Inside the Tailoring Engine

Dennis O’Reilly
doreilly@umich.edu

Overview

- What is the Tailoring Engine?
- Inputs
- Outputs
- What the Tailoring Engine needs to run
- What’s really going on in there
  - The “Pipeline”
- Putting it all together
What Is The Tailoring Engine?

- Written in Python
  - Runs under CPython 2.5 and higher
    - Available for Windows, Mac OS X, Linux
  - Runs under Jython 2.2 and higher
    - Runs in a Java Virtual Machine
- Take a set of messages with rules that define when they should show
- Apply the rules based on a person’s personalized profile
- Generate text from the relevant messages
Inputs

- Dictionaries
- Message Documents
- Subjects

Dictionary Files

- Contains
  - Characteristics
  - Restrictions
  - Sources
  - Base Types
  - Miscellaneous other items
- Format
  - XML
Characteristics

- A unit of participant information used in message logic that directs the MTS Engine to choose only certain messages for participants.
- Has a data type
  - We call this a BaseType, in the UI it’s called Value Type
- Like a variable in most programming languages
  - No assignment, however
- Can be computed functionally
  - Called a “Derived Characteristic”

Base Types

- A definition of known value types for Characteristics to be stored in
- We have
  - String/Text
  - Integer/Number without decimal
  - Float/Number with decimal
  - DateTime
    - YYYY-MM-DD hh:mm:ss
- Possible to add, change, or remove
  - May want the Characteristic types to reflect underlying data storage type system
    - SQL Types
    - Java/C#/<insert your favorite language here> datatypes
Restriction

- Limits on what a characteristic's value may legally be.
- Can apply to multiple Characteristics
- Different Kinds
  - Value
    - Must be one of a set of pre-defined items
    - Like an enum type in C
    - Eg, "Agree", "Disagree", "Neither"
  - Length
    - A string must have a certain number of characters
    - Eg, First Name must be between 1 and 30 characters long.
  - Range
    - A numerical value must be within limits
    - Eg, A US phone number must be between 7 and 10 digits.
  - Function
    - Don't really go there

Message Documents

- Contents
  - Sections
  - Containers
    - Block
    - Select
    - List
  - Leaves
    - Text
    - Graphic
    - Heading
    - Paragraph
- Format
  - XML
What's a Message Doc?

- On-screen demo

Subject

- A collection of characteristics for the purposes of a tailoring run
- Usually represents one person
- Primary Characteristics
  - Used as the basis for tailoring logic and textual replacement
- Selection Characteristics
  - Values intended only for the Selection phase of the Tailoring Engine
- Message Characteristics
  - Values intended only for the Substitution phase of the Tailoring Engine
Testcase Files

- A simplified, serialized subject
- Used for testing, obviously
- Contents
  - Lists of Key-value pairs of specific, typed values
  - Lists are nested on a per-source basis
- Format
  - JSON
    - JavaScript Object Notation
    - http://www.json.org
    - '{"FirstName":"Albert","LastName":"Einstein","IQ":400,"HairLengthInches":9.3}'

Outputs

- (X)HTML Tree
- Error List
  - Objects representing each problem encountered during the tailoring process
- Honest Truth
  - This is what you asked for. Really.
  - Don’t forget about the errors
What the Tailoring Engine needs to run

- Message Document
- Dictionary
- A Subject
- Named command
  - Generally a Section

The Pipeline

- Each stage receives
  - Message Document XML Tree Representation
  - Evaluation Context
- Each stage produces
  - Modified Message Document XML Tree Representation
  - List of Errors
Evaluation Context

- A single object for managing lookups into a Subject during the execution of the Tailoring Engine
- Multiple Evaluation Contexts during a run
  - "Selection"
  - "Substitution"
- Created within the Tailoring Engine
What does “Selection” Mean?

- Walking through the tree and asking
  - “Is this command a candidate for further processing?”
  - Technically: evaluate any element with logic in the ‘if’ attribute
- Not to be easily confused with a ‘select’ command or element
  - Applies to any command

Selection

- The contents of the ‘if’ attribute are evaluated in the passed in EvaluationContext
- If the expression is true, ‘selected=”true”’ is appended to the evaluated element
- Similarly, ‘selected=”false”’ is appended to elements where the logic evaluates falsely
- Any errors raised during expression evaluation are placed in a List
  - Wrapped in a ProcessingError instance
- It is possible for no commands to be selected.
Selection Example

Consider: $\text{Age} = 20$

```xml
<section name="Header1" msgid="5">
    <subelements>
        <text if="Age &lt; 15" selected="false" msgid="6">
            <content>Someday, you too can drive.</content>
        </text>
        <text if="Age &ge; 15" selected="true" msgid="7">
            <content>Turn the key, baby</content>
        </text>
    </subelements>
</section>
```
Selection

Ordering

Limiting

Substitution

Rendering

What does it mean?
- Choose the order in which commands will be placed in final output

When does it happen?
- Children of Select and List commands

Why does it matter?
- Useful when using 'order="random"'
- Highly uninteresting otherwise
- Applies to children of select/list commands
- Relevant elements are tagged with an index attribute
  - index="<positive integer>"
Ordering Example

<section name="Advice">
    <subelements>
        <select orderby="random">
            <subelements>
                <text if="Issue1 == 'seatbelt'">
                    <content>Buckle your seatbelt.</content>
                </text>
                <text if="Issue1 == 'speeding'">
                    <content>Follow all posted speed limits.</content>
                </text>
                <text if="Issue1 == 'turnsignal'">
                    <content>Signal your intentions with your turn signal.</content>
                </text>
                <text if="Issue1 == 'bicycles'">
                    <content>Share the road with bicycles.</content>
                </text>
            </subelements>
        </select>
    </subelements>
</section>

Ordering Example (Selected)

<section name="Advice">
    <subelements>
        <select orderby="random">
            <subelements>
                <text if="Issue1 == 'seatbelt'" selected="true">
                    <content>Buckle your seatbelt.</content>
                </text>
                <text if="Issue1 == 'speeding'" selected="false">
                    <content>Follow all posted speed limits.</content>
                </text>
                <text if="Issue1 == 'turnsignal'" selected="false">
                    <content>Signal your intentions with your turn signal.</content>
                </text>
                <text if="Issue1 == 'bicycles'" selected="true">
                    <content>Share the road with bicycles.</content>
                </text>
            </subelements>
        </select>
    </subelements>
</section>
Ordering Example (Ordered)

```xml
<section name="Advice">
  <subelements>
    <select orderby="random">
      <subelements>
        <text if="Issue1 == 'seatbelt'" selected="true" index="3">
          <content>Buckle your seatbelt.</content>
        </text>
        <text if="Issue1 == 'speeding'" selected="false" index="4">
          <content>Follow all posted speed limits.</content>
        </text>
        <text if="Issue1 == 'turnsignal'" selected="false" index="2">
          <content>Signal your intentions with your turn signal.</content>
        </text>
        <text selected="true" index="1">
          <content>Share the road with bicycles.</content>
        </text>
      </subelements>
    </select>
  </subelements>
</section>
```

Limiting

```
Selection    Ordering    Limiting    Substitution    Rendering
```

University of Michigan Tailoring Workshop
“Limiting”?

- What does it mean?
  - A reduction of selected commands to fit authors’ requirements.
- When does it happen?
  - Children of a select or list command
- Why does it matter?
  - When a select/list has a ‘limit’ attribute
  - Many commands may be ‘selected’, however only a maximum of ‘limit’ are kept.
- Commands are attributed with ‘limited=false’ until ‘limit’ is reached. ‘limited=true’ thereafter

```
<section name="FuelChoice">
  <subelements>
    <select limit="1" orderby="sequential">
      <text if="FuelType == 'Gas'">
        <content>Choose the correct octane for your vehicle's engine.</content>
      </text>
      <text if="FuelType == 'Diesel'">
        <content>Remember to choose the right pump when you fill up.</content>
      </text>
      <text if="FuelType == 'E85'">
        <content>Consider choosing the currently less-expensive E85 fuel.</content>
      </text>
      <text>
        <content>Looks like your vehicle is using an alternative fuel source. Sweet.</content>
      </text>
    </select>
  </subelements>
</section>
```
Limiting Example (Selected)

```xml
<section name="FuelChoice">
    <subelements>
      <select limit="1" orderby="sequential" selected="true">
        <subelements>
          <text if="FuelType == 'Gas'" selected="true">Choose the correct octane for your vehicle's engine.</content>
        </text>
        <text if="FuelType == 'Diesel'" selected="false">Remember to choose the right pump when you fill up.</content>
        <text if="FuelType == 'E85'" selected="false">Consider choosing the currently less-expensive E85 fuel.</content>
        <text selected="true">Looks like your vehicle is using an alternative fuel source. Sweet.</content>
      </select>
    </subelements>
</section>
```

Limiting Example (Ordered)

```xml
<section name="FuelChoice">
    <subelements>
      <select limit="1" orderby="sequential" selected="true">
        <subelements>
          <text if="FuelType == 'Gas'" selected="true" index="1">Choose the correct octane for your vehicle's engine.</content>
        </text>
        <text if="FuelType == 'Diesel'" selected="false" index="2">Remember to choose the right pump when you fill up.</content>
        <text if="FuelType == 'E85'" selected="false" index="3">Consider choosing the currently less-expensive E85 fuel.</content>
        <text selected="true" index="4">Looks like your vehicle is using an alternative fuel source. Sweet.</content>
      </select>
    </subelements>
</section>
```
<section name="FuelChoice">
  <subelements>
    <select limit="1" orderBy="sequential" selected="true">
      <subelements>
        <text if="FuelType == 'Gas'" selected="true" index="1" limited="false">
          <content>Choose the correct octane for your vehicle's engine.</content>
        </text>
        <text if="FuelType == 'Diesel'" selected="false" index="2" limited="true">
          <content>Remember to choose the right pump when you fill up.</content>
        </text>
        <text if="FuelType == 'E85'" selected="false" index="3" limited="true">
          <content>Consider choosing the currently less-expensive E85 fuel.</content>
        </text>
        <text selected="true" index="4" limited="true">
          <content>Looks like your vehicle is using an alternative fuel source. Sweet.</content>
        </text>
      </subelements>
    </select>
  </subelements>
</section>
**“Substitution”?**

- **What does it mean?**
  - Replace placeholders in Message Content with content located elsewhere
- **When does it happen?**
  - Any command with message content
    - Text
    - Heading
    - Graphic
- **Why does it matter?**
  - Can greatly simplify logic
  - Allows for direct insertion of Subject values
    - User-supplied values that are innumerable like names, favorite brand of socks, etc.
    - Simple one-word values
      - Mood: happy, sad, indifferent.

---

### Substitution

- **Where do they come from?**
  - A Subject
    - Survey responses (raw characteristic values)
  - A Dictionary
    - Replacements for pre-defined values
      - LessThanWorthIt -> “DO NOT WANT”
      - TotallyWorthIt -> “Buy It Now”;
  - What do they look like?
    - Substituted values are placed inline with text, prefixed with a dollar sign ($)
      - “Dos Vidanya, Comrade $LastName.”
    - Dollar signs can be inserted literally typing two in a row ($$)
      - “$3.99 are they out of their mind?”
    - Optionally, if there is an ambiguity, they can be wrapped in braces, with a dollar sign prefix
      - “Since you are ${Age}ish”
**Substitution Example**

```xml
<section name="Intro">
    <subelements>
        <text>
            <content>Take a load off, $FirstName.</content>
        </text>
    </subelements>
</section>
```

**Substitution Example (Selected)**

```xml
<section name="Intro">
    <subelements>
        <text selected="true">
            <content>Take a load off, $FirstName.</content>
        </text>
    </subelements>
</section>
```
Substitution Example (Ordered)

```
<section name="Intro">
  <subelements>
    <text selected="true">
      <content>Take a load off, $FirstName.</content>
    </text>
  </subelements>
</section>
```

Substitution Example (Limited)

```
<section name="Intro">
  <subelements>
    <text selected="true">
      <content>Take a load off, $FirstName.</content>
    </text>
  </subelements>
</section>
```
Substitution Example (Substituted)

```xml
<section name="Intro">
  <subelements>
    <text selected="true">
      <content>Take a load off, Dennis.</content>
    </text>
  </subelements>
</section>
```
“Rendering”?  

- What does it mean?  
  - Take the fully attributed tree and bend it into a form useful for output  
- Outputs in different possible formats  
  - HTML  
    - Content is converted to X-HTML, returned in a surrounding <div>  
  - Plain Text  
    - As HTML, but with all tags stripped out.

Rendering Example

```
<section name="HowdThatGetInThere">  
  <subelements>  
    <block class="quotation" tag="blockquote">  
      <subelements>  
        <text>When nothing feels like you're doing something, you can't get more energy-efficient than that.</text>  
        <text class="attribution">Stephen T. Colbert</text>  
      </subelements>  
    </block>  
  </subelements>  
</section>
```

```
<div id="HowdThatGetInThere">  
  <blockquote class="quotation">  
    When nothing feels like you're doing something, you can't get more energy-efficient than that.  
    <span class="attribution">Stephen T. Colbert</span>  
  </blockquote>  
</div>
```
Demo

Q&A