

# **Scenario Similarity Map for Visualizing Classified Scenarios**

Atsushi Ohnishi

Department of Computer Science,

Ritsumeikan Univ., JAPAN

# 1. Introduction

- Scenario is useful to specify system's behaviors and user's actions.
- In the scenario-based system development, incorrect scenarios and insufficient scenarios lead to unsuccessful development.

# Terminology

Scenarios specifies sequence of events and can be classified into three categories.

## 1) normal scenario

- Sequence of events that represents normal and typical behavior of the system

## 2) alternative scenario

Normal scenario : payment with credit card

Alternative scenario : payment with cash

## 3) exceptional scenario

Normal scenario : successful payment with credit card

Exceptional scenario : payment with illegal card

Q1: What is the problem you are working on?

Problem: It is difficult to find reusable scenarios.

## Q2: How and why is visualization expected to help?

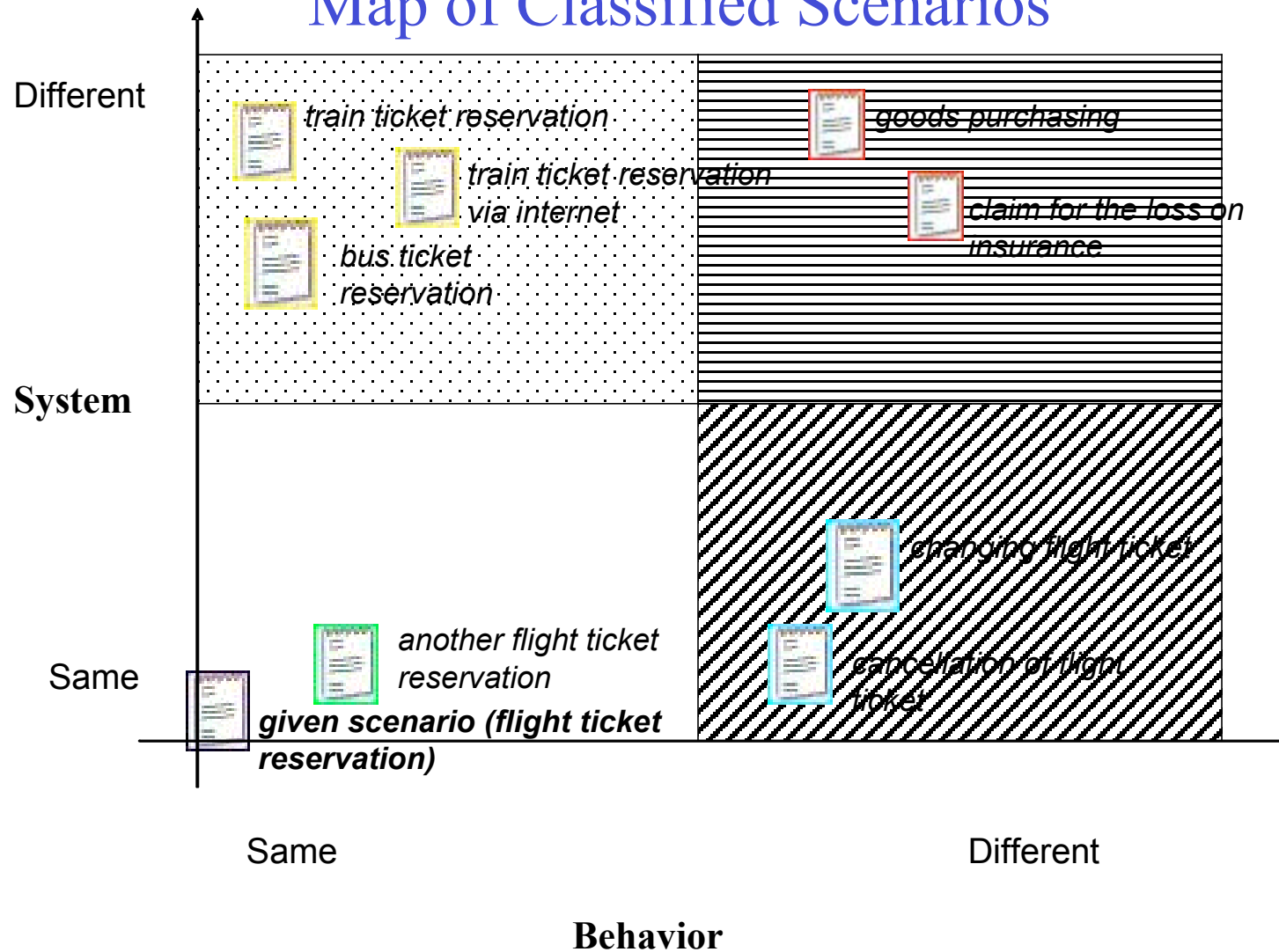
- Solution: We automatically classify scenarios and visualize classified scenarios. Users can find scenarios that behave similarly and scenarios of similar problem domain.

Q3: Who and what is the visualizing for? How are they expected to use it?

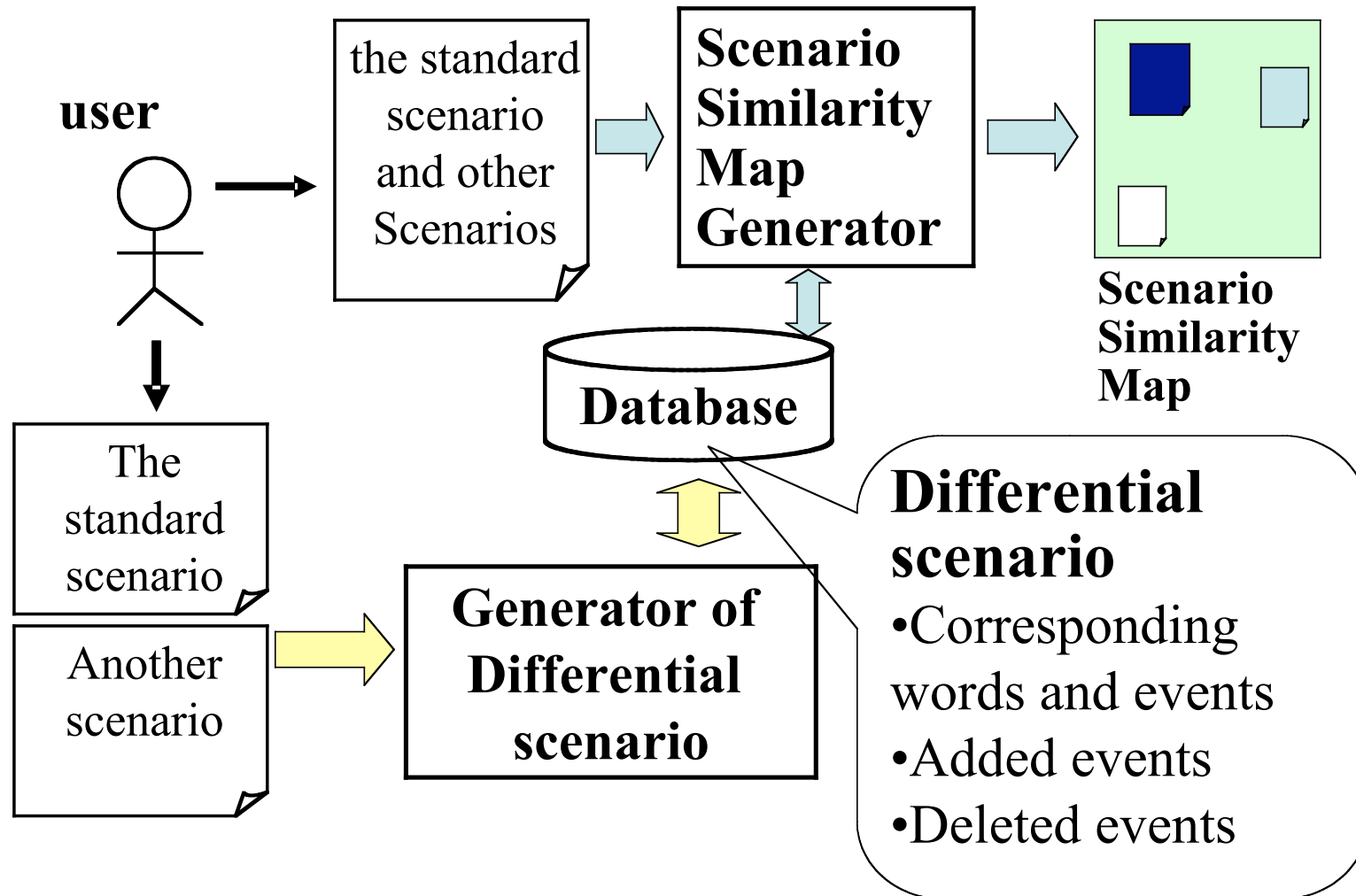
- Users and system developers can use the visualized classification of scenarios in order to get scenarios of similar behaviors and scenarios of similar problem domain.

# Q4: What visualization have you created?

## Map of Classified Scenarios



# Q5: How was it derived and constructed?





## Q6: How does it work?

- User specifies a standard scenario. Then differential information (differential scenario) between the standard scenario and each of scenarios can be derived.
- The differential scenario gives the basis of classification

# Show and tell

## 2. Scenario and Scenario language

Scenario: Sequence of events

- Scenario description language has been developed in order to
  - 1) specify scenario correctly
  - 2) transform a scenario from a certain viewpoint into a scenario from a different viewpoint
  - 3) integrate scenarios from different viewpoints
  - 4) verify a scenario with rules.

# Type of event sequence

Scenario: Sequence of events

Type of sequence:

- 1) sequential (*event1, event2, ...*)
- 2) selective (**if .. then event1 else event2**)
- 3) iterative (**do event1, event2, ... until ..**)
- 4) fork-join (**AND-fork, OR-fork,  
XOR-fork, ... join**)

# Scenario Example

*[title: A customer purchases a train ticket of reservation seat]*

*[viewpoints: staff, customer]*

*[pre-condition: the customer has not a ticket & has not reserved a seat]*

*[post-condition: the customer will get a ticket & reserved a seat]*

1. A staff asks a customer about his request.

2. He sends the customer's request to reservation center with a terminal.

...

Sequence of events

A condition that satisfies at the end of the scenario is specified.

A condition that satisfies at the start of the scenario is specified.

<b>Action (verb)</b>	<b>cases</b>
Transform/derive	agent,source,goal,object
Collate/compare	agent,source,object
Report on/ sense/monitor	agent,source,operation,instrument
Query/ask	agent,object,goal
Command/control	agent,recipient,operation, instrument
Inform/notify	agent,source,operation,instrument
Move/transfer/input	agent,source,goal,instrument

## Event description

- Mr. X moves from Tokyo to Delhi by airplane.

action	agent	source	goal	instrument
move	Mr. X	Tokyo	Delhi	airplane

## 3. Differential scenario

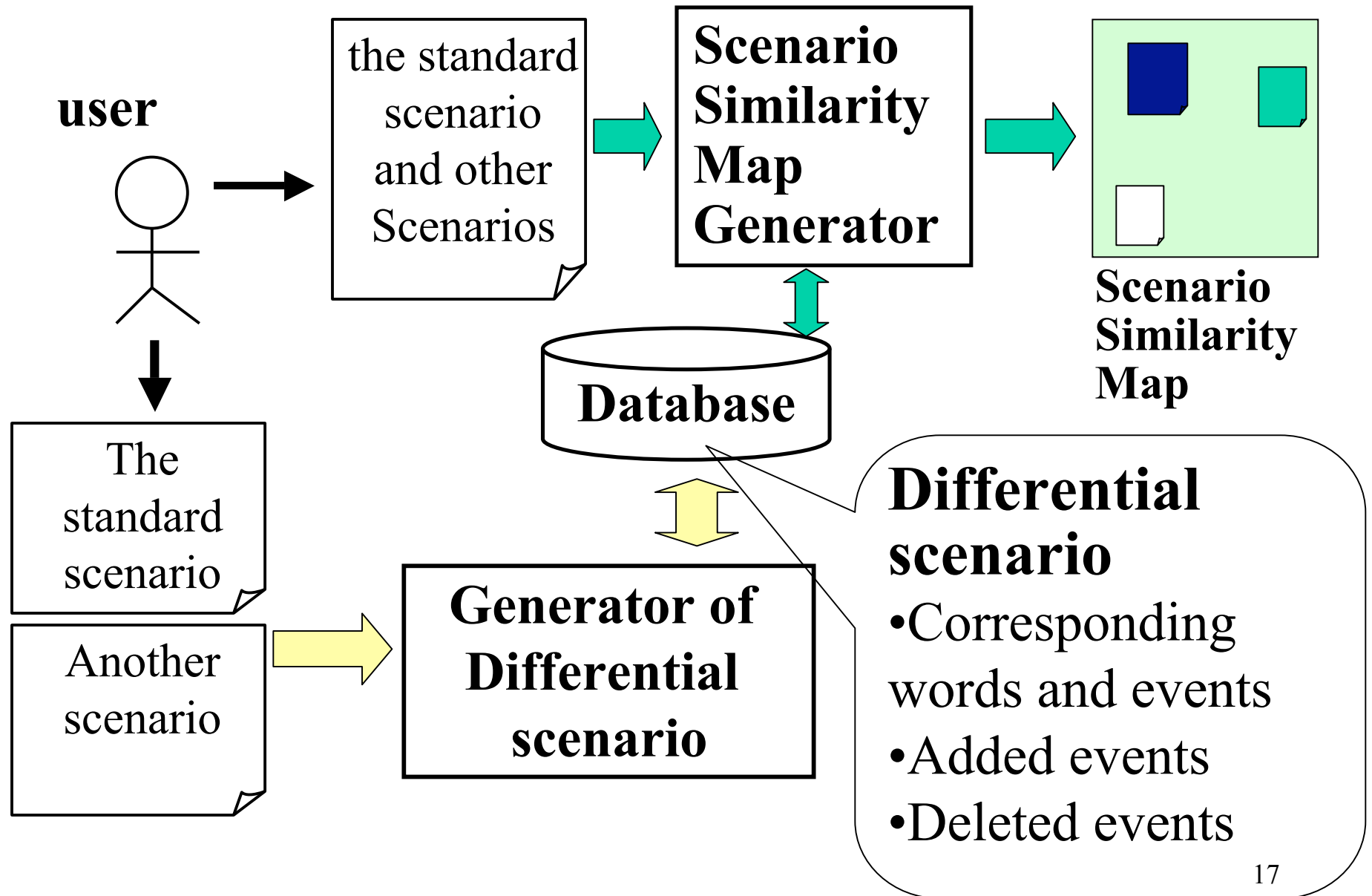
Systems of the same domain behave similarly.

Differential scenario consists of

- 1) A list of corresponding words
- 2) Deleted events
- 3) Added events



# Outline



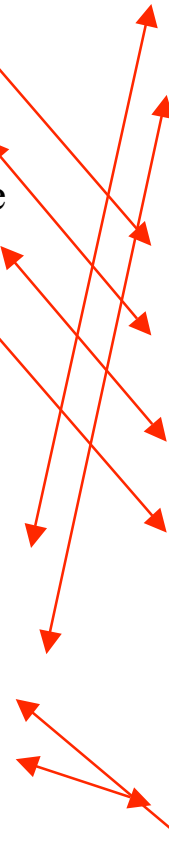
## Corresponding words and events between two scenarios

*[Title: A customer reserves a book at an e-library system]*

1. A **customer** enters a **title of book** to **e-library system**.
2. The e-library system retrieves **books** from book-database using the title.
3. The customer gets the result of retrieval from the system.
4. He **selects** a book for reservation.
5. The system inquires detailed information to the database using ISBN code.
6. The customer gets the detailed information from the system.
7. He enters his user-id and password to the system.
8. **If** (the id and password are correct) **then** he is authenticated by the system.
9. The system **reserves** the book.
10. The system modifies the database.
11. The customer gets a message of successful reservation from the system.

*[Title:: A user reserves CD in a rental CD system]*

1. A **user** enters user-id and password to **CD reservation system**.
2. **If** (user-id and password are correct) **then** the user is authenticated by the system.
3. He enters keywords of CD to the system.
4. The system retrieves **CDs** from CD-database with the **keywords of CD**.
5. The user gets candidates from the system.
6. The user **selects** the candidates.
7. He gives credit card information to the system.
8. The system certifies the credit card information.
9. The system modifies the CD-database
10. The system **reserves** the CDs.



# Differential factors

- *The ratio of different actors (DA) =  
The number of different actors / the number of  
actors in the standard scenario*
- *The ratio of different data (DD) =  
The number of different data / the number of  
data in the standard scenario*
- *The ratio of different events (DE) =  
The number of added or deleted events / the  
number of events in the standard scenario*

# Differential factors and classification

DA	DD	DE	system	problem domain	behavior
0	0	0	same	same	same
low	-	low	probably same	same	almost same
high	-	low	probably different	same	almost same
low	low	high	probably same	same	different
low	high	high	probably different	same	different
high	low	high	probably different	same	different
high	high	high	different	different	different



Same system,  
different behavior

Same system,  
similar behavior

Different domain

Different system,  
same behavior

Different

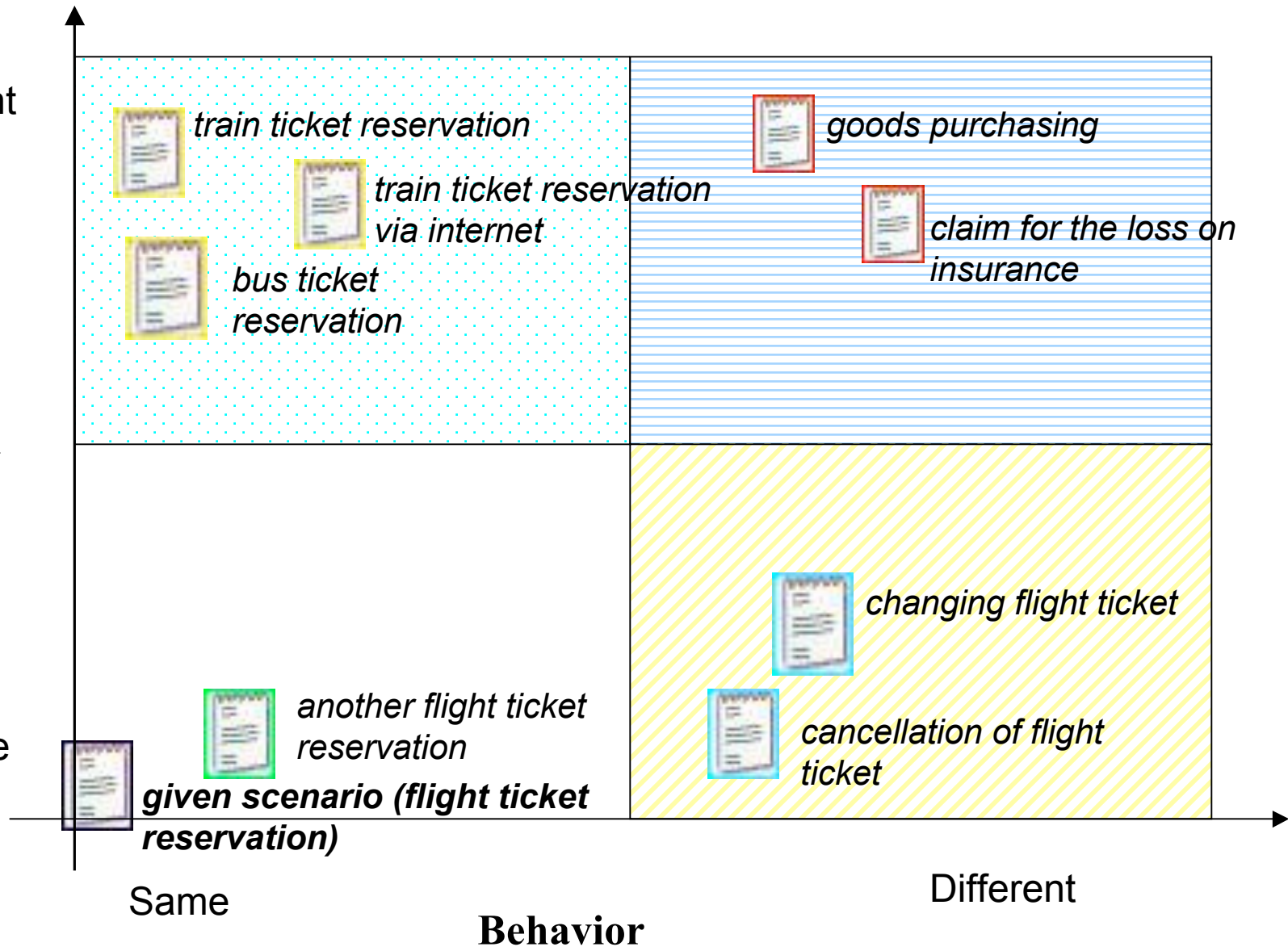
System

Same

Same

Behavior

Different



**Scenario Similarity Map**

# Scenario Similarity Map for Visualizing Classified Scenarios

By Keitaro Osawa, and Atsushi Ohnishi

Department of Computer Science, Ritsumeikan University, JAPAN

E-mail: ohnishi@cs.ritsumeai.ac.jp

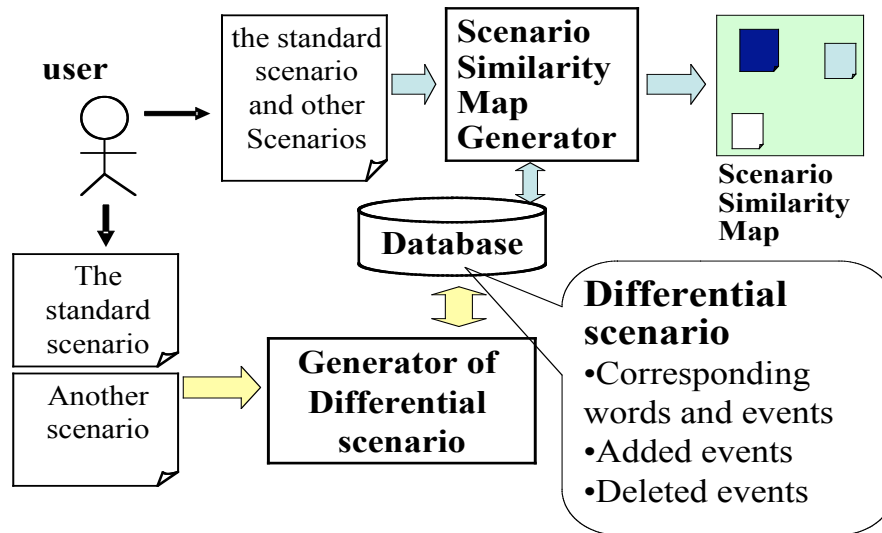


Fig. 1: Outline of the Proposed Method

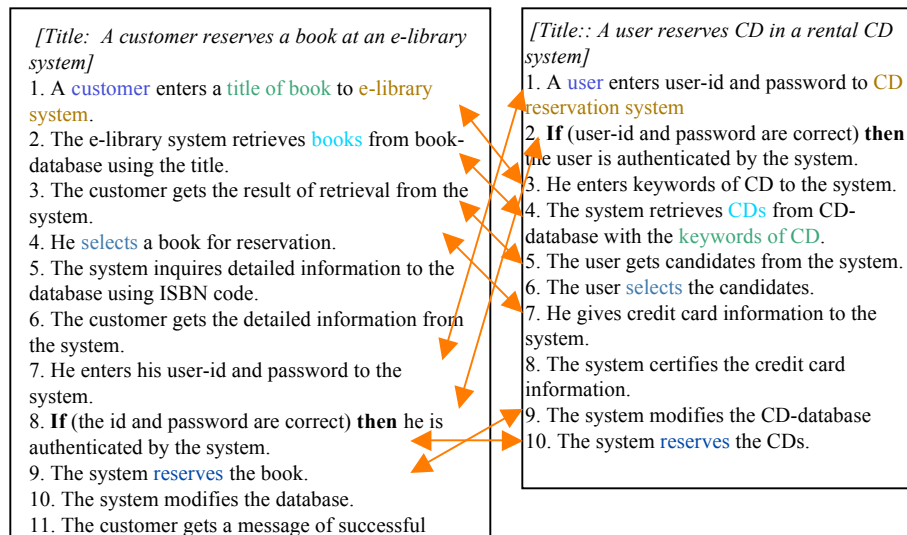


Fig. 2: Corresponding words and events between two scenarios

The ratio of different actors (DA) =  
The number of different actors / the number of actors in scenario A  
The ratio of different data (DD) =  
The number of different data / the number of data in A  
The ratio of different events (DE) =  
The number of added or deleted events / the number of events in A

DE:high → different behavior, DA or DD: high → different system

Fig. 3: Three factors of Differential Scenario A and B

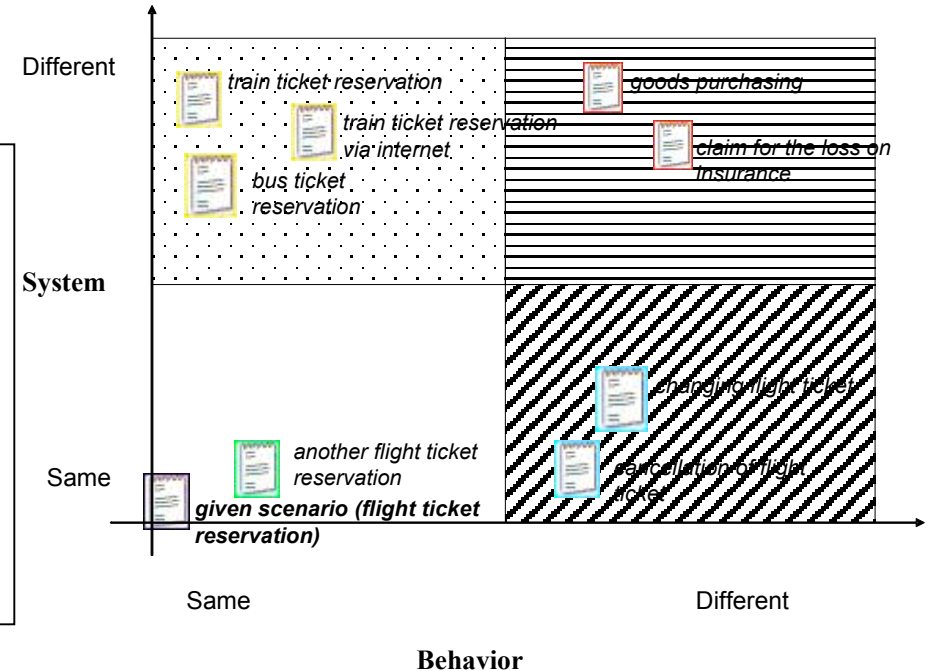


Fig. 4: Scenario Similarity Map (SS Map)

## Q8: Has your visualization been used in practice?

- Not yet.
- We will have an experiment to validate our work. Two groups of subjects. One group classify scenarios by hand. The other one classify with our method.

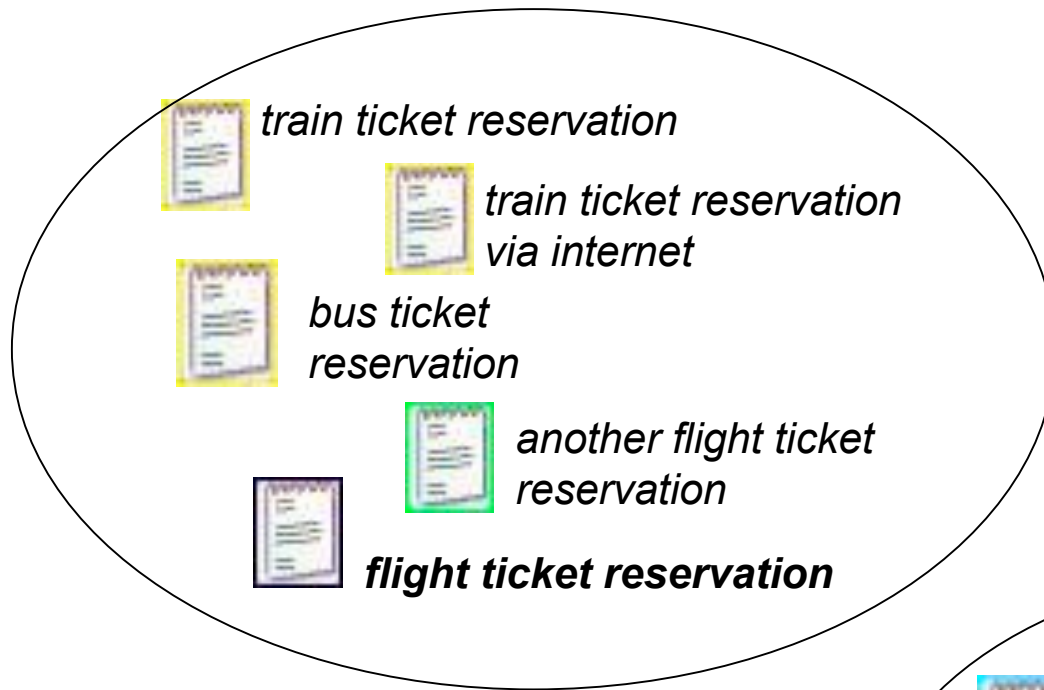


## Q9: Critique your visualization.

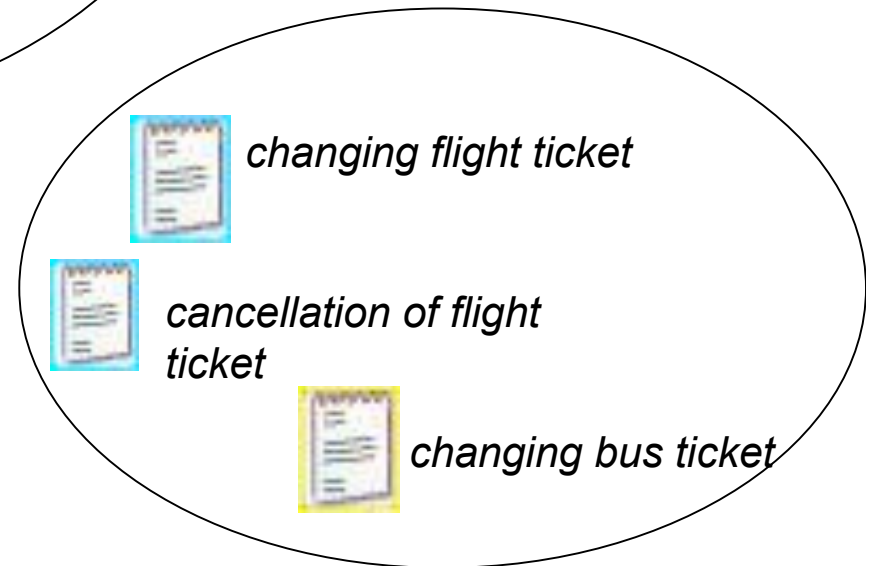
- Strength: Easily find similar scenarios.
- Weakness: Differential information is derived between standard scenario and each of other scenarios. No classification among the other scenarios.

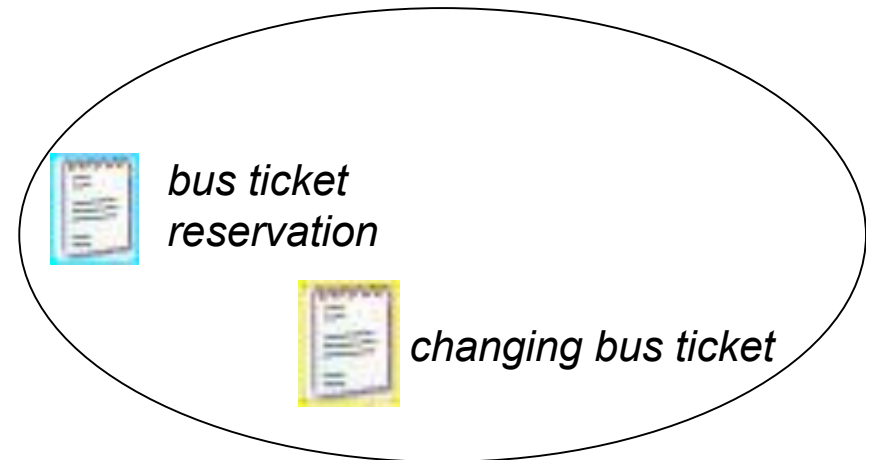
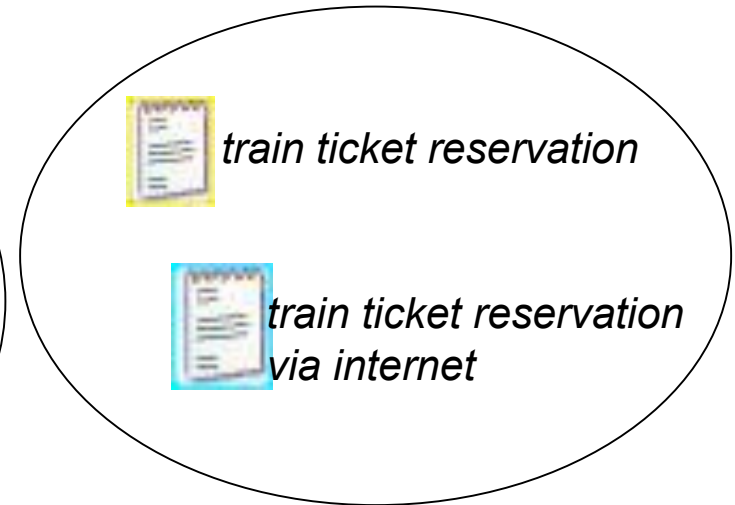
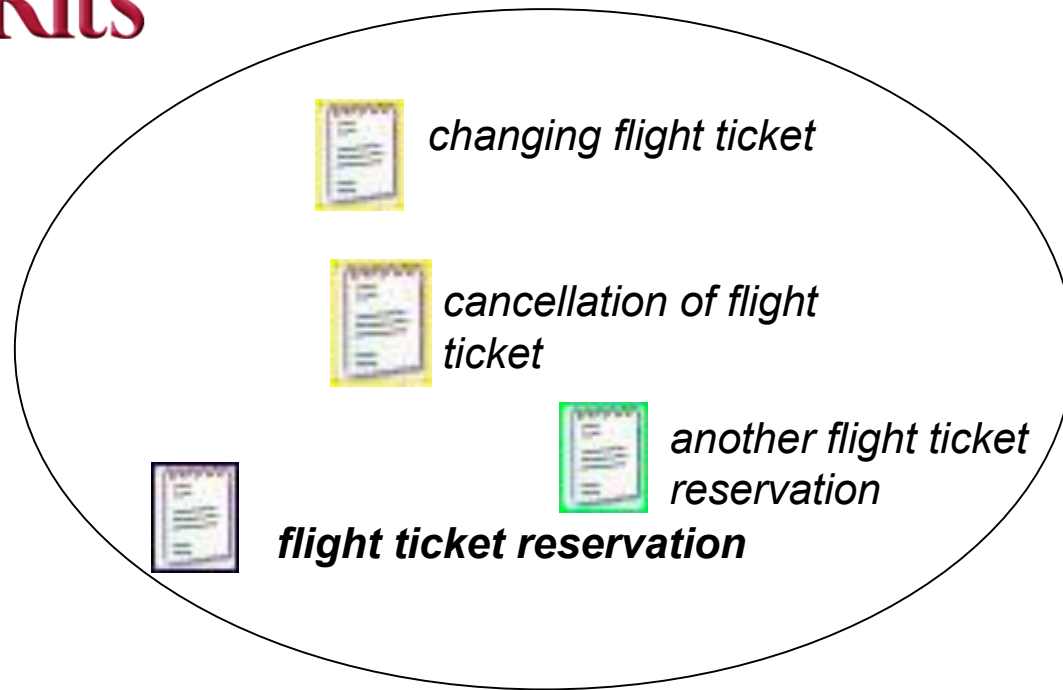
## Q10: Next steps?

- Not specifying a standard scenario, but specify “behavior or system” then scenarios will be classified with the specified keyword.

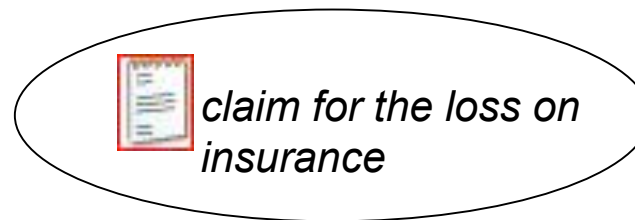
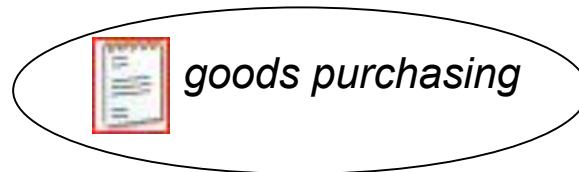
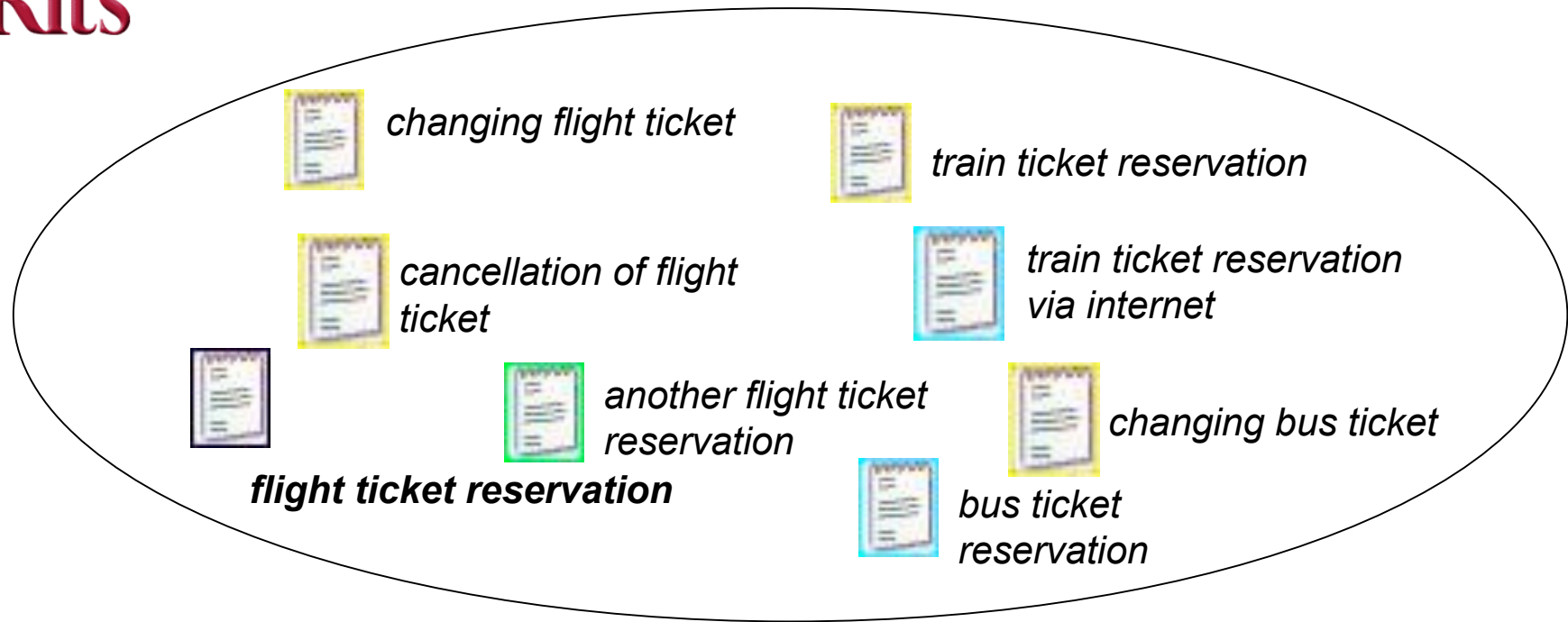


## Scenarios of similar behaviors





## Scenarios of same systems



## Scenarios of similar systems

# Thank you!

- Any other questions or comments?