

# Octave Gtk

## Software Requirements Specification

Ramasamy C, Muthiah A, Hemant Kumar, R Saravana Manickam  
RedHat ID: RHSID041383  
National Institute of Technology,  
Tiruchirapalli, India

December 12, 2004

### Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	Document conventions . . . . .	2
1.2	Intended audience and reading suggestions . . . . .	2
1.3	Scope of development . . . . .	3
1.4	Defenitions, Acronyms, Abbreviations . . . . .	3
1.5	References . . . . .	3
1.6	Document overview . . . . .	3
<b>2</b>	<b>General Description</b>	<b>3</b>
2.1	Product Perspective . . . . .	3
2.2	Product Function . . . . .	4
2.3	User classes and characteristics . . . . .	4
2.4	Operating Environment . . . . .	4
2.5	Design and Implementation constraints . . . . .	5
2.6	User Documentation . . . . .	5
2.7	Assumptions and dependencies . . . . .	5
2.8	Overview of data requiremetns . . . . .	5
2.9	General constraints, assumptions, dependencies, guidelines . . . . .	5
2.10	User view of product use . . . . .	6
<b>3</b>	<b>External interface requirements</b>	<b>6</b>
3.1	User Interface . . . . .	6
3.2	Hardware Interface . . . . .	6
3.3	Software Interface . . . . .	6
3.4	Communication Interface . . . . .	6
<b>4</b>	<b>System Features</b>	<b>6</b>

<b>5</b>	<b>System Features</b>	<b>6</b>
5.1	System Features . . . . .	6
5.2	Stimulus/response sequences . . . . .	6
5.3	Functions requirements . . . . .	7
<b>6</b>	<b>Other non-functional requirements</b>	<b>7</b>
6.1	Performance requirements . . . . .	7
6.2	Safety requirements . . . . .	7
6.3	Security requirements . . . . .	7
6.4	Software quality attribute . . . . .	7
<b>7</b>	<b>Other non-functional requirements</b>	<b>7</b>
7.1	Performance requirements . . . . .	7
7.2	Safety requirements . . . . .	7
7.3	Security requirements . . . . .	7
7.4	Software quality attribute . . . . .	7
7.5	Business rules . . . . .	8
7.6	Special user requirements . . . . .	8
7.7	Backup and recovery . . . . .	8
7.8	Data migration . . . . .	8
7.9	Data retention . . . . .	8
7.10	User training . . . . .	8
7.11	Installation . . . . .	8
<b>8</b>	<b>Other requirements</b>	<b>8</b>

# 1 Introduction

The Software Requirements Specification **SRS** of **Octave-Gtk** are laid out in this document. It is expected to be useful in installation, configuration, and documentation purposes of end users, designers **Octave-Gtk** and hackers in general.

## 1.1 Document conventions

1. References made, are indicated using [number].eg : [1]
2. **Bold facet** is used to refer to projects.eg : **Octave-Gtk**
3. Abbreviations are italicised.eg : *GNU*

## 1.2 Intended audience and reading suggestions

1. *Octave Users, Scientists, Engineers* : Octave users will find out how **Octave-Gtk** can be useful to their work
2. *Administrator* : Administrators, will generally need to know about security, installation, and system requirements
3. *Developers* : As a **Octave-Gtk** hacker, developers are expected to understand the whole document, and to read

### 1.3 Scope of development

**Octave-Gtk** aims to bring the full featured toolkit like **GTK** to GNU Octave, to make scientific computing tools with a GUI front end scripted from Octave itself. Also in the process, we would like to build a Octave GUI, using **Octave-Gtk** bindings in Octave itself, to show the power of **Octave-Gtk** itself.

This directly translates to end users, as a new and powerful paradigm for constructing GUI's with minimal knowledge of the constructs, and easy interface for scientific programs, in a really short time. **Octave-Gtk** will compete with other *Rapid Application Development* tools, in its ease of use, and short design-test-release cycle.

### 1.4 Defenitions, Acronyms, Abbreviations

1. **Octave-Gtk** : *Octave – Gtk is the GTK binding for GNU Octave.*
2. **GTK** : *Gimp Tool Kit*
3. *GNU Octave* : *GNU Octave is a high performance scientific computation tool*
- 4.
- 5.
- 6.
- 7.
- 8.

### 1.5 References

1. *GNU Octave* : [www.octave.org](http://www.octave.org) *GNU Octave interpreter, Octave Language, Sources, Octave mailing lists*
2. *GTK* : [www.gtk.org](http://www.gtk.org) *GTK API documentation, GTK source, GTK mailing lists*
3. *Octave Wiki* : <http://wiki.octave.org> *Documentation on extending Octave, Code docson writing dynamic*
4. *Octave – Forge* : <http://octave.sourceforge.net> *Extrapackages, libraries and tools for GNU Octave.*
5. *PyGtk* : [www.pygtk.org](http://www.pygtk.org) *PythonGtk – binding, by James Henstridge. Referenced documentation on writing*

### 1.6 Document overview

## 2 General Description

### 2.1 Product Perspective

**Octave-Gtk** is a Octave Gtk binding, which helps you access the GTK C API from the GNU Octave's, interpreted language Octave. This cross language interoperability is achieved by **Octave-Gtk** binding code which enables type safe,

idempotent access of functions and objects from the either domains of Octave and C, in a clean, and transparent manner, hiding the details to the end user.

## 2.2 Product Function

1. Create and use the GObject system and types
2. Make GUI programming easy with **GTK**
3. Allow creation of custom GUI components, which are interoperable with the C and Octave
4. Allow Octave scripts to interactively plot, perform actions with GUI
5. Create GTK/GNOME programs with Octave, so that numeric computation is not duplicated

## 2.3 User classes and characteristics

1. Octave Users, Scientists, Engineers: Users will generally learn from *octave-gtk-demo* command, on how to use **Octave-Gtk** for their octave work. Also GTK C API will guide them in learning the binding.
2. Administrator: Administrators will just need to make sure their GNU/Linux system has GNU Autotools installed. Installation is a matter of
  - (a) `./configure --prefix=/you/usr/directory`
  - (b) `make`
  - (c) `make install`
3. *Developers : To hack Octave-Gtk, and enhance the code generator, please read through the source code in the*

## 2.4 Operating Environment

1.  $\zeta$ =GTK+-2.0
2.  $\zeta$ =Octave-2.1.50 compiled with dynamic loading enabled
3.  $\zeta$ =Python
4.  $\zeta$ =GNU/Linux or UNIX like system
5. any 32bit processor

## 2.5 Design and Implementation constraints

Implementation constraints are limited to the performance of the Octave interpreter. Consequently the callbacks and glue code, make **Octave-Gtk** programs **slower** than their corresponding GTK counterparts.

Octave GTK code has to be updated as and when GTK API changes. As of now, this cannot be overcome unless we design some kind of simpler API on top of what **Octave-Gtk** provides.

## 2.6 User Documentation

For programming convenience, all the function names in Octave-Gtk are same as the standard C function names defined in GTK+ API. The names of C classes has been also preserved for sake of programming convenience. Thus for User Documentation the official GTK+ can be referred except in few special cases. For all such cases our additional user documentation lists such peculiarities.

## 2.7 Assumptions and dependencies

Not applicable.

## 2.8 Overview of data requirements

Not applicable.

## 2.9 General constraints, assumptions, dependencies, guidelines

The library has been tested on following hardwares.

1. Intel P4, 1.5GHz  
256MB of SDRAM  
Fedora Core-2  
8MB of Video RAM
2. 8 Intel Xeon, 2.70GHz processor  
8GB of RAM  
RHEL
3. Intel PIII, 833MHz  
128MB of RAM  
Mandrake-9.1, With Kernel-2.4  
8MB of Video RAM

There is no memory limitations but because of bulky Octave interpreter the response time may be slightly more.

## 2.10 User view of product use

This library aims to make GUI programming(with Octave language)a cake-walk. Use of Octave language coupled with GTK+ makes GUI programming easy.Users will be relieved from the headache of handling pointers and can use powerful Octave language.

## 3 External interface requirements

### 3.1 User Interface

Not Applicable.

### 3.2 Hardware Interface

Not Applicable.

### 3.3 Software Interface

GTK library version 2.0 or later, is the target of this **Octave-Gtk**binding, and we need *GTK+-2.0*. Also Octave extensions work iff Octave is compiled with *dynamic loading* enabled, in versions 2.1.50 or later. This is necessary for the working of **Octave-Gtk**binding binray installation.

For building **Octave-Gtk**though you need *Python* installed on your computer along with the above software, and their developement packages. i.e: development packages of GTK and Octave are also needed for building/hacking **Octave-Gtk**.

### 3.4 Communication Interface

Not Applicable.

## 4 System Features

## 5 System Features

This library aims to make GUI programming(with Octave language)a cake-walk.

### 5.1 System Features

Not Applicable

### 5.2 Stimulus/response sequences

Not Applicable

### **5.3 Functions requirements**

Not Applicable.

## **6 Other non-functional requirements**

Not Applicable.

### **6.1 Performance requirements**

Not Applicable.

### **6.2 Safety requirements**

Not Applicable.

### **6.3 Security requirements**

Not Applicable.

### **6.4 Software quality attribute**

The Library is under GPL and thus can be freely modified and redistributed.  
We have followed the cyclic software development method.

## **7 Other non-functional requirements**

Not Applicable.

### **7.1 Performance requirements**

Not Applicable.

### **7.2 Safety requirements**

Not Applicable.

### **7.3 Security requirements**

Not Applicable.

### **7.4 Software quality attribute**

The Library is under GPL and thus can be freely modified and redistributed.  
We have followed the cyclic software development method.

## 7.5 Business rules

**Octave-Gtk** is **GPL**'ed as it links with GNU Octave, which is **GPL**'ed itself. Business rules, are not applicable otherwise.

## 7.6 Special user requirements

Not Applicable.

## 7.7 Backup and recovery

Not Applicable.

## 7.8 Data migration

Migration from Matlab API is not supported.

## 7.9 Data retention

Not Applicable.

## 7.10 User training

Please read the examples from octave-gtk-demo code, that helps you start off easily.

1. GTK C API documentation & Tutorial
2. Octave-Gtk API FAQ
3. Octave-Gtk examples && '\$ octave-gtk-demo'

## 7.11 Installation

Installation from source is like this.

1. `./configure --prefix=/you/usr/directory`
2. `make`
3. `make install`

Installation with RPM is easier.

1. `rpm -iv octave-gtk-src-0.1.rpm # source install`
2. `rpm -iv octave-gtk-i586-src-0.1.rpm # binary install`

## 8 Other requirements

Not Applicable