



Pro2[®] Web User Interface

Table of Contents

| | |
|------------------------------------|---------------|
| Copyright..... | 5 |
| Preface..... | 7 |
| Pro2 web interface..... | 9 |
| Dashboard..... | 10 |
| Watch boxes..... | 10 |
| Thread control..... | 11 |
| Graphs..... | 12 |
| Dashboard settings..... | 12 |
| Manage replication..... | 12 |
| Select source..... | 13 |
| Get Source side schema..... | 14 |
| Select target..... | 14 |
| Generate target schema..... | 16 |
| Mapping..... | 17 |
| CDC mapping..... | 18 |
| Advanced configuration..... | 19 |
| Generate code..... | 20 |
| Properties..... | 21 |
| Actions..... | 31 |
| Bulk Load..... | 32 |
| Source DB Utilities..... | 33 |
| Target DB Utilities..... | 34 |
| How to schedule jobs in Pro2..... | 36 |
| Tools..... | 39 |
| WAN..... | 43 |
| CDC..... | 45 |
| Reports/Troubleshooting..... | 46 |
| Settings..... | 49 |
| Pro2 Enterprise View..... | 49 |
| LAN/WAN..... | 50 |
| Processor Stop..... | 50 |

Copyright

© 2022 Progress Software Corporation and/or its subsidiaries or affiliates. All rights reserved.

These materials and all Progress® software products are copyrighted and all rights are reserved by Progress Software Corporation. The information in these materials is subject to change without notice, and Progress Software Corporation assumes no responsibility for any errors that may appear therein. The references in these materials to specific platforms supported are subject to change.

#1 Load Balancer in Price/Performance, 360 Central, 360 Vision, Chef, Chef (and design), Chef Habitat, Chef Infra, Code Can (and design), Compliance at Velocity, Corticon, Corticon.js, DataDirect (and design), DataDirect Cloud, DataDirect Connect, DataDirect Connect64, DataDirect XML Converters, DataDirect XQuery, DataRPM, Defrag This, Deliver More Than Expected, DevReach (and design), Driving Network Visibility, Flowmon, Inspec, Ipswitch, iMacros, K (stylized), Kemp, Kemp (and design), Kendo UI, Kinvey, LoadMaster, MessageWay, MOVEit, NativeChat, OpenEdge, Powered by Chef, Powered by Progress, Progress, Progress Software Developers Network, SequeLink, Sitefinity (and Design), Sitefinity, Sitefinity (and design), Sitefinity Insight, SpeedScript, Stylized Design (Arrow/3D Box logo), Stylized Design (C Chef logo), Stylized Design of Samurai, TeamPulse, Telerik, Telerik (and design), Test Studio, WebSpeed, WhatsConfigured, WhatsConnected, WhatsUp, and WS_FTP are registered trademarks of Progress Software Corporation or one of its affiliates or subsidiaries in the U.S. and/or other countries.

Analytics360, AppServer, BusinessEdge, Chef Automate, Chef Compliance, Chef Desktop, Chef Workstation, Corticon Rules, Data Access, DataDirect Autonomous REST Connector, DataDirect Spy, DevCraft, Fiddler, Fiddler Classic, Fiddler Everywhere, Fiddler Jam, FiddlerCap, FiddlerCore, FiddlerScript, Hybrid Data Pipeline, iMail, InstaRelinker, JustAssembly, JustDecompile, JustMock, KendoReact, OpenAccess, PASOE, Pro2, ProDataSet, Progress Results, Progress Software, ProVision, PSE Pro, Push Jobs, SafeSpaceVR, Sitefinity Cloud, Sitefinity CMS, Sitefinity Digital Experience Cloud, Sitefinity Feather, Sitefinity Thunder, SmartBrowser, SmartComponent, SmartDataBrowser, SmartDataObjects, SmartDataView, SmartDialog, SmartFolder, SmartFrame, SmartObjects, SmartPanel, SmartQuery, SmartViewer, SmartWindow, Supermarket, SupportLink, Unite UX, and WebClient are trademarks or service marks of Progress Software Corporation and/or its subsidiaries or affiliates in the U.S. and other countries. Java is a registered trademark of Oracle and/or its affiliates. Apache and Kafka are either registered trademarks or trademarks of the Apache Software Foundation in the United States and/or other countries. Any other marks contained herein may be trademarks of their respective owners.

Please refer to the NOTICE.txt or Release Notes – Third-Party Acknowledgements file applicable to a particular Progress product/hosted service offering release for any related required third-party acknowledgements.

September 2022

Product version: Progress OpenEdge Pro2 6.4

Preface

Purpose

This manual explains how to navigate through the OpenEdge® Pro2 web user interface. The Pro2 Web Interface enables users to monitor replication statistics such as processed record counts per thread and table performance. It provides details about the dashboard, the manage replication tab, properties and actions windows. Additionally, this section also explains in detail the replication process in Pro2.

For the latest documentation updates see the OpenEdge Product Documentation Overview page on Progress Communities:

<https://community.progress.com/technicalusers/w/openedgegeneral/1329.openedge-product-documentation-overview.aspx>.

Audience

This book is intended for database administrators (DBA's) and consultants to help them to have an easy and simple way of configuring the replication process. It assumes a fundamental knowledge of both OpenEdge and DataServer for MS SQL and Oracle.

Organization

[Dashboard](#)

Describes the statistics such as processed record count per thread and thread load in visually informative charts and graphs.

[Manage replication](#)

Describes the process of creating a new replication process, managing the existing replication process and deleting replication process.

[Properties](#)

Describes the process to edit, create and delete properties in the replbasev4.ini file.

[Actions](#)

Discusses the idea of the modifications and the enhancements that can be done to the replication thread in detail.

Documentation conventions

See [Documentation Conventions](#) for an explanation of the terminology, format, and typographical conventions used throughout the OpenEdge content library, including information about the following:

- Using ABL documentation
- Examples of syntax descriptions
- OpenEdge messages

Pro2 web interface

The Pro2 web user interface simplifies replication setup, configuration, and monitoring and provides a you with the means to complete complex tasks such as bulk-loading. The Pro2 web user interface is browser independent and runs in an instance of PAS for OpenEdge. Use this interface to monitor replication statistics such as processed record counts per thread and table performance. The web user interface includes the following tabs:

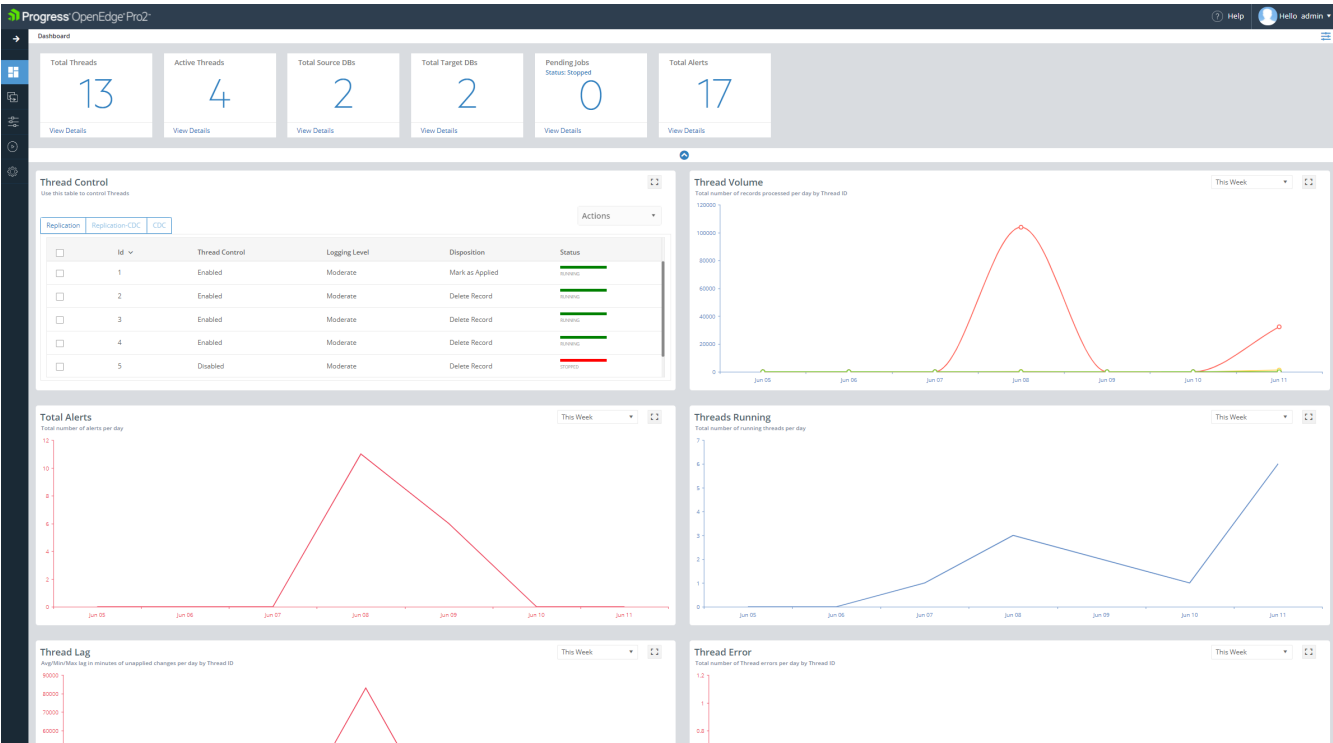
- Dashboard
- Manage Replication
- Properties
- Action
- Settings

For details, see the following topics:

- [Dashboard](#)
- [Manage replication](#)
- [Properties](#)
- [Actions](#)

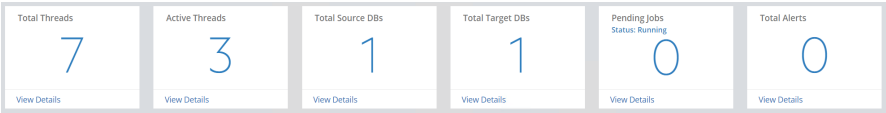
Dashboard

Dashboard displays statistics such as processed record count per thread, thread load, and others, in visually informative charts and graphs as well as thread managements options.



Watch boxes

The Watch boxes on the dashboard offer information on the various moving parts in Pro2. They consist of tiles that display information on total replication threads, the threads that are active, the total number of source and target databases, pending jobs, and total alerts.



You can find more information, and options to delete or add, by selecting **View Details** on each Watch box. Additionally, you can find the **Thread Type**, **Thread ID**, **Replication Control**, **Logging Level**, **Disposition of the thread**, and **Status** on the Details screen.

Thread control

Thread Control displays Replication, Replication-CDC, or CDC threads that can be acted on and displays the status for those threads.

Thread Control
Use this table to control Threads

Replication Replication-CDC CDC Actions

| <input type="checkbox"/> | Id | Thread Control | Logging Level | Disposition | Status |
|--------------------------|----|----------------|---------------|-----------------|---------|
| <input type="checkbox"/> | 1 | Enabled | Moderate | Mark as Applied | RUNNING |
| <input type="checkbox"/> | 2 | Enabled | Moderate | Delete Record | RUNNING |
| <input type="checkbox"/> | 3 | Enabled | Moderate | Delete Record | RUNNING |
| <input type="checkbox"/> | 4 | Enabled | Moderate | Delete Record | RUNNING |
| <input type="checkbox"/> | 5 | Disabled | Moderate | Delete Record | STOPPED |

Thread Control

You can use Thread Control to enable or disable threads. This will enable them, or prevent them from replicating.

Logging levels

- **None**—No logging
- **Minimum**—Writes start and stop time of replication process and writes number of records processed per cycle
- **Moderate**—Minimum logging levels plus any errors
- **Extended**—Moderate logging level plus warnings
- **Verbose**—Extended logging level plus information for each individual replqueue record processed

Typically, logging level is set to verbose during initial implementation and testing. It is possible to for the Logs to become too large verbose mode. To avoid to avoid this, lower the logging level after your initial implementation.

Disposition

The Disposition box allows you to choose what happens to the replication record after it is processed. Typically, Queue Disposition is set to **Delete Record**. When **Delete Record** is selected, the Replication Processor deletes the replication record once it is processed. When **Mark Record Applied** is selected, the Replication Processor modifies the replication record by setting its “Applied” flag to true. No further action is taken on applied records with this setting and a separate `PurgeQueue.p` procedure needs to be run when the queue records are no longer needed and can be deleted.

Status

The Replication Status shows the status of the replication. The status shows as **RUNNING** when the Replication Processor is either actively cycling through the queue or is in a paused state between cycles. This is also indicated by a green status bar.


If the Processor is neither active nor sleeping, then the status is **STOPPED**, indicated by a red status bar.

Graphs

There are five graphs present on the dashboard Total Alerts, Thread Volume, Thread Running, Thread Lag, and Thread Error

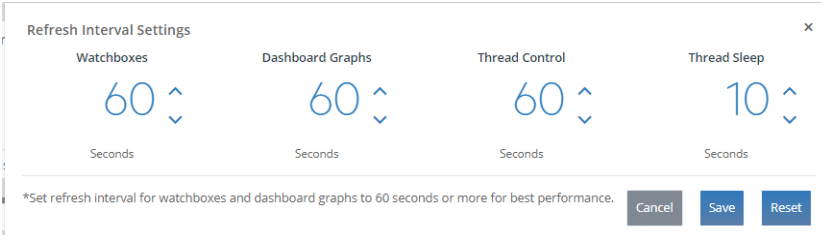
- **Total Alerts**—This graph specifies the alerts that are triggered when a replication thread stalls or stops. You can check the alerts for the current week or the week before.
- **Thread Volume**—This graph represents the volume of replication threads running at a point and time. You can have a consolidated data of the same for the current week and the week before.
- **Thread Running**—This graph represents the replication threads running at a point and time. You can have a consolidated data of the same for the current week and the week before.
- **Thread Lag**—This graph represents the replication records lag time between creation and processing, at a point and time per day, by thread number.
- **Thread Error**—This graph represents thread errors while processing records at a point and time per day by thread number.

Dashboard settings

Use the filter button  to customize the dashboard according to your needs.

You can modify the refresh interval settings for:

- Watch boxes
- Dashboard Graphs
- Thread Control
- Thread Sleep



The dialog box titled "Refresh Interval Settings" contains four input fields with up and down arrows. The first three fields are labeled "Watchboxes", "Dashboard Graphs", and "Thread Control", each with a value of "60" and the unit "Seconds" below. The fourth field is labeled "Thread Sleep" with a value of "10" and the unit "Seconds" below. At the bottom, there is a note: "*Set refresh interval for watchboxes and dashboard graphs to 60 seconds or more for best performance." and three buttons: "Cancel", "Save", and "Reset".

The refresh interval is measured in seconds and automatically refreshes the dashboard at the desired interval. Click **Save** to apply changes. To set the refresh interval settings to the default values, click **Reset**.

Note: The minimum refresh time permitted is 10 seconds. However, you can turn off automatic refresh by setting the interval to 0 (Zero).

Manage replication

This tab is responsible for creating a new replication set up, managing the existing replication setup, and deleting the existing replication setup.

Creating a new replication process consists of the following steps:

1. Select Source
2. Get Source side schema (WAN configuration only)
3. Select Target
4. Generate Target Schema
5. Mapping
6. CDC Mapping
7. Advanced Configuration
8. Generate Code

The **+New** tab begins the creation of new replication setup. Clicking on this button opens the **Create Replication** wizard page.

Select source

This window allows you to indicate the Source Database details to be replicated. You are required to fill the following fields:

- Source Logical Database Name
- Source Physical Database Name
- Database Source Path
- Source Database Mode
- Database Host Name
- Database Port/Service
- User Name
- Password

The screenshot displays the 'Progress OpenEdge Pro2' web interface. The top navigation bar includes 'Help' and 'Hello admin'. The left sidebar shows a menu with 'Pro2: WAN', 'Dashboard', 'Manage Replication', and a list of steps: 'Select Source', 'Get Source side schema', 'Set Target', 'Generate Target Schema', 'Mapping', 'CDC Mapping', 'Advanced Configuration', and 'Generate Code'. The 'Select Source' step is currently active. The main content area features a dashboard with six metrics: 'Total Threads' (17), 'Active Threads' (4), 'Total Source DBs' (2), 'Total Target DBs' (2), 'Pending jobs Since Running' (0), and 'Total Alerts' (0). Below the dashboard, a section titled 'Create Replication : Select the Source Database to be Replicated' contains a note about WAN connections. The form fields include: 'Source Logical Database Name' (with a dropdown showing 'sports'), 'Source Physical Database Name' (with a dropdown showing 'sports_omp'), 'Database Source Path' (text input with 'C:\localdb'), 'Source Database Mode' (dropdown with 'Trigger' selected), 'Database Host Name' (text input with 'localhost'), 'Database Port/Service' (text input), 'User Name' (text input), and 'Password' (password input). At the bottom of the form are 'Test Connection', 'Cancel', and 'Save' buttons.

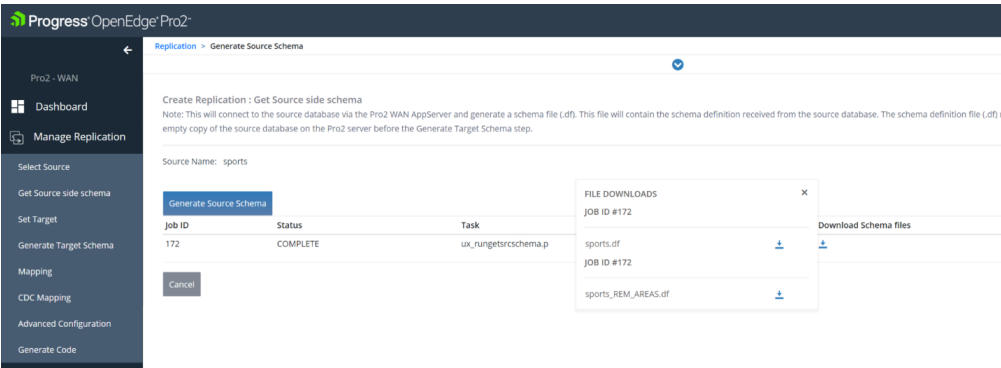
You can also test if you are successfully connected to the source database directly by clicking the Test Connection button. If successful, you will receive a message. Once the test is successful, click **Next** to select the **Target DB**.

Note: For WAN configuration, these data items are for connecting to the local empty source database except for Source Database Mode field.

Get Source side schema

This window allows you to generate the schema file necessary to build a local source empty database. This connects to the source database via the Pro2 WAN application server and generates a schema file (.df). This file contains the schema definition received from the actual source database.

The schema definition file (.df) must be downloaded and applied to the local empty source database copy which has created on the Pro2 server machine.



Select target

Use this window to enter the target database details. The target is where the source data will be replicated to. Because Pro2 replicates data to a database on your behalf, you must provide the information for the following fields.

| Field name | Description |
|-----------------------------------|---|
| Target Database Type | Use this field to designate what type of database your target database is. Choose either Microsoft SQL Server, Oracle database, or OpenEdge database. |
| Target Database Name | This field indicates the logical name of the target database. |
| Schema Holder Database Path | Use this field to enter the schema holder database. |
| Target Schema Image | The target schema image contains the schema information, such as table and column definitions, from the target database schema. The target schema image is stored in the schema holder database. |
| Schema Holder Database | The schema holder is a small standalone Progress OpenEdge database that does not hold actual data. It stores the schema image of the target database and is required to enable mapping between the source and target database for non-OpenEdge target databases. The schema image of the target database is pulled into the schema holder by the OpenEdge DataServer update function. |
| Target Open Database Connectivity | The target ODBC connection is configured through the Windows ODBC setup tool. After it is defined in windows, parameters are added to a .pf file which enables a connection to the target database with the parameters that you define. |
| User Name | The user name for the target database. |
| Password | The password for the target database. |

The screenshot shows the Progress OpenEdge Pro2 web interface. The left sidebar contains navigation links: Pro2 - LAN, Dashboard, Manage Replication (selected), Select Source, Set Target, Generate Target Schema, Mapping, CDC Mapping, Advanced Configuration, Generate Code, Properties, Actions, and Settings. The main content area is titled 'Replication > Select Target' and contains the following fields:

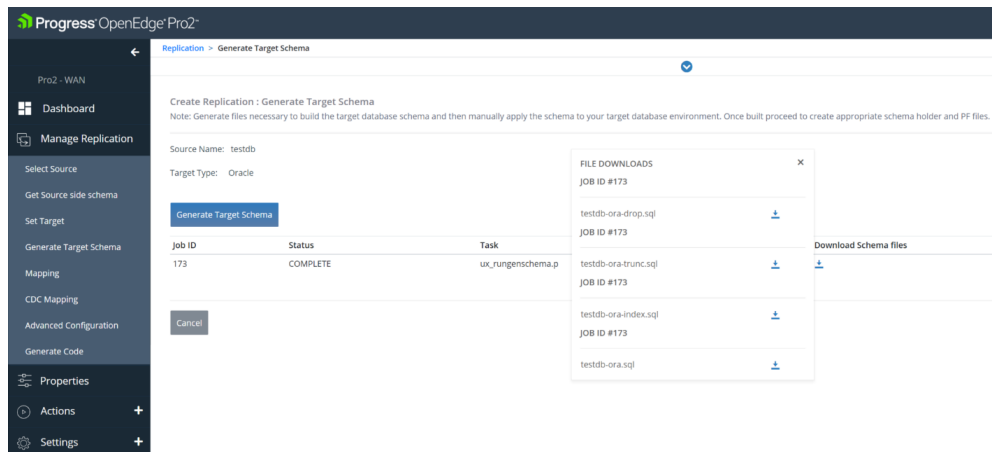
- Create Replication :** Select the database into which Pro2 will replicate
- Source Database Name:** sports
- Target Database Type:** MSS (dropdown menu)
- Target Database Name:** sportstgt
- Schema Holder Database Path:** C:\P2PSQLdb
- Target Schema Image:** sportssql
- Schema Holder Database:** sportssh
- Target ODBC Connection:** sportsodbc
- User Name:** admin
- Password:** *****

Note: The .pf files (replProc.pf and sports.pf) are automatically updated in the background. The name of the .pf file is based on the logical name of the source database.

Click **Next** to move to the **Generate Target Schema** window.

Generate target schema

This window allows you to generate files necessary to build the target database schema and then manually apply the schema to your target database environment by downloading the files. After it is built, you need to proceed to create or update the appropriate schema holder to sync with the target database schema. After the schema holder is ready, click **Next** to reach the **Mapping** window.



Mapping

This page loads both source and target side tables automatically with a dynamic connection. Auto mapping requires a selection of one or more source tables while manual mapping requires a 1:1 link of either a single source table-field to a single target table-field. Unmapping table-fields is similar to manual mapping. You must choose one or more table-fields to unmap from one another.

Replication > Mapping

Create Replication : Mapping Source to Target Tables

Note: Auto mapping requires selection of one or more source tables and manual mapping requires a 1 : 1 linking (single source table or field to a single target table or field)

UnMapped Tables: sports

☐ Source Tables

☐ Item
 ☐ Local-Default
 ☐ Order
 ☐ Order-Line
 ☐ Ref-Call
 ☐ Salesrep
 ☐ State

☐ Target Tables

☐ Item
 ☐ Local-Default
 ☐ Order
 ☐ Order-Line
 ☐ Ref-Call
 ☐ Salesrep
 ☐ State

Mapped Tables

| Source Tables | Target Tables |
|-----------------------------------|---------------|
| <input type="checkbox"/> Customer | Customer |
| <input type="checkbox"/> Invoice | Invoice |

The **Source Tables** box is used to designate what data is to be mapped to the target database. It populates the Source Tables and Target Tables boxes with the appropriate unmapped tables.

The **Source Tables/Target Tables Viewers** selection boxes contain the table names of tables in the selected database that are not already included in the mapping table.

The **Map** button takes the selected table from the Source Tables box and the selected table from the Target Tables box and creates a Table mapping record for them. Additionally, you are not required to map the entire table, only the fields that you want to replicate.

The **Unmap** button removes the mapping of the tables and fields that you indicate.

Use the **Automap** feature to auto-map source database tables to target database tables. You can select multiple tables within the Source Tables box and the application automatically maps those tables with matching target tables. Pro2 continues to create Field Cross-Reference records for any matching fields of those tables. This is possible for only tables and fields that have a direct matching table name or field name in the corresponding target.

Once you have mapped the required tables, click **Next** to access the **Advanced Configuration** window.

CDC mapping

This page displays the mapped source database tables and enables you to create a CDC active or inactive policy for the selected tables.

Pro2 replication is currently designed to capture records that have changed. In CDC, this is known as a Policy. Activating a policy forces the OpenEdge database to immediately start tracking changes to the selected tables. Deactivating a policy forces the OpenEdge database to immediately stop tracking changes to the selected tables.

To enable CDC mapping:

1. From the **Manage Replication** window on the Pro2 web interface, select the **Source Database**.
2. Ensure that the **Source DB Mode** is **CDC**. Choose the **Test DB Connection** button. A message will indicate if CDC is enabled on the source database.
3. Click the **Mapping** option and ensure that all tables to be replicated have been mapped from source to target.
4. Select **CDC Mapping**.
5. Select the source tables to be mapped for CDC and click **Add**.
6. Select the **Policy Settings** for each of the **Source Tables** in the **Add / Activate Policy** column.



Create Replication : Add or Remove CDC Policy

Note: Activating a policy forces the OpenEdge database to immediately START tracking changes to the selected table(s). Inactivating a policy forces the OpenEdge

Source Database: sports

Mapped Source Tables

| | |
|--------------------------|--------------|
| <input type="checkbox"/> | Source Table |
| <input type="checkbox"/> | Salesrep |
| <input type="checkbox"/> | State |

► Add

◄ Remove

Table List

Add / Activate Policy

Activate Policy

Policy Settings ▼

| | | |
|--------------------------|--------------|-----------------|
| <input type="checkbox"/> | Source Table | Policy Status ▼ |
| <input type="checkbox"/> | Customer | ACTIVE |
| <input type="checkbox"/> | Invoice | ACTIVE |
| <input type="checkbox"/> | Item | ACTIVE |
| <input type="checkbox"/> | Order | ACTIVE |
| <input type="checkbox"/> | Order-Line | ACTIVE |

Advanced configuration

This window allows you to change thread assignments and other optional control flags.

Progress OpenEdge Pro2

? Help
 Hello admin ▼

Replication > Advanced Configuration

v5 Prop Test2 - LAN

Dashboard

Manage Replication

Select Source

Set Target

Generate Target Schema

Mapping

CDC Mapping

Advanced Configuration

Generate Code

Properties

Actions +

Settings +

Create Replication : Advanced Configuration

Change thread assignments and other optional control flags.

The Global Controls section allows setting items on all selected tables in the list.

Global Controls

Select Table ▼

Select Thread ▼

Select GenQRec ▼

Select ProcQRec ▼

Select Include/Exclude ▼

| <input type="checkbox"/> | Source Tables | Thread# | Gen Queue Records | Process Queue Recor... | Include/Exclude |
|--------------------------|---------------|---------|-------------------------------------|-------------------------------------|-------------------------------------|
| <input type="checkbox"/> | Customer | 1 ▼ | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| <input type="checkbox"/> | Invoice | 1 ▼ | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| <input type="checkbox"/> | Item | 1 ▼ | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| <input type="checkbox"/> | Order | 1 ▼ | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| <input type="checkbox"/> | Order-Line | 1 ▼ | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| <input type="checkbox"/> | Salesrep | 1 ▼ | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| <input type="checkbox"/> | State | 1 ▼ | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |

Pro2: Pro2® Web User Interface: Version 6.4

19

Global Controls

You can use the **Global Controls** drop down menus to update **Thread #**, **Gen Queue Records**, **Process Queue Records**, and **Include/Exclude** flags by selecting source tables. If you use the **Select All Source Tables** option, then all of the tables that are listed are selected and are affected by your following actions. This includes tables on multiple pages.

Table List

Click the **Table List** button to enter a list of comma-separated table names that you want to update. The checkboxes for only the tables specified in the list are selected, and you can then perform actions on the selected tables. You can change the **Thread #**, **Gen Queue Records**, **Process Queue Records**, and **Include/Exclude** parameters for the given list of tables by using the **Table List** button.

Source Tables

The **Source Tables** column consists of the mapped and cross-referenced tables that are mapped in the Mapping section.

Thread

The **Thread #** specifies which thread the table is in. Mapped tables can be shared with up to five separate threads. By default, all mapped tables are assigned to thread 1.

Gen Queue Records

The **Gen Queue Records** is used to stop/start generating **replqueue** records for the selected table (i.e., turn off replication but leave triggers in schema), as well as disable CDC policies or triggers depending on the method you are using. It can be used to set the **ReplTableXref.genqrec** flag to **FALSE** for this table. The default setting for **ReplTableXref.genqrec** is True. If you need to change **ReplTableXref.genqrec** to False for an entire database, the following ABL code can be run from the Progress Editor to stop generating queue records for all tables: `FOR EACH Repl_TableXRef: GenQRec = FALSE. END.`

Note: If Generate Queue Record is off, replication queue records is not generated for that table even if Process Queue Records is on.

Process Queue Records

Process Queue Records is used to monitor on any **replqueue** records that are generated. If the checkbox is unchecked, the **Replqueue** records are not processed (i.e., replicated). Queue records stay in **replqueue** until **Process Queue Records** flag is selected. It is selected by default. **Process Queue Records** functions similarly to the option for **Generate Queue Record** but sets **ReplTableXref.procqrec** to **FALSE**.

Include/Exclude

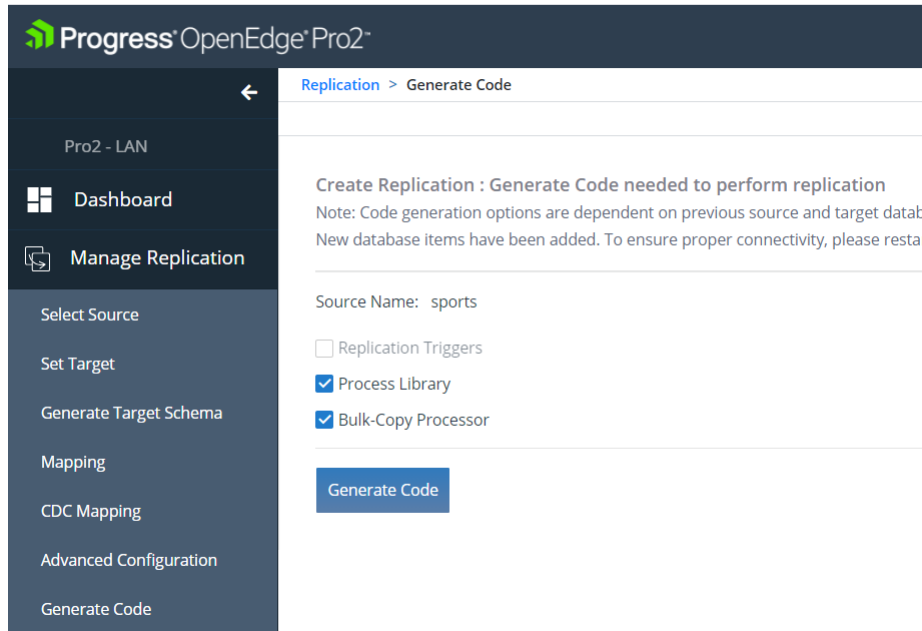
Include/Exclude is selected by default. If this is unchecked, any changes to the schema for this table will not be included in the Differential Schema generated by the **Generate Schema** window.

Generate code

This window allows you to choose replication triggers, process library, bulk-copy processor, and run the Generate Code procedure. Code generation options are dependent on previous source and target database types and connection requirements.

After you click **Generate Code**, you are redirected to the dashboard with a newly created replication instance.

Note: To ensure proper connectivity, restart your PAS for OpenEdge instance and jobrunner process.



Properties

The initial settings for all properties are loaded from the `replbaseversion#.ini` file. The file can be found in the root Pro2 install folder. You can modify the properties manually or use the **Properties** tab in the user interface.

Use the following table as a reference for the configurable Pro2 properties and their default configurations.

Table 1: General properties

| Property | Default Value | Description |
|-----------------------|---------------|---|
| ADMIN_TITLE | Pro2 | You can define the title of a particular Pro2 instance. |
| ALERT_1_PER_DAY | YES | <p>Sends 1 replication thread status email per-day for all three thread types. After sending one status alert email, Pro2 will not send another on the same day for the same thread unless the alarms are cleared.</p> <hr/> <p>Note: This property overrides all other alert properties and settings.</p> <hr/> |
| APPSRV_CDCPROC_SRCODE | NO | Use this property to can check the status of CDC threads that are running on the application server. If this property exists, and the value is "YES", the thread status will be visible. This property is disabled by default. |

| Property | Default Value | Description |
|--------------------------|-----------------|--|
| Bulk_Max_Cache | 25 | Sets the maximum number of <code>rowids</code> that are cached per-table during bulk loads. This value is used during bulk load restarts. |
| CDCADMIN_BASENAME | cdcadmin | Contains the name of the log file for <code>cdcadmin</code> records during each <code>cdcadmin</code> process cycle. Database name, date, queue thread number, and log file suffix are appended to the file name. For example, <code>cdcadmin_sports_20201117-1.log</code> . |
| CDC_MAX_PROCESSED | 5000 | The number of rows per table a CDC Admin thread processes before it begins processing additional tables in the same block. If its value of 0 is set, then all rows in a table are processed before the thread moves on to the next table. |
| CDC_PURGE_DAYS | 10 | CDC records with create dates older than this value are purged during the CDC purge process. This value is used in conjunction with a disposition of "mark as applied". |
| CDC_QUEUE_COMPRESSION | NO | Enables or disables queue compression for CDC threads. |
| CHUNK_SIZE | 1000000 | This property defines the size of file to be transferred. |
| DATA_RETENTION_DAYS | 14 | This property sets the maximum number of days that a Pro2 database dashboard and job data is stored before being purged from the database. |
| DBDOWN_RETRIES | 5 | Sets the maximum number of attempts to reconnect database instance if it stops working. |
| DB_LOCATION | db | Contains the location of the Pro2 database instances (repl, Pro2, source and target database instances). |
| DELETE_TRIG_DIRECTORY | bprepl/repl_d | Contains the location of the REPLICATION-DELETE schema trigger procedures. |
| DEL_TRIG_TEMPLATE | tplt_repltrig.p | Contains the name of the template used in generating the REPLICATION-DELETE schema trigger procedures. |
| ENTERPRISE_PUSH_LOG_FILE | enterprise_push | Contains the name of log file for the enterprise push job. The date and log file suffix are appended to name. It is used by <code>Repllogchk</code> . |
| EXTENT_DELIMITER | ## | SQL and Oracle schema generation, and SQL and Oracle differential use this value as the delimiter for definitions of Progress array fields. |

| Property | Default Value | Description |
|-------------------------|--------------------------|---|
| FETCH_WITH_LOCK | NO | If this property value is YES, then an exclusive record lock is initiated while the record is fetched from the source database, otherwise the record is fetched with out being locked. This is a LAN only property. The WAN counterpart property is APPSRV_FETCH_WITH_LOCK. |
| HEARTBEAT_CHECK_MINUTES | 15 | Sets the time stamp to detect a replication threads most recent replication activity and to detect if a replication thread is stalling. The heart beat check is applicable to CDC and Replication-CDC threads. |
| INCLUDE_DIRECTORY | bprepl/repl_inc | Contains the table programs present in the source database. Each table program contains the field assignment from source to target. |
| INCLUDE_LOB | NO | Enables or disables the inclusion of LOB fields in the replication process, schema generation process, schema differential process, and bulk-load process. |
| JAVA_TRIGGER_DIR | bprepl/repl_jtrig/ | The folder name to where Java triggers are generated. |
| JOBRUNNER_LOG_FILE | jobrunner | Contains the name of file for logging job details during each jobrunner cycle. Date and log file suffix are appended to name. It is used by Repllogchk. |
| LOG_ALERT_DAILY_LIMIT | 10 | Sets the daily limit on replication log generation process. |
| LOGICAL_DELETE_TABLES | XXX_so_mstr,XXX_scd_det | A comma delimited list of table names that are logically deleted, as opposed to physically deleted, from the target database . |
| LOG_DIRECTORY | bprepl/repl_log | The folder location of Pro2 log files. |
| LOG_FILE_SUFFIX | .log | An extension used for Pro2 log file names. |
| LOG_RETENTION_DAYS | 10 | Sets the maximum number of days that a log file is stored before being purged from the disk. |
| MASS_LOAD_DIRECTORY | bprepl/repl_mproc | This property designates the folder that Pro2 Bulk Load procedures are generated. |
| MASS_LOAD_LOG_DIR | bprepl/repl_mproclog | Designates the Bulk Load log files storage folder location. |
| MASS_LOAD_TEMPLATE | tpl_mproc_restat_atopshp | Sets the name of the template procedure that is used during the bulk-load generation. |

| Property | Default Value | Description |
|---------------------------|--------------------------|---|
| MASS_LOAD_TEMPLATE_ORACLE | tmpl_mreplproc_oracle.p | Sets the name of the template procedure that is used during the bulk-load procedure for a target Oracle database. If this property does not exist, then the template is considered present on the "MASS_LOAD_TEMPLATE" property. |
| MASS_LOAD_TEMPLATE_PRO2 | tmpl_mreplproc_pro2pro.p | Sets the name of the template procedure that is used during the bulk-load generation when the source as well as target databases belong to OpenEdge. If this property does not exist, then the template is considered present on the "MASS_LOAD_TEMPLATE" property. |
| MASS_LOAD_TEMPLATE_SQL | tmpl_mreplproc_sql.p | Sets the name of the template procedure that is used during the bulk-load generation for a target Microsoft SQL database. If this property does not exist then the template is considered present on the "MASS_LOAD_TEMPLATE" property. |
| MAX_CHAR_WIDTH | 250 | Sets the maximum character width character width for the source precession field at the time of SQL or Oracle generation and differential |
| MAX_DEC_PREC | 38 | Maximum number of positions to the right of the decimal when generating a target schema or differential for Microsoft SQL Server or Oracle. |
| MERGED_TRIGGER_TEMPLATE | tmpl_mrgdtrig.i | Contains the default information required to list tables that require merged triggers. |
| MERGED_TRG_DIRECTORY | bprepl/repl_mgtrig | Contains the directory listing tables of merged trigger procedures. |
| MERGED_TRG_TEMPLATE | tmpl_mrgdtrig.i | The name of the template that is used to generate merged triggers. |
| MSS_DB_PF | MSS-DB-EXAMPLE.pf | <p>Example .pf file for Microsoft SQL server database connections for the Pro2 user interface. This example file can be used to automatically generate a database .pf file after all values of the source and target page are passed from the manage replication wizard.</p> <hr/> <p>Note: Actual .pf file should be named after the logical database name.</p> <hr/> |

| Property | Default Value | Description |
|-------------------------|--|---|
| ORA_DB_PF | ORA-DB-EXAMPLE.pf | <p>Example .pf file for Oracle database connections for Pro2 user interface. This example file can be used to automatically generate a database .pf file after all values of the source and target page are passed from the manage replication wizard.</p> <hr/> <p>Note: Actual .pf file should be named after the logical database name.</p> <hr/> |
| ORACLE_NATIVE_ROWID | NO | <p>If the property exists and contains a value of "YES", the Pro2Oracle SQL/Differential generator changes the standard output to:</p> <ol style="list-style-type: none"> 1. Suppress output the progress_recid column 2. Suppress output the progress_recid index 3. Suppress output of the target table specific sequence definitions 4. The datatype for prrowid changes from varchar2(36) to number |
| OUTPUT_BASENAME | replproc | Contains the replication processor log file name. The times tamp, queue, thread number, and log file suffix are appended to the base file name, for example: replproc20210209-1.log. |
| OUTPUT_BASENAME_REPLCDC | replcdcproc | Contains the Replication-CDC processor log file name. The time stamp, queue thread number, and log file suffix are appended to the base name, for example: replproc20210209-1.log. |
| PF_DIRECTORY | bprepl/Scripts | Directory containing the .pf files, scripts, and shortcuts to start the Pro2 web user interface, replication processors, and bulk-load procedures. |
| PRO2_SYNC_DIRECTORY | bprepl/repl_pro2trigs | Contains the folder path which is used in the utils/ReplPro2TrigInsert.p program to insert triggers into Pro2 sync tables. |
| PRO2_SYNC_TABLES | <ul style="list-style-type: none"> • Repl_Control • Repl_CustDefs • Repl_CustFlds • Repl_DBXref • Repl_TableXref • Repl_FieldXref • Repl_Properties • Repl_ThreadControl | Used in utils/Replpro2TrigInsert.p and utils/Replpro2TrigDelete.p program files which are used to delete or insert the replication triggers from Pro2 database tables. |

| Property | Default Value | Description |
|-------------------------------------|---------------------|---|
| PROC_DIRECTORY | bprepl/repl_proc | Sets the operating system directory where the replication processor procedure library is located. |
| PROC_CDC_DIRECTORY | bprepl/replcdc_proc | Sets the operating system directory where the Replication-CDC processor procedure library is located. |
| PROC_CDC_TEMPLATE | tmpl_replcdcproc.p | Sets the name of the template procedure that is used during the creation of the Replication-CDC procedure library for a LAN configuration. |
| PROC_CDC_TEMPLATE_PROORA | tmpl_replcdcproc.p | The name of the template that is used to generate Replication-CDC procedures for Oracle based target databases. If this property does not exist, then the template is considered present on the "PROC_CDC_TEMPLATE" property. |
| PROC_CDC_TEMPLATE_PROPRO | tmpl_replcdcproc.p | The name of the template that is used to generate replication procedures for OpenEdge based target databases. If this property does not exist, then the template is considered present on the "PROC_CDC_TEMPLATE" property. |
| PROC_CDC_TEMPLATE_PROSQL | tmpl_replcdcproc.p | The name of the template that is used to generate replication procedures for Microsoft SQL Server based target databases. If this property does not exist, then the template is considered present on the "PROC_CDC_TEMPLATE" property. |
| PROGRESS_DB_Pf | PRO-DB-EXAMPLE.pf | <p>Example .pf file for OpenEdge database connections for Pro2 user interface. This example file can be used to automatically generate a database .pf file after all values of the source and target page are passed from the manage replication wizard.</p> <hr/> <p>Note: Actual .pf file should be named after the logical database name.</p> <hr/> |
| PROC_TEMPLATE | tmpl_replproc.p | Sets the name of the template procedure used during the creation of the replication procedure library. |
| PROC_TEMPLATE_PROSQL | tmpl_replproc.p | The name of the template that is used to generate replication procedures for Microsoft SQL Server based targets. If this property does not exist, then the template is considered present on the "PROC_TEMPLATE" property. |
| PROC_TEMPLATE_PROORA | tmpl_replproc.p | The name of the template that is used to generate replication procedures for Oracle based targets. If this property does not exist, then the template is considered present on the "PROC_TEMPLATE" property. |

| Property | Default Value | Description |
|------------------------------|---|--|
| PROC_TEMPLATE_PRO2PRO | tmpl_replproc_pro2pro.p | The name of the template that is used to generate replication procedures for OpenEdge based targets. If this property does not exist, then the template is considered present on the "PROC_TEMPLATE" property. |
| PRROWID_ADD_DB | NO | When this property is set to YES Pro2 prefixes the source database name to the PRROWID column that is stored in the target database. |
| REFRESH_DASHBOARD | 60 | Sets the refresh time interval for the Pro2 user interface dashboard. |
| REFRESH_REPLICATION_INTERVAL | 60 | Sets the refresh time interval for the replication threads on the Pro2 user interface dashboard. |
| REPL_RESTART_ON_SKIPPED | YES | Use this property to indicate if a thread should go through the replqueue before attempting to reprocess skipped events. Threads check for shutdown events every 1000 reads of the replqueue table. If set to YES, during the shutdown check the thread indicates in the thread log that REPL_RESTART_ON_SKIPPED is enabled and leaves the replqueue processing block. It then goes through the standard sleep interval and restarts processing from the top of the queue, oldest records first. The property is set to YES by default. |
| REFRESH_WATCHBOXES | 60 | Sets the refresh time interval for all the Watchboxes present on the Pro2 user interface dashboard. |
| REPL_LOG_BASENAME | <ul style="list-style-type: none"> repllog replproc | Contains the name of file for logging replication of records during each replication cycle. Date, queue thread number, and log file suffix are appended to name for example: repllog20210209-1.log. |
| REPLCDC_LOG_BASENAME | <ul style="list-style-type: none"> replcdclog replcdcproc | Contains the name of file for logging Replication-CDC of records during each replication cycle. Database name, Date, queue thread number, and log file suffix are appended to the file name. For example replcdclog_sports_20210308-1.log, replcdclog_qadddb_20210308-1.log. |
| SCHEMA_DEL_DIRECTORY | bprepl/repl_d | Contains the directory for the Pro2 generated database replication delete trigger procedures. |
| SCHEMA_WRI_DIRECTORY | bprepl/repl_w | Contains the directory for the Pro2 generated database replication write trigger procedures. |

| Property | Default Value | Description |
|--------------------------|---|--|
| SMTP_AUTH | <i>username , password</i> | Contains the username and password authentication details for email alerts. Note: The username and password details must be separated by a comma. |
| SMTP_DEBUG | NO | When this property is set to YES a debug log for troubleshooting email send issues is created in the <code>bprepl/repl_log</code> folder. |
| STATUS_CHECK_DEBUG | NO | When this property is set to YES a debug log for troubleshooting the status check procedures is created in the <code>bprepl/repl_log</code> folder. |
| STATUS_CHECK_MAXQUEUECNT | 10 | Used in the status check function, this property sets the number of rows per thread for the <code>replqueue</code> count. |
| STATUS_CHECK_MAXCDCCNT | 10 | Used in the status check function, this property sets the number of rows per thread for the CDC queue count. |
| SQL_DIRECTORY | <code>bprepl/repl_sql</code> | Directory for the direct Microsoft SQL assign include files, bulk copy procedures and replication procedures. Directory to store the delta .df file for Pro2 to Pro2 databases, SQL generation and differential, generate linked server SQL, and more. |
| SQL_EXCLUDE_LIST | <ul style="list-style-type: none"> • <code>ReplControl</code> • <code>ReplCustAsgn</code> • <code>ReplCustDefs</code> • <code>ReplCustFlds</code> • <code>ReplDBXRef</code> • <code>ReplFieldXRef</code> • <code>ReplProperties</code> • <code>ReplQueue</code> • <code>ReplTableXRef</code> • <code>ReplThreadControl</code> | Contains the excluded list of tables in the MS SQL database. |
| TEMPLATE_DIRECTORY | <code>bprepl/repl_tmpl</code> | Directory containing the templates used for various code generation such as the bulk-load procedure, replication procedure, and more. |

| Property | Default Value | Description |
|--------------------------|-----------------|---|
| TRIGGER_FORCE_LOWERCASE | NO | When this property is set to YES the file names that are used for trigger generation are set to lower case and the case of the source table names is ignored. |
| TGT_USE_LITERAL_NAMES | NO | Property to use the same naming convention as the source database for tables and columns without modifying them to meet the requirements of the target database (SQL only). |
| TRIGGER_EXTENSION | .t | The extension that is used when Pro2 trigger files are generated. |
| THREADACT_DETAIL_MINUTES | 30 | This property is used to assign the DETUNIT field from thread activity details table. |
| USE_3RD_PARTY_EMAIL | NO | When this property is set to YES Pro2 calls bprepl/Scripts/mail.bat or mail.sh. These files contain third party specific email calls. |
| USE_SQL_TRIGGERS | NO | This property is used for backwards compatibility with data servers that require SQL triggers for maintaining the PROGRESS_RECID column. |
| VARCHAR_TYPE | varchar | This property is used to generate the field type with a property value at the time of SQL or Oracle schema generation, for example: varchar or nvarchar. |
| WRITE_TRIG_DIRECTORY | bprepl/repl_w | Sets the operating system directory where the REPLICATION-WRITE trigger procedures are located. |
| WRI_TRIG_TEMPLATE | tplt_repltrig.p | Contains the template file used by GenReplTrigs.p to create the replication triggers. |

Table 2: WAN only properties

| Property | Default Value | Description |
|--------------------------|------------------------|--|
| APPSRV_CDC_PROC_TEMPLATE | tmpl_asCDCTgtDb.p | Sets the name of the template procedure that is used during the creation of the Replication-CDC procedure library for a WAN configuration. |
| APPSRV_FETCH_WITH_LOCK | NO | If this property value is YES, then an exclusive record lock is initiated while the record is fetched from the application server. |
| APPSRV_GET_PROGS_DIR | bprepl/AppSrv/as_mproc | Stores the WAN process libraries. |

| Property | Default Value | Description |
|-------------------------------|--|--|
| APPSRV_LOG_DIRECTORY | bprepl/repl_log | Stores the Pro2 application server log file on the source database server. |
| APPSRV_LOG_FILE | appsrv.log | Sets the name of the Pro2 application server log file on the source DB server. |
| APPSRV_MASS_LOAD_DIRECTORY | bprepl/AppSrv/as_mproc | Stores the details of a bulk-load procedure on the Pro2 server. |
| APPSRV_MASS_LOAD_TEMPLATE | tmpl_ASmproc.p | Contains the default information used in generating WAN bulk-load procedures. |
| APPSRV_PROC_DIR_TGTDB | bprepl/repl_as_tgt | Contains details of WAN replication procedures on the Pro2 server. |
| APPSRV_PROC_TEMPLATE | tmpl_asTgtDb.p | Contains default information used in generating WAN replication procedures. |
| ASCII_BULK_EXP_DATA_FULL_PATH | /apps/Pro2/XXX/bprepl/repl_export/data | Contains the location on target database server of ASCII bulk export data files. |
| ASCII_BULK_EXP_DIR | /apps/Pro2/XXX/bprepl/repl_export | Sets the Bulk export root directory. |
| ASCII_BULK_LOAD_DATA_FULLPATH | C:\Pro2\XXX\bprepl\repl_export\data | Contains the location of the target database server for ASCII bulk export data files. |
| ASCII_BULK_TGT_ROOT_DIR | C:\Pro2\XXX | Contains the location of the target database server for ASCII bulk load SQL scripts. |
| APPSRV_FETCH_COUNT | 1000 | Specifies how many records are sent across a WAN connection to the Pro2 server at each replqueue records request. |
| APPSRV_FETCH_COUNT_MASS_BULK | 50000 | Specifies the number of bulk-load replications that are sent across the WAN connection to the Pro2 server at each replqueue records request. |
| APPSRV_QUEUE_COMPRESSION | NO | Enables or disables queue compression for replication threads for WAN configuration. |

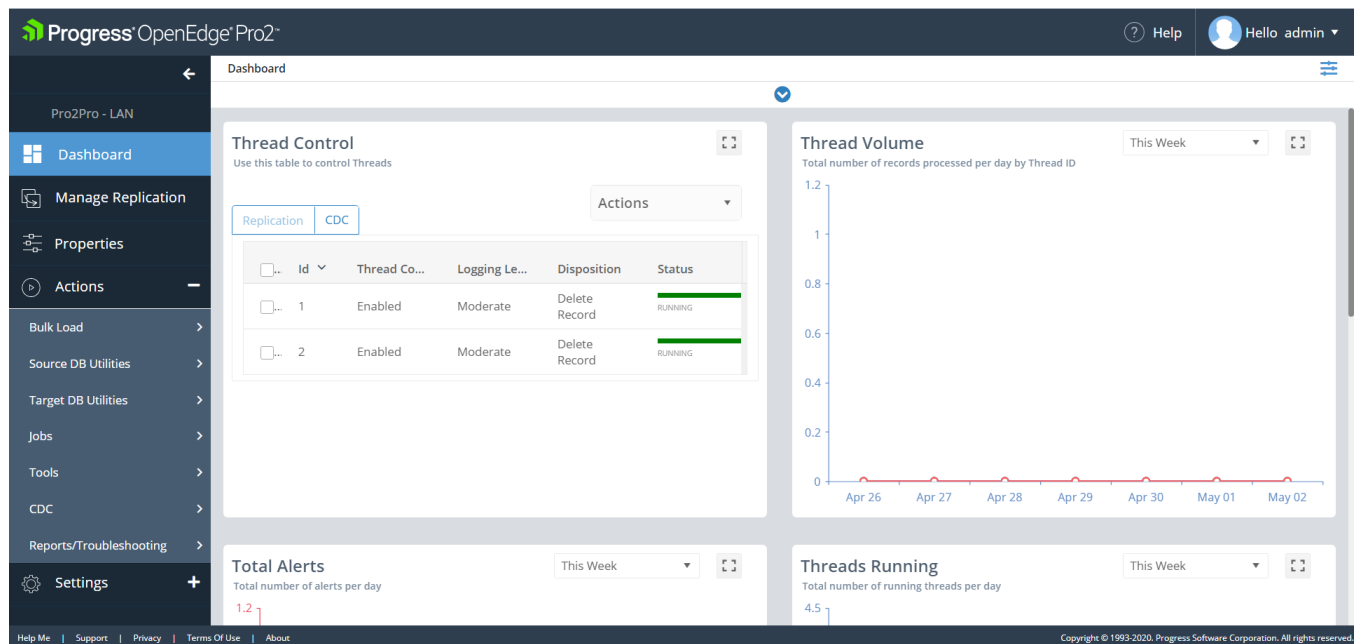
| Property | Default Value | Description |
|-------------------------------|---------------|--|
| APPSRV_FETCH_WITH_LOCK | NO | If this property value is YES, then an exclusive record lock is initiated while the record is fetched from the source database, otherwise the record is fetched with out being locked. |
| Oracle_Bulk_Transaction_Count | 25 | This property is used only in bulk-load WAN programs such as <code>tmpl_ASmproc_oracle</code> . You can use this property to set the limit of transactions to the property value. |

Actions

The **Actions** tab gives a detailed idea of the modifications and the enhancements that can be done to the replication thread. The tab consists of the following functionalities:

- Bulk Load
- Source DB Utilities
- Target DB Utilities
- Jobs
- Tools
- Reports/Troubleshooting

Note: Apart from these functionalities there are two other functionalities namely, WAN and CDC which are enabled only when the **Source DB Connection** is set to **WAN**. and the **Source DB Mode** is set to **CDC** respectively.



Bulk Load

Bulk loading is used when you need to import or export large amounts of data relatively quickly.

Run Bulk Loads: This menu item starts the Bulk Load Utility that performs a mass-record copy of the entire source database over to the DataServer target. You can also choose to bulk load only a few select tables based on the number of records they contain. For example, to begin with, you may want to bulk load only those tables that have over 100,000 records. Or, alternatively, you may want to bulk load only those tables that have less than 100,000 records. You can accomplish this by performing a database or table analysis using the DBANALYS or TABANALYS command in proutil and then using the output to create filters in the Bulk Load Utility.

Enter the appropriate information to begin a mass-record copy of source side table data to the appropriate target side table structure. The **Run** button begins the Bulk-load process.

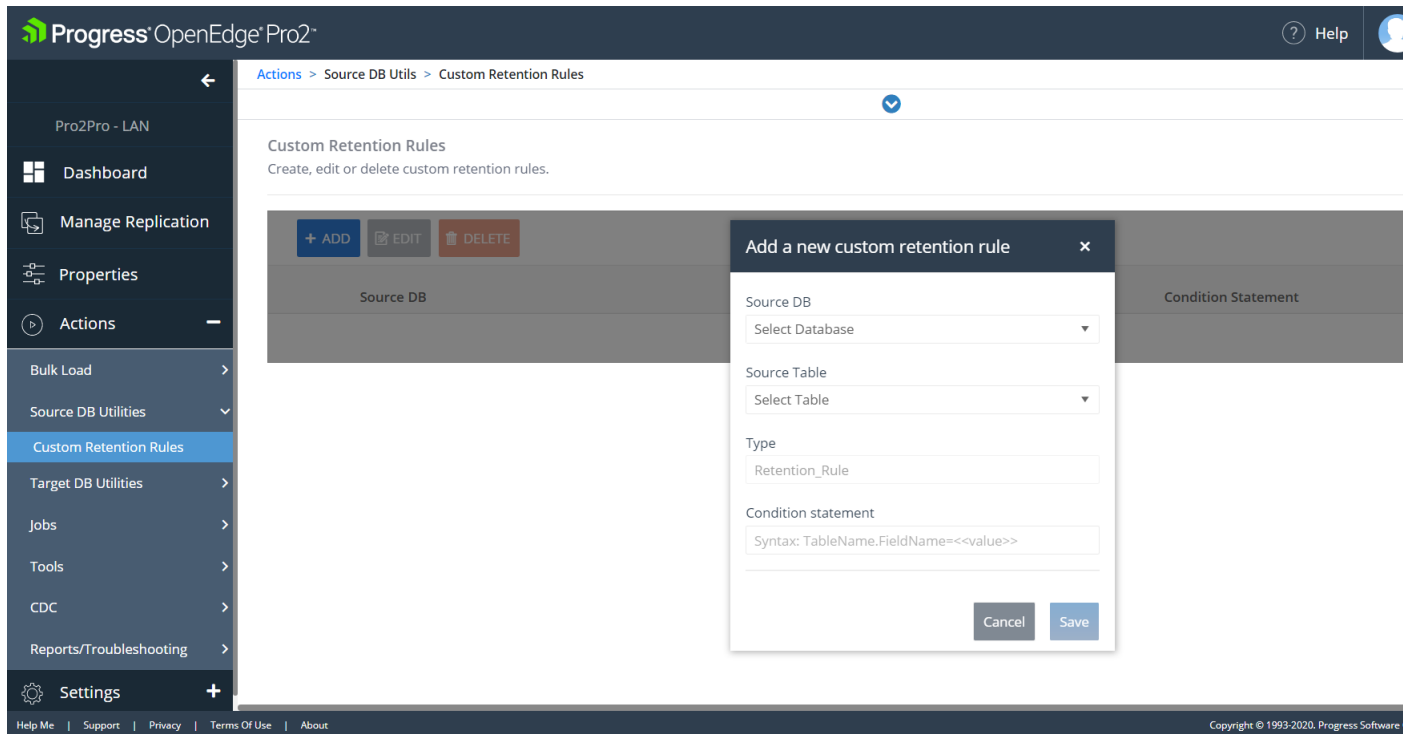
Bulk Load Report: This menu item generates a Bulk Load report after the Run Bulk Loads command is executed. You can generate the report by clicking **Proceed**. This displays the Bulk-load report on the screen. You can also choose to export the report to PDF or Excel.

For more information about the bulk load process, see [Bulk load with Pro2](#) .

Source DB Utilities

Custom Retention Rules: This menu item allows you to maintain the custom data retention rules used on specific tables. These rules are applicable in Pro2 for tasks like Bulk-Load, Bulk-ASCII-Export, Verification, and Replication. An IF statement is used to exclude Data that satisfies the condition. To set custom retention rules, perform the following steps:

1. Click **Add** to open the **Add a new custom retention rule** window



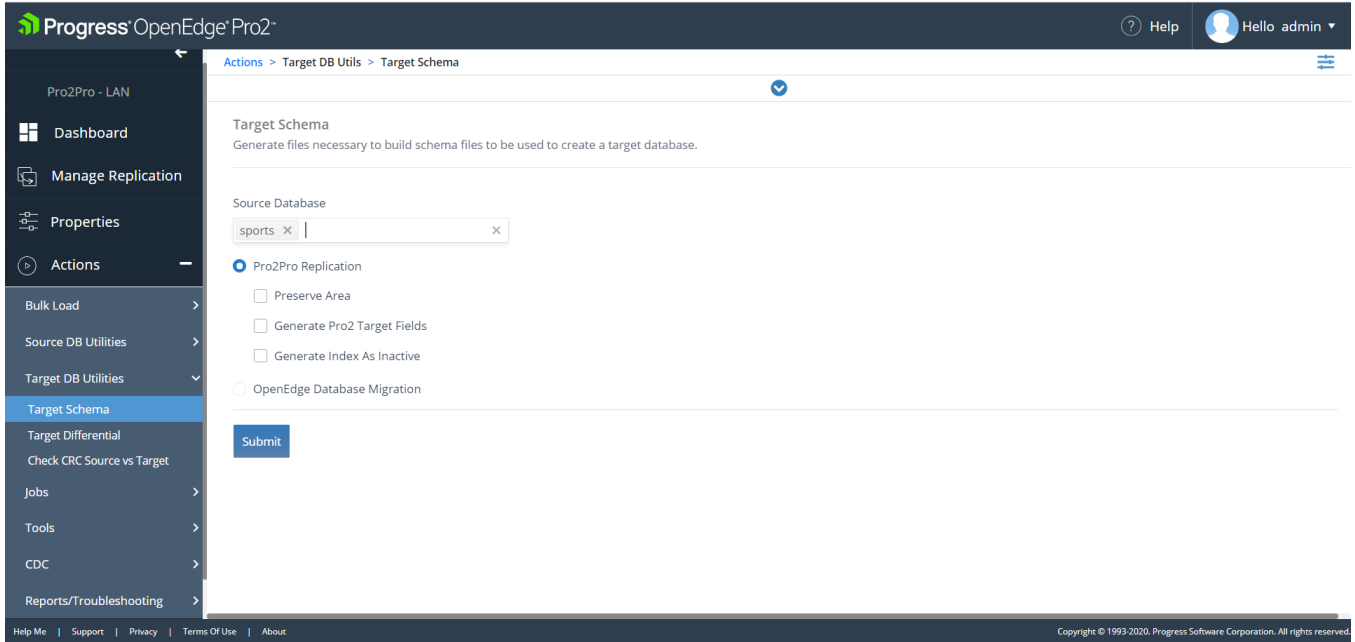
2. Select the **Source DB** and the respective **Source Table**
3. Choose the rule **Type** to specify a new **Custom Retention** rule
4. Add a condition statement
5. Click **Save**

Note: After a Custom Retention Rule is modified you need to regenerate the bulk-load and replication process library code to bring the change in to effect.

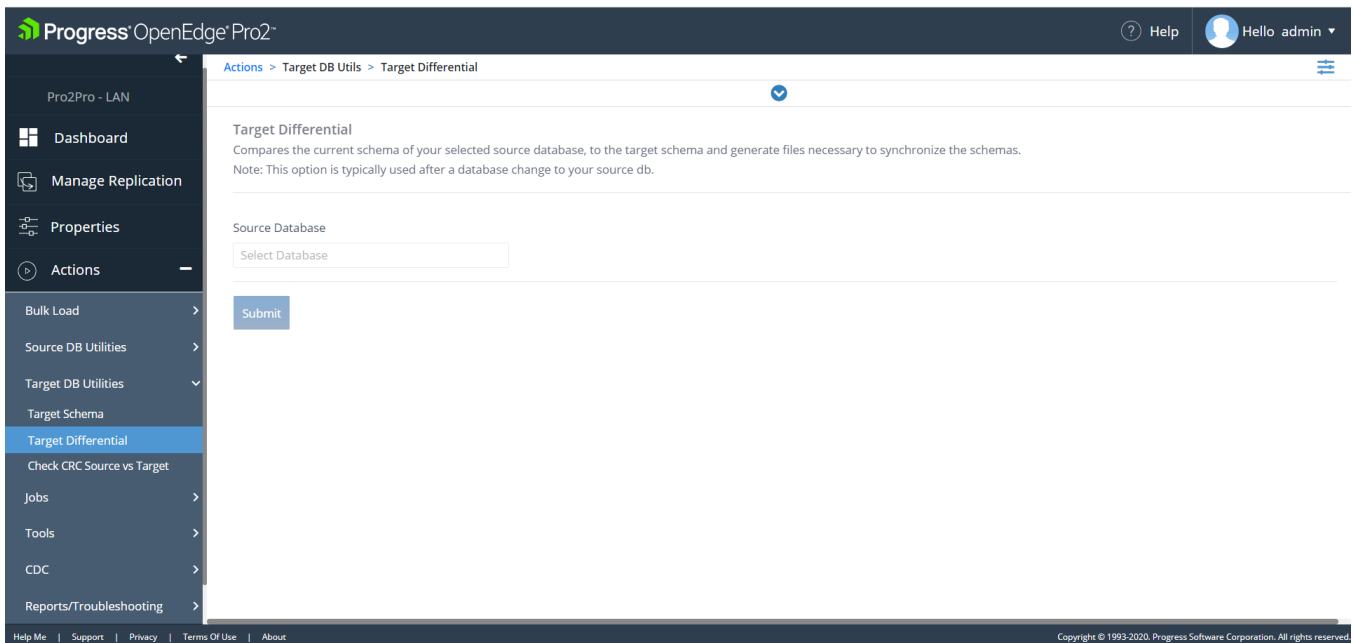
Target DB Utilities

The target database utilities are:

- **Target Schema:** The Target Schema menu item generates the necessary database script to build a copy of the Progress schema within the target database server. You are prompted for an output file name and then asked for the source database from which you wish to prepare a database Schema. The resulting DB file can then be executed within an empty database container to build a target database with the same schema as the source OpenEdge database.



- **Target Differential:** The Target Differential menu item generates the necessary queries to upgrade the current target schema to match to current source schema. Typically, this utility is used after a schema change to the source database. The resulting database file can then be used to upgrade the target schema to match the source schema.



Note: When any change is made to the schema of the target database you must rebuild the schema holder database before Pro2 will see the changes to the target schema.

Oracle specialization: Use the following properties to generate Oracle specific output:

- `ORACLE_USE_LOGICAL`: To convert logical field to number (1)
- `ORACLE_USE_SCALE`: To add precision and scale for decimal fields.

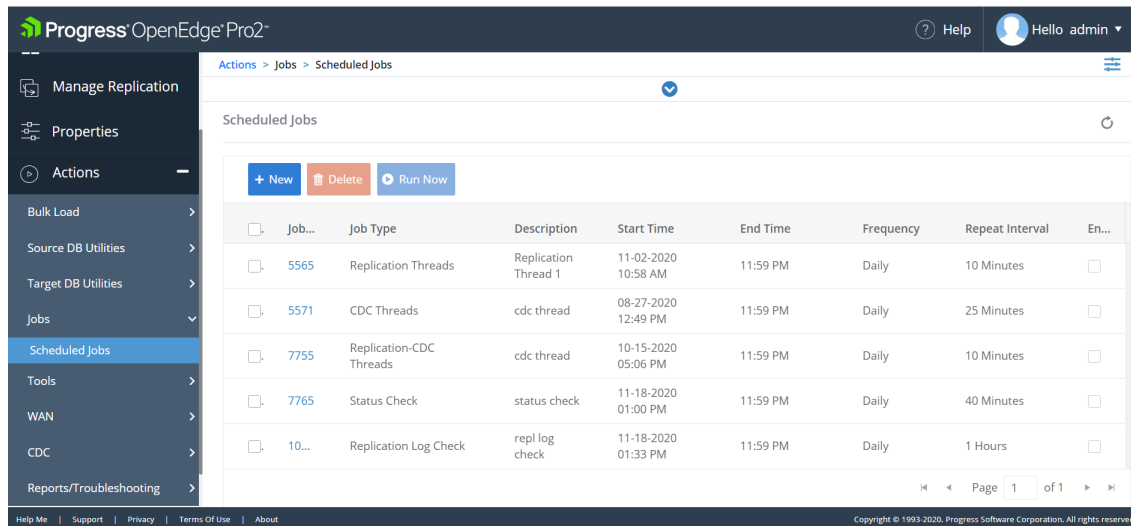
By default, the value of these properties is set to **NO**. To use these properties, set the value to **YES**. For Oracle specific targets, use the following properties. These properties are responsible for Add support for precision and scale.

- **Check CRC Source vs Target:** This option is used to validate that there are no differences between your two databases by generating a CRC report.

How to schedule jobs in Pro2

Pro2 is primarily used for replicating and synchronizing selected data and database objects from a source database to a target database. You can choose to manually perform your replications or schedule them to replicate automatically.

You can also schedule a variety of jobs by using the job scheduler. The job scheduler enables you to schedule recurring replications, CDC tasks, file purges, log checks, and more so that you no longer need to perform these tasks manually.



To schedule a job:

1. From the **Actions** tab of the side menu, click **Jobs > Scheduled jobs > New**.

2. Under the **Type** drop-down list, select the one of the following job types:

- Applied Queue Record Purge
- CDC Purge
- CDC Threads
- Enterprise Push
- File Purge
- Replication Log Check
- Replication Threads
- Replication-CDC Threads
- Status Check

3. Enter a **Description** for the job. A description helps you differentiate between similarly scheduled jobs.

4. If you are creating a new job, then ensure that the **Enable** check-box is selected. If you want to disable a scheduled job, but not remove a scheduled job, then you can do so by using this check-box.

5. In the **Settings** section, choose the job's start and end date as well as the start and end time.

You can schedule a job to occur once, daily, weekly, or monthly depending upon your business needs.

If the job is scheduled to repeat monthly, you can indicate the start date, end date, start time, end time, the month or months to repeat on, and number of days per month.

Note:

- In order to schedule a job once, select **One Time** from the **Settings** section and provide the start date and start time only. The default values for the **Start Date** and **Start Time** fields are the current date and time, respectively.
 - To schedule a job daily, weekly, or monthly, you need to provide the end date and end time, along with other parameters. By default, the **End Date** field is blank and the **End Time** is 11:59 PM, or 86340 seconds past midnight.
 - For the monthly frequency, if you leave the **Perform the job on day(s) of the selected months** field blank, then the job is scheduled for the day it is presently for each of the selected months.
-

6. Set the **Repeat task every field.**

Use this field to indicate when the job should be repeated. You can set the number of hours or minutes. This field is dependent on the repeat cadence. For example, if the job is scheduled to repeat daily, set this field to 0 to repeat the job once per day.

7. Click **Save.**

To run a job on demand:

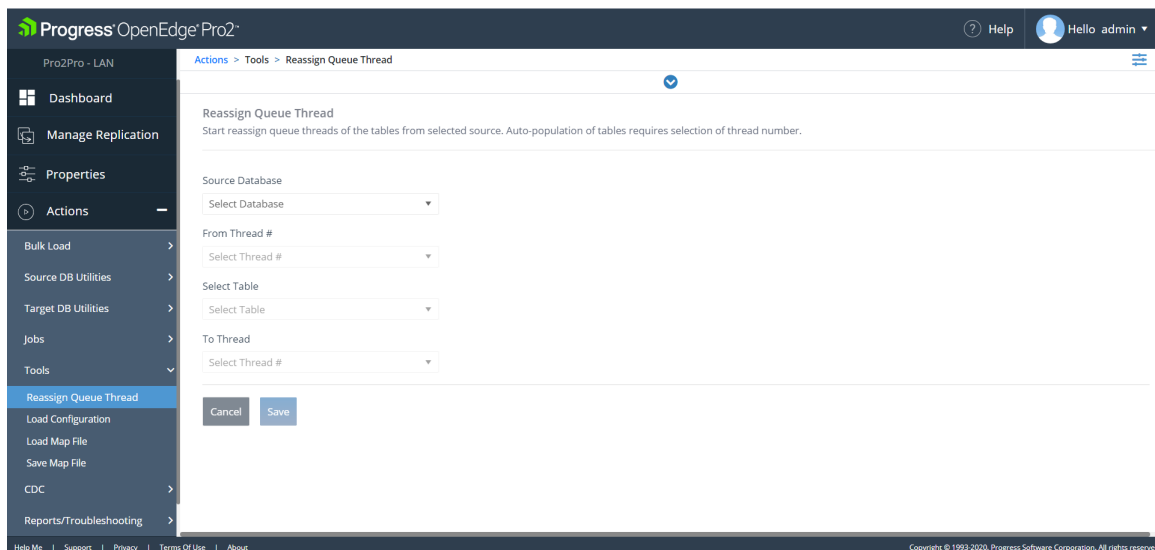
1. Select a previously created job by clicking its check-box.
2. Click **Run Now**.

Tools

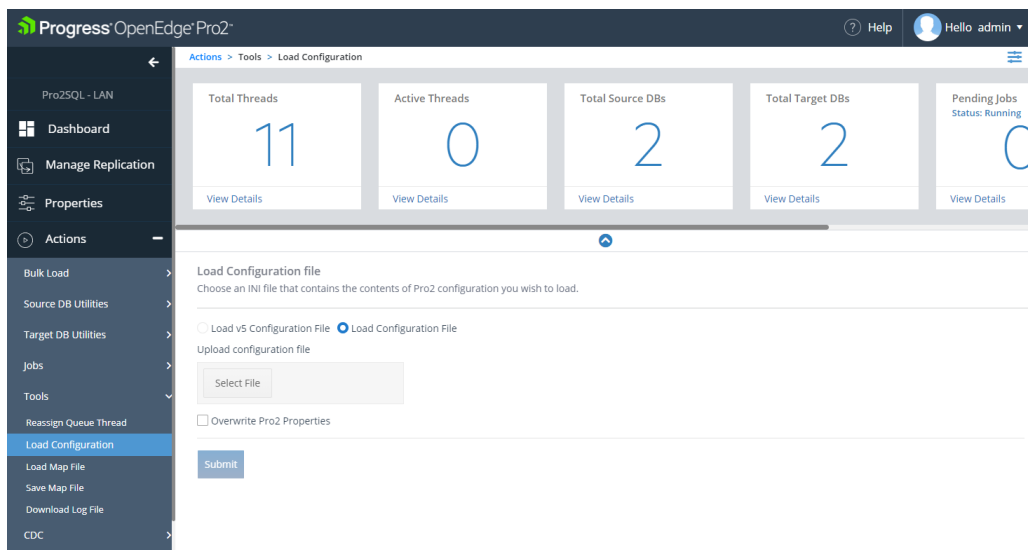
The tools tab consists of:

- **Reassign Queue Thread:** The Reassign Queue Thread menu item enables you to reassign the queue records from one thread to another. The Reassign Queue Thread functionality also allows you to choose the database and the table name as well.

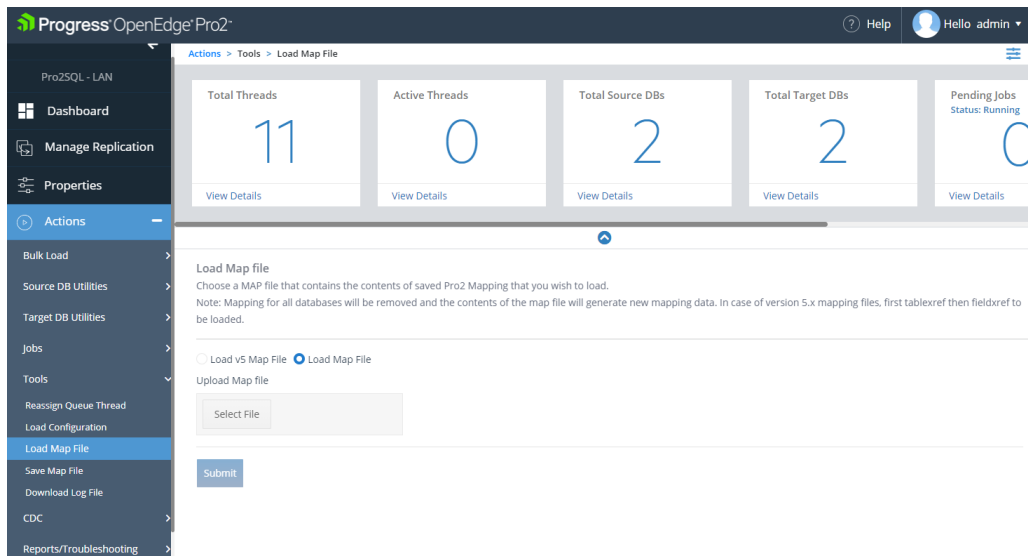
Note: The Reassign Queue Thread is not complete until all the input items are selected.



- **Load Configuration:** This menu item loads a list of replication control properties and values using a text file. You can specify the .ini file that has the Pro2 configuration settings that you want to load. To migrate from Pro2 v5.5 to v6.4 or above, select the **Load v5 Configuration File** radio button. To setup your current Pro2 installation, you can select the **Load Configuration File** radio button. The **Load Configuration File** is the default option.



- **Load Map File:** This menu item loads table and field mapping cross-reference information from a text file. You can specify the map file that has the Pro2 Mapping data that you want to load. To migrate from Pro2 v5.5 to v6.4 or above, select the **Load v5 Map File** radio button. To upgrade your current Pro2 v6.x installation, you can select the **Load Map File** radio button. The **Load Map File** is the default option.



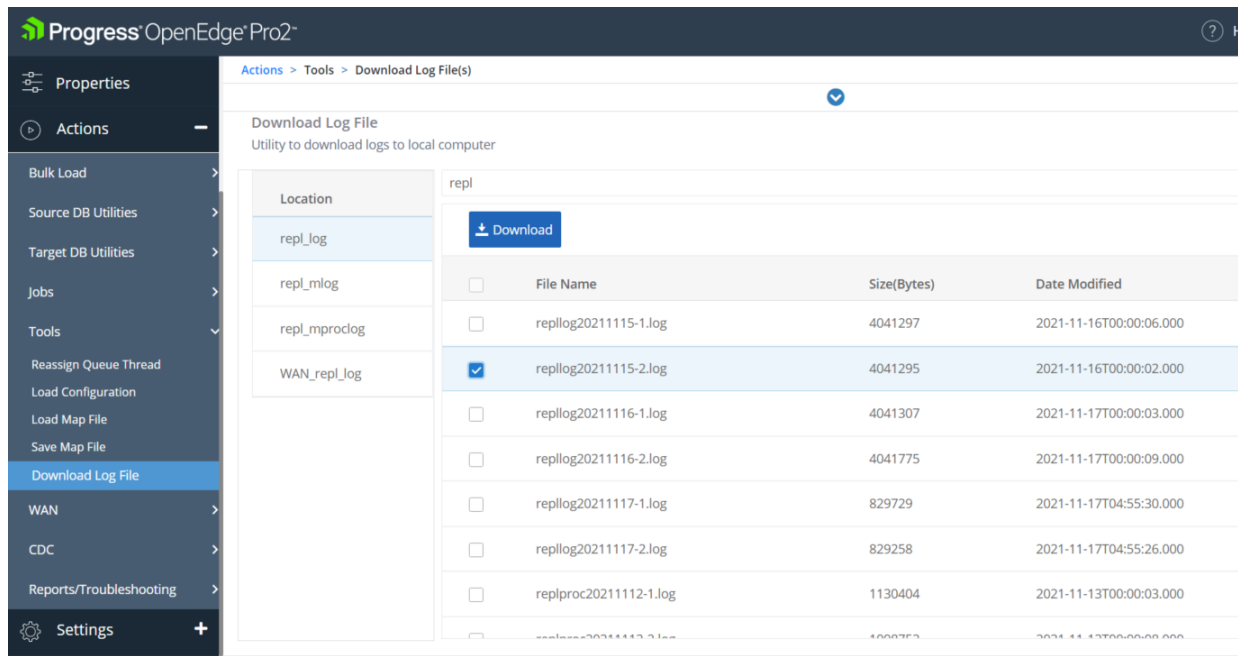
- **Save Map File** : Saves current table and field mapping cross-reference information to a text file.

Download logs

From the Tools menu, you can access the Pro2 Log Download page. The Pro2 Log Download page offers the ability to download log files for each replication type, giving you a complete view of the replication process. This enables you to remotely debug issues without the need to log in your Pro2 Instance machine.

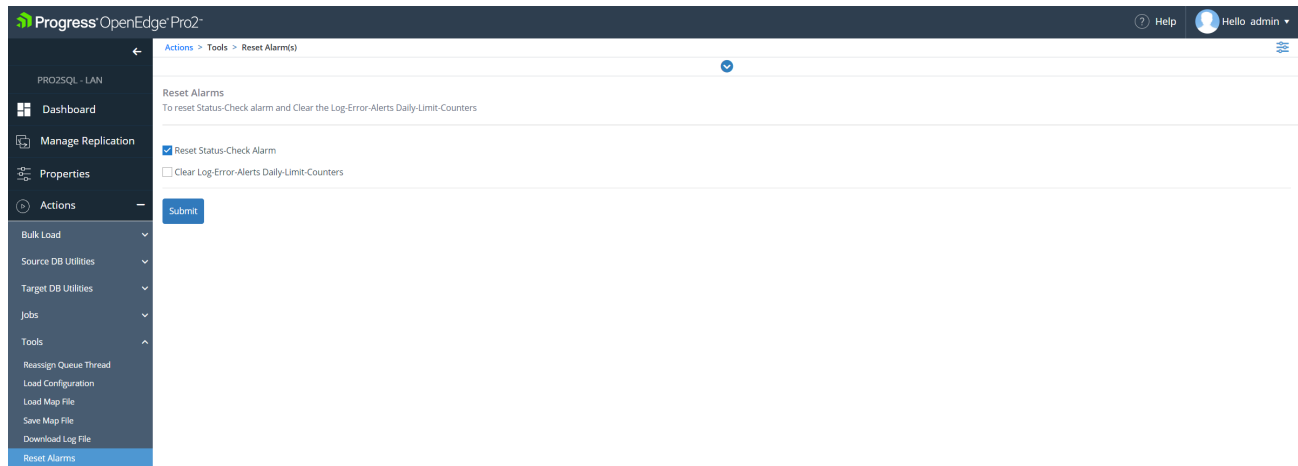
You can download:

- Replication logs
- Replication-CDC logs
- CDC logs
- Replication log-check logs
- Status check logs



Reset alarms

You can access the **Reset Alarms** page from the **Tools** menu.



The **Reset Alarm** utility offers the following options:

- **Reset Status-Check Alarm**
- **Clear Log-Error-Alerts Daily-Limit-Counters**

Reset Status-Check Alarm: By default, the `StatusCheck` procedure sends only one email notification per day to prevent an overload of email messages. After the email is received and replication is restarted, you can reset the status check alarm to send the alerts again, if replication unexpectedly stops on that same day.

You can reset the email notification by using the following procedure:

1. In the **Pro2** web user interface, navigate to the **Actions** tab, and select **Tools**.
2. Select **Reset Alarms**.

To reset the status check alarm, the following control records are deleted from the `repl_control` table:

```
/* Reset status check alarm */
{&Pro2DB}Repl_Control.GroupID = "ALARM"
{&Pro2DB}Repl_Control.CodeID = "LASTALARM"
{&Pro2DB}Repl_Control.CodeVal1 = ""
{&Pro2DB}Repl_Control.CodeVal2 = ""
```

Clear Log-Error-Alerts Daily-Limit-Counters: The `ReplLogCheck` procedure resets the value of the `LOG_ALERT_DAILY_LIMIT` property to **blank**. To set the daily limit on the replication log generation process again, you need to create the `LOG_ALERT_DAILY_LIMIT` property and assign it a value. The default value of this property is 10.

To know more about Pro2 properties and their default configurations, see [Properties](#) on page 21.

To reset the daily limit counters for the replication logs, the following control records are deleted from the `repl_control` table:

```
/* Reset Log Alerts Daily-Limit-Counters */
{&Pro2DB}Repl_Control.GroupID = "REPLICATION"
{&Pro2DB}Repl_Control.CodeID = "LOGALARM"
```

WAN

Note: This tab is enabled only when the **Source DB Connection** is set to **WAN**.

When the target database is across a wide area network (WAN) from the source database, the Pro2 Server must reside on the same local-area-network as the target database server.

When the target database is in a different geographic location than the source Progress database, implementation of Pro2 must include a Progress application server on the source database server in addition to the standard configuration on the Pro2 server. Instead of running the replication procedures via typical client/server mode connections to the source and repl databases, Pro2 will send requests to the application server which returns data to the calling procedure on the Pro2 server. The Pro2 server must be on the same local-area-network as the target database server.

If Pro2 is set up in a WAN environment, you need to check application server connection, synchronize repl table data, build application server processor libraries, transfer files to and from WAN server. Apart from this, you also need to get source side schema, delta DF file and check CRC local vs source.

Check application server connection

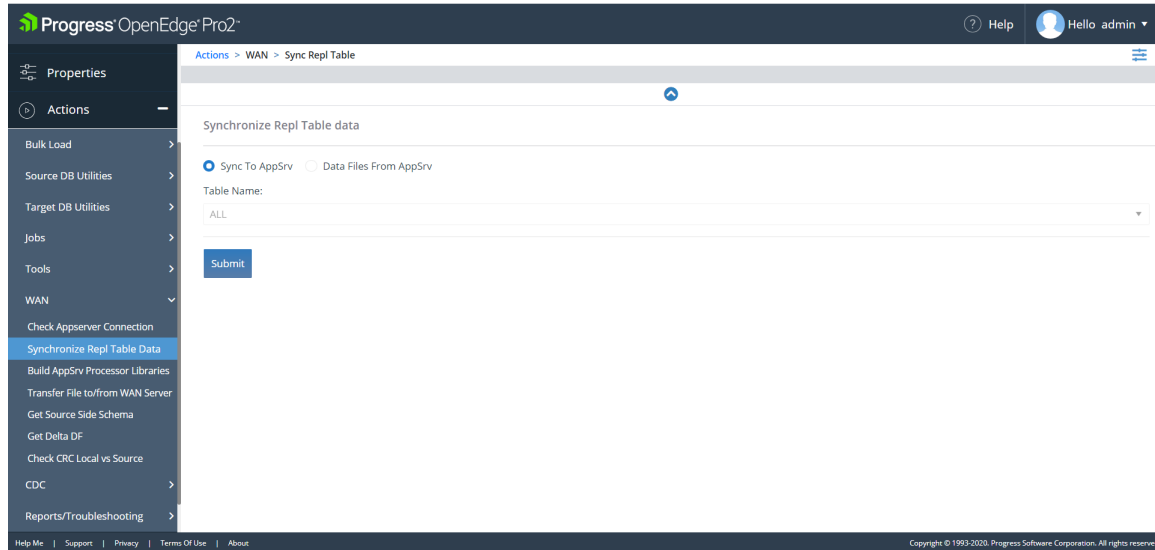
Once the `AS_Src` environments have been configured, the application server connection and communication should be tested and verified before attempting to start replication. You can do that by selecting **Submit**. Pro2 notifies you if the test was successful or if it failed.

Synchronize Repl Table Data

The **Synchronize Repl Table Data** functionality enables you to synchronize the repl tables data to application server and download the repl table data files from application server.

To synchronize repl table data, Click **Tools** → **WAN** → **Synchronize Repl Table Data** in the web UI. The Sync Repl Tables window will open. Here you can choose whether to sync the repl tables to application server by selecting **Sync to AppSrv** radio button, or you can choose to download the repl table data files from the application server using the **Data Files From AppSrv** radio button.

The **Sync to AppSrv** button enables you to synchronize the WAN repl database table values while excluding replqueue from your local copy. After performing a sync of WAN replication table data, Pro2 displays all the repl tables that are updated. It also shows the number of records that are updated, deleted, or locked per repl table. The Data Files From AppSrv button enables you to download the WAN repl database table contents while excluding replqueue to your local copy.



Note: If you choose ALL from the drop-down list (Table Name field), then the data files for all table files are created except for replqueue table.

Build AppSrv Processor Libraries

This menu item enables you to build application server process libraries for the selected databases. To build application server process libraries, select the **Build AppSrv Processor Libraries** from the WAN menu and select the databases that you want and click **Submit**.

Transfer File to/from WAN Server

You can send, receive or delete files from the WAN server using the **Transfer File to/from WAN Server** menu item. To perform these actions, select Transfer Files to/from WAN Server menu item. Choose the file that you want to perform the action for and click **Copy to WAN** or **Copy to Local** depending upon your choice. You can also choose to delete the files from WAN by clicking the **Delete** button in the WAN Directory section.

Get Source Side Schema

To get the schema of the source side databases, select Get Source Side Schema and select the databases that you want to include and click **Submit**.

Get Delta DF

You can use the **Get Delta DF** menu item to retrieve the Delta DF for a source database. To do this, Pro2 connects to a local empty copy of the source DB and generates a Delta DF from the DF of the source DB by removing all area information from the source DF and downloading it through PAS for OpenEdge.

To retrieve the Delta DF, from the WAN menu, click **Get Delta DF** and select the databases for which you want to retrieve the Delta DF and click **Submit**.

Check CRC Local vs Source

You can use the Check CRC Local vs. Source menu item to check whether the CRC for Mapped tables in the source databases and the local databases match. To do this, from the WAN menu, select **Check CRC Local vs. Source**.

CDC

Note: This tab is enabled only when the Source DB Mode is set to CDC.

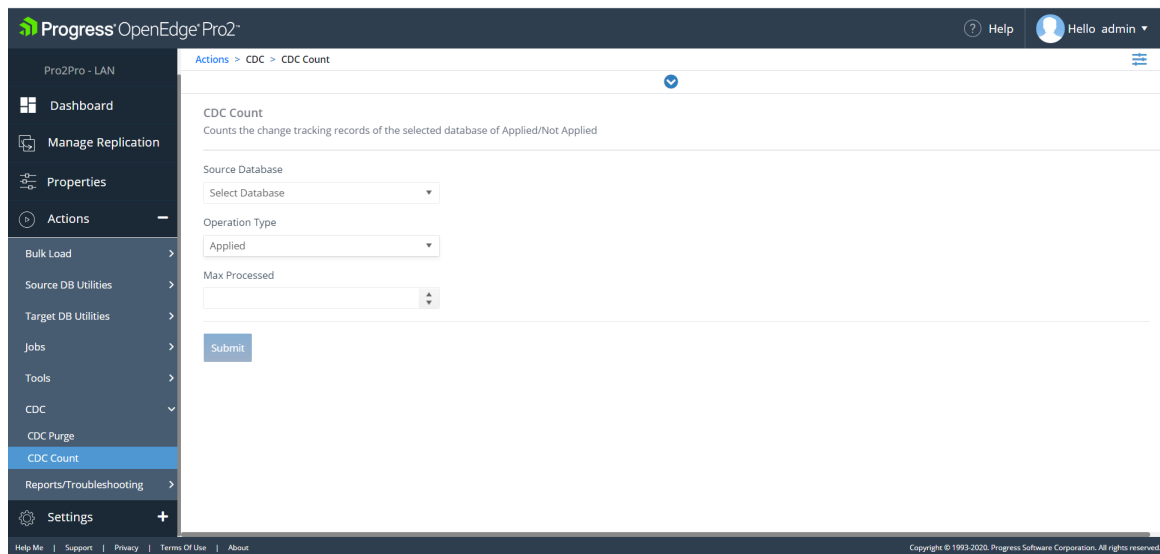
CDC Purge

The CDC Purge menu item can be used to delete the data of change records other than the old data based on user input about the number of days of data that should be deleted and the maximum number of records that should be deleted per batch. To delete change data records, from **CDC** → **CDC Purge**, select the source database and the number of days of data that you want to retain. Specify the number of records to be processed per batch and click Submit. If you set the Number of days to 0, all records are purged regardless of the date. If you set the Max Processed field to 0, all records are purged in one batch.

CDC Count

The CDC Count menu item can be used to get the actual record count of the CDC change tracking table based on whether the records are processed or not processed. This can be done by selecting the database from **Select Database** and then selecting **Operation Type** as **Applied/Not Applied** from **CDC** → **CDC Count**.

Users can get the processed records by selecting **Applied** from the Operation Type drop down. To get the list of records that are not processed, users should select **Not Applied** from the Operation Type drop down. By setting the Max Processed field, you can control the maximum number of records that are processed by the CDC Count function. If you set the Max Processed to 0, all records are counted in one batch.



Reports/Troubleshooting

The **Reports/Troubleshooting** menu option contains the following reports:

- Get Replqueue Count
- Get Replqueue List
- Get Table Configuration
- Get Table Activity Count

Get ReplQueue Count

The **Get ReplQueue Count** retrieves the replqueue count for a specific thread number or all the threads. Along with the thread number, you can specify the maximum number of replqueue records that you want to count for the threads.

Generate the **Get ReplQueue Count** report:

1. In the **Thread Number** field, enter the thread number for which replqueue count report is to be generated. The default value for this field is 0, which retrieves the replqueue count for all the threads.
2. In the **Max Record Count** field, enter the number of records to be included in the report. The default value for this field is 0, which retrieves the count for all the replqueue records.
3. Optional. Select the **Applied** checkbox to count only the applied replqueue records and the **Audited** checkbox to count only the audited replqueue records.
4. Click **Submit**.

The job scheduler runs the `Get ReplQueue Count` job request to generate the **Get ReplQueue Count** report.

5. After the job request runs successfully, download the `rqcounts.txt` file from the job record to view the report output.

The `rqcounts.txt` file gets saved in the user download directory.

The `rqcounts.txt` file contains the following fields: **Thread, Source DB, Source Table, # Records, Unapplied, Applied, Audited, No Map, No GenQRec, To be Replicated.**

Get Replqueue List

The **Get Replqueue List** report lists the replqueue records with the not applied status. The report organizes the records by thread number, source table, and the event date.

Generate the **Get Replqueue List** report:

1. In the **Replication Thread Number** field, specify the thread number to retrieve the respective replqueue records.
2. In the **Start Date** and **End Date** fields, set the date range for the records to be included in the report. The default value for the **End Date** field is the current date.

Note: The **Start Date** is an optional field. If you do not specify a start date, the report lists all the unprocessed replqueue records up to the entered end date.

3. In the **Max Record Count** field, enter the number of records to be included in the report. The default value for this field is 0, which retrieves the count of all the unprocessed replqueue records.

4. Click **Submit**.

The job scheduler runs the `Get Replqueue List` job request to generate the **Get Replqueue List** report.

5. After the job request runs successfully, download the `RQList.csv` file from the job record to view the report output.

The `RQList.csv` file gets saved in the user download directory.

The `RQList.csv` file contains the following fields: **Record ID**, **Sequence Number**, **Thread Number**, **Source Database**, **Source Table**, **Event Date**, **Event Type**, **Applied**.

Get Table Configuration

The **Get Table Configuration** report generates a list of tables and the threads to which the tables are mapped. The report includes the flag values (yes or no) for the **Gen Queue Records**, **Process Queue Records**, and the **Include/Exclude** properties of the tables.

Generate the **Get Table Configuration** report:

1. Select the **Source Database** from the list. The report runs for one database at a time.
2. In the **Table include list** field, specify the table name in case of a single table. If you want to run the report for multiple tables, enter a list of table names separated by commas. The default value for this field is * asterisk, which retrieves the records for all the mapped tables within the selected database.

3. Click **Submit**.

The job scheduler runs the `Get Table Configuration` job request to generate the **Get Table Configuration** report.

4. After the job request runs successfully, download the `TableConfigList.csv` file from the job record to view the report output.

The `TableConfigList.csv` file gets saved in the user download directory.

The `TableConfigList.csv` file contains the following fields: **Source Database**, **Source Table**, **Thread Number**, **Gen Queue Rec**, **Process Queue Rec**, **Include/Exclude**.

Get Table Activity Count

The **Get Table Activity Count** report contains the replqueue activity count for a specific table or multiple tables from the source database. The three activity event types are: create, update, and delete records.

Generate the **Get Table Activity Count** report:

1. Select the **Source Database** from the list. The report runs for one database at a time.
2. In the **Table include list** field, specify the table name in case of a single table for retrieving the replqueue activity count. If you want to run the report for multiple tables, enter a list of table names separated by commas.

Note: For this report, * asterisk is not supported in the **Table include list** field. Fetching activity records for all the tables within the specified source database may take a long time to run the report to completion.

3. Click **Submit**.

The **Alert** dialog box notifies the time it will take to run the report if the queue backlog is large.

4. Click **Ok** to proceed.

The job scheduler runs the `Get Table Activity` job request to generate the **Get Table Activity Count** report.

5. After the job request runs successfully, download the `TableactivityList.csv` file from the job record to view the report.

The `TableactivityList.csv` file gets saved in the user download directory.

The `TableactivityList.csv` file contains the following fields: **Source Database**, **Source Table**, **Create Record**, **Update Record**, **Delete Record**.

Settings

Use the **Settings** tab of the modifications and the enhancements that can be done to the replication thread. The tab consists of the following functionalities: Pro2 Enterprise View and LAN/WAN.

For details, see the following topics:

- [Pro2 Enterprise View](#)
- [LAN/WAN](#)
- [Processor Stop](#)

Pro2 Enterprise View

The Pro2 Enterprise View is a centralized dashboard that aggregates mission-critical data, reports, and metrics across all of your Pro2 instances. From the Enterprise View dashboard, you can quickly access each of your Pro2 instances to manage and act on any reports or replications from the instance.

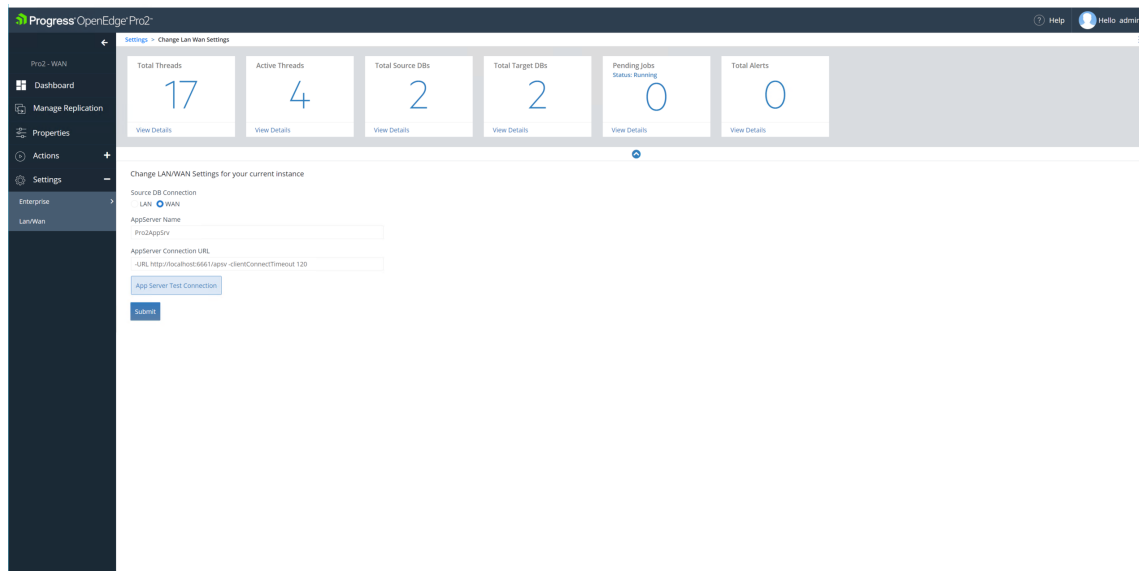
For example, an e-commerce enterprise has three instances of OpenEdge Pro2 running on three separate data centers: one in the USA, one in India, and one in the UK. Managing data and replications across these three instances can be a hassle. With the Enterprise View, you can observe and navigate to each Pro2 instance with ease and efficiency.

The Pro2 Enterprise View dashboard is included with the latest release of Pro2, but it can also be installed as an add-on to your existing instances. If you are interested in the Pro2 Enterprise View Dashboard, contact your Progress OpenEdge representative, and ask about upgrading to the latest version, or about adding the Pro2 Enterprise View Dashboard to your existing instances.

For more information on Pro2 Enterprise View, see [Pro2 FAQs](#).

LAN/WAN

On the **LAN/WAN** page, you can adjust the settings of your LAN or WAN configuration. For example, you can update the application server name of your deployment.



The following is a description of the fields on the page:

- **Source database connection:** use this field to select whether your deployment is LAN or WAN.
- **Application server name:** in this field you define the name of the application server that you want to associate with your deployment.
- **Application server connection URL:** enter your application server URL to connect your it to your Pro2 deployment.
- **Application server test connection:** after you have filled out all of the fields, use this button to test your Pro2 deployment's connection to your the application server.

Processor Stop

On the **Processor Stop** page, you can schedule to stop the running replication threads after you have scheduled to start the threads using the job scheduler. The **Pro2** processor stops the replication threads automatically at the specified frequency. The available scheduling frequencies are: **NEVER**, **WEEKLY**, and **DAILY**.

- **NEVER**— Keeps the replication threads running indefinitely. It is the default scheduling frequency.
- **WEEKLY**— Allows you to specify the day of the week and the time when the processor stops the running replication threads.
- **DAILY**— Allows you to specify the time of the day when the processor stops the running replication threads.

Pro2 - WAN

- Dashboard
- Manage Replication
- Properties
- Actions
- Settings
- Enterprise
- Lan/Wan
- Processor Stop

Total Threads
11
[View Details](#)

Active Threads
5
[View Details](#)

How often do you want to stop the processor?
This page is scheduled to stop the running Replication thread.

☐ NEVER ☐ WEEKLY ☒ DAILY

at:
18:33

[Submit](#)

You can view the status of the replication threads on the **Thread Control** display of the **Dashboard** page.

For more information on scheduling the replication threads to start running, see [How to schedule jobs in Pro2](#) on page 36.

