Reverse Engineering Finite State Machines from Rich Internet Applications

Domenico Amalfitano

Anna Rita Fasolino

Porfirio Tramontana

Dipartimento di Informatica e Sistemistica

University of Naples Federico II, Italy

Motivation

- In the last years, Rich Internet Applications (RIAs) have emerged as a new generation of web applications offering greater usability and interactivity than traditional ones.
  - Web pages have been transformed in complex GUIs with synchronous and asynchronous interactions with the user and with the resources
- Key problems:
  - to define models that are suitable for RIAs representation
  - to define Reverse Engineering techniques for the abstraction of the proposed model

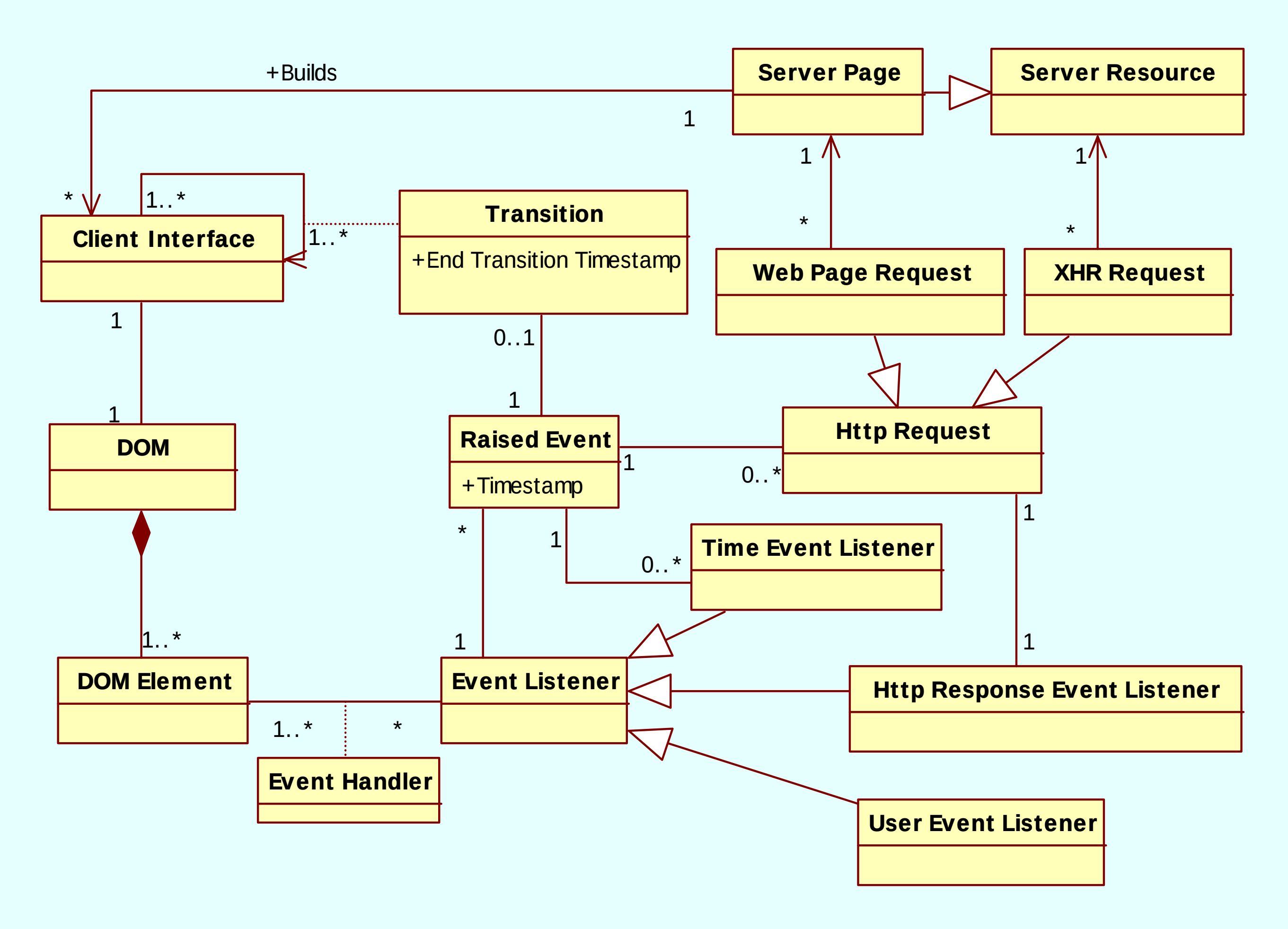
RIA pages vs plain HTML pages

- Complex interactions between the user and the RIA in the browser environment are designed and implemented in Javascript
    - A simple model comprehending only pages and forms (e.g. the Conallen model) is not more suitable to describe RIAs
- Javascript event handlers can be associated to any DOM element and are able to modify the DOM instance itself
- XHR objects are also able to manage Http communications

RIA Interactions Conceptual model

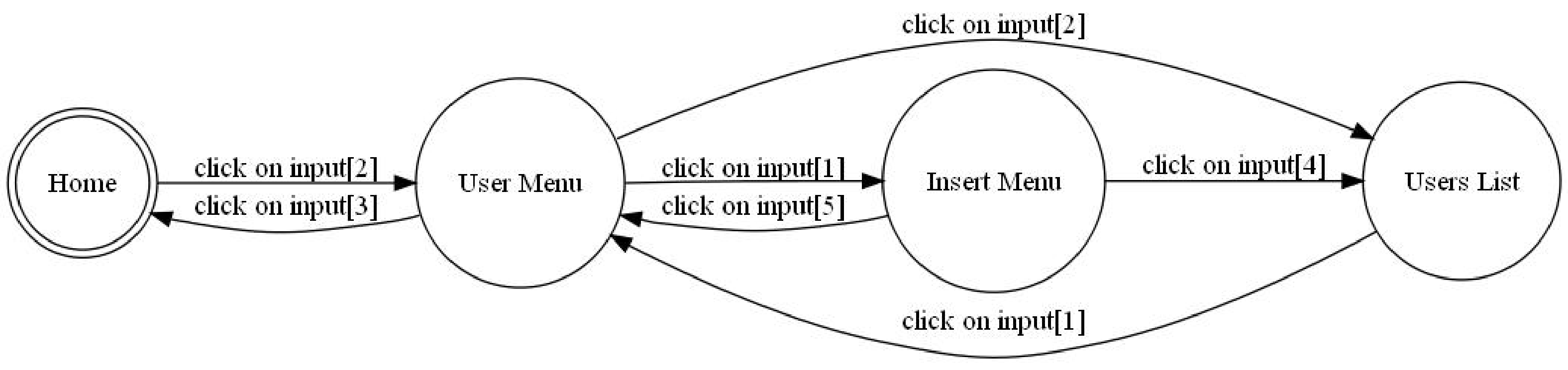
- A Client Interface correspond to the DOM instanced at a given time in the browser

- The raise of an event to which an event handler is associated can cause a transition to another client interface (i.e. a variation of the DOM)



Transition Graph

- RIA interactions can be modelled as a Transition Graph, where
  - each node is associated with a distinct Client Interface
  - each edge corresponds to the raise of an event
    - The event handler execution causes the variation between Client Interfaces.



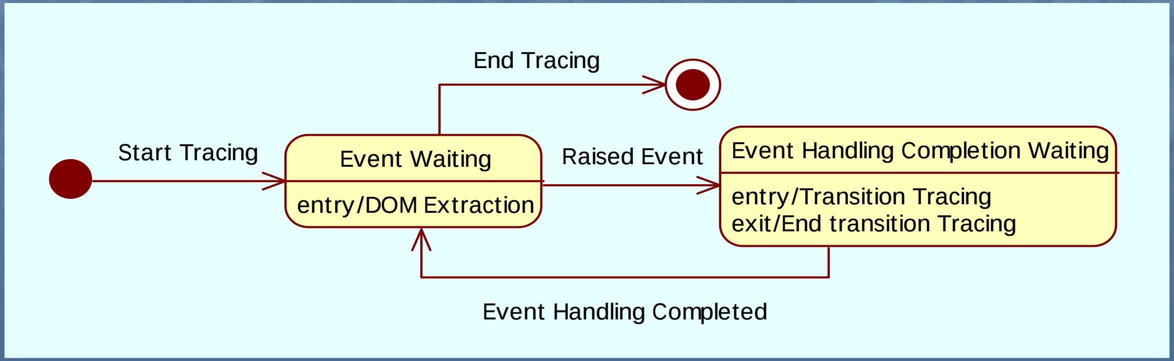
Reverse Engineering Process 1/2

- In order to reconstruct the model of a RIA, a three steps process has been proposed

- The Extraction step is obtained by dynamic analysis

- A user navigates the RIA while it is monitored.
  - Probes applied externally to the DOM structure allow the tracing of the complete cycle of event raising/handling with a non-invasive approach

  - For each observed Client Interface the complete set of tags, attributes and event handlers is stored



Reverse Engineering Process 2/2

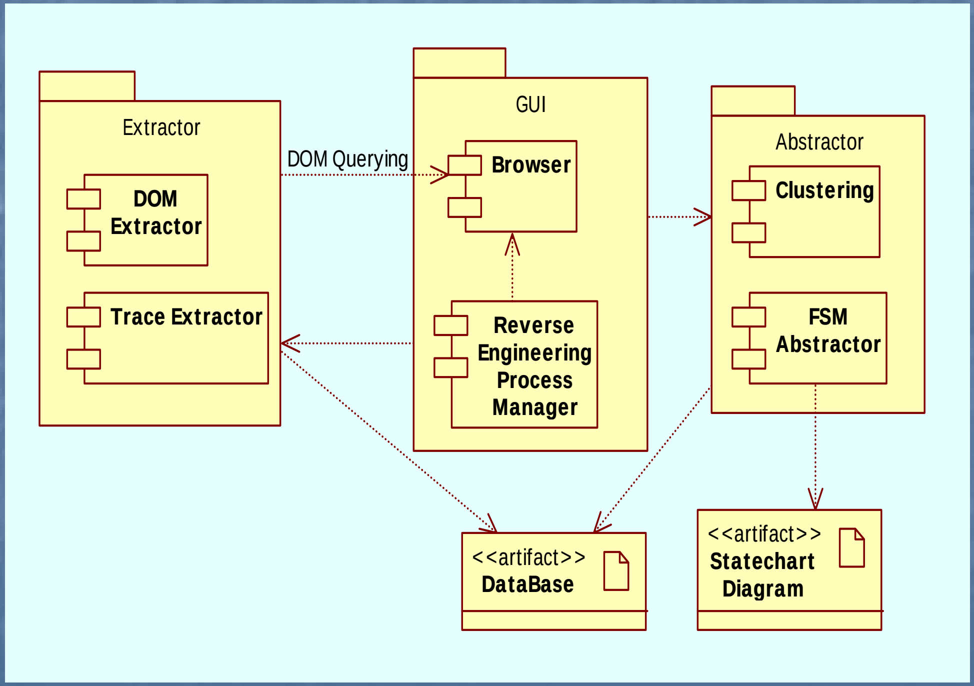
- In this step the retrieved information are analyzed in order to abstract the FSM Model of the RIA
- Two open issues:
  - When a Client Interface should be considered equivalent to a previously visited one?
  - What DOM elements and properties should be considered in order to compare two Client Interfaces?

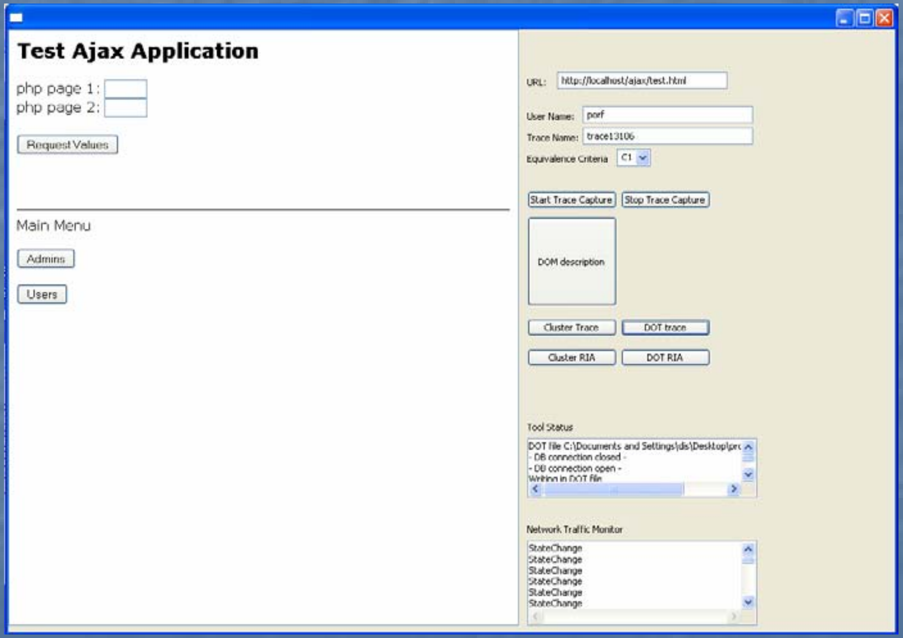
- Two possible criteria to cluster together a pair of Client Interfaces have been proposed:
  - Criterion C1 considers equivalent two Client Interfaces if the corresponding DOM structures include the same set of DOM elements with the same registered event listeners and handlers
  - Criterion C2 considers equivalent two Client Interfaces if the criterion C1 is satisfied and the set of Http requests and time event listeners are equivalent, too.

- A Concept Assignment step, assisted by an expert, must be carried in order to assess if the applied equivalence criterion has been able to cluster together Client Interfaces corresponding to actual Interaction States

RE-RIA Tool

- The Tool supports the proposed Reverse Engineering Process
- It is realised in Java with JavaXPCOM library in order to embed a Mozilla instance into the tool





Test Ajax Application

php page 1:
php page 2:

URL: http://localhost/ajax/test.html

User Name: porf

Trace Name: trace13106

Equivalence Criteria C1

Tool Status

```
DOT file C:\Documents and Settings\dis\Desktop\prc
- DB connection closed -
- DB connection open -
Writing in DOT file
```

Network Traffic Monitor

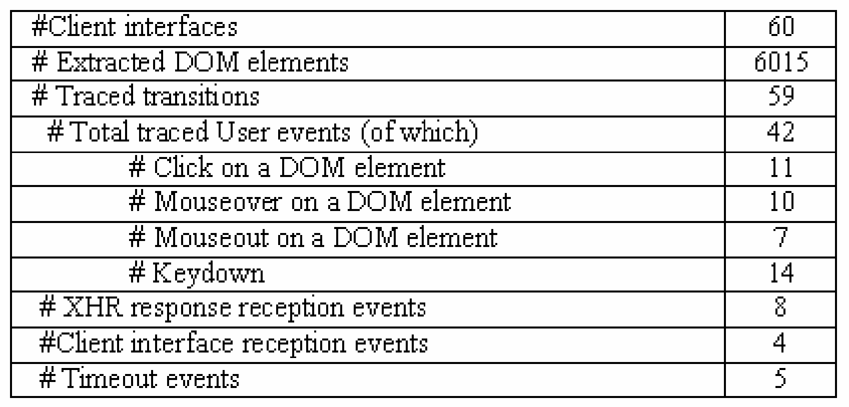
```
StateChange
StateChange
StateChange
StateChange
StateChange
StateChange
```

An example (1/2)

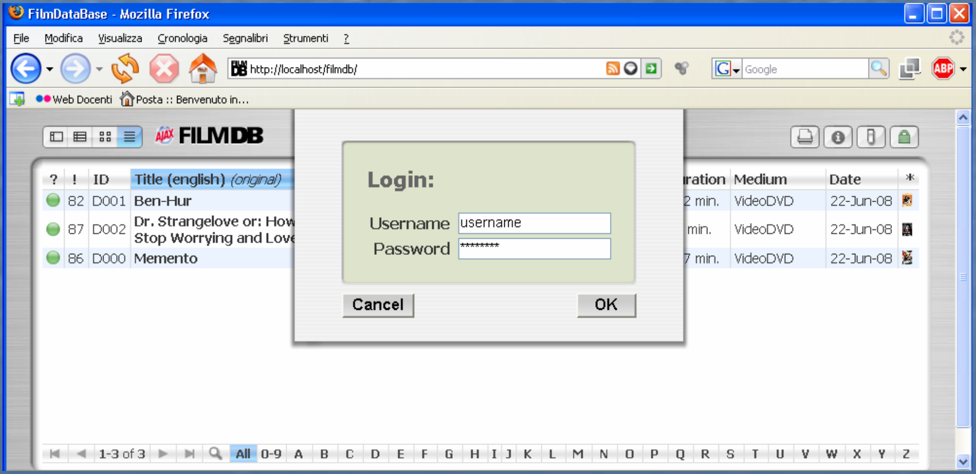
- Based on *FilmDB* Ajax application for data management of a personal movie archive.

- Five Users Sessions corresponding to the entering/exiting the personal movie area use case scenarios were exercised by a user and traced and collected by the tool

- A Transition Graph with 60 nodes and 59 edges was preliminary obtained



#Client interfaces	60
# Extracted DOM elements	6015
# Traced transitions	59
# Total traced User events (of which)	42
# Click on a DOM element	11
# Mouseover on a DOM element	10
# Mouseout on a DOM element	7
# Keydown	14
# XHR response reception events	8
#Client interface reception events	4
# Timeout events	5



?	!	ID	Title (english) *(original)*	ration	Medium	Date
●	82	D001	Ben-Hur	2 min.	VideoDVD	22-Jun-08
●	87	D002	Dr. Strangelove or: How Stop Worrying and Love	min.	VideoDVD	22-Jun-08
●	86	D000	Memento	7 min.	VideoDVD	22-Jun-08

Login:

Username: username

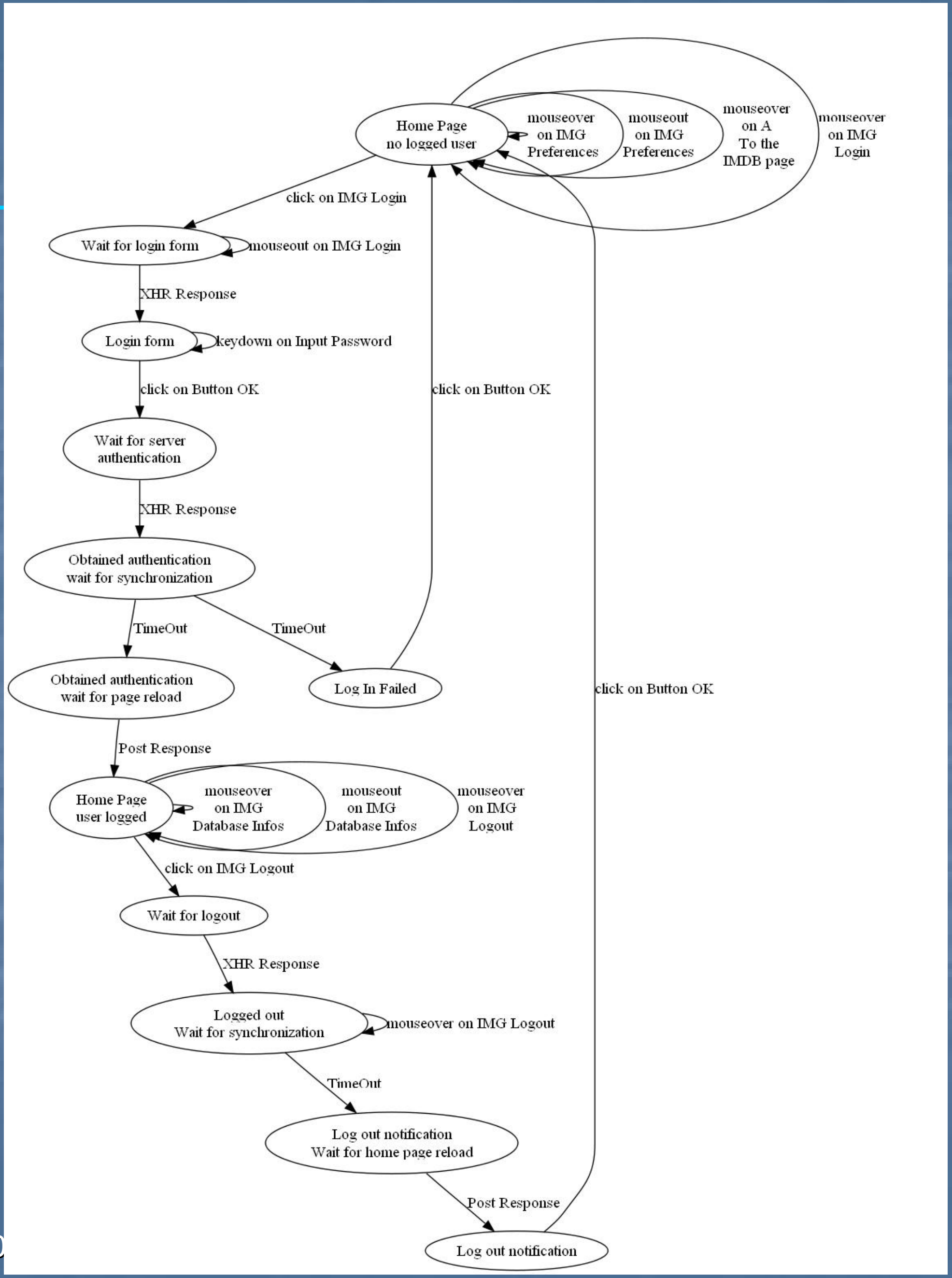
Password: *********

Example (2/2)

- The C1 criterion was able to reduce the TG to 8 nodes and 22 transitions

- A Concept Assignment activity revealed that some nodes could not be associated with meaningful states, but had to be split into further logical states.

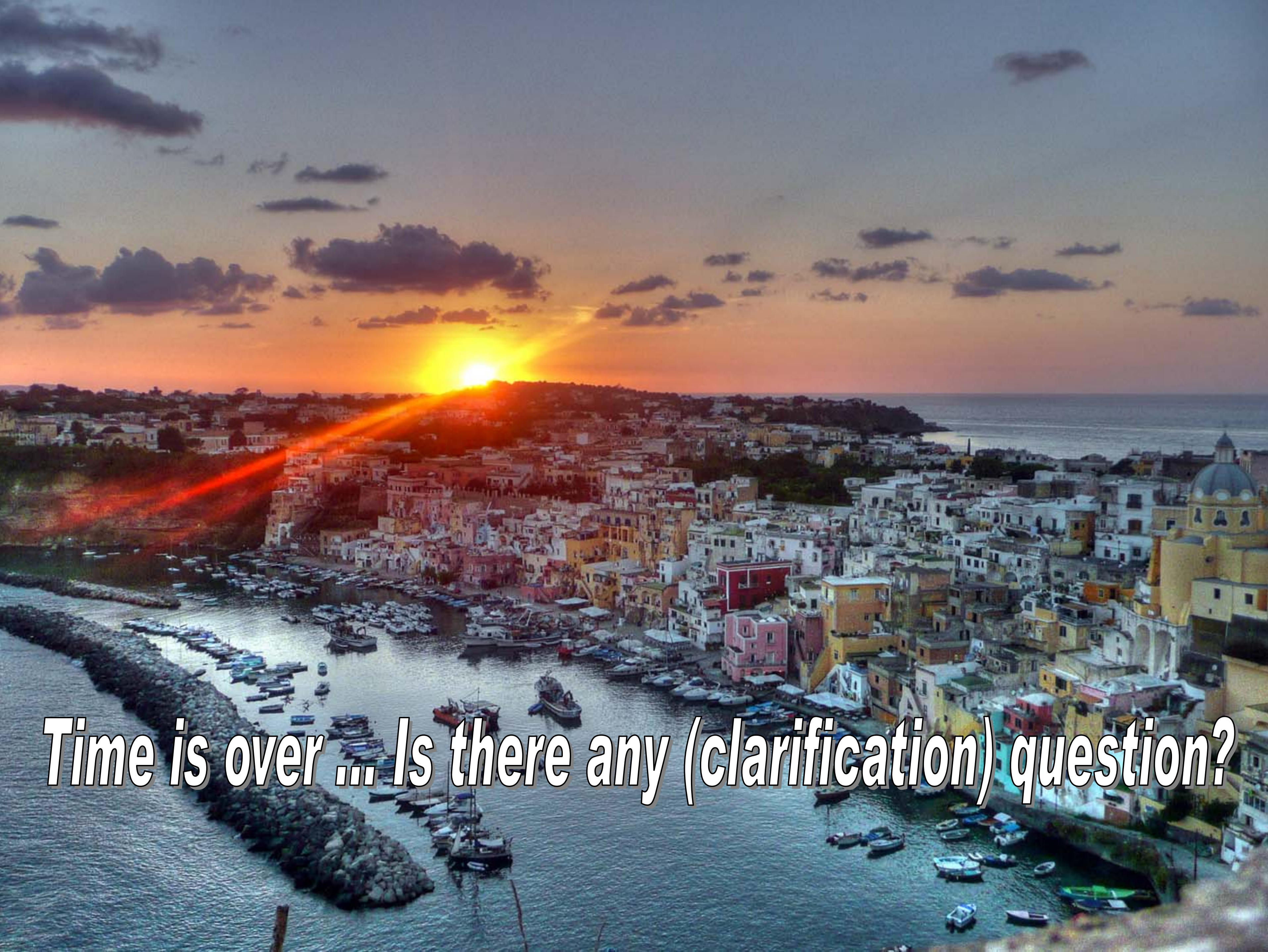
- The C2 criterion produced a TG with 12 nodes and 23 transitions that was validated by the Concept Assignment activity



Conclusions and Future Works

- This paper presented the initial results of a research project that aims at defining and validating techniques for reverse engineering software representation models from Rich Internet Applications.

- The proposing clustering criteria seems to be useful for the abstraction of a compact behaviour model of the RIA interfaces
  - Further criteria that are able to reduce the Concept Assignment step effort will be proposed in future
- The suitability of the obtained model to specific tasks, such as GUI testing, quality evaluation, migration to services will be explored in future works



Time is over ... Is there any (clarification) question?