

Integration of DB oriented CAD systems with Product Lifecycle Management

Roberto Penas, SENER Ingeniería y Sistemas S.A., Tres Cantos/Spain, roberto.penas@sener.es
Carlos González, SENER Ingeniería y Sistemas S.A., Tres Cantos/Spain, carlos.gonzalez@sener.es

Abstract

This paper describes the solution designed to integrate a DB oriented shipbuilding CAD system with product life cycle management systems.

It is organized into three different phases:

- 1. Define Integration architecture through a neutral framework and first prototypes.*
- 2. Solution release into a shipbuilding CAD: FORAN. Solution based on document management and on demand data publishing and synchronization between CAD and PLM, including metadata and 3D graphics.*
- 3. Extended functionality by enabling automatic publishing and synchronization mechanisms, access control managed by PLM system and extended document management and 3D graphics support.*

The software solution described here is called FPLM.

1. Introduction

1.1. Nomenclature

A nomenclature is required for papers/technical notes using a large number of symbols, abbreviations and acronyms.

FORAN: Shipbuilding CAD application owned by SENER, Ingeniería y Sistemas.

FPLM: Software solution for connecting FORAN to a PLM system and based on the architecture described in this document.

PLM: Acronym for Product Lifecycle Management system.

Java: Software development platform owned by Sun Microsystems Inc.

CORBA: Common Object Request Broker Architecture

IDL: CORBA Interface Description Language.

1.2. Overview

FORAN system is used here as a sample shipbuilding DB oriented CAD to be connected with a generic PLM.

With most file oriented CAD systems, integrations with PLM systems usually consist of implementing a series of tools to put CAD files under PLM control, by setting up some rules, procedures and user interfaces.

However, the challenge here is that in DB oriented CAD systems, this integration is performed in a different approach, based on two main blocks:

- A set of tools for publishing and synchronizing product data between CAD and PLM databases.
- A set of document management tools to enable putting CAD generated documents under PLM control within CAD environment.

1.3. Development phases

These are the phases to develop the integration between the shipbuilding CAD (FORAN) and

different PLM systems:

1. **Initial.** Integration architecture through a neutral framework and first prototypes. This phase is now completed.
2. **Second.** First integration release into a shipbuilding CAD: FORAN. Solution based on document management and on demand data publishing and synchronization between CAD and PLM, including metadata and 3D graphics. This phase is completed.
3. **Third.** Extended functionality by enabling automatic publishing and synchronization mechanisms, access control managed by PLM system and extended document management and 3D graphics support. This phase is currently a work in progress.

2. PHASE I – Architecture definition and solution prototype design

2.1. Architecture overview

The connection solution consists of a number of components:

- A main Java process called *FPLM Server* which is the basic connection point between the PLM processes and FORAN functionality.
- A set of *CORBA IDL*'s that define the methods interface between FORAN and FPLM Server.
- A set of Java classes and methods that run embedded in FPLM Server that is the specific implementation for PLM system. This component is called *FPLM Plug-In* and consists of two parts:
 - A set of classes specific for each PLM system that runs embedded in FPLM Server, and are in charge of prepare the data and invoke the PLM functionality through Web Services
 - A package called FPLM FORAN Adapter embedded into PLM server that consists of:
 - A set of PLM Web Services that are invoked from *FPLM Server* included in PLM itself. Some Web Services are out-of-the-box (included in the standard PLM), while others are specific for FORAN integration.
 - A set of templates to define a marine entity definition package, including entity types, attributes for that entity types, lifecycles and workflows.
- A set of FORAN user interface client utilities, invoked from FORAN modules, to perform operations with PLM from FORAN, and that implement the CORBA IDL's explained above. These client utilities are generically called *FPLM Client*.

2.2. Architecture diagram

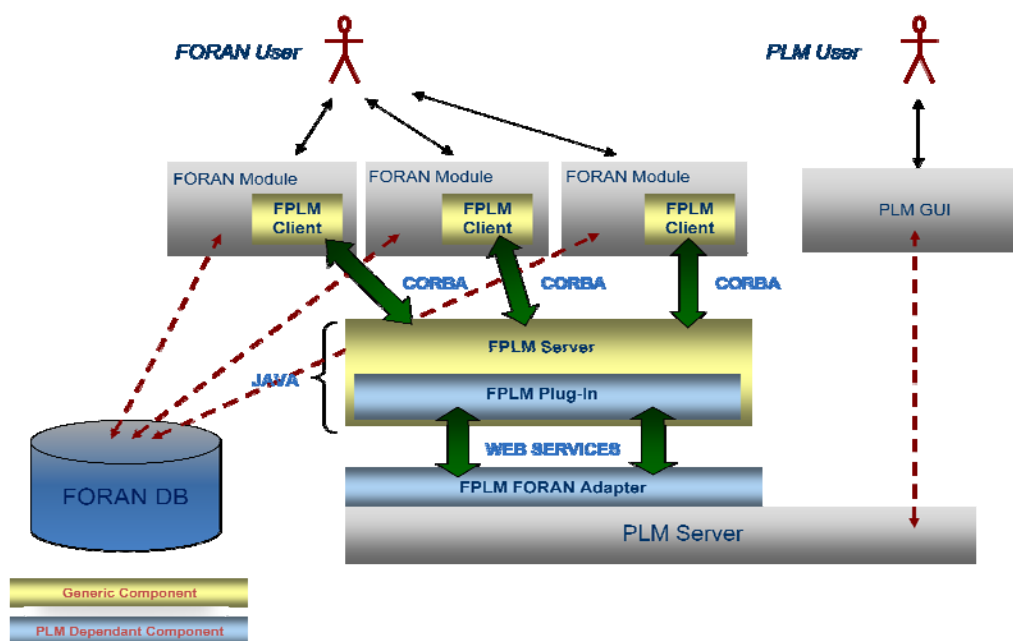


Figure 1: FPLM Architecture Diagram

2.3. Architecture components

2.3.1. FPLM Client

FPLM Client consists of a set of utilities that will be integrated into different FORAN modules to interact with PLM server.

FPLM Client is developed with the standard FORAN user interface tools, so it has exactly the same look and feel as other FORAN user interfaces.

FPLM Client utilities are included in these big groups:

1. ***PLM connection setup***. Basic tools to select a PLM server alias, perform user authentication and set local user parameters.
2. ***Document Manager tools***. A set of document manager tools to be invoked from FORAN modules that generate documents. Tools to manage links between model and documents are included.
3. ***Data publishing and synchronize tools***. A set of tools to publish and synchronize FORAN model tree data into PLM product structures.

2.3.2. FPLM Server

FPLM Server is a Java process in charge of managing FORAN business objects and their mapping to PLM objects.

FPLM Server will include a set of functional libraries called *FPLM Plug-In*.

FPLM Server relies on a couple of configuration files:

- *plmconfig.xml* registers the available PLM server aliases and their basic configuration.
- *entitytypes.xml* registers the entity definition types and their attributes that must exist in PLM entity type definition.

If either *plmconfig.xml* or *entitytypes.xml* do not contain the appropriate entries, *FPLM Server process* cannot start properly.

FPLM Server also relies on a standard CORBA ORB server that will be the communication port between FORAN *FPLM Client* and Server.

2.3.3. FPLM Plug-in

A set of specific PLM class libraries that will run embedded into *FPLM Server*.

FPLM Plug-In will manage the business logic and communicate it to PLM Server.

This component includes two parts:

- A set of Java classes and methods that run embedded into *FPLM Server*. These methods are in charge of invoking the specific Web Service calls to the PLM server.
- The PLM Adapter.- It is a set of Web Services and PLM templates that are embedded or run in the PLM server.

2.3.4. FPLM CORBA IDL Specification

A standard CORBA server is the communication interface between *FPLM Client* utilities and *FPLM Server* process.

There is a CORBA IDL set of classes and methods that will be the specification interface to communicate FORAN with PLM. *FPLM Client* will make use of those IDL implementations in the client side. *FPLM Server* will make use of those IDL implementations in Java in the server side.

This approach helps the client side applications to be fully independent on the kind of PLM system we are using.

3. PHASE II – FPLM Release

3.1. Overview

The first FPLM implemented solution has been released with latest FORAN version (v70R1.0) and consists of a set of PLM utilities fully embedded into FORAN modules that are classified in the following groups:

- FPLM connection setup.
- Local document manager utility.
- Local document manager extension for drawing generation module (FDESIGN).
- Local document manager for diagram modules.
- Model tree manager utility.

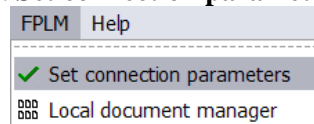
The main goal of this first FPLM release was to provide a comprehensive set of document management tools integrated into FORAN modules, as well as to provide a on demand tree publishing and synchronization between FORAN and PLM systems. This has been fully achieved.

3.2. FPLM connection setup

3.2.1. Functional Overview

A new FPLM menu is enabled. To perform authentication, you can either:

- Go to FPLM Menu and select **Set connection parameters** option:



- Click the **Set connection parameters** icon: 

The authentication window is shown in next figure:

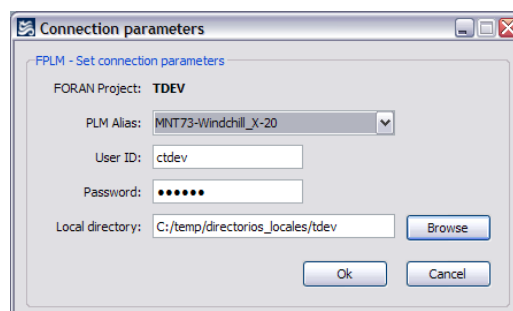


Figure 2: Set connection and authentication parameters

Where:

- **PLM Alias** is the PLM server identifier according to the FPLM Server configuration. You must select the appropriate PLM from the drop down list to authenticate to.
- **User ID** is the PLM user Id. By default, the current FORAN user Id is used. You can reuse the current FORAN user Id if it also exists in PLM user definition schema.
- **Password** is the PLM password for the user. By default, the current FORAN user password is used.
- **Local directory** is the user's local directory from which files will be transferred to and from PLM server. By default, the current FORAN user local directory is used.

Click OK button to authenticate in PLM system. If authentication is successful, you have then just

opened your connection session.

3.3. FPLM local document manager

3.3.1. Functional Overview

This is a local file manager to interchange files and documents with PLM server, ready to be invoked from all FORAN modules and within the context of a concrete FORAN project. This utility will be aware of the FORAN module from which it is invoked and will manage the file types proper of that module only, but will have no direct interaction with the data opened in the module session.

The basic function of this utility is to provide a basic mechanism to publish into PLM all files generated in FORAN modules (data files, reports ...).

This module lists local files, showing their state with regard to PLM. If it is needed to add a file to the local list from PLM, then a search capability will be used to allow document selection, download and, also, check-out options.

The basic metadata for a file document is stored in a file with the same name and *fplm* extension, placed in a directory called *.fplm* under working directory.

A specific IDL interface method is used to pass the appropriate data to FPLM Server.

This utility can be invoked from a FORAN module, and is prepared to manage the file types that can be generated or managed from this module.

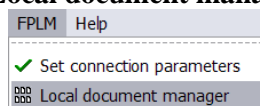
The common operations that can be performed with local document manager are the following:

- Manage document content types contextualized for the FORAN module used.
- Lists all local files managed by FORAN user and shows their PLM status.
- Search documents of a specific content type in PLM and add them to local files list.
- A set of dynamically enabled/disabled operations with selected documents.
 - Download file content from PLM server.
 - Check-out a document for local modification.
 - Undo a previous check-out.
 - Check-in a document after a local modification.
 - Upload a new document to PLM server.
 - Create a new document version.
 - Remove local file. This operation removes the file from local disc and from users file list. It does not remove the document in PLM. If the user needs working with the file again, he/she can search it again in PLM and return it to local list.
- Upload a document from a FORAN module, and manage from a different one. This is possible for certain document content types.

There is an editable XML template to configure the content types and FORAN modules compatibility map.

Local document manager can be invoked by either:

- Go to FPLM Menu and select **Local document manager** option:



- Click the **Local document manager** icon: 

The following figure shows current FPLM local document manager tool:

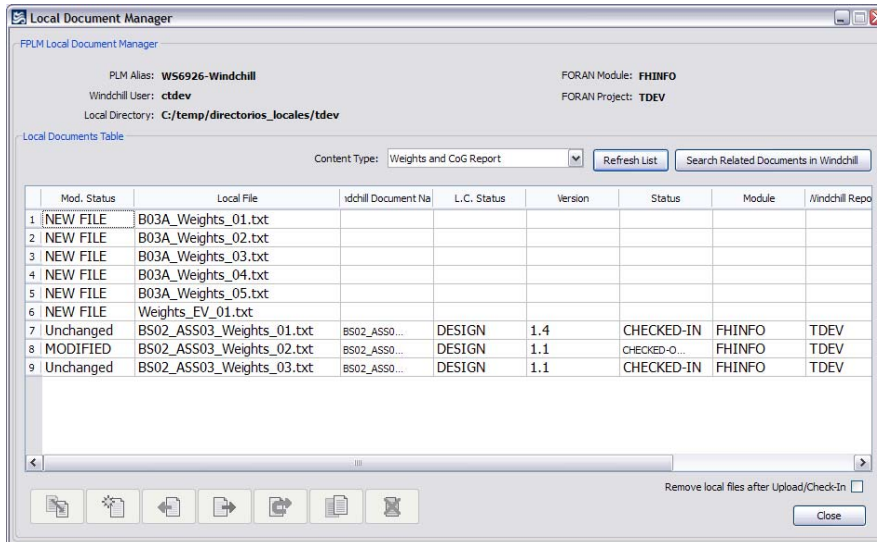


Figure 3: FPLM Local document manager

For searching a document in PLM system and add it to local list for user edition, we should click the *Search Related Documents in PLM* button, once a content type has been selected. The attributes are dynamically supported; this means that those attributes added to FORAN document types are automatically supported in FPLM interface. Once we have found the requested documents, we just need to select them and add them to the local list by clicking the *File Download and Add to Local List* button. This is shown in next figure:

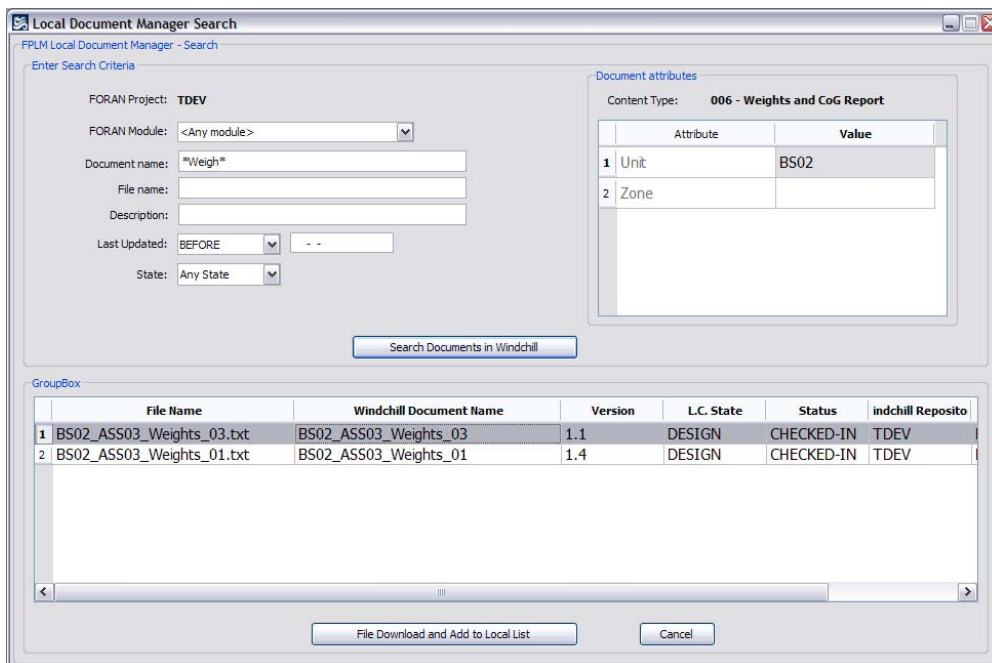


Figure 4: Search documents in PLM system.

3.3.2. Extension for FDESIGN

When using FPLM Local Document Manager in FDESIGN module, we can also perform specific tasks that allow further interaction and eliminates potential errors in data manipulation:

- Open a document file into current FDESIGN module session, by selecting a file in the list and clicking the *Open in session* button.
- Upload current opened file into PLM as new document.

- Check-in the current opened document file.
- Manage drawing BOM. Parts in PLM will be linked to the drawing that describes them.

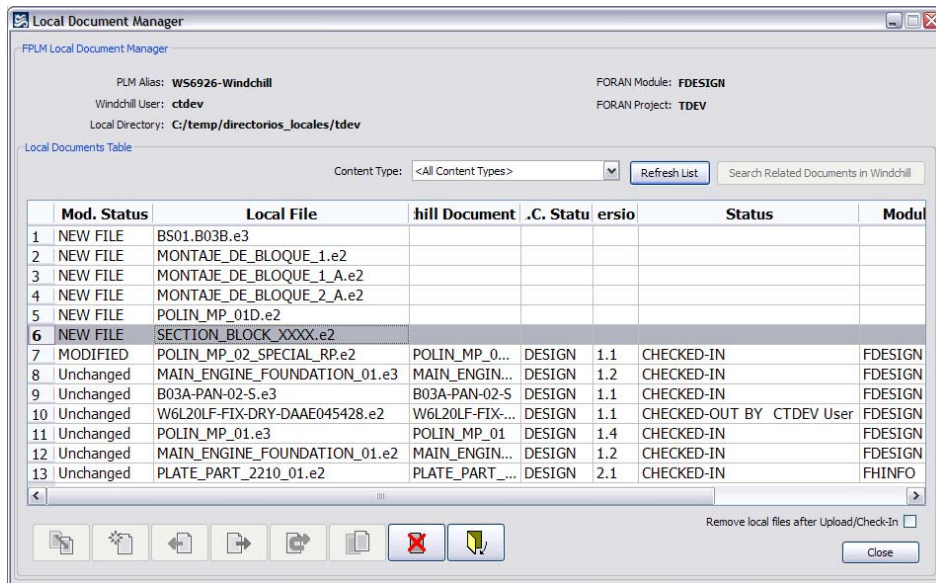


Figure 5: FPLM Local document manager in FDESIGN. Observe the *Open in session* button



Figure 6: FPLM icons in FDESIGN. Observe *Upload* and *Check-in* buttons.

In addition, a Set product data option is included to help PLM system to determine the appropriate part versions to be used for the applicability context. This concept is more widely explained in *Model tree manager* section.

3.3.3. Local Document Manager for Diagram Modules

In FORAN, schematic diagram graphics and metadata are stored in FORAN DB, therefore it has been implemented a special case for managing this type of documents in a PLM system.

In FORAN schematic design FORAN modules, the uniqueness diagram definition is through FORAN keys, like project Id, system Id, and sheet number, allowing the use of simplified diagrams published as PLM documents. Document Upload, Check-out and Check-in are automatic operations and user just need to invoke a generic *Publishing schema* operation that automatically detects whether the diagram document exists in PLM or not, and that performs those operations automatically when needed.

Additionally, a PDF snapshot is generated at publishing time, using that file as primary content for document.

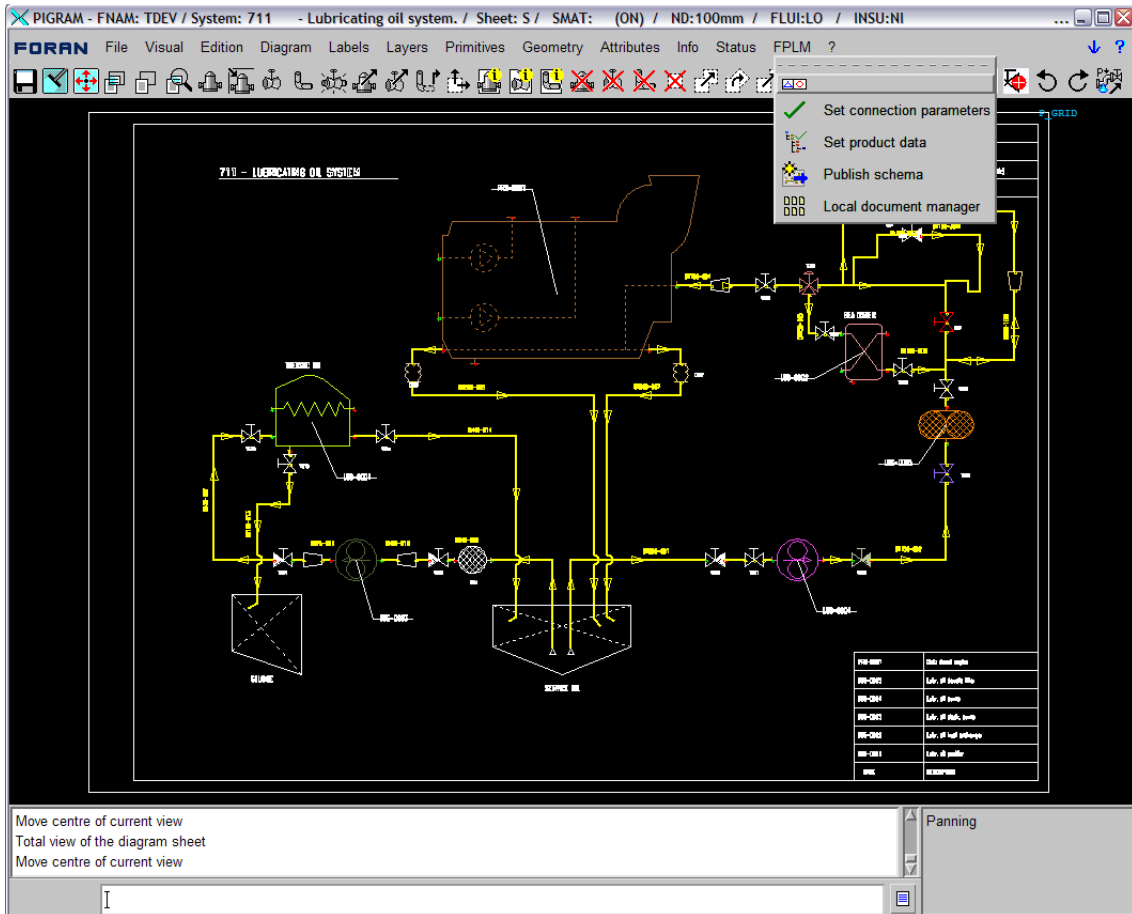


Figure 7: A FORAN P&ID and the existing FPLM options, including *publishing schema*.

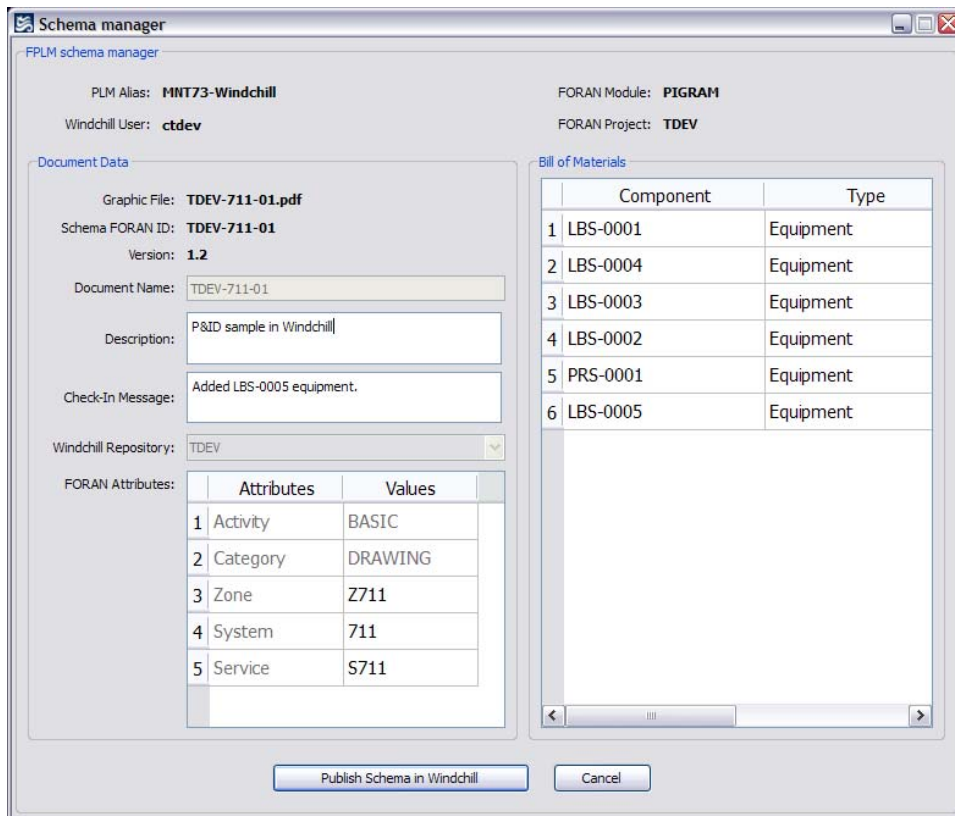


Figure 8: A *Publishing schema* window.

As well, a Set product data option is included to help PLM system to determine the appropriate part versions to be used for the applicability context. This concept is more widely explained in *Model tree manager* section.

3.4. Model tree manager

3.4.1. Functional Overview

This is a set of integrated tools into build strategy definition module (FBUILDS) to allow on demand tree publishing and synchronization capabilities between FORAN and PLM system, as well as enabling 3D mock-up visualization of FORAN ship model into a PLM system.

Two additional menu options are included in this module: *Set product data* and *Publish data cleanup*.

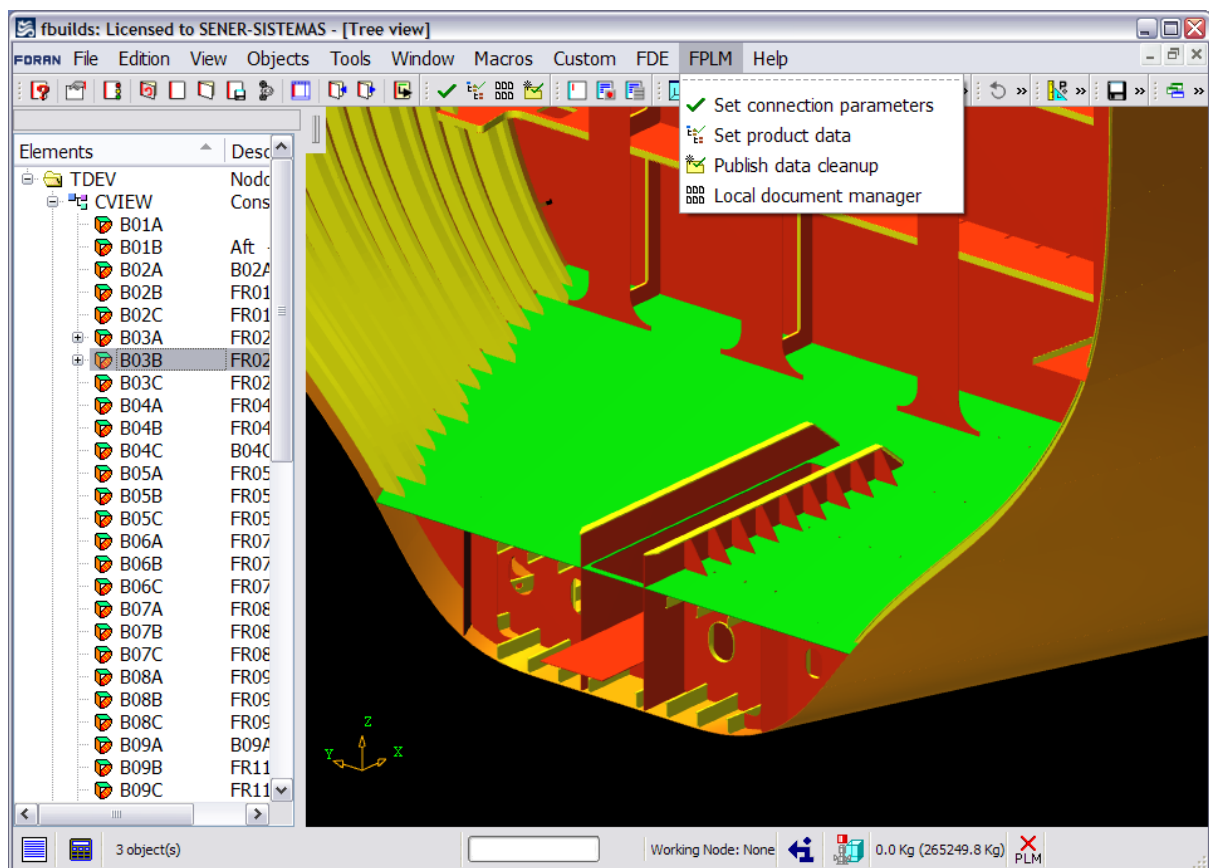


Figure 9: FPLM menu options in FBUILDS module.

3.4.2. Set product data

This option helps to establish the appropriate environment for data to be created and/or updated as a result of publishing and synchronization operations. Samples of data to be provided here are:

- A default repository or folder in PLM to locate the new parts to be created.
- A configuration specification to select the appropriate part versions in PLM system to be used. Usually, two choices are available: use latest versions or use versions compliant with a specific construction number.
- The construction number to be used if applicability by construction number option is selected. The value of construction number is obtained by default from FORAN DB directly.
- An optional engineering change notice number to be added in the transaction.

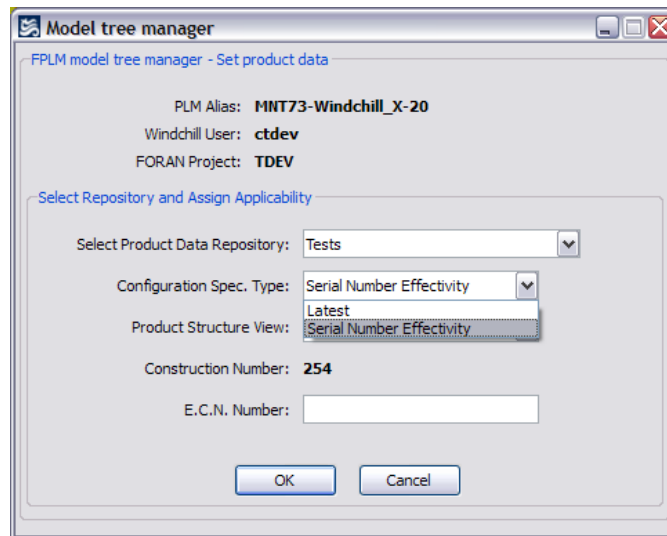


Figure 10: Set product data window.

3.4.3. Locking FBUILDS nodes

In order to enable publishing and synchronization between FORAN and PLM trees and product structures, FORAN tree node must be locked, and this operation will first ensure that PLM system authorizes the operation. According to PLM system response, FORAN node locking will be enabled (if PLM modification access on the tree items is granted) or disabled (otherwise). If, for instance, a product structure item in PLM is checked out, it means that it is under modification, so we will not be allowed to lock the tree in FORAN for the same assembly, in order to avoid simultaneous edition both in FORAN and PLM systems. If the locking is enabled, all PLM items in the tree will remain frozen meanwhile. As well, if locking is enabled, we can be in the situation of that the tree has been modified in PLM and those changes have not been applied to FORAN yet. This case is also managed and, if produced, FORAN will not enable tree publishing until it is first synchronized from PLM.

The objective of this FORAN-PLM locking device is to prevent simultaneous editions on the same tree structure in both systems:

- If a tree is locked in FORAN, it cannot be edited in PLM and vice-versa.
- If the tree is up to date in PLM, then synchronization from PLM is not required and this option will be automatically disabled.
- If a tree has been modified in PLM, the changes have to be first applied to FORAN tree before publishing to PLM is enabled. FPLM will inform about the items affected in PLM tree in a message, as shown in figure 11.

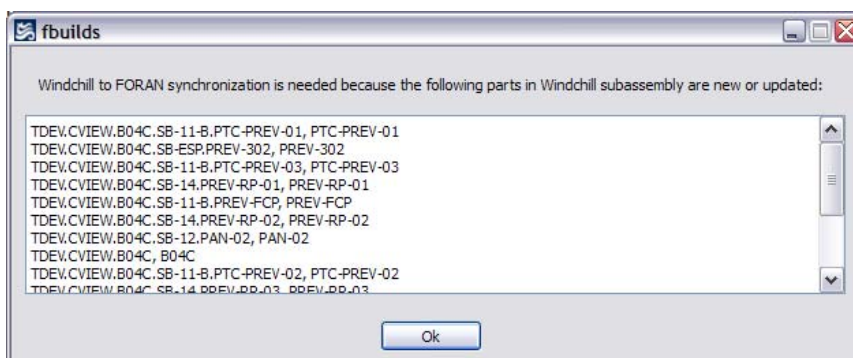


Figure 11: Sample of items modified in a PLM tree that make data synchronization mandatory.

When the user unlocks a tree node in FBUILDS, an instruction is sent to PLM system to unfreeze the assembly, so it could then be edited again. When finished, the system will show the following message:

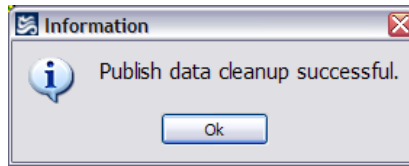


Figure 12: Message informing that the publishing data is updated to FORAN tree nodes and that PLM assembly has been unfrozen.

3.4.4. Model tree manager invocation

To invoke model tree manager, we just need to select the locked FORAN node, and choose the right button option called FPLM -> Model tree manager. The operation window appears. This window has just three elements:

- *Publishing to PLM* button. It will be disabled if a synchronization operation is required.
- *Synchronization from PLM* button. Once data has been synchronized, the publishing to PLM button will be enabled.
- *Generation of 3D Viewables* checkbox. If clicked, FPLM will generate 3D files that will be converted to native visualization format for the PLM used, if this conversion option is available for that PLM. Publishing process will then associate the visualization files to the published items into PLM system.

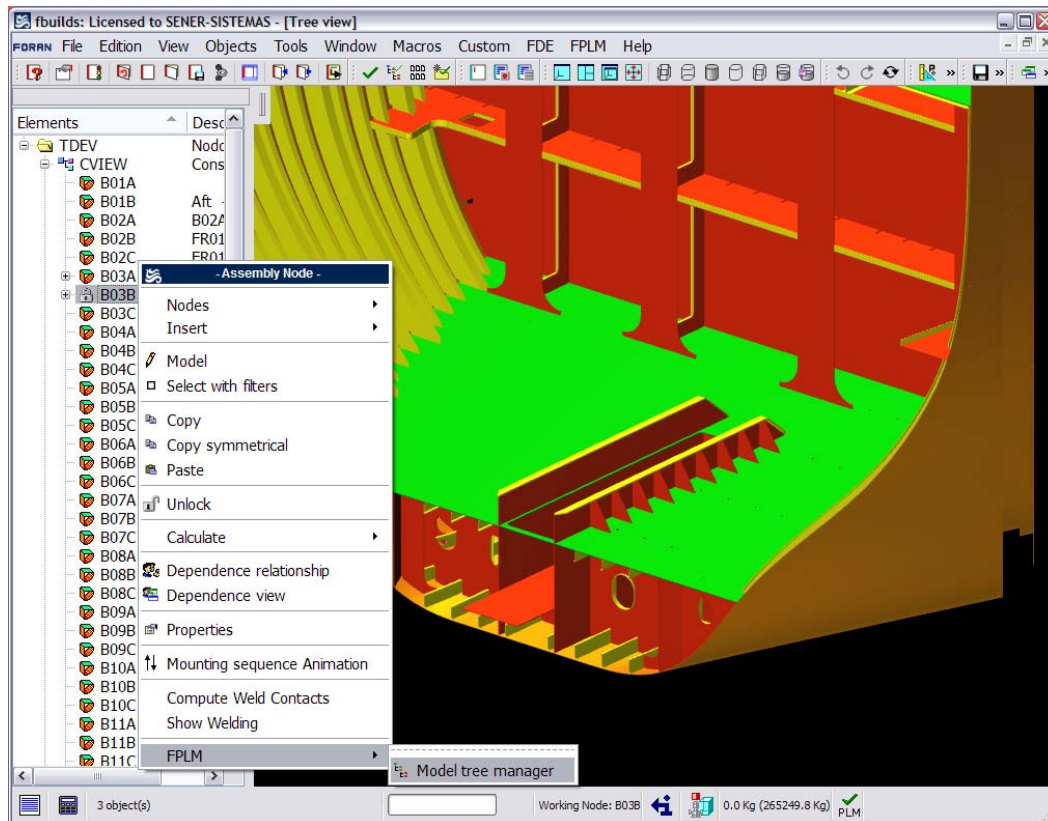


Figure 13: Model tree manager invocation

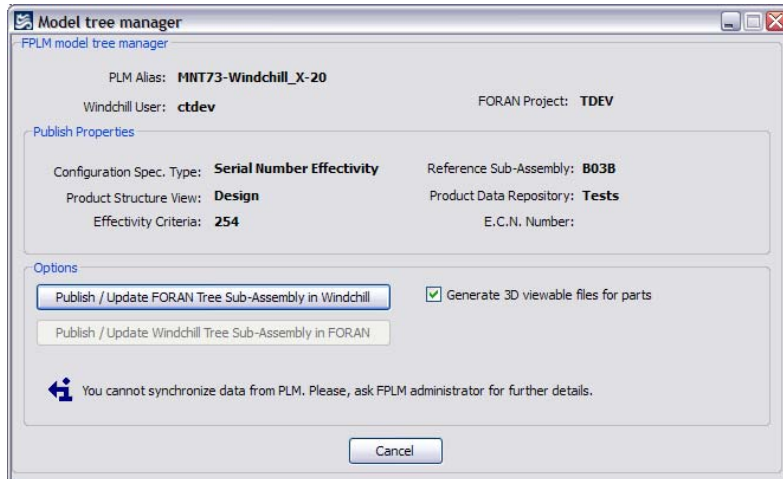


Figure 14: Model tree manager window. Synchronize option is disabled when PLM data is up to date.

Once the publishing operation has been launch, a message is shown to the user. Depending on the particular PLM used, publishing operation will be performed on-line or off-line.

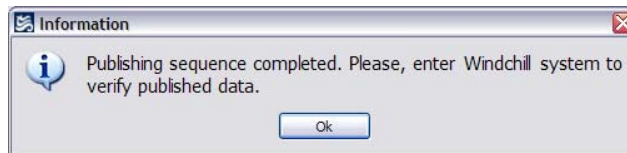


Figure 15: Publishing launch information message.

In PLMs that perform on-line publishing, the message shown in figure 15 means that the publishing has been successfully performed. In PLMs with off-line publishing, this message will indicate that all the data has been successfully sent to PLM to be published. In order to be informed about publishing results, the user must unlock FORAN node and check node attributes. If the operation is not finished, the user may then click the *Publishing data cleanup* option, which will update publishing status on the involved FORAN nodes.

The Synchronization is performed always in on-line mode and, when finished, it will show a message indicating that the operation is finished.

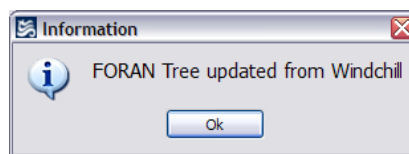


Figure 16: Synchronization end message.

4. PHASE III – FPLM Next Releases

4.1. Overview

For next FPLM releases, the target is to reach further significant milestones on FORAN to PLM integration, by providing a tight interconnection between the two systems. The most relevant features will be:

- **FORAN-PLM support table.** A FORAN table to register relevant data from PLM for all FORAN items. Will be the main PLM data repository for FORAN entities, and will store the

unique PLM identifiers for FORAN items and other PLM data like maturity indicators and access control flags.

- **New PLM-to-FORAN synchronization mechanism.** A new mechanism will be introduced to synchronize FORAN data according to changes in PLM. It will be called FSYNC and will work on a scheduled basis: every X minutes will proceed with data synchronization for entities created, modified or deleted in PLM.
- **New FORAN-to-PLM publishing mechanism.** A new device will be introduced to periodically publishing new, modified or deleted FORAN items into PLM. Depending on the PLM used, it will include 3D graphic support. Data to be published will come from FORAN model tables and FORAN-PLM one.
- **FORAN items creation in PLM.** Some FORAN items could be first created in PLM, according to certain rules, and then their FORAN keys be transferred to FORAN through FORAN-PLM support table through synchronization mechanism.
- **FORAN items access control driven by PLM.** The FORAN-PLM table will store data for allowing/disallowing access control on FORAN items. This means that, for example, if we put some items in read only state in PLM, we could not modify the equivalent items in FORAN.
- **New FORAN Product Tree.** A new FORAN product tree structure will be automatically created and updated in PLM system, to exactly reflect the current existing FORAN data. This tree will be automatically edited by the new FORAN-to-PLM publishing mechanism.
- **Local document manager extension for additional items.** Some specific drawing types will have a dedicated mechanism for managing their document files in PLM, including BOM links generation. Those are:
 - **Spools.**
 - **Isometrics.**
 - **Nestings.**

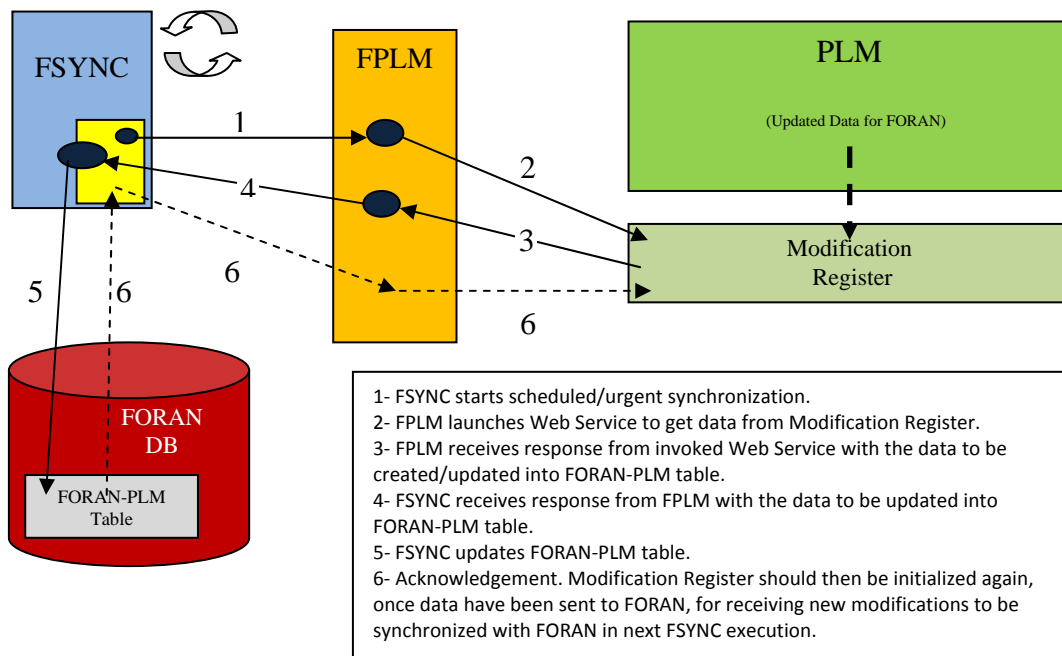


Figure 17: Process diagram for FSYNC module.

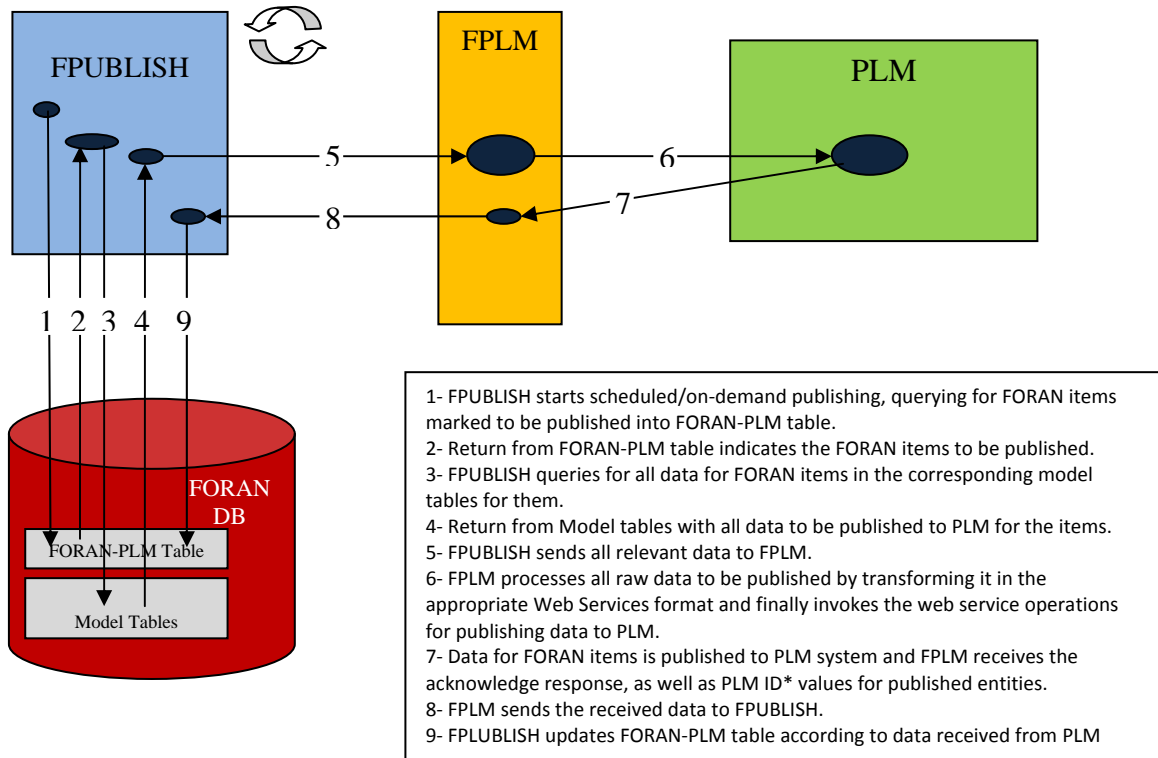


Figure 18: Process diagram for FPUBLISH module.

5. References

Penas, R. Sanchez, L. Gomez, A. Pastor, L. SENER, Ingeniería y Sistemas, S.A. (2009), *A NEUTRAL FRAMEWORK FOR THE INTEGRATION OF CAD IN PRODUCT MODEL LIFECYCLE SYSTEMS*, ICCAS 2009, Shanghai.