

T2EX



Milestones of TRON/T-Kernel

T2EX

Realized extension in the best way for T-Kernel

T-Kernelに最適な形での機能拡張を実現

- While utilizing the small footprint (resource saving) and the robustness (stable operation) of T-Kernel, its advantages over general information-system OS, files and networks have been added in consideration of performance and security
- 情報系OSと比べて軽量（省資源）、堅牢（安定動作）といったT-Kernelのメリットを活かしつつ、性能面や安全面にも配慮しながら、ファイル、ネットワークなどの機能を追加

T-Kernel + T2EX

- Offered for free as software extension of T2
- T2の追加機能（アドオン）として提供

Basic concept of T2EX



Maintains real-time processing

リアルタイム性を堅持

- Used for task-based programming on T-Kernel
- Designed to be used with T-Kernel API
- T-Kernelのタスクベースでのプログラミングが前提
- T-Kernel APIとの併用を前提とした設計



Added network communication function

ネットワーク通信機能の追加

- Offers a TCP/IP protocol stack
- Offers a file system
- TCP/IPプロトコルスタックを提供
- ファイルシステムを提供



Security-conscious design

安全面に配慮した設計

- Offers an efficient memory protection function that does not necessarily need virtual memory
- Ensures all API functions of T2EX are thread-safe
- 仮想記憶によらない効率的なメモリ保護機能を提供
- 全てのAPI関数におけるスレッド安全性を保障

Open × Open = ∞

Moreover, The API of T2EX is designed with the consideration of portability from POSIX

POSIXからの移植性を考慮したAPI設計

Portability

Utilizes software assets



- 移植性
- ソフトウェア資産の有効活用

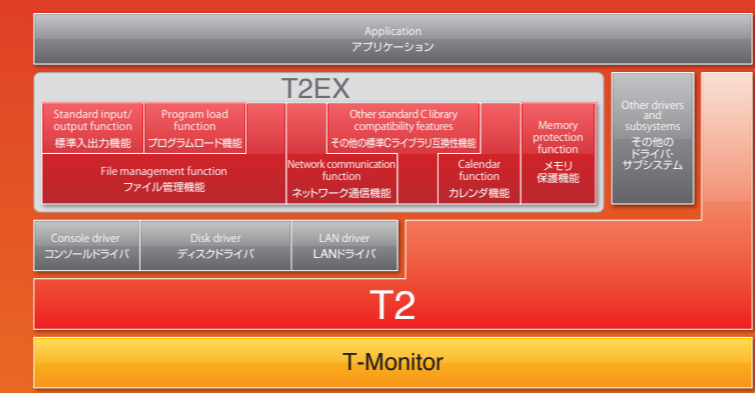
Education

Utilizes already-acquired knowledge



- 教育
- 既得知識の有効活用

Features offered by T2EX



Towards the most powerful embedded OS with T2EX
T2EXで最強の組み込みOSへ

Programming with T2EX

Sample Programming

プログラミング例

```
void processTask( INT stacd, VP exinf )
{
    :
    while( 1 ){
        ercd = tk_rcv_mbf( mbfid, &target, TMO_FEVR );
        if( ercd < 0 ){
            printf( "Receive Error(%08X)\n", ercd );
            :
        }
        tfd = fs_open( target, O_RDONLY );
        if( tfd < 0 ){
            printf( "File Open Error(%08X)\n", tfd );
            :
        }
        ercd = fs_read( tfd, buf, BUFSZ );
        if( ercd < 0 ){
            printf( "File Read Error(%08X)\n", ercd );
            :
        }
        fs_close( tfd );
        ercd = tk_wai_sem( semid, 1, TMO_FEVR );
        if( (ercd == E_RLWAI) || (ercd == E_DISWAI) ){
            break; /* Exit Task */
        } else if( ercd < 0 ){
            :
        }
        outputToDevice( buf, BUFSZ );
        tk_sig_sem( semid, 1 );
    }
    tk_ext_tsk();
}

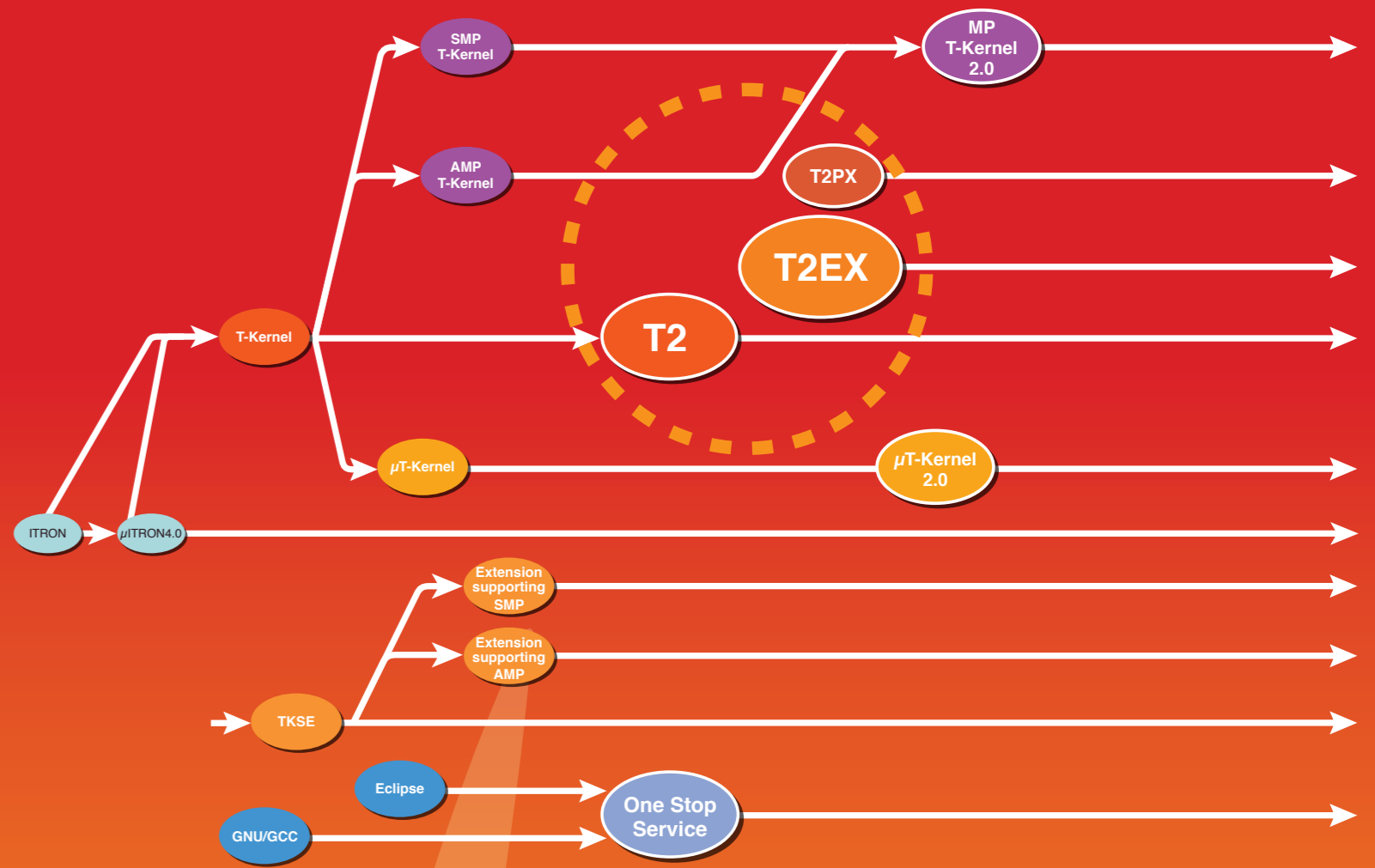
```

- T2 API function T2のAPI関数
- T2EX API function T2EXのAPI関数

API offered by T2EX (excerpted)

T2EXが提供するAPI(一部)

- File management function
ファイル管理機能
fs_open
fs_close
fs_read
fs_write
fs_create
fs_unlink
fs_mkdir
fs_rmdir
fs_rmdir
:
- Calendar function
カレンダー機能
dt_localtime
dt_strftime
dt_mktime
dt_gmtime
dt_setsyatz
dt_getsyatz
dt_tzset
:
- Program load function
プログラムロード機能
pm_load
pm_loadpgg
pm_status
pm_unload
:
- Network communication function
ネットワーク機能機能
so_accept
so_bind
so_close
so_connect
so_listen
so_read
so_write
so_send
so_sendto
:
- Standard input/output function
標準入出力機能
printf
scanf
fopen
fclose
fread
fwrite
fprintf
scanf
:
- Other standard C library compatibility features
標準Cライブラリ互換機能
Functions defined in various header files, etc.
以下の各ヘッダファイルなどで定義されている関数群
assert.h
ctype.h
limits.h
math.h
stdarg.h
stdlib.h
stdint.h
string.h
:





TRON Ecosystem

Topmost Share
For 13 consecutive years

Many TRON Users
in the world

13年連続でNo.1のシェアを誇る

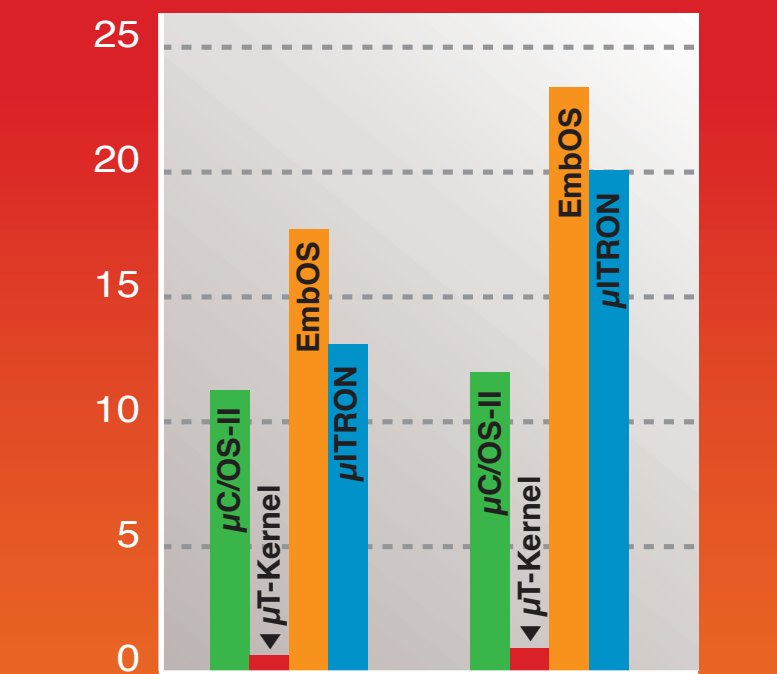
世界中に多くのTRONユーザ

Real-time Performance

リアルタイムパフォーマンス

- Adopts preemptive priority scheduling system to realize high real-time performance
 - Making it easy to build hard real-time systems
- プリエンティブな優先度ベーススケジューリング方式を採用し、高いリアルタイム性能を実現
 - ハードリアルタイムシステム実現を容易に

Source: T.Nguyen, B. Anh, S. Tan, "REAL-TIME OPERATING SYSTEMS FOR SMALL MICROCONTROLLERS", IEEE MICRO Sept./Oct. 2009 pp.30-45



- With the system calls that use microsecond resolution, fully utilizes high-end CPU
- Physical timer function improves the development efficiency and the portability of the program that operates the hardware timer of the MCU
- マイクロ秒単位の指定可能なシステムコールにより、ハイエンドCPUの性能をフル活用
- 物理タイマ機能により、CPUのハードウェアタイマを操作するプログラムの開発効率、移植性を向上

Small Memory Footprint

スモールメモリフットプリント

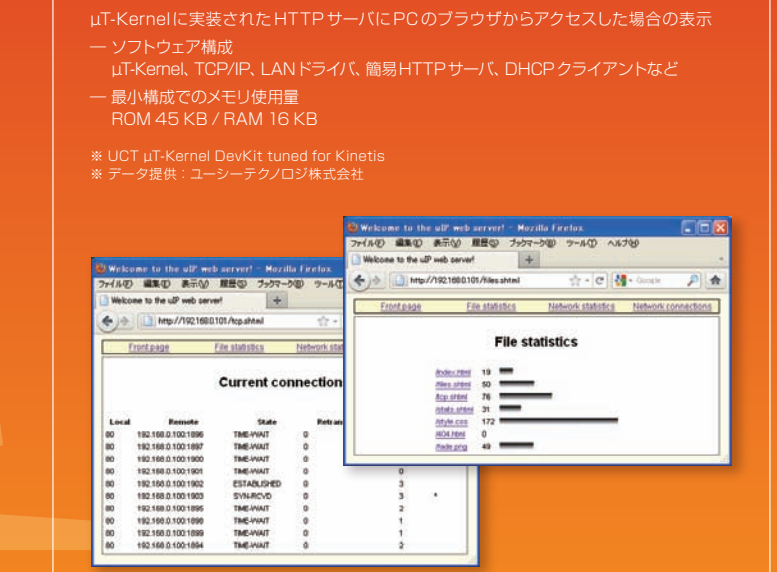
- Rich API is offered on small memory footprint
 - The number of available API entry points: 274
 - Code Size: 99 KB
 - RAM Size: 102 KB
- Freely modifiable and easy to downsize according to the target system needs
 - 豊富なAPIとSmall Memory Footprintを両立
 - 提供するAPI数: 274
 - Code Size: 99 KB
 - RAM Size: 102 KB
 - 自由に改変可能で、ターゲットシステムに合わせた小型化が容易

Example implementation of an HTTP server using μT-Kernel

An HTTP server implemented under μT-Kernel accessed from a PC browser: screen images below

- Software configuration: μT-Kernel, TCP/IP, LAN driver, simple HTTP server, DHCP client, etc.
- Minimum memory configuration: ROM 45 KB / RAM 16 KB

* UCT μT-Kernel DevKit tuned for Kinels
* Data: Courtesy of Ubiquitous Computing Technology Corporation



Energy Saving

省電力

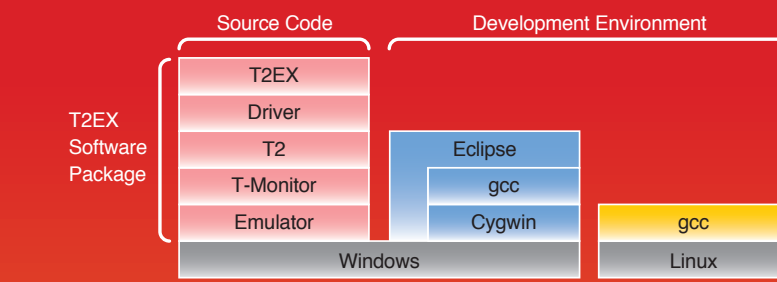
- Energy saving effect of replacing a PC-based system using a dedicated embedded computer
 - Power usage can be lowered by a factor of 1/100 to 1/10
- No energy purchase by using natural energy such as sun light and energy saving functions of T2
 - T2の省エネルギー機能の進化と、太陽光等自然エネルギーの組合せでゼロ・エナジー実現可能!
 - T2の省電力機能を強化し、拡張機能と連携することで、より高度な省電力機能を実現

One Stop Service

You can start development right away with a collection of tools

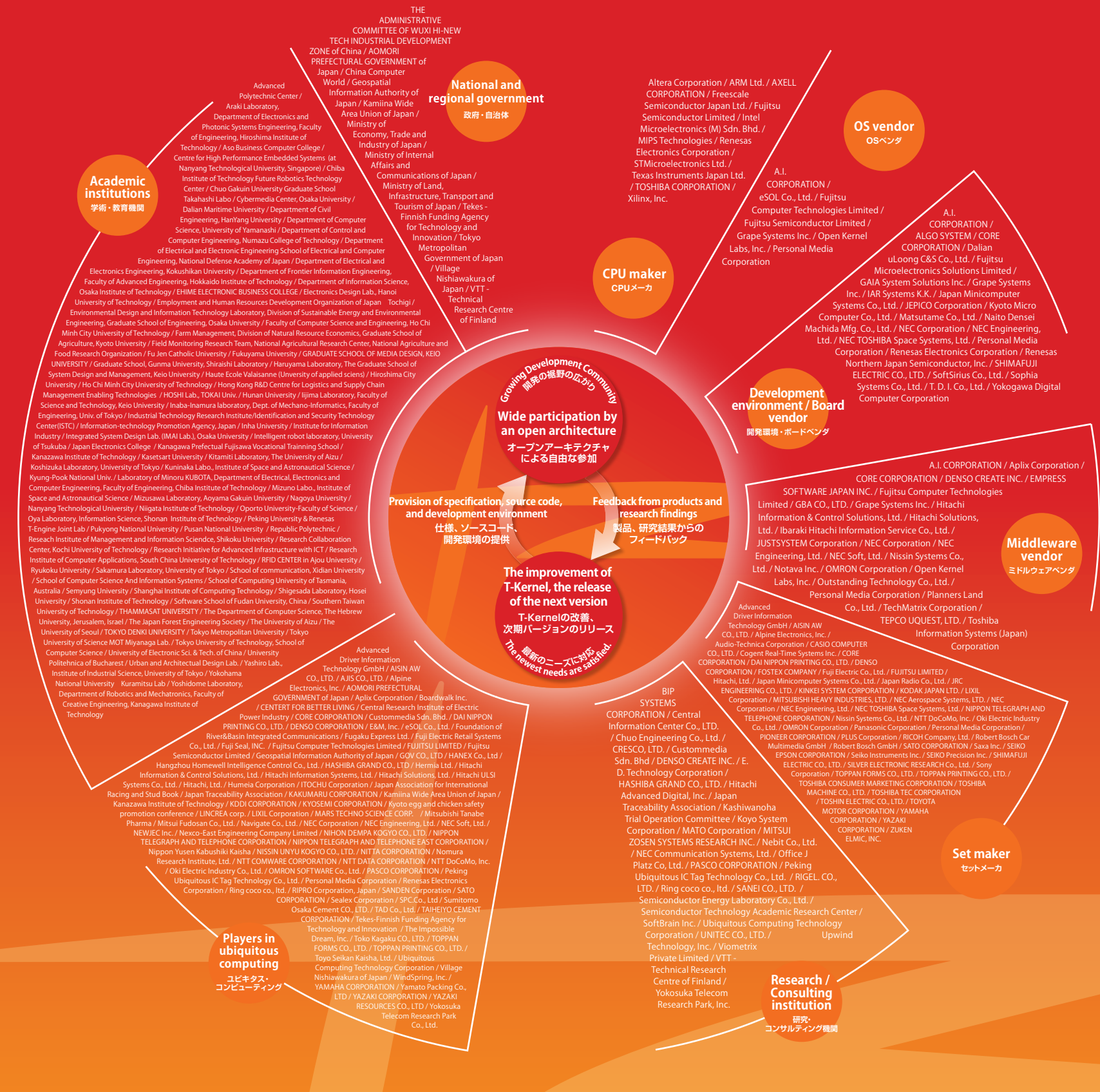
一連のツールですぐに開発を始められる

- All source code of T2 + T2EX
- Source code of sample device drivers
- Source code of monitor program T-Monitor
- Integrated development environment (compiler, debugger, etc.) and emulator
- T2 + T2EXの全ソースコード
- サンプルとなる各種デバイスドライバのソースコード
- モニタプログラムT-Monitorのソースコード
- 統合開発環境 (コンパイラ、デバッガなど) エミュレータ



Open License

- Licensed under flexible T-License 2.0
 - Original source code can be redistributed
 - Source code can be modified and distributed
 - Developers can set conditions to distribute the modified source code
- T-Kernel Traceability Service
 - Offers traceability function with ucode
 - Solves conflicts between versions that often arise in open source software
- 自由度の高いT-License 2.0の下で配布
 - オリジナルソースの再配布が可能
 - 改変したソースの配布が可能
 - 改変したソースを配布する条件を開発者が指定可能
- T-Kernel Traceability Service
 - ucodeによるトレーサビリティ機能を提供
 - オープンソースソフトウェアでおこりがちなバージョン間の不整合問題を解決

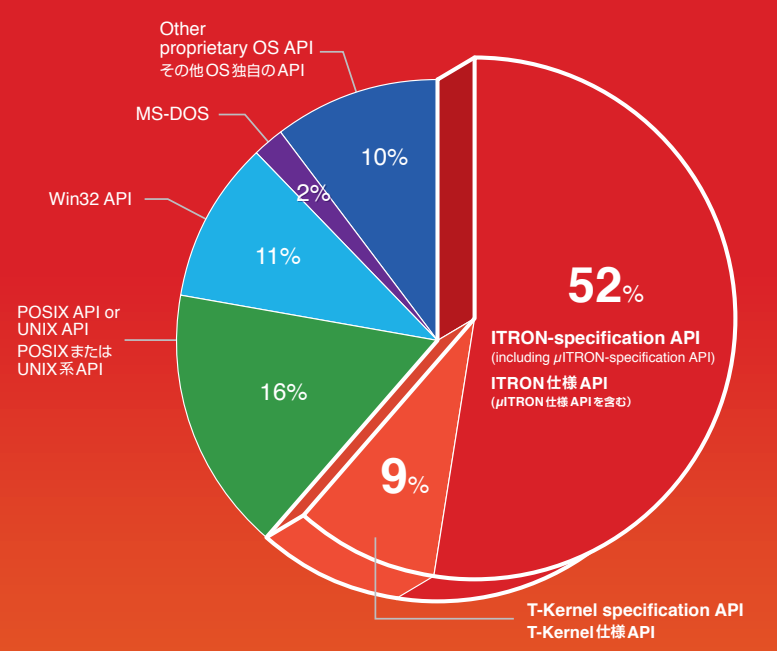


TRON Project
Directed by Ken Sakamura

Usage in the RTOS field of TRON operating systems

RTOS分野での利用実績

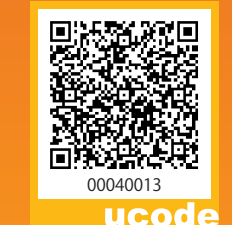
- No. 1 in "API of embedded operating systems" category since the start of survey in 1999
- 「組み込みOSのAPI」1999年調査開始以来連続No.1



Source: 2010-2011 user questionnaire results (242 answers) at Embedded Technology Exhibitions 2010年, 2011年 Embedded Technology / 組み込み総合技術展 利用アンケートNo.1 総数: 242



- TRON has many users in the world, and is used in many products.
- TRONは世界中に多くのユーザがあり、多様な製品で使われています。



www.t-engine.org