

# Borland Together Architect 1.0

가

Together 가 .  
Together . 가

:

- 가
- : Together
- : Together
- 가

가

Together 가 . Together (model diagram)  
(class diagram)

가 , , ,  
가 Java™ , C++ , Visual Basic® , C#  
. Together

Together 가 Together

- Together
- 

:

가 .



가

Together [www.borland.com](http://www.borland.com), [www.borland.co.kr](http://www.borland.co.kr)

가 .

. Together Architect 15

Together 가 .

<http://bdn.borland.com/together> .

---

Copyright © 2005 Borland Corporation. All rights reserved.

: 2005 3 4

# Together

1 :

Together

CashSales

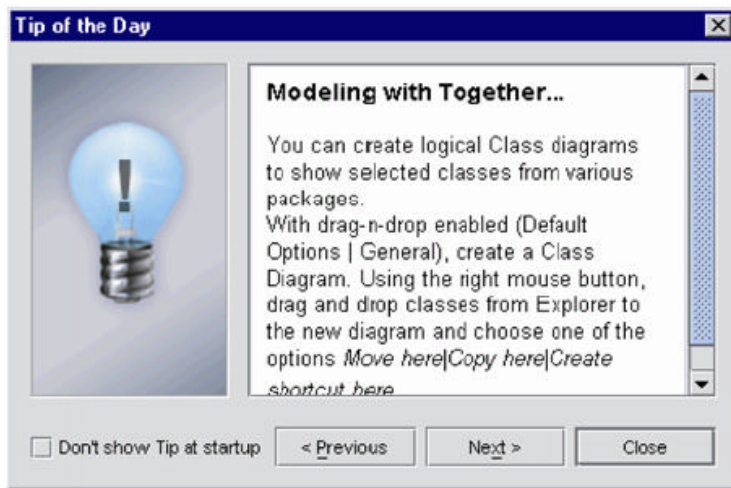
Java

Together

- - 
  - 
  - Pane
  - Properties Inspector
- 

Together “Tip of the Day”

. Together가



Together

Together

Explorer pane

Explorer pane

Explorer pane

Together

Directory

가

Together

Together

Together

: Java **CashSales**

Together

CashSales

**Samples | Java | CashSales**

Directory

CashSales.tpr

Explorer

Directory

C:\

Samples

C++

csharp

java

CashSales

data\_management

problem\_domain

Requirements

server

user\_interface

util

Architecture View.dfClass

Architecture View.txvcls

CashSales.tpr

CashSales.tws

Component.dfComponent

Component.txvcmp



CashSales

Pane

. Together

Pane

. Pane

	Explorer pane	
	Editor pane	
	Designer pane	UML
	Message pane	가

CashSales

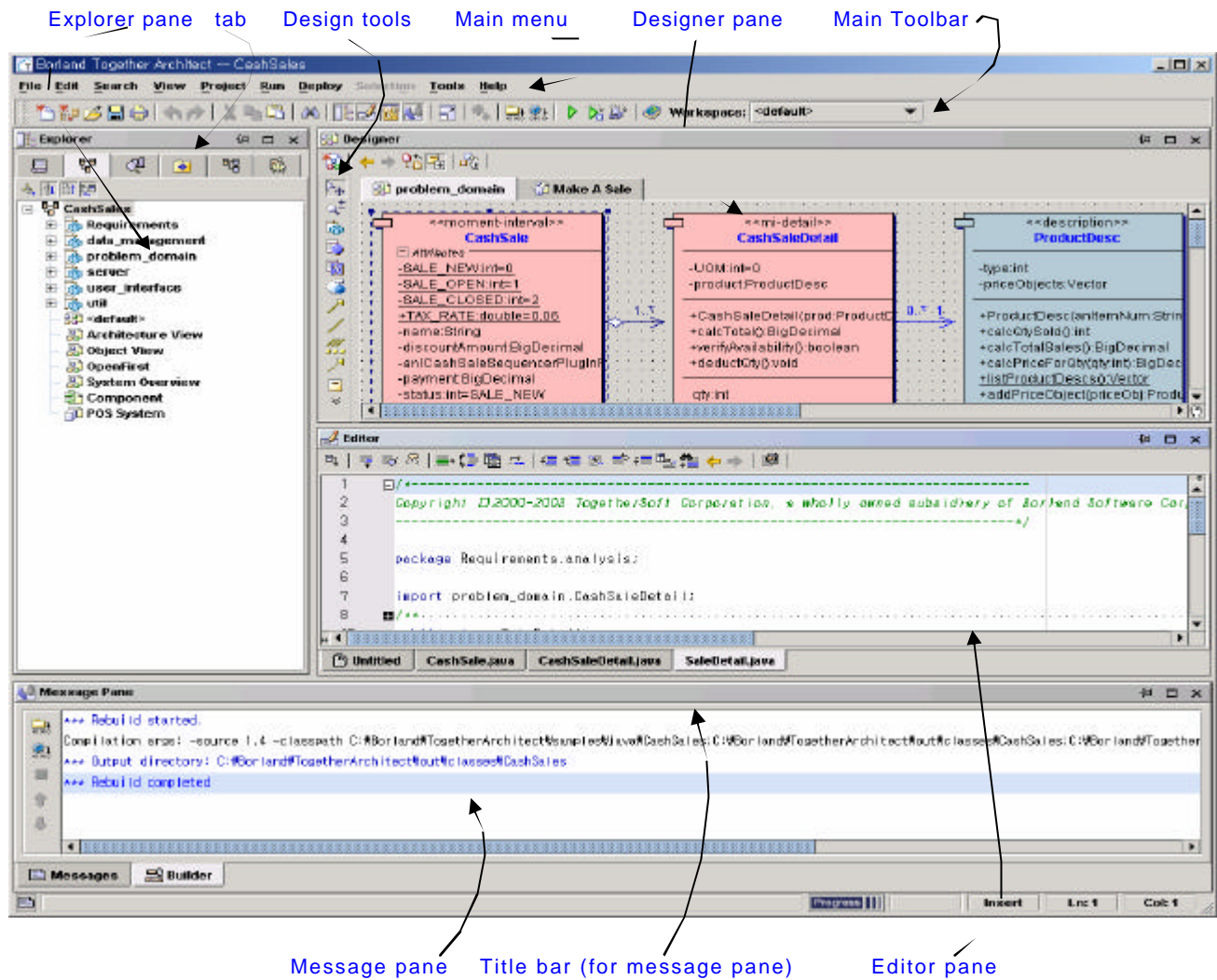
Java

가

Together

. Designer pane

가



가 Editor pane  
 Pane  
 Pane  
 Pane  
 가 Pane

File	가
Edit	“ ” / ( )

가 Pane,



Editor pane

Editor

- 1. Editor pane
- 2. Editor pane
- 3.

pane

(toggle) . Pane

Pane

Pane Pane

Designer pane

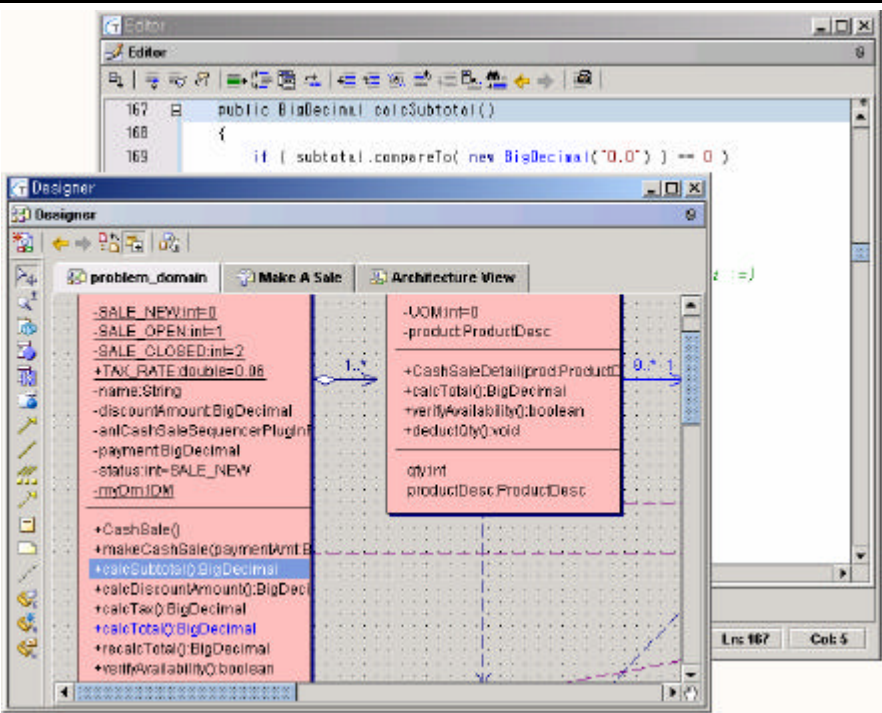
Designer pane

- 1. Designer pane
- 2.

Designer pane

Designer Editor가

Pane



Pane





Pane

## Property Inspector

Together

( )가 . ( (class node) (link)  
Together 가 .  
property Inspector가 .

: **CashSales** property Inspector .

property Inspector Designer Explorer

pane

**Properties** ProductDesc

Inspector가

( ) Inspector

Inspector Explorer pane . Inspecotor

( )

Select in Model Tree

New

Add to Favorites

Properties... Alt+Enter

Rename F2

Add Linked...

Search for Usages...

Inspector: ProductDesc

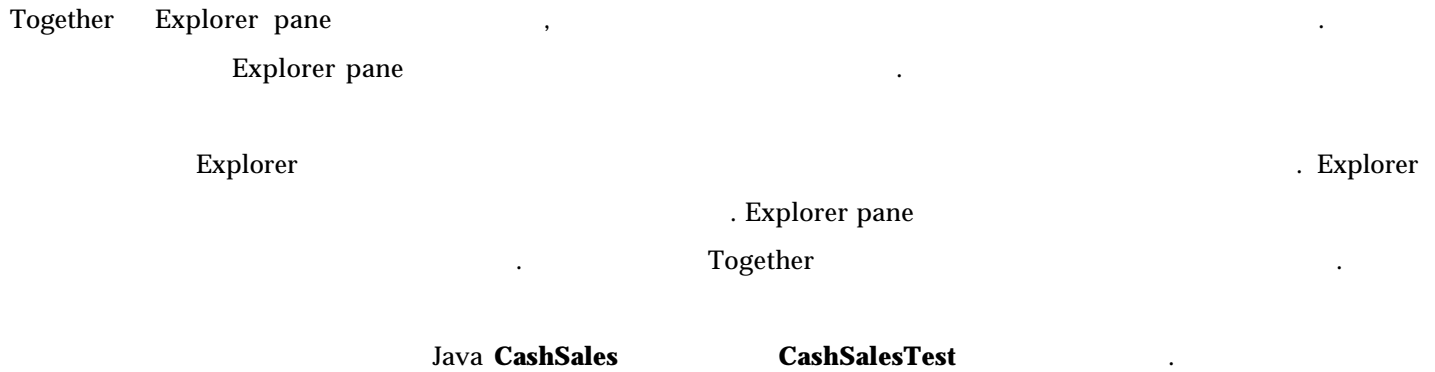
Inspector: ProductDesc

Properties	Hyperlink	View	Description
Name			Value
name			ProductDesc
package			problem_domain
stereotype			description
alias			
file			ProductDesc.java
public			<input checked="" type="checkbox"/>
final			<input type="checkbox"/>
abstract			<input type="checkbox"/>
extends			
implements			
invariants			
persistent			<input checked="" type="checkbox"/>

Press Ctrl+Alt+I to finish editing and close Inspector

## Together

## 2 : Explorer Pane





- :
- Explorer Pane
  - Directory :
  - Model :
  - Server, Favorites, Diagram :
  - Component :
  - Module : Together
  - Testing :

---

### Explorer Pane

Explorer pane

	Directory	
	Model	
	Servers	Together
	Favorites	
	Diagram	
	Components	가
	Modules	
	Xplorer	XML

	UI Builder	
	Testing	

Directory Modules . 가 . UI Builder, X(ML)plorer, Testing 가 .

: Java **CashSales**

### Directory :

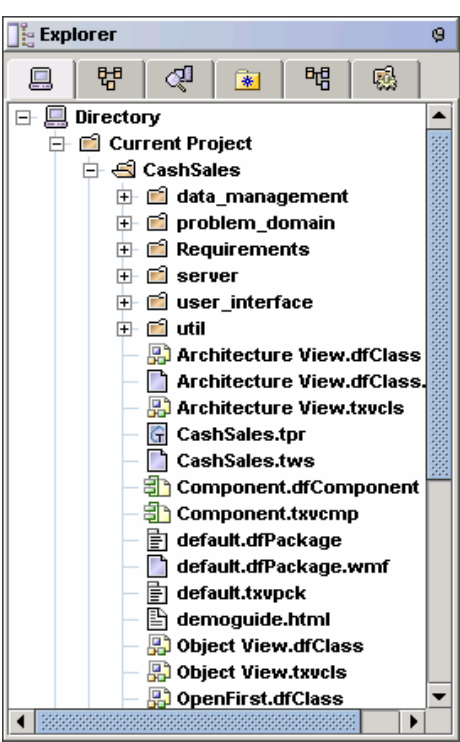
Directory

Directory
Together
가
가

- Current project( 가 )
- 
- Samples – Together
- User projects – Together
- Template – C++, C#, Java, CORBA IDL, Visual Basic, VB.NET Together

Current Project 가

: Explorer pane
Java CashSales
demoguide.html



: naviator.config
Directory

### Model :

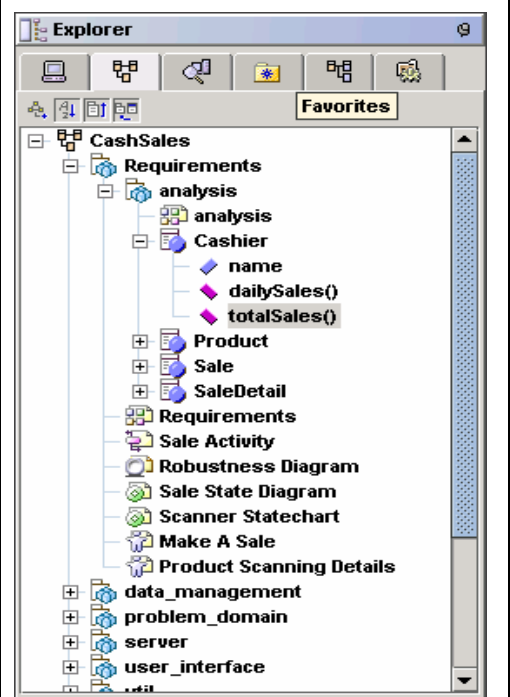
Model 가 . 가





UML (sequence diagram) “ “

Model

UML	Java
Sale State Diagram	CashSales
Requirements	Requirements
analysis	analysis
Cashier	Cashier
name	name
dailySales()	dailySales()
totalSales()	totalSales()

Designer pane



Model	
	가
	
	
	

**Server, Favorites, Diagrams :**

Server

Framework가

Server

Test Suites

Testing

Favorites

Diagram

pane

Designer

가

Server , 가

Favorites ,

Diagrams

: Favorites

problem\_domain

ProductDesc

가

( )

Favorites 가

1. Explorer Model

2. **proble\_domain** Add to Favorites

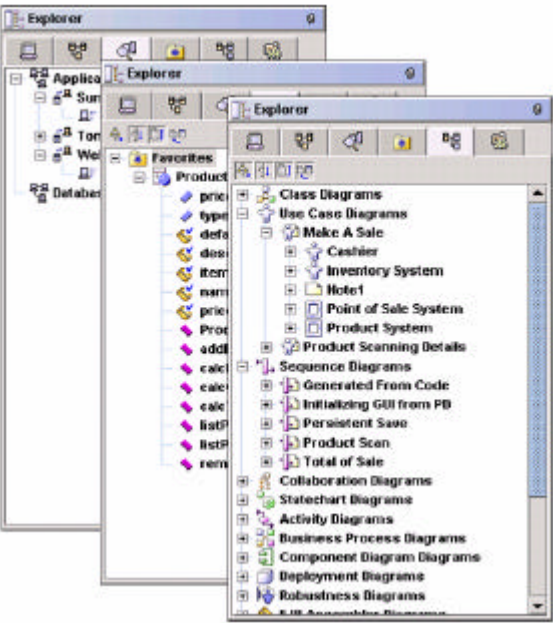
3. **ProductDesc** Add to Favorites

Favorites 가 Editor

Designer (class node)

Favorites Designer

View | Explorer



Component :

Component (component model) . “ 가 ”

가 :

1. **Project | Properties**

2. **Include Components**

Explorer Project Properties Component Together Coad

가 .

: Editor Coad

**CustomerAccDesc** .

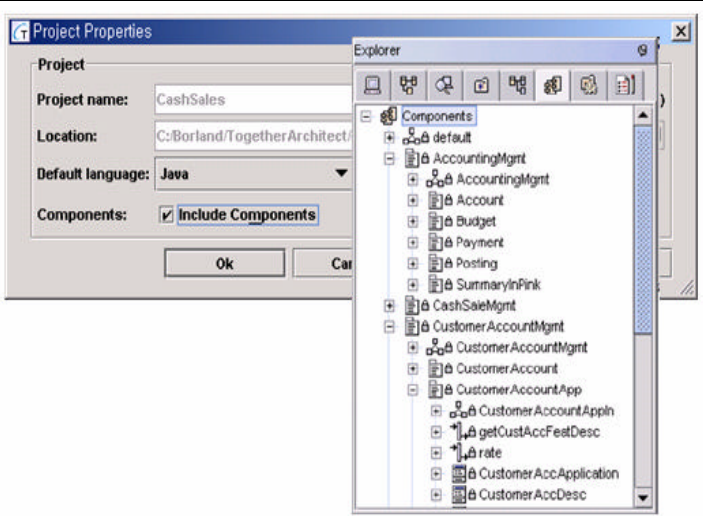
- 1.
2. CustomerAccountMgt | CustomerAccountApp
- CustomerAccDesc .

Editor 가

Together modules/components

. Explorer Component

component



**Modules : Together**

Java API Together API Together

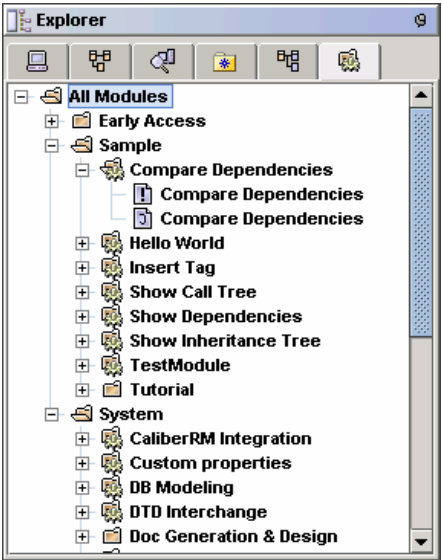
Together Java

\$TGHS/modules/com/togethersoft/moduls

Module Together , 3

: Together Hello World

1. Modules Explorer pane
2. (+) Sample | Hello World
3. Hello World(Java)( )
- Run



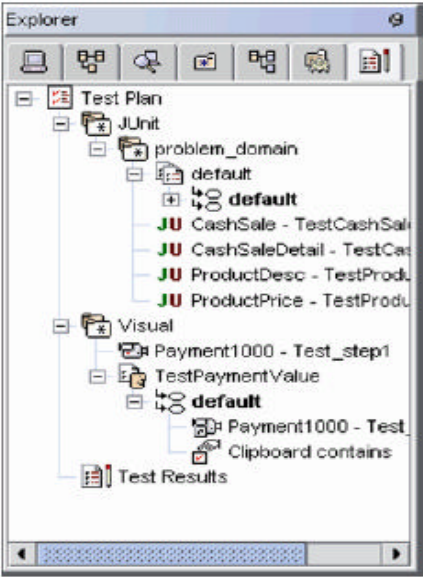
Explorer Modules

- ( )
- Java
- Java
- Tcl ( )

Sample/Tutorial Editor Message pane

Testing :

Together		CashSalesTest
Explorer	Testing	.
<div>:</div>		
<div>1. Directory</div> <div>2. <b>Samples   java   CashSales.Test</b></div> <div>3. <b>CashSalesTests.tpr</b></div> <div>4. Together</div>		<div>. Yes</div>
CashSales	가	CashSalesTests가
(Together		.) Testing
Framework		Explorer Pane
Testing		





Together

3 : Together

Together . Explorer pane  
. Editor pane  
Explorer, Designer, Editor pane . Designer Editor pane

CashSaels

- :
- Explorer
  - Explorer
  - Explorer Designer , Editor
  - Designer

Explorer

Designer pane

: CashSales Java	Total of Sale	(sequence diagram)
------------------	---------------	--------------------

Together가Designer pane

File | Open

Explorer가Model ( )

Diagram ( )

Explorer( ) Designer pane

ExplorerOpen in New Tab  
Designer

Total of Sale  
Modelproblem\_domainDiagramSequence Diagrams,

Open

Open in New Tab

New

Add as Shortcut

Add to Favorites

Properties...Alt+Enter

RenameF2

Update Package Dependencies

CutCtrl+X

CopyCtrl+C

CloneCtrl+Shift+C

PasteCtrl+V

Paste ShortcutCtrl+Shift+V

DeleteDelete

Hyperlink To

Tools

가 가 ,

. Together UML

	/ (class)		(sequence)		(state chart)		(component)
	(use case)		(collaboration)		(activity)		(deployment)

Explorer Model . Designer pane

Together가EnterpriseJ2EE

: Sale Activity

Designer pane

가

Designer pane

Close

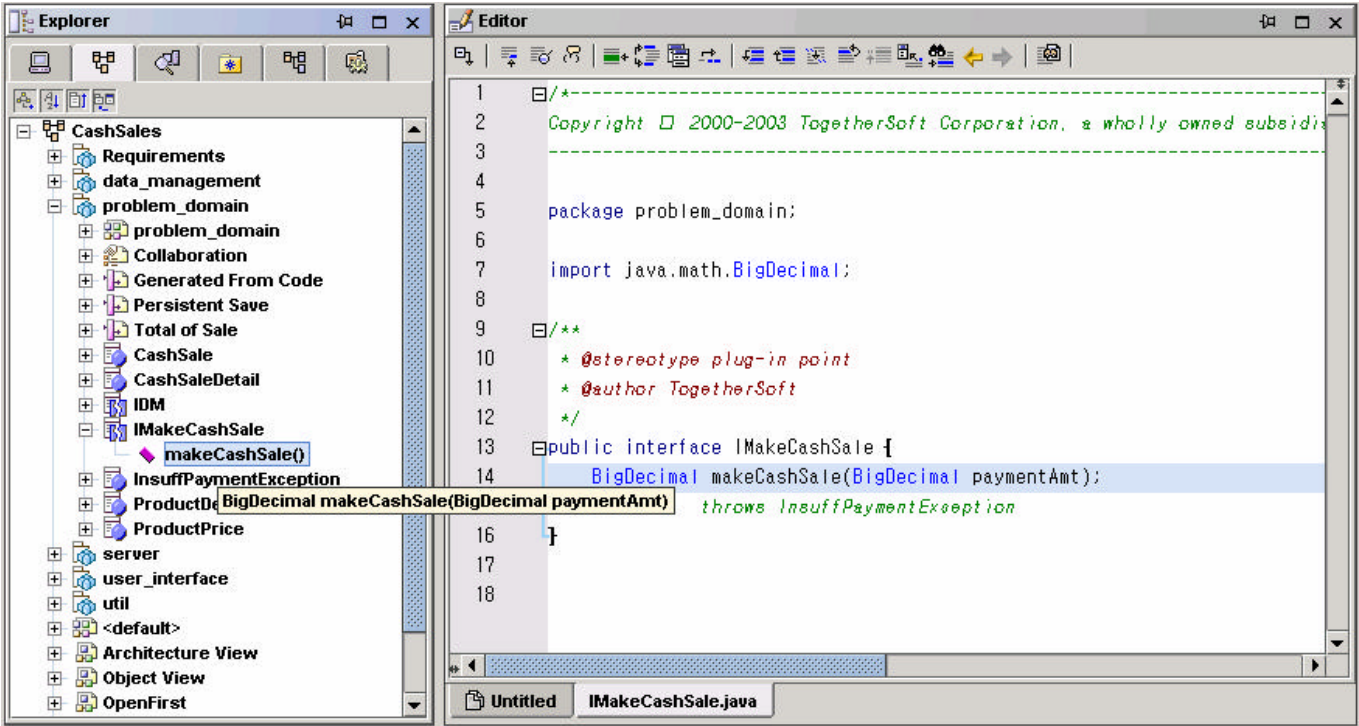
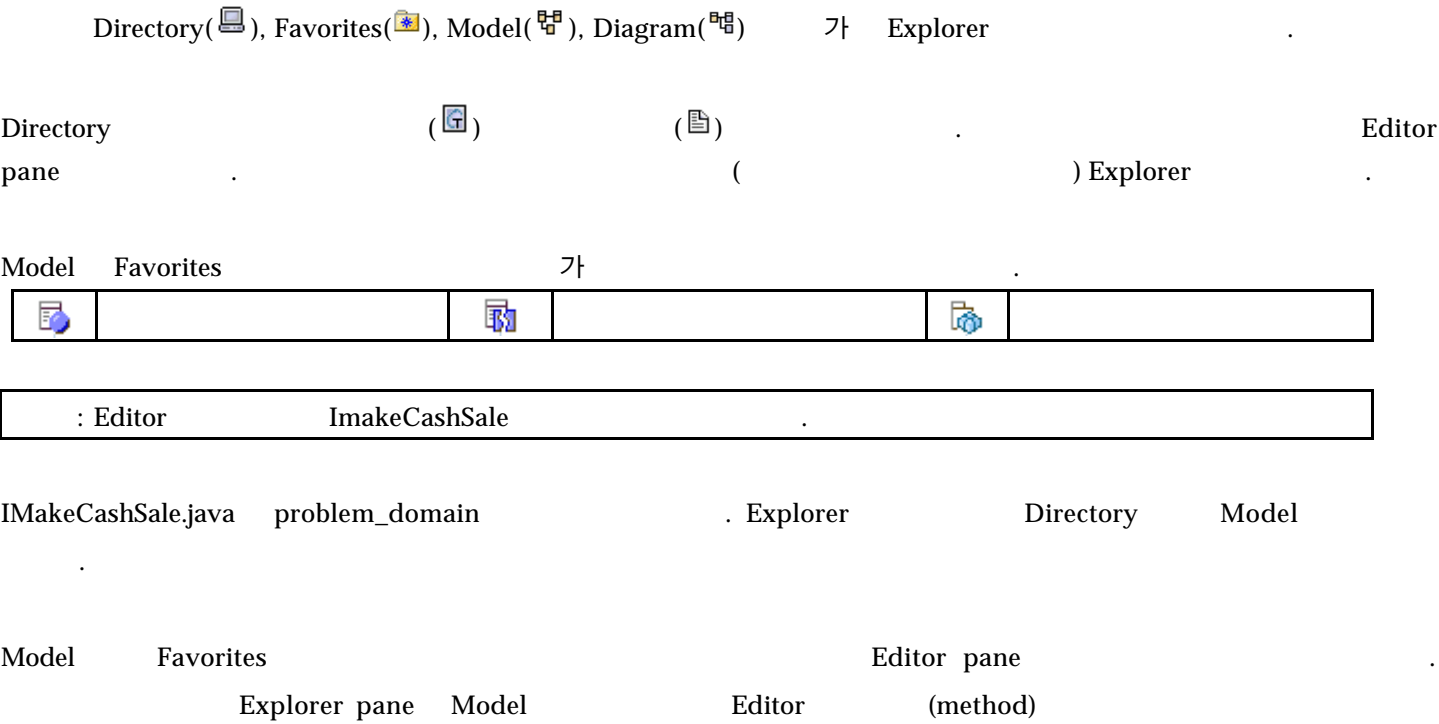
Ctrl + F4

Sale Activity

Close

Close All Diagrams

Explorer

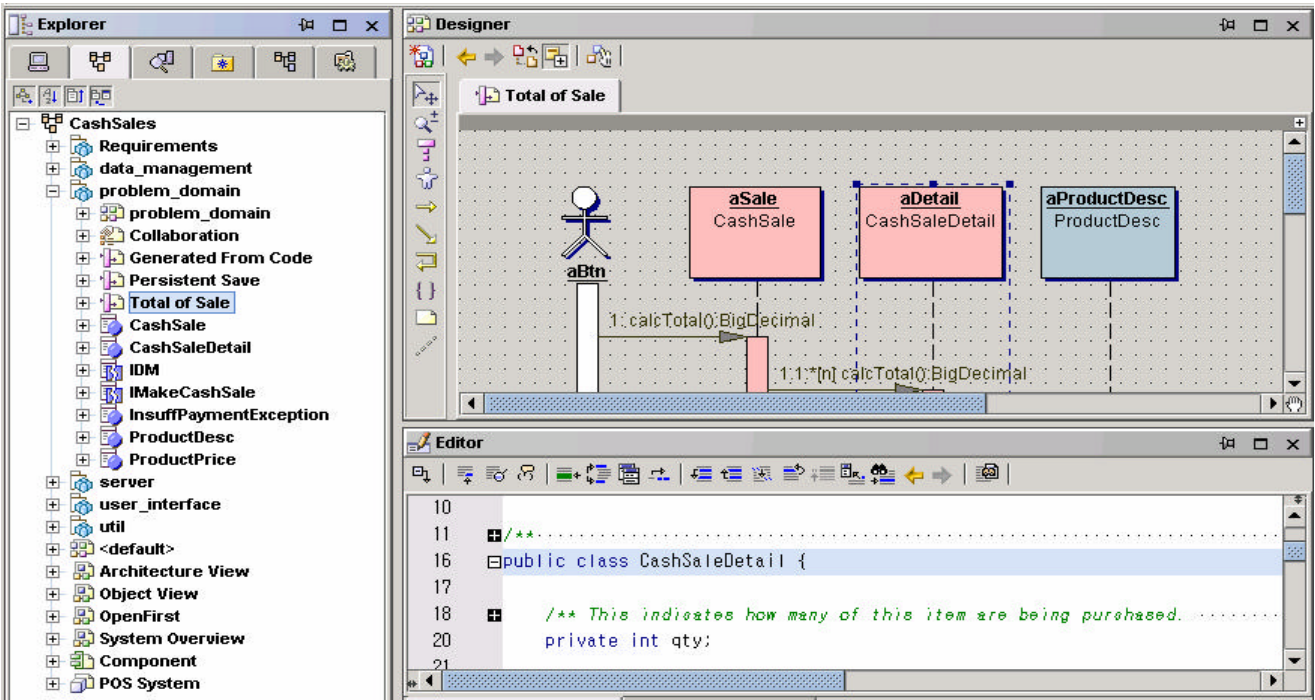


Explorer      Designer ,      Editor

. CashSales

: Designer pane      Editor      . Total of Sale  
CashSaleDetail.java

- 1. Explorer pane      Model      problem\_domain
- 2. Model      Total of Sale
- 3.      Designer pane      aDetail
- 4. Designer pane      aDetail      Editor



: Total of Sale      CashSale.calcTotal()

Editor

가      Editor

Editor      Designer

: Explorer    Editor    POSFrame\_AboutBox.java    Editor    user\_interface

Explorer    Model    user\_interface  
POSFrame\_AboutBox    Editor

Editor      Designer      :

1.                                    가

2.                                    Select in Diagram

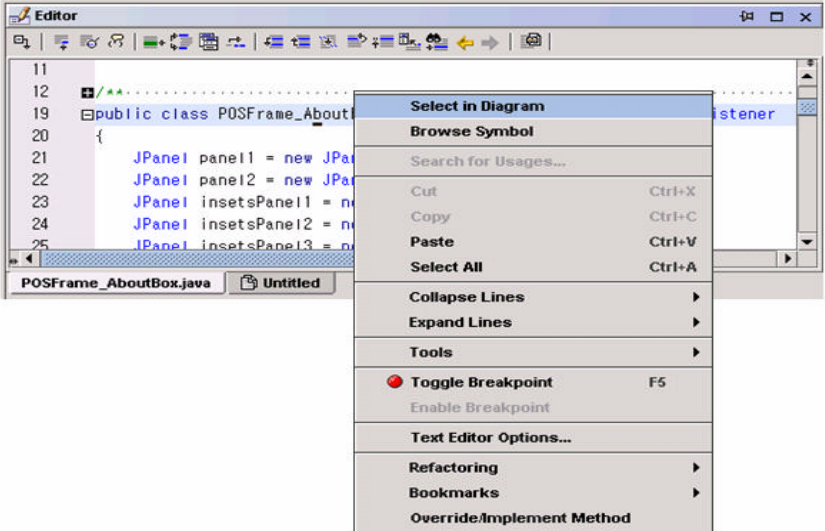
Together

, Together

Designer pane

(method)

Editor



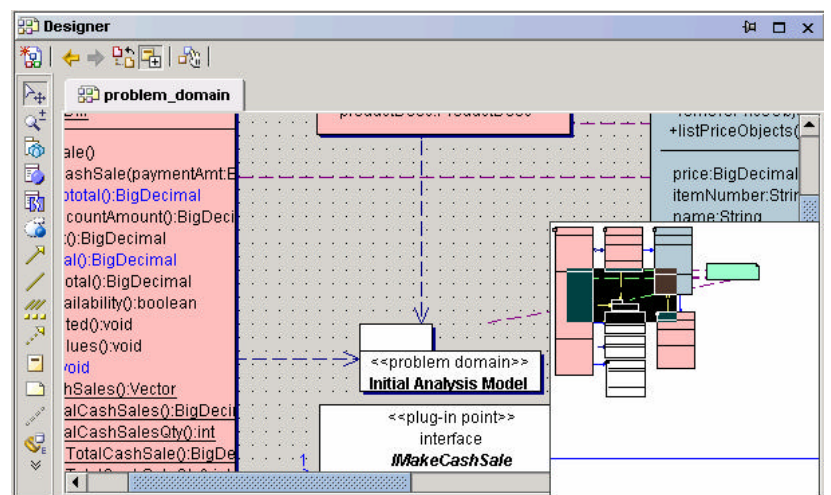
Designer

Designer                                    ( )  
   가

: Pane                                    가                                    Designer pane

- |  |
|--|
|  |
|--|

가



# Together

4 : , ,

configuration , Together 가  
Inspector Options 가  
Together .

CashSales .

:

- 
- 
- 
- 

(workspace)

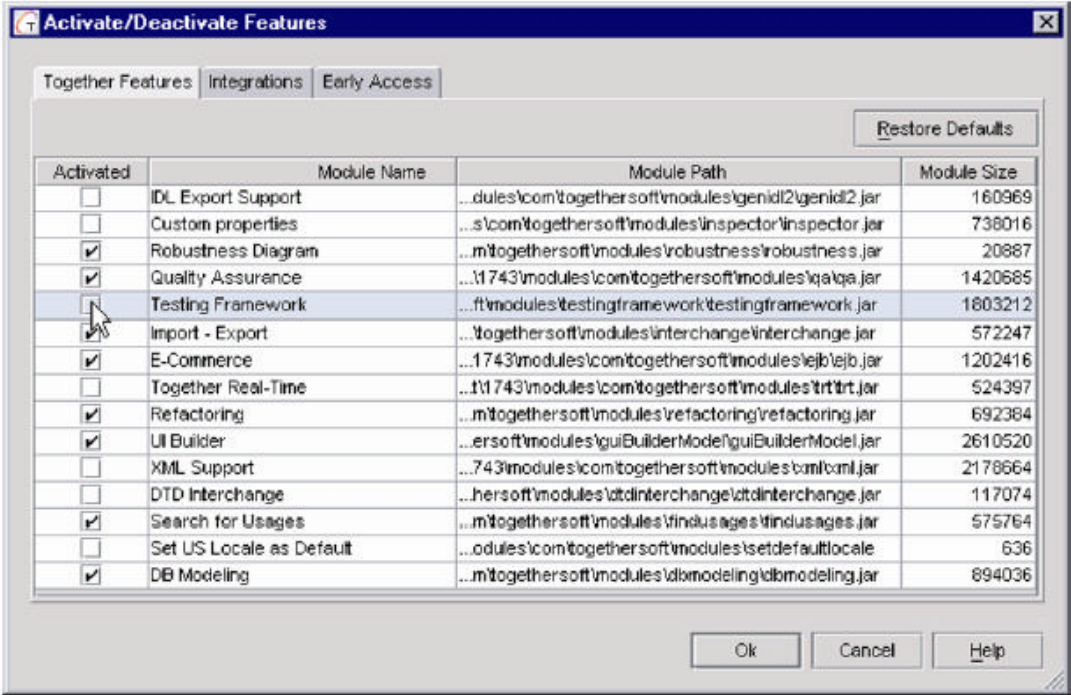


Together ( ) 가

: UI Builder	. Testing Framework
--------------	---------------------

Together .

1. **Tools | Activate/Deactivate Features** .
2. .



: Together



Tools | Options

Default, Project, Diagram

Diagram

Project, Default

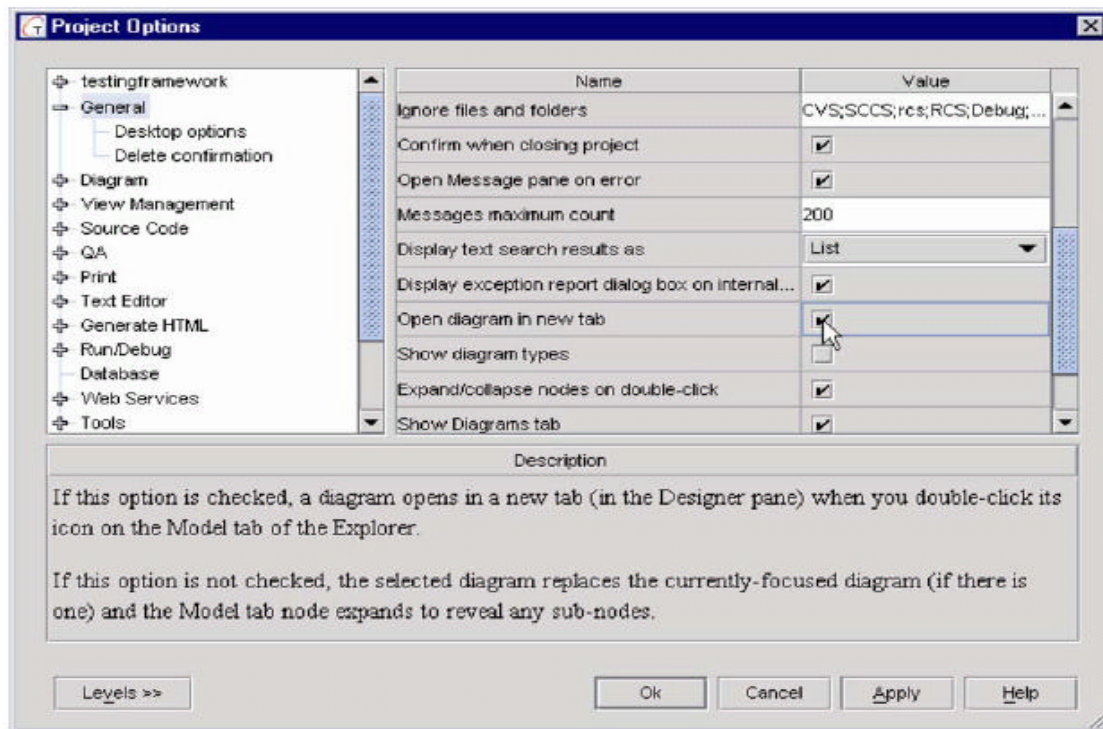
: Explorer    Model

Tools | Options | Project

가

General    (    )    “Open diagram in new tab”





가  
가 Pane

:	(Direct)	(Rectilinear)	.
---	----------	---------------	---

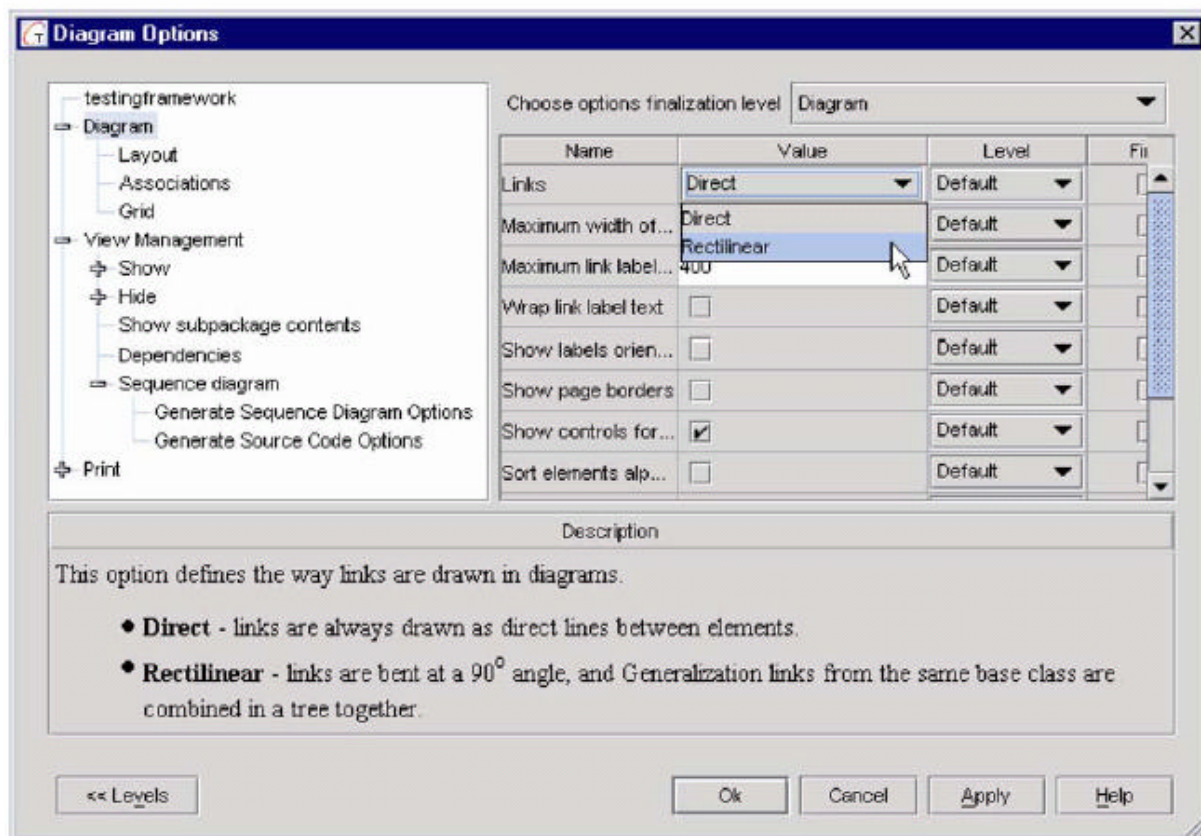
### Diagram Options

Tools | Options | Diagram

. Diagram

(

) Links



**Diagram | Grid**

**Levels**



The diagram illustrates the process of creating a workspace in UML. It shows a sequence of steps:
 

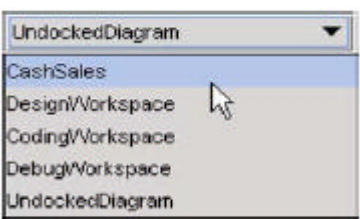
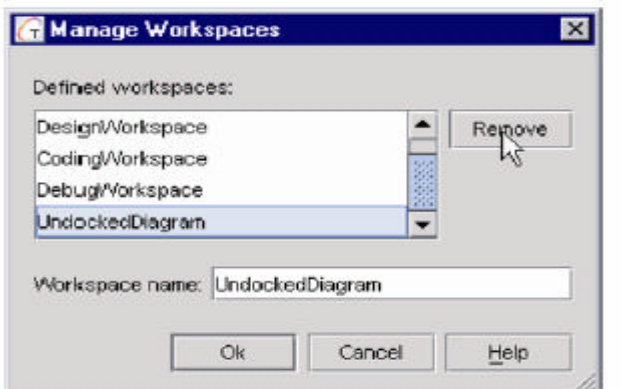
- 1. Pane (workspace)
- 2. View | Workspaces | Create
- 3. Workspace
- 4. Create
- 5. Work space 가

 A screenshot of the UML software interface shows the 'View' menu open, with 'Workspaces' selected. The 'Create Workspace' dialog box is displayed, showing 'Workspace name: UndockedDiagram' and buttons for 'Ok', 'Cancel', and 'Help'.

```

: (workspace) ( ) .
UndockedDiagram .

```

<p>Workspace</p>	
<p>UndockedDiagram      CashSales</p> <p>가    View  </p> <p>Workspaces   Manage Workspaces</p> <p>Remove</p>	

# Together

1 :

Together 가 .  
.  
Together Together .  
Together  
. Together LiveSource™ .

- 
- 
- <default>
- 
- 
- (Tip) (Trick)

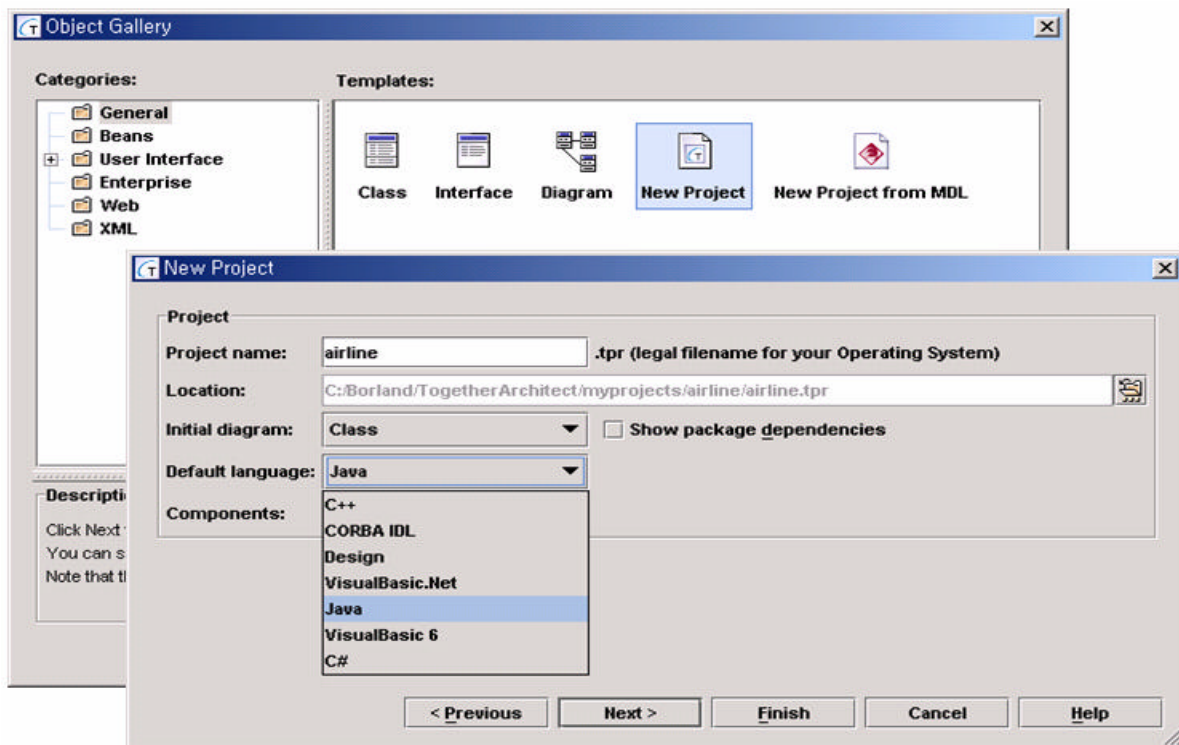
Together “ ”

Together .

:	airline	Java	.
---	---------	------	---

Together File | New . Object Gallery  
1. Categories pane General .(General .)  
2. Template pane New Project .

Together (airline) (Java) New Project 가 .  
Finish .  
Object Gallery가 New Project 가 .



,Together

\$TOGETHER\_HOME\$/myprojects


. Together

Java, VisualBasic, VisualBasic.Net, CORBA IDL,

C# 가

Design

가

- (tpr  )
- (default.dfPackage)
- 

airline 가 Together

airline

- default.dfPackage
- airline.tpr
- airline.tws( )

Windows

Windows

가

. (

Project Default .)

- default.dfPackage.wmg (Windows )

Explorer pane

Directory

User Projects

\$TOGETHER\_HOME\$/myproj

ects

. airline

User Projects

.

airline

Current Project

.

Together

가 Default, Project, Diagram level

.

Editor

.

가

.

:

가

airline

.

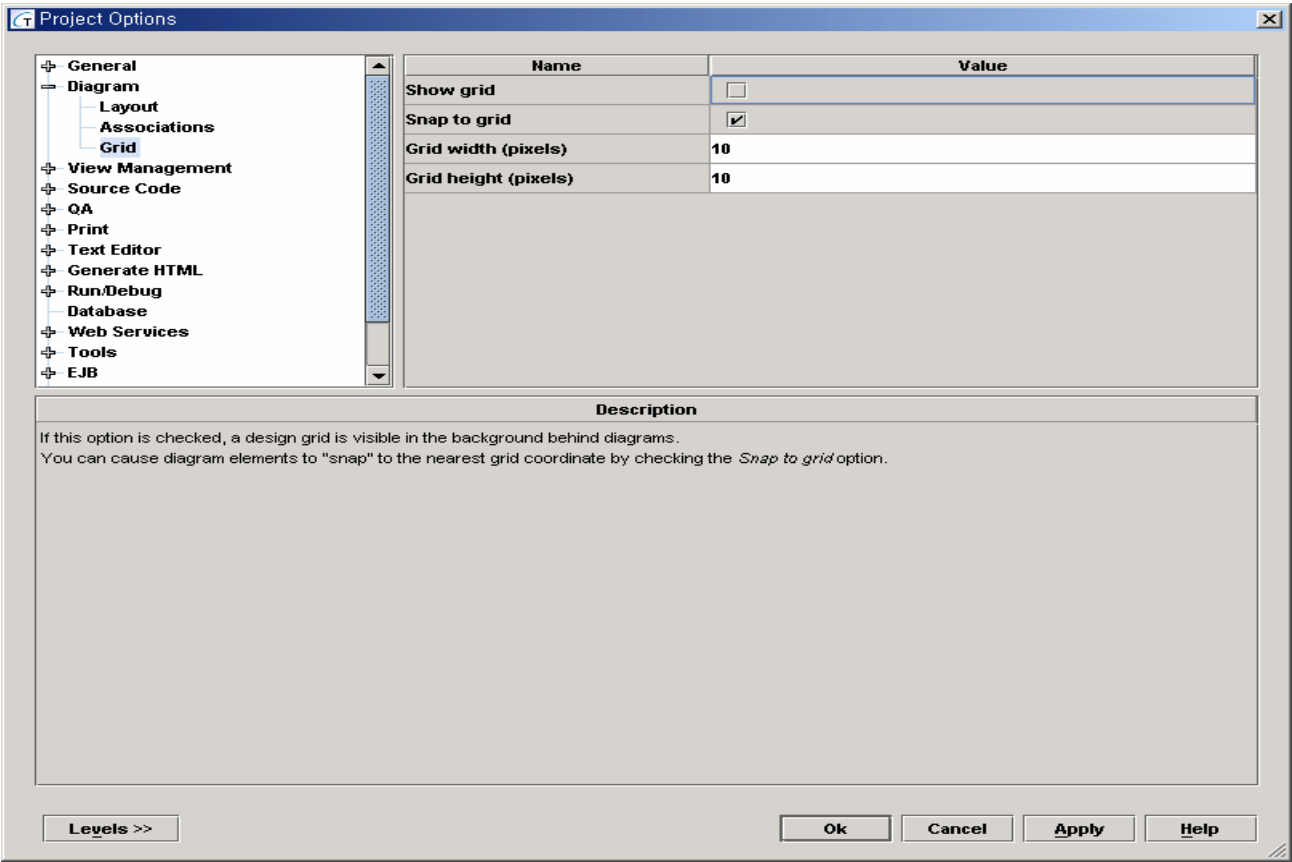
Tools | Options | Project Level

.

Together

가

.



Diagram

Grid

.


Show Grid








가 . Grid off . Ok

Cancel .

<default>

Together <default> . <default>



() . (.)

 airline +  diagrams +  src +  airline.tpr +  airline.tws	가 (airline) (diagrams) 가 (src)가 . .tpr Together . .tws Together (workspace) . ASCII .
 airline +  <default>	. 가 .

<default> . Airline

.

: <default>	ProblemDomain .
-------------	-----------------

Designer (  ) 가 . in-place . Enter .	
--	---

Togther LiveSource . Togther

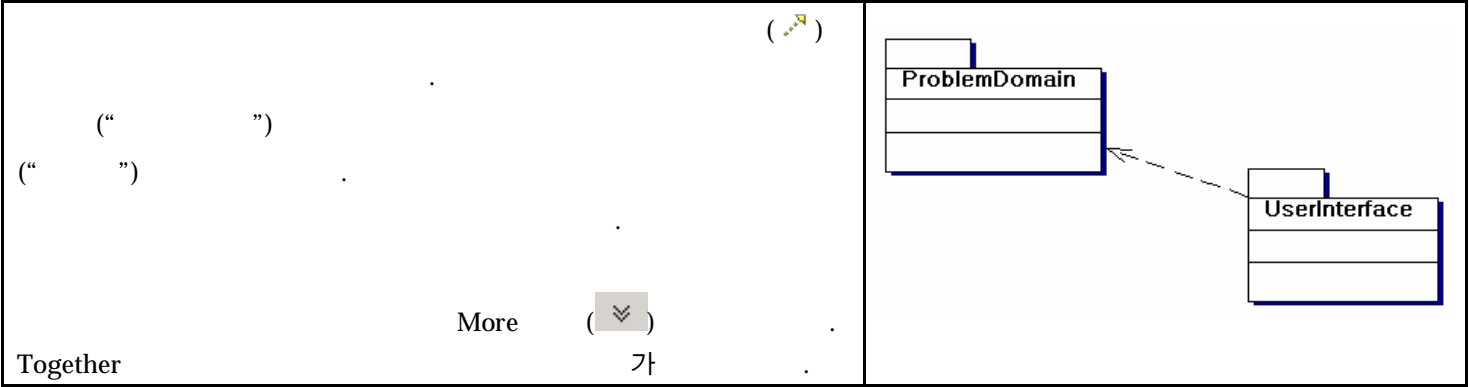
. Together





Designer pane      UML      가      .

: UserInterface      ProblemDomain      .



**(Tip)**      **(Trick)**

- .
- .tpr      Together      . Together      ASCII      .
- .
- Ctrl      가

Together

2 :

가 가

가

- 
- 
- 
- 

가 가

:

- 
- (actor), (use case), (system boundary)
- 
- 
- (Tip) (Trick)


airline . <default> .(Designer pane  
)

: **MakeReservation**

**MakeReservation** 가 .

가 가 .

1. **File | New**

2. Categories **General** Templates **New Diagram**
- Designer **New Diagram**(, **Ctrl + Alt + N**)

Together

가

Use Case

( <default>

Package

)

Together

가

☒ Include in current diagram

가

(Designer

pane

가 )

가

New Diagram

UML

Together

Class

Use Case

Sequence

Collaboration

Statechart

Activity

Component Diagram

Deployment

Diagram name:

예약

Package name:

<default>

☒ Include in current diagram

Description:

비행기 티켓 예약을 위한 시나리오입니다.

Click Ok to create a new diagram.

Ok

Cancel

Help

Ok

Explorer pane

Model, Directory, Diagram

Designer pane





Designer pane


:

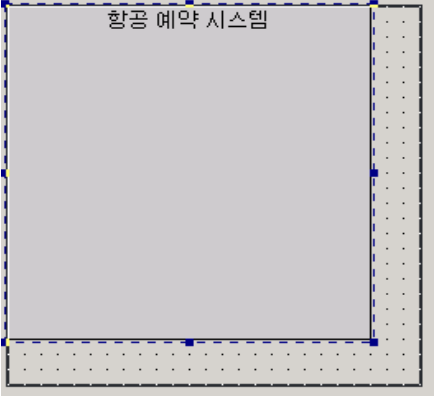
(system boundary)

Designer pane

	(Actor)
	(Use Case)
	(Communicates)
	(Extend)
	(Include)

	(Generalization)
	(System boundary)
	(Note)
	(Note Link)





: , ,



Designer pane



승객

in-place

Ctrl
Designer
가
Designer
. Ctrl

: , , 가



In-place



가 , 가


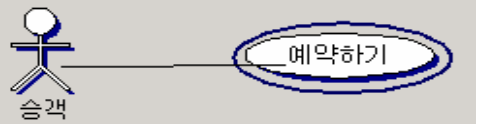
가 가 . 가 가  
가 , 가 가 .

:

- 1.
- 2.
- 3.
- 4.
- 5.

Designer

( / )

가 가 가	
가 가	

가  
, 가  
( )

:

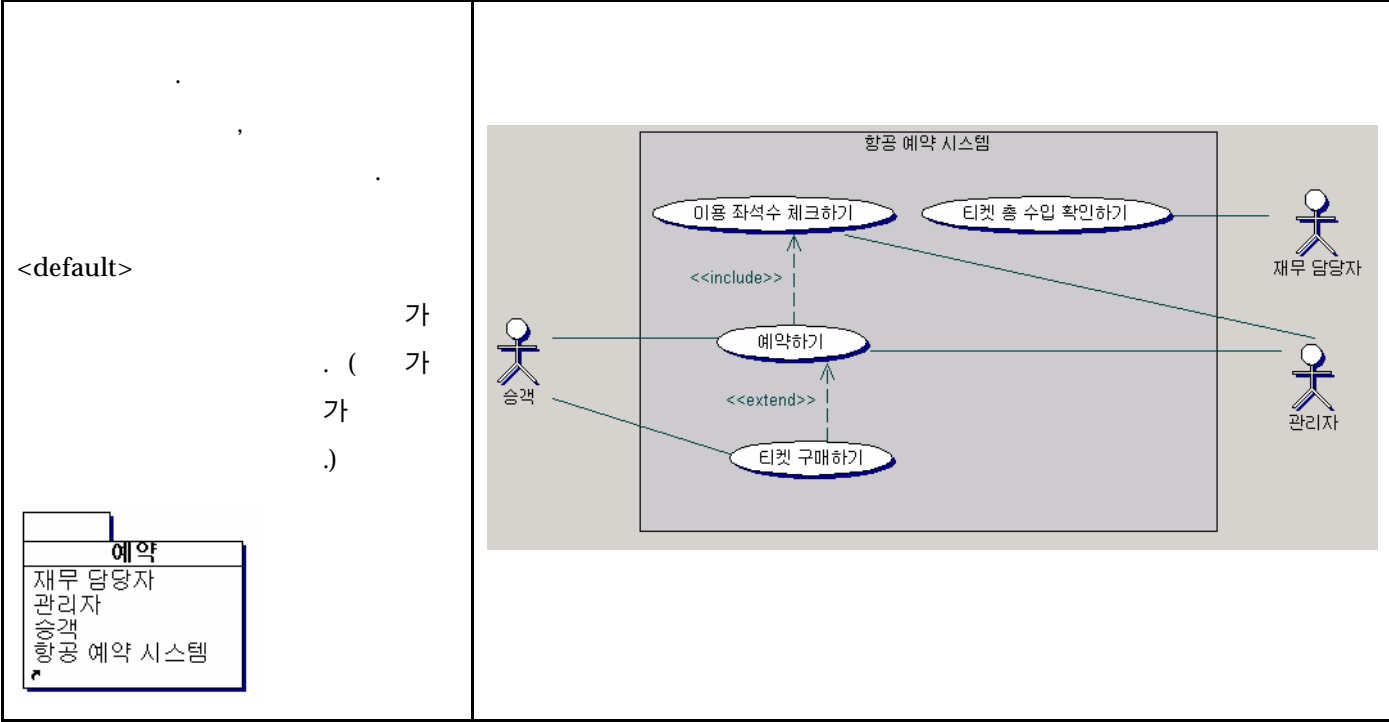
- 1.
- 2.

Designer

0

가 가

가



Together

: ( 가 ) .

Delete(Delete) Delete .

Delete(                      Delete)                      .                      가

**(Tip)**                      **(Trick)**

**(Tip)**                      **(Trick)**

- , 가 가 .  
 ■ in-place .  
 ■ Rename( F2) .  
 ■ Inspector Properties .  
 ● , 가 가  
 . / .  
 ● Undo ( Ctrl + Z) . Redo ( Ctrl + Y)  
 . (misc.config Undo 4096KB .  
 .)



Together

3 : (activity diagram)

가 .

- 
- 
- 가 가
- 

(Activity Diagram) “ ”가 .

:


- 
- 
- , ,
- (Fork), (Joins), (Decisions)
- (Tip) (Trick)



“ ” ? 가 가 .

: <default> .

Designer pane

New Diagram 

New Diagram

Together

☒ Include in current diagram

“include in current diagram”

<default>

가 가

New Diagram

UML Together

Class Use Case Sequence Collaboration Statechart

Activity Component Diagram Deployment

Diagram name: 예약 요청

Package name: <default>










☒ Include in current diagram

Description:

이 다이어그램은 이용 가능한 좌석수 체크와 티켓 발행을 포함하여 어떻게 예약하는지에 대해서 설명합니다.

Click Ok to create a new diagram.

Ok Cancel Help

	(Activity)		(Decision)
	(Start)		(End)
	(Horizontal Fork/Join)		(Vertical Fork/Join)
	(Transition)		(Swinlane)
	(Note)		(Note Link)


,

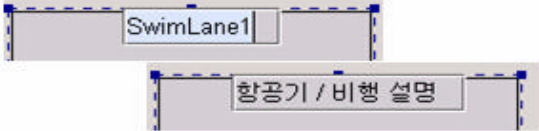
- / :
- :
- / :

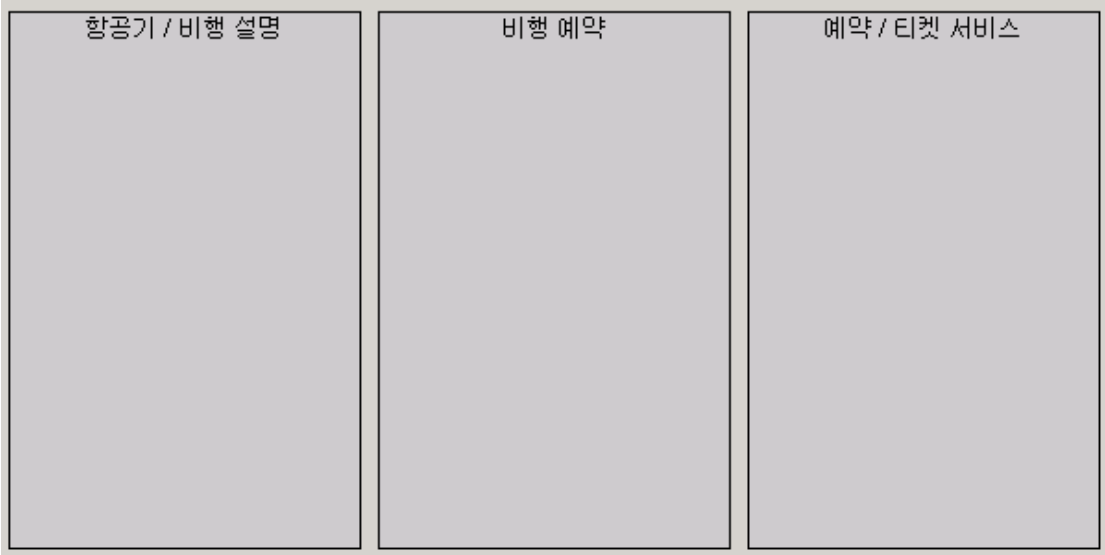
(class diagram)

- :
1. /
  2. 가

3. /

 Designer pane  
in-place





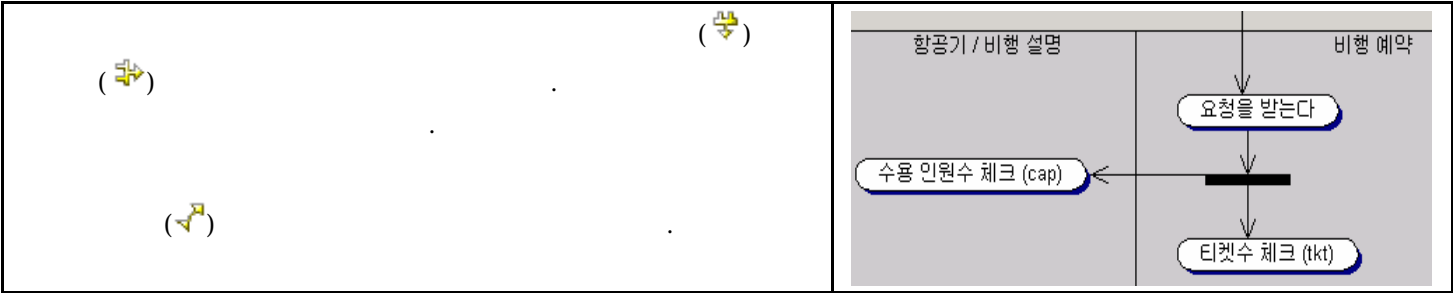
: ( ) .

Designer pane 가 .

가 .

: “ ” .

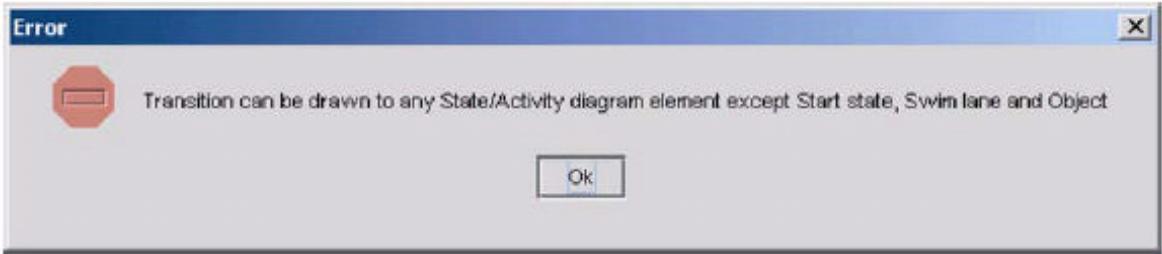




가

(entity)

Message pane



: (Joins)

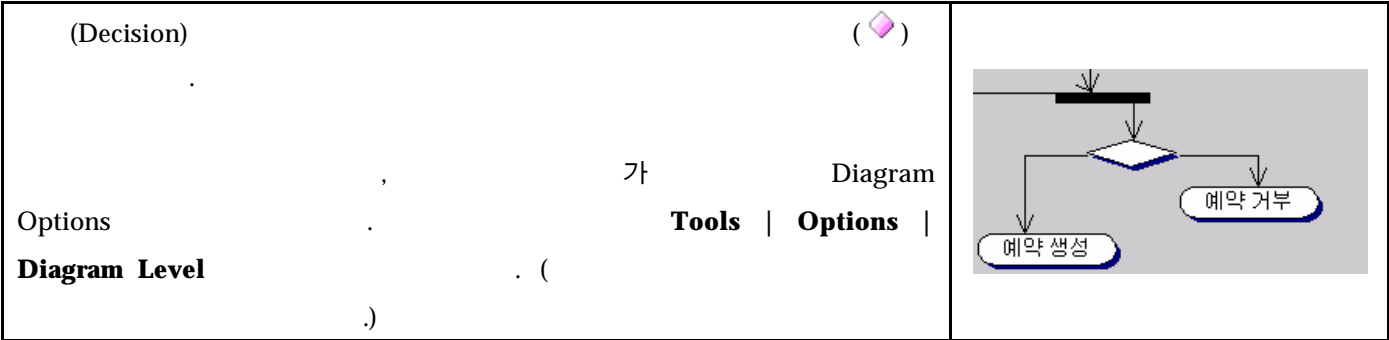
가

: (Decision)

가

가

가



가

: 가

- #tk<cap
- #tk>=cap

Property Inspector

Property Inspector가

Link . Requirements

가 . Description

가 .

Inspector

Link

Description

HTMLdoc

Requirements

Name	Value
client	Decision1
supplier	Create reservation
event name	
event arguments	
guard condition	#tk < cap
action expression	
send clause	
send time	

Press Ctrl+Enter to finish editing and close Insp...

:가 (decision) ,

1.

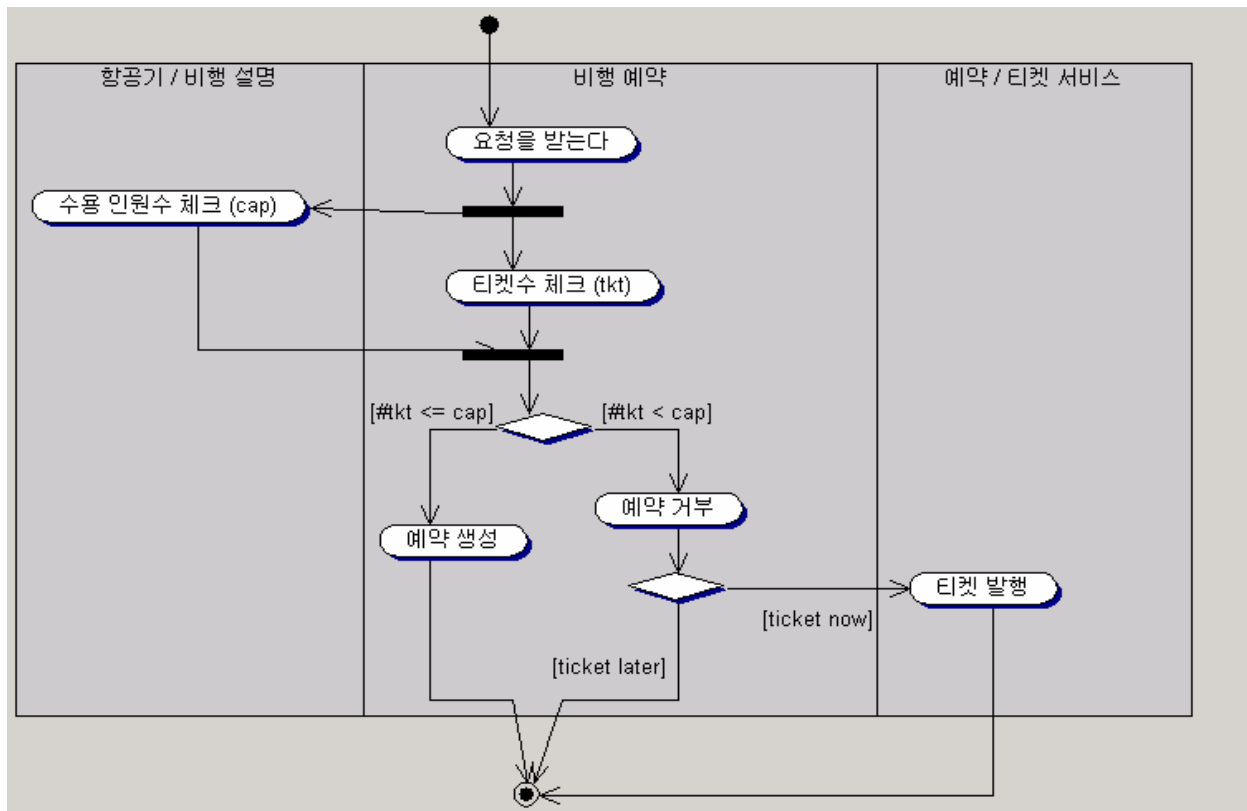
2. , ticket now 가

3. (End) , ticket later 가

4. (End)

5. (End)

가 . 가



**(Tip)**

**(Trick)**

•

• Together

• Together

“Choose Destination” 가

• Tools | Options | Diagram Options

•

가

•

( )

Together

4 :

(task) Together LivceSource™  
Together

:

- 
- LiceSource™
- Together
- (attribute) 가
- (operation) 가
- 
- Property
- (Tip) (Trick)

airline

- 
- 
- 가 가
- 

가

	FlightDescription	ScheduleFlight	Reservation	Ticket	Agent
	DepartureTime arrivalTime origin	date	name	basePrice	name




	destination capacity				
	getCapacity() setCapacity()	makeReservation() numberOfTickets()	ticketPurchased() calcPrice()	calcPrice()	makeReservation()

Designer pane	ProblemDomain	
	: ProblemDomain	FlightDescription, ScheduledFlight, Reservation, Ticket, Agent

Ctrl

Designer

(, Ctrl + L)

. Together

Designer pane

Editor pane

가

<default>

ProblemDomain

가

ProblemDomain

+

가

ProblemDomain

+FlightDescription

+ScheduledFlight

+Ticket

+Agent

+Reservation

Designer

<default>

예약 요청

예약

ProblemDomain

FlightDescription

ScheduledFlight

Ticket

Agent

Reservation

Editor

```

1  /* Generated by Together */
2
3  package ProblemDomain;
4
5  public class ScheduledFlight {
6      }
7

```

ScheduledFlight.java

Together . Explorer pane

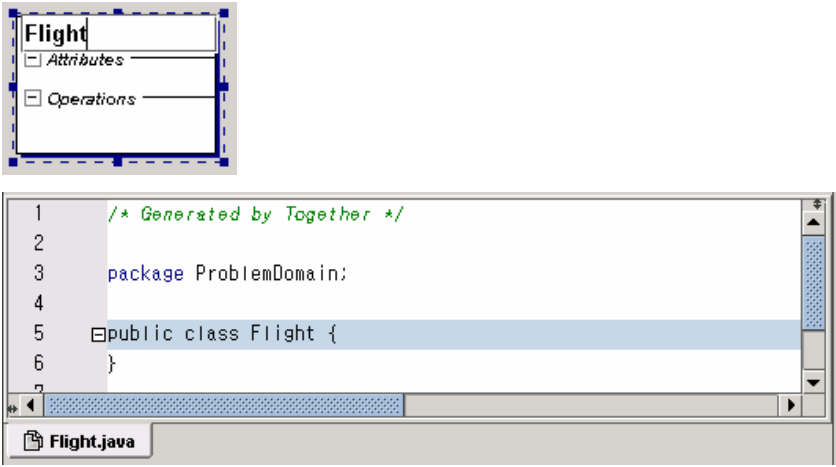
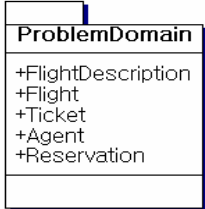
: (Inner class)가 , 가 . Designer pane

LiveSource™

Together , 가 .

Together LiveSource™

: Designer pane	ScheduledFlight	Flight
-----------------	-----------------	--------

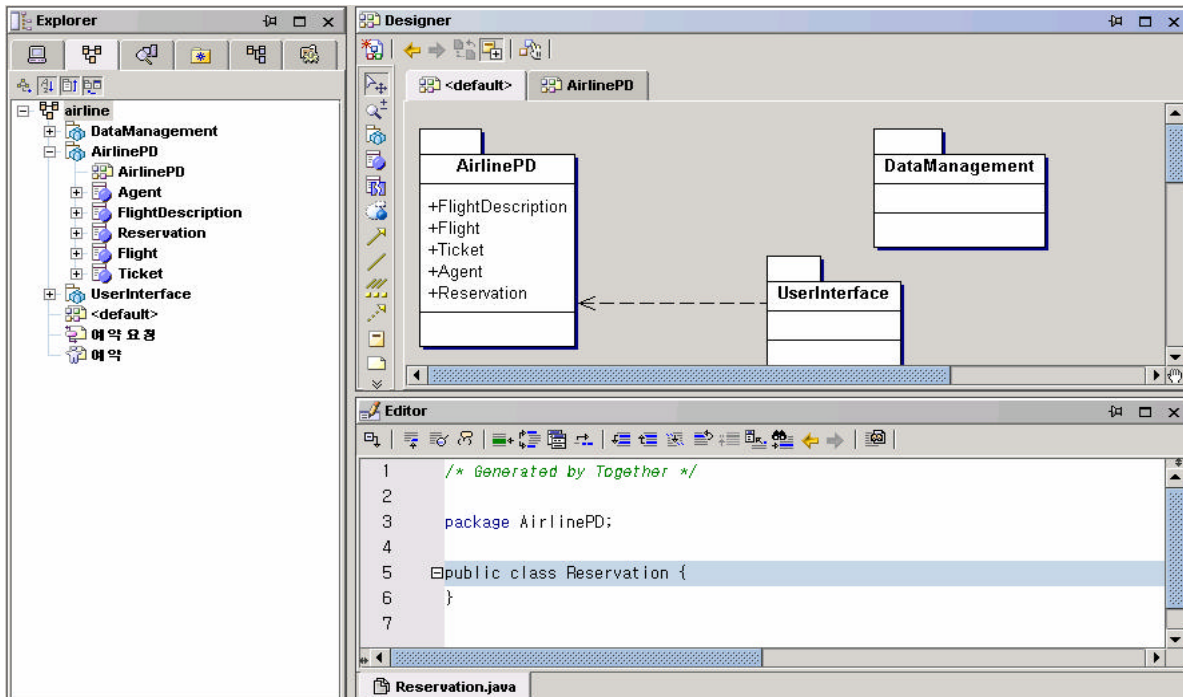
Designer pane	
<default> ProblemDomain	Together 

: <default>	ProblemDomain	AirlinePD
-------------	---------------	-----------

<default> ProblemDomain

Together 가 . (Explorer pane Directory

.)



## Together

Together

. Together

Editor

**Tools | External Editor**

Select in Diagram		
Browse Symbol		
Search for Usages...		
Cut	Ctrl+X	
Copy	Ctrl+C	
Paste	Ctrl+V	
Select All	Ctrl+A	
Collapse Lines		
Expand Lines		
Tools		
Toggle Breakpoint	F5	Format Source Ctrl+T
Enable Breakpoint		External editor
Text Editor Options...		EditPlus
Refactoring		C++Builder
Bookmarks		
Override/Implement Method		
		Make File Shift+F8
		Rebuild File Ctrl+Shift+F8

:	FlightDescription.java	.	Origin	destination	String	가
---	------------------------	---	--------	-------------	--------	---

WinVi Editor      Windows

WinVi Editor    Windows    . (Tools | Options | Default Level    Tools

.)

The diagram illustrates the integration of WinVisio and Borland Together. On the left, a box labeled 'WinVi' is connected to a box labeled 'FlightDescription.java'. An arrow points from 'FlightDescription.java' to a box labeled 'Together'. The 'Together' box is divided into two sections: 'Editor' and 'Designer'. Below the 'Together' box, a screenshot of the 'FlightDescription.java - WinVi' window is shown. The window displays the following Java code:

```

/* Generated by Together */

package AirlinePD;

public class FlightDescription {
    private String origin;
    private String destination;
}

~
~
~
~
~
~

```

The screenshot also shows the 'Together' menu bar (File, Edit, Search, Options, Windows, Help) and a toolbar with various icons. The status bar at the bottom indicates the file path as 'C:\Borland\TogetherArchite...' and shows the current line and column as '00001' and '001' respectively.

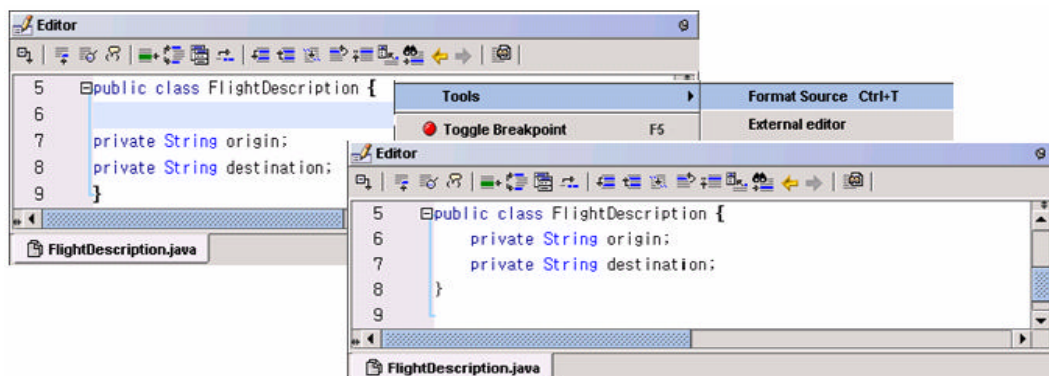
<pre> : origin(String      ) destination(String      ) </pre>	Java
---	------

Format Source Editor pane

Editor pane

가 Editor

## FlightDescription.java가



**(attribute)**     가

가

(class node)

```
: Designer pane      가      Reservation      name(String      )      가      .
```

가

**New | Attribute(                      Ctrl + A)**

Figure 1 illustrates the UML class diagram and the corresponding Java code for the Reservation class. The UML class diagram shows a class named Reservation with a private attribute named attribute1 of type int. The Java code shows the package declaration package AirlinePD; and the class definition public class Reservation { private String name; }.

Designer pane

가

가 가

●

가

Insert

●

## Clone



Copy

## Paste

●

Cut

Paste

- Ctrl +

: Reservation name

1. (initial value) null
2. “Requirement description” Last name first
3. Author

Javadoc

Inspector

property

(

.)

Inspector Properties

Requirements

author

description

Ctrl + Enter

Inspector가

Editor

```

3 package AirlinePD;
4
5 public class Reservation {
6     /**
7      * @author Kailynn
8      * @description Last name first
9      */
10    private String name = "";
11 }
12

```

Reservation.java

- :

AirlinePD

가

.

Editor가

Designer pane
1. Agent – name (Reservation )
  2. FlightDescription – departureTime, arrivalTime ( capacity .)
  3. Flight – date
  4. Ticket -- basePrice

Together

Layout | Actual Size

가

가

FlightDescription

-origin:String

-destination:String

-departureTime:Date

(operation) 가

가 가 Ctrl + O

: **makeReservation** void Flight 가 .

.

1. **name** : **String** .

2. **tKind** : **int** .

가 UML ( ) Java/C++ ( )

가 .

UML	( ):	myMethod(myParameter:int):double
Java/C++	( )	double myMethod(int myParameter)

(attribute) 가 private . 가 public .

Inspector, 가 .

: Designer pane Flight **Agent makeReservation()** .

(operation) Ctrl + (operation)

Copy , Paste . Designer

pane Copy .

: **Reservation** 가 .

- **String** **name**, **int** **tKind** 가
- **Boolean** **ticketPurchased()**
- **double** **calcPrice()**

**Reservation**

New | **Constructor** 가 .

New | **Property** New | **Member by Pattern**

가 .

Attribute Ctrl+A

Operation Ctrl+O

**Constructor**

Property Ctrl+B

Member by Pattern...

Inner Class

---

Together .

Designer pane

View Management (  )

Reservation

Designer detail level

“Implementation”

Reservation

-name:String=""

+Reservation(name:String,tKind:int)

+ticketPurchased():Boolean

+calcPrice():double

View Management

Diagram Options: View Management

View Management

Show

Hide

Show subpackage contents

Dependencies

Sequence diagram

Name	Value
Diagram detail level	Implementation
Member format	Default
Referenced classes	Analysis
	Design
	Implementation

Implementation

Description

This option defines how much information will be displayed for an element in diagrams.  
The detail level that is used by default depends on the role you selected during installation.

• Default - the default detail level for the current role.

☐ Business Modeler role - Analysis level

☐ Designer and Developer roles - Design level

☐ Programmer role - Implementation level

• Analysis - names only (no visibility signs).

• Design - names and types (visibility signs are shown).

• Implementation - names and types, parameters for operations, and initial values of attributes (visibility signs are shown).

Levels >>

Ok

Cancel

Apply

Help

- : Flight Ticket 가 .

● Flight – numberOfTickets() int .

● Ticket – Reservation calcPrice() .

Together getter setter property .



FlightDescription

Property

Property

Together

getter

setter

Together

getter

setter

JavaBean

getter

setter

property

property

New

```
5 public class FlightDescription {
6     private String origin;
7     private String destination;
8     private Date departureTime;
9     private Date arrivalTime;
10    private int capacity;
11
12    public int getCapacity(){
13        return capacity;
14    }
15
16    public void setCapacity(int capacity)
17        this.capacity= capacity;
18 }
```

FlightDescription

-origin:String  
-destination:String  
-departureTime:Date  
-arrivalTime:Date

capacity:int

Property

JavaBean

가

Default Options

General

Diagram

View Management

Show

Hide

Show subpackage contents

JavaBeans / C++ Properties

Dependencies

Sequence diagram

Banned Destinations

Source Code

QA

Print

Text Editor

Generate HTML

Run/Debug

Database

Web Services

Name	Value
Recognize JavaBeans	<input type="checkbox"/>
Recognize C++ Properties	<input checked="" type="checkbox"/>
Recognize VBasic Properties	<input checked="" type="checkbox"/>
Show attributes and accessors	<input type="checkbox"/>
Recognize classes without public/defa...	<input type="checkbox"/>
Create getter as const	<input type="checkbox"/>

Description

If this option is checked, classes in the class diagram are recognized as JavaBeans, and the *Bean* tab is added to the Inspector for the classes. Note that this option affects performance. Uncheck it for improved performance.

Levels >>

Ok Cancel Apply Help

Recognize JavaBeans

Property

getter

setter

FlightDescription

-origin:String  
-destination:String  
-departureTime:Date  
-arrivalTime:Date  
-capacity:int

+getCapacity():int  
+setCapacity(capacity:int):void

Property 가 Recognize JavaBeans가

: capacity	50	.
------------	----	---

Properties (initial value)가

property

: Ticket	.
Ticket	.
Ticket	calcPrice() (method)

 Abstract Property Inspector . Abstract

Ticket 가 (UML )

Inspector . Inspector . Together  
( 가 (class node)가 )  
Javadoc (Javadoc Description )

: <b>Flight</b>	“moment-interval”	(stereo type)	.
-----------------	-------------------	---------------	---

Inspector Properties 가 . Moment-interval **Peter**  
**Coad** **Java Modeling in Color with UML**

Together “moment-interval” . Flight “moment-interval” “moment-interval-detail” . “role” . “party”, “place”, “thing” . “description” .	<div><div>&lt;&lt;moment-interval&gt;&gt; <b>Flight</b></div><div>-date:int</div><div>+makeReservation(name:String,tKind:int +numberOfTickets():int +calcPrice():double</div></div>
--	---

Together	
View	Inspector 가

- :
가
.
- **FlightDescription** : description
  - **Reservation** : mi-detail (moment-interval detail)
  - **Ticket** : thing
  - **Agent** :role

가 .

(Tip)
(Trick)

- 가 (operation) 가 Ctrl + O, (property) 가 Ctrl + B . (attribute) 가 Ctrl + A ,
- ( , , )
- Ctrl+ .
- (attribute) (operation)
- Together
- . Together 가 , 가 가 .
- Property Inspector
- Together 가 . property New | Property( Ctrl + B) , 가 getter setter 가 , .

Together

5 :

가

- :
  - (class association)
  - (association)
  - 
  - 
  - 
  - 
  - 가
  - (Tip) (Trick)

(class association)

AirlinePD . Designer pane

: Flight FlightDescription

Designer pane (association) ( )

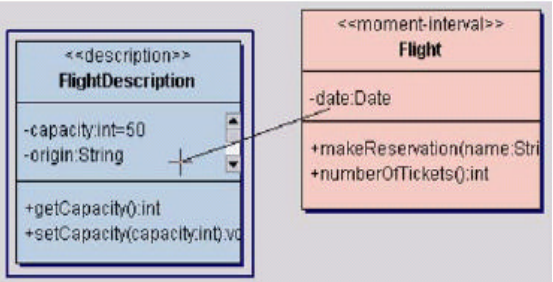
( ) 가 가 Together

Flight FlightDescription

Designer pane

Editor Flight

InkFlightDescription 가 가



. ( **lnk** .)

1.	<b>Reservation</b>	<b>Ticket</b>
2.	<b>Agent</b>	<b>Flight</b>

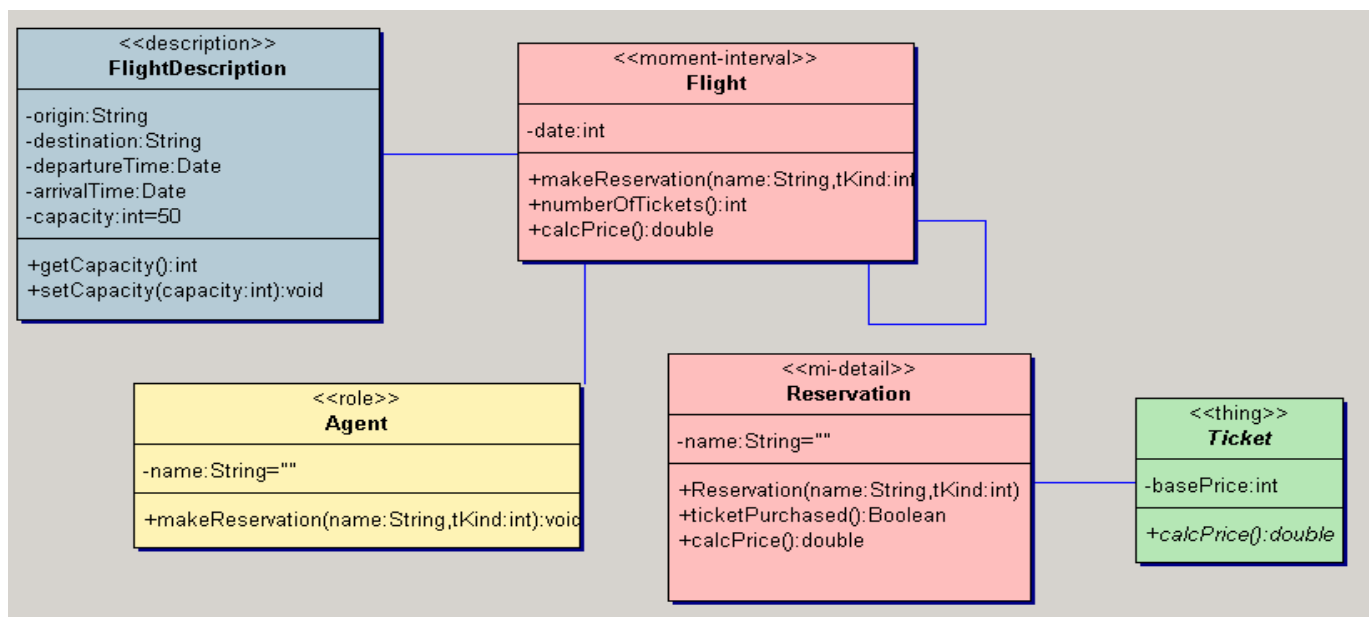
Together

- (Rectilinear) : (segment) (sequence) .
- (Direct) : . (가 .)

**Tools | Options | Diagram Level**

가 (self link)

:	Direct	.	Flight	.
---	--------	---	--------	---




(association)

Inspect가 .

	: Flight	FlightDescription	가	.
●	0... *	:	(Flight)	.
●	1	:	(FlightDescription)	.

가 가  
가  
Inspector



:	Reservation	Ticket	.
Agent	Flight	가	.
		가	.

Property Inspector  
가 .

directed	Automatic
type	Automatic
	Directed
	Undirected

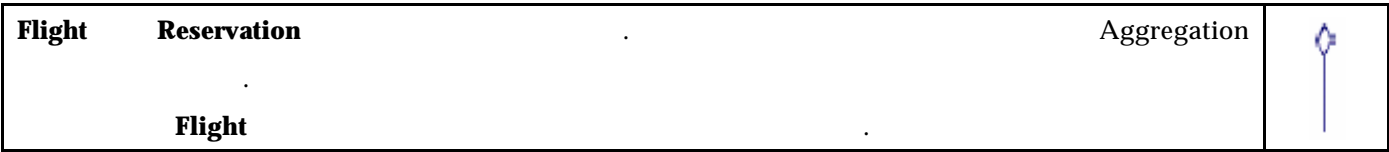
	: Flight	.
●	(supplier role)	“actual” . 0...1 .
●	(client role)	“plan” . 1 .

Flight      Inspect .

Flight

Reservation

Reservation



가

Layout | All (

Ctrl + K)

Together

(+)

가 가

4



(Generalization)

(abstract class) Ticket

. Agent Reservation

(interface)

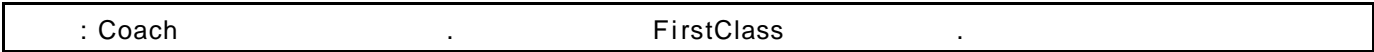


Coach가 Ticket  
(operation)

(Generalization)



Together



Copy( Ctrl + C)

Paste( Ctrl + V)

Javadoc

:

(Rectilinear)

Tools

| Options | Diagram Level

Diagram

Diagram | Layout

Agents      Reservations      가

:

INamed

INamed    String      getName()

Agent   Reservation   INamed


Designer pane

(  )

(operation)

Together

Designer    pane

(Generalization)    (  )

Together

implements

3      package AirlinePD;

4

5      /\*\*

10    public class Reservation implements INamed {

11      public Reservation(String name, int tKind) {

12      }

13

14      public boolean ticketPurchased() {

15      }

Reservation.java

getName()

(operation)

Agent    Reservation



가

“ 가 ”

: AirlinePD

Designer pane

Note



Note



Flight

Inspector

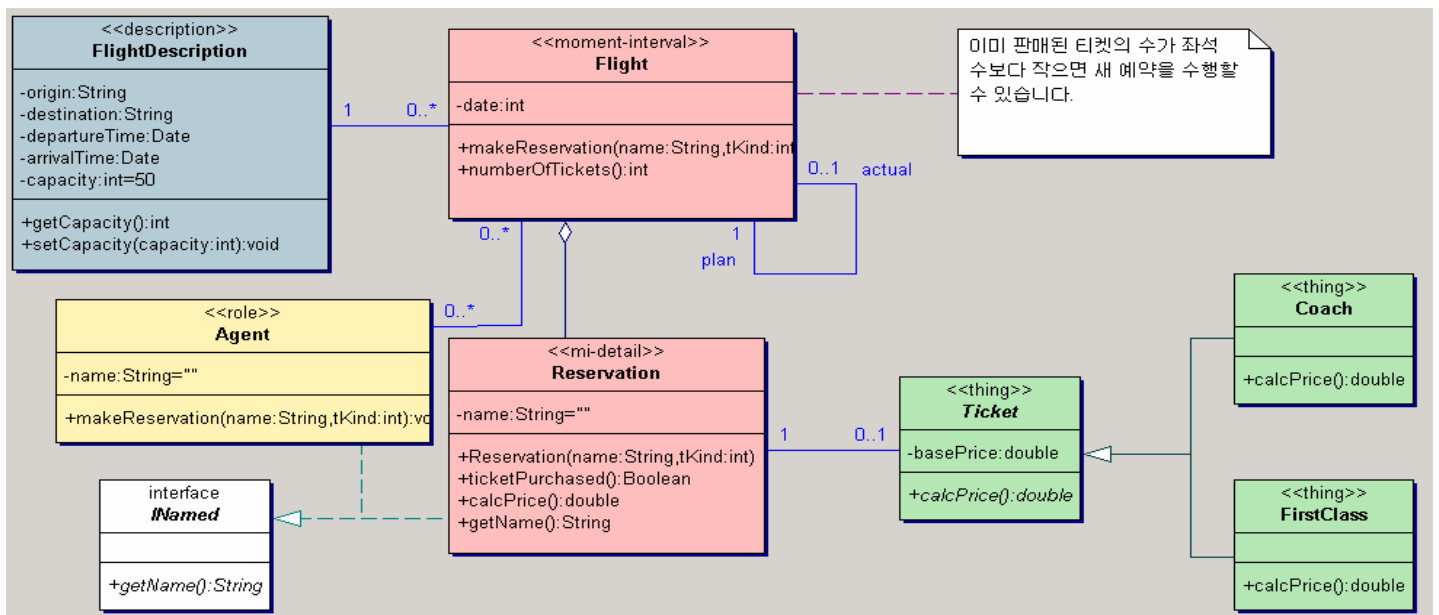
Requirements

Inspector: (note)

HTMLdoc		Requirements	
Text		Properties	
		View	
Name		Value	
type		Business Rule	
number		Business Rule	
description		Feature	
document		Performance	
author		Product Requirement	
priority		User Need	
difficulty		Medium	
testcase		Medium	

Press Ctrl+Alt+I to finish editing and close In...

AirlinePD



---

(Tip) (Trick)

- Designer pane 가 . Layout | All ( Ctrl + K) Together
- , , Undo( Ctrl + Z) Undo가 가 . Undo Redo( Ctrl + Y) .
- Undo Redo / .
- (Rectilinear)
- (Direct) .
- (operation) (method)
- .
- Ink . Tools | Options | Default Level .
- Diagram | Associations | Show as attributes | All .
- “ ”(Directed) Tools | Options | Diagram Level .
- Diagram | Association | Draw Directed | All .

Together

6 :

. Together (template pattern)

가

GoF Coad

- :
- 
- 
- 
- 
- (Tip) (Trick)

Together Stub Implementation, Property, Main (method) 가 Java  
. Stub Implementations void가 (operation) return . Property getter  
setter . Main main() .

:	(method)	Stub Implementation	.
<b>Coach</b>	calcPrice()		
<b>Reservation</b>	ticketPurchased()		
<b>Flight</b>	numberOfTickets()		
	FirstClass calcPrice()	calcPrice() Coach FirstClass	.
Reservation	calcPrice()		.

Stub

Pattern(

Together

Pattern

Ctrl + R)

(member)

.

(link pattern)

Choose

Select in Model Tree

New Operation

Properties...

Rename

Search for Usages...

Choose Pattern...

Cut

Copy

Clone

Paste

Delete

Insert

ALT+Enter

F2

Ctrl+R

Ctrl+X

Patterns

member

Cactus

JUnit

JUnitX

Property

Stub Implementation

Main method

link

Editor

가

.

8

9

10

11

12

public class Coach extends Ticket {

public double calcPrice(){

// Write your code here

return 0;

}

Coach.java

: Property	Reservation	Agent	name	.	get	set
------------	-------------	-------	------	---	-----	-----

Properties

.

Properties

Pattern

, , property getter, setter

Choose Pattern - Property - Pattern properties

Patterns

member

Cactus

JUnit

JUnitX

Property

Stub Implementation

Main method

link

Parameters

Name	Value
Name	name
Type	java.lang.String
Get method	<input checked="" type="checkbox"/>
Set method	<input type="checkbox"/>
Attribute	<input checked="" type="checkbox"/>

Description

Property pattern

This pattern makes a property skeleton in the specified class.

In the Name and Type fields you can specify the name and the type for a newly created property.

The Get method, Set method and Attribute checkboxes control whether to generate "get" or "set" accessors, or a private attribute to keep the values of a property.

For Java:

Note that although this pattern creates a real bean property in accordance with the bean specification, the class itself might not be shown as a property if the Red attributes and accessors options are not checked (Tools | Options | >level< - View Management - JavaBeans/C++ Properties).

< Previous

Next >

Finish

<<mi-detail>>

Reservation

-name:String=""

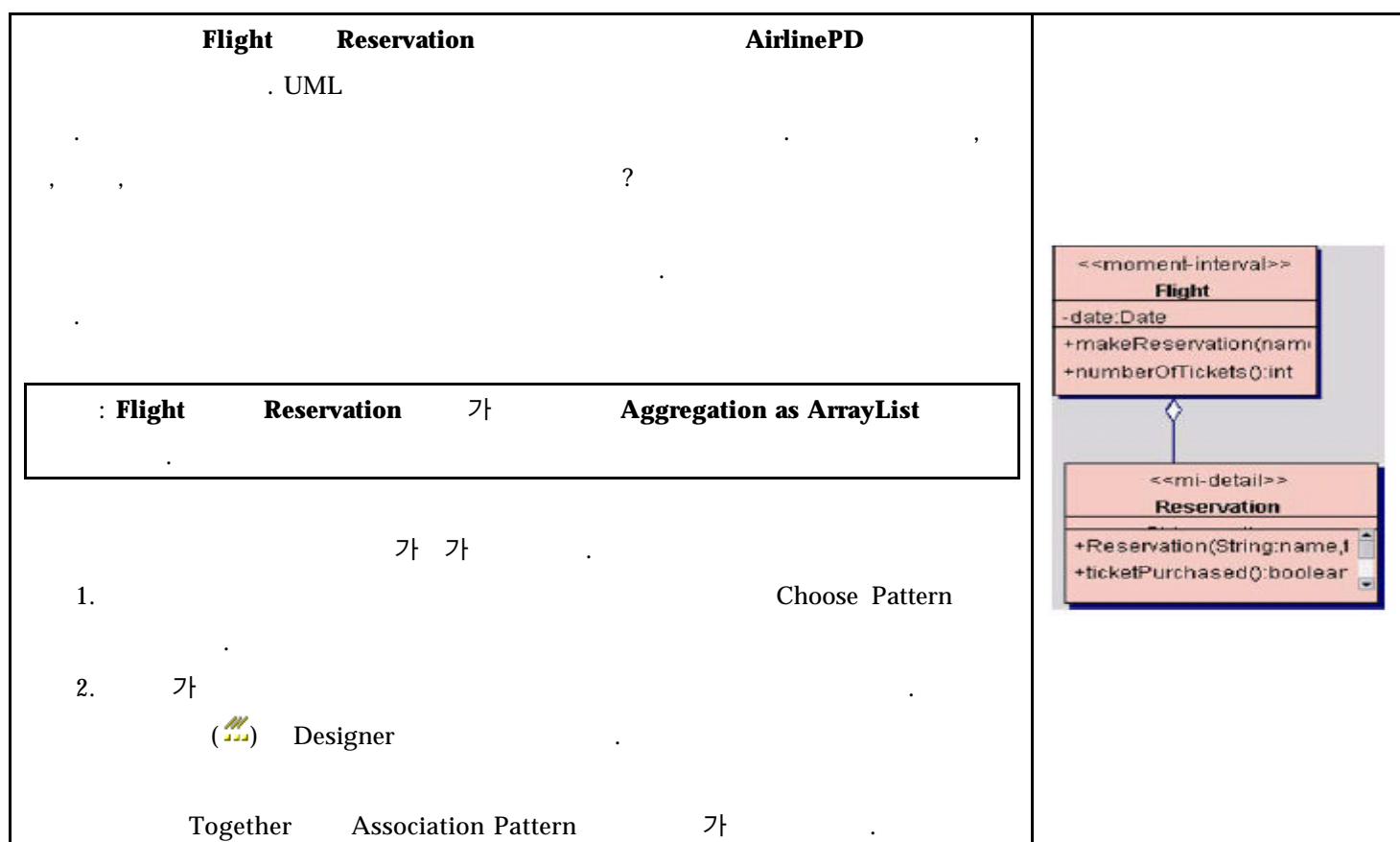
+Reservation(name:String,tKind:int)

+ticketPurchased():Boolean

+calcPrice():double

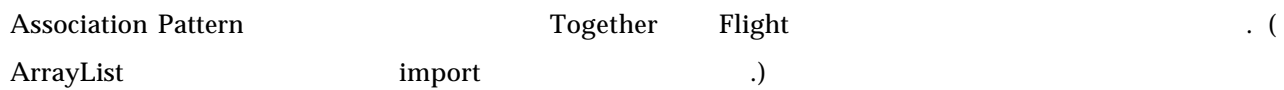
+getName():String

Name	Together	( )	. Attribute
Together	.		
	property		Member by Pattern



```

Pattern selection pane
Association as ArrayList
. java.util.collections
.
```




: <code>UIInterface</code>	<code>Flight</code>	가 (shortcut)	.
----------------------------	---------------------	--------------	---


Explorer pane Model 가 .

AirlinePD Flight Add as

Shortcut .

UserInterface 가 ( )가 가





: Main ClassDriver.

Class by Pattern ( ) . Designer pane Class by Pattern

Main ClassParameters pane

Name

Class1Driver

Preview paneTogether가

Patterns

Java Beans

JUnit

JUnitX

Robustness

TagLibs

UI Components

Application

Bean

Servlet

Stub Implementation

Main Class

Reference HttpServlet

Std Exception

Parameters

NameValue

NameDriver

Preview

public class Driver {  
public static void main(String[] argv) {  
}  
}

Main Class

BeanServlet

: DriverAirlinePD.Flight가Driver

myFlight.

가

가

Designer가

Editor paneTogetherimport

1. Together가

2. Flight가

Layout

Hide

Show Hidden Members...

Visibility

Abstract

Final

Class

Attributes

Operations

Inner Classes

All

Designer

AirlinePD

UserInterface

Driver

myFlight Flight

main(argv, String[] void

Editor

package UserInterface;

import AirlinePD.Flight;

public class Driver {

private Flight myFlight;

public static void main(String[] argv) {

}

}

Driver.java

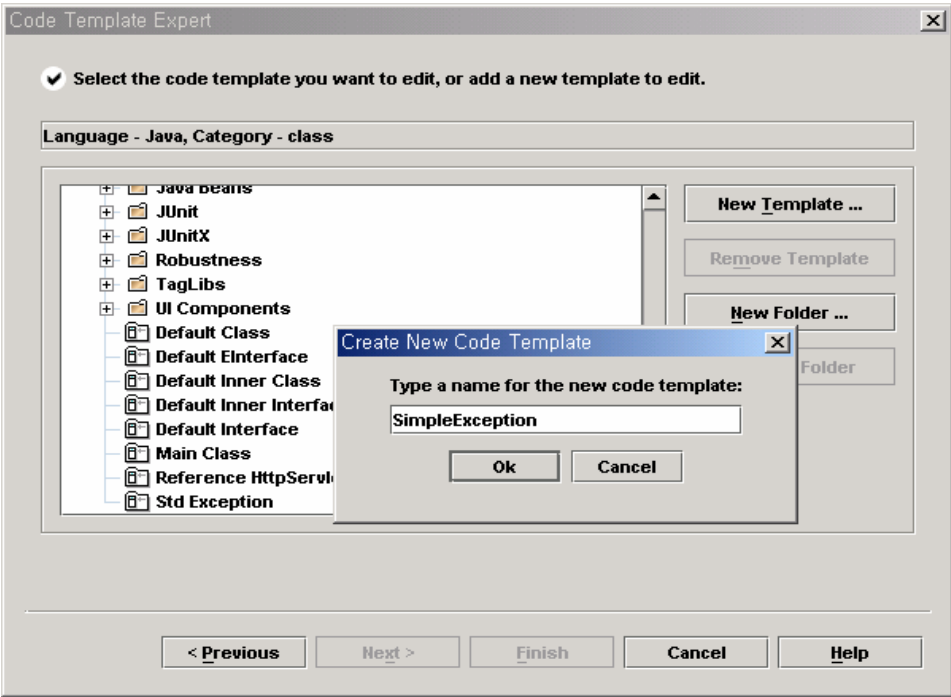
Java Together templates/JAVA/CLASS .

- 1. %Name%.java :
- 2. FolderName.properties : property .

Together Code Template Expert가  
Expert

: Template Expert SimpleException .

- 1. Tools | Code Template Expert .
- 2. Code Template Expert Pane 1  
Template Language : Java, Template Category : Class
- 3. Code Template Expert Pane2 New Template .
- 4. SimpleException Ok .





5. SimpleException . Next Pane3

: Exception1

- Pane3 ( ):
1. Pane 3 : Default name Exception1 .( Exception1 .)
  2. Pane3 : Exception Template description . ( HTML) .

Code Template Expert

Code Template Expert

☒ Set code template properties and edit the template code and its description.

Language - Java, Category - class, Template name - SimpleException

Default name:

Exception1

Generate prologue and epilogue:

☒

Hide in choose list:

☐

Paste all created classes to one file:

☐

Template description:

<HTML> <H3>SimpleException pattern</H3> </HTML>

Edit template text:

Edit Template Code...

< Previous

Next >

Finish

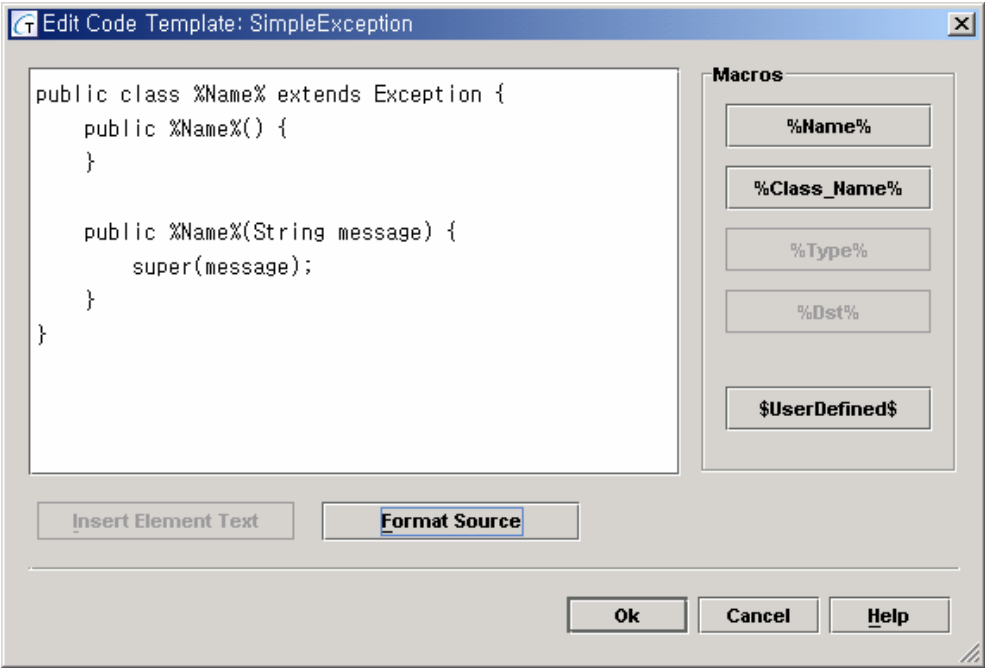
Cancel

Help

:

1. Pane3 Edit Template Code
2. %Name% %Name%
3. Format Source
4. 가 Ok

5. Pane Finish



: AirlinePD                      ReservationException    Exception

AirlinePD  
Class by Pattern                      SimpleException

property

ReservationException

Preview

ReservationException

Description pane

HTML

Name

가

Together

가

Choose Pattern - SimpleException

Patterns

- Oracle9i
- Sun Java Center J2EE Patterns
  - WebServices
  - Coad Classes
  - GoF
  - Cactus
  - HttpUnit
  - Java Beans
  - JUnit
  - JUnitX
  - Robustness
  - TagLibs
- UI Components
  - Application
  - Bean
  - Servlet
  - Stub Implementation
  - Main Class
  - Reference HttpServlet
  - SimpleException
  - Std Exception

Parameters

Name	Value
Name	ReservationException

Preview

^ Generated by Together ^/

```
public class ReservationException extends Exception {  
    public ReservationException() {  
    }  
  
    public ReservationException(String message) {  
        super(message);  
    }  
}
```

Description

SimpleException pattern

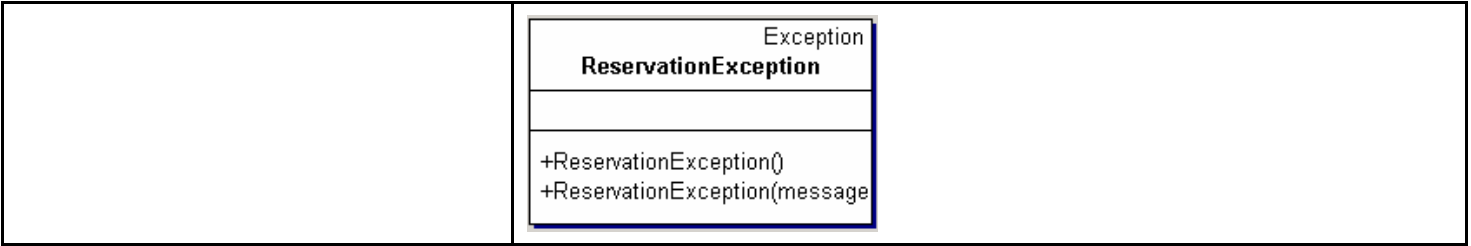
< Previous

Next >

Finish

Cancel

Help



Flight.makeReservation()      ReservationException

**(Tip)**      **(Trick)**

- (      )  
   . Together  
   . (Designer pane      Designer pane      ,      ,  
   .)
- 가 (Shortcut)      .      가      Delete  
   가      Delete      Together      가
- . Code Template Expert      .      Expert      Pane      Class      Link
- (Std Exception      )      .      .

Together

7 :

Together

GoF

Abstract Factory

Together

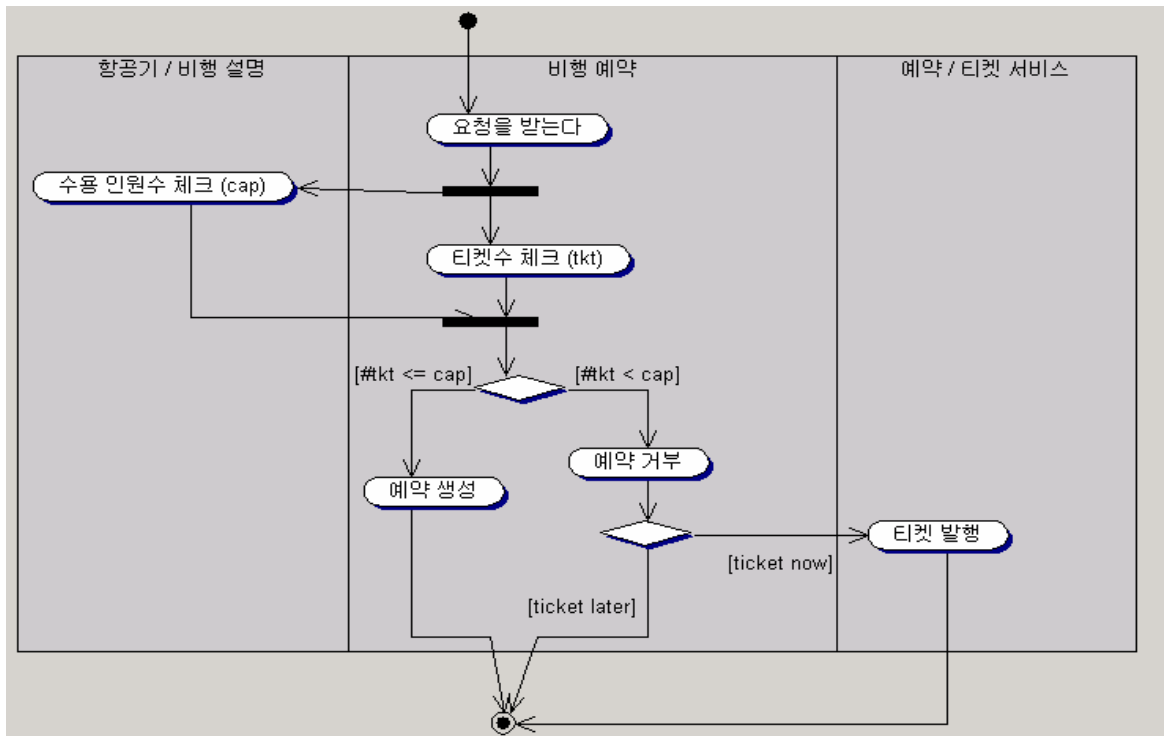
:

- 
- GoF
- 
- (Tip) (Trick)

GoF “Gang of Four” Design Patterns, Elements of Reusable Object-Oriented Software Eric Gamma, Richard Helm, Ralph Johnson, John Vlissides 가

Flight makeReservation()

가



Flight.makeReservation()

```

public void makeReservation (String name, int tKind) throws ReservationException {
    int t = this.numberOfTickets();
    int c = InkFlightDescription.getCapacity();
    //          가
    if (t < c) {
        Reservation r = new Reservation(name, tKind);
        InkReservation.add(r);
    }
    else throw new ReservationException();
}

```

: Flight      makeReservation()

Together  
  
(code sense)  
  
**InkFlightDescription**  
  
Editor      **FlightDescription**

```
public void makeReservation(String name, int tKind) throws ReservationExcep
    int t = this.numberOfTickets();
    int c = InkFlightDescription.
```

clone()      Object

equals(Object)      boolean

finalize()      void

**getCapacity()**      **int**

getClass()      Class

hashCode()      int

notify()      void

notifyAll()      void

setCapacity(int)      void

toString()      String

wait()      void

Reservation	Stub	name	tKind	name
tKind	가			
tKind가 1	Coach	tKind가 2	FirstClass	

: **Reservation**

**Reservation**

```
public Reservation (String name, int tKind) {
    this.name = name;
    //
    InkTicket = null;
    if (tKind == 1)
        InkTicket = new Coach();
    else if (tKind ==2)
        InkTicket = new FirstClass();
}
```

# GoF

airline 가

Reservation

Factory      Ticket      **TicketMaster**      GoF      abstract

: GoF      Abstract factory      **Abstract Factory, Concrete Factory, Abstract**

**Product**

- Abstract Factory      ICreateTickets
- Abstract Product      ticket
- Concrete Factory      TicketMaster

Designer pane      Class by Pattern      (🌐)      Designer pane      .

Pane      가      . 11      GoF      .      GoF

Abstract Factory      .

Pane

**ICreateTickets      TicketMaster**

**Ticket**

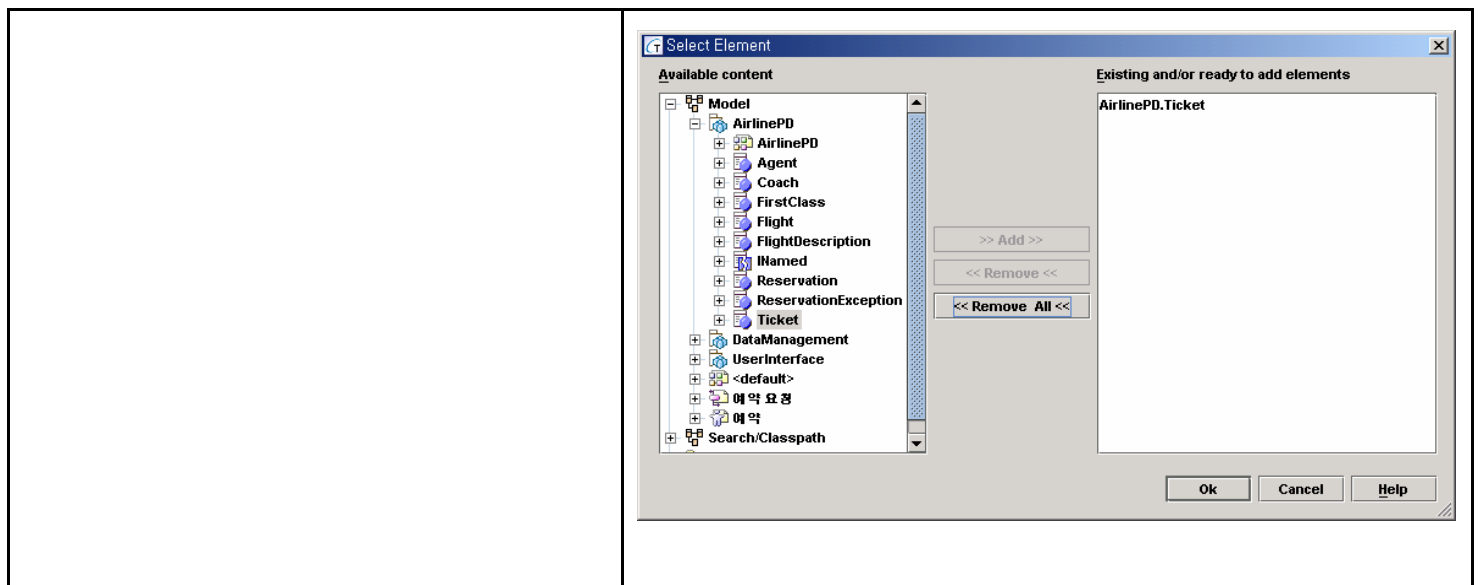
**Abstract product**

(📄)

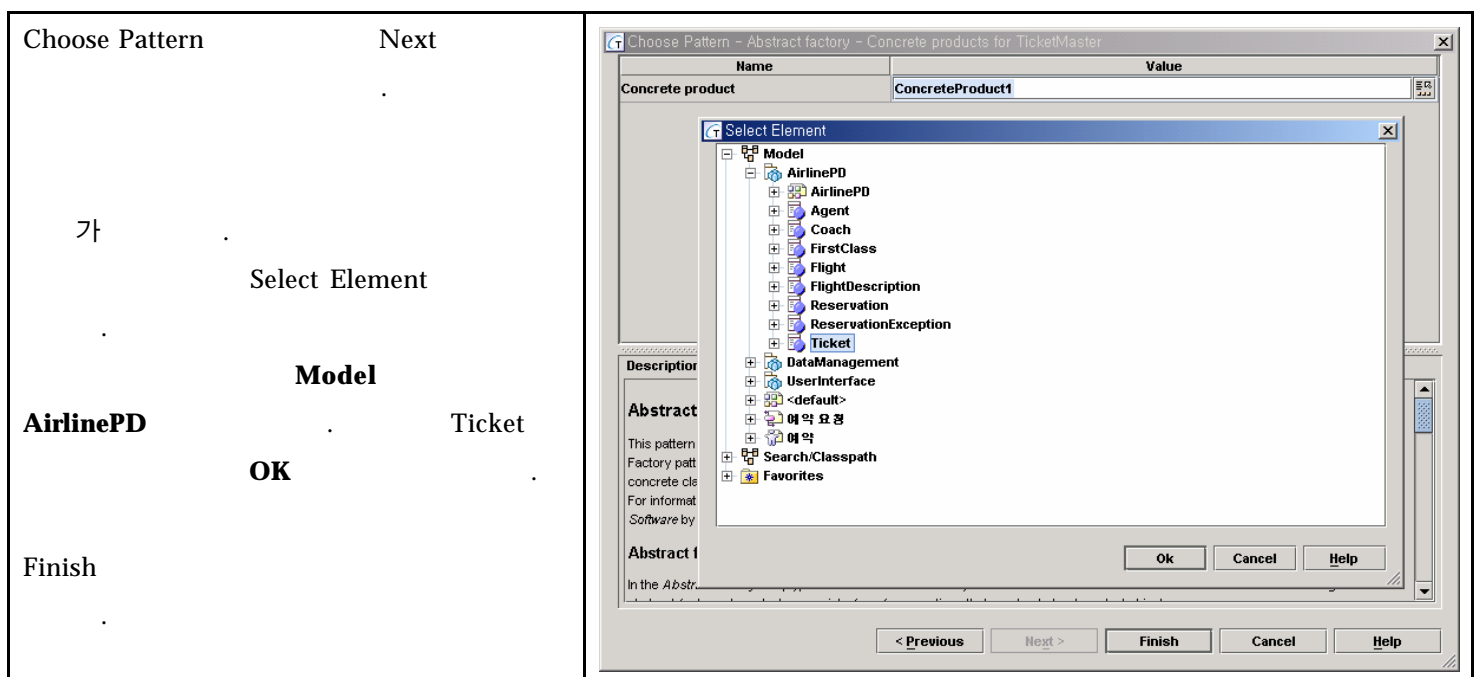
**Patterns**

- Patterns
  - Coad
  - EJB
  - EJB Clients
  - J2EE
  - Oracle9i
  - Sun Java Center J2EE Patterns
  - WebServices
  - Coad Classes
  - GoF
    - Abstract factory**
    - Adapter
    - Chain of Responsibility
    - Composit
    - Decorator
    - Factory M
    - Observer
    - Proxy
    - Singleton
    - State
    - Visitor

Name	Value
Abstract factory	ICreateTickets
Abstract product	AirlinePD.Ticket
Concrete factory	TicketMaster
Copy documentation	<input checked="" type="checkbox"/>
Create pattern links	<input checked="" type="checkbox"/>



: Product      Ticket



TicketMaster, ICreateTickets 가 .



:

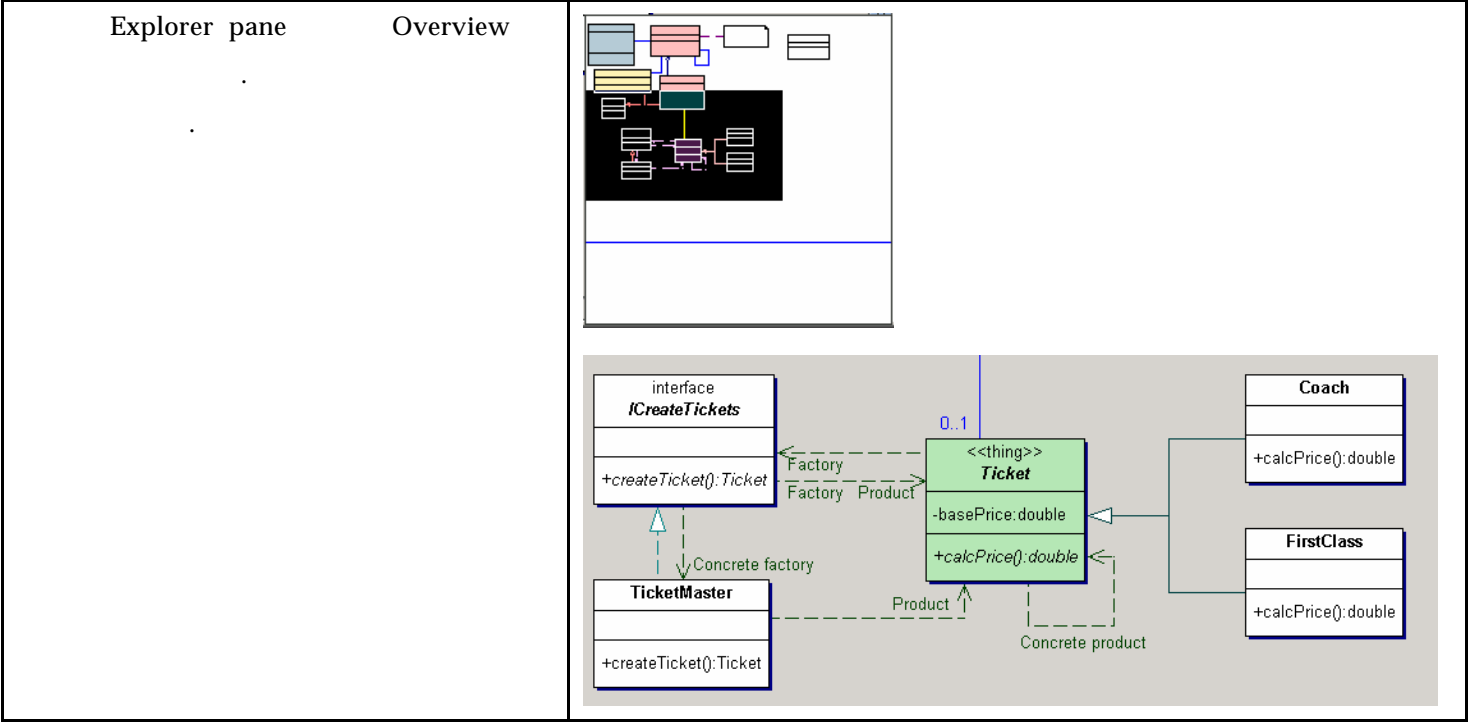
AirlinePD

.

Designer pane

Layout | All(      Ctrl + K)

.



가      Ticket      Reservation

TicketMaster      가

: int      IcreateTickets      (operation)      TicketMaster

1.      IcreateTickets      (operation)
- createTicket(tKind:int):Ticket
2. Ctrl      IcreateTickets      TicketMaster      (operation)      가
- TicketMaster      createTicket()

: Ticket      Reservation      TicketMaster

Editor pane

가      Ticket

Ticket t(inkTicket )

```
public Ticket createTicket (int tKind) {
    Ticket t = null;
    if (tKind == 1)
        t = new Coach();
    else if (tKind == 2)
        t = new FirstClass();
    return t;
}
```

1 2 . ( .)

: TicketMaster Ticket Reservation .

Reservation .

```
public Reservation (String name, int tKind) {
    this.name = name;
    TicketMaster tm = new TicketMaster();
    InkTicket = tm.createTicket(tKind);
}
```

: (compile) . .

Make build Project ( ) . Message pane Builder . Message pane .	<div><div>01010101 <b>Make Project</b> Shift+F7</div><div>01010101 <b>Rebuild Project</b> Ctrl+Shift+F7</div></div>
가 Editor pane .	

(Tip) (Trick)

- GoF Composite

가

- Designer pane

가

- **Ctrl**

Together (operation)

- Designer pane Editor (method)

Designer

. Designer pane

---

Copyright © 2005 Borland Corporation. All rights reserved.

: 2005 3 4

Together

8 :


Together

- 
- (operation)
- (method)

- :
- - /
  - 
  - 
  - 
  - 
  - (Tip) (Trick)

AirlinePD problem  
domain

: AirlinePD	FindRevenue
-------------	-------------

New Diagram Designer ( ) 가 New Diagram (  ) .

New Diagram

New Diagram

UML Together

Class Use Case Sequence Collaboration Statechart

Activity Component Diagram Deployment

Diagram name: 총수입 산출하기

Package name: AirlinePD

☐ Include in current diagram






Description:

비행 예약을 위한 총수입을 산출한다.

Click Ok to create a new diagram.

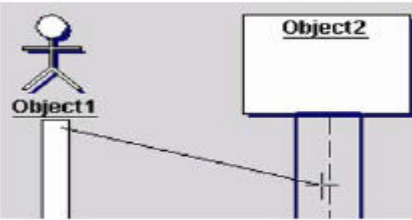
Ok Cancel Help

( )

		
	(Object)	
		
	(Self Message)	
		( )

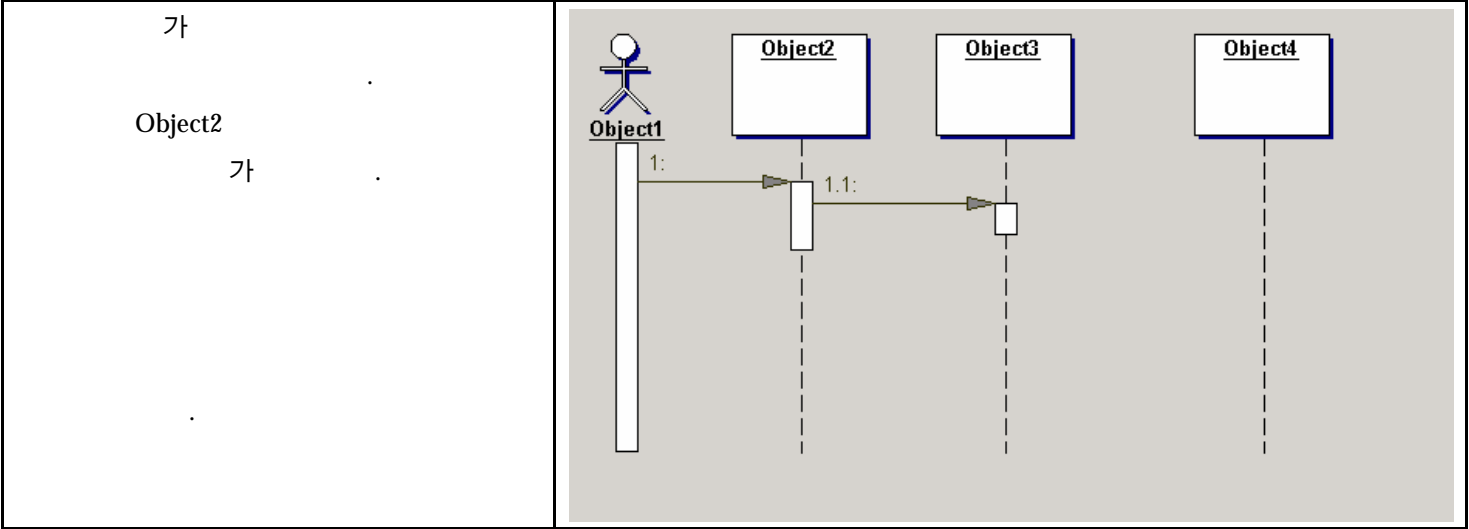
$$\vdots$$

Together

<p>: Object2 가 .</p>	
<p>가 ( )가 .</p>	

Together	가 가
----------	-----

: Object2	(activation bar)	Object3	(Lifeline)	가
-----------	------------------	---------	------------	---



/ (operation)

(operation) . 가

:	가	.
Object2	Flight	.
Object3	Reservation	.
Object4	Ticket	.
가	(Object4)	InkTicket .

Choose Class

가

Object2

More

.

(

.)

가

.

in-place

Choose Class

Unlink Class

Select Class in Model Tree

Import Class

Generate Implementation

Agent

Coach

FirstClass

Flight

FlightDescription

ICreateTickets

INamed

Reservation

ReservationException

Ticket

TicketMaster

More...

InkTicket

Ticket

New

Add to Favorites

Class

Interface

Together

: Flight    Reservation    가    calcPrice()

가

Choose Operation

가

Choose Operation

Unlink operation

Generate Implementation

<constructor>(String,int)

calcPrice():double

getName():String

ticketPurchased():Boolean

More...

:    Inspector    Flight    가

ticketRevenue():double    (operation)

Property(

Property Inspector)

Inspector

(operation)

Together

가

Rename

Question

?

Operation ticketRevenue():double not found in class Flight.  
Create new operation ticketRevenue():double  
or  
Rename operation  
operation1():void to  
ticketRevenue():double ?

Rename

Create

Cancel

(method)

, Together

Rename, Create, Cance

가

Inspector: (message link)

Requirements

Operation

Link

View

Description

HTMLdoc

Name	Value
client	Object1
supplier	Object2
operation	ticketRevenue():double
label	
sequence number	1
creation	<input type="checkbox"/>
destruction	<input type="checkbox"/>
arguments	
return	
return arrow	<input type="checkbox"/>
condition	
iteration	
constraint	

Press Ctrl+Alt+I to finish editing and close Inspector

Together

Tools | Options | Diagram

Level

Diagram Options

Diagram

View Management

Show

Hide

Show subpackage contents

Dependencies

Sequence diagram

Print

Name	Value
Show classname below objectname	<input checked="" type="checkbox"/>
Show message numbers	<input checked="" type="checkbox"/>
Show message description	<input type="checkbox"/>
Create return arrows	<input type="checkbox"/>

Description

These options specify how to display Sequence diagrams.

Levels >>

Ok

Cancel

Apply

Help



. Show message numbers

Reservation

“ 가 0 .”

: Reservation (operation)  
ticketPurchased() hasTicket

Designer ( )  
( ) (operation)  
ticketPurchased()

Property Inspector

Property Inspector Link “return”  
. Together  
Together  
boolean hasTicket = this.ticketPurchased();

Inspector: (message link)	
HTMLdoc Requirements Operation	
Link	View Description
Name	Value
client	Object3
supplier	Object3
operation	ticketPurchased():Boolean
label	
sequence number	1.1.1
creation	<input type="checkbox"/>
destruction	<input type="checkbox"/>
arguments	
return	hasTicket
return arrow	<input type="checkbox"/>

: Reservation

1. if 가 .
2. else if 가 .

(activity bar)

(if )

Reservation

가

Choose statement type

☒ if

☐ else-if

☐ else

☐ for

☐ while

☐ do-while

☐ try

☐ catch

☐ finally

☐ switch

☐ case

☐ default

Ok

Cancel

(if )

가

Inspector

if

“statement expression”

if(hasTicket)

Inspector: StatementBlock1

DescriptionHTML.docRequirements

PropertiesHyperlinkView

NameValue

statement typeif

statement expres...hasTicket

Press Ctrl+Alt+I to finish editing and close Inspector

: ifTicket가calcPrice()cost

if

Return property (operation)

Inspector

Property

Inspector: (message link)

HTMLdoc

Requirements

Operation

Link

View

Description

Name	Value
client	Object3
supplier	InkTicket
operation	calcPrice():double
label	
sequence number	1.1.2.1
creation	<input type="checkbox"/>
destruction	<input type="checkbox"/>
arguments	
return	cost
return arrow	<input type="checkbox"/>

Press Ctrl+Alt+I to finish editing and close Inspector

1.1.1: hasTicket:=ticketPurchased():Boolean

if(hasTicket)

1.1.2.1: cost:=calcPrice():double

else

Together (method) 가 Reservation calcPrice() return

: Reservation calcPrice()

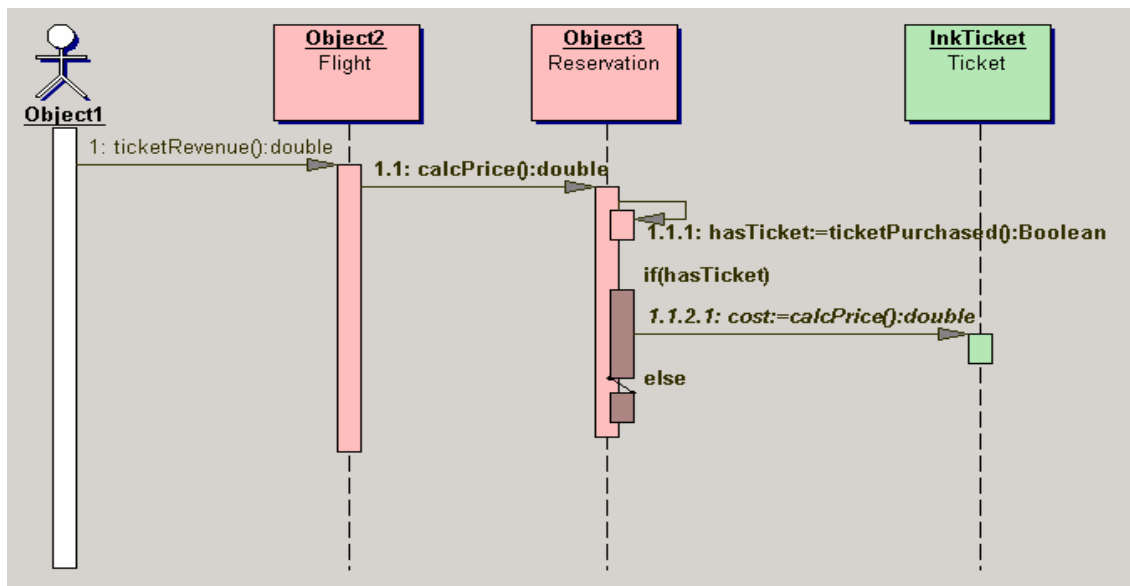
Reservation calcPrice()  
**Generate Implementation**

Show as Sequence

Show as Collaboration

Generate Implementation

가 Together  
**Generate Implementation**



### Reservation.calcPrice()

```

public double calcPrice() {
    //      #1.1.1   this:AirlinePD.Reservation
    boolean hasTicket = this.ticketPurchased();
    if (hasTicket) {
        //      #1.1.2.1   InkTicket:AirlinePD.Ticket
        double cost = InkTicket.calcPrice();
    }
    else { }
}

```

return

1. if                      return                      return cost;
2. else    return                      return 0;

: Reservation.calcPrice()    Flight.ticketRevenue()

Together           가  
Make



Generate Sequence Diagram  
(operation)

Flight가

makeReservation()

Generate Sequence Diagram Expert

java.util

AirlinePD

<<moment-interval>>  
Flight

-date:int

+makeReservation(name:String,tKind:int)  
+numberOfTickets():int  
+ticketRevenue():double

Generate Sequence Diagram...

Hyperlink To

Tools

Refactoring

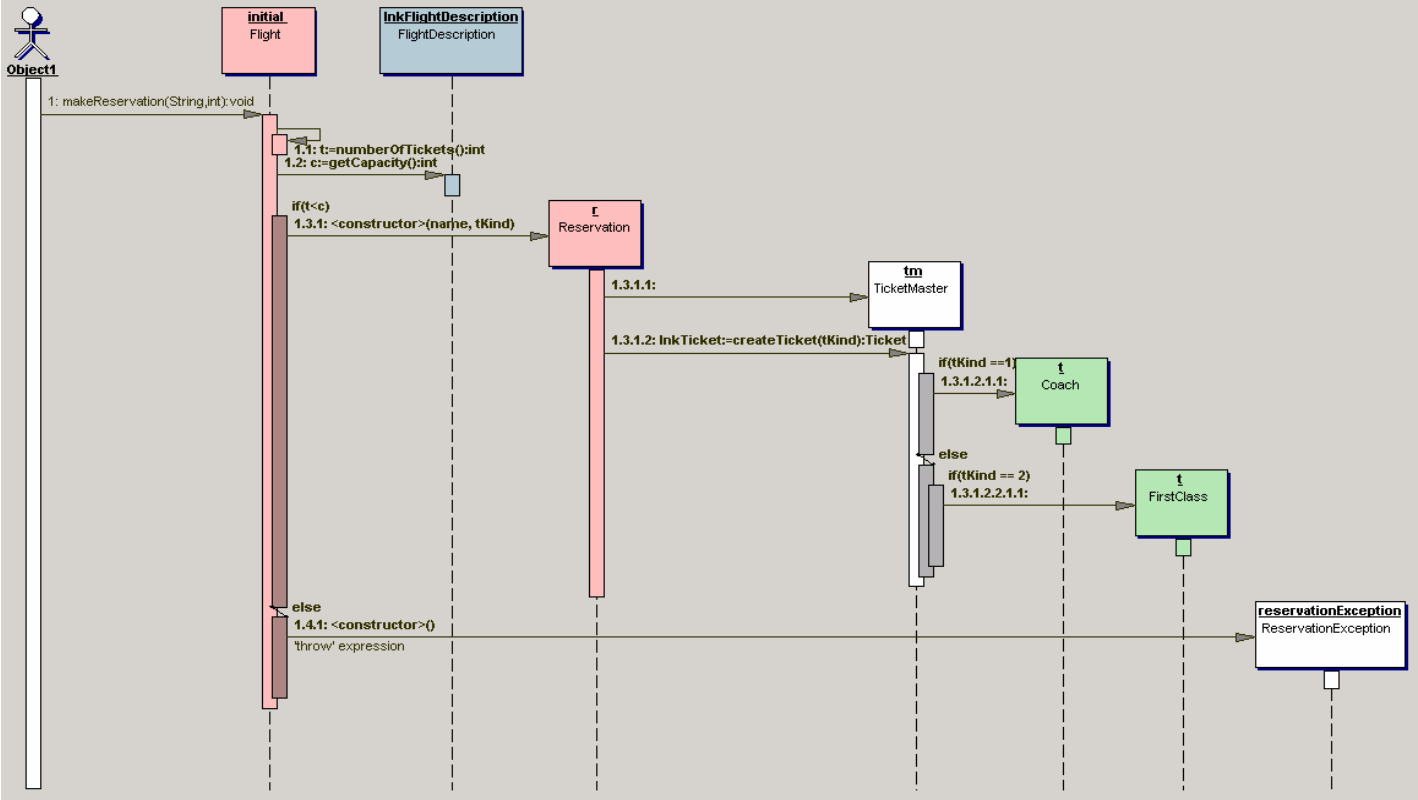
Generate Sequence Diagram

Package/class	Show on diagram...	Show implementati...
AirlinePD	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Coach	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Flight	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
FlightDescription	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Reservation	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
TicketMaster	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
java.util	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
ArrayList	<input type="checkbox"/>	<input type="checkbox"/>

OkCancel

Flight.makeReservation(1)

(method)



: Fligh.makeReservation(1)

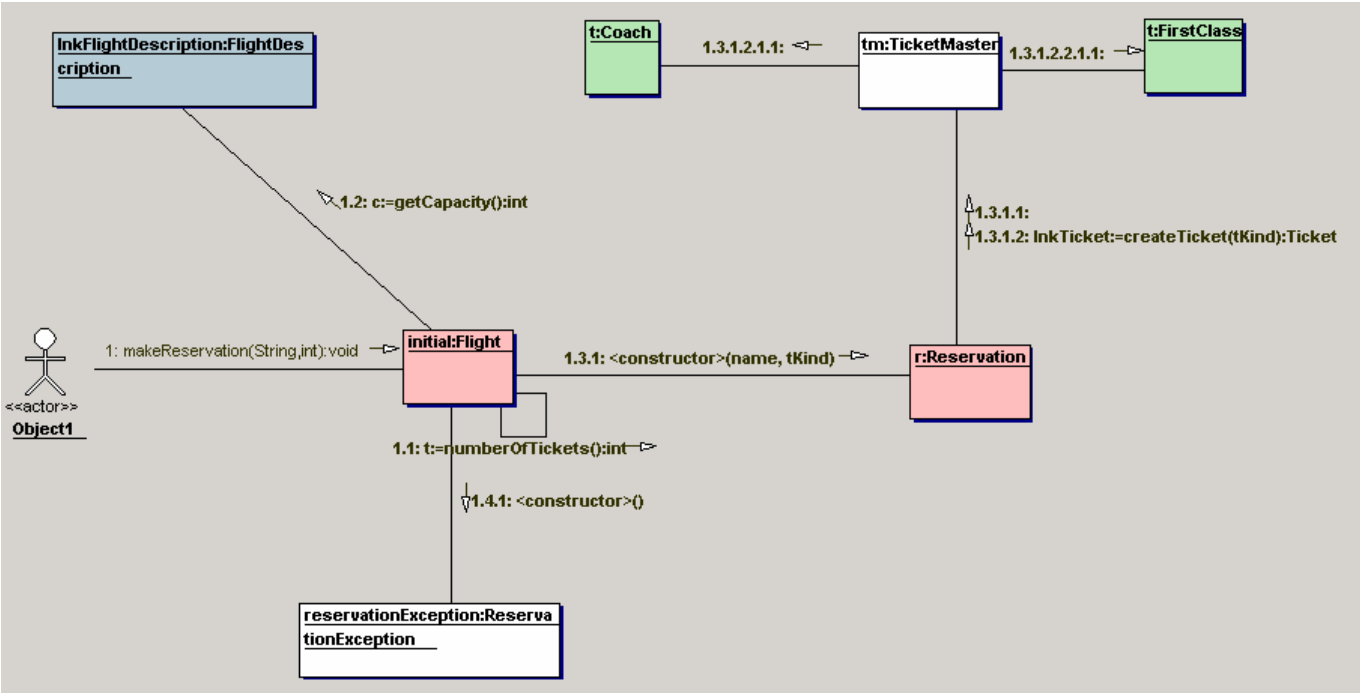
(Collaboration diagram)

(Collaboration Diagram)

. (Together

.)

	<div>Show as Sequence</div> <div>Show as Collaboration</div> <div>Generate Implementation</div>
	<div>Show as Sequence</div> <div>Show as Collaboration</div> <div>Generate Implementation</div>



Together ( ) 가

(operation) Together makeReservation Flight

. AirlinePD Flight.makeReservation(1)

Property Inspector 가

:	가	.
1.	/ Flight.makeReservation(1)	
2.	Request Reservation	
3.	AirlinePD Flight	
4.	AirlinePD Agent	

Inspector    Hyperlink  
Element  
  
가  
Select Element    가

: makeReservation(1)

**(Tip)      (Trick)**

- Together      Together      가
- Property Inspector  
Hyperlink To
- Together
- in-place
- void가      Inspector



Together

9 :

Together HTML, RTF, PDF  
(project entity)  
가 Together

- :
- HTML
  - RTF
  - (Tip) (Trick)

HTML

Together Javadoc property  
Javadoc

: airline HTML

HTML

Project | Documentation

Scope All

Options

( , ), HTML

( , ), 가

Project

Documentation Generate HTML... Ctrl+Shift+H

Make Project Shift+F7 Generate Using Template...

Rebuild Project

Generate HTML

Scope

☐ Current package ☐ Current diagram

☐ Current package with subpackages ☒ All

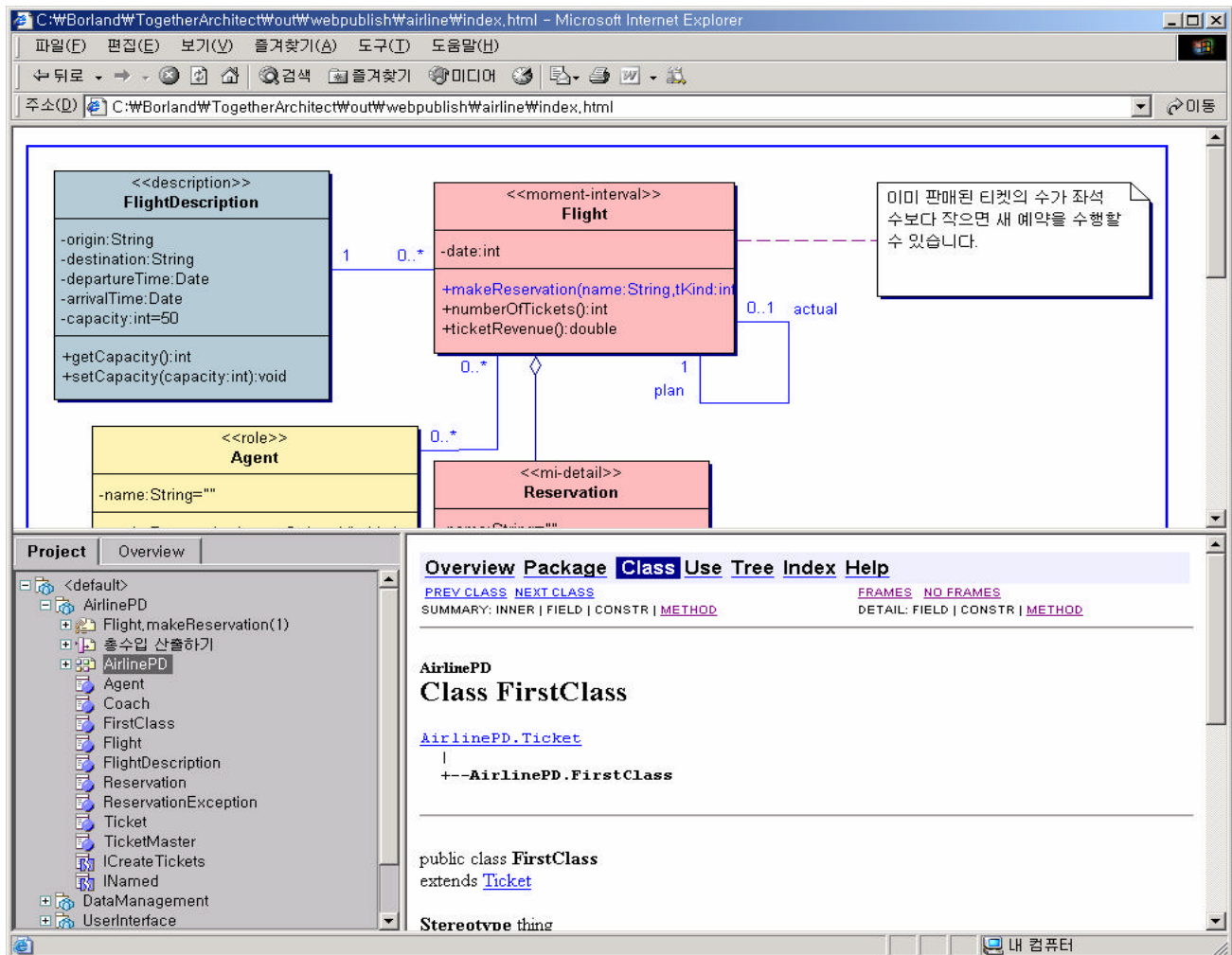
Options

Output folder: C:\Borland\TogetherArchitect\out\webpublish\airline\

☒ Include diagrams ☒ Include navigation tree

☒ Launch HTML browser

Ok Cancel Options Help



Together

HTML

가

. Project

. Overview

가

## RTF

Together

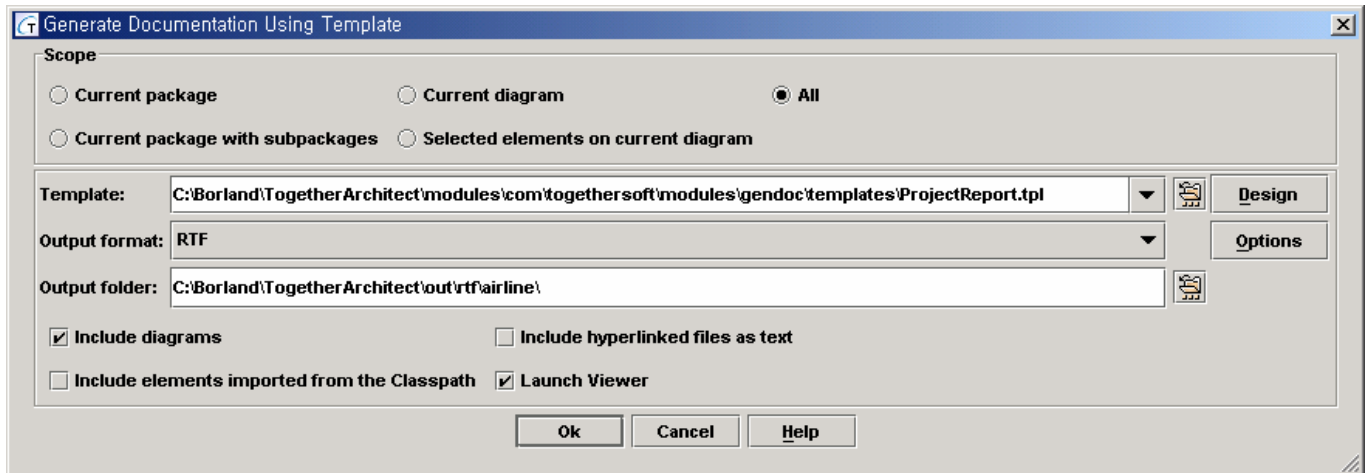
, HTML, PDF, RTF

: airline

RTF

Together      RTF

## Project | Documentation | Generate using Template



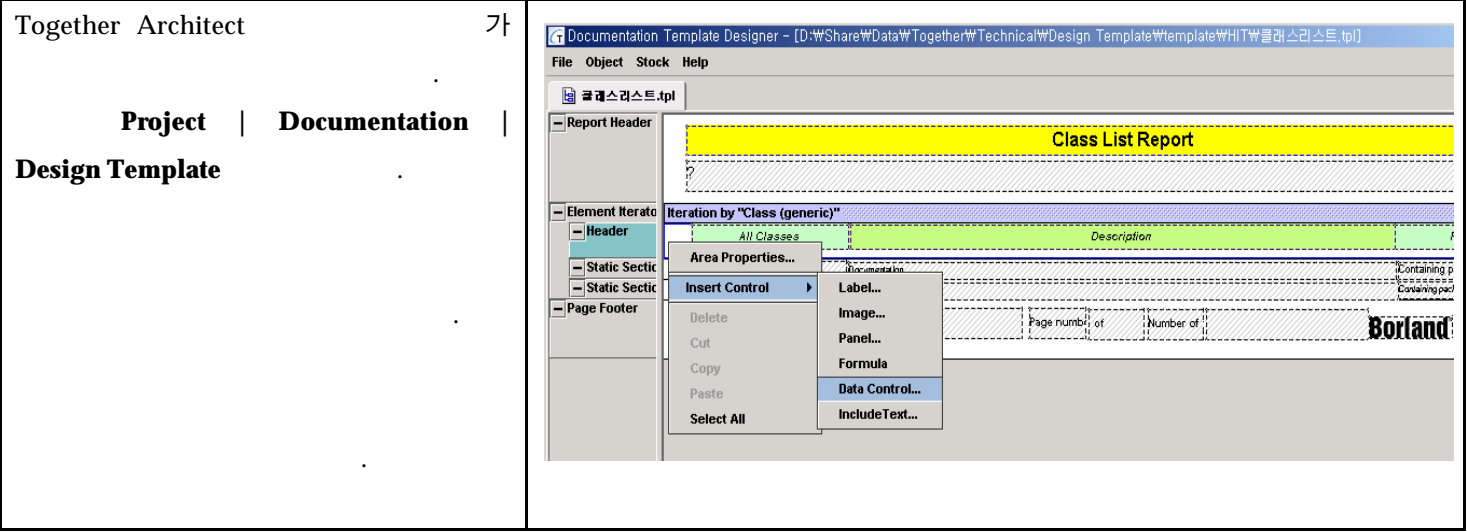
Page 1 of 27

## airline Project Report

Thu Mar 03 20:45:50 KST 2005

### Table Of Contents

Root Package .....	3
Class Diagrams .....	3
Class Diagram <default> .....	3
Package Node Detail .....	4
Package DataManagement .....	4
Package UserInterface .....	4
Package AirlinePD .....	4
Activity Diagrams .....	4
Activity Diagram 예약 요청 .....	4
Swim Lane Detail .....	5
Swim Lane 비행 예약 .....	5
Activity 예약 거부 .....	5
Activity 예약 생성 .....	5
Activity 요청을 받는다 .....	5
Activity 티켓수 체크 (bit) .....	5
Decision Decision1 .....	5
Decision Decision2 .....	5
SyncBar SyncBar1 .....	6
SyncBar SyncBar2 .....	6
Swim Lane 예약 / 티켓 서비스 .....	6
Activity 티켓 발행 .....	6
Swim Lane 항공기 / 비행 설명 .....	6
Activity 수송 인원수 체크 (cap) .....	6
State Detail .....	6
EndState EndState1 .....	6
StartState StartState1 .....	6
UseCase Diagrams .....	7
UseCase Diagram 예약 .....	7



: Together Architect가 airline PDF .

File | Print Documentation PDF 가 .

(Tip) (Trick)

- Together Property Inspector
- , Together “pick folder” (📁)
- HTML . RTF
- . Diagram (page break)
- . Diagram **Show page borders**
- **Layout All for Printing** Together
- 가 .

Together

10 : Audit Metric

Together , Metric .  
Audit Metric . Audit ,  
가 . Metric .

Together Java( ), C++, VB.NET( )eh .  
airline . 가 Together CashSales

- :
- Audit
  - Audit
  - Metric
  - (Tip) (Trick)

Audit

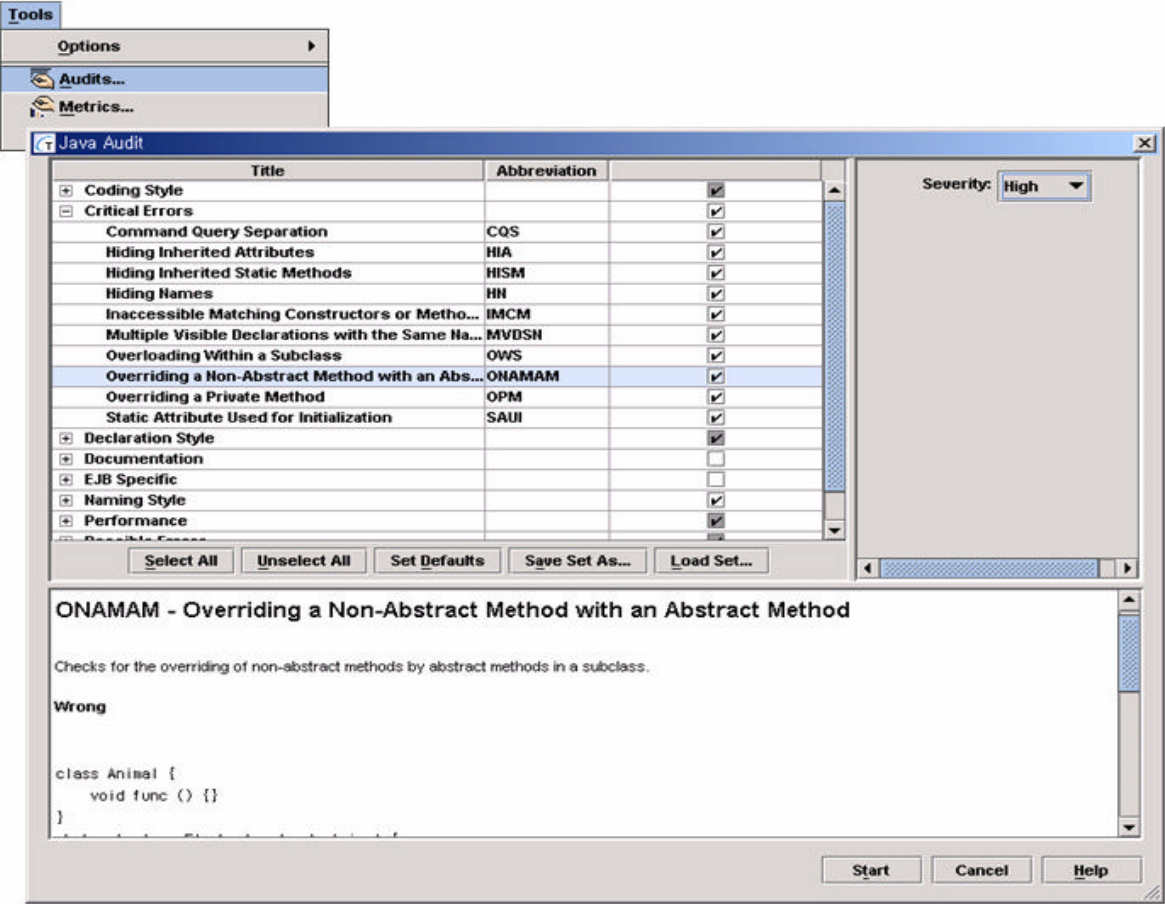
Quality Assurance / 가 가 Together **Tools**  
| **Acitivate/Deactivate Features** . Quality Assurance .  
airline .

: airline	Audit	.
-----------	-------	---

Quality Assurance **Tools | Audits** Audit . (  
가 Audit  
**Quality Assurance | Audits** .)

Together Audit 가 .

Airline project audits      **Select All**      . (Load Set      Java      Sun  
Audit      .)



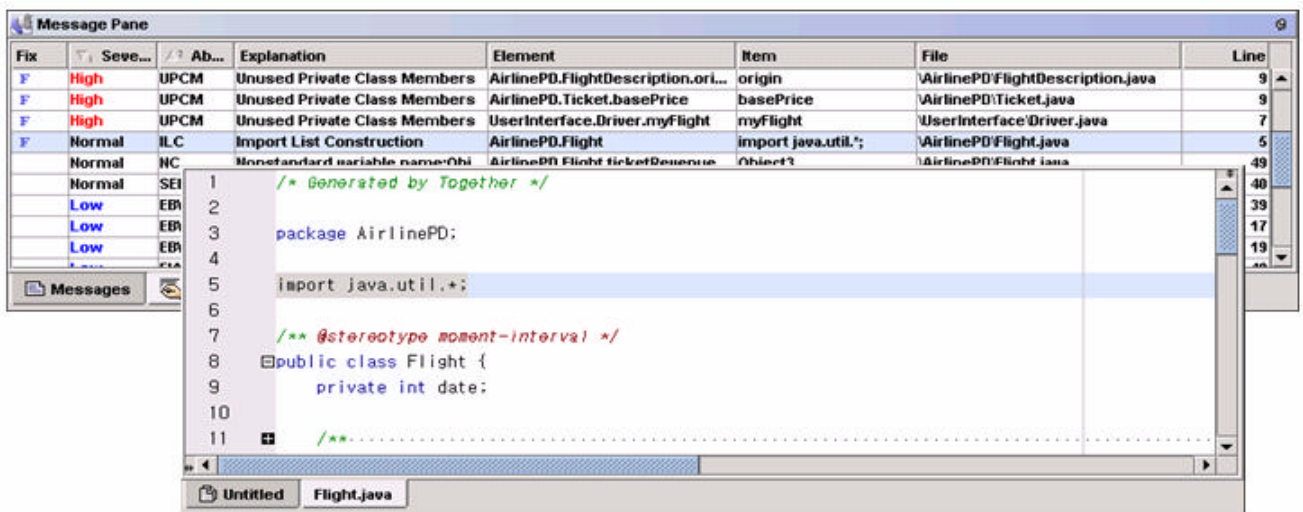
Java Audit      가      Audit      . Java Audit  
Low, Medium, High

Audit      .

Java Audit      Audit      .  
가      .

Airline      Audit      Together      Message pane      .

- 
- 가      .
- Editor pane      가      .



Audit import 가 .

## Audit

Together Audit 가 Audit 가 F .

Audits 가 .

: 가 Audit .

Audit , , 가 .

가 Audit Auto

Correct .

Audit

Auto Correct...

Show Description...

Export

Print...

Sort...

☐ Group By

☐ Sort Ascending

☒ Sort Descending

Auto Correct

Correct this violation ?

Violation

ILC - Import List Construction

\AirlinePD\Flight.java

import java.util.;

Assume this answer for

☒ This violation only

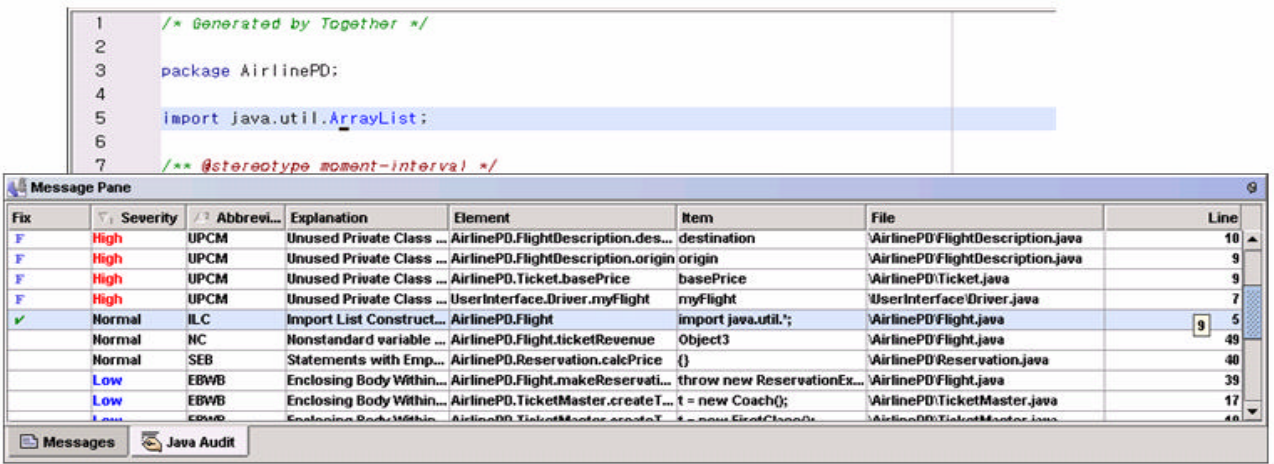
☐ All violations of this type

☐ All violations of this type in selected range

Yes No Close

Audit . Audit Audit

가 . Editor 가 .



: Audit

HTML

Audit

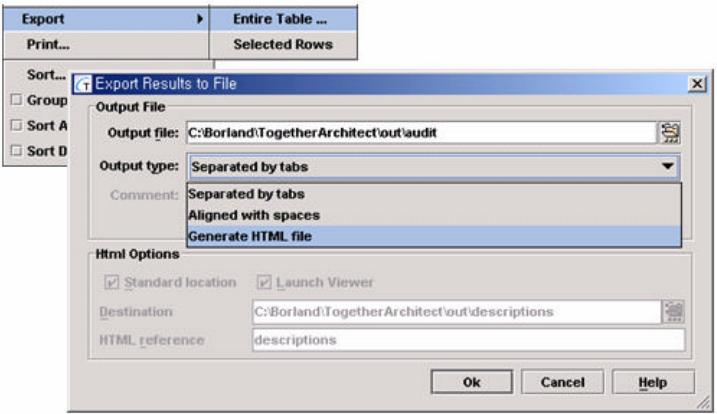
Entire Table

Together

HTML

Export |

HTML



Metric

: airline

Metric

Metric

Tools | Metrics

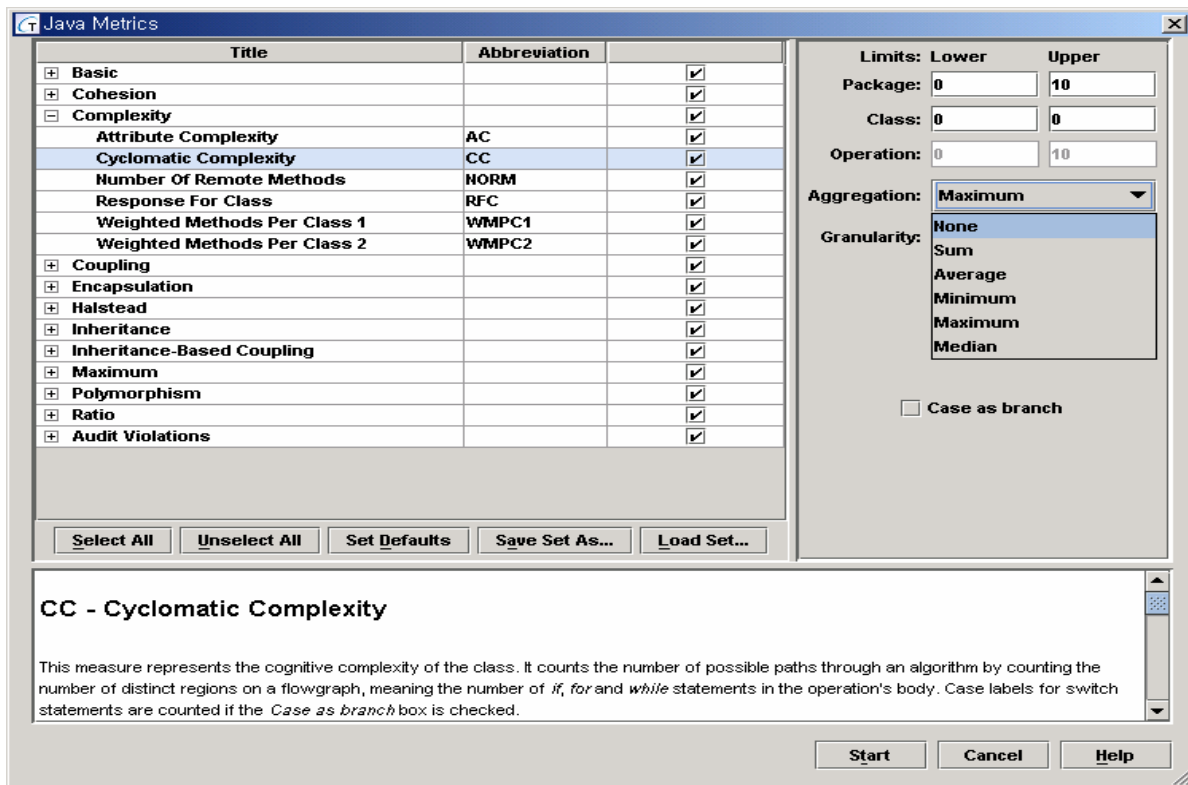
Metric

가 Metric  
Metric

Select All  
(granularity)



Metric 가



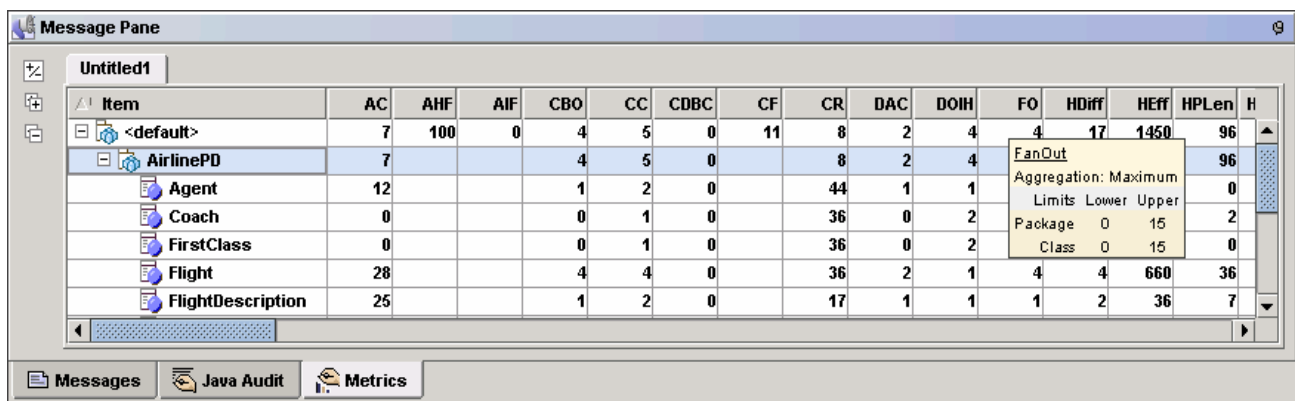
Message pane

가

Metric

Audit

가 가



: Fan Out

. Airline

Fan Out Metric

Metric

가

Metric

Show Description

Description

Bar

Show Description...

Export

Untitled1 - FO

Bar Graph

Description

FO - FanOut

Counts the number of reference formal parameters, return types, types and supertypes are not co

Supported languages: Java, C++

FO FanOut

Untitled1 - FO

Bar Graph

Description

Agent	1
Coach	0
FirstClass	0
Flight	4
FlightDescription	1
ICreateTickets	0
INamed	0
Reservation	2
ReservationException	0
Ticket	0

☐ Selected Rows Only
☒ Auto Update

Class

FO FanOut

Print...

Close

: airline

Metric

● CBO – (Coupling Between Objects) – ( = 4)

● NOC – (Number of Classes) – ( = 30, 30)

Metric

AirlinePD

Kiviat

Metric

Metrics

가

가

Metric

Message pane

Metrics

Kiviat

( NOC)

(CBO, CC, FO)

Metric

Bar graph, Distribution

graph, Kiviat graph

Kiviat ( )

Metric

Untitled5 - AirlinePD

☐ Auto Update

Print...

Close

Metric	AirlinePD	
Kiviat graph	. Together	
Metric	.	

---

**(Tip)**      **(Trick)**

- Audit Metric .
- Metric .
- Kiviat Metric .

# Together

## 11 : Java

Together Editor Message pane Java  
airline CashSales

Windows Together Java2™ SDK 1.4.2 Javac.exe  
Java2 SDK


:  
• (Compile)

•  
•  
•  
•  
• (Tip) (Trick)

(compile)

Airline Driver output 가 (Driver  
UserInterface .)


: Driver.main()  
System.out.println(" Goodbye Airline Project.");


 )

**Project**

Together Java Make

Together Message pane

(Make, )

(Rebuild, )

**Message Pane**

\*\*\* Make started.  
Compilation args: -source 1.4 -classpath C:\Borland\TogetherArchitect\myprojects\airline\src;C:\Borland\Togethe  
\*\*\* Output directory: C:\Borland\TogetherArchitect\out\classes\airline  
\*\*\* Make completed

Messages Builder



configuration

Message pane Together

main (method)가

가

The image shows two screenshots from an IDE. The top screenshot displays the 'Run Arguments and Parameters' dialog box. It has two tabs: 'Application' and 'Applet'. The 'Applet' tab is selected, showing 'Run configuration: Vanilla'. Under 'Select class with \'main\'', 'UserInterface.Driver' is entered. 'Program arguments' is empty, and 'VM options' is '-classic'. The bottom screenshot shows the 'Message Pane' with a green play button icon and the text: 'C:\Borland\TogetherArchitect\jdk\bin\java -classic -classpath C:\Borland\TogetherArchitect\... ..Driver' followed by 'Goodbye Airline Project.' Below the pane are buttons for 'Messages', 'Builder', and 'Run/Debug'.

: CashSales problem\_domain

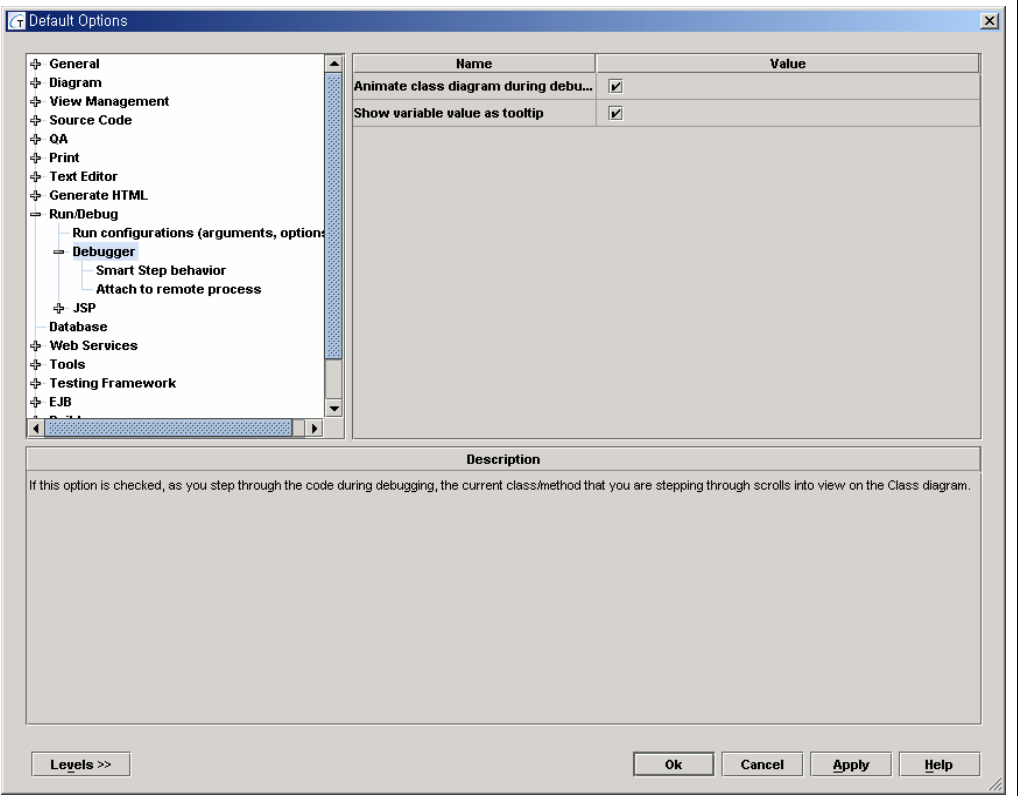
Explorer pane Directory 가 CashSaels.tpr  
Sample | java | CashSales | CashSales.tpr  
Together

: problem\_domain CashSaleDetail (break  
point)




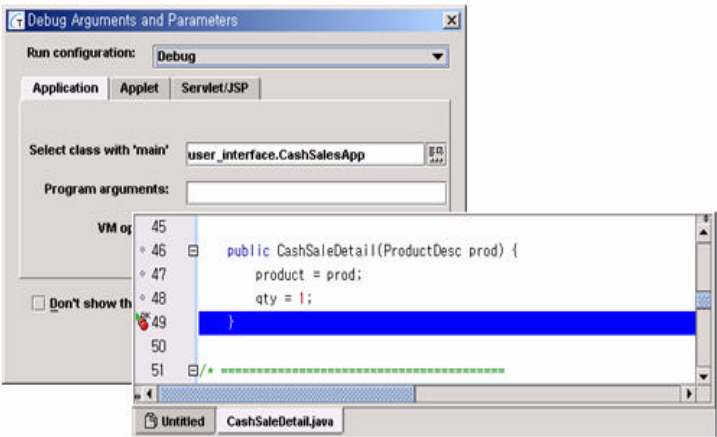
Project Default Options  
가 Run/Debug  
Animate class diagram during  
debugging

Together



: Debug run configuration  
(break point) CashSales . Editor

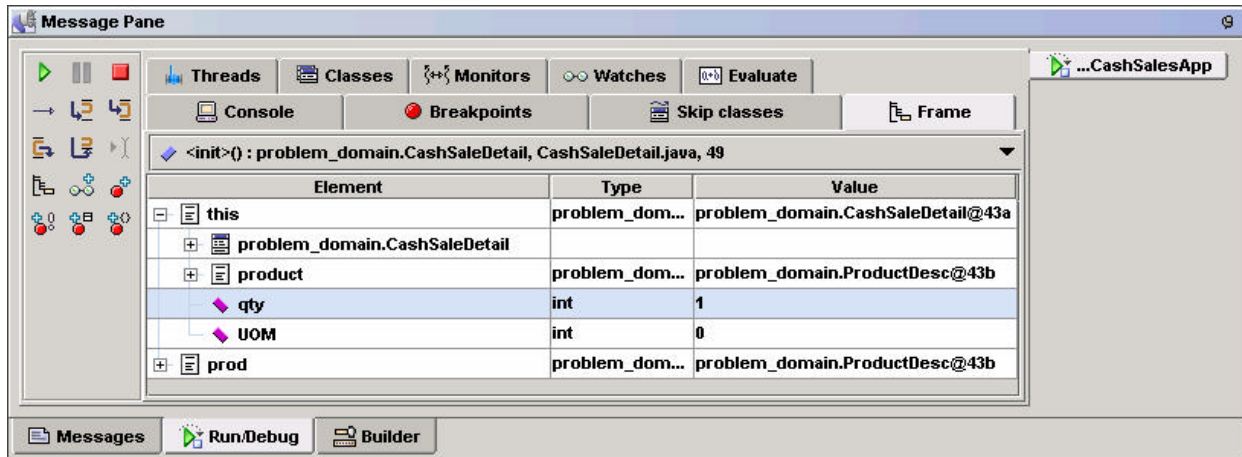
Run in Debugger (  )  
가  
CashSales Debug run  
configuration . **Ok**  
가 “Scan”  
Editor



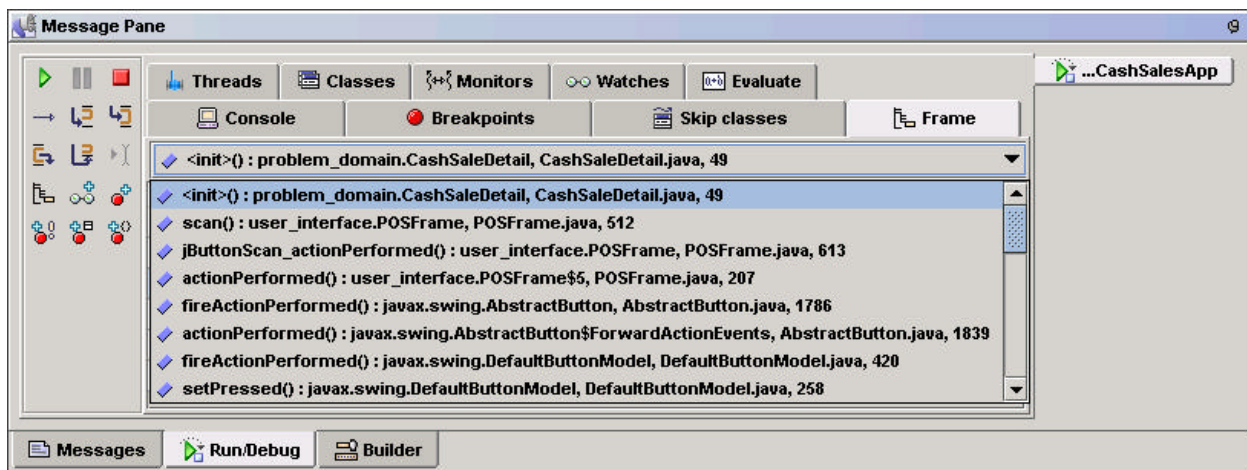
: CashSaleDetail.qty



- . Debugger pane

**CashSaleDetail.qty**


(method)



가



47



The screenshot shows the Eclipse IDE's Message Pane during a debug session. The pane displays a table of variables for the current frame:

Element	Type	Value
this	problem_do...	problem_domain.CashSaleDetail
problem_domain.CashSaleDetail		
product	problem_do...	problem_domain.ProductDe
qty	int	1
UOM		
prod		

A context menu is open over the 'qty' variable, showing options: 'Create Attribute Watch...', 'Create Attribute Breakpoint', 'Change Value...', and 'Show Hex Value'. The 'Change Value...' option is selected. A 'Change Value' dialog box is open, prompting 'Enter a new int:' with the value '47' entered. The dialog has buttons for '확인' (OK) and '취소' (Cancel).

가

Item	Name	Unit	Qty	Price
3	Vienna Sausages	₩3	47	₩141
8	Finch Food	₩2	1	₩2

**(Trick)**

- (build) Make

**Project | Rebuild Project**

Make

- Debugger . Debugger

(breakpoint)

**Remove breakpoint**

Delete

- (method) Smart Step “Step into” ()

가

. (Debugger

가

가

“Skip

classes”

Java

가

“Skip classes”

가

가

.)

- C++ Together

Together

configuration

C++

가