

A guide to robotpkg

Anthony Mallet — `anthony.mallet@laas.fr`

September 30, 2024

Copyright ©2006-2011,2013 LAAS/CNRS.
Copyright ©1997-2010 The NetBSD Foundation, Inc.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE NETBSD FOUNDATION, INC. AND CONTRIBUTORS “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE FOUNDATION OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

2.5	Creating binary packages for everything	22
2.5.1	Initial setup	22
2.5.2	Running bulk builds	23
2.5.3	Generating pretty reports	24
2.5.4	Automated bulk builds	24
3	The robotpkg developer's guide	25
3.1	Package files, directories and contents	25
3.1.1	Makefile	25
3.1.2	distinfo	28
3.1.3	PLIST	28
3.1.4	patches/*	29
3.2	General operation	29
3.2.1	Adding build options to a package	29
3.2.2	Customizing the PLIST	31
3.2.3	Customizing the semi-automatic PLIST generation . .	33
3.2.4	Incrementing versions when fixing an existing package	33
3.2.5	Substituting variable text in the package files	34
3.3	The build phase	34

Introduction

1.1 What is robotpkg?

The robotics research community has always been developing a lot of software, in order to illustrate theoretical concepts and validate algorithms on board real robots. A great amount of this software was made freely available to the community, especially for Unix-based systems, and is usually available in form of the source code. Therefore, before such software can be used, it needs to be configured to the local system, compiled and installed. This is exactly what The Robotics Packages Collection (robotpkg) does. robotpkg also has some basic commands to handle binary packages, so that not every user has to build the packages for himself, which is a time-costly, cumbersome and error-prone task.

The robotpkg project was initiated in the [Laboratory for Analysis and Architecture of Systems](#) (CNRS/LAAS), France. The motivation was, on the one hand, to ease the software maintenance tasks for the robots that are used there. On the other hand, roboticists at CNRS/LAAS have always fostered an open-source development model for the software they were developing. In order to help people working with the laboratory to get the LAAS software running outside the laboratory, a package management system was necessary.

Although robotpkg was an innovative project in the robotics community (it started in 2006), a lot of general-purpose software packages management systems were readily available at this time for a great variety of Unix-based systems. The main requirements that we wanted robotpkg to fulfill were listed and the best existing package management system was chosen as a starting point. The biggest requirement was the capacity of the system to adapt to the nature of the robotic software, being available mostly in form

of source code only (no binary packages), with unfrequent stable releases. robotpkg had thus to deal mostly with source code and automate the compilation of the packages. The system chosen as a starting point was **The NetBSD Packages Collection** (pkgsrc). robotpkg can be considered as a fork of this project and it is still very similar to pkgsrc in many points, although some simplifications were made in order to provide a tool geared toward people that are not computer scientists but roboticists.

Due to its origins, robotpkg provides many packages developed at LAAS. It is however not limited to such packages and contains, in fact, quite some other software useful to roboticists. Of course, robotpkg is not meant to be a general purpose packaging system (although there would be no technical restriction to this) and will never contain widely available packages that can be found on any modern Unix distribution. Yet, robotpkg currently contains roughly one hundred and fifty packages, including:

- architecture/genom - The LAAS Generator of Robotic Components
- architecture/openrtm - The robotic distributed middleware from AIST, Japan
- middleware/yarp - The “other”, yet famous, robot platform
- ...just to name a few.

1.2 Why robotpkg?

robotpkg provides the following key features:

- Easy building of software from source as well as the creation and installation of binary packages. The source and latest patches are retrieved from a master download site, checksum verified, then built on your system.
- All packages are installed in a consistent directory tree, including binaries, libraries, man pages and other documentation.
- Package dependencies, including when performing package updates, are handled automatically.
- The installation prefix, acceptable software licenses and build-time options for a large number of packages are all set in a simple, central configuration file.
- The entire framework source (not including the package distribution files themselves) is freely available under a BSD license, so you may extend and adapt robotpkg to your needs, like robotpkg was adapted from pkgsrc.

