

CFGS Desenvolupament d'Aplicacions Multiplataforma (DAM) CFGS Desenvolupament d'Aplicacions Web (DAW)

Mòdul 5 – Entorns de Desenvolupament. UF3 – Introducció al disseny orientat a objectes

Tutorial programari Modelio



Índex de contingut

Introducció.....	3
Instal·lació i execució.....	3
Creació d'un projecte.....	5
Creació de diagrames.....	6
Treball amb projectes i diagrames.....	7
Diagrama de classes.....	9
Algunes accions d'edició comunes a tots els diagrames.....	13
Diagrama de casos d'ús.....	14
Diagrama de seqüència.....	15
Diagrama d'activitats.....	16
Diagrama d'estats.....	16
Generació de codi a partir del diagrama de classes.....	17
Generació de classes a partir del codi.....	20
Errors coneguts.....	24
Problemes amb els tipus primitius.....	24
Problema en obrir: Impossible to load diagram content.....	24
No es mostren a la classe els atributs afegits.....	25

Introducció

Modelio és un entorn de modelatge extensible de codi obert (llicència GPL) desenvolupat per Modeliosoft. Suporta UML2 i BPMN. S'ha desenvolupat amb Eclipse i molts dels seus plug-ins. És d'ús bastant **amigable**. Podeu ampliar informació aquí: <https://www.modelio.org/documentation-menu/user-manuals.html>

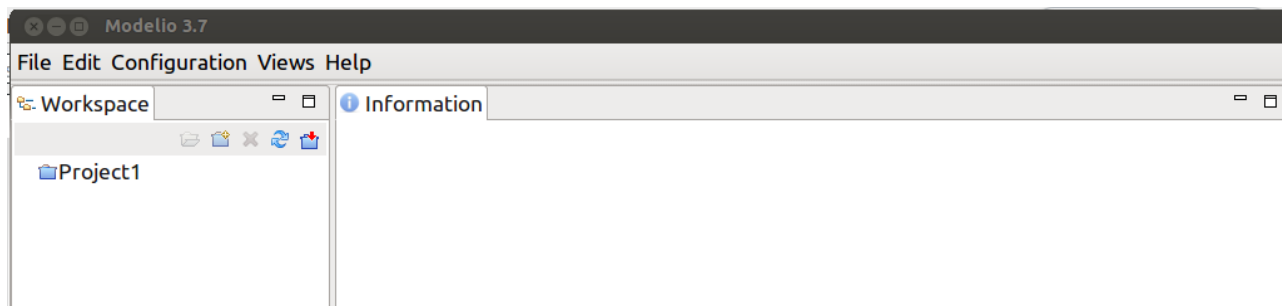
Instal·lació i execució

1. Comproveu els requeriments aquí: <https://www.modelio.org/downloads-links/requirements.html> Cal remarcar els que posa al final:
 - Java 8
 - A MAC OS X cal que s'hagi instal·lat JDK 8 (i no només JRE 8, com als altres sistemes).
 - Ha d'estar definit un navegador "del sistema" (si és Internet Explorer, ha de ser la versió 10 o superior).
2. Descarregueu-lo des de <https://www.modelio.org/downloads/download-modelio.html>. Cal que seleccioneu la descàrrega que correspon a la versió adequada del vostre sistema operatiu. És molt recomanable que comproveu la correctesa de la descàrrega tot calculant el "checksum".
3. Executeu l'instal·lador que us acabeu de baixar si el vostre sistema és *Windows* o *GNU Linux*. Si el vostre sistema és *MAC OS*, descomprimiu el fitxer que us heu descarregat.
4. Us apareixerà la pantalla de benvinguda. Només surt el primer cop que s'executa el programa. Podeu veure-la a continuació:

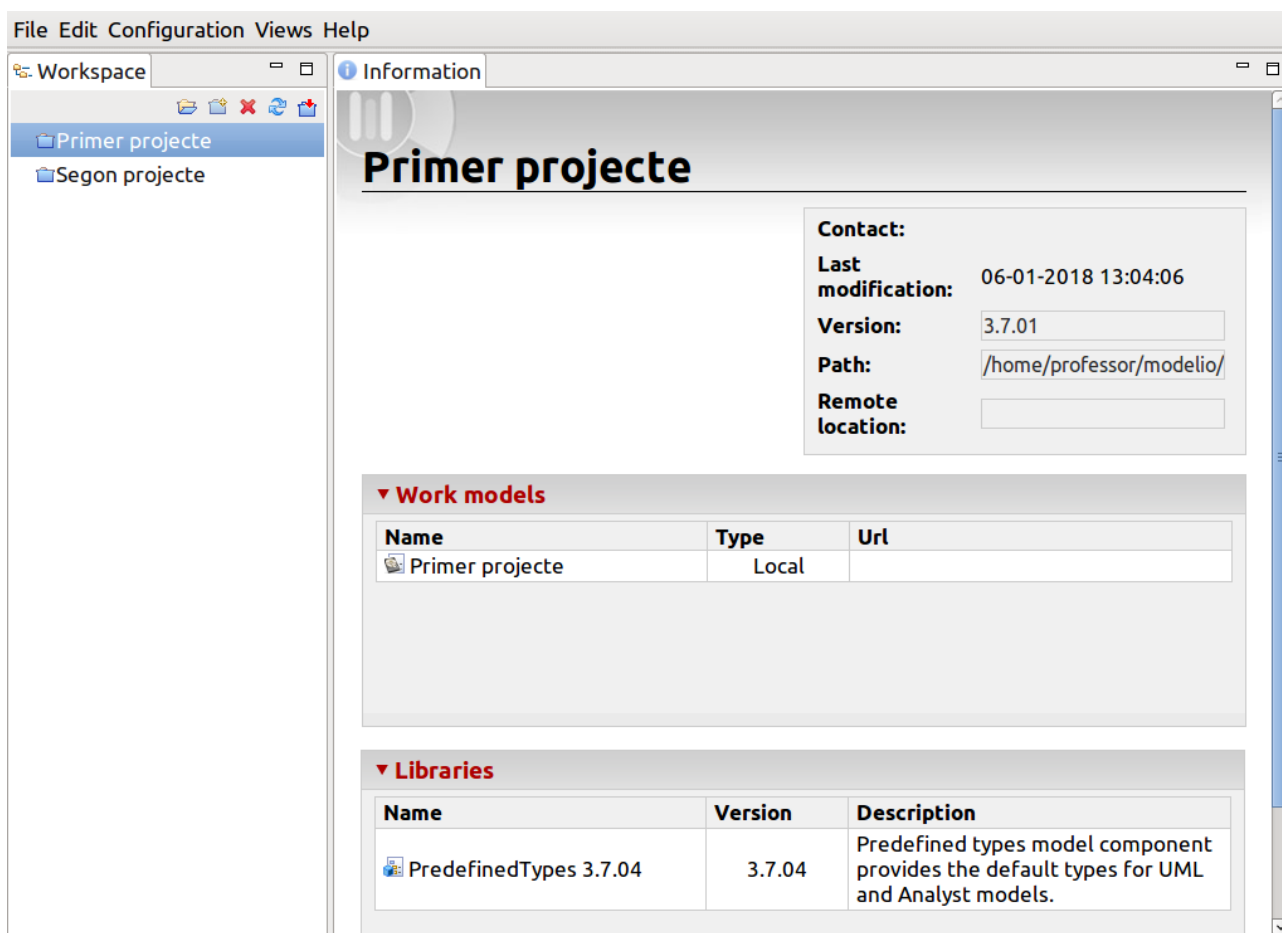


Si voleu, podeu explorar les diferents opcions que ens presenta.

Nosaltres la tancarem i ens apareixerà la pantalla que veurem habitualment i que trobareu a continuació, a la pàgina següent:

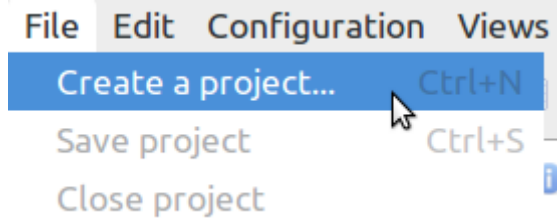


Ara apareix buida, però a la pestanya **Workspace** ens aniran apareixent els diferents projectes que anem creant-hi i a la pestanya **Information** podrem visualitzar-los, un cop oberts.

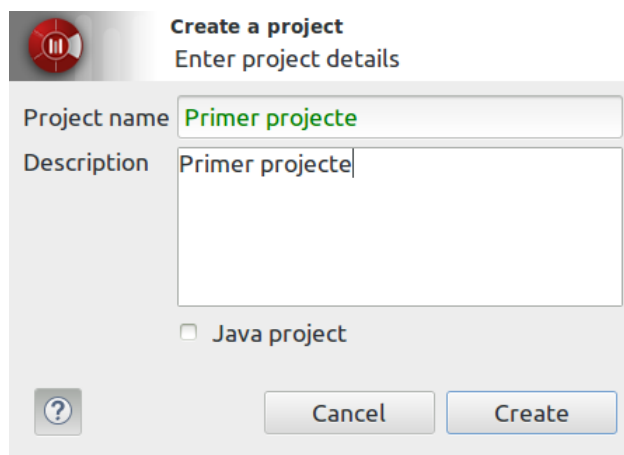


Creació d'un projecte

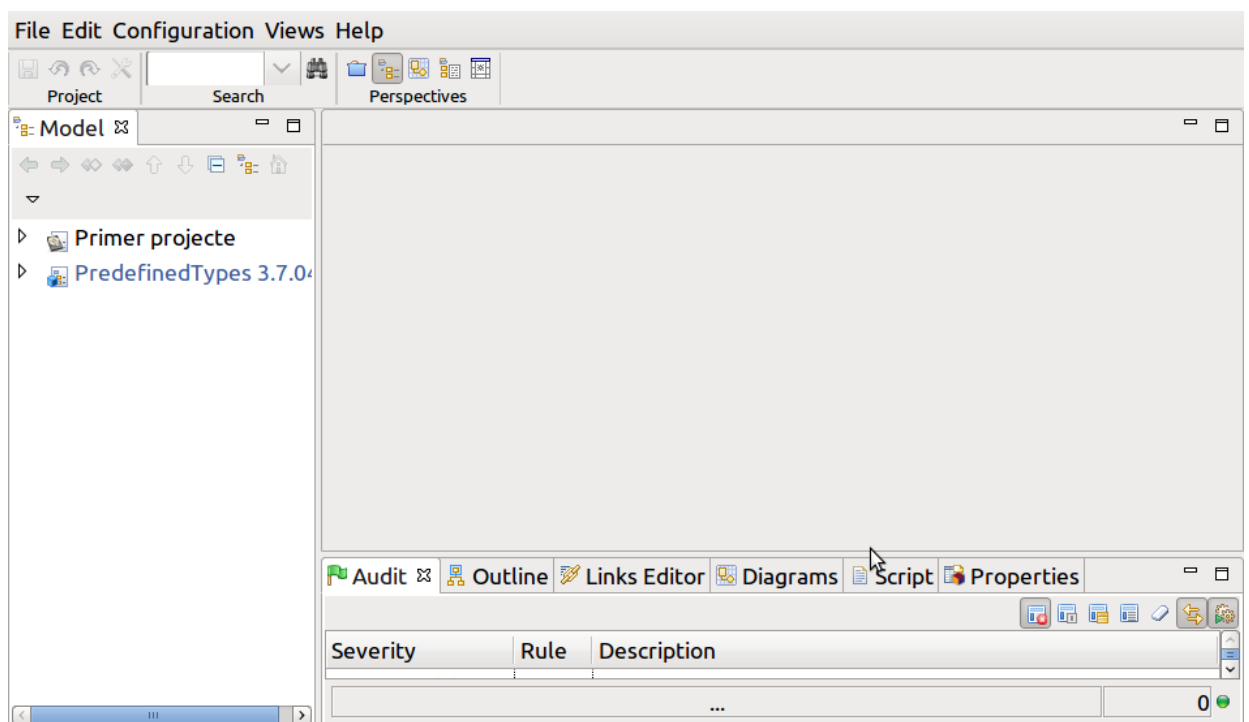
1. Seleccioneu l'opció File → Create a Project...



2. Entreu-hi el nom, si voleu, una descripció i feu clic a **Create**. Si abans feu clic a l'opció *Java project* s'activarà el dissenyador Java. La seva finalitat és tenir sincronitzat el codi Java amb els diagrames UML. No utilitzarem ara aquesta opció, perquè la generació de codi es veu en un altre apartat.



Ens apareixerà aquesta pantalla:



Error conegut:

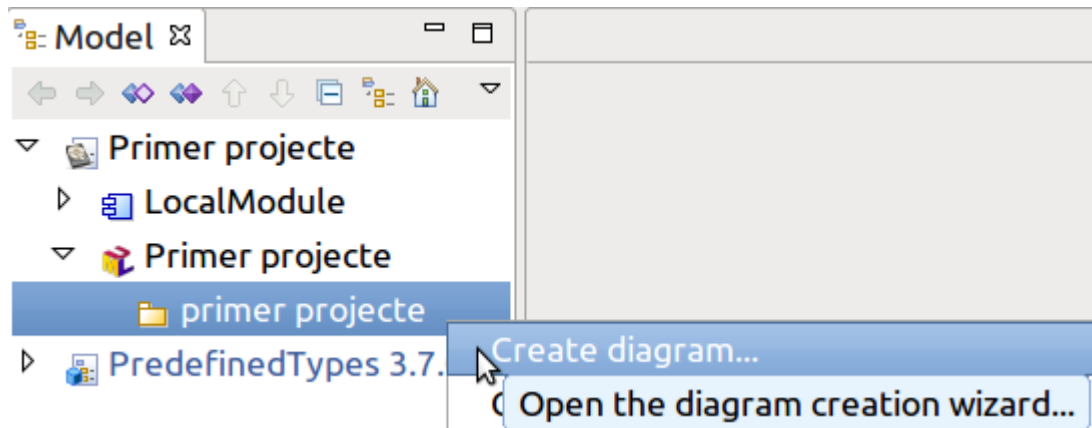
java.io.IOException: The 'PredefinedTypes X.X.XX' fragment's 'Standard vY.Y.YYYY' metamodel is outdated compared to the current 'Standard vZ.Z.ZZZZ' version.

.....

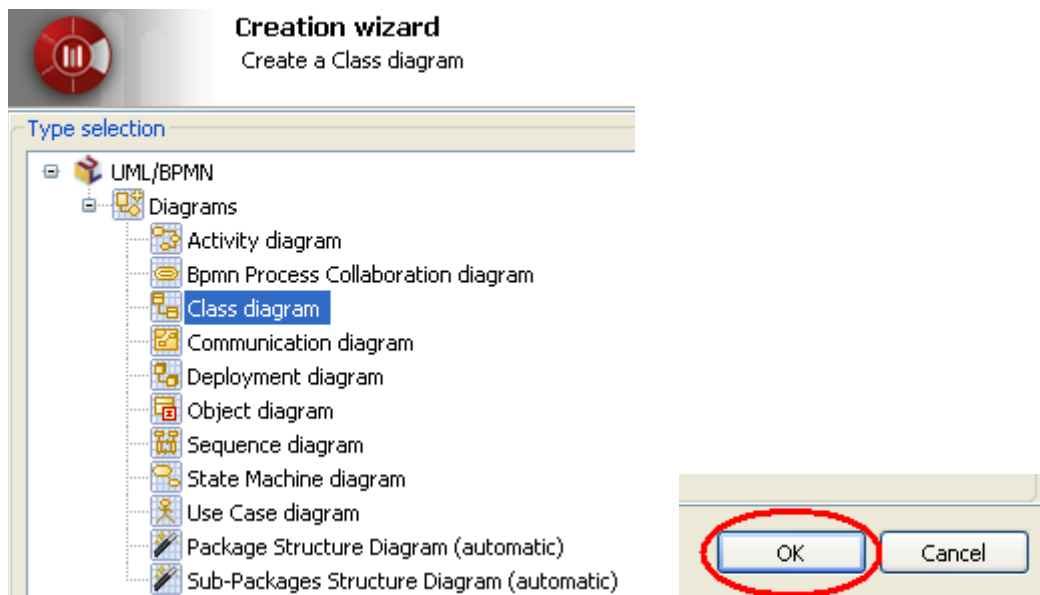
Enllaç: [Problemes amb els tipus](#)

Creació de diagrames

1. Cal fer clic amb el botó secundari a sobre de la icona que representa el projecte i seleccionar *Create diagram...*

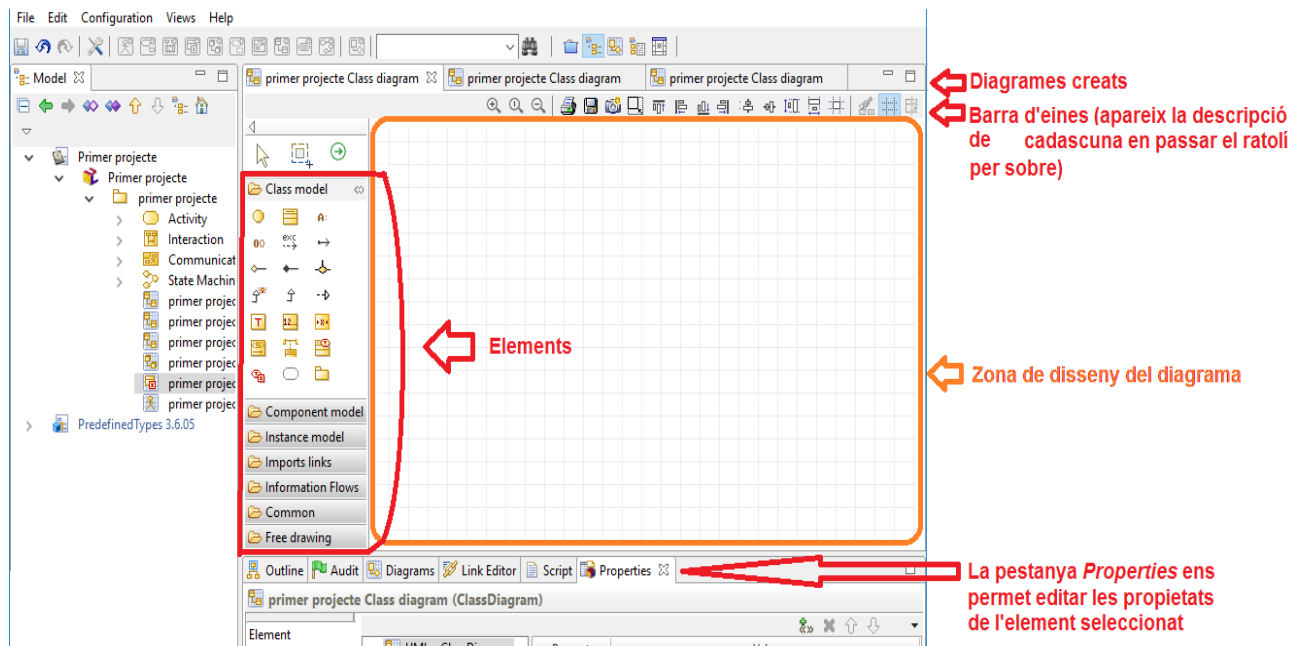


2. Després, cal triar el tipus de diagrama que vulguem i fer clic a OK



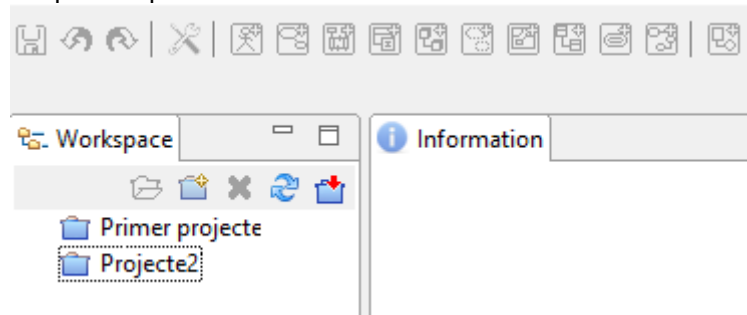
Perquè us surtin tots els tipus de diagrames, heu de desmarcar "Show only applicable elements".

3. Ens apareix a la finestra principal una pestanya amb el nou diagrama.



Treball amb projectes i diagrames

Quan obrim el programa a la pestanya *Workspace* (espai de treball) ens apareix una icona per cadascun dels projectes existents a l'espai de treball. Per obrir el projecte amb què treballarem, només caldrà fer doble clic a sobre de la carpeta que el representa.



Error conegut:

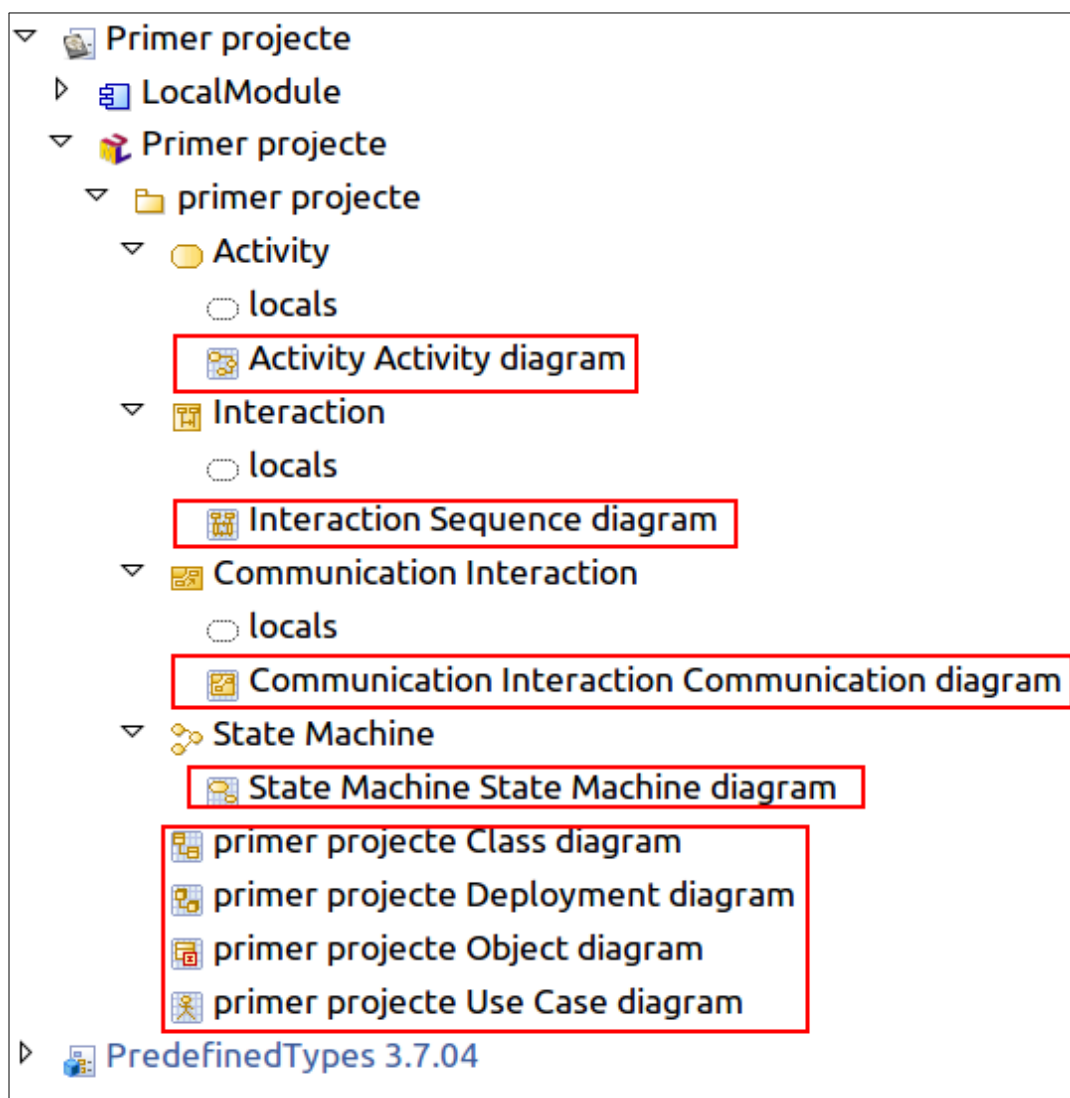
Impossible to load diagram content.

Check that you're using the right edition of Modelio or that you have a license for this kind of diagrams.

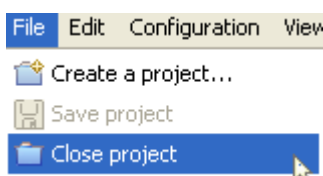
.....

Enllaç: [Problema en obrir: Impossible to load diagram content](#)

Un cop obert el projecte, si volem obrir un diagrama que no està obert, només cal desplegar l'arbre del projecte i fer doble clic a sobre del diagrama que volem obrir (a la imatge següent estan emmarcats els diferents diagrames on podem fer doble clic).

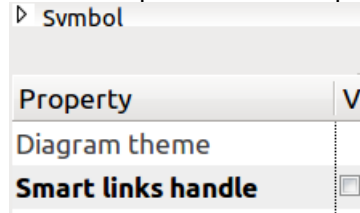


Podem tancar un diagrama tancant la seva pestanya; podem tancar el projecte seleccionant l'opció *File* → *Close Project*



A tots els diagrames convé desactivar l'opció **Smart links handle**, ja que no l'aprofitarem i ens pot incomodar una mica. Es fa seguint els següents passos:

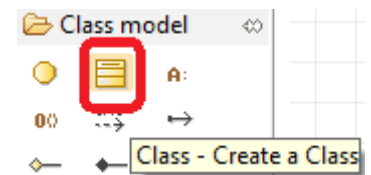
- Tenir el diagrama seleccionat. Es fa clicant un punt del diagrama que no tingui cap element.
- Fer clic a la columna *Symbol*. Apareixerà una pantalla com la que hi ha a continuació:



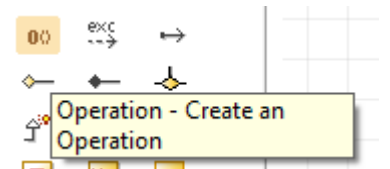
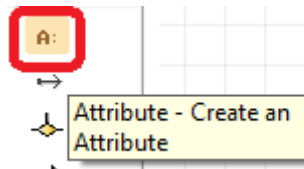
- Desactivar l'opció *Smart links handle*.
- Fer novament clic a la columna *Symbol* per minimitzar-la.

Diagrama de classes

- **Afegir-hi una classe:** fer clic a la icona corresponent i, després, fer clic al punt del diagrama on volem inserir-lo.



- **Afegir-hi un atribut o una operació a una classe:** fer clic a la icona corresponent i, després, fer clic a la classe on ha d'anar.




Després de donar d'alta una operació en una classe, caldrà assegurar-se que la classe té prou amplada per mostrar tots els paràmetres; per tant, si cal, caldrà redimensionar-la (s'explica més endavant com fer-ho).

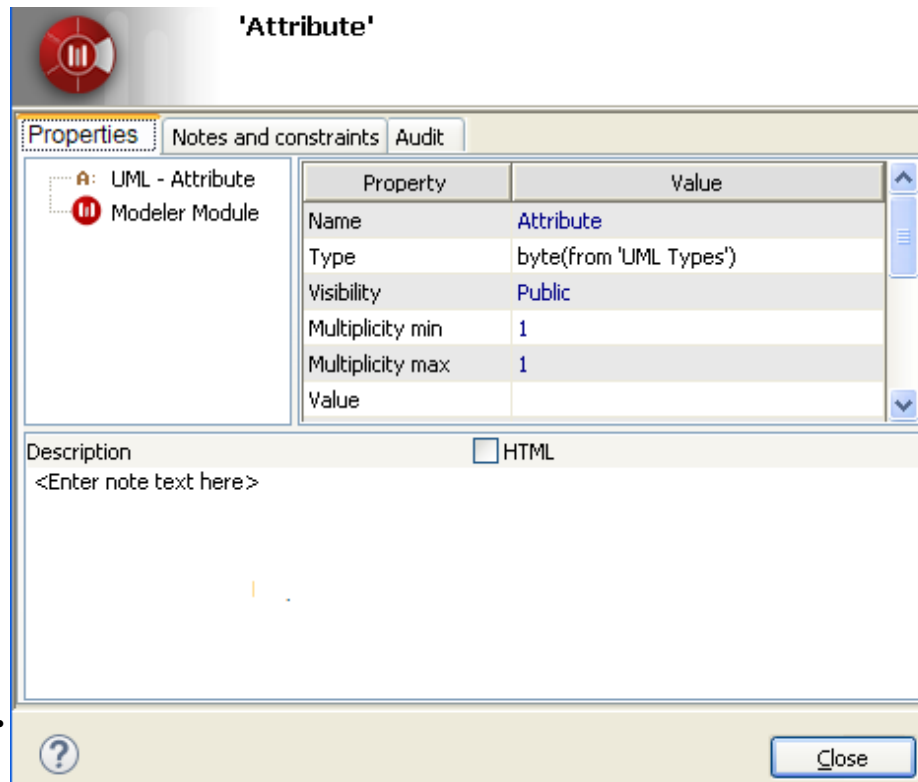
Error conegut:

No es mostren a la classe els atributs afegits.

Enllaç: [No es mostren a la classe els atributs afegits.](#)

- **Seleccionar un element:** n'hi ha prou amb fer-hi clic a sobre o, si volem seleccionar-ne més d'un a la vegada, podem:
 - Fer clic a sobre d'ells, un rera l'altre, i mantenir a la vegada la tecla *Ctrl* pulsada.
 - Assenyalar amb el ratolí una àrea que els inclogui, tot arrossegant el ratolí (de manera similar a com es seleccionen les icones d'una àrea a l'escriptori).
- **Canviar el nom d'un element:** cal seleccionar-lo i picar F2. També es pot seleccionar l'element i tornar-hi a fer clic.
- **Canviar la grandària d'una classe:** cal seleccionar la classe i arrossegar amb el ratolí un dels petits quadres negres que hi apareixen. 

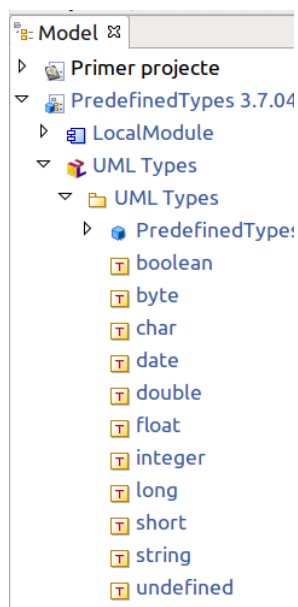
- **Modificar les propietats d'un atribut o una operació:** cal fer doble clic a sobre de l'element que volem editar. Sortirà una finestra com aquesta:



A la pestanya UML (la primera de la imatge) podem editar les diferents propietats de l'element.

Cal tenir en compte el següent:

- ◆ Tipus (Type): podem escriure qualsevol dels tipus UML predefinits. Els podeu veure a la pestanya *Model* de la part esquerra de la pantalla (com es veu a la imatge següent).



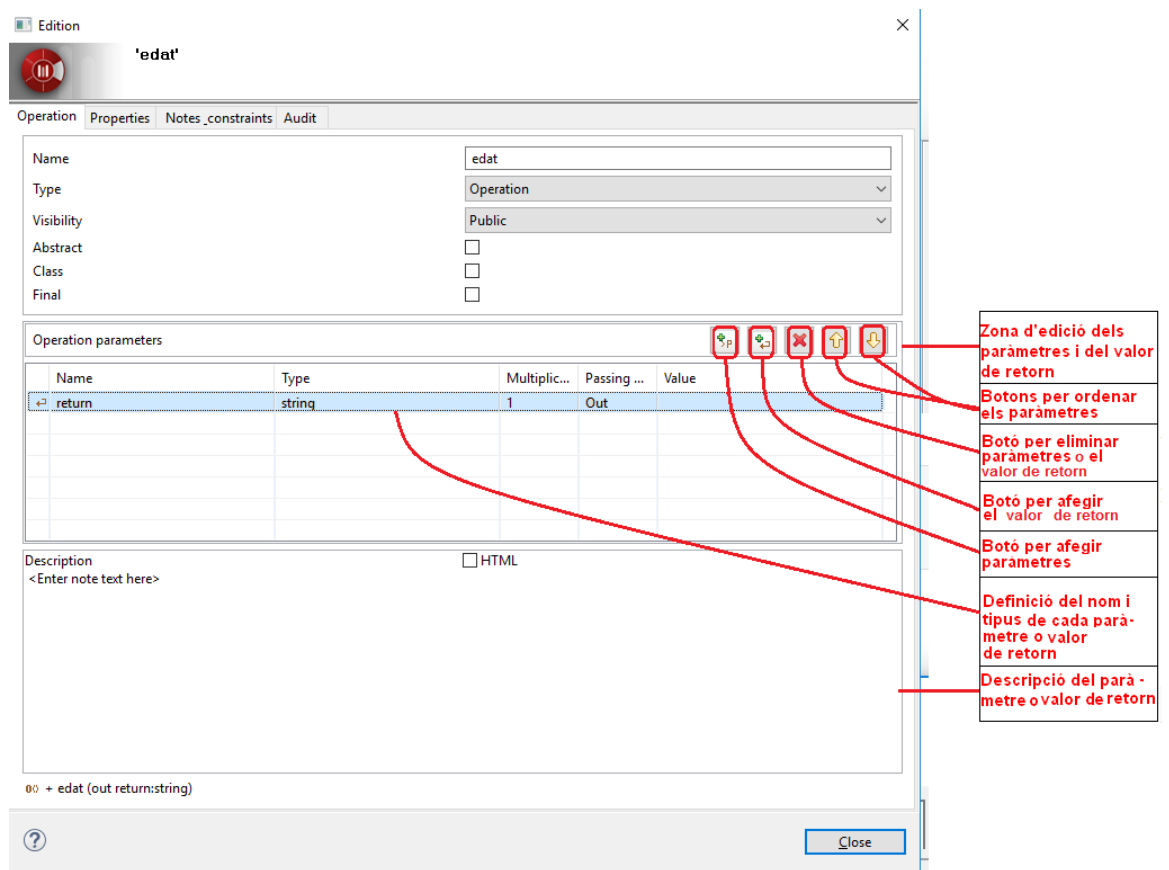
Després d'entrar el nom del tipus cal picar <Enter> perquè el sistema validi el tipus i, a més, enregistri el canvi realitzat.

Una altra manera d'entrar-los és deixar el quadre en blanc o amb la(es) lletra(es) inicial(s) del tipus i picar la combinació de tecles *Ctrl-barra espaiadora*; apareixerà un desplegable amb els tipus que encaixen amb el que hem teclejat.

◆ Edició d'operacions:

Apareix la pestanya *Operation* oberta i podem editar diferents característiques pròpies de les operacions (que no apareixen, per tant, als atributs). La principal són els paràmetres i el valor de retorn.




A la següent imatge, que podeu trobar a la pròxima pàgina, es veu la utilitat dels diferents controls.




• **Afegir-hi una associació:** cal seleccionar la icona associació (↔) i fer clic a totes dues classes de manera consecutiva. Un cop afegida, podem modificar les seves propietats fent-hi doble clic. Si treiem la selecció de la propietat *navigable*¹ desapareix la fletxa del gràfic. Les cardinalitats (multiplicity) també poden editar-se directament des del diagrama (sense obrir la pàgina de propietats).

¹La propietat *navigable* és per indicar des de quina classe de la relació es pot accedir a l'altra; per defecte (sense fletxes), les relacions són bidireccionals.


• **Afegir-hi una agregació o una composició:** cal fer el següent:

1. Seleccionar la icona corresponent ( per agregació o  per composició).
2. Fer clic en primer lloc a la classe base i després a una de les altres.
3. Modificar les propietats segons necessitem.
4. Seleccionar la icona de l'associació ().
5. Fer clic a una altra de les classes implicades.
6. Fer clic a l'agregació o composició.
7. Modificar les propietats segons necessitem (especialment *navigable* i les cardinalitats).
8. Repetir el procés per a cadascuna de la resta de les classes.



Cal tenir present que, quan volem crear una associació, agregació o composició, després de seleccionar la icona corresponent al tipus de relació, només podem fer clic a sobre dels elements que es posen de color verd en passar-hi per sobre.

• **Afegir-hi una associació N-ària:** cal seleccionar la icona *associació N-ària* () i, després a totes les classes implicades; a mesura que es fa clic, va apareixent una nova línia; per acabar, cal fer clic a la intersecció de les línies. Apareixerà un rombe (◊). Per editar les propietats cal fer doble clic a sobre d'aquest romb.

• **Afegir-hi una generalització:** cal:

1. Seleccionar la icona generalització ().
2. Fer clic a una de les subclasses i, després, a la superclasse (**en aquest ordre**); ens apareixerà una fletxa.
3. Seleccionar novament la icona *generalization*.
4. Fer clic a una altra de les subclasses i, després, a la primera fletxa insertada (que es posarà de color verd, igual que passava al cas de l'agregació i composició).
5. Repetir els dos passos anteriors per cada subclasse.

• **Afegir-hi una classe associativa:** a diferència d'altres programes de disseny UML, no pot fer-se en una sola operació; cal seguir les següents passes:

1. Crear la "classe" que contindrà els atributs de l'associació; es crea com si fos una classe normal.
2. Crear una associació (amb la icona ) entre les classes que volem relacionar.
3. Seleccionar la icona de la classe associativa (), fer clic a l'associació creada al pas 2 i a la classe creada al pas 1. No és possible equivocar-se en l'ordre perquè el programa només permet fer-ho en la seqüència correcta i es posa verd en cada moment l'element seleccionable en passar-hi per sobre amb el ratolí.

Algunes accions d'edició comunes a tots els diagrames

- ◆ **Seleccionar un element:** n'hi ha prou amb fer-hi clic a sobre o, si volem seleccionar-ne més d'un a la vegada, podem:
 - Fer clic a sobre d'ells, un rera l'altre, i mantenir a la vegada la tecla *Ctrl* pulsada.
 - Assenyalar amb el ratolí una àrea que els inclogui, tot arrossegant el ratolí (de manera similar a com es seleccionen les icones d'una àrea a l'escriptori)
- ◆ **Esborrar un element:** cal seleccionar-lo i, a continuació, fer clic a la tecla *Supr.*
- ◆ **Gravar el diagrama com un gràfic:** cal fer clic a la icona , que es troba a la barra de sobre del diagrama (al costat de la icona *impressora*), i ens apareixerà el diàleg habitual per desar fitxers.
- ◆ **Copiar el diagrama al porta-retalls com un gràfic:** cal fer clic a la icona , que es troba a la barra de sobre del diagrama i la imatge gràfica es copia automàticament al porta-retalls; a continuació pot enganxar-se a qualsevol programa d'edició.
- ◆ **Moure un element:** només cal posar el ratolí a sobre i arrossegar fins portar-lo a la posició desitjada.
- ◆ **Canviar la forma d'un element:** cal seleccionar l'element, posar el ratolí a sobre d'un dels quadres negres que apareixen després d'haver fet la selecció i arrossegar-lo fins aconseguir la forma desitjada.
- ◆ **Connectar dos elements amb un enllaç:** normalment només cal seleccionar la icona corresponent a l'enllaç a la barra d'elements i, després, fer clic als elements que volem unir en l'ordre adient; quan s'està fent aquest segon pas, l'element sobre el que tenim el ratolí apareix de color verd si podem fer-hi clic i de color vermell en cas contrari.
- ◆ **Gestió de la barra d'elements:**

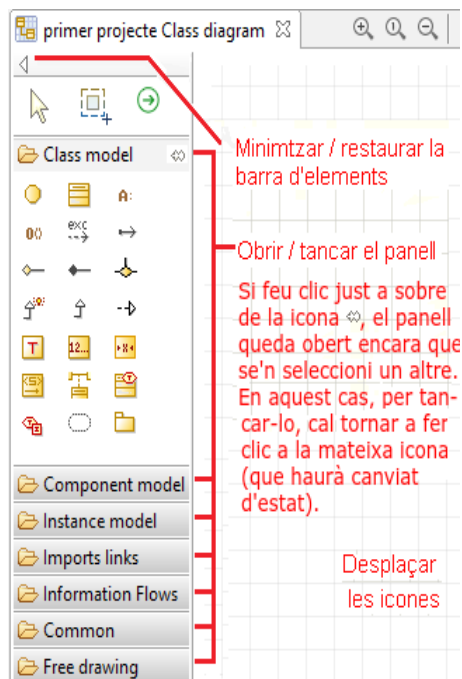

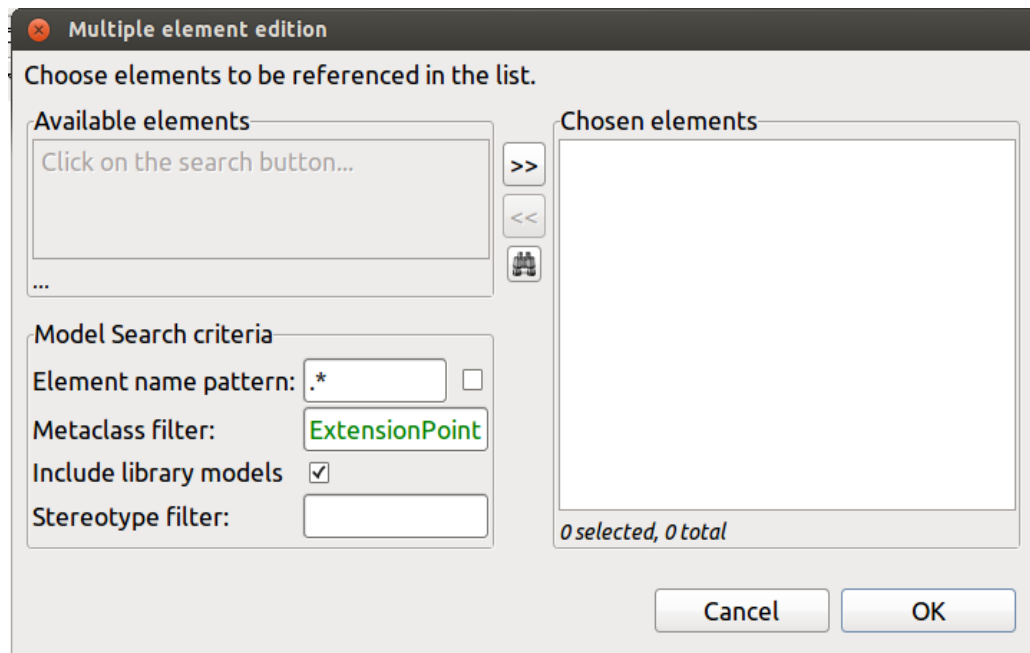




Diagrama de casos d'ús

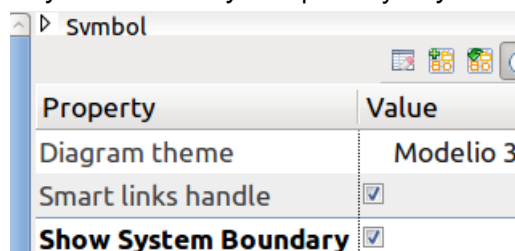
La manera de treballar és similar a la resta de diagrames: cal fer clic a la barra d'elements sobre la icona de l'element que volem representar i, a continuació, a sobre del diagrama. Coses a tenir en compte són:

- Per incloure punts d'extensió (*Extension point*) en un cas d'ús cal fer clic en primer lloc a sobre de la icona  i, a continuació, a sobre del cas d'ús on hi anirà. Un cas d'ús pot tenir més d'un punt d'extensió.
- Les línies dels diagrames de casos d'ús estan a l'apartat *Links*. Un cop afegida la línia, si correspon a un `<<Extend>>`, cal indicar a quin punt d'extensió correspon. Es fa seleccionant la línia i omplint la propietat *Extension point*. Veureu que, en posar-hi el ratolí, apareix una finestra com aquesta



En aquesta pantalla, en primer lloc, cal fer clic sobre el botó ; al quadre "Available elements" apareixeran tots els punts d'extensió del diagrama; cal seleccionar-lo(s), fer clic al botó >> i, per últim, al botó OK.



- Podem afegir una condició a un enllaç fent clic a la icona *constraint*  de l'apartat *Common* i a continuació en l'element al qual volem posar la condició. Els elements als quals és possible posar-la es mostren destacats amb color verd. Ens apareixen dos punts tancats entre claus (és a dir {;}). A continuació, a les propietats, podem omplir les propietats *Name* i *Body*. El que posem a la primera és el que sortirà abans dels dos punts; el que posem a la segona és el que sortirà després dels dos punts.
- Perquè es dibuixin els **límits del sistema**, cal fer el següent:
 - Clicar en primer lloc a un punt del diagrama sense cap element, per seleccionar el diagrama i, a continuació, fer clic a la columna *Symbol*, que apareix a la dreta.
 - Seleccionar-hi *Show System Boundary* a la pestanya *Symbol* que ens apareix, tal com es mostra a la figura següent:

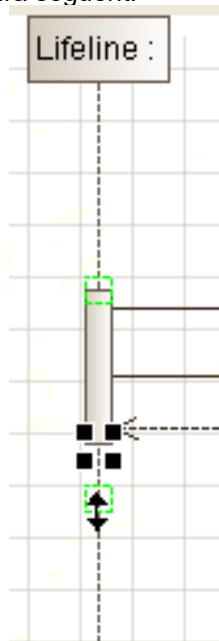


- Després podeu tornar a fer clic a *Symbol* per minimitzar aquesta columna.
- Per últim, podem canviar el nom al sistema fent clic al requadre que ens ha aparegut i modificant la propietat *Name*.

Diagrama de seqüència

També funciona de manera similar a la resta de diagrames:

- Per afegir-hi *línies de vida* només cal seleccionar la icona corresponent () a l'apartat *Nodes* i, a continuació, fer clic al diagrama. Podem donar-les nom modificant la propietat *Name*.
- Un cop afegides les línies de vida, podem afegir els missatges; són tots a l'apartat *Messages*. Per fer-ho, cal fer clic a les dues *línies de vida* implicades, primer la línia origen i després la línia de destinació. En fer-ho, s'afegeix en el destinatari un rectangle que representa una activació; a més, si el missatge era síncron, es crea automàticament un missatge de resposta.
- Poden afegir-se activacions directament seleccionant la icona corresponent () i fent clic a sobre de la *línia de vida*.
- Es pot canviar la longitud d'una activació tot seleccionant l'extrem que volem modificar i arrossegant amb el ratolí, tal com es veu a la figura següent.



Si hi ha un missatge que surt o arriba de l'activació, per canviar la longitud d'aquesta cal desplaçar verticalment el missatge.

Diagrama d'activitats

També funciona de manera similar a la resta de diagrames:














- Per afegir-hi els elements, es selecciona la icona corresponent de la barra d'elements (apartat **Control nodes**) i es fa clic al lloc on el volem posar:
 - Estat inicial 
 - Estat final 
 - Acció 
 - Fork / Join . Un cop afegit al diagrama, podem canviar la mida d'aquest element, arrossegant amb el ratolí el(s) seu(s) extrem(s). Es pot arrossegar quan s'ha seleccionat l'element (fent-hi clic), s'ha posat el ratolí a prop dels límits de la figura i té aproximadament aquesta forma:  o . Pot costar una mica. La manera més fàcil és:
 - Assegurar-se que l'opció *Smart links handle* està desactivada (podeu veure com fer-ho al breu apartat [treball amb projectes i diagrames](#)).
 - Acostar-nos amb el ratolí al join que volem pel costat esquerre. Veureu que és més fàcil aconseguir que agafi aquesta forma: .
 - Decisió .
- Per afegir-hi els fluxos de control, es selecciona la icona  de l'apartat *Flows* i, a continuació, es fa clic als dos elements del diagrama que es volen unir. Aquests elements poden ser qualsevol dels indicats als punts anteriors. Un cop afegits els fluxos de control al diagrama, pot modificar-se la seva forma tal com s'indica a l'apartat [Algunes accions d'edició comunes a tots els diagrames](#) (al punt "Canviar la forma d'un element").
- En una decisió cal posar la condició de cadascuna de les dues branques. Pot fer-se seleccionant la fletxa que representa el flux i omplint la seva condició "Guard".

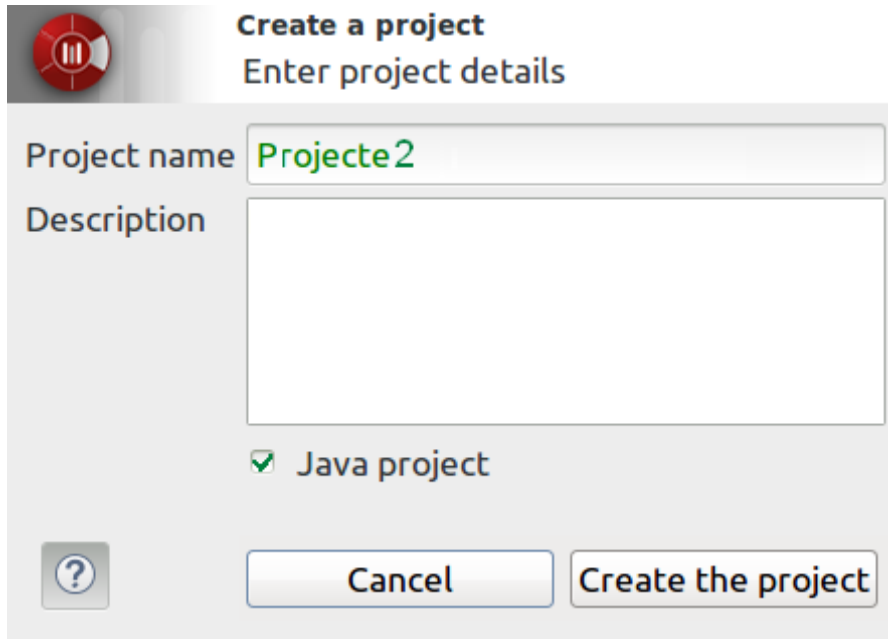
Diagrama d'estats

També funciona de manera similar a la resta de diagrames:

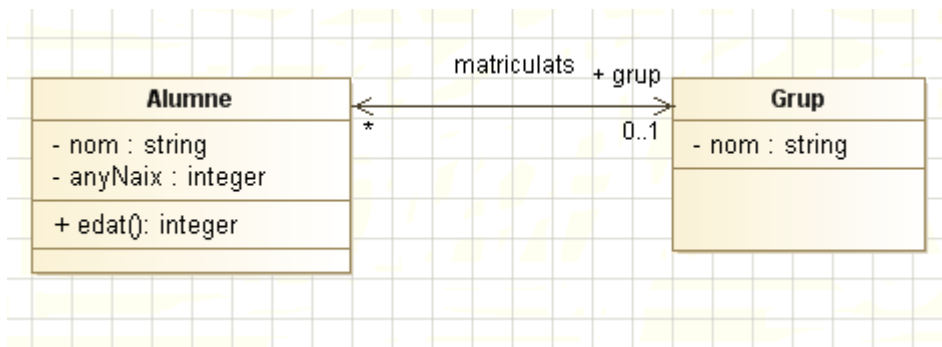
- Per afegir-hi els elements, es selecciona la icona corresponent de la barra d'elements (apartat **States**) i es fa clic al lloc on el volem posar:
 - Estat inicial 
 - Estat final 
 - Estat 
- Per afegir-hi les transicions, es selecciona la icona  del mateix apartat **States** i, a continuació, es fa clic als dos elements del diagrama que es volen unir. Aquests elements poden ser qualsevol dels indicats al punt anterior. El nom de cada transició s'entra en la seva propietat *Guard*.

Generació de codi a partir del diagrama de classes

En primer lloc, crearem un nou projecte. Aquesta vegada seleccionarem l'opció *Java Project*.



A continuació, crearem un diagrama de classes. Per exemple:

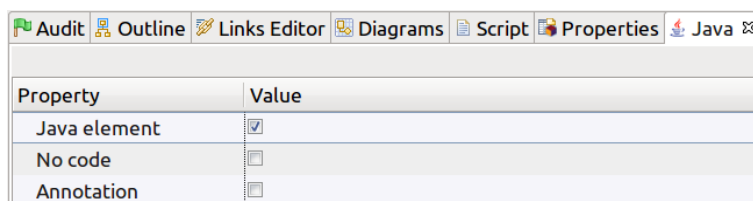


Si seguim el mateix exemple, cal parar atenció a les cardinalitats. Cal fixar-se també que s'ha assenyalat l'associació com a navegable pels dos costats perquè es generin els membres que permetin aquesta doble navegació.



El següent pas és fer que tots els elements que apareixen es considerin "*elements Java*".

Això es fa de la següent manera:

- Classes: cal seleccionar la classe i, a la pestanya *Java* de la part inferior, activar *Java element*

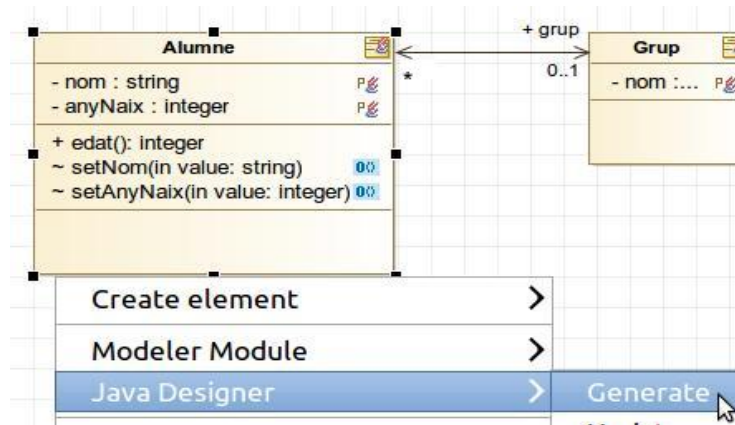


- Propietats: també cal seleccionar-les i, anàlogament, activar les caselles "Java Property", "Getter", "Getter visibility", "Setter" i "Setter visibility" com apareixen a la figura adjunta.

Després de fer-ho, ens apareixeran les classes i les propietats amb una icona  o  a la dreta. Indica que, a l'hora de generar codi es tindran en compte.

Property	Value
No code	<input type="checkbox"/>
Java Property	<input checked="" type="checkbox"/>
Wrapper	<input type="checkbox"/>
Getter	<input checked="" type="checkbox"/>
Getter Visibility	Public
Setter	<input checked="" type="checkbox"/>
Setter Visibility	Public
Static	<input type="checkbox"/>

- Associacions: cal assenyalar-la com a Java Property dins de les seves propietats, de manera similar a com hem fet amb els altres elements.
- Per generar el codi, cal anar a **cada classe** i seleccionar, des del menú contextual (surto amb el botó secundari) l'opció *Java Designer* → *Generate*



- Per veure el resultat, cal seleccionar, al mateix menú, *Java Designer* → *Edit*. S'obrirà una pestanya per la classe des de la qual hem triat l'opció.

Al nostre cas, el resultat ha estat aquest, respectivament:

```
import com.modeliosoft.modelo.javadesigner.annotations.mdl;
import com.modeliosoft.modelo.javadesigner.annotations.objid;

@objid("ea148390-1b93-4ee6-b5ea-96e2e70982d2")
public class Alumne {
    @mdl.prop
    @objid("b619ad4f-e045-4d54-90ec-a1e25a4a1662")
    private String nom;

    @mdl.propgetter
    public String getNom() {
        // Automatically generated method. Please do not modify this code.
        return this.nom;
    }

    @mdl.propsetter
    public void setNom(String value) {
        // Automatically generated method. Please do not modify this code.
        this.nom = value;
    }

    @mdl.prop
    @objid("5f7344ef-9cbf-4474-a520-9c2760267bc5")
    private int anyNaix;

    @mdl.propgetter
    public int getAnyNaix() {
        // Automatically generated method. Please do not modify this code.
        return this.anyNaix;
    }

    @mdl.propsetter
    public void setAnyNaix(int value) {
        // Automatically generated method. Please do not modify this code.
        this.anyNaix = value;
    }

    @mdl.prop
    @objid("13c5bbad-eb82-437f-8376-83a2fca94135")
    private Grup grup;

    @mdl.propgetter
    public Grup getGrup() {
        // Automatically generated method. Please do not modify this code.
        return this.grup;
    }

    @mdl.propsetter
    public void setGrup(Grup value) {
        // Automatically generated method. Please do not modify this code.
        this.grup = value;
    }

    @objid("28f1391c-5152-4b82-890c-beb8c32fcae3")
    public int edat() {
    }
}
```

```
import java.util.ArrayList;
import java.util.List;
import com.modeliosoft.modelo.javadesigner.annotations.mdl;
import com.modeliosoft.modelo.javadesigner.annotations.objid;

@objid("24a5fbd8-dccd-40e2-9826-8209eb4a06e6")
public class Grup {
    @mdl.prop
    @objid("338ab93b-e6f7-400e-bf3a-f8196f24e84a")
    private String nom;

    @mdl.propgetter
    public String getNom() {
        // Automatically generated method. Please do not modify this code.
        return this.nom;
    }

    @mdl.propsetter
    public void setNom(String value) {
        // Automatically generated method. Please do not modify this code.
        this.nom = value;
    }

    @objid("75030486-5d6f-4c37-b3ec-5cb131c6d884")
    public List<Alumne> = new ArrayList<Alumne> ();
}
```

Com es veu, el resultat és bastant bo. Tot i així, cal fer algun retoc a mà a la classe *Grup*; concretament, l'ArrayList de la classe Grup no té nom, és públic i li falta el getter. A més, com era d'esperar, a la classe *Alumne* cal completar el mètode edat.

Veiem també que utilitza unes anotacions pròpies per incloure informació per gestionar internament els diferents elements.

Generació de classes a partir del codi

En aquest apartat veurem com generar el diagrama de classes a partir del codi.

Concretament, afegirem les següents classes al diagrama de classes:

```
import java.util.ArrayList;
import java.util.List;

public class Botiga {

    private String nom;

    private String adreca;

    private List<ClientHabitual>
        clients = new
        ArrayList<ClientHabitual>();

    public String getNom() {
        return nom;
    }

    public void setNom(String nom) {
        this.nom=nom;
    }

    public String getAdreca() {
        return adreca;
    }

    public void setAdreca(String
        adreca) {
        this.adreca=adreca;
    }

    public List<ClientHabitual>
        getClients() {
        return this.clients;
    }

}

public class ClientHabitual {

    private String nom;

    private String adreca;

    private Botiga botiga;

    public String getNom() {
        return nom;
    }

    public void setNom(String nom) {
        this.nom=nom;
    }

    public String getAdreca() {
        return adreca;
    }


    public void setAdreca(String
        adreca) {
        this.adreca=adreca;
    }

    public Botiga getBotiga() {
        return botiga;
    }

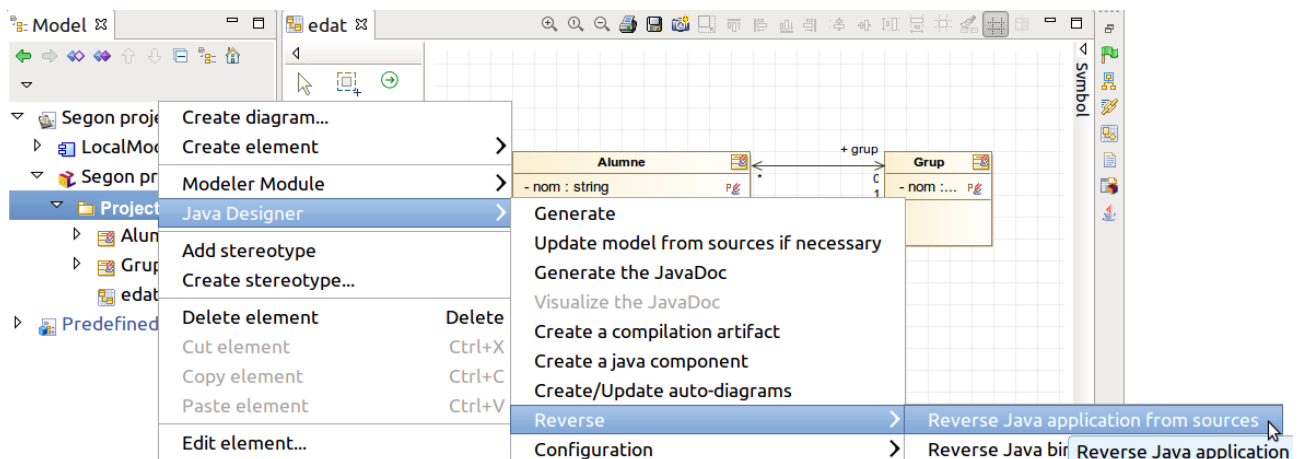
    public void setBotiga(Botiga
        botiga) {
        this.botiga=botiga;
    }

}
```

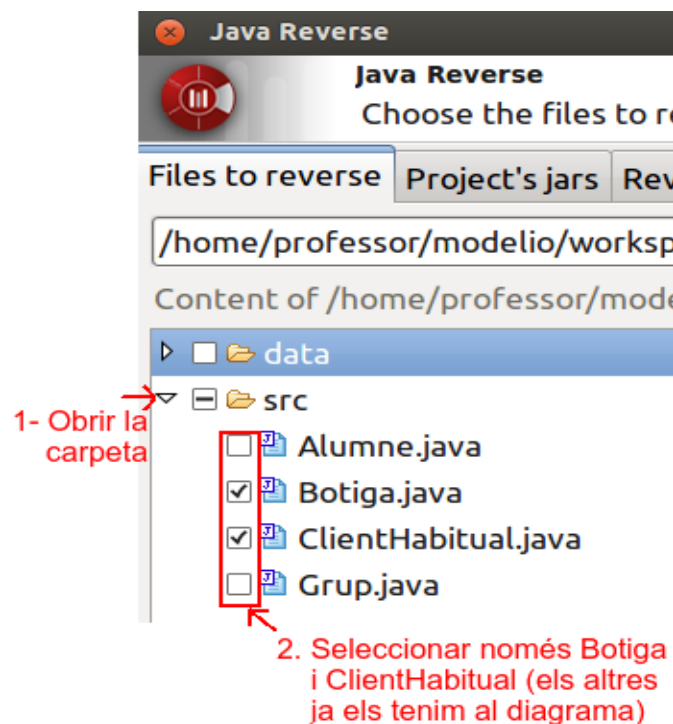
Aquestes classes no tenen res a veure amb les anteriors.

Per fer-ho, crearem a la carpeta *src* del projecte un fitxer *.java* per cadascuna de les classes amb el contingut corresponent. La carpeta del projecte es troba seleccionant el projecte a l'explorador i clicant la icona *Project configuration*, .

A continuació, seleccionem l'opció *Java Designer* → *Reverse* → *Reverse Java application from sources* (com veieu, també pot fer-se a partir de fitxers *.class* o *.jar*) com es mostra a la imatge següent.

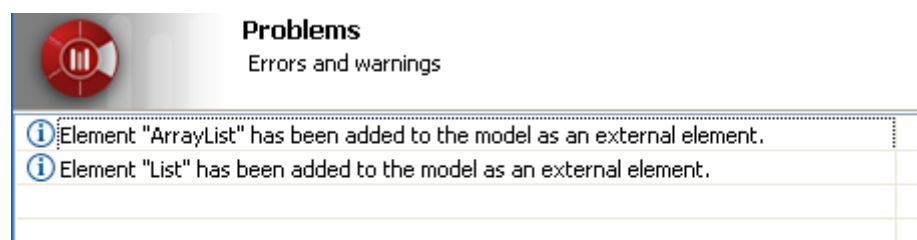


Ens sortiran unes pantalles com les següents on hem de seleccionar els fitxers amb les classes que volem afegir al nostre projecte:

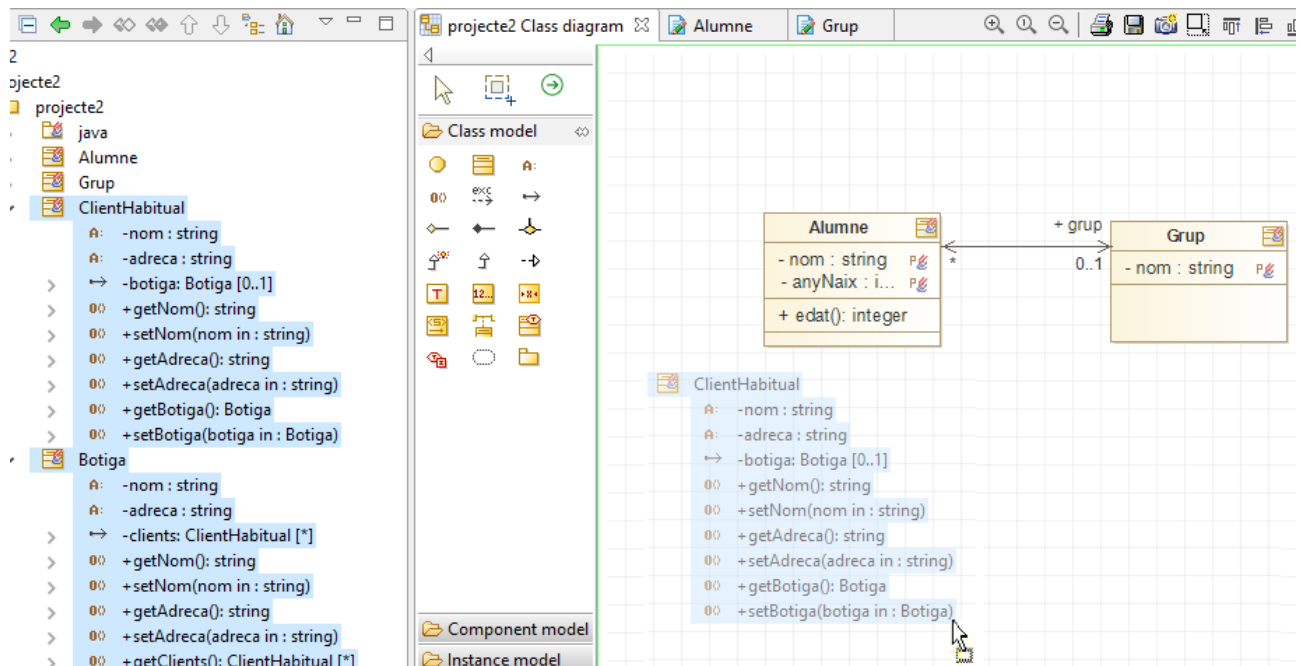


Seguint l'assistent, cal fer clic a *Next* un parell de vegades tot deixant les opcions que ja surten per defecte.

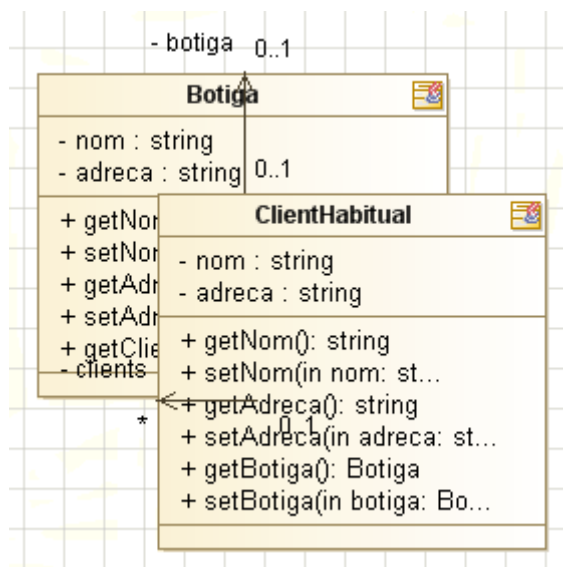
A la pantalla que ens surt hem de fer clic a *Reverse* i ens sortirà aquesta pantalla, que indica que tot ha anat bé:



Podem comprovar-ho a la pestanya *Model* (a la part esquerra de la pantalla), on veurem que s'han afegit les dues classes al projecte. Ara només resta incorporar-les al diagrama de classes. Ho farem desplegant-les, seleccionant-les totes (amb els seus membres inclosos) i arrossegant-les cap al diagrama. Per seleccionar-les totes n'hi ha prou amb fer clic a la primera, desplegar-les totes i, mantenint la tecla de *majúscules* premuda, fer clic al darrer element que volem incorporar. Gràficament:

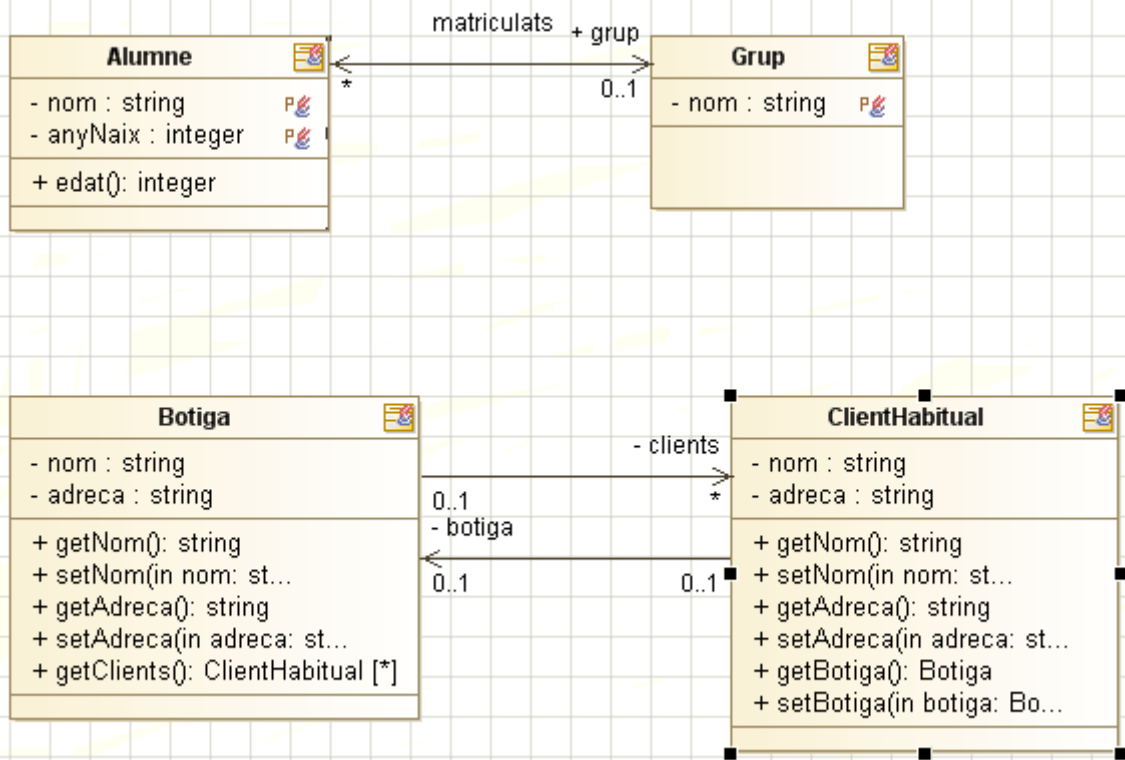


Les noves classes presenten un aspecte similar a aquest:



Com veiem, les classes surten amuntegades.

Podem endreçar-les, tot movent les classes de la manera que s'ha vist abans. El resultat serà semblant al mostrat a la imatge següent.



Com hem afegit classes al projecte anterior, ens apareixen les classes que ja tenia el projecte i les que acabem d'incorporar.

Errors coneguts

Problemes amb els tipus primitius

Us podeu trobar:

- Bé que en crear el projecte us aparegui un error similar a aquest (s'utilitza X, Y i Z per referir-nos a nombres de versions):

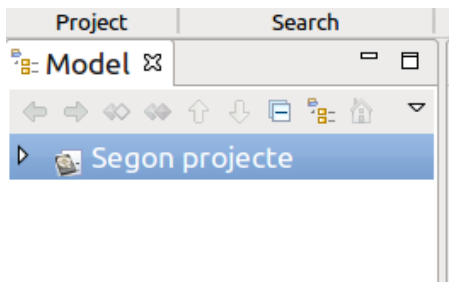
```
java.io.IOException: The 'PredefinedTypes X.X.XX' fragment's 'Standard vY.Y.YYYY' metamodel is outdated compared to the current 'Standard vZ.Z.ZZZZ' version.
```

.....

- Bé que us aparegui (per exemple, a la pestanya *Audit* de la part inferior dreta) un error semblant a aquest (el tipus primitiu podria ser un altre): "The 'integer' element is missing.

Aquests problemes es deuen a que el projecte no té el mòdul dels tipus primitius correctament configurat. Es soluciona de la següent manera:

1. Obriu el projecte i seleccioneu-lo.



2. Seleccioneu l'opció del menú *Configuration --> Modules*. Apareixerà una llista dels mòduls que utilitza el projecte. Aquí poden passar dues coses:

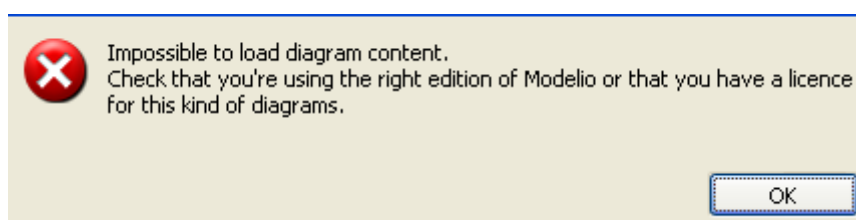
- Que no hi aparegui el mòdul *ModelerModule*.
- Que aparegui el mòdul *ModelerModul*, però que a la columna *Compatibility* aparegui el text en blau *Update recommended*.

Si apareix, cal seleccionar-lo i fer clic al botó *Remove...* per eliminar-lo del projecte.

Tant si no apareix com si l'hem eliminat, cal fer clic al botó *Add...*, seleccionar *ModelerModule* de la llista (a la columna *Compatibility* ha d'aparèixer *Compatible*) i fer clic al botó *Deploy in the project*. Veureu que ara ja apareix correctament a la llista de mòduls, que al projecte apareix un nou element anomenat *PredefinedTypes x.x.x* i que desapareixen els avisos de la pestanya *Audit*.

Problema en obrir: Impossible to load diagram content

Si en obrir un projecte us surt el missatge que hi ha a continuació, probablement es tracta d'un error conegut que es soluciona tancant consecutivament les pestanyes dels diagrames oberts fins que desaparegui l'error; després ja podem tornar a obrir el diagrama que ens interessi.



No es mostren a la classe els atributs afegits.

S'ha observat que si es copia directament (arrossegant-lo amb el ratolí a la vegada que es prem la tecla Ctrl) un atribut des d'una classe a una altra, en algunes ocasions aquest atribut no es mostra a la nova classe. La manera de fer que es mostri és afegint un atribut nou des del quadre de l'esquerra on són els elements dels diagrames de classe (en podeu veure una part a la dreta d'aquest text).

