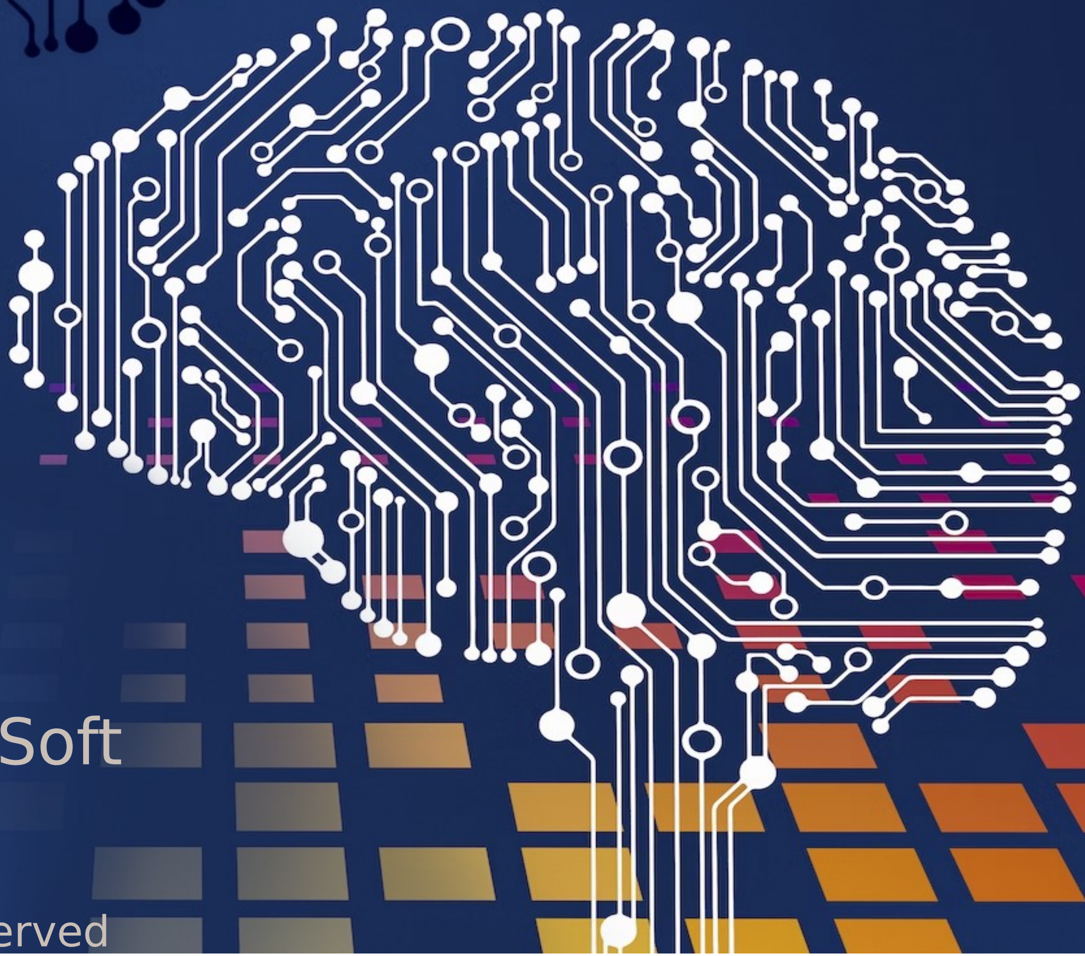
XPath AI Extension Functions



Octavian Nadolu, Syncro Soft
octavian_nadolu@oxygenxml.com
@OctavianNadolu

Transform Content Using AI

- Examples:
  - Make the text easy to read and understand
  - Correct spelling and grammar
  - Rephrase to use active voice
  - Rephrase to have 20 words
  - Rephrase paragraph to answer to the following question
  - Translate content

Generate Content Using AI

- Examples:
    - Generate a template file
    - Add missing cells from table
    - Generate content based on a schema
    - Generate documentation for code
    - Explain current code

Verify Content Using AI

Examples of verification with AI:

- Is the text easy to read and understand?
- Is spelling and grammar correct?
- Is active/passive voice used in the description?
- Is this written according to the style guide?
- Does the topic answers to this <question>?

XPath AI Extension Functions

- Call AI engine API using an XPath extension function
- Implementation can differ depending on the AI model you are using
- Extension functions
  - Transform the content using an instruction from the user, return the generated content

    ```
    xs:string ai:transform-content(xs:string instruction, xs:string content)
    ```

  - Verify the content using an instruction from the user, return true or false

    ```
    xs:boolean ai:verify-content(xs:string instruction, xs:string content)
    ```

Transform Content Function

- Function provides a specific built-in prompt
- User specifies the instruction and the content to be transformed

```
ai:transform-content(instruction, content)
```

The function has the string parameters:

- instruction - The instruction to be performed on the content.
- content - The content to be transformed.
- It returns a string that represents the transformed content.

"You are a developer and you need perform the following task:"
+ *Rephrase to use active voice* + *content*

Verify Content Function

- Function can provide a specific built-in prompt
- User can specify the instruction and the content to be verified

```
ai:verify-content(instruction, content)
```

The function has the string parameters:

- instruction - The OpenAI instruction to be performed on the content.
- content - The content to be verified.
- It returns a boolean value that represents the result of the verification.

```
“You are a technical writer and you need to verify the following
and respond with true or false:“
+ Is active voice used in the description? + content
```

Transform Content Function+

- Provide a more detailed context using (user, agent) pairs

```
ai:transform-content(instruction, (user, agent)*, content)
```

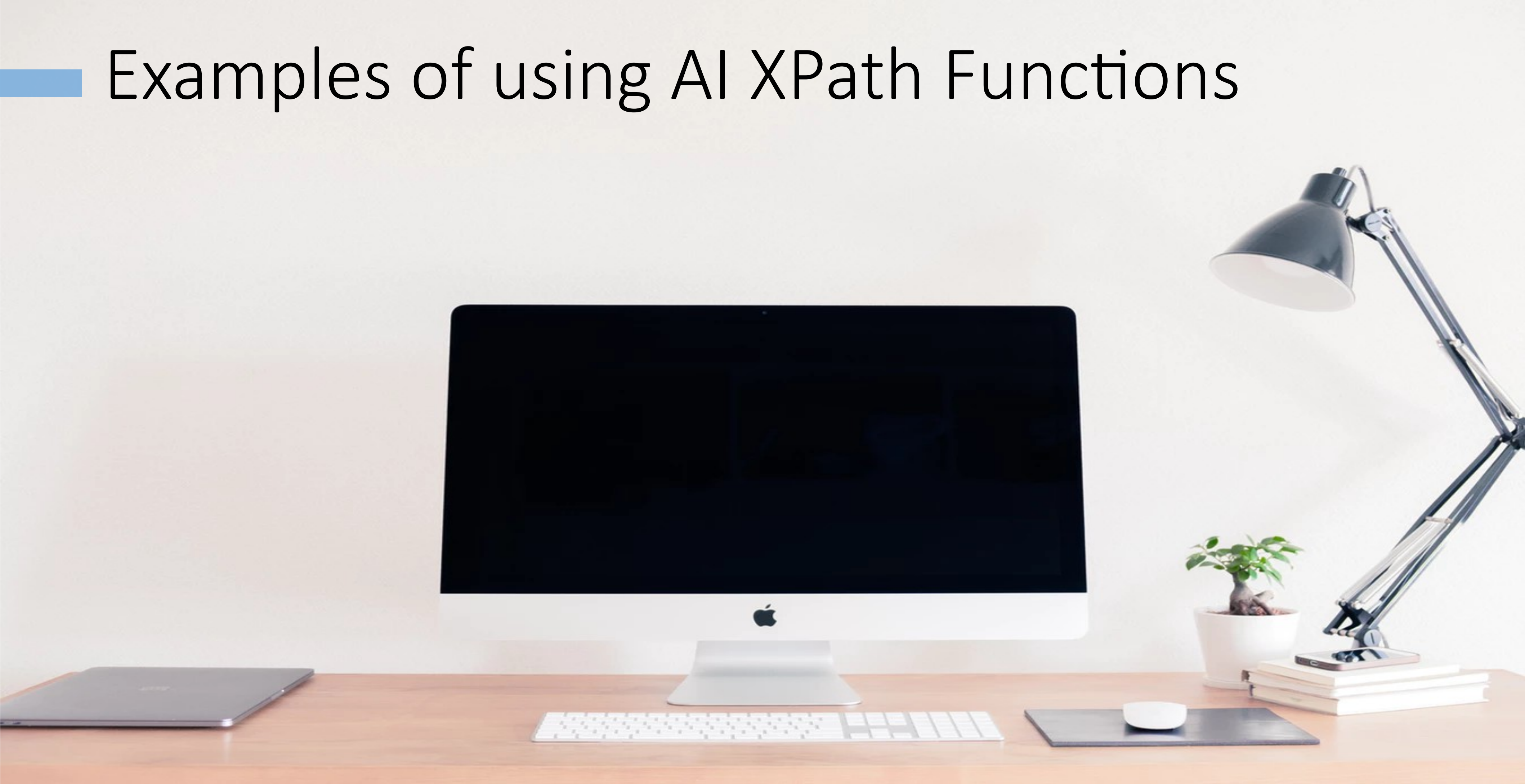
“Act as a technical writer and perform the following task:“
+ Rephrase to use active voice. Choose the best variant based to the strong points
+ (Rephrase and provide 3 variants + agentResult)
+ (Provide the strong points for each variant + agentResult)
+ content

Verify Content Function+

- Provide a more detailed context using (user, agent) pairs
  - `ai:verify-content(instruction, (user, agent)*, content)`

  -
    ```
    “You are a technical writer and you need to verify the following
    and respond with true or false:“
    + Is active voice used in the description?
    + (user + agent)
    + (user + agent)
    + content
    ```

Examples of using AI XPath Functions



AI functions in Schematron and SQF

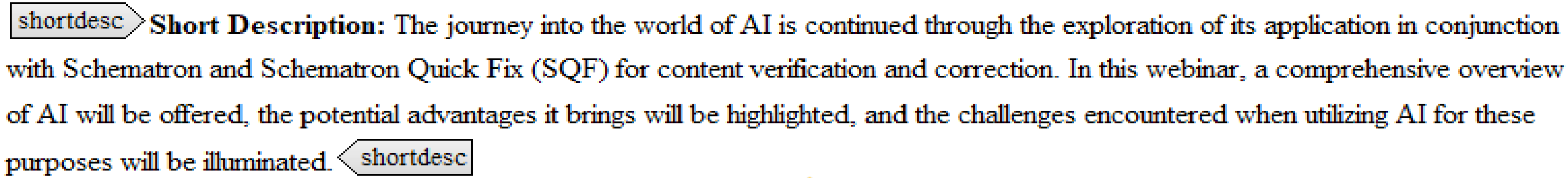
- Schematron - verify document content automatically using AI

- SQF - correct problems in document using AI

Verify content using Schematron

- Example of a rule that checks if the text uses active voice

In the description we should use active voice



shortdesc **Short Description:** The journey into the world of AI is continued through the exploration of its application in conjunction with Schematron and Schematron Quick Fix (SQF) for content verification and correction. In this webinar, a comprehensive overview of AI will be offered, the potential advantages it brings will be highlighted, and the challenges encountered when utilizing AI for these purposes will be illuminated. shortdesc

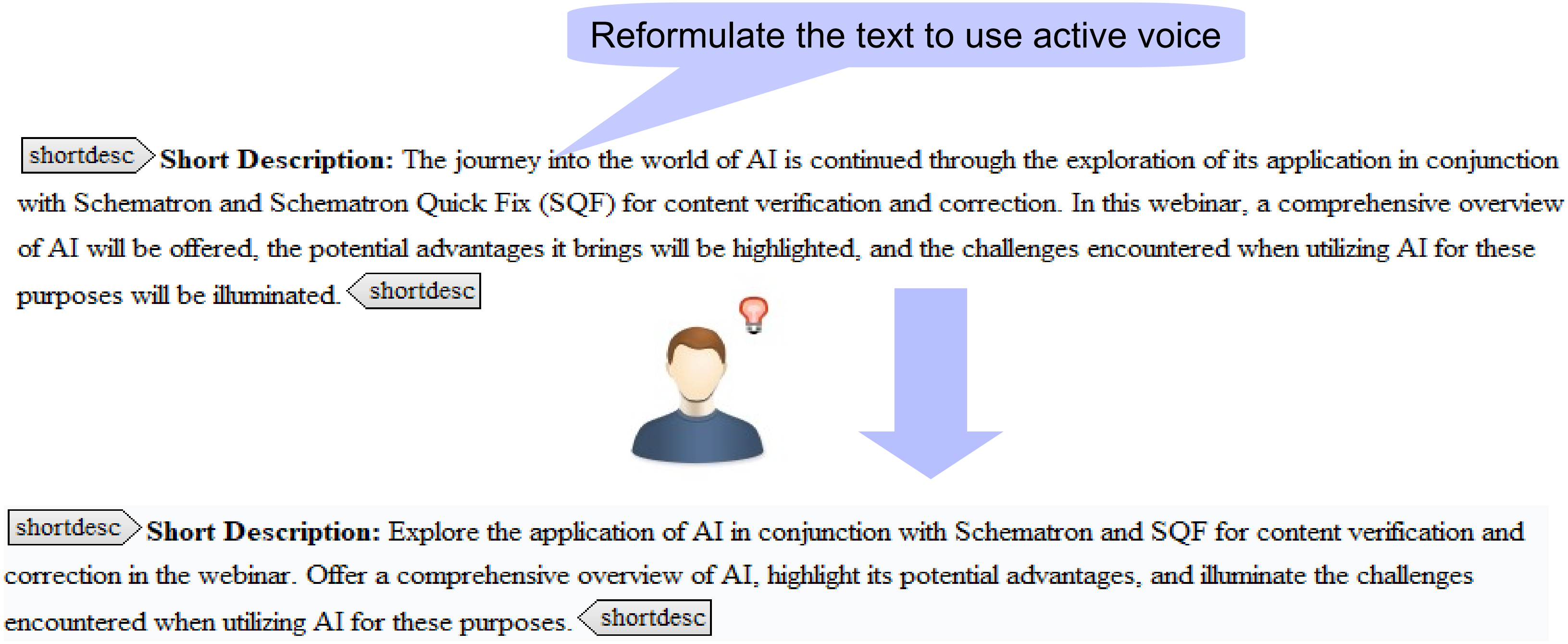
Check the text voice

- Rule that verifies if the text voice is active

```
<sch:rule context="shortdesc">
    <sch:assert test="ai:verify-content('Is active voice used?', .)">
        In the description we should use active voice.</sch:assert>
</sch:rule>
```

Correct the text voice

- Example fix that reformulates the text to use active voice



Reformulate the text to use active voice

shortdesc **Short Description:** The journey into the world of AI is continued through the exploration of its application in conjunction with Schematron and Schematron Quick Fix (SQF) for content verification and correction. In this webinar, a comprehensive overview of AI will be offered, the potential advantages it brings will be highlighted, and the challenges encountered when utilizing AI for these purposes will be illuminated. shortdesc

shortdesc **Short Description:** Explore the application of AI in conjunction with Schematron and SQF for content verification and correction in the webinar. Offer a comprehensive overview of AI, highlight its potential advantages, and illuminate the challenges encountered when utilizing AI for these purposes. shortdesc

Correct the text voice

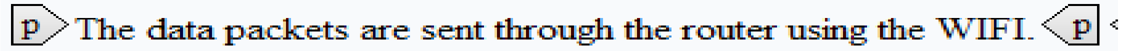
- SQF fix that that reformulates the text to use active voice

```
<sqf:fix id="rephrase">
    <sqf:description>
        <sqf:title>Reformulate the text to use active voice</sqf:title>
    </sqf:description>
    <sqf:replace match="text()" select="ai:transform-content('
             Reformulate to use active voice', .)"/>
</sqf:fix>
```

Check technical terms

- Example of a rule that checks if the technical terms are not explained adequately

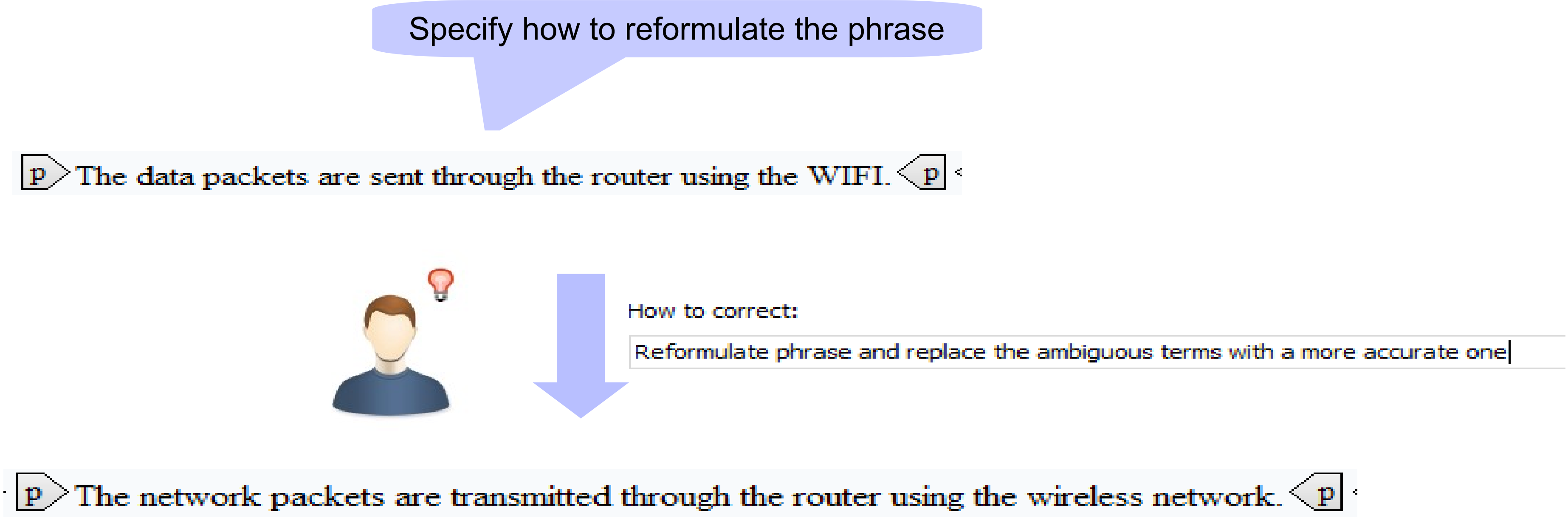
The text uses WIFI term that is not explained adequately



p The data packets are sent through the router using the WIFI. p

Correct terms

- Example fix that allows the user to specify how to reformulate the phrase



Specify how to reformulate the phrase

p The data packets are sent through the router using the WIFI. p

How to correct:

Reformulate phrase and replace the ambiguous terms with a more accurate one

p The network packets are transmitted through the router using the wireless network. p

Correct terms

- SQF fix that allows the user to specify the prompt that will be sent to the AI

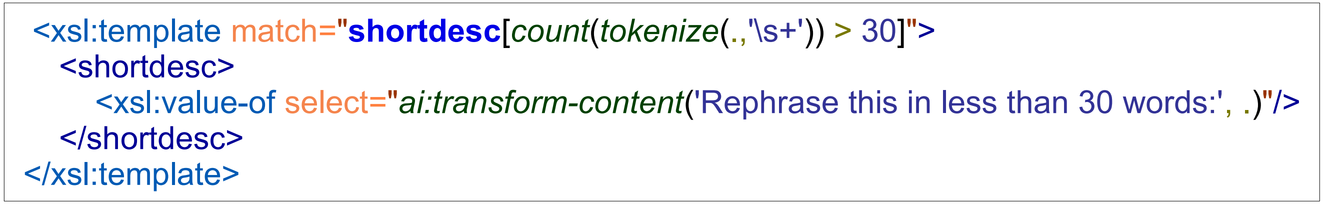
```
<sqf:fix id="reformulateUser">
    <sqf:description>
        <sqf:title>Specify how to reformulate the phrase</sqf:title>
    </sqf:description>
    <sqf:user-entry name="userInput" default="'
         Reformulate phrase and replace the ambiguous terms with a more accurate one'">
        <sqf:description><sqf:title>How to correct:</sqf:title>sqf:description>
    </sqf:user-entry>
    <sqf:replace match="text()" select="ai:transform-content($userInput, .)"/>
</sqf:fix>
```

AI functions in XSL and XQuery

- Transform content using AI
- Create refactoring actions based on AI

Refactor content using XSL and AI

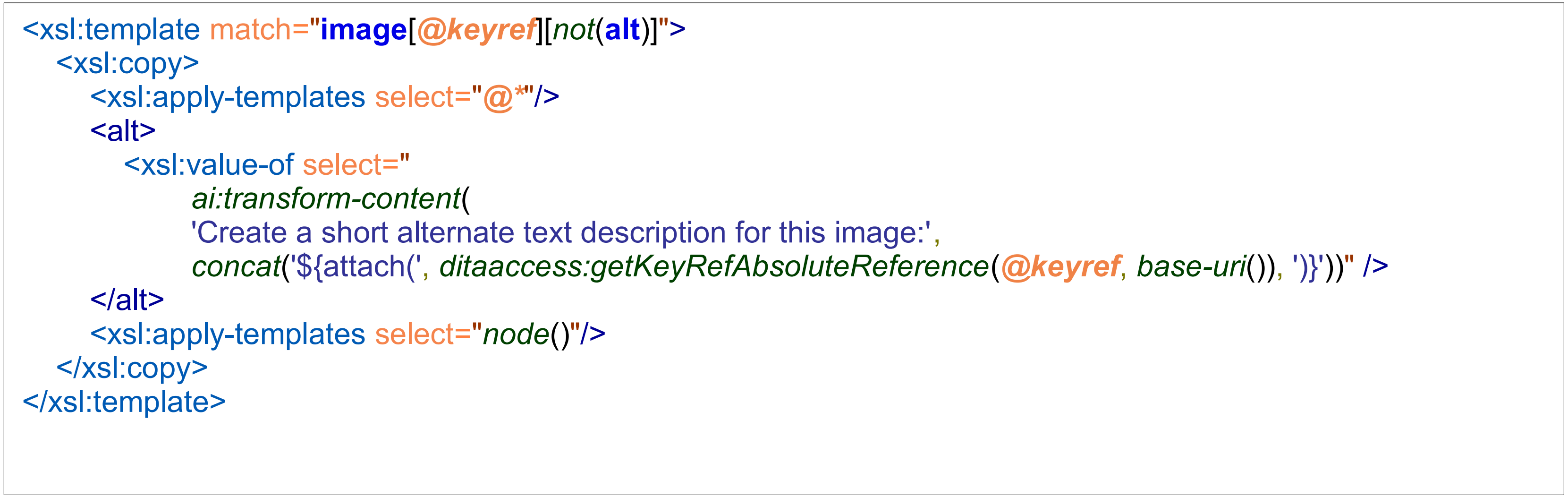
- Rephrase the short description element content to have less than 30 words



```
 <xsl:template match="shortdesc[count(tokenize(.,'\s+')) > 30]">
    <shortdesc>
       <xsl:value-of select="ai:transform-content('Rephrase this in less than 30 words:', .)"/>
    </shortdesc>
</xsl:template>
```

Refactor content using XSL and AI

- Rephrase the short description element content to have less than 30 words



```
<xsl:template match="image[@keyref][not(alt)]">
    <xsl:copy>
        <xsl:apply-templates select="@*"/>
        <alt>
            <xsl:value-of select="
                    ai:transform-content(
                    'Create a short alternate text description for this image:',
                    concat('${attach(', ditaaccess:getKeyRefAbsoluteReference(@keyref, base-uri()), ')}'))" />
        </alt>
        <xsl:apply-templates select="node()"/>
    </xsl:copy>
</xsl:template>
```

Refactor content using XSL and AI

- Translate text to French

```
<xsl:template match="text()">
    <xsl:value-of select="ai:transform-content('Act as a translation specialist.
                    Translate following content to French:', text())"/>
</xsl:template>
```

Refactor content using XSL and AI

- Translate text to French and Verify

```
<xsl:template match="text()">
  <xsl:variable name="agentFrench" select="ai:transform-content('Translate to French in 3 variants ', .)"/>

  <xsl:variable name="agentEnglish" select="ai:transform-content('Translate content from French to English ', $agentFrench)"/>

  <xsl:value-of  select="ai:transform-content('Compare the English translation with original content
    and use the correct French translation. ',
    'Translate to french in 3 variants ', $agentFrench,
    'Translate from french to english ', $agentEnglish, .)"/>
</xsl:template>
```

Conclusion

- AI can be used to generate and transform content
- Use XPath AI functions to automate the process
- Use AI in Schematron and SQF to verify and correct content
- Use AI in XSL and XQuery to transform and refactor content
- Always check the AI response

Resources

- oxygenxml.com/doc/ug-editor/topics/ai_positron.html
- blog.oxygenxml.com/topics/ai_positron.html
- https://github.com/f/awesome-chatgpt-prompts
- https://platform.openai.com/docs/guides/chat

Questions?

Octavian Nadolu
Project Manager at Syncro Soft

octavian.nadolu@oxygenxml.com
Twitter: @OctavianNadolu
LinkedIn: octaviannadolu