Stabilizing the DPDK ABI and what it means for you

Stephen Hemminger (Microsoft Azure)
Ray Kinsella (Intel)
DPDK Summit – Mountain View - 2019
Technology Adoption Curve

Constant ABI churn limits the reach of latest releases – to Enthusiasts and Visionaries.

How do we encourage the silent majority, to adopt new releases of DPDK?

Increasing conservatism resulting in slower release adoption

ABI Breakages discourages new technology adoption by the silent majority.
Comparative ABI Churn

GStreamer Application Binary Interface
- 100% backward compatible within Major Versions (1.x).
- Stable since 1.4.5, typically < 1% change between Major Versions.


DPDK Application Binary Interface
- 8.7% median ABI churn between quarterly releases.
- LTS release is API stable for 2 years, however limited backporting of new features or HW.


Thanks to ABI Laboratory for providing the ABI analysis tools.
We care about user-space interfaces to an insane degree. We go to extreme lengths to maintain even badly designed or unintentional interfaces. Breaking user programs simply isn't acceptable.

Linus Torvalds, 2005
What is an ABI?

**API: Application Programming Interface**
- What a programmer expects to see.
- Programing specification comprised of: functions, datatypes, headers ...
- Typically managed very carefully by software engineers.

**ABI: Application Binary Interface**
- What the application expects to see.
- Runtime binary interface comprised of: Public symbols and symbols versions
- Typically managed very carefully by DevOPs Engineers, OSVs, CSPs etc

Understanding ABI is key part of deploying a stable application
What is an ABI?

Application

ABI

API

function calls

Programming Language
Functions Datatypes
Return Types Constants

Instruction set
Executable & Linker Format
Calling Conventions.
...

DPDK
Proposals on ABI Stability

- Major ABI versions are declared every year and are then supported for one year, typically aligned with the LTS release.
- The ABI version is managed at a project level in DPDK, with the ABI version reflected in all library’s soname.
- The ABI should be preserved and not changed lightly. ABI changes must follow the outlined deprecation process.
- The addition of symbols is generally not problematic. The modification of symbols is managed with ABI Versioning.
- The removal of symbols is considered an ABI breakage, once approved these will form part of the next ABI version.
- Libraries or APIs marked as Experimental are not considered part of an ABI version and may change without constraint.
- Updates to the minimum hardware requirements, which drop support for hardware which was previously supported, should be treated as an ABI change.

[dpdk-dev] [PATCH v7 0/4] doc: changes to abi policy introducing major abi versions
The addition of symbols is generally not problematic. The modification of symbols should be managed with ABI versioning.

New platform, features & improvements continue to be released.
Where the ABI policy applies

<table>
<thead>
<tr>
<th>In Scope</th>
<th>Out of Scope</th>
</tr>
</thead>
<tbody>
<tr>
<td>• lib</td>
<td>• lib (experimental)</td>
</tr>
<tr>
<td>• drivers (with public API's)</td>
<td>• api (annotated experimental)</td>
</tr>
<tr>
<td></td>
<td>• examples</td>
</tr>
<tr>
<td></td>
<td>• usertools</td>
</tr>
<tr>
<td></td>
<td>• devtools</td>
</tr>
<tr>
<td></td>
<td>• doc</td>
</tr>
</tbody>
</table>
ABI Stability: major ABI versions

v20 ABI is declared aligned with v19.11 LTS
v21 symbols are added and v20 symbols are modified, support for v20 ABI continues.

v21 ABI is declared aligned with v20.11 LTS, remaining v20 symbols are removed.

v21 ABI is declared aligned with v20.11 LTS, remaining v20 symbols are removed.

ABI Version Compatibility

v20
v21
v22
v23

v20 ABI is declared aligned with v19.11 LTS
v21 symbols are added and v20 symbols are modified, support for v20 ABI continues.

v21 ABI is declared aligned with v20.11 LTS, remaining v20 symbols are removed.

ABI Versions

v20
v21

DPDK Releases

v19.11
v20.02
v20.05
v20.08
v20.11
v21.02
v21.05
v21.08
Next Steps

• Agreement at the DPDK Technical Board following Userspace Bordeaux.

• Supporting work - preparation for ABI stability
  • Build Changes: Global ABI Version & Build Symbol Check.
    • [dpdk-dev] [PATCH v5] Implement the new ABI policy and add helper scripts - Anatoly Burakov & Marcin Baran.
    • [dpdk-dev] [RFC 0/6] Add ABI compatibility checks to the meson build – Kevin Laatz

• Supporting infrastructure to test ABI stability in the UNH lab (WiP).

• Ensure testing of ABI Compatibility as part of the RC Test Cycle (WiP).

What does this mean for your adoption of DPDK releases?
Questions?

Stephen Hemminger
stephen@networkplumber.org