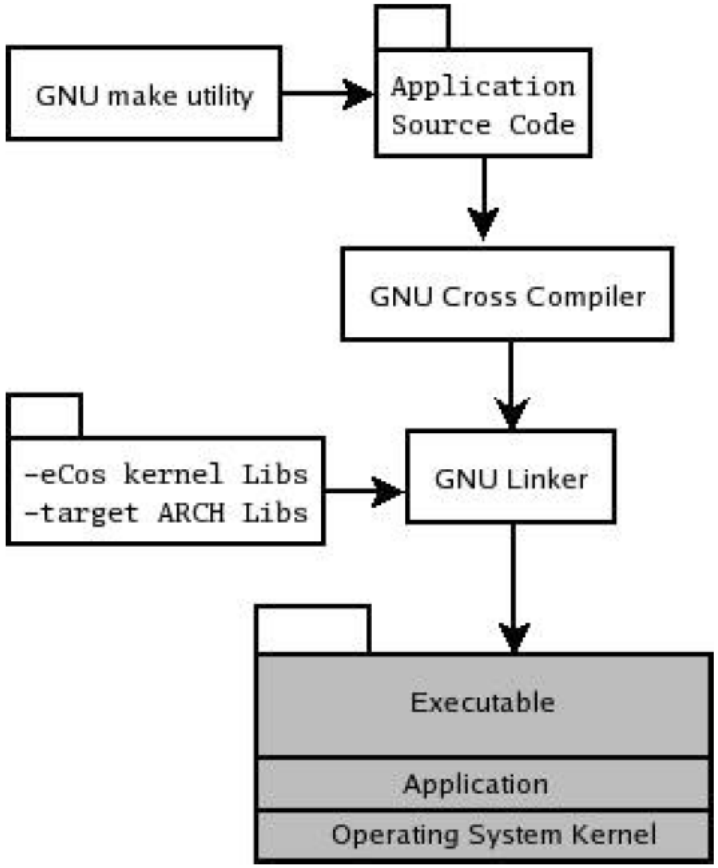
eCos

James

Overview

- eCos, an ASOS (application specific operating system)



Main features

- Free and open source
- Small footprint
- Highly configurable
  - Each part of the system is a package
  - Configuration tool
- Easily portable
  - Hardware abstraction layer
- Code reusable
  - Standard APIs
- A static framework

eCos kernel

- Has kernel mode, but no user mode
  - eCos and the application run in supervisor mode
- Does not include memory allocation or device drivers
- Support multiple threads or single thread
- A choice of schedulers
  - Bitmap
  - Multiple level queue
    - Optionally support timeslicing
- Manipulating threads
  - Priorities
  - Creation, etc.
- A range of synchronization primitives
  - Mutex, condition variable, semaphore, and etc.

Hardware abstraction layer (HAL)

- Implemented in C and assembly language
- Possible to bypass HAL for better performance but lose portability
- Sub-Modules
  - Architecture
  - Variant
  - Board

Device driver

- Configured into the system just like all other components
- Drivers can be initialized when actual usage starts
- Written in C
- Access hardware via an I/O “handle”
- Support run-time configuration

Memory & Storage

- eCos memory pools
  - Variable size allocation pools
  - Fixed size allocation pools
  - stdlib malloc pools
- File system
  - POSIX compilant file and IO operations
  - Can support multiple FS

Other eCos features

- TCP/IP network stack
- BSD sockets
- USB support
- Power management
  - Provide a framework for controlling the power mode
  - Provide policy module for applications
  - Power modes
    - Active/idle/sleep/off
- POSIX compatible

Limitation

- No dynamic module loading/unloading
- Everything is configured beforehand
- Limited capability of reconfiguration
- Expect better supports