



| z/TPF V1.1

TPF Users Group Spring 2008

Title: z/OS Compiler Support for TPF 4.1,

Migrating with updated C/C++ Single Source Rules

Name: Peter Lemieszewski

Venue: Open Source Subcommittee

AIM Enterprise Platform Software
IBM z/Transaction Processing Facility Enterprise Edition 1.1.0

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.

© 2008 IBM Corporation

•z/OS Compiler Support

- The following compilers are supported:
 - **z/OS compiler with TARGET(OSV2R10) option**
 - **OS/390 V2R10 compiler**
- For Support:
 - **Contact your TPF Customer Support Representative**

TPF4.1 and z/OS Compiler Support – Release Roll Out

<u>TPF 4.1 PUT</u>	<u>DATE</u>	<u>COMPILER</u>	<u>End of Service</u>
“PUT 22”	11/2007	z/OS 1.7	9/2008
“PUT 22 +”	01/2008	z/OS 1.8 – new APARs	9/2009
“PUT 23”	11/2008	z/OS 1.9	9/2010
“PUT 24”	11/2009	z/OS 1.10	12/2010
“PUT 25”	11/2010	z/OS 1.11	12/2010

Migrating with Updated Single Source Rules

- **OTRKYWDc: `_Export` and visibility**

- `//TPF4.1` Code

```
static void * _Export gimmeVoidPointer (int x);
```

- z/OS requires `_Export` to precede function name

Migrating with Updated Single Source Rules

- **OTRKYWDc: `__Export` and visibility**

//The same code written for z/TPF

```
__attribute__((visibility("default"))) static void * gimmeVoidPointer (int x);
```

- Gcc scoping rules require visibility attribute to be placed so that they apply to function name

- **Not the return type**

- This is not what we want:

- `static void * __attribute__((visibility("default"))) gimmeVoidPointer (int x);`

-

Migrating with Updated Single Source Rules

- **OTRKYWDc: _Export and visibility**
- Using updated OTRKYWDc rule (version 3.2.6):

```
//original line of code: static void * _Export gimmeVoidPointer (int x);  
#ifndef __370__  
    #define _Export  
    #define _Export_zTPF __attribute__((visibility("default")))  
#else  
    #define _Export_zTPF  
#endif  
_Export_zTPF static void * _Export get (int x);
```

Migrating with Updated Single Source Rules

- **OTRPACKd: `_Packed` keyword**
- Consider the following code

```
_Packed struct
{
    char i;
    int c;
    float f;
} UFO;
_Packed struct UFO flying;
struct UFO saucer;
```

Migrating with Updated Single Source Rules

- **OTRPACKd: `_Packed` keyword**
- `_Packed struct // The _Packed keyword does not apply here`
- `{`
- `char i;`
- `int c;`
- `float f;`
- `} UFO;`
- `_Packed struct UFO flying; // This structure is packed!`
- `struct UFO saucer; // This structure is not packed!`
- z/OS allows packed and unpacked instances of a structure
- Gcc does **not** allow packed and unpacked instances of a structure!

Migrating with Updated Single Source Rules

- **OTRPACKd: `_Packed` keyword**
- **One possible solution – allows OTRPACKc to be applied**

```
struct //original structure
```

```
{  
    char i;  
    int c;  
    float f;
```

```
} UFO;
```

```
struct // new structure written for z/TPF migration.
```

```
{  
    char i;  
    int c;  
    float f;
```

```
} UNKNOWNFLYINGOBJECT;
```

```
_Packed struct UFO flying; // This structure is packed! It can be migrated
```

```
struct UNKNOWNFLYINGOBJECT saucer; // This structure is not packed! It can be migrated
```

Trademarks

- IBM, z/OS, System z, and WebSphere are trademarks of International Business Machines Corporation in the United States, other countries, or both.
- Linux is a trademark of Linus Torvalds in the United States, other countries, or both.
- Other company, product, or service names may be trademarks or service marks of others.
- UNIX is a registered trademark of The Open Group in the United States and other countries.
- Other company, product, or service names may be trademarks or service marks of others.
- **Notes**
- Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput improvements equivalent to the performance ratios stated here.
- All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.
- This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.
- All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.
- Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the performance, compatibility, or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.
- Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.
- This presentation and the claims outlined in it were reviewed for compliance with US law. Adaptations of these claims for use in other geographies must be reviewed by the local country counsel for compliance with local laws.