Towards an Adaptive Audience Response System Using Role-Concepts

Tommy Kubica
Chair of Computer Networks, Institute of Systems Architecture, Faculty of Computer Science, Technische Universität Dresden

RoSI-Workshop
Lichtenwalde // Thursday, November 29, 2018
Motivation
Mobile Phones in the Classroom: If you can’t beat them, join them

– Scornavacca et al., 2009
Audience Response Systems
Audience Response Systems

Motivation

Surveys

Learning Questions

Questions from the Audience

Towards an Adaptive Audience Response System Using Role-Concepts
Chair of Computer Networks // Tommy Kubica
RoSI Workshop // November 29, 2018

Slide 4
Audience Response Systems Classification by Ebner et al.

Front Channel Functions

Back Channel Functions

Slide 5

Sources: [2,3]
Audience Response Systems Classification by Ebner et al.

Front Channel Functions
- variety of existing systems
  hard to select an appropriate system

Back Channel Functions

“Choosing the appropriate Audience Response System in different Use-Cases”
- International Conference on Education, Training and Informatics: ICETI 2019

Sources: [2,3]
Choosing the appropriate Audience Response System in different Use-Cases

- **Goal of the paper:** Support the lecturer/admin while selecting an appropriate ARS
- **Personal goal:** Get an overview about existing systems
- **3 “contributions”:**
  - Index Card for choosing an appropriate ARS
  - Divided into proprietary and didactic view
  - Investigation and Classification of 50 systems
  - Web-based tool to support the selection process
Choosing the appropriate Audience Response System in different Use-Cases

- Goal of the paper: Support the lecturer/admin while selecting an appropriate ARS
- Personal goal: Get an overview about existing systems
- 3 “contributions”:
  - Index Card for choosing an appropriate ARS
  - Divided into proprietary and didactic view
  - Investigation and Classification of 50 systems
  - Web-based tool to support the selection process

Findings

1) Different combinations / variants of functions
2) Limited to support specific didactic concepts and stages

→ Adaptive Audience Response Systems
Adaptive Audience Response Systems
Adaptive Audience Response Systems

Motivation

- The system is able to support **different didactic concepts:**
  1. that *allow tool support*
  2. that have a *separate communication channel*

- The system adapt its behavior and view to **different stages**

**Examples:**
Adaptive Audience Response Systems
Peer Instruction – Recap

Brief introduction lecture (10-15 min) -> ConcepTest students vote

- Correct answers < 30 % -> Another version of introduction lecture
- Correct answers 30 – 70 % -> Peer Discussion
- Correct answers > 70 % -> Short discussion of misconceptions

Repeat the voting -> Next topic

Presentation -> ConcepTest -> (Discussion)

Source: [5]
Adaptive Audience Response Systems
Peer Instruction – Current Implementations

**Presentation:**
- few systems provide plugins for presentation software or web-based solutions
- functions are (automatically) unlocked

**ConcepTest:**
- most systems support *Learning Questions*
- few systems concentrate on PI (*do not show correct answer, repetition, ...*)

**(Peer Discussion):**
- few systems support *Question and Answer* functionality
- functionality is provided globally
Adaptive Audience Response Systems
Peer Instruction – Current Implementations

Presentation:
- Few systems provide plugins for presentation software or web-based solutions
- Functions are (automatically) unlocked

ConceptTest:
- Most systems support Multiple Choice Questions
- Few systems concentrate on PI (do not show correct answer, repetition, …)

(Peer Discussion):
- Few systems support Question and Answer functionality
- Functionality is provided globally

Findings

1) Systems support different stages
2) No system adapts its system behavior and view to current stage

→ Introduce Role-Concepts in Audience Response Systems
Adaptive Audience Response Systems
Peer Instruction – Using Role-Concepts

Separation of concerns
- consider stages as well as didactic concepts more modular → better re-use

Describe context-adaptive systems more easy
- adapt system behavior and view to different stages and “roles”

System evolution
- possibility to add new or change existing didactic concepts in the future
Adaptive Audience Response Systems Using Role-Concepts
Adaptive ARS Using Role-Concepts
Separation of Concerns

- Recap:
  - main container Lecture
  - different user roles for different functions
- Lecture as compartment
- Basic functionality encapsulated into roles

→ How to integrate didactic concepts?
Adaptive ARS Using Role-Concepts

Separation of Concerns

- Define didactic concepts (e.g. Peer Instruction) as compartment, too:
  - Inner compartment
  - Separate compartment

+ access inner roles

+ better re-use
Adaptive ARS Using Role-Concepts
Separation of Concerns

- Define didactic concepts (e.g., Peer Instruction) as compartment, too:
  - Inner compartment
  - Separate compartment

Challenges

1) Investigate both variants
2) Investigate combination of different didactic concepts

→ Show Feasibility

+ access inner roles
+ better re-use
Adaptive ARS Using Role-Concepts

Describe context-adaptive Systems

• Peer Instruction consists out of different stages:
  • **Presentation**
  • **ConcepTest**: affects the progression
  • **(Peer Discussion)**

• 4-6 iterations within a 90 minutes lecture
  • Each iteration consist out of at least 2 changes
  • Roles can help to adapt the *system behavior and view* depending on the current stage

• Roles can also help within these stages:
  • Peer Discussion: match with random person → *Answerer* and *Peer*
  • ConcepTest: *FirstTest, RepetitionTest*, ...

• Roles can be applied for other concepts, too
Adaptive ARS Using Role-Concepts
Describe context-adaptive Systems

- Peer Instruction consists out of different stages:
  - Presentation
  - ConcepTest: affects the progression
  - (Peer Discussion)
- Roles can help to adapt the system behavior and view depending on the current stage
  - 4-6 iterations at 2-3 changes within a 90 minutes lecture
- Roles can also help within these stages:
  - Peer Discussion: Match with random person → Answerer and Peer
  - ConcepTest: FirstTest, RepetitionTest
- Roles can be applied for other concepts, too

**Challenges**

1) Roles for (stages of) didactic concepts
2) Adapt system behavior **and** views according to stage

→ Show Feasibility
Adaptive ARS Using Role-Concepts

System Evolution

- New didactic concepts are introduced
- or existing didactic concepts changes
  - e.g. condition for progression of ConcepTest

→ There is a need to add / change didactic concepts during the systems lifetime

- Roles enable the possibility to update running applications
Adaptive ARS Using Role-Concepts

System Evolution

- New didactic concepts are introduced
- or existing didactic concepts changes
  - e.g. condition for progression of ConcepTest

→ There is a need to add / change didactic concepts during the systems lifetime

Challenges

1) Use Roles to add a new didactic concept
2) Use Roles to change an existing didactic concept

→ Show Feasibility
Done so far
Done so far

- Minimal role runtime written in Ruby
  - supported functions: `newPlayer()`, `newCompartment()`, `bind()`, `unbind()`, `invokeRole(function_name, attributes)`, ...
  - based on LyRT: “A Dynamic Instance Binding Mechanism Supporting Run-Time Variability of Role-Based Software Systems”

<table>
<thead>
<tr>
<th>Id</th>
<th>Com.Id</th>
<th>CoreId</th>
<th>PlayerId</th>
<th>RoleId</th>
<th>Type</th>
<th>Lvl</th>
<th>Seq</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>company</td>
<td>bob</td>
<td>bob</td>
<td>developer</td>
<td>PPR</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>company</td>
<td>bob</td>
<td>bob</td>
<td>accountant</td>
<td>PPR</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>company</td>
<td>ely</td>
<td>ely</td>
<td>freelance</td>
<td>PPR</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>company</td>
<td>ely</td>
<td>freelance</td>
<td>taxpayer</td>
<td>RPR</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>tax</td>
<td>company</td>
<td>company</td>
<td>taxpayer</td>
<td>PPR</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Source: [6]
Done so far

- Adjusted library to work with AMCS
- Implemented a minimal working example to show feasibility
  - move `createQuestion` to a role → single students are allow to create questions
- First results:

```
<table>
<thead>
<tr>
<th>Id</th>
<th>Com.Id</th>
<th>CoreId</th>
<th>PlayerId</th>
<th>RoleId</th>
<th>Type</th>
<th>Lvl</th>
<th>Seq</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>company</td>
<td>bob</td>
<td>bob</td>
<td>developer</td>
<td>PPR</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>company</td>
<td>bob</td>
<td>bob</td>
<td>accountant</td>
<td>PPR</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>company</td>
<td>ely</td>
<td>ely</td>
<td>freelance</td>
<td>PPR</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>company</td>
<td>ely</td>
<td>freelance</td>
<td>taxpayer</td>
<td>RPR</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>tax</td>
<td>company</td>
<td>company</td>
<td>taxpayer</td>
<td>PPR</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
```
Done so far

- Adjusted library to work with AMCS
- Add basic example to show feasibility
  - move createQuestion to a role → single students are allowed to create questions
- First results:
  1) Global instance pool and lookup table will be insufficient
  2) Apply concepts of SCROLL → hold instances per compartment
  2) Lookup-Table per compartment

<table>
<thead>
<tr>
<th>Id</th>
<th>Com.Id</th>
<th>Coreld</th>
<th>PlayerId</th>
<th>RoleId</th>
<th>Type</th>
<th>LId</th>
<th>Seq</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>company</td>
<td>bob</td>
<td>bob</td>
<td>developer</td>
<td>PPR</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>company</td>
<td>bob</td>
<td>bob</td>
<td>accountant</td>
<td>PPR</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>company</td>
<td>ely</td>
<td>ely</td>
<td>freelance</td>
<td>PPR</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>company</td>
<td>ely</td>
<td>company</td>
<td>taxpayer</td>
<td>RPR</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>tax</td>
<td>ely</td>
<td>company</td>
<td>taxpayer</td>
<td>PPR</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Source: [6]
Next Steps and Conclusions
Next Steps

**Milestone 1**
Begin within RoSI
1 June

**Milestone 2**
First Paper
23 August

**Milestone 3**
Internal Chair Meeting
5/6 November

**Milestone 4**
RoSI Workshop / First TAB
29/30 November

**Milestone 5**
Paper Submission
no date yet

---

2018
June | July | Aug | Sept | Oct | Nov | Dec | Jan | Feb | Mar | Apr | May

2019

- **Related Work**
  May/June – 3 August

- **RoSI Courses / Orientation**
  2 July – 28 September

- **Integration in Ruby**
  17 September – 7 December

- **Adjust AMCS**
  8 October – x

- **Re-Use Use-Case**
  19 November – 21 December

- **Adaptive Use-Case**
  7 January – 22 February

- **Create Paper**
  1 February - 8 March

---

"Choosing the appropriate Audience Response System in different Use-Cases"
Conclusions

**Audience Response Systems:**
- solve issues in classical teaching scenarios
  - rely on specific didactic concepts and stages

**Adaptive Audience Response Systems:**
- support different didactic concepts
- adapt system behavior and view to current stages

**Adaptive Audience Response Systems Using Role-Concepts:**
- separation of concerns
- describe context-adaptive systems more easy
- system evolution
Learning is not the Product of Teaching. Learning is the Product of the Activity of Learners.

– John Holt
References

[1] Example created by Dr. Felix Kapp


[3] Saarbrücker Informatik-Professor sorgt per Software für Lernerfolg während der Vorlesung: https://idw-online.de/de/news631780 – last successful access on Nov, 1st 2018

[4] “Choosing the appropriate Audience Response System in different Use-Cases”: Own paper for ICETI 2019
