Compiler Support and Build Environments

Applications Subcommittee

JT Plotzke

2025 TPF Users Group Conference
May 05-07, Austin, TX

IBM Z

IBM

Disclaimer

Any reference to future plans are for planning purposes only. IBM reserves the right to change those plans at its discretion. Any reliance on such a disclosure is solely at your own risk. IBM makes no commitment to provide additional information in the future.

As-Is
Compiler and C/C++ Library Support

GNU C library (glibc) 2.3.2

GCC 7

- Initial: PJ45408 (release tpf-17r1-1) – Mar 2019
- Latest: PJ46997 (release tpf-17r1-7) – Feb 2023
- Full support for C++11/C++14 standards

Systems/C and Systems/C++ (Dignus) 2.25

- Initial: PJ46531 (release 2.25.24) – Feb 2022
- Latest: PJ46990 (release 2.25.41) – Jul 2023
- Mirrors GCC 7 in terms of libstdc++ run time support

Latest compiler releases listed at Compiler requirements

Future Deliverables
glibc 2.41 support

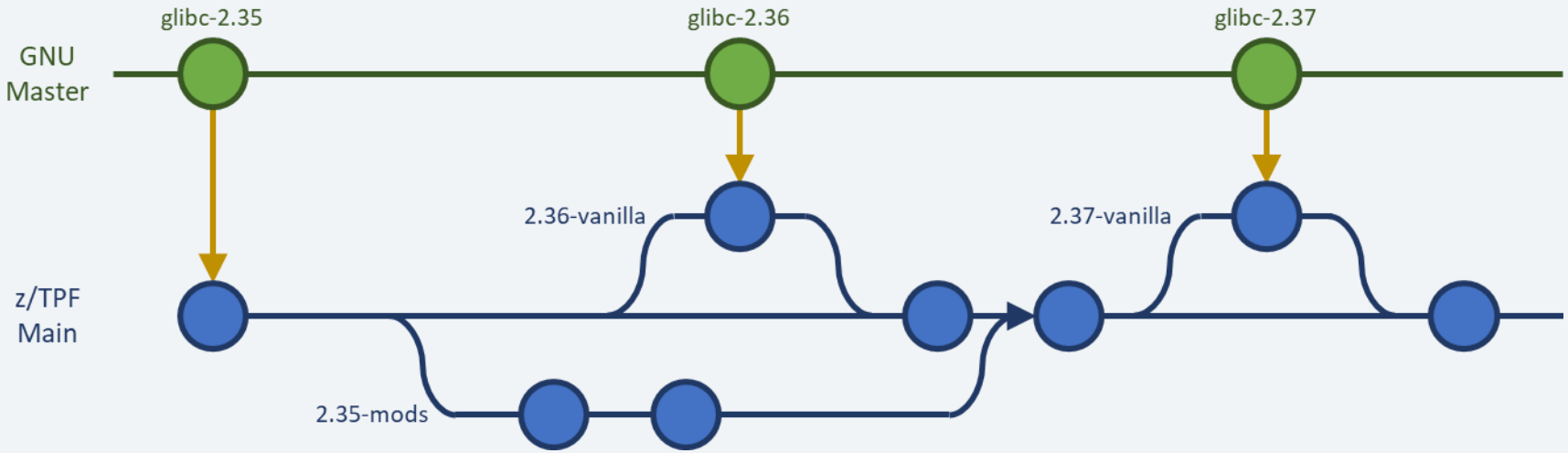
PJ46900 will upgrade support for C applications by providing access to new standards, APIs, and portability (Target 2025)

- Improved POSIX compliance across C APIs
- New thread library support will be incorporated into CISO
- POSIX compliant threads library simplifies writing or porting thread applications to z/TPF
- C11 standard support upon release (maximum supported by GCC 7)
- Ability to support up to C23 standard in the future
- New release of GCC 7 (tpf-17r1-8) and minor version of Dignus (2.25.*xx*) will be prerequisites
- Prerequisite to our ability to support new compiler versions

Future Deliverables
glibc 2.41 support

Will revamp our internal process for maintaining glibc

- Internally using a Git source repository to incorporate each new glibc version



Future Strategy
Compiler and C/C++ Library Support

Upgrade glibc and compiler versions more frequently, and make it easier and beneficial to migrate immediately

- New version of glibc released by GNU every 6 months
- New version of GCC released by GNU every year
- Will not be releasing each version for the sole purpose of currency, but need the ability to do so as needed and the infrastructure to be quicker than before
- Newer C and C++ standards to be supported
- Aim to minimize effort (amount needed to recompile and test) to migrate to newer compiler
- Incentivize migrating to new compiler when available versus when old compiler is no longer available
- Following compiler strategy previously shared, new cadence to begin with this cycle

Future Strategy
Compiler Support

As-Is Support: GCC 4.6 and GCC 7

To-Be Support: GCC 7 and GCC 15, continue new pattern of:

1. (Announce) Deprecate oldest supported GCC version (4.6)
2. (APAR) Discontinue support for GCC *v.old* (4.6)
3. (APAR) Pre-release GCC *v.next* (15) compiler with any necessary code compatibility updates for compilation
4. (APAR) Add support for GCC *v.next* (15)
5. (APAR) Update Dignus compiler support as needed to maintain compatibility with GCC *v.next* (15)

Future Strategy
Compiler Support

3. Pre-release GCC 15 with compatibility updates (APAR)

- APAR would focus on z/TPF header updates required for clean compiles
- Post a beta version of the GCC 15 cross compiler
- Introduce a new `TPFGCC_VERSION := NEXT` maketpf.cfg option
- When using the `NEXT` version, only perform compilation not linkage
- Provides the ability for compile time testing and warning resolution to be done for user applications at the same time as the z/TPF product
- Better overlaps migration timelines with compiler release timelines

Future Strategy
Compiler Support

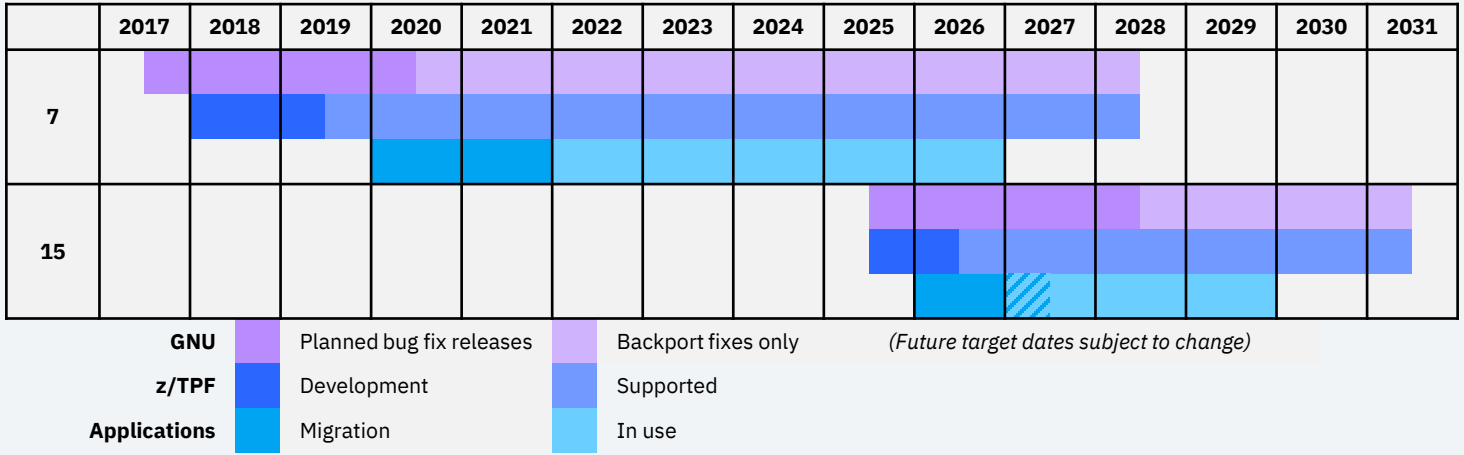
4. Add support for GCC 15 (APAR)

- APAR itself already standard business approach
- Options for **`TPFGCC_VERSION`** are now: **`7 15 COMPAT`**
- When using the **`COMPAT`** version:
  - GCC 15 would be used but with as many new warnings as possible disabled
  - Also revert any default compile options back to their GCC 7 defaults
- Can adopt the new compiler before a new application warnings are addressed

Future Strategy
Compiler Support

NEXT and **COMPAT** compiler versions enable migrations to happen sooner and easier than before

- More overlap between z/TPF development and user application migration



Future Strategy
Compiler and C/C++ Library Support

Newer C and C++ standards to be supported when:

- Latest glibc/libstdc++ libraries fully support the C/C++ standard
- All compilers at least partially implement support for the standard

	C90	C99	C11	C17	C23	C++03	C++11	C++14	C++17	C++20	C++23	C++26
glibc 2.41	F	F	F	F	F							
libstdc++ 7						F	F	F	X			
libstdc++ 15						F	F	F	F	F	X	X
GCC 7	F	F	F			F	F	F	X			
Dignus 2.25	F	F	F			F	F	F	F			
GCC 15	F	F	F	F	F	F	F	F	F	F	X	X
Dignus 2.30	F	F	F		F	F	F	F	F			

F Full support

X Experimental support

C/C++ standards supported on z/TPF after glibc 2.41 upgrade

C/C++ standards supported on z/TPF after GCC 15 upgrade

Future Deliverables
Compiler and C/C++ Library Support

Following compiler strategy:	Target Timeframe
1. Deprecate GCC 4.6	Aug 2022* (announcement)
2. Discontinue GCC 4.6 support	Feb 2024* (PJ47248)
3. Pre-release GCC 15 beta compiler and code compatibility updates	2025
4. GCC 15 and C++17 support	2026
5. Dignus 2.30 and C++17 support	2026
Preexisting GCC 7 support (2019-) continues throughout	**Dropping GCC 4.6 happened earlier than normal due to glibc upgrade*

To-Be
Compiler and C/C++ Library Support

GNU C library (glibc) 2.41

GCC 7

- Full support for C11/C++11/C++14 standards
- Experimental support for C++17 standards

GCC 15

- Target 2026
- Full support for C11/C++17 standards

Systems/C and Systems/C++ (Dignus) 2.30

- Mirrors GCC 15 in terms of libstdc++ run time support

Build Environments

As-Is
Build Environments

SUSE Linux Enterprise Server (SLES) 12

- Initial: PJ44753 – May 2017
- Latest: PJ46098 – May 2020 (SLES 12 SP4)
- z/TPF support for SLES is deprecated as of Apr 2025 (announcement)
- z/TPF support for SLES will be withdrawn in Apr 2026

Red Hat Enterprise Linux (RHEL) 9

- Initial: PJ48047 – Jul 2024
- Latest: PJ48128 – Dec 2024 (RHEL 9.5)

Latest system levels listed at Linux for IBM Z requirements

Recent Deliverables

PJ48047 – Currency updates to support RHEL 9 (Jul 2024)

- Code compatibility updates to various z/TPF utility and offline programs for building on RHEL 9

PJ48128 – Currency updates to support RHEL 9.5 (Dec 2024)

- MakeTPF-only update to GCC version audit

Recent Deliverables
Currency updates to support RHEL 9

PJ48047 provides currency updates to support building and loading z/TPF from a RHEL 9 environment

- RHEL 9 uses GCC 11 (with new C/C++ standard defaults of -std=gnu17 and -std=gnu++17)
- GCC 11 detected new errors and warnings in various utility and offline programs
- Some warnings were safely suppressed in .mak files while others revealed legitimate source code changes needed to be made
- Online programs were also affected due to shared source segments with offline programs
- Dignus compilers cannot yet build applications with RHEL 9 without breaking z/TPF debugger support
- Further details: https://community.ibm.com/community/user/ibmz-and-linuxone/blogs/jt-plotzke/2024/08/28/rhel-9-compatibility

Recent Deliverables
Currency updates to support RHEL 9.5

PJ48128 provides currency updates to support building and loading z/TPF from a RHEL 9.5 environment

- No additional code changes or build instructions beyond what PJ48047 already provided
- MakeTPF-only update to allow for GCC 11.5 to be used for offline compiles (previous max was 11.4)
- Will revisit need for minor version audit within MakeTPF tools with next required currency update (when a Linux for IBM Z system level begins using GCC 12+)

Future Strategy
Build Environments

We will only guarantee support for building and loading z/TPF from a RHEL environment, based on Red Hat's latest support

- Each version of RHEL is supported by Red Hat for 10 years (5 years full support + 5 years maintenance)
- New version released by Red Hat every 3 years

(Release and support policies subject to change in the future at Red Hat's discretion)

OS version	General availability	Full support ends	Maintenance support ends
RHEL 8	May 2019	May 2024	May 2029
RHEL 9	May 2022	May 2027	May 2032
RHEL 10	Mid-2025		

Future Strategy
Build Environments

TPF Lab will use the latest, stable RHEL for its primary build environment, and version *n-1* for its secondary environment

- Regular builds are performed on both Linux on IBM Z systems
- Primary Linux will upgrade from RHEL *n-1* to RHEL *n* after RHEL *n* has been generally available for 2 years to ensure stability
- RHEL *n* compatibility APAR is shipped
- Secondary Linux will be upgraded from RHEL *n-2* to RHEL *n-1* at the same time
- TPF will continue to support for RHEL *n-2* on an as needed basis for the duration of its 10-year Red Hat lifecycle

Future Deliverables
Build Environments

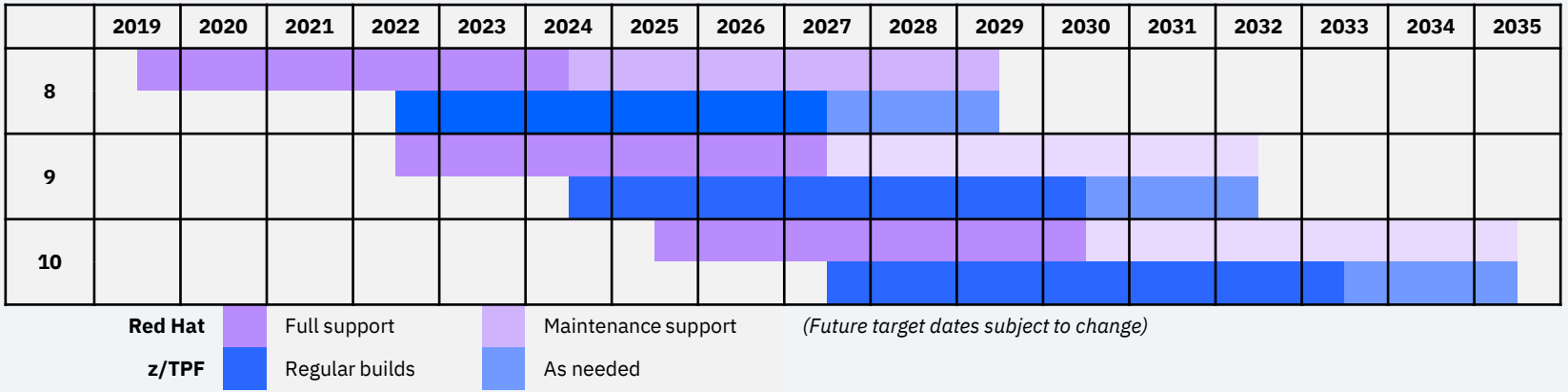
RHEL 10 compatibility (Target 2027)

- Red Hat to release RHEL 10 imminently (“mid-2025”), z/TPF compatibility to follow in 2 years
- Regular builds to be performed on both RHEL 9 and RHEL 10
- Continue to support RHEL 8 as needed until target 2029

To-Be
Build Environments

RHEL 8 and RHEL 9: Target 2024-2027

RHEL 9 and RHEL 10: Target 2027-2030



Be a sponsor user

Sponsor users assist in design and implementation, and your feedback drives our development cycle.

Target personas

- Application developer
- System programmer
- Build manager

Begins

2H 2025 (GCC 15 upgrade)

Interested? Contact

Dan Gritter (dgritter@us.ibm.com)
JT Plotzke (jtplotzk@us.ibm.com)

Thank you