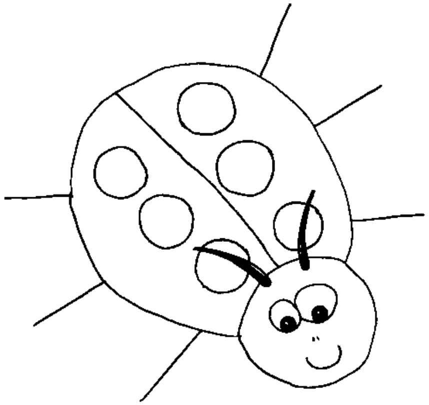


Visual Models in Test Design

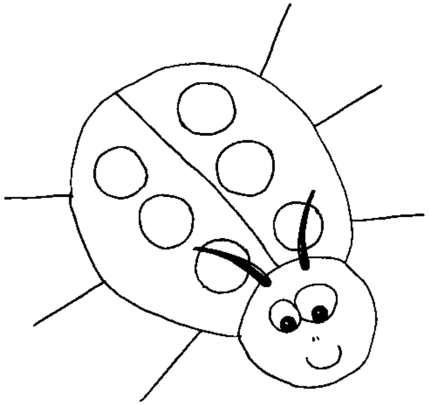
Robert Sabourin

President

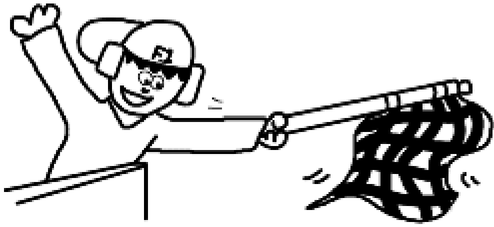
AmiBug.Com, Inc.

Montreal, Canada

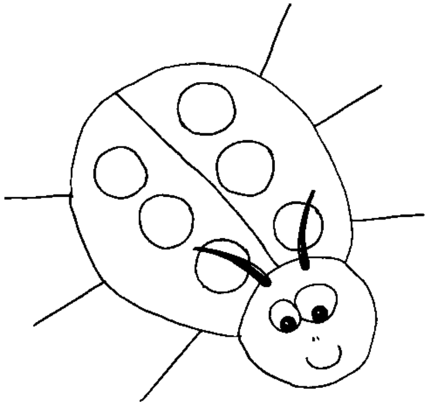
rsabourin@amibug.com



Visual Models in Test Design



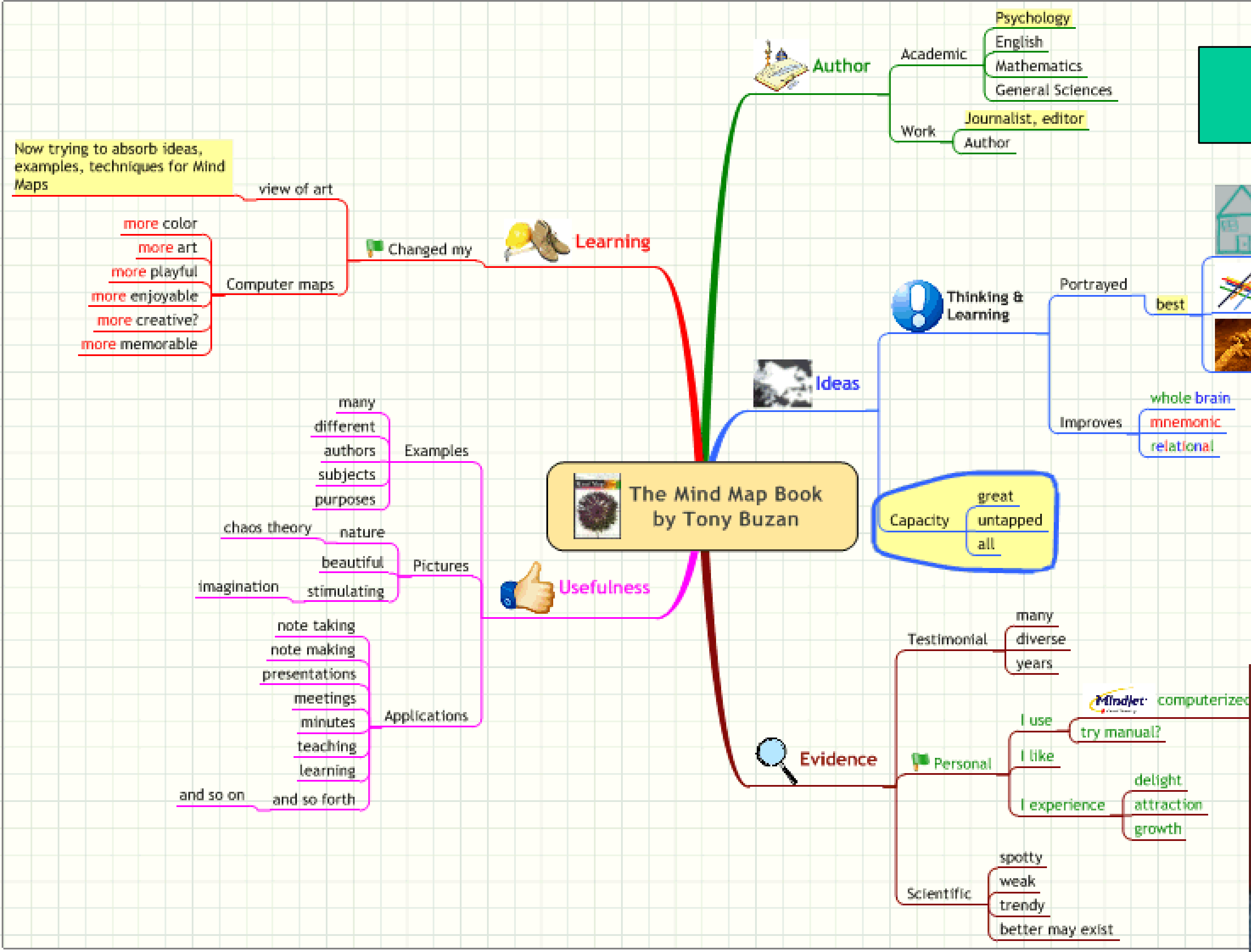
- Robert Sabourin , *Software Evangelist*
- President
- AmiBug.Com Inc.
- Montreal, Quebec, Canada
- rsabourin@amibug.com
- www.amibug.com



Edsger W. Dijkstra

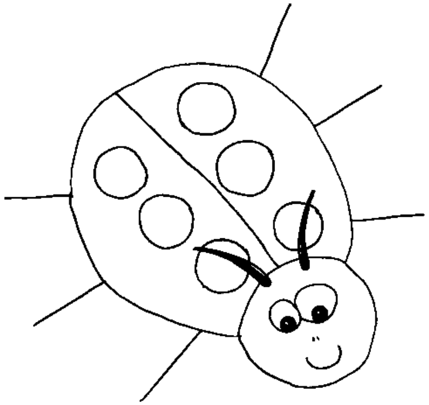
- “Program testing can be used to show the presence of bugs, but never to show their absence”

Mind Maps

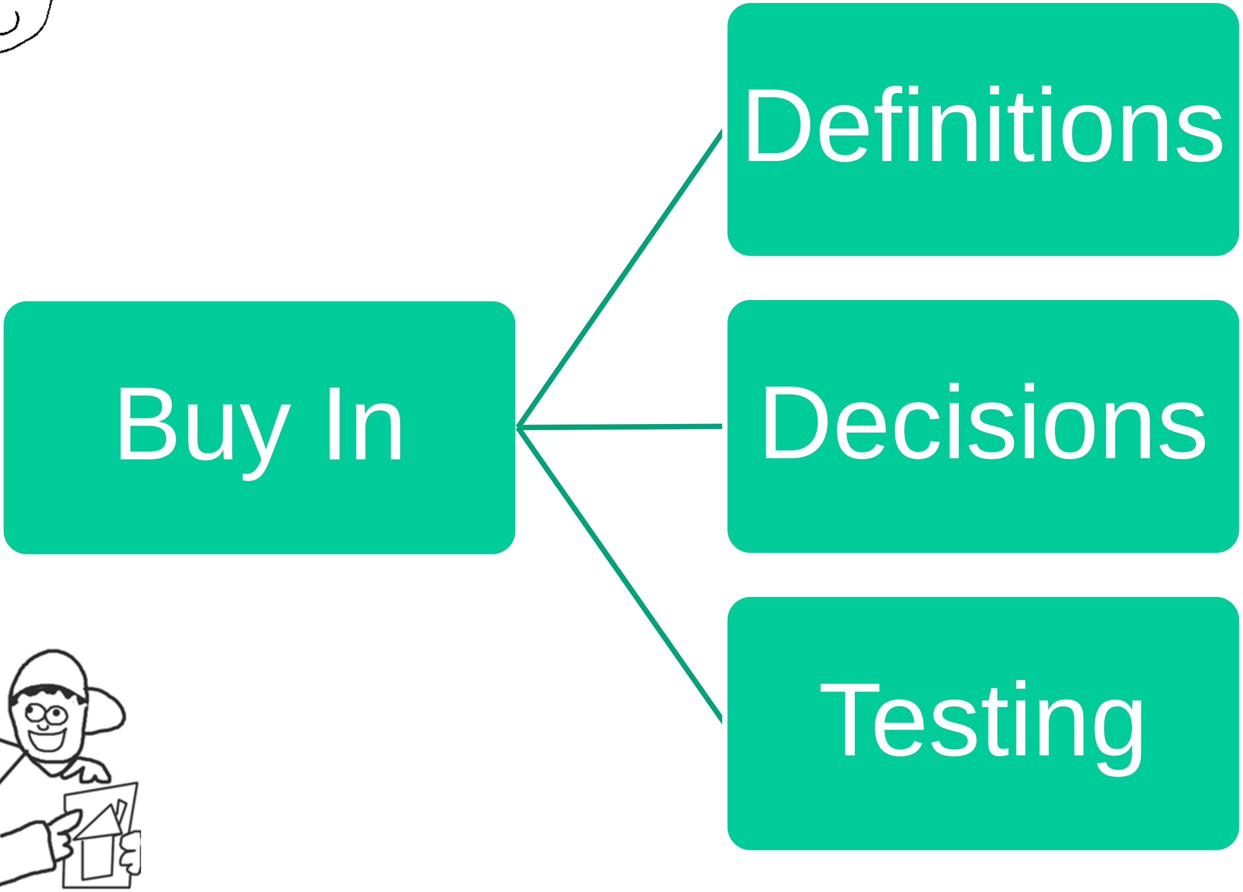


- The Mind Map Book by Tony Buzan
  - Author
    - Academic
      - Psychology
      - English
      - Mathematics
      - General Sciences
    - Work
      - Journalist, editor
      - Author
  - Learning
    - Changed my
      - view of art
        - Now trying to absorb ideas, examples, techniques for Mind Maps
      - Computer maps
        - more color
        - more art
        - more playful
        - more enjoyable
        - more creative?
        - more memorable
  - Ideas
    - Thinking & Learning
      - Portrayed
        - best
          - drawing
          - color
          - spatial
      - Improves
        - whole brain
        - mnemonic
        - relational
    - Capacity
      - great
      - untapped
      - all
  - Usefulness
    - Examples
      - many
      - different
      - authors
      - subjects
      - purposes
    - Pictures
      - nature
        - chaos theory
      - beautiful
      - stimulating
        - imagination
    - Applications
      - note taking
      - note making
      - presentations
      - meetings
      - minutes
      - teaching
      - learning
      - and so forth
        - and so on
  - Evidence
    - Testimonial
      - many
      - diverse
      - years
    - Personal
      - I use
        - computerized
        - try manual?
      - I like
      - I experience
        - delight
        - attraction
        - growth
    - Scientific
      - spotty
      - weak
      - trendy
      - better may exist

Tony Buzan



Before Test Design

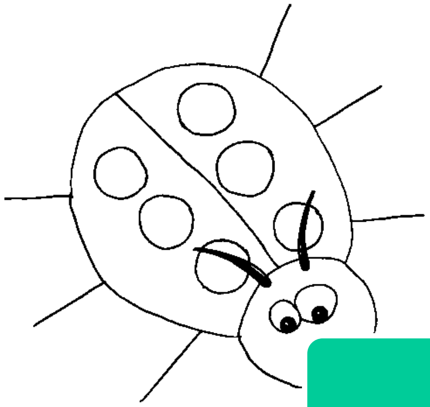


Buy In

Definitions

Decisions

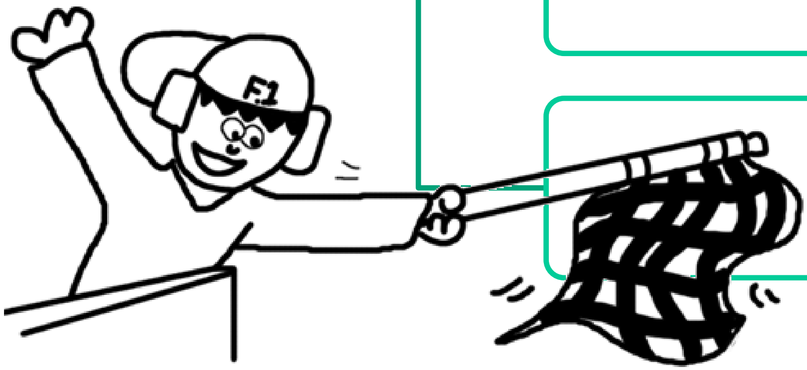
Testing

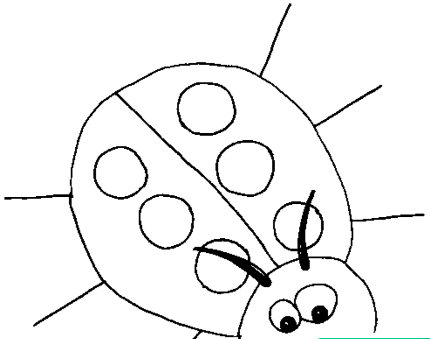


Before Test Design

Definitions

- Quality
- Bugs
- Done

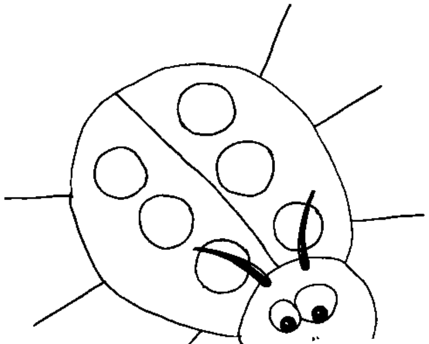




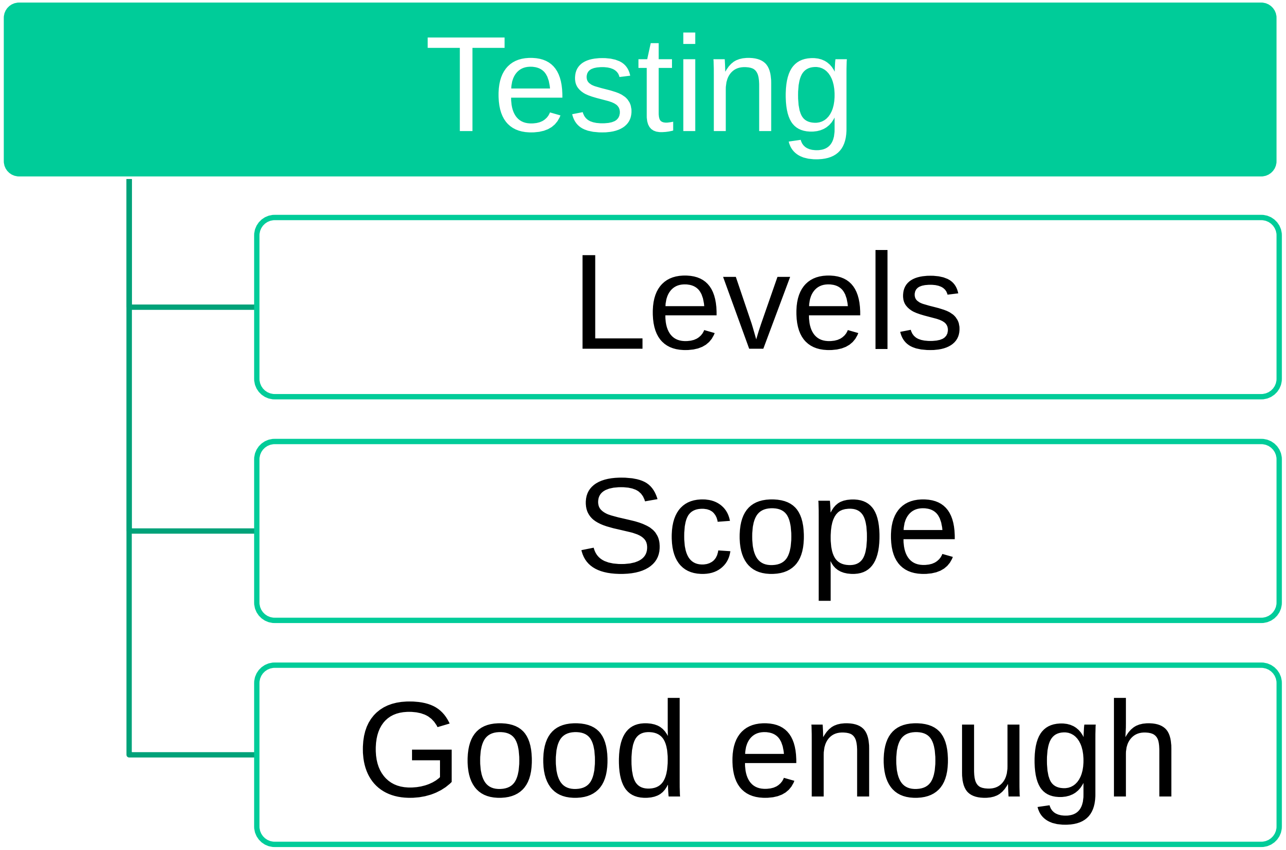
Before Test Design

Decisions

- Requirements
- Bugs
- Tests



Before Test Design

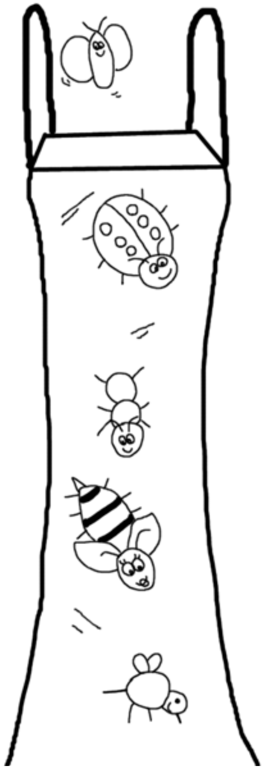


Testing

Levels

Scope

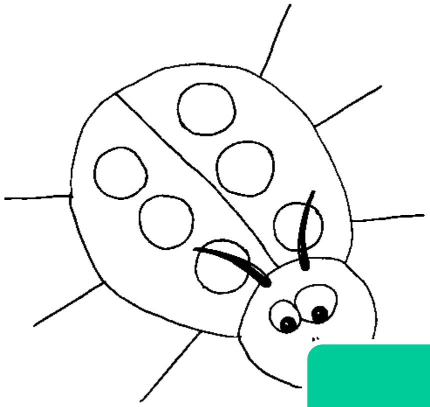
Good enough



Before Test Design

Test Objective

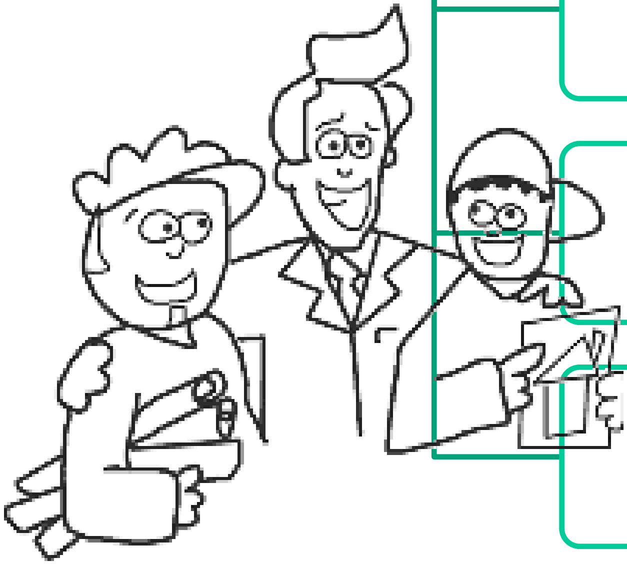
- Identified
- Important
- Knowledge



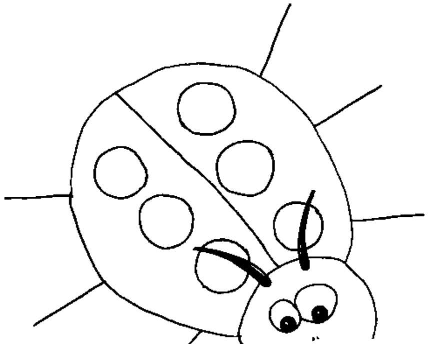
Value of Test Design

Visual design

- Blueprint
- Focuses work
- Reusable tool

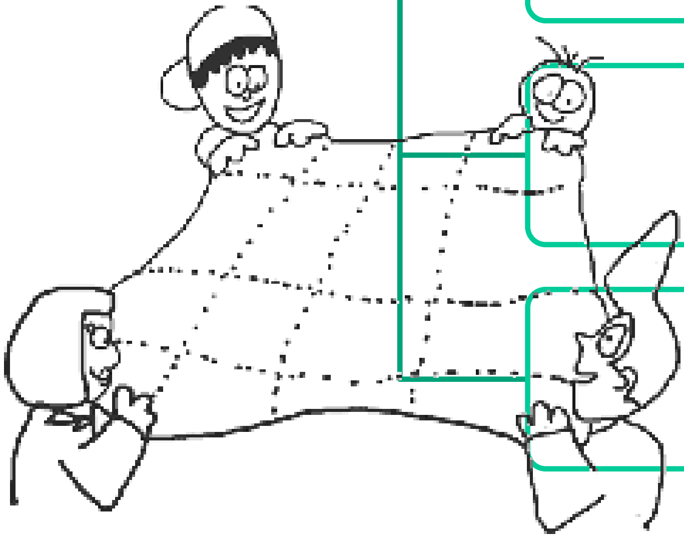


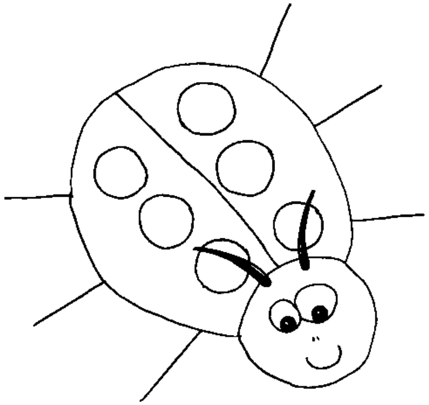
Value of Test Design



Collaboration

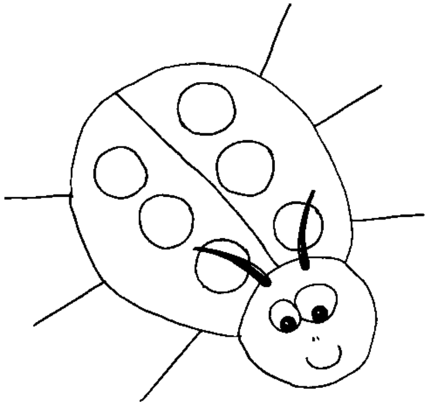
- Customers
- Developers
- Testers





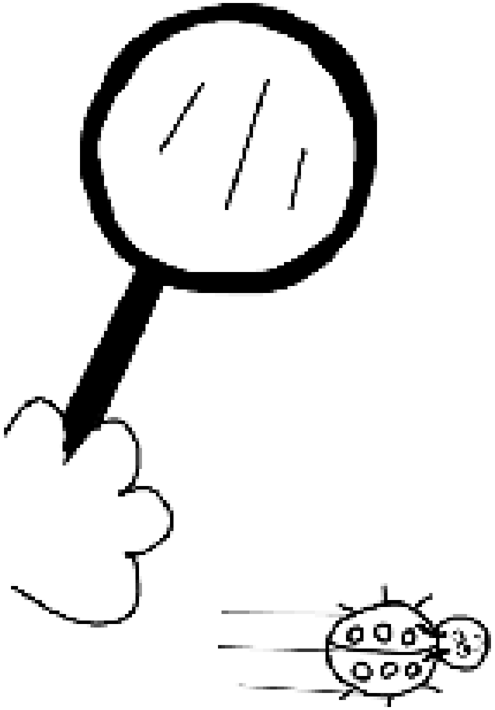
Test Design

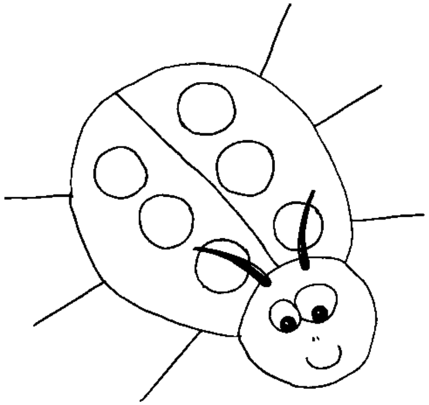
Variable identification



Identify Variables

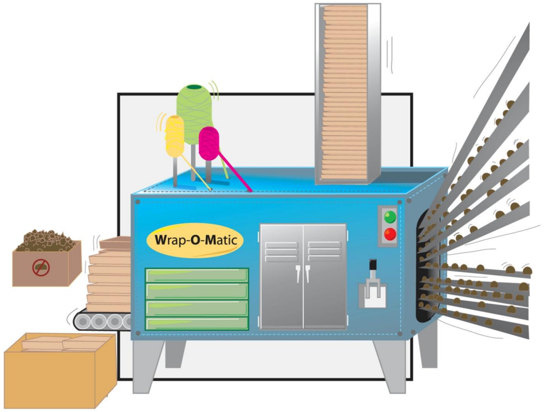
- What is a variable?
  - To *VARY* is to *CHANGE*
    - A variable is something which can change
  - Software behavior depends on the *VALUES* of many *VARIABLES*
    - Anything which influences the behavior of software could be a variable





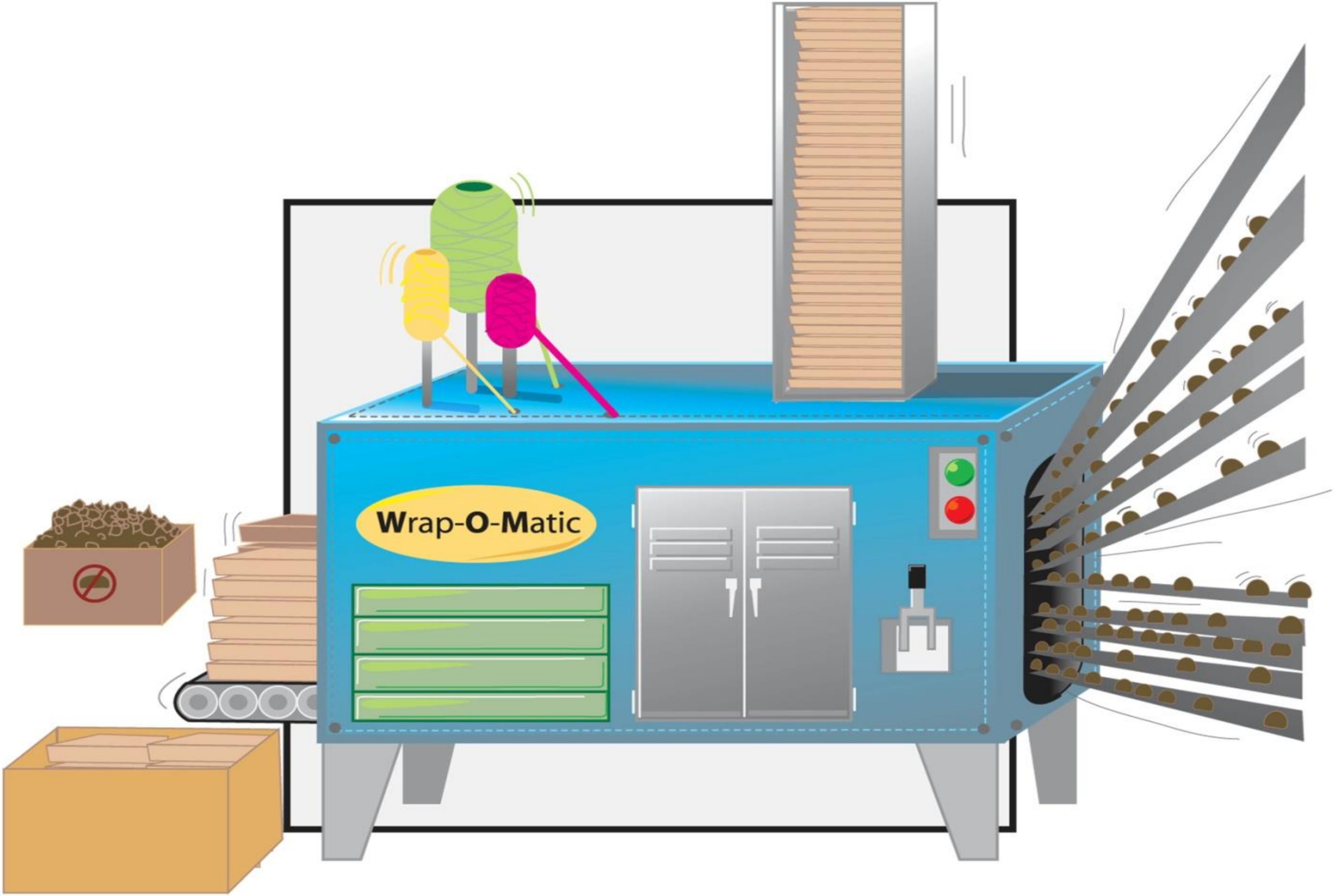
Identify Variables

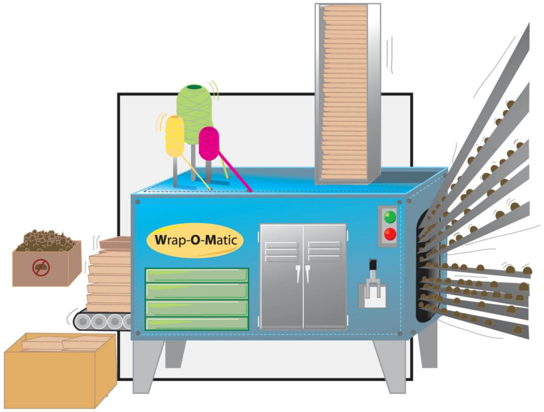
- Review test basis
- Identify variables
  - Influencers
  - Outcomes
- Potential Sources
  - Conditions
  - Environment
  - Rules
  - Constraints
  - Actions
  - States

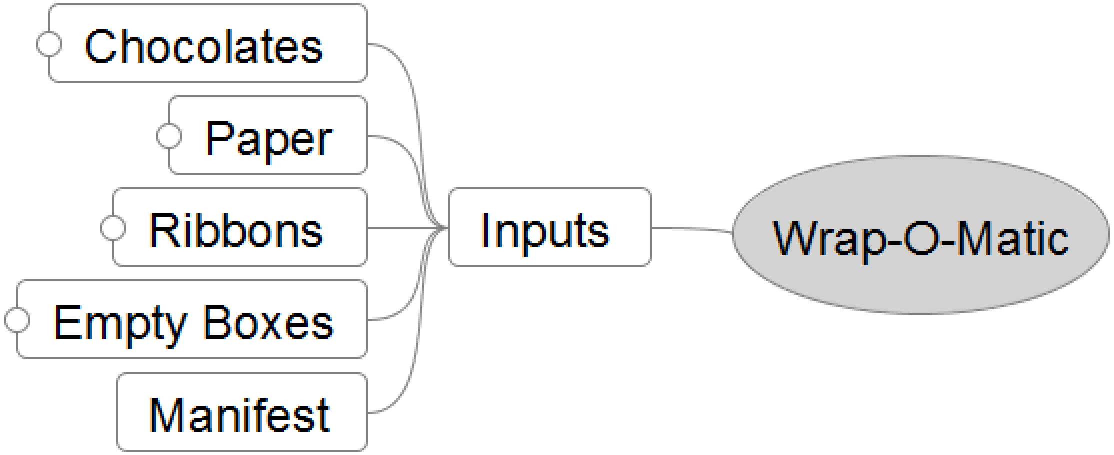


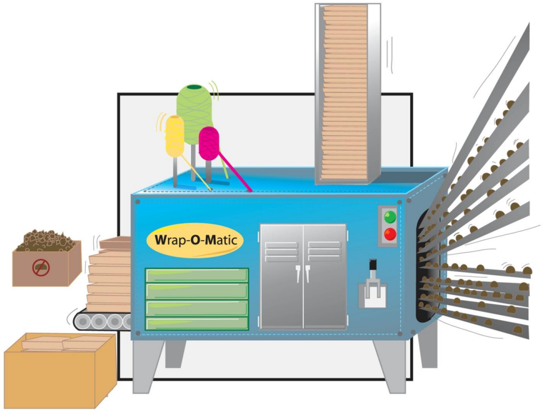
Wrap-O-Matic

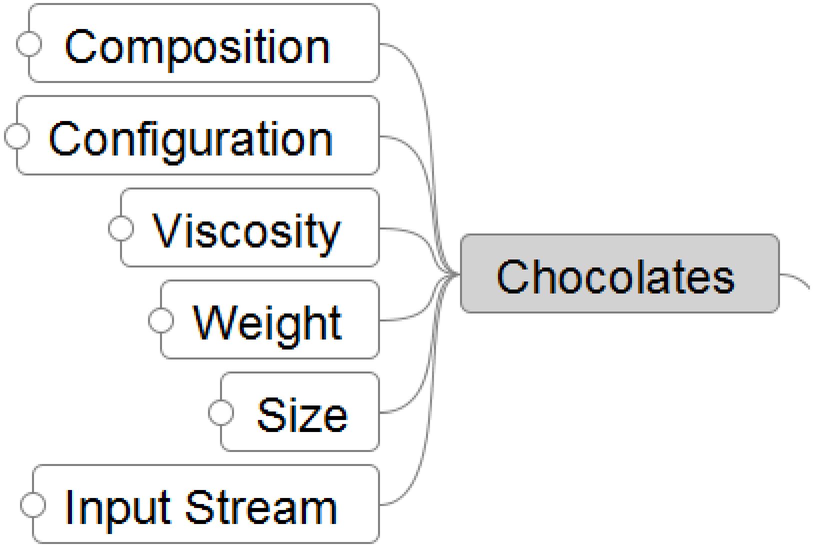
Variable identification


Wrap-O-Matic


Wrap-O-Matic


Chocolates
Paper
Ribbons
Empty Boxes
Manifest
Inputs
Wrap-O-Matic





Composition

Configuration

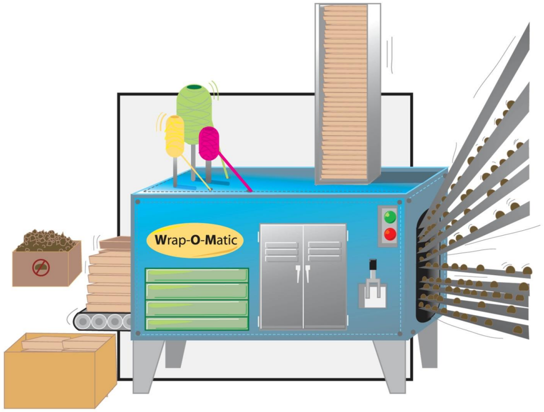
Viscosity

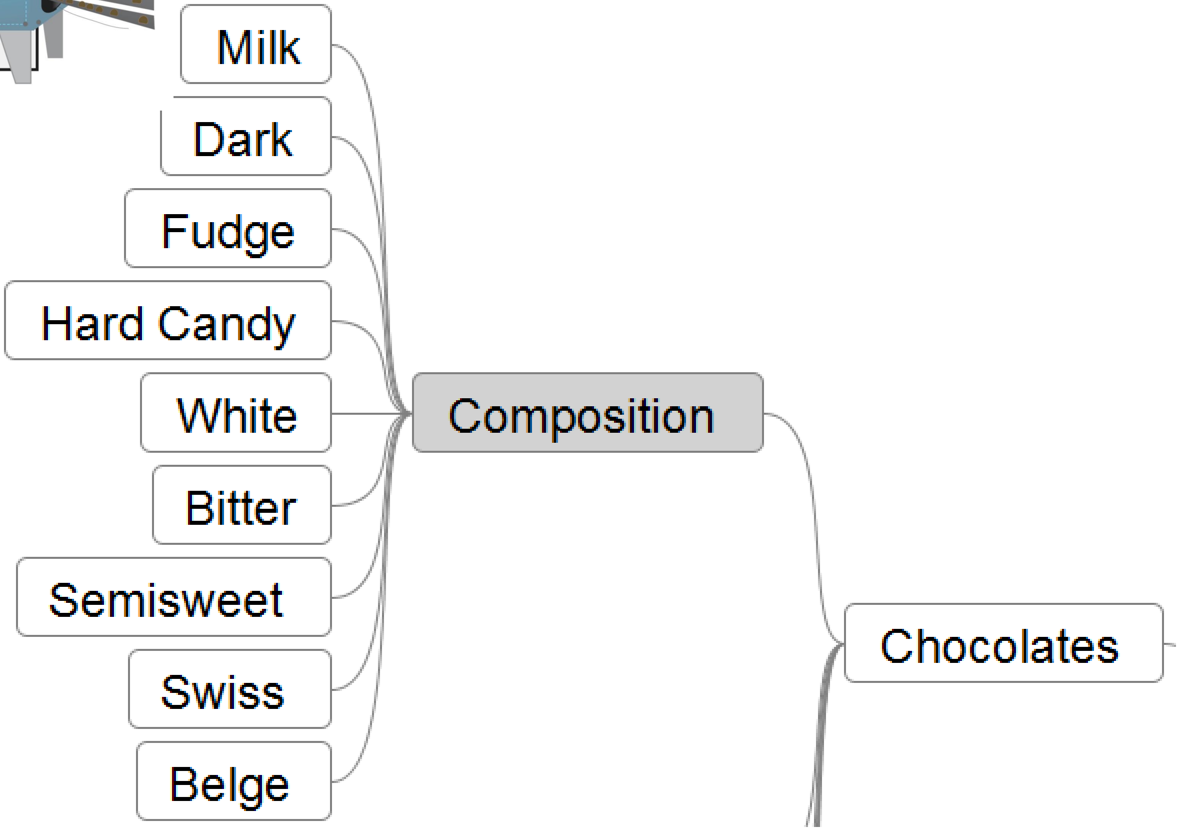
Weight

Size

Input Stream

Chocolates





Milk

Dark

Fudge

Hard Candy

White

Bitter

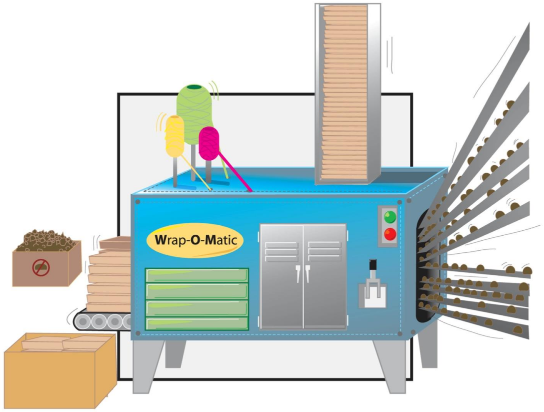
Semisweet

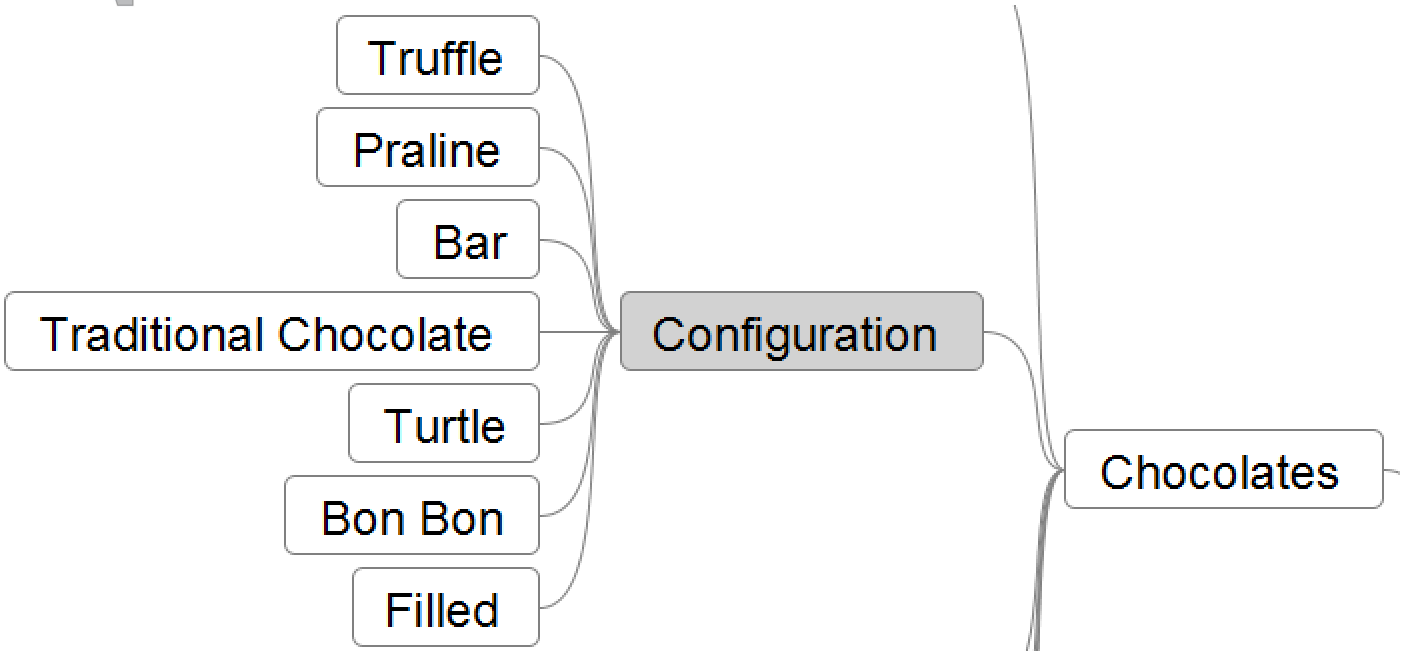
Swiss

Belge

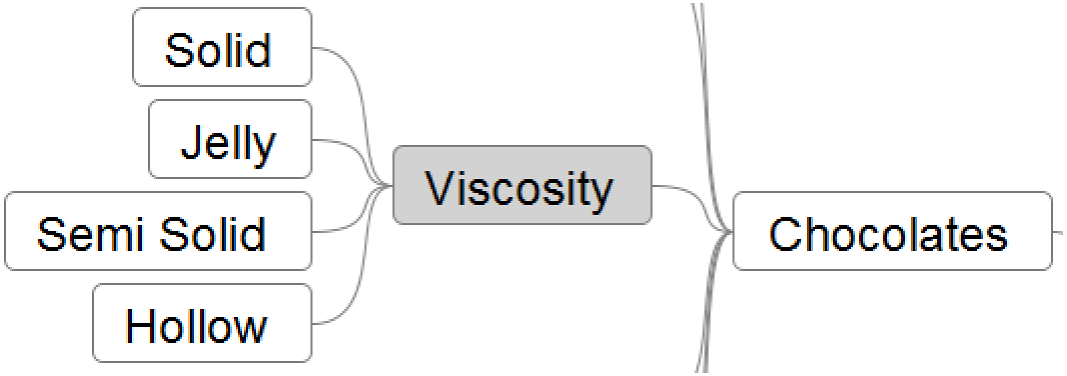
Composition

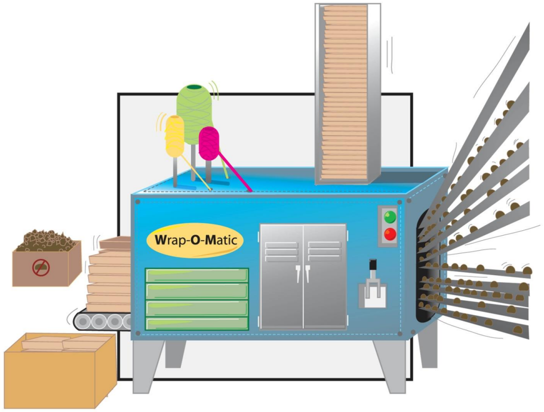
Chocolates

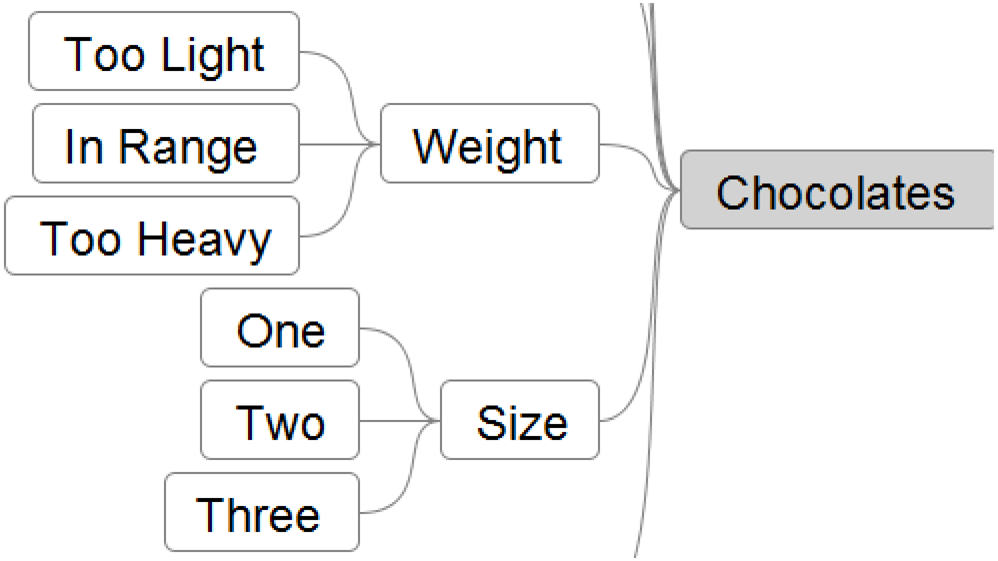

Wrap-O-Matic


Truffle
Praline
Bar
Traditional Chocolate
Turtle
Bon Bon
Filled
Configuration
Chocolates

Wrap-O-Matic


Solid
Jelly
Semi Solid
Hollow
Viscosity
Chocolates




Too Light

In Range

Too Heavy

Weight

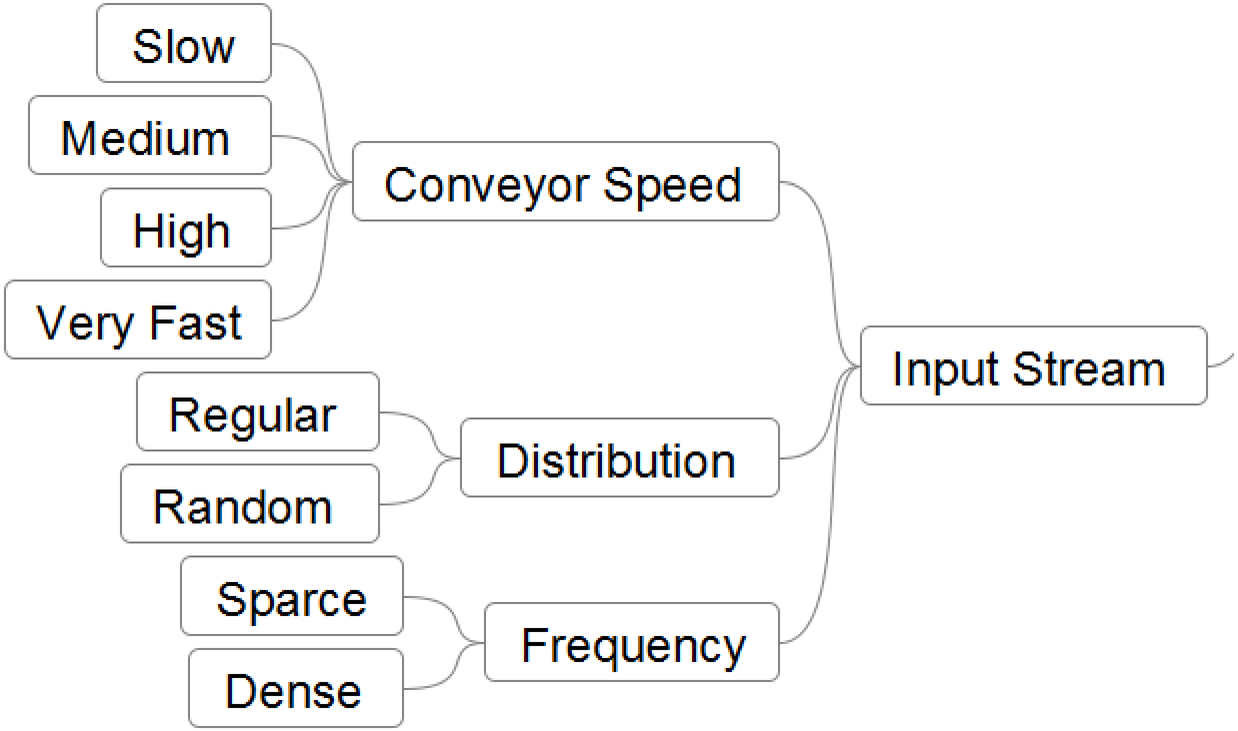
One

Two

Three

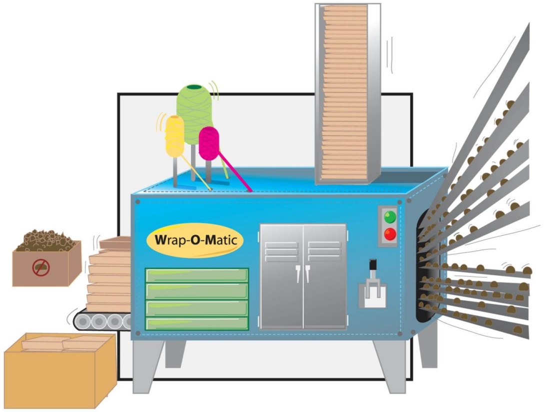
Size

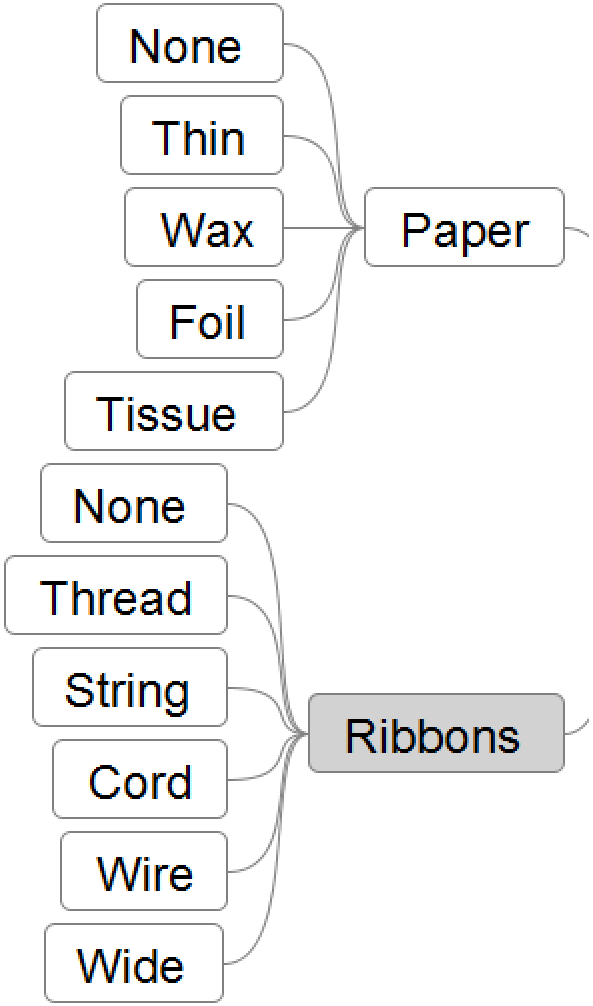
Chocolates



Input Stream

- Conveyor Speed
  - Slow
  - Medium
  - High
  - Very Fast
- Distribution
  - Regular
  - Random
- Frequency
  - Sparce
  - Dense





None

Thin

Wax

Foil

Tissue

Paper

None

Thread

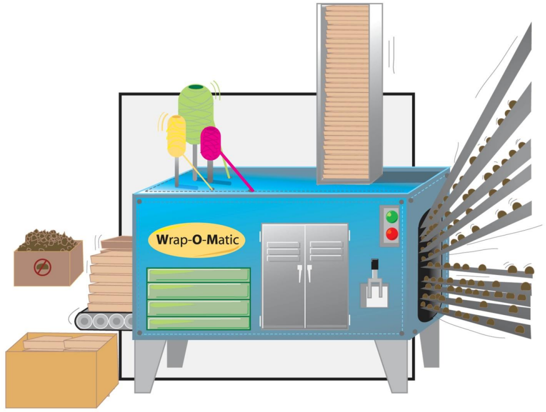
String

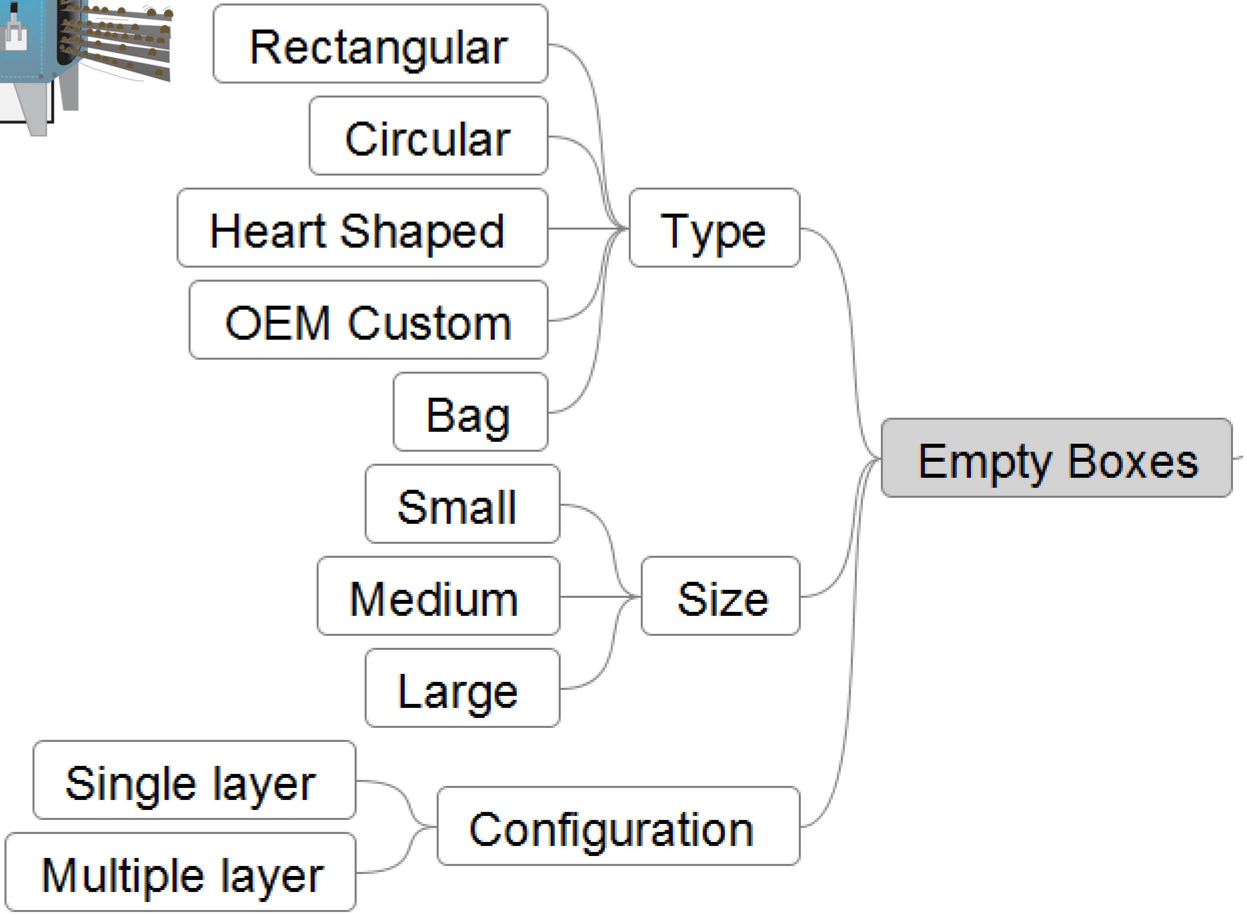
Cord

Wire

Wide

Ribbons



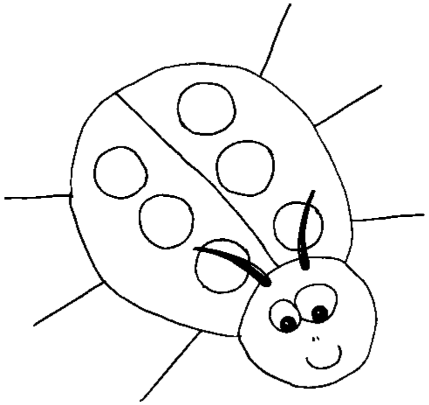


Empty Boxes

Type
- Rectangular
- Circular
- Heart Shaped
- OEM Custom
- Bag

Size
- Small
- Medium
- Large

Configuration
- Single layer
- Multiple layer

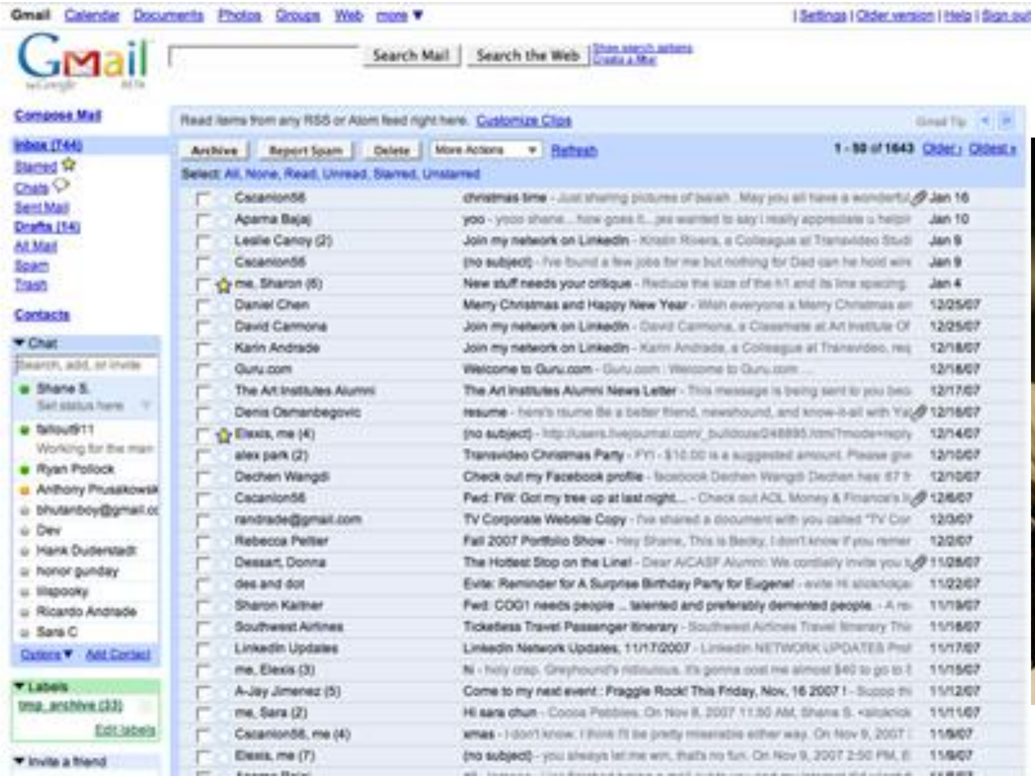


Gmail Image Attachment

Variable identification

Gmail™

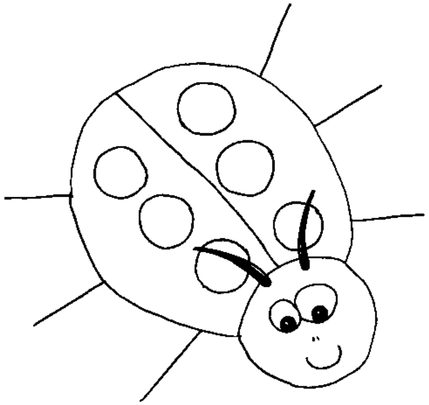
by Google

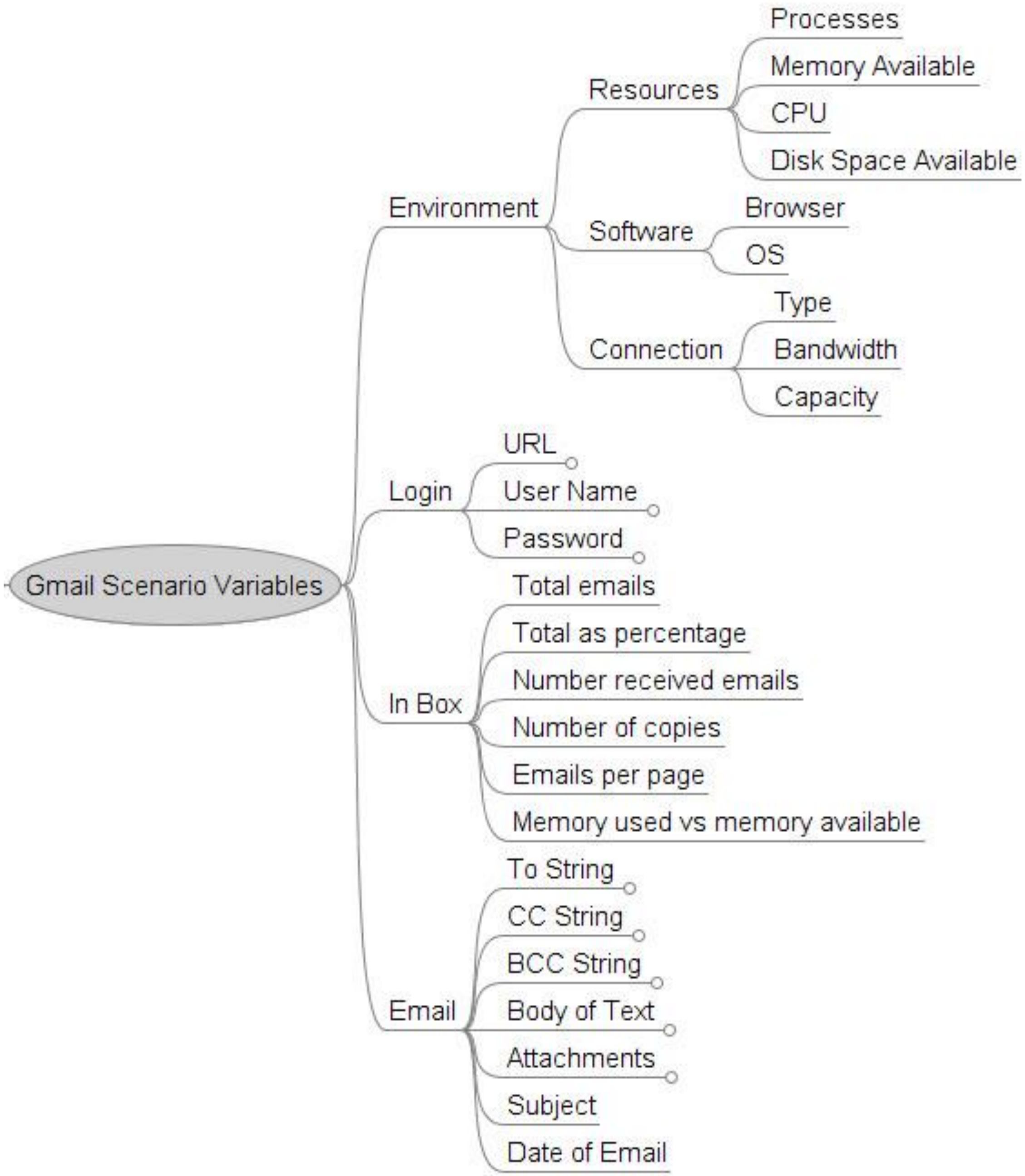


Read items from any RSS or Atom feed right here. Customize Clips

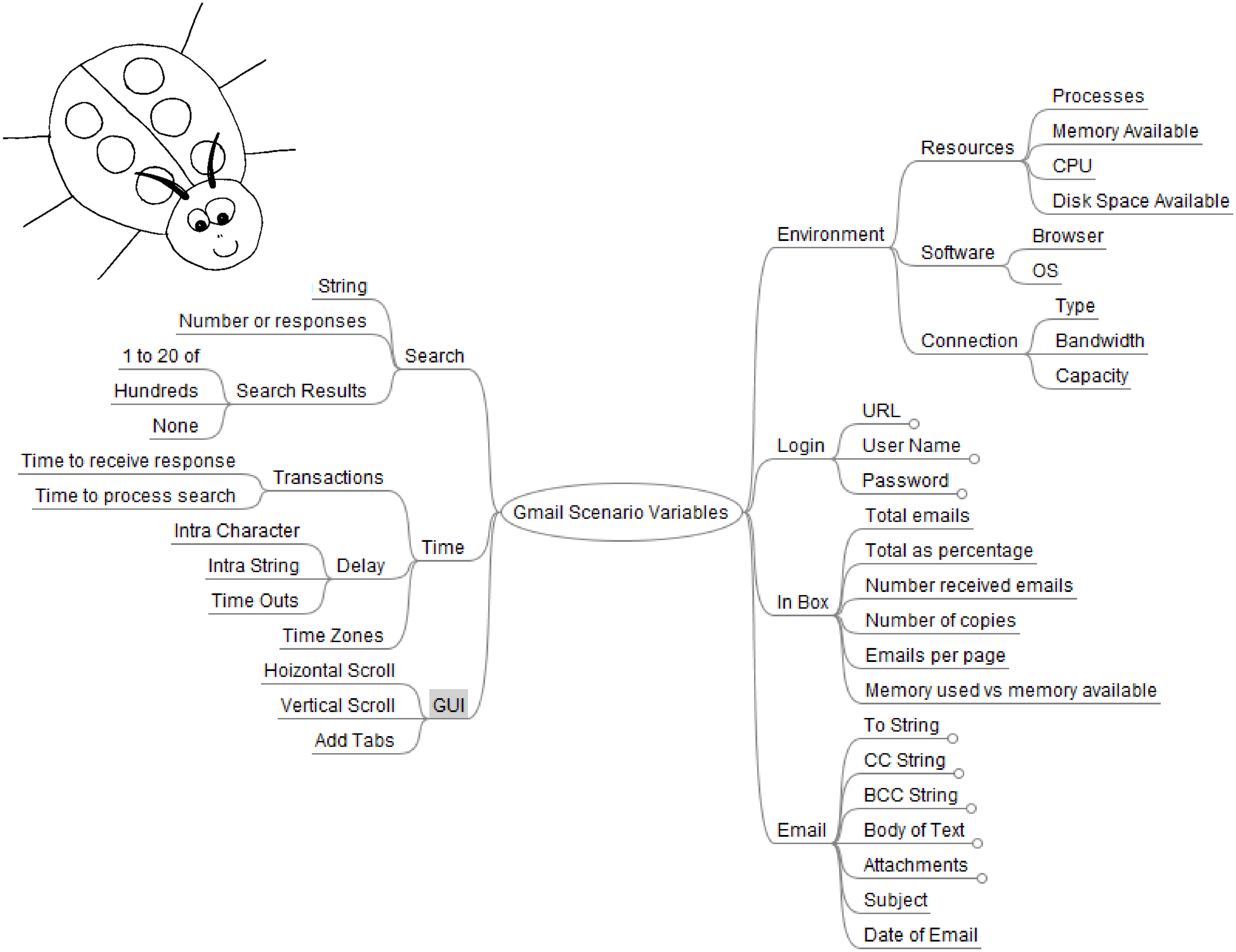
Select: All, None, Read, Unread, Starred, Unstarred

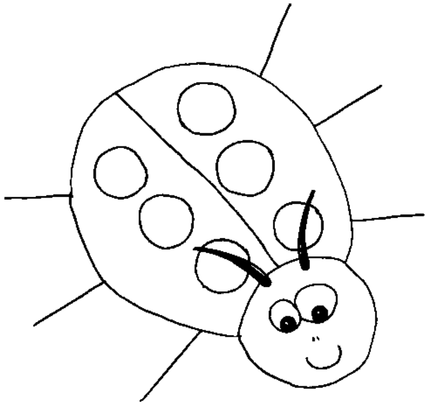
□	Cacanion56	**christmas time** - Just sharing pictures of Isaiah . May you all have a wonderful	Jan 16
□	Aparna Bajaj	**yoo** - yooo shane... how goes it... jes wanted to say i really appreciate u helpin	Jan 10
□	Leslie Canoy (2)	**Join my network on LinkedIn** - Kristin Rivera, a Colleague at Transvideo Studi	Jan 9
□	Cacanion56	**(no subject)** - I've found a few jobs for me but nothing for Dad can he hold wire	Jan 9
□	me, Sharon (6)	**New stuff needs your critique** - Reduce the size of the h1 and its line spacing	Jan 4
□	Daniel Chen	**Merry Christmas and Happy New Year** - Wish everyone a Merry Christmas an	12/25/07
□	David Carmona	**Join my network on LinkedIn** - David Carmona, a Classmate at Art Institute Of	12/25/07
□	Karin Andrade	**Join my network on LinkedIn** - Karin Andrade, a Colleague at Transvideo, req	12/18/07
□	Guru.com	**Welcome to Guru.com** - Guru.com : Welcome to Guru.com ...	12/18/07
□	The Art Institutes Alumni	**The Art Institutes Alumni News Letter** - This message is being sent to you beca	12/17/07
□	Denis Osmanbegovic	**resume** - here's rsume Be a better friend, newshound, and know-it-all with Ya	12/16/07
□	Elexis, me (4)	**(no subject)** - http://users.livejournal.com/_bulldoze/248895.html?mode=reply	12/14/07
□	alex park (2)	**Transvideo Christmas Party** - FYI - $10.00 is a suggested amount. Please give	12/10/07
□	Dechen Wangdi	**Check out my Facebook profile** - facebook Dechen Wangdi Dechen has 67 fr	12/10/07
□	Cacanion56	**Fwd: FW: Got my tree up at last night...** - Check out AOL Money & Finance's li	12/6/07
□	randrade@gmail.com	**TV Corporate Website Copy** - I've shared a document with you called "TV Cor	12/3/07
□	Rebecca Peltier	**Fall 2007 Portfolio Show** - Hey Shane, This is Becky, I don't know if you remer	12/2/07
□	Dessart, Donna	**The Hottest Stop on the Line!** - Dear AiCASF Alumni: We cordially invite you t	11/28/07
□	des and dot	**Evite: Reminder for A Surprise Birthday Party for Eugene!** - evite Hi slickdoga	11/22/07
□	Sharon Kaltner	**Fwd: COO1 needs people ... talented and preferably demented people.** - A re	11/19/07
□	Southwest Airlines	**Ticketless Travel Passenger Itinerary** - Southwest Airlines Travel Itinerary This	11/18/07
□	LinkedIn Updates	**LinkedIn Network Updates, 11/17/2007** - LinkedIn NETWORK UPDATES Prof	11/17/07
□	me, Elexis (3)	**hi** - holy crap. Greyhound's ridiculous. It's gonna cost me almost $40 to go to	11/15/07
□	A-Jay Jimenez (5)	**Come to my next event : Fraggle Rock! This Friday, Nov. 16 2007 !** - Suppp thi	11/12/07
□	me, Sara (2)	**Hi sara chun** - Cocoa Pebbles. On Nov 8, 2007 11:50 AM, Shane S. <slickdick	11/11/07
□	Cacanion56, me (4)	**xmas** - I don't know. I think I'll be pretty miserable either way. On Nov 9, 2007	11/9/07
□	Elexis, me (7)	**(no subject)** - you always let me win, that's no fun. On Nov 9, 2007 2:50 PM, E	11/9/07





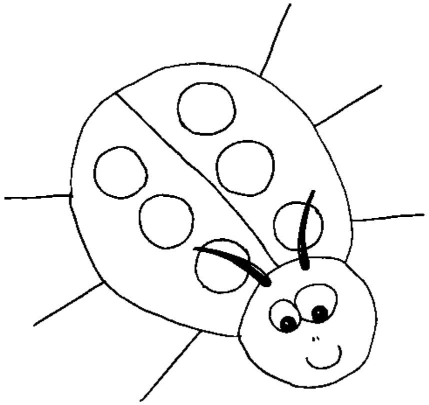
- Gmail Scenario Variables
  - Environment
    - Resources
      - Processes
      - Memory Available
      - CPU
      - Disk Space Available
    - Software
      - Browser
      - OS
    - Connection
      - Type
      - Bandwidth
      - Capacity
  - Login
    - URL
    - User Name
    - Password
  - In Box
    - Total emails
    - Total as percentage
    - Number received emails
    - Number of copies
    - Emails per page
    - Memory used vs memory available
  - Email
    - To String
    - CC String
    - BCC String
    - Body of Text
    - Attachments
    - Subject
    - Date of Email

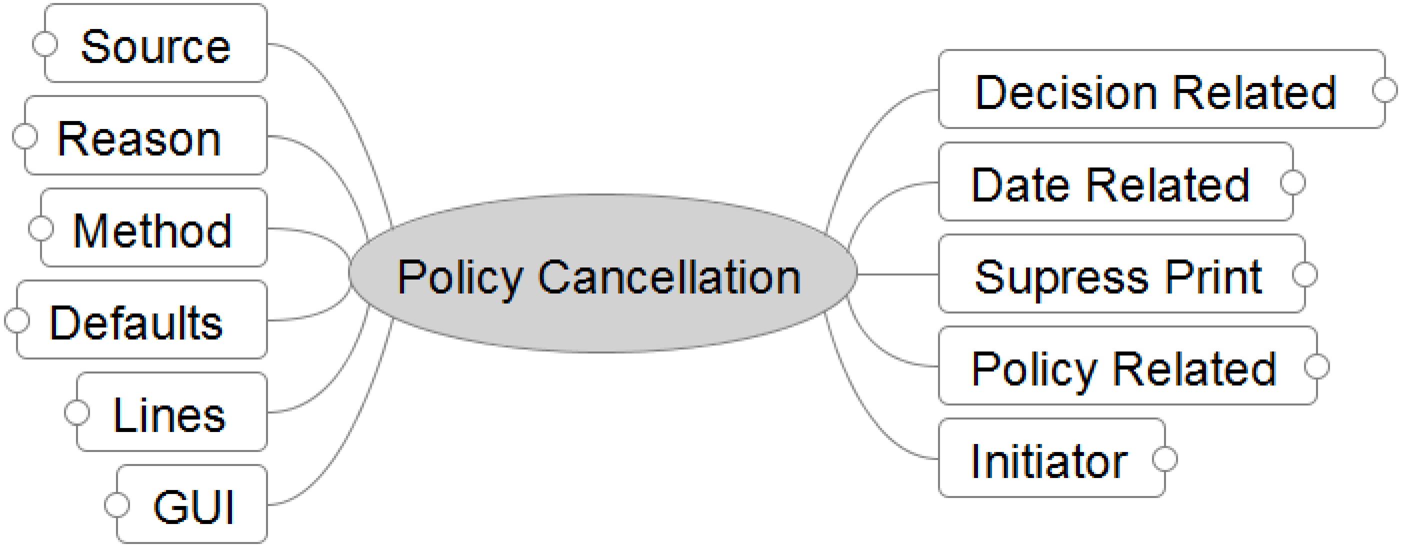

Gmail Scenario Variables
Search
String
Number or responses
Search Results
1 to 20 of
Hundreds
None
Time
Transactions
Time to receive response
Time to process search
Delay
Intra Character
Intra String
Time Outs
Time Zones
GUI
Hoizontal Scroll
Vertical Scroll
Add Tabs
Environment
Resources
Processes
Memory Available
CPU
Disk Space Available
Software
Browser
OS
Connection
Type
Bandwidth
Capacity
Login
URL
User Name
Password
In Box
Total emails
Total as percentage
Number received emails
Number of copies
Emails per page
Memory used vs memory available
Email
To String
CC String
BCC String
Body of Text
Attachments
Subject
Date of Email



Insurance Policy Cancelation

Variable identification





Policy Cancellation

Source

Reason

Method

Defaults

Lines

GUI

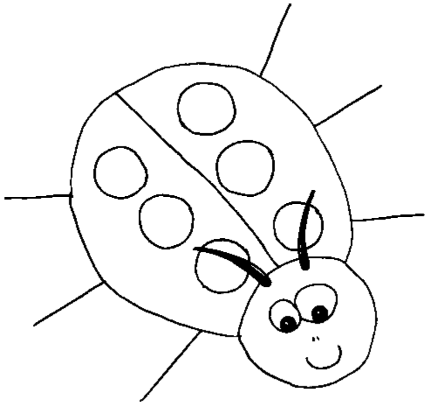
Decision Related

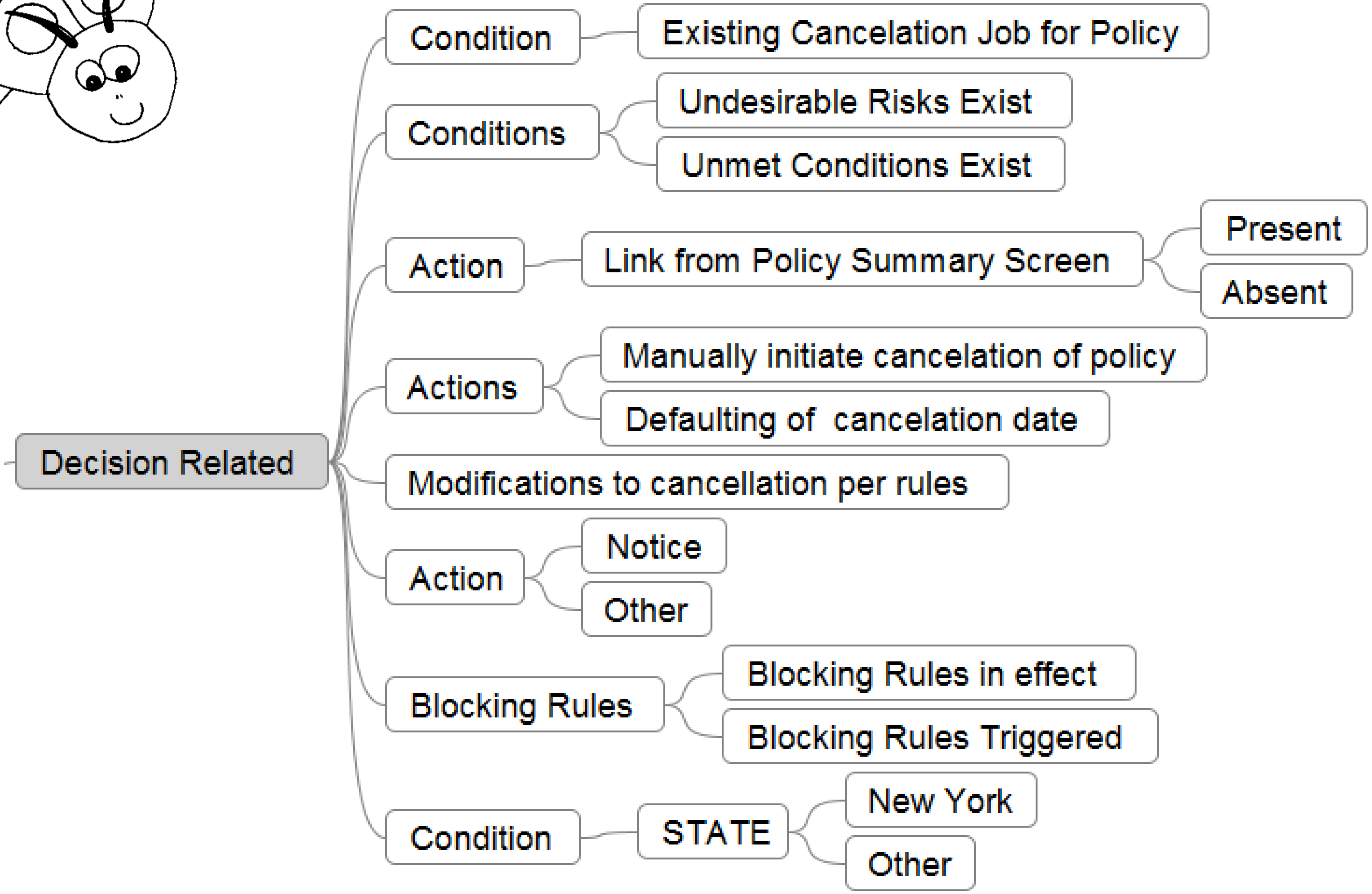
Date Related

Supress Print

Policy Related

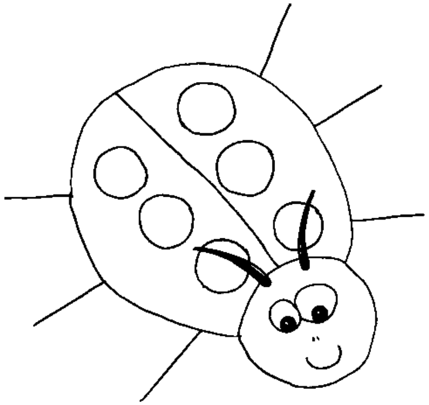
Initiator

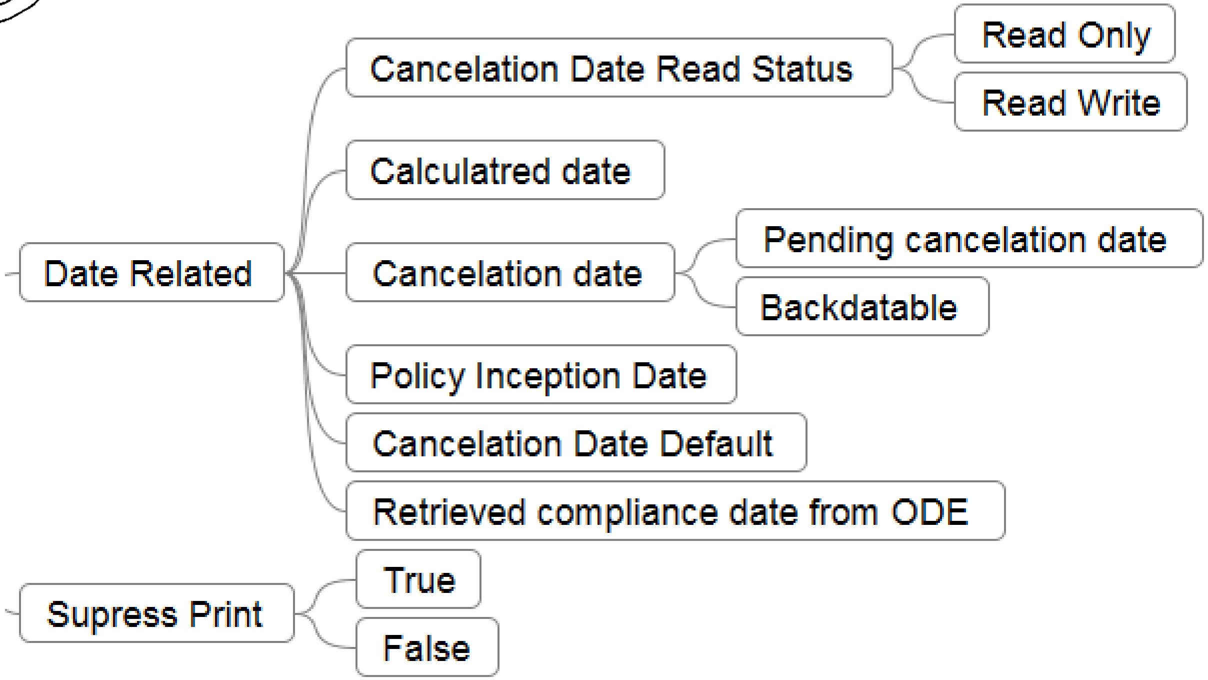




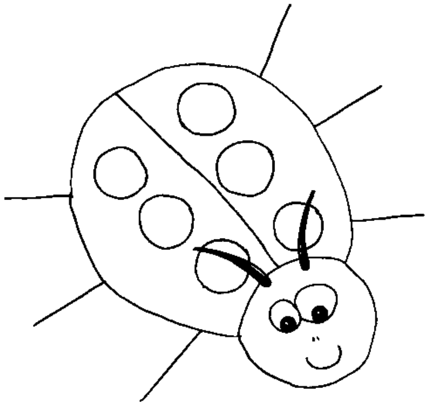
Decision Related

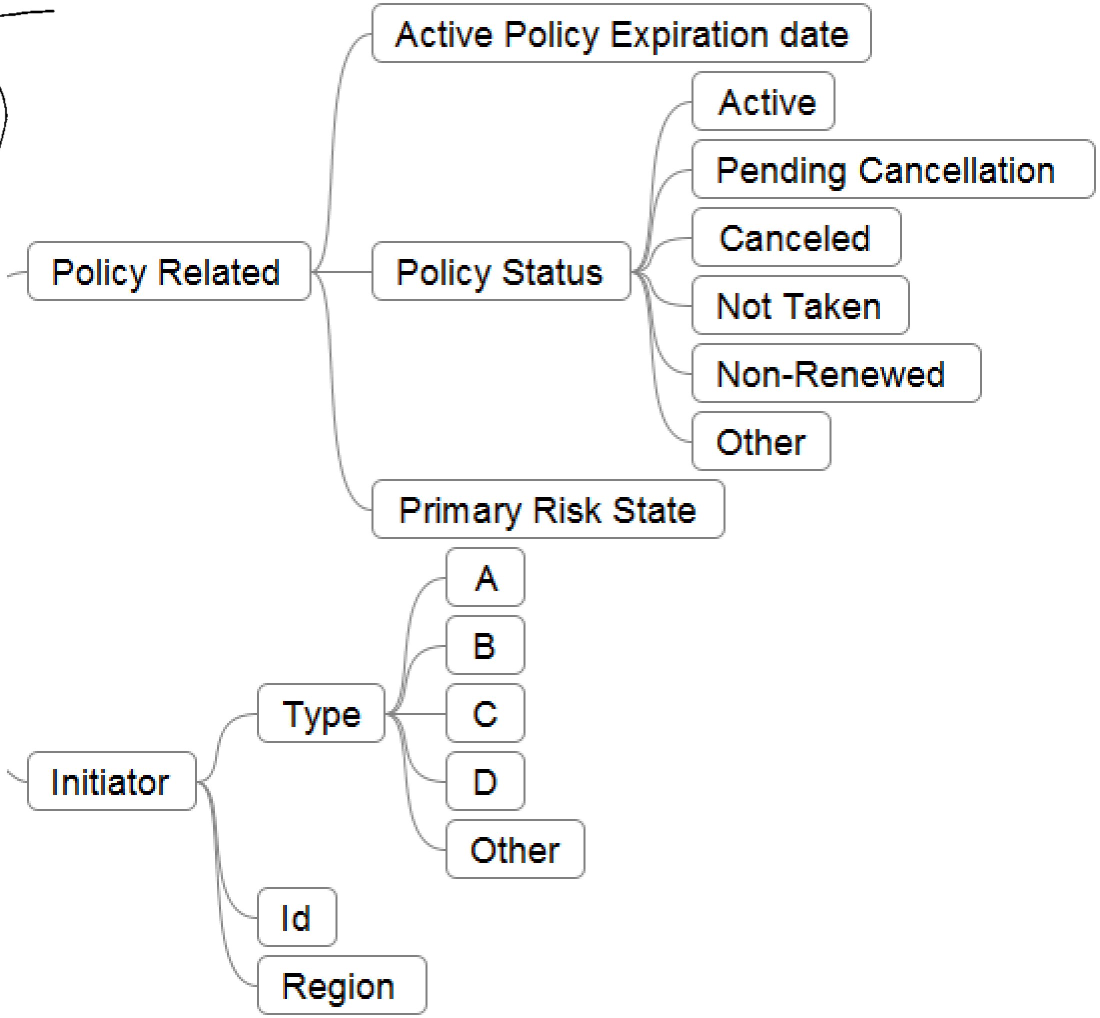
- Condition
  - Existing Cancelation Job for Policy
- Conditions
  - Undesirable Risks Exist
  - Unmet Conditions Exist
- Action
  - Link from Policy Summary Screen
    - Present
    - Absent
- Actions
  - Manually initiate cancelation of policy
  - Defaulting of cancelation date
- Modifications to cancellation per rules
- Action
  - Notice
  - Other
- Blocking Rules
  - Blocking Rules in effect
  - Blocking Rules Triggered
- Condition
  - STATE
    - New York
    - Other



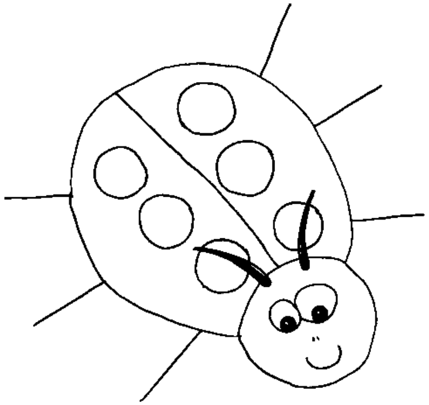


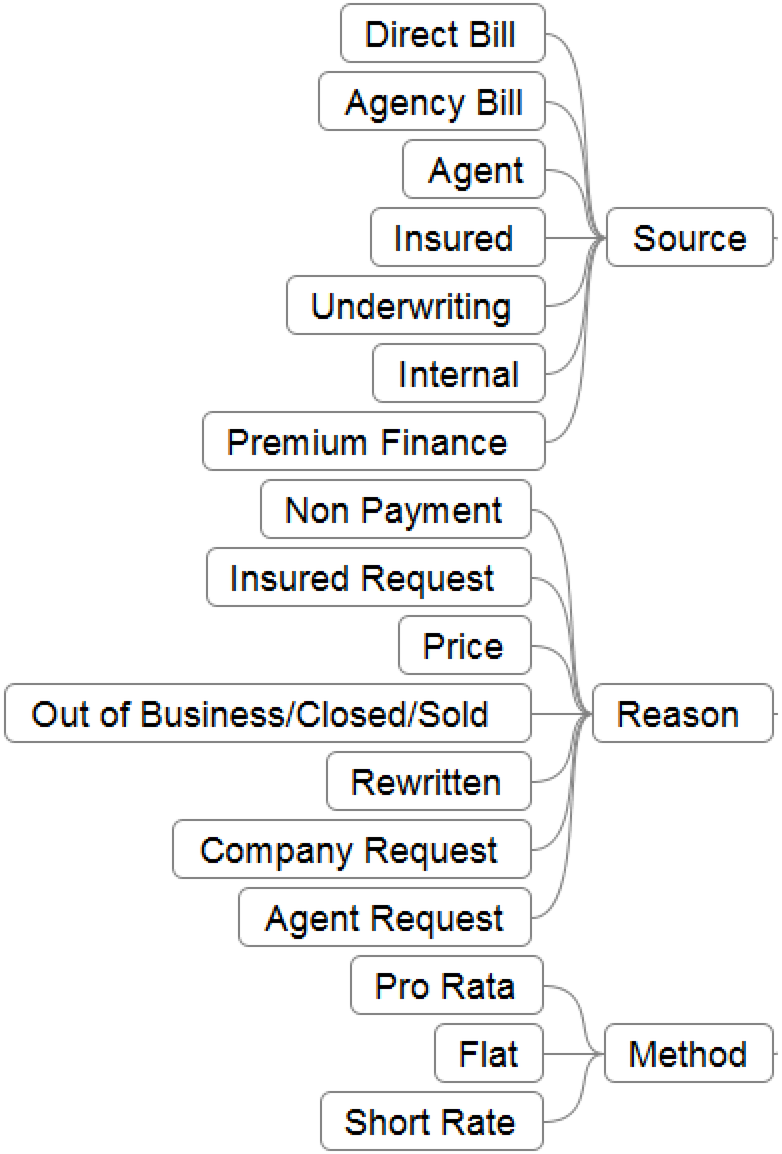
- Date Related
  - Cancelation Date Read Status
    - Read Only
    - Read Write
  - Calculatred date
  - Cancelation date
    - Pending cancelation date
    - Backdatable
  - Policy Inception Date
  - Cancelation Date Default
  - Retrieved compliance date from ODE
- Supress Print
  - True
  - False





- Policy Related
  - Active Policy Expiration date
  - Policy Status
    - Active
    - Pending Cancellation
    - Canceled
    - Not Taken
    - Non-Renewed
    - Other
  - Primary Risk State
- Initiator
  - Type
    - A
    - B
    - C
    - D
    - Other
  - Id
  - Region

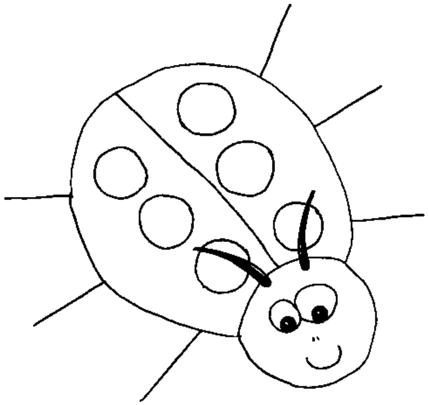


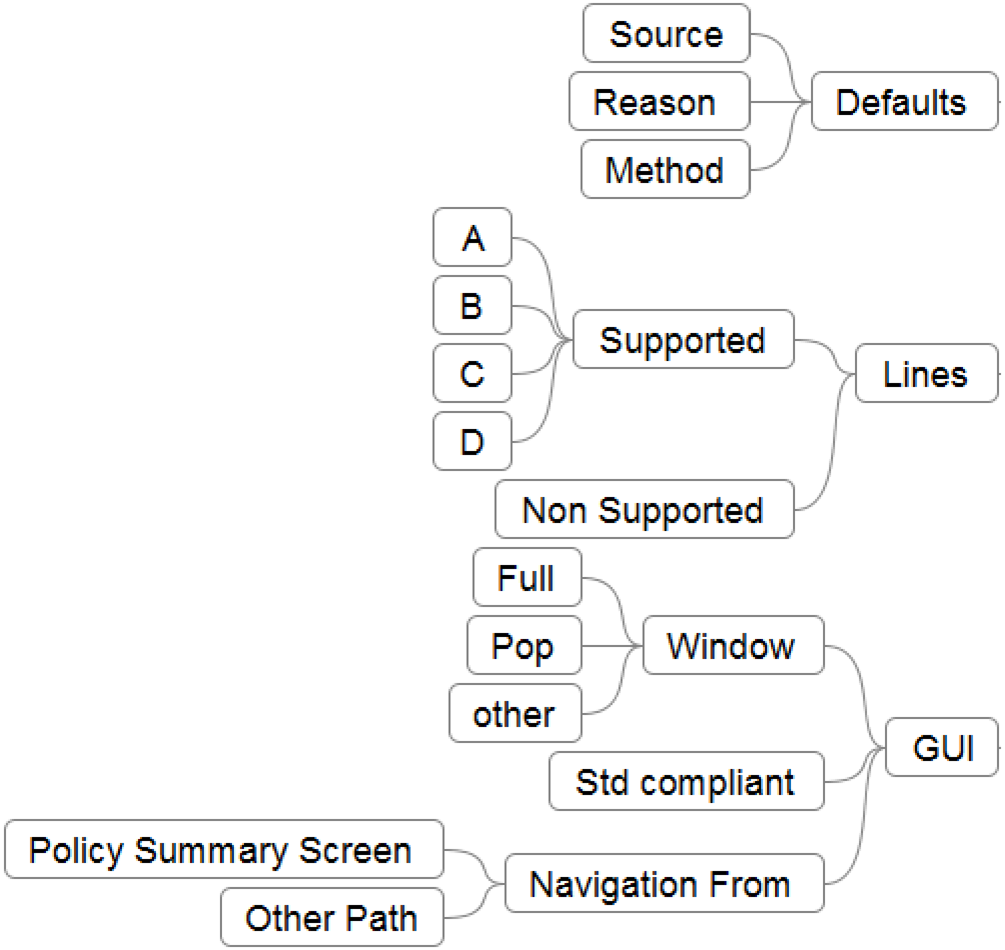


Direct Bill
Agency Bill
Agent
Insured
Underwriting
Internal
Premium Finance
Source

Non Payment
Insured Request
Price
Out of Business/Closed/Sold
Rewritten
Company Request
Agent Request
Reason

Pro Rata
Flat
Short Rate
Method

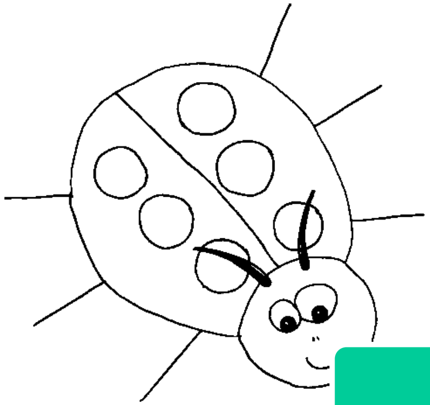




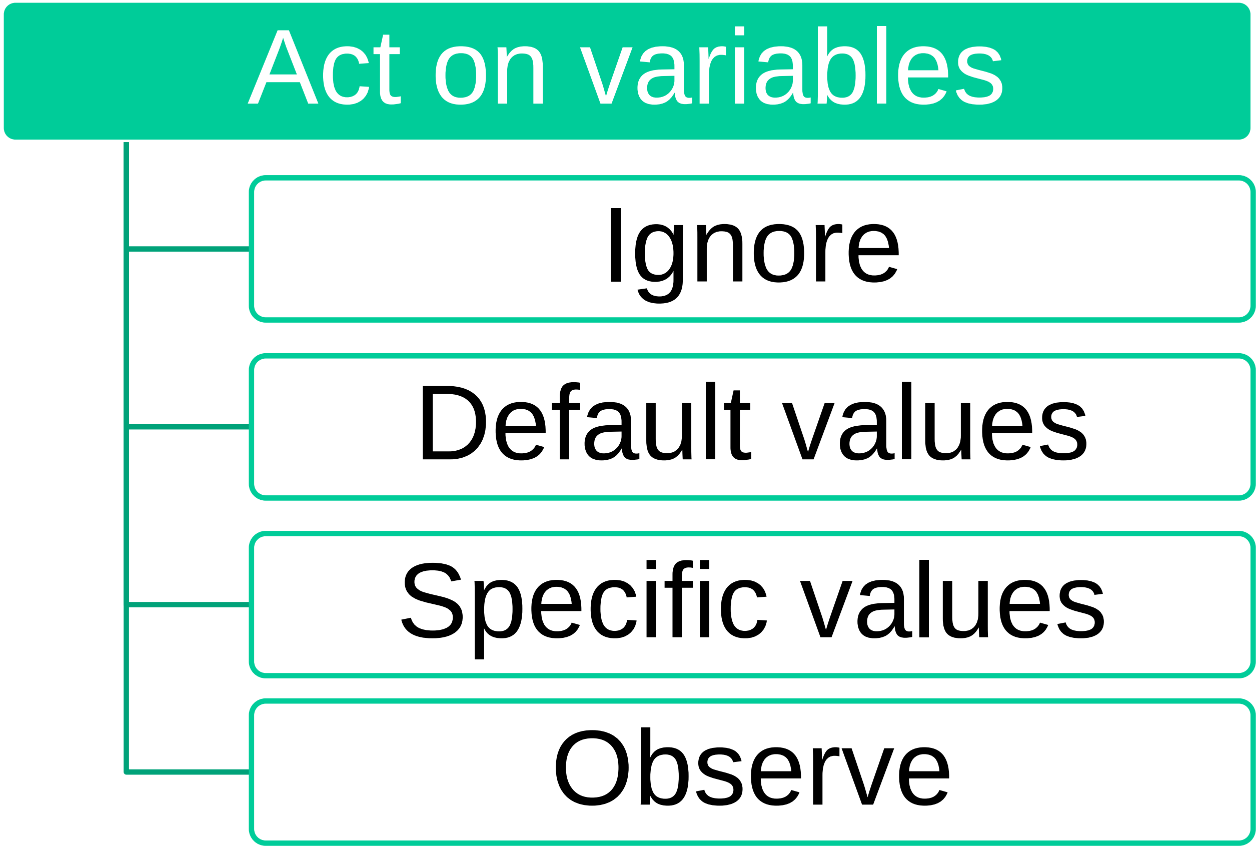
Source
Reason
Method
Defaults

A
B
C
D
Supported
Non Supported
Lines

Full
Pop
other
Window
Std compliant
Policy Summary Screen
Other Path
Navigation From
GUI

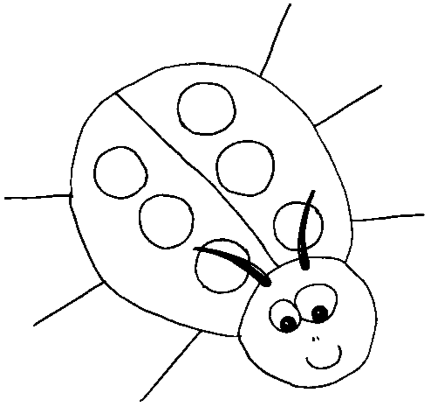


Identify Variables



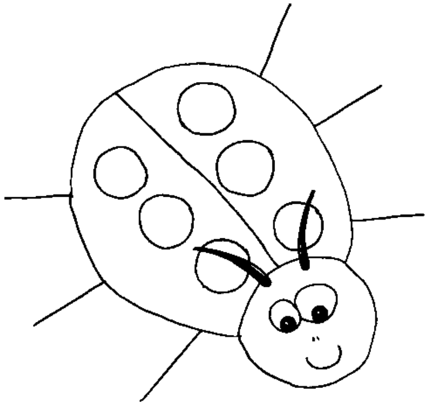
Act on variables

- Ignore
- Default values
- Specific values
- Observe

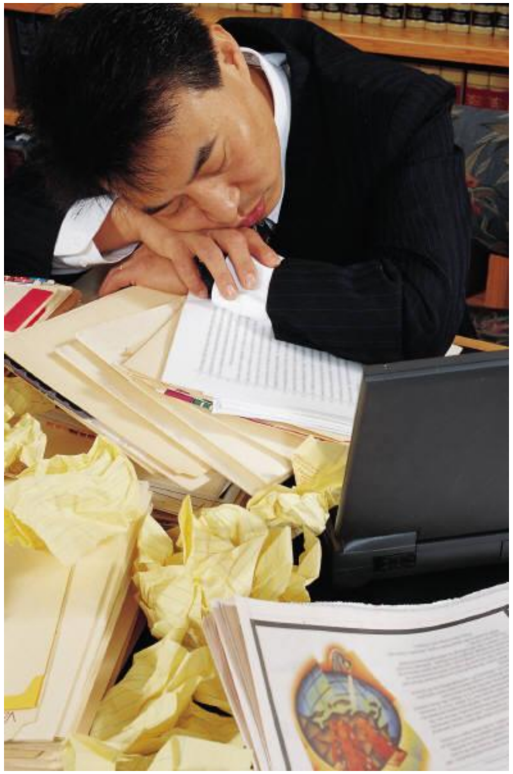


Test Design

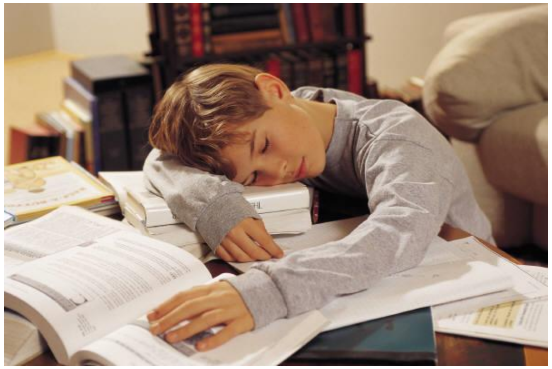
Equivalence Classes



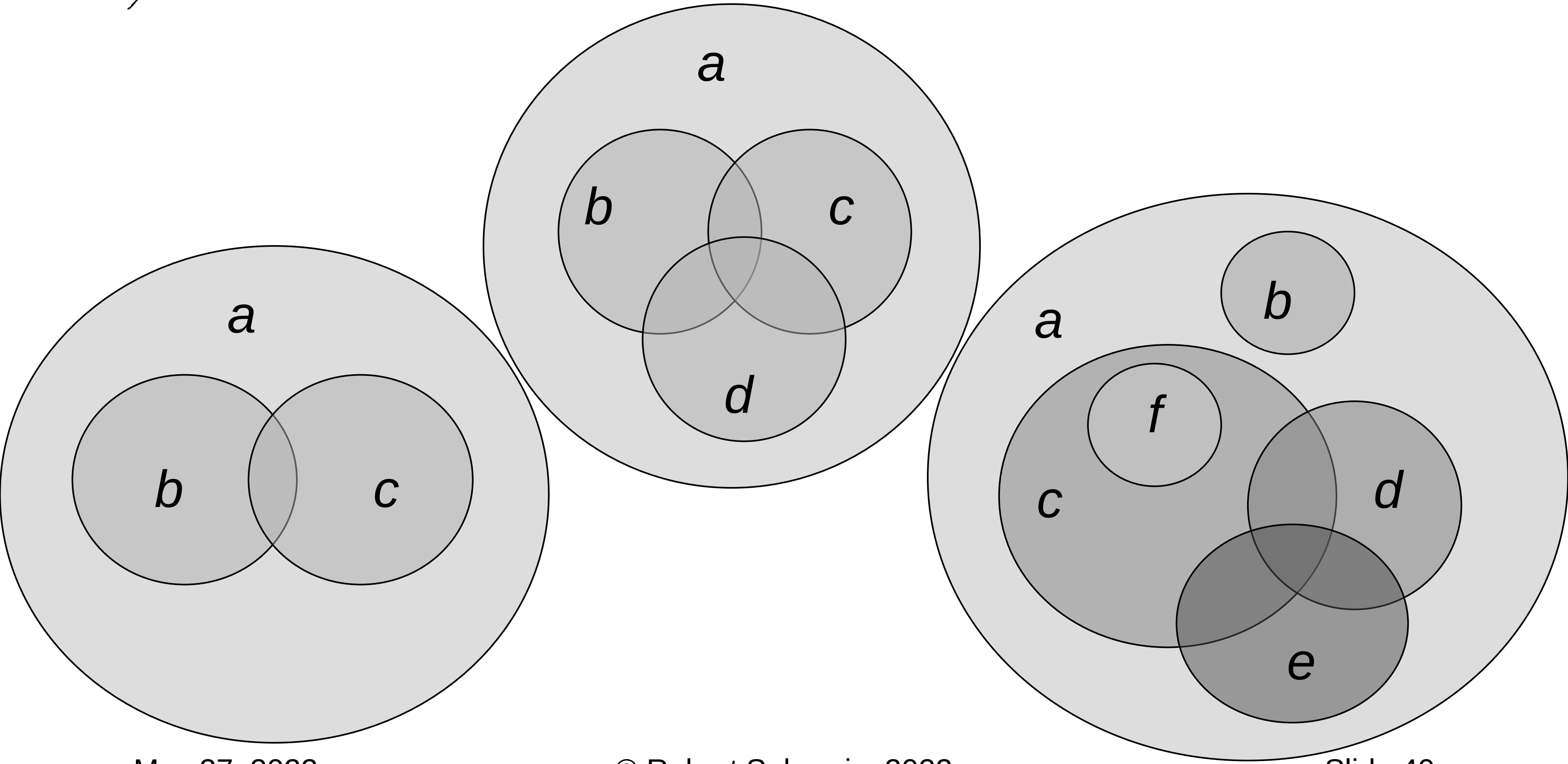
Taking AIM

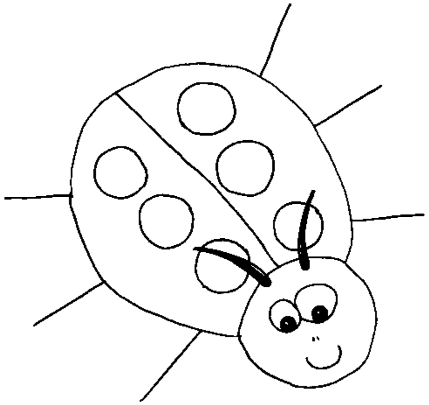


- Equivalence class
  - A *subset* of all possible test values to a variable
  - Each member assumed provide the same info
  - Each variable may have *many classes*
  - Equivalence class are *not mutually exclusive*
  - Focus testing
  - Reduce the number of test cases



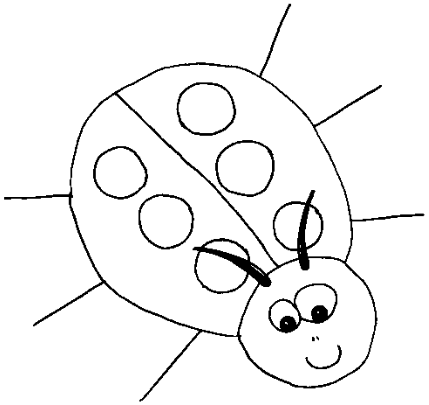
Sets - Venn Diagram
Equivalence Classes



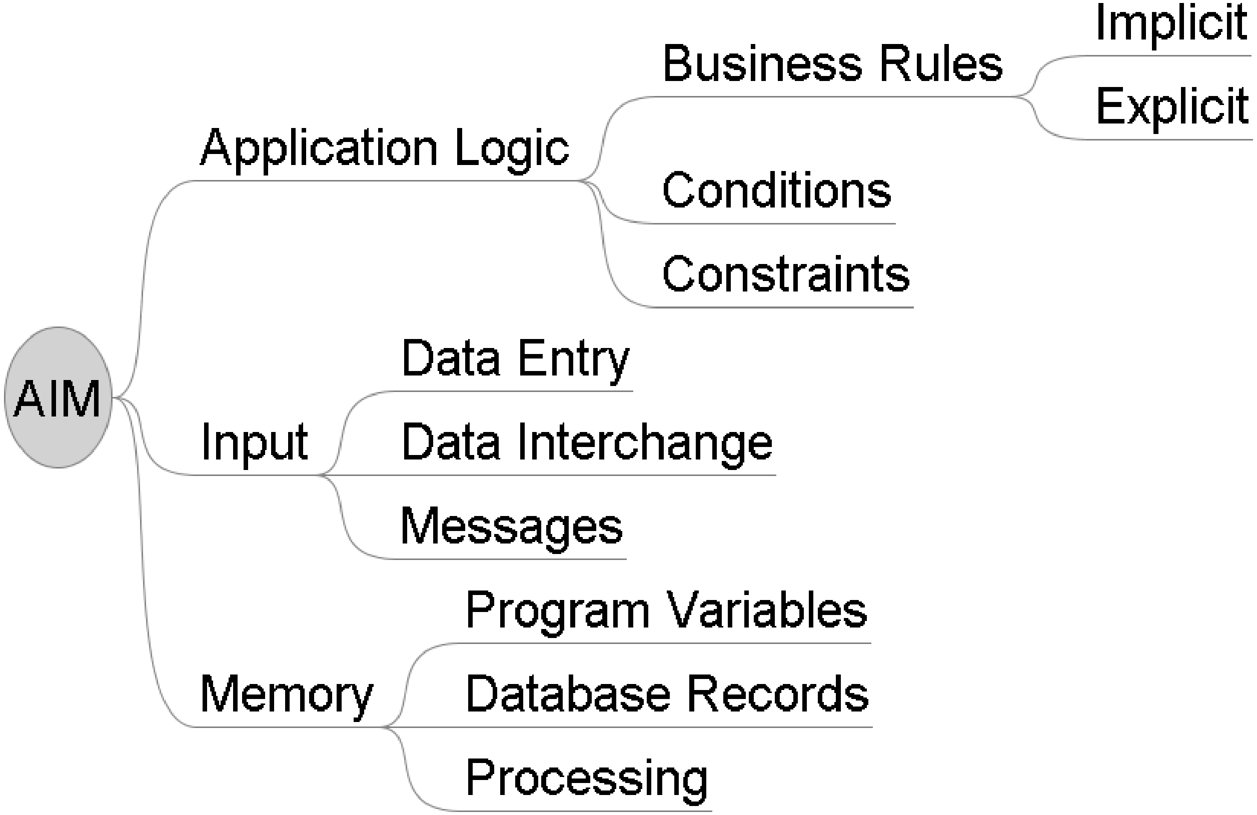


Equivalence Classes

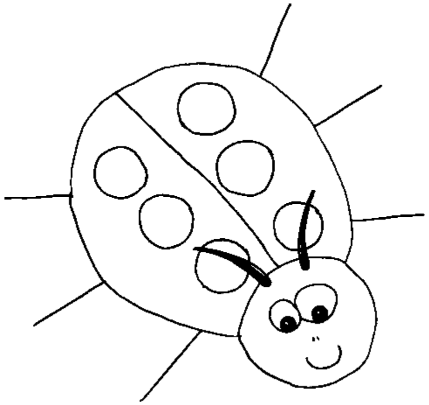
- Sources
  - *Requirements*
    - Business logic
    - Capabilities
    - Ranges
    - Constraints
  - *Code*
    - Decisions
    - Intermediate computations
  - Data
    - Input fields
    - Database
    - Internal structures



AIM - Equivalence Classes



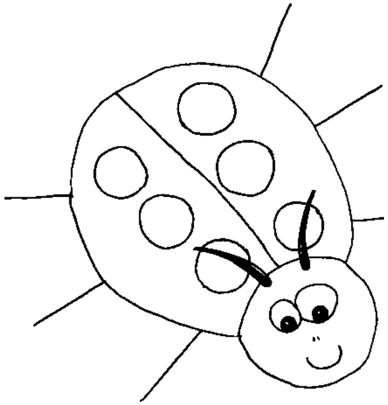
- AIM
  - Application Logic
    - Business Rules
      - Implicit
      - Explicit
    - Conditions
    - Constraints
  - Input
    - Data Entry
    - Data Interchange
    - Messages
  - Memory
    - Program Variables
    - Database Records
    - Processing

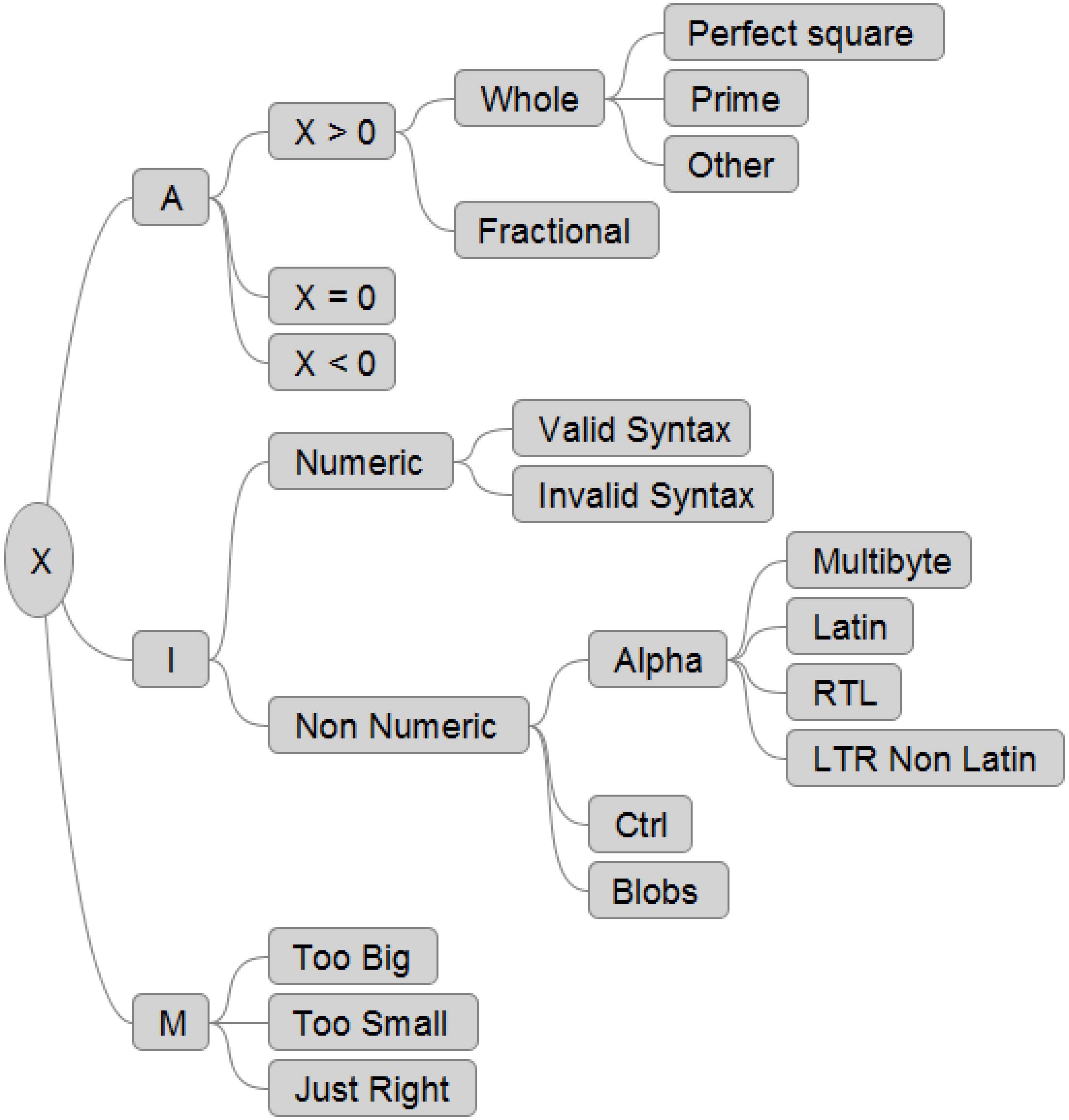


Square Root Function

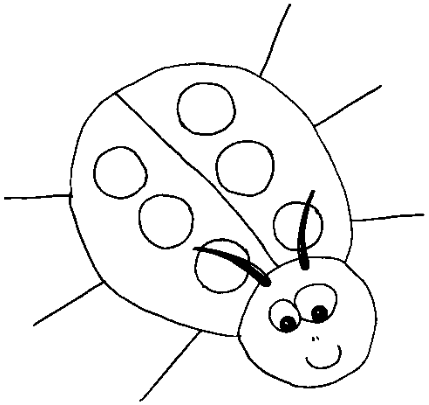
AIM - Equivalence Classes

Example: SQRT X
Equivalence Classes



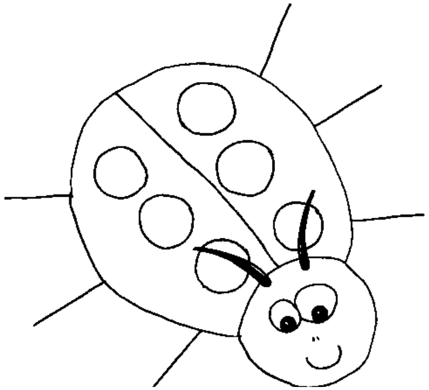


- X
  - A
    - X > 0
      - Whole
        - Perfect square
        - Prime
        - Other
      - Fractional
    - X = 0
    - X < 0
  - I
    - Numeric
      - Valid Syntax
      - Invalid Syntax
    - Non Numeric
      - Alpha
        - Multibyte
        - Latin
        - RTL
        - LTR Non Latin
      - Ctrl
      - Blobs
  - M
    - Too Big
    - Too Small
    - Just Right

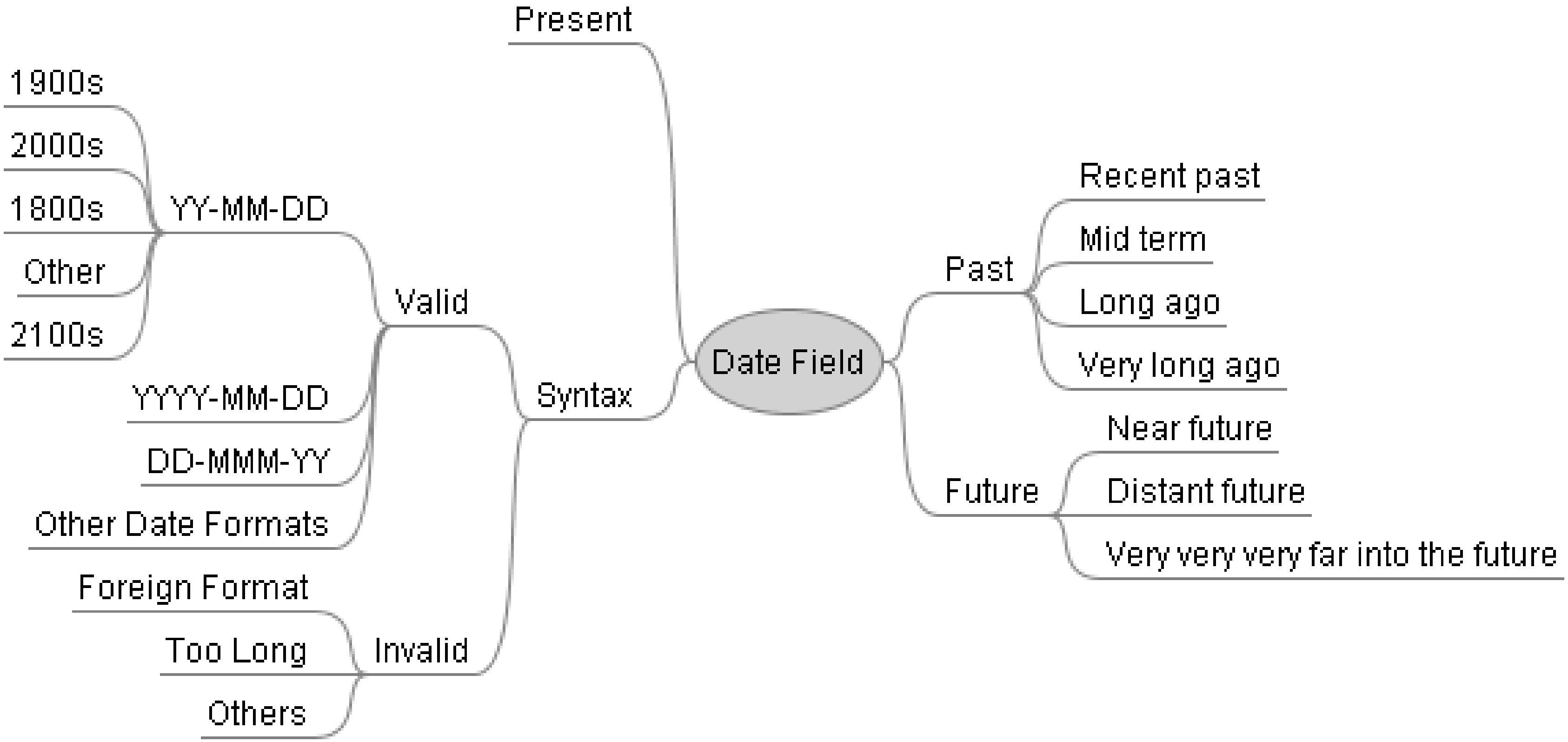


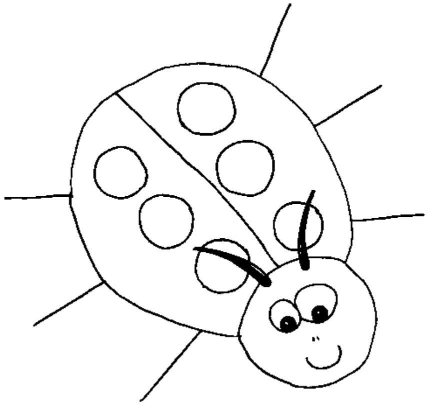
Date Field

AIM - Equivalence Classes



Example: Date Field Equivalence Classes Mind Map

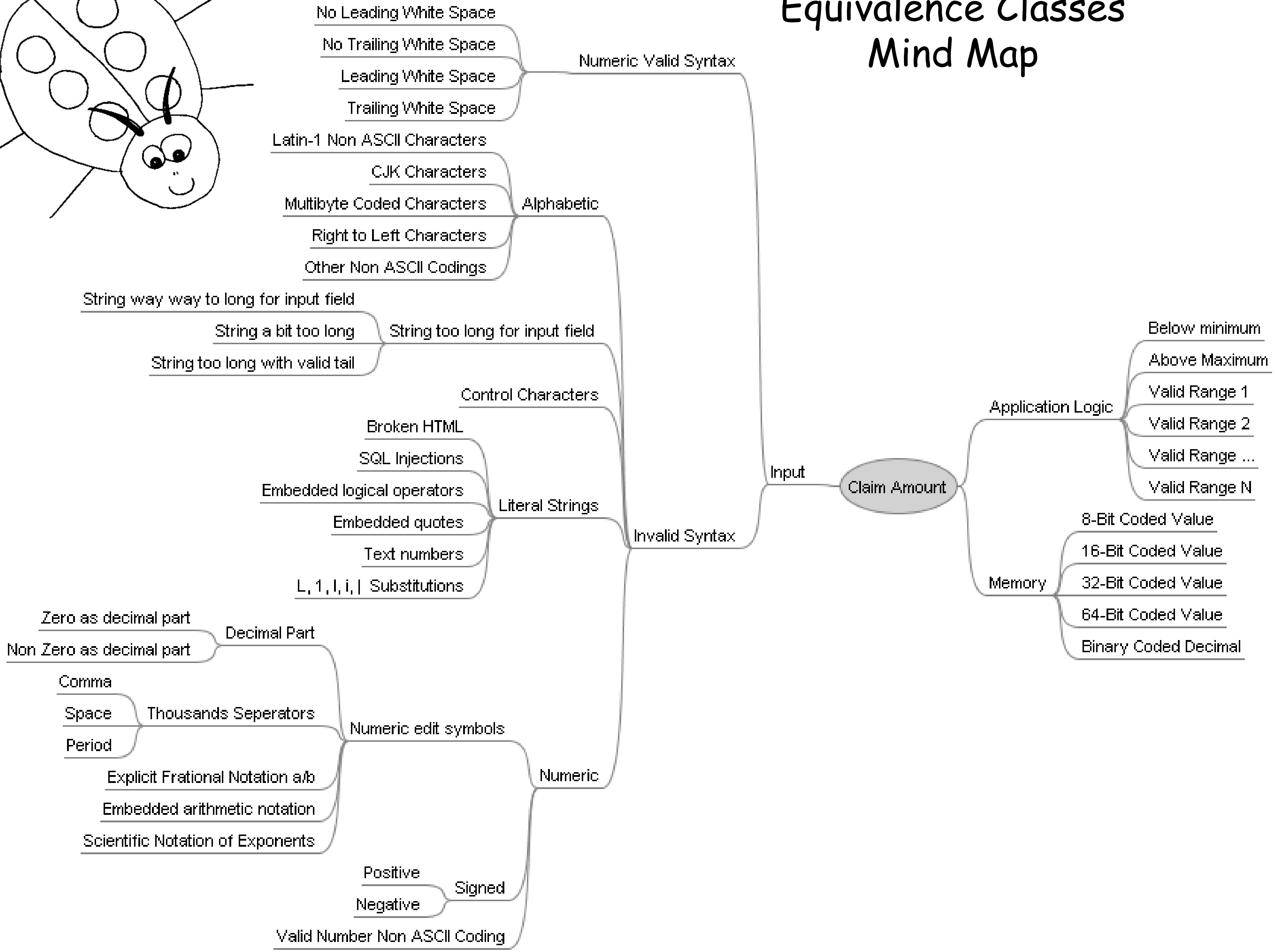




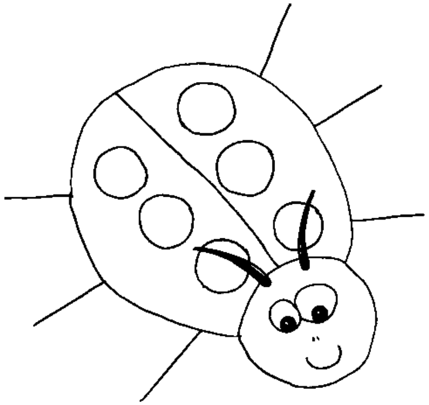
Insurance Claim Amount

AIM - Equivalence Classes

Example: Claim Amount Equivalence Classes Mind Map

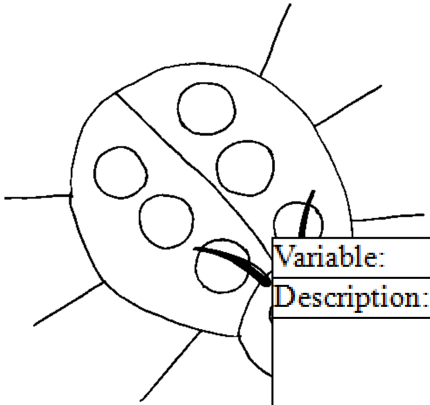


- Claim Amount
  - Input
    - Numeric Valid Syntax
      - No Leading White Space
      - No Trailing White Space
      - Leading White Space
      - Trailing White Space
    - Invalid Syntax
      - Alphabetic
        - Latin-1 Non ASCII Characters
        - CJK Characters
        - Multibyte Coded Characters
        - Right to Left Characters
        - Other Non ASCII Codings
      - String too long for input field
        - String way way to long for input field
        - String a bit too long
        - String too long with valid tail
      - Control Characters
      - Literal Strings
        - Broken HTML
        - SQL Injections
        - Embedded logical operators
        - Embedded quotes
        - Text numbers
        - L, 1, l, i, | Substitutions
      - Numeric
        - Numeric edit symbols
          - Decimal Part
            - Zero as decimal part
            - Non Zero as decimal part
          - Thousands Seperators
            - Comma
            - Space
            - Period
          - Explicit Frational Notation a/b
          - Embedded arithmetic notation
          - Scientific Notation of Exponents
        - Signed
          - Positive
          - Negative
        - Valid Number Non ASCII Coding
  - Application Logic
    - Below minimum
    - Above Maximum
    - Valid Range 1
    - Valid Range 2
    - Valid Range ...
    - Valid Range N
  - Memory
    - 8-Bit Coded Value
    - 16-Bit Coded Value
    - 32-Bit Coded Value
    - 64-Bit Coded Value
    - Binary Coded Decimal

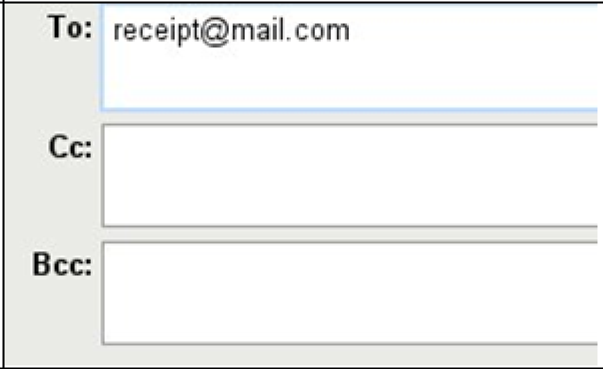


Gmail “to composed”

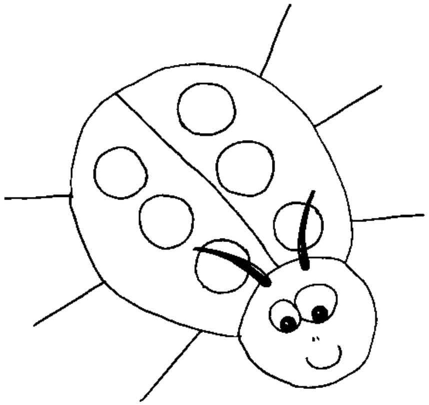
AIM - Equivalence Classes

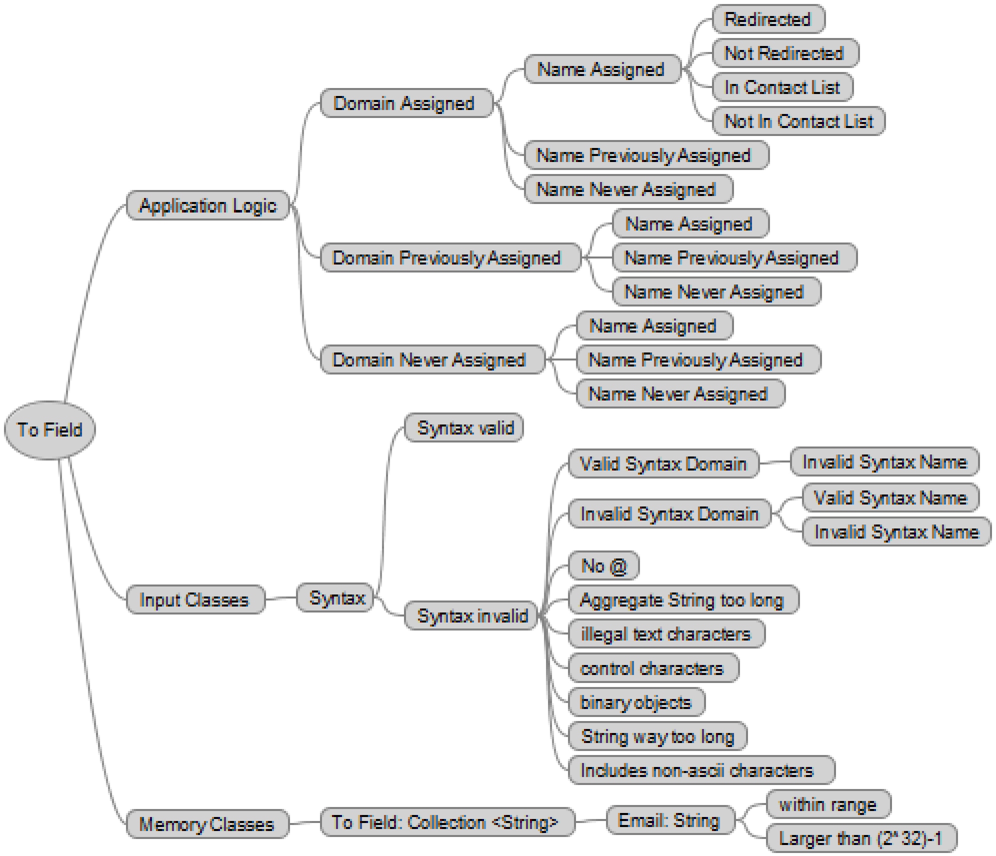


Example: Gmail To Composed Equivalence Classes Mind Map

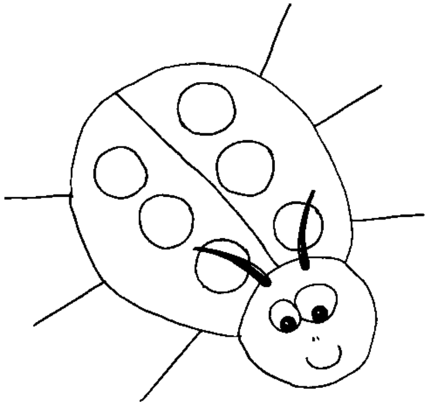
Variable:	To (composed email)
Description:	This is the To field in a composed email
	 To: receipt@mail.com Cc: Bcc:
Gmail API Source:	g4j-gmail_api/GMComposedMessage.java (Line 19)
Other Source:	Java Collection: http://java.sun.com/j2se/1.4.2/docs/api/java/util/Collection.html A collection object is used to implement a list to store emails Java String Implementation: http://www.docjar.com/html/api/java/lang/String.java.html (Line 119: number of characters is stored as an *int*) Java *int* Implementation: http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Integer.html (constant MAX_VALUE states that maximum value of int is 2^31-1 , thus maximum length of *String* is 2^31-1)

Example: Gmail To Composed Equivalence Classes Mind Map

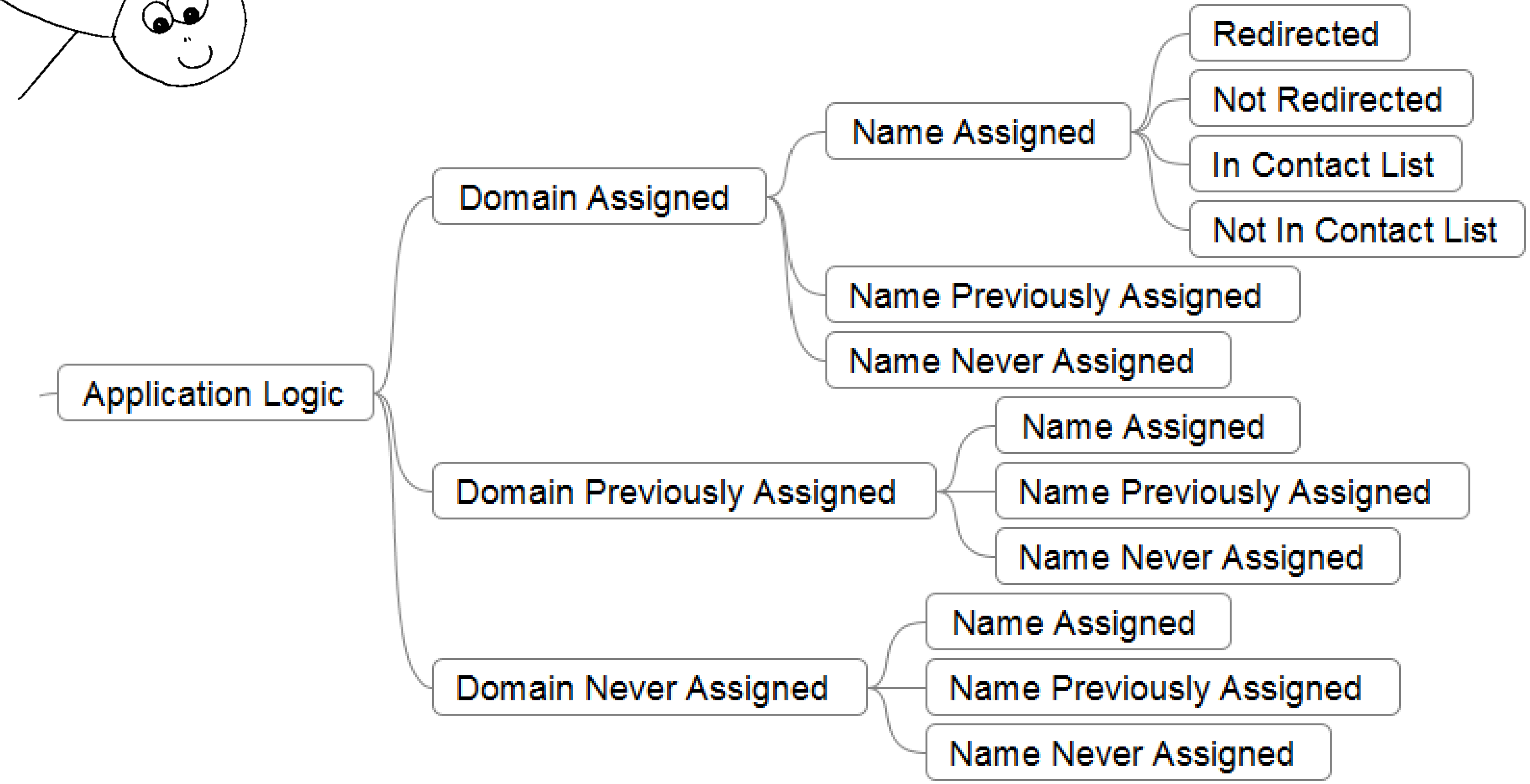




- To Field
  - Application Logic
    - Domain Assigned
      - Name Assigned
        - Redirected
        - Not Redirected
        - In Contact List
        - Not In Contact List
      - Name Previously Assigned
      - Name Never Assigned
    - Domain Previously Assigned
      - Name Assigned
      - Name Previously Assigned
      - Name Never Assigned
    - Domain Never Assigned
      - Name Assigned
      - Name Previously Assigned
      - Name Never Assigned
  - Input Classes
    - Syntax
      - Syntax valid
      - Syntax invalid
        - Valid Syntax Domain
          - Invalid Syntax Name
        - Invalid Syntax Domain
          - Valid Syntax Name
          - Invalid Syntax Name
        - No @
        - Aggregate String too long
        - illegal text characters
        - control characters
        - binary objects
        - String way too long
        - Includes non-ascii characters
  - Memory Classes
    - To Field: Collection <String>
      - Email: String
        - within range
        - Larger than (2*32)-1



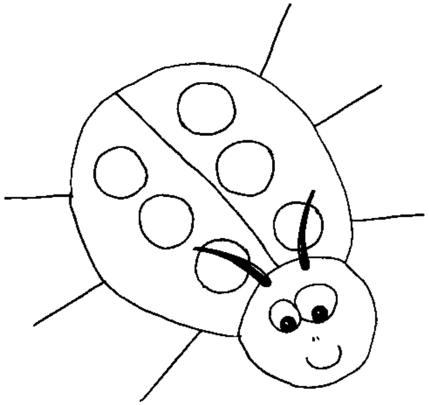
Example: Gmail To Composed Equivalence Classes Mind Map

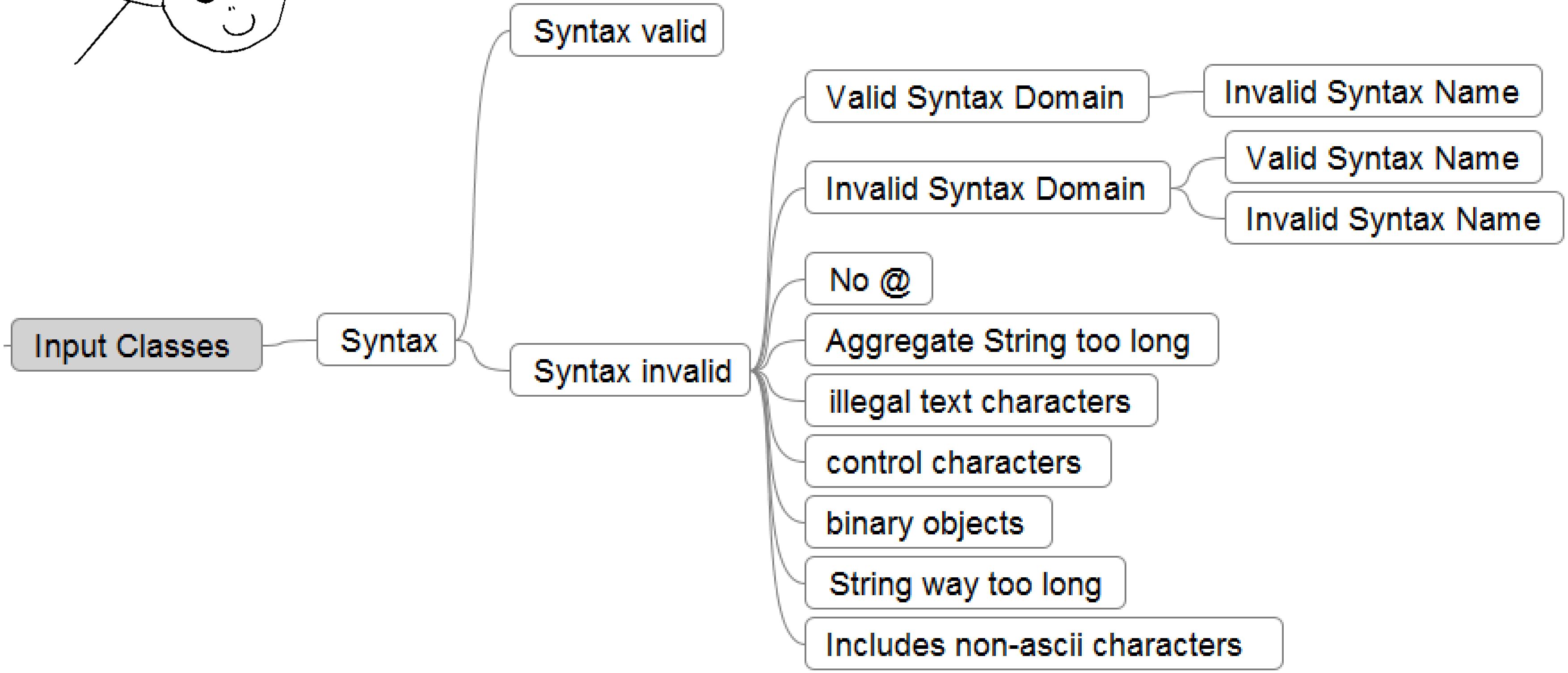


Application Logic

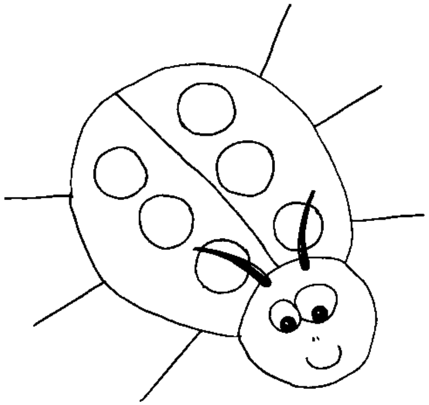
- Domain Assigned
  - Name Assigned
    - Redirected
    - Not Redirected
    - In Contact List
    - Not In Contact List
  - Name Previously Assigned
  - Name Never Assigned
- Domain Previously Assigned
  - Name Assigned
  - Name Previously Assigned
  - Name Never Assigned
- Domain Never Assigned
  - Name Assigned
  - Name Previously Assigned
  - Name Never Assigned

Example: Gmail To Composed Equivalence Classes Mind Map

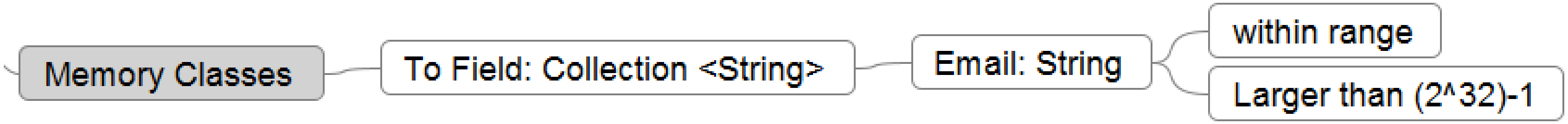




- Input Classes
  - Syntax
    - Syntax valid
    - Syntax invalid
      - Valid Syntax Domain
        - Invalid Syntax Name
      - Invalid Syntax Domain
        - Valid Syntax Name
        - Invalid Syntax Name
      - No @
      - Aggregate String too long
      - illegal text characters
      - control characters
      - binary objects
      - String way too long
      - Includes non-ascii characters



Example: Gmail To Composed Equivalence Classes Mind Map



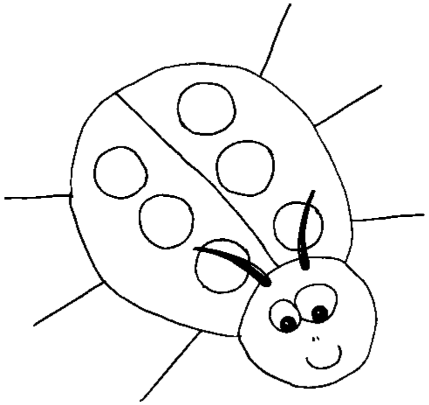
Memory Classes

To Field: Collection <String>

Email: String

within range

Larger than (2^32)-1

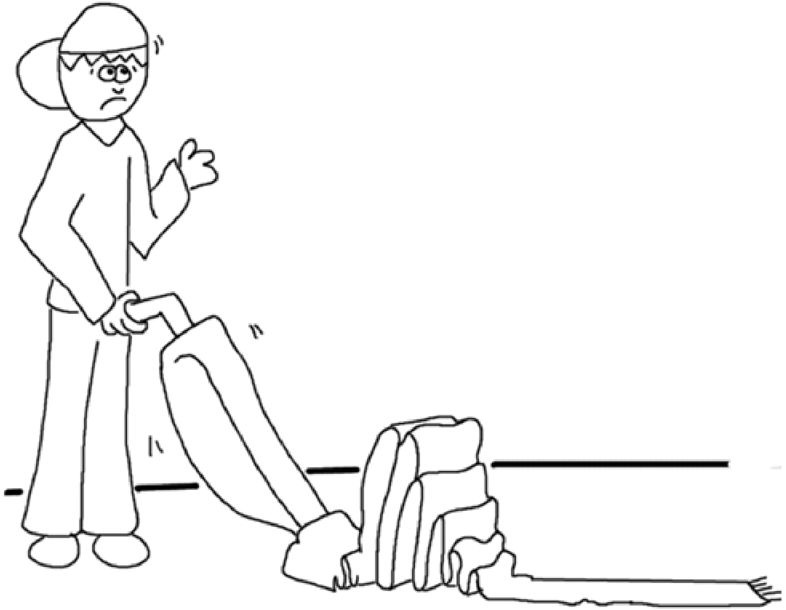


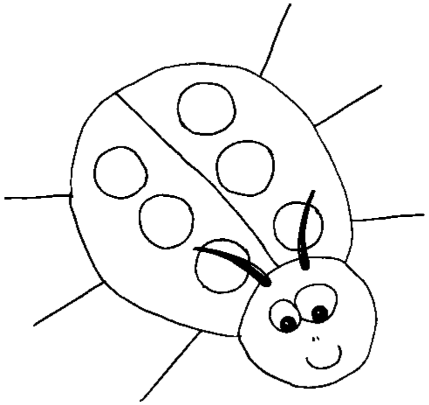
Test Design

Boundaries

Boundary Testing

- Classes with continuous ranges of values
  - Test around extremes
  - Lower & upper boundaries
  - Edge conditions





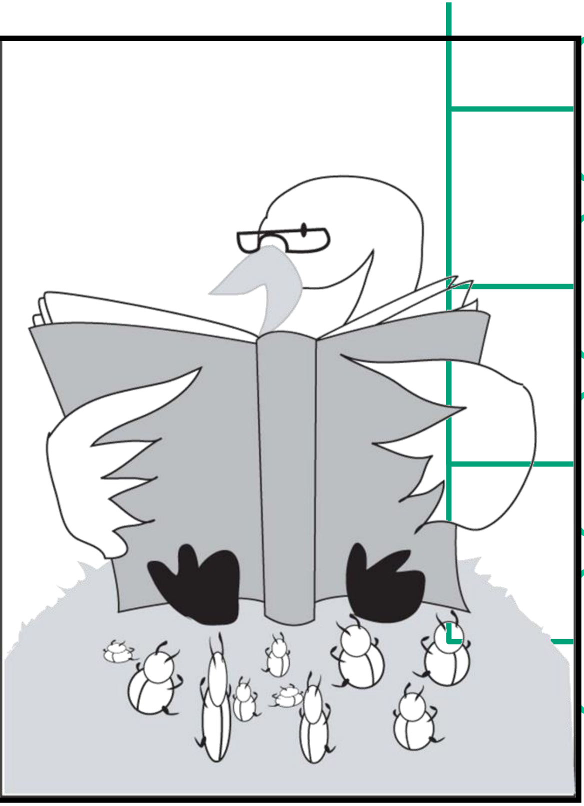
Test Design

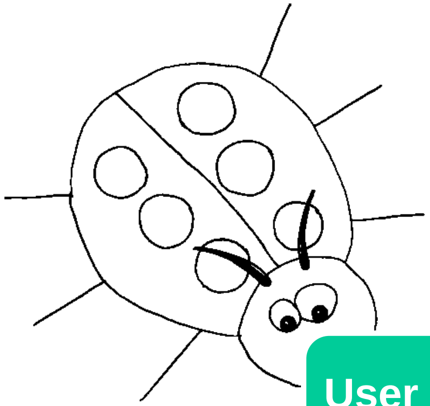
Agile Story Acceptance Tests

Story Testing

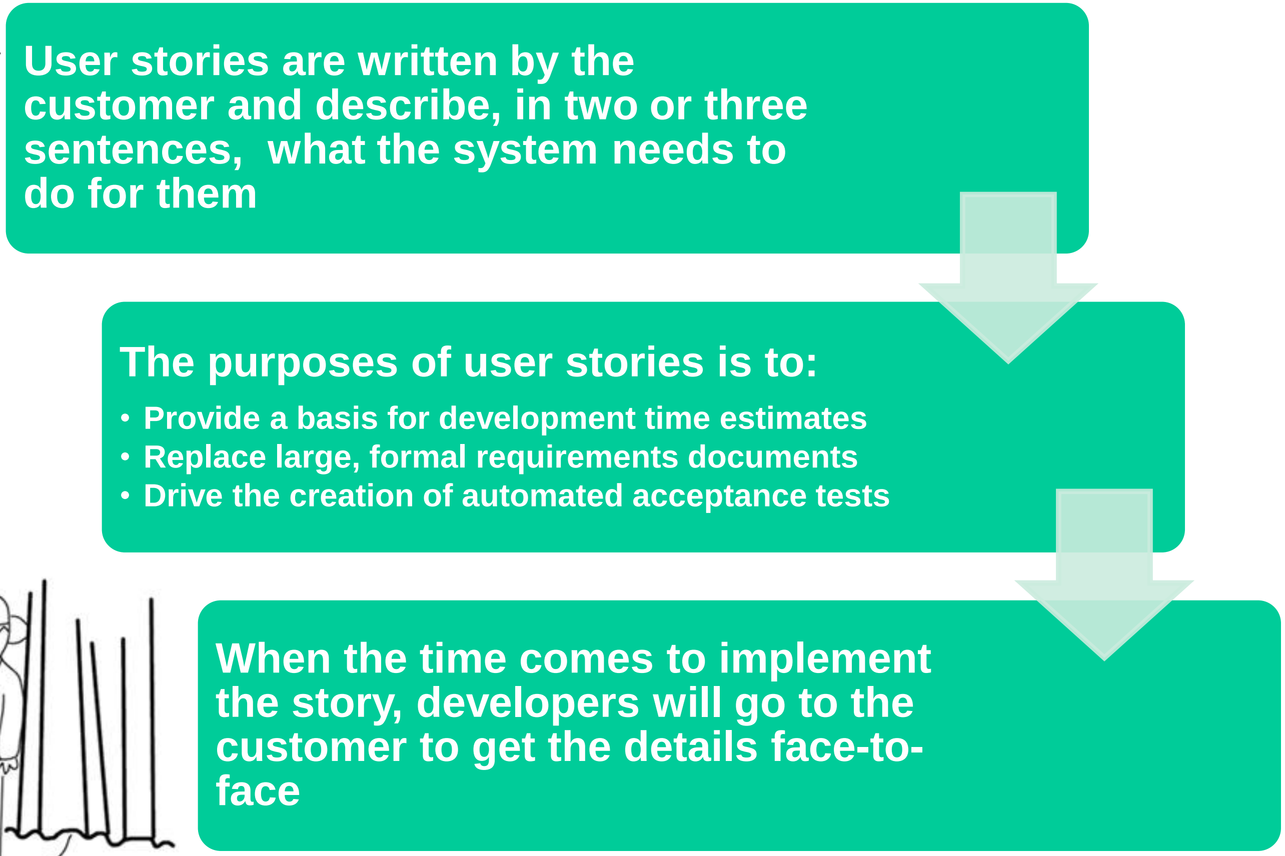
Story Tests

- Elicited from customer
- Clear examples
- Confirm implementation
- Demonstrate correctness





Story Testing

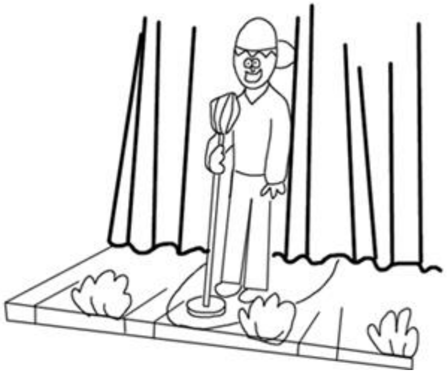


User stories are written by the customer and describe, in two or three sentences, what the system needs to do for them

The purposes of user stories is to:

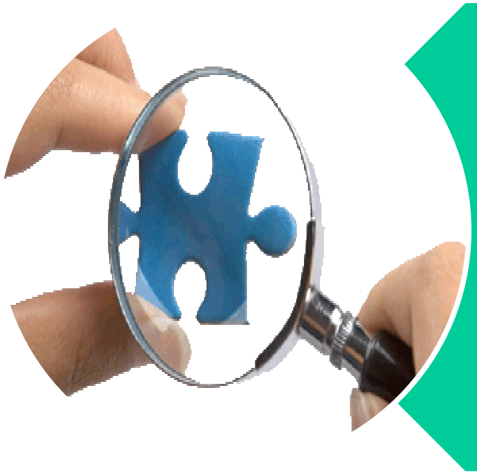
- Provide a basis for development time estimates
- Replace large, formal requirements documents
- Drive the creation of automated acceptance tests

When the time comes to implement the story, developers will go to the customer to get the details face-to-face

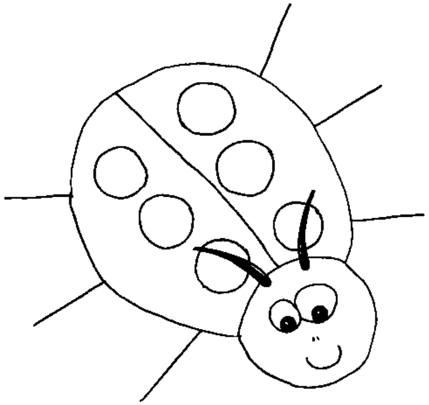


A User Story answers these questions:

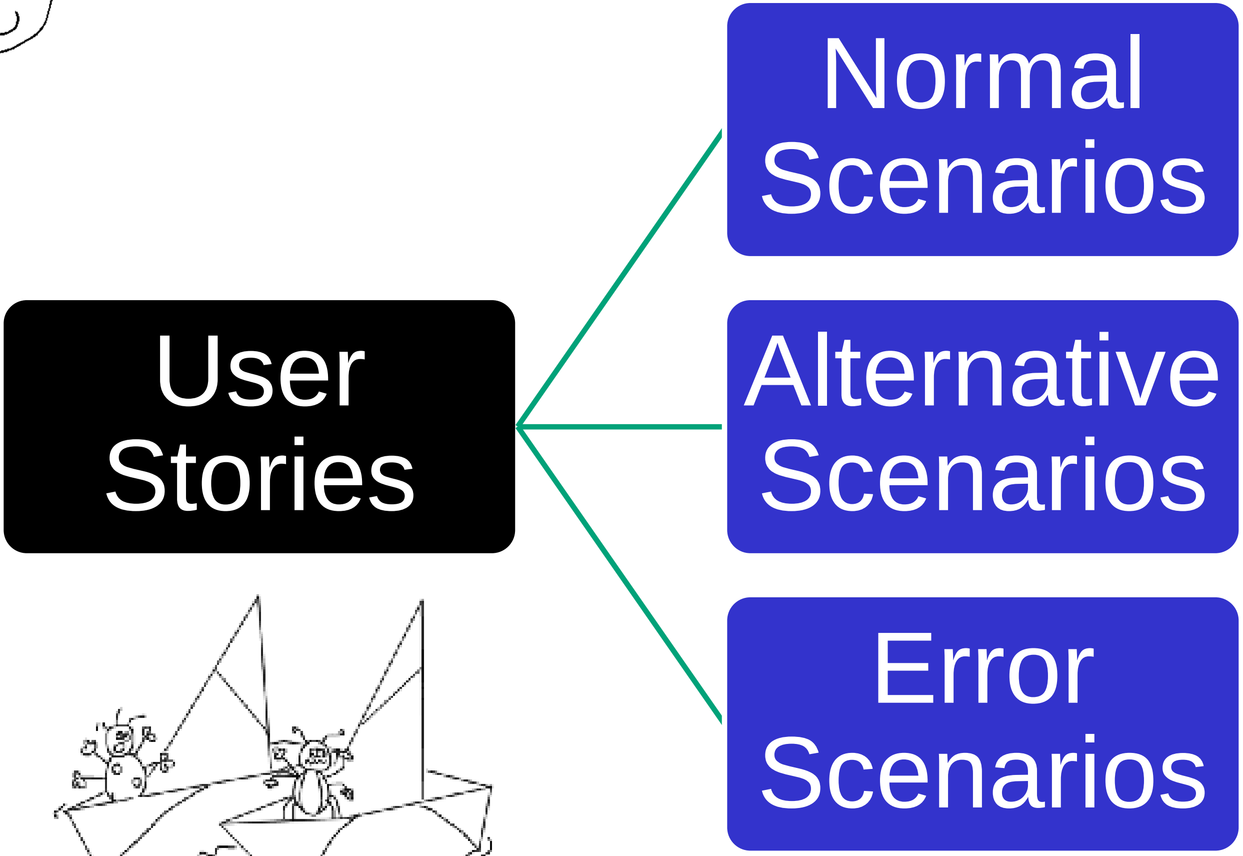
- Who is the user?
- What do they want to do?
- For what benefit?



User Story Tests provide examples of different scenarios to confirm understanding of typical, alternate and error situations a user may encounter



Story Testing

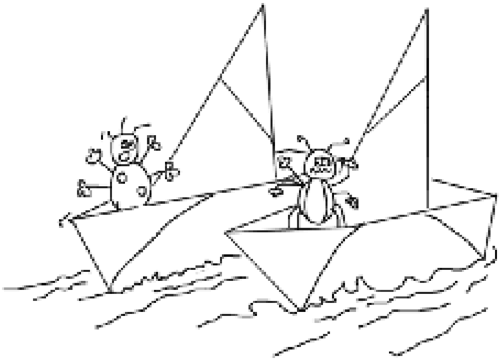


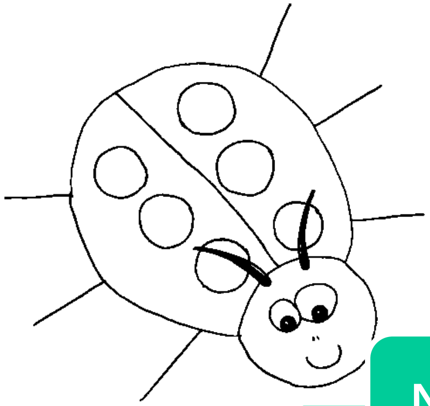
User Stories

Normal Scenarios

Alternative Scenarios

Error Scenarios

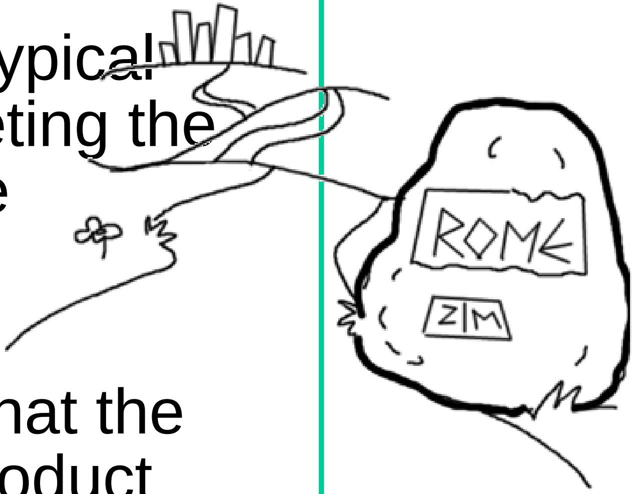




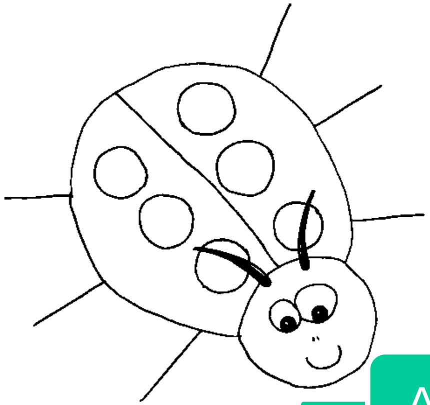
Story Testing

Normal Flows

- Define scenarios which are typical examples of the user completing the story with a positive outcome



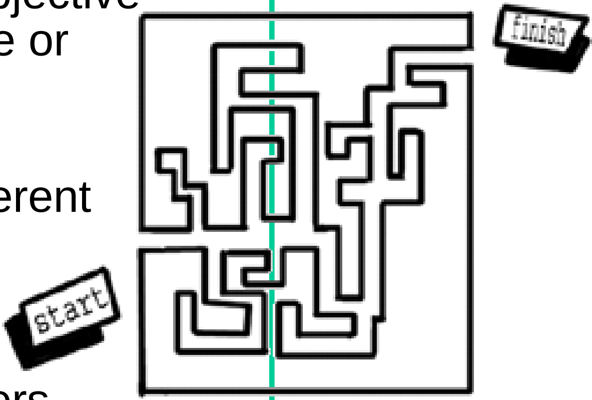
- A normal flow is a clear representative example of what the customer really needs the product to do

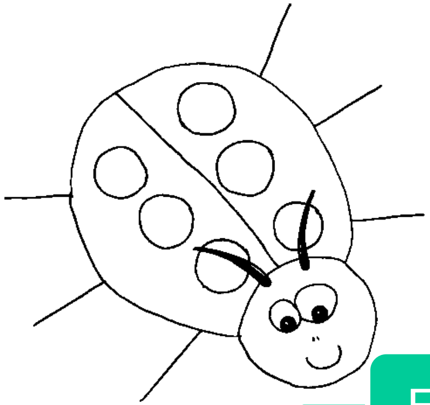


Story Testing

Alternate Flows

- Define scenarios which are examples of the user completing the story with a positive outcome but with a variation in the path to achieve the objective due to differences in user date, system state or other factors
- Alternative flows help explore the many different ways a user can accomplish the story
- Elaborating alternative flows improves implementation estimates and clears up users understanding of the user stories scope

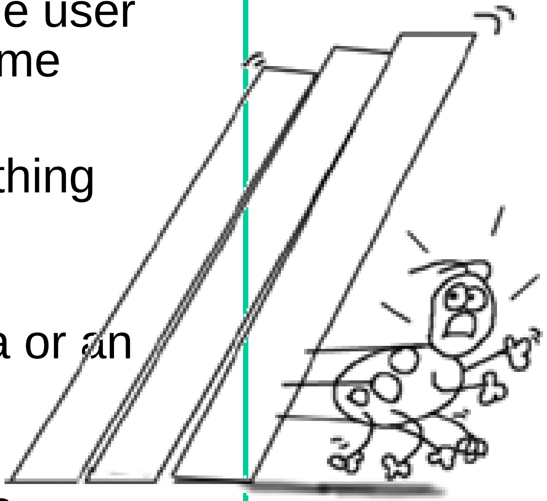


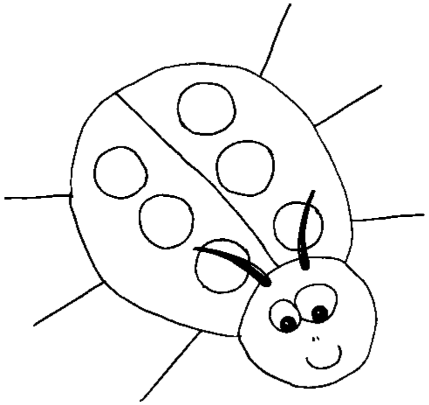


Story Testing

Error Flows

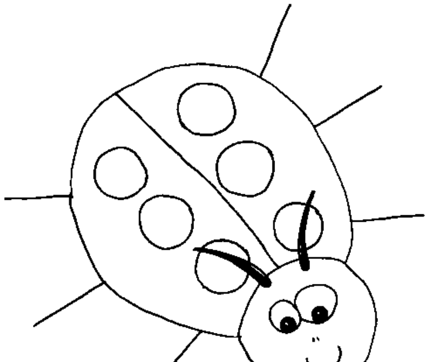
- Define scenarios which are examples of the user completing the story with a negative outcome
- These are alternative flows in which something goes has gone wrong
- Error flows can be triggered by invalid data or an inappropriate system state
- Eliciting error flows helps the implementers understand the type of error handling expected



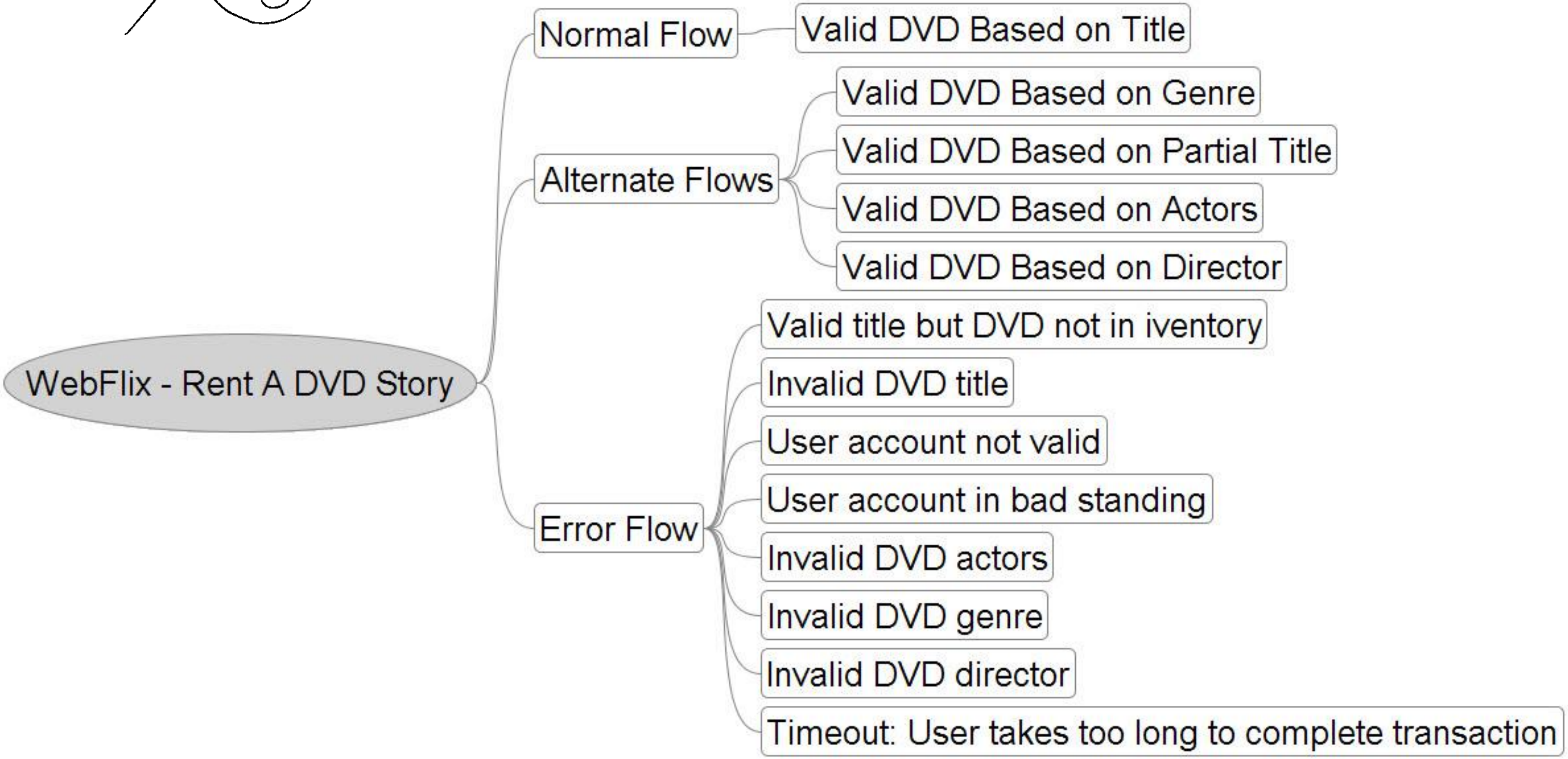


WebFlix DVD Rental

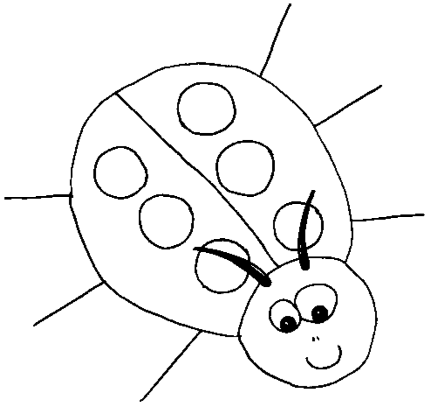
Story Testing



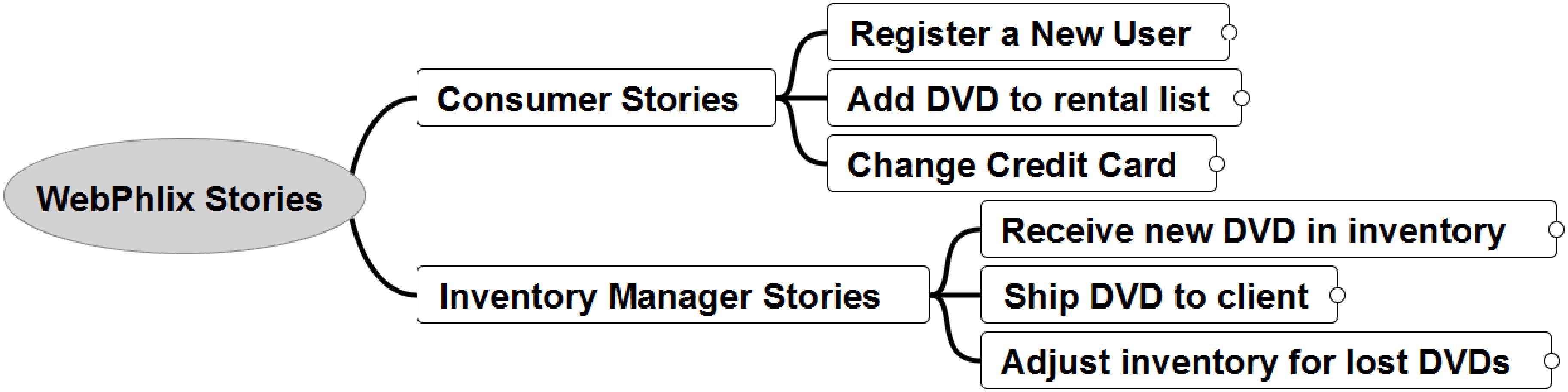
Story Testing



- WebFlix - Rent A DVD Story
  - Normal Flow
    - Valid DVD Based on Title
  - Alternate Flows
    - Valid DVD Based on Genre
    - Valid DVD Based on Partial Title
    - Valid DVD Based on Actors
    - Valid DVD Based on Director
  - Error Flow
    - Valid title but DVD not in iventory
    - Invalid DVD title
    - User account not valid
    - User account in bad standing
    - Invalid DVD actors
    - Invalid DVD genre
    - Invalid DVD director
    - Timeout: User takes too long to complete transaction



Story Testing



WebPhlix Stories

Consumer Stories

Register a New User

Add DVD to rental list

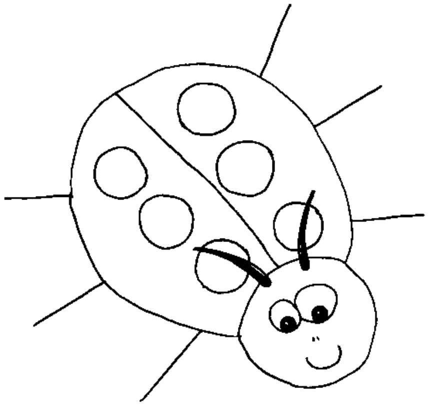
Change Credit Card

Inventory Manager Stories

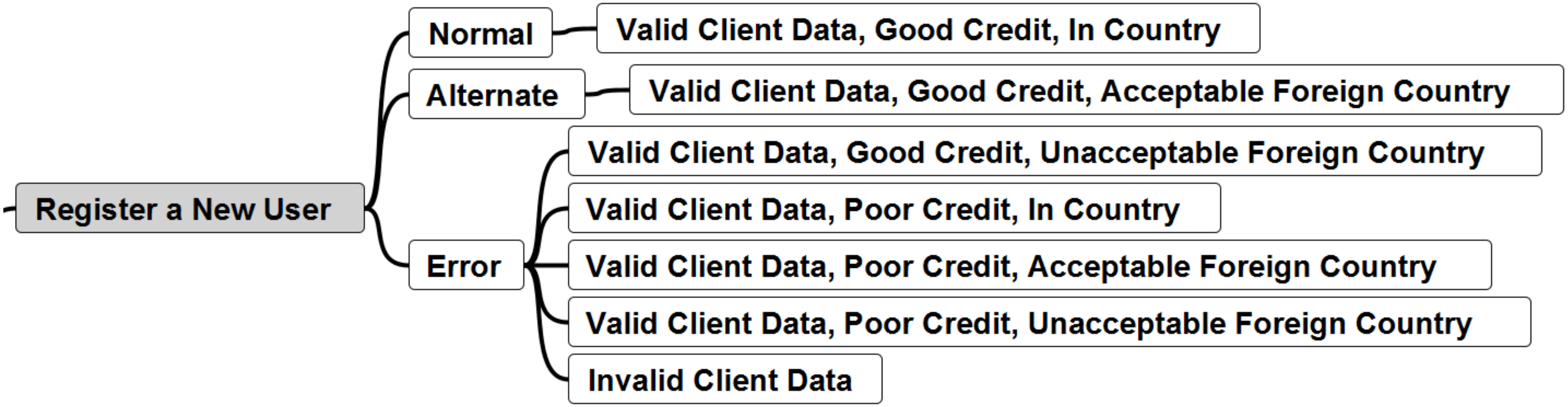
Receive new DVD in inventory

Ship DVD to client

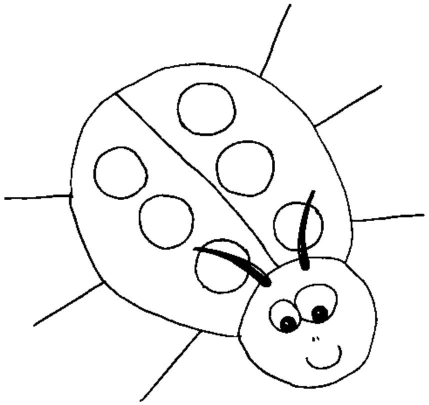
Adjust inventory for lost DVDs



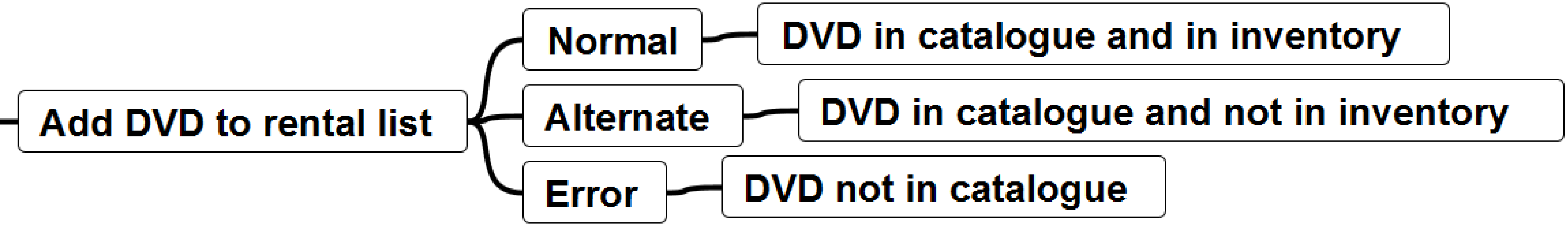
Story Testing



- Register a New User
  - Normal
    - Valid Client Data, Good Credit, In Country
  - Alternate
    - Valid Client Data, Good Credit, Acceptable Foreign Country
  - Error
    - Valid Client Data, Good Credit, Unacceptable Foreign Country
    - Valid Client Data, Poor Credit, In Country
    - Valid Client Data, Poor Credit, Acceptable Foreign Country
    - Valid Client Data, Poor Credit, Unacceptable Foreign Country
    - Invalid Client Data



Story Testing



Add DVD to rental list

Normal

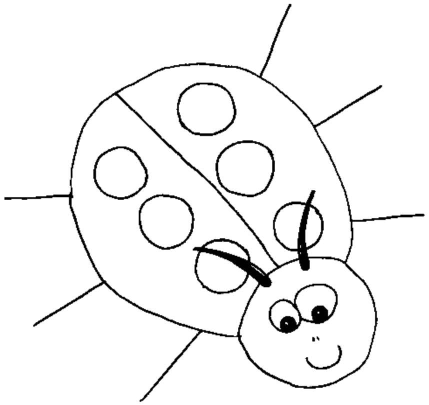
DVD in catalogue and in inventory

Alternate

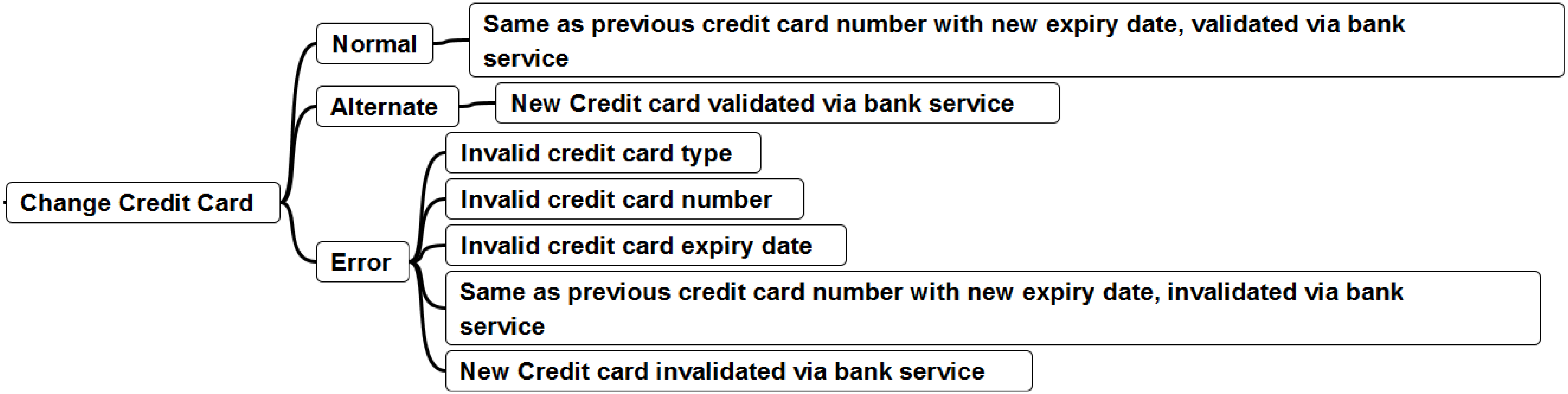
DVD in catalogue and not in inventory

Error

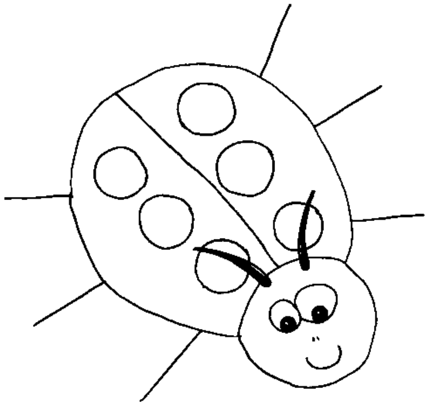
DVD not in catalogue



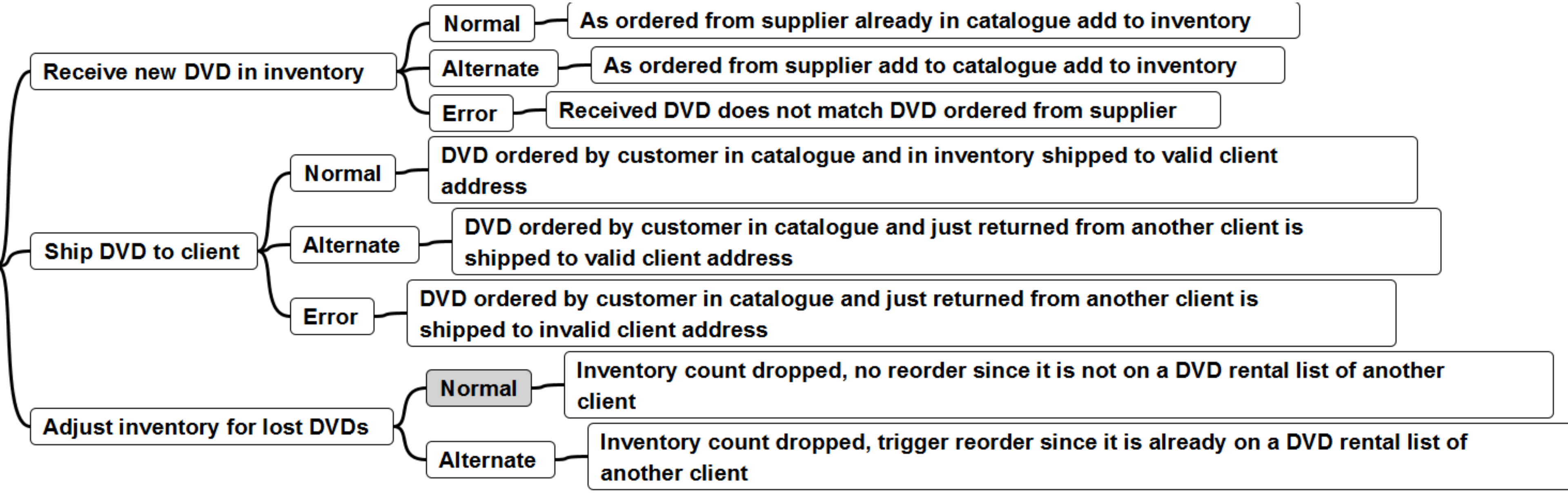
Story Testing



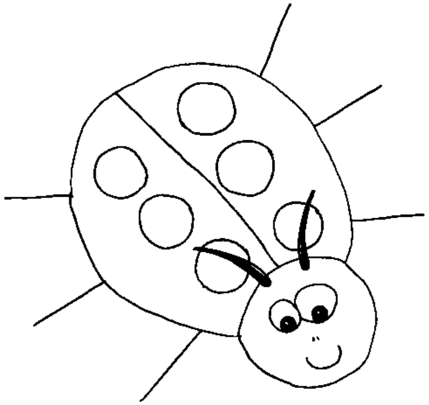
- Change Credit Card
  - Normal
    - Same as previous credit card number with new expiry date, validated via bank service
  - Alternate
    - New Credit card validated via bank service
  - Error
    - Invalid credit card type
    - Invalid credit card number
    - Invalid credit card expiry date
    - Same as previous credit card number with new expiry date, invalidated via bank service
    - New Credit card invalidated via bank service



Story Testing

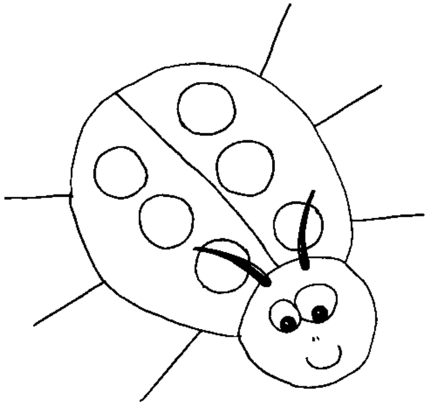


- Receive new DVD in inventory
  - Normal
    - As ordered from supplier already in catalogue add to inventory
  - Alternate
    - As ordered from supplier add to catalogue add to inventory
  - Error
    - Received DVD does not match DVD ordered from supplier
- Ship DVD to client
  - Normal
    - DVD ordered by customer in catalogue and in inventory shipped to valid client address
  - Alternate
    - DVD ordered by customer in catalogue and just returned from another client is shipped to valid client address
  - Error
    - DVD ordered by customer in catalogue and just returned from another client is shipped to invalid client address
- Adjust inventory for lost DVDs
  - Normal
    - Inventory count dropped, no reorder since it is not on a DVD rental list of another client
  - Alternate
    - Inventory count dropped, trigger reorder since it is already on a DVD rental list of another client

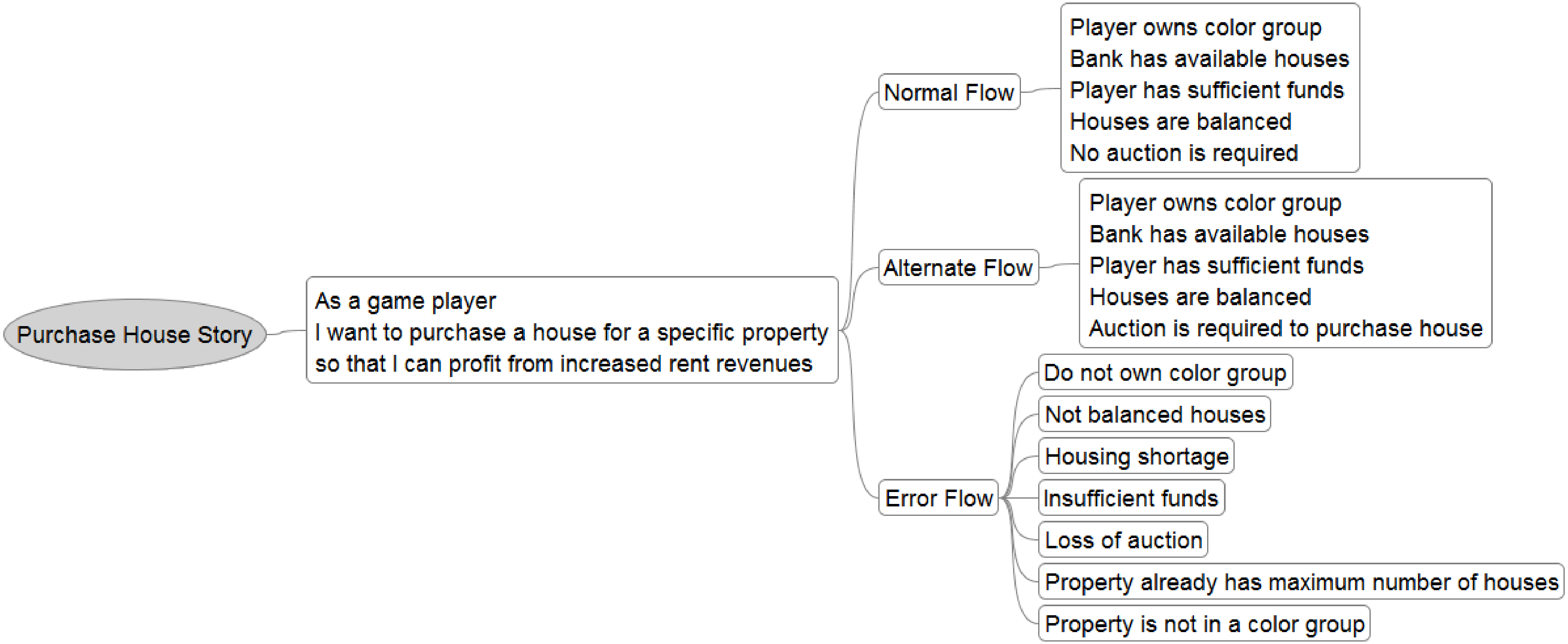


Monopoly® Game

Story Testing



Story Testing



Purchase House Story

As a game player
I want to purchase a house for a specific property
so that I can profit from increased rent revenues

Normal Flow

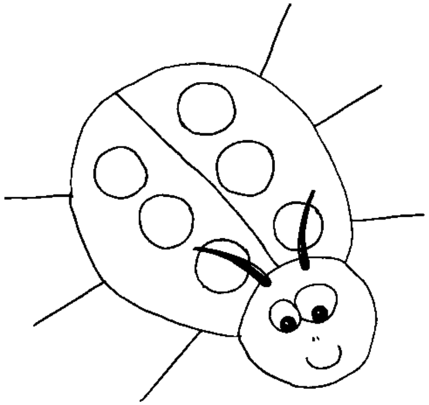
Player owns color group
Bank has available houses
Player has sufficient funds
Houses are balanced
No auction is required

Alternate Flow

Player owns color group
Bank has available houses
Player has sufficient funds
Houses are balanced
Auction is required to purchase house

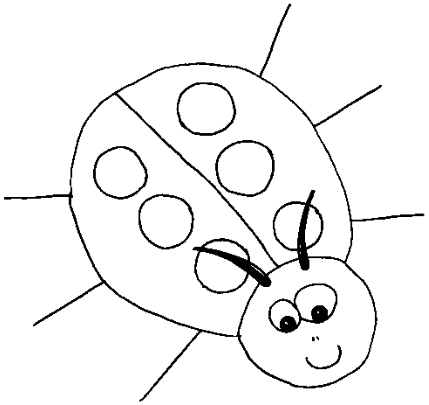
Error Flow

- Do not own color group
- Not balanced houses
- Housing shortage
- Insufficient funds
- Loss of auction
- Property already has maximum number of houses
- Property is not in a color group



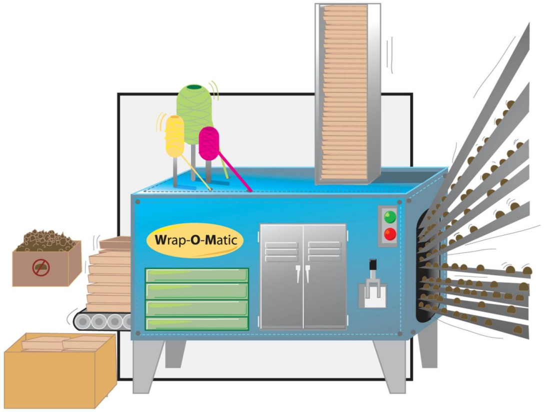
Test Design

Usage Scenario Tests



Scenario Based Testing

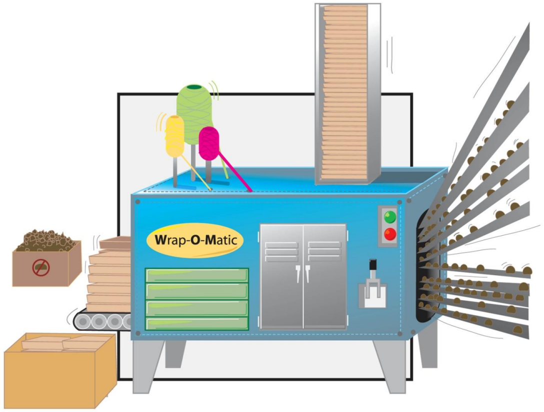
- Scenarios
  - Typical, *real*, usage scenarios for each user type
    - *Could be based on Use Case Analysis or Story Boards*
  - Individual scenarios can cover many functions!

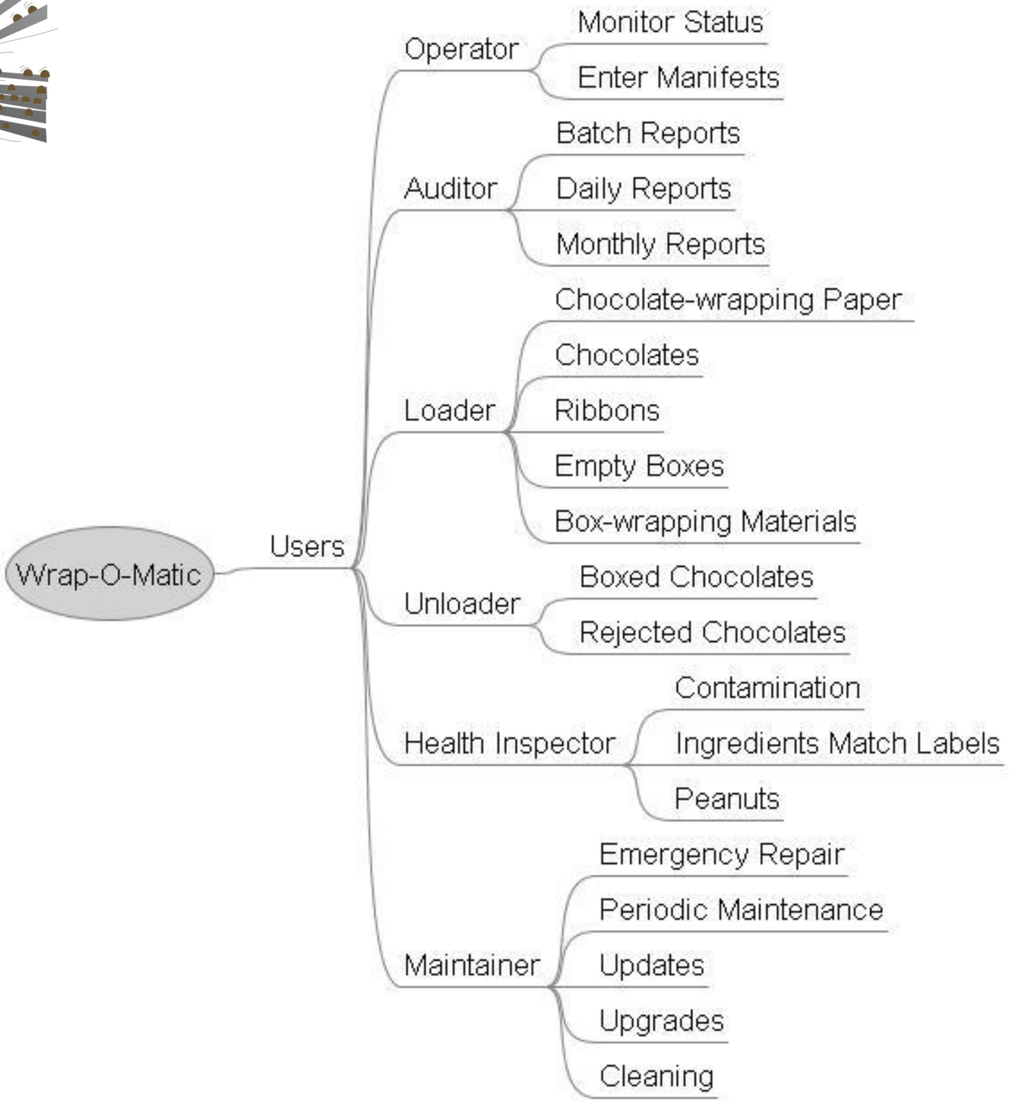


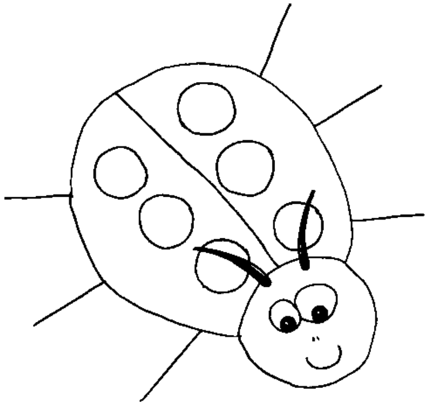
Wrap-O-Matic

Usage Scenarios

Scenario Based Testing

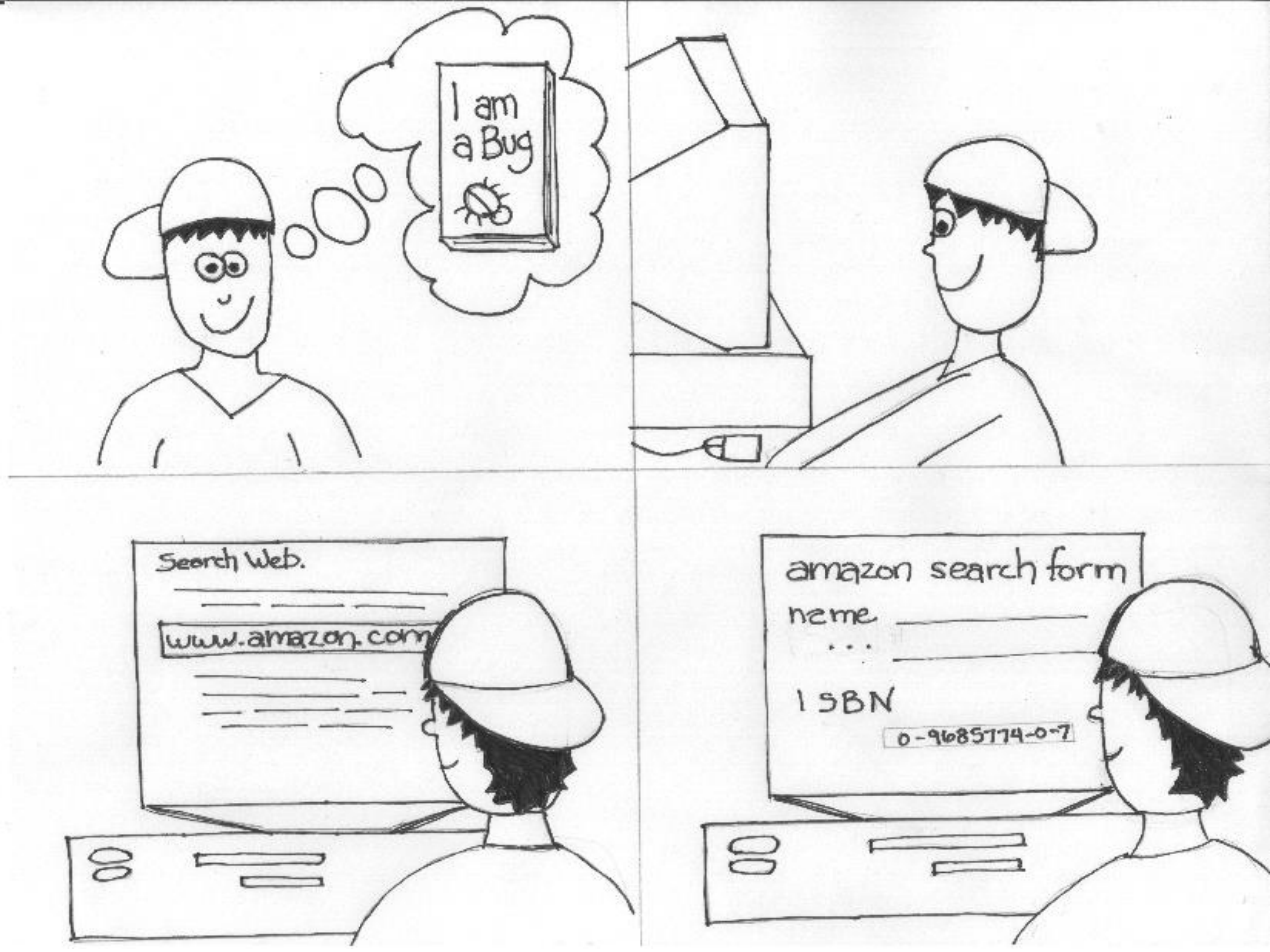


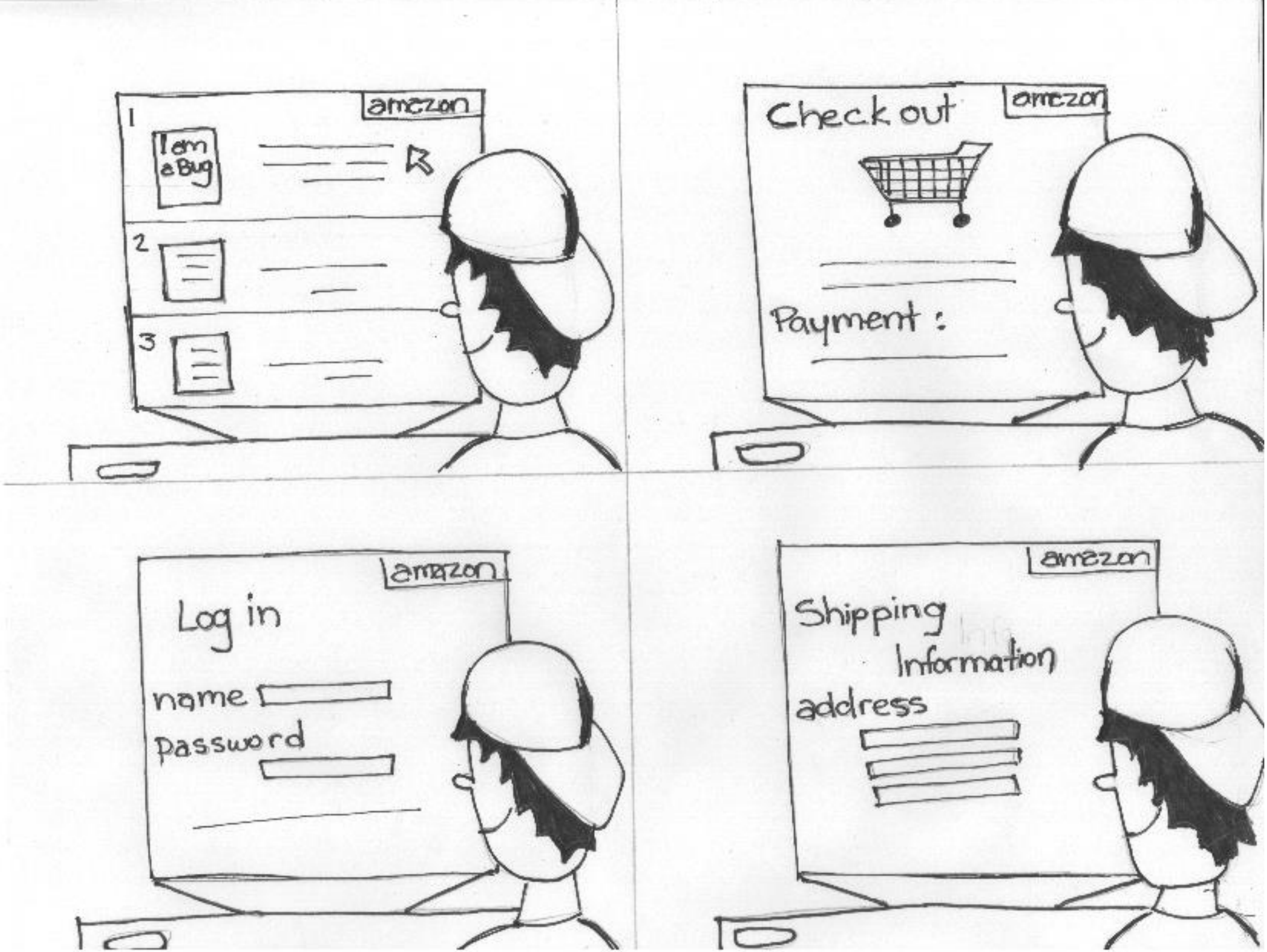


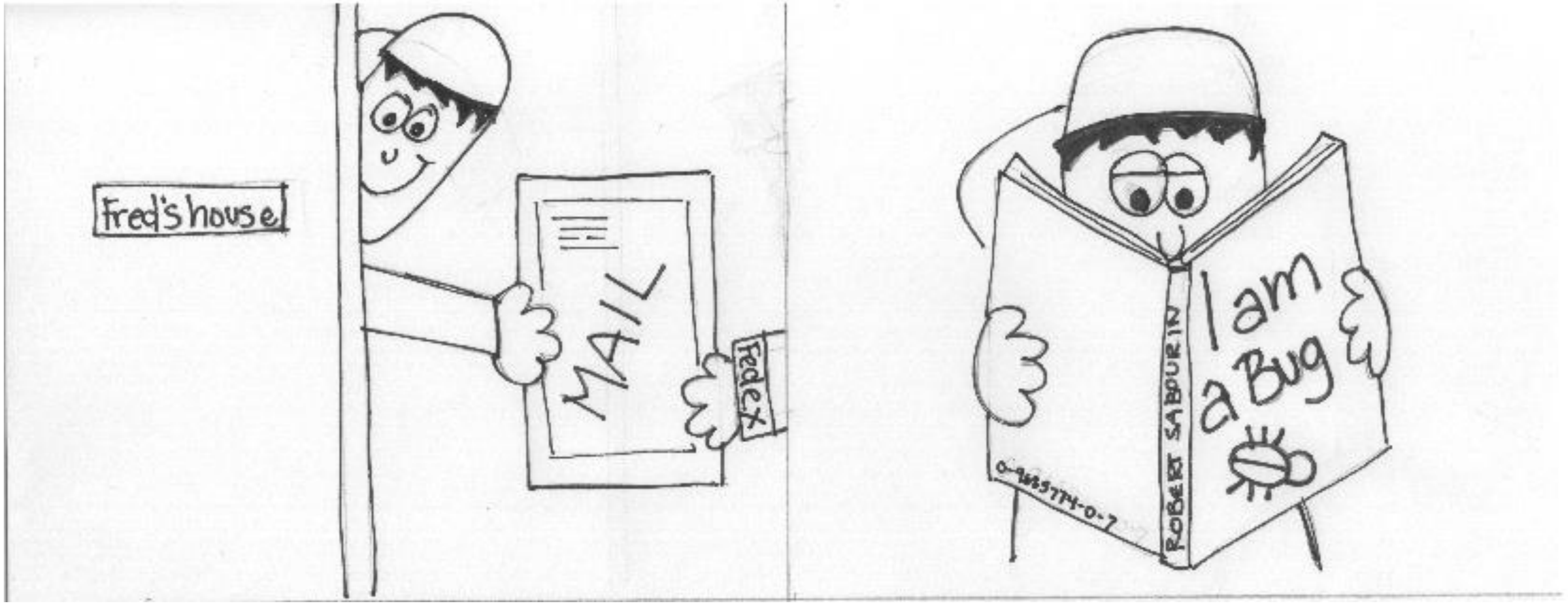


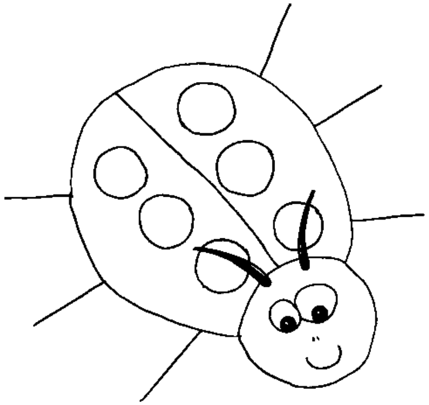
Buying a Book

Usage Scenarios


I am a Bug
Search Web.
www.amazon.com
amazon search form
name
...
ISBN
0-9685774-0-7

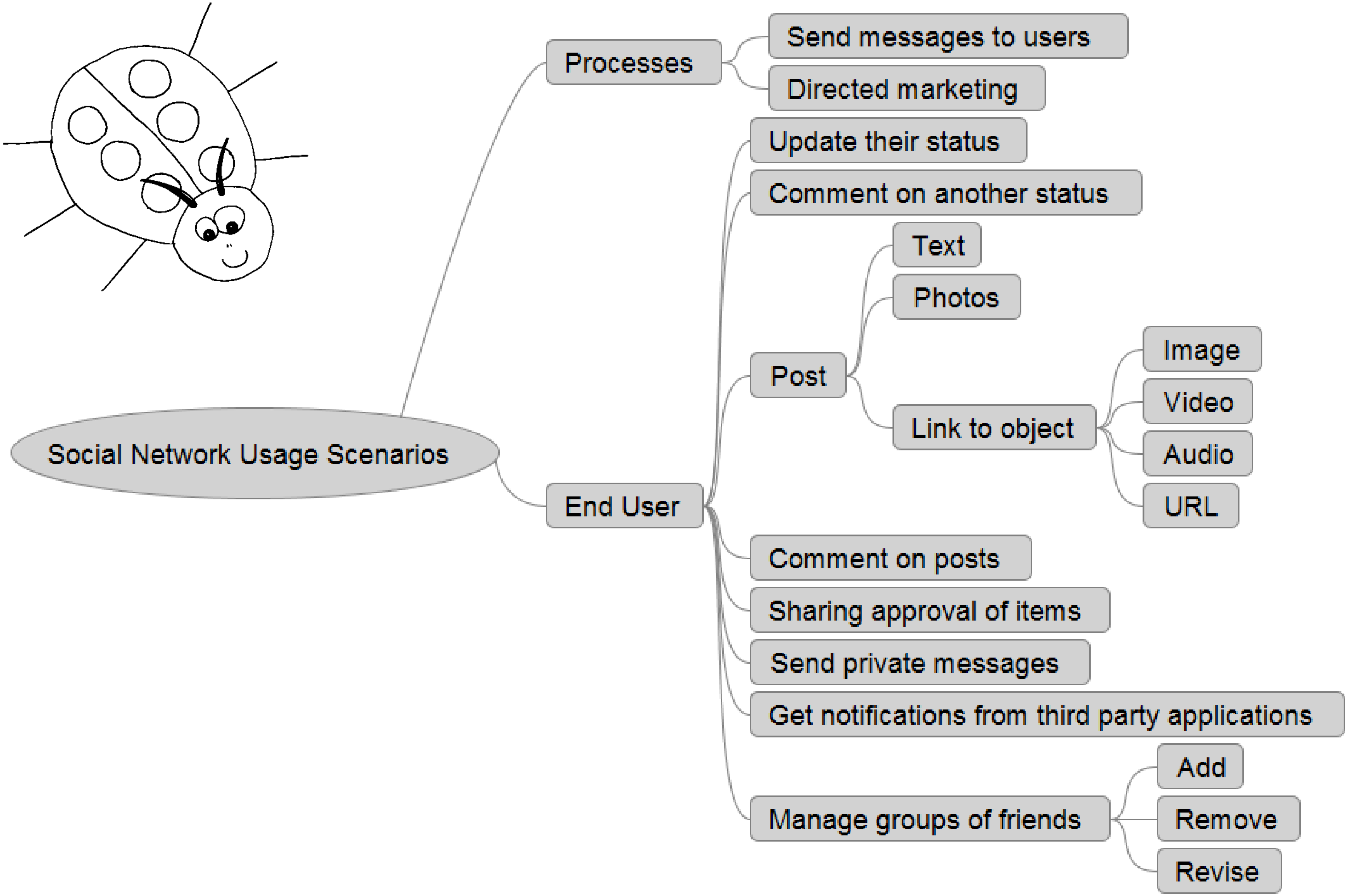

amazon
1
I am a Bug
2
3
amezon
Check out
Payment :
amazon
Log in
name
password
amazon
Shipping
Information
address


Fred's house
MAIL
Fedex
ROBERT SABOURIN
I am a Bug
0-9685774-0-7

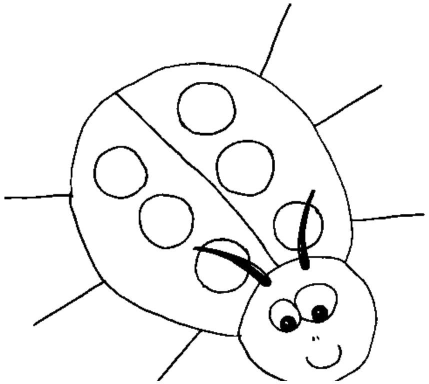


Social Networking

Usage Scenarios



- Social Network Usage Scenarios
  - Processes
    - Send messages to users
    - Directed marketing
  - End User
    - Update their status
    - Comment on another status
    - Post
      - Text
      - Photos
      - Link to object
        - Image
        - Video
        - Audio
        - URL
    - Comment on posts
    - Sharing approval of items
    - Send private messages
    - Get notifications from third party applications
    - Manage groups of friends
      - Add
      - Remove
      - Revise



Scenario Based Testing

Sharing a Picture With a Group of Friends

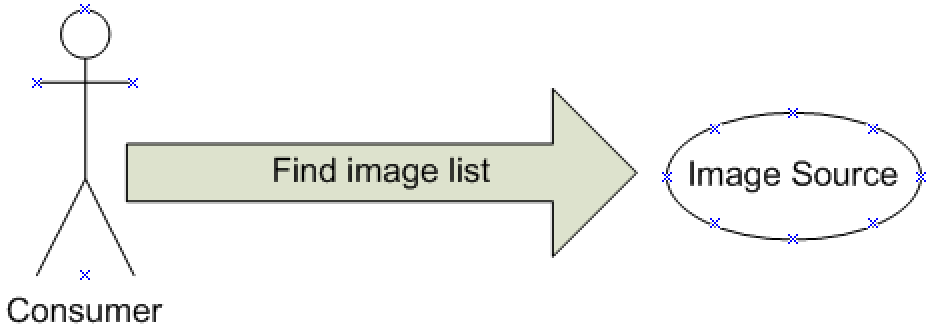
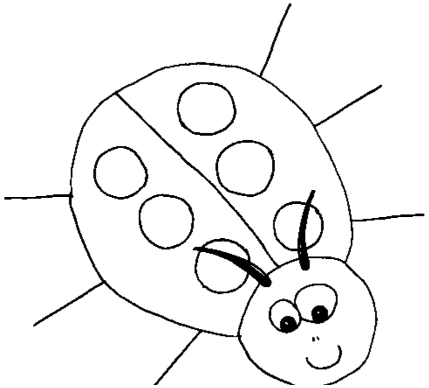
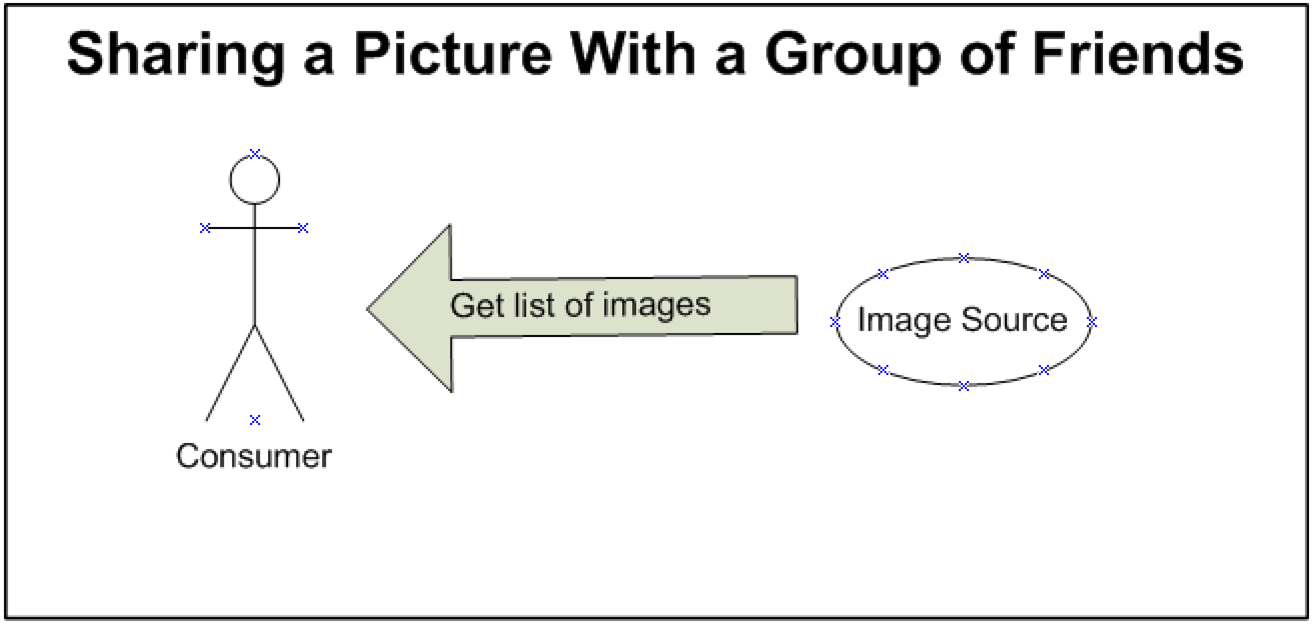


Image Sources

- Different sources
- Local
- Camera
- Phone
- Memory card
- USB stick

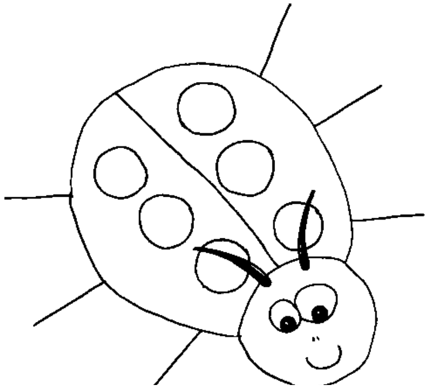


Scenario Based Testing



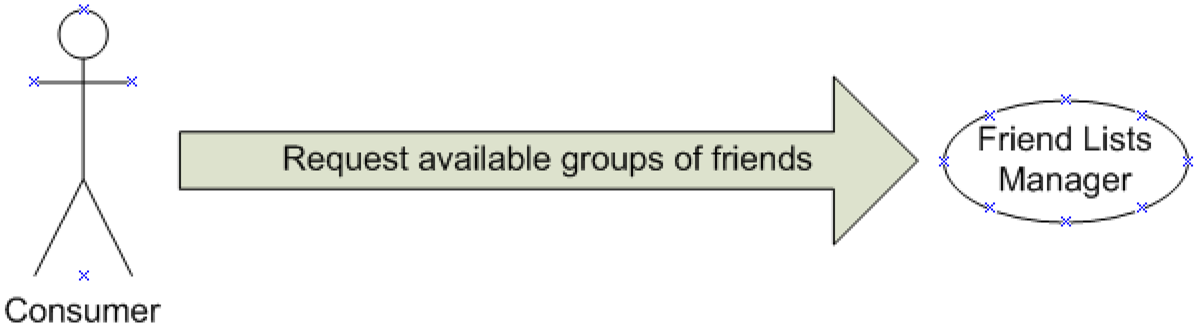
Images

- Image formats
- Resolutions
- Sizes
- Colors
- None
- Many

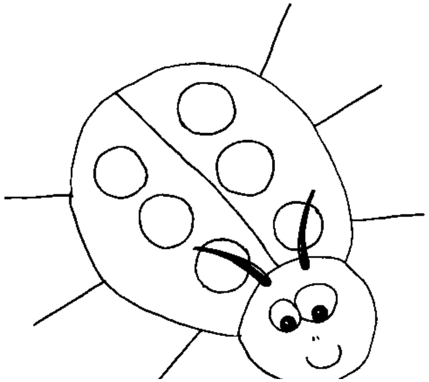


Scenario Based Testing

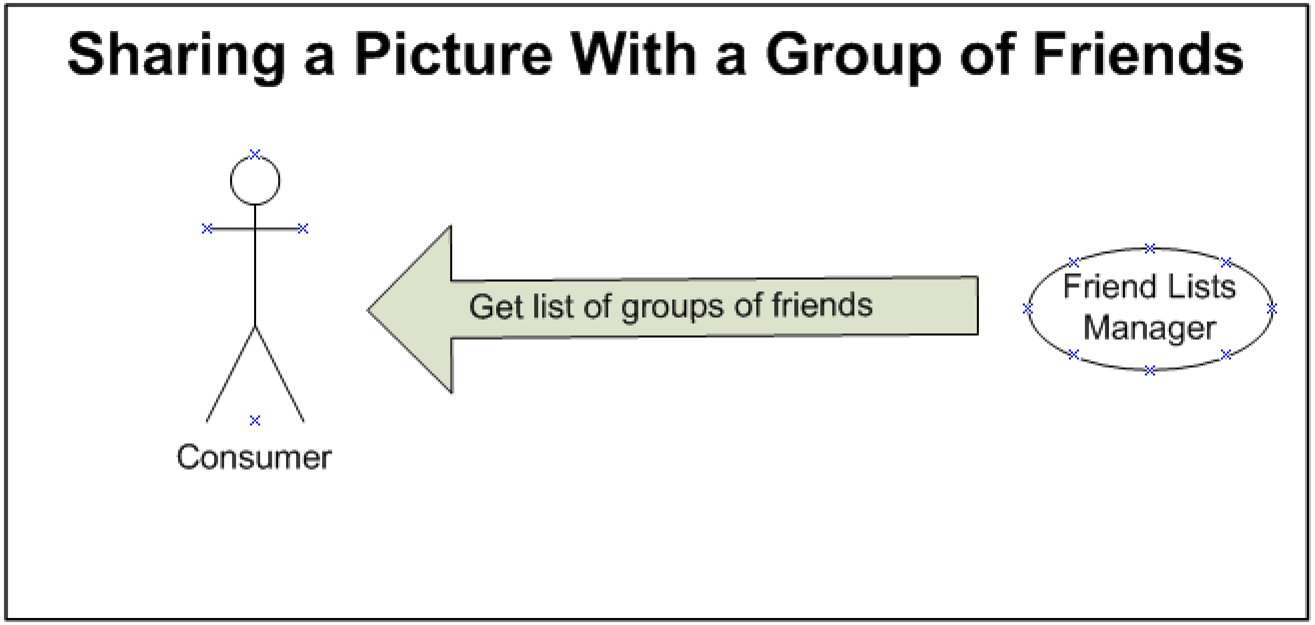
Sharing a Picture With a Group of Friends



Lists of
- Private
- Public



Scenario Based Testing



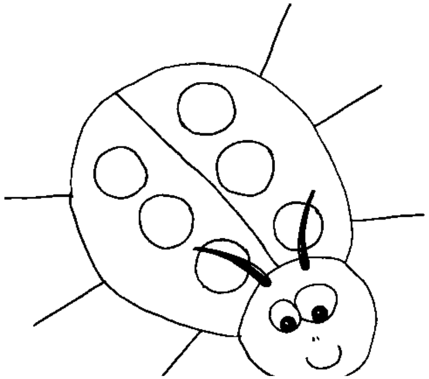
Sharing a Picture With a Group of Friends

Consumer

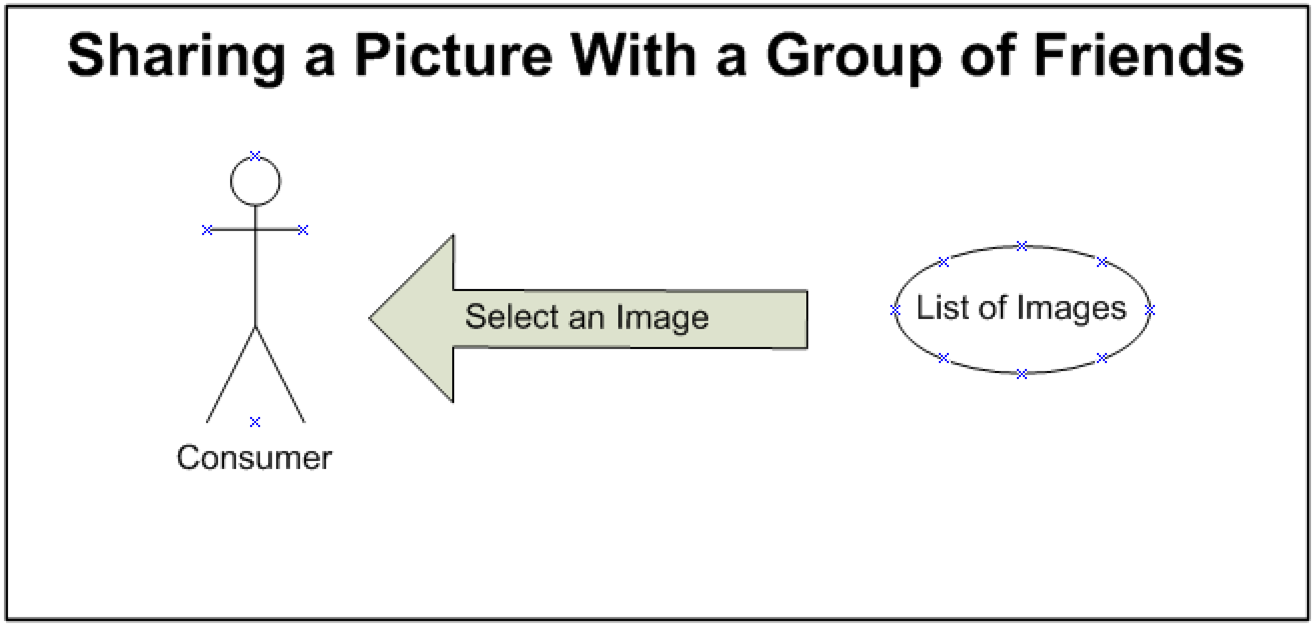
Get list of groups of friends

Friend Lists Manager

Lists
- None
- Many



Scenario Based Testing



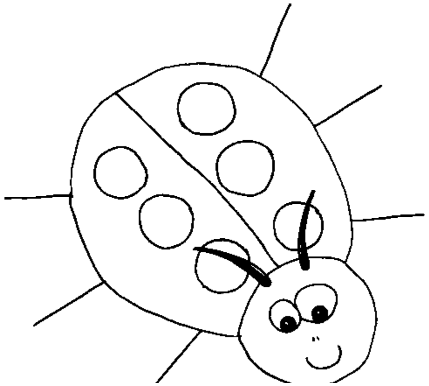
Sharing a Picture With a Group of Friends

Consumer

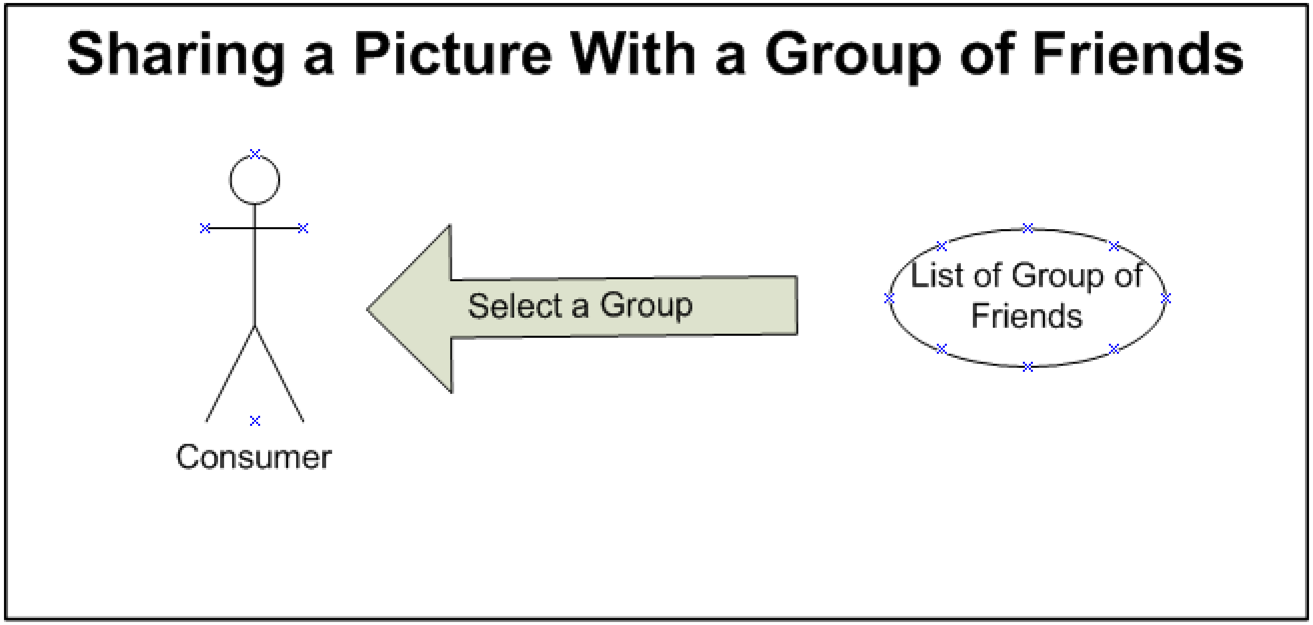
Select an Image

List of Images

Select by
- Name
- Type
- Preview
- One
- None
- Many

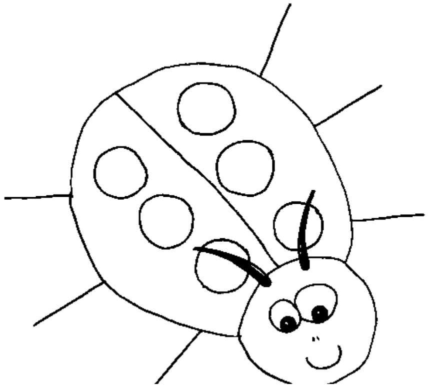


Scenario Based Testing

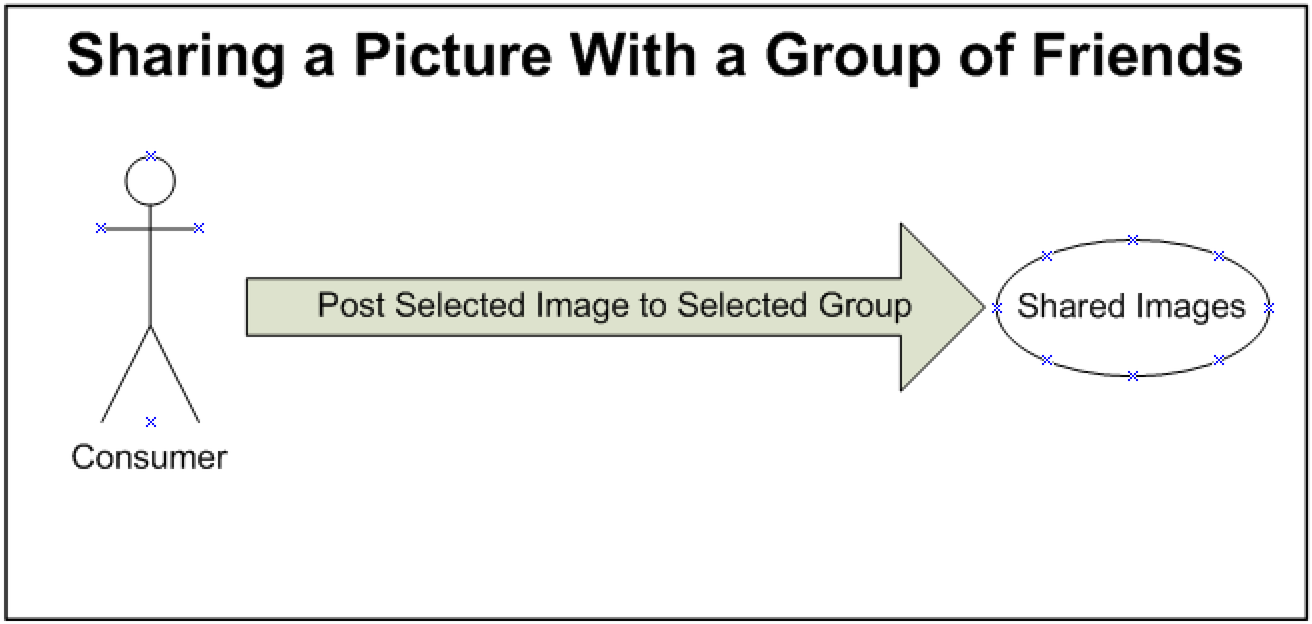


Sharing a Picture With a Group of Friends

Select by
- Name
- Membership
- Custom made for sharing this picture



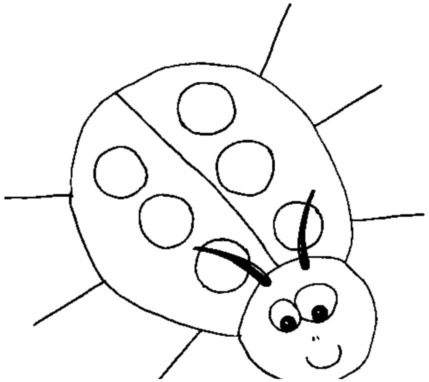
Scenario Based Testing



Sharing a Picture With a Group of Friends

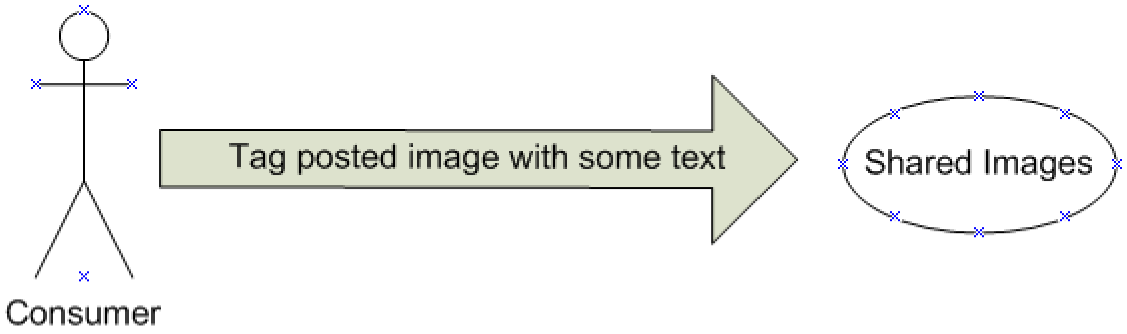
Posted

- Single image
- Multiple images
- Single group
- Multiple groups



Scenario Based Testing

Sharing a Picture With a Group of Friends



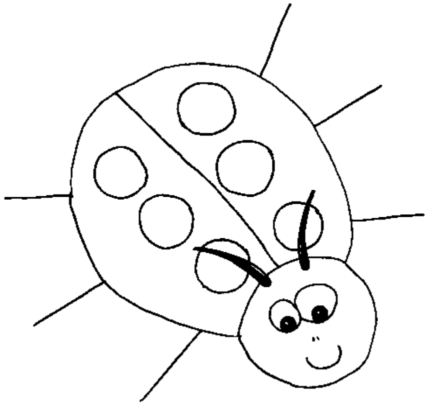
Consumer

Tag posted image with some text

Shared Images

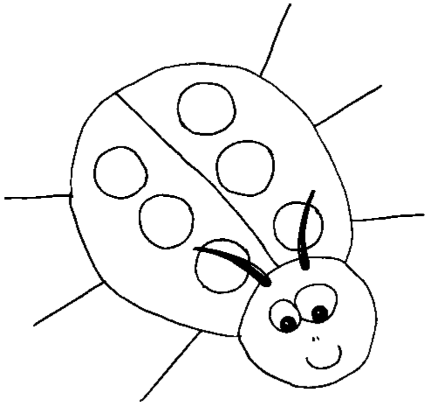
Tagged

- No text
- Some text
- Link
- Different text for different image
- Different text for different groups



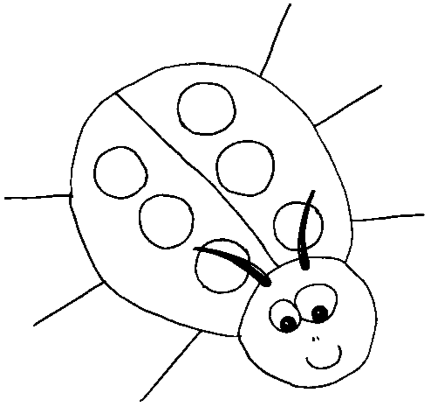
Scenario Based Testing

- User Experience Test Case
  - ❑Parameterize experience
  - ❑Walk through scenario from start to end
  - ❑Use pre selected input for each case
  - ❑Always run every test as if it were a user experience



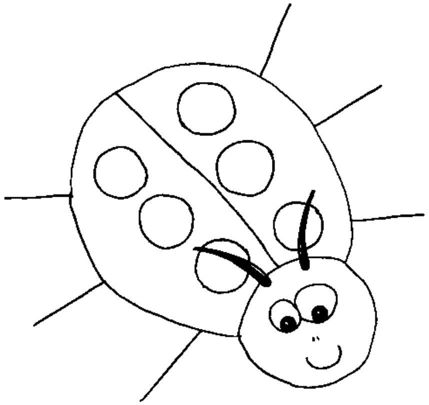
TOFT Testing

- Task Oriented Functional Testing
  - Can the user accomplish useful tasks correctly?



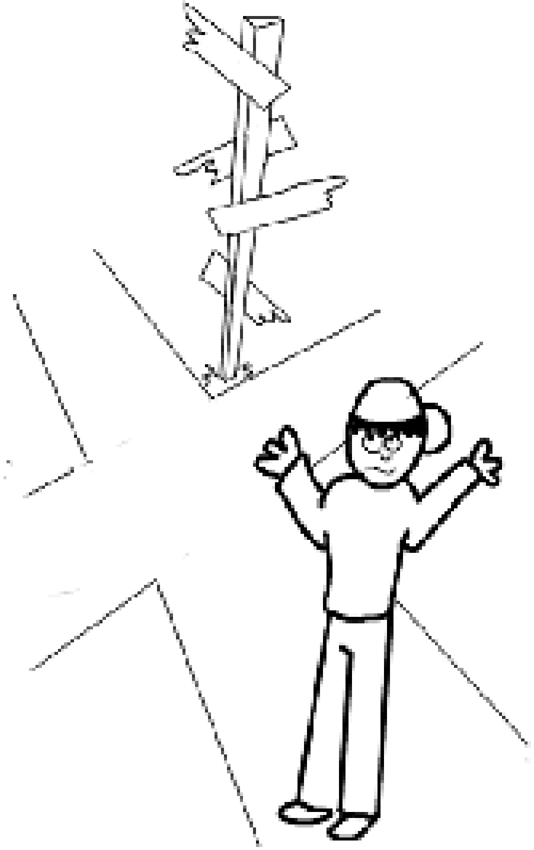
Test Design

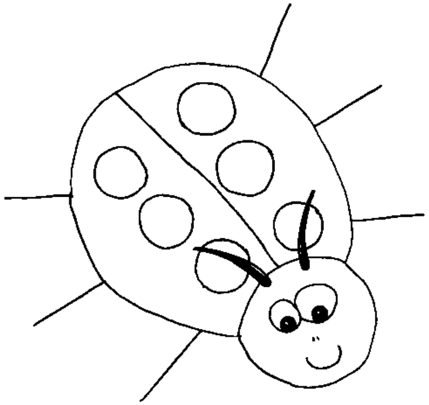
Control Flow Testing



Control Flow Testing

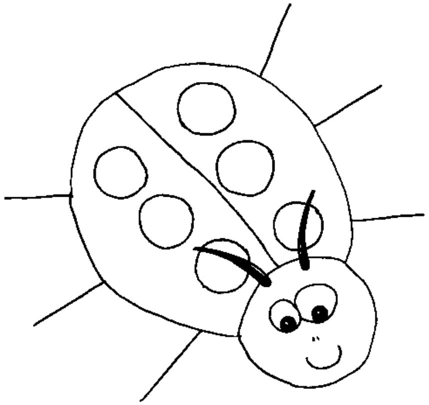
- Exercise paths through a system
  - Data Flow
  - Transaction Flow
  - Code Flow
  - Process Flow





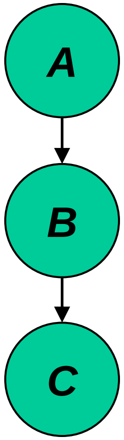
Control Flow Testing

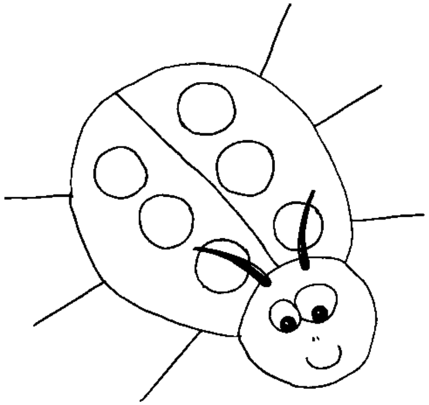
- Model flow
  - – Create control flow diagram
  - – Find basis paths
    - Minimal set of transactions
    - Exercise at least once
      - – Each step
      - – Each decision



Control Flow Testing

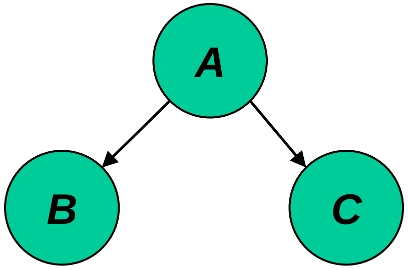
- Process Steps
  - A flows to B
  - B flow to C

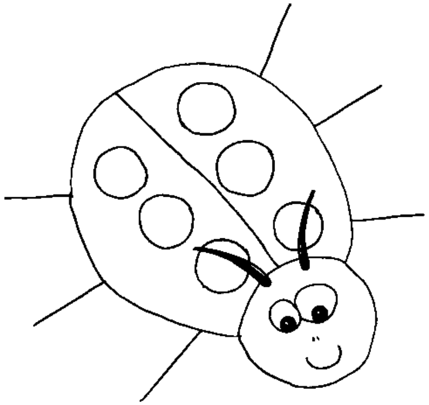




Control Flow Testing

- – Decisions
  - A flow to B or C

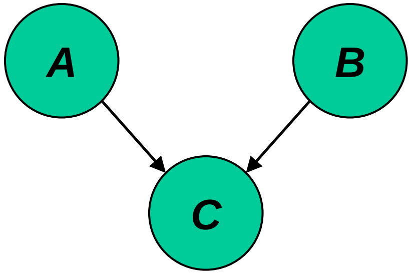


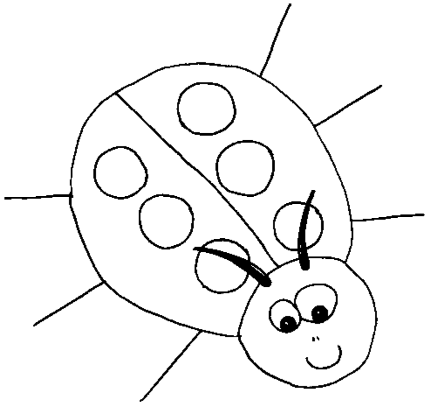


Control Flow Testing

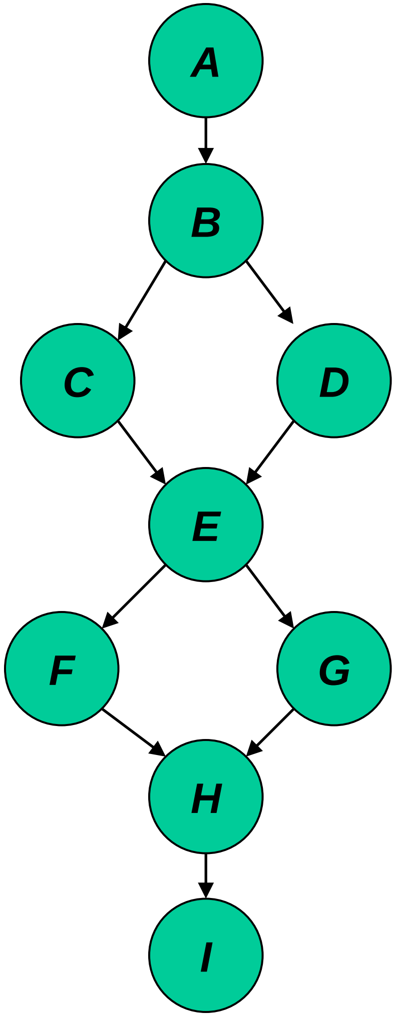
– Junctions

- A or B flow to C





Control Flow Testing

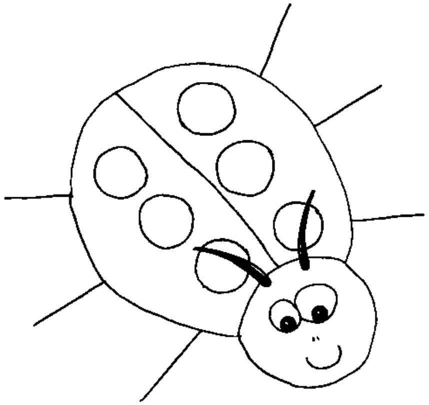


N=9
E=10
P=10-9+2
P=3

– Minimal basis paths

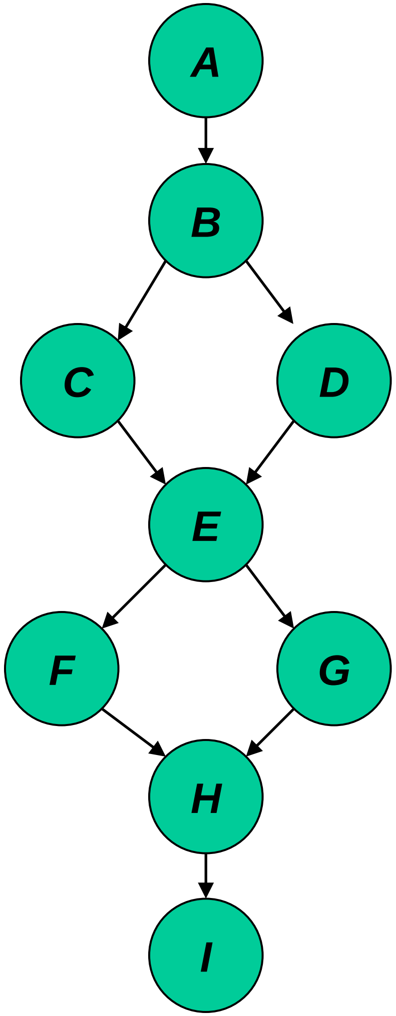
- N – number of nodes
- E – number of edges
- P – number of basis paths

- P = E – N + 2
- McCabe Cyclomatic Complexity

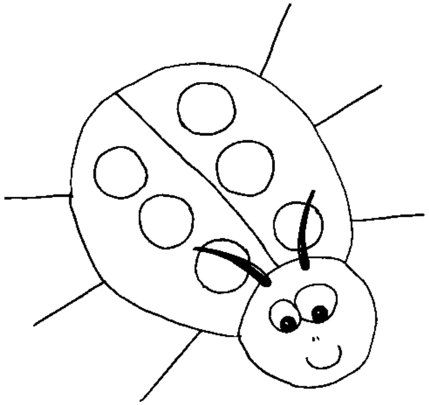


Control Flow Testing

- A set of basis paths:
  - A B D E G H I
  - A B C E G H I
  - A B D E F H I



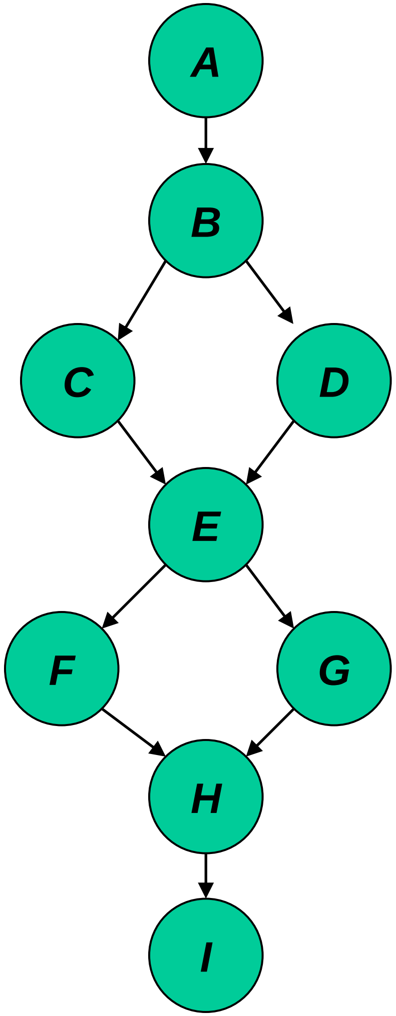
N=9
E=10
P=10-9+2
P=3



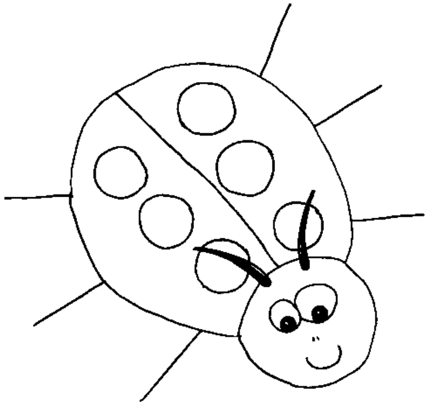
Control Flow Testing

– Finding basis paths:

1. Start with a *typical* baseline
2. Flip first decision keep rest as similar as possible
3. Continue flipping decisions on baseline
4. After all decisions on baseline have been flipped continue on next path
5. Stop when all paths have been exhausted



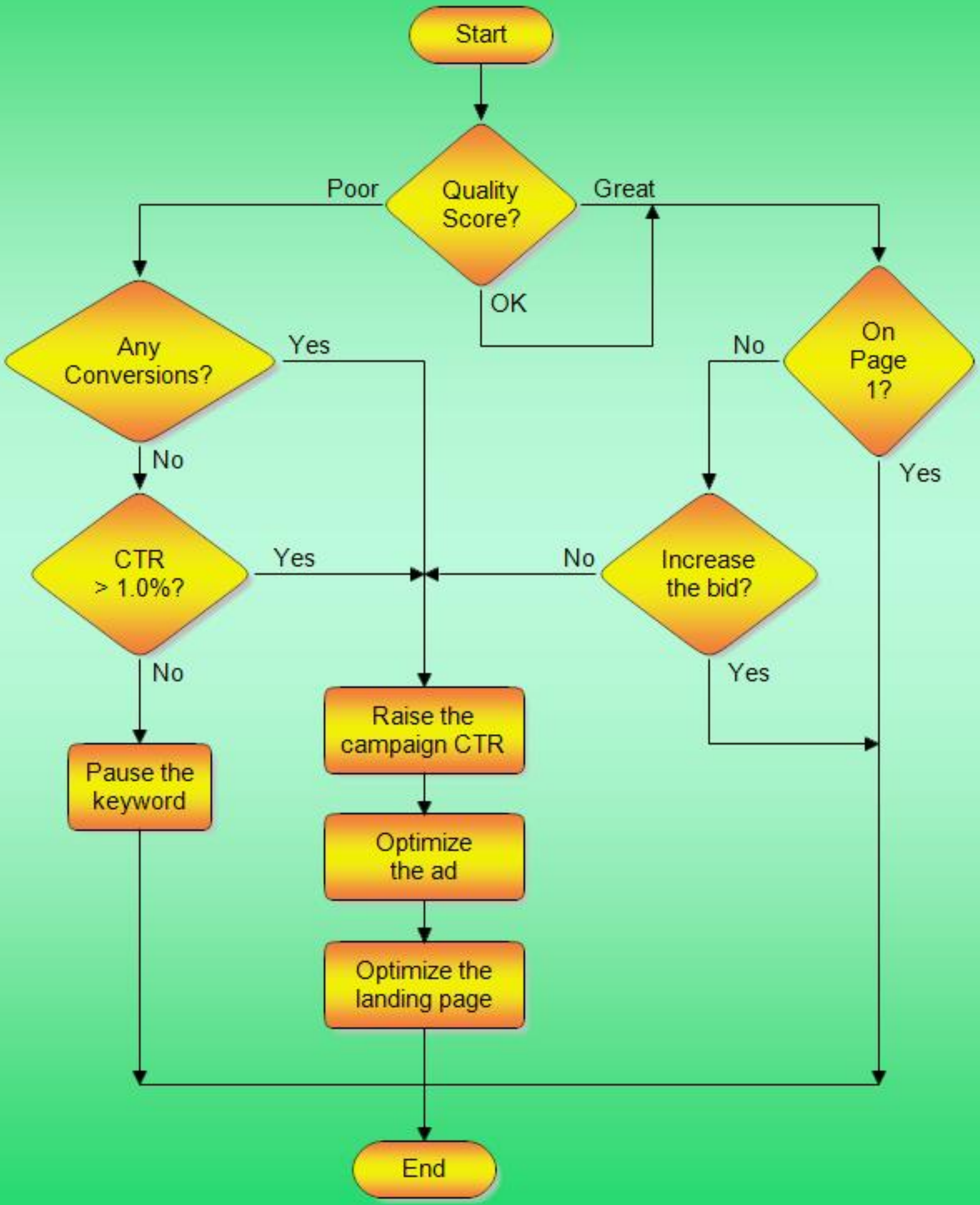
N=9
E=10
P=10-9+2
P=3



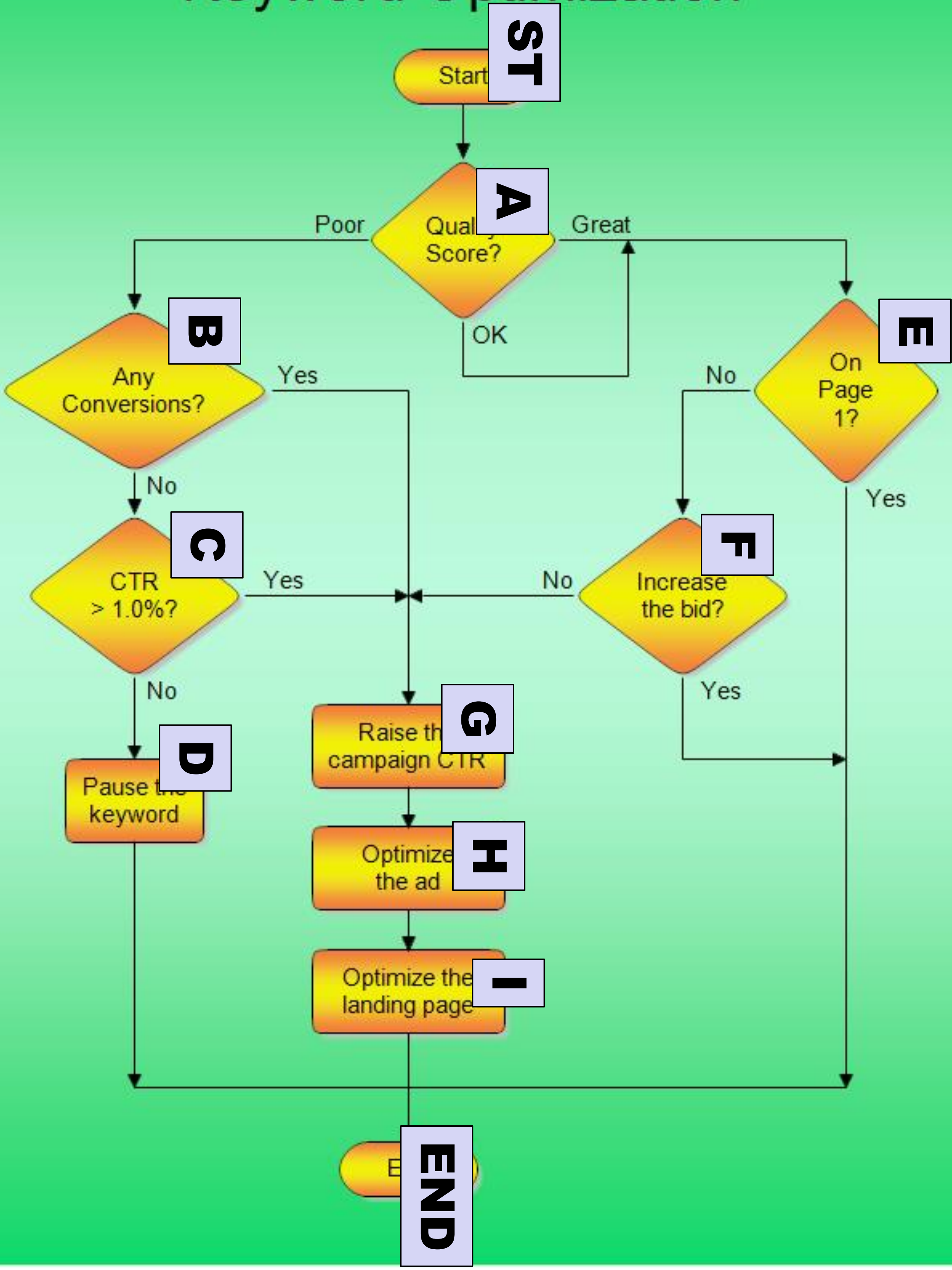
Keyword Optimization

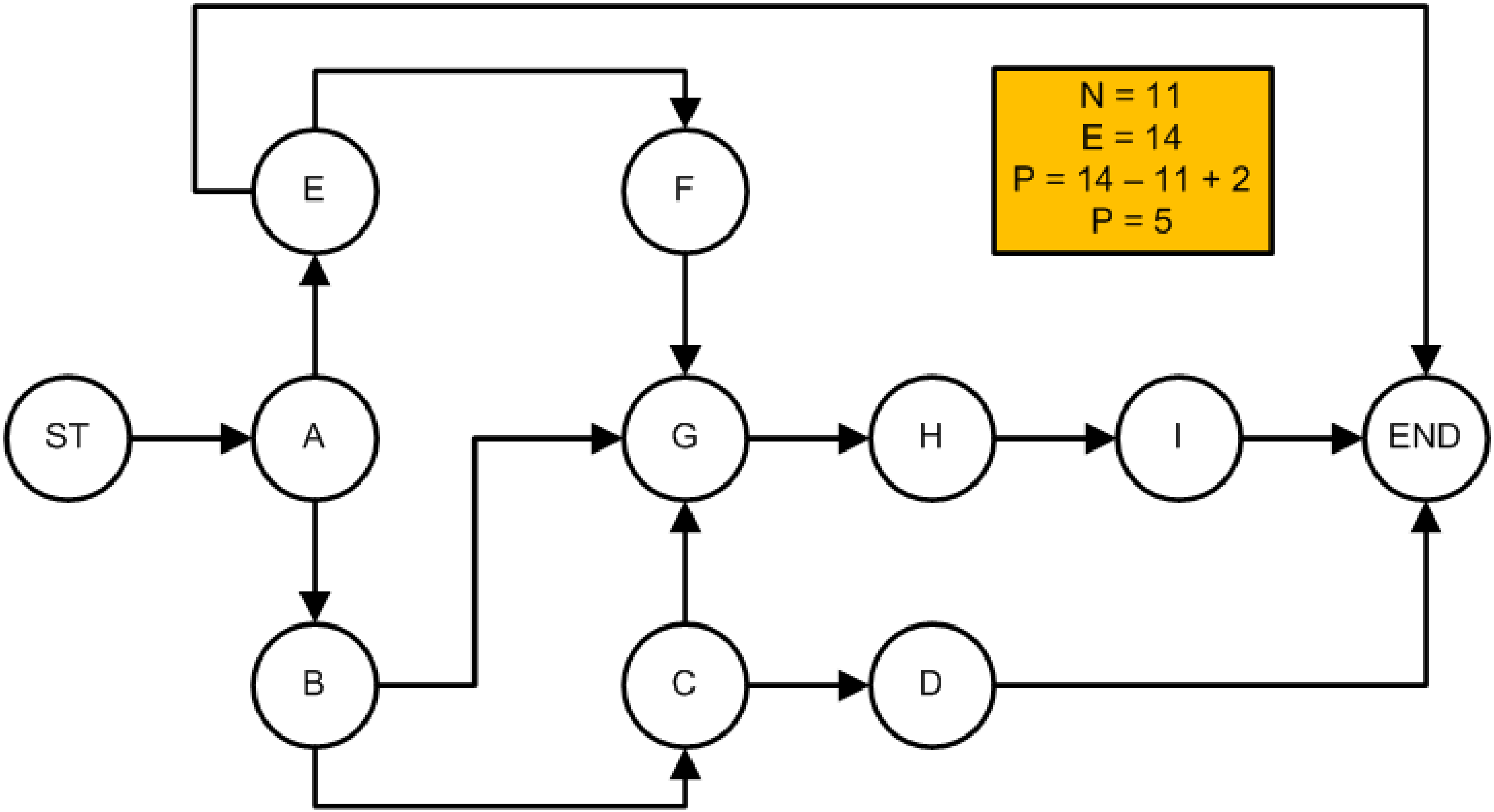
Control Flow Diagram

Keyword Optimization

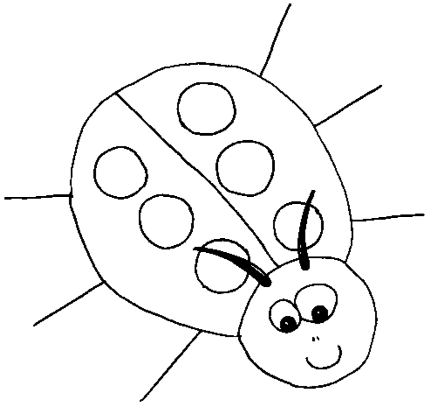


Keyword Optimization



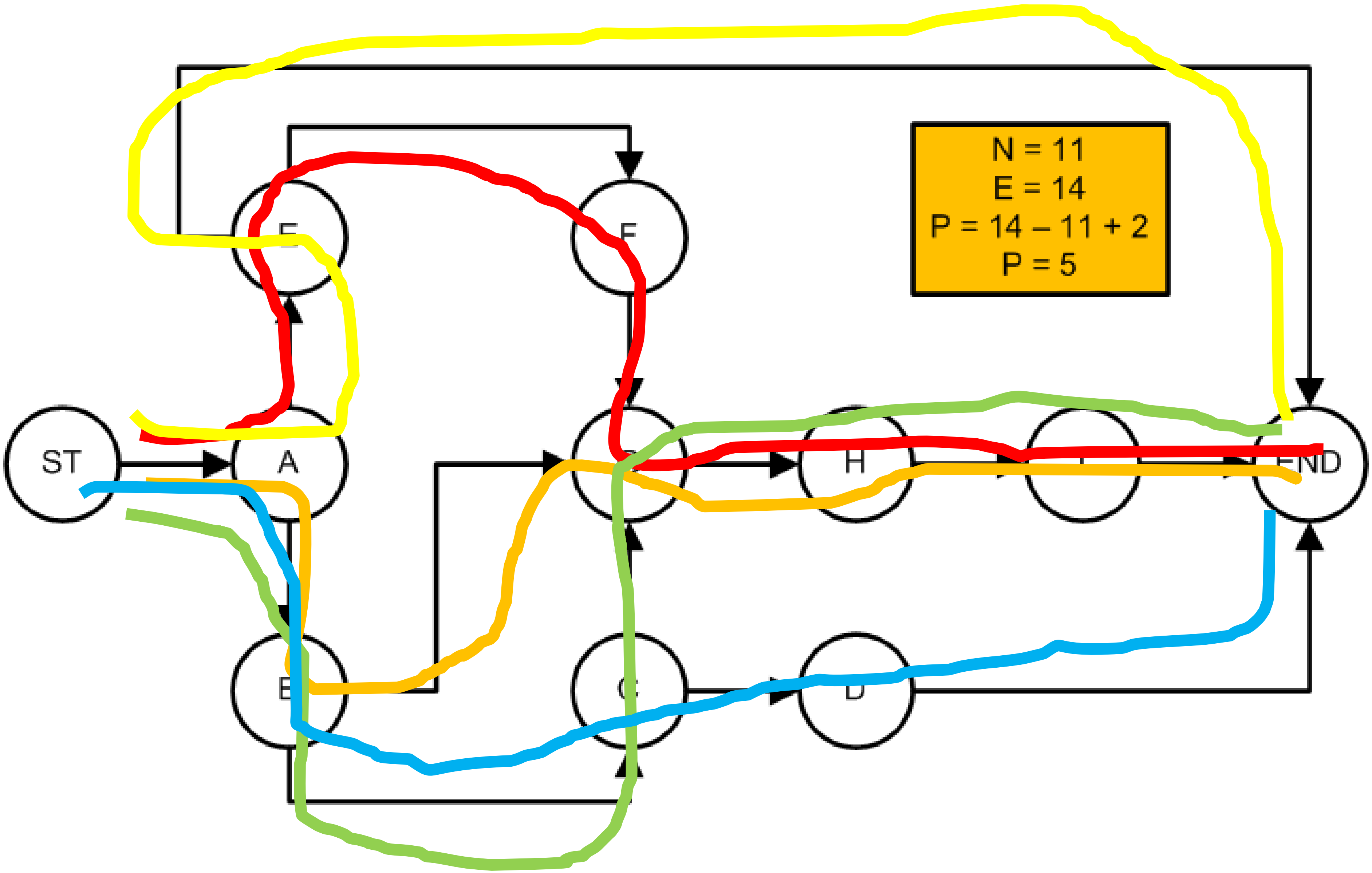


$N = 11$
$E = 14$
$P = 14 - 11 + 2$
$P = 5$

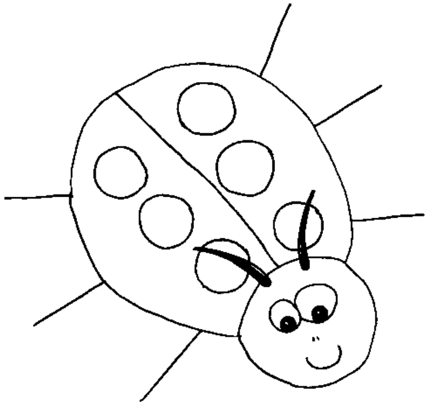


Control Flow Testing

Basis Paths								
1	ST	A	E	F	G	H	I	END
2	ST	A	B	G	H	I	END	
3	ST	A	E	END				
4	ST	A	B	C	G	H	I	END
5	ST	A	B	C	D	END		

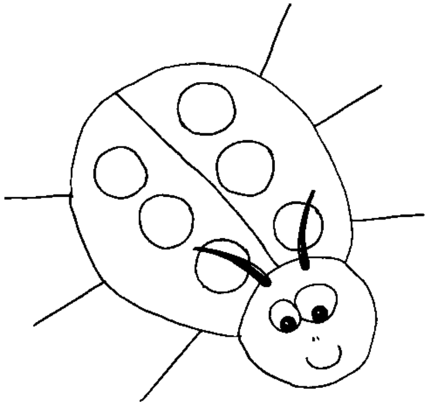


N = 11
E = 14
P = 14 – 11 + 2
P = 5



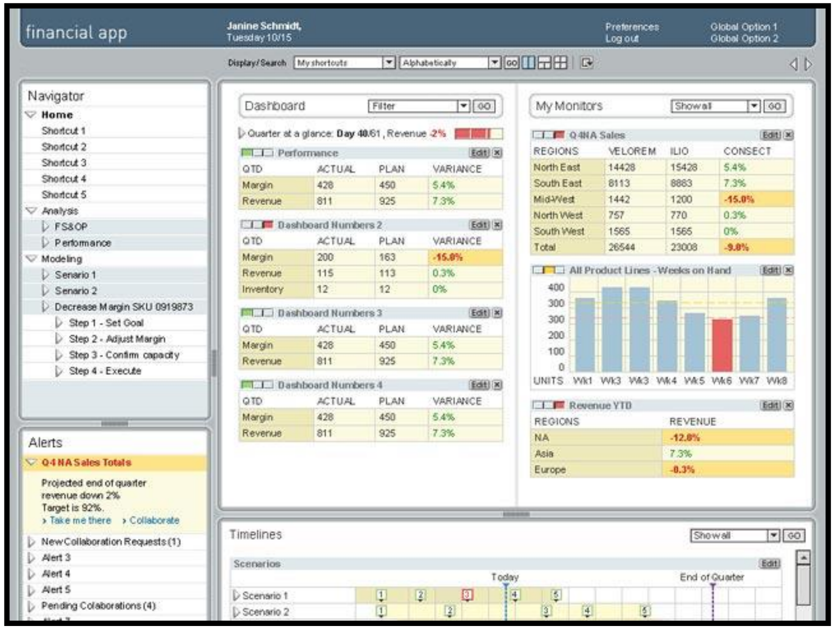
Insurance Workflow

Control Flow Diagram



Control Flow Testing

- Application screens are selected with three controls:
  - *(a) has 5 options*
  - *(b) has 6 options*
  - *(c) has 2 options*



Quarter at a glance: **Day 40/61**, Revenue -2%

Performance

QTD	ACTUAL	PLAN	VARIANCE
Margin	428	450	5.4%
Revenue	811	925	7.3%

Dashboard Numbers 2

QTD	ACTUAL	PLAN	VARIANCE
Margin	200	163	-15.0%
Revenue	115	113	0.3%
Inventory	12	12	0%

Dashboard Numbers 3

QTD	ACTUAL	PLAN	VARIANCE
Margin	428	450	5.4%
Revenue	811	925	7.3%

Dashboard Numbers 4

QTD	ACTUAL	PLAN	VARIANCE
Margin	428	450	5.4%
Revenue	811	925	7.3%

Q4NA Sales

REGIONS	VELOREM	ILIO	CONSECT
North East	14428	15428	5.4%
South East	8113	8883	7.3%
Mid-West	1442	1200	-15.0%
North West	757	770	0.3%
South West	1565	1565	0%
Total	26544	23008	-9.0%

All Product Lines - Weeks on Hand

Revenue YTD

REGIONS	REVENUE
NA	-12.0%
Asia	7.3%
Europe	-0.3%

Alerts

Q4 NA Sales Totals

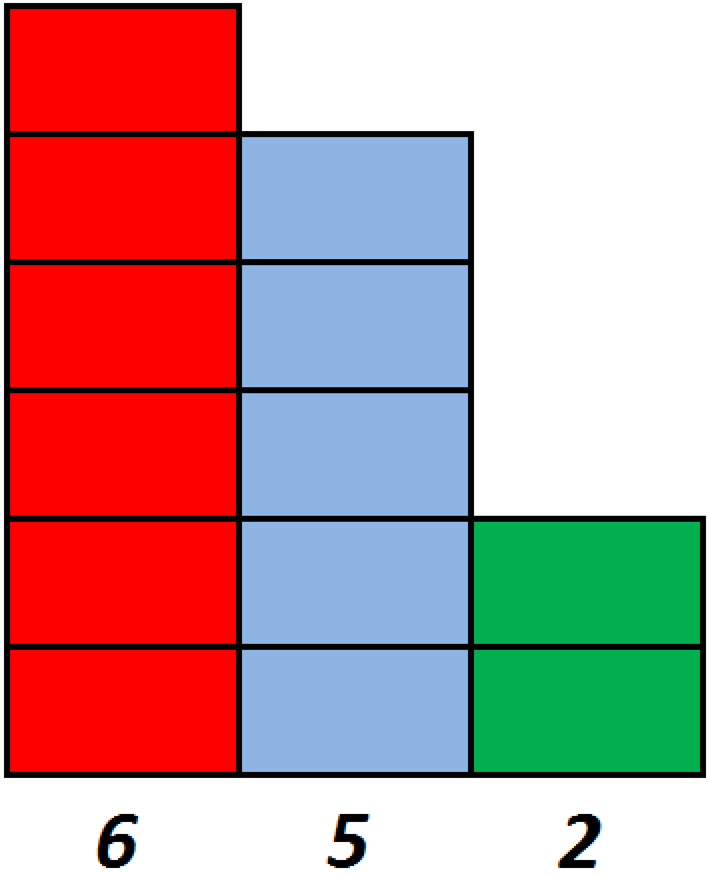
Projected end of quarter revenue down 2%.
Target is 92%.

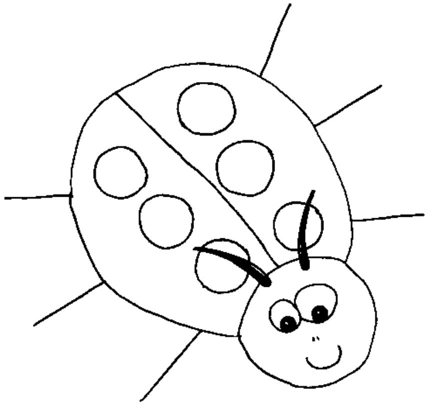
- How many screens can a user choose?

Control Flow Testing

- Total Combinations = 6 x 5 x 2 = 60

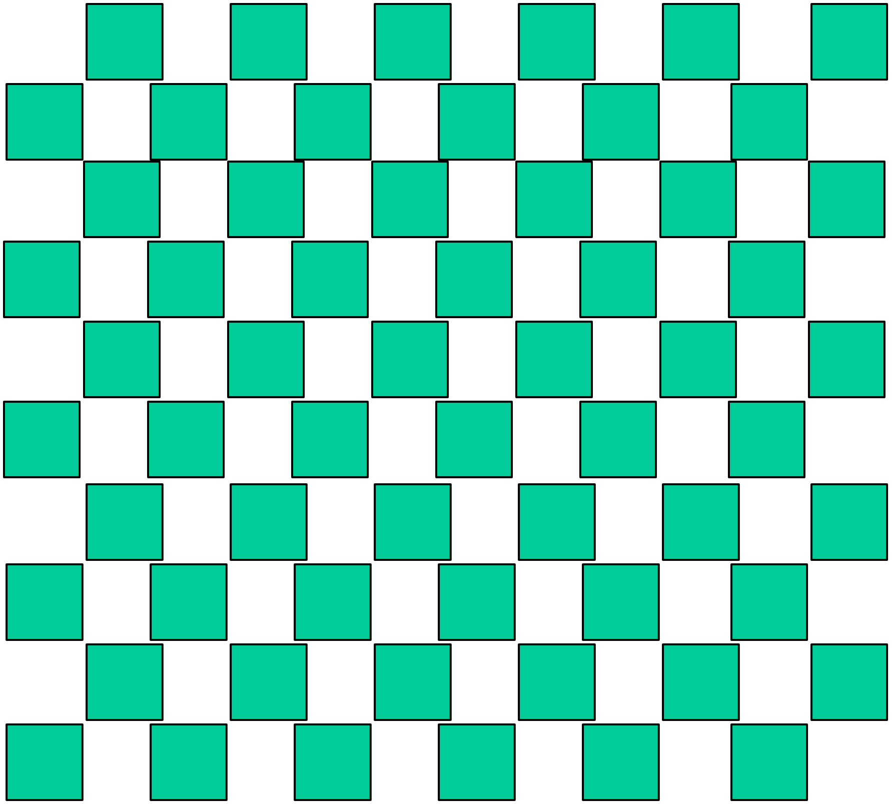
- To exercise each combination once a total of 60 tests would be required.

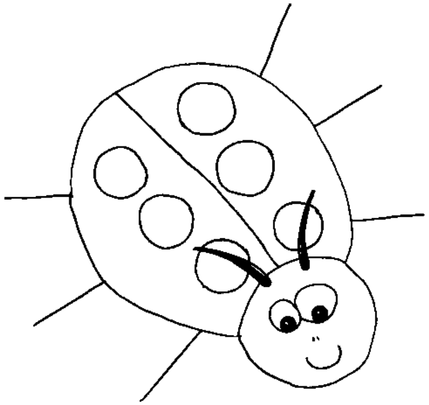




Control Flow Testing

- How many tests would be required to exercise all possible screens in every possible order?.





Control Flow Testing

- To exercise all screens in every possible order would require

$$n! > \sqrt{2\pi n}\left(\frac{n}{e}\right)^n.$$

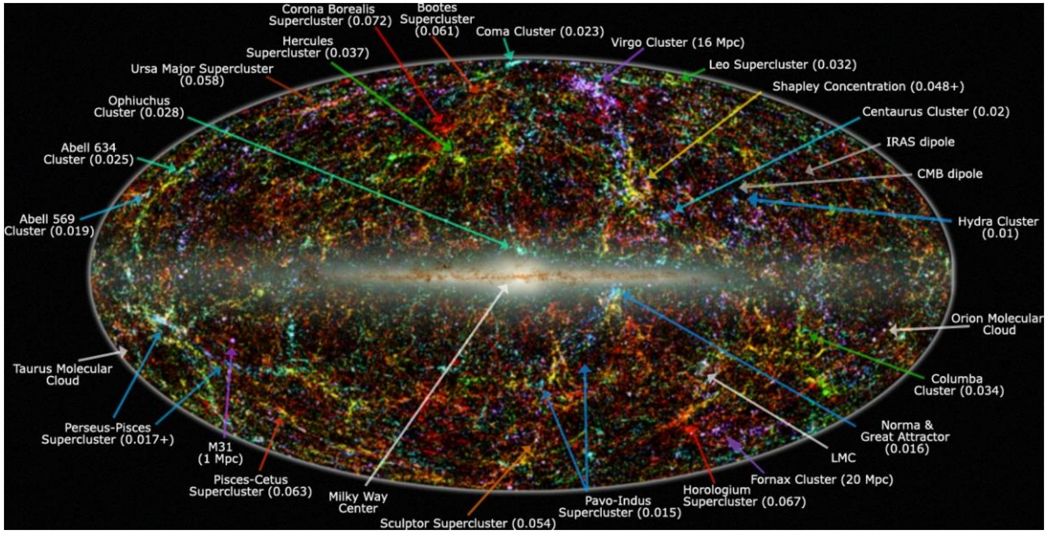
60! Test cases

$60! = 60 \times 59 \times 58 \times \ldots 3 \times 2 \times 1$

$60! \approx 8.32 \times 10^{**}81$

Control Flow Testing

How many atoms are in the observable universe?


Corona Borealis Supercluster (0.072)
Bootes Supercluster (0.061)
Coma Cluster (0.023)
Virgo Cluster (16 Mpc)
Hercules Supercluster (0.037)
Leo Supercluster (0.032)
Ursa Major Supercluster (0.058)
Shapley Concentration (0.048+)
Ophiuchus Cluster (0.028)
Centaurus Cluster (0.02)
Abell 634 Cluster (0.025)
IRAS dipole
CMB dipole
Abell 569 Cluster (0.019)
Hydra Cluster (0.01)
Orion Molecular Cloud
Taurus Molecular Cloud
Columba Cluster (0.034)
Perseus-Pisces Supercluster (0.017+)
Norma & Great Attractor (0.016)
M31 (1 Mpc)
LMC
Pisces-Cetus Supercluster (0.063)
Fornax Cluster (20 Mpc)
Milky Way Center
Horologium Supercluster (0.067)
Pavo-Indus Supercluster (0.015)
Sculptor Supercluster (0.054)

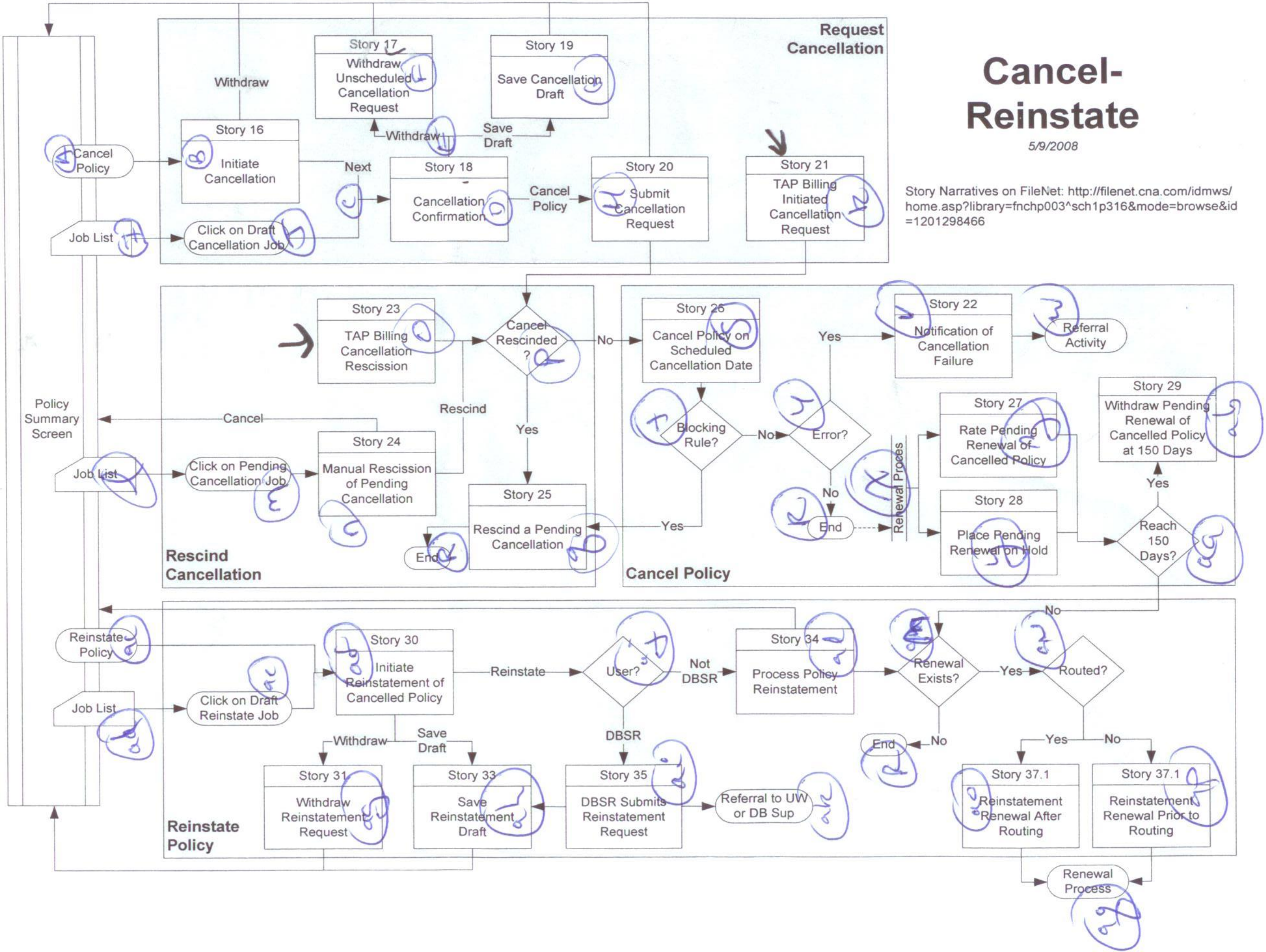
From 7.0 × 10**79

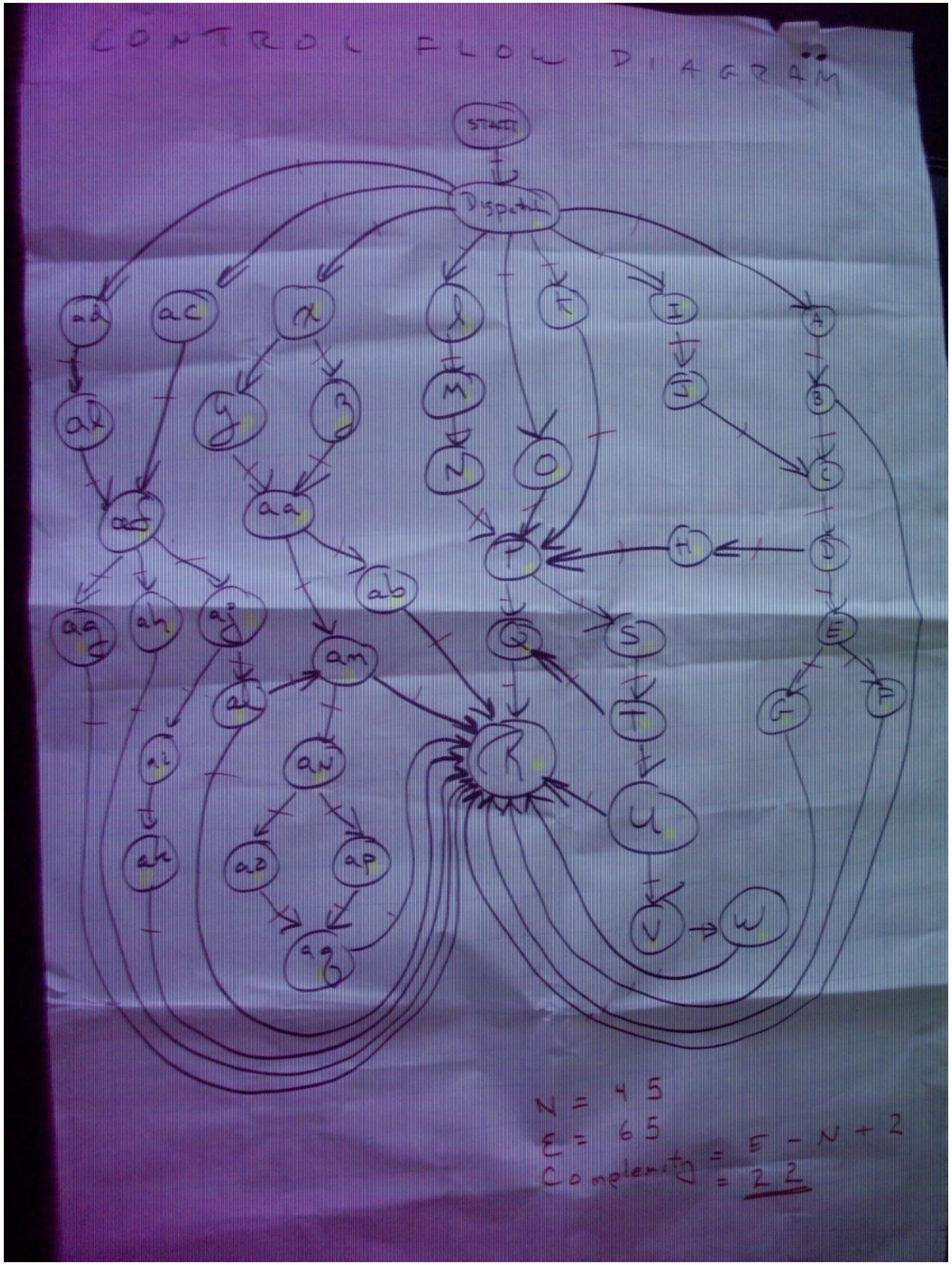
To 1.5 × 10**82

Cancel-Reinstate

5/9/2008

Story Narratives on FileNet: http://filenet.cna.com/idmws/home.asp?library=fnchp003^sch1p316&mode=browse&id=1201298466


Request Cancellation
Cancel Policy
Job List
Click on Draft Cancellation Job
Withdraw
Story 16
Initiate Cancellation
Story 17
Withdraw Unscheduled Cancellation Request
Withdraw
Save Draft
Story 19
Save Cancellation Draft
Next
Story 18
Cancellation Confirmation
Cancel Policy
Story 20
Submit Cancellation Request
Story 21
TAP Billing Initiated Cancellation Request
Policy Summary Screen
Rescind Cancellation
Story 23
TAP Billing Cancellation Rescission
Cancel Rescinded ?
No
Yes
Rescind
Cancel
Job List
Click on Pending Cancellation Job
Story 24
Manual Rescission of Pending Cancellation
Story 25
Rescind a Pending Cancellation
End
Cancel Policy
Story 26
Cancel Policy on Scheduled Cancellation Date
Blocking Rule?
Yes
No
Error?
Yes
No
End
Story 22
Notification of Cancellation Failure
Referral Activity
Renewal Proces
Story 27
Rate Pending Renewal of Cancelled Policy
Story 28
Place Pending Renewal on Hold
Reach 150 Days?
Yes
Story 29
Withdraw Pending Renewal of Cancelled Policy at 150 Days
No
Reinstate Policy
Reinstate Policy
Job List
Click on Draft Reinstate Job
Story 30
Initiate Reinstatement of Cancelled Policy
Reinstate
User?
Not DBSR
DBSR
Story 34
Process Policy Reinstatement
Renewal Exists?
Yes
No
End
Routed?
Yes
No
Withdraw
Save Draft
Story 31
Withdraw Reinstatement Request
Story 33
Save Reinstatement Draft
Story 35
DBSR Submits Reinstatement Request
Referral to UW or DB Sup
Story 37.1
Reinstatement Renewal After Routing
Story 37.1
Reinstatement Renewal Prior to Routing
Renewal Process
A
B
C
D
E
F
G
H
I
J
K

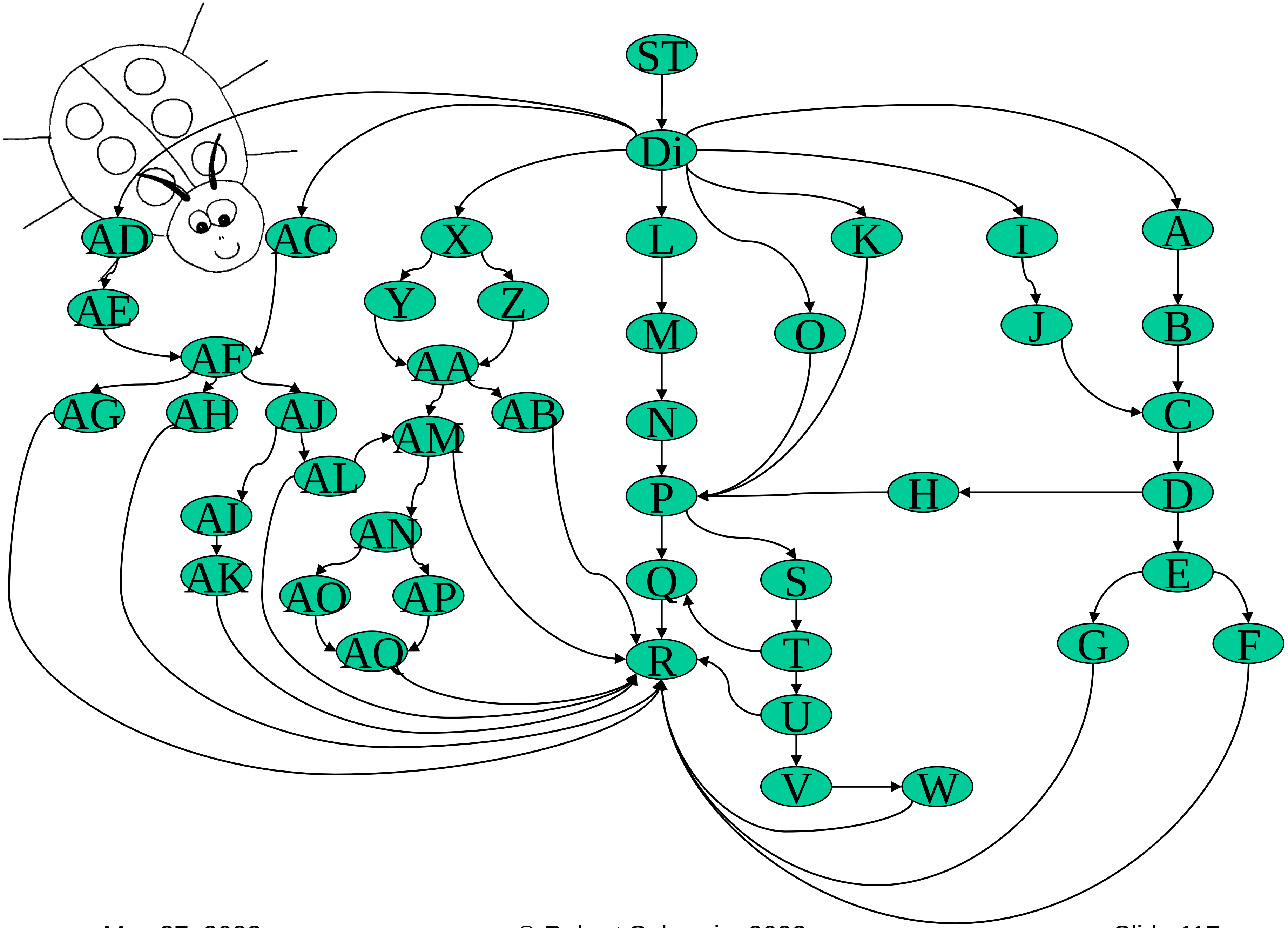


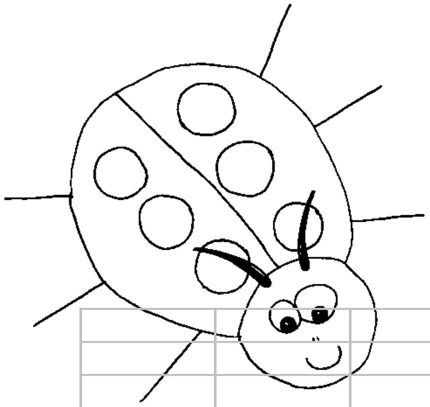
$N = 45$

$E = 65$

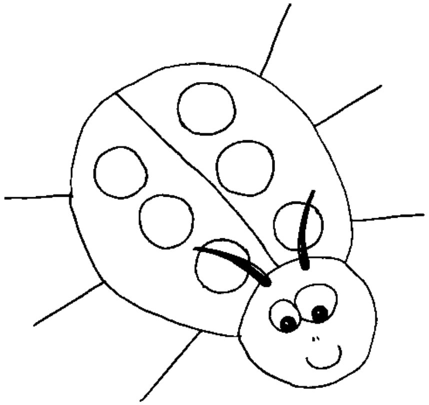
$\text{Complexity} = E - N + 2$

$= \underline{22}$


ST
Di
AD
AC
X
L
K
I
A
AE
Y
Z
M
O
J
B
AF
AA
C
AG
AH
AJ
AB
N
AM
AL
P
H
D
AI
AN
AK
Q
S
E
AO
AP
AQ
R
T
G
F
U
V
W

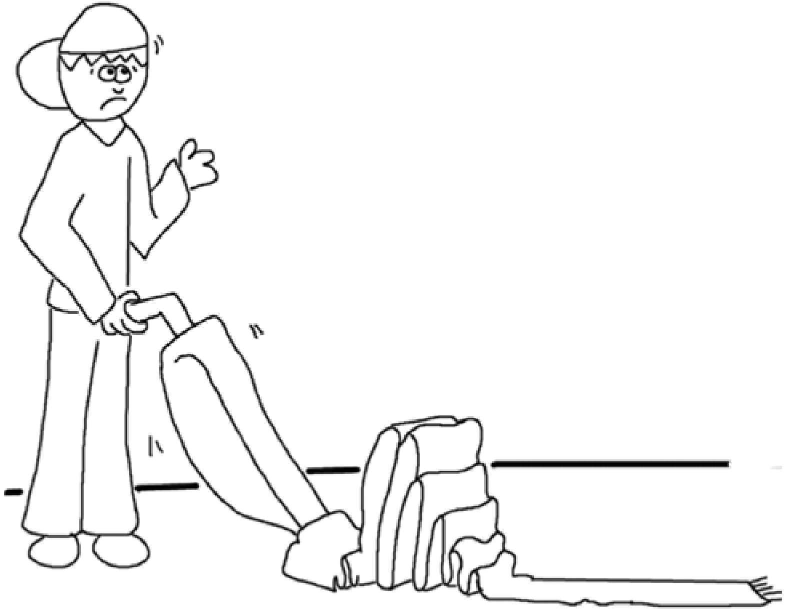


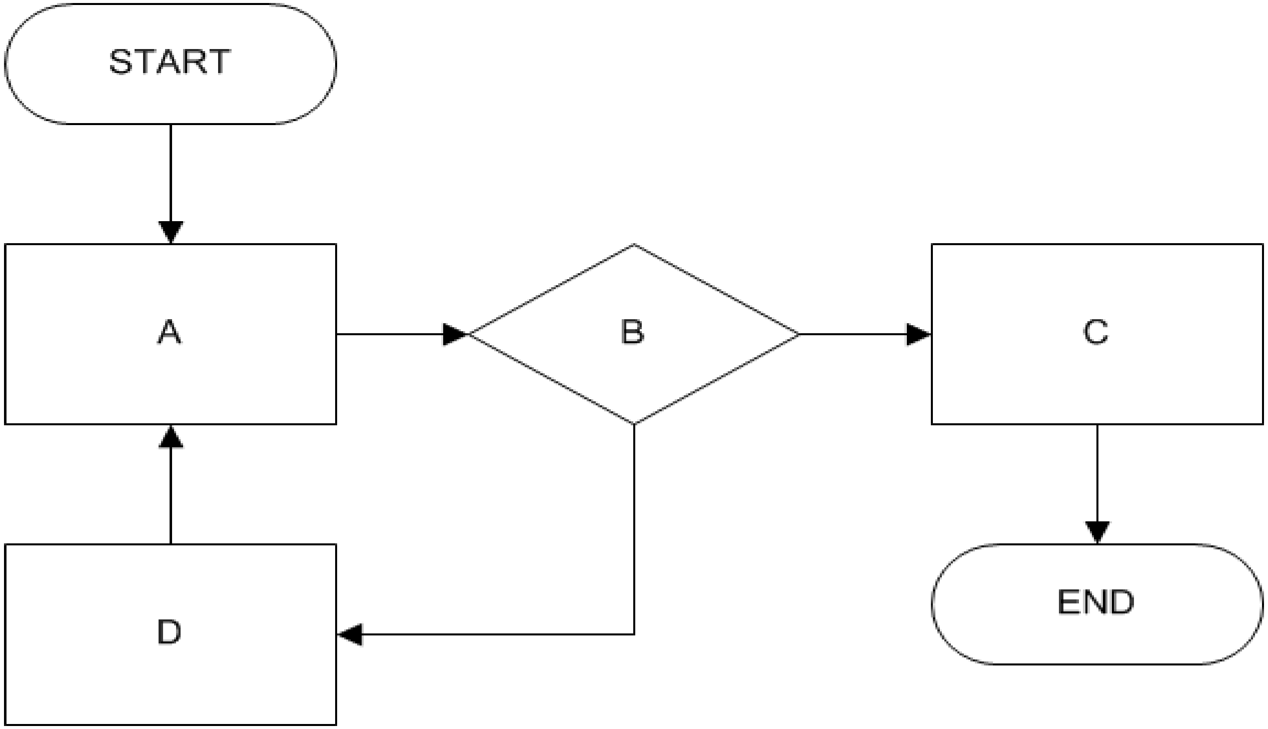
						Control Flow Diagram to Identify Basis Paths						
						Basis Paths						
p01	Start	Dispatch	L	M	N	P	Q	R				
p02	Start	Dispatch	X	Y	AA	AB	R					
p03	Start	Dispatch	AC	AF	AJ	AI	AK	R				
p04	Start	Dispatch	AD	AE	AF	AJ	AI	AK	R			
p05	Start	Dispatch	O	P	Q	R						
p06	Start	Dispatch	K	P	Q	R						
p07	Start	Dispatch	I	J	C	D	H	P	Q	R		
p08	Start	Dispatch	A	B	C	D	H	P	Q	R		
p09	Start	Dispatch	L	M	N	P	S	T	Q	R		
p10	Start	Dispatch	L	M	N	P	S	T	U	R		
p11	Start	Dispatch	L	M	N	P	S	T	U	V	W	R
p12	Start	Dispatch	X	Z	AA	AB	R					
p13	Start	Dispatch	X	Z	AA	AM	R					
p14	Start	Dispatch	X	Z	AA	AM	AN	AP	AQ	R		
p15	Start	Dispatch	X	Z	AA	AM	AN	AO	AQ	R		
p16	Start	Dispatch	AC	AF	AG	R						
p17	Start	Dispatch	AC	AF	AH	R						
p18	Start	Dispatch	AC	AF	AJ	AL	AM	R				
p19	Start	Dispatch	AC	AF	AJ	AL	R					
p20	Start	Dispatch	I	J	C	D	E	F	R			
p21	Start	Dispatch	I	J	C	D	E	G	R			
p22	Start	Dispatch	A	B	R							
	Nodes	45										
	Edges	65										
	Complexity	22	E-N+2									



Control Flow

- Some interesting control flow test ideas
  - All nodes
  - All edges
  - All paths

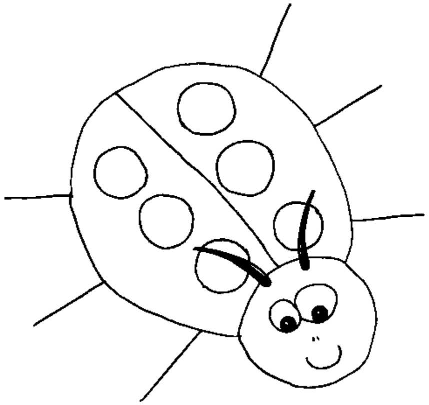




```
Edge List:

START
START A
A B
B D
D A
B C
C END
END
```

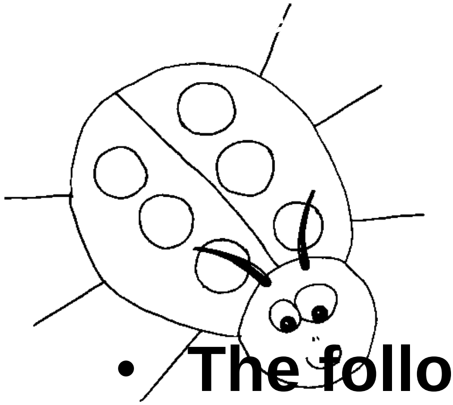
```
Basis Paths:

start node: START
end node: END
Number of nodes: 6  Number of edges: 6
Cyclomatic Complexity: 2
----------------------------

START,A,B,C,END
START,A,B,D,A,B,C,END

Number of basis paths: 2
```

```
Edge Paths:

start node: START
end node: END
----------------------------

START,A,B,D,A,B,C,END

Number of paths for edge coverage: 1
```



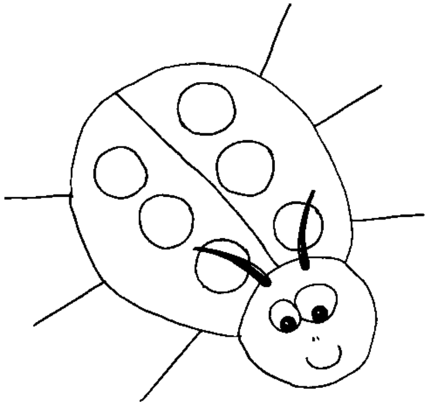
Test Design

Decision Tables



- **The following instructions are taken from FAFSA SM, the Free Application for Federal Student Aid form:**

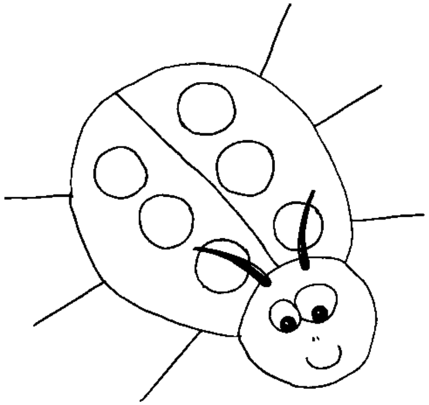
Step Four: Who is considered a parent in this step? Read these notes to determine who is considered a parent for purposes of this form. Answer all questions in Step Four about them, even if you do not live with them. Are you an orphan, or are you or were you (until age 18) a ward/dependent of the court? If Yes, skip Step Four. If your parents are both living and married to each other, answer the questions about them. If your parent is widowed or single, answer the questions about that parent. If your widowed parent is remarried as of today, answer the questions about that parent and the person whom your parent married (your stepparent). If your parents are divorced or separated, answer the questions about the parent you lived with more during the past 12 months. (If you did not live with one parent more than the other, give answers about the parent who provided more financial support during the last 12 months, or during the most recent year that you actually received support from a parent.) If this parent is remarried as of today, answer the questions on the rest of this form about that parent and the person whom your parent married (your stepparent).



Decision Tables

Construction

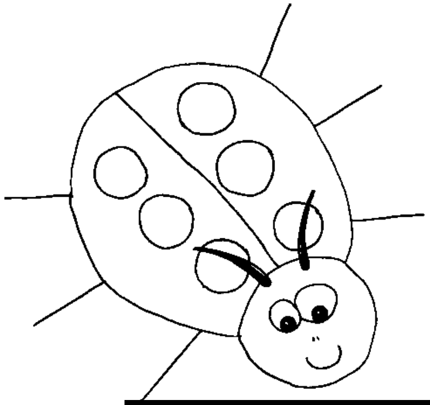
1. Identify Conditions
2. Identify Actions
3. Relate Conditions to Actions with Rules
4. Logic Reduction
5. Tests each Rule



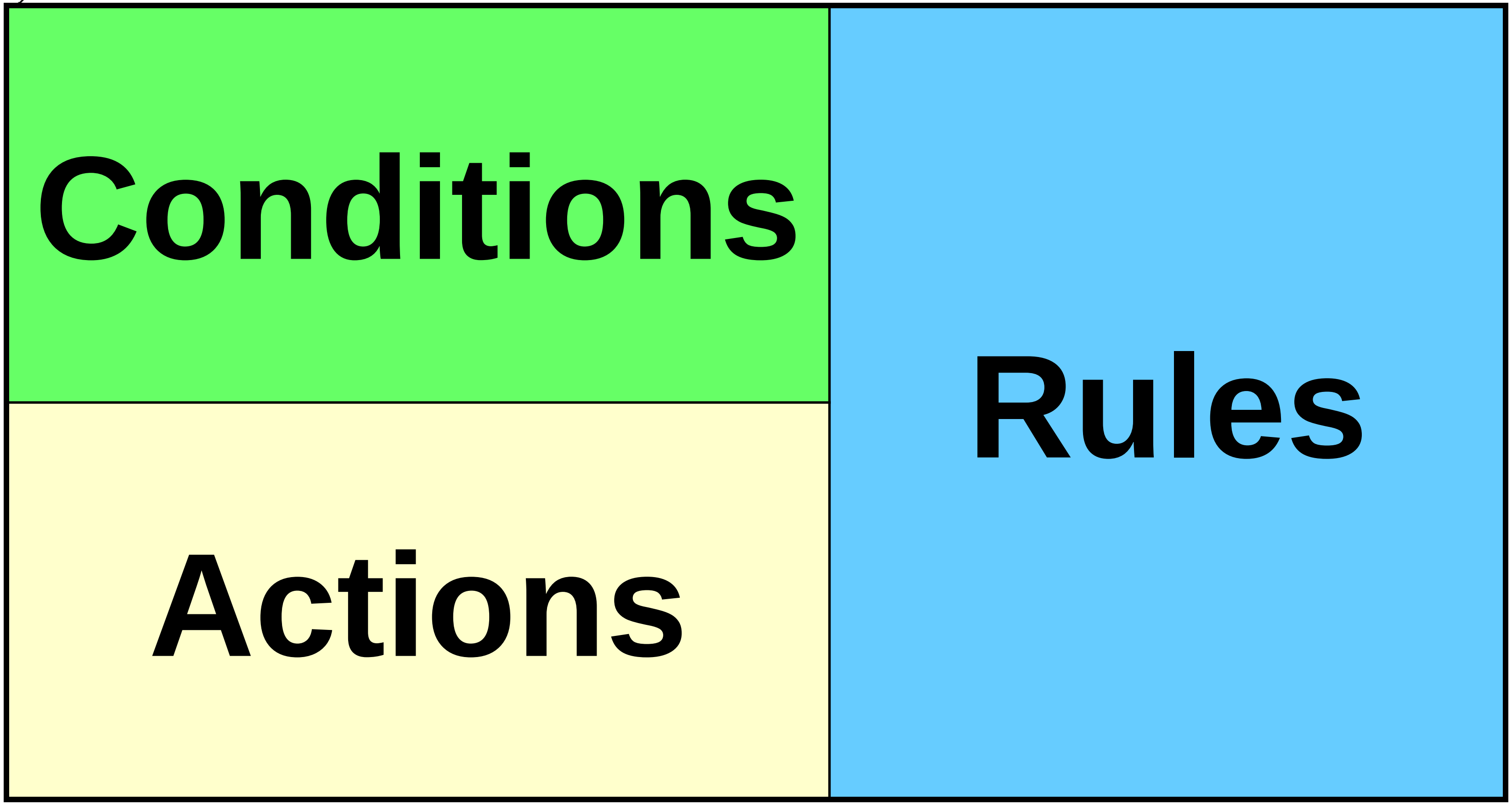
Decision Tables

Logic Modeling

- ❑ Application Logic
- ❑ Business Rules
- ❑ Regulations
- ❑ Multiple Conditions
- ❑ Multiple Actions



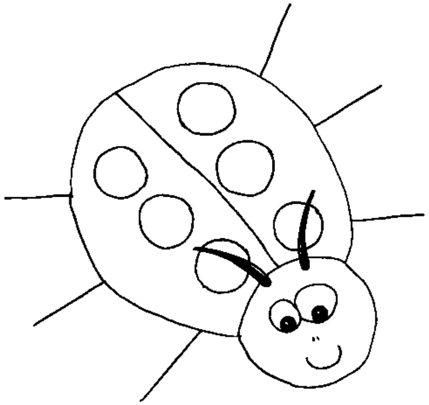
Decision Tables



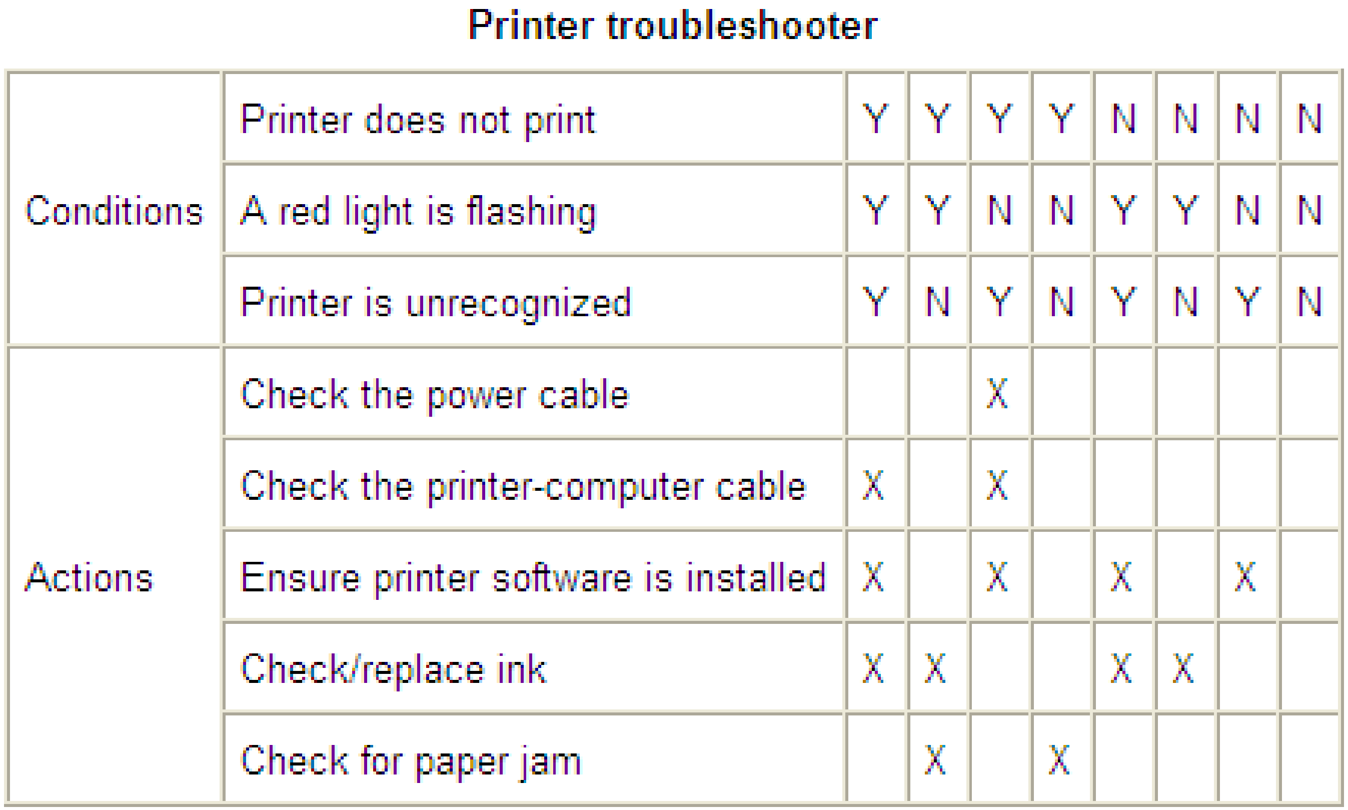
Conditions

Actions

Rules



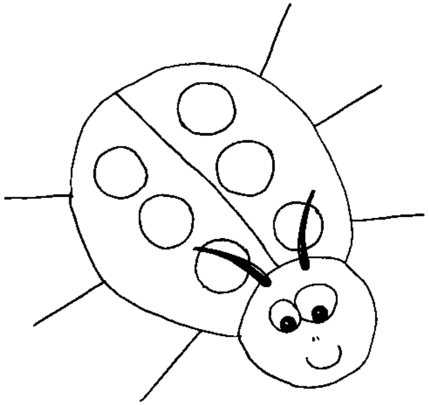
Decision Tables



Printer troubleshooter

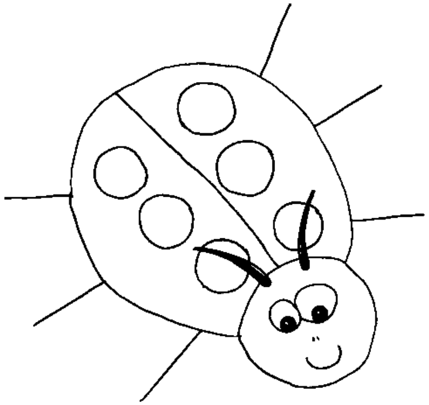
Conditions	Printer does not print	Y	Y	Y	Y	N	N	N	N
	A red light is flashing	Y	Y	N	N	Y	Y	N	N
	Printer is unrecognized	Y	N	Y	N	Y	N	Y	N
Actions	Check the power cable			X					
	Check the printer-computer cable	X		X					
	Ensure printer software is installed	X		X		X		X	
	Check/replace ink	X	X			X	X		
	Check for paper jam		X		X				

www.wikipedia.com



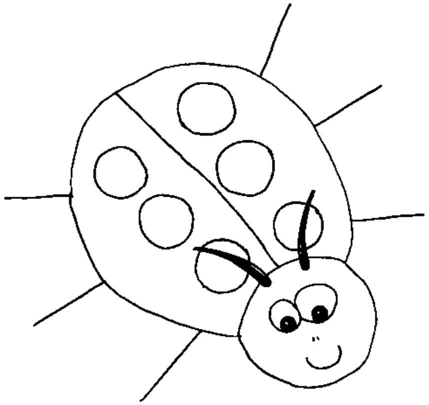
University Acceptance

Decision Table



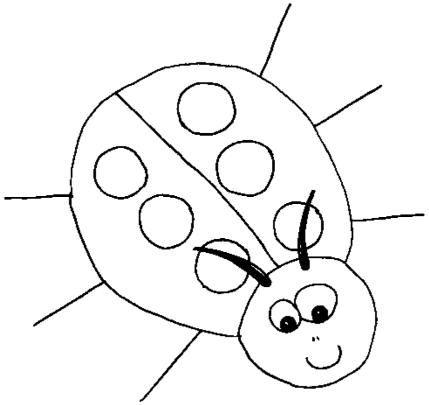
Decision Tables Example

Conditions	R01	R02	R03	R04	R05	R06	R07	R08	R09	R10
SAT SCORE	High	High	High	High	High	High	High	High	High	High
Prerequisites	All	All	All	Core Only	Core Only	Core Only	Partial	Partial	Partial	None
Residency	In State	Out of State	Foreign	In State	Out of State	Foreign	In State	Out of State	Foreign	*
Actions										
Accept	x	x	x							
Redirect							x	x	x	x
Conditional				x	x	x				
Reject										
Grant	x	x		x	x		x	x		



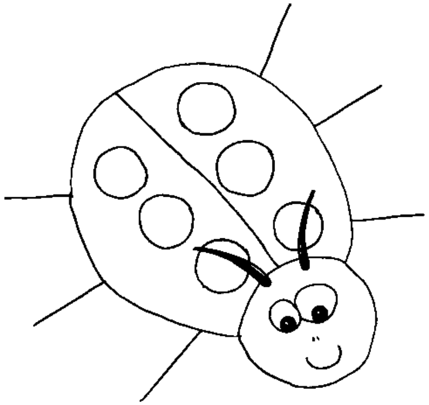
Decision Tables Example

Conditions	R11	R12	R13	R14	R15	R16	R17	R18	R19	R20
SAT SCORE	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium
Prerequisites	All	All	All	Core Only	Core Only	Core Only	Partial	Partial	Partial	None
Residency	In State	Out of State	Foreign	In State	Out of State	Foreign	In State	Out of State	Foreign	*
Actions										
Accept	x	x	x							
Redirect							x	x	x	
Conditional				x	x	x				
Reject										x
Grant	x			x			x			



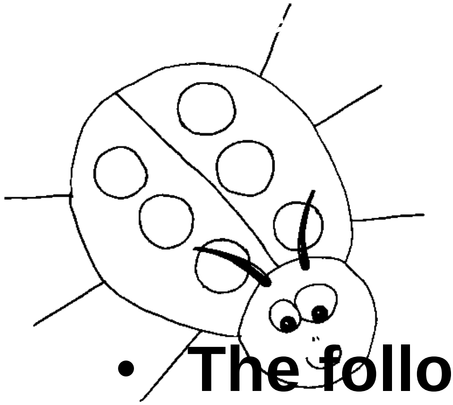
Decision Table Example

Conditions	***R21***	***R22***	***R23***	***R24***
SAT SCORE	Low	Low	Low	Low
Prerequisites	All	Core Only	Partial	None
Residency	*	*	*	*
Actions				
Accept				
Redirect				
Conditional	x			
Reject		x	x	x
Grant				



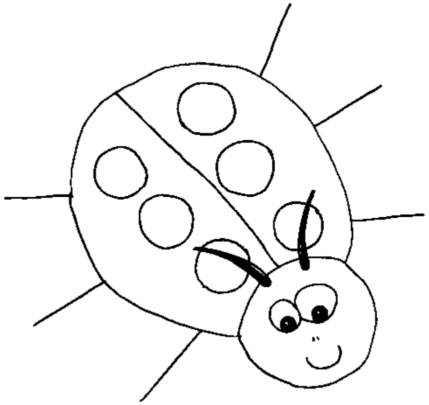
DVD Rental

Decision Table



- **The following rules describe how we process DVDs that are not returned. Create a decision table that represents these rules:**

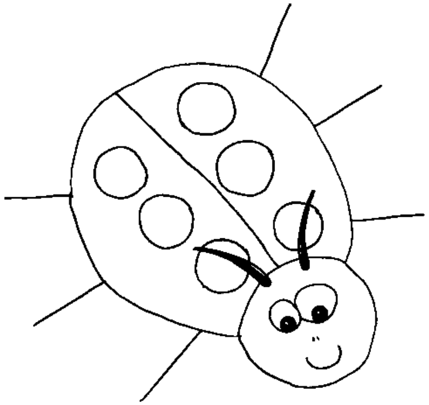
> **“When a DVD is “lost” (it has not been returned in 90 days) the following rules apply: If the customer has not lost any DVDs in the preceding 12 months we do not suspend the account nor do we bill for the lost DVDs. If the customer has lost 1-2 DVDs in the preceding 12 months and if they have not filed a complaint with the USPS we do not suspend or bill, If the customer has lost 1-2 in the preceding 12 months and if they have filed a complaint with the USPS we do not suspend or bill. If the customer has lost 3-4 DVDs in the preceding 12 months and if they have not filed a complaint then we suspend their account but do not bill for the lost DVDs. If the customer has lost 3-4 DVDs in the preceding 12 months and if they have filed a complaint then we do not suspend their account nor do we bill them. If the customer has lost 5 or more DVDs in the preceding 12 months we suspend their account and bill them for the lost DVDs.”**



Decision Table Example

Binary Ranges

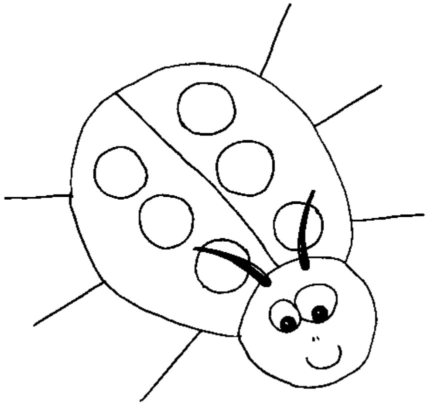
Conditions	*Rule 1*	*Rule 2*	*Rule 3*	*Rule 4*	*Rule 5*	*Rule 6*
Lost DVDs in past 12 Months (YES or NO)	NO	YES	YES	YES	YES	YES
Lost 1-2 DVDS in past 12 Months (YES or NO)	NO	YES	YES	YES	YES	YES
Complaint to USPS	Dcare	NO	YES	NO	YES	Dcare
Lost 3-4 DVDs in past 12 Months (YES or NO)	NO	NO	NO	YES	YES	YES
Lost 5 or more DVDs in past 12 Months (YES or NO)	NO	NO	NO	NO	NO	YES
Actions						
Suspend account	NO	NO	NO	YES	NO	YES
Bill for lost DVDs	NO	NO	NO	NO	NO	YES



Decision Table Example

Value Ranges

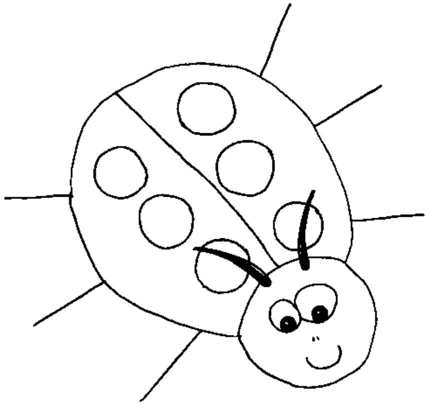
Conditions	*Rule 1*	*Rule 2*	*Rule 3*	*Rule 4*	*Rule 5*	*Rule 6*
Number of DVDs lost in past 12 Months	0	1-2	1-2	3-4	3-4	5 +
Filled Complaint with USPS	Dcare	YES	NO	YES	NO	Dcare
Actions						
Suspend account	NO	NO	NO	NO	YES	YES
Bill for lost DVDs	NO	NO	NO	NO	NO	YES



Decision Table Example

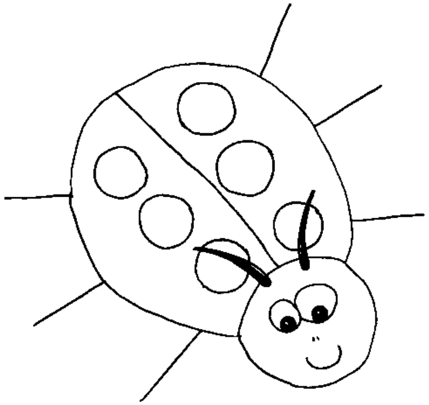
Reduced

Conditions	*Rule 1*	*Rule 2*	*Rule 3*	*Rule 4*
Number of DVDs lost in past 12 Months	0-2	3-4	3-4	5 +
Filled Complaint with USPS	*	YES	NO	Dcare
Actions				
Suspend account	NO	NO	YES	YES
Bill for lost DVDs	NO	NO	NO	YES

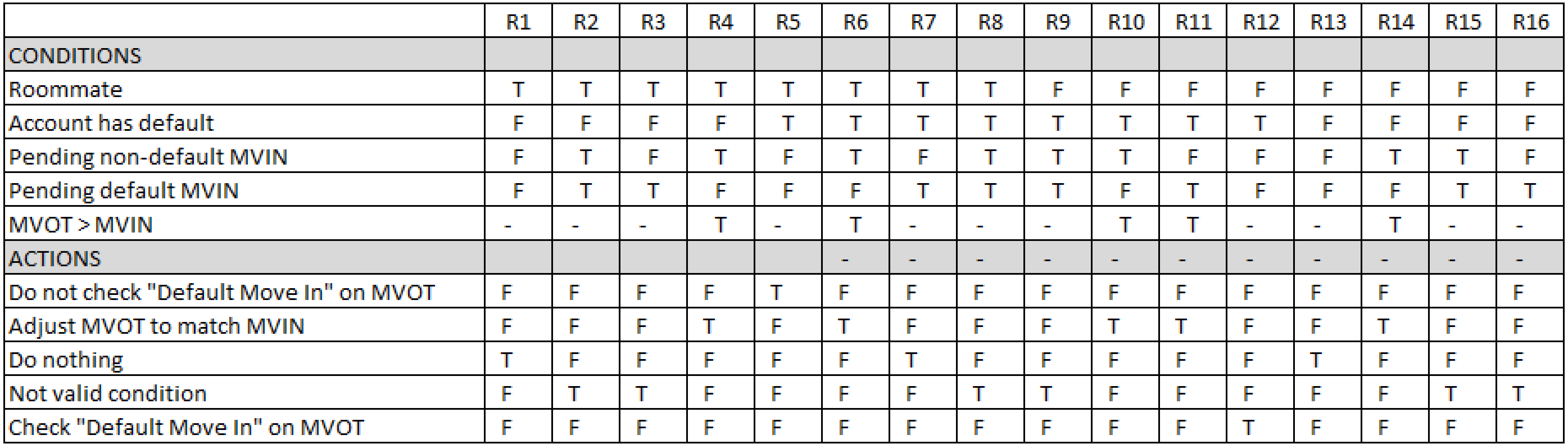


Customer Business Rules

Decision Tables

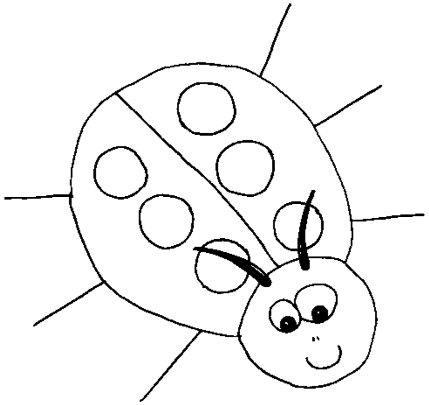


Decision Tables

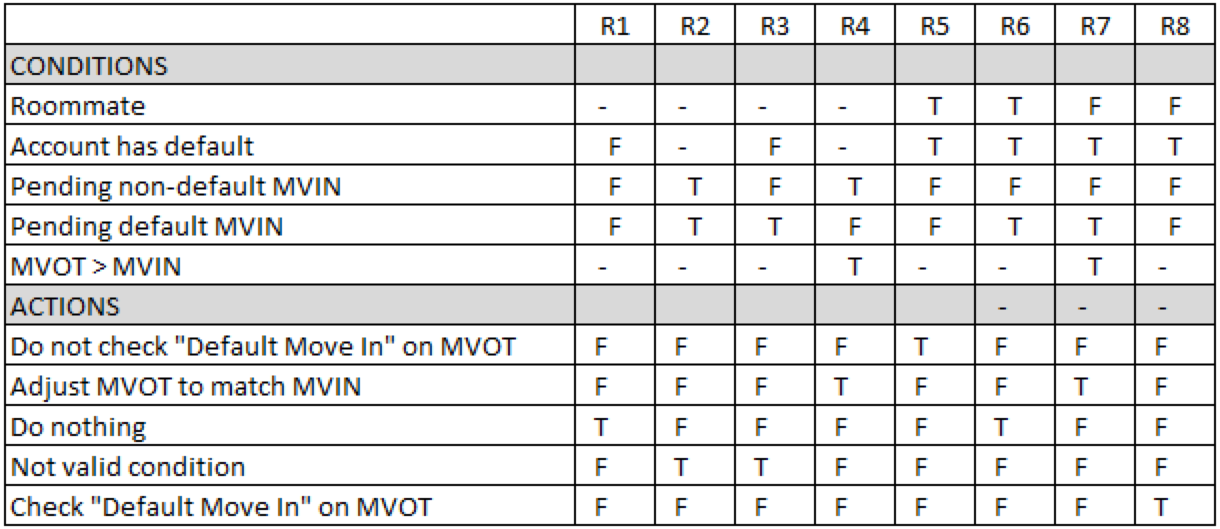


	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13	R14	R15	R16
CONDITIONS																
Roommate	T	T	T	T	T	T	T	T	F	F	F	F	F	F	F	F
Account has default	F	F	F	F	T	T	T	T	T	T	T	T	F	F	F	F
Pending non-default MVIN	F	T	F	T	F	T	F	T	T	T	F	F	F	T	T	F
Pending default MVIN	F	T	T	F	F	F	T	T	T	F	T	F	F	F	T	T
MVOT > MVIN	-	-	-	T	-	T	-	-	-	T	T	-	-	T	-	-
ACTIONS						-	-	-	-	-	-	-	-	-	-	-
Do not check "Default Move In" on MVOT	F	F	F	F	T	F	F	F	F	F	F	F	F	F	F	F
Adjust MVOT to match MVIN	F	F	F	T	F	T	F	F	F	T	T	F	F	T	F	F
Do nothing	T	F	F	F	F	F	T	F	F	F	F	F	T	F	F	F
Not valid condition	F	T	T	F	F	F	F	T	T	F	F	F	F	F	T	T
Check "Default Move In" on MVOT	F	F	F	F	F	F	F	F	F	F	F	T	F	F	F	F

Elicited from customer

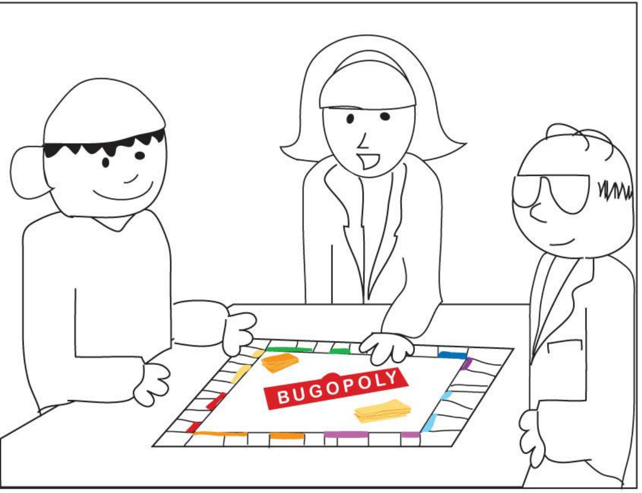


Decision Tables



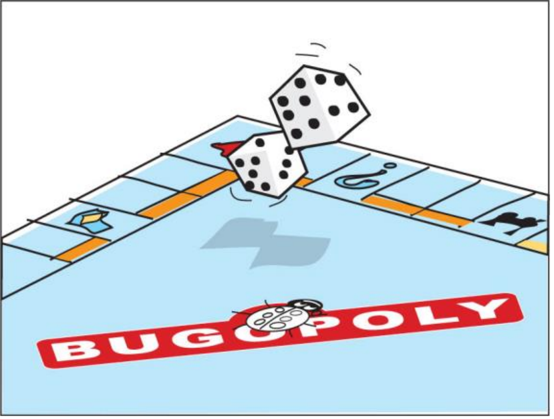
	R1	R2	R3	R4	R5	R6	R7	R8
CONDITIONS								
Roommate	-	-	-	-	T	T	F	F
Account has default	F	-	F	-	T	T	T	T
Pending non-default MVIN	F	T	F	T	F	F	F	F
Pending default MVIN	F	T	T	F	F	T	T	F
MVOT > MVIN	-	-	-	T	-	-	T	-
ACTIONS						-	-	-
Do not check "Default Move In" on MVOT	F	F	F	F	T	F	F	F
Adjust MVOT to match MVIN	F	F	F	T	F	F	T	F
Do nothing	T	F	F	F	F	T	F	F
Not valid condition	F	T	T	F	F	F	F	F
Check "Default Move In" on MVOT	F	F	F	F	F	F	F	T

Reduced



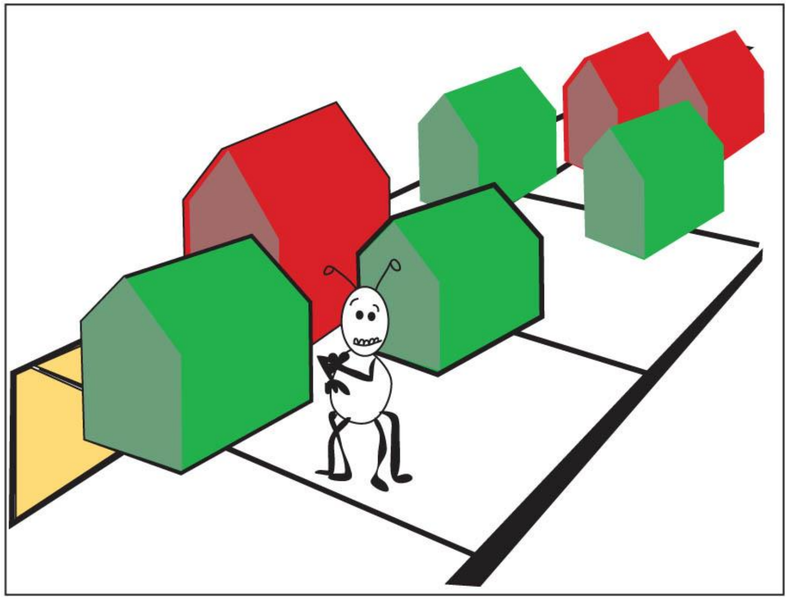
Monopoly® House Purchase Rules

Decision Tables



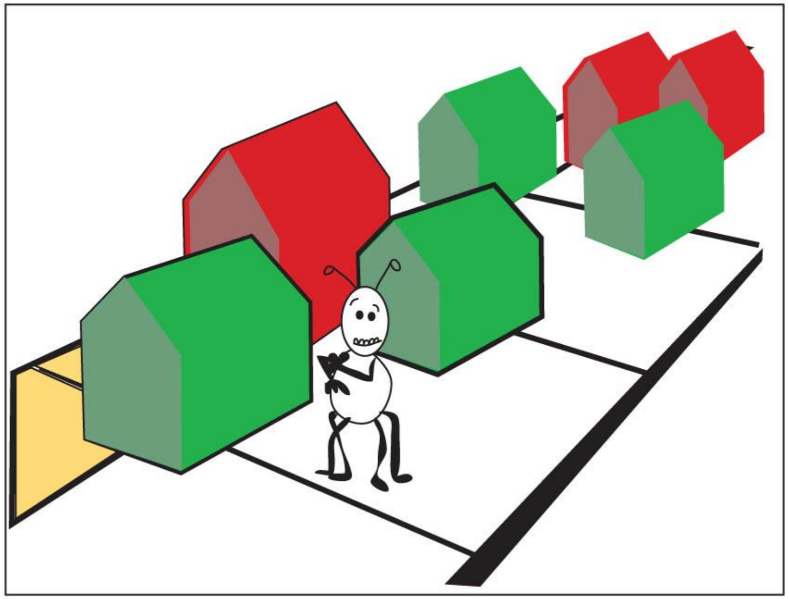
Decision Tables

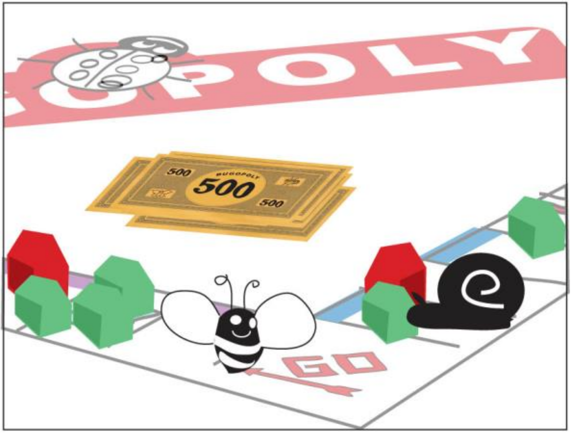
- House Purchase Rules
- Property must be in a color group
- A player must own all properties of a color group
- No properties can be mortgaged



Decision Tables

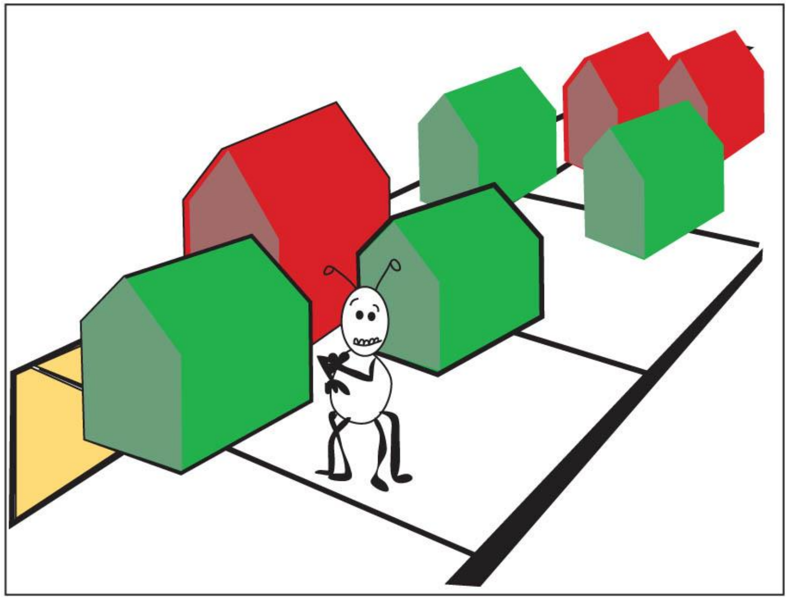
- House Purchase Rules
- Houses must be available for sale
- Houses must be built progressively
- Each property can have a maximum of 4 houses

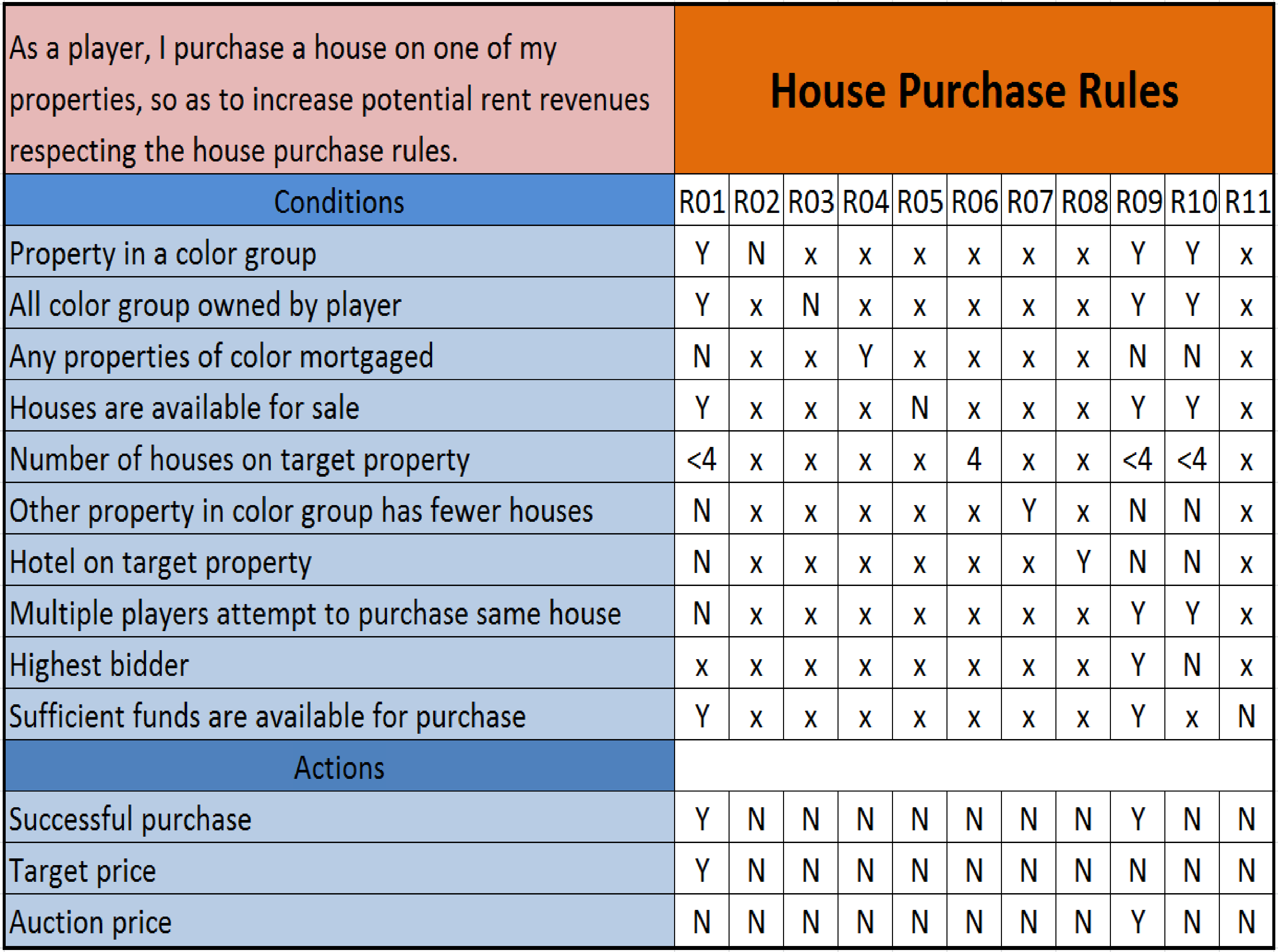




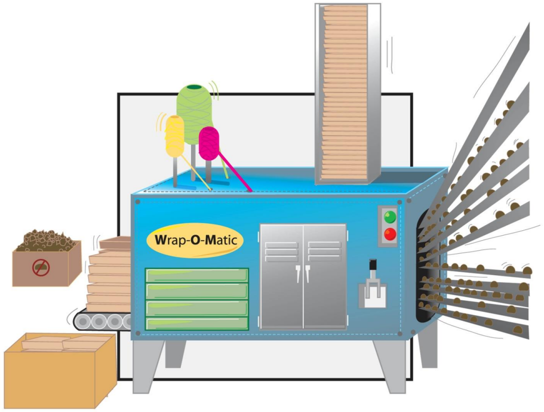
Decision Tables

- House Purchase Rules
- If multiple players attempt to purchase the same house the highest auction bidder gets it



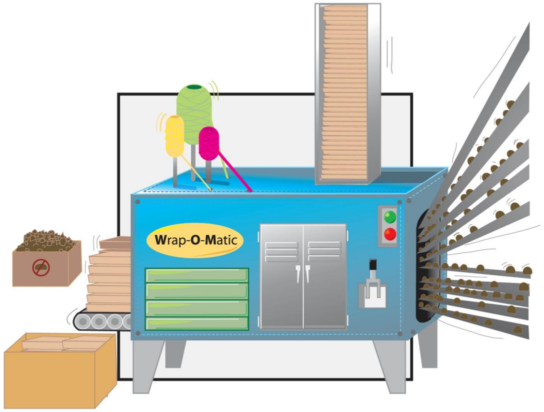


As a player, I purchase a house on one of my properties, so as to increase potential rent revenues respecting the house purchase rules.	House Purchase Rules										
Conditions	R01	R02	R03	R04	R05	R06	R07	R08	R09	R10	R11
Property in a color group	Y	N	x	x	x	x	x	x	Y	Y	x
All color group owned by player	Y	x	N	x	x	x	x	x	Y	Y	x
Any properties of color mortgaged	N	x	x	Y	x	x	x	x	N	N	x
Houses are available for sale	Y	x	x	x	N	x	x	x	Y	Y	x
Number of houses on target property	<4	x	x	x	x	4	x	x	<4	<4	x
Other property in color group has fewer houses	N	x	x	x	x	x	Y	x	N	N	x
Hotel on target property	N	x	x	x	x	x	x	Y	N	N	x
Multiple players attempt to purchase same house	N	x	x	x	x	x	x	x	Y	Y	x
Highest bidder	x	x	x	x	x	x	x	x	Y	N	x
Sufficient funds are available for purchase	Y	x	x	x	x	x	x	x	Y	x	N
Actions											
Successful purchase	Y	N	N	N	N	N	N	N	Y	N	N
Target price	Y	N	N	N	N	N	N	N	N	N	N
Auction price	N	N	N	N	N	N	N	N	Y	N	N



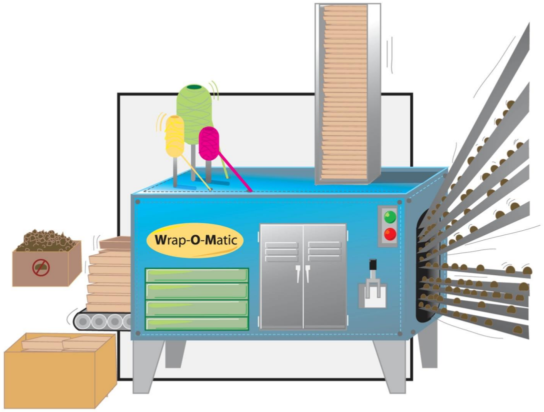
Wrap-O-Matic

Decision Tables



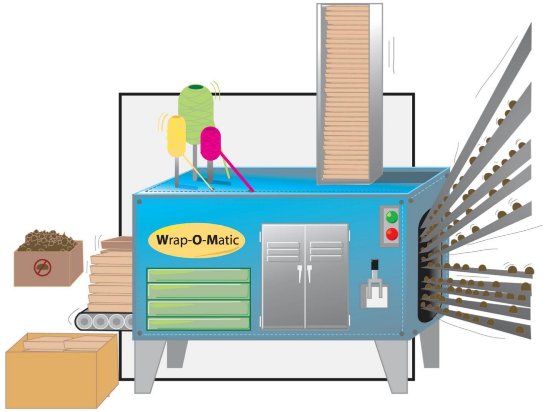
Decision Tables

- Wrapping Rules:
  - Disallows ribbons applied to unwrapped chocolates.
  - Disallows hollow chocolates tied with metallic ribbon.
  - Uses the gentle wrapping algorithm with tissues wrappers.



Decision Tables

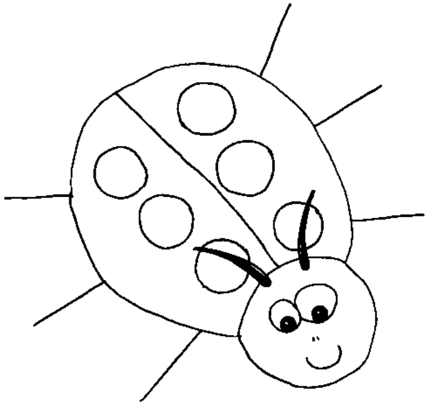
- Wrapping Rules:
  - Uses the rapid wrapping algorithm whenever chocolates do not have ribbons and do not have tissue wrappers.
  - Uses the gentle algorithm whenever hollow chocolates are tied with ribbons.
  - Uses the normal algorithm for all other cases.



Decision Tables

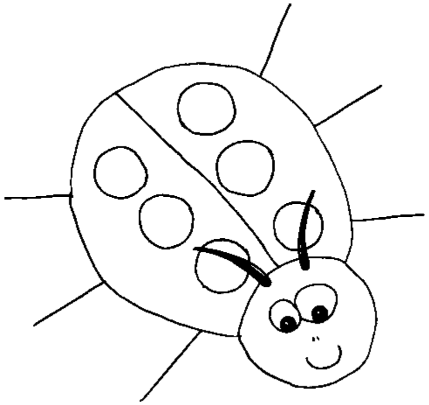
		Rules														
		R01	R02	R03	R04	R05	R06	R07	R08	R09	R10	R11	R12	R13	R14	R15
Conditions	Viscosity	Hollow									Not Hollow					
	Ribbon	Metallic			Other			None			Metallic or Other			None		
	Wrapper	Metallic or Paper	Tissue	None	Metallic or Paper	Tissue	None	Metallic or Paper	Tissue	None	Metallic or Paper	Tissue	None	Metallic or Paper	Tissue	None
Actions	Disallow	x	x	x	.	.	x	.	.	.	.	.	x	.	.	.
	Rapid Algorithm	.	.	.	-	.	-	X	.	x	.	.	.	x	.	x
	Normal Algorithm	.	.	.	.	.	.	.	.	.	x	.	.	.	.	.
	Gentle Algorithm	.	-	.	x	x	x	.	x	.	.	x	.	.	x	.

Wrapping Rules



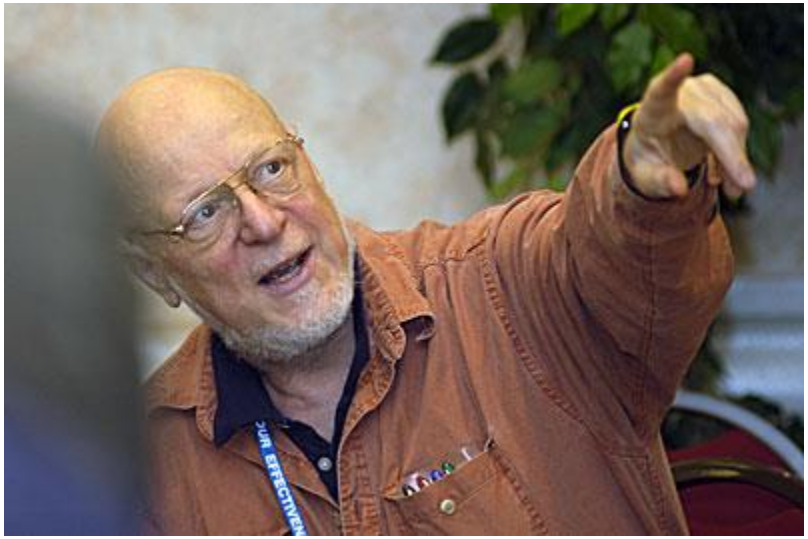
Test Design

State Models

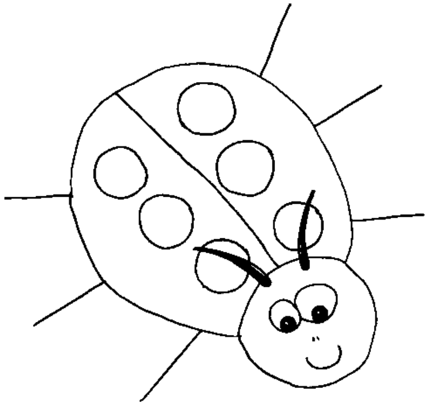


Gerald M. Weinberg

"A state is a situation which can be recognized if it occurs again"



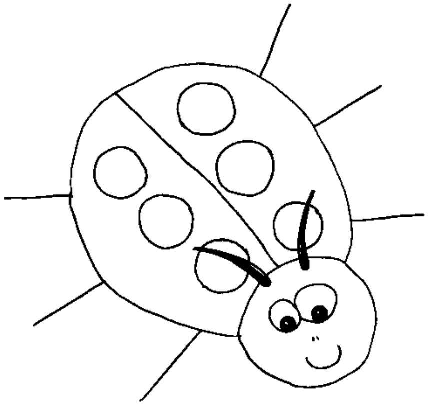
An Introduction to General Systems Thinking
Dorset House



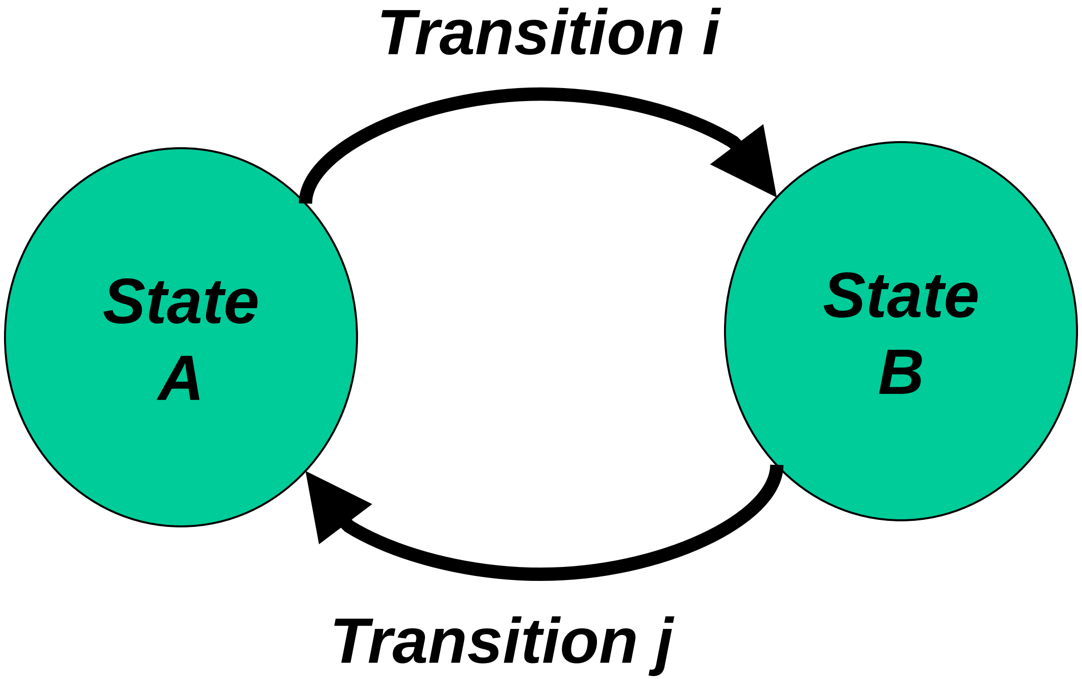
State Models

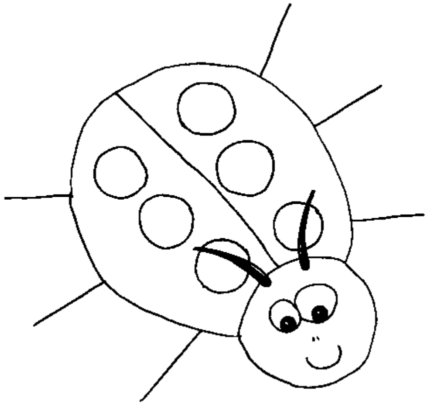
“*Stateful*” Systems

- ❑ Transactions
- ❑ Embedded Systems
- ❑ Process
- ❑ Workflow
- ❑ User Interface



State Models

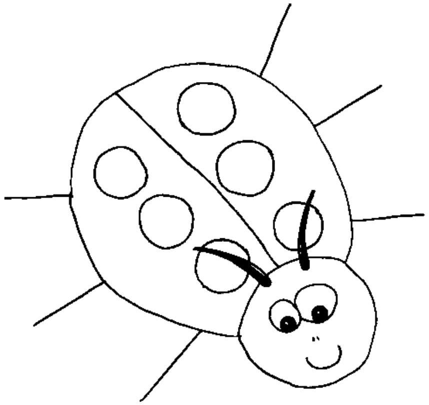




State Models

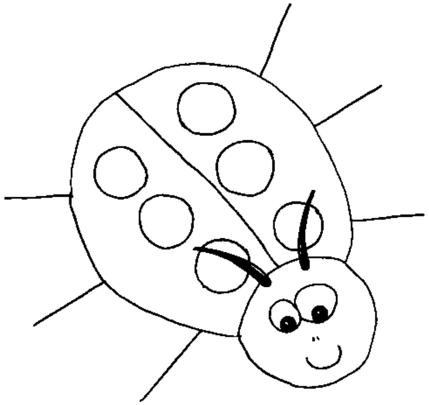
Construction

1. Identify: States
2. Identify: Transitions
3. Identify: Triggers and Outcomes
4. Test: *Get to Each State*
5. Test: *Exercise Each Transition*
6. Test: *Cover Each Path*

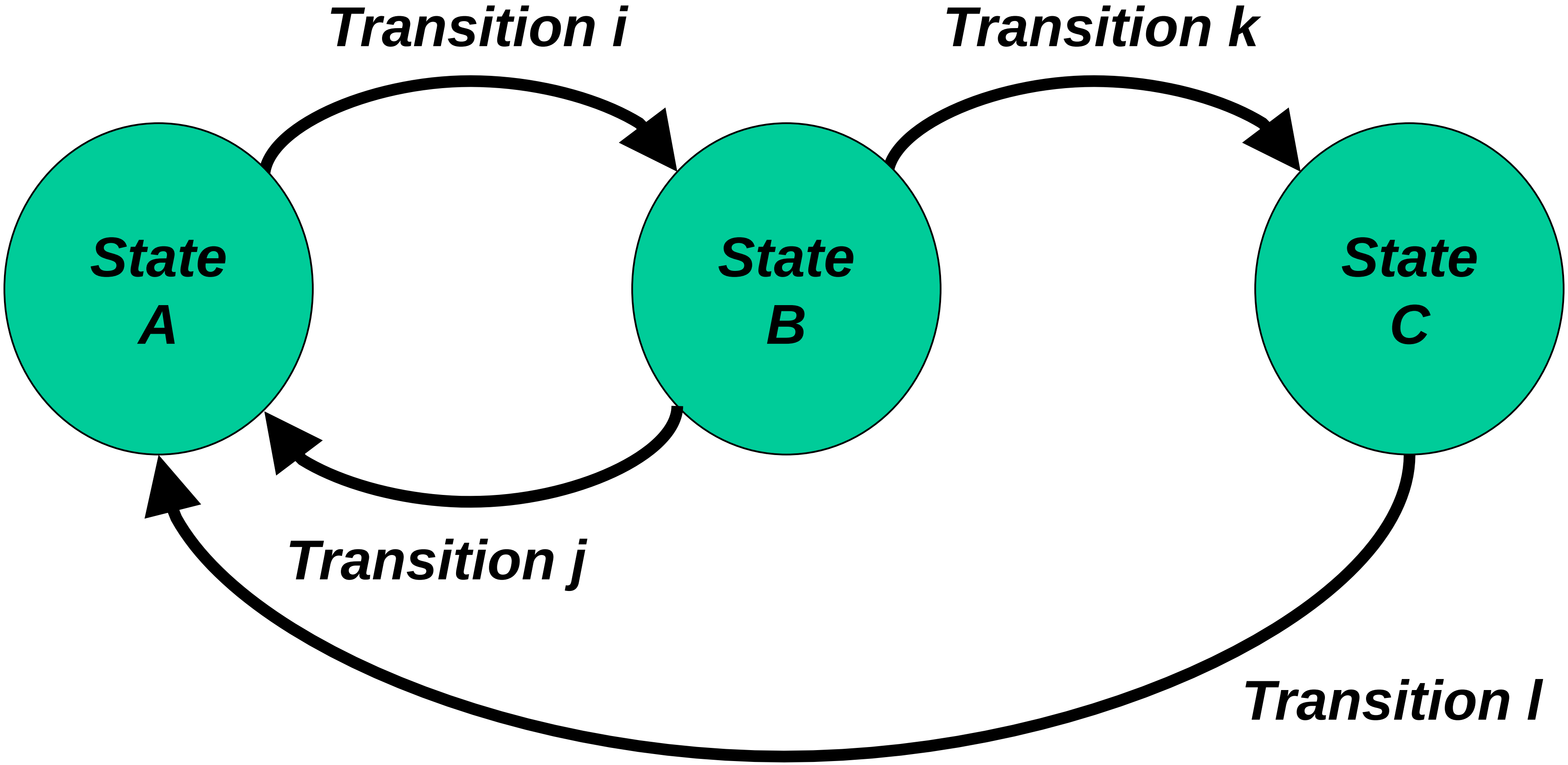


San Diego Singles Pattern

State Models



State Models
San Diego Singles Pattern



Transition i

Transition k

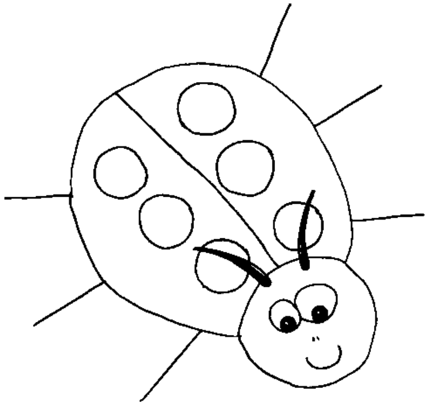
State A

State B

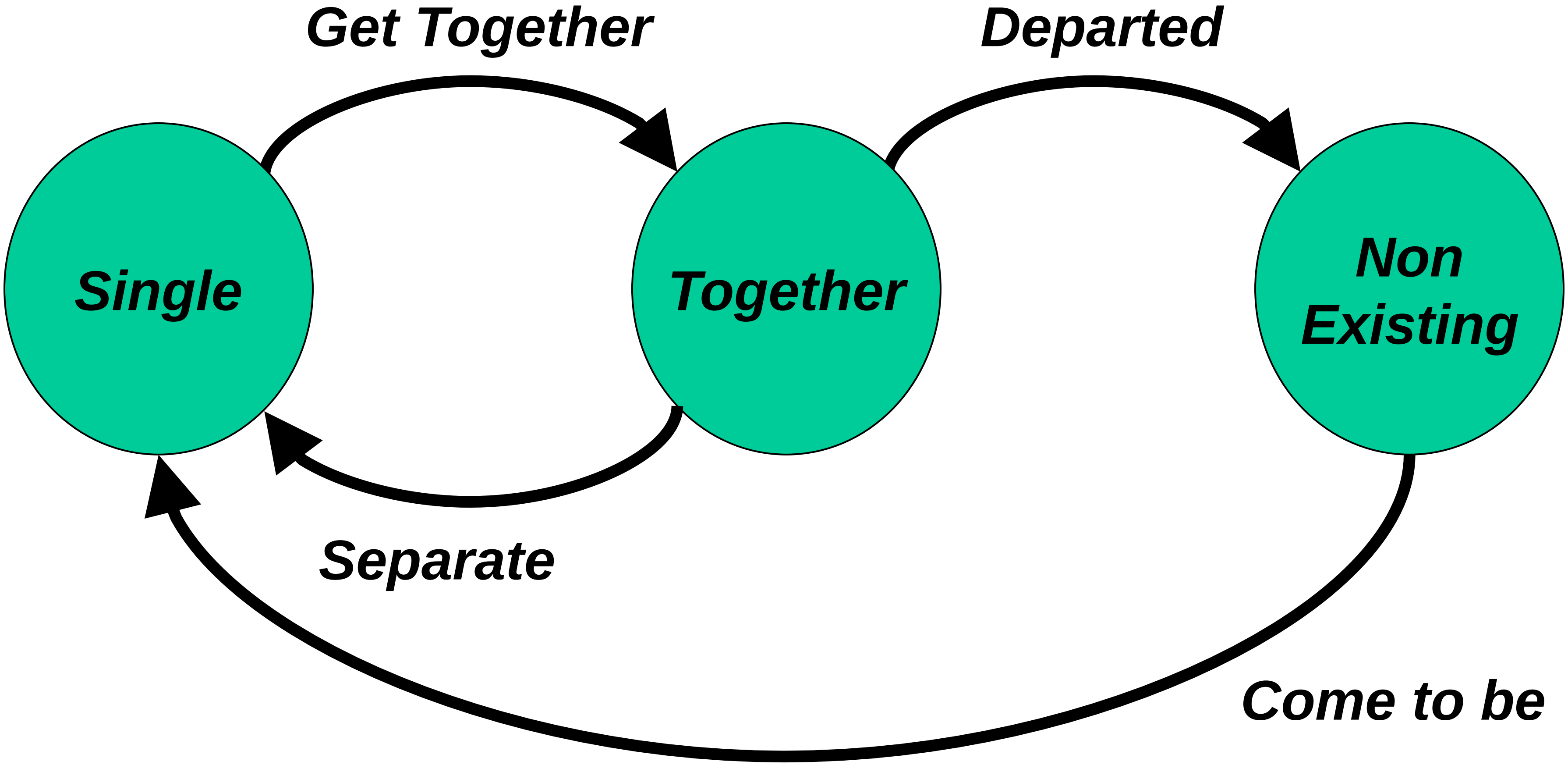
State C

Transition j

Transition l



State Models
San Diego Singles Pattern



Single

Together

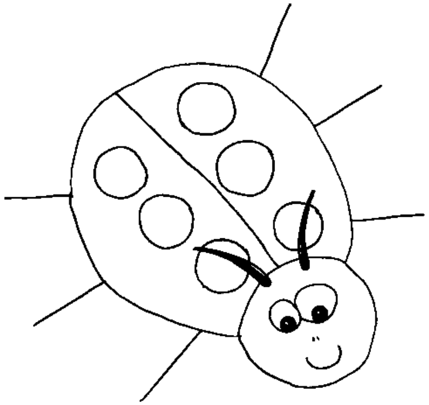
Non Existing

Get Together

Departed

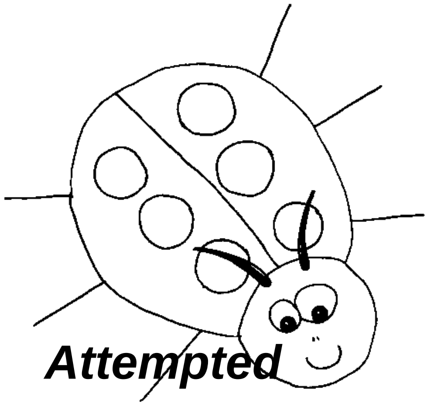
Separate

Come to be

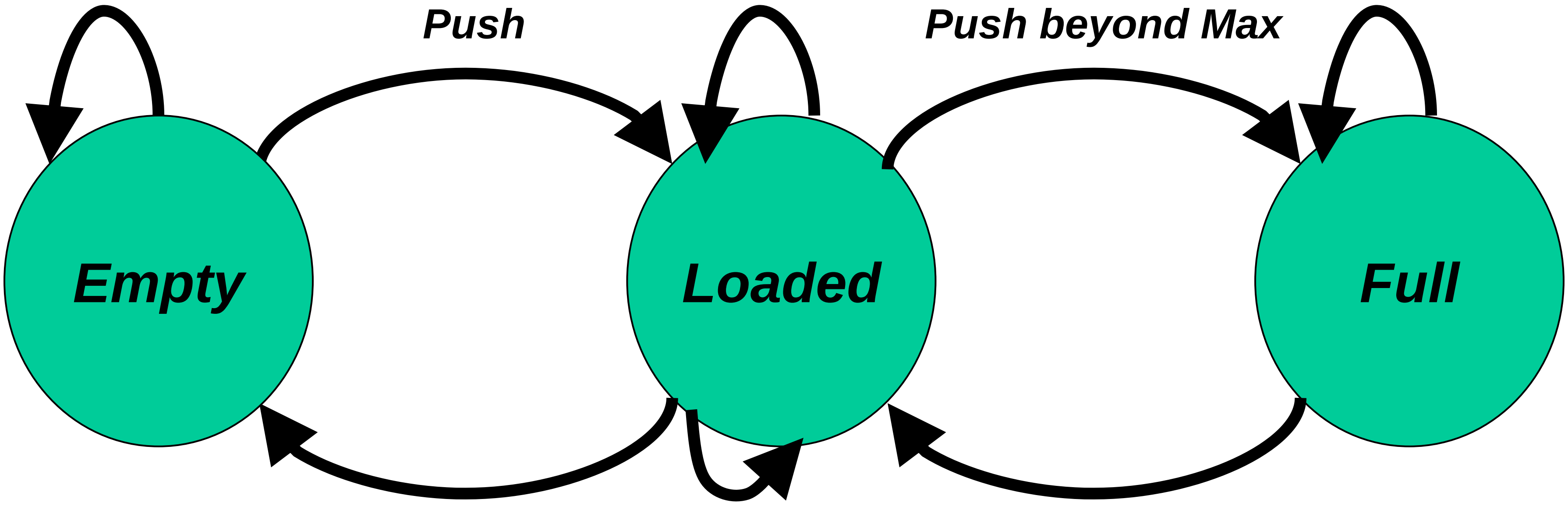


Stack Pattern

State Models



State Models
Stack Pattern



Attempted Pop

Push

Push

Push beyond Max

Attempted Push

Empty

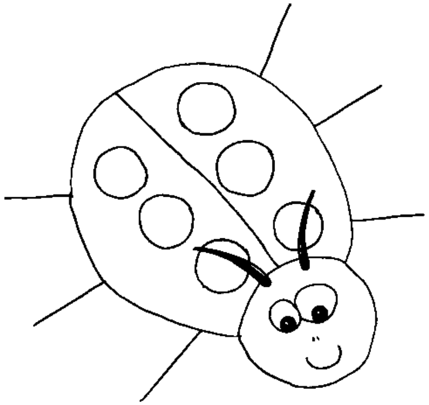
Loaded

Full

Pop beyond Min

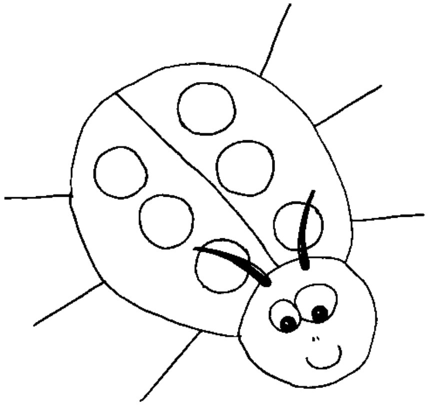
Pop

Pop

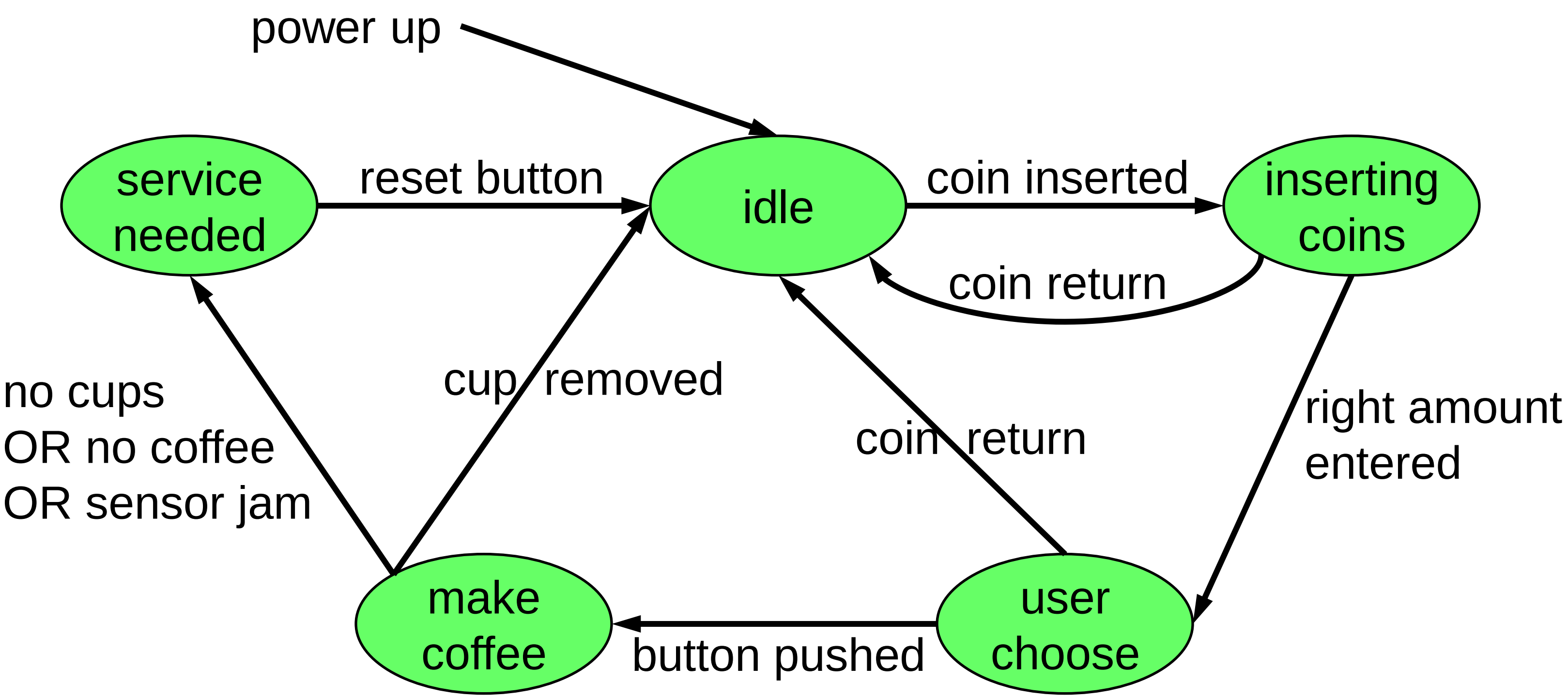


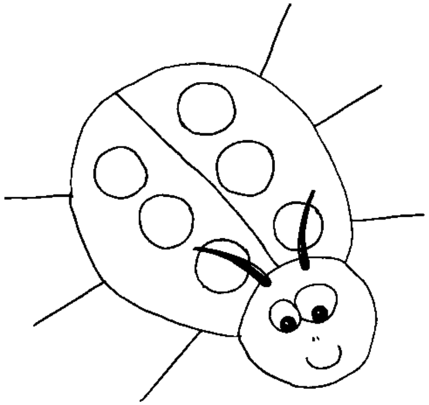
Embedded Systems

State Models



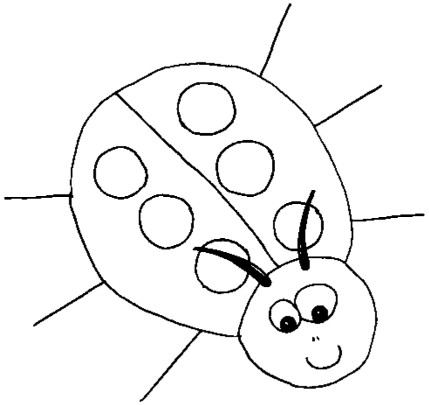
State Models
Coffee Machine



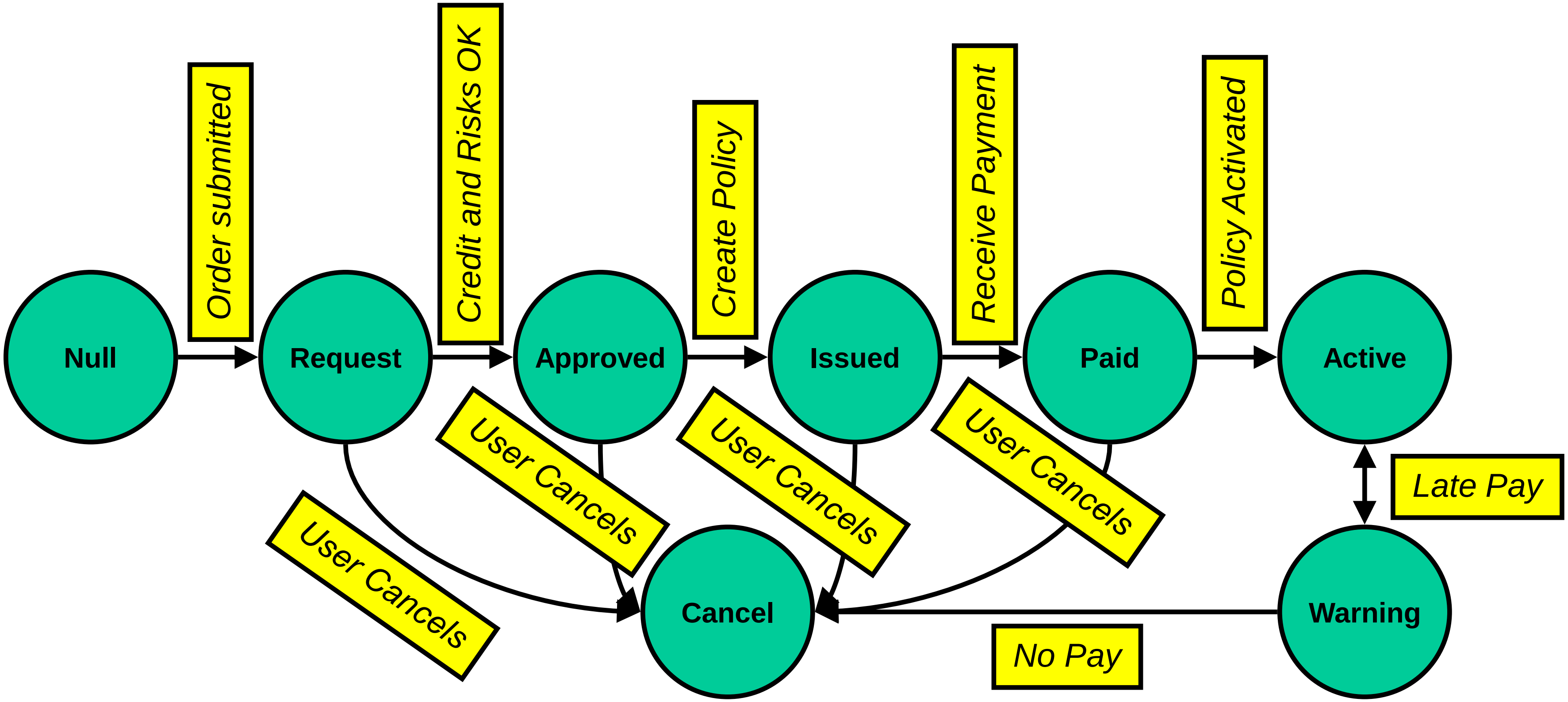


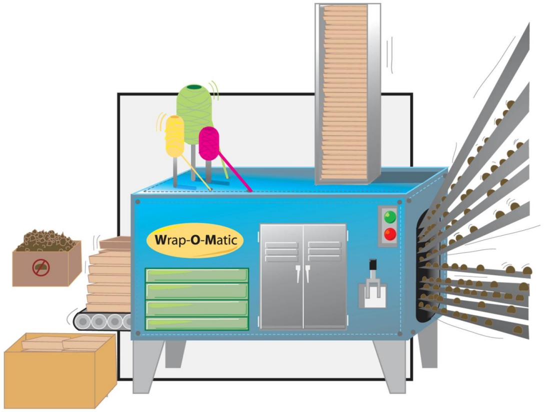
Insurance Transactions

State Models



State Models Transaction

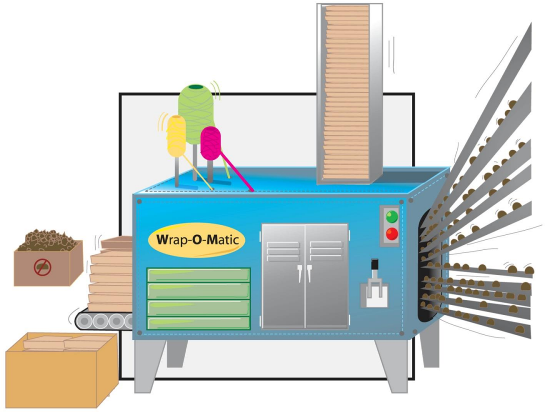


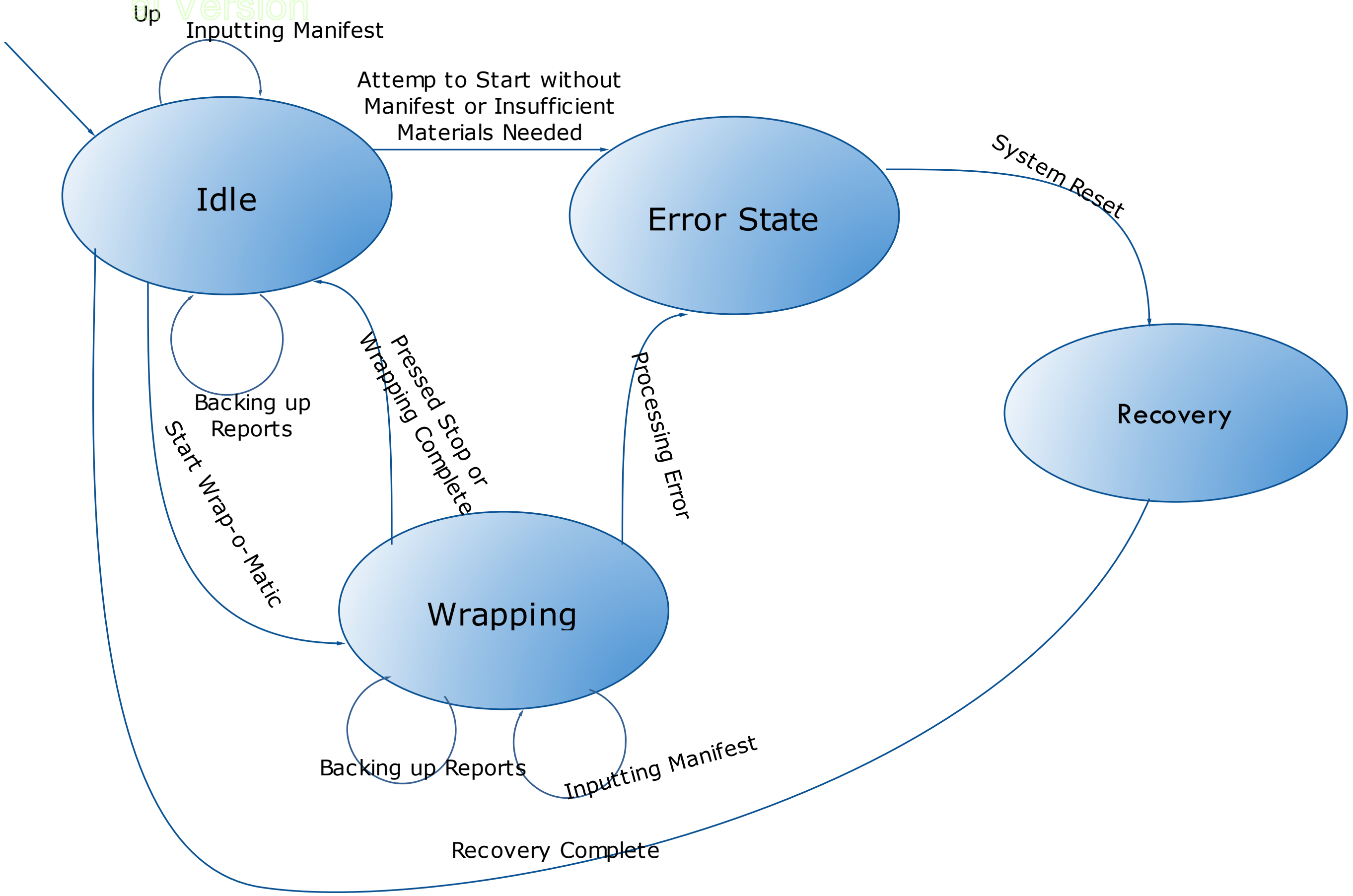


Wrap-O-Matic

State Models

State Models

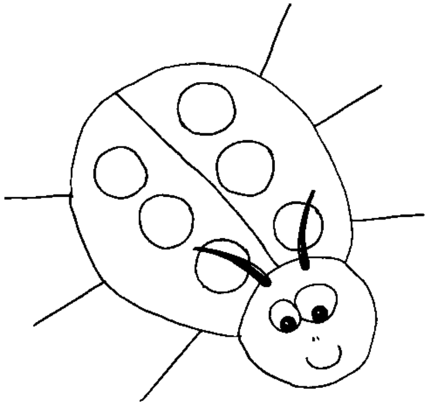




State Models

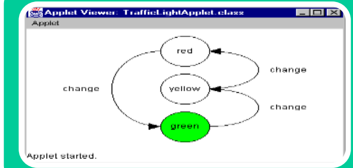
Tools of the Trade

- ❑ Baskets
- ❑ Index Cards
- ❑ Excel
- ❑ Visio
- ❑ Power Point
- ❑ Case Tools



State Models

Model Based Testing

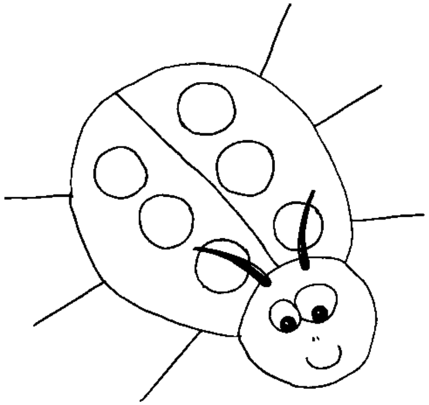


Applet started.

Finite State Machines

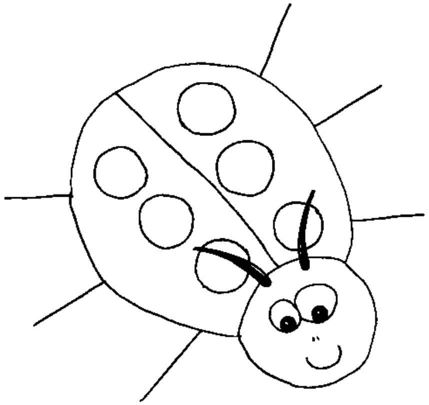
Robots

Probes



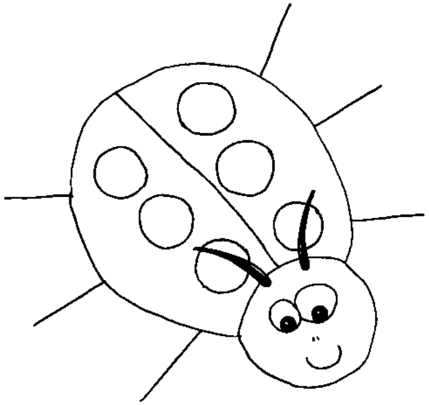
Test Design

Combination Testing

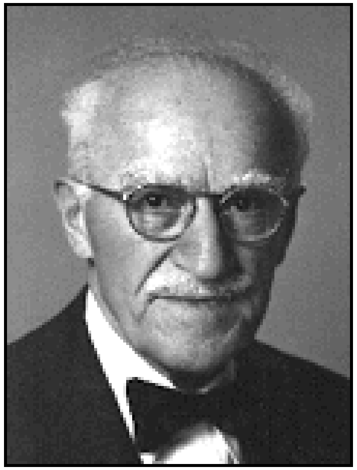


Pareto Principle

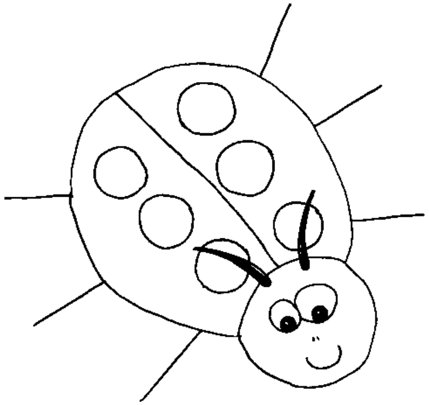
- Vilfredo Pareto, 1848 - 1923, Economist
  - 80% of the wealth was in the hands of 20% of the population



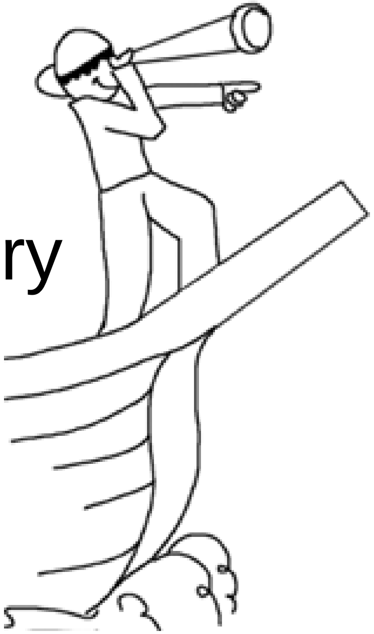
Pareto Principle



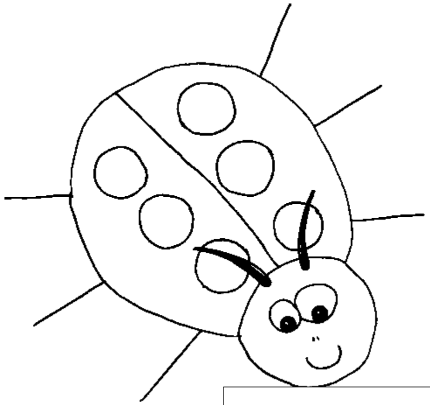
- Joseph Juran, 1903 - 2008, Quality Control Engineer
  - 1950 Quality Control Handbook
  - 20% of the study population accounts for 80% of the measure under consideration
  - “ *… vital few and trivial many …* ”



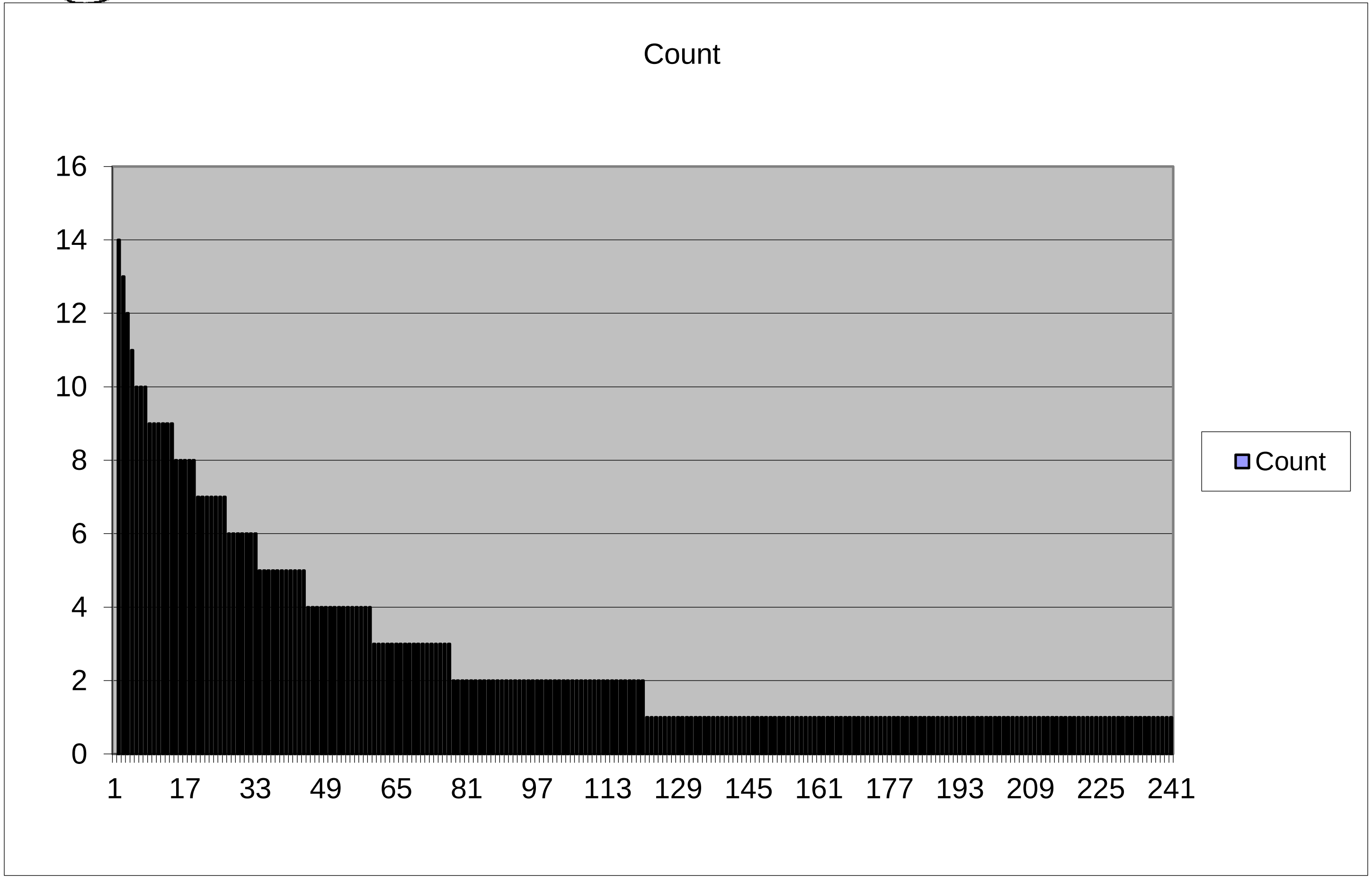
Pareto Analysis

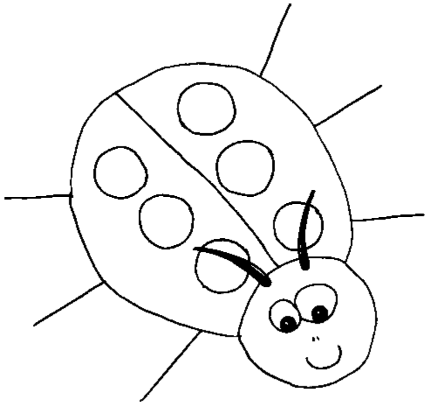


- Pareto Combinations
  - – Start from transaction history
  - – Create histogram
  - – Identify sweet spot
    - 20% of the transaction types
    - 80% of the time



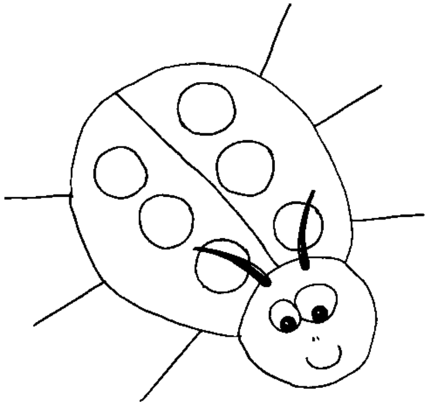
Pareto Analysis





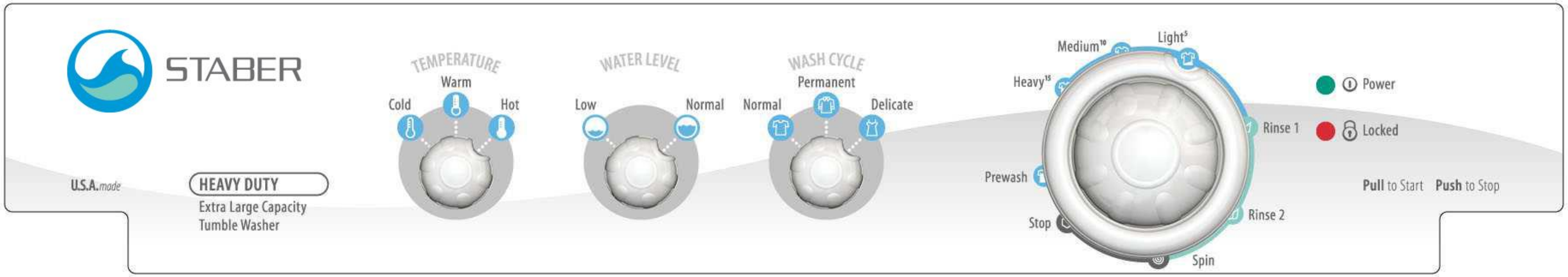
Pairwise Combinations

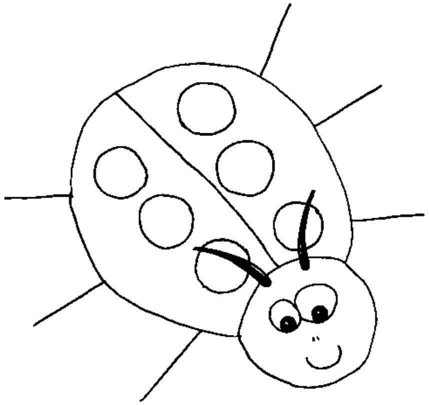
- Identify dependent variables of interest
- For each variable identify values (classes of special concern)
- For each pair of variables define at least one test case which exercise all possible combinations of values



CHERRY
10 AMPS
125/250 VAC
1/3 H.P
125/250 VAC
MIDWEST APPLIANCE PARTS
07-06
6 3094010
E6965C0
COM
NC
NO
INDIA

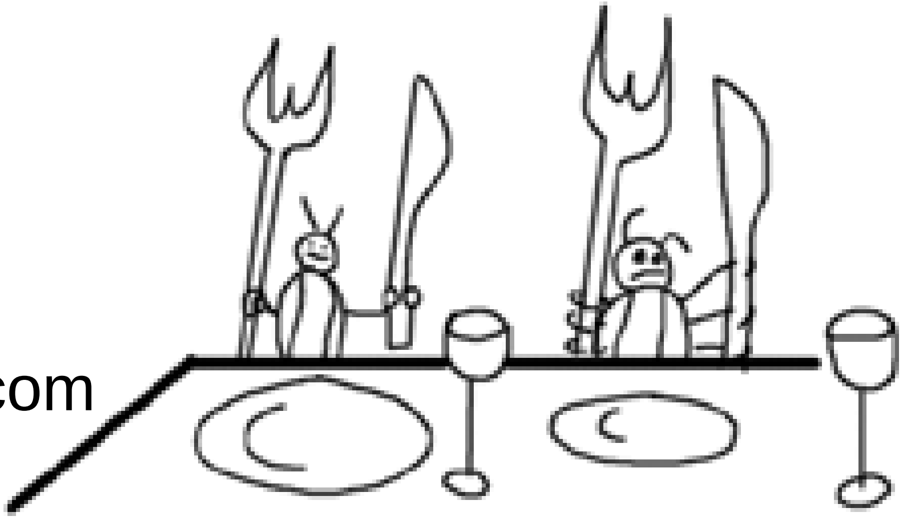
POWER
HEAVY DUTY
NORMAL
CASUAL
DELICATE
Estimated Time Remaining
20
AUTOMATIC CYCLES
TIMED CYCLES
TIMED DRY
QUICK REFRESH
ENHANCED TOUCH UP
DRY CYCLES
TIME ADJUST
DRYNESS
Less
Damp

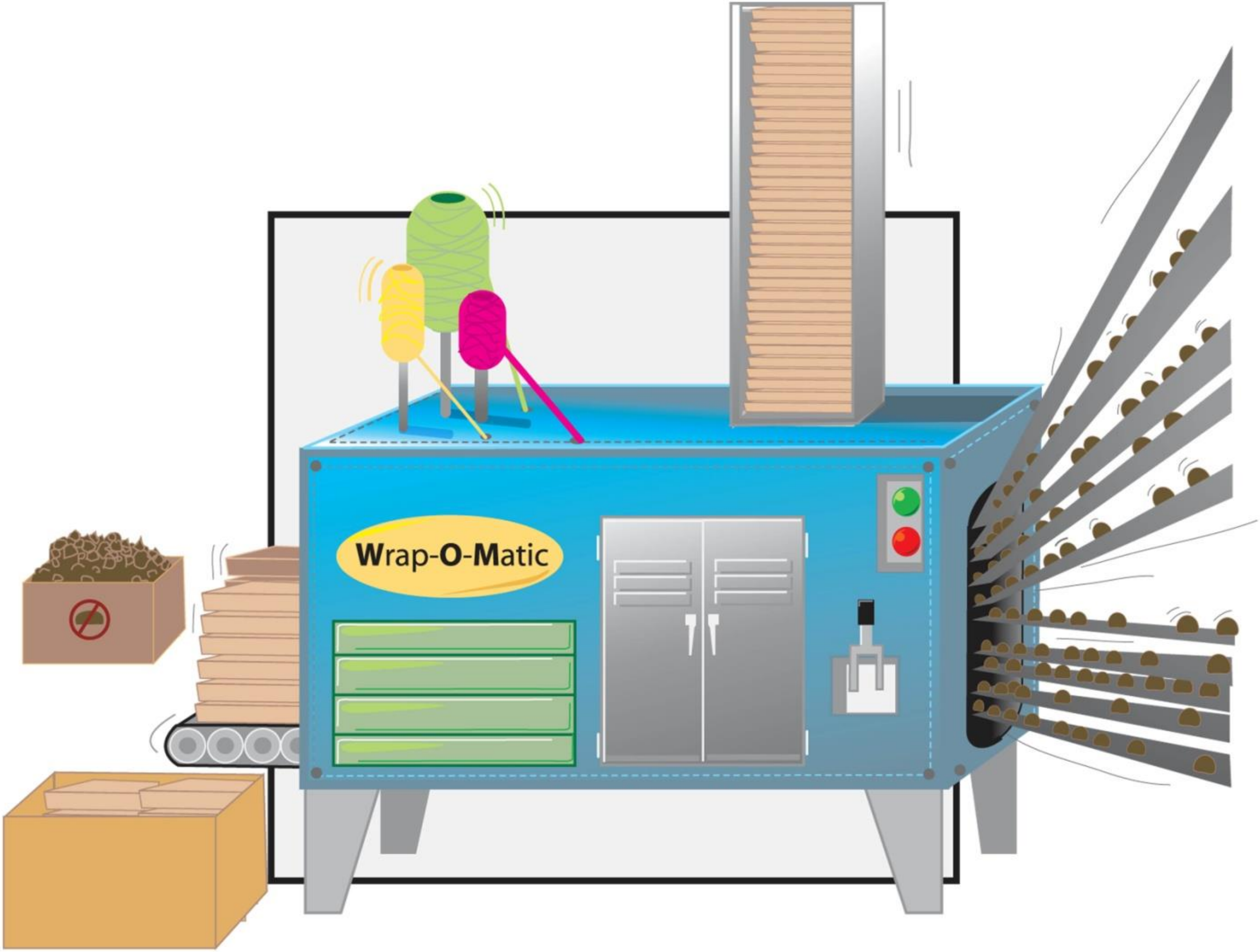

STABER
U.S.A. made
HEAVY DUTY
Extra Large Capacity
Tumble Washer
TEMPERATURE
Cold
Warm
Hot
WATER LEVEL
Low
Normal
WASH CYCLE
Normal
Permanent
Delicate
Heavy
Medium
Light
Rinse 1
Prewash
Rinse 2
Stop
Spin
Power
Locked
Pull to Start
Push to Stop

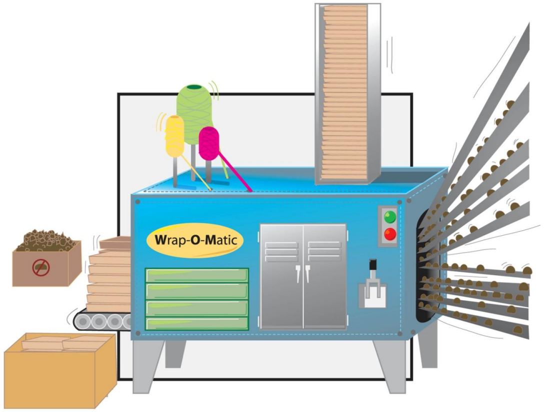


Pairwise Combinations

- Pairwise Combinations Test Tools
  - allpairs
    - www.satisfice.com
  - pict
    - www.microsoft.com
    - www.amibugshare.com

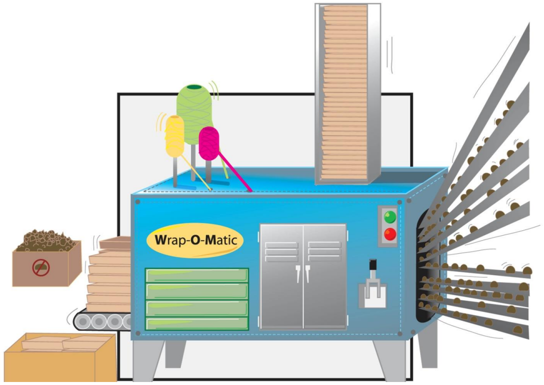



Wrap-O-Matic



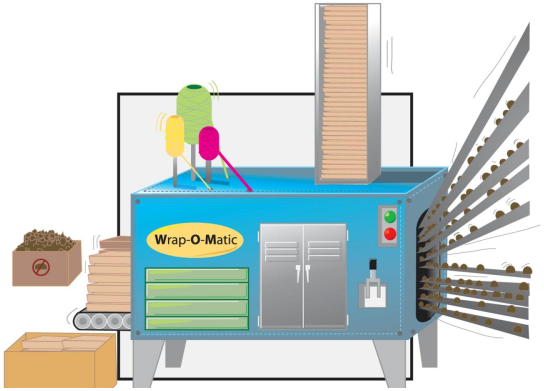
Pairwise Combinations

```
#
# This is a sample model for testing volume create/delete functions
#
Choc:          White, Dark, Milk, Bitter, Semi, Swiss, Belg, Fudge
Config:        Truffle, Bar, Turtle, BonBon, Praline, Filled
Paper:         Wax, Foil, Paper, Tissue, None
Ribbon:        Thread, Cord, Wide, Wire, None
ConvSp:        Slow, Med, Fast, VeryFast
BoxType:       Heart, Rect, Circle, Bag
Visc:          Solid, Jelly, SemiSolid, Hollow
BoxSiz:        Small, Med, Large
Weight:        TooLight, InRange, TooHeavy
Size:          One, Two, Three
```



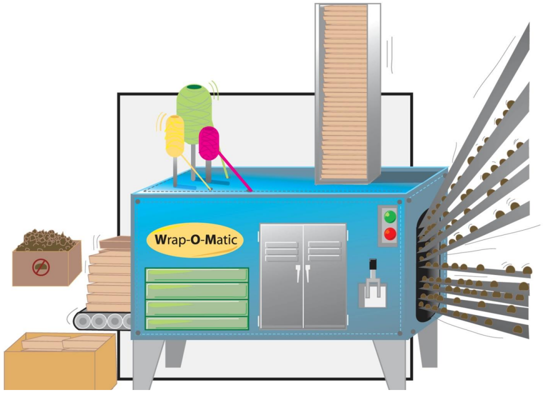
Pairwise Combinations

ID	Choc	Config	Paper	Ribbon	ConvSp	BoxType	Visc	BoxSiz	Weight	Size
1	Swiss	Filled	Tissue	Thread	Med	Rect	Hollow	Small	TooLight	Two
2	Swiss	Bar	None	Wire	Slow	Heart	Jelly	Med	TooHeavy	Three
3	Belg	Turtle	Tissue	Cord	Fast	Bag	Solid	Large	InRange	One
4	Dark	Bar	Wax	None	VeryFast	Circle	SemiSolid	Large	TooLight	One
5	Belg	Truffle	Paper	Wide	VeryFast	Heart	SemiSolid	Small	TooHeavy	Two
6	Swiss	Truffle	Foil	Wide	Fast	Circle	Hollow	Med	InRange	One
7	Belg	Bar	Foil	None	Med	Rect	Jelly	Small	InRange	Three
8	Semi	BonBon	Paper	Thread	VeryFast	Bag	Solid	Med	TooLight	Three
9	Bitter	Bar	Wax	Thread	Slow	Heart	Solid	Large	InRange	Two
10	Semi	Praline	Wax	Cord	Med	Circle	Jelly	Small	TooHeavy	One
11	Fudge	Filled	Wax	Wire	Fast	Bag	Hollow	Large	TooHeavy	Three
12	Semi	Filled	None	Cord	Slow	Rect	SemiSolid	Med	InRange	Two
13	Semi	Turtle	Foil	None	Fast	Heart	Jelly	Large	TooLight	Two
14	Bitter	Praline	Tissue	Wide	Med	Bag	SemiSolid	Med	TooLight	Three
15	Fudge	Truffle	None	Wire	Slow	Rect	Solid	Small	TooLight	One



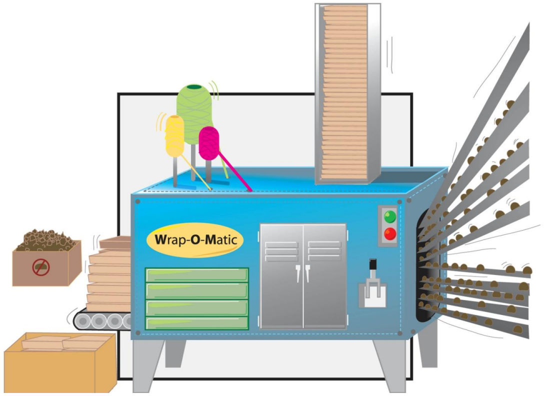
Pairwise Combinations

ID	Choc	Config	Paper	Ribbon	ConvSp	BoxType	Visc	BoxSiz	Weight	Size
16	White	BonBon	Foil	Wire	VeryFast	Rect	Hollow	Large	TooHeavy	Two
17	Dark	Filled	Paper	None	Med	Heart	Solid	Small	TooHeavy	One
18	White	Praline	Paper	Wide	Slow	Circle	Solid	Large	InRange	Three
19	Milk	Filled	Tissue	None	VeryFast	Bag	Jelly	Med	TooHeavy	Two
20	Dark	BonBon	None	Thread	Fast	Circle	Jelly	Small	InRange	Two
21	White	Bar	None	Cord	Slow	Bag	Hollow	Med	TooLight	One
22	Bitter	Turtle	Paper	Thread	VeryFast	Rect	Hollow	Small	TooHeavy	One
23	Belg	BonBon	Tissue	Wire	Slow	Circle	SemiSolid	Med	TooLight	One
24	Milk	BonBon	None	Cord	Med	Heart	SemiSolid	Large	InRange	Three
25	Swiss	Praline	Foil	Cord	VeryFast	Rect	SemiSolid	Large	InRange	Two
26	Milk	Bar	Paper	Wide	Fast	Rect	Hollow	Small	TooLight	One
27	Bitter	BonBon	Wax	Wide	Fast	Bag	Jelly	Small	InRange	Two
28	Belg	Filled	Wax	None	Slow	Heart	Hollow	Med	InRange	Two
29	Milk	Truffle	Foil	Thread	Slow	Bag	Jelly	Large	InRange	Three
30	Swiss	Turtle	Wax	Wire	Med	Bag	SemiSolid	Med	InRange	Three



Pairwise Combinations

ID	Choc	Config	Paper	Ribbon	ConvSp	BoxType	Visc	BoxSiz	Weight	Size
31	Milk	Praline	None	Wire	Fast	Heart	Solid	Large	InRange	Three
32	Semi	Truffle	Wax	Wide	Med	Rect	Hollow	Small	InRange	One
33	Bitter	Truffle	Tissue	Cord	Fast	Circle	SemiSolid	Large	TooHeavy	Three
34	Fudge	Bar	Tissue	Thread	VeryFast	Heart	SemiSolid	Med	InRange	Two
35	Dark	Truffle	Foil	Wide	Slow	Bag	Hollow	Med	InRange	Three
36	Swiss	BonBon	None	None	VeryFast	Bag	Solid	Small	TooLight	Three
37	Fudge	BonBon	Paper	Cord	Med	Circle	Jelly	Small	InRange	Two
38	Fudge	Praline	Foil	None	Slow	Heart	Hollow	Small	InRange	One
39	White	Turtle	None	Wide	Fast	Circle	Jelly	Small	InRange	One
40	White	Truffle	Wax	Thread	Med	Heart	SemiSolid	Med	InRange	Three
41	White	Filled	Tissue	None	Slow	Circle	SemiSolid	Small	TooHeavy	Three
42	Fudge	Filled	Foil	Wide	Slow	Bag	Solid	Large	TooHeavy	One
43	Belg	Praline	None	Thread	Slow	Circle	Solid	Med	TooLight	One
44	Bitter	Turtle	Foil	None	Slow	Bag	SemiSolid	Small	TooLight	Three
45	Bitter	Filled	None	Wire	Slow	Heart	Jelly	Med	TooLight	One



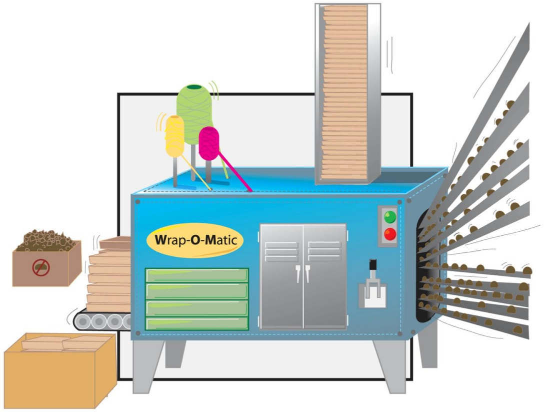
Pairwise Combinations

ID	Choc	Config	Paper	Ribbon	ConvSp	BoxType	Visc	BoxSiz	Weight	Size
46	Dark	Praline	Paper	Wire	Med	Rect	Hollow	Med	TooLight	One
47	Semi	Bar	Tissue	Wire	VeryFast	Rect	SemiSolid	Small	InRange	Three
48	Milk	Turtle	Wax	None	Med	Circle	SemiSolid	Large	InRange	Two
49	Dark	Turtle	Tissue	Cord	VeryFast	Bag	Hollow	Med	InRange	One
50	Swiss	Truffle	Paper	None	Slow	Bag	Jelly	Large	InRange	Two
51	Fudge	Turtle	Paper	None	Fast	Heart	Hollow	Large	InRange	Three

Constrained Pairwise Combinations

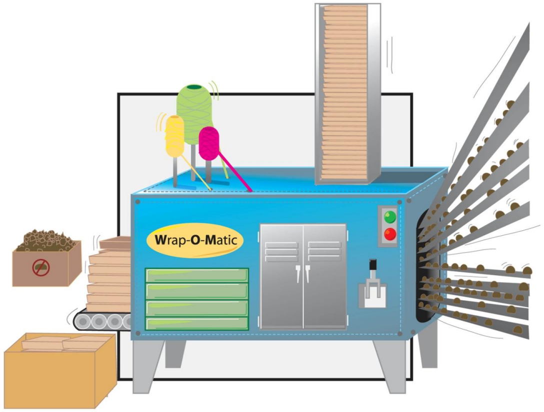
```
#
# This is a Wrap O Matic combinations testing example
#
Choc:                White, Dark, Milk, Bitter, Semi, Swiss, Belg, Fudge
Config:              Truffle, Bar, Turtle, BonBon, Praline, Filled
Paper:               Wax, Foil, Paper, Tissue, None
Ribbon:              Thread, Cord, Wide, Wire, None
ConvSp:              Slow, Med, Fast, VeryFast
BoxType:             Heart, Rect, Circle, Bag
Visc:                Solid, Jelly, SemiSolid, Hollow
BoxSiz:              Small, Med, Large
Weight:              TooLight, InRange, TooHeavy
Size:                One, Two, Three
#
# No Wire on Hollow chocolates
#
IF [Ribbon] in {"Wire"} THEN [Visc] in {"Solid", "Jelly", "SemiSolid"};
#
# No Ribbon if No Paper
#
IF [Paper] in {"None"} THEN [Ribbon] in {"None"};
```

Wrapping Rules

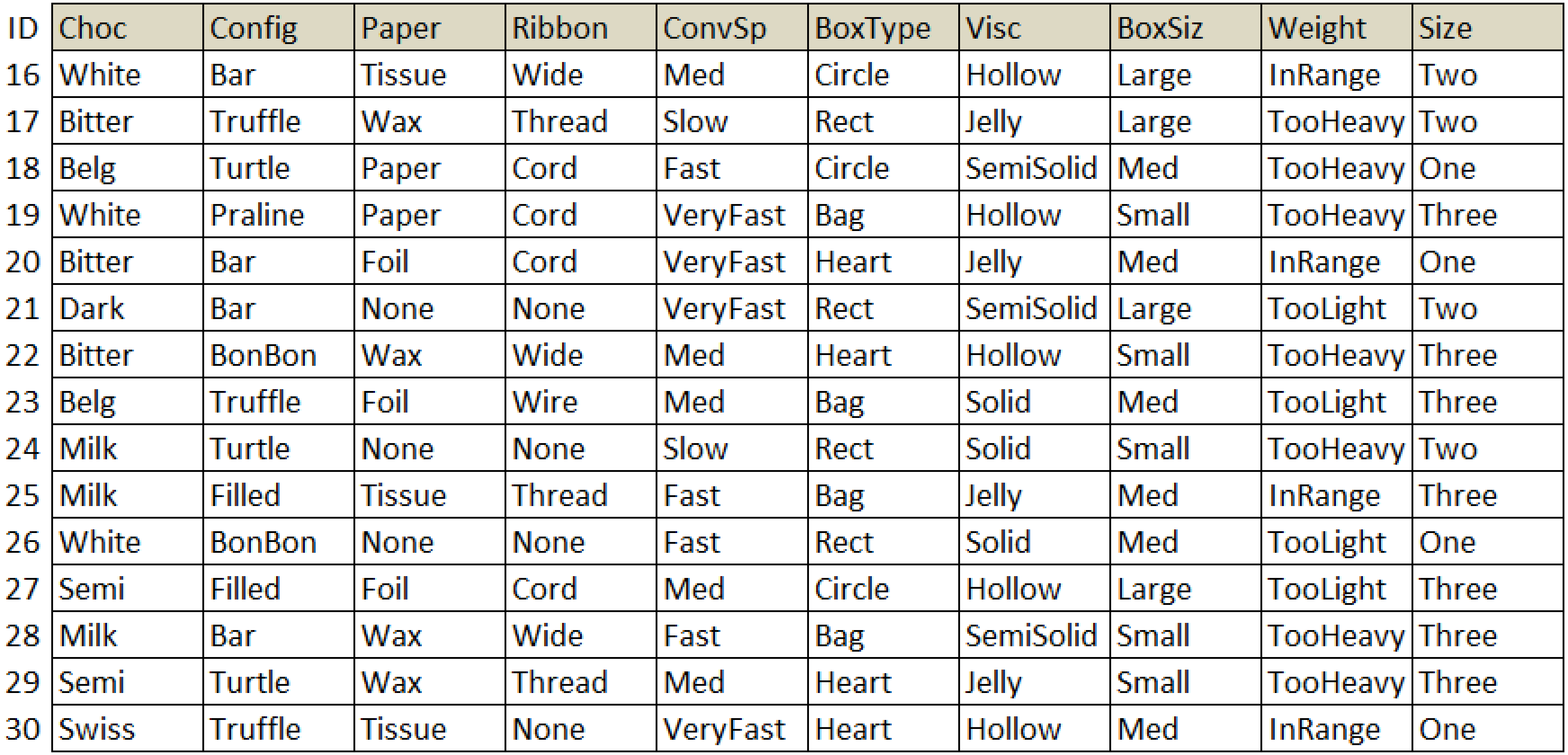


Constrained Pairwise Combinations

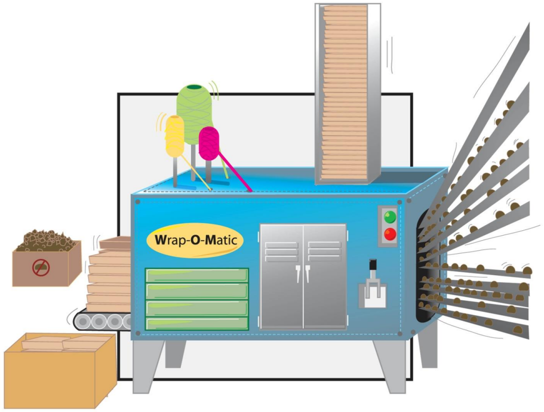
ID	Choc	Config	Paper	Ribbon	ConvSp	BoxType	Visc	BoxSiz	Weight	Size
1	Swiss	Filled	None	None	Slow	Circle	SemiSolid	Small	TooLight	Two
2	Fudge	Turtle	Tissue	Wide	VeryFast	Bag	Jelly	Large	InRange	One
3	Dark	Praline	Paper	Thread	Med	Heart	Solid	Med	TooHeavy	Three
4	Fudge	BonBon	Foil	Cord	Fast	Rect	Hollow	Large	TooHeavy	Two
5	Milk	Praline	Wax	Cord	VeryFast	Circle	Hollow	Med	TooLight	One
6	Belg	Praline	Wax	Wire	Fast	Heart	Jelly	Small	InRange	Two
7	Swiss	Praline	Foil	None	Med	Rect	SemiSolid	Large	InRange	Three
8	Bitter	Turtle	Foil	Wide	Fast	Bag	Solid	Small	TooLight	Three
9	Dark	BonBon	Paper	Thread	Slow	Bag	SemiSolid	Small	InRange	One
10	Semi	Bar	Tissue	Wire	VeryFast	Rect	Solid	Small	TooHeavy	One
11	Milk	BonBon	Paper	Wire	Med	Circle	Jelly	Large	TooLight	Three
12	Semi	BonBon	None	None	Slow	Bag	Solid	Med	InRange	Two
13	Fudge	Praline	Tissue	Wide	Slow	Heart	SemiSolid	Med	TooLight	Three
14	Fudge	Filled	Wax	None	Med	Heart	Solid	Large	TooHeavy	One
15	Belg	Bar	Foil	Thread	Slow	Rect	Hollow	Med	TooLight	Three



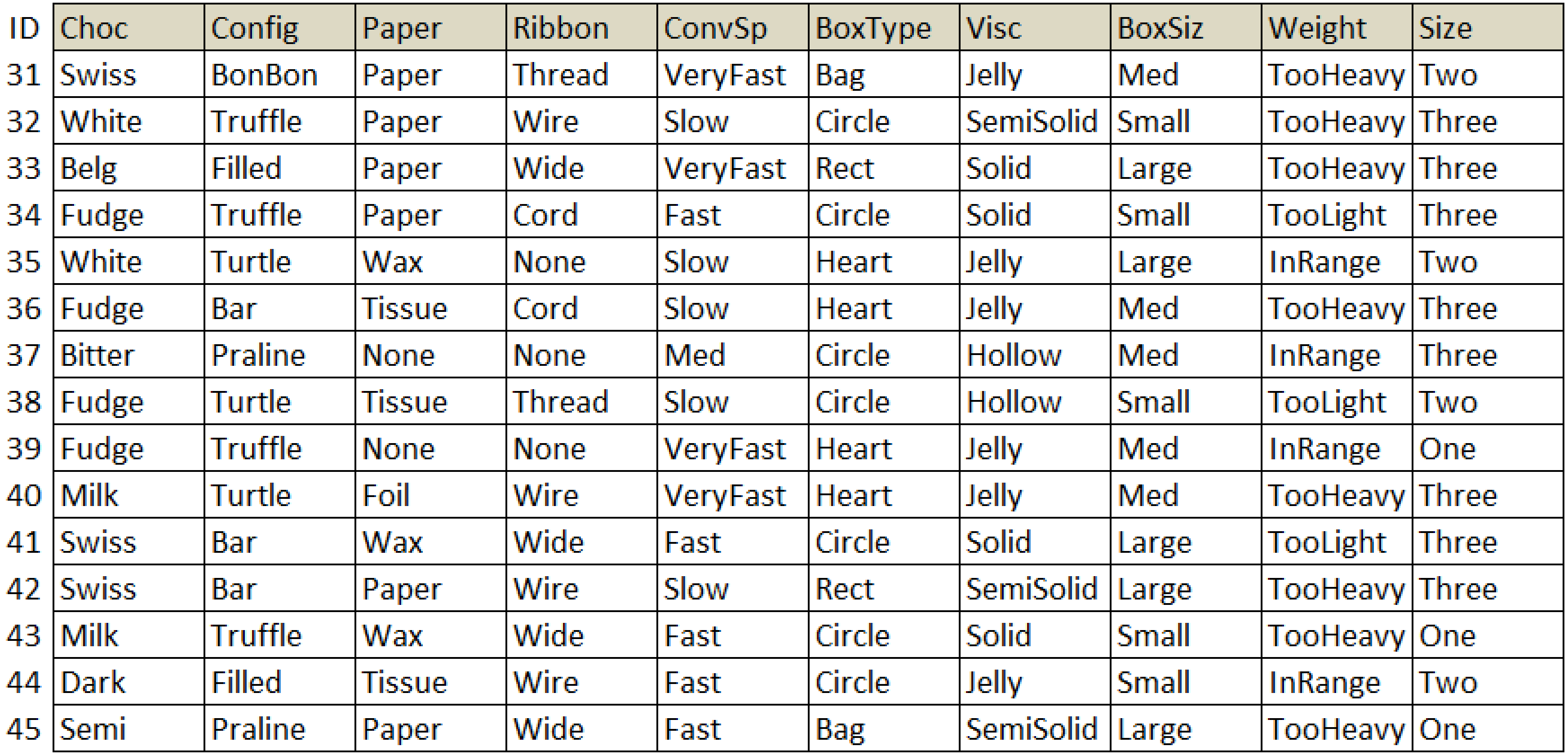
Constrained Pairwise Combinations



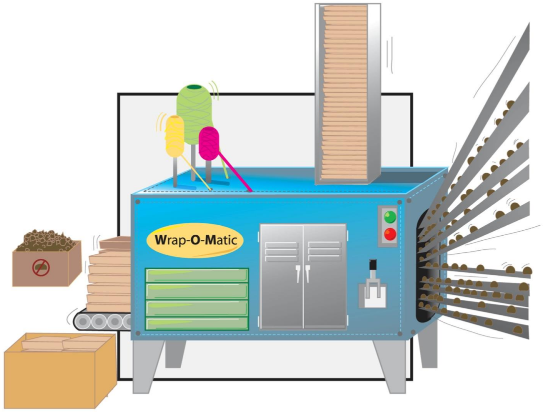
ID	Choc	Config	Paper	Ribbon	ConvSp	BoxType	Visc	BoxSiz	Weight	Size
16	White	Bar	Tissue	Wide	Med	Circle	Hollow	Large	InRange	Two
17	Bitter	Truffle	Wax	Thread	Slow	Rect	Jelly	Large	TooHeavy	Two
18	Belg	Turtle	Paper	Cord	Fast	Circle	SemiSolid	Med	TooHeavy	One
19	White	Praline	Paper	Cord	VeryFast	Bag	Hollow	Small	TooHeavy	Three
20	Bitter	Bar	Foil	Cord	VeryFast	Heart	Jelly	Med	InRange	One
21	Dark	Bar	None	None	VeryFast	Rect	SemiSolid	Large	TooLight	Two
22	Bitter	BonBon	Wax	Wide	Med	Heart	Hollow	Small	TooHeavy	Three
23	Belg	Truffle	Foil	Wire	Med	Bag	Solid	Med	TooLight	Three
24	Milk	Turtle	None	None	Slow	Rect	Solid	Small	TooHeavy	Two
25	Milk	Filled	Tissue	Thread	Fast	Bag	Jelly	Med	InRange	Three
26	White	BonBon	None	None	Fast	Rect	Solid	Med	TooLight	One
27	Semi	Filled	Foil	Cord	Med	Circle	Hollow	Large	TooLight	Three
28	Milk	Bar	Wax	Wide	Fast	Bag	SemiSolid	Small	TooHeavy	Three
29	Semi	Turtle	Wax	Thread	Med	Heart	Jelly	Small	TooHeavy	Three
30	Swiss	Truffle	Tissue	None	VeryFast	Heart	Hollow	Med	InRange	One



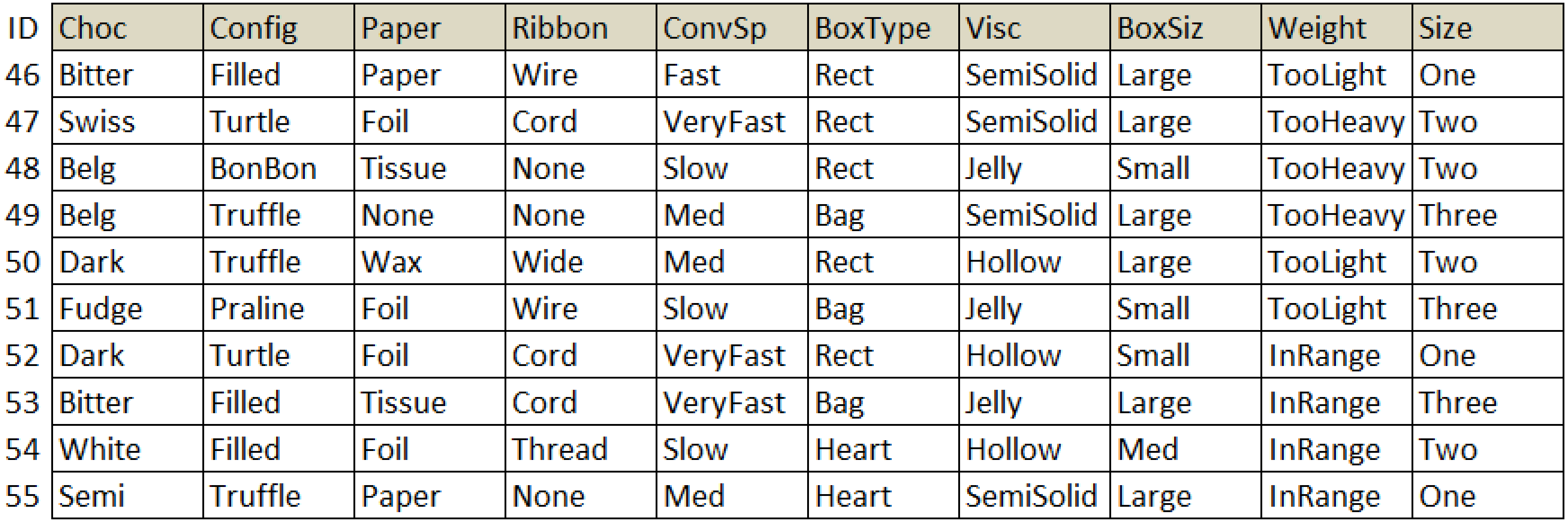
Constrained Pairwise Combinations



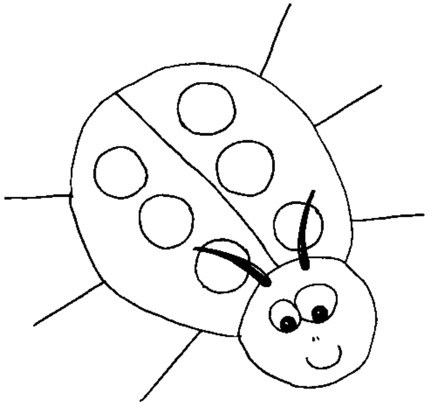
ID	Choc	Config	Paper	Ribbon	ConvSp	BoxType	Visc	BoxSiz	Weight	Size
31	Swiss	BonBon	Paper	Thread	VeryFast	Bag	Jelly	Med	TooHeavy	Two
32	White	Truffle	Paper	Wire	Slow	Circle	SemiSolid	Small	TooHeavy	Three
33	Belg	Filled	Paper	Wide	VeryFast	Rect	Solid	Large	TooHeavy	Three
34	Fudge	Truffle	Paper	Cord	Fast	Circle	Solid	Small	TooLight	Three
35	White	Turtle	Wax	None	Slow	Heart	Jelly	Large	InRange	Two
36	Fudge	Bar	Tissue	Cord	Slow	Heart	Jelly	Med	TooHeavy	Three
37	Bitter	Praline	None	None	Med	Circle	Hollow	Med	InRange	Three
38	Fudge	Turtle	Tissue	Thread	Slow	Circle	Hollow	Small	TooLight	Two
39	Fudge	Truffle	None	None	VeryFast	Heart	Jelly	Med	InRange	One
40	Milk	Turtle	Foil	Wire	VeryFast	Heart	Jelly	Med	TooHeavy	Three
41	Swiss	Bar	Wax	Wide	Fast	Circle	Solid	Large	TooLight	Three
42	Swiss	Bar	Paper	Wire	Slow	Rect	SemiSolid	Large	TooHeavy	Three
43	Milk	Truffle	Wax	Wide	Fast	Circle	Solid	Small	TooHeavy	One
44	Dark	Filled	Tissue	Wire	Fast	Circle	Jelly	Small	InRange	Two
45	Semi	Praline	Paper	Wide	Fast	Bag	SemiSolid	Large	TooHeavy	One



Constrained Pairwise Combinations

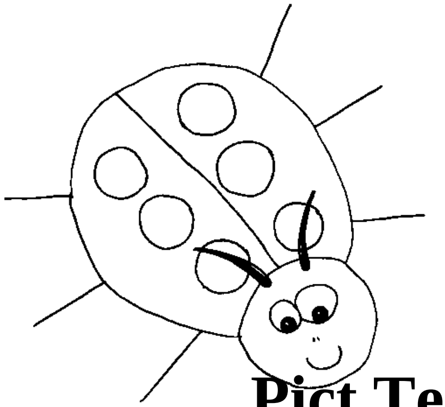


ID	Choc	Config	Paper	Ribbon	ConvSp	BoxType	Visc	BoxSiz	Weight	Size
46	Bitter	Filled	Paper	Wire	Fast	Rect	SemiSolid	Large	TooLight	One
47	Swiss	Turtle	Foil	Cord	VeryFast	Rect	SemiSolid	Large	TooHeavy	Two
48	Belg	BonBon	Tissue	None	Slow	Rect	Jelly	Small	TooHeavy	Two
49	Belg	Truffle	None	None	Med	Bag	SemiSolid	Large	TooHeavy	Three
50	Dark	Truffle	Wax	Wide	Med	Rect	Hollow	Large	TooLight	Two
51	Fudge	Praline	Foil	Wire	Slow	Bag	Jelly	Small	TooLight	Three
52	Dark	Turtle	Foil	Cord	VeryFast	Rect	Hollow	Small	InRange	One
53	Bitter	Filled	Tissue	Cord	VeryFast	Bag	Jelly	Large	InRange	Three
54	White	Filled	Foil	Thread	Slow	Heart	Hollow	Med	InRange	Two
55	Semi	Truffle	Paper	None	Med	Heart	SemiSolid	Large	InRange	One



Multiple Constraints

Combinations



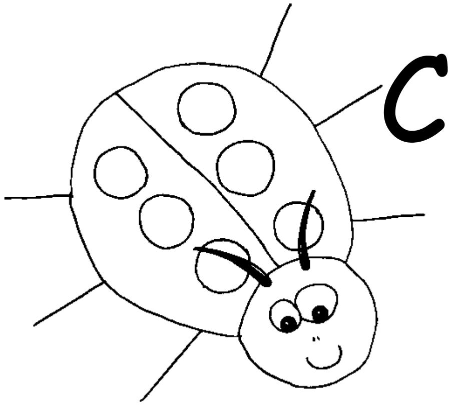
Constrained Pairwise Combinations

Pict Text Input (1)

```
#
# Multiple RC case
#
BES:            Single, Multiple
#
RCs:            1, 2, 3, 4, 5
#
AS1:            On, Off, None
Router1:        On, Off, None
BBIM1:          On, Off, None
MDS_CS1:        On, Off, None
BAS1:           On, Off, None
#
AS2:            On, Off, None
Router2:        On, Off, None
BBIM2:          On, Off, None
MDS_CS2:        On, Off, None
BAS2:           On, Off, None
```

Pict Text Input (2)

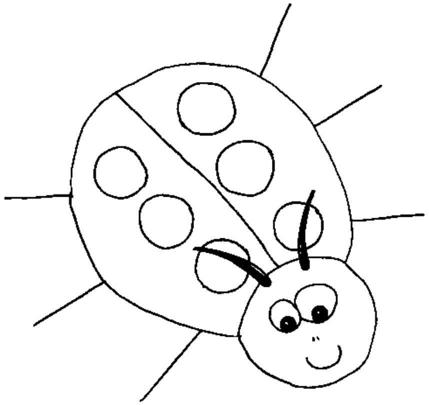
```
#
AS3:            On, Off, None
Router3:        On, Off, None
BBIM3:          On, Off, None
MDS_CS3:        On, Off, None
BAS3:           On, Off, None
#
AS4:            On, Off, None
Router4:        On, Off, None
BBIM4:          On, Off, None
MDS_CS4:        On, Off, None
BAS4:           On, Off, None
#
AS5:            On, Off, None
Router5:        On, Off, None
BBIM5:          On, Off, None
MDS_CS5:        On, Off, None
BAS5:           On, Off, None
```



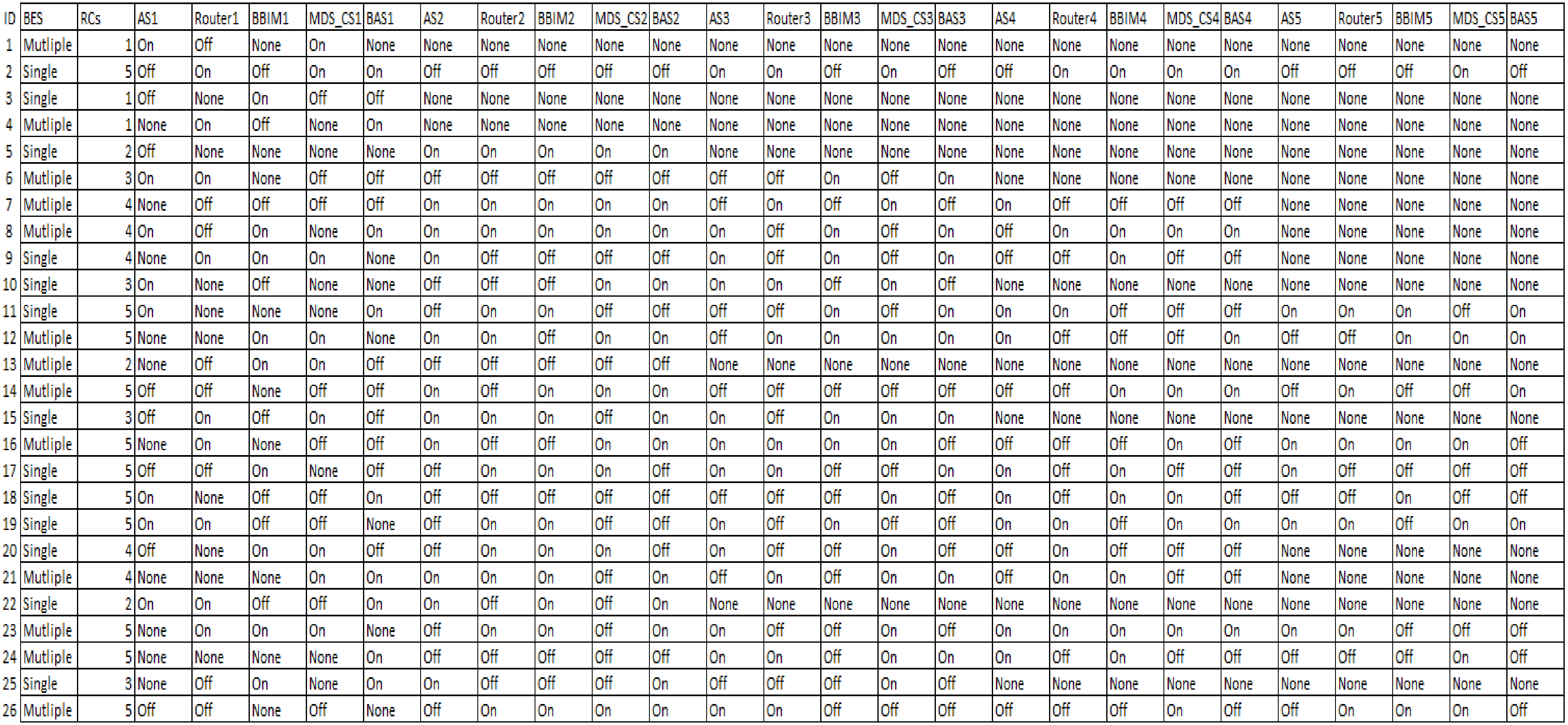
Constrained Pairwise Combinations

Pict Text Input (3)

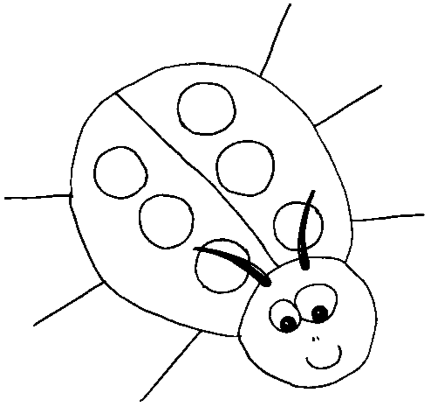
```
IF [RCs] < 5 THEN [AS5] in {"None"} ELSE [AS5] in {"On" , "Off"} ;
IF [RCs] < 5 THEN [Router5] in {"None"} ELSE [Router5] in {"On" , "Off"} ;
IF [RCs] < 5 THEN [BBIM5] in {"None"} ELSE [BBIM5] in {"On" , "Off"} ;
IF [RCs] < 5 THEN [MDS_CS5] in {"None"} ELSE [MDS_CS5] in {"On" , "Off"} ;
IF [RCs] < 5 THEN [BAS5] in {"None"} ELSE [BAS5] in {"On" , "Off"} ;
#
IF [RCs] < 4 THEN [AS4] in {"None"} ELSE [AS4] in {"On" , "Off"} ;
IF [RCs] < 4 THEN [Router4] in {"None"} ELSE [Router4] in {"On" , "Off"} ;
IF [RCs] < 4 THEN [BBIM4] in {"None"} ELSE [BBIM4] in {"On" , "Off"} ;
IF [RCs] < 4 THEN [MDS_CS4] in {"None"} ELSE [MDS_CS4] in {"On" , "Off"} ;
IF [RCs] < 4 THEN [BAS4] in {"None"} ELSE [BAS4] in {"On" , "Off"} ;
#
IF [RCs] < 3 THEN [AS3] in {"None"} ELSE [AS3] in {"On" , "Off"} ;
IF [RCs] < 3 THEN [Router3] in {"None"} ELSE [Router3] in {"On" , "Off"} ;
IF [RCs] < 3 THEN [BBIM3] in {"None"} ELSE [BBIM3] in {"On" , "Off"} ;
IF [RCs] < 3 THEN [MDS_CS3] in {"None"} ELSE [MDS_CS3] in {"On" , "Off"} ;
IF [RCs] < 3 THEN [BAS3] in {"None"} ELSE [BAS3] in {"On" , "Off"} ;
#
IF [RCs] < 2 THEN [AS2] in {"None"} ELSE [AS2] in {"On" , "Off"} ;
IF [RCs] < 2 THEN [Router2] in {"None"} ELSE [Router2] in {"On" , "Off"} ;
IF [RCs] < 2 THEN [BBIM2] in {"None"} ELSE [BBIM2] in {"On" , "Off"} ;
IF [RCs] < 2 THEN [MDS_CS2] in {"None"} ELSE [MDS_CS2] in {"On" , "Off"} ;
IF [RCs] < 2 THEN [BAS2] in {"None"} ELSE [BAS2] in {"On" , "Off"} ;
```



Constrained Pairwise Combinations

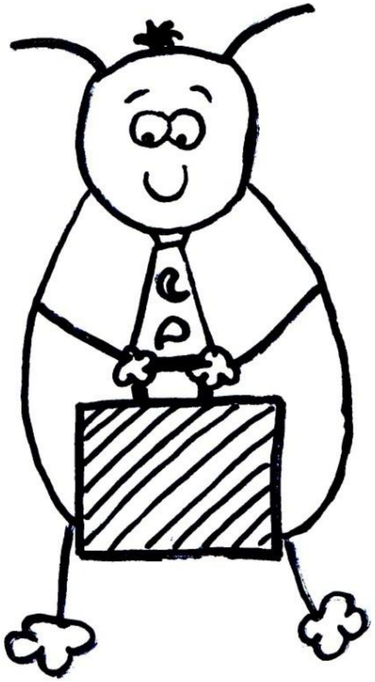


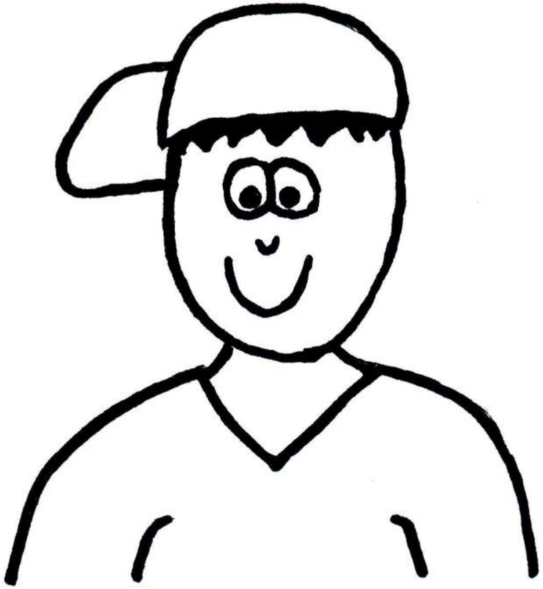
ID	BES	RCs	AS1	Router1	BBIM1	MDS_CS1	BAS1	AS2	Router2	BBIM2	MDS_CS2	BAS2	AS3	Router3	BBIM3	MDS_CS3	BAS3	AS4	Router4	BBIM4	MDS_CS4	BAS4	AS5	Router5	BBIM5	MDS_CS5	BAS5
1	Mutliple	1	On	Off	None	On	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None
2	Single	5	Off	On	Off	On	On	Off	Off	Off	Off	Off	On	On	Off	On	Off	Off	On	On	On	On	Off	Off	Off	On	Off
3	Single	1	Off	None	On	Off	Off	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None
4	Mutliple	1	None	On	Off	None	On	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None
5	Single	2	Off	None	None	None	None	On	On	On	On	On	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None
6	Mutliple	3	On	On	None	Off	Off	Off	Off	Off	Off	Off	Off	Off	On	Off	On	None	None	None	None	None	None	None	None	None	None
7	Mutliple	4	None	Off	Off	Off	Off	On	On	On	On	On	Off	On	Off	On	Off	On	Off	Off	Off	Off	None	None	None	None	None
8	Mutliple	4	On	Off	On	None	On	On	On	On	On	On	On	Off	On	Off	On	Off	On	On	On	On	None	None	None	None	None
9	Single	4	None	On	On	On	None	On	Off	Off	Off	Off	On	Off	On	Off	On	Off	Off	On	Off	Off	None	None	None	None	None
10	Single	3	On	None	Off	None	None	Off	Off	Off	On	On	On	On	Off	On	Off	None	None	None	None	None	None	None	None	None	None
11	Single	5	On	None	None	None	On	Off	On	On	Off	Off	Off	Off	On	Off	On	On	On	Off	Off	Off	On	On	On	Off	On
12	Mutliple	5	None	None	On	On	None	On	On	Off	On	On	Off	On	On	On	On	On	Off	Off	Off	On	Off	Off	On	On	On
13	Mutliple	2	None	Off	On	On	Off	Off	Off	Off	Off	Off	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None
14	Mutliple	5	Off	Off	None	Off	Off	On	Off	On	On	On	Off	Off	Off	Off	Off	Off	Off	On	On	On	Off	On	Off	Off	On
15	Single	3	Off	On	Off	On	Off	On	On	On	Off	On	On	Off	On	On	On	None	None	None	None	None	None	None	None	None	None
16	Mutliple	5	None	On	None	Off	Off	On	Off	Off	On	On	On	On	On	On	Off	Off	Off	Off	On	Off	On	On	On	On	Off
17	Single	5	Off	Off	On	None	Off	Off	On	On	On	Off	On	On	Off	Off	On	On	Off	On	Off	Off	On	Off	Off	Off	Off
18	Single	5	On	None	Off	Off	On	Off	Off	Off	Off	Off	Off	Off	Off	On	Off	On	Off	On	On	Off	Off	Off	On	Off	Off
19	Single	5	On	On	Off	Off	None	Off	On	On	Off	Off	On	Off	On	Off	Off	On	On	Off	On	On	On	On	Off	On	On
20	Single	4	Off	None	On	On	Off	Off	On	On	On	Off	On	Off	Off	On	Off	Off	On	Off	Off	Off	None	None	None	None	None
21	Mutliple	4	None	None	None	On	On	On	On	On	Off	On	Off	On	Off	On	On	Off	On	On	Off	Off	None	None	None	None	None
22	Single	2	On	On	Off	Off	On	On	Off	On	Off	On	None	None	None	None	None	None	None	None	None	None	None	None	None	None	None
23	Mutliple	5	None	On	On	On	None	Off	On	On	Off	On	On	Off	Off	On	Off	On	On	On	On	On	On	On	Off	Off	Off
24	Mutliple	5	None	None	None	None	On	Off	Off	Off	Off	Off	On	On	Off	On	On	On	Off	On	Off	Off	Off	Off	Off	On	Off
25	Single	3	None	Off	On	None	On	On	Off	Off	Off	On	Off	Off	Off	On	Off	None	None	None	None	None	None	None	None	None	None
26	Mutliple	5	Off	Off	None	Off	None	Off	On	On	On	On	On	On	Off	Off	Off	Off	Off	Off	On	Off	Off	On	On	On	Off

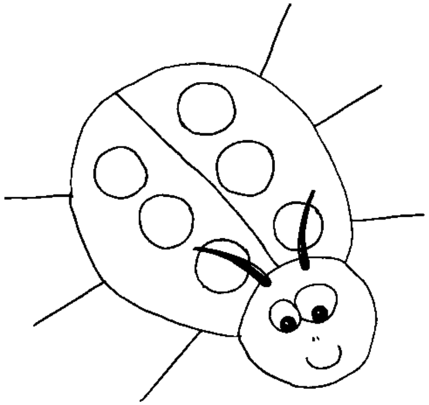


Thank You

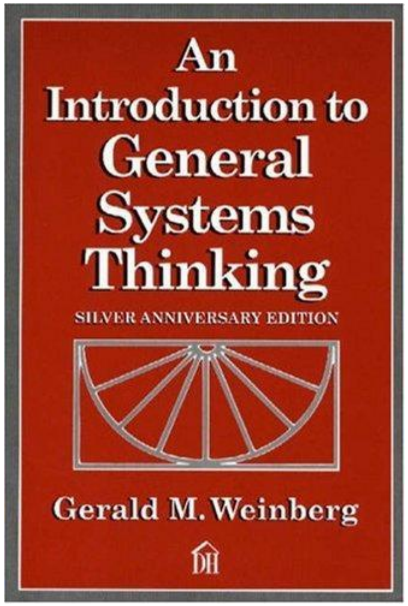
- Questions?

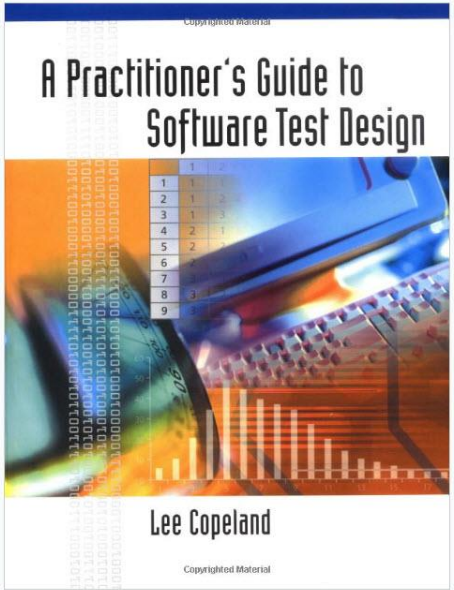


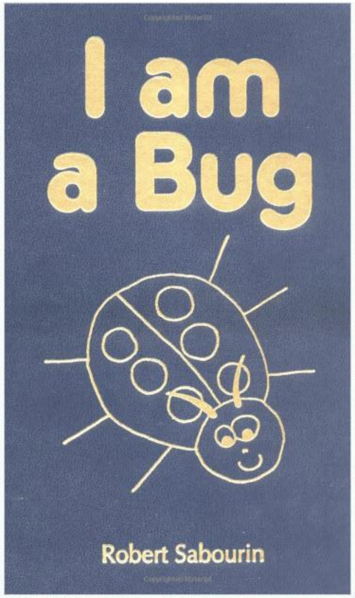


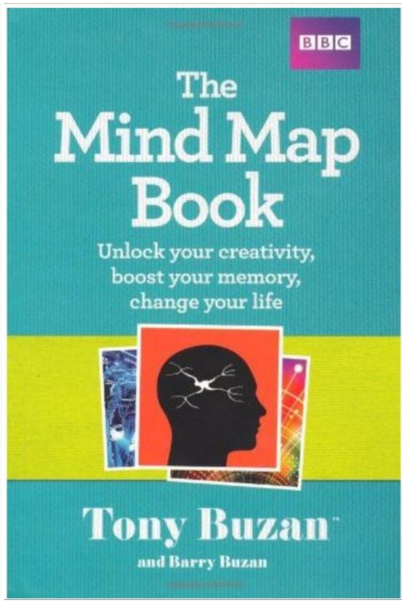


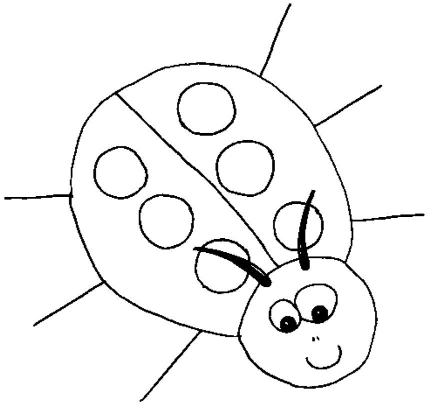
Some References


An Introduction to General Systems Thinking
SILVER ANNIVERSARY EDITION
Gerald M. Weinberg
DH

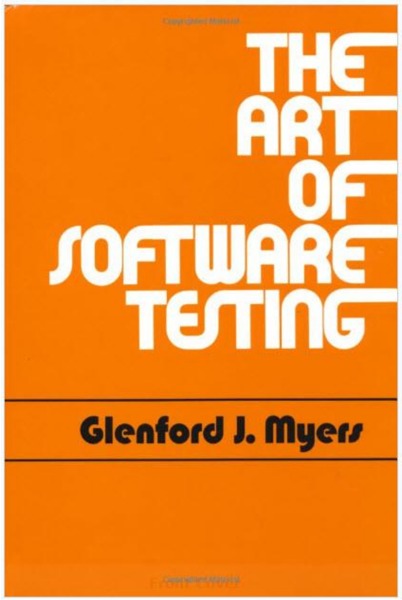


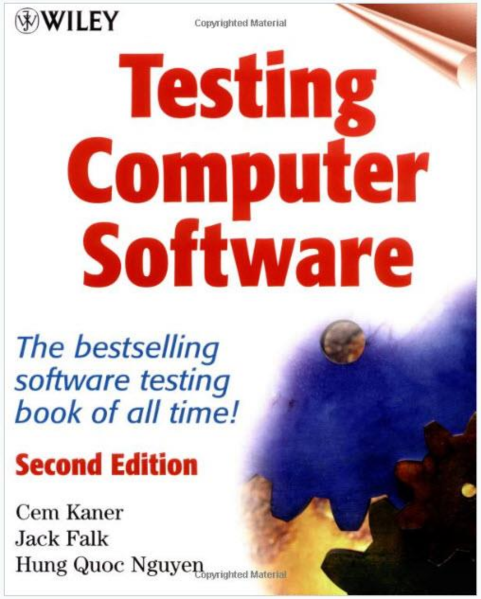


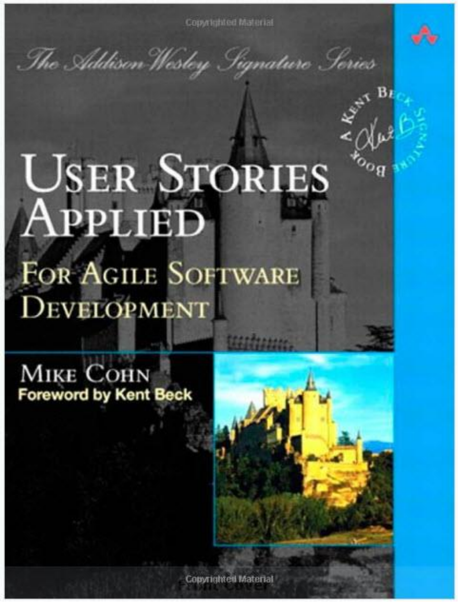


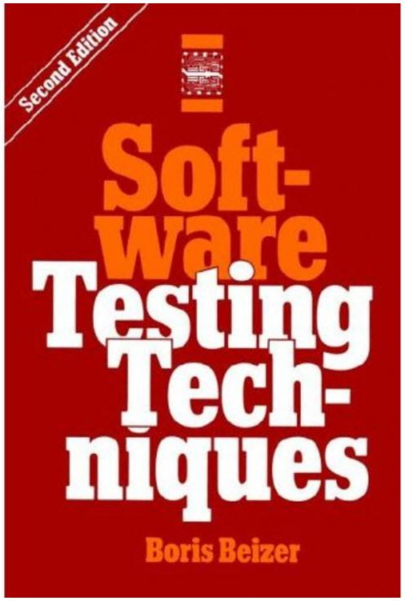


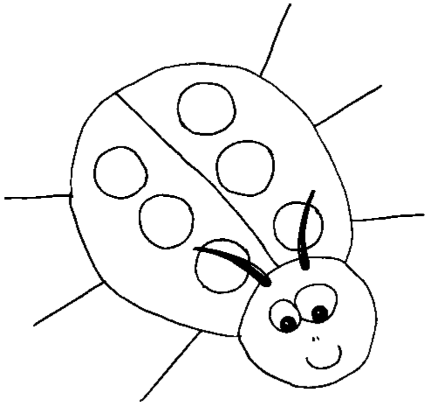
Some References



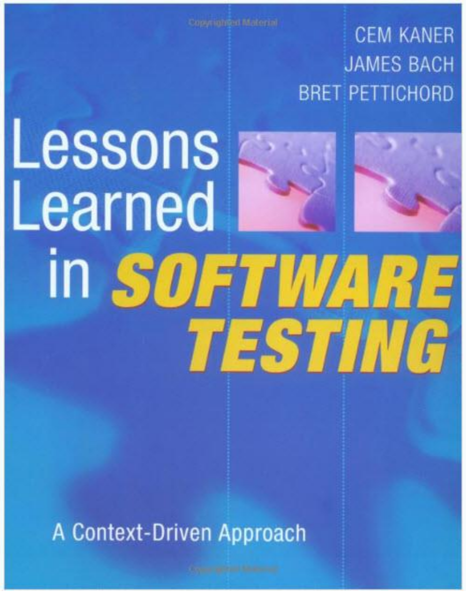

WILEY
Copyrighted Material
Testing Computer Software
The bestselling software testing book of all time!
Second Edition
Cem Kaner
Jack Falk
Hung Quoc Nguyen

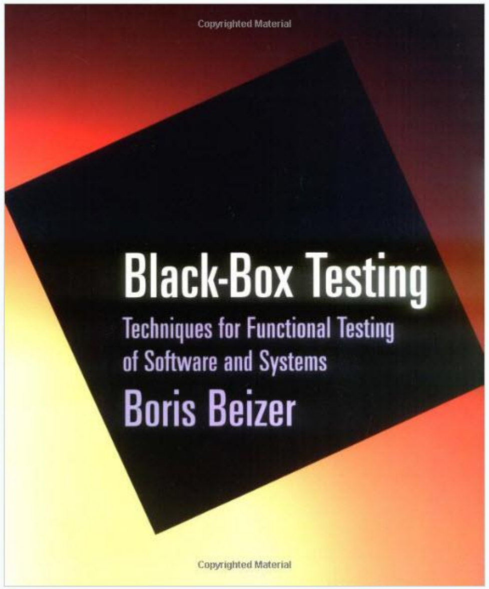

Copyrighted Material
The Addison Wesley Signature Series
A Kent Beck Signature Book
User Stories Applied
For Agile Software Development
Mike Cohn
Foreword by Kent Beck
Copyrighted Material

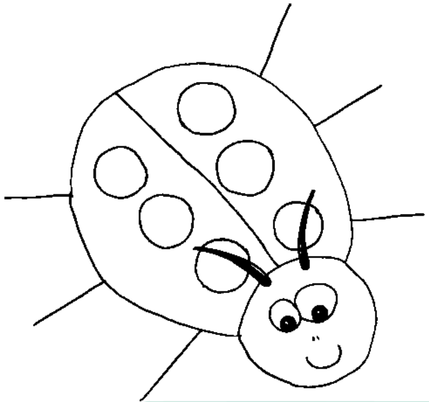




Some References

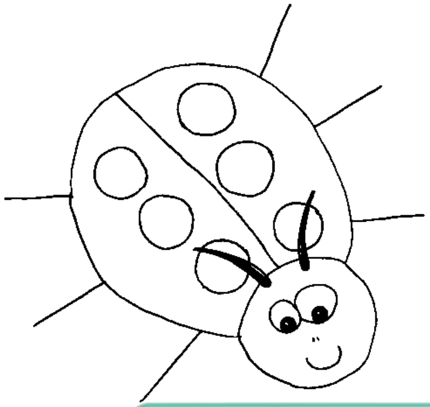

Copyrighted Material
CEM KANER
JAMES BACH
BRET PETTICHORD
Lessons Learned in SOFTWARE TESTING
A Context-Driven Approach





Some Tools

Fitnesse	http://fitnesse.org/
Cucumber	http://cukes.info/
Freemind	http://freemind.sourceforge.net/wiki/index.php/Main_Page
PICT	http://download.microsoft.com/download/f/5/5/f55484df-8494-48fa-8dbd-8c6f76cc014b/pict33.msi
AllPairs	http://www.satisfice.com/testmethod.shtml
e2gRuleWriter	http://expertise2go.com/e2g3g/e2g3gdoc/e2gRuleWriterRef.htm#SOFTWARE



Some Web References

Gmail Examples	http://www.amibugshare.com/examples/Gmail_test.zip
PICT Online	http://www.amibugshare.com/pict/
Mind Mapping	http://www.amibugshare.com/examples/Mind_Mapping.zip
Articles	http://www.amibugshare.com/articles/
Examples	http://www.amibugshare.com/examples/
Case Studies	http://www.amibugshare.com/case_studies/