

# TMB-IDE

current status & road ahead

Arni Magnusson

ADMB/TMB Developer Workshop

Seattle, 21 June 2016

# TMB-IDE

---

Emacs `tmb-mode` without the Emacs  
Version 1.6-2 (2015-10-28)

# TMB-IDE

## **Background**

overview, components, history

# TMB-IDE

## **Background**

overview, components, history

## **Current status**

three flavors: mode, ide, virtual

# TMB-IDE

## **Background**

overview, components, history

## **Current status**

three flavors: mode, ide, virtual

## **Road ahead**

comparison to admb-ide, easy to maintain

# Overview

Original idea behind TMB-IDE was to provide something similar to ADMB-IDE:

- 1 Quickly **install** TMB on a wide variety of machines, for example, in a workshop setting

# Overview

Original idea behind TMB-IDE was to provide something similar to ADB-IDE:

- 1 Quickly **install** TMB on a wide variety of machines, for example, in a workshop setting

Turns out to be more challenging than ADB-IDE ...

- should TMB-IDE install R? where?
- should TMB-IDE install the TMB package? where?
- should TMB-IDE install GCC? where?

# Overview

Original idea behind TMB-IDE was to provide something similar to ADMI-IDE:

- 1 Quickly **install** TMB on a wide variety of machines, for example, in a workshop setting

Turns out to be more challenging than ADMI-IDE ...

- should TMB-IDE install R? where?
- should TMB-IDE install the TMB package? where?
- should TMB-IDE install GCC? where?

- 2 Efficient and **full-featured environment** to develop TMB models



# Overview

Original idea behind TMB-IDE was to provide something similar to ADMB-IDE:

- 1 Quickly **install** TMB on a wide variety of machines, for example, in a workshop setting

Turns out to be more challenging than ADMB-IDE ...

- should TMB-IDE install R? where?
- should TMB-IDE install the TMB package? where?
- should TMB-IDE install GCC? where?

- 2 Efficient and **full-featured environment** to develop TMB models

This is an easier objective.

# Components

## TMB

R package + stuff

# Components

## TMB

R package + stuff

## GCC

compiler

## GDB

debugger

# Components

## TMB

R package + stuff

## GCC

compiler

## GDB

debugger

## Emacs

editor framework

# Components

## TMB

R package + stuff

## GCC

compiler

## GDB

debugger

## Emacs

editor framework

## TMB Mode

syntax colors

commands

- build
- run
- view
- template
- window layout

# Three alternative environments

## TMB Mode

Lightweight Emacs mode to develop TMB models

*Designed for users who already use Emacs as their main editor*

**tmb.el** (22 KB)

# Three alternative environments

## TMB Mode

Lightweight Emacs mode to develop TMB models

*Designed for users who already use Emacs as their main editor*

**tmb.el** (22 KB)

## TMB-IDE

Config file that lets Emacs behave like a basic editor

*The convenient features of TMB Mode for non-Emacs users*

**.emacs** (5 KB)

# Three alternative environments

## TMB Mode

Lightweight Emacs mode to develop TMB models

*Designed for users who already use Emacs as their main editor*

**tmb.el** (22 KB)

## TMB-IDE

Config file that lets Emacs behave like a basic editor

*The convenient features of TMB Mode for non-Emacs users*

**.emacs** (5 KB)

## Virtual TMB

Fully configured virtual machine

*VirtualBox disk that can be installed on any computer*

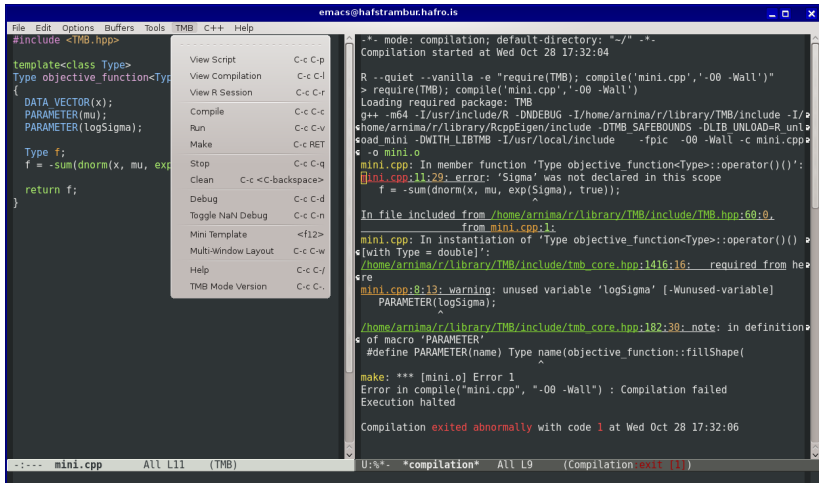
**VirtualTMB.vdi** (6 GB)



# First and current release

	<b>TMB Mode</b>	<b>TMB-IDE</b>	<b>Virtual TMB</b>
2015 Sep	1.0	1.1	1.1
2015 Oct		1.6	1.6
2015 Nov	3.1		

# TMB Mode



emacs@hafstrambur.hafro.is

File Edit Options Buffers Tools TMB C++ Help

```
#include <TMB.hpp>

template<class Type>
Type objective_function<Type>
{
    DATA VECTOR(x);
    PARAMETER(mu);
    PARAMETER(logSigma);

    Type f;
    f = -sum(dnorm(x, mu, exp(logSigma)));

    return f;
}
```

View Script C-c C-p  
View Compilation C-c C-l  
View R Session C-c C-r  
Compile C-c C-c  
Run C-c C-v  
Make C-c RET  
Stop C-c C-q  
Clean C-c <C-backspace>  
Debug C-c C-d  
Toggle NaN Debug C-c C-n  
Mini Template <f12>  
Multi-Window Layout C-c C-w  
Help C-c C-f  
TMB Mode Version C-c C-.

```
 -*- mode: compilation; default-directory: "~/ " -*-
Compilation started at Wed Oct 28 17:32:04

R --quiet --vanilla -e "require(TMB); compile('mini.cpp','-O0 -Wall')"
> require(TMB); compile('mini.cpp','-O0 -Wall')
Loading required package: TMB
g++ -m64 -I/usr/include/R -DNDEBUG -I/home/arnima/r/library/TMB/include -I/home/arnima/r/library/RcppEigen/include -DTMB_SAFEBOUNDS -DLIB_UNLOAD=RUNLOAD -DWITH_LIBTMB -I/usr/local/include -fpic -O0 -Wall -c mini.cpp
-o mini.o
mini.cpp: In member function 'Type objective_function<Type>::operator()()':
mini.cpp:11:29: error: 'Sigma' was not declared in this scope
    f = -sum(dnorm(x, mu, exp(Sigma), true));
                             ^
In file included from /home/arnima/r/library/TMB/include/TMB.hpp:68:0,
                 from mini.cpp:1:
mini.cpp: In instantiation of 'Type objective_function<Type>::operator()()':
[with Type = double]:
/home/arnima/r/library/TMB/include/tmb_core.hpp:1416:16: required from here
mini.cpp:8:13: warning: unused variable 'logSigma' [-Wunused-variable]
    PARAMETER(logSigma);
                ^
/home/arnima/r/library/TMB/include/tmb_core.hpp:182:30: note: in definition of macro 'PARAMETER'
#define PARAMETER(name) Type name(objective_function::fillShape(
                          ^
make: *** [mini.o] Error 1
Error in compile("mini.cpp", "-O0 -Wall") : Compilation failed
Execution halted

Compilation exited abnormally with code 1 at Wed Oct 28 17:32:06
```

U:-- mini.cpp All L11 (TMB) U:-- \*compilation\* All L9 (Compilation:exit (1))

tmb.el (22 KB)

## TMB-IDE

The screenshot shows the TMB-IDE Emacs interface. The left pane displays the source code for `mini.cpp`, which includes a template class `Type` with an `objective_function` member and a `sum` function. The right pane shows the compilation output, indicating an error in `mini.cpp` at line 11, column 29: `'Sigma' was not declared in this scope`. The error is linked to the definition of `logSigma` in `tmb_core.hpp`. The status bar at the bottom shows the file `mini.cpp` at line 11, column 28, with a compilation error (code 1) at line 9, column 0.

```
File Edit Options Buffers Tools TMB C++ Help
#include <TMB.hpp>

template<class Type>
Type objective_function<Type>
{
    DATA VECTOR(x);
    PARAMETER(mu);
    PARAMETER(logSigma);

    Type f;
    f = -sum(dnorm(x, mu, exp(logSigma)));

    return f;
}

View Script <f11>
View Compilation
View R Session
Compile <f8>
Run <f9>
Make
Stop
Clean <f7>
Debug <f10>
Toggle NaN Debug
Mini Template <f12>
Multi-Window Layout
Help <f1>
TMB Mode Version

-.*. mode: compilation; default-directory: "~/".
Compilation started at Wed Oct 28 17:35:26

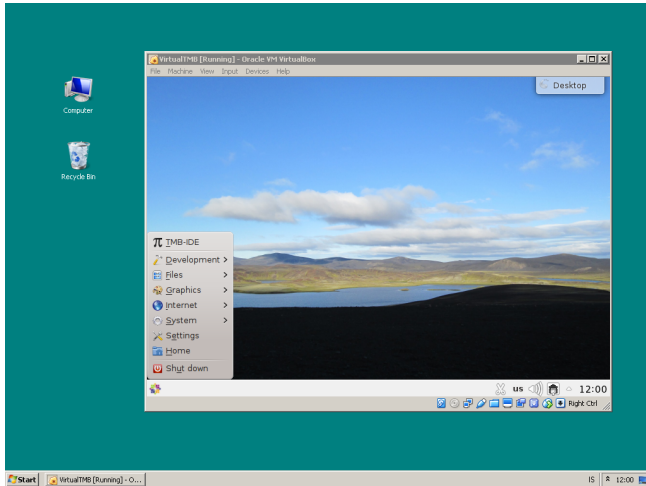
R --quiet --vanilla -e "require(TMB); compile('mini.cpp', '-fno-gnu-unique ->
> require(TMB); compile('mini.cpp', '-fno-gnu-unique -O0 -Wall')
Loading required package: TMB
g++ -m64 -I/usr/include/R -DNDEBUG -I/home/arnima/r/library/TMB/include -I>
mini.cpp: In member function 'Type objective_function<Type>::operator(){}':
mini.cpp:11:29: error: 'Sigma' was not declared in this scope
    f = -sum(dnorm(x, mu, exp(Sigma), true));
                           ^
In file included from /home/arnima/r/library/TMB/include/TMB.hpp:60:0,
from mini.cpp:1:
mini.cpp: In instantiation of 'Type objective_function<Type>::operator(){}':
/home/arnima/r/library/TMB/include/tmb_core.hpp:1416:16: required from he
mini.cpp:8:13: warning: unused variable 'logSigma' [-Wunused-variable]
    PARAMETER(logSigma);
    ^
/home/arnima/r/library/TMB/include/tmb_core.hpp:182:30: note: in definition
#define PARAMETER(name) Type name(objective_function::
    ^
make: *** [mini.o] Error 1
Error in compile("mini.cpp", "-fno-gnu-unique -O0 -Wall") :
Compilation failed
Execution halted

Compilation exited abnormally with code 1 at Wed Oct 28 17:35:28

-:-- mini.cpp All (11,28) (TMB) U:~*.*. *compilation* All (9,0) (Compilation:exit [1])
```

`.emacs` (5 KB)

# Virtual TMB



VirtualTMB.vdi (6 GB)

# Comparison to ADMB-IDE

## TMB Mode

Similar to ADMB Mode: lightweight standard Emacs mode, provides syntax highlighting and convenient commands

# Comparison to ADBM-IDE

## TMB Mode

Similar to ADBM Mode: lightweight standard Emacs mode, provides syntax highlighting and convenient commands

## TMB-IDE

Comparison to ADBM-IDE: just one config file, no installer

# Comparison to ADBM-IDE

## TMB Mode

Similar to ADBM Mode: lightweight standard Emacs mode, provides syntax highlighting and convenient commands

## TMB-IDE

Comparison to ADBM-IDE: just one config file, no installer

## Virtual TMB

Analogous to ADBM-IDE, but a virtual machine (6 GB) instead of an installer (0.2 GB)

Quick to start a workshop, practical introduction to Linux, but requires maintenance and learning from the user

## Comparison to ADMB-IDE

The need for TMB-IDE is not as urgent as ADMB-IDE:

- Without ADMB-IDE, it was challenging to set up ADMB and GCC  
*It's not that difficult to set up TMB*



## Comparison to ADMB-IDE

The need for TMB-IDE is not as urgent as ADMB-IDE:

- Without ADMB-IDE, it was challenging to set up ADMB and GCC  
*It's not that difficult to set up TMB*
- Without ADMB-IDE, the majority of ADMB workshop participants used Notepad and a Dos shell  
*TMB can be used in the same environment as other R work (RGui, RStudio, Emacs, etc.)*

## Comparison to ADMB-IDE

The need for TMB-IDE is not as urgent as ADMB-IDE:

- Without ADMB-IDE, it was challenging to set up ADMB and GCC  
*It's not that difficult to set up TMB*
- Without ADMB-IDE, the majority of ADMB workshop participants used Notepad and a Dos shell  
*TMB can be used in the same environment as other R work (RGui, RStudio, Emacs, etc.)*

Nevertheless, Emacs with TMB Mode (and TMB-IDE) provides dedicated TMB functionality beyond what any other editor can offer

## Comparison to ADMB-IDE

The need for TMB-IDE is not as urgent as ADMB-IDE:

- Without ADMB-IDE, it was challenging to set up ADMB and GCC

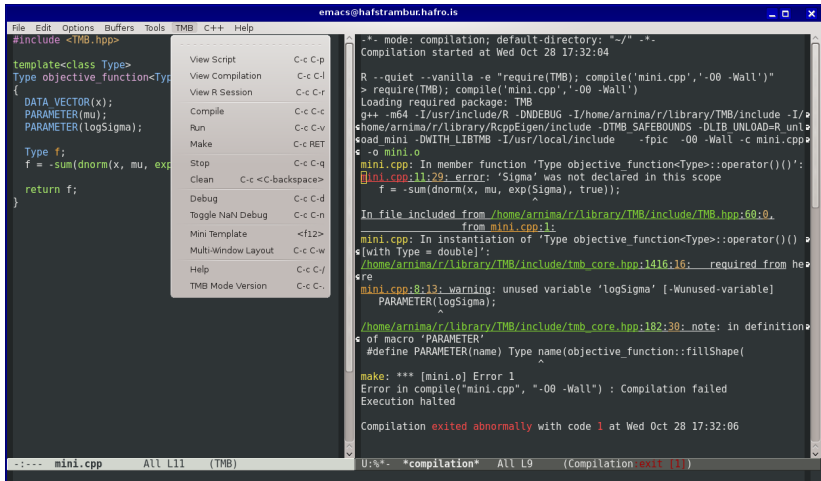
*It's not that difficult to set up TMB*

- Without ADMB-IDE, the majority of ADMB workshop participants used Notepad and a Dos shell

*TMB can be used in the same environment as other R work  
(RGui, RStudio, Emacs, etc.)*

Nevertheless, Emacs with TMB Mode (and TMB-IDE) provides dedicated TMB functionality beyond what any other editor can offer

It's what Kasper and most of the core TMB developers are using



```
File Edit Options Buffers Tools TMB C++ Help
#include <TMB.hpp>

template<class Type>
Type objective_function<Type>
{
    DATA VECTOR(x);
    PARAMETER(mu);
    PARAMETER(logSigma);

    Type f;
    f = -sum(dnorm(x, mu, exp
    return f;
}

View Script C-c C-p
View Compilation C-c C-l
View R Session C-c C-r
Compile C-c C-c
Run C-c C-v
Make C-c RET
Stop C-c C-q
Clean C-c <C-backspace>
Debug C-c C-d
Toggle NaN Debug C-c C-n
Mini Template <f12>
Multi-Window Layout C-c C-w
Help C-c C-f
TMB Mode Version C-c C-.
```

```
 -*- mode: compilation; default-directory: "~/ " -*-
Compilation started at Wed Oct 28 17:32:04

R --quiet --vanilla -e "require(TMB); compile('mini.cpp','-O0 -Wall')"
> require(TMB); compile('mini.cpp','-O0 -Wall')
Loading required package: TMB
g++ -m64 -I/usr/include/R -DNDEBUG -I/home/arnima/r/library/TMB/include -I/home/arnima/r/library/RcppEigen/include -DTMB_SAFEBOUNDS -DLIB_UNLOAD=R unload_mini -DWITH_LIBTMB -I/usr/local/include -fpic -O0 -Wall -c mini.cpp
g++ -o mini.o
mini.cpp: In member function 'Type objective_function<Type>::operator()()':
mini.cpp:11:29: error: 'Sigma' was not declared in this scope
    f = -sum(dnorm(x, mu, exp(Sigma), true));
                           ^
In file included from /home/arnima/r/library/TMB/include/TMB.hpp:60:0,
                 from mini.cpp:1:
mini.cpp: In instantiation of 'Type objective_function<Type>::operator()() [with Type = double]':
/home/arnima/r/library/TMB/include/tmb_core.hpp:1416:16:   required from here
mini.cpp:8:13: warning: unused variable 'logSigma' [-Wunused-variable]
    PARAMETER(logSigma);
    ^
/home/arnima/r/library/TMB/include/tmb_core.hpp:182:30: note: in definition of macro 'PARAMETER'
#define PARAMETER(name) Type name(objective_function::fillShape(
                           ^
make: *** [mini.o] Error 1
Error in compile("mini.cpp", "-O0 -Wall") : Compilation failed
Execution halted

Compilation exited abnormally with code 1 at Wed Oct 28 17:32:06
```

```
U:--- mini.cpp All L11 (TMB)
U:--- *compilation* All L9 (Compilation.exit (1))
```

# Easy to maintain

Not a big bundle, just three files to maintain:

① `tmb.el`

Relatively frequent updates

② `.emacs`

Occasional updates, paired with a particular version of `tmb.el`

③ `VirtualTMB.vdi`

Rarely updated – mainly useful for experimenting, improving performance on a Windows computer, and as a last resort on computers with a restricted user account

# Download

## TMB Mode

<https://github.com/kaskr/adcomp/tree/master/emacs>

also linked from

<http://www.hafro.is/~arnima/tmb.html>

# Download

## TMB Mode

<https://github.com/kaskr/adcomp/tree/master/emacs>

also linked from

<http://www.hafro.is/~arnima/tmb.html>

## TMB-IDE

<http://www.hafro.is/~arnima/tmb.html>

# Download

## TMB Mode

<https://github.com/kaskr/adcomp/tree/master/emacs>

also linked from

<http://www.hafro.is/~arnima/tmb.html>

## TMB-IDE

<http://www.hafro.is/~arnima/tmb.html>

## Virtual TMB

<http://www.hafro.is/~arnima/tmb.html>