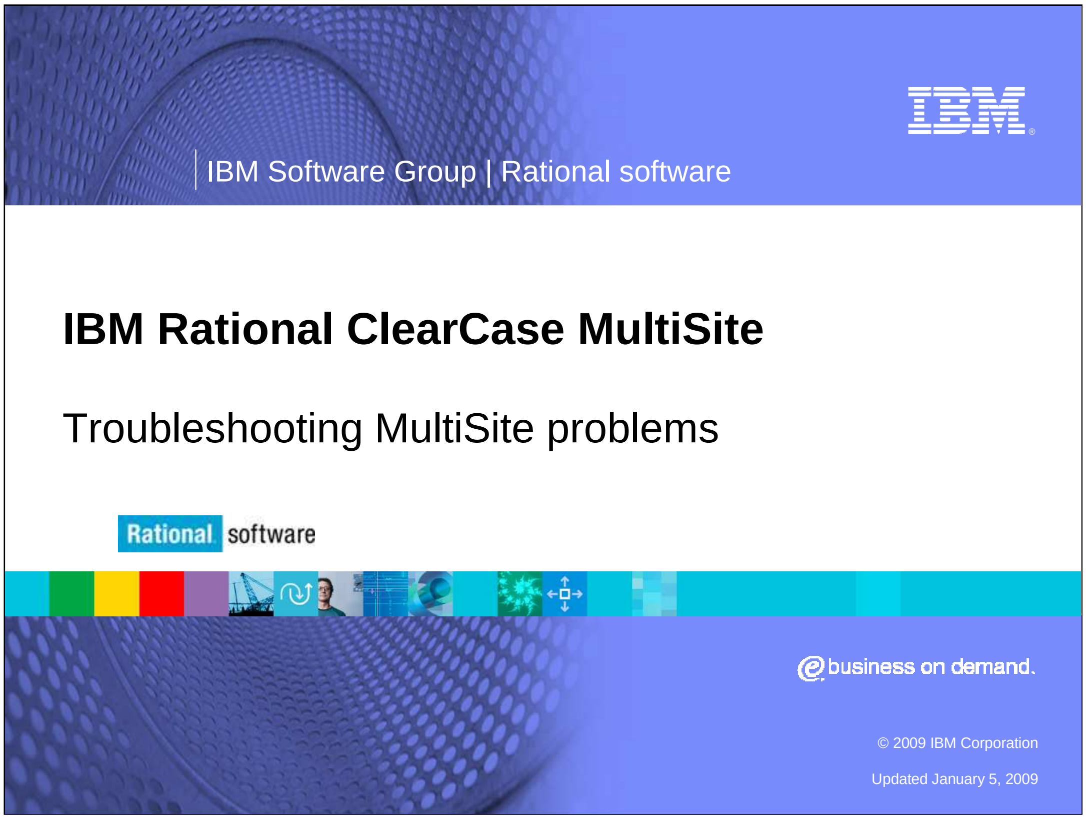


IBM Software Group | Rational software

IBM Rational ClearCase MultiSite

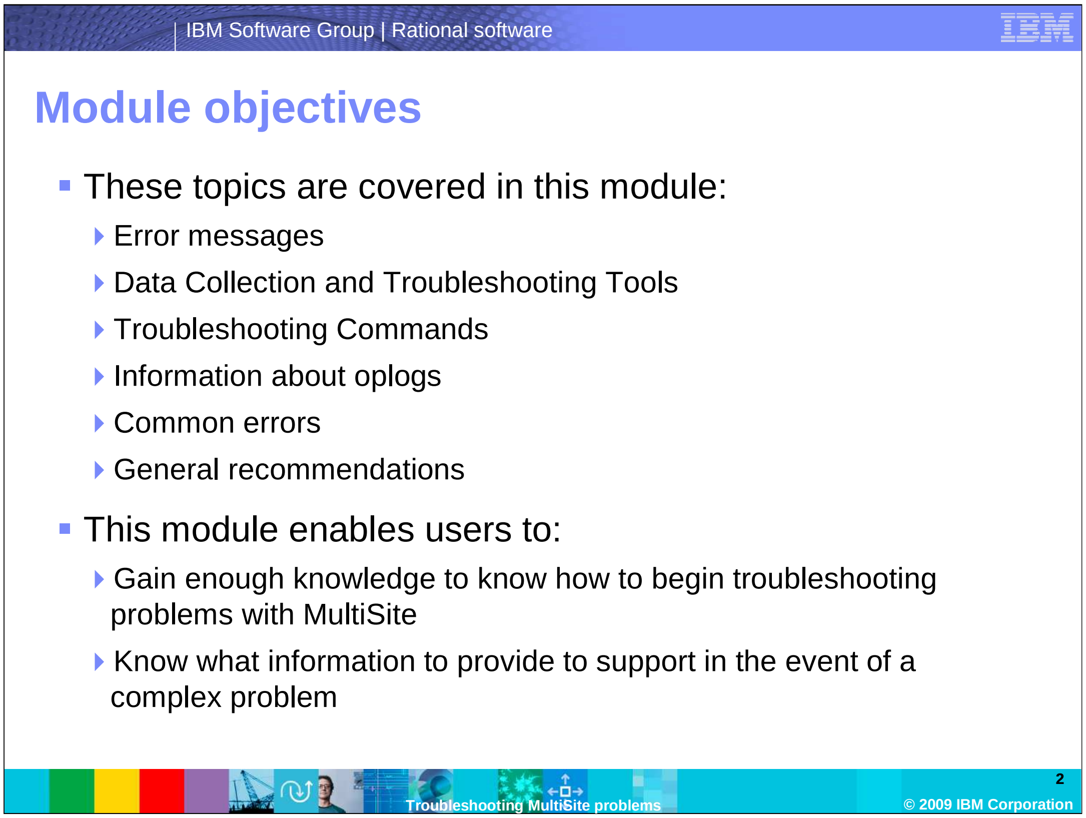
Troubleshooting MultiSite problems

business on demand.

Updated January 5, 2009

This module covers troubleshooting multi-site problems for IBM Rational® ClearCase MultiSite®.

Users should be familiar with general ClearCase® concepts and terminology, and should complete the Introduction to IBM Rational ClearCase MultiSite and Creating a Replicated Environment modules before continuing. These modules help to familiarize users with MultiSite terminology and concepts, and how to create a replicated environment.



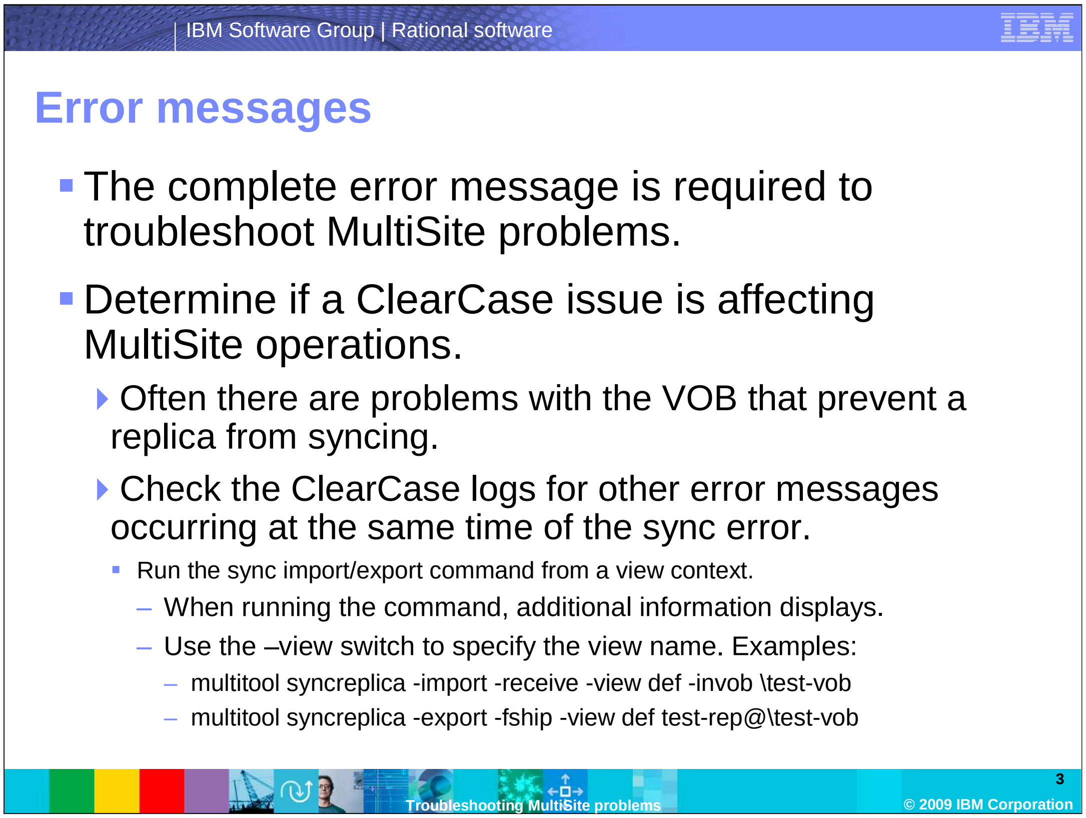
Module objectives

- These topics are covered in this module:
  - Error messages
  - Data Collection and Troubleshooting Tools
  - Troubleshooting Commands
  - Information about oplogs
  - Common errors
  - General recommendations
- This module enables users to:
  - Gain enough knowledge to know how to begin troubleshooting problems with MultiSite
  - Know what information to provide to support in the event of a complex problem

This module discusses Error messages, Data Collection and Troubleshooting Tools, Troubleshooting Commands, Information about oplogs, Common errors, and

General recommendations.

This module informs users how to begin troubleshooting problems with MultiSite. After completion of this module, users will also be familiar with what information to provide to support in the event of a complex problem.



IBM Software Group | Rational software

Error messages

- The complete error message is required to troubleshoot MultiSite problems.
- Determine if a ClearCase issue is affecting MultiSite operations.
  - Often there are problems with the VOB that prevent a replica from syncing.
  - Check the ClearCase logs for other error messages occurring at the same time of the sync error.
    - Run the sync import/export command from a view context.
      - When running the command, additional information displays.
      - Use the –view switch to specify the view name. Examples:
        - multitool syncreplica -import -receive -view def -invob \test-vob
        - multitool syncreplica -export -fship -view def test-rep@\test-vob

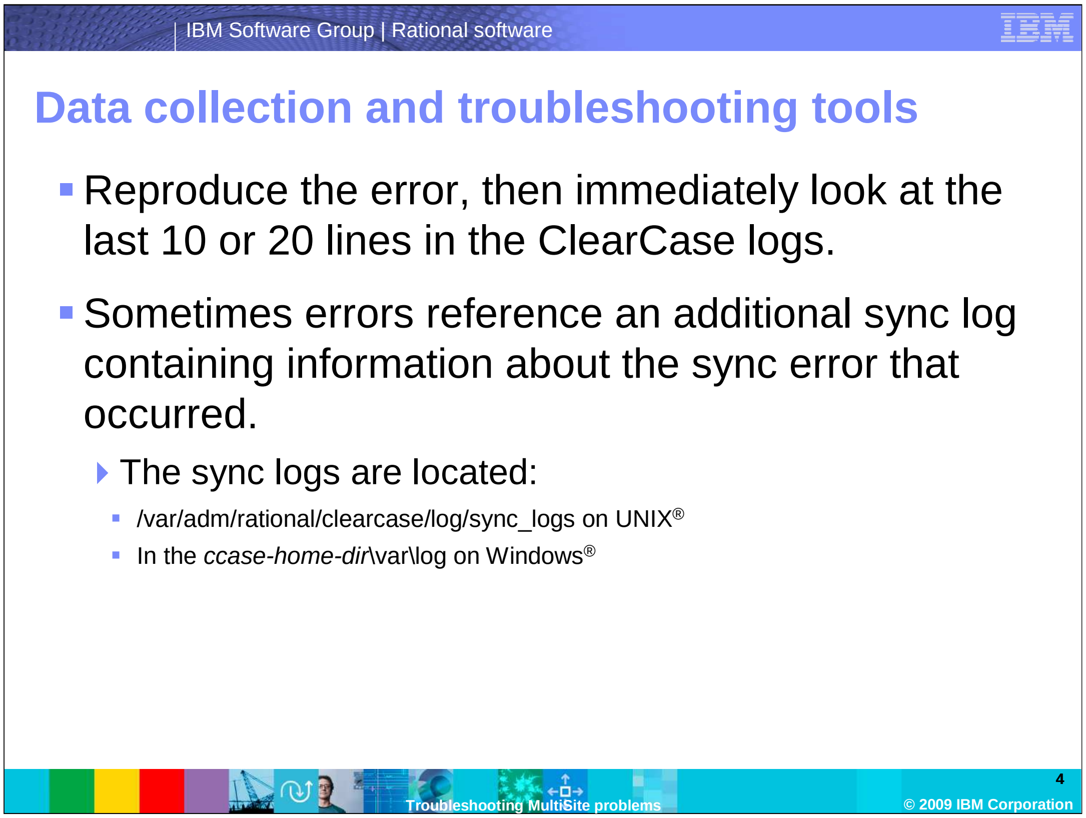
Troubleshooting MultiSite problems

3

© 2009 IBM Corporation

To troubleshoot MultiSite problems, the complete error message is required. Without a complete error message, it is impossible to proceed. It is also important to identify whether or not it is a ClearCase issue affecting MultiSite operations. Often there are problems with the VOB that prevent a replica from syncing, such as VOB container corruption. This prevents MultiSite from generating synchronization packets. When the error message occurs, check the ClearCase logs for other error messages occurring at the same time as the sync error.

When reproducing the error message, run the failing sync import/export command from a view context. In doing this, a user should see additional information to help diagnose the problem. For example, object identifiers are resolved to names. Use the –view switch to specify the view name. Examples of the command appear on the slide.



IBM Software Group | Rational software

Data collection and troubleshooting tools

- Reproduce the error, then immediately look at the last 10 or 20 lines in the ClearCase logs.
- Sometimes errors reference an additional sync log containing information about the sync error that occurred.
  - The sync logs are located:
    - /var/adm/rational/clearcase/log/sync_logs on UNIX®
    - In the *ccase-home-dir*\var\log on Windows®

Troubleshooting MultiSite problems

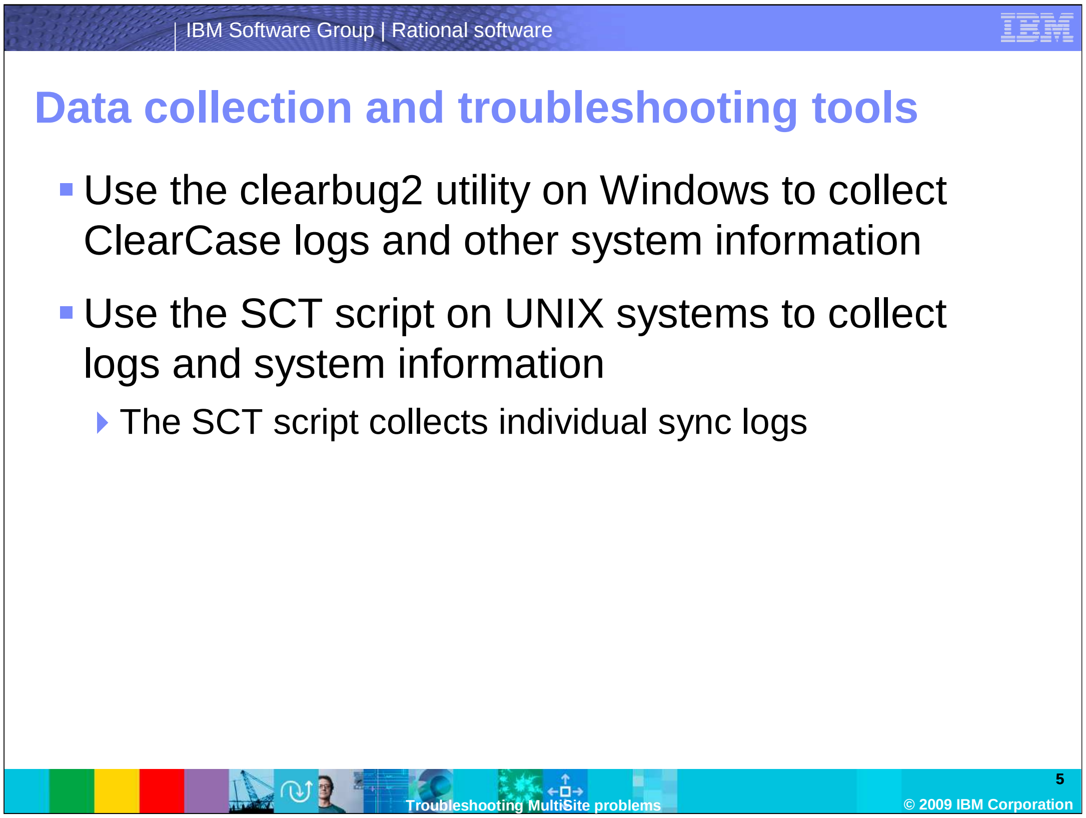
4

© 2009 IBM Corporation

Users have access to several troubleshooting methods.

A user can reproduce the error. Helpful error messages and information to diagnose the problem appear in the last 10 or 20 lines of ClearCase logs.

A user can also run cleartool getlog -last 20 -all >redirect to a file, or examine the logs in the ClearCase Administration Console. Sometimes, the errors reference an additional sync log. These logs contain information about the sync error that took place. The locations of the sync logs are indicated on the slide.

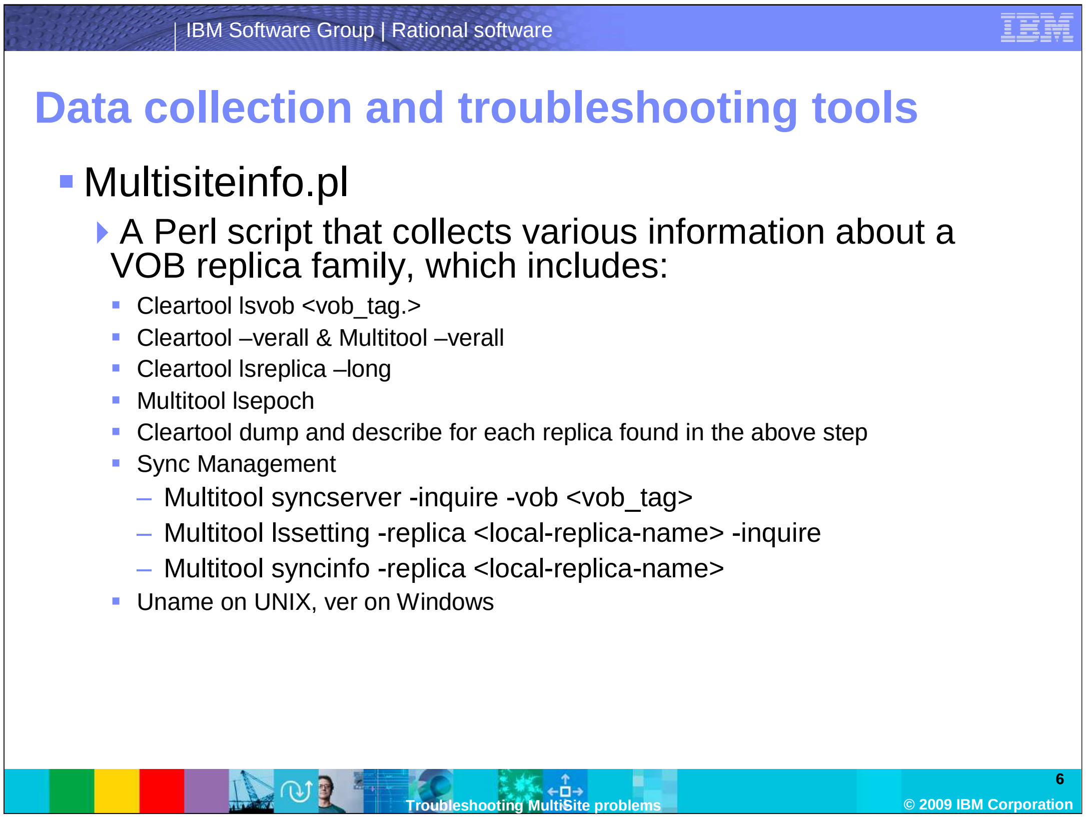


Data collection and troubleshooting tools

- Use the clearbug2 utility on Windows to collect ClearCase logs and other system information
- Use the SCT script on UNIX systems to collect logs and system information
  - The SCT script collects individual sync logs

Use the clearbug2 utility on Windows to help collect ClearCase logs and other information about the system. It is important to note that clearbug2 does not collect the individual sync logs. These are gathered separately and are sent to support.

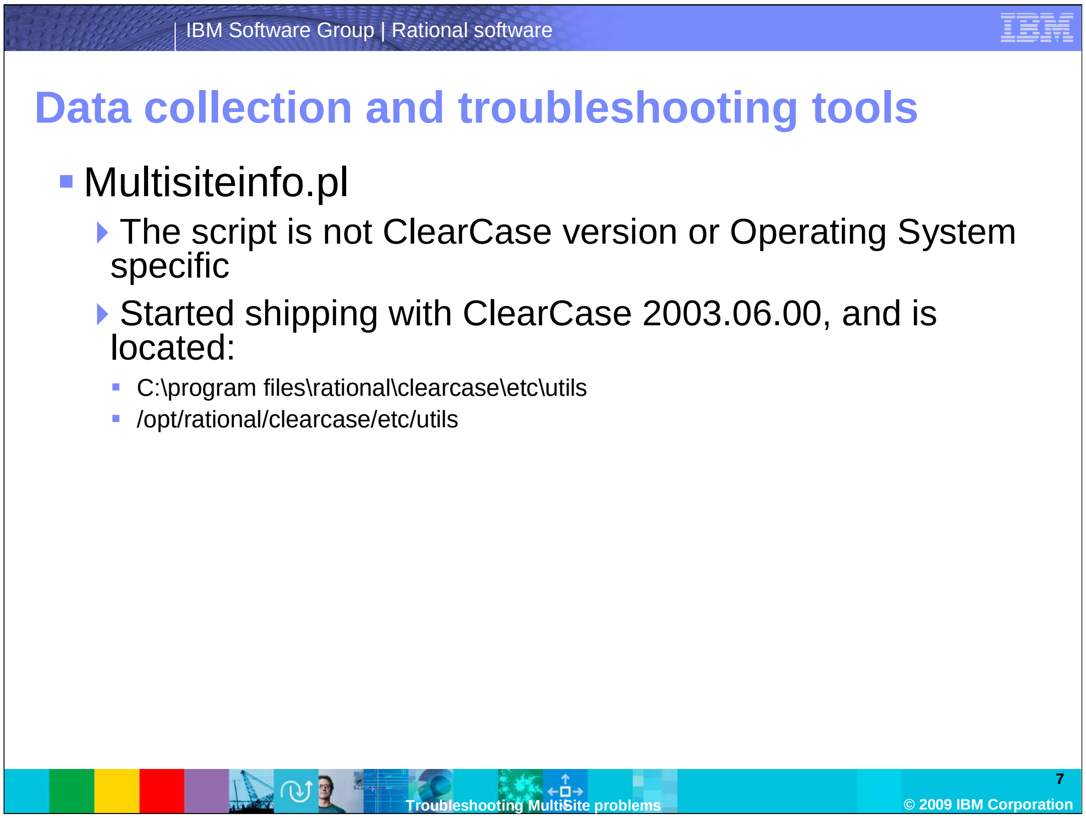
Use the SCT script on UNIX systems to collect logs and system information. The SCT script collects individual sync logs.



Data collection and troubleshooting tools

- Multisiteinfo.pl
  - A Perl script that collects various information about a VOB replica family, which includes:
    - Cleartool lsvob <vob_tag.>
    - Cleartool –verall & Multitool –verall
    - Cleartool lsreplica –long
    - Multitool lsepoch
    - Cleartool dump and describe for each replica found in the above step
    - Sync Management
      - Multitool syncserver -inquire -vob <vob_tag>
      - Multitool lssetting -replica <local-replica-name> -inquire
      - Multitool syncinfo -replica <local-replica-name>
    - Uname on UNIX, ver on Windows

Multisiteinfo.pl is a Perl script that collects various pieces of information about a VOB replica family. It includes several items, which are shown on this slide.



IBM Software Group | Rational software

Data collection and troubleshooting tools

- Multisiteinfo.pl
  - The script is not ClearCase version or Operating System specific
  - Started shipping with ClearCase 2003.06.00, and is located:
    - C:\program files\rational\clearcase\etc\utils
    - /opt/rational/clearcase/etc/utils

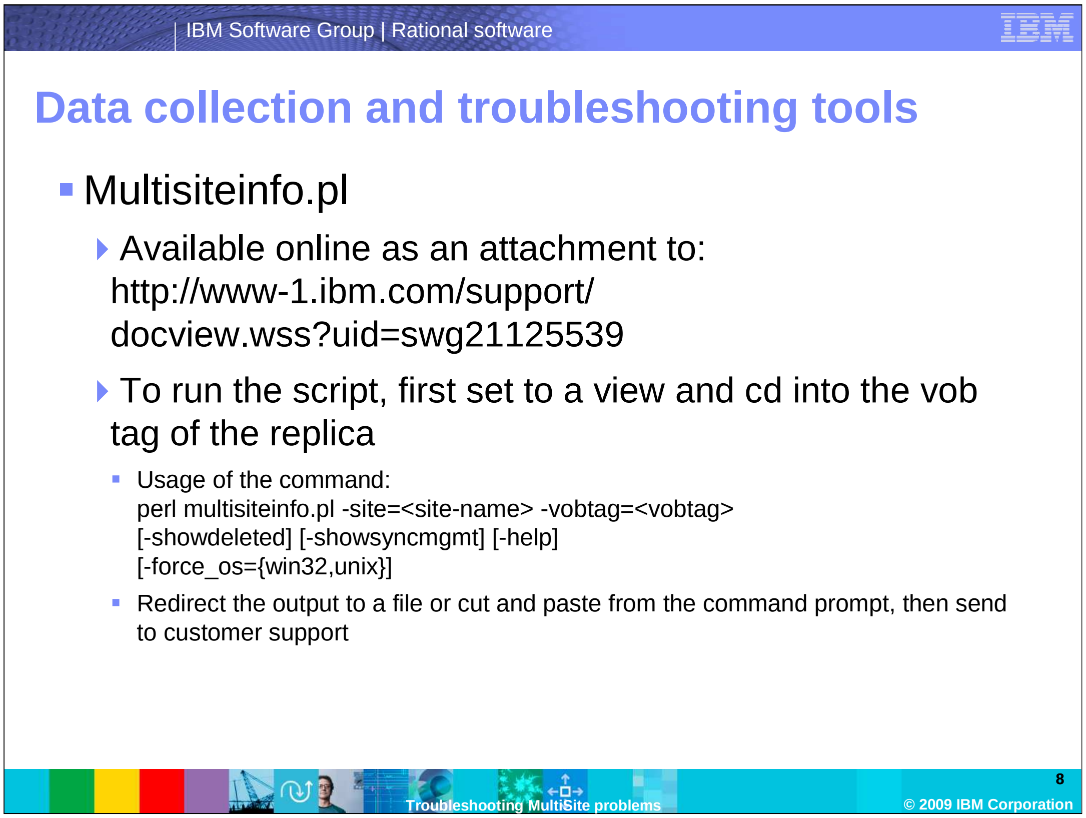
Troubleshooting MultiSite problems

7

© 2009 IBM Corporation

The multisiteinfo.pl script is not ClearCase version or Operating System specific, and started shipping with ClearCase 2003.06.00 in the directory shown on the slide.

There are other utilities that support can provide to customers on an as needed basis. The tool is sent based on what is needed for the specific case.

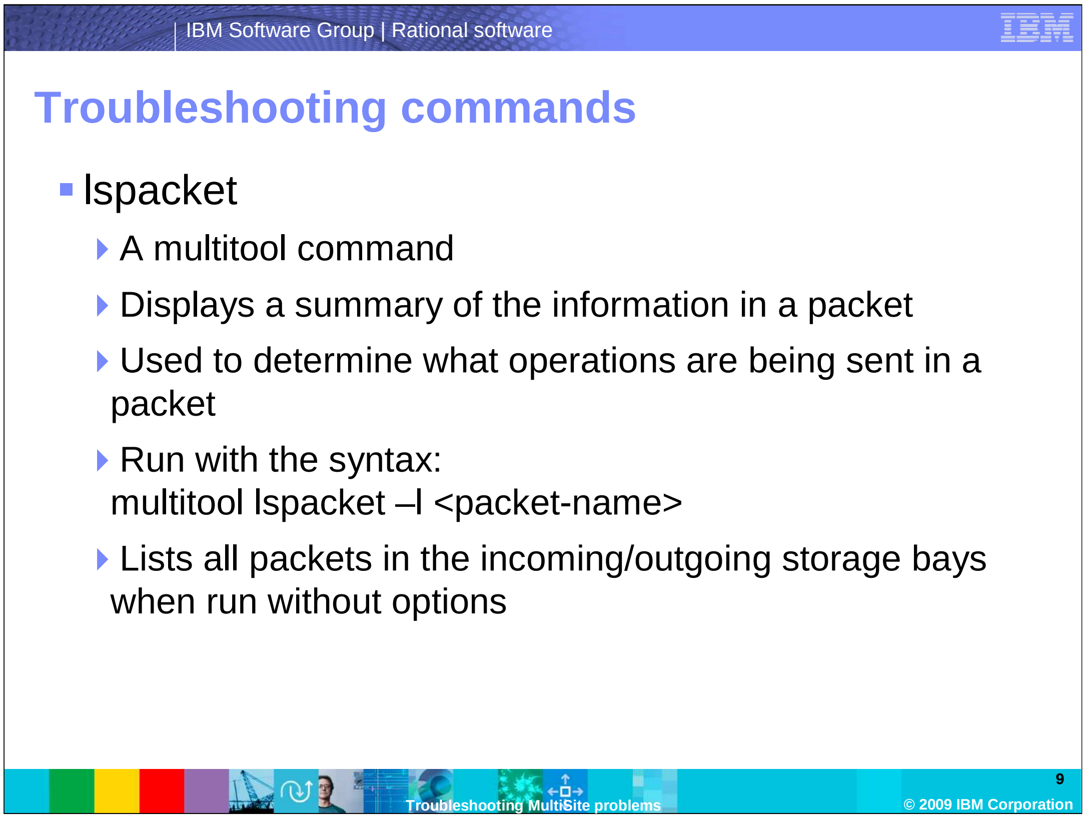


Data collection and troubleshooting tools

- Multisiteinfo.pl
  - Available online as an attachment to: http://www-1.ibm.com/support/docview.wss?uid=swg21125539
  - To run the script, first set to a view and cd into the vob tag of the replica
    - Usage of the command:
      perl multisiteinfo.pl -site=<site-name> -vobtag=<vobtag>
      [-showdeleted] [-showsyncmgmt] [-help]
      [-force_os={win32,unix}]
    - Redirect the output to a file or cut and paste from the command prompt, then send to customer support

The multisiteinfo.pl script is also available online as an attachment to the technote listed on this slide.

To run the script, first set to a view and cd into the vob tag of the replica. The usage of the command is shown on the slide. At the very end of the command, redirect the output to a file or cut and paste it from the command prompt. Send this file can to customer support for further analysis.



Troubleshooting commands

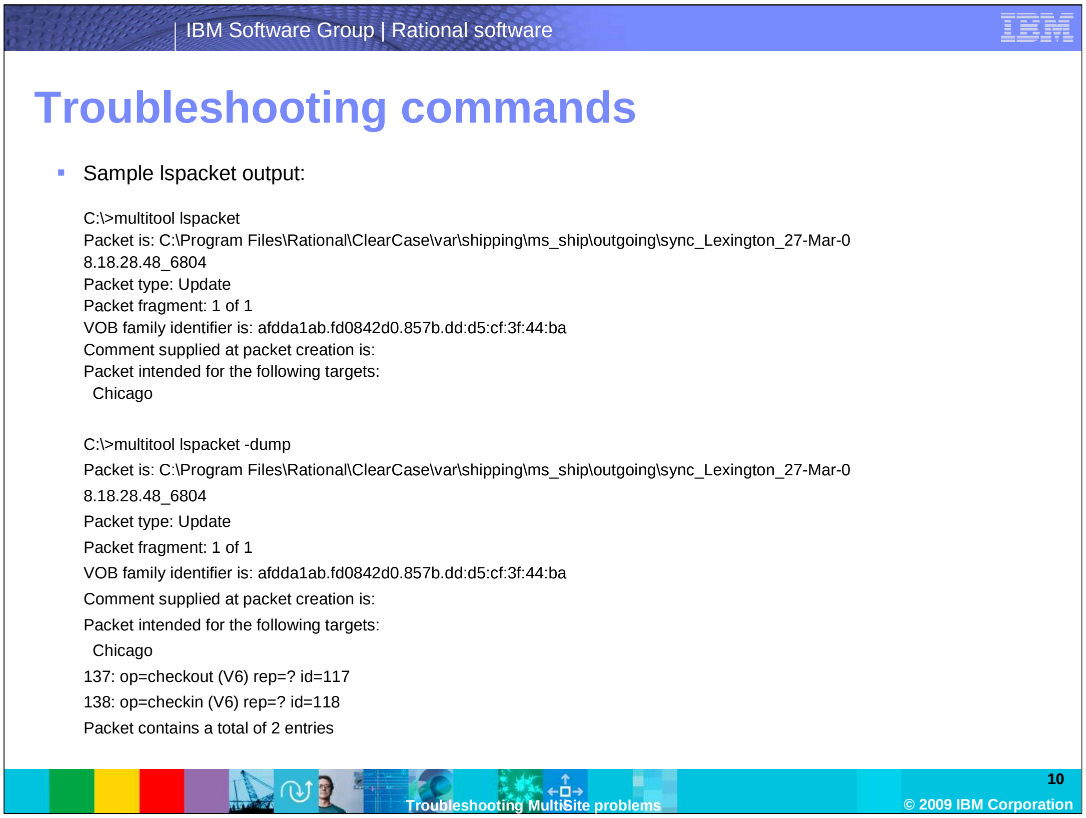
- lspacket
  - A multitool command
  - Displays a summary of the information in a packet
  - Used to determine what operations are being sent in a packet
  - Run with the syntax:
    multitool lspacket –l <packet-name>
  - Lists all packets in the incoming/outgoing storage bays when run without options

This section of the module details troubleshooting commands when using Rational ClearCase MultiSite.

Lspacket is a multitool command that displays a summary of the information in a packet. It is used when a user must determine what operations are being sent in a packet. It is also useful for determining how many packets are sitting on a system, whether they are waiting to be imported, or are leftover stale packets.

The command is run with the syntax shown on the slide.

When run without options, it lists all packets in the incoming and outgoing storage bays. Using the –dump (undocumented) switch allows more detailed output, including oplog details. Users should be set to a view to properly resolve the object names.

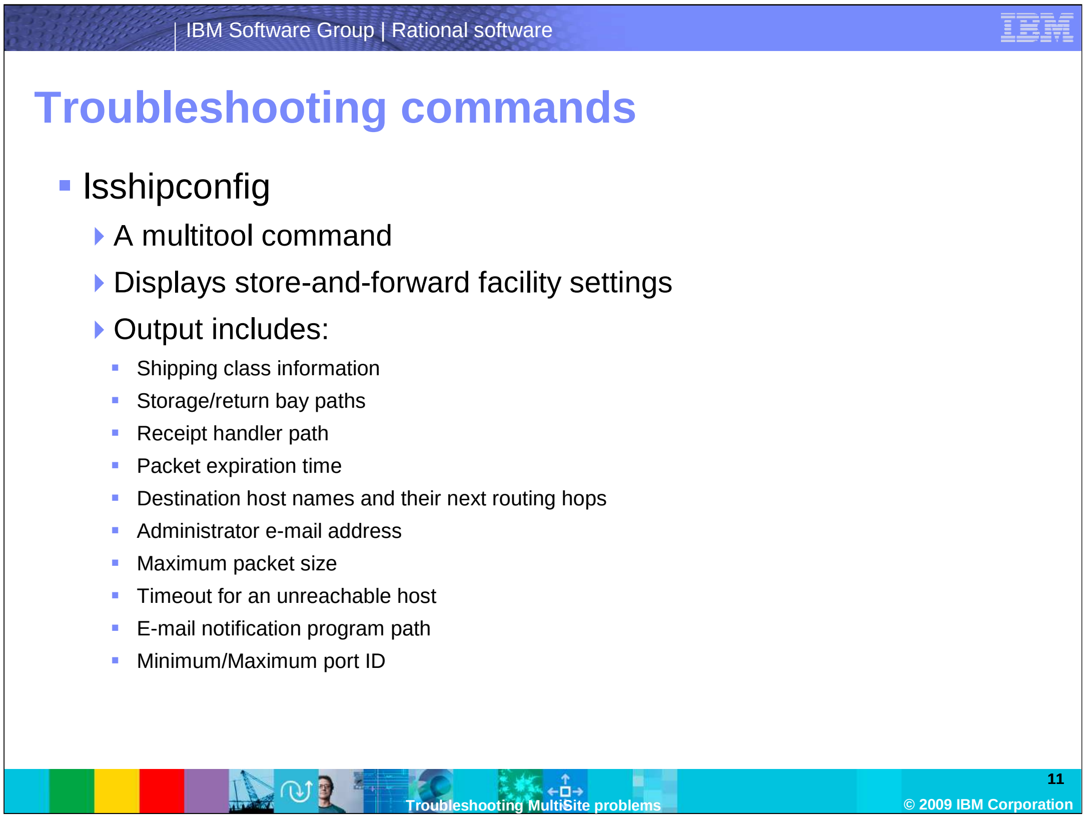


Troubleshooting commands

- Sample lspacket output:

```
C:\>multitool lspacket
Packet is: C:\Program Files\Rational\ClearCase\var\shipping\ms_ship\outgoing\sync_Lexington_27-Mar-0
8.18.28.48_6804
Packet type: Update
Packet fragment: 1 of 1
VOB family identifier is: afdda1ab.fd0842d0.857b.dd:d5:cf:3f:44:ba
Comment supplied at packet creation is:
Packet intended for the following targets:
  Chicago

C:\>multitool lspacket -dump
Packet is: C:\Program Files\Rational\ClearCase\var\shipping\ms_ship\outgoing\sync_Lexington_27-Mar-0
8.18.28.48_6804
Packet type: Update
Packet fragment: 1 of 1
VOB family identifier is: afdda1ab.fd0842d0.857b.dd:d5:cf:3f:44:ba
Comment supplied at packet creation is:
Packet intended for the following targets:
  Chicago
137: op=checkout (V6) rep=? id=117
138: op=checkin (V6) rep=? id=118
Packet contains a total of 2 entries
```

This slide lists an lspacket sample output. It shows the destination target replicas, the path of the packet, and the type of packet. Note the difference between the output with the –dump switch and the one without. The –dump switch will list what operations are replayed in each packet. This information is useful when troubleshooting lost packet scenarios.



IBM Software Group | Rational software

Troubleshooting commands

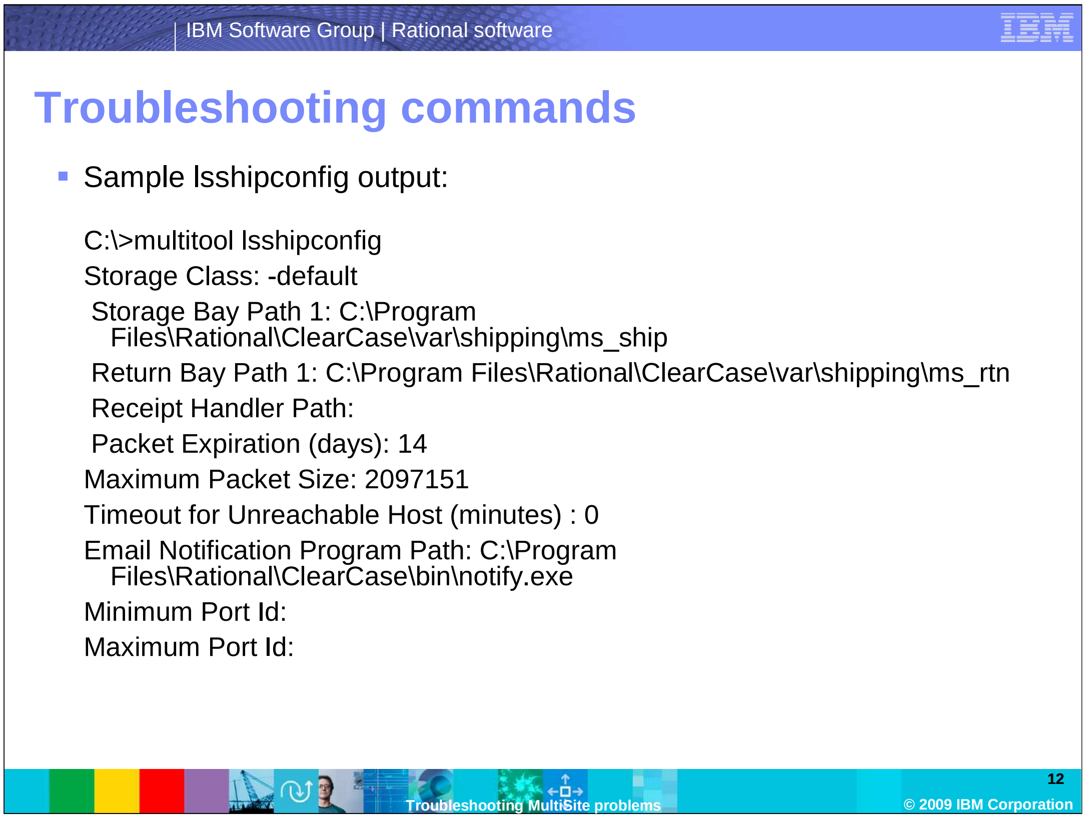
- lsshipconfig
  - A multitool command
  - Displays store-and-forward facility settings
  - Output includes:
    - Shipping class information
    - Storage/return bay paths
    - Receipt handler path
    - Packet expiration time
    - Destination host names and their next routing hops
    - Administrator e-mail address
    - Maximum packet size
    - Timeout for an unreachable host
    - E-mail notification program path
    - Minimum/Maximum port ID

Troubleshooting MultiSite problems

11

© 2009 IBM Corporation

Lsshipconfig is a multitool command that displays store-and-forward facility settings. The output for this command includes shipping class information, storage/return bay paths, the receipt handler path, the packet expiration time, destination host names and their next routing hops, the administrator’s e-mail address, the maximum packet size, the timeout for an unreachable host, the e-mail notification program path, and the minimum/maximum port ID set from the CLEARCASE_MIN_PORT and MAX_PORT variables.

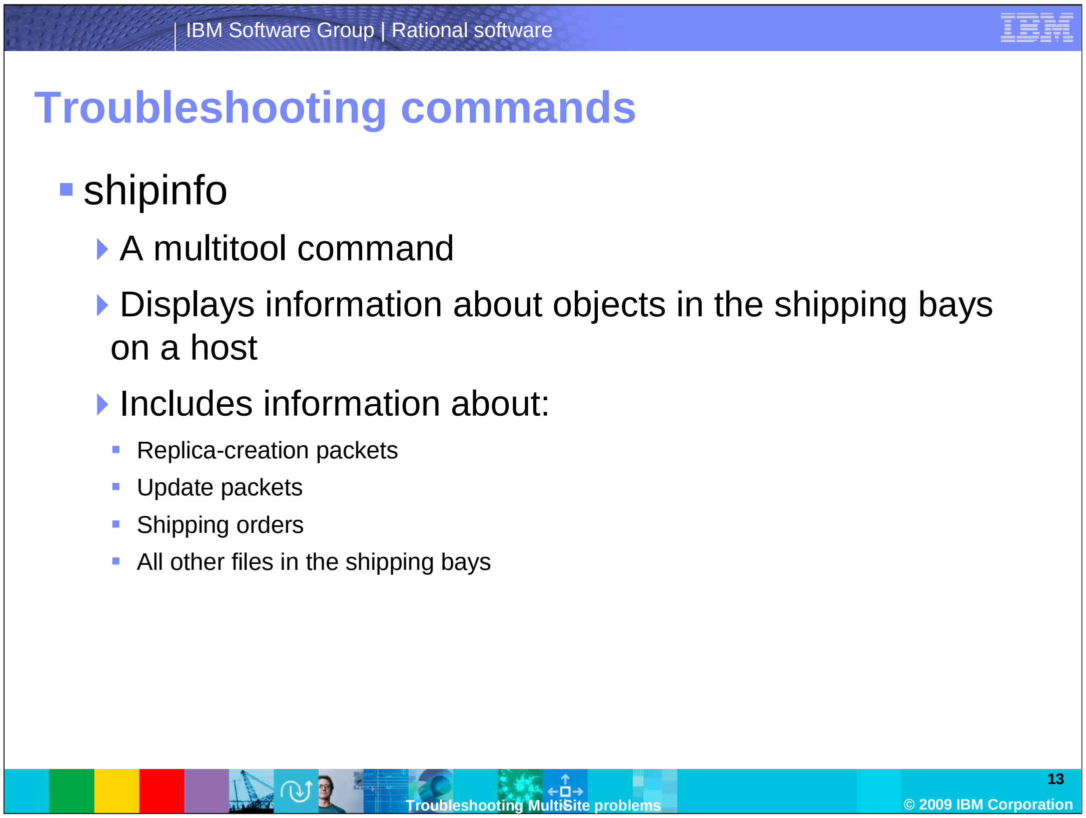


Troubleshooting commands

- Sample lsshipconfig output:

```
C:\>multitool lsshipconfig
Storage Class: -default
 Storage Bay Path 1: C:\Program
   Files\Rational\ClearCase\var\shipping\ms_ship
 Return Bay Path 1: C:\Program Files\Rational\ClearCase\var\shipping\ms_rtn
 Receipt Handler Path:
 Packet Expiration (days): 14
Maximum Packet Size: 2097151
Timeout for Unreachable Host (minutes) : 0
Email Notification Program Path: C:\Program
   Files\Rational\ClearCase\bin\notify.exe
Minimum Port Id:
Maximum Port Id:
```

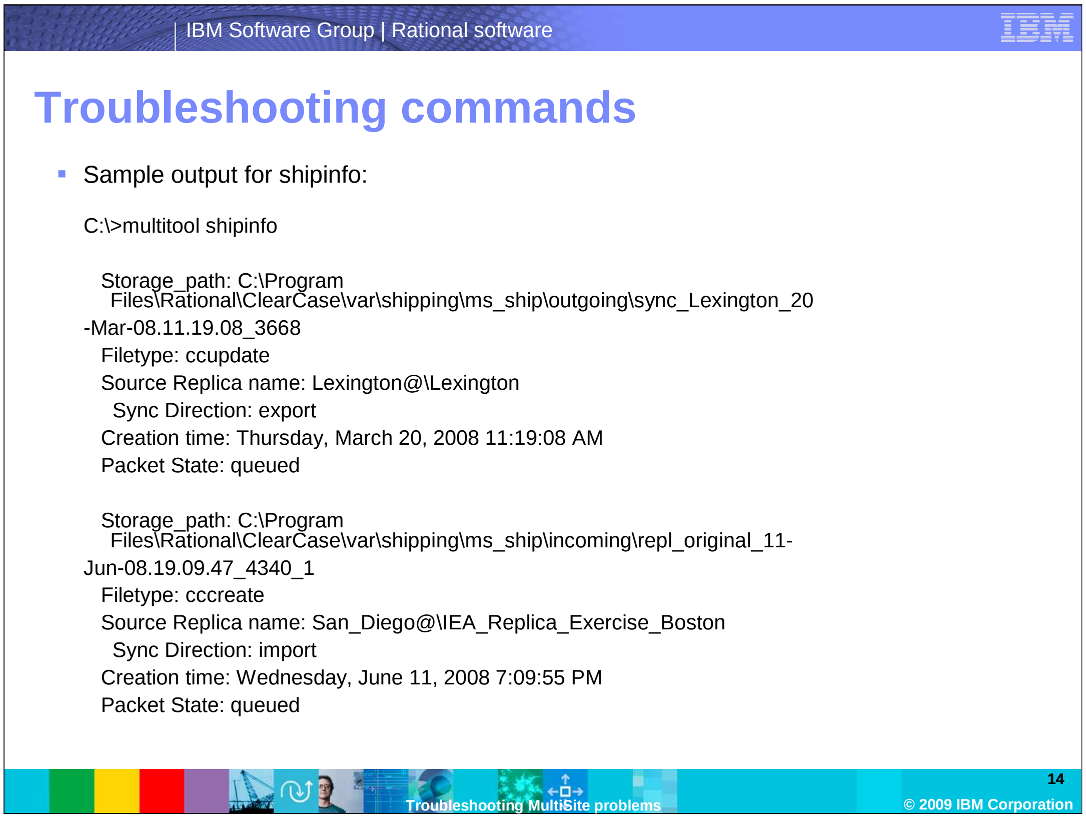
This slide lists a sample output from the lsshipconfig command. The output shows that the system is set up to use the default storage bay class, that it is not using a receipt handler, that the packet expiration is set to the default of 14 days, that the maximum packet size is 2097151 kilobytes, and that the system does not have the CLEARCASE_MIN_PORT and CLEARCASE_MAX_PORT environment variables set.



Troubleshooting commands

- shipinfo
  - A multitool command
  - Displays information about objects in the shipping bays on a host
  - Includes information about:
    - Replica-creation packets
    - Update packets
    - Shipping orders
    - All other files in the shipping bays

Shipinfo is a multitool command that displays information about objects in the shipping bays on a host. It includes information about replica-creation packets, update packets, shipping orders, and all other files in the shipping bays.



Troubleshooting commands

- Sample output for shipinfo:

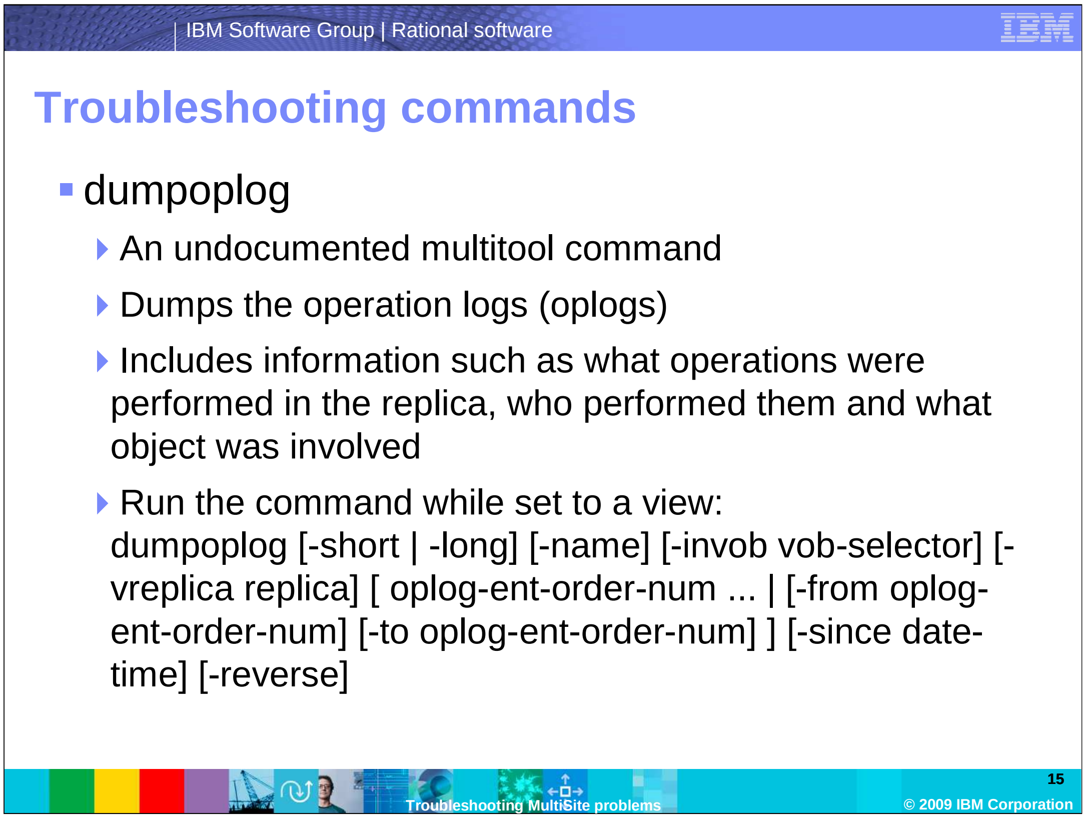
```
C:\>multitool shipinfo

  Storage_path: C:\Program
    Files\Rational\ClearCase\var\shipping\ms_ship\outgoing\sync_Lexington_20
-Mar-08.11.19.08_3668
  Filetype: ccupdate
  Source Replica name: Lexington@\Lexington
    Sync Direction: export
  Creation time: Thursday, March 20, 2008 11:19:08 AM
  Packet State: queued

  Storage_path: C:\Program
    Files\Rational\ClearCase\var\shipping\ms_ship\incoming\repl_original_11-
Jun-08.19.09.47_4340_1
  Filetype: cccreate
  Source Replica name: San_Diego@\IEA_Replica_Exercise_Boston
    Sync Direction: import
  Creation time: Wednesday, June 11, 2008 7:09:55 PM
  Packet State: queued
```

This slide lists a sample output from the shipinfo command. Shipinfo is useful in determining if there are old or stale replica creation packets on a system, and for determining what packets are waiting to be imported or exported.

This sample shows that there are two packets in the shipping bays.

The first packet is an update sync packet, sitting in the outgoing shipping bay. It originated from the Lexington replica, and is an export packet.

The second packet is a replication packet, and is sitting in the incoming shipping bay. The source replica is the San Diego replica. If the packet was already used to create the replica, the creation packet can be deleted.



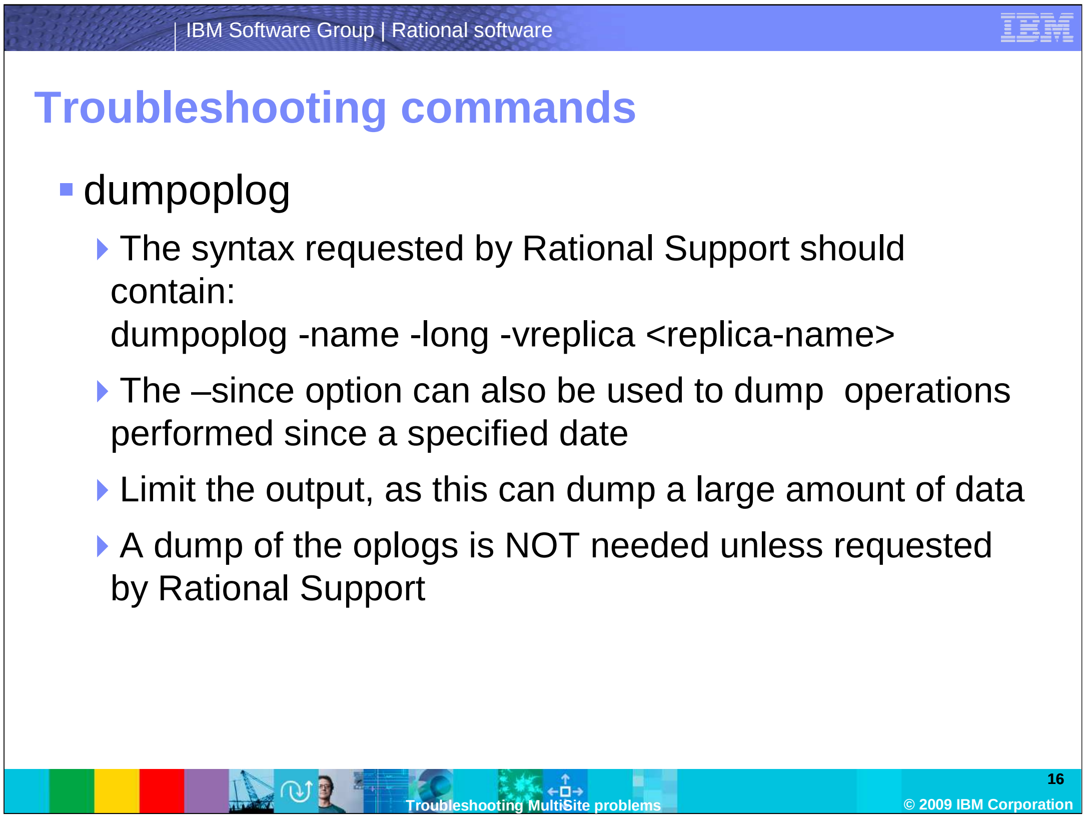
Troubleshooting commands

- dumpoplog
  - An undocumented multitool command
  - Dumps the operation logs (oplogs)
  - Includes information such as what operations were performed in the replica, who performed them and what object was involved
  - Run the command while set to a view:

```
dumpoplog [-short | -long] [-name] [-invob vob-selector] [-vreplica replica] [ oplog-ent-order-num ... | [-from oplog-ent-order-num] [-to oplog-ent-order-num] ] [-since date-time] [-reverse]
```

Dumpoplog is an undocumented multitool command that dumps the operation logs (oplogs). It includes information about the operations performed in the replica, who performed the actions, and the object that was involved.

When running the command, be sure to be set to a view. This slide displays the usage of the command.



Troubleshooting commands

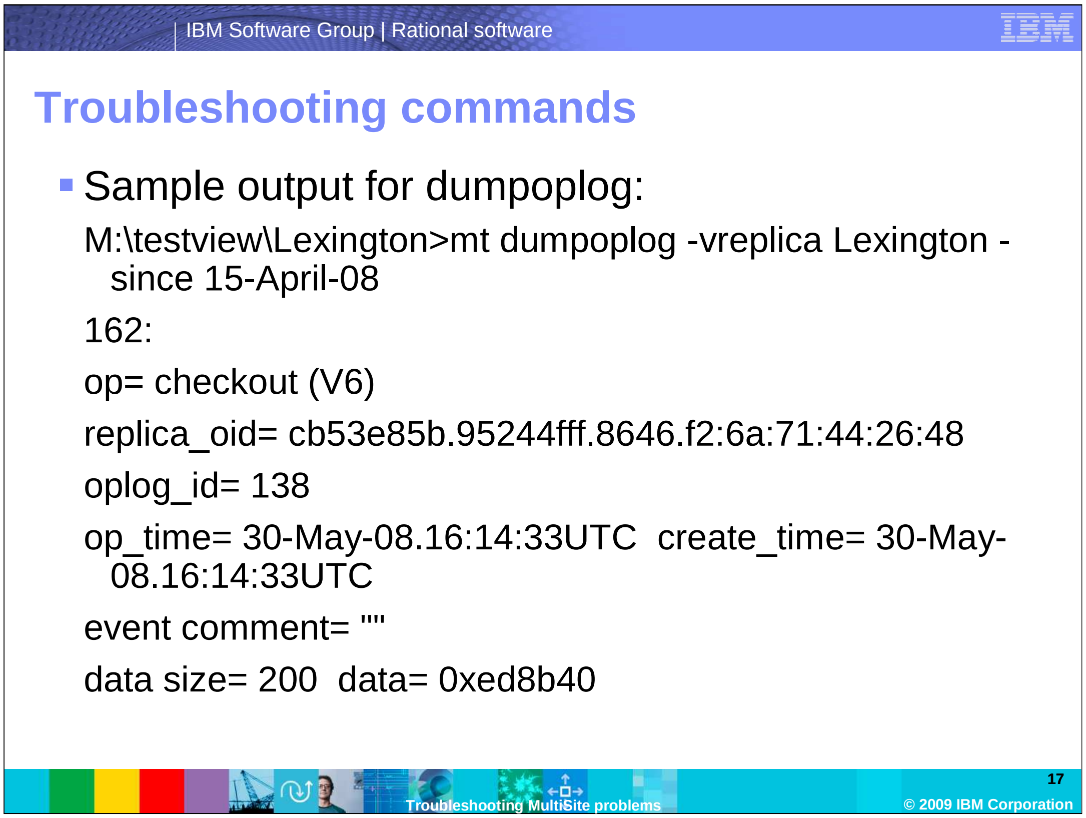
- dumpoplog
  - The syntax requested by Rational Support should contain:
    dumpoplog -name -long -vreplica <replica-name>
  - The –since option can also be used to dump operations performed since a specified date
  - Limit the output, as this can dump a large amount of data
  - A dump of the oplogs is NOT needed unless requested by Rational Support

When running dumpoplog, note the syntax requested by Rational Support. It should contain what is listed on this slide.

The –since option can also be used to dump operations performed since a specified date.

Dumpoplog can dump a large amount of data, so users should be sure to limit the output. If the correct switches are not specified, the command dumps every recorded oplog in the VOB database.

A dump of the oplogs is NOT normally required, unless specifically requested by Rational Support.

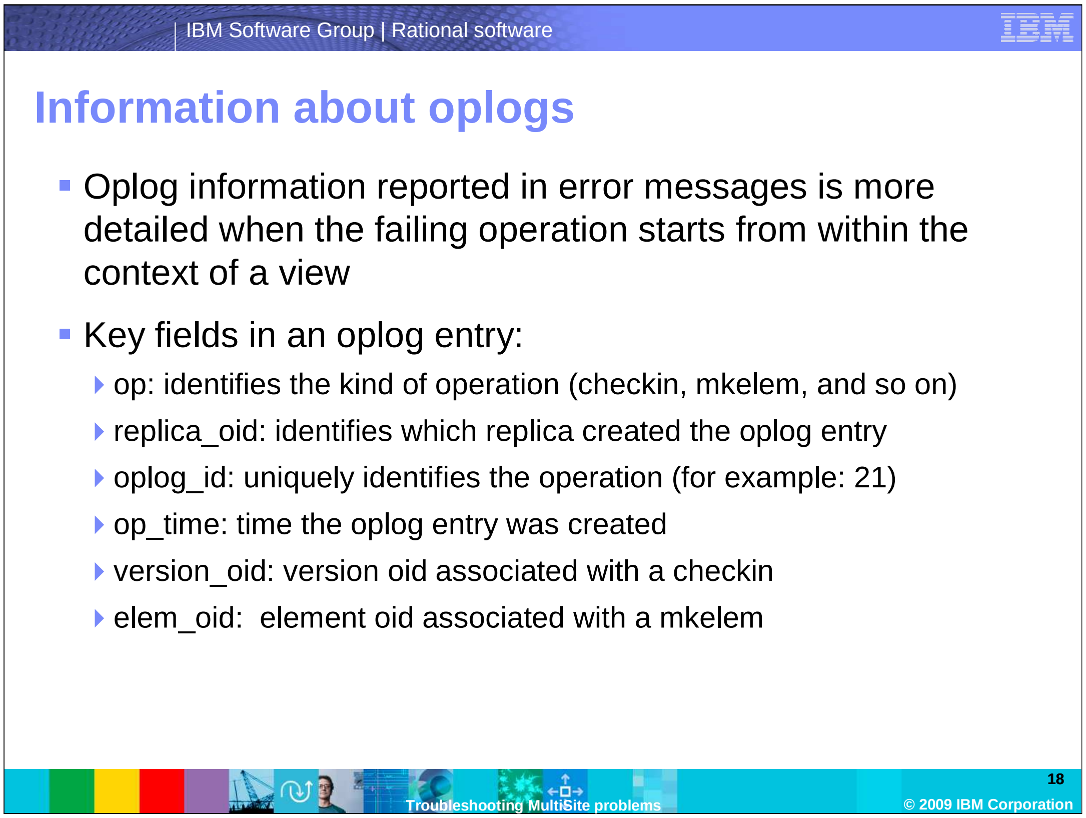


Troubleshooting commands

- Sample output for dumpoplog:

```
M:\testview\Lexington>mt dumpoplog -vreplica Lexington -
  since 15-April-08
162:
op= checkout (V6)
replica_oid= cb53e85b.95244fff.8646.f2:6a:71:44:26:48
oplog_id= 138
op_time= 30-May-08.16:14:33UTC  create_time= 30-May-
  08.16:14:33UTC
event comment= ""
data size= 200  data= 0xed8b40
```

This slide lists a sample output from the dumpoplog command. It shows that the oplog number is 126, that the operation performed was a checkout, the creation time, and the replica OID. The dumpoplog command is very useful in troubleshooting divergence problems. This command helps determine when the divergence occurred.



IBM Software Group | Rational software

Information about oplogs

- Oplog information reported in error messages is more detailed when the failing operation starts from within the context of a view
- Key fields in an oplog entry:
  - op: identifies the kind of operation (checkin, mkelem, and so on)
  - replica_oid: identifies which replica created the oplog entry
  - oplog_id: uniquely identifies the operation (for example: 21)
  - op_time: time the oplog entry was created
  - version_oid: version oid associated with a checkin
  - elem_oid: element oid associated with a mkelem

Troubleshooting MultiSite problems

18

© 2009 IBM Corporation

The oplog information reported in error messages is more detailed when the failing operation starts from within the context of a view.

There are several key fields contained in an oplog entry.

Op identifies the kind of operation, such as checkin, mkelem, and so on.

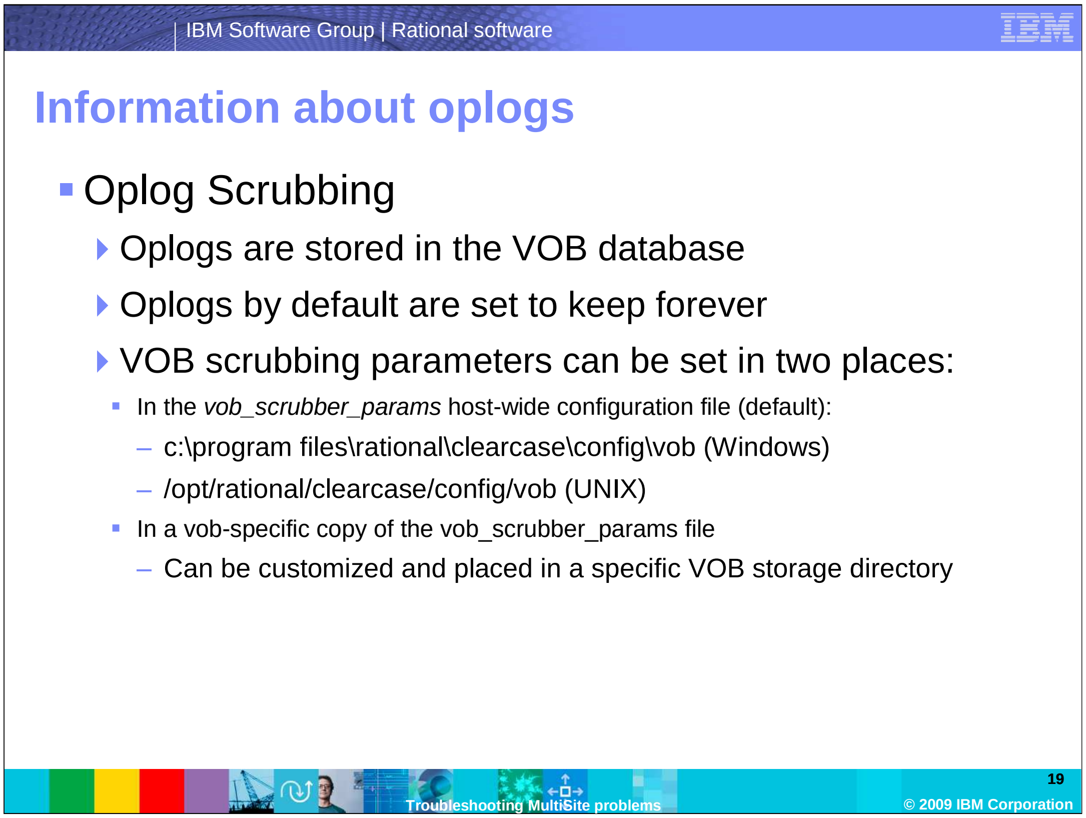
Replica_oid identifies which replica created the oplog entry.

Oplog_id uniquely identifies the operation, such as, for example, 21.

Op_time is the time the oplog entry was created.

Version_oid is the version oid associated with a checkin.

Elem_oid is the element oid associated with a mkelem.



IBM Software Group | Rational software

Information about oplogs

- Oplog Scrubbing
  - Oplogs are stored in the VOB database
  - Oplogs by default are set to keep forever
  - VOB scrubbing parameters can be set in two places:
    - In the *vob_scrubber_params* host-wide configuration file (default):
      - c:\program files\rational\clearcase\config\vob (Windows)
      - /opt/rational/clearcase/config/vob (UNIX)
    - In a vob-specific copy of the vob_scrubber_params file
      - Can be customized and placed in a specific VOB storage directory

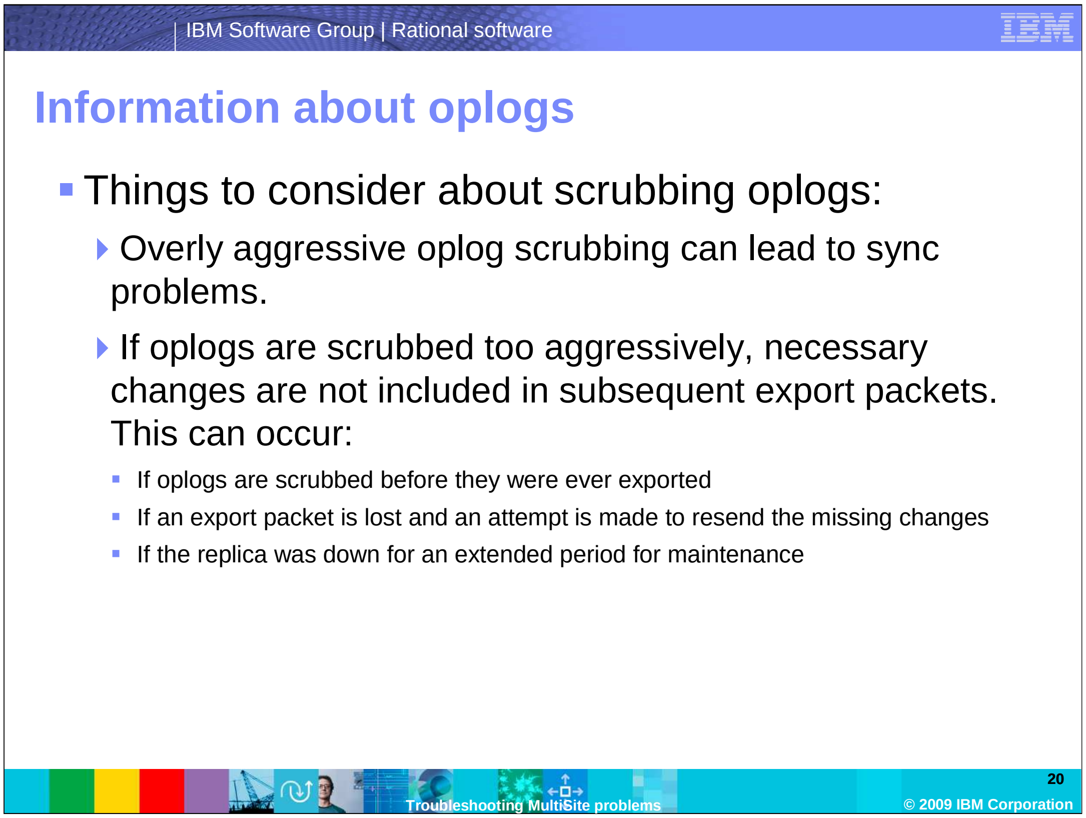
Troubleshooting MultiSite problems

19

© 2009 IBM Corporation

Oplogs can be scrubbed when disk space becomes a problem. They are stored in the VOB database, and by default, are set to keep forever.

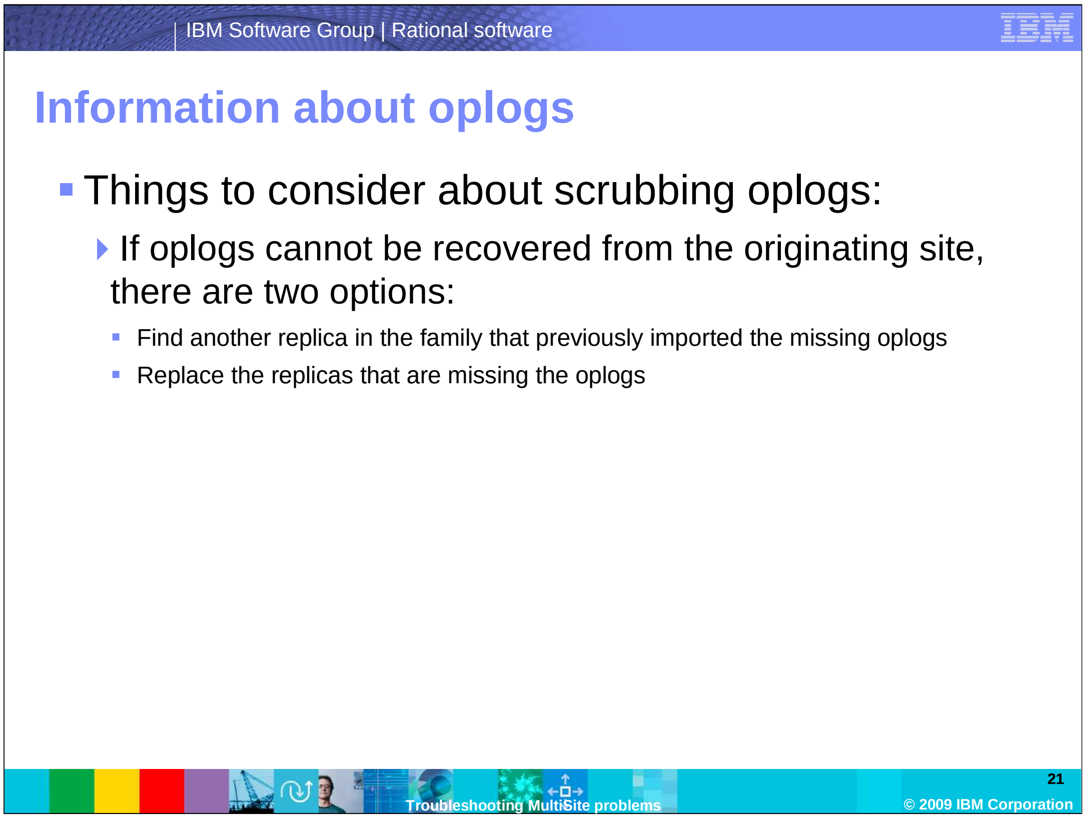
VOB scrubbing parameters can be set in two places. They can be set in the *vob_scrubber_params* host-wide configuration file, which is kept in one of the locations shown on the slide. Parameters can also be stored in a vob-specific copy of the vob_scrubber_params file, which can be customized and placed in a specific VOB storage directory.



Information about oplogs

- Things to consider about scrubbing oplogs:
  - Overly aggressive oplog scrubbing can lead to sync problems.
  - If oplogs are scrubbed too aggressively, necessary changes are not included in subsequent export packets. This can occur:
    - If oplogs are scrubbed before they were ever exported
    - If an export packet is lost and an attempt is made to resend the missing changes
    - If the replica was down for an extended period for maintenance

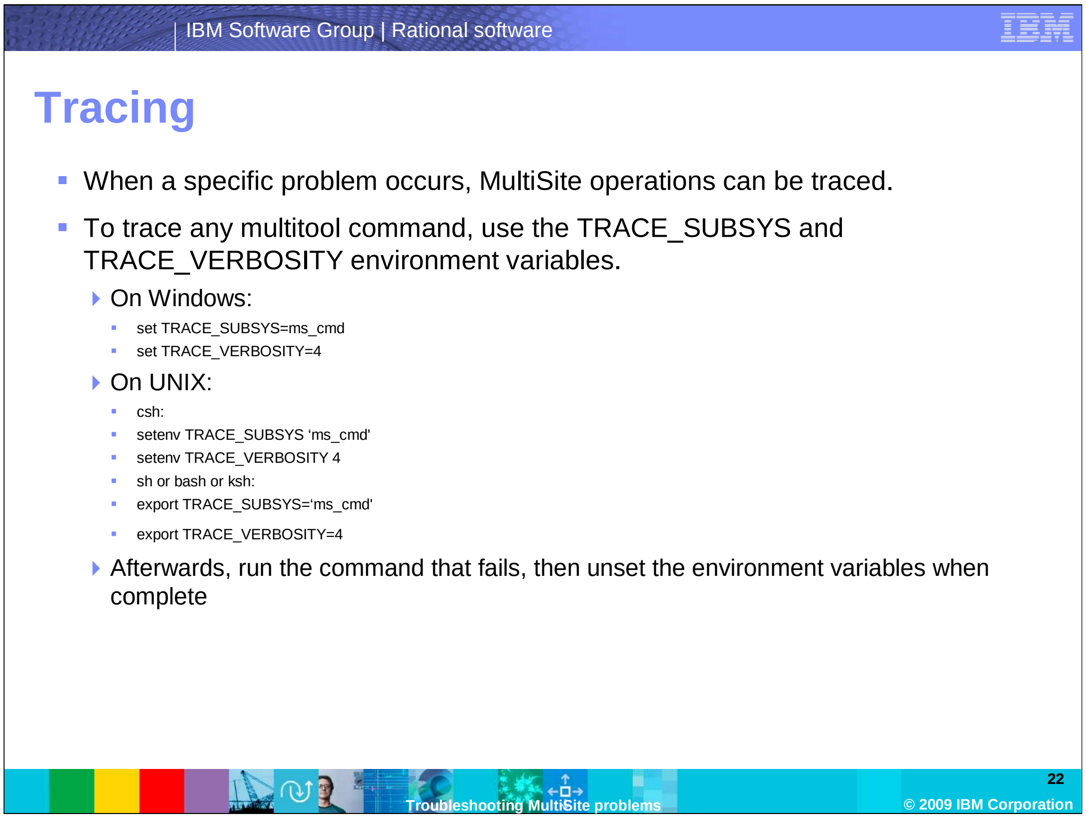
There are some important factors to consider when scrubbing oplogs, as this can be a risky procedure if not done and planned for properly. Overly aggressive oplog scrubbing can lead to sync problems. Also, if the oplogs are scrubbed too aggressively, necessary changes are not included in subsequent export packets. This can occur if oplogs are scrubbed before they were ever exported, if an export packet is lost and an attempt is made to resend the missing changes, or if the replica was down for an extended period for maintenance.



Information about oplogs

- Things to consider about scrubbing oplogs:
  - If oplogs cannot be recovered from the originating site, there are two options:
    - Find another replica in the family that previously imported the missing oplogs
    - Replace the replicas that are missing the oplogs

If the oplogs cannot be recovered from the originating site, then there are two options. Either find another replica in the family that previously imported the missing oplogs, or replace the replicas that are missing the oplogs.



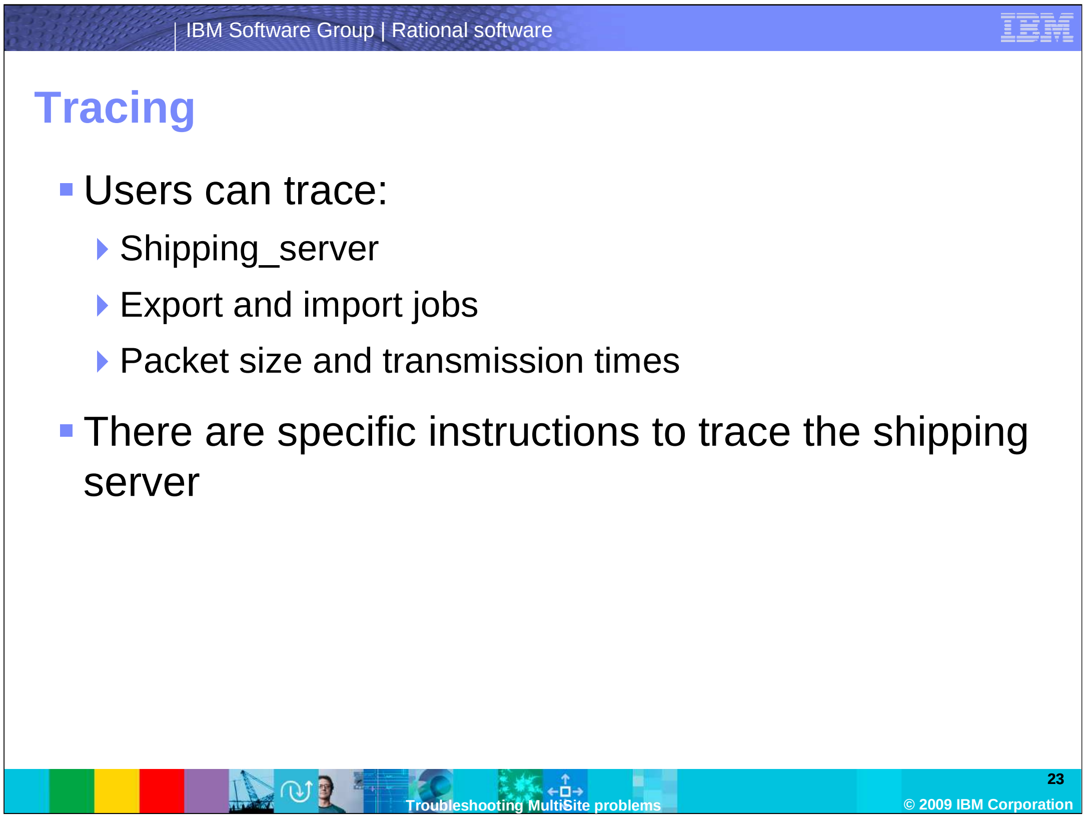
Tracing

- When a specific problem occurs, MultiSite operations can be traced.
- To trace any multitool command, use the TRACE_SUBSYS and TRACE_VERBOSITY environment variables.
  - On Windows:
    - set TRACE_SUBSYS=ms_cmd
    - set TRACE_VERBOSITY=4
  - On UNIX:
    - csh:
    - setenv TRACE_SUBSYS 'ms_cmd'
    - setenv TRACE_VERBOSITY 4
    - sh or bash or ksh:
    - export TRACE_SUBSYS=‘ms_cmd'
    - export TRACE_VERBOSITY=4
  - Afterwards, run the command that fails, then unset the environment variables when complete

When a specific problem occurs, MultiSite operations can be traced. Tracing is not required on every problem with MultiSite. Only the more complex issues require tracing outputs for resolution.

To trace any multitool command the TRACE_SUBSYS and TRACE_VERBOSITY environment variables are used. Examples of this appear on the slide.

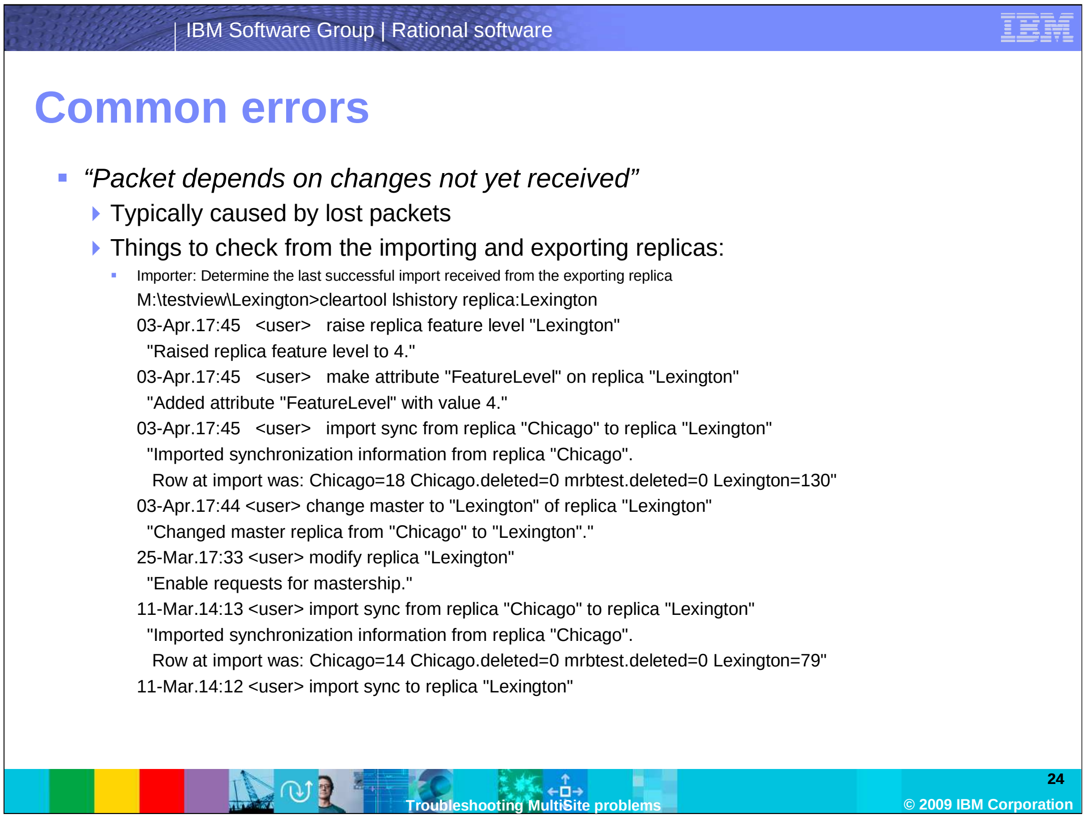
Once the variables are set, users can run the command that fails. After reproducing the problem, unset the environment variables.



Tracing

- Users can trace:
  - Shipping_server
  - Export and import jobs
  - Packet size and transmission times
- There are specific instructions to trace the shipping server

Users can also trace the shipping_server, export and import jobs, and the packet size and transmission times. There are specific instructions to trace the shipping server. There is a technote at the end of this module that discusses how to set up and run the tracing.



Common errors

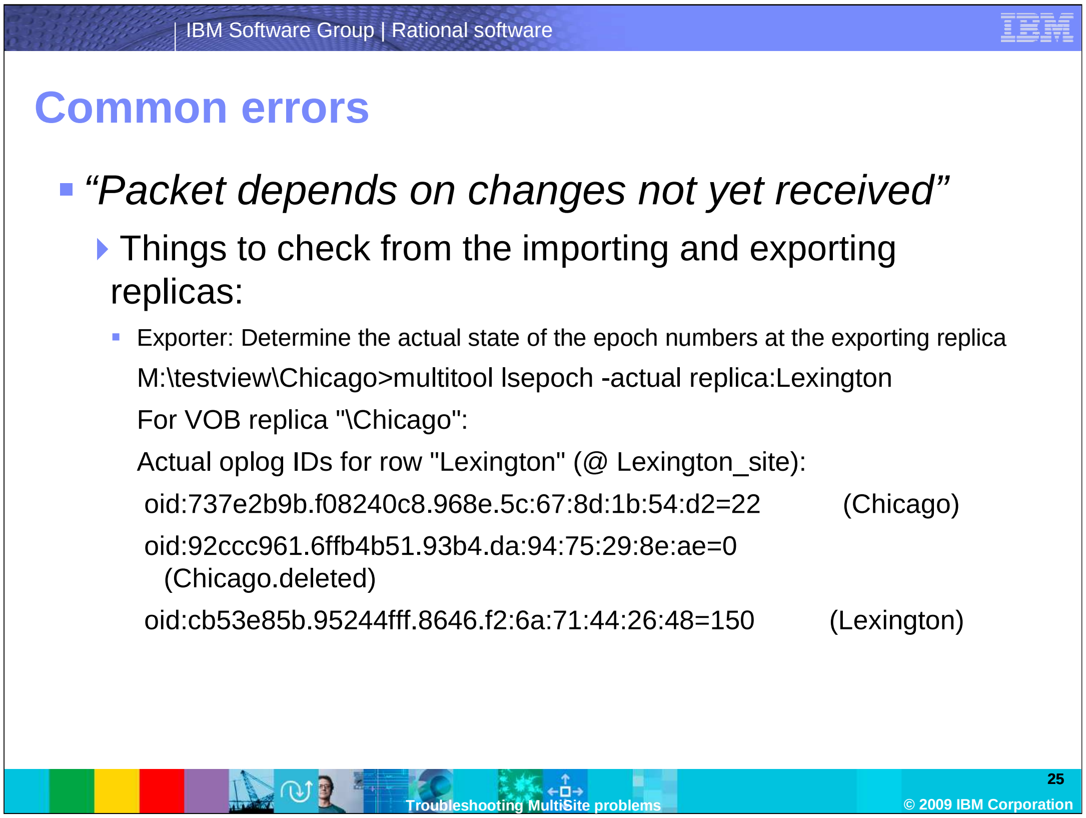
- *“Packet depends on changes not yet received”*
  - Typically caused by lost packets
  - Things to check from the importing and exporting replicas:
    - Importer: Determine the last successful import received from the exporting replica

```
M:\testview\Lexington>cleartool lshistory replica:Lexington
03-Apr.17:45   <user>   raise replica feature level "Lexington"
  "Raised replica feature level to 4."
03-Apr.17:45   <user>   make attribute "FeatureLevel" on replica "Lexington"
  "Added attribute "FeatureLevel" with value 4."
03-Apr.17:45   <user>   import sync from replica "Chicago" to replica "Lexington"
  "Imported synchronization information from replica "Chicago".
   Row at import was: Chicago=18 Chicago.deleted=0 mrbtest.deleted=0 Lexington=130"
03-Apr.17:44 <user> change master to "Lexington" of replica "Lexington"
  "Changed master replica from "Chicago" to "Lexington"."
25-Mar.17:33 <user> modify replica "Lexington"
  "Enable requests for mastership."
11-Mar.14:13 <user> import sync from replica "Chicago" to replica "Lexington"
  "Imported synchronization information from replica "Chicago".
   Row at import was: Chicago=14 Chicago.deleted=0 mrbtest.deleted=0 Lexington=79"
11-Mar.14:12 <user> import sync to replica "Lexington"
```

There are a set of errors that occur commonly in day to day operations of MultiSite.

“Packet depends on changes not yet received” is typically caused by lost packets. To troubleshoot this error, check items from the importing and exporting replicas.

At the Importer, determine the last successful import received from the exporting replica. To do this, use the cleartool lshistory command. Here, the lshistory sample command output details that the last successful import at the Lexington replica took place on April 3.

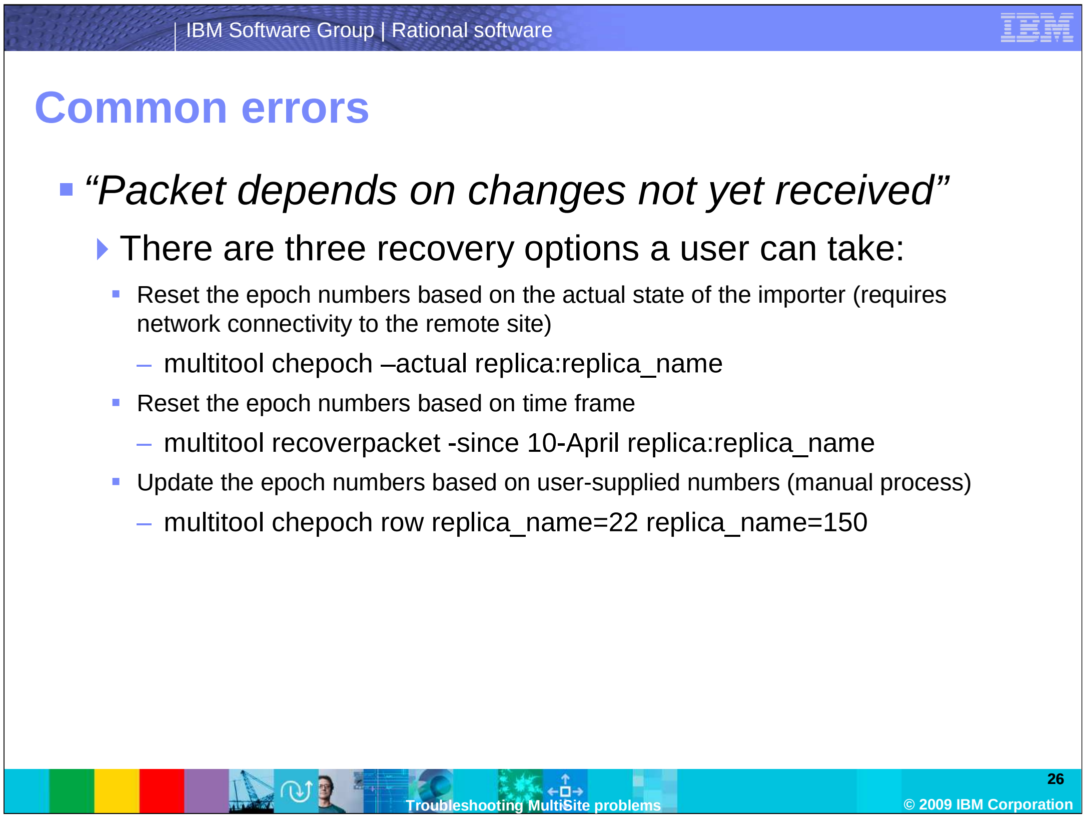


Common errors

- *“Packet depends on changes not yet received”*
  - Things to check from the importing and exporting replicas:
    - Exporter: Determine the actual state of the epoch numbers at the exporting replica

```
M:\testview\Chicago>multitool lsepoch -actual replica:Lexington
For VOB replica "\Chicago":
Actual oplog IDs for row "Lexington" (@ Lexington_site):
 oid:737e2b9b.f08240c8.968e.5c:67:8d:1b:54:d2=22          (Chicago)
 oid:92ccc961.6ffb4b51.93b4.da:94:75:29:8e:ae=0
   (Chicago.deleted)
 oid:cb53e85b.95244fff.8646.f2:6a:71:44:26:48=150        (Lexington)
```

From the exporting site, determine what the epoch table is with the lspoch –actual command. The epoch rows must be updated. This command contacts the remote site to retrieve the actual state of that replica’s epoch table.



IBM Software Group | Rational software

Common errors

- *“Packet depends on changes not yet received”*
  - There are three recovery options a user can take:
    - Reset the epoch numbers based on the actual state of the importer (requires network connectivity to the remote site)
      - multitool chepoch –actual replica:replica_name
    - Reset the epoch numbers based on time frame
      - multitool recoverpacket -since 10-April replica:replica_name
    - Update the epoch numbers based on user-supplied numbers (manual process)
      - multitool chepoch row replica_name=22 replica_name=150

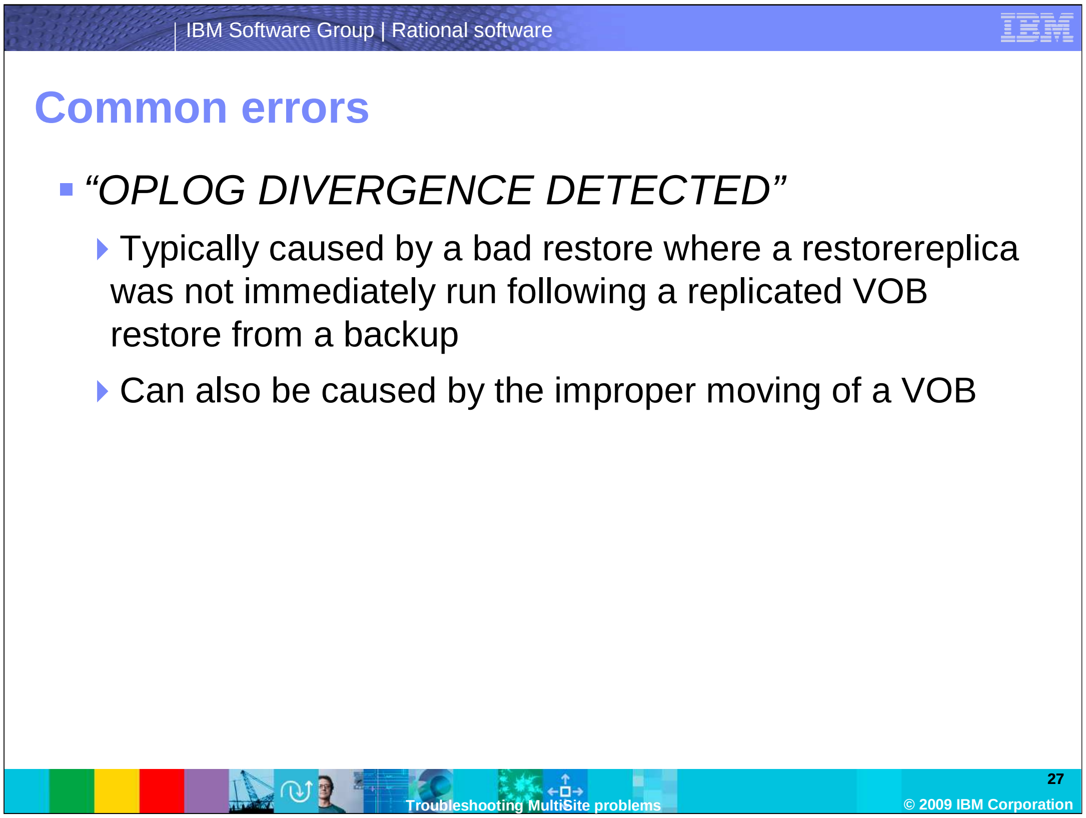
Troubleshooting MultiSite problems

26

© 2009 IBM Corporation

There are three recovery options a user can take to resolve the “*packet depends on changes not yet received*” error. The commands for each action are detailed on the slide.

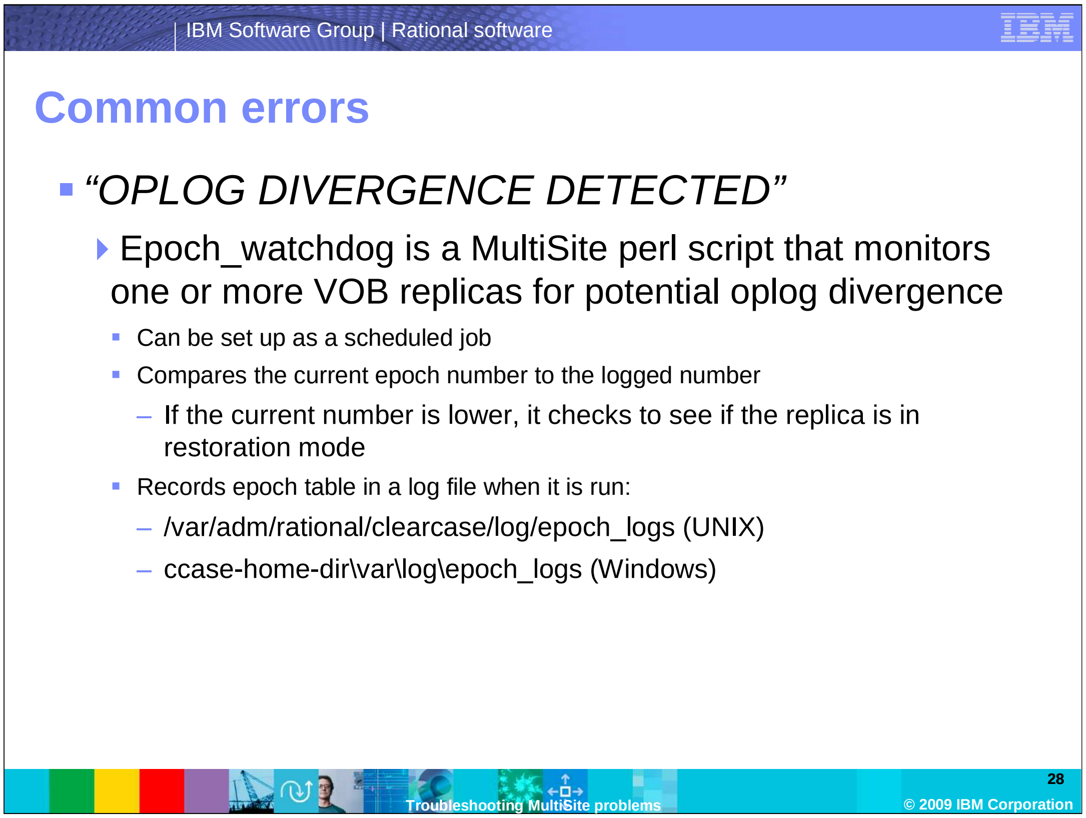
A user can reset the epoch numbers based on the actual state of the importer, which requires network connectivity to the remote site. A user can reset the epoch numbers based on time frame. Finally, a user can update the epoch numbers based on user-supplied numbers, which is a manual process.



Common errors

- *“OPLOG DIVERGENCE DETECTED”*
  - Typically caused by a bad restore where a restorereplica was not immediately run following a replicated VOB restore from a backup
  - Can also be caused by the improper moving of a VOB

“OPLOG DIVERGENCE DETECTED” is typically caused by a bad restore where a restorereplica was not immediately run after a replicated VOB restore from a backup. It can also be caused by the improper moving of a VOB. There are several methods and tools used to detect divergence.



IBM Software Group | Rational software

Common errors

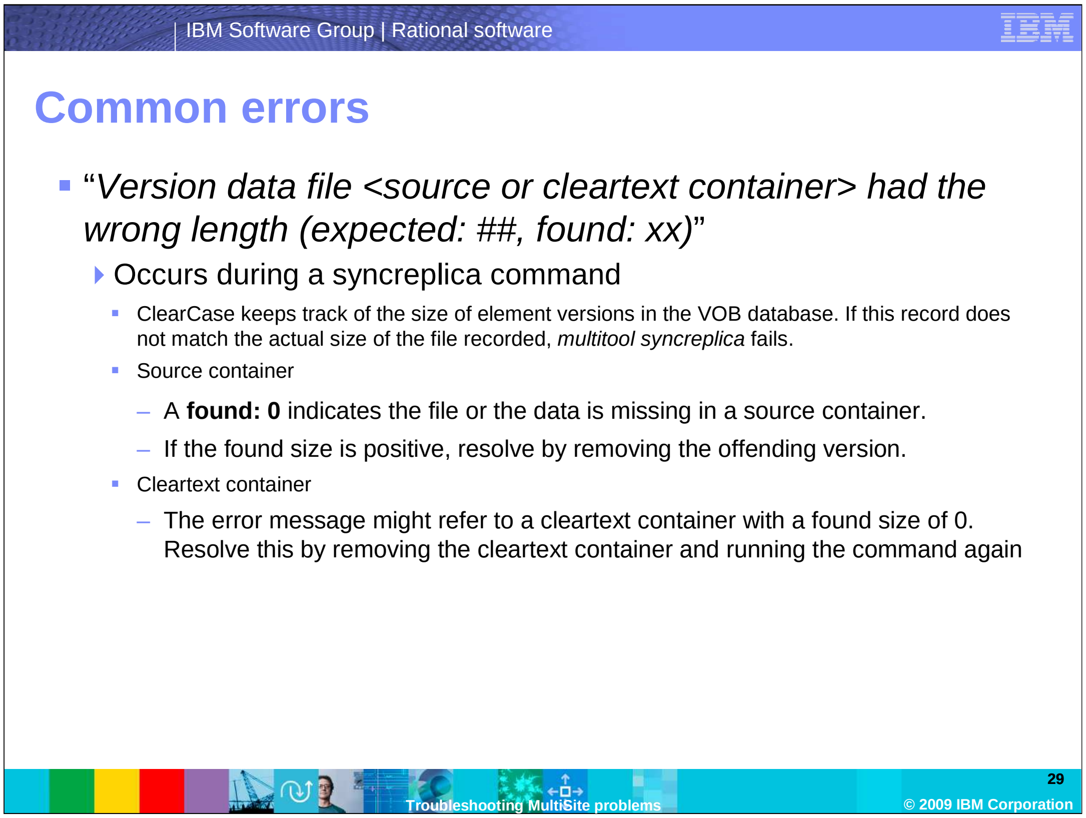
- *“OPLOG DIVERGENCE DETECTED”*
  - Epoch_watchdog is a MultiSite perl script that monitors one or more VOB replicas for potential oplog divergence
    - Can be set up as a scheduled job
    - Compares the current epoch number to the logged number
      - If the current number is lower, it checks to see if the replica is in restoration mode
    - Records epoch table in a log file when it is run:
      - /var/adm/rational/clearcase/log/epoch_logs (UNIX)
      - ccase-home-dir\var\log\epoch_logs (Windows)

Troubleshooting MultiSite problems

28

© 2009 IBM Corporation

Epoch_watchdog is a MultiSite perl script that monitors one or more VOB replicas for potential oplog divergence. The script can be set up as a scheduled job, and compares the current epoch number to the logged number. If the current number is lower, it checks to see if the replica is in restoration mode. If there is no restoration in progress for the related VOB, it attempts to lock the VOB. When it is run, it records the epoch table in a log file in the location shown on the slide.



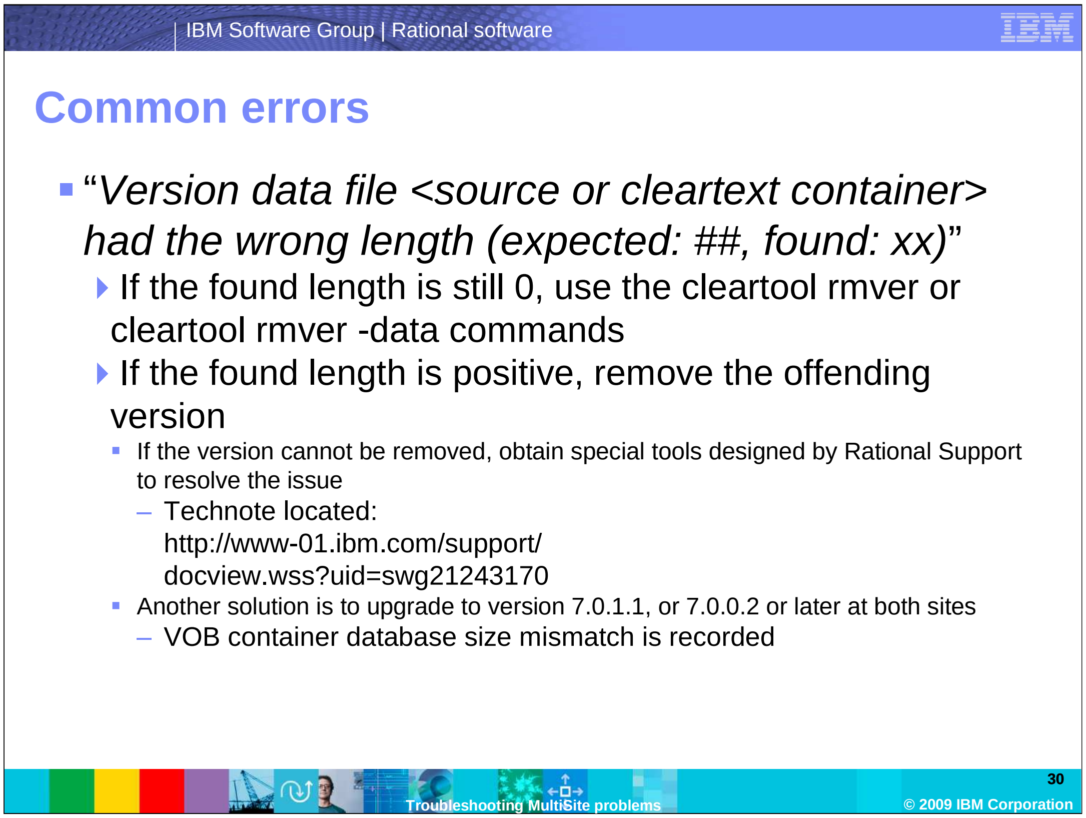
Common errors

- “*Version data file <source or cleartext container> had the wrong length (expected: ##, found: xx)*”
  - Occurs during a syncreplica command
    - ClearCase keeps track of the size of element versions in the VOB database. If this record does not match the actual size of the file recorded, *multitool syncreplica* fails.
    - Source container
      - A **found: 0** indicates the file or the data is missing in a source container.
      - If the found size is positive, resolve by removing the offending version.
    - Cleartext container
      - The error message might refer to a cleartext container with a found size of 0. Resolve this by removing the cleartext container and running the command again

“*Version data file <source or cleartext container> had the wrong length (expected: ##, found: xx)*” occurs during a syncreplica command. ClearCase keeps track of the size of element versions in the VOB database. If this record does not match the actual size of the file recorded by the file system, *multitool syncreplica* will fail.

If the error occurs on a source container and **found: 0** is recorded, this indicates that the file or the data is missing in a source container, resulting in the size of zero. To resolve this problem, replace the source container at the affect replica. If the found size is positive, removing the offending version might resolve the issue.

If the error occurs on a cleartext container and the found size is 0, resolve the issue by removing the cleartext container and running the command again.



IBM Software Group | Rational software

Common errors

- *"Version data file <source or cleartext container> had the wrong length (expected: ##, found: xx)"*
  - If the found length is still 0, use the cleartool rmver or cleartool rmver -data commands
  - If the found length is positive, remove the offending version
    - If the version cannot be removed, obtain special tools designed by Rational Support to resolve the issue
      - Technote located: http://www-01.ibm.com/support/docview.wss?uid=swg21243170
    - Another solution is to upgrade to version 7.0.1.1, or 7.0.0.2 or later at both sites
      - VOB container database size mismatch is recorded

Troubleshooting MultiSite problems

30

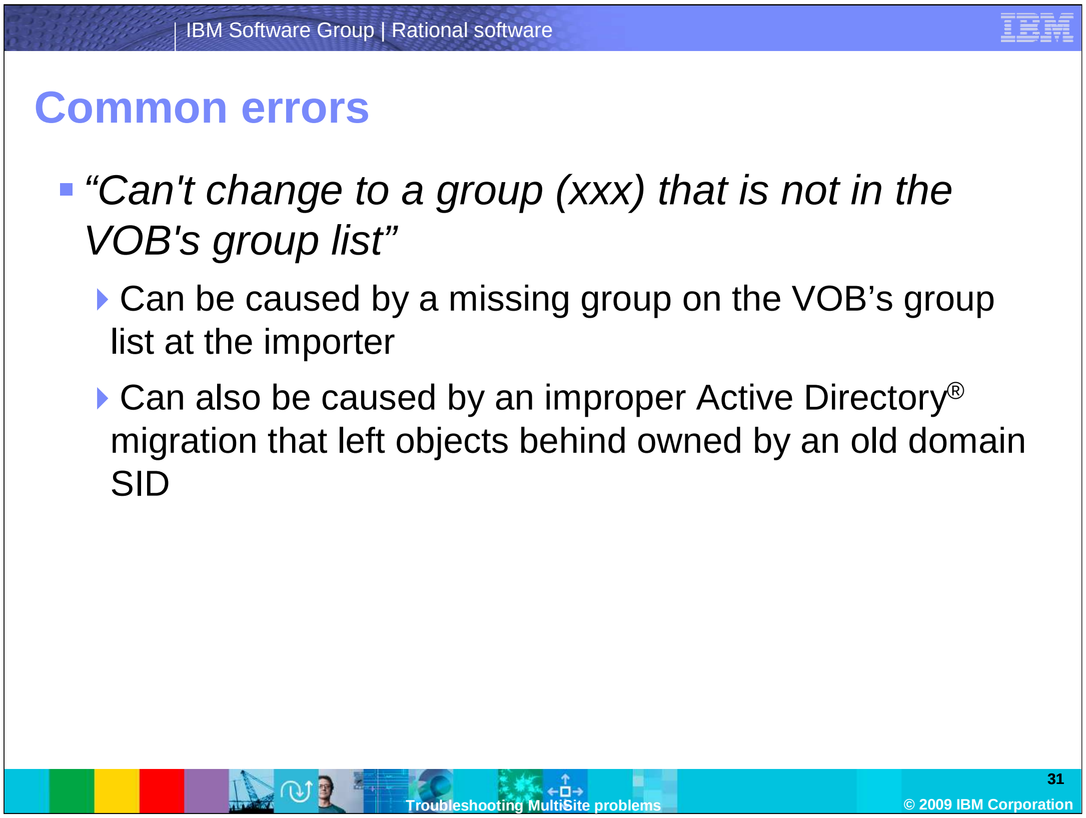
© 2009 IBM Corporation

If the found length is still 0 on cleartext errors, use the cleartool rmver or cleartool rmver -data command to remove the version.

If the found length is positive, remove the offending version.

If users cannot remove the offending version to resolve the issue, obtain special tools designed by Rational Support. There is a set of data required to obtain the tool indicated in the technote link listed on this slide.

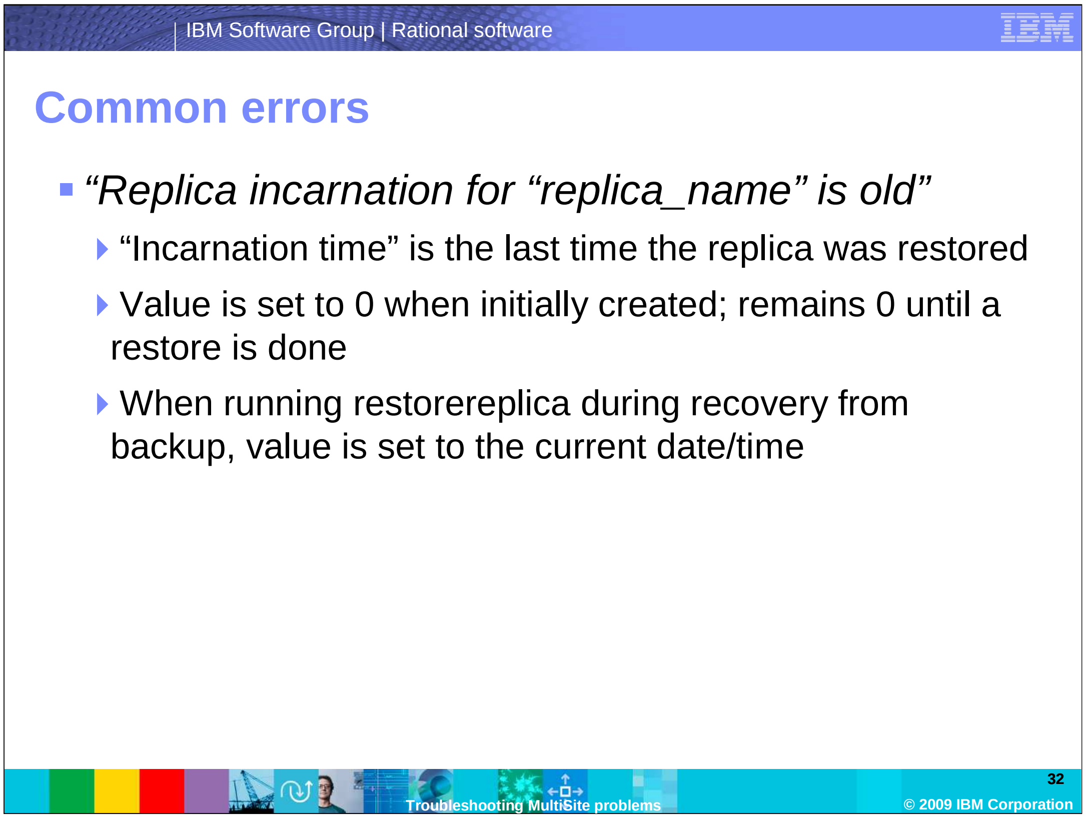
Another solution to resolve this issue is to upgrade to version 7.0.1.1, or 7.0.0.2 or later at both sites. This is because defect APAR IC48182 was open, and the fix allows the import or export to proceed even though a size mismatch is evident. The patch must be applied to both the sending and receiving sites, or the syncreplica command will still fail.



Common errors

- *“Can't change to a group (xxx) that is not in the VOB's group list”*
  - Can be caused by a missing group on the VOB’s group list at the importer
  - Can also be caused by an improper Active Directory® migration that left objects behind owned by an old domain SID

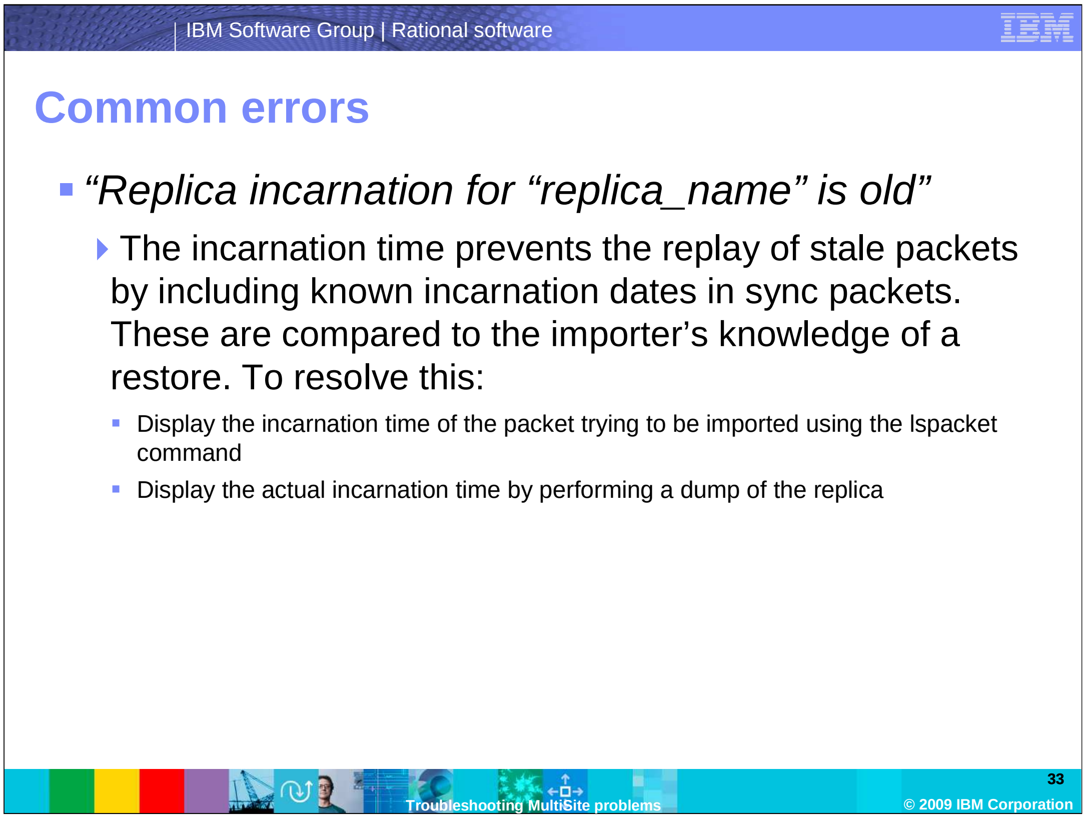
“Can't change to a group (xxx) that is not in the VOB's group list” is caused by a missing group on the VOB’s group list at the importing site. It can also be caused by an improper Active Directory migration that left behind objects owned by an old domain SID. To resolve this error, add the missing group to the VOB’s group list using the cleartool protectvob –add_group command.



Common errors

- *“Replica incarnation for “replica_name” is old”*
  - “Incarnation time” is the last time the replica was restored
  - Value is set to 0 when initially created; remains 0 until a restore is done
  - When running restorereplica during recovery from backup, value is set to the current date/time

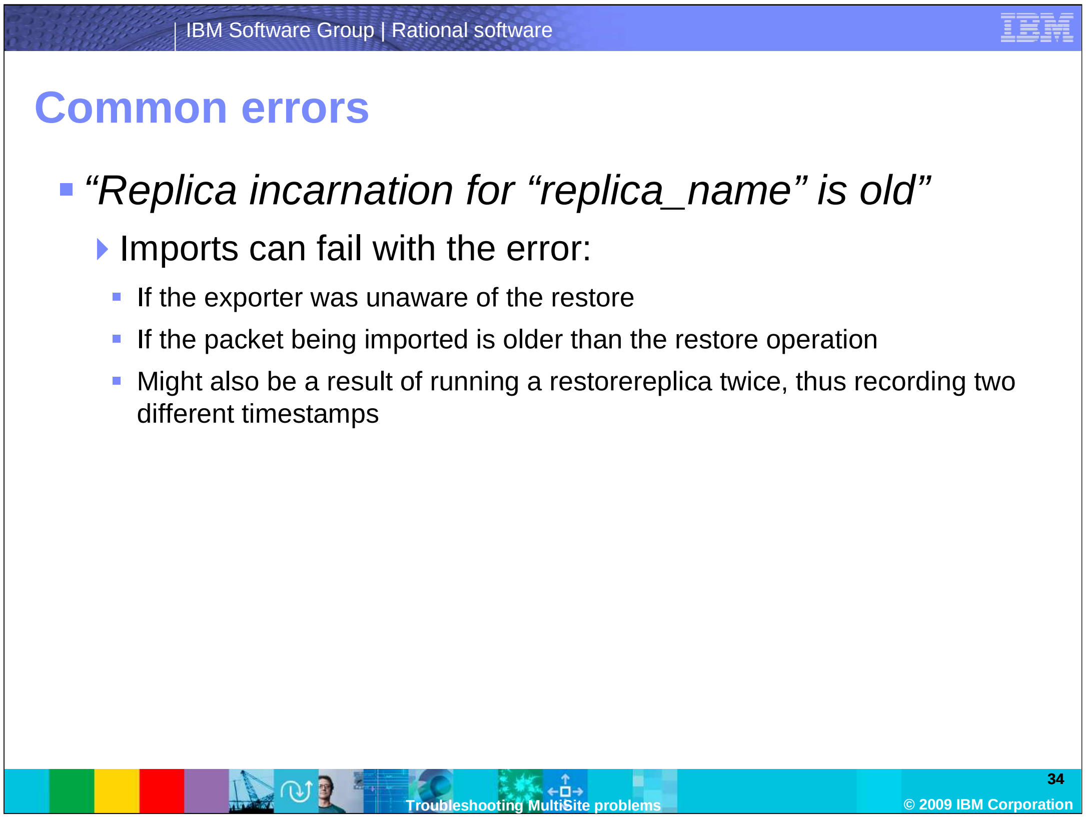
“Replica incarnation for “replica_name” is old” occurs when the “incarnation time” for a replica, the last time the replica was restored, is outdated. The value is set to 0 when a replica is initially created, and remains at 0 until a restore is performed. When running a restorereplica during a recovery from backup, the value is set to the current date/time.



Common errors

- *“Replica incarnation for “replica_name” is old”*
  - The incarnation time prevents the replay of stale packets by including known incarnation dates in sync packets. These are compared to the importer’s knowledge of a restore. To resolve this:
    - Display the incarnation time of the packet trying to be imported using the lspacket command
    - Display the actual incarnation time by performing a dump of the replica

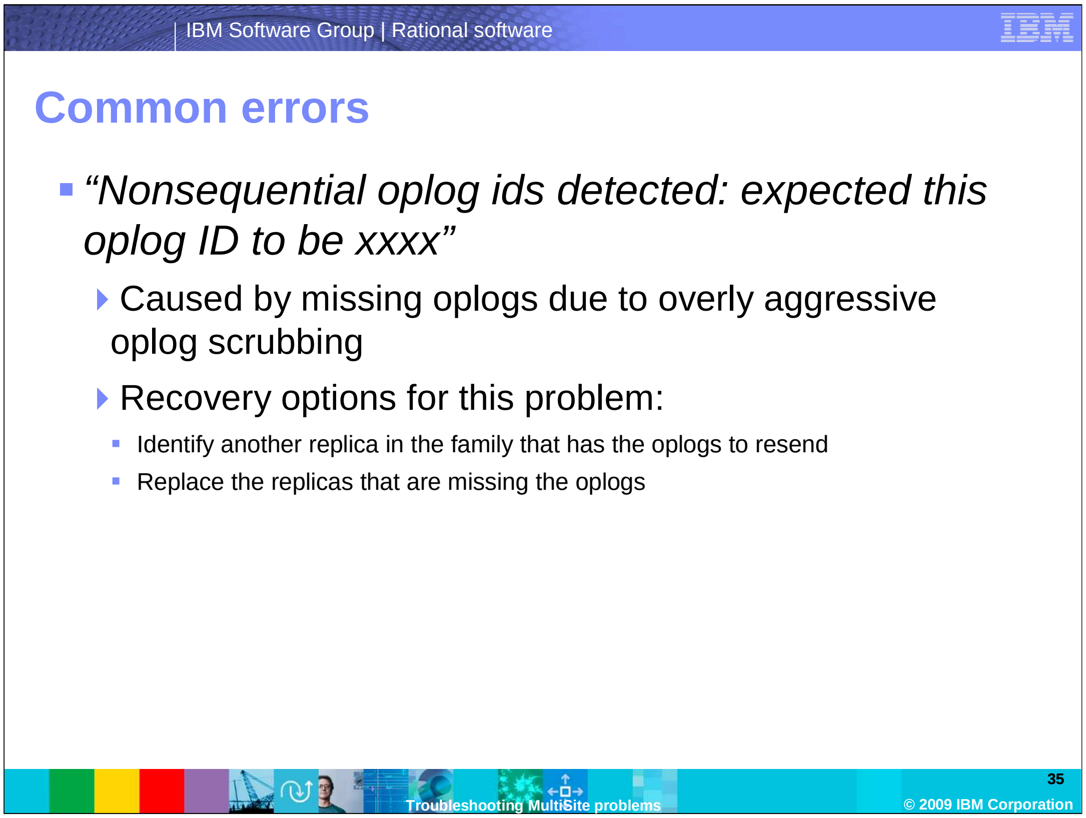
The incarnation time prevents the replay of stale packets by including the known incarnation dates in sync packets, which are compared to the importer’s knowledge of a restore. To resolve the issue, display the incarnation time of the packet trying to be imported using the lspacket command. Then, display the actual incarnation time by performing a dump of the replica. To dump the replica, use the cleartool dump replica:replica_name command.



Common errors

- *“Replica incarnation for “replica_name” is old”*
  - Imports can fail with the error:
    - If the exporter was unaware of the restore
    - If the packet being imported is older than the restore operation
    - Might also be a result of running a restorereplica twice, thus recording two different timestamps

Imports can fail with this error if the exporter was unaware of the restore, or if the packet being imported is older than the restore operation. Try to resend a new packet, as the current one might be stale. The error might also be a result of running a restorereplica twice, thus recording two different timestamps.

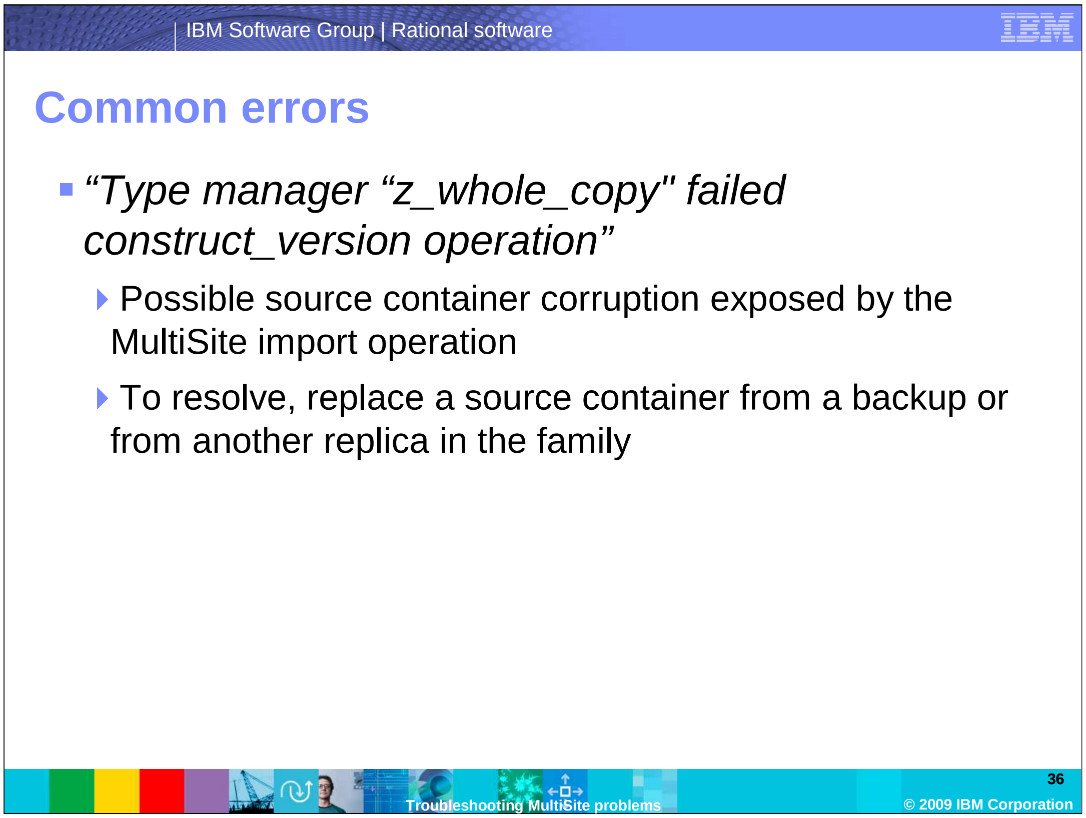


Common errors

- *“Nonsequential oplog ids detected: expected this oplog ID to be xxxx”*
  - Caused by missing oplogs due to overly aggressive oplog scrubbing
  - Recovery options for this problem:
    - Identify another replica in the family that has the oplogs to resend
    - Replace the replicas that are missing the oplogs

“Nonsequential oplog ids detected: expected this oplog ID to be xxxx” is the result of missing oplogs due to overly aggressive oplog scrubbing.

There are two recovery options for this problem. Either identify another replica in the family that has the missing oplogs to resend, or, if no other replicas in the family have the oplogs, replace the replicas missing the oplogs.



IBM Software Group | Rational software

Common errors

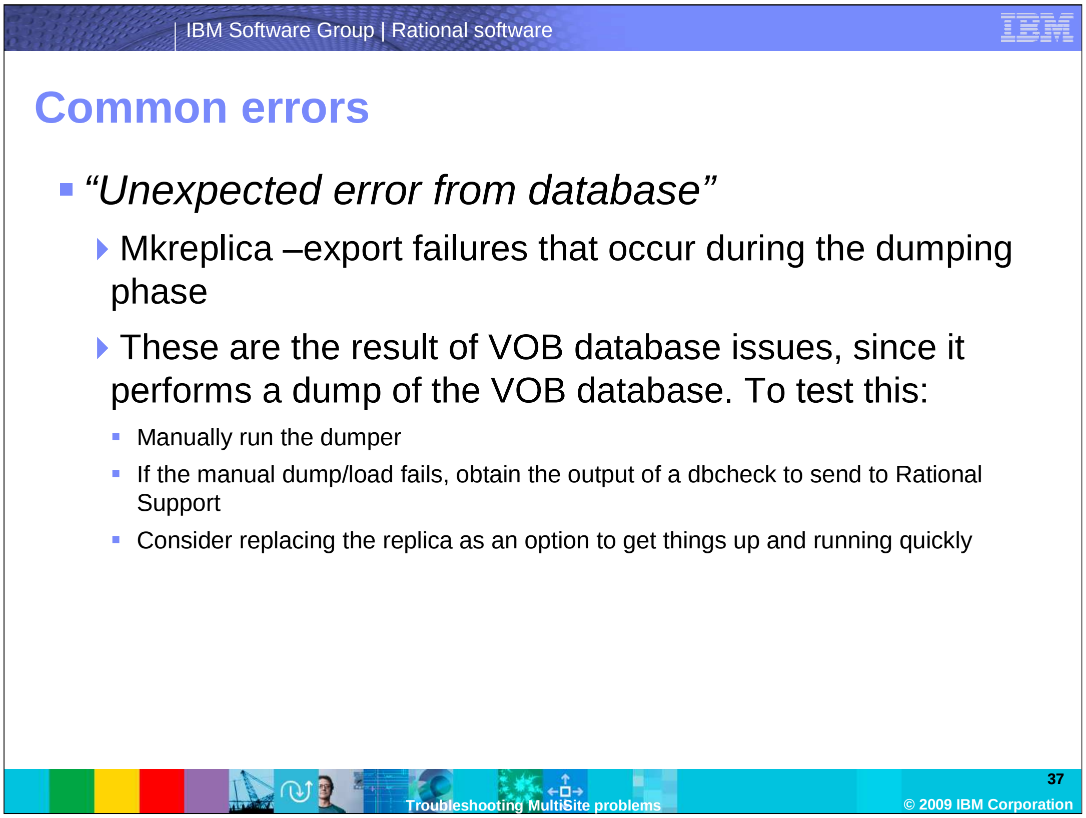
- *“Type manager “z_whole_copy" failed construct_version operation”*
  - Possible source container corruption exposed by the MultiSite import operation
  - To resolve, replace a source container from a backup or from another replica in the family

Troubleshooting MultiSite problems

36

© 2009 IBM Corporation

“Type manager “z_whole_copy" failed construct_version operation” is caused by a possible source container corruption exposed by the MultiSite import operation. Users might need to replace a source container from a backup or from another replica in the family to resolve this issue.

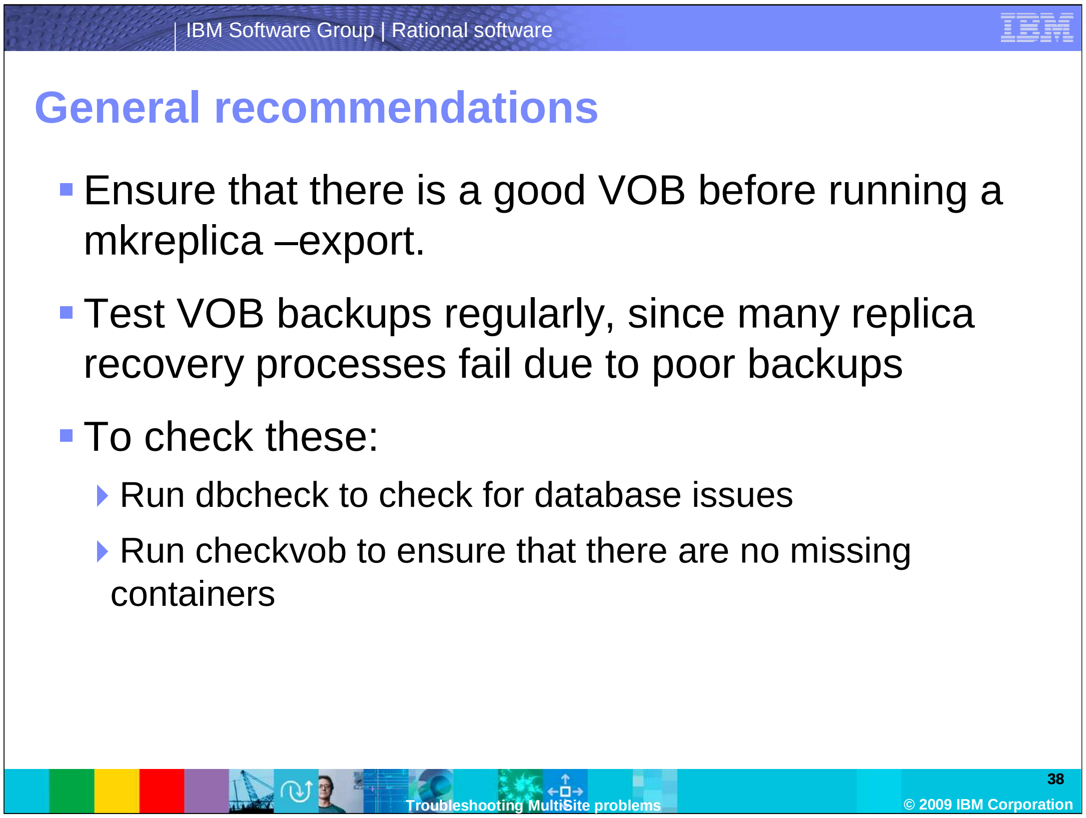


Common errors

- *“Unexpected error from database”*
  - Mkreplica –export failures that occur during the dumping phase
  - These are the result of VOB database issues, since it performs a dump of the VOB database. To test this:
    - Manually run the dumper
    - If the manual dump/load fails, obtain the output of a dbcheck to send to Rational Support
    - Consider replacing the replica as an option to get things up and running quickly

“Unexpected error from database” is caused by mkreplica –export failures that occur during the dumping phase. These are the result of VOB database issues, since it performs a dump of the VOB database.

Users can determine if the dumper is the problem by running the db_dumper command manually. If the manual dump or load fails, obtain the output of a dbcheck to send to Rational Support. Users might want to consider replacing the replica as an option to get things up and running quickly.



IBM Software Group | Rational software

General recommendations

- Ensure that there is a good VOB before running a mkreplica –export.
- Test VOB backups regularly, since many replica recovery processes fail due to poor backups
- To check these:
  - Run dbcheck to check for database issues
  - Run checkvob to ensure that there are no missing containers

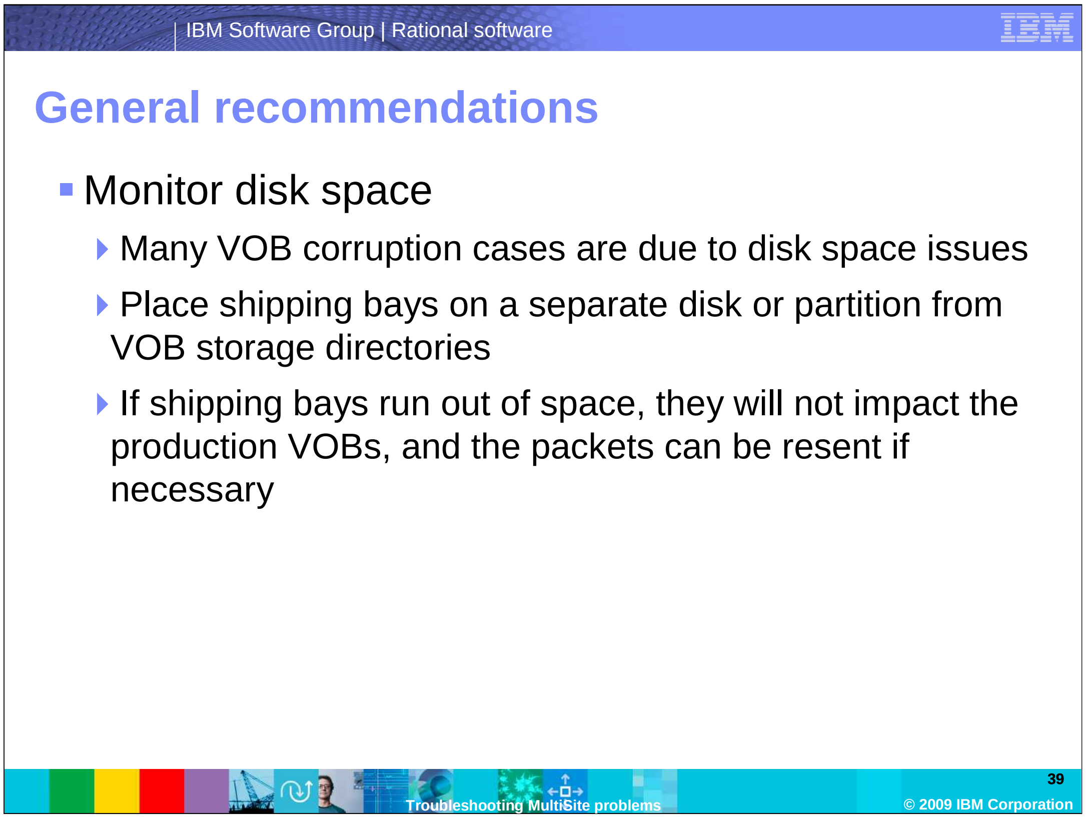
Troubleshooting MultiSite problems

38

© 2009 IBM Corporation

To ensure that replicas stay up and running and remain healthy, check to see that a VOB is good before running mkreplica –export. Also, many replica recovery processes fail because the backup was not good. If the back of a replicated VOB is bad, the only other option to restore the replica is to create a new one from a replicated site.

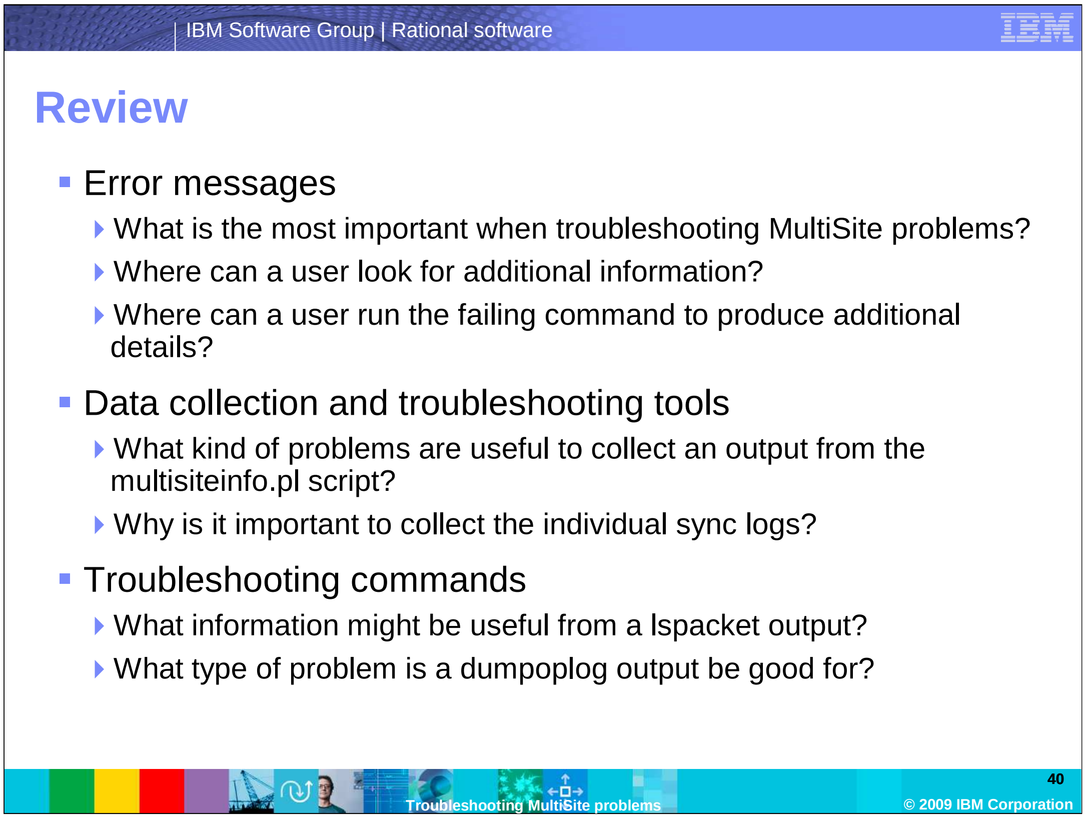
To prevent these issues, users should do two things for VOBs and backups. Run dbcheck to ensure the integrity of the VOB database. Also, run checkvob to ensure that there are no missing containers in the backup. Test these on a regular basis.



General recommendations

- Monitor disk space
  - Many VOB corruption cases are due to disk space issues
  - Place shipping bays on a separate disk or partition from VOB storage directories
  - If shipping bays run out of space, they will not impact the production VOBs, and the packets can be resent if necessary

It is also a good idea to monitor disk space on a regular basis. Many VOB corruption cases are due to disk space issues. Place shipping bays on a separate disk or partition them from the VOB storage directories. This ensures that the diskspace on the VOB server will not fill up. If the shipping bays run out of space, they will not impact the production VOBs, and the packets can be resent, if necessary. To prevent shipping bays from running out of space, monitor them regularly. Also, check to see if packets are accumulating in the incoming/outgoing bays. If they are, remove any old packets, and consider setting up a new shipping class.



IBM Software Group | Rational software

Review

- Error messages
  - What is the most important when troubleshooting MultiSite problems?
  - Where can a user look for additional information?
  - Where can a user run the failing command to produce additional details?
- Data collection and troubleshooting tools
  - What kind of problems are useful to collect an output from the multisiteinfo.pl script?
  - Why is it important to collect the individual sync logs?
- Troubleshooting commands
  - What information might be useful from a lspacket output?
  - What type of problem is a dumpoplog output be good for?

Troubleshooting MultiSite problems

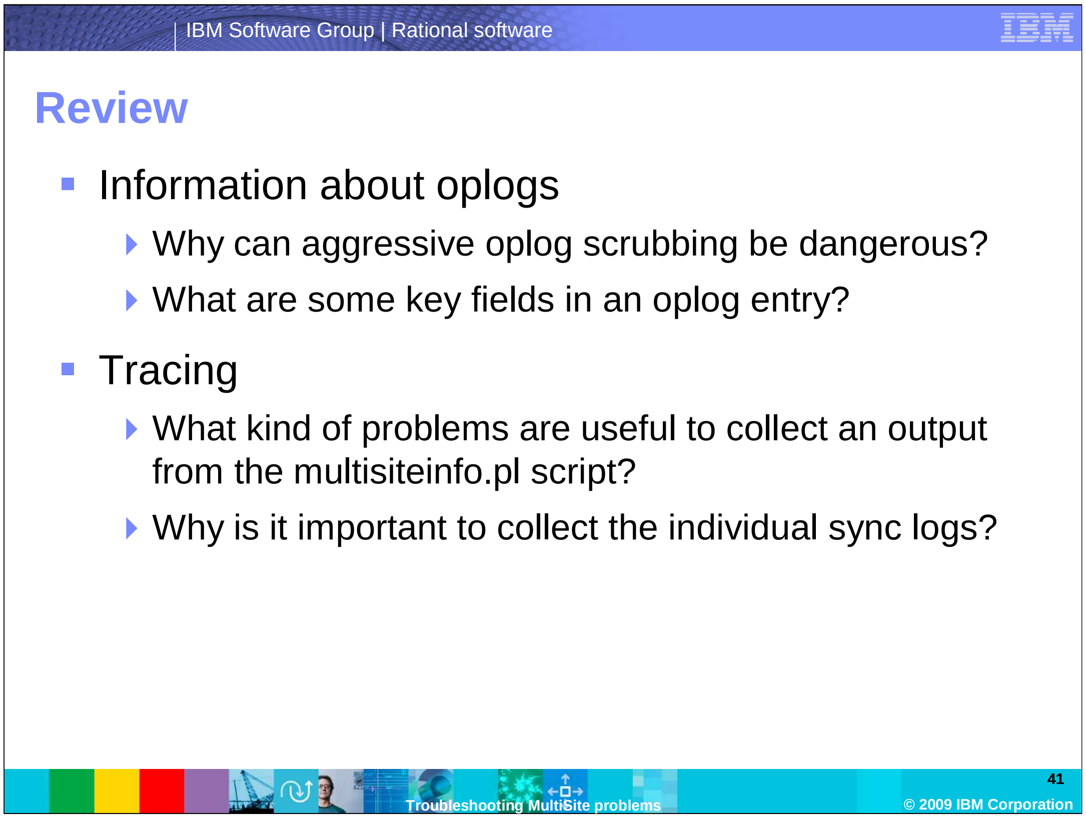
40

This module covered troubleshooting for MultiSite problems.

The error messages section discussed what is important to note and collect about the messages that occur during MultiSite operations. This includes determining what is most important when troubleshooting MultiSite problems, where to look for additional information, and where to run the failing command to produce additional details.

The Data Collection and Troubleshooting Tools section discussed the important information to collect when a problem arises. This includes the types of problems useful to collect an output from the multisiteinfo.pl script, and why it is important to collect the individual sync logs.

The Troubleshooting Commands section discussed the different commands that assist in the troubleshooting process for MultiSite issues. This includes the information that might be useful from a lspacket output, and the type of problem that a dumpoplog output is good for.

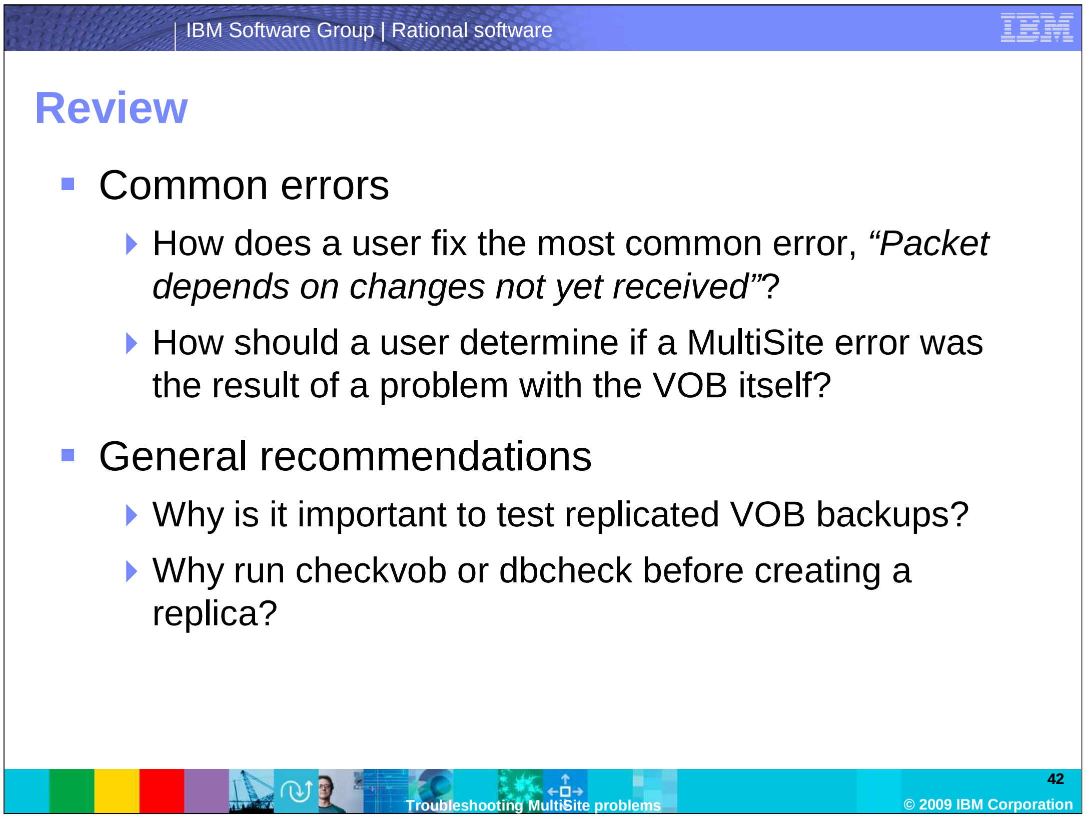


Review

- Information about oplogs
  - Why can aggressive oplog scrubbing be dangerous?
  - What are some key fields in an oplog entry?
- Tracing
  - What kind of problems are useful to collect an output from the multisiteinfo.pl script?
  - Why is it important to collect the individual sync logs?

The Information about oplogs sections, oplog scrubbing, and the different parts of an oplog were discussed. This includes why aggressive oplog scrubbing can be dangerous, and what some key fields in an oplog entry are.

The Tracing values were introduced to provide more detailed information for more complex issues. This includes the kinds of problems that are useful to collect an output for from the multisiteinfo.pl script, and why it is important to collect the individual sync logs.

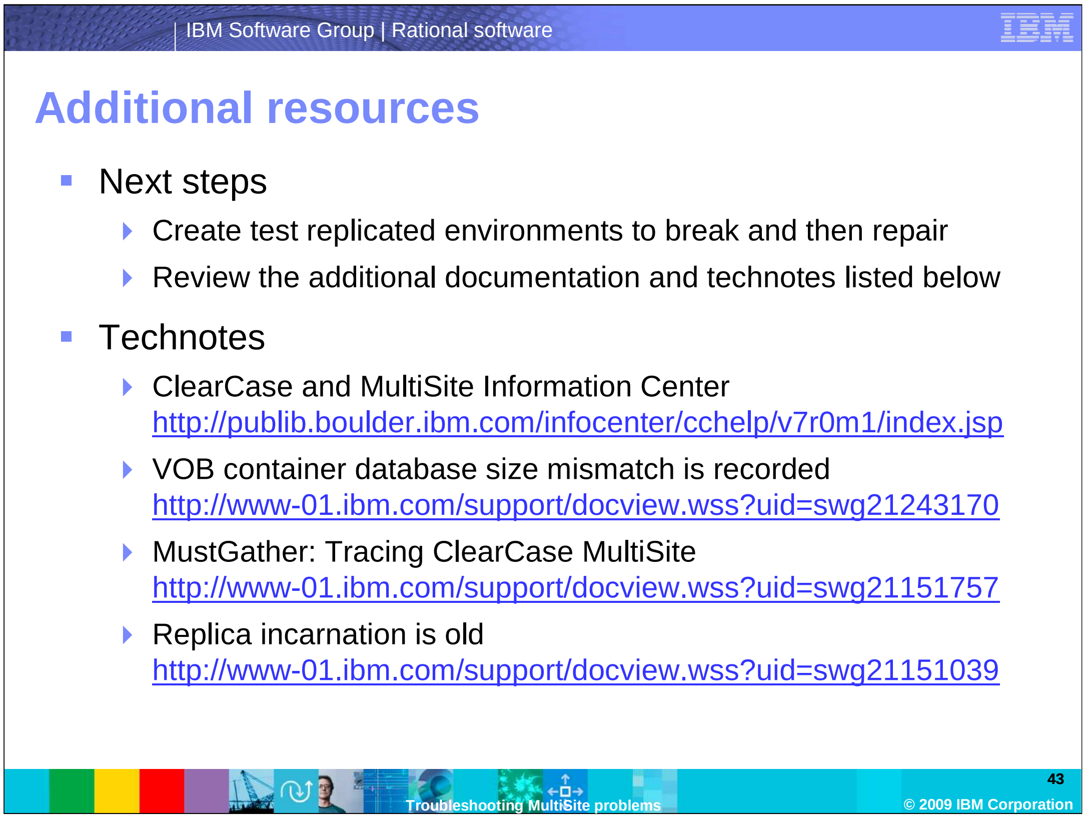


Review

- Common errors
  - How does a user fix the most common error, *“Packet depends on changes not yet received”*?
  - How should a user determine if a MultiSite error was the result of a problem with the VOB itself?
- General recommendations
  - Why is it important to test replicated VOB backups?
  - Why run checkvob or dbcheck before creating a replica?

In the Common Errors section, multiple errors were introduced that are commonly reported in a replicated environment. This section includes how a user fixes the most common error, *“Packet depends on changes not yet received,”* and how to determine if a MultiSite error was the result of a problem with the VOB itself.

The General Recommendations section ensures that users have good, healthy VOBs before replicating, and how to keep them healthy and running once they are replicated. This section also includes why it is important to test replicated VOB backups, and why users should run checkvob or dbcheck before creating a replica.

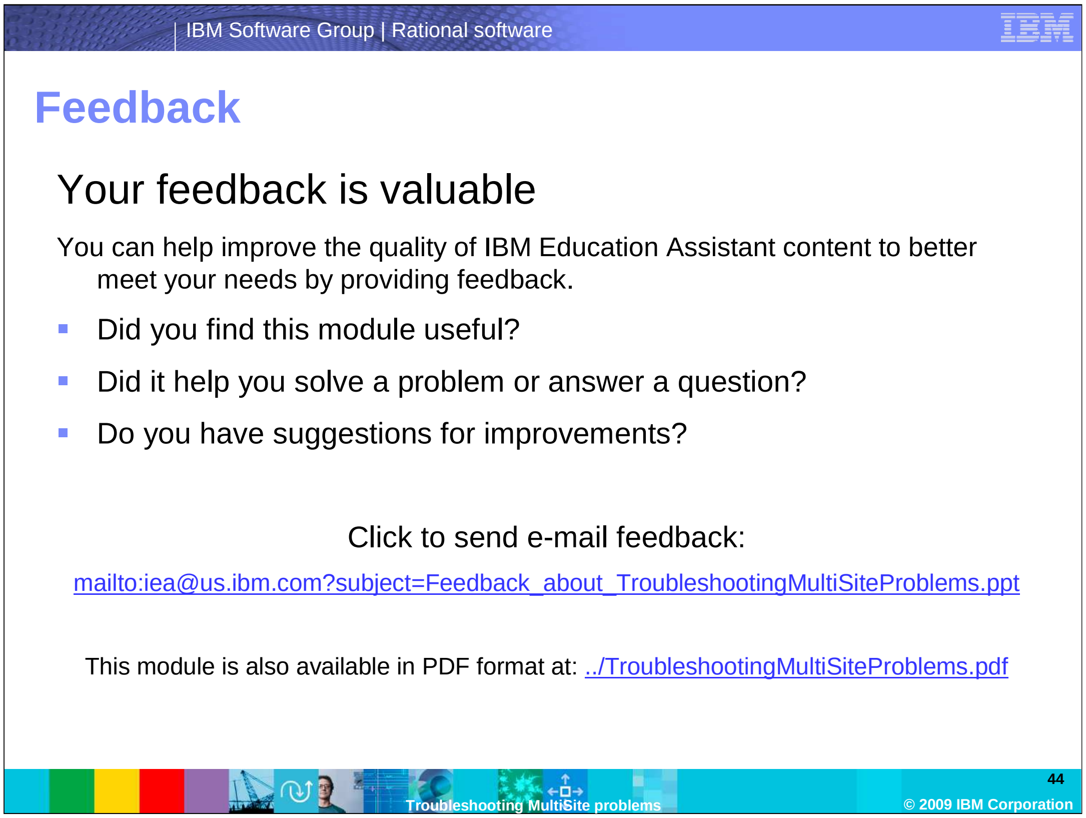


Additional resources

- Next steps
  - Create test replicated environments to break and then repair
  - Review the additional documentation and technotes listed below
- Technotes
  - ClearCase and MultiSite Information Center
    http://publib.boulder.ibm.com/infocenter/cchelp/v7r0m1/index.jsp
  - VOB container database size mismatch is recorded
    http://www-01.ibm.com/support/docview.wss?uid=swg21243170
  - MustGather: Tracing ClearCase MultiSite
    http://www-01.ibm.com/support/docview.wss?uid=swg21151757
  - Replica incarnation is old
    http://www-01.ibm.com/support/docview.wss?uid=swg21151039

After completing this module, create test replicated environments to break and then repair. A hands-on experience better helps to understand what to do in a situation when a problem with a replicated environment occurs. It also helps to understand how to use the troubleshooting commands, and what the outputs look like.

Finally, it can be very beneficial to review the additional documentation and technotes listed on this slide. These technotes go into greater details on some of the errors and information discussed in this module.



Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send e-mail feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_TroubleshootingMultiSiteProblems.ppt

This module is also available in PDF format at: ../TroubleshootingMultiSiteProblems.pdf

You can help improve the quality of IBM Education Assistant content by providing feedback.

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:

ClearCase   Rational   ClearCase MultiSite

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.

Active Directory, Windows, and the Windows logo are registered trademarks of Microsoft Corporation 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.

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.

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.