

Table of Contents

PPDM Disclaimer _____	4
About This Document _____	5
Introduction _____	7
Before You Start _____	8
<i>About the PPDM Lite 1.0 Data Store</i> _____	8
<i>About Spatially Enabling PPDM Lite 1.0</i> _____	8
<i>About the Scripts and Procedures</i> _____	9
Create PPDM Lite: A Step by Step Guide _____	9
<i>Step 1. Create a PPDM Lite 1.0 Database</i> _____	10
<i>Step 2. Load PPDM Lite 1.0 from PPDM 3.7</i> _____	10
<i>Step 3. Create Synchronization Procedures</i> _____	11
<i>Step 4. Create Triggers to Initialize the Synchronization Procedures</i> _____	12
<i>Step 5. Create Java Programs to Generate GIS Objects</i> _____	13
<i>Step 6. PL/SQL Spatial Wrappers</i> _____	16
<i>Step 7. Create Triggers on Spatial Tables</i> _____	16
<i>Step 8. Create a Sample Geodatabase</i> _____	17
Model Overview _____	19
<i>Integration</i> _____	19
<i>Data Diagrams</i> _____	20
<i>Tables and Columns</i> _____	22
<i>Application</i> _____	23
<i>Area</i> _____	23
<i>Business Associate</i> _____	24
<i>Contracts</i> _____	24
<i>Entitlements</i> _____	25
<i>Facility</i> _____	25
<i>Financial</i> _____	26
<i>Field/Pool</i> _____	27
<i>Interest Sets</i> _____	28
<i>Land</i> _____	28

<i>PPDM Meta Data</i>	29
<i>Projects</i>	30
<i>License</i>	31
<i>Production Volumes</i>	31
<i>Records Management</i>	32
<i>Reserves</i>	33
<i>Seismic</i>	34
<i>Spatial</i>	35
<i>Support Facility</i>	36
<i>Stratigraphy</i>	37
<i>Well</i>	38
Implementation Considerations	39
<i>Load of the Rings</i>	39
<i>Constraints in PPDM Lite 1.0</i>	39
<i>Check Constraints</i>	39
<i>Units of Measure</i>	40
<i>PPDM GUID</i>	40
<i>Modifying PPDM Lite 1.0</i>	40
<i>Frequently Asked Questions (FAQ)</i>	41
Appendix A: PPDM Lite 1.0 Rings	43

PPDM Disclaimer

USER agrees on receipt of the PPDM model and supporting material that its use by USER constitutes acceptance by the USER that:

1. The PPDM products and supporting material are provided “as is” without any warranty or condition of any kind, either statutory, express or implied, including, without limiting the foregoing, implied warranties of merchantability of fitness for a particular purpose, or warranty as to non-infringement with existing or future copyright, patent or other intellectual property rights claimed or asserted by any third party;
2. PPDM shall not be liable to User nor to anyone else who uses or otherwise obtains the PPDM products and supporting material, either directly or indirectly from User, for direct, indirect, special or consequential damages or other loss or cost however arising;
3. User agrees to indemnify and hold PPDM harmless from any third party claims arising directly or indirectly, or however arising, from the PPDM products and supporting material provided to the User;
4. The user acknowledges that the PPDM products and supporting material are under copyright notice and that all rights are reserved by PPDM, the Public Petroleum Data Model Association.

About This Document

This reference guide has been prepared to help managers, analysts, database administrators, programmers and data managers understand how to implement PPDM Lite 1.0. Readers at many levels, from managerial to technical implementers will benefit from reading various sections of this document. General, high-level business information is contained at the beginning of the document, with each section becoming progressively more technical and detailed.

Sometimes the terms we use in this and other PPDM documents need to be defined. We provide definitions in a separate Glossary, which you can obtain from PPDM.

This reference guide contains the following sections:

- Introduction
Provides an executive overview of PPDM Lite 1.0.
- Business Process Overview
Summarizes PPDM Lite 1.0 and provides examples of related business processes.
- Integration
Discusses how PPDM Lite 1.0 is integrated with PPDM 3.7 modules and provides information about related references guides.
- Model Overview
Includes the data diagram in the Data Model.
- Tables and Columns
Identifies the data model tables for PPDM Lite 1.0, how they should be used, what they contain, and recommends how they should be used. This section should be used in conjunction with the PPDM Table Report available for download from the PPDM Web Site (www.ppdm.org).
- Implementation Considerations
Discusses issues related to implementing the PPDM Lite 1.0 model, architectural methodologies used in design, or special considerations for implementation that are not related to a specific table.
- Frequently Asked Questions

An FAQ that addresses technical and business questions about the PPDM Lite 1.0.

- Appendix A – PPDM Lite 1.0 Rings

This is a listing of all tables within PPDM Lite and their load order.

Introduction

The PPDM 3.7 data model, released to the PPDM Membership in spring 2004, contains over 1,200 tables and 24,000 columns covering 45 subject areas in the energy sector. This robust and comprehensive data model is very effective for managing data and processes in transactional systems, or for use as a corporate master data store.

However, for some functions, such as management decision support systems, or GIS browsing applications it is more practical to present a warehouse style data summary to users. In 2004, the PPDM membership recommended that this kind of high level, rolled up version of PPDM 3.7 be designed.

The resulting light weight model, released publicly, is PPDM Lite 1.0. In some implementations, PPDM Lite 1.0 may be thought of as a simple warehouse or data mart model. If desired, PPDM Lite 1.0 may be spatially enabled using a GIS system. For most implementations, PPDM Lite 1.0 will be maintained through automated processes from PPDM 3.7. Synchronization procedures are provided to members of the PPDM Association.

The PPDM Association provides examples illustrating how to spatially enable PPDM Lite using ESRI, Oracle and Postgres, but the reader should be aware that there are other spatial applications that could be used effectively. The PPDM Association does not recommend any specific set of GIS tools; the examples are provided simply to illustrate possible implementation scenarios.

Every implementation of PPDM Lite 1.0, whether spatially enabled or not, should be designed to meet the needs of the user community within the company, the requirements of corporate data management recommendations and the existing technical environment.

The seven steps listed in this document will guide you through implementing PPDM Lite 1.0. Many implementation levels are possible, depending on your business needs. To choose the appropriate level an understanding of the business drivers in your company is needed. These scenarios range from implementing PPDM Lite 1.0 as a standalone warehouse without a GIS to a full implementation of PPDM Lite that is replicated from your PPDM 3.7 master data store and spatially enabled through a GIS such as ArcIMS.

Some implementations may not require all the steps in this guide. For example, an implementation that periodically refreshes PPDM Lite 1.0 from your master PDM 3.7 data store (rather than handling the synchronization in real time) requires steps 1, 2, 5, 6 and 7. In some cases, companies may choose to implement a stand alone version of PPDM Lite 1.0 without any managed replication from a master data store. For this case, step 1, followed by the spatial enabling processes in Step 7 are all that is needed. (Please note that the PPDM Association strongly recommends that all data replication should always be managed.)

This document will inform the reader about what is available in PPDM Lite 1.0, and help you decide what modules and processes are appropriate for your implementation.

Before You Start

About the PPDM Lite 1.0 Data Store

Many organizations will use PPDM Lite 1.0 to create a high level summary data store that is replicated from existing master data stores. In practice, it's likely that data from many master data stores (land, wells, production etc) will be used to populate PPDM Lite 1.0.

The PPDM Association and its work groups recommend, in the strongest possible terms, that data replication be managed. This means that the data replication should be automated to the fullest extent possible and never conducted manually (if data in the master data store is updated, PPDM Lite 1.0 should be updated automatically).

Our recommendation is that PPDM Lite 1.0 should not be considered to be a transactional system, but treated only as a reporting system. Data corrections or additions should be made to the master data store and then replicated into PPDM Lite 1.0 through automated processes. These replication processes may run in real time (as they occur) or periodically (nightly or hourly).

Before you begin this process, develop a complete data management plan that will clarify how these replication processes will be managed.

About Spatially Enabling PPDM Lite 1.0

When you spatially enable PPDM Lite 1.0, you will have many decisions to make. Among the most important is the decision about how to create spatial layers or features.

PPDM Lite generalizes the storage of geodetic location according to geometry type. If you spatially enable a well, for example, and choose to spatially enable the well as a linear reference system using the directional survey, the locations for the well will be in L_SP_LINE_POINT. If you choose the spatially enable wells using only the surface or base hole location, the locations will be stored in L_SP_POINT.

Decide what geometry type(s) you would like to capture for each business object type. There are a variety of reasonable business requirements that might encourage you to create more than one type of geometry for a business object. Each geometry type for a business object type should be defined as a new spatial layer in your GIS system.

While it is possible to create a single spatial layer for each geometry type, it will probably not be practical to do this. Each business object type typically has different behavior and descriptive attributes. Combining more than one type of information in a single layer can cause user confusion, frustration and dissatisfaction. We recommend that each type of business object be defined as a separate feature, and different geometries for the same business object be created as separate features.

About the Scripts and Procedures

The procedures provided in the sample scripts are mostly written for Oracle databases; they should be considered to be sample scripts or examples of what each step will do. You will need to convert the syntax of each process to work with whichever database and application suite you are using.

Please do not treat these procedures as if they were vendor provided software. They should be considered guidelines or samples only. We expect your implementation team to review and modify the scripts to suit your needs.

Note that the PPDM Association provides very limited advice and guidance for technical implementation issues. Problems and questions should be directed to the PPDM discussion lists (spatial-1@ppdm.org); we urge industry to collaborate and cooperate with implementation issues.

Create PPDM Lite: A Step by Step Guide

The steps in this reference guide have been numbered to correspond to the products that are available on the PPDM Spatial download page (http://www.ppdm.org/standards/spatial/ppdm_lite.html).

As you download and review each section of sample code, you should review the appropriate section of this guide.

Step 1. Create a PPDM Lite 1.0 Database

Description

Any company may download the PPDM Lite 1.0 DDL and create a high level E&P database containing over 20 modules that summarize many of the business objects that an energy company needs to track.

PPDM Lite 1.0 was designed as a light weight warehouse or rolled up data model that summarizes key information in the full PPDM 3.7 (PPDM 3.7 is available to members of the PPDM Association). PPDM Lite provides users with a simple, easy to understand data set that contains key information commonly used by managers, GIS users and decision makers.

PPDM Lite 1.0, with a GIS user interface, may be thought of as a Spatial Data Warehouse. Used as a summarized data source with a web based dashboard application, PPDM Lite 1.0 may be considered a Data Warehouse.

Key Business Processes

Successful implementation of PPDM Lite requires some workflow analysis to be done. With over 20 modules that can be implemented, an implementation plan detail how data will be captured and how data will be used and/or displayed. There are many reasons to implement PPDM Lite; either as a standalone system or as a front-end to a PPDM 3.7 transactional Master Data Store.

Decide where your master data store for each data type exists and develop a plan to manage replication of key data in PPDM Lite 1.0.

Download the PPDM Lite DDL and create a data base by running the DDL scripts provided. You should be able to create PPDM Lite in any SQL 92 compliant database, including Oracle, SQL*Server, MySQL or Postgres.

Step 2. Load PPDM Lite 1.0 from PPDM 3.7

Description

Since PPDM Lite 1.0 is a subset of PPDM 3.7, the scripts in this step may be run to load the PPDM Lite 1.0 tables. These scripts were written to extract appropriate information from PPDM 3.7 and load it into PPDM Lite 1.0.

After you have done the initial data population, you will need processes to keep PPDM Lite 1.0 synchronized with PPDM 3.7; these processes are documented in steps 3 & 4.

Key Business Processes

Implementation decisions will define how the population of PPDM Lite 1.0 will be done. Most implementations will start with an initial load of data from the master data store. This set of processes will help you prepare for this initial population.

Since PPDM Lite 1.0 is a replicated system, it assumes that data validation and quality control have been done in the master data store. This means that the data coming into PPDM Lite 1.0 should be valid and correct. PPDM Lite 1.0 does not contain reference tables, since this level of validation should have already been completed in the master data store.

1. Review the source data and be sure you understand where it will come from and how it will replicate.
2. Check the data quality level in the master data store and determine whether it will meet the needs of your users. A spatially enabled PPDM Lite 1.0 will often provide users with a useful data review mechanism, but we recommend that any data corrections be made to the master data store, and not to PPDM Lite 1.0.
3. Review the scripts provided and customize them as needed.
4. Run the final scripts to complete the initial load of data into PPDM Lite 1.0
5. If you decide to synchronize PPDM Lite 1.0 with your master data store(s) on a scheduled basis, rather than in real time, schedule your final set of load scripts to be run according to the schedule. In this case, all data in PPDM Lite 1.0 should be deleted prior to the refreshing load.

Step 3. Create Synchronization Procedures

Description

If your business drivers are such that you can or must implement a system where the PPDM Lite 1.0 is synchronized with your master data store (PPDM 3.7) in real time, use steps 3 and 4.

Step 3 contains the PL/SQL procedures that will move the data from your transaction system to PPDM Lite in real time (as they are made to the master data store). Each procedure is called from the Insert, Update and Delete triggers that are attached to the PPDM 3.7 tables.

If you choose to update PPDM Lite 1.0 using a schedule, you can call these procedures independently through a batch job or similar process.

Key Business Processes

Your implementation will be determined by how important is it at your company to have information in PPDM Lite 1.0 synchronized in real time from your master PPDM 3.7 system. If data updates must be immediate, then step 4 should be implemented. If data updates can be done on a scheduled basis, you can leave out step 4, and conduct the updates according to your schedule.

These procedures consist of a set of procedures that reside in PPDM 3.7; they are used to synchronize data changes with PPDM Lite 1.0. Any time a relevant table experiences an update, insert or deletion, the appropriate trigger will invoke a procedure that will synchronize the change in PPDM 3.7 with PPDM Lite 1.0.

1. Review the scripts provided and customize them as needed.
2. Run the final scripts to set PPDM 3.7 to supply data updates to PPDM Lite 1.0.

Step 4. Create Triggers to Initialize the Synchronization Procedures

Description

When data is inserted, updated or deleted in your PPDM 3.7 transactional system (to the tables that are the master store for PPDM Lite 1.0, these triggers will invoke the procedures created in Step 3. The procedures will automatically update your PPDM Lite 1.0 database to match PPDM 3.7.

Key Business Processes

Two preferred methods for updating PPDM Lite 1.0 have been identified. If you choose to simply re-create PPDM Lite 1.0 from your master data store on a scheduled basis, these triggers and the procedures in Step 3 are not necessary. Instead, you should truncate all the tables in PPDM Lite 1.0 and reload them using the scripts in Step 1.

If you want to update PPDM Lite 1.0 with changes made to PPDM 3.7, either in real time or on a scheduled basis, you should create these triggers. The procedures can be modified to run in real time or on a scheduled basis, as you need.

These procedures consist of a set of table level triggers that reside on the tables in PPDM 3.7 that are used to populate PPDM Lite 1.0.

1. Review the scripts provided and customize them as needed.
2. Run the final scripts to set the triggers that invoke procedures to synchronize PPDM Lite 1.0.

Step 5. Create Java Programs to Generate GIS Objects

Description

For the spatial implementation of PPDM Lite 1.0, this is the step that will actually create the spatial layer objects. It is very important to realize that if this is implemented in a transactional system, each addition or update or deletion of a location (well, seismic survey, facility, etc) will result in this process firing. This could be a huge resource drain on your system or lead to unavailability of the spatial object whilst it is being rebuilt. Consider this implementation carefully. The example list here deals with an SDE setup in an Oracle database.

This set of JAVA programs are used to generate new spatial binary (SDE Binary) objects when location data in PPDM Lite 1.0 is changed. For example, if a well location in spatial tables is changed, the JAVA program deployed in Oracle will generate a new SDE binary spatial object for the changed location when inserting, update spatial row when updating and delete a row when deleting from database

For installation purposes, it is necessary to create or install the triggers and procedures in an order reversed to the sequence in which they will run. Chronologically, when a location (such as a well location) is changed or created, a trigger set on the appropriate table will fire (these triggers are created in step 7). The trigger will invoke a PL*SQL wrapper (created in step 6) that will in turn invoke the appropriate Java procedure (created in this step). The Java procedures are divided into many different samples; each needs to be reviewed and modified as necessary prior to implementation at your site.

Java procedures have been developed for three GIS formats (SDE Binary, Oracle Spatial and Postgres POSTGIS format). Within each of those categories, Java procedures are called to deal with INSERTS, UPDATES and DELETES to each of the spatial objects in PPDM Lite 1.0. As a consequence, there are quite a few procedures.

Please thoroughly review the entire set of procedures before you begin this step. It is useful to deal with the code for Steps 5, 6 and 7 as a unit.

Key Business Processes

Please go over this checklist before you start to make sure that you have all the necessary components installed and have documented the necessary reference information.

Java version j2re1.4.x must be installed

JDeveloper9i must be installed

The SDE API must be installed

Oracle 9i must be installed (if you are using a different RDBMS, use the scripts as a reference and code them to work in your environment).

For your Oracle database, ensure you know

Server name

Port number

Database name

User name and password

For each type of data base change (Insert, update & delete), and for every business object type that is to be spatially enabled, there is a separate Java program that must be reviewed, modified and compiled. The program names follow these conventions:

On INSERT to business table → Step5_TrInsertLWell.java;

On UPDATE to business table → Step5_TrUpdateLWell.java;

On DELETE from business table → Step5_TrDeleteLWell.java.

For each of the files you intend to use, you will need to find and modify certain system variables to correspond to your environment. These variables are listed in the following table, along with the value they contain by default. If you are not sure what the variable value should be for your implementation, please contact your system administrator or DBA.

Variable Type	Variable Name	Existing Value
string	db_server	“spare”
int	db_port	1521
string	db_database	“v92”
string	db_user	“ot37”
string	db_pass	“ot37”

When these variables have been changed, the java code should be compiled in JDeveloper9i. It can then be deployed (load the java class into the database) from JDeveloper9i as well.

This document does not describe how to create or run Java code. Please see your system administrator or developer for assistance as needed. To check to see if your Java code was deployed properly, you may open a SQL*Plus session and issue the following statement:

```
SELECT * FROM USER_OBJECTS
WHERE OBJECT_NAME = 'TrgInsertLWell';
```

The next step is to grant permission to the run the java class in the database. This is usually done by the DBA. The example below grants the Oracle schema of PPDM37 the right to run the java class.

```
BEGIN
DBMS_JAVA.GRANT_PERMISSION(
  grantee => 'PPDM37',
  permission_type => 'SYS:java.net.SocketPermission',
  permission_name => '*',
  permission_action => 'connect');
END;
/
```

Steps required to generate SDE binary

1. A new SDE table needs to be created to store the Binary spatial information
2. An SDE layer (point, polyline, polygon) that creates the feature needs to be created.
3. The new SDE layer is registered with SDE.
4. Java scripts for generating spatial binary is run by retrieving spatial locations from PPDM Lite Spatial tables and inserts spatial objects into SDE table.

Steps required to generate SDO

1. A new SDO table needs to be created to store the Oracle spatial information
2. Insert metadata to SDO table. This consists of min, max values of x, y and z.
3. Register SDO Layer to SDE in order to be recognized and displayed in Arc utility. *** *This step is only needed if you want to recognize and display the SDO through ArcGIS.*
4. Register table to SDE in order to be recognized and displayed in Arc utility. *** *This step is only needed if you want to recognize and display the SDO through ArcGIS.*
5. Run Java Class according to which GIS objects to be created (well, seismic)
6. Create spatial index to perform spatial queries

Steps required to generate PostGIS

1. Create table (ex: create table l_well (uwi varchar(20))

2. Add a spatial column to the table using OpenGIS “AddGeometryColumn” function
3. Run Java Class according to which GIS objects to be created (well, seismic)

Step 6. PL/SQL Spatial Wrappers

Description

These wrappers are in place to call the external java routine from Oracle’s PL/SQL. It is a two step process to complete these as the PL/SQL triggers are unable to call Java directly. These wrappers map the PL*SQL parameters to Java parameters. There is a wrapper for each of the triggers sets; INSERT, UPDATE and DELETE.

Key Business Processes

The decision here must be when or how often to re-create your spatial layer. The trigger is designed to fire out at every DML event. So every time an object, such as change to a well (insert, update or delete) the trigger will fire. As we mentioned before, this will cause a lot of overhead.

Setting up a frequently run batch job might be a better way of implementing your update procedures. You need to decide (with your users) how often updates must be reflected in the replaced PPDM Lite 1.0 system. Each data type may have different refresh requirements. Each implementation at each company will be different, depending upon the business drivers.

The following is an example of how the ‘add_l_well’ procedure is set up.

```

CREATE OR REPLACE PROCEDURE add_l_well
(
  add_line_id          IN VARCHAR2,
  add_point_seq_no    IN NUMBER
)
AUTHID CURRENT_USER
AS LANGUAGE JAVA
NAME 'TrgInsertLWell.TrgInsertLWell (JAVA.lang.String,short)';
/

```

Step 7. Create Triggers on Spatial Tables

Description

This is the last step in the implementation process. It can be called from the PL/SQL wrapper as detailed in the previous step or it can be called directly from a command line.

Key Business Processes

Again, business driven decision must be made about how often each data type should be refreshed. If you are implementing PPDM Lite as a stand-alone system or want to update or create spatial objects for data after it is loaded (step 2), a direct jump to this step can be done.

The actual trigger looks like this:

```
CREATE OR REPLACE TRIGGER aud_l_well
AFTER DELETE OR INSERT OR UPDATE ON L_SP_LINE_POINT
FOR EACH ROW
BEGIN
  IF INSERTING AND :NEW.point_seq_no = 2 THEN
    add_l_well(:NEW.line_id,:NEW.point_seq_no);
  ELSIF DELETING AND :OLD.point_seq_no = 2 THEN
    del_l_well(:OLD.line_id,:OLD.point_seq_no);
  ELSIF UPDATING AND :OLD.point_seq_no = 2 THEN
    upd_l_well(:OLD.line_id, :NEW.line_id, :NEW.point_seq_no);
  END IF;
END aud_well;
/
```

Step 8. Create a Sample Geodatabase

Description

This optional step was included to demonstrate how PPDM Lite 1.0 can be implemented using ESRI tools. There are many ways that this database can be spatially enabled, this is sample representation is intended to serve as an illustration.

Please note that the sample data included in the sample Geodatabase is not very good quality. The data has been artificially shifted in location, and much of the technical data has been altered.

Key Business Processes

Download the MDB file in zipped format above. This Personal Geodatabase has been created using ArcSDE 9.1. It contains all PPDM Lite table with some sample data. The SDE binaries have been generated based on data stored in the spatial tables, which include:

SPATIAL_L_SP_POINT_SDE3 for points

SPATIAL_L_SEIS_POINT_SDE for lines

SPATIAL_L_SP_BOUNDARY_SDE for polygons

Model Overview

Integration

PPDM Lite 1.0 has been designed to be implemented in many different ways, including as a stand-alone database, a GIS enabled database, a GIS front-end to PPDM 3.7 or an integrated section within PPDM 3.7.

Most implementations of PPDM Lite 1.0 will require tight physical integration with a master data store, usually PPDM 3.7. As you plan your project to use PPDM Lite 1.0, you should review the capabilities of the complete master data store PPDM 3.7 to understand what data is available and how it will be summarized in PPDM Lite 1.0.

Occasionally, user will want to access more information out of the master data store than is available in PPDM Lite 1.0. In these cases, it may be most practical to join the master and summary databases together through the use of views, materialized views or query. We urge you to consider the impact of each enhancement to PPDM Lite 1.0 before making them.

The policies, practices and procedures that are in place at your company will determine the level of accuracy and completeness of your master data stores and the currency or timeliness with which they are updated. Take this into consideration and ensure that your customers are aware of any issues or problems that may arise.

Your business analysis, conducted before the project begins, will tell you which of the 20 modules in PPDM Lite 1.0 you should implement. This section describes the models and tables in PPDM Lite 1.0.

Modules

- *Business Associates:* describe people, companies, regulatory agencies, and other stakeholders for the data captured in PPDM Lite 1.0.
- *Contracts:* contractual details may be captured in this module,
- *Entitlements:* information about the security or entitlement rights that you have to any type of data and what you are able to do with it.
- *Facility:* tracks production facilities.
- *Financial:* helps you gather a high level financial summary from your accounting system and link that with relevant business objects.
- *Field/Pool:* Geographically and geologically defined areas that produce hydrocarbons.

- *Interest Sets*: Capture partnership shares of various types, such as working interests, production interests or royalty interests and associates these interests with business objects.
- *Land Rights*: tracks mineral and surface rights to onshore or offshore areas.
- *Meta Model*: information about where the data that resides in the PPDM Lite 1.0 came from. The PPDM Model tracks details about the tables, columns, constraints and uses in PPDM Lite.
- *Projects*: track any set of tasks or activities, such as for field acquisition, interpretation, or processing.
- *Licenses*: Licenses are formal, regulatory approvals to conduct work
- *Production*: Production volume data for various business objects, such as wells, completions, fields, pools etc.
- *Records Management*: track the physical location of digital and hard copy products, circulation, retention, etc.
- *Reserves*: captures information about hydrocarbon production potential.
- *Seismic*: describes the acquisition, processing, interpretation and divesture throughout the life cycle of seismic data.
- *Spatial*: use this module to reference any positional information to geodetic or cartographic information.
- *Stratigraphy*: Describes geologic formations, faults, surfaces and other structures. In PPDM Lite 1.0 this module is used to provide reference lists of valid formation and faults.
- *Support Facility*: describes facilities used to support business operations such as roads, transmission towers, docks and drilling platforms.
- *Wells*: describes downhole well details.

Data Diagrams

The diagram on this page is the legend for the tables discussed later in this document. Note that some or all of these elements may be present in data diagrams provided by the Association. Some elements are removed from final products to reduce file size:

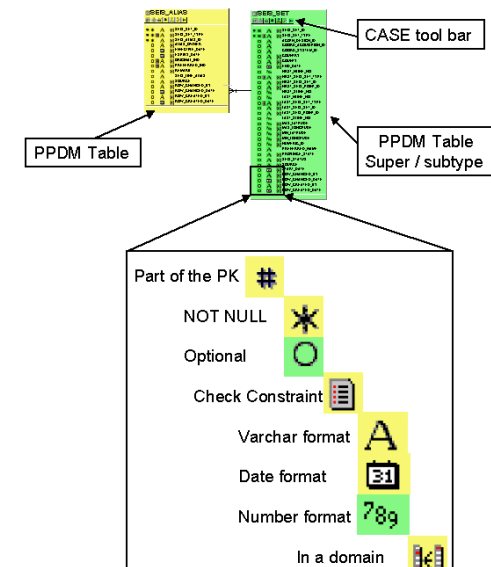


Figure 6: This illustration shows the functions of each icon used in the data diagrams provided with PPDM Lite 1.0.

The data diagrams can be obtained from the PPDM Association as part of the final model documentation or as a set of PowerPoint diagrams. The PowerPoint diagrams will provide the best resolution for printed quality.

Tables and Columns

The following tables exist in PPDM Lite 1.0. Each table is described in the following section; you can jump to a table description by clicking on the hyperlinked table name below. Note that for detailed content descriptions for each table, you should refer to the PPDM Lite 1.0 table documentation.

[L_APPLICATION](#)

[L_AREA](#)

[L_BUSINESS_ASSOCIATE](#)

[L_BA_ALIAS](#)

[L_CONTRACT_COMPONENT](#)

[L_CONT_EXTENSION](#)

[L_ENT_GROUP](#)

[L_FACILITY](#)

[L_FACILITY_ALIAS](#)

[L_FINANCE](#)

[L_FIN_COMPONENT](#)

[L_FIELD](#)

[L_POOL](#)

[L_INTEREST_SET](#)

[L_INT_SET_PARTNER](#)

[L_LAND_XREF](#)

[L_LAND_SALE_OFFERING](#)

[L_PPDM_DATA_SOURCE](#)

[L_PPDM_ORIGIN](#)

[L_PPDM_COLUMN](#)

[L_PROJECT_ALIAS](#)

[L_LICENSE](#)

[L_PDEN_STRAT_UNIT](#)

[L_PDEN_XREF](#)

[L_RM_INFORMATION_ITEM](#)

[L_RM_INFO_ITEM_DESC](#)

[L_RESENT_CLASS](#)

[L_RESENT_XREF](#)

[L_SEIS_ALIAS](#)

[L_SEIS_RECORD](#)

[L_AREA_ALIAS](#)

[L_BA_ADDRESS](#)

[L_CONTRACT](#)

[L_CONT_ALIAS](#)

[L_ENTITLEMENT](#)

[L_ENT_COMPONENT](#)

[L_FACILITY_SUBSTANCE](#)

[L_FACILITY_XREF](#)

[L_FIN_COST_SUMMARY](#)

[L_FIN_XREF](#)

[L_FIELD_ALIAS](#)

[L_POOL_ALIAS](#)

[L_INT_SET_COMPONENT](#)

[L_LAND_SALE](#)

[L_LAND_RIGHT](#)

[L_LAND_ALIAS](#)

[L_PPDM_EXCEPTION](#)

[L_PPDM_TABLE](#)

[L_PROJECT](#)

[L_PROJECT_COMPONENT](#)

[L_PDEN](#)

[L_STRAT_UNIT](#)

[L_PDEN_VOL_SUMMARY](#)

[L_RM_INFO_ITEM_CONTENT](#)

[L_RESERVE_ENTITY](#)

[L_RESENT_COMPONENT](#)

[L_SEIS_SET](#)

[L_SEIS_POINT](#)

[L_CS_COORDINATE_SYSTEM](#)

L_SP_COMPONENT	L_SPATIAL_DESCRIPTION
L_SP_POLYGON	L_SP_BOUNDARY
L_SP_LINE	L_SP_LINE_POINT
L_SP_MINERAL_ZONE	L_SP_ZONE_SUBSTANCE
L_SP_GEOMETRY	L_SP_POINT
L_SF_SUPPORT_FACILITY	L_SF_COMPONENT
L_SF_XREF	L_SF_DESCRIPTION
L_SF_ALIAS	L_STRAT_NAME_SET
L_STRAT_WELL_SECTION	L_WELL
L_WELL_ALIAS	L_WELL_SERVICE

Application

L_APPLICATION

This table captures requests for authorization for an activity. Applications may be made for AFE's, to regulatory agencies or to partners to conduct operations according to the terms of a contract. Includes the type of application made, major dates (submitted and received), the decision and any remarks about the application.

[Back to the list of table names](#)

Area

L_AREA

An area can be any geographic place, including project areas, organizational areas or environmentally restricted areas. High level information about the area type, the preferred name as well as remarks.

[Back to the list of table names](#)

L_AREA_ALIAS

Alias table that contains what the area is called and what it could be or is called in other contexts. All names, codes and identifiers (including the preferred name) for an area should be stored in this table.

[Back to the list of table names](#)

Business Associate

L_BUSINESS_ASSOCIATE

A business associate may be any company, agency or person with whom you do business. This may be considered to be a contact data base, and would normally be managed from a central corporate database.

[Back to the list of table names](#)

L_BA_ADDRESS

The address table contains physical, postal or billing addresses. Companies that have more than one branch may have more than one valid address.

[Back to the list of table names](#)

L_BA_ALIAS

Alias tables track all names, codes and identifiers for each business associate. This table allows useful functionality for linking company information between multiple applications, including your financial and billing systems and your geotechnical applications. All names, codes and identifiers should be stored in this table, including current and out dated versions.

[Back to the list of table names](#)

Contracts

L_CONTRACT

The contract table may be used to reference contracts of any type. This is a high level table that contains contract name, status, date and penalty percentages.

[Back to the list of table names](#)

L_CONTRACT_COMPONENT

Cross reference table that associates contracts with all the business objects (wells, land rights, seismic data) that the contract relates to.

[Back to the list of table names](#)

L_CONT_ALIAS

Alias table that is used to track what a contract is called or is to be referenced as by various partners, systems or applications. Out dated aliases may also be

tracked in this table. All names, codes and identifiers should be stored in this table, including current and out dated versions.

[Back to the list of table names](#)

L_CONT_EXTENSION

Most often used for the extension of land right related contracts. If the contract was extended, use this table to capture the type of extension, and associated terms of the extension.

[Back to the list of table names](#)

Entitlements

L_ENTITLEMENT

Entitlements describe the rights you have to data or information, such as a well, facility, report or area. These entitlements may be captured for any business object or table in PPDM Lite 1.0.

[Back to the list of table names](#)

L_ENT_GROUP

Use this table to define groups of users or business associates who have a common set of entitlements. Often, a group may be departmental, functional or organizational.

[Back to the list of table names](#)

L_ENT_COMPONENT

Cross reference table that allows specific business objects such as wells or facilities to be associated with a specific set of rights or entitlements.

[Back to the list of table names](#)

Facility

L_FACILITY

A facility is a physical construct that is directly associated with the production, management or processing of hydrocarbons. A facility may be a pipeline, storage tank, AV unit, separator, valve etc. This table contains very general facility data such as operator, type, production date, status and status type.

[Back to the list of table names](#)

L_FACILITY_SUBSTANCE

This table captures information, varied over time, about what substances a facility is capable of handling and what its average, minimum and maximum capacities are.

[Back to the list of table names](#)

L_FACILITY_ALIAS

Alias table that is used to track what a facility is called or is to be referenced as by any business associate or application. All names, codes and identifiers should be stored in this table, including current and out dated versions.

[Back to the list of table names](#)

L_FACILITY_XREF

Shows how two facilities are related to each other. This table may be used to capture physical connections between facilities, such as connections between pipelines and other pipelines or facilities.

[Back to the list of table names](#)

Financial

L_FINANCE

This table will allow you to associate your PPDM Lite 1.0 implementation with your accounting systems; in this way information about finances may be drawn directly from your accounting system. Defines the type of financial transaction, the budgeted and actual costs, the issue date and if there is a tax credit available for it.

[Back to the list of table names](#)

L_FIN_COST_SUMMARY

Allows the capture of multiple costs to a single financial cost centre(AFE). Each cost breakdown may be associated with a specific business object or process using the L_FIN_COMPONENT table.

[Back to the list of table names](#)

L_FIN_COMPONENT

Use this table to associate the overall finance record or a specific component with a specific business object.

[Back to the list of table names](#)

L_FIN_XREF

Use this table to track the relationships between financial systems, cost center numbers, AFE's or other financial information.

[Back to the list of table names](#)

Field/Pool

L_FIELD

High level table of the field name, type, the geographic boundaries it is confined in and the discovery date.

[Back to the list of table names](#)

L_FIELD_ALIAS

Alias table that is used to track what a field was called or is to be referenced by. All names, codes and identifiers for the field should be stored in this table, including current and out dated versions.

[Back to the list of table names](#)

L_POOL

Pool descriptions are usually set by regulatory agencies; this table may be used to define the owning field, related stratigraphic formations and the geographic boundaries.

[Back to the list of table names](#)

L_POOL_ALIAS

Alias table that is used to track what a pool was called or is to be referenced by. All names, codes and identifiers should be stored in this table, including current and out dated versions.

[Back to the list of table names](#)

Interest Sets

L_INTEREST_SET

Interest sets, also called partnerships or division of interests, are captured in this table. Any kind of interest set may be stored in this table, including working interests, royalty interests, production interests and construction interests. Interest sets version over time using the INT SET SEQ NO in the full version of PPDM. In PPDM Lite, only the most up to date version of the interest set is stored.

[Back to the list of table names](#)

L_INT_SET_PARTNER

Tracks who your partner(s) are, what role each has and what the gross & net interests are. As with the interest set, only the most up to date version of partner shares are managed in PPDM Lite 1.0. Full information is captured in PPDM 3.7.

[Back to the list of table names](#)

L_INT_SET_COMPONENT

This table is used to tie the interest set back to the business objects, such as well, seismic, land (onshore or offshore) and facilities that they govern.

[Back to the list of table names](#)

Land

L_LAND_RIGHT

A land right represents the rights you have to a geographic area, whether it is onshore or offshore and whether the rights are owned outright or obtained through contractual or lease arrangements with the owner. Land rights may include rights only to the surface or to the subsurface (mineral zones). Header information captured about the land right such as type, status, primary term, cost center are included in this table.

The column LAND RIGHT TYPE is controlled by a check constraint. Allowed values correspond to allowed values in PPDM 3.7; the PPDM Association recommends that these values be left as published, so that replication processes are not compromised. Allowed values are LAND_TITLE, LAND_AGREEMENT, LAND_AGREE_PART, LAND_UNIT, and LAND_UNIT_TRACT.

[Back to the list of table names](#)

L_LAND_ALIAS

Alias table that is used to track what a land right was called or is to be referenced as. All names, codes and identifiers should be stored in this table, including current and out dated versions.

[Back to the list of table names](#)

L_LAND_XREF

The relationships between land rights are captured here. Typically, these relationships may include associations between a land title and the leases that have been granted on that land, leases and subleases, tracts or splits.

[Back to the list of table names](#)

L_LAND_SALE

Land sales capture information about geographic areas (sometimes including stratigraphic or substance definitions) that are made available by offering to the E&P industry. Land sale bids may be based on financial offerings, work promises or both. This table captures information such as the opening, closing and sale dates, the jurisdiction offering it and remarks about it.

[Back to the list of table names](#)

L_LAND_SALE_OFFERING

In a land sale, one or more offerings may be presented for bidding. Each offering may consist of one or more parcels, which may or may not be geographically contiguous. Each offering is bid on as a whole, and granted as a whole.

[Back to the list of table names](#)

PPDM Meta Data

L_PPDM_DATA_SOURCE

Captures the database(source) that is the master data store for a table in PPDM Lite 1.0.

[Back to the list of table names](#)

L_PPDM_EXCEPTION

This table is used by the data management routines to capture load errors at run time. After any data replication is done, this table should be reviewed to ensure that the load was successfully completed.

[Back to the list of table names](#)

L_PPDM_ORIGIN

Table that tracks the exact row in the master data store that the PPDM Lite data came from. As a result of this detail, the table may be quite large. It is powerful in that you can drill back very precisely to the place in the master data store where the data came from.

[Back to the list of table names](#)

L_PPDM_TABLE

A meta-data table which contains information regarding the tables contained within the PPDM Lite schema.

[Back to the list of table names](#)

L_PPDM_COLUMN

Contains meta-data regarding the columns in the PPDM Lite schema. This includes data type and primary key information.

[Back to the list of table names](#)

Projects

L_PROJECT

A project is any set of tasks or activities that are managed. The project name, type, status and start/completion date are capture here. PPDM 3.7 contains a great deal of information about projects.

[Back to the list of table names](#)

L_PROJECT_ALIAS

Alias table that is used to track what a project was called or is to be referenced by. All names, codes and identifiers should be stored in this table, including current and out dated versions.

[Back to the list of table names](#)

L_PROJECT_COMPONENT

This table is used to associate the project with the business objects in PPDM Lite 1.0 that are associated with the project.

[Back to the list of table names](#)

License

L_LICENSE

A license is a formal authorization to conduct operations, and is usually granted by a regulatory authority. This is a rolled up table that contains a summary of detailed license information from the following tables in PPDM 3.7: BA_LICENSE, FACILITY_LICENSE, SEIS_LICENSE and WELL_LICENSE.

The primary key is two part: the first part is a unique License ID, which is typically derived from the source table. The second is the name of the table in PPDM 3.7 that is the master for the row of data. Allowed values are BA_LICENSE, FACILITY_LICENSE, SEIS_LICENSE and WELL_LICENSE.

[Back to the list of table names](#)

Production Volumes

L_PDEN

A PDEN (Producing entity) can be used to capture production information for a variety of producing objects. A PDEN can be categorized as a well, production string, production string completion, pool, field, land right, business associate (such as a partner) or geographic area (such as a county). It represents any entity for which production can be reported against.

The column PDEN TYPE is controlled by a check constraint. Allowed values correspond to allowed values in PPDM 3.7; the PPDM Association recommends that these values be left as published, so that replication processes are not compromised. Allowed values are PDEN_OTHER, PDEN_WELL, PDEN_RESENT_CLASS, PDEN_RESENT, PDEN_PROD_STRING, PDEN_PR_STR_FORM, PDEN_POOL, PDEN_LEASE_UNIT, PDEN_FACILITY, PDEN_FIELD, PDEN_BUSINESS_ASSOC, PDEN_AREA.

[Back to the list of table names](#)

L_PDEN_STRAT_UNIT

This table allows a PDEN entity to have multiple stratigraphic units from which it can produce.

[Back to the list of table names](#)

L_PDEN_XREF

This table tracks PDENs when they are cross-referenced to each other. This allows you to capture information about which PDENs are physically connected at the time of reporting, or new PDENs that replace old PDENs.

[Back to the list of table names](#)

L_PDEN_VOL_SUMMARY

This table allows you to capture volumes that are associated with any activity (sales, flaring, production, injection, loss, fuel use etc) for any time period (monthly, yearly) and for any method (estimated, actual, forecast).

Note that while PPDM 3.7 supports the management of amended volumes, PPDM Lite 1.0 will capture only the most current reported production volumes. Amendment histories should be stored in your full version of PPDM 3.7. Volumes for the most commonly reported substances are captured in this table, including oil, gas, sulphur, water, BOE, CO2 and Nitrogen.

[Back to the list of table names](#)

Records Management

L_RM_INFORMATION_ITEM

The records management module is used to capture information about products and how they are managed. Although unstructured documents such as reports and images are the most commonly managed in this module, other products such as equipment and fossils may also be managed. The L_RM_INFORMATION_ITEM table captures details about the information contained in a product – specifically, what the product is in business terms.

This table captures information such as the name of the item, the kind of item it is, the media that it is stored on (tape, film, paper etc) and who the authors are.

The column INFO_ITEM_TYPE is controlled by a check constraint. Allowed values correspond to allowed values in PPDM 3.7; the PPDM Association recommends that these values be left as published, so that replication processes are not compromised. Allowed values are .

[Back to the list of table names](#)

L_RM_INFO_ITEM_CONTENT

This table will capture information about the business objects that are related to a product, such as the wells, seismic sets, facilities, production reports and so on.

[Back to the list of table names](#)

L_RM_INFO_ITEM_DESC

This table is very vertical in nature, and will allow any descriptive information to be captured for a product. Captures any detailed description data about the record.

[Back to the list of table names](#)

Reserves

L_RESERVE_ENTITY

A reserve entity represents any single producing well (or well completion or string) or group of wells that are used to determine reserves. Stores the product type, created by and update schedule for the reserves analysis. Each reserve entity is qualified by the primary product being produced.

[Back to the list of table names](#)

L_RESENT_CLASS

For each reserve entity, reserve estimates may be defined as categorized by the reserve classes (proven, unproved, developed, undeveloped etc). For each reserve entity, captures the confidence factor (90 % probable reserves) and effective date of a reserve estimate.

[Back to the list of table names](#)

L_RESENT_COMPONENT

This table associates the reserve entity with all the wells, production strings and production string formations that are part of the reserve entity.

[Back to the list of table names](#)

L_RESENT_XREF

This table may be used to define the relationships between reserve entities, usually when one replaces another.

[Back to the list of table names](#)

Seismic

L_SEIS_SET

A seismic set represents a set of seismic data, either an entire survey (sometimes called a project or prospect) or a seismic line. PPDM 3.7 defines several additional types of seismic sets that are not supported in PPDM Lite 1.0.

This table tracks the first & last point in a seismic set, the type of seismic set (controlled by a check constraint), the political jurisdiction that it was shot in and a tie to the Finance module.

The column SEIS SET TYPE is controlled by a check constraint. Allowed values correspond to allowed values in PPDM 3.7; the PPDM Association recommends that these values be left as published, so that replication processes are not compromised. Allowed values are SEIS_INTERP_SET, SEIS_3D, SEIS_SET_PLAN, SEIS_PROC_SET, SEIS_LINE, SEIS_ACQTN_SURVEY, SEIS_WELL, SEIS_SEGMENT.

[Back to the list of table names](#)

L_SEIS_ALIAS

Alias table that is used to track what a seismic set was called or is to be referenced by. All versions of names, codes and identifiers should be loaded in this table.

[Back to the list of table names](#)

L_SEIS_POINT

This table captures the name and location of individual points on a seismic set. These points may exist on the surface (field data, as shots, receivers or both) or in the subsurface (CDP data).

[Back to the list of table names](#)

L_SEIS_RECORD

This table contains information about every record created for every shot on a seismic line. The quality of each resulting record may be tracked. The potential size of this table could be very large; considerable care should be exercised for implementation.

[Back to the list of table names](#)

Spatial

L_SPATIAL_DESCRIPTION

This is the main driving table for the spatial component in PPDM Lite 1.0. This table contains summary information about the type of geometry that has been captured, the coordinate systems used and other important spatial reference information.

[Back to the list of table names](#)

L_SP_BOUNDARY

This table contains the points which outline the perimeter of a polygon. The points must be sequenced spatially in either a clockwise or counter clockwise direction, using the value specified in L_SP_POLYGON.

[Back to the list of table names](#)

L_CS_COORDINATE_SYSTEM

This table contains basic reference information about the coordinate systems referenced by spatial data. It contains a summary of much more detailed data in PPDM 3.7:

[Back to the list of table names](#)

L_SP_COMPONENT

This table is used to identify the business objects that are defined by this spatial description. The relationship supports the business rules that an object may have one or more spatial descriptions and that a spatial description may describe one or more business objects. It would be common for example, for a land right and a contract to be described using the same spatial description.

[Back to the list of table names](#)

L_SP_GEOMETRY

This table provides a method for spatially enabling descriptions through association with appropriate geometries in a spatial engine such as SDE.

[Back to the list of table names](#)

L_SP_LINE

This table stores line data. A line can be described as the linear extent of a spatial object such as a road, pipeline or transmission line.

[Back to the list of table names](#)

L_SP_LINE_POINT

This table contains the points along a line.

[Back to the list of table names](#)

L_SP_MINERAL_ZONE

Definition of the subsurface stratigraphic zone(s) included in the spatial description. This is important for many types of land rights, in which rights are granted to specific (and well defined) stratigraphic zones and often to specific substances in each zone.

[Back to the list of table names](#)

L_SP_POINT

The location of a point geometry.

[Back to the list of table names](#)

L_SP_POLYGON

A polygon is defined as the outline of an area and this table stores this information. Polygons may describe outlines of fields, pools, AOI agreements, land titles, land parcel lots, surface restrictions and others within PPDM Lite 1.0.

[Back to the list of table names](#)

L_SP_ZONE_SUBSTANCE

Describes substances (and their related zone) which are specifically included or excluded from the spatial description. For example, land rights may be granted from surface to basement, except for gas in Zone A.

[Back to the list of table names](#)

Support Facility

L_SF_SUPPORT_FACILITY

A support facility provides operational support for activities. These facilities may include roads, transmission towers, airstrips, vessels, docks and so on. Sub types are used to fully describe each type of support facility.

[Back to the list of table names](#)

L_SF_COMPONENT

This table is used to track relationships between support facilities of all types and other business objects.

[Back to the list of table names](#)

L_SF_XREF

This table is used to capture physical or business relationships between support facilities.

[Back to the list of table names](#)

L_SF_DESCRIPTION

This generic table may be used to track descriptive information about support facilities.

[Back to the list of table names](#)

L_SF_ALIAS

Support facilities may have more than one name, code or identifier. All versions, including the preferred version, should be stored here.

[Back to the list of table names](#)

Stratigraphy

L_STRAT_NAME_SET

A stratigraphic name set is an unordered collection of stratigraphic units, that may be in use for a geographic area, a project, a company etc.

[Back to the list of table names](#)

L_STRAT_UNIT

A stratigraphic unit may be lithologic (a body of rock), chronostratigraphic (defined based on time), an event (unconformity, fault, thrust sheet) or other type of stratigraphy. It is most commonly used for lithostratigraphic units.

[Back to the list of table names](#)

Well

L_WELL

A table for general and header information about a well. A well is an actual or proposed hole in the ground, designed to exchange fluids between a subsurface reservoir and the surface (or another reservoir) or to enable the detection and measurement of rock properties. A wellbore is a cylindrical hole created by a drill. A well may consist of zero, one, or more wellbores; their relationships are described by PARENT_UWI and WELL_RELATIONSHIP. Information from other well tables (e.g. key dates and depths) may be included (or de-normalized) for convenience. The term "well" is used in the column name and comments to mean "well", "wellbore", either, or both (depending on context)..

[Back to the list of table names](#)

L_WELL_ALIAS

The Well Alias table contains names and identifiers that a well may otherwise be known as. This would include previous or alternate well identifiers assigned to the well by a regulatory agency and the reason for the alias.

[Back to the list of table names](#)

L_WELL_SERVICE

This is a rollup table that stores the type of service (core, DST, perf, etc), the date run, the top and base depth and the top and base stratigraphic unit. This table is populated from almost all of the well tables in PPDM 3.7 that have a foreign key from the WELL table.

[Back to the list of table names](#)

L_STRAT_WELL_SECTION

This table contains information on well tops. This includes pick (positional) data on formations, markers, contacts and horizons that can be correlated from well to well within a geographic area.

[Back to the list of table names](#)

Implementation Considerations

Load of the Rings

While all data managers recognize the value of loading data into a data model with referential integrity constraints enabled, this is often viewed as a difficult and onerous undertaking. The PPDM Association recommends that you use Referential Integrity (i.e. don't disable FK constraints) when loading new data.

With any model that is constrained by referential integrity, the trick to start loading this is to find ground zero. PPDM Lite 1.0 is no different. The title of this section came from a discussion held by the Data Management Work Group during the summer of 2004.

The order in which tables must be populated when your foreign keys are enabled has been determined and published (see Appendix A). If you load tables in this order, it will be possible to leave your foreign keys (referential integrity) in place. Eight sets of tables (called Rings) have been defined. Each ring contains tables and columns to be inserted or updated.

Start by populating the tables in Ring 1. When all the tables in Ring 1 are complete, you can load all the tables in Ring 2. Continue loading, Ring by Ring, until all tables are loaded.

Constraints in PPDM Lite 1.0

It is essential that anyone who is considering using PPDM Lite 1.0 be aware that this model does not contain auditing information about any of the data objects as PPDM 3.7 does. It does have the same constraints, where appropriate, as version 3.7. Please review the Constraints Reference Guide for PPDM 3.7 first. Improper use or population of constrained columns in PPDM Lite can compromise the quality of your data and the reliability of your queries. This document may be obtained by members from the PPDM Association or downloaded from the PPDM web site at www.ppdm.org.

Check Constraints

PPDM Lite makes use of the same check constraints as in version 3.7. An example of this is the super-sub type example given next.

- Super-sub type implementations use check constraints to enforce the integrity of the super-sub type relationship. Currently these relationships are in use for Seismic, Business Associates, Records Management, Support Facilities, Production Entities and Land Rights.

Let's use Seismic Sets as an example. This structure consists of a parent table (SEIS_SET) and several sub-type tables (SEIS_3D, SEIS_ACQTN_SURVEY, SEIS_INTERP_SET, SEIS_LINE, SEIS_PROC_SET, SEIS_SEGMENT, SEIS_SET_PLAN and SEIS_WELL). Each of the tables has a two-part primary key: SEIS_SET_ID and SEIS_SET_TYPE.

SEIS_SET_ID is assigned by the user and can have any value as long as it is unique for that type of seismic set. SEIS_SET_TYPE was designed to maintain the integrity of the super-sub type structure and can only have the values assigned to it by check constraints; these values are the table names of the eight valid sub-types. In SEIS_SET, the SEIS_SET_TYPE can have any of the table names, but in each of the sub-types, it can only have the name of the table it is owned by.

Units of Measure

Throughout PPDM Lite all values are to be of the same Unit of Measure. This requires any modification of data be done at load time. This makes querying of the data set easier and more reliable.

PPDM GUID

The Global Unique Identifier (GUID) has been added to every table in PPDM Lite. Applications that are designed to take advantage of this column should implement the sample code to alter the PPDM_GUID column to be NOT NULL and add a Unique Index to each PPDM_GUID column.

Modifying PPDM Lite 1.0

Subsetting PPDM

The PPDM Lite 1.0 data model is designed to allow users to implement portions that support their business without needing to manage modules that are not required. Good data management practices are also supported; this means that data redundancy is reduced in the Model whenever possible.

All information about Seismic will be found in the seismic module; information about contracts is stored in the Contracts module, details about objects that are retained for long term use are stored in the Records Management module and so on. Depending on your business requirements, you can implement all or some of the modules.

In general, it is usually simplest to install the entire PPDM data model and simply restrict usage to the portions that are useful to you. Additional tables can be

implemented as your business requirements expand, or as your data and processes are able to support capture in a data model.

Feedback to PPDM

Much of the growth of the PPDM model can be attributed to Industry feedback. All implementers are requested and encouraged to provide feedback to the Association about changes they have made for implementation. Feedback can be submitted to changes@ppdm.org.

Frequently Asked Questions (FAQ)

Question? Our database is based on PPDM 3.6. Will that work with PPDM Lite?

Answer: With some work; Yes. The data transfer mechanism was designed for PPDM 3.7. However, if you have a data map of PPDM 3.6 to PPDM 3.7 (available at the PPDM web site), you should be able to modify the various procedures shown in steps 2 thru 7 for it to work just fine.

Question? We don't use PPDM as our Master Data Store. Could another database be used to populate PPDM Lite?

Answer: Yes, but there will be some work involved. You will have to create a data map between your existing data store and PPDM Lite. From that you will have to review and probably change all of the pl/sql routines for data loading and updating.

Question? Why are there so few well tables?

Answer: The well tables were designed to allow for a huge amount of rollups. If you look at most well tables, they contain; a run type, a run date, a top depth, a bottom depth and corresponding stratigraphic markers. It seemed to make sense (and follow the mandate of the PPDM Spatial project) to try and put all of these tables together in a single table; called WELL_SERVICE.

Question? Do I have to implement all of PPDM Lite?

Answer: No, just like PPDM Full, you can put various modules into production without affecting the entire model. It is easier to create all of the tables in the schema but then only populate the ones that you intend to use. Please refer to Appendix A for the load order of the tables.

Appendix A: PPDM Lite 1.0 Rings

This table contains the name of the table and ring level the table must be loaded in.

Ring	PPDM Lite Table
1	L_PPDM_DATA_SOURCE
2	L_CS_COORDINATE_SYSTEM
2	L_PPDM_TABLE
3	L_APPLICATION
3	L_AREA
3	L_BUSINESS_ASSOCIATE
3	L_ENTITLEMENT
3	L_FIELD
3	L_INTEREST_SET
3	L_PPDM_COLUMN
3	L_PROJECT
3	L_SF_SUPPORT_FACILITY
4	L_AREA_ALIAS
4	L_BA_ADDRESS
4	L_BA_ALIAS
4	L_CONTRACT
4	L_ENT_GROUP
4	L_FACILITY
4	L_FIELD_ALIAS
4	L_FINANCE
4	L_INT_SET_PARTNER
4	L_LAND_SALE
4	L_POOL
4	L_PROJECT_ALIAS
4	L_RESERVE_ENTITY
4	L_RM_INFORMATION_ITEM
4	L_SEIS_SET
4	L_SF_ALIAS
4	L_SF_DESCRIPTION
4	L_SPATIAL_DESCRIPTION
4	L_STRAT_NAME_SET
5	L_CONT_ALIAS
5	L_CONT_EXTENSION
5	L_FACILITY_ALIAS
5	L_FACILITY_SUBSTANCE
5	L_FACILITY_XREF
5	L_FIN_COST_SUMMARY
5	L_FIN_XREF
5	L_LAND_RIGHT
5	L_LAND_SALE_OFFERING
5	L_POOL_ALIAS
5	L_PPDM_ORIGIN
5	L_RESENT_CLASS
5	L_RESENT_XREF
5	L_RM_INFO_ITEM_DESC
5	L_SEIS_ALIAS
5	L_SEIS_POINT
5	L_SF_XREF
5	L_SP_GEOMETRY
5	L_SP_LINE
5	L_SP_MINERAL_ZONE
5	L_SP_POINT
5	L_SP_POLYGON
5	L_STRAT_UNIT
5	L_WELL
6	L_LAND_ALIAS
6	L_LAND_XREF
6	L_LICENSE
6	L_PDEN
6	L_SEIS_RECORD
6	L_SP_BOUNDARY
6	L_SP_LINE_POINT
6	L_SP_ZONE_SUBSTANCE
6	L_STRAT_WELL_SECTION
6	L_WELL_ALIAS
6	L_WELL_SERVICE
7	L_PDEN_STRAT_UNIT
7	L_PDEN_VOL_SUMMARY
7	L_PDEN_XREF
8	L_CONTRACT_COMPONENT
8	L_ENT_COMPONENT
8	L_FIN_COMPONENT
8	L_INT_SET_COMPONENT
8	L_PPDM_EXCEPTION
8	L_PROJECT_COMPONENT
8	L_RESENT_COMPONENT
8	L_RM_INFO_ITEM_CONTENT
8	L_SF_COMPONENT
8	L_SP_COMPONENT

