Five Things I Wish I'd Known About Use Cases Before I Abused Them

Phil Robinson
Lonsdale Systems
lonsdale@iinet.net.au
www.iinet.net.au/~lonsdale/

Topics

- Brief introduction to Use Cases
- The five things:
  1. Use cases describe tool use
  2. Use cases describe actor's goals
  3. Actors define the system boundary
  4. Use cases do not have conditional logic but do have data
  5. Use cases are not visual models

Brief Introduction to Use Cases

- Use cases were 'invented' by Ivar Jacobson
- An integral part of Jacobson's OOSE method
- Core element of the UML
- Underpinning of many 'iterative, incremental' methods
- Rather over-hyped and trendy!

A Use Case

A Use Case Narrative

When a front desk clerk checks in a new guest, they use the application to search for the guest's prior reservation details.

When the reservation details are found, the clerk creates a check-in entry. The application copies the guest's personal details from the reservation entry to the check-in entry.

To complete the check-in, the clerk enters the guest's credit card details and allocates the guest a room.

A Formal Use Case Description

MAIN FLOW

1. The front desk clerk enters the guest's name
2. The application displays prior reservations which match the name
3. The front desk clerk enters the guest's check-in date and time
4. The application copies the guest's personal details from the reservation entry to the check-in entry
5. The front desk clerk enters the guest's credit card details
6. The application allocates the guest a room

ALTERNATE FLOWS

a) At step 2, the guest does not have a prior reservation
   1. The front desk clerk enters the guest's personal details

b) At step 6, there is no room available which matches the guest's preferences
   1. The application displays all available rooms
   2. The front desk clerk allocates the guest a room

Why Use Cases Are Easy to Understand But Hard to Write?

- Short answer
  - Users read them
  - Developers write them!
- Long answer
  - Developers are familiar with 'functions' that describe what an application does
  - Use cases describe how a user actually 'uses' an application
  - The difference between 'usage' and 'function' is not immediately obvious
  - Descriptions of the use case technique
    - No underlying conceptual model
    - Emphasis on visual models
    - Lack of in-depth descriptions
    - Lots of trivial examples
    - Use cases can be 'all things to all people'

What is an 'Abuse Case'

- An 'abuse case' is something pretending to be a use case just to be 'trendy'
- Popular culprits are
  - Functions
  - Modules
  - Screens
  - Windows
  - Structured English
  - Pseudo code

1. Use Cases Describe Tool Use

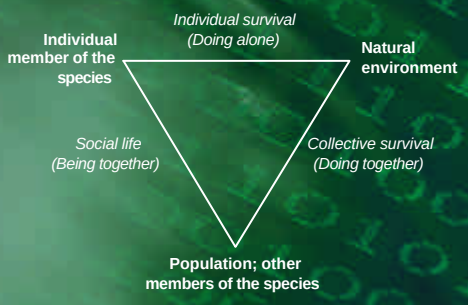
Activity Theory

- Software engineering already has odd things like:
  - the 'three amigos'
  - 'entity beans'
  - the 'gang of four'
  - 'lazy materialisation'
- So why not introduce some obscure Soviet psychology?

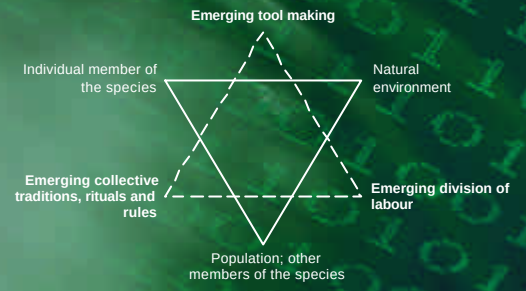
Activity Theory

- Developed by Russian psychologists Vygotsky, Leontiev and others during the early 1900s
- Not much interest in the west (except Scandinavia)
- Provides a very relevant model of human activity

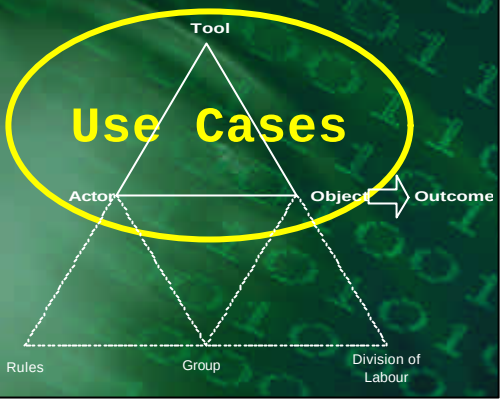
Animal Activity



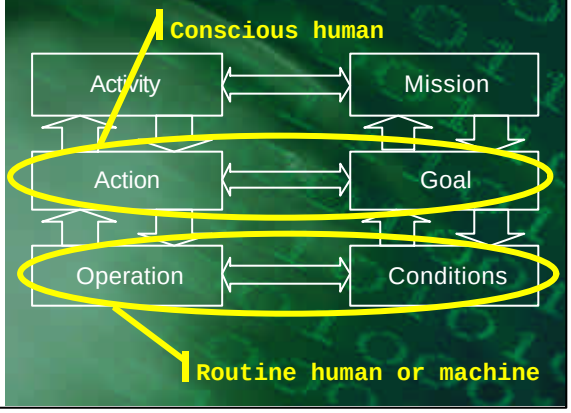
Transition From Animal to Human Activity



Human Activity



The Hierarchical Structure of Activities



Activity Theory Summary

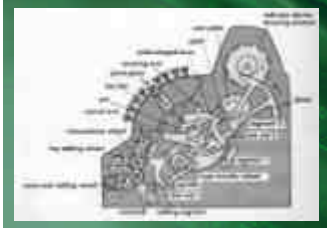
- Tools 'mediate' activity
- Tool use changes an activity
- Actions achieve goals
- Operations can be automated
- Use cases describe a tool's role in an activity

Use Cases

- The check-out clerk records the amount of the sale item.
- The check-out clerk calculates the total of the sale items. Access is provided to a secure storage area.

No details of the user interface

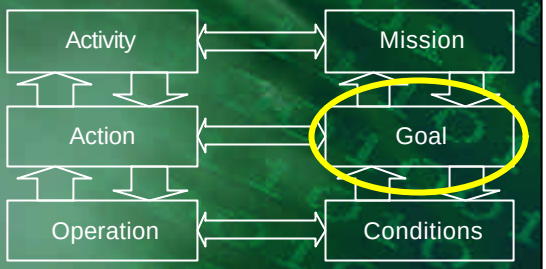
An Abuse Case



When a key is depressed, the electric motor is started which drives the camshaft anti-clockwise. This causes the block, which is rotatably mounted on the shaft, to thrust against the sickle-shaped lever, so that this lever rotates clockwise about its pivot. This causes the swivel arm to swivel until it is stopped by the key bar. As a result of this, the bottom pivot of the sickle-shaped lever becomes a fixed point. The sickle-shaped lever is thus compelled to rotate in the clockwise direction. In doing this it carries the counting arm along with it until this arm, too, is arrested by the key bar. Acting through the link rod 1, the counting arm moves the adding segment into position. The pin and segment 1 rotate the type transfer wheel and thereby set the printing wheels in position. The link rod 2, attached to the segment 1, works the indicator roll through the agency of the segment 2. The adding segment engages with the intermediate cog-wheel of the adding mechanism. Then the sickle lever differential opens, and the counting arm and the swivel arm return to their initial positions. The amount is transferred both to the top adding wheel and to the bottom sectorised adding wheel. On release of the adding segment, the camshaft rotates and performs the transfer of the tens.

2. Use Cases Describe Actor's Goals

Goals Are Related to Conscious Actions



Actor Goals

- Record the amount of a sale item
- Calculate the sale total
- Keep the cash secure

Goals and Use Cases

- An actor perform a sequence of operations to achieve a goal
- Some operations may be automated
- Use cases describe the requirements of the automated operations
- Use cases must be relevant to the actor's goal

Goals and Scenarios

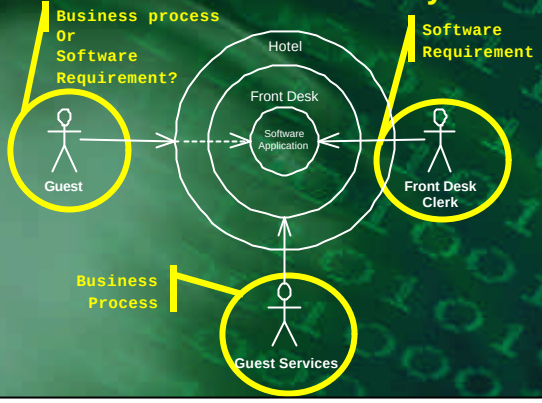
- Goals may succeed or fail
- Goal
  - Record returned rental video details
- Scenarios
  - No worries!
  - The video is overdue
  - The video is damaged
  - The video was previously recorded as lost
  - The video is already recorded as returned

3. Actors Define the System Boundary

Who is the Actor?

- Store assistant
  - Uses the cash register to record sales
- Customer
  - Uses the store to purchase goods
- Both have a 'Sale' use case

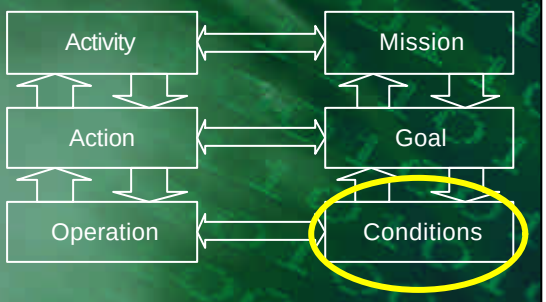
'Check-In' Use Case Where is the Boundary?



Software Requirements Two Crude But Effective Tests!

4. Use Cases Do Not Have Conditional Logic But Do Have Data

Conditions Are Related to Routine Operations



Structured English

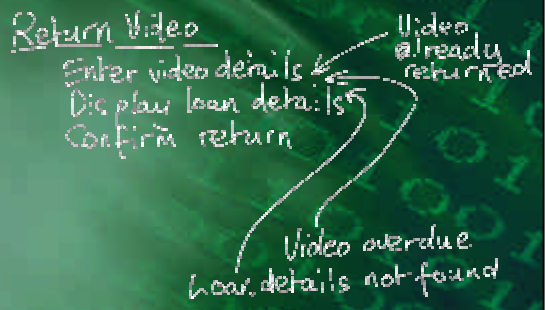
```
The store staff enter returned video details
If the rental details cannot be found
   If the video previously recorded as lost
      do something…
   Endif
   If the video is already recorded as returned
      do something…
   Endif
Else
   If the video is overdue
      do something…
   Else
      do something
   Endif
Endif
```

Is this correct?

Steps and Conditions

1. The store staff enter returned video details
2. do something…
   - 1.1 Overdue
     do something…
   - 1.2 Rental details cannot be found
     - 1.2.1 Previously recorded as lost
       do something…
     - 1.2.2 Already recorded as returned
       do something…

Use Case as Lists



Use Cases as 'Mind Maps'

Use Case Data

- The check-out clerk records the amount of the sale item.
- The check-out clerk calculates the total of the sale items.
- Access is provided to a secure storage area.

BUT what if there are 50 data elements?

Use Case Data: Option 1

- Include the data elements as a comma separated list in the body of the use case

The store staff enter the member's name, address, telephone number, e-mail address and date of birth.

Use Case Data: Option 2

- Describe the data elements as a bullet list under the use case step

The store staff enter the member's:

- Name
- Address
- Telephone number
- E-mail address
- Date of birth

Use Case Data: Option 3

- Give the data elements a collective name and describe them in a separate set of data requirements

Use case

The store staff enter the memberDetails

Data Requirements

memberDetails = name + address + telephoneNumber + emailAddress + dateOfBirth

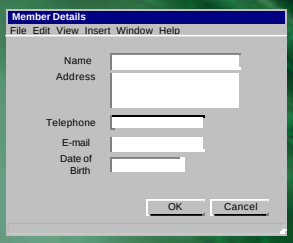
Use Case Data: Option 4

- Describe changes of state to objects in the domain model
- Derive the use data elements

The store staff enter the memberDetails.

The system updates the member's name, address, telephoneNumber, emailAddress and dateOfBirth

Use Case Data: Option 5



- Capture the required data elements in a prototype

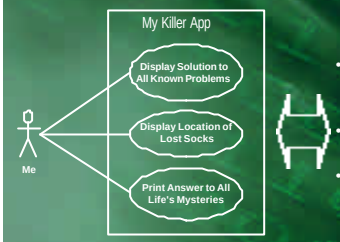
5. Use Cases Are Not Visual Models

A UML Use Case Diagram

Opportunities for 'Analysis Paralysis'

- 'Include' (formerly 'uses')
- 'Extends' and extension points
- 'Generalise'
- Emphasis on visual layout
- Wall-sized diagrams

Spot the Difference!



- Display Solution to All Known Problems
- Display Location of Lost Socks
- Print Answer to all Life's Mysteries

Summary

1. Use cases describe tool use
2. Use cases describe actor's goals
3. Actors define the system boundary
4. Use cases do not have conditional logic but do have data
5. Use cases are not visual models

Questions?

Phil Robinson
Lonsdale Systems
lonsdale@iinet.net.au
www.iinet.net.au/~lonsdale/