



Installation Document

Open SensorWeb Architecture

OSWA

SensorWeb 3.1 Version

Tom Kobialka
Peng Deng
Ronald Siswara Wong
Jehanzeb Khan

May 11, 2009

Contents

1. Introduction	4
1.1. Purpose	4
1.2. How to Use This Document	4
1.3. References	4
1.4. Document Overview	4
2. Software Installation	5
2.1. Installing Intel Mote 2 Development Environment for TinyOS 1.1 (Rev 5.0)	5
2.2. Install JNI Library for getenv	22
2.3. Installing Ant	22
2.4. Installing Tomcat	22
2.5. Running with the MOTES connected	23
2.5.1. "Write Failed" Error while running Application with SerialForwarder:	24
2.6. Install and Running with SunSPOT	24
2.7. Java Build Errors	28
2.8. JAR files	28
2.9. Globus	28
2.10. TcpTunnelGui	28
2.11. SVN	29
2.12. Tomcat with eclipse edit	29
2.13. Logging	29
2.14. Documentation	29
3. Common Installation Problems	30
3.1. Errors when compiling TinyOS	30
3.2. Exception when trying to start up Tomcat	30
3.3. Failed to Initialize SensorCollectionService	32
3.4. Axis Jar Problem	33
4. Postgres Installation	35
4.1. Set up	35
4.1.1. Step 0 - Setting Up Cygserver	35
4.1.2. Step 1. Start Cygserver for Shared Memory Support	35
4.1.3. Step 2. Create a New Database Cluster	35
4.1.4. Step 3. Start up the postmaster	35
4.2. Command Notes	38
4.3. Installing JDBC Driver	38
4.4. Running Postgres	38
4.5. Shutting Down Postgres	38
5. Getting the Services Running	46
5.1. Before we start	46

5.2.	Configuring Eclipse	46
5.3.	Machine Specific Configuration	47
5.4.	Building the OSWA Project	48
5.5.	Error Encountered When Building the OSWA Project	49
5.6.	Building the OSWA Project - 2	50
5.7.	Adding Method to Existing WebServices	54

1. Introduction

1.1. Purpose

This document describes all the installation process for the OSWA project starting from scratch. This document covers all aspect of its installation including all various softwares needed, and how to deploy the web services in your machine.

1.2. How to Use This Document

When installing the OSWA project from scratch, readers are encouraged to follow the order specified below:

- Section 2 (Not all software are compulsory and readers are encouraged to check Section 3 when encounter some problems to see if they have already been documented).
- Section 4 (This section could be skipped but most service and routine in OSWA will not work without this).
- Section 5 (Compulsory)

1.3. References

- Various OSWA installation notes written by Tom Kobialka
- <http://www.postgresql.org/docs/7.4/interactive/index.html>
- http://www.postgresql.org/docs/faqs.FAQ_CYGWIN.html
- <http://archives.postgresql.org/pgsql-cygwin/2004-07/msg00150.php>
- <http://archives.postgresql.org/pgsql-cygwin/2004-07/msg00136.php>
- <http://www.geocities.com/cfflorendo/postgresql-cygwin.html>

1.4. Document Overview

The remainder of this document consists of the following sections:

- Section 2: Software Installation - All the various software that needs to be installed before the OSWA service can be installed and deployed.
- Section 3: Common Installation Problems - Describe some of the common problems encountered while following section 2 of this document.
- Section 4: Postgres Installation - Setting up the Postgres database to ensure it is ready to be used by the service.
- Section 5: Getting the Services Running - All the configuration needed to install and deploy the service in your machine.

2. Software Installation

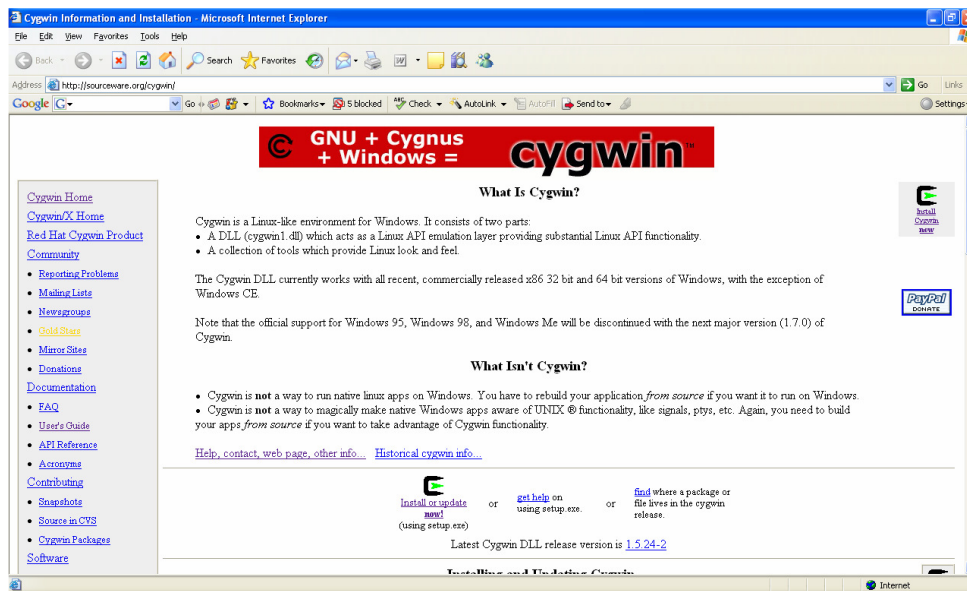
2.1. Installing Intel Mote 2 Development Environment for TinyOS 1.1 (Rev 5.0)

The Intel Mote 2 development environment requires Cygwin, Wasabi tool suite, xflash programming tool and TinyOS/Nesc. The Intel Mote 2 platform code is available

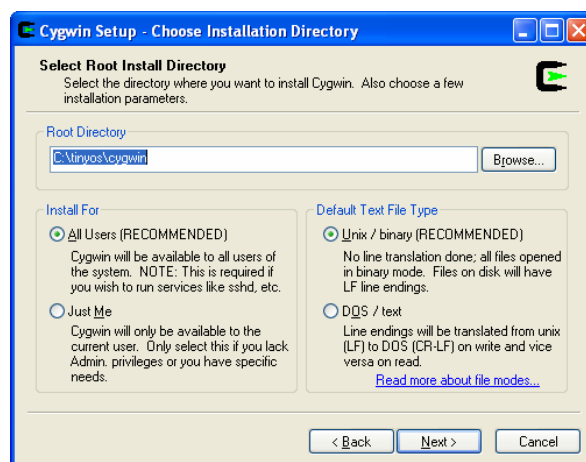
on sourceforge and can be downloaded from there. Please follow the detailed instructions below to install and configure the development environment.

Step 1 – Installing Cygwin

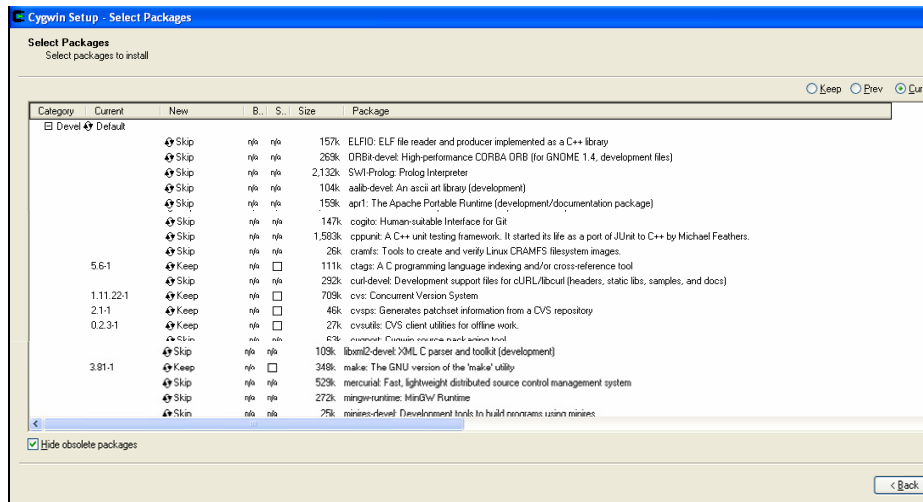
1. Obtain the latest cygwin version from <http://sourceware.org/cygwin/>.



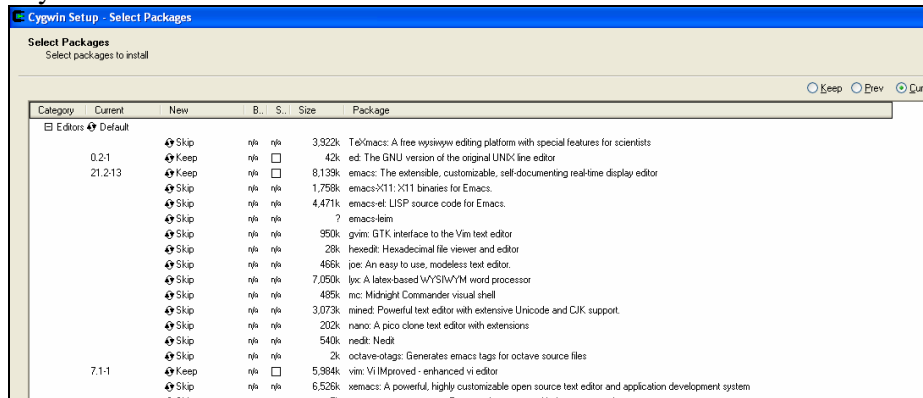
2. Install cygwin at C:\tinyos\cygwin
3. Select options for “install for all users” and “use unix file types”.



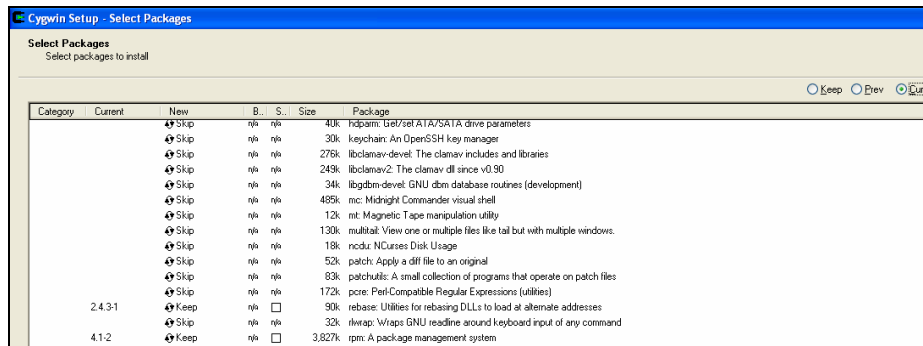
- By default, the Devel option doesn't include the CVS options. Make sure you go to the Devel list, and change the *cv*s, *cvsp*s and *cvutil* sublists from skip to install. Also, select *make* sublist because it will be needed in step 6, Test the Setup.



- You need to install an editor for editing files. I chose three different editors to try out.



- You will need to install an utility for executing rpm command that is used for installing NesC program in step 4 below.



- Cygwin will point your home directory to "C:\Documents and Settings\your_loginname". This is liable to create problems with other apps,

so change it to something like “/home/your_loginname”. This can be done by editing the /etc/passwd file, scrolling down to the line that starts with “your_loginname” and replacing the string “/cygdrive/c/Documents and Settings/your_loginname” by “/home/your_loginname”. Restart cygwin and check your home directory.

Here are the commands for editing passwd file with vi editor.

```

C /cygdrive/c/tinynos/cygwin/etc
Michael Zaremba@ISU_Robotics ~
$ cd c:
Michael Zaremba@ISU_Robotics /cygdrive/c
$ cd tinynos/cygwin/etc
Michael Zaremba@ISU_Robotics /cygdrive/c/tinynos/cygwin/etc
$ vi passwd_

```

Note the directory below is already correct. I believe the above step for changing passwd file is no longer necessary. Somebody must have installed a patch to automatically perform this home directory correction. To exit vi editor without saving changes, type “:q!” and depress enter key. To save changes and exit vi editor, type “:wq” and depress enter key.

```

C /cygdrive/c/tinynos/cygwin/etc
SYSTEM:*:18:544:.,S-1-5-18::
Administrators:*:544:544:.,S-1-5-32-544::
Administrator:unused_by_nt/2000/xp:500:513:U-ISU_ROBOTICS\Administrator,S-1-5-21-4202648761-318352513-1247360309-500:/home/Administrator:/bin/bash
Guest:unused_by_nt/2000/xp:501:513:U-ISU_ROBOTICS\Guest,S-1-5-21-4202648761-318352513-1247360309-501:/home/Guest:/bin/bash
HelpAssistant:unused_by_nt/2000/xp:1004:513:Remote Desktop Help Assistant Account,U-ISU_ROBOTICS\HelpAssistant,S-1-5-21-4202648761-318352513-1247360309-1004:/home/HelpAssistant:/bin/bash
LEGO:unused_by_nt/2000/xp:1007:513:LEGO,U-ISU_ROBOTICS\LEGO,S-1-5-21-4202640761-318352513-1247360309-1007:/home/LEGO:/bin/bash
Michael Zaremba:unused_by_nt/2000/xp:1005:513:U-ISU_ROBOTICS\Michael Zaremba,S-1-5-21-4202648761-318352513-1247360309-1005:/home/Michael Zaremba:/bin/bash
SUPPORT_388945a0:unused_by_nt/2000/xp:1002:513:CN=Microsoft Corporation,L=Redmond,S=Washington,C=US,U-ISU_ROBOTICS\SUPPORT_388945a0,S-1-5-21-4202648761-318352513-1247360309-1002:/home/SUPPORT_388945a0:/bin/bash
~
~
~
~
~
~
"passwd" 8 lines, 1019 characters

```

Step 2 – Downloading the TinyOS1.1 tree

1. If you already have an installation of the TinyOS1.1 tree from sourceforge, just update the tree and proceed to step 3. Note: The next screen shot shows directory of tinynos-1.x previously installed. I updated by running same code in next step.
2. To install the TinyOS1.1 tree from scratch, go to c:/ and type the following:
`cvs -z3 -d:pserver:anonymous@tinynos.cvs.sourceforge.net:/cvsroot/tinynos co -P tinynos-1.x`

If you have an account on sourceforge, replace anonymous with your login name and “pserver” with “ext”. When prompted for password enter your password (or no password for the anonymous account, just press Enter key)

Note this will take a few minutes to download.

```
Michael Zarenba@ISU_Robotics /cygdrive/c/tinyos-1.x
$ cvs -z3 -d:pserver:anonymous@tinyos.cvs.sourceforge.net:/cvsroot/tinyos co -P
tinyos-1.x
cvs checkout: warning: failed to open /home/Michael Zarenba/.cvspass for reading
: No such file or directory
cvs checkout: Updating tinyos-1.x
U tinyos-1.x/ChangeLog
U tinyos-1.x/INTEL-LICENSE.txt
U tinyos-1.x/Makefile
cvs checkout: Updating tinyos-1.x/apps
U tinyos-1.x/apps/Makefile
cvs checkout: Updating tinyos-1.x/tos/types
U tinyos-1.x/tos/types/AM.h
U tinyos-1.x/tos/types/ByteEEPROMInternal.h
U tinyos-1.x/tos/types/dbg.h
U tinyos-1.x/tos/types/dbg_nodes.h
U tinyos-1.x/tos/types/list.h

Michael Zarenba@ISU_Robotics /cygdrive/c/tinyos-1.x
$
```

Step 3 – Configure TinyOS1.1 tree

1. The Intel Mote 2 platform files are split into a pxa27x and imote2 subdirectories and can be found in tinyos1.x/beta/platform/. Create a link to each of these directories under tinyos1.x/tos/platform
 - a) Go to tinyos-1.x/tos/platform
 - b) Note space between “s ..”. Type: `ln -s ../../beta/platform/imote2 imote2`
 - c) Type: `ln -s ../../beta/platform/pxa27x pxa27x`

Screen shot shows platform’s directories before and after linking.


```

C /cygdrive/c/tinyos-1.x/tos/platform
Michael Zarenba@ISU_Robotics ~
$ cd c:/tinyos-1.x/tos/platform

Michael Zarenba@ISU_Robotics /cygdrive/c/tinyos-1.x/tos/platform
$ ls -la
total 0
drwxr-xr-x+ 15 Michael Zarenba None 0 Jul 25 13:55 .
drwxr-xr-x+ 9 Michael Zarenba None 0 Jul 6 15:36 ..
drwxr-xr-x+ 2 Michael Zarenba None 0 Jul 16 17:24 CUS
drwxr-xr-x+ 3 Michael Zarenba None 0 Jul 6 15:35 atmega8
drwxr-xr-x+ 3 Michael Zarenba None 0 Jul 16 19:26 avrnote
drwxr-xr-x+ 4 Michael Zarenba None 0 Jul 6 15:35 mica
drwxr-xr-x+ 3 Michael Zarenba None 0 Jul 6 15:35 mica128
drwxr-xr-x+ 4 Michael Zarenba None 0 Jul 6 15:35 mica2
drwxr-xr-x+ 3 Michael Zarenba None 0 Jul 6 15:35 mica2dot
drwxr-xr-x+ 3 Michael Zarenba None 0 Jul 6 15:35 micaz
drwxr-xr-x+ 3 Michael Zarenba None 0 Jul 16 17:24 msp430
drwxr-xr-x+ 6 Michael Zarenba None 0 Jul 6 15:36 pc
drwxr-xr-x+ 3 Michael Zarenba None 0 Jul 6 15:36 rene2
drwxr-xr-x+ 3 Michael Zarenba None 0 Jul 6 15:36 telos
drwxr-xr-x+ 3 Michael Zarenba None 0 Jul 6 15:36 telosb

Michael Zarenba@ISU_Robotics /cygdrive/c/tinyos-1.x/tos/platform
$ ln -s ../../beta/platform/imote2 imote2

Michael Zarenba@ISU_Robotics /cygdrive/c/tinyos-1.x/tos/platform
$ ln -s ../../beta/platform/pxa27x pxa27x

Michael Zarenba@ISU_Robotics /cygdrive/c/tinyos-1.x/tos/platform
$ ls -la
total 2
drwxr-xr-x+ 15 Michael Zarenba None 0 Jul 25 13:59 .
drwxr-xr-x+ 9 Michael Zarenba None 0 Jul 6 15:36 ..
drwxr-xr-x+ 2 Michael Zarenba None 0 Jul 16 17:24 CUS
drwxr-xr-x+ 3 Michael Zarenba None 0 Jul 6 15:35 atmega8
drwxr-xr-x+ 3 Michael Zarenba None 0 Jul 16 19:26 avrnote
lrwxrwxrwx 1 Michael Zarenba None 26 Jul 25 13:58 imote2 -> ../../beta/platform/imote2
drwxr-xr-x+ 4 Michael Zarenba None 0 Jul 6 15:35 mica
drwxr-xr-x+ 3 Michael Zarenba None 0 Jul 6 15:35 mica128
drwxr-xr-x+ 4 Michael Zarenba None 0 Jul 6 15:35 mica2
drwxr-xr-x+ 3 Michael Zarenba None 0 Jul 6 15:35 mica2dot
drwxr-xr-x+ 3 Michael Zarenba None 0 Jul 6 15:35 micaz
drwxr-xr-x+ 3 Michael Zarenba None 0 Jul 16 17:24 msp430
drwxr-xr-x+ 6 Michael Zarenba None 0 Jul 6 15:36 pc
lrwxrwxrwx 1 Michael Zarenba None 26 Jul 25 13:59 pxa27x -> ../../beta/platform/pxa27x
drwxr-xr-x+ 3 Michael Zarenba None 0 Jul 6 15:36 rene2
drwxr-xr-x+ 3 Michael Zarenba None 0 Jul 6 15:36 telos
drwxr-xr-x+ 3 Michael Zarenba None 0 Jul 6 15:36 telosb

```

2. Setup the TOSDIR and TOSROOT variables in your .profile.

- a) If you installed the tinyos tree in /cygdrive/c/tinyos-1.x, add the following lines to your .profile file:


```
export TOSROOT=/cygdrive/c/tinyos-1.x
export TOSDIR=$TOSROOT/tos
```

Note: If you have installed the tinyos tree in a different location, replace /cygdrive/c/tinyos-1.x with the path of your installed tree.

- b) Set the new make utility to be your makefile by adding the following line to your .profile:


```
export MAKERULES=$TOSROOT/tools/make/Makerules
```

Screen shot of .bash_profile before editing.

```
Michael Zarenba@ISU_Robotics ~
$ cd c:/

Michael Zarenba@ISU_Robotics /cygdrive/c
$ cd tinyos/cygwin/home/'Michael Zarenba'

Michael Zarenba@ISU_Robotics /cygdrive/c/tinyos/cygwin/home/Michael Zarenba
$ ls -la
total 32
drwxrwxrwx+ 3 Michael Zarenba None    0 Jul 25 15:07 .
drwxrwxrwx+ 3 Michael Zarenba None    0 Jul  6 14:21 ..
-rw----- 1 Michael Zarenba None  3305 Jul 17 16:37 .bash_history
-rwxr-xr-x  1 Michael Zarenba None  1150 Jul 25 15:07 .bash_profile
-rwxr-xr-x  1 Michael Zarenba None  3116 Jul  6 14:13 .bashrc
drwxr-xr-x+ 3 Michael Zarenba None    0 Jul 17 16:45 .emacs.d
-rw----- 1 Michael Zarenba None 12200 Jul 11 09:59 .foo.c.swp
-rwxr-xr-x  1 Michael Zarenba None  1461 Jul  6 14:13 .inputrc
-rw----- 1 Michael Zarenba None  1611 Jul 25 14:34 .viminfo

Michael Zarenba@ISU_Robotics /cygdrive/c/tinyos/cygwin/home/Michael Zarenba
$ cat .bash_profile
# base-files version 3.7-1

# To pick up the latest recommended .bash_profile content,
# look in /etc/defaults/etc/skel/.bash_profile

# Modifying /etc/skel/.bash_profile directly will prevent
# setup from updating it.

# The copy in your home directory (~/.bash_profile) is yours, please
# feel free to customise it to create a shell
# environment to your liking.  If you feel a change
# would be beneficial to all, please feel free to send
# a patch to the cygwin mailing list.

# ~/.bash_profile: executed by bash for login shells.

# source the system wide bashrc if it exists
if [ -e /etc/bash.bashrc ] ; then
    source /etc/bash.bashrc
fi

# source the users bashrc if it exists
if [ -e "${HOME}/.bashrc" ] ; then
    source "${HOME}/.bashrc"
fi

# Set PATH so it includes user's private bin if it exists
# if [ -d "${HOME}/bin" ] ; then
#   PATH=${HOME}/bin:${CPATH}
# fi

# Set MANPATH so it includes users' private man if it exists
# if [ -d "${HOME}/man" ] ; then
#   MANPATH=${HOME}/man:${MANPATH}
# fi

# Set INFOPATH so it includes users' private info if it exists
# if [ -d "${HOME}/info" ] ; then
#   INFOPATH=${HOME}/info:${INFOPATH}
# fi
$
```

To edit .bash_profile with vi editor, use the below code:

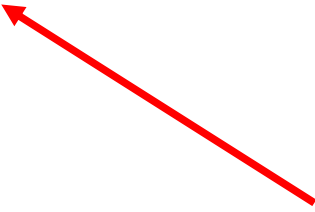
```
Michael Zarembo@ISU_Robotics /cygdrive/c/tinyos/cygwin/home/Michael Zarembo
$ vi .bash_profile_
```

Screen shot showing vi editor with three added lines of code. Note: To exit and save changes while using vi editor, type “:wq” and depress enter key. After exiting vi editor, you must restart cygwin for changes in .bash_profile to take effect.

```

/cygdrive/c/tinyos/cygwin/home/Michael Zarembo
# base-files version 3.7-1
# To pick up the latest recommended .bash_profile content,
# look in /etc/defaults/etc/skel/.bash_profile
# Modifying /etc/skel/.bash_profile directly will prevent
# setup from updating it.
# The copy in your home directory (~/.bash_profile) is yours, please
# feel free to customise it to create a shell
# environment to your liking.  If you feel a change
# would be beneficial to all, please feel free to send
# a patch to the cygwin mailing list.
# ~/.bash_profile: executed by bash for login shells.
# source the system wide bashrc if it exists
if [ -e /etc/bash.bashrc ] ; then
source /etc/bash.bashrc
fi
# source the users bashrc if it exists
if [ -e "$HOME/.bashrc" ] ; then
source "$HOME/.bashrc"
fi
# Set PATH so it includes user's private bin if it exists
# if [ -d "$HOME/bin" ] ; then
# PATH=$HOME/bin:$PATH
# fi
# Set MANPATH so it includes users' private man if it exists
# if [ -d "$HOME/man" ] ; then
# MANPATH=$HOME/man:$MANPATH
# fi
# Set INFOPATH so it includes users' private info if it exists
# if [ -d "$HOME/info" ] ; then
# INFOPATH=$HOME/info:$INFOPATH
# fi
export TOSROOT=/cygdrive/c/tinyos-1.x
export TOSDIR=$TOSROOT/tos
export MAKERULES=$TOSROOT/tools/make/Makerules
~
~
~
~
~
~
~
~
~
~
END

```



Step 4 – Install NesC

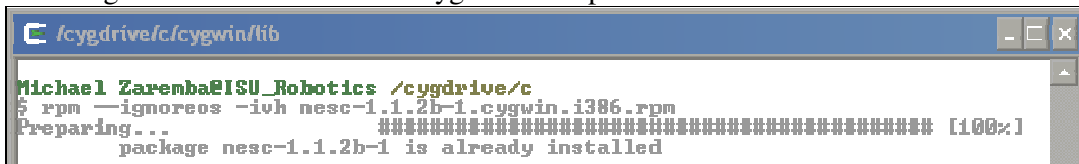
1. Download NesC from the TinyOS website:

<http://www.tinyos.net/dist-1.1.0/tinyos/windows> . Get the latest version nesc-1.xxxx.cygwin.i386.rpm and load to a directory of your choice.



Name	Last modified	Size	Description
Parent Directory	26-Sep-2005 10:07	-	
galsc-0.1.0-1.cygwin.i386.rpm	28-Apr-2004 16:30	1.0M	
micaz-installer/	16-Jun-2004 13:32	-	
mspgcc-win32tinyos-20041204-1.cygwin.i386.rpm	06-Dec-2004 15:16	12.5M	
nesc-1.1-1w.cygwin.i386.rpm	29-Sep-2003 11:47	992k	
nesc-1.1.1-2.cygwin.i386.rpm	31-Mar-2004 16:14	1006k	
nesc-1.1.2a-1.cygwin.i386.rpm	30-Jul-2004 16:35	1.5M	
nesc-1.1.2b-1.cygwin.i386.rpm	14-Mar-2005 18:27	965k	
old/	06-Dec-2004 22:39	-	

2. Using cygwin, navigate to your chosen directory and install using “rpm --ignoreos -ivh nesc-1.xxxx.cygwin.i386.rpm”.



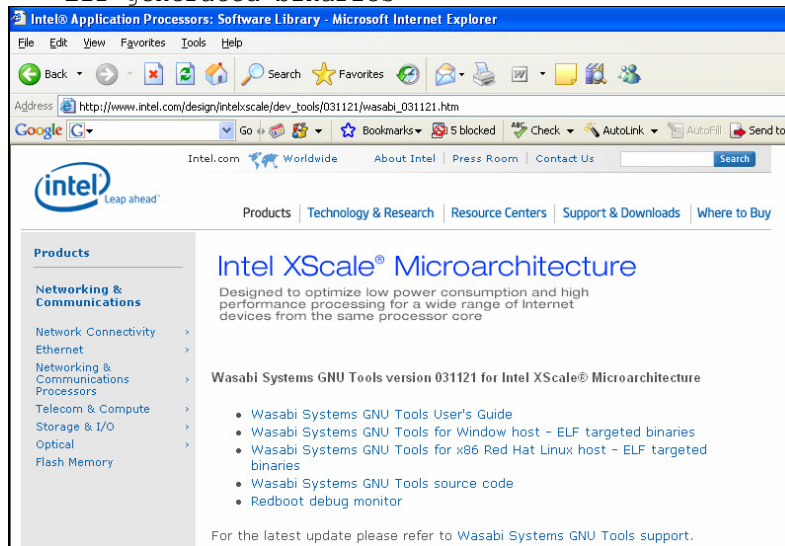
```
Michael Zaremba@ISU_Robotics /cygdrive/c
$ rpm --ignoreos -ivh nesc-1.1.2b-1.cygwin.i386.rpm
Preparing...
package nesc-1.1.2b-1 is already installed
```

Step 5 – Install the Wasabi tool suite

1. The Wasabi toolsuite can be downloaded from

http://www.intel.com/design/intelxscale/dev_tools/031121/

- a) Accept the license terms
- b) Choose the Wasabi Systems Gnu Tools for Windows Host - Elf generated binaries



- c) Install the Wasabi tools to c:/wasabi
- d) Add /cygdrive/c/wasabi/usr/local/bin to the beginning of your path by adding the following to your .profile file:

```
PATH=/cygdrive/c/wasabi/usr/local/bin:$PATH
```

```
C /cygdrive/c/tinyos/cygwin/home/Michael Zaremba
Michael Zaremba@ISU_Robotics ~
$ cd c:/
Michael Zaremba@ISU_Robotics /cygdrive/c
$ cd tinyos/cygwin/home/'Michael Zaremba'
Michael Zaremba@ISU_Robotics /cygdrive/c/tinyos/cygwin/home/Michael Zaremba
$ vi .bash_profile
Michael Zaremba@ISU_Robotics /cygdrive/c/tinyos/cygwin/home/Michael Zaremba
$ cat .bash_profile
# base-files version 3.7-1

# To pick up the latest recommended .bash_profile content,
# look in /etc/defaults/etc/skel/.bash_profile

# Modifying /etc/skel/.bash_profile directly will prevent
# setup from updating it.

# The copy in your home directory (~/.bash_profile) is yours, please
# feel free to customise it to create a shell
# environment to your liking.  If you feel a change
# would be beneficial to all, please feel free to send
# a patch to the cygwin mailing list.

# ~/.bash_profile: executed by bash for login shells.

# source the system wide bashrc if it exists
if [ -e /etc/bash.bashrc ] ; then
source /etc/bash.bashrc
fi

# source the users bashrc if it exists
if [ -e "${HOME}/.bashrc" ] ; then
source "${HOME}/.bashrc"
fi

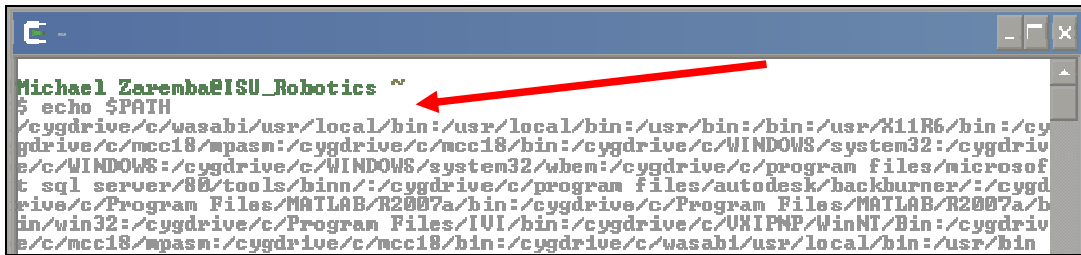
# Set PATH so it includes user's private bin if it exists
# if [ -d "${HOME}/bin" ] ; then
#   PATH=${HOME}/bin:${PATH}
# fi

# Set MANPATH so it includes users' private man if it exists
# if [ -d "${HOME}/man" ] ; then
#   MANPATH=${HOME}/man:${MANPATH}
# fi

# Set INFOPATH so it includes users' private info if it exists
# if [ -d "${HOME}/info" ] ; then
#   INFOPATH=${HOME}/info:${INFOPATH}
# fi

export TOSROOT=/cygdrive/c/tinyos-1.x
export TOSDIR=${TOSROOT}/tos
export MAKERULES=${TOSROOT}/tools/make/Makerules
PATH=/cygdrive/c/wasabi/usr/local/bin:${PATH}
```

- e) Don't set GCC_EXEC_PREFIX like the user guide says. Restart cygwin to make sure the path change has taken effect.



```
Michael Zarenba@ISU_Robotics ~  
$ echo $PATH  
/cygdrive/c/wasabi/usr/local/bin:/usr/local/bin:/usr/bin:/bin:/usr/X11R6/bin:/cygdrive/c/mcc18/wpasn:/cygdrive/c/mcc18/bin:/cygdrive/c/WINDOWS/system32:/cygdrive/c/WINDOWS:/cygdrive/c/WINDOWS/system32/wbem:/cygdrive/c/program files/microsoft sql server/80/tools/binn:/cygdrive/c/program files/autodesk/backburner:/cygdrive/c/Program Files/MATLAB/R2007a/bin:/cygdrive/c/Program Files/MATLAB/R2007a/bin/win32:/cygdrive/c/Program Files/LUI/bin:/cygdrive/c/UNIXPNP/WinNT/Bin:/cygdrive/c/mcc18/wpasn:/cygdrive/c/mcc18/bin:/cygdrive/c/wasabi/usr/local/bin:/usr/bin
```

- f) Make sure that you can execute xscale-elf-gcc. To do this refer to Wasabi_Xscale_Users_Guide1.pdf file and follow step 3 of XP Installation section. Step shown below. You can use a C-editor or text editor like notepad. Save your file to the C drive. Note: Insert a CD into CD drive or you will get an error about not being able to read your CD drive. This is a known bug.

3. A simple test to verify correct installation is to compile this canonical "hello world" program and then run it on the simulator

a. Create the text file, foo.c:

```
int main()  
{  
    printf ("Hello world\n");  
    return 0;  
}
```

b. Compile the program:

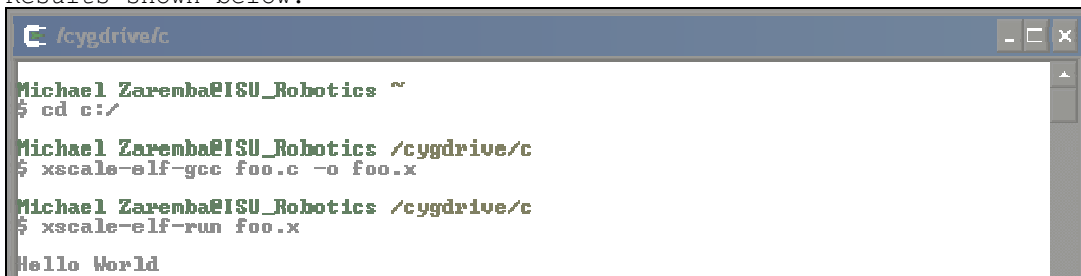
```
xscale-elf-gcc foo.c -o foo.x
```

c. Run the executable foo.x on the simulator:

```
xscale-elf-run foo.x
```

If the toolchain is properly installed, you should see "Hello world" on your console as a result of executing this program on the simulator.

Results shown below:



```
/cygdrive/c  
Michael Zarenba@ISU_Robotics ~  
$ cd c:/  
Michael Zarenba@ISU_Robotics /cygdrive/c  
$ xscale-elf-gcc foo.c -o foo.x  
Michael Zarenba@ISU_Robotics /cygdrive/c  
$ xscale-elf-run foo.x  
Hello World
```

Step 6 – Test the setup

1. Go to \$TOSROOT/apps/blink
2. Type : make imote2 debug. It should complete but complain about dwarf2bd. If it doesn't complete and gives you an error message about an e drive : "cc1: /cygdrive/e/wasabi_drops/...../include: No medium found", insert a CD in your e drive and try again. This is a known issue that we are trying to debug currently. If it doesn't complete but you don't get the error about the e drive, check your environment variables.

Text outputted to cygwin terminal while executing 'make imote2 debug':

```
mkdir -p build/imote2
xscale-elf-as -mcpu=iwmmxt -mfpu=softfpa -defsym BOOTLOADER=1
/cygdrive/c/tinyos-1.x/tos/platform/pxa27x/./imote2/flash.s /cygdrive/c/tinyos-
1.x/tos/platform/pxa27x/./imote2/binarymover.s /cygdrive/c/tinyos-
1.x/tos/platform/pxa27x/barecrt.s /cygdrive/c/tinyos-
1.x/tos/platform/pxa27x/mmu_table.s /cygdrive/c/tinyos-
1.x/tos/platform/pxa27x/util.s -o build/imote2/asms.o
cd /cygdrive/c/tinyos-1.x/tos/platform/pxa27x/lib; make;
make[1]: Entering directory `/cygdrive/c/tinyos-1.x/beta/platform/pxa27x/lib'
xscale-elf-gcc -g -O2 -Wall -I/cygdrive/c/tinyos-1.x/tos/platform/pxa27x -
I/cygdrive/c/tinyos-1.x/tos/platform/pxa27x/lib/DSP -I/cygdrive/c/tinyos-
1.x/tos/platform/pxa27x/lib/ -c -o bufferManagement.o bufferManagement.c
xscale-elf-gcc -g -O2 -Wall -I/cygdrive/c/tinyos-1.x/tos/platform/pxa27x -
I/cygdrive/c/tinyos-1.x/tos/platform/pxa27x/lib/DSP -I/cygdrive/c/tinyos-
1.x/tos/platform/pxa27x/lib/ -c -o downsample.o downsample.c
xscale-elf-gcc -g -O2 -Wall -I/cygdrive/c/tinyos-1.x/tos/platform/pxa27x -
I/cygdrive/c/tinyos-1.x/tos/platform/pxa27x/lib/DSP -I/cygdrive/c/tinyos-
1.x/tos/platform/pxa27x/lib/ -c -o frequency.o frequency.c
xscale-elf-gcc -g -O2 -Wall -I/cygdrive/c/tinyos-1.x/tos/platform/pxa27x -
I/cygdrive/c/tinyos-1.x/tos/platform/pxa27x/lib/DSP -I/cygdrive/c/tinyos-
1.x/tos/platform/pxa27x/lib/ -c -o paramtask.o paramtask.c
xscale-elf-gcc -g -O2 -Wall -I/cygdrive/c/tinyos-1.x/tos/platform/pxa27x -
I/cygdrive/c/tinyos-1.x/tos/platform/pxa27x/lib/DSP -I/cygdrive/c/tinyos-
1.x/tos/platform/pxa27x/lib/ -c -o profile.o profile.c
xscale-elf-gcc -g -O2 -Wall -I/cygdrive/c/tinyos-1.x/tos/platform/pxa27x -
I/cygdrive/c/tinyos-1.x/tos/platform/pxa27x/lib/DSP -I/cygdrive/c/tinyos-
1.x/tos/platform/pxa27x/lib/ -c -o queue.o queue.c
xscale-elf-gcc -g -O2 -Wall -I/cygdrive/c/tinyos-1.x/tos/platform/pxa27x -
I/cygdrive/c/tinyos-1.x/tos/platform/pxa27x/lib/DSP -I/cygdrive/c/tinyos-
1.x/tos/platform/pxa27x/lib/ -c -o systemUtil.o systemUtil.c
xscale-elf-gcc -g -O2 -Wall -I/cygdrive/c/tinyos-1.x/tos/platform/pxa27x -
I/cygdrive/c/tinyos-1.x/tos/platform/pxa27x/lib/DSP -I/cygdrive/c/tinyos-
1.x/tos/platform/pxa27x/lib/ -c -o utils.o utils.c
xscale-elf-gcc -g -O2 -Wall -I/cygdrive/c/tinyos-1.x/tos/platform/pxa27x -
I/cygdrive/c/tinyos-1.x/tos/platform/pxa27x/lib/DSP -I/cygdrive/c/tinyos-
1.x/tos/platform/pxa27x/lib/ -c -o wmmx.o wmmx.c
xscale-elf-ar -rvs libimote2.a bufferManagement.o downsample.o frequency.o
paramtask.o profile.o queue.o systemUtil.o utils.o wmmx.o
r - bufferManagement.o
r - downsample.o
```

```

r - frequency.o
r - paramtask.o
r - profile.o
r - queue.o
r - systemUtil.o
r - utils.o
r - wmmx.o
make[1]: Leaving directory `~/cygdrive/c/tinyos-1.x/beta/platform/pxa27x/lib'
  compiling Blink to a imote2 binary
ncc -o build/imote2/main.exe -g -fnesc-no-inline -I/cygdrive/c/tinyos-
1.x/tos/lib/CC2420Radio -I/cygdrive/c/tinyos-1.x/tos/lib/Flash -
DAUTO_BATTERY_MONITORING=1 -Wall -Wshadow -
DDEF_TOS_AM_GROUP=0x7d -Wnesc-all -target=imote2 -fnesc-
cfile=build/imote2/app.c -board= -DBOOTLOADER -
DIDENT_PROGRAM_NAME="Blink\" -
DIDENT_USER_ID="MichaelZaremba\" -
DIDENT_HOSTNAME="ISU_Robotics\" -
DIDENT_USER_HASH=0xe8f84054L -DIDENT_UNIX_TIME=0x46af6ce8L -
DIDENT_UID_HASH=0x5d90bbfdL Blink.nc -lm build/imote2/asms.o
~/cygdrive/c/tinyos-1.x/tos/platform/pxa27x/lib/libimote2.a
  compiled Blink to build/imote2/main.exe
    0 bytes in ROM
    0 bytes in RAM
    0 bytes in STACK
    262144 bytes available in HEAP
xscale-elf-objcopy --output-target=binary build/imote2/main.exe
build/imote2/main.bin.out
dwarf2bd -nc build/imote2/main.exe
make: dwarf2bd: Command not found
make: *** [bin] Error 127

```

Note: Skip step 7, 8, 9 and 10 if using USB for programming. These steps are for xflash programming with JTAG cable.

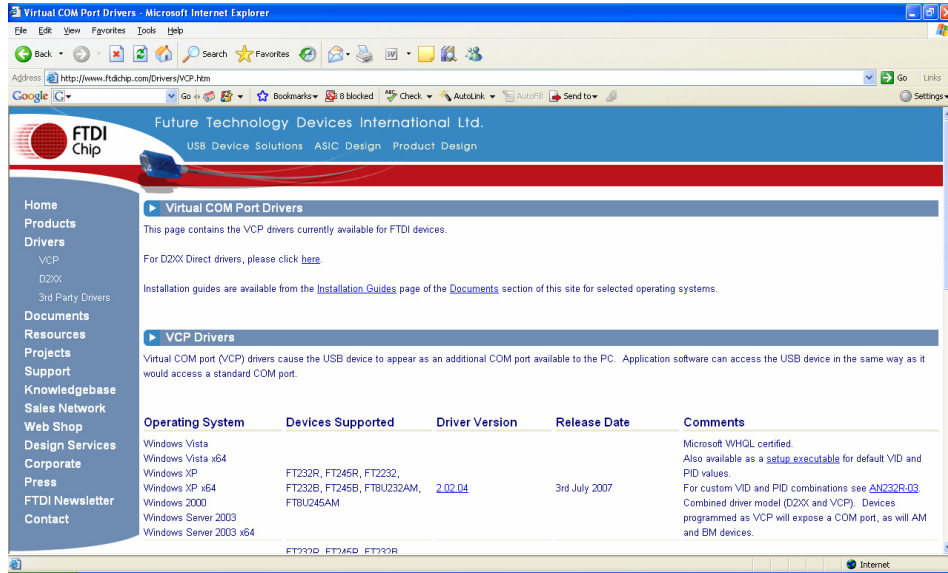
Skip step 7.

Step 7 – Test the setup again

- Go to \$TOSROOT/apps/blink
- Type : make imote2 debug
- It should complete.
 - If you get a dwarf2bd error, you probably didn't restart your machine after installing the SDT tools. Try restarting your machine and repeat the step above. Note: dwarf2bd.exe is installed in c:\nordheim\xdbbin
 - You can check if the path is setup correctly by typing which dwarf2bd at the cygwin prompt, you should get /cygdrive/c/nordheim/xdbbin/dwarf2bd

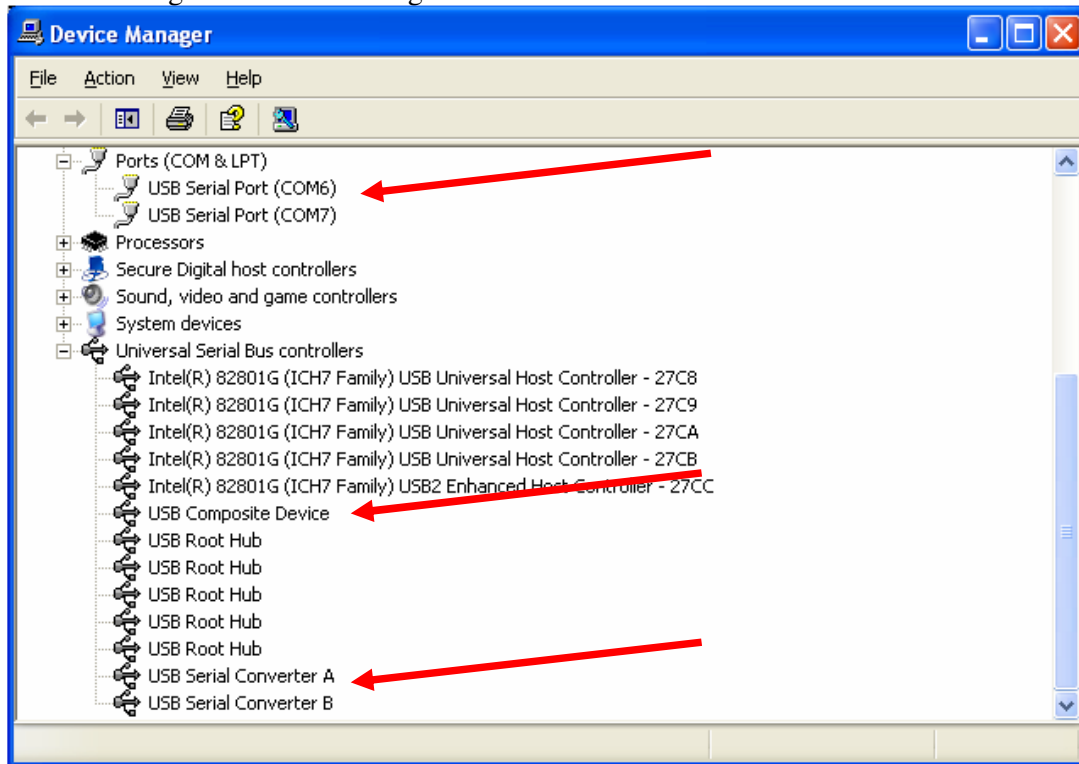
Step 8 – Setup Intel Mote 2 debug board drivers

- The Intel Mote 2 debug/programming board contains a dual port FTDI chip to map two of the mote UART ports to USB on your pc.
- Before connecting Intel Mote 2 debug board to your pc, download the drivers from <http://www.ftdichip.com/Drivers/VCP.htm> (for Windows XP). Current version is 2.02.04.



- Unzip the file to a folder.
- Connect the debug board to your computer via a USB cable. When prompted for a driver, follow the installation instructions, specify the folder that you just unzipped to look for a driver.

Device Manager results of loading drivers:



Skip step 9

Step 9 – Copy required files and setup the environment

- Copy imote2.fcf and imote2.xsf from \$TOSROOT/contrib/imote2/installation to

c:\imote2

- Copy imote2.xdb from \$TOSROOT/contrib/imote2/installation to C:\nordheim\xflash\imote2.xdb (replace c:\norheim with your toolsuite location if different)
- Copy boards.ini from \$TOSROOT/contrib/imote2/installation to c:\nordheim\xflash\boards.ini (overwrite existing file).
- If you have the Intel JTAG dongle, add the following line to your .profile file :
 - export PXA27X_JTAG_DEV=""INTEL(R) JTAG CABLE""Note, the INTEL JTAG cable might complain unless your PC has been hard power cycled with its parallel port in ECP mode at base 0x378
- if you have a Macraigor Raven JTAG dongle, you don't need to do anything.

Step 10 – Test the setup

- Restart Cygwin
- Plug in an Intel Mote 2 into the advanced connector of the debug board (J1 & J3).
- Push the power button on Intel Mote 2 (little button above the processor that should be on the top side of the board when the mote is attached to the advanced connectors).
- Go to `tinys-1.x/apps/Blink` and type: `make install imote2 debug`.
- This will build, link and download the blink application to the Intel Mote 2.
- After the download is done, the mote should be blinking red at 2Hz (50% duty cycle).

Step 11 – Load file to Intel Mote 2 using USBLoaderHost

1. Connect USB cable to Intel Mote 2.
2. Go to `$TOSROOT/contrib/imote2/tools/bin` directory.
3. Type:
`./USBLoaderHost.exe -p $TOSROOT/apps/Blink/build/imote2/main.bin.out`.

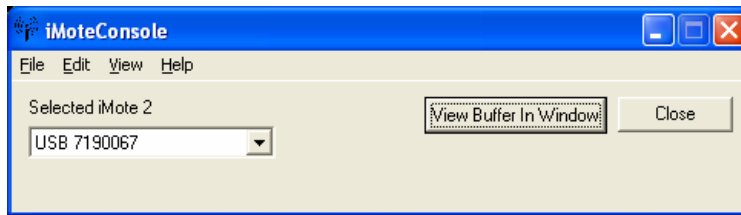
After program loads the led should flash red color.

Step 12 – Test the Intel Mote 2 debug shell

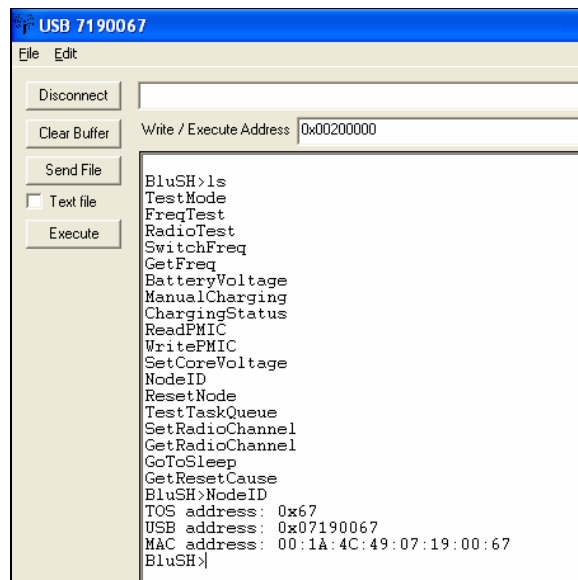
The Intel Mote 2 has a command shell that gets linked in by default to all applications. This enables sending debug messages to a console, or invoking commands in the mote from the console. The debug shell can be connected to either the USB port on the mote (directly connecting the Intel Mote 2 to USB not to the debug board), or through the UART (connecting the mote to the PC through the debug board, hence going through the USB->UART translation). The two approaches are mentioned below:

1. Debug through USB:
 - a. Connect the USB cable to the Intel Mote 2 board directly (J2) on one side and to the PC on the other side.
 - b. Push the power button of the Intel Mote 2. The mote will be recognized as a USB human interface device.
 - c. Use the `ImoteConsole` program to communicate with the mote, this can be found in `$TOSROOT/contrib/imote2/tools/bin` directory.
 - i. You may have to do this first:
`chmod a+x $TOSROOT/contrib/imote2/tools/bin/iMoteConsole.exe`
 - d. Start `iMoteConsole` by entering `./iMoteConsole.exe`.

- e. Select the Imote2 from the drop down menu, the name of the device will be “USB ID”, where ID is the least significant 4 digits of the serial number shown on the back of the Intel Mote 2 board.

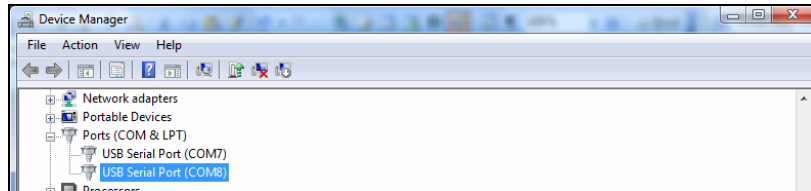


- f. Click on “View Buffer In Window” option. This will start a new window, push the Connect button. Go to the large box (click anywhere inside the box), and push the enter button, you will see a “BluSH>” prompt.
- g. Type “ls” to see available commands. Type NodeID, the mote will print out the serial number of the mote.

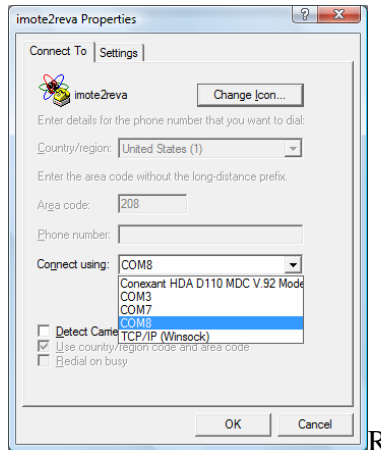


2. Debug through UART:

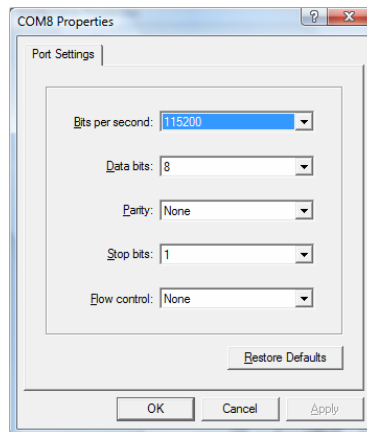
- a. Connect the Intel Mote 2 to the programming board. If you installed the programming board successfully in step 9 above, it will create two “USB serial port” instances, you can find which COM ports were mapped by going to control panel, system, hardware, device manager. The ports will be listed under “Ports (COM & LPT)”. If you don’t see these listed, please go back and check Step 9 above.



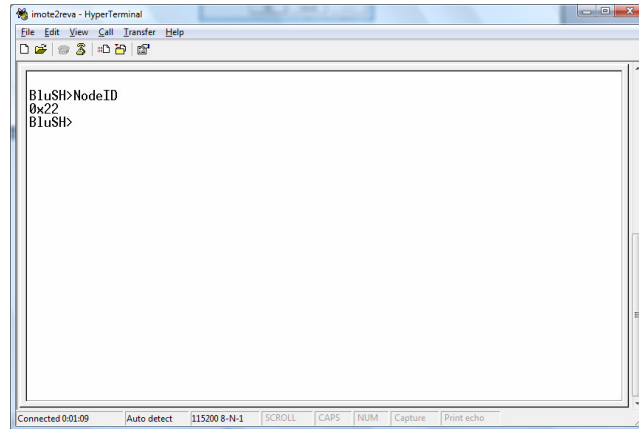
- b. The second USB serial port listed is the one dedicated to the console.



- c. Start hyperterm, choose the second com port listed and configure it to 115200, 8, none, 1, none.



- d. Push the connect button. Push the enter button, you will see a “BluSH>” prompt. Type NodeID, the mote will print out the serial number of the mote.



2.2. Install JNI Library for getenv

To install JNI Library, execute:

```
> cd $TOSROOT/tools/java/jni
> make install
```

2.3. Installing Ant

Download Ant from <http://ant.apache.org/>

To install Ant, simply follow the instructions provided at <http://ant.apache.org/manual/index.html>

Make sure ANT_HOME is set prior to building and don't forget to add \$ANT_HOME/bin to your \$PATH

2.4. Installing Tomcat

- Download and install Tomcat (Windows Executable version) from <http://tomcat.apache.org/>
at the time of writing this document, the version of the Tomcat I'm using is 5.5.20
- Do not use Tomcat Version 6.0 as it is missing the values for 'server.loader' and 'shared.loader' in catalina.properties file.
- Additional installation instructions are available at <http://tomcat.apache.org/tomcat-5.5-doc/setup.html>
- Set JAVA_HOME environment variable to point to the pathname of the directory where the JRE is located. In my machine, the JAVA_HOME is set to C:\Program Files\Java\jdk1.5.0_02. Your JAVA_HOME may be different depending on where you installed your JDK
Note that the installer will try to determine this package by default using the registry but ensure manually that the location is correct for your machine

For the purposes of the remainder of this document, the symbolic name "\$CATALINA_HOME" is used to refer to the full pathname of the release directory. As an example, in my computer, the symbol \$CATALINA_HOME in my computer represents C:\Program Files\Apache Software Foundation\apache-tomcat-5.5.20

NOTE: *CATALINA_HOME* represents the root of your tomcat installation.

- Add to /etc/bash.bashrc:

```
> export CATALINA_HOME="/cygdrive/c/Program\ Files/Apache\
Software\ Foundation/Tomcat\ 5.5/"
```

- To start up Tomcat in Windows simply execute the file \$CATALINA_HOME/bin/startup.bat
- To shut it down, simply execute the file \$CATALINA_HOME/bin/shutdown.bat
- Visit <http://localhost:8080/> in you web browser to ensure tomcat is working

2.5. Running with the MOTES connected

- Edit application.properties under tomcat/webapps/SensorWeb/classes/WEB-INF
uncomment all the lines that start with "REAL"
- Go to c:/Surge-View/
execute SerialForwarder.exe
set: serail@COM1:57600
- Ensure the motes and bass station are connected to the serial port
- Restart the server
- Open cmd.exe and go to Surge-View directory
- Execute:

```
> Surge 125 > surgeout.out
```

- Go to Sensor-Source directory and Execute

```
> ant run
```

- Select the second tab and press "Get Observation"

2.5.1. "Write Failed" Error while running Application with SerialForwarder:

- In case of "write failed" error, you will see a constant red light on your board as well as notes. First make sure that the error is not in your code by going through the tomcat trace logs. If the error is not in the code then you either have to reset your board.

2.6. Install and Running with SunSPOT

2.6.1 Install NetBeans and J2SE

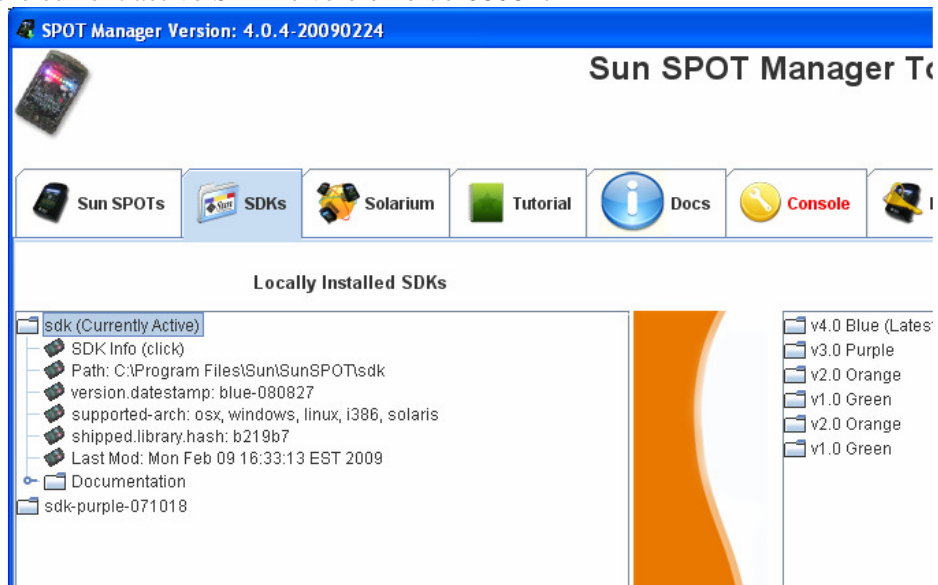
The current running system using NetBeans 6.5 and J2SE 1.6.11

2.6.2 Install SPOTManager from <http://www.sunspotworld.com/SPOTManager/>

It will automatically install ANT and NetBeans plugin

2.6.3 Please use SunSPOT SDK Version Blue (4.0), and ensure your SunSPOTs and SunSPOT basestation have the version Blue firmware running

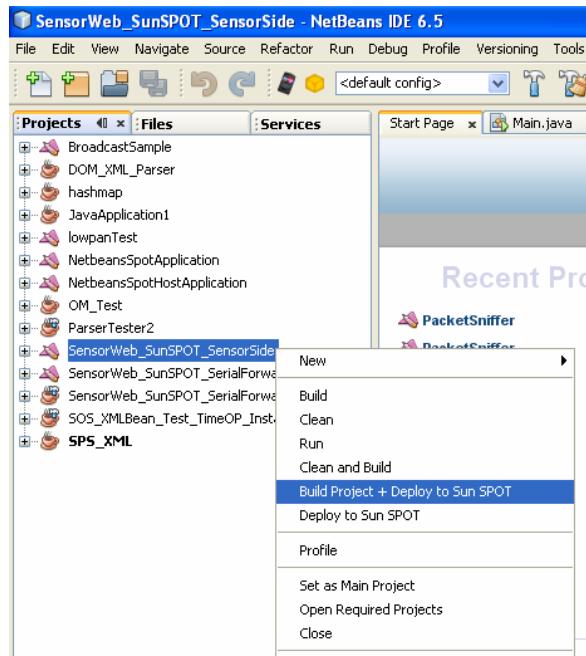
To check you have right version of SDK on your PC, run SPOTManager. You should see the current active SDK is version blue-080827



To check SunSPOTs and basestation have the right firmware version, please connect SunSPOT to USB port. Then, in the SPOTManager, under the "Sun SPOTs" tab, click SPOT info button and see the printout. If the printout says "[java] spot.sdk.version: blue-080827", you have the right configuration otherwise you need to upgrade or downgrade your firmware. For more information please refer to <http://www.sunspotworld.com/>



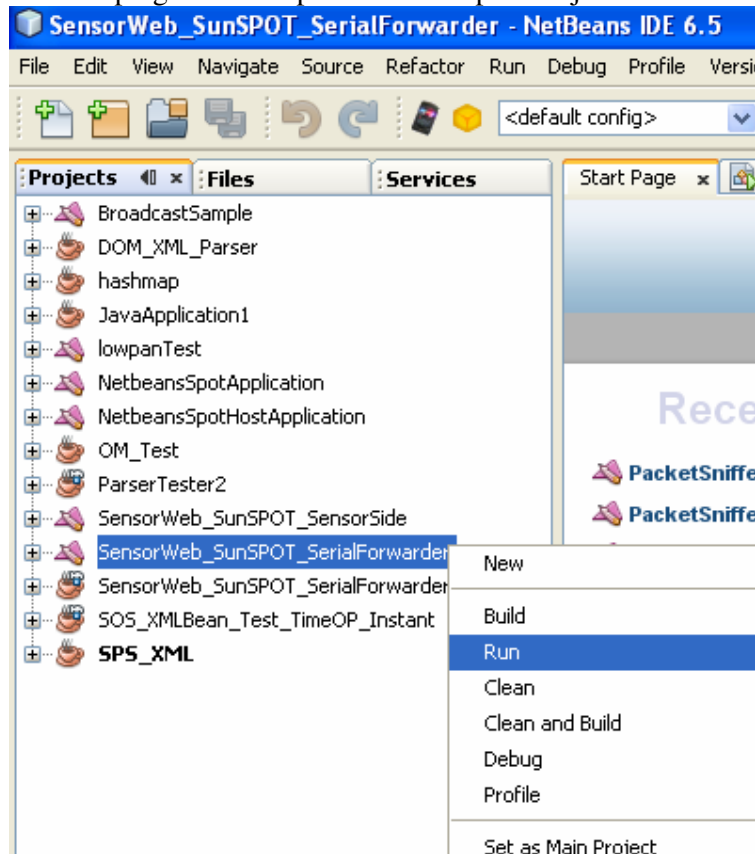
2.6.4 Build and deploy “SensorWeb_SunSPOT_SensorSide” to SunSPOTs
 Open “SensorWeb_SunSPOT_SensorSide” from NetBeans. Right click on the opened project and select “Build project + Deploy to Sun SPOT”, if you did not see this option, you should install NetBeans SunSPOT plugin from <https://netbeans-spot.dev.java.net/>



2.6.5 Connect SunSPOT basestation to host PC and run

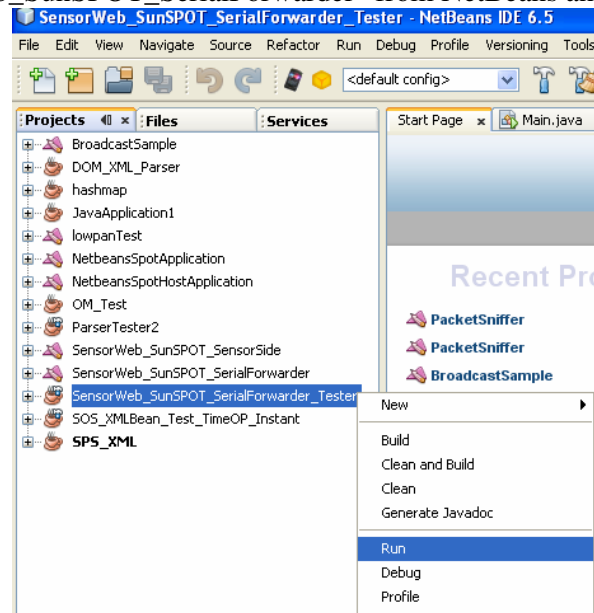
“SensorWeb_SunSPOT_SerialForwarder”

Open “SensorWeb_SunSPOT_SerialForwarder” from NetBeans. Right click on the opened project and select “Run”, if you did not see this option, you should install NetBeans SunSPOT plugin from <https://netbeans-spot.dev.java.net/>

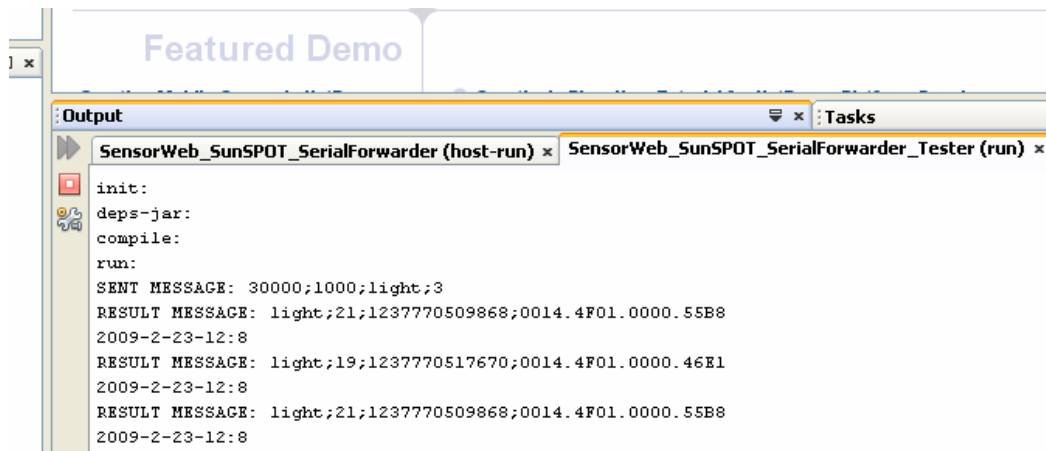
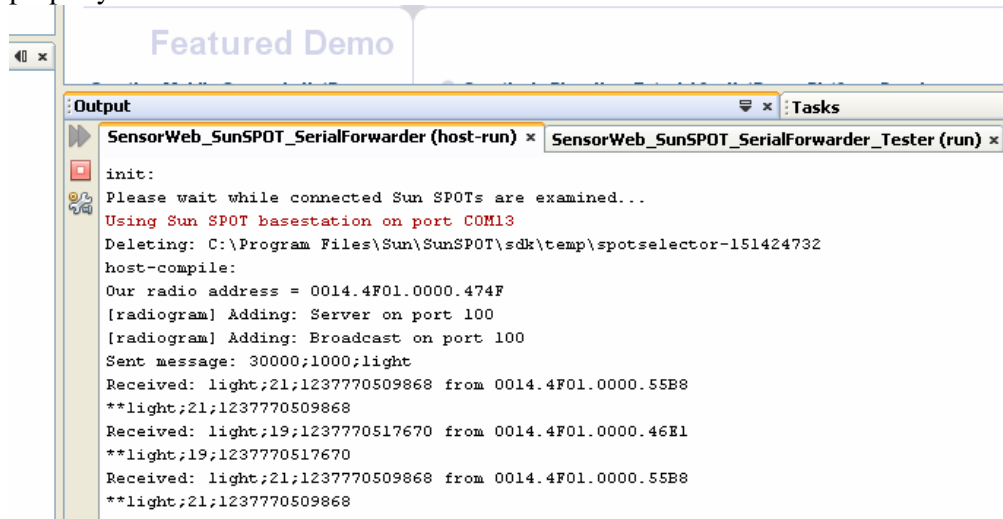


2.6.6 Run “SensorWeb_SunSPOT_SerialForwarder_Tester”

Open “SensorWeb_SunSPOT_SerialForwarder” from NetBeans and run it.

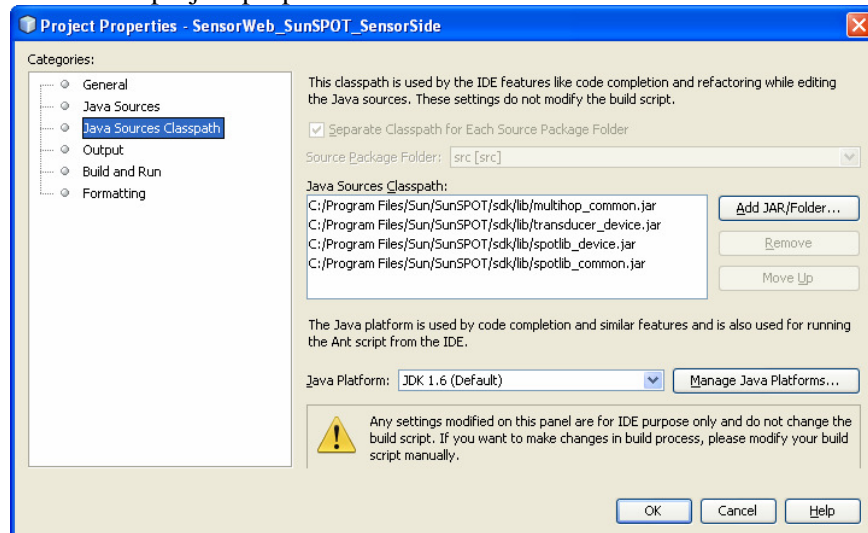


If you can see similar output below, your SunSPOT SerialForwarder is running properly



2.6.7 Common errors

You may need to change the entries of required Java libraries according to your SunSPOT SDK in project properties



2.7. Java Build Errors

BUILD FAILED

build.xml:41: Unable to find a javac compiler;
com.sun.tools.javac.Main is not on the classpath.
Perhaps JAVA_HOME does not point to the JDK

Eclipse -> preferences ->Ant-> classpath

ensure that tools.jar from the JDK you are using is located
in the classpath. If not go to "Add External Classpath"
and add it from c:/program files/java/jdk_1.5.0_2

2.8. JAR files

To extract files, including manifest file from jar archive, execute:

```
> jar xfv jaxrpc.jar
```

To view contents of jar file, execute:

```
> jar tf filename.jar
```

2.9. Globus

To setup Eclipse environment follow the guide "Using Eclipse to Develop Grid Services" available at:

http://grid.lamost.org/files/Eclipse_grid.pdf

To learn about grid services check out the "Borja's Grid Services Tutorial" available at:

<http://gdp.globus.org/gt4-tutorial/>

2.10. TcpTunnelGui

Follow these steps to use TcpTunnel for monitoring purpose:

1. Download soap.jar from <http://ws.apache.org/wsif/>
2. Extract the file into directory of your choice (for me, it was C:\workspace\)
3. Run it

```
> cd /cygdrive/c/workspace/soap-2_3_1/lib  
> java -cp soap.jar org.apache.soap.util.net.TcpTunnelGui 8081 localhost  
8080
```

2.11. SVN

Start a new repository:

```
cd /home/projects/nosa/ (create this directory first!)  
svnadmin create --fs-type fsfs svnrepository_1.0
```

```
svn import -m "Post- Techfest, with reworked SensorML repository code"  
/home/staff/tkob/workspace_baseline_3_July_06/  
file:///home/projects/nosa/svnrepository\_1.0/
```

```
svnlook svn+ssh://mundula.melbuni.edu.au/home/project  
s/nosa/svnrepository_1.0
```

2.12. Tomcat with eclipse edit

To ensure tinyOS jars are copied into the wsrf/WEB-INF/lib directory

add:

```
<copy toDir="${ webapp.webinf.lib.dir} ">  
  <fileset dir="${ tinyos.lib.loc} ">  
    <include name="*.jar"/>  
  </fileset>  
</copy>
```

under <target name="_baseTomcatDeploy"> (approx line 116)

under \$GLOBUS_LOCATION/share/globus_wsrf_common/tomcat/tomcat.xml

2.13. Logging

Change in /cygdrive/c/Program Files/Java/jdk1.5.0_02/jre/lib/logging.properties
console.logging.level from FINE to Finest

2.14. Documentation

Documentation on Web Services

<http://ws.apache.org/axis/java/index.html>

When adding new webservices add to:

```
C:\Program Files\Apache Software Foundation\Tomcat  
5.5\webapps\SensorWeb\WEB-INF\server-config.wsdd
```

3. Common Installation Problems

This section contains the common problem encountered while trying to install and compile various software specified in Section 2.

3.1. Errors when compiling TinyOS

Problem:

When I tried to compile TinyOS, I get an error and the following scree of output is displayed:

```
Global.h: In member function @~@~Xvoid TPt<TRec>::MkRef()@~@~Y:
Global.h:45: error: @~@~XNULL@~@~Y was not declared in this scope
Global.h: In member function @~@~Xvoid TPt<TRec>::UnRef()@~@~Y:
Global.h:46: error: @~@~XNULL@~@~Y was not declared in this scope
Global.h: In constructor @~@~XTPt<TRec>::TPt()@~@~Y:
Global.h:48: error: @~@~XNULL@~@~Y was not declared in this scope
Global.h: In member function @~@~XTRec* TPt<TRec>::operator->()
const@~@~Y: Global.h:59: error: @~@~XNULL@~@~Y was not declared
in this scope
Global.h: In member function @~@~XTRec& TPt<TRec>::operator*()
const@~@~Y: Global.h:60: error: @~@~XNULL@~@~Y was not declared
in this scope
Global.h: In member function @~@~XTRec& TPt<TRec>::operator[](int)
const@~@~Y: Global.h:61: error: @~@~XNULL@~@~Y was not declared
in this scope

make[4]: *** [AvrDummy.o] Error 1
make[4]: Leaving directory `/home/tkob/tinyos-1.1.0/tools/src/uisp/src'
make[3]: *** [all-recursive] Error 1
make[3]: Leaving directory `/home/tkob/tinyos-1.1.0/tools/src/uisp'
make[2]: *** [all] Error 2
make[2]: Leaving directory `/home/tkob/tinyos-1.1.0/tools/src'
make[1]: *** [all-recursive] Error 2
make[1]: Leaving directory `/home/tkob/tinyos-1.1.0/tools'
make: *** [all] Error 2
```

Solution:

find the Globals.h file and edit it, replacing all instances of NULL with 0.

3.2. Exception when trying to start up Tomcat

Problem:

When I tried to start up Tomcat using the startup.bat in the \$CATALINA_HOME folder, it gives me the following error

```
SEVERE: Error initializing endpoint
java.net.BindException: Address already in use: JVM_Bind:8080
```

```
at
org.apache.tomcat.util.net.PoolTcpEndpoint.initEndpoint(PoolTcpEndpoi
nt.java:297)
at
org.apache.coyote.http11.Http11BaseProtocol.init(Http11BaseProtocol.ja
va:138)
at
org.apache.catalina.connector.Connector.initialize(Connector.java:1016)
at
org.apache.catalina.core.StandardService.initialize(StandardService.java:5
80)
at
org.apache.catalina.core.StandardServer.initialize(StandardServer.java:79
1)
at org.apache.catalina.startup.Catalina.load(Catalina.java:503)
at org.apache.catalina.startup.Catalina.load(Catalina.java:523)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at
sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorIm
pl.java:39)
at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAc
cessorImpl.java:25)
at java.lang.reflect.Method.invoke(Method.java:324)
at org.apache.catalina.startup.Bootstrap.load(Bootstrap.java:266)
at org.apache.catalina.startup.Bootstrap.main(Bootstrap.java:431)
Dec 19, 2006 1:07:57 PM org.apache.catalina.startup.Catalina load
SEVERE: Catalina.start
LifecycleException: Protocol handler initialization failed:
java.net.BindException: Address already in use: JVM_Bind:8080
at
org.apache.catalina.connector.Connector.initialize(Connector.java:1018)
at
org.apache.catalina.core.StandardService.initialize(StandardService.java:5
80)
at
org.apache.catalina.core.StandardServer.initialize(StandardServer.java:79
1)
at org.apache.catalina.startup.Catalina.load(Catalina.java:503)
at org.apache.catalina.startup.Catalina.load(Catalina.java:523)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at
sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorIm
pl.java:39)
at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAc
cessorImpl.java:25)
at java.lang.reflect.Method.invoke(Method.java:324)
at org.apache.catalina.startup.Bootstrap.load(Bootstrap.java:266)
at org.apache.catalina.startup.Bootstrap.main(Bootstrap.java:431)
```

Solution:

This is one of the most common problems found in using Tomcat. This happens simply because there is another web server (or any other process) that has already claimed port 8080. This can also occur when you have already started a Tomcat previously and didn't shut it down properly. To shut it down properly, use the script shutdown.bat available at \$CATALINA_HOME/bin

To find out more solution for this issue, please refer to \$CATALINA_HOME/running.txt

3.3. Failed to Initialize SensorCollectionService

Problem:

```
Failed to initialize 'globus/SensorCollectionService' service. Caused by
java.lang.InstantiationException:
localhost.globus.SensorWeb.services.SensorCollectionService.service.Sen
sorCollectionService
    at java.lang.Class.newInstance0(Class.java:335)
    at java.lang.Class.newInstance(Class.java:303)
    at
org.globus.axis.providers.RPCProvider.getNewServiceInstance(RPCProvi
der.java:115)
    at
org.globus.axis.description.ServiceDescUtil.initializeProviders(ServiceDe
scUtil.java:190)
    at
org.globus.axis.description.ServiceDescUtil.initializeService(ServiceDesc
Util.java:135)
    at
org.globus.wsrp.container.ServiceManager$InitPrivilegedAction.initialize(
ServiceManager.java:252)
    at
org.globus.wsrp.container.ServiceManager.initializeService(ServiceMana
ger.java:215)
    at
org.globus.wsrp.container.ServiceManager.start(ServiceManager.java:148
)
    at
org.globus.wsrp.container.ServiceDispatcher.init(ServiceDispatcher.java:1
31)
    at
org.globus.wsrp.container.ServiceContainer.start(ServiceContainer.java:2
37)
    at
org.globus.wsrp.container.ServiceContainer.<init>(ServiceContainer.java:
195)
    at
```



```

org.globus.wsrfl.container.GSIServiceContainer.<init>(GSIServiceContai
ner.java:45)
    at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native
Method)
    at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstruct
orAccessorImpl.java:39)
    at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingC
onstructorAccessorImpl.java:27)
    at java.lang.reflect.Constructor.newInstance(Constructor.java:494)
    at
org.globus.wsrfl.container.ServiceContainer.createContainer(ServiceConta
iner.java:136)
    at
org.globus.wsrfl.container.ServiceContainer.startSecurityContainer(Servic
eContainer.java:449)
    at
org.globus.wsrfl.container.ServiceContainer.main(ServiceContainer.java:4
03)
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at
sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorIm
pl.java:39)
    at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAc
cessorImpl.java:25)
    at java.lang.reflect.Method.invoke(Method.java:585)
    at org.globus.bootstrap.BootstrapBase.launch(BootstrapBase.java:95)
    at org.globus.bootstrap.Bootstrap.main(Bootstrap.java:37)

```

Solution:

All my classes weere mixed up. My org_sensorweb_sensor_collectionsevice.jar file had no content! I fixed this by following Borja's tutorial more closely and modifying my class paths in my wsdl file to more closely align with his. Special care was taken to separate my actual implementation (.impl) from the generated stubs. Which is what I got mixed up.

3.4. Axis Jar Problem

Problem:

```

Exception in thread "main" java.lang.NoSuchMethodError:
org.apache.axis.description.TypeDesc.<init>(Ljava/lang/Class;Z)V
    at
org.apache.axis.message.addressing.EndpointReferenceType.<clinit>(End
pointReferenceType.java:333)

```

```
at  
org.sensorweb.service.globus.client.SensorWebClient.main(SensorWebClient.java:42)
```

Solution:

Different axis.jar files were being used to compile my code, compared to those used to execute my code. I followed the Eclipse grid services document to create a new project where I had complete control over my jar files.

4. Postgres Installation

This section contains guide on set up and initialisation of PostgreSQL 7.4.5 under Cygwin / Windows XP

Note that the postgres package that we need to use is PostgreSQL is the version 7.4.5 and it is already included in the TinyOS default install.

4.1. Set up

4.1.1. Step 0 - Setting Up Cygserver

Type in the following command in your Cygwin

```
> export CYGWIN=server
> cygserver-config
```

Note that a client application which wants to use the services provided by cygserver *must* have the environment variable CYGWIN set so that it contains the word "server". So if you don't need any other special CYGWIN setting, just set it to "server".

It is advisable to add this setting to the Windows system environment.

4.1.2. Step 1. Start Cygserver for Shared Memory Support

To do this, enter the command:

```
> /usr/sbin/cygserver.exe &
```

This program needs to be running anytime you start the PostgreSQL server (postmaster) or initialize a database (initdb).

4.1.3. Step 2. Create a New Database Cluster

Use the initdb command to create a new database cluster.

Enter the following command in your cygwin which will create a new database cluster in the /var/postgresql/data directory

```
> /usr/sbin/initdb -D /var/postgresql/data -W
```

NOTE: As a reference, a database cluster is a database storage area on disk. This is a collection of databases accessible by a single instance of a running database server

4.1.4. Step 3. Start up the postmaster

We can now start the database server using one of the following command

```
> /usr/sbin/postmaster -D /var/postgresql/data
```

or

```
> /usr/sbin/postmaster -D /var/postgresql/data >logfile 2>&1 &
```

The first command will leave the server running into the foreground while the second command will start the postmaster in the background. Note also that the second command will redirect the entire log message to the file called logfile.

This will start the postmaster, and if successful you will see some initial log entries, and an entry "LOG: database system is ready".

Before anyone can access the database, you must start the database server (i.e. postmaster).

NOTE: For some of the common problems associated with postmaster start up failure, please refer to <http://www.postgresql.org/docs/7.4/static/postmaster-start.html>

* Create Users:

Note: This is not totally necessary, by default the user that executed createdb will already be in the database, however you may want to set a password (see below).

```
> CREATE USER tkob;
```

Use:

```
> DROP USER name;  
to remove a user.
```

Command line equivalents also exist:

```
createuser name  
dropuser name
```

To list users in database:

```
"select * from pg_user;" OR \du
```

* Set password:

```
ALTER USER tkob PASSWORD '*****';
```

Step 4. Create a Database

```
> createdb sensorml_repository ----DONE
```

To remove a database:

```
dropdb sensorml_repository
```

Once you have created a database, you can access it by running the PostgreSQL interactive terminal program, called psql, which allows you to interactively enter, edit, and execute SQL commands

```
> psql sensorml_repository
```

NOTE: this is for terminal access only, for in-program access you will need to use the java connector (<http://www.postgresql.org/docs/7.4/interactive/client-interfaces.html>)

For help:

```
> mydb=> \h
```

To get out of psql, type

```
> mydb=> \q
```

To read in SQL commands from a specific file:

```
> mydb=> \i sensorml_db_populate.sql
```

Commands:

```
\l - list all databases on server
```

```
Use a Specific Database: \c databaseName;
```

```
List All Objects in the Database: \d
```

```
List Tables: \dt
```

```
(or \d[ivsS] to list indexes, views, sequences, Schema objects)
```

```
Count All Rows in a Table: select count(*) from tableName;
```

```
List All Rows in a Table: select * from tableName;
```

4.2. *Command Notes*

To remove a table:

```
> DROP TABLE tablename;
```

The INSERT statement is used to populate a table with rows

UP TO: <http://www.postgresql.org/docs/7.4/interactive/tutorial-join.html>

4.3. *Installing JDBC Driver*

This will allow your Java code to talk to Postgres:

```
> cd /cygdrive/c/workspace/Sensor -Source/lib  
> cp /usr/share/postgresql/java/postgresql.jar .
```

!!!IMPORTANT!!!

Because Java only uses TCP/IP connections, the PostgreSQL server must be configured to accept TCP/IP connections. This can be done by setting `tcpip_socket = true` in the `postgresql.conf` file or by supplying the `-i` option flag when starting `postmaster`.

Also: uncomment Port 5432 in configuration file.

4.4. *Running Postgres*

```
> export CYGWIN=server  
> /usr/sbin/cygservice &  
> /usr/sbin/postmaster -i -D /var/postgresql/data &
```

`-i` allows clients to connect over TCP/IP as opposed to just those from the local machine.

4.5. *Shutting Down Postgres*

```
> pg_ctl stop -w -D /var/postgresql/data -s -m smart  
> kill $(ps -f | grep cygservice | awk '{print $2}')
```

--Place the text below in `sensorml_db_populate.sql` under `cygwin` directory

-- Tom Kobialka tkob@cs.mu.oz.au

```

-- NICTA NOSA Project
-- 10th April 2006

-- Skeleton SQL TABLEs for SensorML Repository

-- Postgres Startup
-- =====
-- export CYGWIN=server
-- /usr/sbin/cygserver &
-- postmaster -i -D /var/postgresql/data &

-- Postgres Shutdown
-- =====
-- pg_ctl stop -w -D /var/postgresql/data -s -m smart
-- kill $(ps -f | grep cygserver | awk '{print $2}')

CREATE TABLE Sensor(
    SensorId varchar(80) PRIMARY KEY          -- Sensor ID
);

CREATE TABLE StationaryPlatform(
    PlatformId varchar(80) PRIMARY KEY       -- Platform ID
);

CREATE TABLE PlatformDescription(
    PlatformId varchar(80) REFERENCES StationaryPlatform (PlatformId), --
    Foreign Key
    descriptionContent varchar(250)
);

CREATE TABLE namespace(
    SensorId varchar(80) REFERENCES Sensor (SensorId),
    nsPrefix varchar(80),
    nsUri varchar(250)
);

CREATE TABLE SensorDescription(
    SensorId varchar(80) REFERENCES Sensor (SensorId), -- Foreign Key
    descriptionContent varchar(250)
);

CREATE TABLE identification(
    SensorId varchar(80) REFERENCES Sensor (SensorId), -- Foreign Key
    identifierName varchar(80),
    idTermQualifier varchar(80),
    idTermContent varchar(80)
);

CREATE TABLE classification(
    SensorId varchar(80) REFERENCES Sensor (SensorId), -- Foreign Key

```

```

    classifierName varchar(80),
    clTermQualifier varchar(80),
    clTermContent varchar(80)
);

CREATE TABLE inputs(
    SensorId varchar(80) REFERENCES Sensor (SensorId), -- Forigen Key
    inputId varchar(80) PRIMARY KEY
);

CREATE TABLE inputList(
    inputId varchar(80) REFERENCES inputs (inputId),
    inputName varchar(80),
    inputType varchar(80),
    inputDefinition varchar(80),
    inputUom varchar(80),
    min int,
    max int
);

CREATE TABLE outputs(
    SensorId varchar(80) REFERENCES Sensor (SensorId), -- Forigen Key
    outputId varchar(80) PRIMARY KEY
);

CREATE TABLE outputList(
    outputId varchar(80) REFERENCES outputs (outputId),
    outputName varchar(80),
    outputType varchar(80),
    outputDefinition varchar(80),
    outputUom varchar(80),
    min int,
    max int
);

CREATE TABLE method(
    methodId varchar(80) REFERENCES Sensor (SensorId), -- Forigen Key
    methodName varchar(150)
);

-- Tom Kobialka tkob@cs.mu.oz.au
-- NICTA NOSA Project
-- 23 June 2006

-- Sample Data
-- Manual Insertion into Postgres database.. Should be done dynamically

-- Postgres Startup
-- =====
-- export CYGWIN=server

```



```

-- /usr/sbin/cygsrver &
-- postmaster -i -D /var/postgresql/data &

-- Postgres Shutdown
-- =====
-- pg_ctl stop -w -D /var/postgresql/data -s -m smart
-- kill $(ps -f | grep cygsrver | awk '{print $2}')

-- WorkStation
INSERT INTO Sensor VALUES('Sensor_126');
INSERT INTO namespace
VALUES('Sensor_126','sml','http://www.opengis.net/sensorML');
INSERT INTO namespace
VALUES('Sensor_126','gml','http://www.opengis.net/gml');
INSERT INTO namespace
VALUES('Sensor_126','swe','http://www.opengis.net/swe');

INSERT INTO StationaryPlatform VALUES('Platform_126');
INSERT INTO PlatformDescription VALUES('Platform_126','Dell
Workstation, PC 686, Melbourne University 2006');

INSERT INTO SensorDescription VALUES('Sensor_126','No Sensors are
installed on the workstation');
INSERT INTO identification
VALUES('Sensor_126','longName','urn:ogc:def:identifier:longName','Workstati
on');
INSERT INTO classification
VALUES('Sensor_126','longName','urn:ogc:def:classifier:observationClassifica
tion','in-situ');

INSERT INTO inputs VALUES('Sensor_126','Sensor_126');
INSERT INTO inputList
VALUES('Sensor_126',NULL,NULL,NULL,NULL,NULL,NULL);

INSERT INTO outputs VALUES('Sensor_126','Sensor_126');
INSERT INTO outputList
VALUES('Sensor_126',NULL,NULL,NULL,NULL,NULL,NULL);

INSERT INTO method
VALUES('Sensor_126','urn:ogc:def:process:1.0:sensor');

-- Base Station
INSERT INTO Sensor VALUES('Sensor_0');
INSERT INTO namespace
VALUES('Sensor_0','sml','http://www.opengis.net/sensorML');
INSERT INTO namespace
VALUES('Sensor_0','gml','http://www.opengis.net/gml');
INSERT INTO namespace
VALUES('Sensor_0','swe','http://www.opengis.net/swe');

```

```

INSERT INTO namespace
VALUES('Sensor_0','xLink','http://www.w3.org/2001/XMLSchema-instance');

-- Platform
INSERT INTO StationaryPlatform VALUES('Platform_0');
INSERT INTO PlatformDescription VALUES('Platform_0','Crossbow
MIB510CA Base Station for Mica2 Motes with MPR400CB receiver, copyright
2003');

-- Sensors
INSERT INTO SensorDescription VALUES('Sensor_0','Crossbow Sensor 0
node, Mica Sensor board MTS300CA');
INSERT INTO identification
VALUES('Sensor_0','longName','urn:ogc:def:identifier:longName','Base
Station');
INSERT INTO classification
VALUES('Sensor_0','longName','urn:ogc:def:classifier:observationClassificatio
n','in-situ');

INSERT INTO inputs VALUES('Sensor_0','Sensor_0');
INSERT INTO inputList
VALUES('Sensor_0','nodeid','Count','urn:ogc:def:data:index',NULL,'0','1000');
INSERT INTO inputList
VALUES('Sensor_0','temperature','Quantity','urn:ogc:def:phenomenon:temperat
ure','urn:ogc:def:unit:celcius',NULL,NULL);
INSERT INTO inputList
VALUES('Sensor_0','light','Quantity','urn:ogc:def:illumination','urn:ogc:def:unit
:lux',NULL,NULL);

INSERT INTO outputs VALUES('Sensor_0','Sensor_0');
INSERT INTO outputList
VALUES('Sensor_0','measuredNodeid','Count','urn:ogc:def:data:index',NULL,'
0','1000');
INSERT INTO outputList
VALUES('Sensor_0','measuredTemperature','Quantity','urn:ogc:def:phenomeno
n:temperature','urn:ogc:def:unit:celcius',NULL,NULL);
INSERT INTO outputList
VALUES('Sensor_0','measuredLight','Quantity','urn:ogc:def:illumination','urn:o
gc:def:unit:lux',NULL,NULL);

INSERT INTO method VALUES('Sensor_0','urn:ogc:def:process:1.0:sensor');

-- Sensor 1
INSERT INTO Sensor VALUES('Sensor_1');
INSERT INTO namespace
VALUES('Sensor_1','sml','http://www.opengis.net/sensorML');
INSERT INTO namespace
VALUES('Sensor_1','gml','http://www.opengis.net/gml');
INSERT INTO namespace
VALUES('Sensor_1','swe','http://www.opengis.net/swe');

```

```

INSERT INTO namespace
VALUES('Sensor_1','xLink','http://www.w3.org/2001/XMLSchema-instance');

-- platform
INSERT INTO StationaryPlatform VALUES('Platform_1');
INSERT INTO PlatformDescription VALUES('Platform_1','MPR400CB 2003
Crossbow receiver');

-- sensors
INSERT INTO SensorDescription VALUES('Sensor_1','Crossbow Sensor 1
node, Mica Sensor board MTS300CA');
INSERT INTO identification
VALUES('Sensor_1','longName','urn:ogc:def:identifier:longName','Sensor 1');
INSERT INTO classification
VALUES('Sensor_1','longName','urn:ogc:def:classifier:observationClassificatio
n','in-situ');

INSERT INTO inputs VALUES('Sensor_1','Sensor_1');
-- INSERT INTO inputList
VALUES('Sensor_1','nodeid','Count','urn:ogc:def:data:index',NULL,'0','1000');
INSERT INTO inputList
VALUES('Sensor_1','sound','Quantity','urn:ogc:def:sound','urn:ogc:def:unit:dB',
NULL,NULL);
INSERT INTO inputList
VALUES('Sensor_1','light','Quantity','urn:ogc:def:illumination','urn:ogc:def:unit
:lux',NULL,NULL);

INSERT INTO outputs VALUES('Sensor_1','Sensor_1');
--INSERT INTO outputList
VALUES('Sensor_1','measuredNodeid','Count','urn:ogc:def:data:index',NULL,'
0','1000');
--INSERT INTO outputList
VALUES('Sensor_1','measuredTemperature','Quantity','urn:ogc:def:phenomeno
n:temperature','urn:ogc:def:unit:celcius',NULL,NULL);
INSERT INTO outputList
VALUES('Sensor_1','measuredLight','Quantity','urn:ogc:def:illumination','urn:o
gc:def:unit:lux',NULL,NULL);
INSERT INTO outputList
VALUES('Sensor_1','measuredSound','Quantity','urn:ogc:def:sound','urn:ogc:de
f:unit:dB',NULL,NULL);

INSERT INTO method VALUES('Sensor_1','urn:ogc:def:process:1.0:sensor');

-- Sensor 2
INSERT INTO Sensor VALUES('Sensor_2');
INSERT INTO namespace
VALUES('Sensor_2','sml','http://www.opengis.net/sensorML');
INSERT INTO namespace
VALUES('Sensor_2','gml','http://www.opengis.net/gml');
INSERT INTO namespace

```

```

VALUES('Sensor_2','swe','http://www.opengis.net/swe');
INSERT INTO namespace
VALUES('Sensor_2','xLink','http://www.w3.org/2001/XMLSchema-instance');
-- platform
INSERT INTO StationaryPlatform VALUES('Platform_2');
INSERT INTO PlatformDescription VALUES('Platform_2','MPR400CB 2003
Crossbow receiver');

-- sensors
INSERT INTO SensorDescription VALUES('Sensor_2','Crossbow Sensor 2
node, Mica Sensor board MTS300CA');
INSERT INTO identification
VALUES('Sensor_2','longName','urn:ogc:def:identifier:longName','Sensor 2');
INSERT INTO classification
VALUES('Sensor_2','longName','urn:ogc:def:classifier:observationClassificatio
n','in-situ');

INSERT INTO inputs VALUES('Sensor_2','Sensor_2');
-- INSERT INTO inputList
VALUES('Sensor_2','nodeid','Count','urn:ogc:def:data:index',NULL,'0','1000');
INSERT INTO inputList
VALUES('Sensor_2','sound','Quantity','urn:ogc:def:sound','urn:ogc:def:unit:dB',
NULL,NULL);
INSERT INTO inputList
VALUES('Sensor_2','temperature','Quantity','urn:ogc:def:phenomenon:temperat
ure','urn:ogc:def:unit:celcius',NULL,NULL);

INSERT INTO outputs VALUES('Sensor_2','Sensor_2');
--INSERT INTO outputList
VALUES('Sensor_2','measuredNodeid','Count','urn:ogc:def:data:index',NULL,'
0','1000');
INSERT INTO outputList
VALUES('Sensor_2','measuredTemperature','Quantity','urn:ogc:def:phenomeno
n:temperature','urn:ogc:def:unit:celcius',NULL,NULL);
INSERT INTO outputList
VALUES('Sensor_2','measuredSound','Quantity','urn:ogc:def:sound','urn:ogc:de
f:unit:dB',NULL,NULL);

INSERT INTO method VALUES('Sensor_1','urn:ogc:def:process:1.0:sensor');

=====
Drop scripts should be placed in some other file.
=====

DROP TABLE method CASCADE;
DROP TABLE outputs CASCADE;
DROP TABLE outputList CASCADE;
DROP TABLE inputs CASCADE;
DROP TABLE inputList CASCADE;
DROP TABLE SensorDescription CASCADE;

```

```
DROP TABLE PlatformDescription CASCADE;
DROP TABLE namespace CASCADE;
DROP TABLE classification;
DROP TABLE identification;
DROP TABLE StationaryPlatform;
DROP TABLE sensor;
DROP TABLE historic_observation;
DROP TABLE user_table;
DROP TABLE id_table;
```

```
=====
Scripts should be placed in separate file.
=====
```

```
-- Tom Kobialka tkob@cs.mu.oz.au
-- NICTA NOSA Project
-- 12th December 2006
```

```
-- Skeleton SQL TABLEs for Sensor Repository Service user repository
```

```
CREATE TABLE historic_observation(
    nodeid varchar(20),
    light varchar(20),
    temperature varchar(20)
);
```

```
-- Tom Kobialka tkob@cs.mu.oz.au
-- NICTA NOSA Project
-- 31st October 2006
```

```
-- Skeleton SQL TABLEs for Web Notification Service user repository
```

```
CREATE TABLE user_table(
    id varchar(20),
    name varchar(80),
    contact varchar(80)
);
```

```
CREATE TABLE id_table(
    id varchar(20)
);
```

5. Getting the Services Running

5.1. Before we start

The OSWA codes require JDK 1.5.0_02. They won't work under JDK 1.4 You need to make sure to set this as your default Java. To set it as your default JAVA, point the environment variable JAVA_HOME and PATH appropriately. Make sure also that Ant and Tomcat refers to JDK 1.5.0_02 in case you have more than one Java installed on your machine.

5.2. Configuring Eclipse

Firstly, install Eclipse and Globus following documentation. It is also helpful to configure Eclipse to be able to develop grid services. Follow the documentation provided in http://grid.lamost.org/files/Eclipse_grid.pdf for guide on how to configure Eclipse to develop grid services.

Then you need to make sure also to get tomcat running with eclipse. The guide to do that can be found in the same guide provided above.

Importing the OSWA codes

All the codes are located in OSWA directory.

To start with, you should have been given the OSWA directory containing all the codes. If you are using Eclipse and you are just starting out, start by importing the Project into your workspace. Choose File->Import. In the dialog, choose General->Existing Projects into Workspace. Follow the prompt as appropriately.

Rearrange the directory.

tinynos-lib contains the library file that you will need. Just place it into the same directory as OSWA as depicted above

It should also be in the same directory as ws-core-4.0.2

This is the way I arrange my workspace:

Workspace

- ws-core-4.0.3
- OSWA
- tinynos-lib
- Surge-View

3 Errors encountered after importing OSWA project into my workspace

Severity and Description	Path	Resource	Location	Creation Time	Id
The project		OSWA	Unknown	1167767733390	26

cannot be built until build path errors are resolved					
Unbound classpath container: 'GT4' in project OSWA		OSWA	Build path	1167767756859	77
Unbound classpath container: 'Tiny-OS' in project OSWA		OSWA	Build path	1167767756859	78

The first error is caused by the next two errors so we will need to fix those errors.

Unbound classpath container is caused by the fact that Eclipse could not find these libraries (this could be defined as user libraries previously). To check this, open.classpath file from OSWA directory and see if there are two user libraries with the name of GT4 and Tiny-OS defined there.

What we need to do is to update the Java Build Path libraries to include those GT4 and Tiny-OS libraries. By doing so will enable the Eclipse editor to find those classes that OSWA sources will import.

Go to Project->Properties page. Select the Java Build Path page and click the Libraries tab. You would notice that all those libraries will be listed and there are two unbound libraries which are GT4 and Tiny-OS itself.

Fix each one of them by clicking Edit. In the dialog that comes up, choose the user library for each of them. If it is empty, you will need to create a new user library for each of them. For the GT4 library, create a new user library with the name GT4 Library, then click the button Add JARs and select and add all of the JARs in \$GLOBUS_LOCATION\lib to the library. Selecting OK and Finish leaves us with the properly configured Libraries tab. The GT4 library should not be unbounded anymore. Note that \$GLOBUS_LOCATION refer to the location where your GT4 is installed.

Do the similar thing with Tiny-OS library. The only difference is that, add the JARs from the tinyos-lib provided together with OSWA.

5.3. Machine Specific Configuration

Some machine specific configuration will needs to be done since some variable is hardcoded and some folder location could be different in two different machine

First, open some of the files listed below and change some of the variable to suit your machine. The comments in each file will help you determine which path to be changed.

- buildservice.properties:
 - globus.bs.loc
 - root.OSWA
 - tomcat.dir
- buildservice.xml:
 - tinyos.lib.loc (Note that there are more than 1 occurrences of this variable to be changed. In fact, there are 4 occurrences of them)
- application.properties:
 - tinydb.catalog

Note also that you might need to edit build_***.xml if hard coded path elements exist (if we miss anything)

Note that buildservice.xml uses buildservice.properties for configuration. You will need to set your paths in this file.

You also need to add application.properties to your Java build path in Eclipse. Go to Window->Preferences then choose Java->Build Path->Classpath Variables. Add new variable entries for GLOBUS_LOCATION and APPLICATION_PROPERTIES. For GLOBUS_LOCATION, point it to the location of the GT4 folder (it is C:/workspace/ws-core-4.0.3/ in my machine). For APPLICATION_PROPERTIES, point it to the location of \$OSWA/application.properties where OSWA represents the location of OSWA source code. You may need to type the location directly instead of using the file selector.

5.4. Building the OSWA Project

Run Ant on buildservice.xml. Start the building target starting from the bottom target.

In buildservice.xml, we have the following build targets:

```
<!--<target name="all" depends="SensorPlanningService" />-->
<!--<target name="all" depends="SensorCollectionService" />-->
<!--<target name="all" depends="SensorRepositoryService" />-->
<!--<target name="all" depends="WebNotificationService" /> -->
```

Initially, uncomment the bottom target (remove the <!-- and -->) and execute Ant build on it.

After that is built, comment the bottom line and uncomment the next above that. Keep on doing that until you reach the top.

In other words, this is the sequence order on which the build should be executed:

1. WebNotificationService
2. SensorRepositoryService
3. SensorCollectionService

4. SensorPlanningService

To execute Ant build on the targets, follow the instructions given in the tutorial for building grid services using Eclipse. (this is the link in case you forget it already http://grid.lamost.org/files/Eclipse_grid.pdf)

5.5. Error Encountered When Building the OSWA Project

Error:

The eclipse editor shows red compilation errors on the package org.jdom and all the other class that is imported from that package.

Solution:

The problem comes from the fact that Eclipse couldn't see the package org.jdom. When you install TinyOS, jdom.jar file is included in \$TOSROOT/tools/java/jars. Add this jar as a new user library to silence all those Editor View's red compile errors. Ensure that you use the jdom.jar provided by TinyOS

Error:

When I tried built the target, it gives me many compilation errors. From closer looks, it seems that most of the problem is because Eclipse couldn't find the package org.jdom.

Solution:

This is because Eclipse could see the package but Ant could not see the package. One solution to this would be to add all the jars to Ant CLASSPATH. One way of doing it is to go to Window->Preferences. In the dialog that comes up, choose Ant->Runtime. Choose Ant Home Entries (Default) and then choose Add External Jars to add jdom.jar from \$TOSROOT/tools/java/jars. Ensure that you use the jdom.jar provided by TinyOS.

Error:

I get the following error, when I try to access <http://localhost:8080/wsrp/services/globus/SCSFactoryService?wsdl>.

```
java.lang.NoClassDefFoundError: Could not initialize class
org.sensorweb.service.globus.scs.impl.SensorCollectionService
```

Solution:

The problem comes from the fact that Tomcat is not able to locate the application.properties file located in the "C:\Program Files\Apache Software Foundation\Tomcat 6.0\common\classes". Open the dialog in Windows>Preferences>Tomcat>JVM and add the above mentioned path in Classpath textbox.

Error:

When I tried to execute the target, it complains that the any of these packages localhost.globus.WebNotificationService*.* does not exist or

localhost.globus.SensorCollectionService*.* does not exist or
localhost.globus.SensorRepositoryService*.* does not exist

Solution:

This is because the Eclipse expecting a stubs but could not find that stubs anywhere. This is most likely because you executed the build target in the wrong order cause there is some dependencies between them. The correct order is

1. WebNotificationService
2. SensorRepositoryService
3. SensorCollectionService
4. SensorPlanningService

5.6. Building the OSWA Project - 2

Not only Eclipse and Ant, Tomcat also needs to have access to Jdom.jar. Make sure you copy this file (the one given by default from TinyOS installation) to \$TOMCAT/webapps/wsrf/WEB-INF/lib where \$TOMCAT represents the root installation of Tomcat. In my machine, it is c:Program Files/Apache Software Foundation/Tomcat 5.5

Tomcat must also have access to all the jars file inside tinyos-lib. To do this, you can simply copy all the jar files from tinyos-lib to \$TOMCAT/webapps/wsrf/WEB-INF/lib.

After all the 4 build target is executed successfully, we could check if the services work by going to their URL. The URL for these services are defined in application.properties file. Note: change port numbers from 8081 to 8080. Then, visit these URLs to check that these services work.

Note that you need to start Tomcat to be able to access them.

For example: The URL for the web notification service is
<http://localhost:8080/wsrf/services/globus/WebNotificationService>

To check on their WSDL:

<http://localhost:8080/wsrf/services/globus/WebNotificationService?wsdl>

In the case where accessing those services shows you an error, it is helpful to debug Tomcat by making it show more detail log message. To change that, simply open the file \$JAVA_HOME/jre/lib/logging.properties and change the global logging level. The variable is .level and it is INFO by default. Change it to FINE or FINEST.

May want to create a small script which will do everything (or as much as possible) automatically.

Error:

While running DemoFrame.java, the GUI loaded up fine but I get the following error when I tried to use the functionality

```
AxisFault
  faultCode:
  {http://schemas.xmlsoap.org/soap/envelope/}Server.userException
  faultSubcode:
  faultString: java.lang.reflect.InvocationTargetException
  faultActor:
  faultNode:
  faultDetail:

  {http://xml.apache.org/axis/}stackTrace:java.lang.reflect.InvocationTargetException
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at
  sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:39)
    at
  sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:25)
    at java.lang.reflect.Method.invoke(Method.java:585)
    at
  org.apache.axis.providers.java.RPCProvider.invokeMethod(RPCProvider.java:384)
    at
  org.globus.axis.providers.RPCProvider.invokeMethodSub(RPCProvider.java:107)
    at
  org.globus.axis.providers.RPCProvider.invokeMethod(RPCProvider.java:90)
    at
  org.apache.axis.providers.java.RPCProvider.processMessage(RPCProvider.java:281)
    at
  org.apache.axis.providers.java.JavaProvider.invoke(JavaProvider.java:319)
  )
    at
  org.apache.axis.strategies.InvocationStrategy.visit(InvocationStrategy.java:32)
    at org.apache.axis.SimpleChain.doVisiting(SimpleChain.java:118)
    at org.apache.axis.SimpleChain.invoke(SimpleChain.java:83)
    at
  org.apache.axis.handlers.soap.SOAPService.invoke(SOAPService.java:450)
    at org.apache.axis.server.AxisServer.invoke(AxisServer.java:285)
    at
  org.apache.axis.transport.http.AxisServlet.doPost(AxisServlet.java:697)
    at javax.servlet.http.HttpServlet.service(HttpServlet.java:709)
    at
  org.apache.axis.transport.http.AxisServletBase.service(AxisServletBase.java:327)
```

```
    at javax.servlet.http.HttpServlet.service(HttpServlet.java:802)
    at
org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:252)
    at
org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:173)
    at
org.apache.catalina.core.StandardWrapperValve.invoke(StandardWrapperValve.java:213)
    at
org.apache.catalina.core.StandardContextValve.invoke(StandardContextValve.java:178)
    at
org.apache.catalina.core.StandardHostValve.invoke(StandardHostValve.java:126)
    at
org.apache.catalina.valves.ErrorReportValve.invoke(ErrorReportValve.java:105)
    at
org.apache.catalina.core.StandardEngineValve.invoke(StandardEngineValve.java:107)
    at
org.apache.catalina.connector.CoyoteAdapter.service(CoyoteAdapter.java:148)
    at
org.apache.coyote.http11.Http11Processor.process(Http11Processor.java:869)
    at
org.apache.coyote.http11.Http11BaseProtocol$Http11ConnectionHandler.processConnection(Http11BaseProtocol.java:664)
    at
org.apache.tomcat.util.net.PoolTcpEndpoint.processSocket(PoolTcpEndpoint.java:527)
    at
org.apache.tomcat.util.net.LeaderFollowerWorkerThread.runIt(LeaderFollowerWorkerThread.java:80)
    at
org.apache.tomcat.util.threads.ThreadPool$ControlRunnable.run(ThreadPool.java:684)
    at java.lang.Thread.run(Thread.java:595)
Caused by: java.lang.NoClassDefFoundError: org/jdom/Element
    at
org.sensorweb.service.globus.scs.impl.SensorCollectionServiceResource.initialise(SensorCollectionServiceResource.java:42)
    at
org.sensorweb.service.globus.scs.impl.SensorCollectionServiceResourceHome.create(SensorCollectionServiceResourceHome.java:31)
    at
org.sensorweb.service.globus.scs.impl.SensorCollectionServiceFactory.cr
```

```
    eateResource(SensorCollectionServiceFactory.java:31)
      ... 32 more

      {http://xml.apache.org/axis/}hostname:alpha

java.lang.reflect.InvocationTargetException
  at
  org.apache.axis.message.SOAPFaultBuilder.createFault(SOAPFaultBuilder.java:221)
    at
    org.apache.axis.message.SOAPFaultBuilder.endElement(SOAPFaultBuilder.java:128)
      at
      org.apache.axis.encoding.DeserializationContext.endElement(DeserializationContext.java:1087)
        at
        org.apache.xerces.parsers.AbstractSAXParser.endElement(Unknown Source)
          at
          org.apache.xerces.impl.XMLNSDocumentScannerImpl.scanEndElement(Unknown Source)
            at
            org.apache.xerces.impl.XMLDocumentFragmentScannerImpl$FragmentContentDispatcher.dispatch(Unknown Source)
              at
              org.apache.xerces.impl.XMLDocumentFragmentScannerImpl.scanDocument(Unknown Source)
                at org.apache.xerces.parsers.XML11Configuration.parse(Unknown Source)
                  at org.apache.xerces.parsers.XML11Configuration.parse(Unknown Source)
                    at org.apache.xerces.parsers.XMLParser.parse(Unknown Source)
                      at org.apache.xerces.parsers.AbstractSAXParser.parse(Unknown Source)
                        at javax.xml.parsers.SAXParser.parse(SAXParser.java:375)
                          at
                          org.apache.axis.encoding.DeserializationContext.parse(DeserializationContext.java:227)
                            at
                            org.apache.axis.SOAPPart.getAsSOAPEnvelope(SOAPPart.java:645)
                              at org.apache.axis.Message.getSOAPEnvelope(Message.java:424)
                                at
                                org.apache.axis.handlers.soap.MustUnderstandChecker.invoke(MustUnderstandChecker.java:62)
                                  at org.apache.axis.client.AxisClient.invoke(AxisClient.java:206)
                                    at org.apache.axis.client.Call.invokeEngine(Call.java:2727)
                                      at org.apache.axis.client.Call.invoke(Call.java:2710)
```

```
at org.apache.axis.client.Call.invoke(Call.java:2386)
at org.apache.axis.client.Call.invoke(Call.java:2309)
at org.apache.axis.client.Call.invoke(Call.java:1766)
at
localhost.globus.SCSFactoryService.bindings.SCSFactoryPortTypeSOAP
BindingStub.createResource(SCSFactoryPortTypeSOAPBindingStub.java
:224)
at org.sensorweb.demo.DemoWorker.invoke(DemoFrame.java:587)
at org.sensorweb.demo.DemoWorker.run(DemoFrame.java:510)
at java.lang.Thread.run(Thread.java:595)
```

Solution:

The problem lies in this line

```
Caused by: java.lang.NoClassDefFoundError: org/jdom/Element
at
org.sensorweb.service.globus.scs.impl.SensorCollectionServiceResource.initialise(Se
nsorCollectionServiceResource.java:42)
```

You need to ensure that the jdom.jar file is copied across your \$TOMCAT/wsrf/lib directory.

5.7. The SPS and Hibernate

To develop for the SPS you will need to install the Hibernate DB (HSQLDB 1.8.0). It is available from <http://hsqldb.org/>.

Part of the source code for the SPS is located in the Gridbusbroker project directory. Once this project is built it will generate several jar files, including: Gridbusbroker-core.jar, Gridbusbroker-globus.jar, Gridbusbroker-scs.jar, Gridbusbroker-ssh.jar, Gridbusbroker-workflow.jar. These jar files are then used by the SPS. They will need to be copied manually to the tomcat/webapps/wsrf directory in order for the SPS to function.

HSQLDB does not need to be installed if you simply want to run the SPS, although some of the tools in the installation may help in debugging. However, hibernate still needs to run, to execute hibernate open a terminal window and run startdb.sh located in Workspace/SensorWeb/Hibernate_DB.

If you wish to make changes to the SPS code, you may also need to change the hsqldb scripts. This will only be the case if you modify classes which are also maintained in the hsqldb. Such as WorkUnit, SCSJob, Task, etc.. To update the hsqldb database with new information do the following:

To create a new script file after the java source has been updated:

- Edit Gridbusbroker/ Broker.Hibernate.cfg.xml – Set the hibernate startup to be as a file by uncommenting

```
<property
name="hibernate.connection.url">jdbc:hsqldb:file:GBB.Persistence
e/gridbusbroker</property>
```

And commenting out:

```
<property
name="hibernate.connection.url">jdbc:hsqldb:hsq://localhost/pe
rsistence/gridbusbroker</property>
```

- Run org.gridbus.broker.util.PersistenceUtil
This should generate the following directory along with some files:
GBB.Persistence/gridbusbroker.*

- Go to your hsqldb install directory, mine is c:/workspace/hsqldb
 - a. Run demo/runManager.bat
 - b. Select “standalone server” mode
 - i. Set the file to be:
c:/workspace/gridbusbroker/GBB.Persistence/gridbusbroker
 - c. In the command window execute:
 - i. ‘SCRIPT gridbusbroker.script’
 - d. Copy the generated script to
c:/workspace/Sensorweb/Hibernate_DB and replace the old script

- Edit Gridbusbroker/ Broker.Hibernate.cfg.xml and this time comment out
- ```
<property
name="hibernate.connection.url">jdbc:hsqldb:file:GBB.Persistence
e/gridbusbroker</property>
```

And replace it with:

```
<property
name="hibernate.connection.url">jdbc:hsqldb:hsq://localhost/pe
rsistence/gridbusbroker</property>
```

- Run startdb.sh and you should now have an updated DB!

## 5.8. Adding Method to Existing WebServices

In order to add a new method to the existing webservises available under OSWA projects (for example SensorPlanningService) you have to follow the steps mentioned below:

1. Add method signature in *SensorPlanningService* interface under “org.sensorweb.service.sps” package.  
e.g,  
*public Element getSCSLocations() throws ServiceException;*
2. Add method implementation in *SensorPlanningService* class under “org.sensorweb.service.globus.sps.impl”

e.g,

```
public Element getSCSLocations() throws ServiceException {
 System.out.println("getSCSLocations() --START");
 Element element = null;
 ...
}
```

3. Next we have to add the method reference in “*SensorPlanningService.wsdl*” located in *OSWA* folder. Since our method is not taking an input parameters we will have to declare the *xsd:type* as

```
<xsd:element name="getSCSLocations" nillable="true">
 <xsd:complexType/>
</xsd:element>
```

4. Now we have to add request and response messages

e.g.

```
<message name="getSCSLocationsRequest">
 <part name="parameters" element="tns:getSCSLocations" />
</message>
<message name="getSCSLocationsResponse">
 <part name="parameters" type="tns1:Element" />
</message>
```

5. Lastly we have to add the operation item.

```
<operation name="getSCSLocations">
 <input message="tns:getSCSLocationsRequest" />
 <output message="tns:getSCSLocationsResponse" />
</operation>
```

6. Replicate these changes in “*SensorPlanningService.wsdl*” located in *OSWA\schema\SensorPlanningService\SensorPlanningService\_instance\*
7. Build the *OSWA* project and your changes can be seen in the stubs generated as a result of build.