Chat Room Demo Assignment

GRUIA-CATALIN ROMAN

12 AUGUST 2018

Part I : Tuple Space

Problem Statement

- Design, implement, and test a package (Java class TSpace) that supports the storage and retrieval of a set (i.e., no duplicates!) of k-tuples where
  - k is an integer from 1 to N
  - N is a small constant integer specified in the constructor
  - a k-tuple is a sequence of k fields
  - a field contains a simple non-null object ω of some class σ
- The operations supported by TSpace are
  - out(tuple) — adds a tuple to the tuple space (if not already there)
  - inp(pattern) — removes a tuple matching the pattern and returns it to the caller
  - rdp(pattern) — returns a tuple matching the pattern

Problem Statement

- A pattern consists of a sequence of objects, some of which may be null
- A tuple τ matches a pattern π if
  - they have the same number of fields
  - each field is of the same type
  - all non-null fields in the pattern equal those in the tuple
- Simple illustration using abstract notation
  - assume an empty tuple space
  - out(<Mary, age, 23>) adds tuple <Mary, age, 23> to the tuple space
  - rdp (<Mary, age, *>) returns the tuple <Mary, age, 23> to the caller
  - rdp (<Mary, *, 23>) returns the tuple <Mary, age, 23> to the caller
  - inp (<*, *, 23>) returns the tuple <Mary, age, 23> to the caller leaving the tuple space empty

Special Considerations

- If no tuple is found in the tuple space, **inp** and **rdp** return null
- If more than one tuple matches the pattern, one tuple is selected in a nondeterministic manner
  - repeated invocations of the **rdp** are not required to return different tuples

Part II : Chat Room

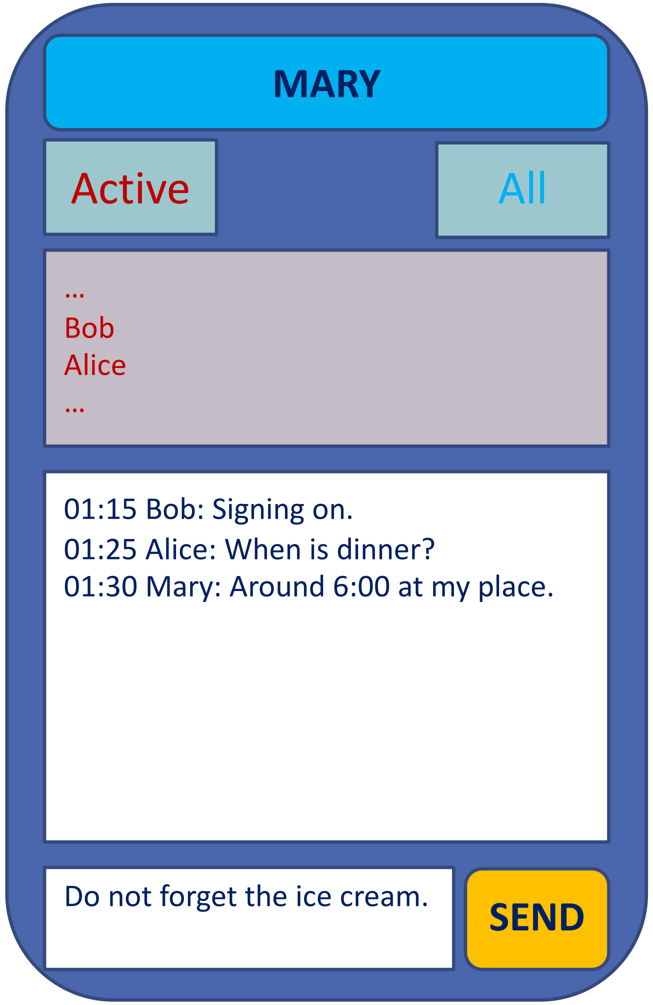
Problem Statement

- Use the TSpace package to build a chat room where
  - each chat room user has a name
  - a user has the ability to see
    - the list of people using the chat room now (active)
    - the list of all the users (all)
  - the last 10 messages received from any users on the selected list are shown in chronological order
  - a user can generate a new message, which will be available to everybody in the chat room

Deign Notes

- The goal of the project is to design the TSpaces class
- The Chat Room is meant to be a simple application to demonstrate that your design work and that you understood the semantics of each operation on the tuple space
- Keep the design simple
  - a top level coordinator that creates users and gives each a turn to act
  - users carry out zero or one action per turn and return control to the coordinator
  - users interact with each other only via the tuple space which is shared by all
  - user interfaces can be either textual or visual and do not need to be very sophisticated

Visual and Textual Interfaces



01:15 Bob: Signing on.
01:25 Alice: When is dinner?
01:30 Mary: Around 6:00 at my place.

Do not forget the ice cream.

```
System: NEW USER Bob
Bob: ON LINE
Mary: SEND Do not forget the ice cream.
Bob: PRINT ACTIVE USERS
  Bob
  Mary
Mary: OFF LINE
Alice: PRINT ALL USERS
  Alice
  Bob
  Mary
Alice: PRINT ACTIVE USERS
  Alice
  Bob
Bob: MESSAGES FROM ALL USERS
  …
```