System Development Environment

More than a distribution

T2 is not a common Linux distribution — it is a flexible open source build kit for Linux-based systems supporting the creation of custom distributions with bleeding edge technology. Whether embedded systems with low memory footprint, secure and highly specialised server setups or complete desktop systems featuring a rich set of applications — the modular framework of T2 can easily be adapted to fit your needs.
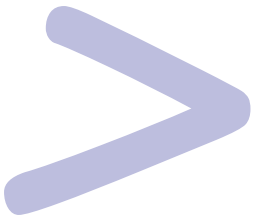
Philosophy

A community driven development process ensures rapid evolution of the tool kit. Both free and professional users have been attracted by the GPL'ed build system and created a large set of extensions and packages for T2. Industrial spin-off projects benefit from the steadily growing user base of the project by gaining access to an enormous amount of resources and innovations.

Standard compliant

Packages are kept clean and distribution specific patches are avoided. Furthermore T2 adheres to the Filesystem Hierarchy Standard (FHS) and Linux Standard Base (LSB). Package configurations are kept simple with focus on security. Inconsistencies and historical cruft is evaded by making as few assumptions about the past as possible.

The Build System

T2 features an automated build system which manages the whole compilation process including an optional installer CD creation. After initial creation of the build-toolchain, all packages are built inside a sandbox environment to monitor installed files and dependencies automatically. Optionally a cross-build between different architectures can be done, for example to use a fast machine to build a system for a thin client. The build system can also modify the execution of various programs to provide a generic way to transform control compiler flags and file manipulations.

ExactCODE

More than a distribution

- • Easy and flexible system adaption
- • Excellent hardware auto-detection (hotplug)
- • Innovative package management
- • Automated build-system
- • Optimization for embedded systems & PDA
- • Solid "System Development Environment"
- • Live, installation CD or ROM image
- • Supports all major processor architectures
- • Allows cross compilation
- • Professional support

Optimised systems with T2 for embedded systems, highly secure servers, live-CDs and desktops

Framework

To create individual custom build target definitions, the T2 framework allows to be used and to customize their configuration — including the package selection and C library. The build system merges all those definitions together and generates a build process. Dependencies of packages are traced and resolved during the compilation process to take away this burden from the user. There is full controle of the build output such as to create a installable CD or ROM images for embedded use or distributing the compiled system over the network.

Portability

Due to the nature of the clean source packages and its automatic build system, T2 is highly portable. It includes support for ARM, AVR, Alpha, Blackfin, HPPA, HPPA64, IA-64, MIPS, MIPS64, PowerPC, PowerPC64, SPARC, Ultra SPARC, SuperH and x86, x86-64. The user can add definitions for other architectures to the SDE if desired.

Optimization

Since the automatic build system controls the whole build, it can transform compiler arguments used by the various packages on-the-fly. This allows adapting the optimization for the specified CPU platform, thus accelerating your product over its current limits.

Packaging system

Unlike competing packaging systems like RPM or DEB, the format used in T2 does not require the developer to write code. Instead, an human readable ASCII tag format is used to specify the package parameters — the automated build system does the work.

Ready to start

T2 already ships with more than 2600 up-to-date packages providing a solid base for your product, including the X.Org windowing system, major desktop environments like KDE, GNOME and many other server and security additions and many specialized embedded packages.

Features

Due to its flexibility and scalability T2 is an interesting system development environment for embedded systems. But in many cases the goal of the user is just creating a good custom Linux distribution — it is a T2 philosophy to deliver vanilla packages and standard compliant

installations. Surely, an excellent product or distribution needs more than kernel, libraries and software. T2 comes with many features which make it easy to maintain, highly configurable and comfortable system.

Hardware auto-detection

The hardware detection system is exclusively designed around the Linux hotplug mechanism. This way the same code-path is used for devices present at boot-up or plugged into the system at run time yielding exactly the same configuration whenever the user decides to connect the devices.

Network configuration

For network configuration a modular framework is included in T2 supporting: profiles, basic IP configuration, multiple interfaces, static routing, stateful firewalling, W-LAN, PPP incl. analogue modems, PPPoE (cable and DSL) as well as CSD and GPRS for use with mobile phones.

Administration

T2 features an Install CD target with a full installer supporting partitioning hard-drives, creating file-systems, defining mount points, package selection and much more. The configuration tool STONE allows administrators to configure system services, network and behaviour. STONE is compatible with the system's standard config files, manual changes in those files are mostly preserved.

A good choice

T2 gives a good framework to develop and build custom Linux based systems. Instead of starting to build everything from scratch, one can benefit from all the existing architectures and a full package repository. But there are even more advantages maintaining the product: A custom target's build can be instantly reproduced and it is easy to perform slight modifications and rebuild without having to do everything again "manually". When updating binary packages, there is a backup mode for modified files, so no config file is accidentally lost. Together the support for different compilers and interesting options like dietLibC and uClibC and optional, space-optimized utilities, T2 gives interesting perspectives for the development of embedded systems.
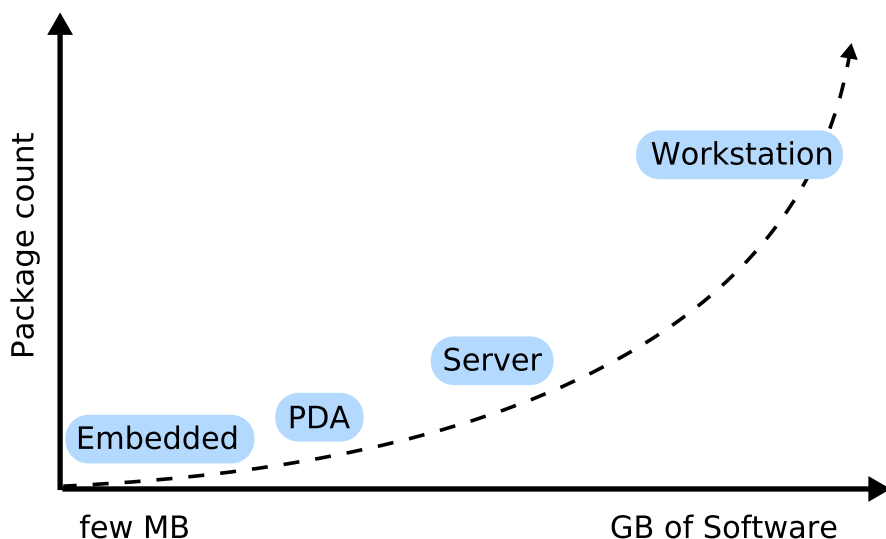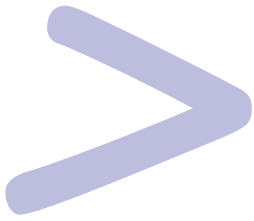
Choose your desired architecture and setup the build mode to generate your favoured system. Benefit from the huge set of maintained targets and packages that are available within T2. After the build, T2 can create a set of boot-able CDs or generate a ROM-Image to distribute the system.

ExactCODE

>

At a glance

- Automated build-system
- Integrated sandbox environment
- Generic transformation of program arguments
- Supports all major processor architectures
- Allows cross compilation
- Distributed compilation (distcc, icecream)
- Selection of Compiler and C-library
- More than 2600 up-to-date packages
- Live, installation CD or ROM image

Quality Assurance

During the development of T2 Linux, measurements are used to assure the high quality level defined by the developers. The sandbox environment is used to review the content installed by the packages, as well as the preconditions needed to build the package. In addition, a regression tester automatically verifies cross builds for the supported architectures and tracks the build status for every package included in T2. Since the system builds itself up in the sandbox environment it is guaranteed to be stable to self-hosting — to bootstrap itself.

Why T2

With all the points outlined above, T2 offers a solid base and a flexible framework to build a custom system specialized for your needs and for integration with your custom applications. The state-of-the-art technology reduces the development time and thus the time to market.

Many end users and system administrators enjoy T2 with its clean and tested packages for daily home or office use.

With developers all around the world, T2 is best suited for commercial scale products, especially in the embedded and high performance clustering market.

What can a high-performance and highly secure Linux do for your system? It's time to find out — try T2 Linux today. Learn more at: www.t2-project.org

ExactCODE