IBM Software Group | Rational software

IBM Rational Build Forge Servers

Globally distributed development

Rational software

@business on demand.

© 2008 IBM Corporation

Converted to video June 29, 2015

This module covers the basics of Globally Distributed Development for IBM Rational® Build Forge® Version 7.0 and above.

This module assumes users are familiar with IBM Rational Build Forge basics. For a primer on Build Forge, exit this module and first review the Introduction to Build Forge module, then continue with this more advanced topic.

Objectives

- Know what globally distributed development (GDD) is in Build Forge and its benefits
- Understand the architecture of GDD in Build Forge

This module provides an overview of the benefits of Globally Distributed Development (GDD), and the problems it is meant to bypass. It also discusses the architecture of GDD.

IBM

What is GDD?

- GDD is the Build Forge mechanism for working around wide area networks (WANs).

- Performance problems develop around sending large amounts of information between very distant locations.
  - ▶ For example, there is a Build log generating 5 MB of data, and an agent in India wants to send that data back to a console in North America. However, delays and general network problems impair the transfer.

- Extraneous data should not be sent, particularly when giving tasks to agents.

Globally distributed development

3

© 2008 IBM Corporation

GDD is a feature added to Build Forge to relieve problems experienced on widely distributed Build Forge configurations. Problems occur in situations where the agent is geographically distant from the console. There are many inconsistencies with performance, as the Build Forge engine and Agent have trouble staying in sync over the Wireless Area Network (WAN). The solution involves finding a way to reduce the amount of traffic between the Agent and the Engine to minimize the exposure to WAN performance inconsistencies.
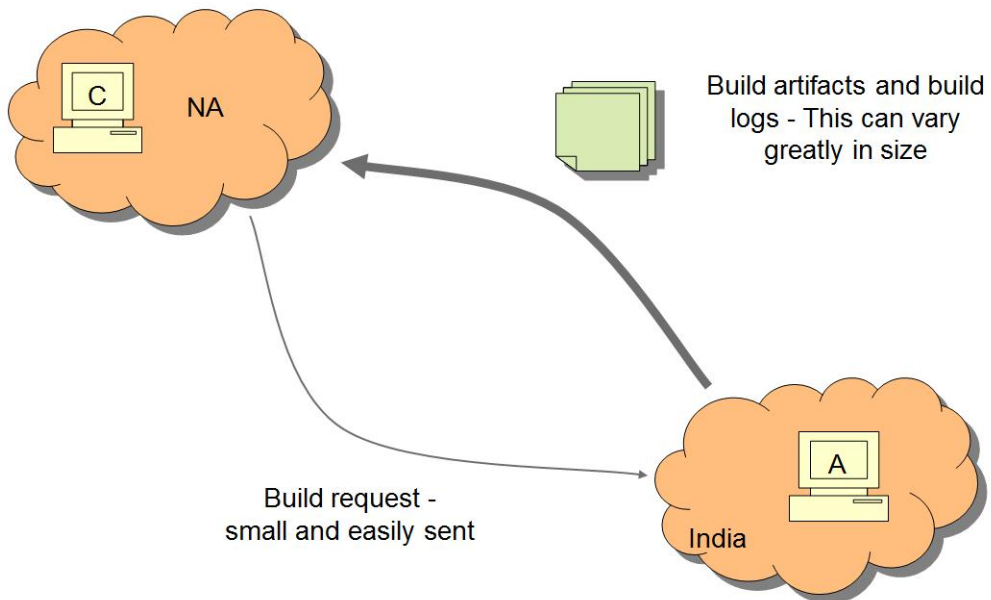
This is a diagram of the problem before GDD is implemented. The Build request is small, and not necessarily affected by distance. The problem emerges when the results need to be sent back to the console. In some cases, this can be hundreds of pages of log files, or large files generated at the Agent. If there are any problems on the WAN during the transmission of that data, bad data will end up in the database.
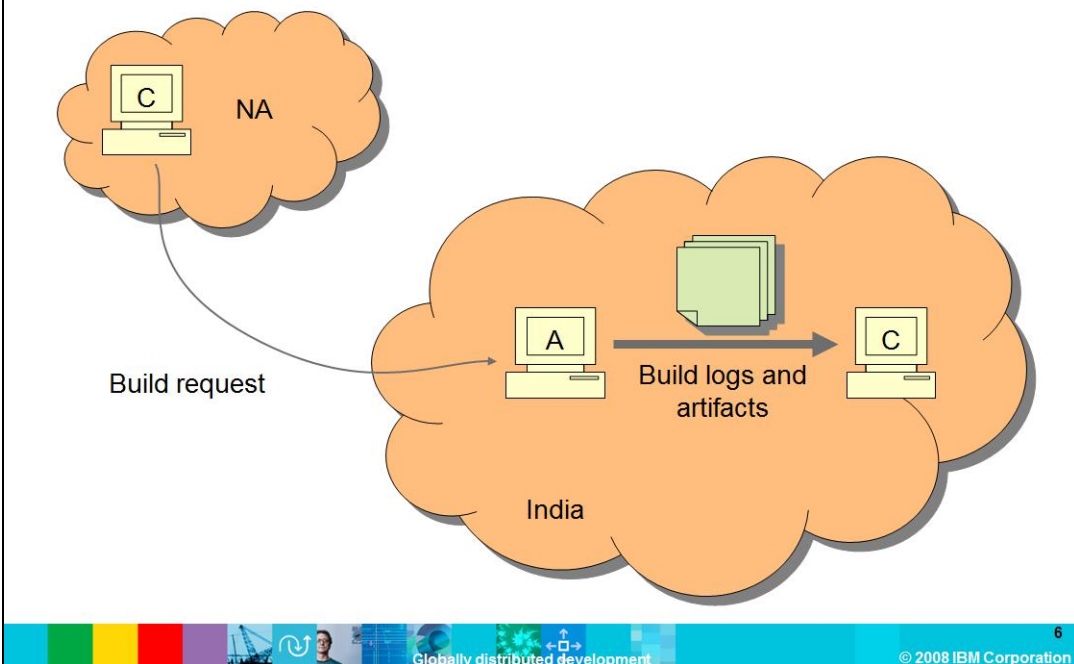
Solution

- Rather than sending back the entire build log and artifacts, keep them where they are generated.

- Install an additional engine in the remote location, and store the build log information there.

- Keep the build configuration information in the main management console.

- Data from the agent does not go over the WAN, but stays in the same location as the agent.

The solution to this situation is to bring the engine closer to the Agent. To accomplish this, add additional engines to the configuration, which will act as local engines for the Agent. This allows the remote engine to store just the project configuration data, leaving the local engine to store the potentially large amount of data that comes back from the Build request. The problem transmit is kept local, thus avoiding the problem.

Solution diagram

NA

C

Build request

A → Build logs and artifacts → C

India

Globally distributed development

© 2008 IBM Corporation

6

This diagram shows the new behavior after GDD is set up. The Build request is still sent from the remote console. However, once the agent runs and completes the task, the Build Logs and files are kept local, as they are only sent to the local Console, and not to the remote one. When a user wants to access those logs or files they can do so from the remote console, and the request is forwarded to the local console where the data actually resides. To the user, there is no difference between this and a non-GDD environment.

IBM

Benefits

- Through GDD, Build Forge regionalizes the engine processes.

- GDD remedies performance problems associated with relying on network traffic over long distances.

- The main management console can be configured to fail over to a secondary site if necessary.

- GDD should not be confused with clustering.
  - Clustering is two engines pointing at the same database, while GDD is a totally separate Build Forge installation with its own database.

Globally distributed development

7

© 2008 IBM Corporation

By implementing GDD, Build Forge allows users to regionalize their processes without penalty for having the agents geographically remote. The data stays local, and the WAN is not overburdened with Build Forge traffic. Also, Build Forge allows those secondary consoles that exist in the other locations to be set up as failover consoles, as they are Build Forge installations complete with their own engine and database. This is different than clustering in Build Forge. Clustering is when two engines point to a single database. This improves performance by balancing the load from a single database across multiple engines. In a GDD environment, each engine has its own engine and database, and improves performance by keeping data transmits local.

Summary

- GDD is the mechanism that Build Forge uses to break down an architecture into regional sections.

- GDD improves performance for widely distributed agents.

In summary, GDD is Build Forge's way of breaking large architectures into regional chunks. This breakdown improves performance by keeping the potentially large data transmits local to the agent that is running the Build.

Trademarks, copyrights, and disclaimers

IBM, the IBM logo, ibm.com, and the following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

Build Forge   Rational

If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of other IBM trademarks is available on the Web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml

Rational is a trademark of International Business Machines Corporation and Rational Software Corporation in the United States, Other Countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

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.

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 2008. 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.