z/TPF EE V1.1
z/TPFDF V1.1
TPF Toolkit for WebSphere® Studio V3
TPF Operations Server V1.2

IBM Software Group

TPF Users Group Fall 2006

Decimal Applications

Name: Edwin W. van de Grift

Venue: Hot Topics

AIM Enterprise Platform Software

IBM z/Transaction Processing Facility Enterprise Edition 1.1.0

© IBM Corporation 2006

Any references 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.

What is a Decimal?

- Decimal
  - **From Wikipedia, the free encyclopedia**
  - The decimal (base ten or occasionally denary) numeral system has ten as its base.
  - It is the most widely used numeral system, probably because humans commonly have a total of ten digits on their hands.

© IBM Corporation 2006

Floating Point Terminology

$$\pm d.dd\ldots d \times \beta^{e}$$

- d.dd…d – **Significand**
  - Where $0 \leq d < \beta$
  - The number of digits in the significand is called the **precision**.
  - Also refered to as **coefficient** or mantissa.
- β – **Base**
  - β is assumed to be even.
- e – **Exponent**

Decimal Floating Point (DFP)
β =10

- A floating point number is **normalized** when the leading digit of the significand is nonzero.

© IBM Corporation 2006

An Example

- The number 0.1 in Decimal versus Binary
  - Decimal ($\beta = 10$) with precision 3

    1.00×10^{-1}
  - Binary ($\beta = 2$) with precision 24

    $1.10011001100110011001101 \times 2^{-4}$
    - **Cannot be represented exactly!**

© IBM Corporation 2006

Decimal Floating Point in GCC 4.2

- GCC 4.2 targeted for Q3 2006
- z/TPF support of GCC 4.2 is likely
- New native types
  - **_Decimal32**
    - 7 decimal digits
  - **_Decimal64**
    - 16 decimal digits
  - **_Decimal128**
    - 34 decimal digits
    - Fits every packed number and more!

© IBM Corporation 2006

Encoding

- 1 Bit Sign
- 5 Bits Combination Field
  - 2 Most significant bits of the exponent. Values 0-2.
  - Most significant digit (4 bits) of the coefficient. Values 0-9.
- 6, 8, 12 Bits Exponent Continuation
  - The exponent is encoded, by adding a bias so that the encoded exponent is greater than or equal to 0. Biases are 101, 398, 6176
- 20, 50, 110 Bits Coefficient Continuation
  - Densely Packed Decimal, 3 digits in 10 bits
  - 7, 16, 34 Digit Coefficient

- Supports
  - ±0
  - ±Infinity
  - Not-a-Number (NaN)
    - Signaling and Quiet

© IBM Corporation 2006

Decimal Encoding Example

	Decimal64
-7.50	A2 30 00 00 00 00 03 D0

- Sign = 1
- Coefficient = 0000000000000750
  - First digit (0) in combination field
  - Remaining digits (15) in coefficient continuation
    - X'11 1101 0000'
- Exponent = -2
  - Encoded exponent -2 + bias (396)
    - X'01 1000 1100'

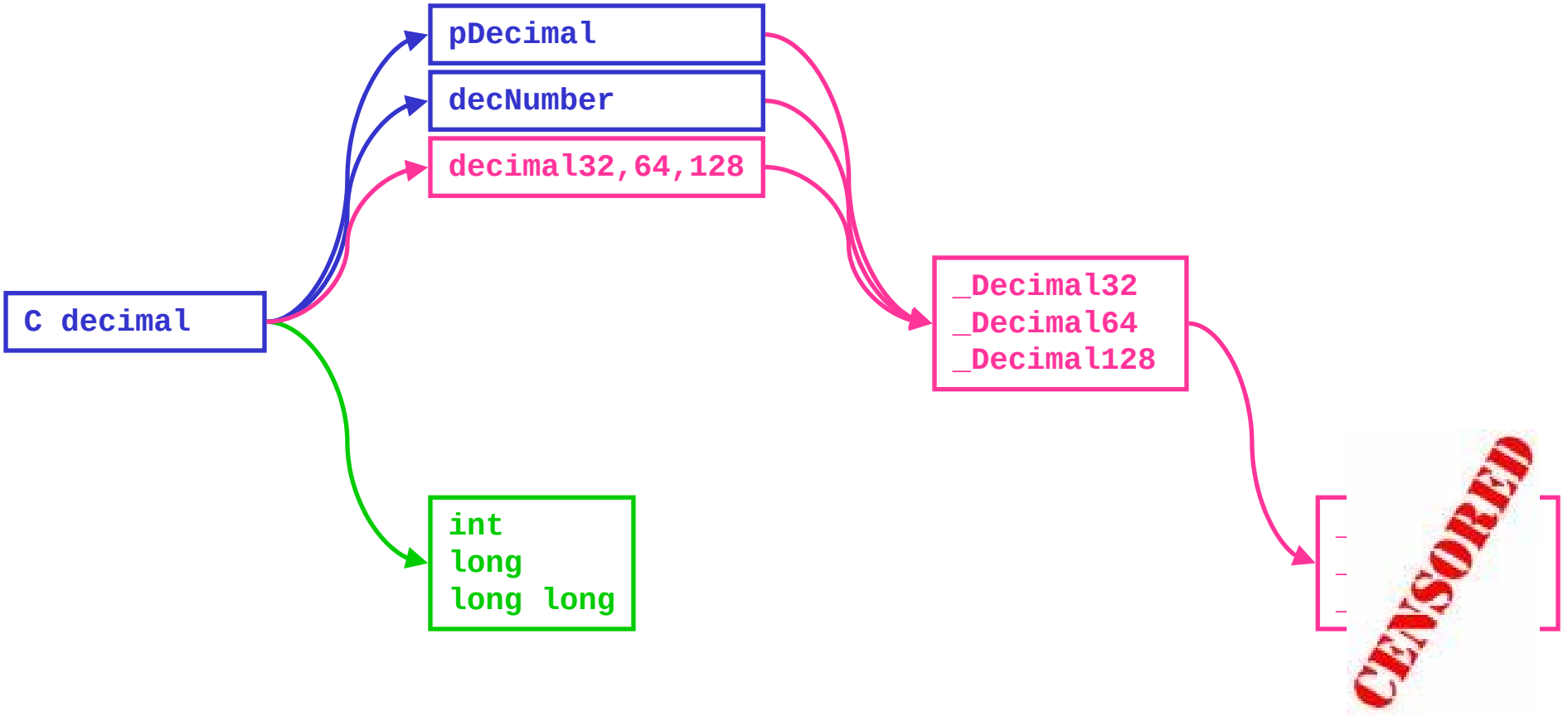
© IBM Corporation 2006

Migrations

Data format:
IBM packed decimal
binary
IEEE decimal floating point

TPF4.1 | **TPF4.1 & z/TPF** | **z/TPF**

software compiler hardware



© IBM Corporation 2006

Sample Code

```
decimal(5,2) d1 = 1.2;
decimal(3) d2 = 42;

decimal(9,4) d3 = d1 * d2;
```

```
_Decimal128 d1 = 1.2DL;
_Decimal128 d2 = 42DL;

_Decimal128 d3 = d1 * d2;
```


© IBM Corporation 2006

Conversion Functions

```
_Decimal32  packedToDecimal32(unsigned char* packed, int width, int precision,
                              _Decimal32 d);
_Decimal64  packedToDecimal64(unsigned char* packed, int width, int precision,
                              _Decimal64 d);
_Decimal128 packedToDecimal128(unsigned char* packed, int width, int precision,
                               _Decimal128 d);


unsigned char* decimal32ToPacked(_Decimal32 d,
                                 unsigned char* packed, int width, int precision);
unsigned char* decimal64ToPacked(_Decimal64 d,
                                 unsigned char* packed, int width, int precision);
unsigned char* decimal128ToPacked(_Decimal128 d,
                                  unsigned char* packed, int width, int precision);
```

- More/other information will come with TPF support of GCC 4.2

© IBM Corporation 2006

References

- Floating Point Standards
  - IEEE 754 governs binary floating-point arithmetic.
    - It specifies number formats, basic operations, conversions, and exceptional conditions.
  - The related standard IEEE 854 generalizes 754 to cover decimal arithmetic as well as binary.
  - Revision IEEE 754r
    - Merging 854 into 754, reducing implementation choices, resolving ambiguities in 754, standardizing fused multiply-add, and including quadruple precision.

© IBM Corporation 2006

References

- http://en.wikipedia.org/wiki/Decimal
- General Decimal Arithmetic, Mike Cowlishaw
  - http://www2.hursley.ibm.com/decimal/
- Let's Get to the (Floating) Point, Chris Hecker
  - http://www.d6.com/users/checker/
- What Every Computer Scientist Should Know About Floating-Point Arithmetic, David Goldberg
  - http://docs.sun.com/source/806-3568/ncg_goldberg.html
- Densely Packed Decimal Encoding, Mike Cowlishaw
  - *IEE Proceedings – Computers and Digital Techniques*, ISSN 1350-2387, Vol. 149, No. 3, pp102-104, IEE, May 2002

© IBM Corporation 2006

Trademarks

IBM is a trademark of International Business Machines Corporation in the United States, other countries, or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.
Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.
Linux is a trademark of Linus Torvalds in the United States, other countries, or both.
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

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.

© IBM Corporation 2006