Progress®

OpenEdge Replication

What Have We Done for You Lately?

Rob Farver

rob.farver@progress.com

PUG Challenge Boston, 2024

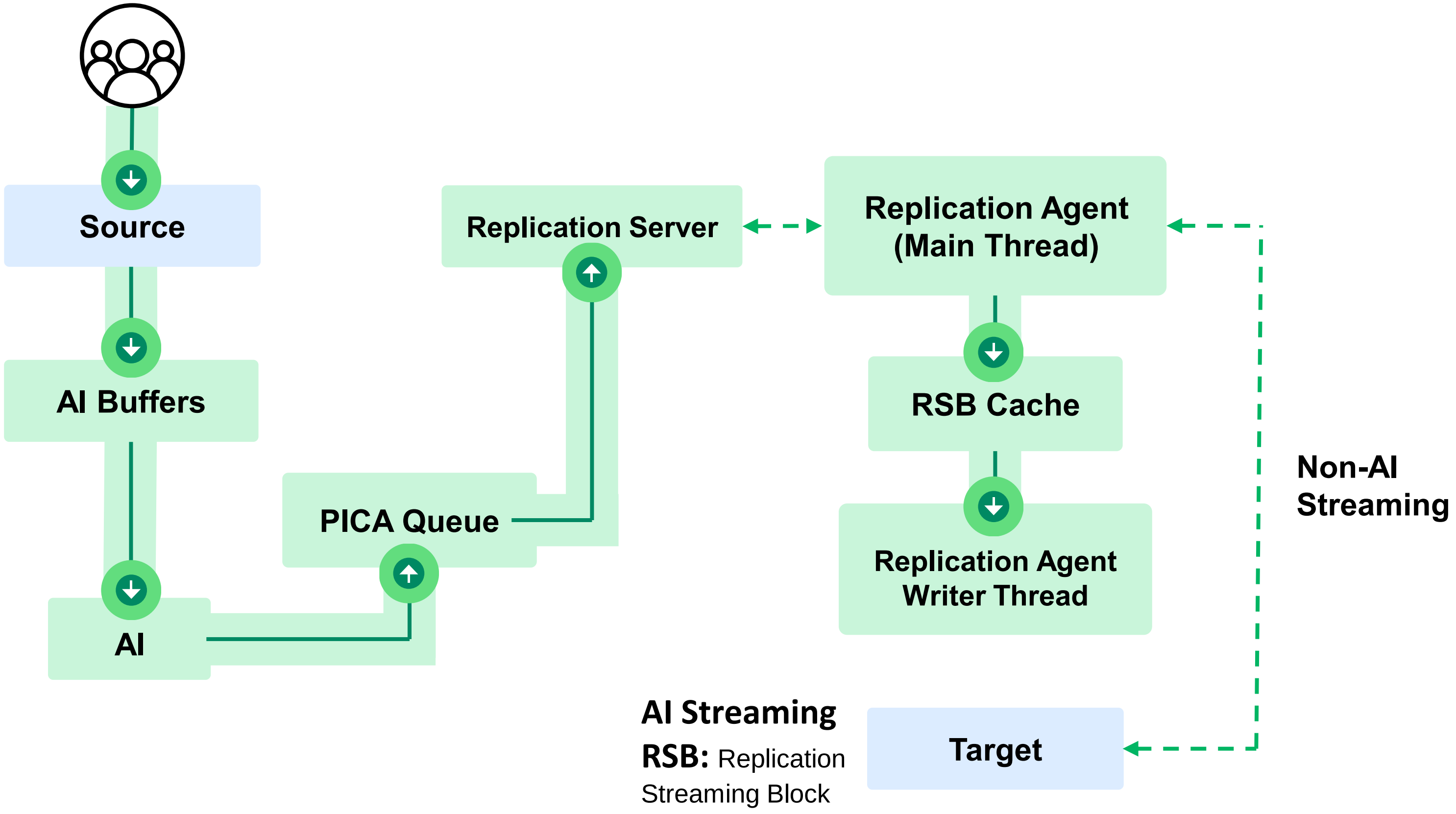
- New OpenEdge Replication features in OpenEdge v12
- Backup & restore enhancements in OpenEdge v12
- Replication sets review
- Replication sets demo
- How can we help?

What’s New in OpenEdge Replication in OpenEdge v12?

AI Streaming (12.2)

- Queues AI blocks on target in RSB cache
- 2 thread target agent
- Reduces source queueing, improves recovery

AI Streaming Replication Overview



RSB: Replication Streaming Block

Modifiable VSTs on Target (12.2)

- _Connect._Connect-Misc
- _MyConnect._MyConn-UserMisc
- _Database-Feature._DBFeature_Enabled
  - Large database entry keys, enable only
- _DbParams._DbParams-Value
  - Numerous startup parameters can be modified

Enhanced Repl Status VSTs (12.2)

- New fields added to the following VSTs
  - _Repl-Server
  - _Repl-Agent

New Transition Properties (12.2)

- dr-transition-to-agents
  - List of agents for recovery transition
  - If not defined, transition-to-agents is used for recovery
- transition-to-agents
  - List of agents for failover transition

Repl Properties Validation (12.2)

- dsrutil *db-name* -C validate
  - When run on the replication source, checks the syntactical correctness of the [server] , [control-agent] and [transition] properties and reports on the remote target broker running the `status.Replication` target.
  - When run on the replication target, checks the [agent] and [transition] properties.

Database Parameters (12.2)

- -pica
  - Now modifiable online
- -picanap
  - Max PICA queue naptime in milliseconds
  - Increase to reduce CPU usage
  - Default 1ms, max 100ms

Source and Target Parameters No Longer Need to Match (12.2)

- The transaction table size is no longer scoped or based on the parameters the database was started with. Instead, the transaction table is simply scoped to its maximum size.

- This alleviates the overhead of having to additionally scale the transaction table on the source database, should related parameters be increased online.

DB Upgrade without Re-baseline (12.8)

- You can convert without rebase when the only database changes from OpenEdge Release 11 are in the schema area
  - conv1112 replsource sch.bak
  - conv1112 repltarget sch.bak
- conv1112 pre-scan to verify can be done
- Blockers
  - TDE w/ RC4 cipher
- Sequence conversion 32bit – 64bit

Automatic Database Reconnection (12.8)

- New client startup parameter -autoReconnect
  - Clients can auto connect to primary target database if source database fails
  - Can be used with or without -dbalt1 and -dbalt2
  - Best practice to configure -dbalt1 to primary target and -dbalt2 to secondary target (if one exists)

Brokers Run Through Recovery Transition (12.8)

- To finish transitions faster, database brokers now continue running on single targets and replication sets
  - This change improves both manual and automatic recovery transitions
  - Can be turned off

Progress

Enforce Replication Sync on Shutdown (12.8)

- New proshut parameter -replSync
  - Syncs source/target
  - Logs to database logfile
  - Can be used with –shutdownTimeout (default 10 minutes)

Truncate BI on Target (12.8)

- Can now truncate BI on target databases
  - Utilizes –replSync
- Allows for BI block size or cluster size changes

Where to Get Details on v12 Changes

- Progress Documentation
  - Learn about OpenEdge 12
    - Review changes for each release of v12
  - Use Database Replication (v12.8)
  - Use Database Replication (v12.2)

Backup and Restore Enhancements in OpenEdge v12

New probkup Parameters (12.8)

- -thread <n>
  - 1 enables, 0 disables multi-threaded backup
- -threadnum <n>
  - Default is lower of number of CPUs or 6
  - Max is number of CPUs
- -wbf (write buffer factor for multi-threading)
  - Number of write buffers, default 1024
- -comlevel <n> (must use –com also)
  - If -com not specified, no compression is done
  - Default level 3, prior versions level 1 (RLE)

Backup Compression (12.8)

-com	-comlevel	Compression algorithm applied
Specified	Not Specified	Applies the ZSTD algorithm using the default compression level 3. Changes the backup header version.
Specified	1	Applies the RLE compression algorithm. Free blocks are compressed to the length of their header, 16 bytes.
Specified	2	Applies the fastest, least compressed ratio of the ZSTD compression algorithm and changes the backup header version.
Specified	3	Applies a ZSTD algorithm that balances speed versus compression ratio and changes the backup header version.
Specified	4	Applies the slowest, most compressed ratio of the ZSTD compression algorithm and changes the backup header version.

New prorest Parameters (12.8)

- -thread <n>
  - 1 enables, 0 disables multi-threaded restore
  - If omitted, disabled by default
- -threadnum <n>
  - If n omitted, max is number of CPUs
  - Max allowable n is twice the number of CPUs
  - If n is higher than max, reduced to max

New prorest Parameters (12.8)

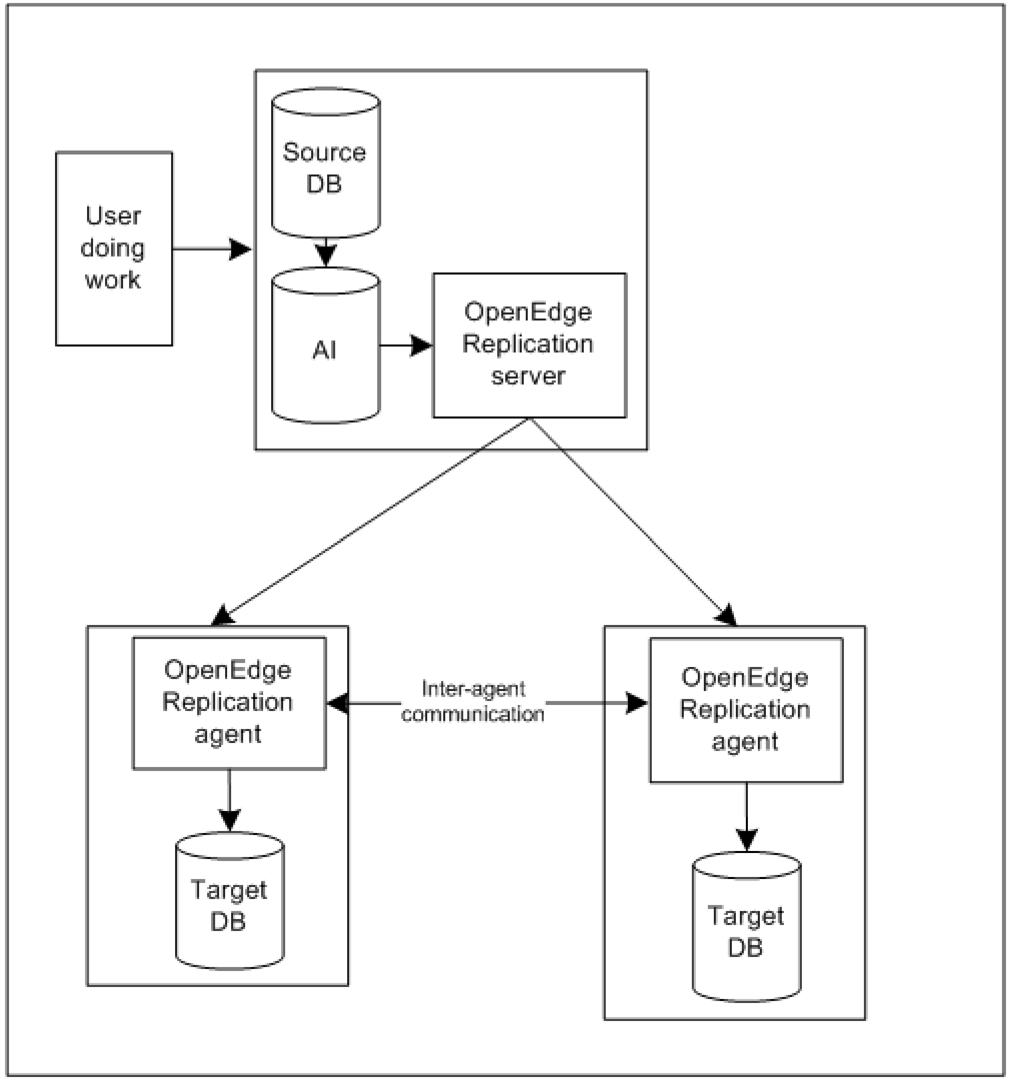
- -wbf <n>
  - Each write buffer is 64KB
  - Default 1024, max 100,000
  - -wbf must be >= -threadnum
  - If -threadnum not specified, adjust -wbf to number of CPUs
  - Reduce -wbf to reduce memory usage
- -noverify
  - Skips CRC checks of backup block
  - Speeds up restore in time-critical situations
  - Should run prorest -vp in parallel to verify backup

Replication Sets

Replication Set

A replication environment that contains a source replica and two target replicas that can transition together.

After imaging is required on both targets in a replication set.


Source DB
User doing work
AI
OpenEdge Replication server
OpenEdge Replication agent
Inter-agent communication
OpenEdge Replication agent
Target DB
Target DB

Inter-agent Communication

- Allows the targets to transition together if the source database is gone.
  - One becomes new source, the other remains a target and replication can continue.
- Targets communicate their location relative to source.
  - Used to determine if AI extents can be unlocked.
  - If one target is behind, the healthier target may need to keep a number of extents locked to synchronize with the less healthy target.
- Allows a health check at the beginning of a recovery transition to determine coordinator.

Inter-agent Communication

- Specify replication-set=1 in the [transition] section of the properties file.

```
[transition]
 replication-set=1
 database-role=reverse
 transition-to-agents=agent1,agent2
 restart-after-transition=1
 source-startup-arguments=-S 51902 -pi replserv
```

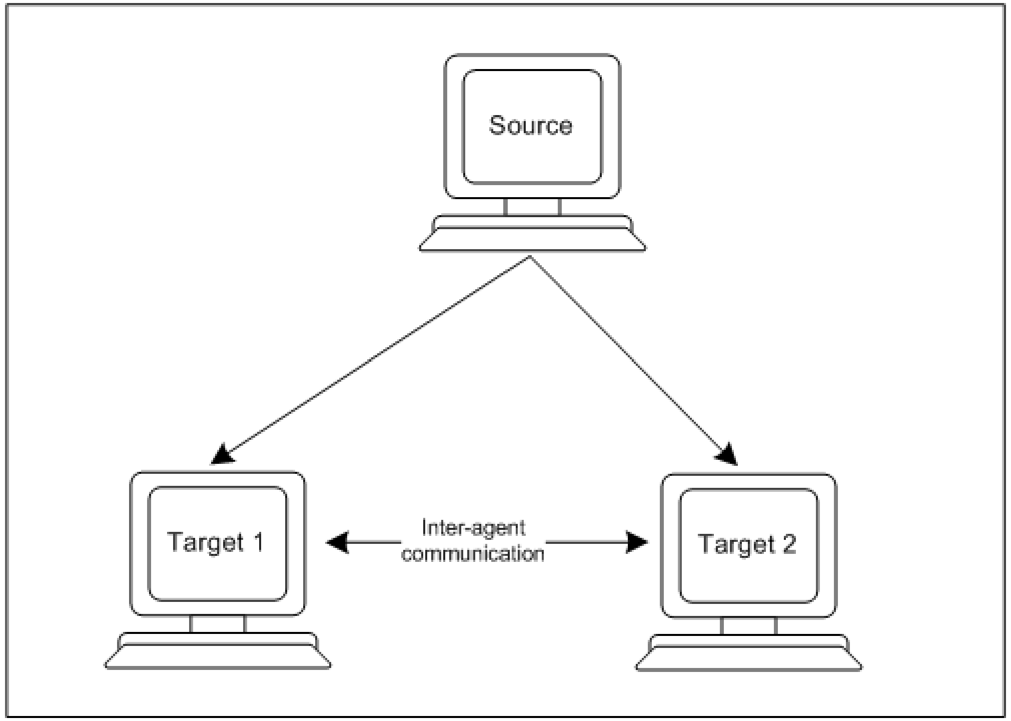
- When agents receive the inter-agent communication message from the server, they will initiate their connections and send a response to the server.

What if…

- Source server goes offline?
  - Targets remain connected if they remain running.
  - When server is restarted, it will re-initiate the network.
- Target agent goes offline?
  - Replication between source and other agent continues.
  - When the agent is restarted and server told to connect to it, the network will be re-initiated.
- Agents are started without server?
  - Agents will depend on their properties files to connect to each other.

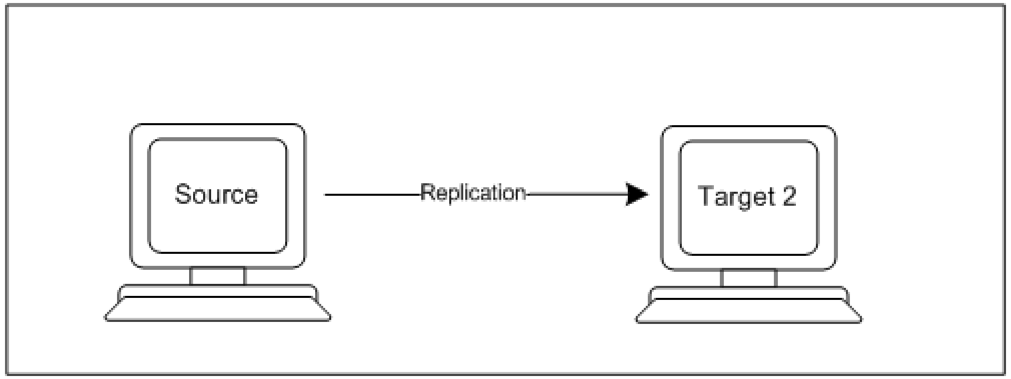
Recovery Transition

The transition of a database from the role of a target replica to a source replica after the failure of the original source replica.



Recovery Transition

The transition of a database from the role of a target replica to a source replica after the failure of the original source replica.



Configuring Recovery Transition (2 ways)

- Manual Transition
  - Transition manually with **dsrutil -C transition** command
  - Source is lost, but AI data still available
  - Apply AI extents before transition
  - Recover as much data as possible
  - Targets can be restarted after failures, then begin transition
- [Server] properties:
  - **`transition=manual`**

Configuring Recovery Transition (2 ways)

- Auto Transition
  - Transition happens automatically after “transition-timeout”
  - Transition as soon as possible (HA)
- [Server] properties:
  - **`transition=auto`**
- Not typically used

Which Agent will Transition?

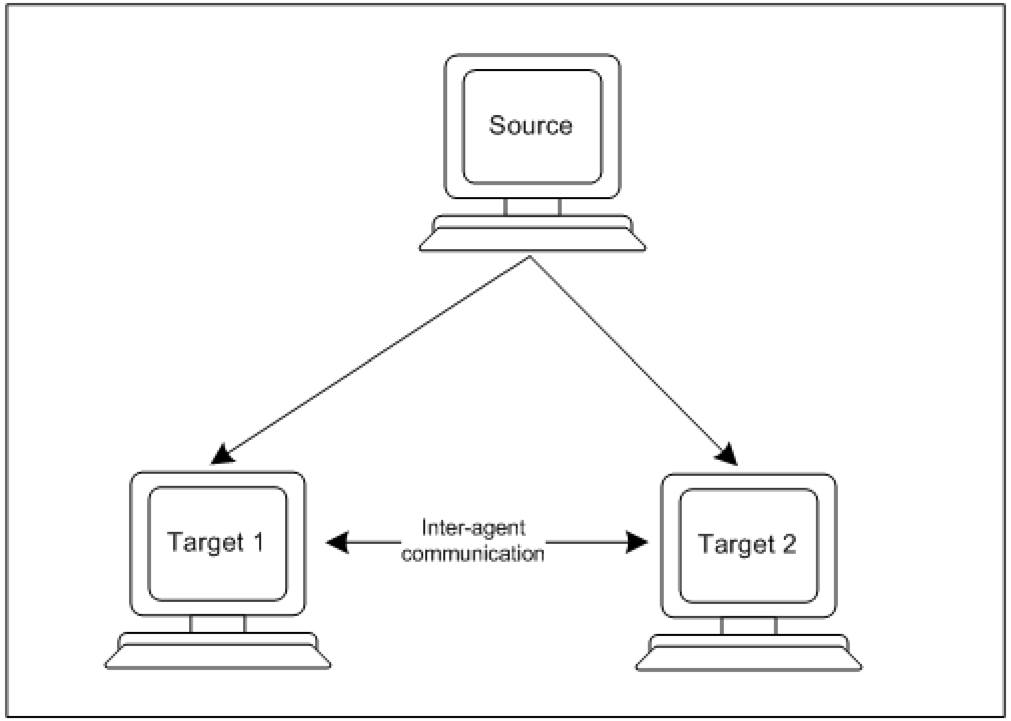
- Priority of the agents to transition is identified in the properties file.
  - The **transition-to-agents** property specifies the priority:

```
[transition]
  replication-set=1
  database-role=reverse
  transition-to-agents=agent1,agent2
  restart-after-transition=1
  source-startup-arguments=-S 51876 -pi replserv
```

  - Can also use the new **dr-transition-to-agents** to specify a different priority for recovery vs. failover transition.

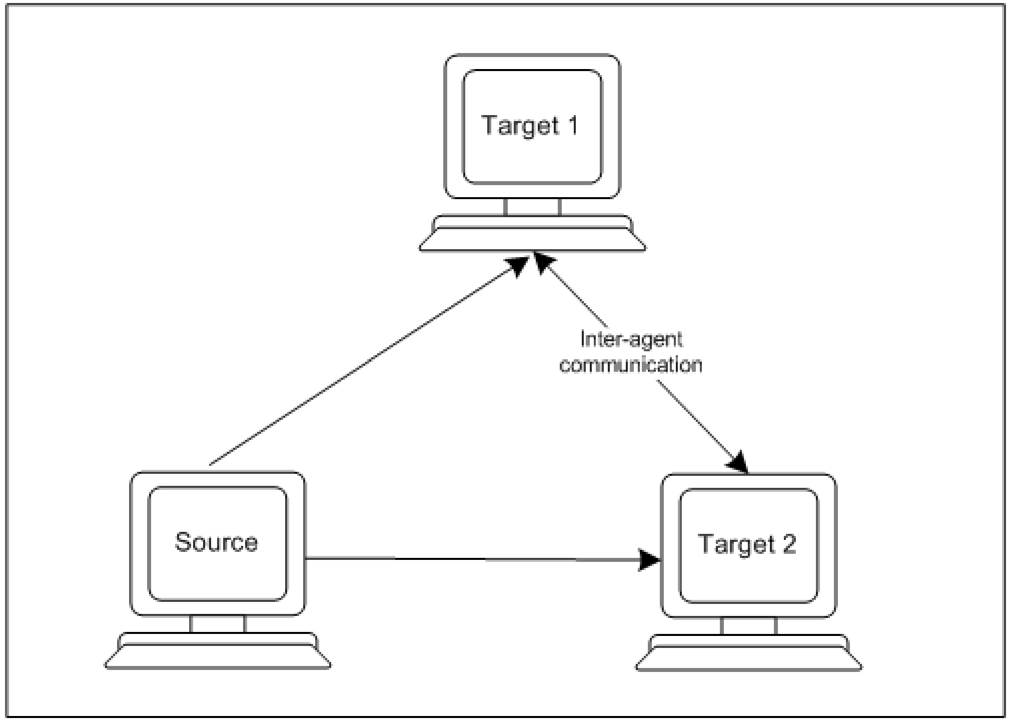
Failover Transition

A transition where a source replica and a target replica switch roles.



Failover Transition

A transition where a source replica and a target replica switch roles.



Initiating Failover

```
dsrutil <source_db_name> -C transition failover
```

How is the target database specified?

```
transition-to-agents=agent1,agent2
```

When is Transition Failover Useful?

- Maintenance is required for the server hosting the source replica
- The server hosting the source replica is being replaced
- The server hosting the source replica is being moved

Best Practices for Failover Transition

- All replica properties files must have these properties set:

```
transition=manual
role=reverse
agent-shutdown-action=recovery
replication-set=1
transition-to-agents=agent1,agent2
```

Best Practices for Failover Transition

- All replica properties files should define a `[control-agents.xxx]` section(s)

- Define the same `transition-to-agents` property in each replica:
  - `transition-to-agents=agent1,agent2`

- All replica properties files should define an `[agent]` section

Best Practices for Failback

- Edit the current source replica’s properties file, identifying new target replica:
  - `transition-to-agents=agent0`
- Terminate and restart the replication server running on replica 1
- Trigger role switch for the replicas:
  - `dsrutil targetdb1 -C transition failover`

Best Practices for Failback

- After transition completes the original configuration is restored:
  - The newest target replica will have reverted back to its original role as a source
  - The new source replica will have reverted back to its original role as a target
- Edit previous source replica’s properties file, restoring the original values:
  - `control-agents=agent2`
  - `transition-to-agents=agent1,agent2`
- Terminate and restart agent1 on replica 1

Replication Set Demo

Demo

- Initial Setup
- Recovery Transition
- Failover Transition
- Failback

Initial Setup

- Setup Script
  - Create add_ai script to add 10 ai files to all databases (source and targets)
  - Shutdown databases if already served
  - Delete and recreate source and target directories and aiarch directory
  - Update repl properties with aiarch directory
  - Copy source and target repl properties to respective directories

Initial Setup

- Create source database, add ai files, enable ai/archiver, enable replication, take repl backup
- Restore backup on targets, add ai files, enable ai/archiver, enable target repl
- Start targets, then source
- Check status

Recovery Failover

- Source lost – transition target to source
  - Verify targets in pre-transition, if not, force manually
  - Perform transition on first target
  - Start target database
  - Start source database
  - Confirm sync
- Reset demo

Planned Failover

- Planned transition to target 1 for maintenance
  - Add source to target 1 **control-agents** property
  - Restart / Reconnect target 1 agent
  - Perform transition failover
  - Start targets (old source and original target 2)
  - Start source (original target 1)
  - Confirm sync
- Reset demo

Planned Failback

- Maintenance complete, fail back to original source
  - Modify target 1 **transition-to-agents** to original source
  - Restart target 1 replserver
  - Perform transition on target 1
  - Restore target 1 **control-agents** and **transition-to-agents** to original recovery values
  - Start targets
  - Start source
  - Confirm sync

How Can We Help?

Progress Professional Services

- We can help you implement OpenEdge Replication
  - Ad hoc Professional Services engagement

- We can manage your OpenEdge Replication
  - Managed OpenEdge Replication Service

Questions?

Progress® OpenEdge®

News You Can Use

Progress®