.
- Step 6: If new analysis for the current dataset is needed, go from step 2 to reconfigure the parameters.
- Step 7: If new dataset will be loaded, go from step 1 to open a new data file.

Performing Gene Clustering

The default options are not for gene clustering, to conduct gene clustering, please do the following in the “Configure Parameters” box.

- Step 1: Check the “gene clustering” tick box.
- Step 2: Select the option of “All of the *** Features”.
- Step 3: Click on the “set” button.

Then you can begin analysis.

Loading Local Mage-ml Data

Please follow the following steps:

- Step 1: Click “File” → “Open a Local/Remote MAGE-ML DataSet”.
- Step 2: In the “Import MAGE-ML Data” box, right click “Local Experiments”, left click on the pop up “Add a new experiment”.

- Step 3: In the “Open a MAGE-ML DataSet” box, highlight the data file that you want to open. For example, select the file “E-CAGE-8.mageml.gz” in the data directory. Click “Open” button.
- Step 4: In the “Import MAGE-ML Data” box, on the left, you see that under “Local Experiments” there is “E-CAGE-8”. Click on the node (shown as a small circle with a bar) of “E-CAGE-8”, you see “2 local BioAssays”.
- Step 5: Right click on “2 local BioAssays”, you see multiple choices. Select a suitable option, e.g. “Get MeasuredBioAssays”. Then on the right side of the “Import MAGE-ML Data” box, you see two arrays, select them. You may not need to check the “Load Data with Heat Map Display” tick box, because when the gene number is big, showing the heat map needs a long time.
- Step 6: Click the “Save & Load Data” button, the “Select a QuantitationType before load” box pops up. Select one of the options, e.g. RAT2. Click the “OK” button.
- Step 7: The data has been loaded to the software. A file “save” box pops up to save the extracted data to a tab-delimited file for later use.

Connecting to Remote Site (caArray Database)

Please register for the caArray database and use your own user name and password. The caArray website is <https://array.nci.nih.gov/caarray/home.action>.

Appendix I VISDA Components

The VISDA includes the following components:

- Components under visda-Deploy-1.0 directory
 - VISDA_CJavaInterface.dll
 - libVISDA_CJavaInterface.so
 - RunVISDA.bat
 - RunVISDA.sh
 - MAGE-ML.dtd
 - java.policy
 - VISDA_Readme_V1.0.txt
 - VISDA_ReleaseNotes_V1.0.txt
 - VISDA_InstallationGuide.doc
 - lib directory: visda.jar, MAGE-OM APIs and other needed jar files
 - doc directory: visda APIs HTML document
 - data directory: data example files
 - Framework packages
 - edu.vt.cbil.visda
 - edu.vt.cbil.visda.comp
 - edu.vt.cbil.visda.data
 - edu.vt.cbil.visda.view
 - edu.vt.cbil.visda.util
 - edu.vt.cbil.visda
 - org.math.plot
 - gov.nih.nci.system
 - gov.nih.nci.common
 - gov.nih.nci.mageom
 - org.biophage
 - Ant buildfile: build.xml
-

Contacting Technical Support

**VISDA
Application
Support** Send email to visda_support.vt.edu to request support.
