Integration of Third-party Components into TAO

The ItemBuilder Example
Overview

• History and Background Information
  Heiko Rölke (roelke@dipf.de), DIPF

• Technical Information
  Michel Dorochevsky (Michel.Dorochevsky@softcon.de), SOFTCON AG

• ItemBuilder Usage and Examples
  Gabriele Gissler (gissler@dipf.de), DIPF
History

• Concrete Idea: 2006
  – PISA 2009 – ERA
  – Item Development in Luxembourg and Germany

• Prototype in Spring 2007
  – HyperText Builder

• All ERA items (re)developed
  – Browser
  – Mail Client
History -2-

PIAAC (since 2008)
• Runtime re-factoring (+ new name)
• Emphasis on complex interaction modes
• Equating/Mode effects

Complex Problem Solving
• MicroDYN
• Finite State Machine controlled items
Main Ideas

• Empower Item Developpers
  – Authoring System
  – WYSIWYG
  – Templates

• Separate Design from Execution
  – Stand-alone Designer
  – Generated Executable
CBA Item Builder Components

CBA Item Builder: standalone authoring tool with preview
CBA Item: result of authoring is deployed on delivery platform
CBA Runtime: executes the item

Many different delivery platforms available:
- **CBA Server** => e.g. PIAAC Translation and Test Portal
- **Web Delivery Environment** => Integration in other Test Platforms (e.g. VMWare Integration with TAO)
- **CBA Execution Environment** => Web, Intranet, Soft Appliance, Memory Stick
CBA Item Builder

Standalone Visual Editor for building CBA Items

- Rich Text Fields (static/interactive)
- InputFields
- Buttons
- Links
- CheckBox
- Images
- ImageMaps
- Video/Audio MediaPlayer
- ...

![Diagram of CBA Item Builder interface](image)
Examples of CBA Items

Typical Examples

- **Simulation of an “Email / Web Scenario”**
  The test person receives an email and should book cinema tickets online.

- **Explore and understand a dynamic model (MicroDYN)**
  The test person should explore and master a dynamic system with input (exogenous) variables influencing output (endogenous) variables.

- **Understand an automaton based on a Finite State Machine (MicroFIN)**
  The test person should interact with a mobile phone and set time to summertime.
Implementation and Architecture

1. Massive re-use of existing Open Source Components/Frameworks for Development and Runtime
   - Item Development: CBA Item Builder
     Java, Eclipse RCP, Eclipse GMF/EMF, openArchitectureWare, in a second step also Xtext
   - Item Delivery: CBA Runtime
     GUI Framework: initially Adobe Flash (asWing), now Eclipse RAP (Rich AJAX Platform)
     Platform: JBoss Application Server, OSGi
     Framework: Spring, EJB
     Apache SCXML (for FSM engine)

2. Architecture
   - model-driven
   - component oriented
Integration in TAO

Example PIAAC 2011

- CBA Item integrated as a web component (stimulus) into the TAO test frame.
- TAO controls the test execution and the lifecycle of the CBA Item at runtime
- TAO communication based on Flash/JavaScript
- TAO Adapter (cba_rte.swf) provides API
CBA TAO Adapter

Stimulus Area (CBA Item Area) implemented using HTML Element “iframe”
API TAO Adapter

CBA TAO API (in Flash)
- StartTask(Task, X, Y, W, H, SnapshotRef)
- EndTask(TaskID)

TAO Callback Messages
- notify()
  Return message
- record_trace()
  Trace events record all user interactions

**Task** corresponds to an “inquiry” in TAO. A CBA Item may contain several tasks e.g starting with different pages.

**Snapshot** represents the state of a CBA Item, allows restoring when going back to a task.
CBA Item Life-Cycle

Life-Cycle of a “CBA Item“ at Runtime

- **Loaded**
  - CBA Item is loaded the first time a task is called (global resource)

- **Active**
  - CBA Item is initialized for the session and the requested task (session state)

StartTask()

- First initialization when calling the task for the first time (no snapshot) or
- Restore last state (using snapshot) when continuing a task

EndTask()

- Saves the current state and returns a snapshot
- Computes the score “on the fly” and returns the score result (useful for adaptive testing)
Java Script API

More General Java Script API (without Flash Adapter)

**Start/Restart a task of a CBA Item**
- function: `StartTask()`
- callback function: `extCBAEvent(logText)`

**Stop a running task**
- function: `StopTask()`
- callback function: `extCBAEndTaskResponse(itemScore, snapshotUrl)`

**Notify about any user interaction**
- callback function: `extCBAEvent(logText)`

**Save an intermediary state**
- function: `RequestSnapshot()`
- callback function: `extCBASnapshotResponse(snapshotUrl)`
CBA Item Builder:

- Editor with graphical user interface
- Easy to handle for non-IT-specialists
- Wide spectrum of graphical possibilities, templates
- WYSYWYG (preview of test item)
- Short development times
- Support of test development process (translation and adaptation of test items)
Usage and Examples

Usage: The user interface of the CBA IB

Set of pages
Project view
Designer view (page editor)
Graphical interface / drawing board
Toolbar
Properties view
Editor for single graphical components
Tool set for graphical components
Usage and Examples

What types of items does the CBA IB support?

- Static / dynamic
- Simple / complex
- Text only / multimedia
- Linear structure / hypertext structure
Usage and Examples

Examples: Simple questionnaire

<table>
<thead>
<tr>
<th>Question</th>
<th>Not at all</th>
<th>Very much</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. How interested are you in starting this test?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. What do you think - will your results be good?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. How useful is this test in your opinion?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. How easy is this test in your opinion?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5. How important is it for you to score well in the test?</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6. How difficult is this test in your opinion?</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Name</th>
<th>Sandra Smith</th>
</tr>
</thead>
<tbody>
<tr>
<td>Profession</td>
<td>Scientist</td>
</tr>
<tr>
<td>Age</td>
<td>35</td>
</tr>
</tbody>
</table>
Usage and Examples

Examples: Simple multiple choice item

**Deductive Thinking - figural part**

In the following tasks you will be given a set of pictures. The pictures vary from box to box following certain rules. You should continue the set consistently and determine the next two pictures in the row.

Please have a look at the example on the right. The set needs to be completed. Choose the correct solution from the possible answers. Below each box with a question mark you will find three possible solutions.

In the example on the right the fourth picture needs to consist of four overlapping squares. The fifth picture needs to consist of five overlapping squares.

You will have **14 minutes** to work on the following **16 tasks**. You are **not allowed to make any notes**.
Usage and Examples

Examples: Simulation of a technology rich environment

- Web Browser Simulation
- Simulation of a web application including search functionality
- Hypertext structure
Usage and Examples

Examples: Problem solving scenario

- Simulation of a dynamic system consisting of states, events and transitions
- Graphical representation of an automaton
- Keys are linked to events and trigger transitions
- Display represents states

INSTRUCTIONS
Set the clock of the mobile phone to summer time.
Usage and Examples

Examples: Interaction and response modes

• Click responses (via mouse)
  – Radiobuttons, Buttons, Checkboxes etc.
  – Clickable image areas
  – Clicking on a hyperlink
  – Choosing an item from a drop down box
  – Menu items

• Highlighting (via mouse)

• Numeric and textual entries (via keyboard)
Usage and Examples

Examples: Click responses

- Radiobuttons (1 out of x)
- Checkboxes (x out of y)
- Button
- Dropdown menu (1 out of x)
- Menu items
- Text link
- Clickable Image Area

Exercise: Please mark the shortest path from your hotel to the workshop.
Usage and Examples

Examples: Highlighting text

Preschool Education
The Little Test Takers

In this era of testing and accountability, preschool is the last frontier. Most preschool educators are coming to accept that with more states getting into the business of educating 3- and 4-year-olds, the political demands for accountability—for showing that preschools and kindergartens work—are inevitable. This means that children as young as 3 have to be observed, evaluated and in some way tested.

In developing ways to measure preschool quality, Mr. Bruner says, "there are concerns about, do you focus on early preliteracy and the cognitive aspects of development, or do you focus on the social and emotional and physical well-being of the children?"

But the purpose generally has been to provide schools with an internal map that shows where students are thriving, and where they are faltering—not to satisfy outside authorities. Considering the emotion that surrounds the mass testing of older children, it should be no surprise that the discussion about how to evaluate 3- and 4-year-olds is developing into a passionate slugfest that involves devilishly technical arguments over how young children learn differently from their older siblings.

"We do want to know how well preschool programs are doing, so policy makers are asking for accountability," says Charles R. Bruner, director of the State Early Childhood Technical Assistance Network.

Source: New York Times
Usage and Examples

Examples: Highlighting text

Preschool Education
The Little Test Takers

In this era of testing and accountability, preschool is the last frontier. Most preschool educators are coming to accept that with more states getting into the business of educating 3- and 4-year-olds, the political demands for accountability—for showing that preschools and kindergartens work—are inevitable. This means that children as young as 3 have to be observed, evaluated and in some way tested.

In developing ways to measure preschool quality, Mr. Bruner says, "there are concerns about, do you focus on early preliteracy and the cognitive aspects of development, or do you focus on the social and emotional and physical well-being of the children?"

But the purpose generally has been to provide schools with an internal map that shows where students are thriving, and where they are faltering—not to satisfy outside authorities. Considering the emotion that surrounds the mass testing of older children, it should be no surprise that the discussion about how to evaluate 3- and 4-year-olds is developing into a passionate tug fest that involves devilishly technical arguments over how young children learn differently from their older siblings.

"We do want to know how well preschool programs are doing, so policy makers are asking for accountability," says Charles Bruner, director of the State Early Childhood Technical Assistance Network.

Source: New York Times
Usage and Examples

Examples: Numeric and textual entries
Questions?