Aspect-Oriented Analysis and Design

The Theme Approach

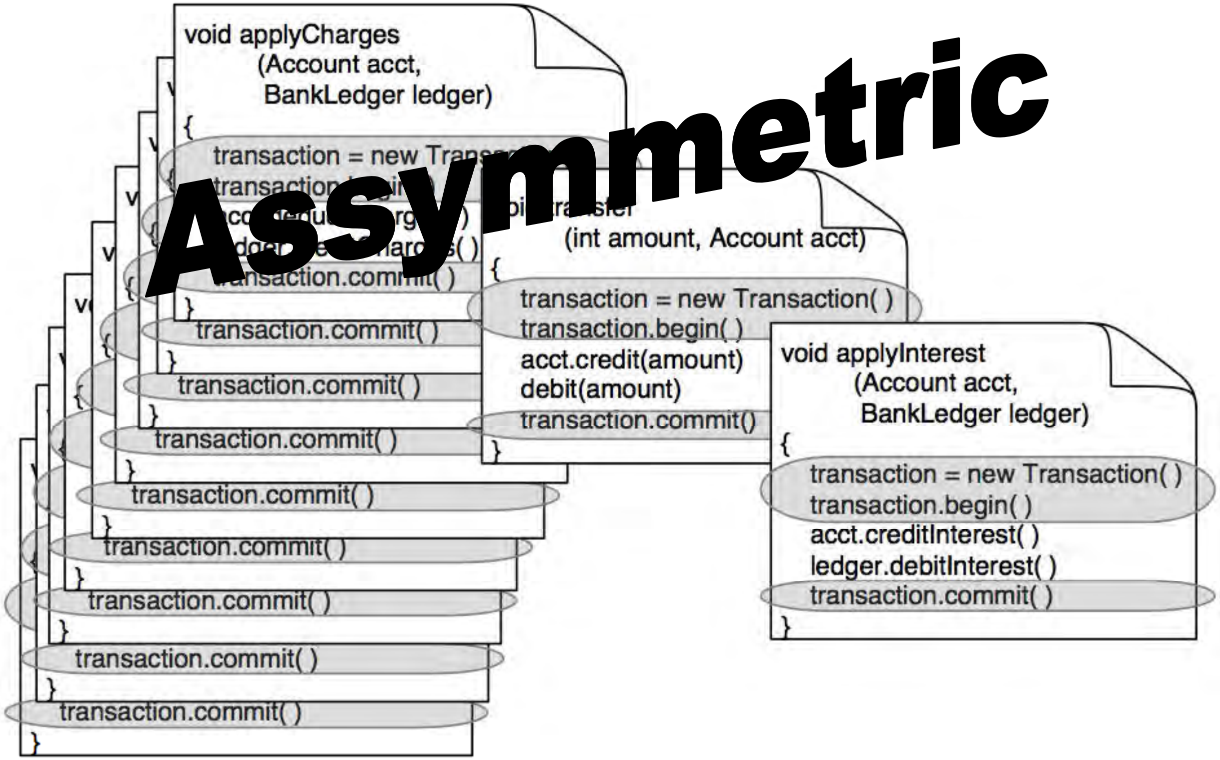
Siobhán Clarke

Trinity College Dublin

BCS Advanced Programming SIG, 11th May 2006

What’s wrong with objects – 1?

- Concerns that crosscut multiple parts of a system cannot be modularized



Assymmetric

```
void applyCharges
        (Account acct,
         BankLedger ledger)
{
  transaction = new Trans
  transaction.begin( )
  acct.deductCharges( )
  ledger.creditCharges( )
  transaction.commit( )
}
```

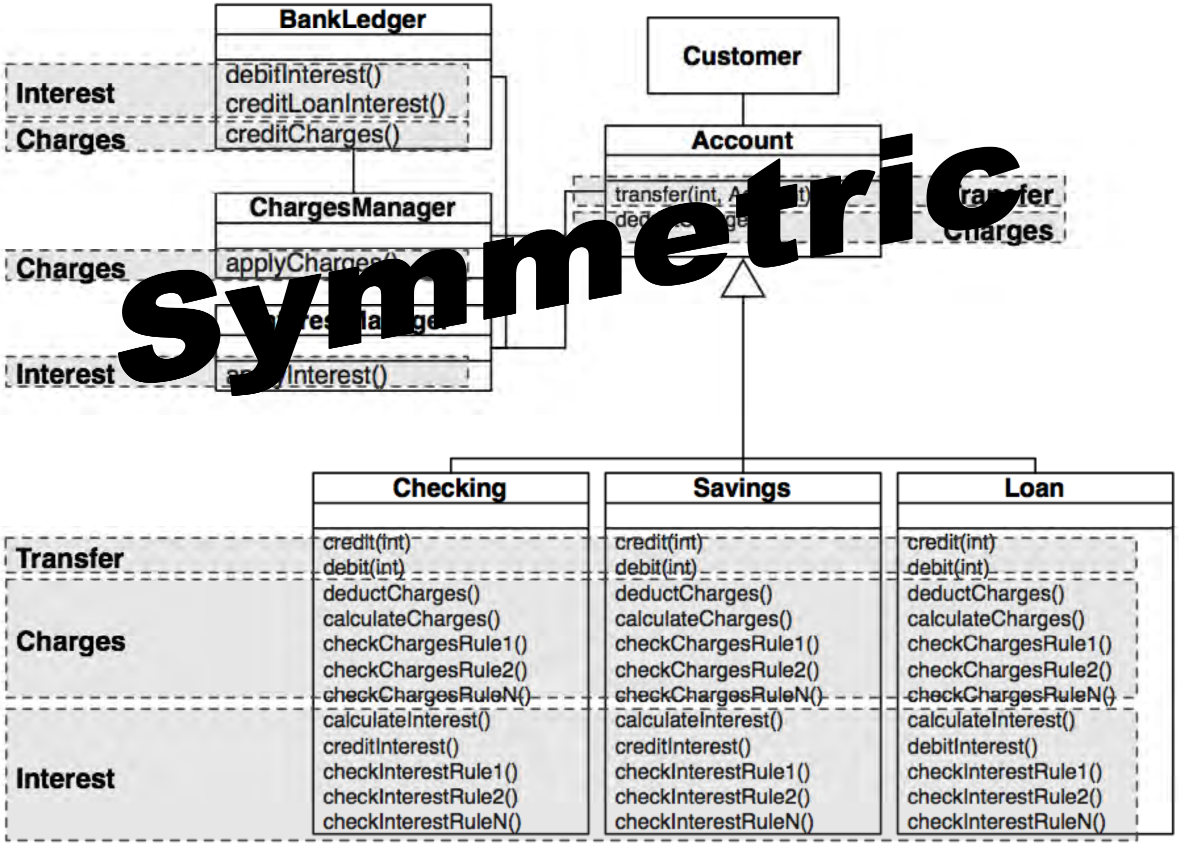
```
  transaction.commit( )
}
  transaction.commit( )
}
  transaction.commit( )
}
  transaction.commit( )
}
  transaction.commit( )
}
  transaction.commit( )
}
  transaction.commit( )
}
  transaction.commit( )
}
```

```
void transfer
        (int amount, Account acct)
{
  transaction = new Transaction( )
  transaction.begin( )
  acct.credit(amount)
  debit(amount)
  transaction.commit()
}
```

```
void applyInterest
        (Account acct,
         BankLedger ledger)
{
  transaction = new Transaction( )
  transaction.begin( )
  acct.creditInterest( )
  ledger.debitInterest( )
  transaction.commit( )
}
```

What’s wrong with objects – 2?

- Class modularization encapsulates multiple concerns

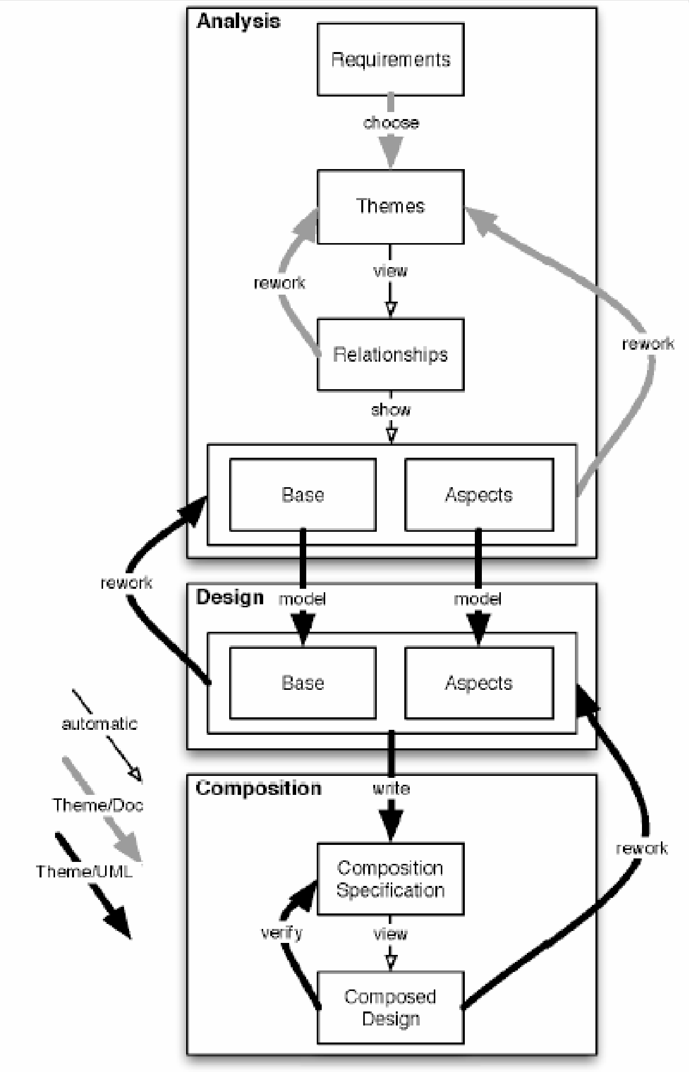


The Theme Approach

Brief History Lesson!

- Early ’90’s: Subject-Oriented Programming
  - Symmetric view of decomposition
- 1997+ Subject-Oriented Design
- 2001: Subject-Oriented Design -> Theme/UML
- 2003+ The Theme Process:
  - Theme/UML – design
  - Theme/Doc – requirements analysis

Overview



Theme/Doc for analysis

View relationships between concerns

Theme/UML for design

Model concerns separately

Theme/Doc

find the themes – select potential concerns from text

R18 A new game will randomly distribute crystals around the game world

R21 If players do not reach their initial location in time, they lose one energy point

R35 Dropped crystals will be re-scattered throughout the game area

R37 Players lose energy at two units per five minute period in a game location

R38 If a player enters a location that has no players or characters in it, they may pick up any crystals or magical items there

R40 When a game starts, players gain 10 units of energy

R42 Energy is gained by two units when a player picks up a crystal entering a location

R61 When two players meet on entering a location, they perform a duel of rock paper scissors

R80 When a player completes a physical test challenge successfully they gain three units of energy and win a crystal

Theme/Doc

find the themes – select potential concerns from text

R18 A new game will randomly **distribute** crystals around the game world

R21 If players do not **reach** their initial location in time, they **lose** one energy point

R35 Dropped crystals will be **re-scattered** throughout the game area

R37 Players **lose** energy at two units per five minute period in a game location

R38 If a player **enters** a location that has no players or characters in it, they may **pick up** any crystals or magical items there

R40 When a game **starts**, players **gain** 10 units of energy

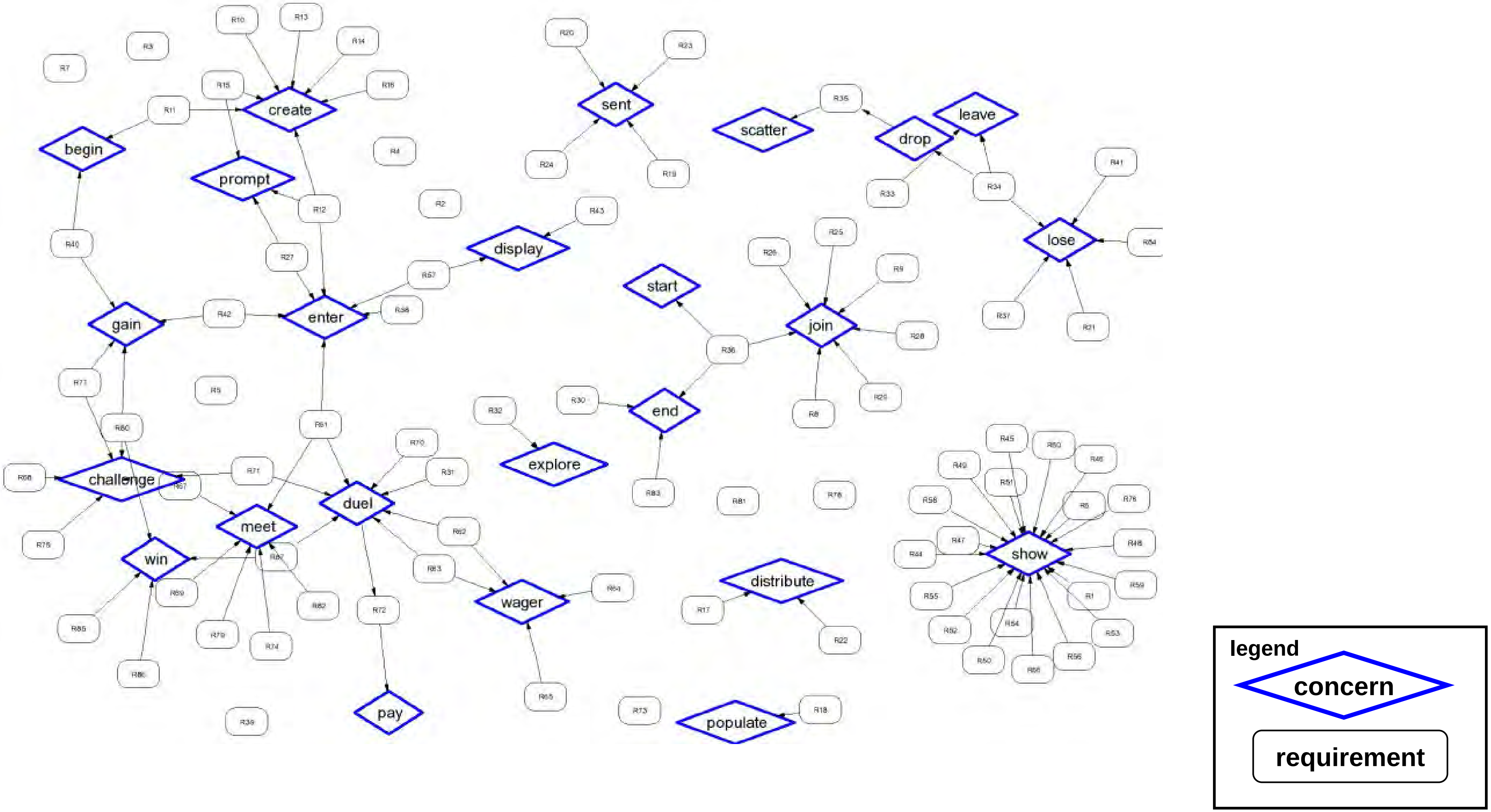
R42 Energy is **gained** by two units when a player **picks up** a crystal **entering** a location

R61 When two players **meet** on **entering** a location, they **perform a duel** of rock paper scissors

R80 When a player **completes** a physical test **challenge** successfully they **gain** three units of energy and **win** a crystal

Theme/Doc

graphing the relationships between the concerns

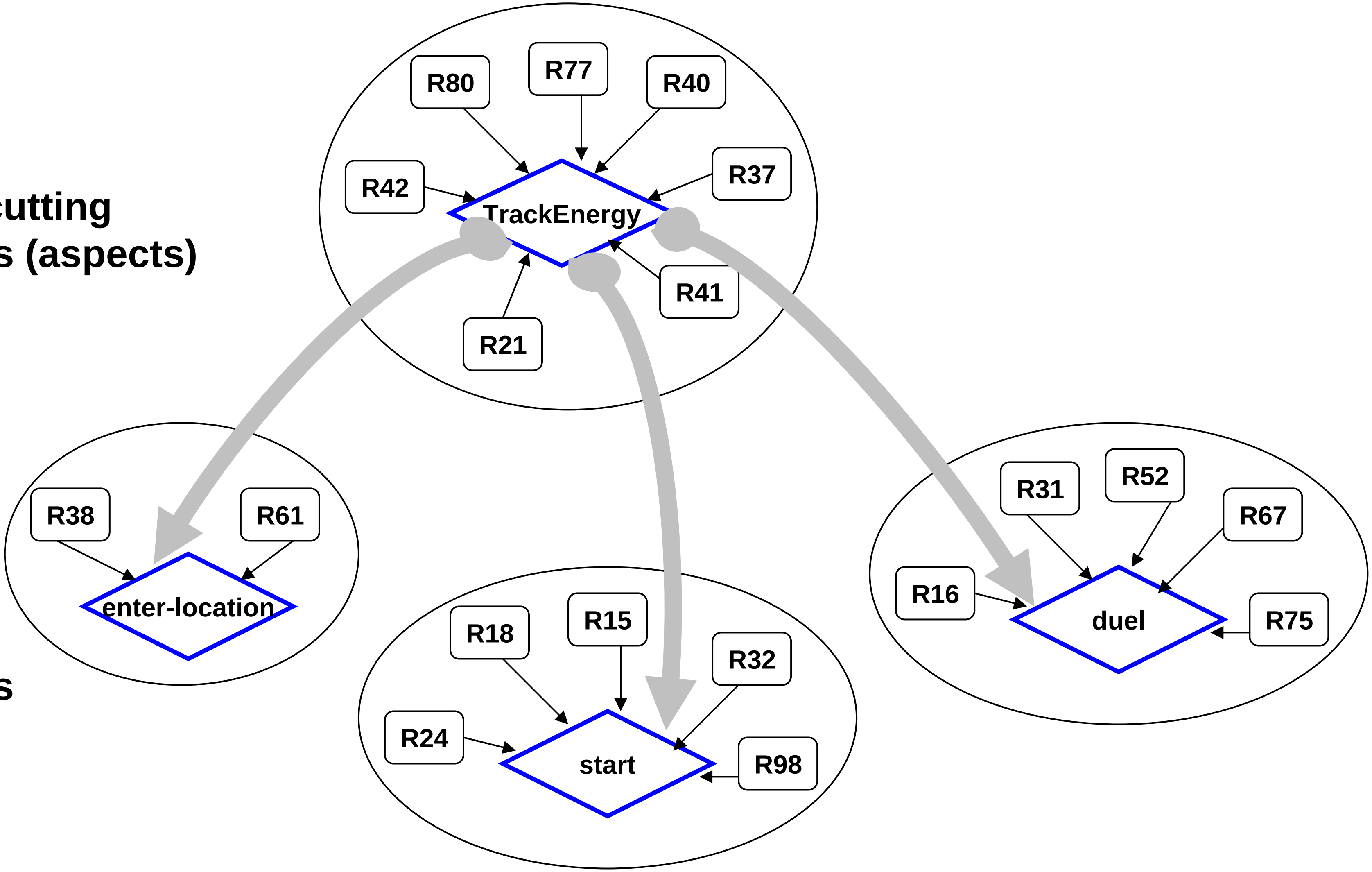


Theme/Doc

a look forward to the final goal

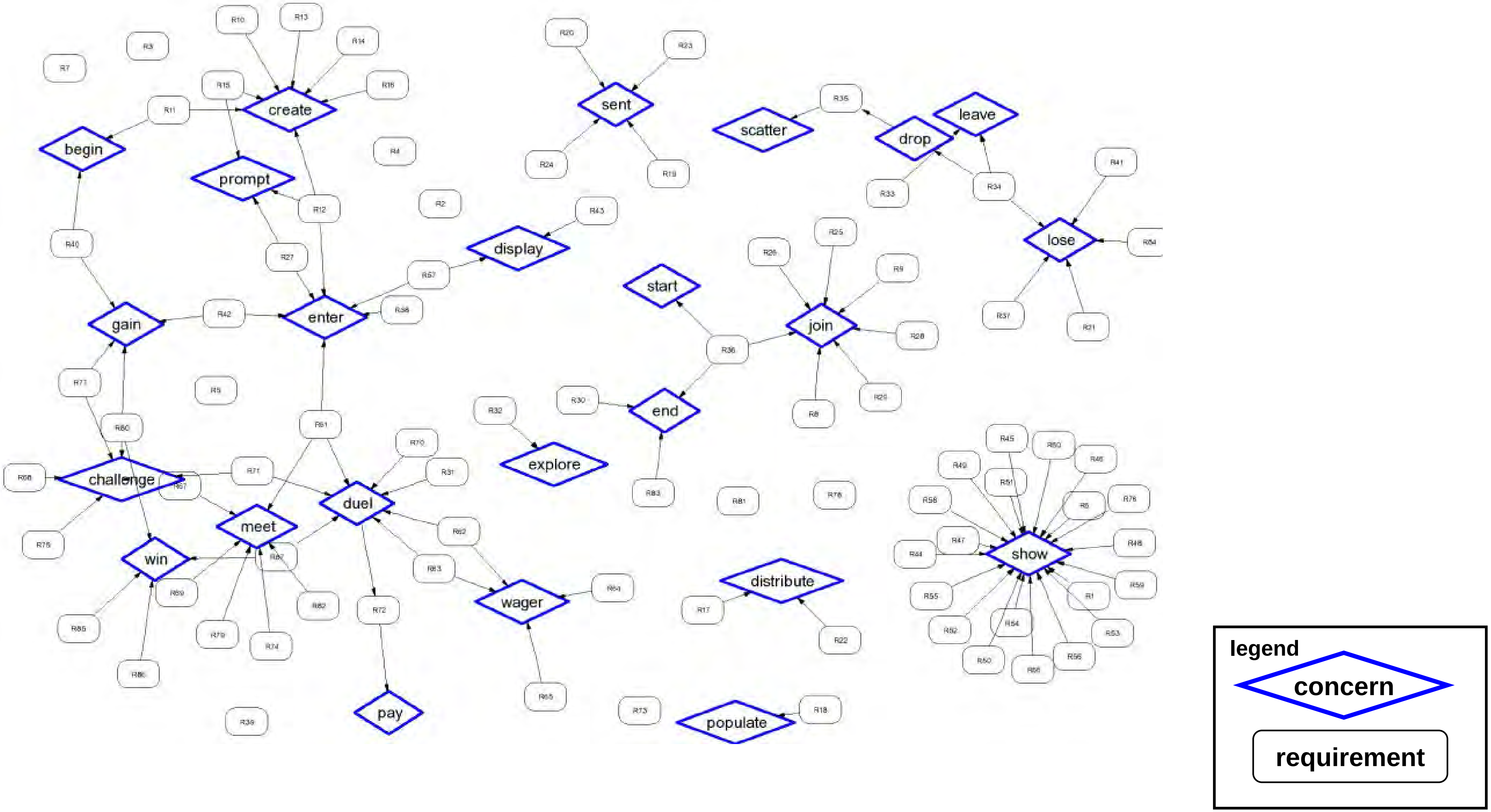
crosscutting themes (aspects)

base themes



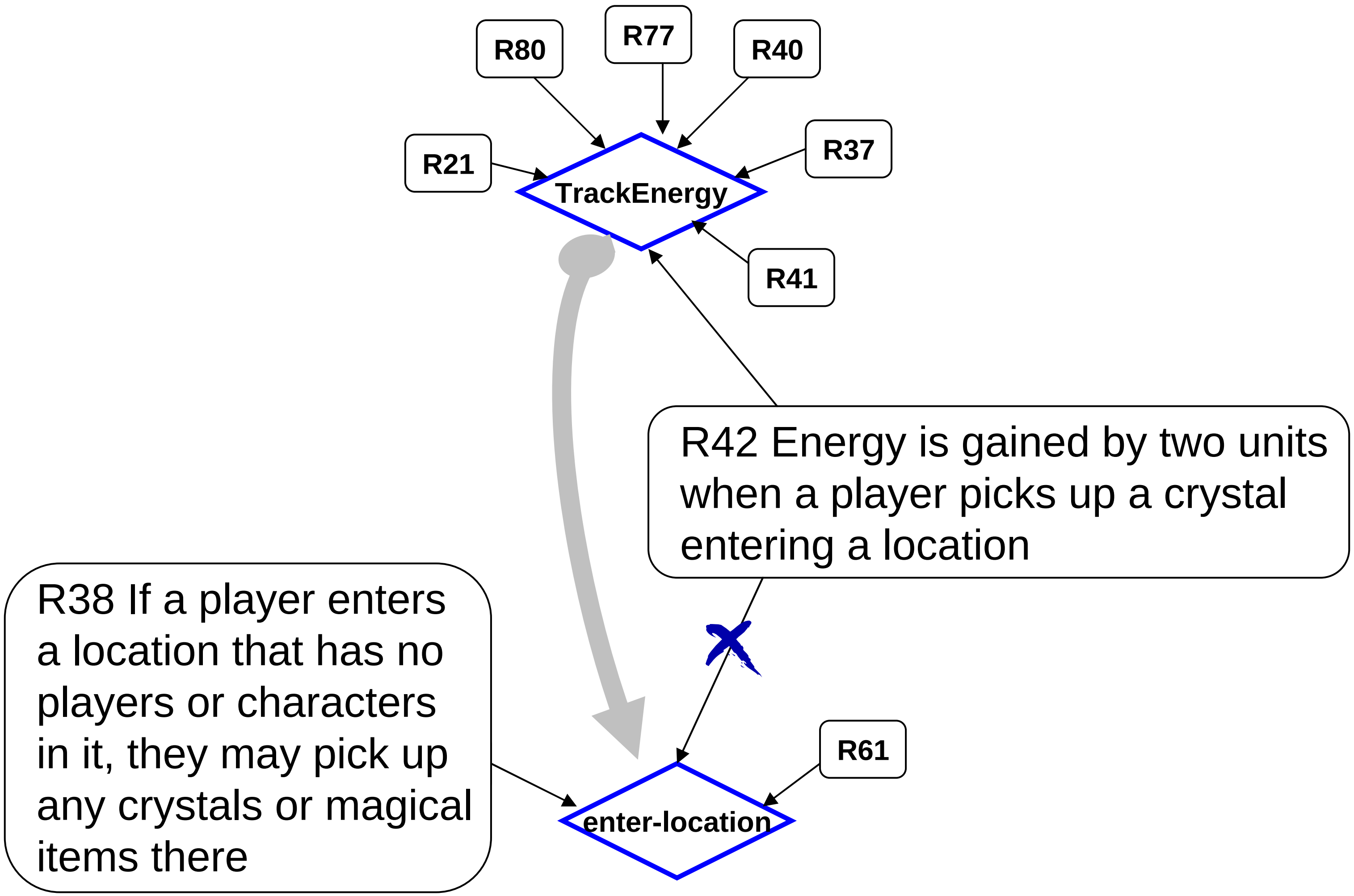
Theme/Doc

graphing the relationships between the concerns



Theme/Doc

getting there.. aspects (probably from shared requirements)



R42 Energy is gained by two units when a player picks up a crystal entering a location

R38 If a player enters a location that has no players or characters in it, they may pick up any crystals or magical items there

Theme/Doc

getting there.. base themes

Theme too general?

split into smaller ones

Themes too similar?

unify synonyms, group

Theme not really useful?

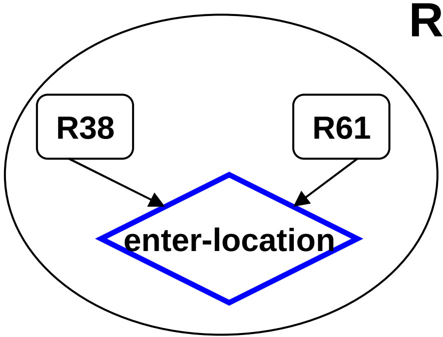
delete it!

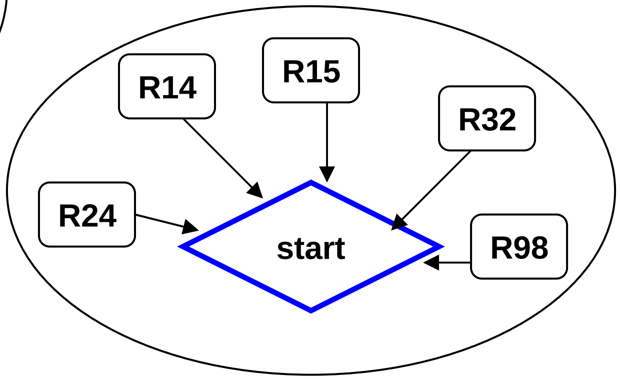
Requirement “orphaned”?

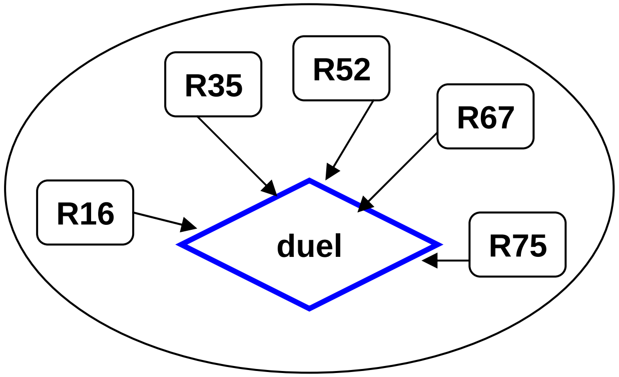
find a home

Requirement ambiguous?

resolve and refine

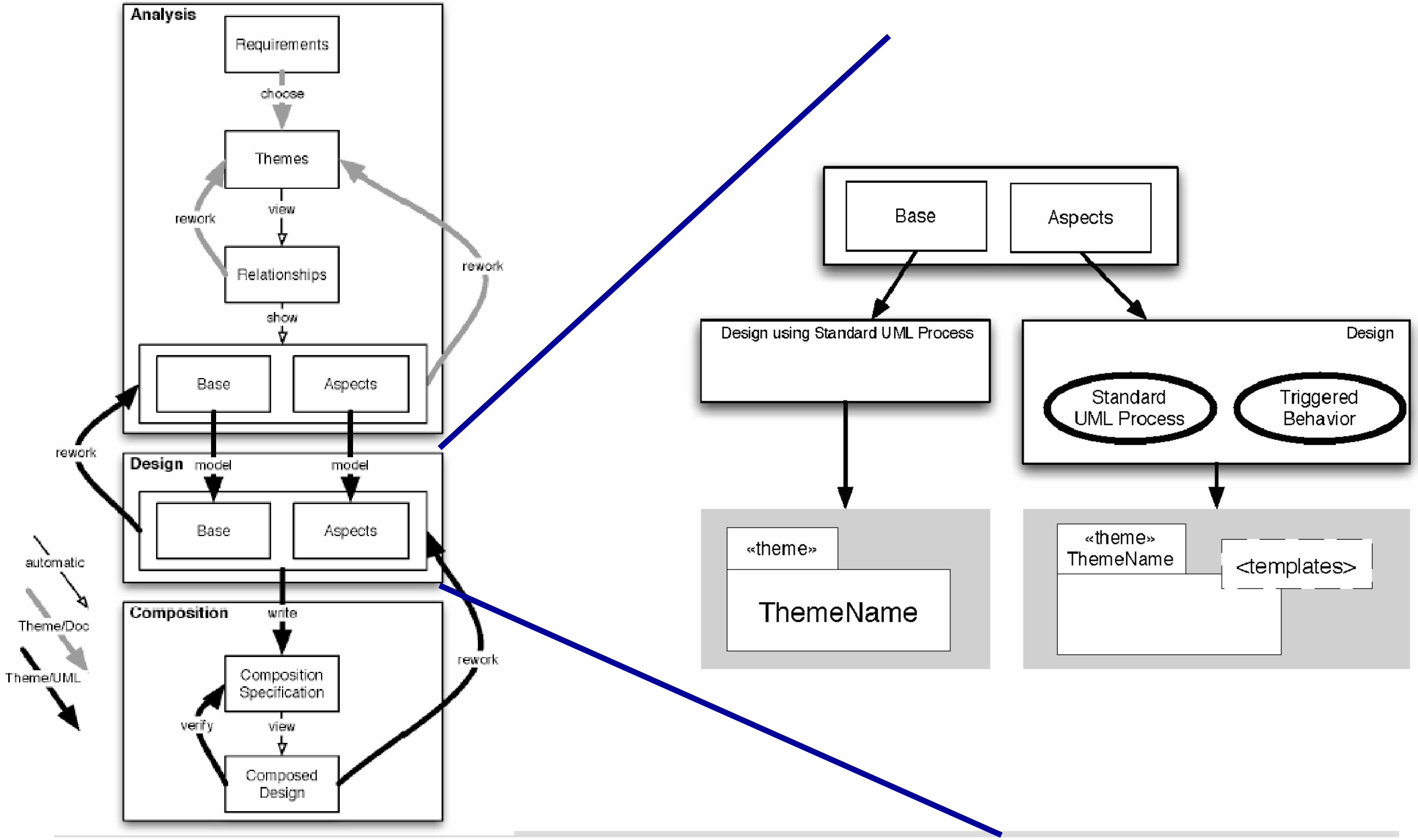






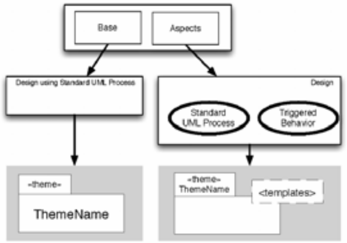
Theme/UML

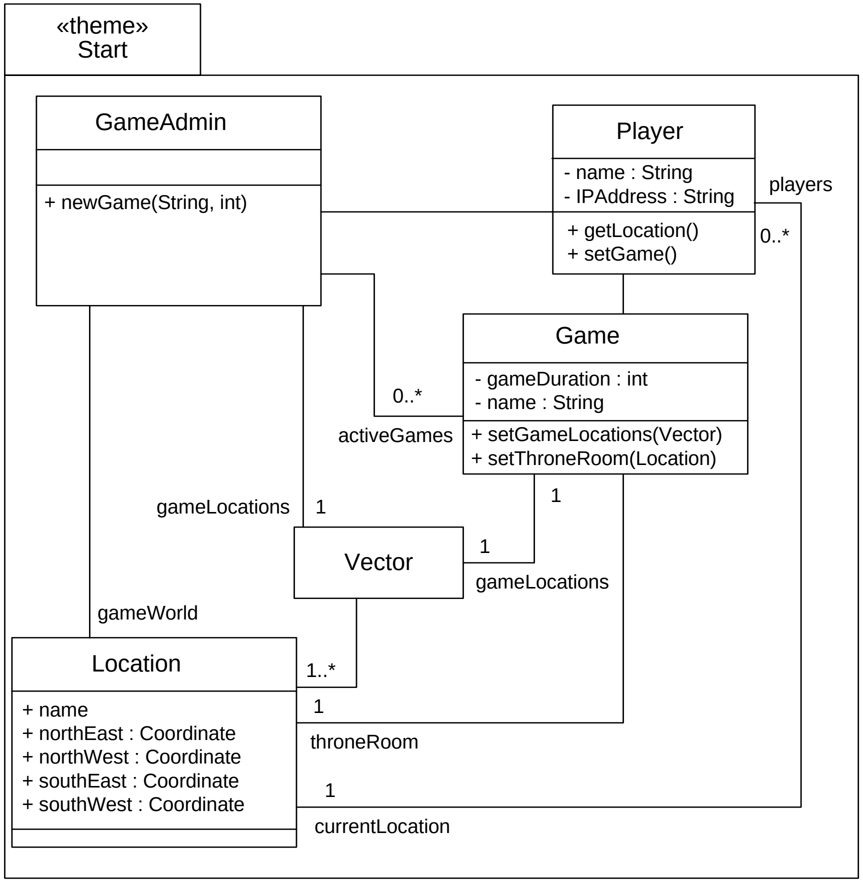
theme **design** process

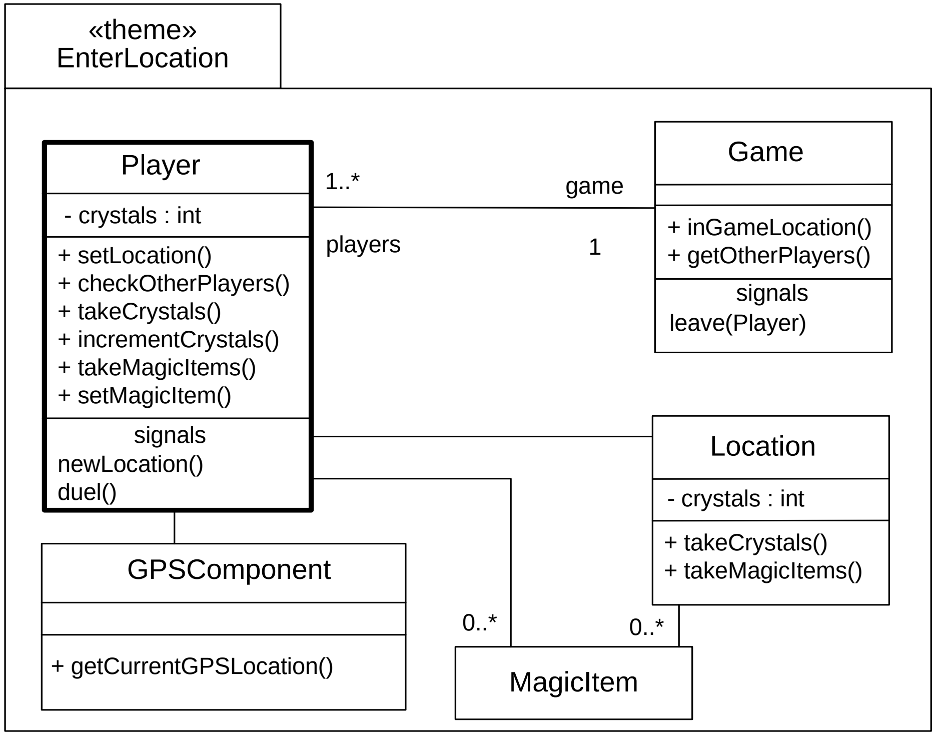


Theme/UML

model themes separately – base themes with standard UML

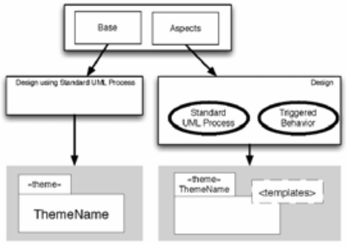

Base
Aspects
Design using Standard UML Process
Design
Standard UML Process
Triggered Behavior
«theme»
ThemeName
«theme»
ThemeName
<templates>

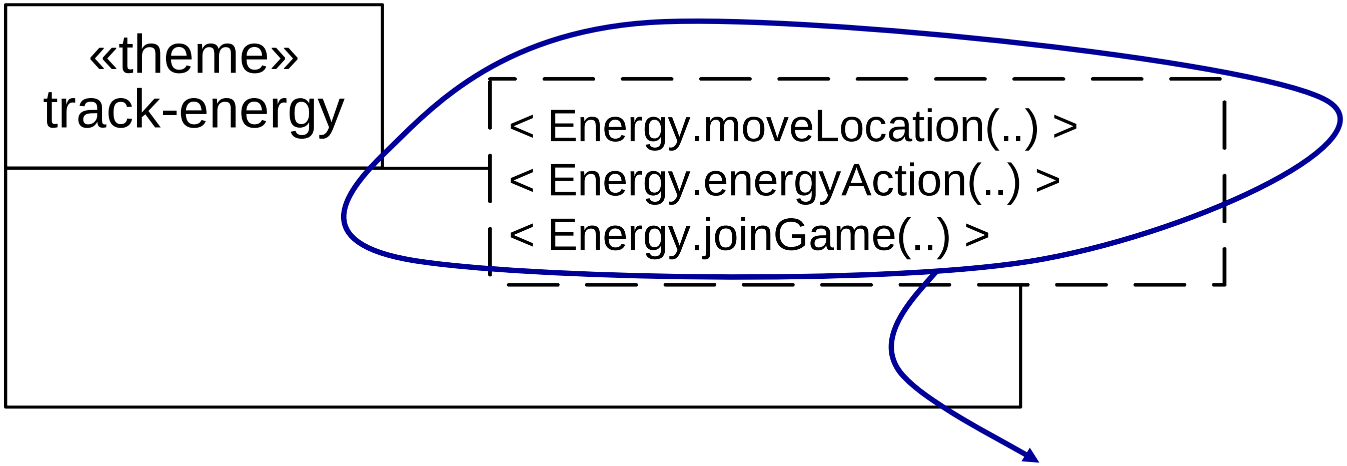

«theme»
Start
GameAdmin
+ newGame(String, int)
Player
- name : String
- IPAddress : String
+ getLocation()
+ setGame()
players
0..*
Game
- gameDuration : int
- name : String
+ setGameLocations(Vector)
+ setThroneRoom(Location)
0..*
activeGames
gameLocations
1
1
Vector
1
gameLocations
gameWorld
Location
+ name
+ northEast : Coordinate
+ northWest : Coordinate
+ southEast : Coordinate
+ southWest : Coordinate
1..*
1
throneRoom
1
currentLocation


«theme»
EnterLocation
Player
- crystals : int
+ setLocation()
+ checkOtherPlayers()
+ takeCrystals()
+ incrementCrystals()
+ takeMagicItems()
+ setMagicItem()
signals
newLocation()
duel()
1..*
players
game
1
Game
+ inGameLocation()
+ getOtherPlayers()
signals
leave(Player)
Location
- crystals : int
+ takeCrystals()
+ takeMagicItems()
GPSComponent
+ getCurrentGPSLocation()
0..*
0..*
MagicItem

Theme/UML

model themes separately – aspect themes with minor UML extensions





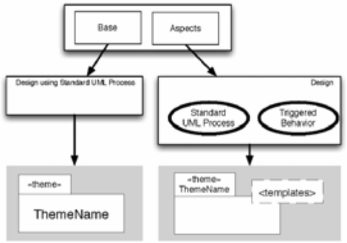
Template parameters used to reason about triggers of behaviour in aspect theme

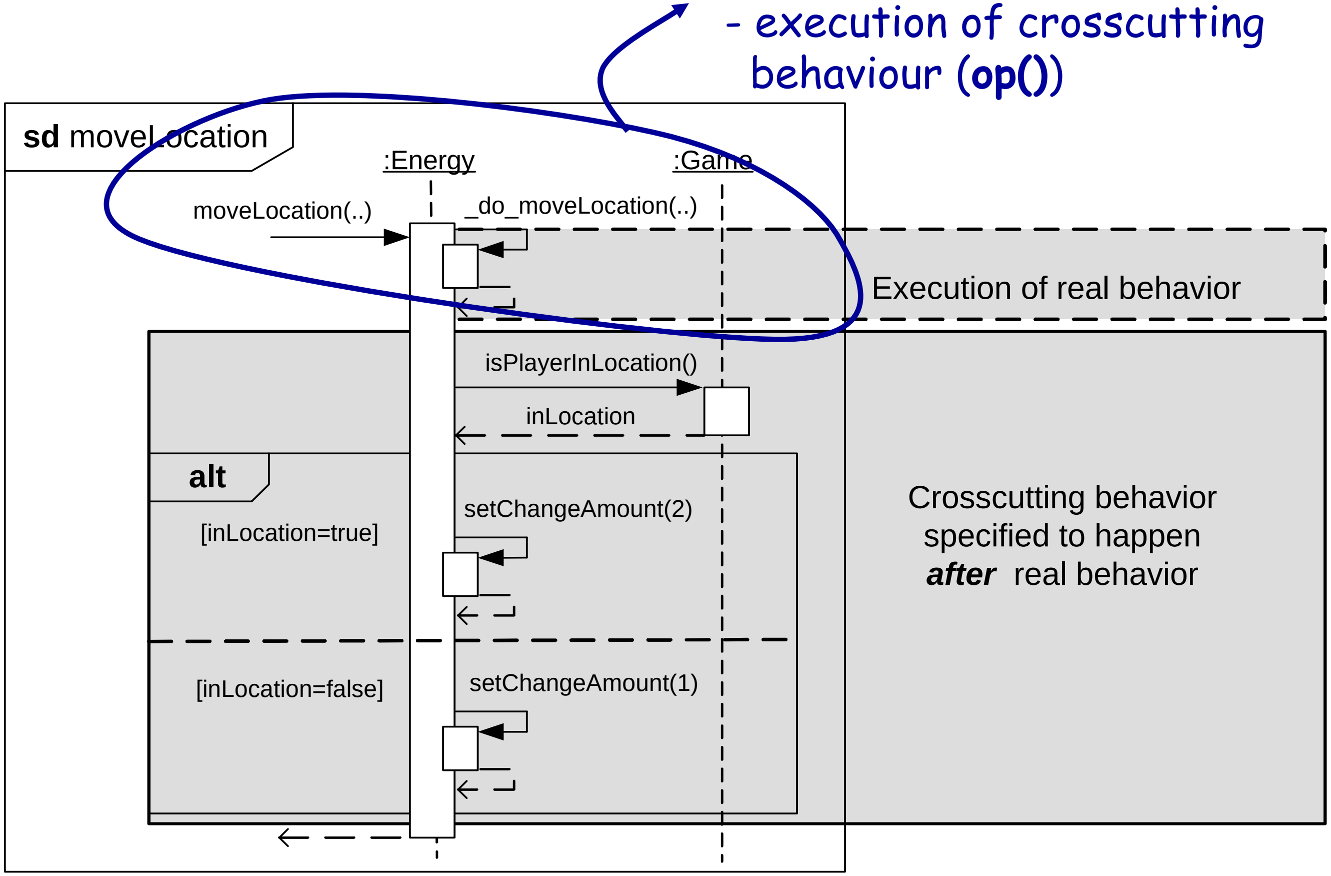
- grouped by behaviour sequence within < > brackets

Theme/UML

model themes separately – aspect themes with minor UML extensions

Distinguish between:
- execution of base operation (**_do_op()**) and
- execution of crosscutting behaviour (**op()**)



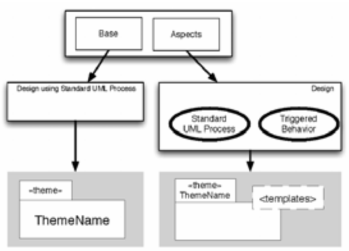


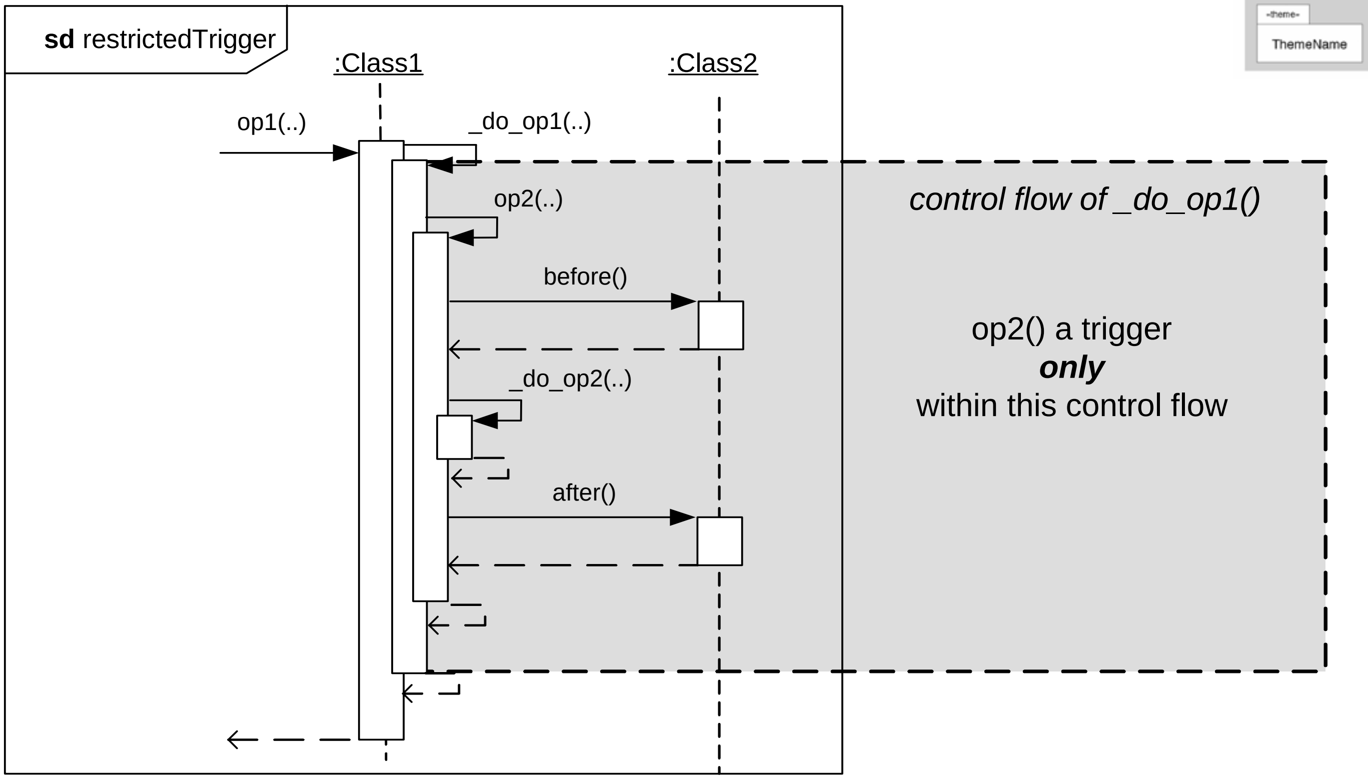
Execution of real behavior

Crosscutting behavior specified to happen ***after*** real behavior

Theme/UML

model themes separately – aspect triggers with control flow restrictions



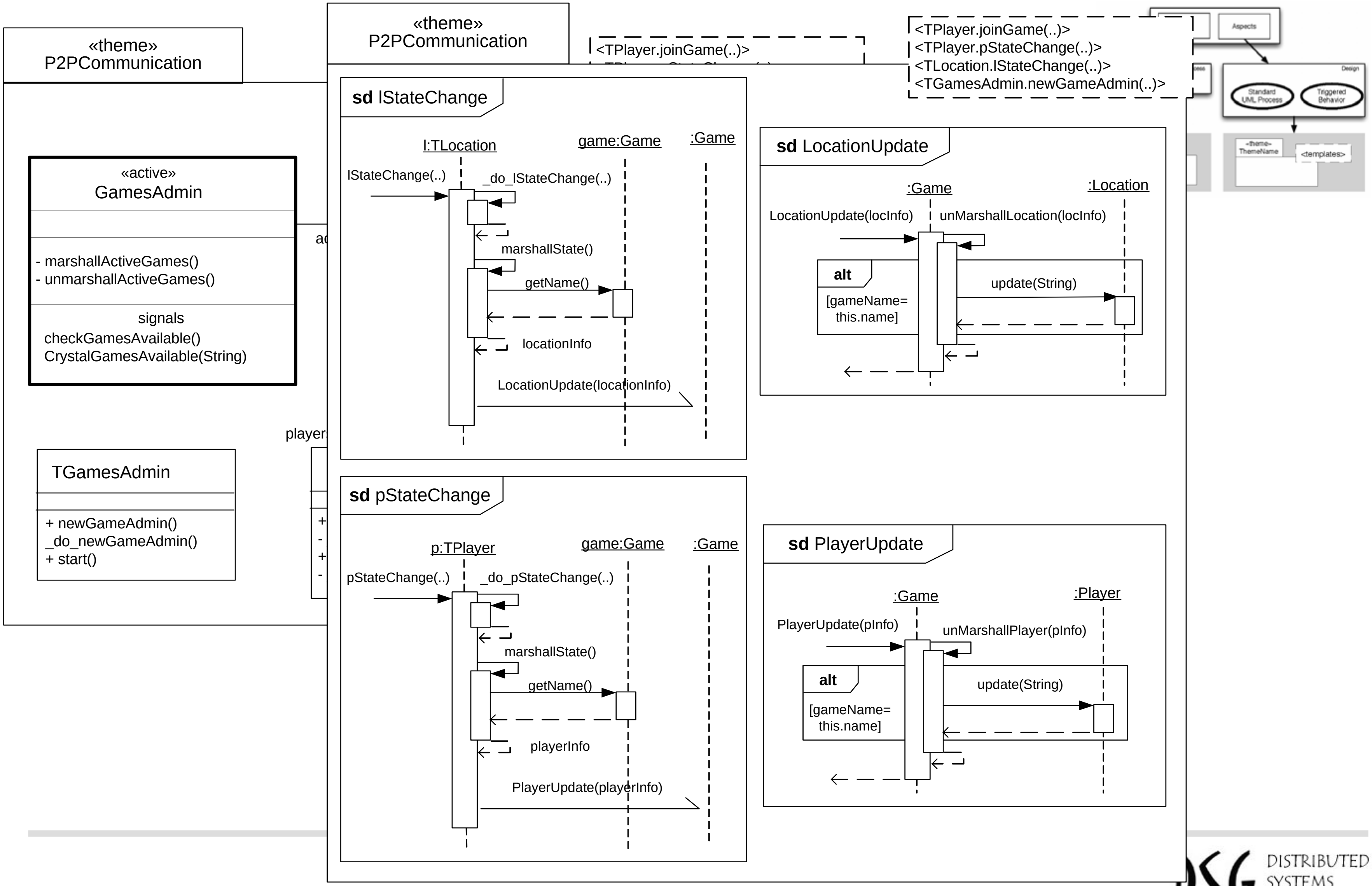


control flow of _do_op1()

op2() a trigger
only
within this control flow

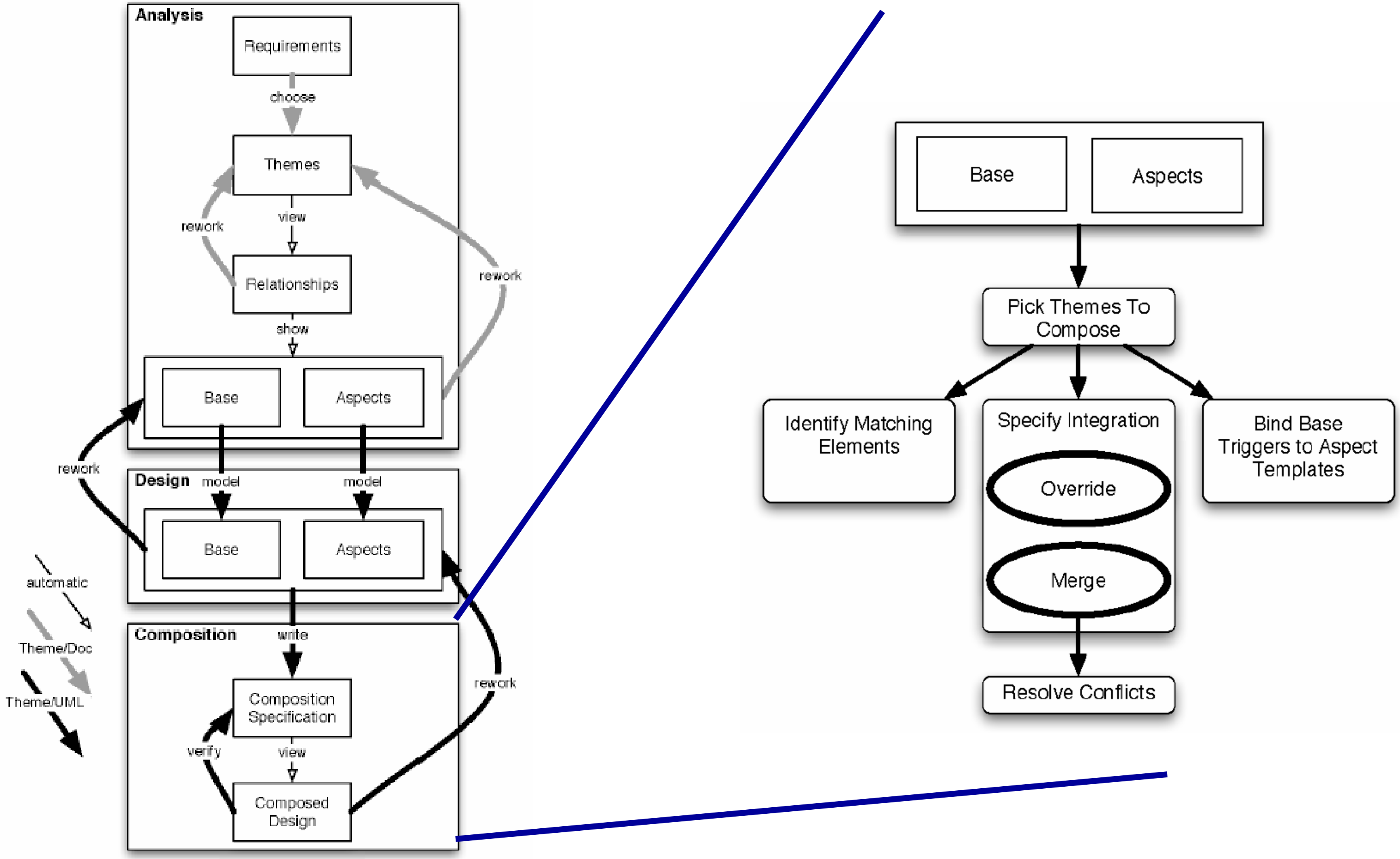
Theme/UML

model themes separately – aspect themes arising from detailed design



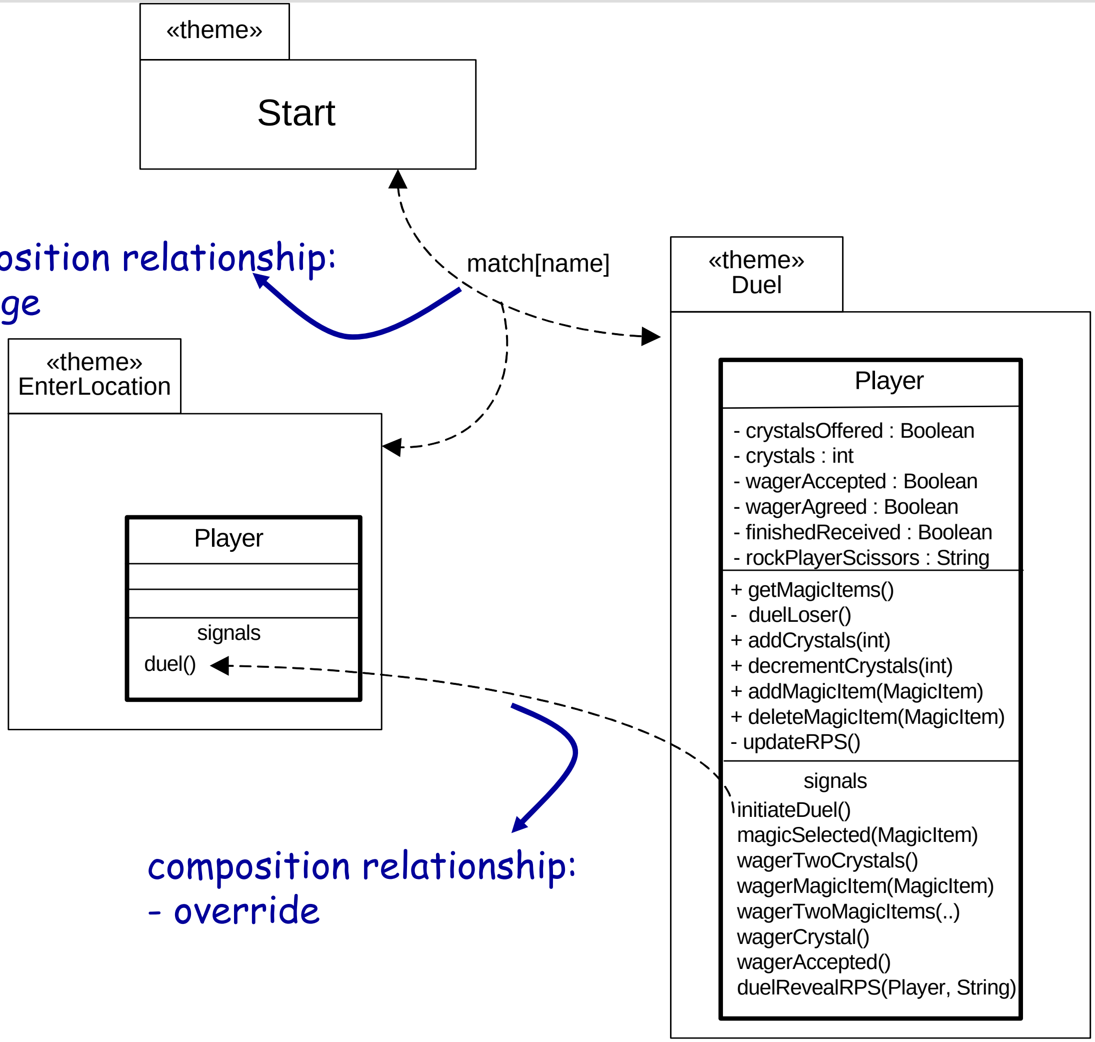
Theme/UML

theme **composition** process



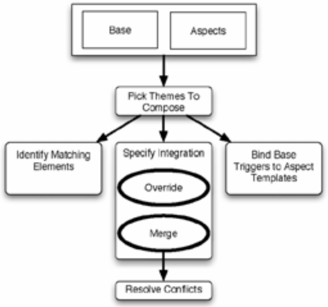
Theme/UML

compose themes – Composition Relationship and **base** themes



composition relationship:
- merge

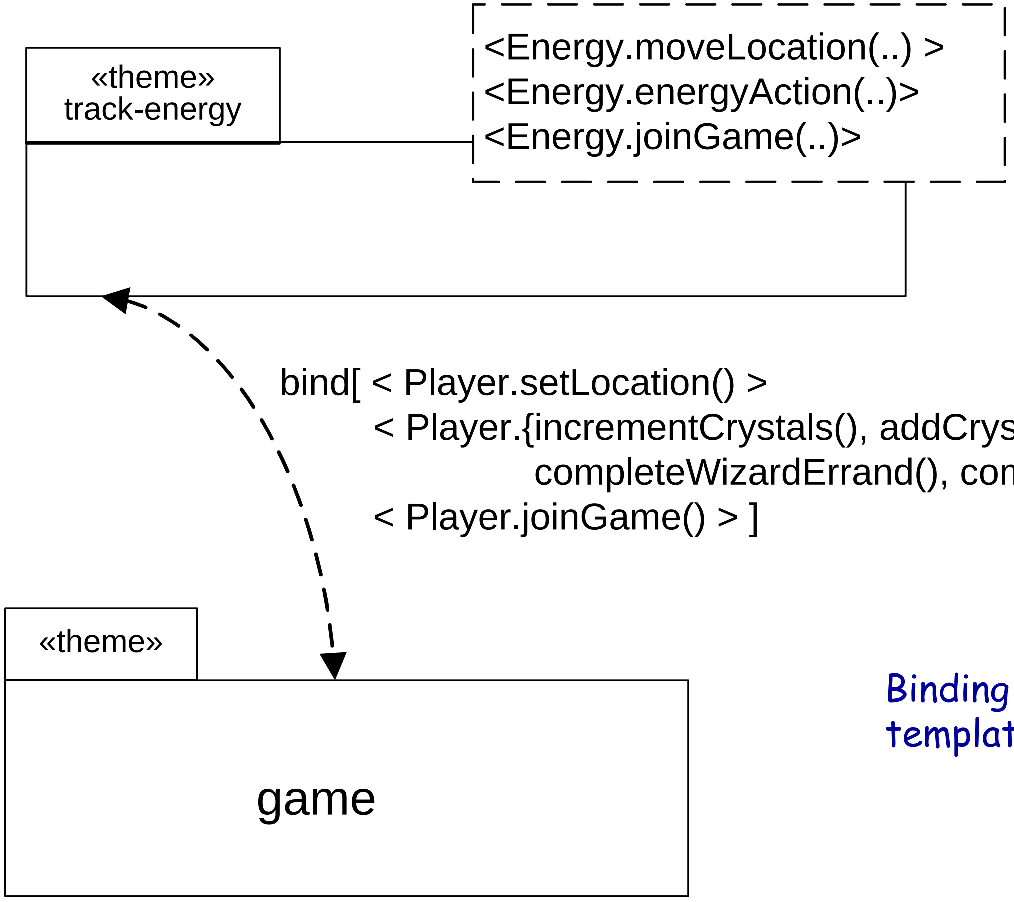
composition relationship:
- override



Composition Relationship tags:
- match[name]
- nomatch
- *for conflict resolution*
    - [prec]
    - reconcil[explicit[{..}]]
    - reconcil[default[Type {..}]]
- ordering sequence for op merge

Theme/UML

compose themes – Composition Relationship and **aspect** themes



«theme»
track-energy

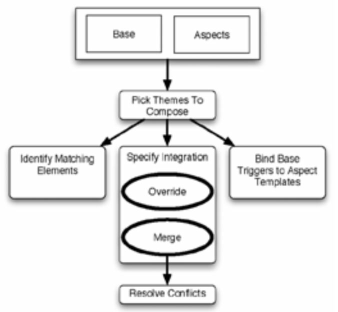
<Energy.moveLocation(..) >
<Energy.energyAction(..)>
<Energy.joinGame(..)>

bind[< Player.setLocation() >
< Player.{incrementCrystals(), addCrystals(),
completeWizardErrand(), completeWarriorTest() } >
< Player.joinGame() >]

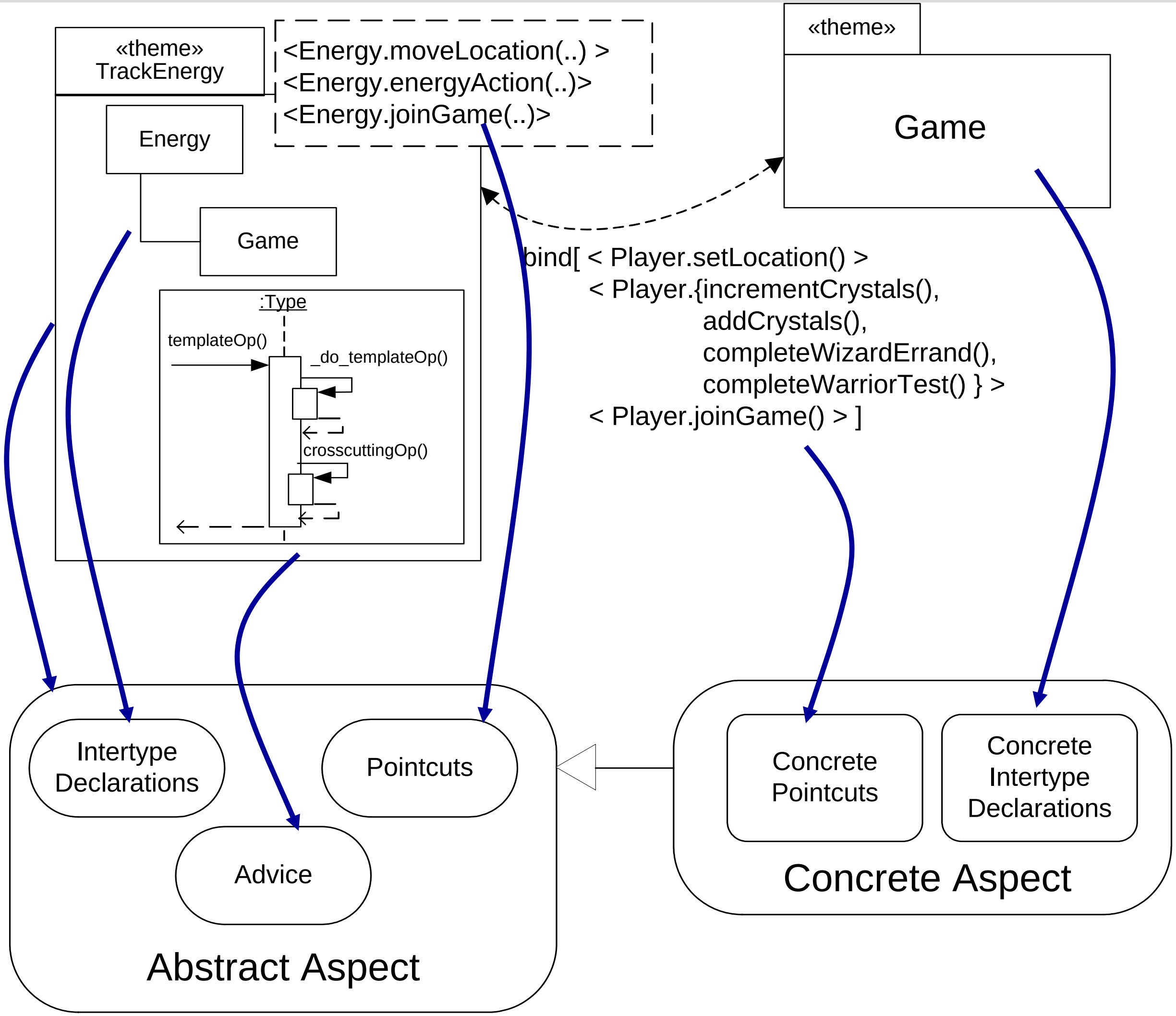
«theme»

game

Binding base design elements to template parameters.



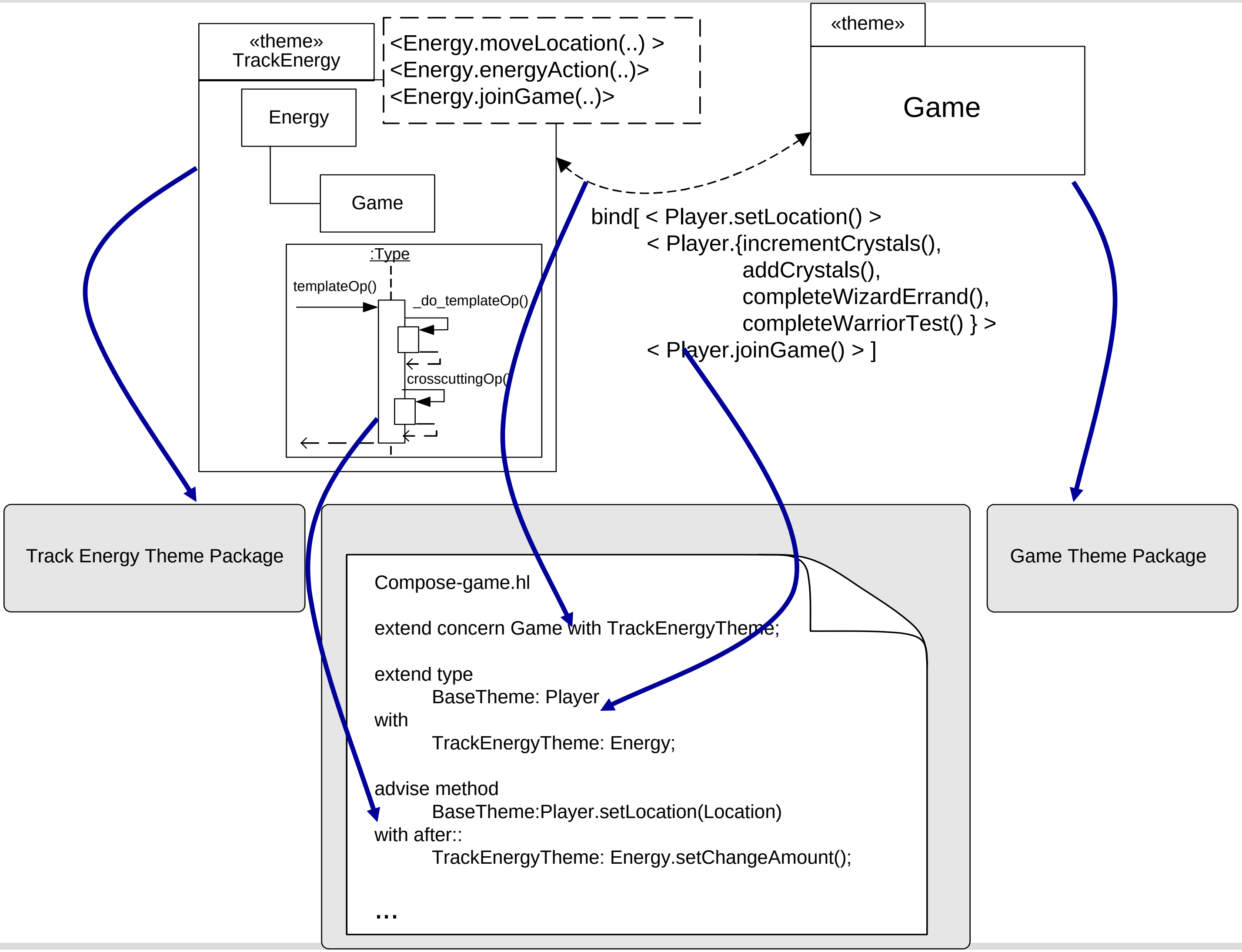
Map to AspectJ



Small flavour – concrete aspect

```
public aspect ConcreteTrackEnergy extends TrackEnergy {
   declare parents: Player implements EnergyEntityI;
   declare parents: Game implements GameI;

   pointcut moveLocation(EnergyEntityI energyEntity):
                          this(energyEntity) &&
                          execution (* Player.setLocation(..));



   pointcut energyAction(EnergyEntityI energyEntity):
                 this(energyEntity) &&
                 (execution (* Player.incrementCrystals(..)) ||
                 execution (* Player.addCrystals(..)) ||
                 execution (* Player.completeWarriorTest(..)) ||
                 execution (* Player.completeWizardErrand(..)));

   pointcut joinGame(Player player):
                 this(player) &&  execution (* Player.joinGame(..));
}
```

Map to CME - aspects



<Energy.moveLocation(..) >
<Energy.energyAction(..)>
<Energy.joinGame(..)>

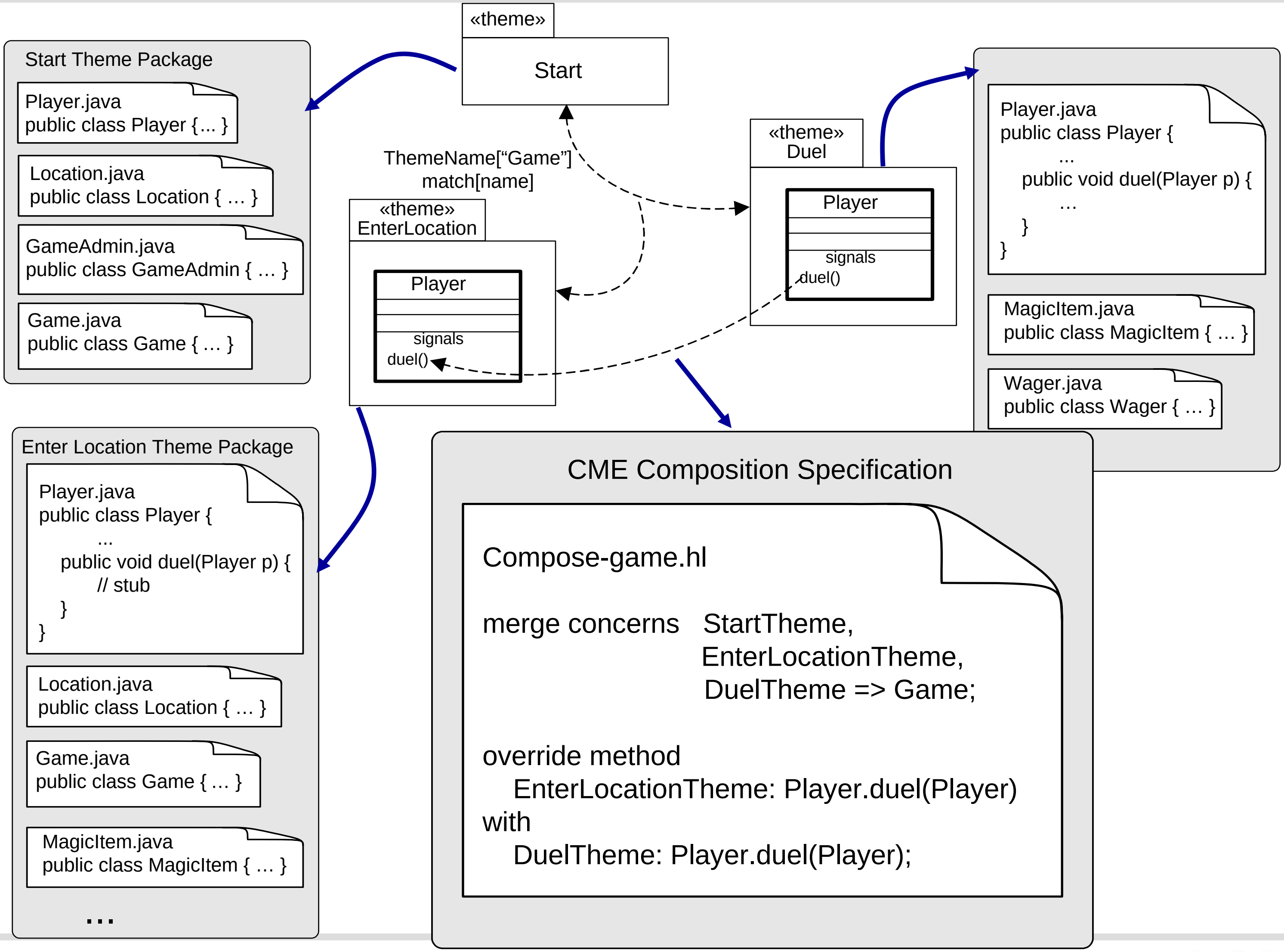
bind[< Player.setLocation() >
< Player.{incrementCrystals(),
addCrystals(),
completeWizardErrand(),
completeWarriorTest() } >
< Player.joinGame() >]

Track Energy Theme Package

Game Theme Package

```
Compose-game.hl

extend concern Game with TrackEnergyTheme;

extend type
        BaseTheme: Player
with
        TrackEnergyTheme: Energy;

advise method
        BaseTheme:Player.setLocation(Location)
with after::
        TrackEnergyTheme: Energy.setChangeAmount();

…
```

Map to CME – base



Start Theme Package

```
Player.java
public class Player {… }
```

```
Location.java
public class Location { … }
```

```
GameAdmin.java
public class GameAdmin { … }
```

```
Game.java
public class Game { … }
```

Enter Location Theme Package

```
Player.java
public class Player {
    …
    public void duel(Player p) {
        // stub
    }
}
```

```
Location.java
public class Location { … }
```

```
Game.java
public class Game { … }
```

```
MagicItem.java
public class MagicItem { … }
```

…

```
Player.java
public class Player {
    …
    public void duel(Player p) {
        …
    }
}
```

```
MagicItem.java
public class MagicItem { … }
```

```
Wager.java
public class Wager { … }
```

CME Composition Specification

```
Compose-game.hl

merge concerns   StartTheme,
                 EnterLocationTheme,
                 DuelTheme => Game;

override method
   EnterLocationTheme: Player.duel(Player)
with
   DuelTheme: Player.duel(Player);
```

Theme publications

Siobhán Clarke, Robert J. Walker. *"Generic Aspect-Oriented Design with Theme/UML“* In Aspect-Oriented Software Development, Addison-Wesley, 2005

Elisa L.A. Baniassad, Siobhán Clarke. *"Theme: An Approach for Aspect-Oriented Analysis and Design"* In Proceedings of the 26th International Conference on Software Engineering (ICSE), Edinburgh, 2004.

Siobhán Clarke. *"Extending standard UML with model composition semantics"* in Science of Computer Programming, Volume 44, Issue 1, pp. 71-100. Elsevier Science, July 2002.

Siobhán Clarke, Robert J. Walker. *"Towards a Standard Design Language for AOSD"* In Proceedings of the 1st International Conference on Aspect-Oriented Software Development (AOSD), Enschede, The Netherlands, April 2002.

Siobhán Clarke, Robert J. Walker. *"Composition Patterns: An Approach to Designing Reusable Aspects"* In Proceedings of the 23rd International Conference on Software Engineering (ICSE), Toronto, Canada, May 2001.

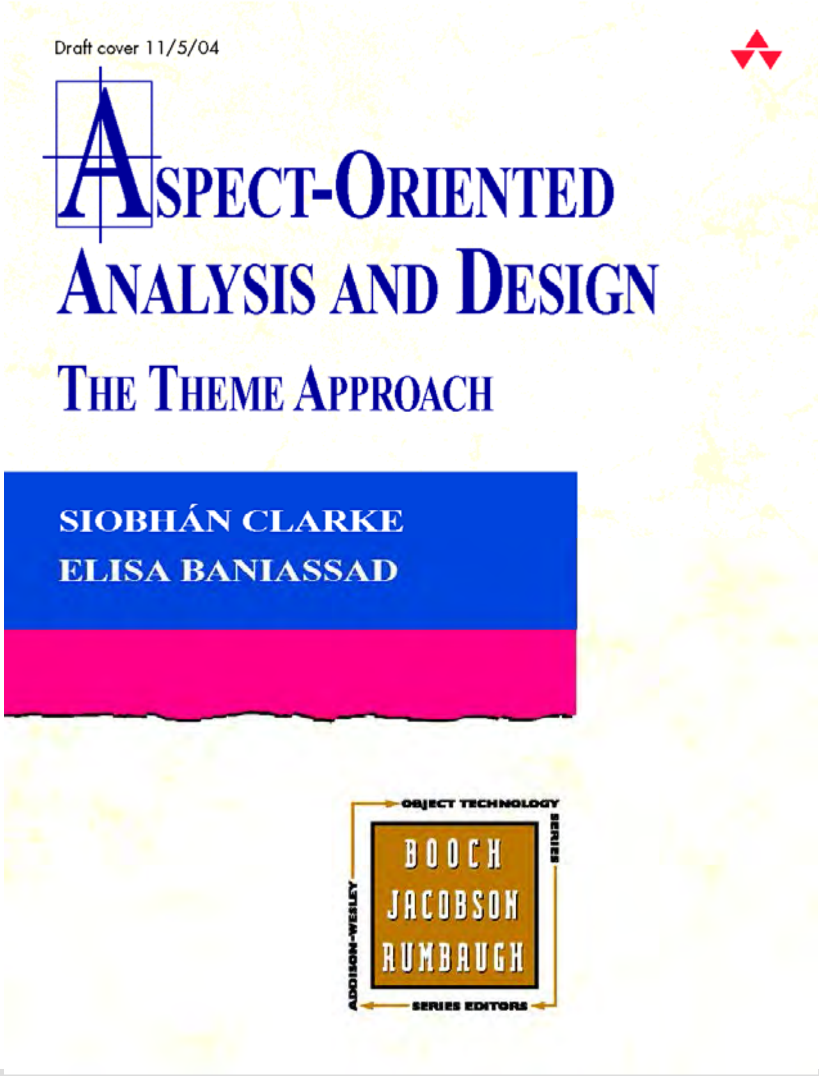
Siobhán Clarke. "Composition of Object-Oriented Software Design Models“ Ph.D. Thesis, January 2001, Dublin City University.

Siobhán Clarke. *"Composing Design Models: An extension to the UML"* In Proceedings of the Third International Conference on the Unified Modeling Language (UML), UK, 2000. Volume 1939, LNCS

Siobhán Clarke, William Harrison, Harold Ossher, Peri Tarr. *"Subject-Oriented Design: Towards Improved Alignment of Requirements, Design and Code"* In Proceedings of Object-Oriented Programming, Systems, Languages and Applications (OOPSLA) Denver, Colorado U.S., 1999.

DISTRIBUTED SYSTEMS GROUP

… and a plug!



Draft cover 11/5/04

ASPECT-ORIENTED ANALYSIS AND DESIGN

THE THEME APPROACH

SIOBHÁN CLARKE
ELISA BANIASSAD

OBJECT TECHNOLOGY SERIES

BOOCH
JACOBSON
RUMBAUGH

SERIES EDITORS

ADDISON-WESLEY

Questions...