

EMMS DATA INTERCHANGE GUIDE

VERSION: 1.05
DOCUMENT REF: ELECMARKDEV-9-83
PREPARED BY: Information Management and Technology (IMT)
DATE: 16 May 2011
Final

Disclaimer

This document is made available to you on the following basis:

1. Purpose – This EMMS Data Interchange Guide (Guide) has been produced by the Australian Energy Market Operator Limited (AEMO) to provide information about the **Data Interchange**, as at the date of publication.
2. No substitute – This Guide is not a substitute for, and should not be read in lieu of, the National Electricity Law (NEL), the National Electricity Rules (Rules) or any other relevant laws, codes, rules, procedures or policies. Further, the contents of this Guide do not constitute legal or business advice and should not be relied on as a substitute for obtaining detailed advice about the NEL, the Rules, or any other relevant laws, codes, rules, procedures or policies, or any aspect of the national electricity market or the electricity industry.
3. No Warranty – While AEMO has used due care and skill in the production of this Guide, neither AEMO, nor any of its employees, agents and consultants make any representation or warranty as to the accuracy, reliability, completeness or suitability for particular purposes of the information in this Guide.
4. Limitation of liability - To the extent permitted by law, AEMO and its advisers, consultants and other contributors to this Guide (or their respective associated companies, businesses, partners, directors, officers or employees) shall not be liable for any errors, omissions, defects or misrepresentations in the information contained in this Guide, or for any loss or damage suffered by persons who use or rely on such information (including by reason of negligence, negligent misstatement or otherwise). If any law prohibits the exclusion of such liability, AEMO's liability is limited, at AEMO's option, to the re-supply of the information, provided that this limitation is permitted by law and is fair and reasonable.

Copyright

Copyright © 2011 Australian Energy Market Operator Limited. All rights reserved.

Trademark Notices

- Microsoft, Windows and SQL Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.
- Oracle and Java are registered trademarks of Oracle and/or its affiliates.

Distribution

Commercial-in-confidence with NEM participants

This Document Identification

Title: EMMS Data Interchange Guide

Version: 1.05

Document ID: ELECMARKDEV-9-83

Responsible Department: Information Management and Technology (IMT)

Notes: Explains DI product set without installation packages.

Documents made obsolete: The release of this document changes only the version of the EMMS Data Interchange Guide.

Contents

1	Introduction	6
1.1	Purpose.....	6
1.2	Audience	6
1.3	Scope.....	6
1.3.1	What's in this guide	6
1.4	Conventions	7
2	Context.....	8
2.1	What Data Interchange is for	8
2.2	How do you use Data Interchange	8
2.3	Who can use Data Interchange	8
2.4	Data Interchange requirements	8
2.5	Data Interchange products and resources.....	9
2.5.1	Data Interchange product set.....	9
2.5.2	Data Interchange documentation set.....	10
3	Components.....	12
3.1	EMMS data interfaces	12
3.1.1	EMMS Data Subscription Services	12
3.1.2	EMMS file server	13
3.2	Data transport protocol.....	14
3.3	Participant Data Replication (PDR)	14
3.3.1	Participant Data Replication Batcher (pdrBatcher)	15
3.3.2	Participant Data Replication Loader (pdrLoader).....	15
3.4	Participant's RDBMS.....	15
3.4.1	MMS Data Model.....	16
3.4.2	Current EMMS baseline data	16
3.4.3	Historical data.....	16
3.5	Replication Manager	16
4	Implementation.....	17
4.1	Minimising potential implementation problems	17
4.2	Production and pre-production data sources	17
4.3	Hardware specifications	18
4.4	Data volumes	18
4.5	Supported database platforms	18
4.6	Typical installation locations	18
4.7	Before starting your DI installation.....	19
4.8	Setting-up a standard DI environment.....	19
5	Considerations	22

5.1	DI Scenarios.....	23
5.1.1	pdrBatcher and pdrLoader to a single database.....	23
5.1.2	pdrBatcher and pdrLoader to a single database, with file recovery.....	23
5.1.3	pdrBatcher to multiple directories and parallel pdrLoaders to multiple databases, without file recovery.....	24
5.1.4	pdrBatcher to multiple directories and parallel pdrLoaders to multiple databases, with recovery.....	25
5.1.5	Serial feed to multiple databases.....	26
5.1.6	Summary of example configurations.....	27
5.2	Bespoke data interchange systems.....	28
5.2.1	Providing a local replica using pdrBatcher.....	28
5.3	Data Sharing.....	28
5.3.1	Outcome of sharing data.....	28
5.3.2	Setting up data sharing.....	29
5.3.3	Removing data sharing.....	29
5.4	Hot recovery.....	30
5.4.1	Failure of database or pdrLoader.....	30
5.4.2	Failure of a stand-alone pdrBatcher.....	30
5.4.3	Failure of server with pdrBatcher, pdrLoader and database.....	30
5.4.4	Failure of the network connection.....	30
5.5	Outline of the file recovery strategy.....	31
5.5.1	File synchronisation.....	31
5.5.2	Trickle feed and normal CSV feed.....	31
5.5.3	AEMO data retention.....	31
6	Maintenance.....	32
6.1	File data feed.....	32
6.2	Monitoring.....	32
6.2.1	Database state.....	32
6.2.2	Operational checks.....	32
6.3	Future upgrades.....	33
7	Glossary.....	34
7.1	Abbreviations.....	34
7.2	Special terms.....	34
8	References.....	35
8.1	AEMO's website.....	35
8.2	EITS Publications.....	35
8.3	Information centre.....	35
9	Needing Help?.....	36
9.1	Check system status.....	36
9.2	Confirm no issues with organisation's internal network.....	36
9.3	Confirm no issues with external service provider.....	36
9.4	Java errors.....	36
9.5	Not connecting or locked out?.....	36
9.6	Files not transferring, or timing out?.....	36

9.7	Missing history	37
9.8	Getting file, but missing report	37
9.9	VPN down	37
9.10	Observing abnormal behaviour?.....	37
9.11	Logging a call with AEMO Help Desk for support	37

1 Introduction

This document describes Data Interchange (DI). DI is a set of cooperating applications to replicate data between AEMO's Electricity Market Management System (EMMS) and a participant's RDBMS conforming to the MMS Data Model, including management and monitoring.

AEMO provides working applications to participants as assistance with the replication of data between AEMO and each participant. Each participant is responsible for satisfying their own requirements for data, including deciding if the higher IT complexity for using DI is a better business decision than using the simpler graphical user interface provided by the EMMS Web Portal.

Successfully implementing and managing Data Interchange requires good IT skills, including networking, database management, batch file management and disaster recovery.

1.1 Purpose

This document provides an overview of:

- The components for managing, monitoring and replicating data between AEMO's Electricity Wholesale Market Management Systems (EMMS) and a participant's RDBMS conforming to MMS Data Model.
- The AEMO-supported replication mechanism applicable to EMMS, being Participant Data Replication (PDR).
- Subscribing to a data feed from AEMO, including files conforming to the MMS Data Model.

1.2 Audience

This document is relevant to:

- Implementers of interfaces to AEMO's systems.
- Maintainers of interfaces to AEMO's systems.
- Parties with responsibility for using interfaces to AEMO's systems.

1.3 Scope

1.3.1 What's in this guide

This guide includes:

- An explanation of the context of Data Interchange.
- A summary of the components comprising Data Interchange.
- Options to consider before attempting you Data Interchange implementation.
- Set-up instructions for careful reference when performing your first Data Interchange implementation.
- Advice on maintaining your Data Interchange implementation.
- A glossary and reference section.
- Advice to help you resolve a problem before contacting the AEMO Help Desk.

For further information on individual Data Interchange components or detailed installation instructions, see the relevant Data Interchange product user guide (see "Data Interchange documentation set" on page 10).

1.4 Conventions



Important Note: important information is in this style.



Note: additional information is in this style.

2 Context

2.1 What Data Interchange is for

DI manages, monitors, and replicates data between AEMO's Electricity Market Management System (EMMS) and a participant's database conforming to the MMS Data Model.

The core elements of DI are:

- AEMO applications generate compressed, structured .CSV files into the EMMS file server, according to the selections made by each participant from an extensive list.
- Participant runs software to download the compressed data from the EMMS file server across a network connection.
- Participant runs software to load this data into one or more local databases at the participant's site, with optional monitoring.
- Participant manages each local database, and the data feed.

2.2 How do you use Data Interchange

Using DI starts with defining a purpose-designed architecture. AEMO provides some examples in this document. It can be as simple as a single data feed to a single database (plus a similar setup for testing with AEMO's pre-production environment). It can be quite sophisticated, such as multiple data feeds from shared Participant IDs to multiple databases (primary receiver, warehouse, settlements, dispatch, trading, etc.) and numerous variations for testing with AEMO's pre-production environment.

AEMO provides individually-configurable applications for both the downloading and the database loading functions (pdrBatcher and pdrLoader, respectively). This document addresses how these can be made to work together, plus how one database (or more) can receive the feed of data.

2.3 Who can use Data Interchange

Data Interchange is useful for:

- NEM participants wishing to replicate data between AEMO's Electricity Wholesale Market Management Systems (EMMS) and their RDBMS conforming to MMS Data Model.
- NEM participants sharing data between participant IDs.

To implement Data Interchange, a participant organisation requires IT staff with knowledge of:

- The Java application environment.
- The organisation's operating system and RDBMS.
- Data flow architecture, with disaster recovery.
- How the NEM systems operate.
- The organisation's business needs for NEM data.

2.4 Data Interchange requirements

To use Data Interchange requires:

- Network connection to AEMO's MarketNet (a private network) (see "[Guide to Market Systems - Gaining Access](#)" on AEMO's web site).
- Access credentials to at least one of the EMMS file servers (production and pre-production) (see welcome e-mail sent to the participant's registered Security Contacts).

- RDBMS to support the MMS Data Model and to suit the hardware platform on which the RDBMS runs (see heading “MMS” on the secure [“EITS Publications”](#) web page). In particular, the MMS Data Model is published in four supported physical models, being two version of Oracle and two versions of Microsoft SQL Server.
- Computer hardware to suit the participant-designed architecture.
- Installation of the Participant Data Replication software (see heading “Participant Data Replication (PDR)” on the secure [“EITS Publications”](#) web page). In particular, PDR requires:
 - Java 1.4 runtime engine.
 - JDBC driver and handler suitable for the target database. The distribution file from AEMO contains JDBC drivers for 16-bit versions of Oracle and SQL Server.
 - Either Windows or Unix operating system.
- For the Replication Manager:
 - Windows operating systems only.
- For accessing EMMS Web Portal (choosing files to receive):
 - Microsoft Internet Explorer version 6 or later, although the recommended version is Microsoft Internet Explorer 7. Participants using Microsoft Internet Explorer 8 can enable Compatibility View.
 - A monitor capable of 1024 x 768 screen resolution.

2.5 Data Interchange products and resources

All Data Interchange software packages and documentation are available from the secure [EITS Publications](#) area on AEMO’s website. The Data Model software and documentation are also available under the Data Subscriptions menu in the EMMS Web Portal (see “MMS Data Subscription Services User Guide” on the secure [EITS Publications](#) web page).

2.5.1 Data Interchange product set

The product set available from AEMO for installation on participant systems, as part of the implementation of the full functionality of Data Interchange are:

Name	Description
MMS Data Model	Sets-up the target tables in your RDBMS, conforming to the MMS Data Model.
MMS Data Model pdrLoader configuration	Maps the .CSV file content to your RDBMS conforming to the MMS Data Model
Participant Data Replication Batchter (pdrBatcher)	Obtains the data from the EMMS file server to feed one or more pdrLoader input folders (created during the pdrLoader set-up), and to send participant files to AEMO (e.g. bids).
Participant Data Replication Loader (pdrLoader)	Loads the data from AEMO into your RDBMS conforming to the MMS Data Model.
Replication Manager	A software application that manages the operation of the pdrLoader using a graphical user interface.

The required web application available from AEMO, to be used by participants as part of the implementation of the full functionality of Data Interchange is:

Name	Description
EMMS Data Subscription Services	<p>The Data Subscription web application in the EMMS Web Portal, allowing registered participants to:</p> <ul style="list-style-type: none"> ▪ Obtain the latest MMS Data Model scripts for building and maintaining their RDBMS conforming to the MMS Data Model. ▪ Subscribe to the report files made available in the EMMS file server. ▪ Unsubscribe from any files that are no longer required. ▪ View the entire list of available files and their details, whether or not your organisation is currently subscribed to them. <p>The recommendation is to subscribe to all available files and selectively load them into your RDBMS as required.</p>

2.5.2 Data Interchange documentation set

Each product is supplied with detailed user guides describing installation and configuration options for the DI product set. The DI documentation set is:

Name	Description
EMMS Data Interchange Guide (this document)	Provides an overview of the components for managing, monitoring and replicating data between AEMO's EMMS and a participant's RDBMS conforming to the MMS Data Model.
EMMS Data Subscription Services User Interface Guide	<p>Describes how to:</p> <ul style="list-style-type: none"> ▪ Download scripts to create or upgrade databases based on the current version of the MMS Data Model. ▪ Download the MMS Data Model Loader Configuration files to set up or upgrade the pdrLoader configuration. ▪ Manage subscriptions to the files required to populate the MMS Data Model and other files of EMMS-related data provided by AEMO that are not part of the MMS Data Model.
MMS Data Model Implementation Note	Available in each MMS Data Model script .ZIP file, contains detailed instructions on the set-up of a database based on the MMS Data Model.
MMS Data Model Loader Configuration Implementation Note	Available in the MMS Data Model Loader Configuration .ZIP file, contains instructions for installing pdrLoader configuration files.
MMS Data Model Package Summary	<p>Available from:</p> <p>http://www.aemo.com.au/data/market_data.html#data_model and in each MMS Data model script .ZIP file, summarises the MMS Data Model report. For each package, it lists the tables in the package, together with a brief summary, and entity diagrams.</p>

Name	Description
MMS Data Model Report	Available from: http://www.aemo.com.au/data/market_data.html#data_model and in each MMS Data model script .ZIP file. Contains a detailed description on the structure and content of the MMS Data Model. It contains: <ul style="list-style-type: none"> ▪ A summary list of the tables that make up the package, together with a brief description. ▪ Entity diagrams, showing the relationship between the tables in the package. ▪ For each table, information about the source of the data, its visibility (whether the data is public or private), primary key columns, index columns, and a detailed description of each data item in the table.
MMS Data Model Upgrade Report	Available in each MMS Data model script .ZIP file, contains a full description of any tables that have changed in that version from the immediately preceding version.
Participant Data Replication Batcher User Guide	Covers the set-up and use of the Participant Data Replication Batcher.
Participant Data Replication Loader User Guide	Contains detailed instructions on options for set-up and configuration of the pdrLoader application.
Replication Manager Installation Guide	Contains general installation instructions, plus a detailed description of the user interface, for the Replication Manager product.

3 Components

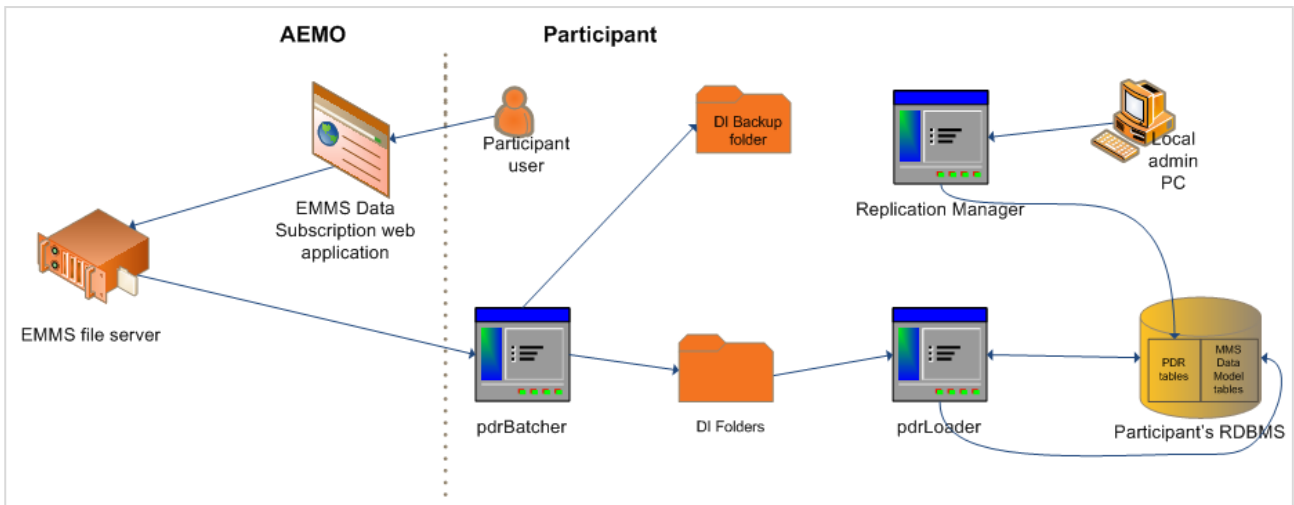


Diagram 1: Data Interchange components

AEMO's production and pre-production systems are totally independently operated, so the set of components shown in the above diagram are duplicated for production and pre-production.

The table below briefly describes the components of the EMMS Data Interchange process.

Replication Component	Description
EMMS Data Subscription Services	EMMS web-based application to manage AEMO's generation of data files.
EMMS file server	Folders (or directories) at AEMO containing files for each participant.
pdrBatcher	The software component responsible for retrieving data from EMMS data interface to local environment.
pdrLoader	The software component responsible for loading the data to the participant's RDBMS.
Participant DI Folders	The folders set-up as part of your Data Interchange architecture in your local store. The DI backup folder is optional, but highly recommended. In the event of data loss, files can be reloaded to your participant's RDBMS, by copying from the DI backup folder.
Participant's RDBMS	The NEM participant's local RDBMS for storing EMMS data of specific interest to that participant. The target data model for AEMO's Data Interchange is the MMS Data Model.
Replication Manager	The software component responsible for monitoring and managing the pdrLoader component, if a GUI interface is preferred to direct handling of the pdrLoader tables.

3.1 EMMS data interfaces

3.1.1 EMMS Data Subscription Services

EMMS Data Subscription Services is an EMMS web-based application for participants to:

- Obtain the MMS Data Model scripts for building and then subsequently maintaining the participant's RDBMS conforming to the MMS Data Model.
- Subscribe to the EMMS NEM report files containing the data needed by the participant, which AEMO then makes available in the EMMS file server.

- Unsubscribe from any files that are no longer required.

EMMS Data Subscription Services provides access to the files you need to upgrade from the immediate previous version of the MMS Data Model to the current version. If you are upgrading from a version prior to the immediate past version, consider starting with the latest version of the MMS Data Model and back-populating historical data using database management tools.

The EMMS Data Subscription Services application also has tools allowing participants to:

- View a catalogue of all available files.
- Easily identify the files to which they are currently subscribed.
- For some file types, download sample files.

AEMO also makes available a small number of additional files that are not supported by the MMS Data Model. That is, the data model does not define tables for storing the data provided in these additional files. The EMMS Data Subscription Services application also allows participants to subscribe and unsubscribe to these additional files.

Participants can also locally activate or deactivate the loading of selected .CSV files into the local MMS Data Model database. This local activation can be achieved by either (or both):

- Replication Manager, a Windows-based application, which is run locally at their own site.
- Direct manipulation of the data in the pdrLoader tables



Important Note: it's important to note that local activation and deactivation is a quite different process from the subscribing and unsubscribing to file delivery through the EMMS Data Subscriptions Services. Subscribing to a file through the EMMS Data Subscriptions Services causes the file to be made available on the EMMS file server, from where the file can be transferred to a local directory, or more. However, for it to be loaded into the local database it must also be activated using the locally-based tool. Similarly, deactivating the loading of files into the database does not stop the delivery of the files; they will simply not be loaded into the instance of the database. To stop a file being created by AEMO, unsubscribe using the EMMS Data Subscriptions Services.

The AEMO systems keep an online archive of all recently generated files to which a participant is subscribed. The archive is needed in case the network fails to deliver a usable file. The archive can be also a key resource when a receiving server collapses, destroying recently received files (although backing up all received files onto a separate system is a safer option for participants).

If a participant locally deactivates a file load without unsubscribing, then that file continues to be maintained in the AEMO archive and can be retrieved (and loaded into the local database) at a later date if the participant subsequently decides they require this data. If, however, a participant unsubscribes a file, then it is not possible to retrieve the files that would have been generated historically but weren't because they weren't subscribed to. File deactivation rather than unsubscribing therefore gives greater flexibility in terms of access to historical data by a participant.

AEMO has sized hardware to retain approximately 6 months worth of files online for all participants.

AEMO's production and pre-production systems are totally independently operated, although the pre-production system does occasionally get refreshed based on the production system.



Note: to learn more, see "EMMS Data Subscription Services User Interface Guide" (see "Data Interchange documentation set", on page 10)

3.1.2 EMMS file server

The EMMS file server is the publishing point from AEMO systems to participant systems. Each participant is responsible for interfacing with the EMMS file server. AEMO currently supports one data interface for EMMS data replication, called Data Interchange.

The EMMS file server is a standard file server at AEMO, with each participant allocated an account and access to common areas. The EMMS file server has been also called the NEM Participant File Server, abbreviated in various places as NEMPART or just Participant Server.

Each participant has access to a private area on the EMMS file server to which AEMO delivers files when data becomes available. Files are generally delivered as a zipped file with contents specific to single event or business process result. The EMMS file server is not considered to be a repository, so data retentions are kept down to maintain acceptable performance, being aware of the shared nature of the server across all participants. The intention of the EMMS file server is for use by Data Interchange.

AEMO only keeps a copy of reports that each participant was subscribed to at the time that a report was generated, and then only for a limited time.

AEMO's production and pre-production systems are totally independently operated, so each environment has its own IP address for its EMMS file server.

3.2 Data transport protocol

Communication with the EMMS file server uses the FTP protocol. All participants must connect to the EMMS file server through AEMO's private network (MarketNet). The pdrBatcher product has the default of using the FTP protocol, thus simplifying the setting up to access the EMMS file server.

AEMO has delegated the responsibility of the security for FTP access set-up to the IT Security contact nominated during the participant registration. To enable (or determine) FTP access using participant ID(s), participants are advised to get in touch with the nominated IT Security contact in their respective organisation.

3.3 Participant Data Replication (PDR)

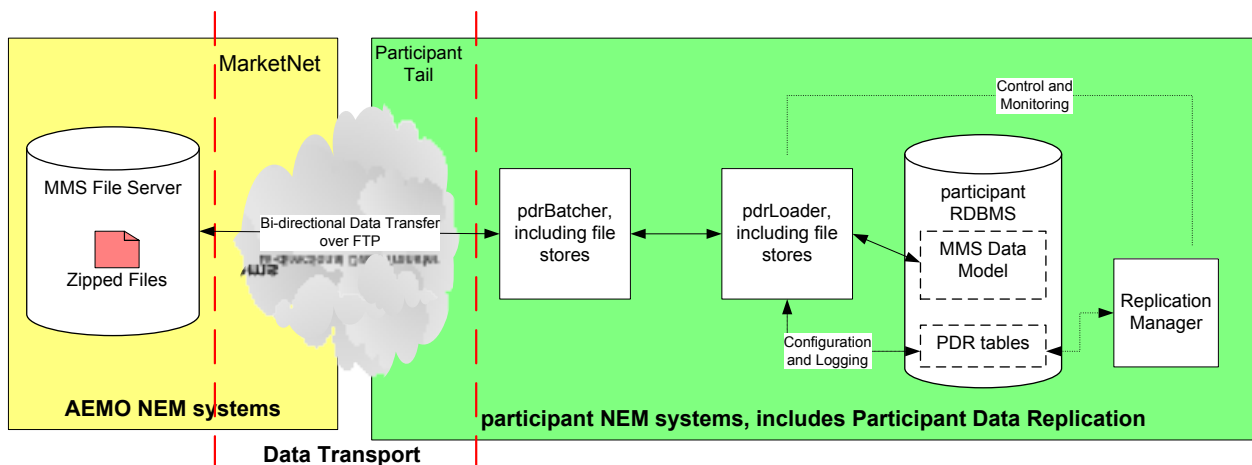


Diagram 2: EMMS Participant Data Replication overview

The two components of Participant Data Replication (PDR) are the batcher application (pdrBatcher) and the loader application (pdrLoader). PDR is built to replicate data from the zipped files available at the EMMS file server to a participant's RDBMS.

- The Participant Data Replication batcher (pdrBatcher) is responsible for moving files between remote and participant DI folders, in one direction or both, using the FTP protocol.
- The Participant Data Replication Loader (pdrLoader) parses specifically-designed .CSV report files, such as NEM report files, into a local database instance implementing the MMS Data Model.

Each of the two PDR applications is highly configurable. The PDR software is written in Java (a platform independent language), so can be installed in any environment supporting Java (for example, Unix, Linux and Microsoft Windows).

The pdrLoader uses a JDBC connection to access the participant's RDBMS and supports the RDBMS systems Oracle and SQL Server (with supported versions as shown in the release details for Participant Data Replication).

PDR is standalone, running outside the MMS Data Model; it uses its own tables on the participant's RDBMS and updates tables in the MMS Data Model.

PDR allows recovery from various failures, including network failure (allowing recovery of missing data), hardware failure (requiring failover to a possibly out-of-synchronisation pdrLoader and RDBMS), and system failure (causing build-up of data at various points). Recovery requires a message facility from the participant system back to the data source (at AEMO), implemented entirely with .CSV files. Use of the recovery features is optional.

3.3.1 Participant Data Replication Batcher (pdrBatcher)

The pdrBatcher transfers files from the EMMS file server to the participant DI folders, or vice versa. The pdrBatcher transfers files using the FTP protocol.

The application design of the pdrBatcher is a state machine with the file locations controlling the state. This means that, on restarting after a stop for any reason, the application automatically recovers by using the stored state in the local directories. Each of the participant DI folders contains the results of the file transfers, and a process in the application periodically moves files to the appropriate DI folder.

The design of pdrBatcher assumes the access to the participant DI folder is highly reliable and efficient, that is, with the DI folders hosted on the machine running the application. If space is a problem, consider using an ancillary process to empty the various local subdirectories by move or delete.

The pdrBatcher is a batch application and hence does not include a graphical user interface. All configuration of the pdrBatcher is done via the `.properties` file. The `.properties` file is only read when the pdrBatcher starts up, so any changes require the application to be restarted.

3.3.2 Participant Data Replication Loader (pdrLoader)

The pdrLoader parses AEMO-supplied .CSV files into the participant's RDBMS. The interface to the database is SQL, so the database can be any RDBMS accepting a standard SQL interface (with Oracle and SQL Server being supported by AEMO).

The pdrLoader polls the participant's DI folders and processes data from files into the participant's RDBMS conforming to the MMS Data Model. If the recovery features are enabled, periodically the pdrLoader sends a manifest request file to AEMO regarding the state of the files sent. AEMO sends a manifest report in response to the request. The pdrLoader compares the manifest report with the local database to identify any missing data, and, if necessary, sends a request to AEMO for any missing files. The catching-up process is at a lower priority than normal file handling.

The application design of the loader is a state machine. This means that, on restarting after a stop for any reason, the application automatically recovers by using the stored state.

3.4 Participant's RDBMS

The participant's RDBMS contains the data received from AEMO into tables matching the MMS Data Model, plus other tables and views needed to support such functions as the extraction, monitoring and replication of that data.

AEMO's production and pre-production systems are totally independently operated, so ensure the data flow from one is not confused with the other.

3.4.1 MMS Data Model

The MMS Data Model is the target model for products transferring data from AEMO to each participant. AEMO's intention is to supply the data transfer products pre-configured to deliver data consistent with the MMS Data Model.

The MMS Data Model includes database tables, indexes and constraints. The model is exposed as a physical model, so is different in presentation for each RDBMS family. However, a single logical model underlies all the physical models published by AEMO for a release.

The MMS Data Model is updated in line with the release process for the EMMS software. The EMMS software release cycle is normally six monthly—May and November. Usually, in association with each release, there are also changes to the MMS Data Model (for example, new tables and/or new columns added to existing tables).

The MMS Data Model is currently delivered for the RDBMS systems Oracle and SQL Server (with supported versions as shown in the release details for MMS Data Model).

3.4.2 Current EMMS baseline data

The baseline data is intended for initial loading into an empty participant's RDBMS, to provide key standing data immediately prior to commencing routine data feed.

Participants can fully populate the relevant tables with baseline data by copying each file in the \MarketData folder into the pdrLoader input folder.

The publication of an EMMS standing data baseline for the participant's RDBMS consists of complete data extracts from selected MMS Data Model tables on a daily basis. The data is in the latest MMS Data Model format and the files are placed in the \MarketData folder on the EMMS file server.

3.4.3 Historical data

AEMO has monthly and annual historical data on DVD, available for subscription. For further information, see "Additional Products and Services" in the "Guide to Market Systems – Maintaining and Extending Access" (in "EITS Publications", on page 35). This historical data is intended for historical analysis, kept and processed separately from the current data flows.

3.5 Replication Manager

AEMO currently supports one product (Replication Manager) to interrogate and manage PDR, as an alternative to direct querying and modification of the PDR tables in the RDBMS.

The Replication Manager product is an interactive user interface allowing participants to configure and interrogate Participant Data Replication (PDR) instances running at the participant's site. The Replication Manager software is installed locally on the system administrator's desktop. Replication Manager must run on a Windows operating system but it can access configuration data from any supported RDBMS regardless of the store's operating system or RDBMS.

4 Implementation

Implementing Data Interchange involves the setting-up or use of multiple products in multiple environments. All the parts cooperate, so they need to be set up accordingly.

This section, "Implementation", is the equivalent of a "Quick Start" guide to Data Interchange. Included here are: setting-up a new DI environment, typical installation locations, what order to install the DI product in, what user access and passwords are required, etc. This section is high level; details are elsewhere in this document plus related references.

All software packages and documentation mentioned in this guide are available from the secure [EITS Publications](#) area on AEMO's website.

4.1 Minimising potential implementation problems

- Read this guide ("Data Interchange Guide") fully before commencing your DI installation. A successful DI installation requires time, understanding and ongoing maintenance.
- For your first DI installation, carefully follow the steps described in "Setting-up a standard DI environment" on page 19. A 'standard DI implementation' is known to work, and is the target for the default installation settings. Creating a 'standard DI implementation' is the most reliable and easiest approach to installation. Once you have a successful pre-production installation, keep it for testing.
- There are numerous set-up possibilities, with some outlined in the "Considerations" on page 22. Your set up depends on your business needs and your resources. A 'standard DI implementation' may, or may not, satisfy all your requirements, however AEMO has chosen a workable, though basic, set-up for the 'standard DI implementation'.
- Each of the pdrBatcher and pdrLoader applications has an installation program intended for installing those applications to suit a 'standard DI implementation'. Unless you have rigorously analysed your requirements and the behaviour of these products, using the installation programs provide the most reliable method for successfully installing these products. The pdrBatcher and pdrLoader .properties files are pre-configured to suit a 'standard DI implementation', which includes file recovery. If you choose to do a manual install, (not recommended for first DI installation), make minimal changes to the .properties files. If you have a valid business need, the .properties files can be further customised at a later date to suit that particular need (after testing).
- Be aware that your initial DI installation does not contain historical data; see "Historical data", on page 16.
- Under no circumstances load the pre-production baseline data into your production environment and vice versa (see "Production and pre-production data sources", below).
- Under no circumstances, point your pre-production environment to the production environment and vice versa (see "Production and pre-production data sources", below).
- All participants are encouraged to use the EMMS pre-production environment to test implementation and to train their participant users, before attempting any changes or operations on EMMS production systems.

4.2 Production and pre-production data sources



Important Note: under no circumstances should a participant point their pre-production system to the AEMO production environment.

MMS data sets provided from AEMO's pre-production are different to that provided from AEMO's production. There can be very significant differences in the MMS data sets and data content provided from the respective environments, depending upon such factors as your own file subscriptions and other participants' activity.

Only periodic EMMS data feeds (for example, DISPATCH) are regularly available from AEMO's pre-production, with non-periodic feeds available on an ad-hoc basis.

All participants are encouraged to use the EMMS pre-production environment to test implementation and to train their participant users, before attempting any changes or operations on EMMS production systems.

4.3 Hardware specifications

Suitable hardware specifications are organisation-specific. Things to consider include redundancy, number of subscriptions, in-house business requirements, level of business impact of delay in availability of data, etc.

The sizing of any server depends on factors such as how much data to keep online, how many concurrent users, what requirement for performance, current and anticipated data volumes, and what data to download (everything or just a subset). AEMO recommends a capacity planning exercise to establish the appropriately-specified hardware platforms.

4.4 Data volumes

AEMO has the following estimates:

- **No. of files:** 2500 files per day (subscription to all suites and reports).
- **Data volumes:** 1.2 GB monthly of compressed public .CSV files, with private data over and above this.

4.5 Supported database platforms

AEMO progressively adopts new versions of supported databases, and retires superseded ones. For details of currently supported database platforms, see "MMS Data Model Implementation Note" (see "Data Interchange products and resources", on page 9).

4.6 Typical installation locations

- **pdrLoader:** on the database server with MMS Data Model, or on a closely-associated application server.
- **pdrBatcher:** on an application server (may be in local network DMZ), or on the database server.

The PDR software is written in Java (a platform independent language), so can be installed in any environment supporting Java (e.g. Unix, Linux, Microsoft Windows).

- **Replication Manager:** on the system administrator's desktop.

Replication Manager is a Microsoft Windows application.

4.7 Before starting your DI installation

Before commencing to set-up your Data Interchange environment, ensure you have the following information available so you can access the AEMO data and your RDBMS:

- “EMMS Data Interchange Guide”
- “EMMS Data Subscription Services User Interface Guide”
- “Participant Data Replication Loader User Guide”
- “Participant Data Replication Batchers User Guide”
- “Replication Manager Installation Guide”

Prepare the following items:

- Appropriately skilled IT staff.
- A network security design.
- The appropriate installation package for your RDBMS (Oracle or MS SQL Server).
- A database administrator or a service provider with a database administrator.
- Your participant ID.
- For each environment (pre-production, then production)
 - The IP address for the EMMS file server; for help see “AEMO’s IP Addresses for Participants”.
 - Your login, password and connection details for your RDBMS.
 - Your login and password to the EMMS Data Subscription web application.
 - Your login and password to the EMMS file server.

4.8 Setting-up a standard DI environment



Important Note: set up your first Data Interchange implementation in your pre-production environment. Once you have a successful pre-production installation (used for testing), you can replicate it in production to receive live NEM market data, then decide on other set-up possibilities to suit your business needs. Be careful to keep the pre-production and production data entirely separate, always.

In this document, the use of the term ‘standard DI implementation’ indicates a fully-operational data interchange instance known to effectively and accurately replicate data from AEMO’s EMMS into a remote database (see “pdrBatcher and pdrLoader to a single database, with file recovery” on page 23). AEMO supports the ‘standard DI implementation’ and each of the products AEMO supplies. Any variations to the ‘standard DI implementation’ are entirely the responsibility of the implementer, even if using documented features of the various products. By virtue of its unique position in the energy market, AEMO cannot provide individual consulting services or advice.

To set up a standard Data Interchange environment, carefully follow the steps described below:

Step	Do	For help
1.	Set-up EMMS Data Subscription Services: <ul style="list-style-type: none"> Configure report subscriptions using the EMMS Data Subscription web application. For a 'standard DI implementation', subscribe to the full MMS Data Model data set. 	<ul style="list-style-type: none"> See "EMMS Data Subscription Services", on page 12. Read "Data Subscription Services User Interface Guide".
2.	Create an RDBMS instance to conform to the MMS Data Model: <ul style="list-style-type: none"> Install the latest MMS Data Model structure (target tables) by downloading and configuring the current MMS Data Model .zip for your database type (Oracle or MS SQL Server). Do not make any changes to the MMS Data Model structure. 	<ul style="list-style-type: none"> See "EMMS file server", on page 13. Read "MMS Data Model Installation Note" and other documents contained in the Documentation folder in the .ZIP file.
3.	Install the Participant Data Replication Loader (pdrLoader): <ul style="list-style-type: none"> For a 'standard DI implementation', install pdrLoader using the "pdrLoader Installation" application to suit your RDBMS instance. Alternatively, extract and configure the pdrLoader software to load data from your participant DI folders to your RDBMS. 	<ul style="list-style-type: none"> See "Participant Data Replication Loader (pdrLoader)", on page 15. Read "Participant Data Replication Loader User Guide".
4.	Configure the pdrLoader application to read the EMMS NEM report data files into the database: <ul style="list-style-type: none"> Download the current version of the MMS_Data_Model_pdrLoader_Configuration.zip and expanding to a working folder. Expand the PUBLIC_PDR_CONFIG_nnn..._nnn..._FULL.zip. Copy the PUBLIC_PDR_CONFIG_nnn..._nnn..._FULL.csv into the Reports folder of pdrLoader. 	<ul style="list-style-type: none"> Read "README.txt" and "MMS Data Model pdrLoader Configuration Implementation Note".
5.	Run the pdrLoader application: <ul style="list-style-type: none"> In Windows, double-click pdrLoader.bat file in the Lib folder. In unix-like environments, execute the pdrLoader.sh file in the Lib folder. 	<ul style="list-style-type: none"> Read "Participant Data Replication Loader User Guide".
6.	Install Replication Manager: <ul style="list-style-type: none"> Extract and configure the Replication Manager software on a local PC running Microsoft Windows to monitor and control the operation of the pdrLoader. Ignore all references to "Replica" in software and documentation, and use "Parser" options. 	<ul style="list-style-type: none"> Read "Replication Manager Installation Guide".
7.	Populate your RDBMS with baseline data: <ul style="list-style-type: none"> Load baseline data by copying files from the EMMS file server \MarketData\Baseline to your pdrLoader Reports folder (baseline data excludes historical data published periodically). 	<ul style="list-style-type: none"> See "EMMS file server", on page 13. See "Production and pre-production data sources" on page 17. Load historical data (optional); usually only for production data warehousing.

Step	Do	For help
8.	Install the Participant Data Replication Batchter (pdrBatcher): <ul style="list-style-type: none">▪ For a 'standard DI implementation', install pdrBatcher using the "pdrBatcher Installation" application.▪ Alternatively, extract and configure the pdrBatcher software to move files from the EMMS file server to your participant DI folders (created during the pdrLoader set-up).	<ul style="list-style-type: none">▪ See "Participant Data Replication Batchter (pdrBatcher)", on page 15.▪ Read "Participant Data Replication Batchter User Guide".
9.	Run the pdrBatcher application: <ul style="list-style-type: none">▪ In Windows, double-click pdrBatcher.bat file in the Lib folder.▪ In unix-like environments, execute the pdrBatcher.sh file in the Lib folder.	<ul style="list-style-type: none">▪ Read "Participant Data Replication Batchter User Guide".

5 Considerations

Data interchange is unlikely to be just from AEMO to a single participant database, especially in a production environment; there are just too many unpredictable events that can cause havoc with business-as-normal. In detail, the considerations for implementing data interchange are likely to be different from one participant to another due to the numerous possibilities.

Just some of the considerations regarding data interchange are:

- What happens when part of the data feed fails?
 - Database failure?
 - Server failure?
 - Network failure?
- What security considerations apply to the machine(s) in the data interchange path?
- Are there more consumers of AEMO data than can be supported from a single participant database? For examples:
 - Multiple locations?
 - High performance?
 - Space availability?
 - Need-to-know or not?
- Are there multiple repositories of AEMO data needed? For examples:
 - To serve different business purposes (short-term trading, scenario analysis, accounting, archive, etc.)?
 - To serve different business groups (internal divisions, sub-contractors, etc.)?
- Are there functional groups requiring combined data feeds? For examples:
 - Multiple participant IDs?
 - Other data sources?
- What inputs do you provide to AEMO, such as bids?
- What other technologies are in use, or available, for data interchange? What are the advantages and disadvantages?
- Is pre-production to mirror the production set-up, be a subset of production, be completely different, or all of the above?

A key issue for each participant is to be sure that you have access to the technical skills to address all these issues. AEMO only has detailed knowledge of one end of the interchange path (the EMMS file server, or, equivalently, from the participant's perspective, the remote data feed). It is inappropriate for AEMO to be involved with each participant's architecture and its maintenance.

As the participant's needs are extended from simple data feed, through file recovery to hot system recovery for multiple target user-groups, the modular approach to application design means many scenarios can be accommodated. To make implementation as practical as possible to cover all the possibilities, AEMO makes the components functionally independent and highly configurable.

Rather than attempt to directly address the considerations, the following sections cover various scenarios for data interchange. Where your situation is different, make the appropriate adjustments in the light of the various examples. Some participants may have a combination of these scenarios.

5.1 DI Scenarios

The PDR products are very flexible with regard to configuration. It is very easy to over-complicate the implementation. Start with as simple a configuration as needed, and change only what is necessary to achieve each incremental improvement (with thorough testing).

AEMO supports the products it supplies with regard to the individual operation of each supplied component. The configuration for the products is individual and totally within the control and responsibility of each participant.

All these examples assume the use of pdrLoader and pdrBatcher.

For simplicity in the diagrams and explanations, the multiple data flows may be shown as one flow. In practice, in the primary direction, there are at least two folders at AEMO (Reports and ReportsTrickle folders). In the reverse direction, AEMO has one folder to receive message files back to AEMO, and at least one for bids (if used).

5.1.1 pdrBatcher and pdrLoader to a single database

The simplest arrangement to consider is running the pdrBatcher and pdrLoader on the same machine as the database is installed. In this scenario, the participant uses the pdrBatcher to receive the files and uses the pdrLoader to load the files into a single database.

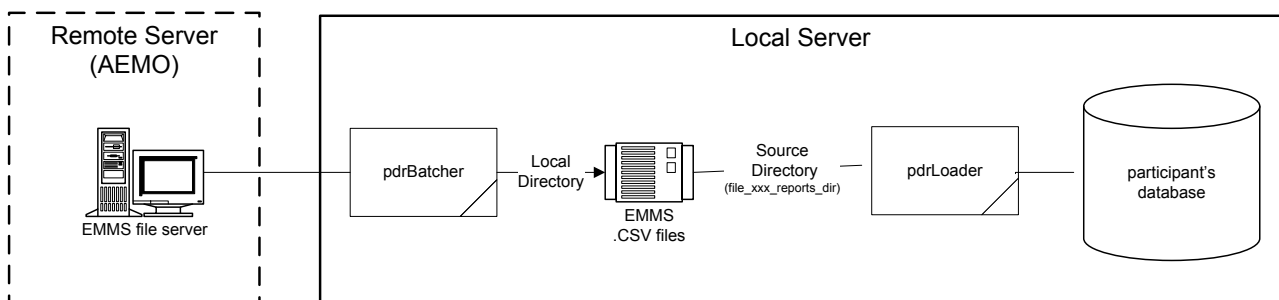


Diagram 3: pdrLoader to a single database

This arrangement achieves the primary objective of feeding data from AEMO's EMMS to a participant's database. There are many shortcomings in such a simple scenario, but can be a good place to start when first becoming familiar with the capabilities of pdrBatcher and pdrLoader.

As one example of a shortcoming, security considerations suggest the pdrBatcher runs on a machine in the DMZ of the participant's network and transfers data to a machine running the pdrLoader inside the fully secured part of the participant's network. The pdrLoader may run on a different machine to the database, or may be on the same machine (preferred).

As another example of a shortcoming, consider the corruption or loss of a file anywhere in the chain. In this scenario, there is no way to reconcile what has been received into the participant's database with what was made available on the EMMS file server. To allow recovery means looking at another scenario (see "pdrBatcher and pdrLoader to a single database, with file recovery" below).



Note: consider backing up every file received by pdrBatcher, by duplicating its output folders. If something unfortunate happens downstream, re-processing missing data is as simple as copying files from the backup folder back into the processing stream (such as, into the input folder for pdrLoader).

5.1.2 pdrBatcher and pdrLoader to a single database, with file recovery

The addition of automated recovery from file loss or corruption to the single database scenario means the addition of a reverse communication channel. The pdrLoader can generate files to be sent back to the EMMS file server, via pdrBatcher, and EMMS responds with reconciliation information for pdrLoader to check.

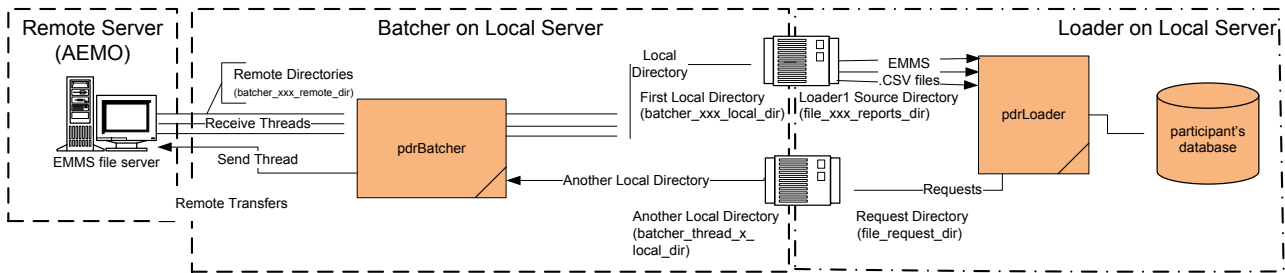


Diagram 4: Single database with recovery

Both pdrBatcher and pdrLoader can have multiple threads. The diagram above shows just two folders as the interface between pdrBatcher and pdrLoader, but there are more. The key to successful interfacing is to identify the location of each interface folder in the file system and record that location against the identifier relevant to each component. This means, for examples:

- An input folder to pdrLoader has to be an output folder from pdrBatcher, so has a different identifier in each of the environments pointing to a single folder.
- A pdrBatcher thread feeding to a given output folder needs to get its input from the folder on the EMMS file server appropriate to the intended purpose of the pdrBatcher output folder.
- Each pdrLoader interface folder must map through to a folder on the EMMS file server.

Some of the files from EMMS are large (for example, the “next day” files) and can take considerable time to transfer and load, potentially holding up processing of more-frequently produced files. Having multiple processing threads in both pdrBatcher and pdrLoader, plus control over filtering and performance of each thread, allows each participant to tune the data flows to suit their own priorities. Examination of the supplied sample `.properties` file for pdrLoader shows an example for allowing high-frequency files to be processed in parallel with large files.



Note: implementing recovery includes setting the pdrLoader property, `file_load_automated_recovery`, to true (as it is in the sample pdrLoader `.properties` file).

5.1.3 pdrBatcher to multiple directories and parallel pdrLoaders to multiple databases, without file recovery

Running with a single database does not allow for database downtime without affecting downstream users. Running parallel pdrLoader instances into separate databases provides a minimum data delay for alternative databases for downstream feeding. This scenario involves running a pdrBatcher instance to multiple directories and parallel pdrLoaders to multiple databases, described here without recovery for simplicity of explanation. The pdrBatcher can transfer each file into multiple directories (two shown below). The first pdrLoader instance loads one database from the first pdrBatcher output directory and the second pdrLoader uses the second Batcher output directory as its source. If either pdrLoader or database fails, the other continues.

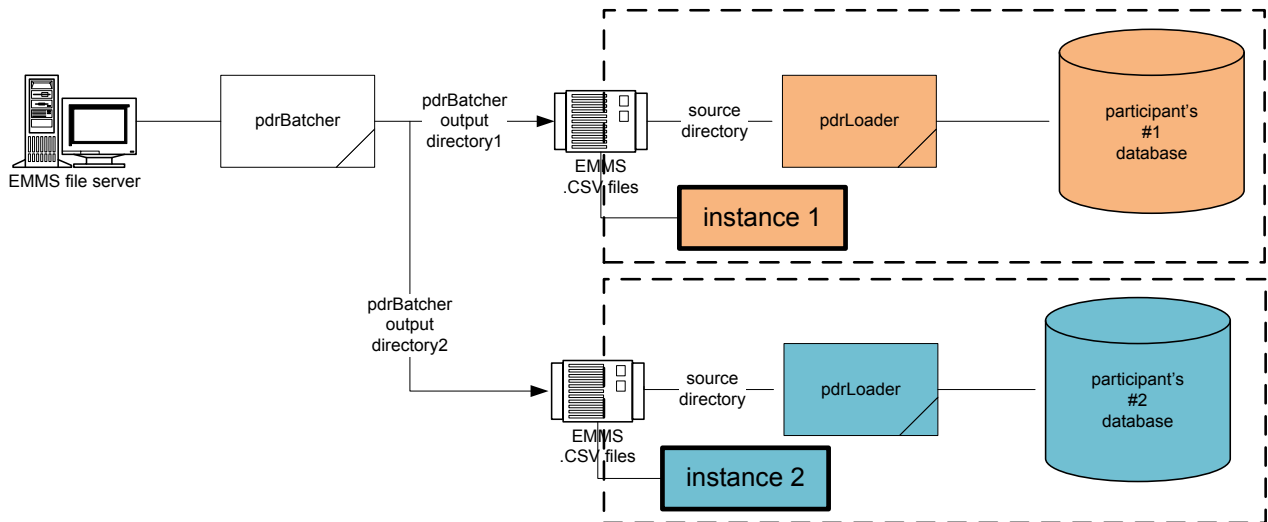


Diagram 5: Parallel-fed databases (without recovery)

There is no guarantee that the databases are in the same state at any given point of time, simply because of the dynamic nature of the data flows.



Note: If two or more user groups require data feed to be as quick as possible, but for different sub-sets of the AEMO data, adapting this scenario is straightforward; adjust the files actually loaded into each user group’s database through its associated pdrLoader configuration.

5.1.4 pdrBatcher to multiple directories and parallel pdrLoaders to multiple databases, with recovery

Adding file recovery to the scenario of running a pdrBatcher to multiple directories and parallel pdrLoaders to multiple databases is incrementally similar to adding recovery to the scenario for a single database. This means all the considerations in “pdrBatcher and pdrLoader to a single database, with file recovery”, on page 23, apply in this scenario also. The AEMO file server has a single folder to receive all requests.

The pdrBatcher transfers the files into two directories in the diagram below, although more are possible. The pdrLoader of the first instance loads one database. The second instance of the pdrLoader uses the second transfer directory as its source. Both pdrLoaders feed back to a single directory and the pdrBatcher sends the files to AEMO.

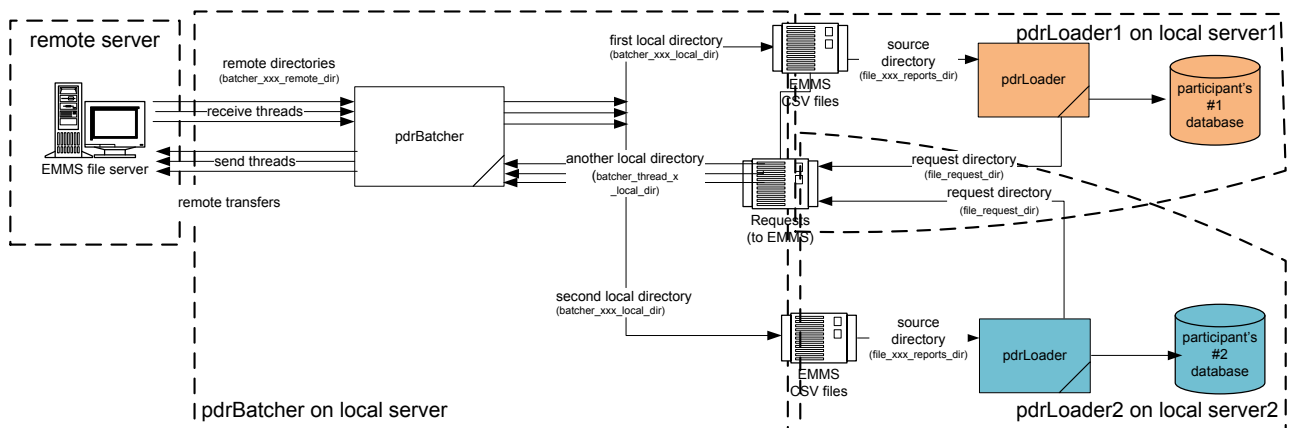


Diagram 6: Parallel-fed databases with recovery

A possible situation is for a file into one pdrLoader to be corrupted, but not the same file into the other pdrLoader. Each pdrLoader instance needs to be individually identifiable, and all messages specific for each pdrLoader need to carry that identifier (see the pdrLoader property,

instance_identifier). An identifier unique to each pdrLoader (within the participant) allows for files specific to that instance to be recognised only by that instance and ignored by all others.

Where a participant requires the same data to be made available to two groups of users, the two pdrLoaders can be configured to process the data in the same way and the pdrBatcher threads feeding each pdrLoader forward the same data. If the timing needs for the two groups are different, then the performance of the two paths can be independently adjusted to provide the required business outcome whilst minimising computing power and cost. Adjustments include creating more threads in the pdrBatcher, changing the filters on pdrBatcher threads, changing the filters on pdrLoader threads, using more (or less) powerful hardware, adding new machines to run pdrBatcher and pdrLoaders, and changing network arrangements and speeds.

If different groups need different data in the database they use, either the pdrLoaders can have different filters, or the pdrBatcher can provide different feeds (by adjusting its filters), or both (if the outcome is really worth the complexity).

Generalising this scenario to three or more groups of users is quite practical from the software functionality point-of-view. The design considerations are a matter of matching the machine and network capabilities with the expected loads.

5.1.5 Serial feed to multiple databases

After having set up a simple data feed to a single database (as in “pdrBatcher and pdrLoader to a single database” on page 23), it is quite practical to add another database without adjusting anything in the simple data feed arrangement. All files successfully processed by pdrLoader (whether loaded or not) are placed in the Archive folder.

The Archive directory of the first pdrLoader is the source for another pdrBatcher to transfer to the input for a second pdrLoader to load the second database. This is the serial feed to multiple databases scenario.

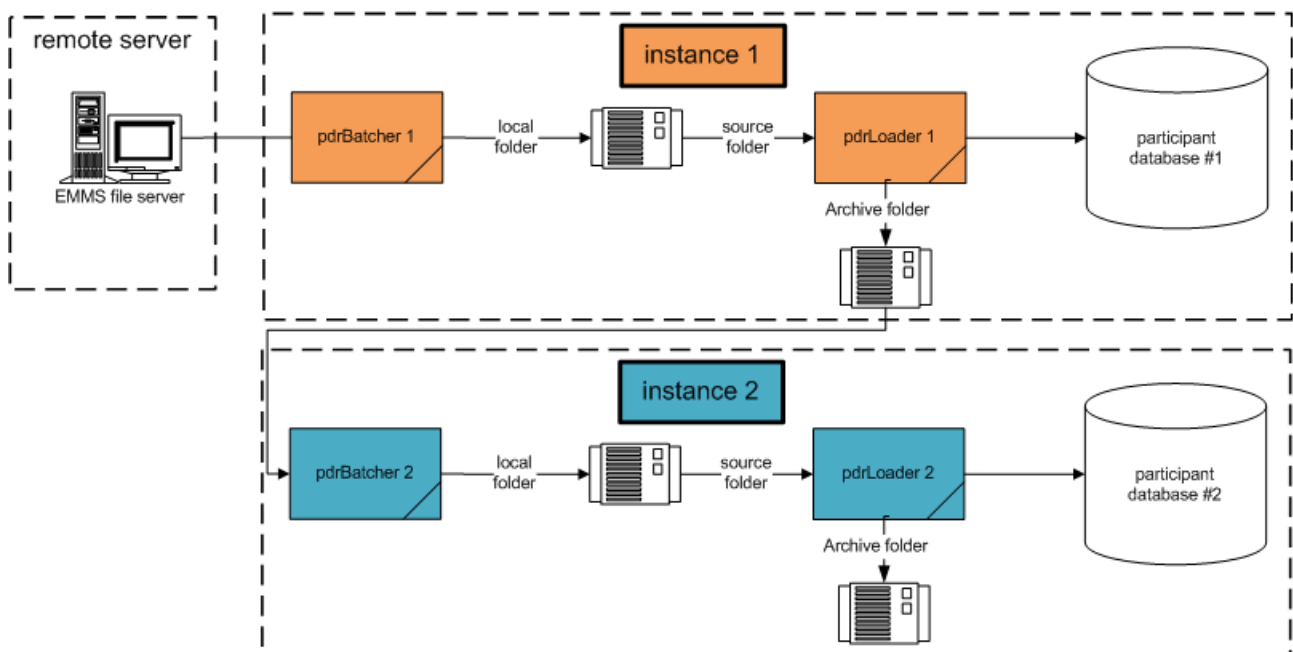


Diagram 7: Serial-fed databases

This technique provides one way to have a reliable local ‘master’ database with the backup database only fractionally behind. A failure of the first DI instance means the second DI instance requires only a change of the source for the second pdrBatcher, so it can become the new ‘master’.

At the time of changeover, files already taken from the EMMS file server by the first DI instance but not completed processing by that instance are effectively lost to the second DI instance. This means the file recovery mechanism is required (see “pdrBatcher and pdrLoader to a single

database, with file recovery” on page 23 for considerations). The recovery process in the pdrLoader of the second DI instance (now the ‘master’) can detect the missing files and request a re-send of them.

A ‘daisy-chain’ of instances can be suitable for several purposes, including provision of full databases for historical analysis as well as disaster recovery. To adjust for the failure of any instance in the ‘daisy-chain’, change the source for the following instance to be one instance earlier.

As an alternative application of the “serial feed to multiple databases” technique, the first pdrLoader may select data of short-term interest for the first database (by selective filtering) and pass through the rest of the files. The second pdrLoader and database can act as a warehouse holding all data.

It is possible to omit the pdrBatcher of instance 2 and have the pdrLoader for instance 2 get its input from the Archive folder of instance 1. In this case, the location of that folder is probably better on the second server to allow for total server failure of server instance 1, which, in turn, suggests the pdrBatcher is better on a different server again (say, in DMZ). The diagram below does not show the file recovery flows, but they are necessary for the correct functioning of this scenario.

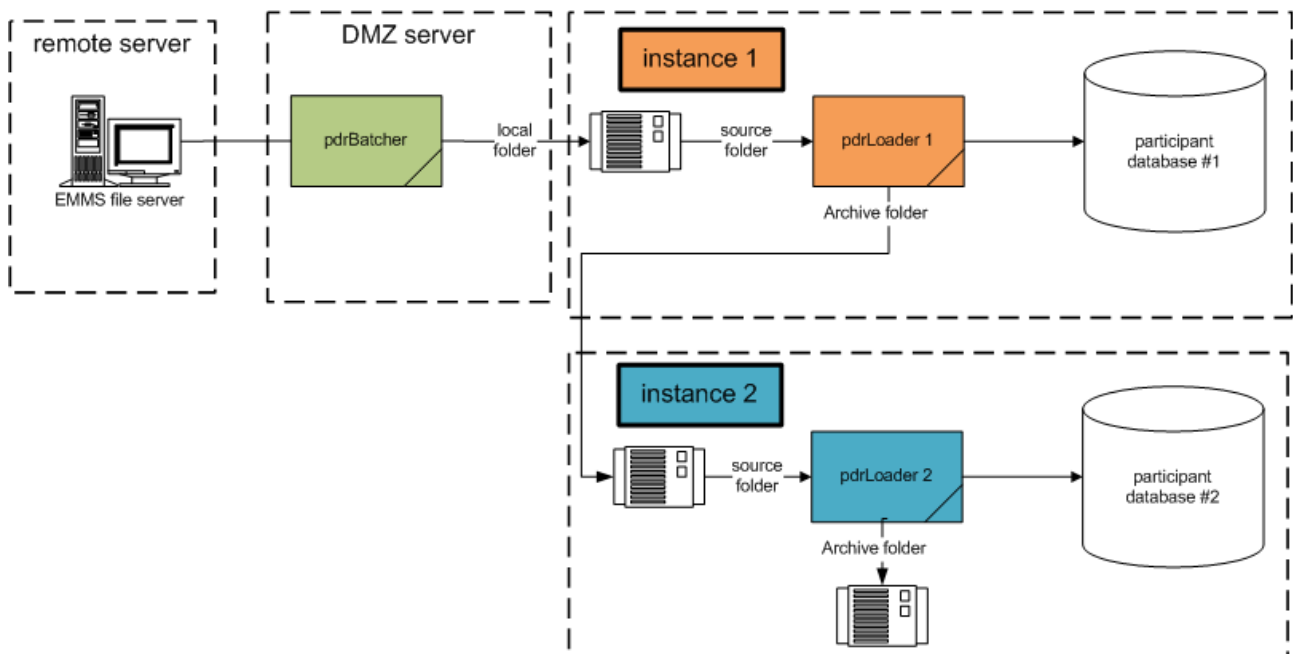


Diagram 8: Serial-fed databases with single pdrBatcher

For a failure of the instance 1 server, the output from the pdrBatcher is switched to the source folder for the instance 2 server. A failure of the DMZ server needs a similarly-configured duplicate server.

5.1.6 Summary of example configurations

The modes of possible failure of each configuration together with satisfying the differing business needs of each participant (for example, data needs and performance) mean there is no single recommendation for the configuration of data interchange based on the PDR applications. The pdrLoader and pdrBatcher are highly configurable applications. The use of multiple instances (each serving a single database) allows for handling a wide variety of business needs, including selective data visibility, disaster recovery and data warehousing.

The issue of performance is highly dependent on data selection, equipment configuration, network factors, database implementation, other workloads and numerous other factors specific to each participant. The PDR applications are very efficient in themselves, so planning a suitable

configuration needs to consider the relative performance of the interfaces to the PDR applications as being of key importance.

5.2 Bespoke data interchange systems

A user not using Data Interchange for file transfer to or from the AEMO fileserver(s) typically has a bespoke system or application using a proprietary means. There are two alternative approaches for such a system to achieve compliance with using passive FTP to the EMMS file server.

- The application or script(s) involved with the file transfer must use passive FTP.
- If the bespoke system cannot use passive FTP, implement a pdrBatcher instance to provide a local replica of the EMMS pre-production file server and then redirect the application or script(s) to the local replica. Repeat for the EMMS production server.

5.2.1 Providing a local replica using pdrBatcher

The steps to use a local replica of the EMMS pre-production file server are:

1. Create a local set of folders in the same hierarchy as the EMMS server.
2. Change the network drive mapping to point to the root of the local set of folders, or redirect the application to use the local folders.
3. Set up pdrBatcher to replicate between AEMO's folders and the local folders using passive FTP.

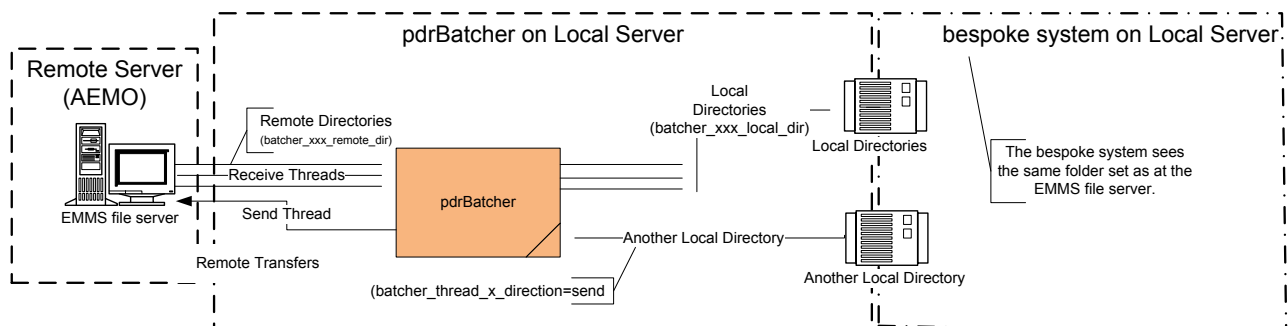


Diagram 9: Replicating the file server folder set

5.3 Data Sharing

AEMO's wholesale Electricity Market Management Systems (EMMS) allows a NEM participant to share data with another participant, subject to an authorised request. Implementing data sharing means the full Data Interchange data set (including private data) for the requesting participant ID is made available to another participant's sets of folders in AEMO's EMMS file servers, as nominated in the request. The sharing of data is for both pre-production and production EMMS.

A data sharing requirement may emerge, for example, due to a merger with, takeover of, or sale of another NEM registered participant. The requirement may be to use one company's IT system to manage access and process the data for two or more participant IDs via just one participant's set of folders in AEMO's EMMS file server.

5.3.1 Outcome of sharing data

Sharing your NEM EMMS Data Interchange data with another EMMS registered participant ID means you must have Data Interchange and the EMMS Data Subscription service set-up and configured at your site.

For example, participant 1 wishes to manage both participant 1 and participant 2 data via participant 1's IT systems and access mechanism.

- After requesting set-up by AEMO, participant 2's data is made available to both participant 1 and participant 2 file shares.

- Participant 1 receives PUBLIC-DATA.ZIP and PARTICIPANT1_DATA.ZIP (includes both participant 1 and participant 2 data).
- Participant 2 receives PUBLIC-DATA.ZIP and PARTICIPANT2_DATA.ZIP (includes only participant 2 data).
- Each participant ID can independently manage which data packages they receive for their Participant ID via EMMS Data Subscription services. A participant receiving data for another participant does not impact the other participant's file subscriptions.

A key saving is the bandwidth and download time for the public data, since it is common to both participant IDs. Having all the data in one database can be both an advantage and a disadvantage.

It is possible to arrange for one participant ID to receive data for more than just one other participant ID.

It is possible to arrange for both participant IDs to get the data for both participant IDs.



Important Note: Unless specifically and clearly requested otherwise, data sharing is one-way (that is, one participant ID gets just its own data and the other participant ID gets the data for both participant IDs).

5.3.2 Setting up data sharing

To set up data sharing:

- Understand what you wish to achieve and ensure your IT systems can process data for multiple participant IDs from the one file. Ensure downstream processes, procedures and systems can handle the database with data for multiple participant IDs.
- Formally request data sharing.
 - The formal request needs to nominate a business contact with the authority to request or authorise the sharing of private data from one business entity to another, plus nominate a technical contact to directly implement changes at the participant site needed to put the data sharing into place.
 - If there are two (or more) companies (that is, having different ABNs), each company must provide a request to the AEMO Help Desk, on their respective company letterhead, signed at senior management level; detailing their specific circumstances. Include the following template paragraph authorising the sharing of private data.
 <Company name> < ABN>for <participant ID> <requests|authorises> AEMO to make available <company name> < ABN>for <participant ID> EMMS full subscription data set (including private data) to <company name> < ABN><participant ID> via the EMMS Data Interchange and Data Subscription service. This is to remain in place until advised in writing. A letter from <Company name> < ABN> for <participant ID> confirming this is forthcoming.
 - Participants with the same ABN for all participant IDs involved only need a single request from the company registered for their participant IDs.
- After confirming full agreement by all affected parties (including technical, as necessary), AEMO processes the sharing request into pre-production.
- Similarly, processing the sharing request into production involves confirming full agreement by all affected parties (including technical, as necessary).

5.3.3 Removing data sharing

To remove data sharing at any time, only one involved participant need ask for its removal.

One situation requiring removal of data sharing is for a change in the legal ownership or control of a participant ID involved in sharing of data.

To formally request withdrawal from a data sharing arrangement, submit the request on a company letterhead and signed at senior management level, detailing their specific circumstances. Include the following template paragraph authorising the withdrawal of sharing of private data.

- <Company name> < ABN>for <participant ID> requests AEMO to remove sharing of <company name> < ABN>for <participant ID> EMMS full subscription data set (including private data) from <company name> < ABN><participant ID>.

A participant purchasing an existing NEM registered company can contact the AEMO Helpdesk to ask if any data sharing arrangement is in place, and, if so, can request the sharing be removed (as above).

5.4 Hot recovery

The following are just some examples of possible modes of failure. Each participant has the responsibility to evaluate their own situation, and to implement their own configuration against the modes of failure that affect them. Testing is important to ensure the processes necessary for recovery are in place. AEMO is not in a position to give advice on individual cases.

Both the pdrBatcher and pdrLoader have activity monitoring features (see the heartbeat file in each) to allow for a monitoring application on a separate server to automatically switch data feeds and raise alarms.

5.4.1 Failure of database or pdrLoader

If the database or pdrLoader fails when running parallel loaders, no configuration changes are needed to have access to an up-to-date database, since each pdrLoader is independent of the other (see “pdrBatcher to multiple directories and parallel pdrLoaders to multiple databases, without file recovery” on page 24).

5.4.2 Failure of a stand-alone pdrBatcher

If a pdrBatcher fails on a separate server to other DI components, starting a duplicate pdrBatcher is as simple as having an alternative pdrBatcher already set up ready to go (say, on another server).

5.4.3 Failure of server with pdrBatcher, pdrLoader and database

If running serial loaders, changing the source for a pdrBatcher instance is quick, easy and automatable (see “Serial feed to multiple databases” on page 26). The next pdrBatcher after the failed system simply needs to be changed for its source to point to the failed system’s source. The return path to AEMO in each pdrBatcher in the chain can all point to the primary pdrBatcher (presumably in DMZ of network) so does not need to be altered for a failure. If the primary pdrBatcher fails, starting a duplicate Batcher is as simple as having the alternative set up ready to go.

5.4.4 Failure of the network connection

All of the configurations described in “DI Scenarios” on page 23 assume the network connection is functional. For both planned maintenance and unplanned failures, ensure you have an alternative connection (that is, one that is unlikely to be similarly affected) and you have tested and practiced failover.



Note: for suggestions on how to check the network connection, see “Needing Help?” on page 36.

It is possible to implement co-primary databases without using a common network connection, but only if you have control of two (or more) participant IDs. Each participant ID provides a data feed to a processing chain, with each processing chain being totally independent of the other. The independence is necessary because each pdrLoader can only send files to the same participant

folder set as the source of the files. Any cross-feeding of files can compromise the file data recovery mechanism.

5.5 Outline of the file recovery strategy

5.5.1 File synchronisation

The pdrLoader periodically issues a manifest request. This only occurs when the pdrLoader is idle and there are no files waiting to be loaded. The manifest request is a small text file, including an instance identifier (unique within the participant environment). It is the instance identifier that allows the specifically requested files out of the EMMS file server to find their way to the correct pdrLoader instance.

The Batcher then transfers this request to the AEMO systems, which process the request file and provide an acknowledgement. The acknowledgement file is itself in the AEMO CSV data format standard (see “References” on page 35) and, if the instance identifier in the response file matches the one defined within the Loader, the Loader processes it.

The acknowledgement of a manifest request simply means that the report request has been lodged into the EMMS systems and the requested reports are scheduled for delivery. The time-frame for delivery does have some dependency on the loading of the EMMS systems, but generally can be expected to start within a minute or so of the request.

If the data is excessively delayed (say, due to a network failure), the pdrLoader effectively restarts the data request process in due course.

5.5.2 Trickle feed and normal CSV feed

The CSV feed is the real-time data feed. The trickle feed provides a mechanism for a low priority catch-up of data. The trickle feed fills in data gaps in historical data when there is idle time. The normal .CSV feed has priority over any backfill operations.

5.5.3 AEMO data retention

AEMO has sized hardware to retain up to 6 months worth of files online for all participants, including private as well as public data. Private data is only created live by AEMO upon active request (that is, via the EMMS Data Subscription web interface) and cannot be retrospectively created.

Whilst AEMO keeps copies of all files sent to each participant, it is even safer if each participant also keeps a copy of all files received. Besides, using a local copy is going to be much quicker than waiting for the trickle feed from the EMMS file server.

6 Maintenance

- Update to the latest release as soon as possible after AEMO issues a release.
- Modifications to the MMS Data Model are at your own risk. Upgrades to the data model are tested and certified against the standard MMS Data Model. If you must modify the MMS Data Model in even the smallest way, keep the modifications separate from any instance fed by the primary pdrLoader process. Test any modification in pre-production first.

6.1 File data feed

To receive data from AEMO, do both of the following:

- Subscribe to the chosen files (using the EMMS Data Subscription web application).
- Ensure the chosen files are set to be loaded by pdrLoader into your RDBMS (using Replication Manager or direct SQL statements).

A file feed can be turned off by either:

- Unsubscribing from the file using the EMMS Data Subscription web application.
- Setting the pdrLoader file load to inactive in your RDBMS (using Replication Manager or direct SQL statements).

The ramifications are different, so choose the method carefully.



Note: if you unsubscribe from a file, it is no longer delivered to your NEM Reports folder on the EMMS file server. If you set the pdrLoader file load to inactive, the file is still generated and transferred but its content is not loaded to your RDBMS. In the latter case, AEMO still maintains a copy of that file in its online archive for up to 6 months, so the file can be re-requested for transfer at a later date. If a participant is not subscribed to a file, AEMO does not produce the file, so it is not possible to retrieve it. AEMO does not have any production facility to produce a file as at an historical date. The recommendation is to subscribe to all available files and selectively load into your RDBMS as required by the business.

Files can be re-requested manually, either using direct SQL commands or using the Replication Manager. One possible reason for re-requesting data is to fill a data gap (perhaps caused when a report is set inactive for a period). Another way to fill a data gap is to locate your own archive copies of the relevant files and simply copy them into the input folder of pdrLoader.

6.2 Monitoring

6.2.1 Database state

To tell if the database is up to date, either:

- Inspect the PDR_MANIFEST_LOG table directly, see the “Participant Data Replication Loader User Guide” for more details (see “Data Interchange documentation set”, on page 10).
- Use the Replication Manager to show all known data discrepancies in this database (refer to “Replication Manager User Guide” for more details).

6.2.2 Operational checks

AEMO recommends that participants put in place appropriate monitoring of their Data Interchange environment to ensure smooth operation.

Key monitoring points include:

- Build up of files in your participant DI Folders.

- Space monitoring at the database and file system levels.
- Log file inspection of the pdrBatcher and pdrLoader processes.
- Timeliness of specific tables in your RDBMS conforming to the MMS Data Model.

For more details, see the “Participant Data Replication Loader User Guide” and the “Participant Data Replication Batcher User Guide” (see “Data Interchange documentation set”, on page 10).

6.3 Future upgrades

Participants are required to implement upgrades to their systems in line with the six monthly EMMS release cycle. These are typically scheduled near the middle and near the end of each year (May and November) and potentially involve additions or changes to the published MMS data (that is, MMS Data Model and its dependencies). The EMMS systems support only the latest and previous versions of data feeds, and AEMO recommends that participants schedule resources to maintain their environment in accordance with the EMMS releases to ensure no loss of data and that all downstream data consumers remain on a supported configuration.

There may also be updates to the PDR product set which, typically, are scheduled outside of the EMMS release cycle. Updates can include maintenance and enhancements to the products themselves. AEMO supports the latest and previous versions of products, and AEMO recommends that participants maintain their installation in accordance with advice from AEMO to ensure remaining on a supported platform.

Every six months, in line with new EMMS releases (May and November), use EMMS Data Subscription Services to:

- Download and apply scripts to upgrade your RDBMS conforming to the MMS Data Model.
- Keep your MMS Data Model RDBMS current, as only the latest data model and legacy versions are supported.
- Download and install new and updated pdrLoader configuration files.
- Subscribe to any new files.
- Unsubscribe from legacy files, waiting until successfully parsing at least one of any new version.
- Implement data retention and archiving policies.
- Monitor the PDR environment to ensure smooth operation.
- Backup all files from AEMO—in the event of data loss you can rebuild the database by running all files in sequence.
- If you must modify the MMS Data Model, keep the modifications separate from the main data feed by pdrLoader. Test any modifications in pre-production first, including failure recovery.

7 Glossary

7.1 Abbreviations

Abbreviation	Abbreviation Explanation
AEMC	Australian Energy Market Commission
CSV	Comma-separated values; a file format for exchanging data using commas as delimiters. For help with the CSV format, see the “AEMO CSV Data Format Standard” (see “References” on page 35).
DI	Data Interchange; either the project or the set of products implementing the project
DMZ	Demilitarized zone, for network security.
EMMS	Electricity Market Management System; software, hardware, network and related processes to implement the National Electricity Market (NEM).
NEM	National Electricity Market
NEMPART	EMMS file server formerly the NEM Participant Server (NEMPART or just Participant Server).
NER	National Electricity Rules
PDR	Participant Data Replication
RDBMS	Relational Database Management System (e.g. Oracle or SQL Server)

Table 1: Abbreviations

7.2 Special terms

Term	Definition
Batcher	Participant Data Replication Batcher
CSV reports	NEM reports
Data Interchange (DI)	EMMS data transfer to and from participants; commonly abbreviated to DI.
DI documentation set	The set of documents to reference for help with Data Interchange.
DI product set	The set of products implementing Data Interchange.
Loader	Participant Data Replication Loader
MMS Data Model	The definition of the interface to participants of data published by AEMO from EMMS.
Participant RDBMS	Local participant’s RDBMS, conforming to the MMS Data Model.
pdrBatcher	Participant Data Replication Batcher
pdrLoader	Participant Data Replication Loader
Replication Manager	Manages the configuration settings and monitors the status of the replication of data from AEMO’s systems to the participant’s RDBMS.
Rules	National Electricity Rules

Table 2: Special terms

8 References

The resources listed in this section contain additional related information that may assist you.



Note: it is important to ensure that you are reading the current version of any document.

- About Administration: for information about using Set Participant (e.g. for agents), see the online help under the Administration menu item, when logged into the EMMS Web Portal.
- AEMO Help Desk: phone: 1300 300 295, option 2; e-mail: helpdesk@aemo.com.au.
Contact AEMO's Help Desk to suggest corrections to this document or to request AEMO support.
- National Electricity Rules: see the AEMC website <http://www.aemc.gov.au>.

8.1 AEMO's website

The following documents are found on AEMO's website:

- "Guide to Market Systems – Gaining Access", for information on access to the Electricity Market Management Systems (EMMS) Web Portal. Participants wishing to use the EMMS Web Portal are required to have access to the AEMO Market Systems via the MarketNet Private Network. MarketNet provides information via web interfaces to participants, available from <http://www.aemo.com.au/registration/nemnet.html>.
- Participant Administration: For information about managing a Web Portal account (accounts are created in MSATS), see the "Participant Rights Administration User Interface Guide", available from <http://www.aemo.com.au/electricityops/userguide.html>.

8.2 EITS Publications

Participant users with the credentials can find the following documents in the secure [EITS Publications](#) area on AEMO's website (application to AEMO's Help Desk or see "About Administration" in the EMMS Web Portal). Documents in "EITS Publications" are available to registered participants only.

- "AEMO CSV Data Format Standard", describes the CSV standard used within flat files provided to participants. Its primary function is to provide sufficient information to allow participants to understand the AEMO CSV data format standard.
- "AEMO's IP Addresses for Participants", provides information about URLs for accessing AEMO's IT systems.
- "Electricity Market Management Systems (EMMS) Web Portal Applications", provides a summary of EMMS Web Portal applications, to assist participants with decisions about AEMO's IT systems.
- "Guide to Market Systems – Maintaining and extending Access", provides high-level, summary information about AEMO's IT systems, to assist participants with decisions about usage of the data interfaces to AEMO's systems.
- "Web Portal Login User Guide": for information on how to log on to AEMO's Web Portals.

8.3 Information centre

The AEMO Information Centre provides an information service for all interested parties, from NEM participants to the general public, providing information regarding AEMO NEM operations and the electricity industry generally.

- Telephone: 1300 361 011. E-mail: infocentre@aemo.com.au.

9 Needing Help?

Watch out for market notices and bulletins about network changes potentially affecting access, but unplanned events do occur. In preparation for handling network and system failures, plan and test alternate configurations (including failovers).

The following may help you resolve your problem, before logging a call with AEMO's Help Desk.



Note: for further resolution issues, see relevant product user guides (see "Data Interchange documentation set", on page 10).

9.1 Check system status

Either, or both:

- Check the relevant Performance Monitor (for IP addresses, see document "IP Addresses for Participants").
- Call the AEMO Help Desk (1300 300 295) and select option 1, which provides you with up-to-date information on the status of the AEMO systems, including MarketNet.

9.2 Confirm no issues with organisation's internal network

A number of participants transfer market information across their internal communications infrastructure. Confirm with your organisation's internal IT department to ensure your organisation's internal network is not contributing to the problem being experienced.

9.3 Confirm no issues with external service provider

A number of participants contract external service providers to implement and manage the data communications between their organisation and AEMO. Confirm with your service provider to ensure their connection to the MarketNet is operating correctly. Please encourage your service provider to notify the AEMO Help Desk of issues at their end. This enables AEMO to provide accurate information to other participants.

9.4 Java errors

- Check your release of pdrBatcher and pdrLoader. Are they the latest?
- Check your Java version.

9.5 Not connecting or locked out?

- Check credentials (passwords do expire).
- Check successful connection using a command-line FTP client from the same environment as where pdrBatcher runs.
- Check using passive FTP, since active FTP is not as reliable.
- Check access to the EMMS file server shares by pdrBatcher is Read/Write.

9.6 Files not transferring, or timing out?

- Check network configuration and performance, including firewalls.
- Try disconnecting and reconnecting.
- Modify properties to suit available network conditions.
- Check with local networking, IT and operational support.

9.7 Missing history

- Check participant implementation has file recovery.
- Is file recovery enabled (in pdrLoader)?
- Is local processing of selected files disabled (check the PDR tables)?
- Check subscriptions in the system used (pre-production or production).
- Is there a build-up of files in a DI folder (locally or remote, to or from AEMO)?
- Check manifest request files are being produced by pdrLoader.
- Are the manifest request files being delivered to the EMMS file server?
- Within a minute or so after a manifest request file is delivered to AEMO, check to see if the file has been removed. If not, is the folder correct?
- When the file has been removed from the EMMS file server by the EMMS, then check to see there is at least one file returned in the trickle feed folder on the EMMS file server. Is the trickle feed configured all the way through to the pdrLoader?

9.8 Getting file, but missing report

- Check subscriptions (remembering that pre-production is occasionally refreshed from production, as notified by e-mail).
- Reports are for changed data; has anything changed to cause report to appear?

9.9 VPN down

If your VPN is down, as a workaround, load files to the DI folders from the public data on AEMO's website, <http://www.aemo.com.au/data/csv.html> (latest release only).

9.10 Observing abnormal behaviour?

- Is release of each product the latest?
- Check logs; any error indicates local IT action is necessary.
- Check with local support, including networks, IT operations, and IT maintenance for any recent changes to any part of the DI environment. Can you confirm there are no recent changes?
- Does the database instance conform to the MMS Data Model?
- Are there any modifications to the databases receiving the data from each pdrLoader?
- Is the error reproducible in a test environment where all AEMO-supplied products have no modifications to the basic install? Can you supply copies of logs and .properties files from the test environment?

9.11 Logging a call with AEMO Help Desk for support

When requesting support for Data Interchange, provide at least (after checking each thoroughly):

- Version of DI Product.
- Logs showing the abnormal behaviour (zipped).
- .properties files.
- Diagram of architecture.