



| IBM Software Group

64-bit support

Glue modules



@business on demand.

© 2007 IBM Corporation
Updated May 14, 2015

This presentation will discuss Glue Modules on a WebSphere® base application server V6.1 on z/OS.

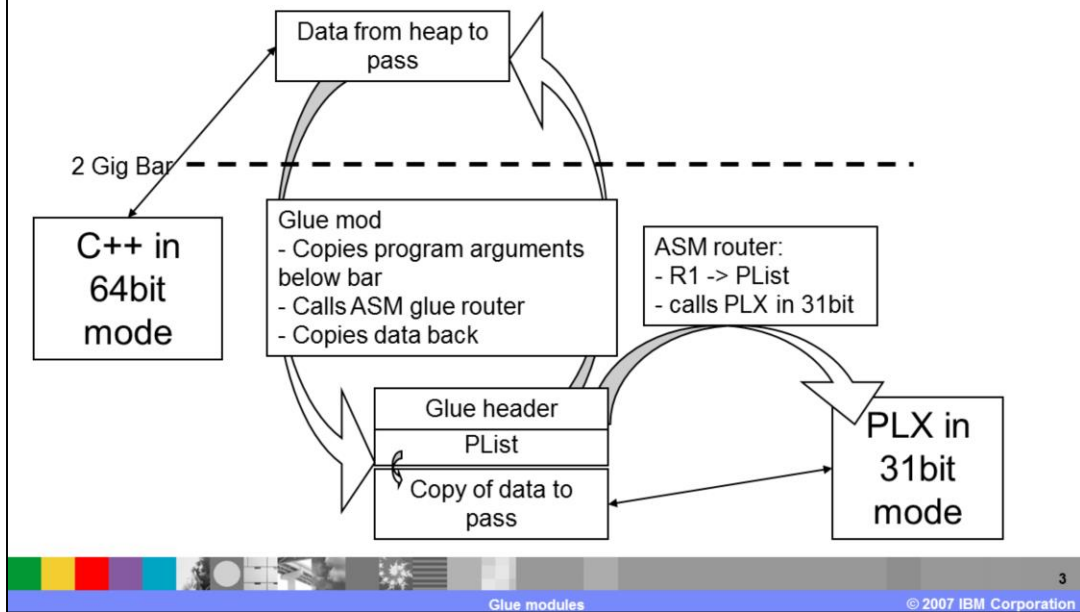
What is glue code?

- C++ programs are compiled in 64bit mode, but PLX programs are still in 31bit mode.
- PLX cannot access data above the 2meg bar.
- Glue code moves data below the bar and switches to 31bit mode.
- On return, glue code switches back to 64bit mode and moves output data to original location above the bar.



The 31-bit server has the C++ programs compiled in 31bit mode, `_LP64` not defined. In this case, the glue code is not used.

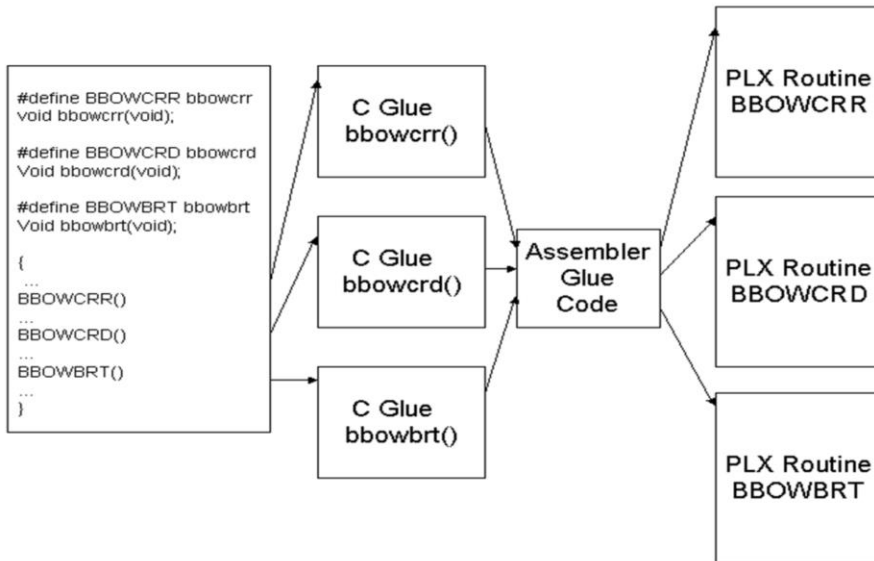
Passing data between 64bit C and 31bit PLX



Data that is above the 2 gigabyte bar is passed to the glue code, which copies the data to a work area that is allocated from below the bar using RAS_MALLOC31. The pointers to this data are passed to the PLX routine as if they were the original parameters.

If the PLX routine passes any data back to the C++ function, that data is copied back to the location pointed to by the original parameters.

Calling the glue code



`#define`'s effectively change calls to upper case names ,the PLX routines, to lower case names used in the glue code.

In the glue code, the PLX code is called by moving the address of the PLX function into a parameter that is passed to the assembler glue code. This address is then loaded into Register 15 and a `BASR R14,R15` command is executed.

Calling the glue code

- Intermediate glue module is bbgqxxxx.cpp
- Standard assembler glue router: bbgqcasm.asm
- C caller: calls PLX module name
- bboXstub.h controls flow of call to glue mod if 64bit mode

Example: bbo3stub.h:

```
#ifdef _LP64
#define BBO3GETS bbo3gets
void * bbo3gets(mvs_storage_type StgType,
               int Len,
               int Subpool);

#else
void * BBO3GETS(mvs storage_type ...etc...)
#endif
```

If compiling for 64 bit, calls to PLX routines are changed to calls to a lower case version of the name. This resolves to the glue code for that PLX routine.

New trace information

- **At glue mod entry:** shows glue mod name and line number and data being passed in to the PLX routine.
- **At glue mod exit:** shows glue mod name and line number and data being passed back to the C++ function.
- **At RAS_MALLOC31:** shows file, function name, line number, and size & address of allocated area.
- **At RAS_DELETE:** shows file, function name, line number, and pointer to deleted area.



There are some new trace points available; they are listed here.

Glue Modules: The good, the bad, and the ugly

- Enables compatibility
 - ▶ 64bit exploiters
 - ▶ Legacy 31bit code
- Performance hit to copy data
 - ▶ Extreme cases, PLX 64bit code blocks
- New dependency on interface
 - ▶ If you change a C to PLX interface, the glue module must also be updated!



Glue code enables compatibility, allowing you to have the same code for 64-bit and 31-bit . The differences between the two modes are resolved using `_LP64`.

Trademarks, copyrights, and disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM WebSphere

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance 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 or performance improvements equivalent to the ratios stated here.

© Copyright International Business Machines Corporation 2007. All rights reserved.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.

