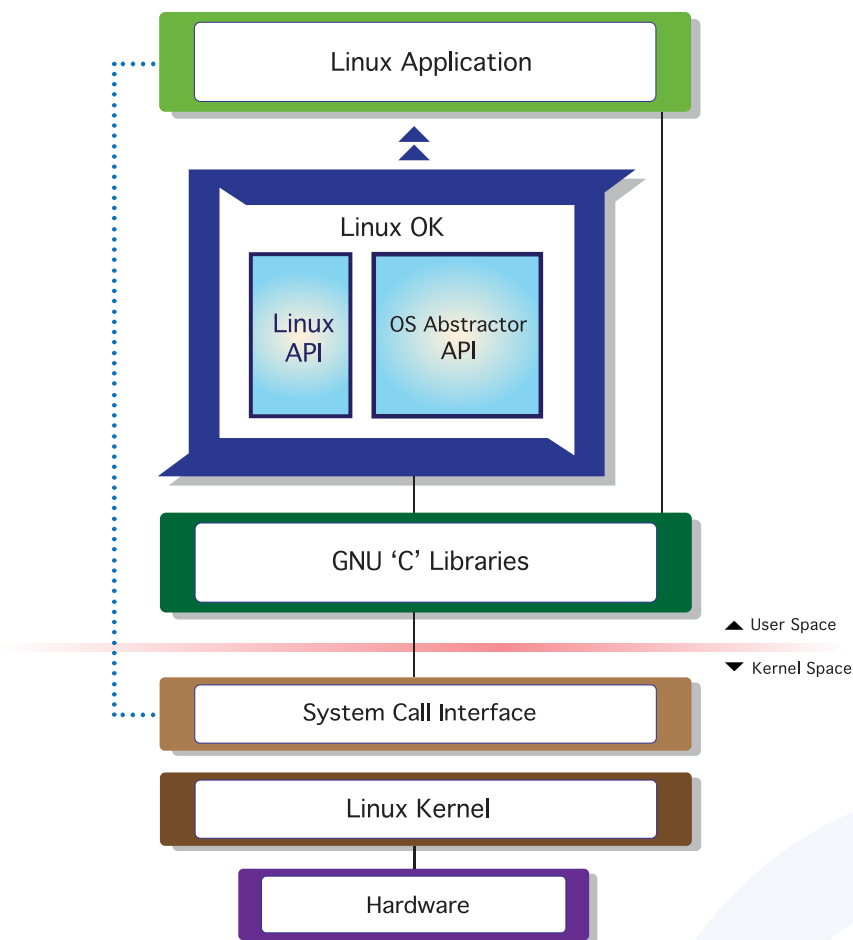


Developers are feeling additional pressure to get a performance increase from their embedded Linux due to increased real-time demands on applications, while having limited resources on their existing target hardware platform. To achieve additional performance, they will need to dig into the Linux operating system code to re-build the kernel and spend a considerable amount of time re-designing the application to improve performance.

MapuSoft's Linux OK™ (Linux Optimization Kit) framework provides several kernel and application specific optimization features all aimed at increasing CPU performance, speeding up boot-time and reducing memory footprint for Linux applications. The following diagram shows how applications use Linux OK.



The Linux Optimization Kit provides header files and template code to re-direct the application to use the Linux OK interfaces. It also generates optimized interface code, based on the level of individual API usage in the application, and according to GUI settings input by the user which captures resource requirements and target environment settings. Linux OK improves application performance by reducing run-time creation and deletion of OS resources by pooling and reusing them across applications/processes. Linux OK also reduces unnecessary task and user/kernel mode switching.

## Linux OK Highlights:

- Enhances Linux application performance
  - > Statically create and reuse OS resource control blocks across one or more applications
  - > Statically create and reuse threads from pool
  - > Enhance performance without breaking Linux API compliance or modifying your application
  - > GUI to configure target environment and resource requirements for various Linux kernel versions and variants (32/64 bit; SMP/UP systems)
- OS Abtractor Interface for advanced features
  - > Tiered memory pools to eliminate memory fragmentation
  - > Shared memory pool to eliminate data copy across applications
  - > Automatically recover from software fatal errors through a soft reset
  - > Use the same Interface for inter-process communication within and across individual application processes
- Scalability and Optimization specific to your application
  - > Profiler to identify performance bottlenecks and level of API usage so that specific optimizations can be made
  - > Generate project files for your target

## Linux OK Contents:

- Application Common Operating Environment (AppCOE): An eclipse based IDE for development and deployment of Linux applications developed in C, C++ and Ada
- Linux Interface (an equivalent subset of GNU 'C' Lib)
- OS Abtractor Interface for advanced optimization
- Code generator for the Linux and OS Abtractor interface components that are specific to the application(s)
- Profiler to graphically view performance data regarding your application, API usage and other report generations

## Related Links

- You can find the supported Linux APIs here:  
[http://www.mapusoft.com/wp-content/uploads/documents/Release\\_Notes-POSIX-APIs.pdf](http://www.mapusoft.com/wp-content/uploads/documents/Release_Notes-POSIX-APIs.pdf)
- You can find the supported OS Abtractor APIs here:  
[http://www.mapusoft.com/wp-content/uploads/documents/Release\\_Notes-OS-Abtractor-APIs.pdf](http://www.mapusoft.com/wp-content/uploads/documents/Release_Notes-OS-Abtractor-APIs.pdf)
- A free evaluation can be downloaded here:  
<http://mapusoft.com/downloads/>
- For user manual & technical documentation visit this link:  
<http://www.mapusoft.com/techdata/>
- For any technical or sales questions please submit a ticket at the MapuSoft support site at this link:  
<http://mapusoft.com/support/>