

# A PIM for MDA Illustrations

(with revisions and clarifications 13 May 2002)

## 1 Introduction

This is a very tiny example PIM. The purpose of this model is to provide a starting point for an illustration of PIM to PSM transformation. Vendors of MDA methods or tools are invited to use this example to illustrate the technique or techniques they use to specify the transformation of a PIM to a PSM.

Others who wish to contribute to our understanding of MDA are also urged to participate.

Vendors and method makers are requested to provide in their response whatever detail their method or tool provides. Others may provide whatever detail they feel is necessary to illustrate their approach, and no more.

This PIM and the discussion of what to put in the MDA illustrations based on this PIM are deliberately minimal. In particular, care has been taken, while describing what to put in the illustrations, to presume as little as possible about how a PIM to PSM transformation is done.

## 2 The platforms

### 2.1 Single program

This platform is a programming language and the services required to execute programs written in that language.

### 2.2 CORBA

This platform is standard CORBA and the services required to activate CORBA objects and support communication between them.

### 2.3 Components

#### 2.3.1 CORBA Components

This platform is CORBA Components and the services required to instantiate and execute CORBA components and support communication between them.

#### 2.3.2 Java Enterprise Edition

This platform is J2EE and the services required to instantiate and execute Java objects and support communication between them.

## 3 The platform independent model

### 3.1 The specification

This is a tiny model for a federation of automated trading systems. The business is trading gold. The trading systems act as agents of gold dealers to buy and sell gold.

Dealer systems in the community notify others of their current bid or offer. The notification includes a number of gold bars they will buy or sell and the price.

The trade practices specify:

- a bid or offer must specify a quantity in multiples of ten bars
- the price must be quoted in USA dollars per bar, to the cent,
- acceptance of a bid or offer must be for a quantity in multiples of ten bars.
- each bar delivered must:
  - be cast as one nominal 100 troy ounce bar of refined gold,
  - assay not less than .995 fineness,
  - not vary from the nominal weight by more than  $\pm 5\%$ ,
  - bear the stamp of a refiner approved by the New York Mercantile Exchange,
  - bear a serial number of that refiner, and
  - be stored in a warehouse designated for the storage of metals by the Exchange.

To make an offer to sell, a dealer's system notifies another of the offer, quoting the quantity offered and the price per bar. Such an offer is good until withdrawn. If another offer is made, the previously outstanding offer is thereby withdrawn. An offer with quantity zero withdraws the previously outstanding offer and makes no offer. An offer may be made to one or many other dealer's systems.

To accept an offer to sell, a dealer that received the offer provides the quantity bought to the offering dealer and the offering dealer responds with a confirmation number.

The clearing and settlement of trades are handled by other systems. This model is abstract, in that it hides all detail of the communication of trades to clearing and settlement systems.

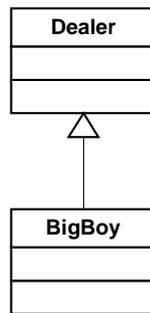
(For example, the selling dealer system provides to the clearing system the counterparty identifier, confirmation number, quantity and price, and nominates gold bars for delivery by specifying for each bar: refiner, serial number, warehouse where stored, blanket warehouse receipt number, and the actual weight of the bar. The buying dealer system provides to the clearing system the counterparty identifier, confirmation number, quantity and price.)

To make a bid to buy, a dealer notifies another (or several) of the offer, quoting the quantity bid and the price per bar. Such a bid is good until withdrawn. If another bid is made, the previously outstanding bid is thereby withdrawn. A bid with quantity zero withdraws the previously outstanding bid and makes no new bid.

To accept a bid to buy, a dealer that received the bid provides the quantity sold to the bidding dealer and the bidding dealer responds with a confirmation number. The clearing and settlement of trades are handled by other systems in the same manner as trades that originate with an offer.

A dealer may make both a bid and an offer to another dealer or dealers.

The federation is a peer-to-peer system. Any participant may make an offer or bid (or both) and any may accept an offer or bid.



As with some other aspects of society, however, not all peers are equal. Smaller dealers trade freely with each other.

The big boys, on the other hand, while trading freely with each other, require that a smaller dealer who wishes to trade with one of them, first request to establish a trading relationship. When one of the big boys receives such a request it may accept or reject the proposed liaison. One established, either party might terminate the liaison.

Through their dominance of the standards organization of the gold trade, the big boys have caused there to be two standard interfaces for trading systems in the federation.

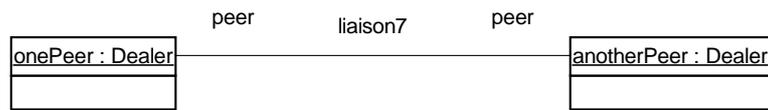


This diagram specifies that the system have two types of objects.<sup>1</sup> Dealers interact by making requests of each other and by providing notifications<sup>2</sup> to each other. Dealers are also able to request establishment of a peer-to-peer liaison with certain dealers. Once the liaison is established, the two peers interact as any other pair of dealers.

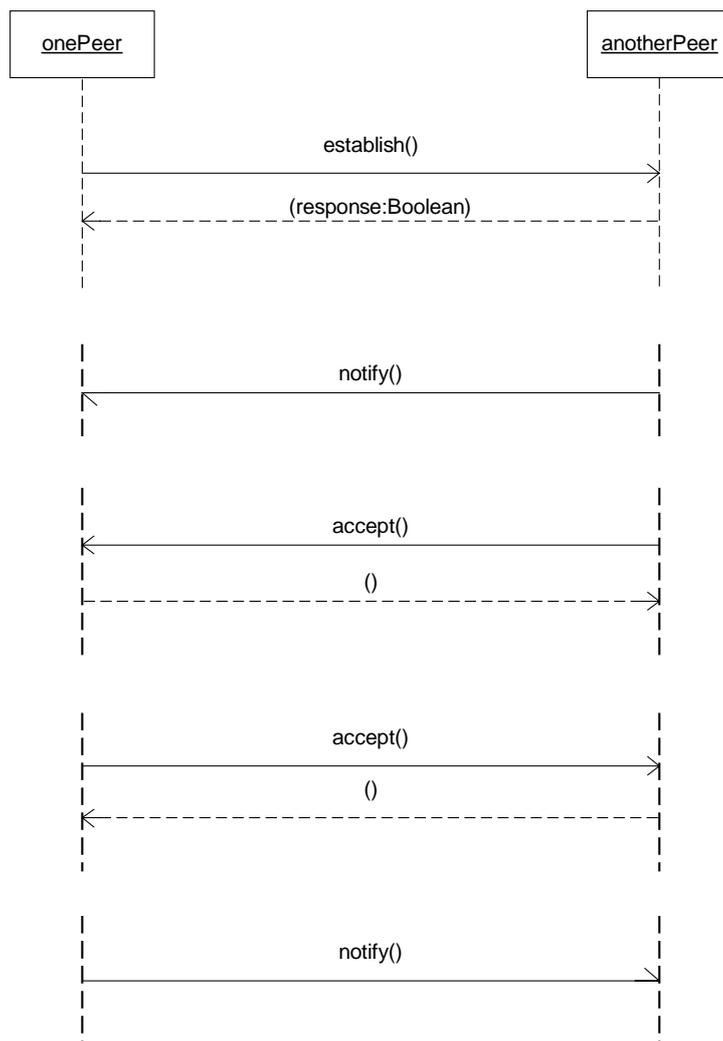
This platform independent model does not specify how the liaison is established, nor how requests are made or notifications given.

The operations are:

- establish: An interrogation, which establishes a liaison between an object and the invoking object.
- terminate: An announcement, which ends the liaison between two objects.
- notify: An announcement that is an offer to sell or a bid to buy. The parameter, bidOffer distinguished bids from offers.
- accept: An interrogation that buys or sells and returns a confirmation. If the parameters do not match an outstanding bid or offer, the a negative confirmation is returned.

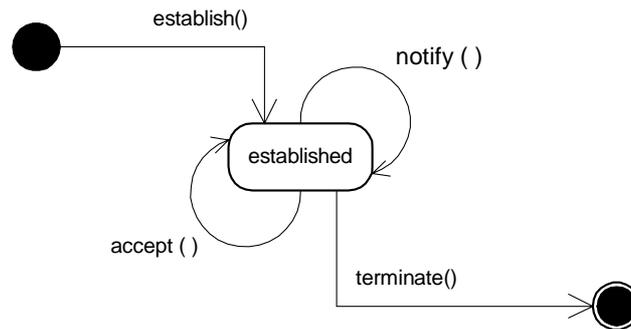


This diagram provides an example of a liaison between one peer and another. As a peer may have several peers at the same time, the diagram gives a name to this particular liaison. Note that some liaisons are established by explicit action, while others are preexisting, by virtue of the practices of the trade.



This drawing presents fragments of sequence diagrams, showing parts of possible interactions between the two objects.

The purpose of the drawing is simply to show the kinds of interactions that any platform specific model must enable. It does not specify the sequence in which they must or will occur.



This diagram specifies the behavior enabled by a liaison between a peer and one of the big boys. The establishing behavior takes place when the establish operation is invoked. While the liaison is established, the two peers may trade. The terminating behavior takes place when the terminate operation is invoked by either party.

This drawing shows the kinds of interactions that any platform specific model must enable.

## 3.2 Regarding this specification

If you have any questions about this specification, that is to be expected. As with any project, you will want to ask questions. Please send your questions to:

<mailto:joaquin@acm.org>

### 3.2.1 Clarity

If the specification is not clear, let's clear it up.

### 3.2.2 Detail

If the specification lacks detail, this may be for one of several reasons:

- It is platform independent.
- The lacking details are part of a platform independent model, but are hidden by abstraction in this tiny example.
- There is an error or failing in the specification.

If you are in doubt, or feel the third is the reason, please write.

If you wish to add details to the PIM, for example, to specify bidOffer, the specify the data type of the returned confirmation, or how to indicate a negative confirmation, please do so. On the other hand, if you can, and wish to, provide your illustration without these details, please do that.

### 3.2.3 Correctness

It is possible that the drawings do not conform to the UML 1.4 specification, or to your understanding or use of UML. Let's discuss that.

## **4 The platform specific models**

Here are several platforms. Please prepare a platform specific model for one or more of these platforms.

### **4.1 Single program**

This platform is a programming language and the services required to execute programs written in that language. Produce a single program, executing on a single machine. The objects in the example will all be in that program.

### **4.2 CORBA**

This platform is standard CORBA with the services required to activate CORBA objects and support communication between them. The programming language is not specified. When a programming language is chosen for each object, the CORBA language mapping to that language will be used for that object.

For concreteness, please choose a single language for each object, in order to illustrate the transformation.

Make each object in the PIM a CORBA object. Provide a means of communication using any CORBA capability, or more than one.

### **4.3 Components**

In the component platform models, place each of the objects in the PIM in a separate container and provide a means of communication.

#### **4.3.1 CORBA Components**

This platform is CORBA Components and the services required to instantiate and execute CORBA components and support communication between them.

#### **4.3.2 Java Enterprise Edition**

This platform is J2EE and the services required to instantiate and execute Java objects and support communication between them.

## 5 The deliverables

For each of the PIM to PSM transformations you illustrate, deliver

- a description of the approach used
- the resulting platform specific model
- a record of the transformation

The more complete your illustration, the more useful it will be. But any discussion at all will be very useful. Please do not hesitate to contribute whatever you have time to prepare.

### 5.1 Description of approach

Describe how MDA works for you. And how you have applied that to produce the transformation of the example PIM to a PSM.

Please be as complete as you have time to be. Our purpose is to provide OMG members with illustrations of the use of the model driven architecture approach. To help insure a good understanding of your approach, please put yourself in your reader's shoes. Do not assume what others might not take for granted or even think of. It will be better to state the obvious, than to have misunderstandings caused by different assumptions.

If you use the same approach in each transformation, describe that approach only once.

### 5.2 Platform specific model

The platform specific model shows the result of the transformation.

This model need not be a UML 1.4 model, if that is not what you produce. It might be a model in some proposed form of UML 2. It might be some other form of model. If the result of the transformation, or the result of a second transformation, is source code for a computer program to implement the system, provide that.

The result may include more than a PSM. Or, the PSM may include more than the model. Or may include more than one model. As: examples: For a single program platform, there may be code and a make file. For a CORBA platform, there may be both IDL and code. For a component platform, there may be a deployment descriptor. Or, for any of these, there might be only a UML model.

Our purpose is to find out how you do a MDA transformation, not to prescribe how that is done, nor what are the results. Please provide the results of a transformation as you do it.

### 5.3 Record of transformation

Provide a record of the transformation. Show (using the technique illustrated by your response):

- Any additional information or other thing used, in addition to the PIM, for the PIM to PSM transformation.
- Each part of the transformation, namely:
  - a collection of (one or more) elements of the PIM
  - the corresponding collection of (one or more) elements of the PSM
  - the data that determined this part of the transformation.

The data that determines some part of a transformation might be in the form of a model, some form of rule or rules, some other form of data, or might be imbedded in a transformation tool. It might simply be a natural language description of what determined this part of the transformation.

This record is, perhaps, the most important thing that you can provide. We have said no more about it, in order to avoid over specifying this request. The goal is to learn in full detail how you do a PIM to PSM transformation.

### 5.4 Format

Please place all of your illustration in a Microsoft Word or Adobe PDF document.

If you use a Microsoft Word document, when adding drawings to the document, please use Edit PasteSpecial Picture. This will ensure that everyone can see your drawings, however you have prepared them.

If you wish to make available other documents or files, please do so.

In particular, if you have prepared your models using a modeling tool, and want to provide the model file, or have used some other tool and want to provide the files used with that tool, please do. If you can provide XMI files, so much the better. If you are a tool vendor, please provide the files used with your tool.

## 6 The next step

When you have completed your illustration, please:

<mailto:joaquin@acm.org>

With your permission, we will collate your document with others and post them as an OMG document. Otherwise, if you ask, they will be used only by ORMSC to learn about your approach.

Again, please do not hesitate to submit any document you are able to prepare. An illustration of any part of an MDA approach will be valuable.

Thank you for your work.

---

<sup>1</sup> The following terms are used in this document with the meaning given the terms in ISO/IEC 10746, Reference Model of Open Distributed Processing (ODP): object, type, establish, establishing behavior, liaison, terminate, terminating behavior, operation, announcement, interrogation.

<sup>2</sup> The use of the term 'notification' neither requires nor suggests use of the CORBA notification service or a similar service. The distinction is simply that a peer expects a response to a request, but does not expect a response to a notification.