

Advanced XML development

XML Projects

George Cristian Bina

george@oxygenxml.com

[@georgebina](#) on Twitter

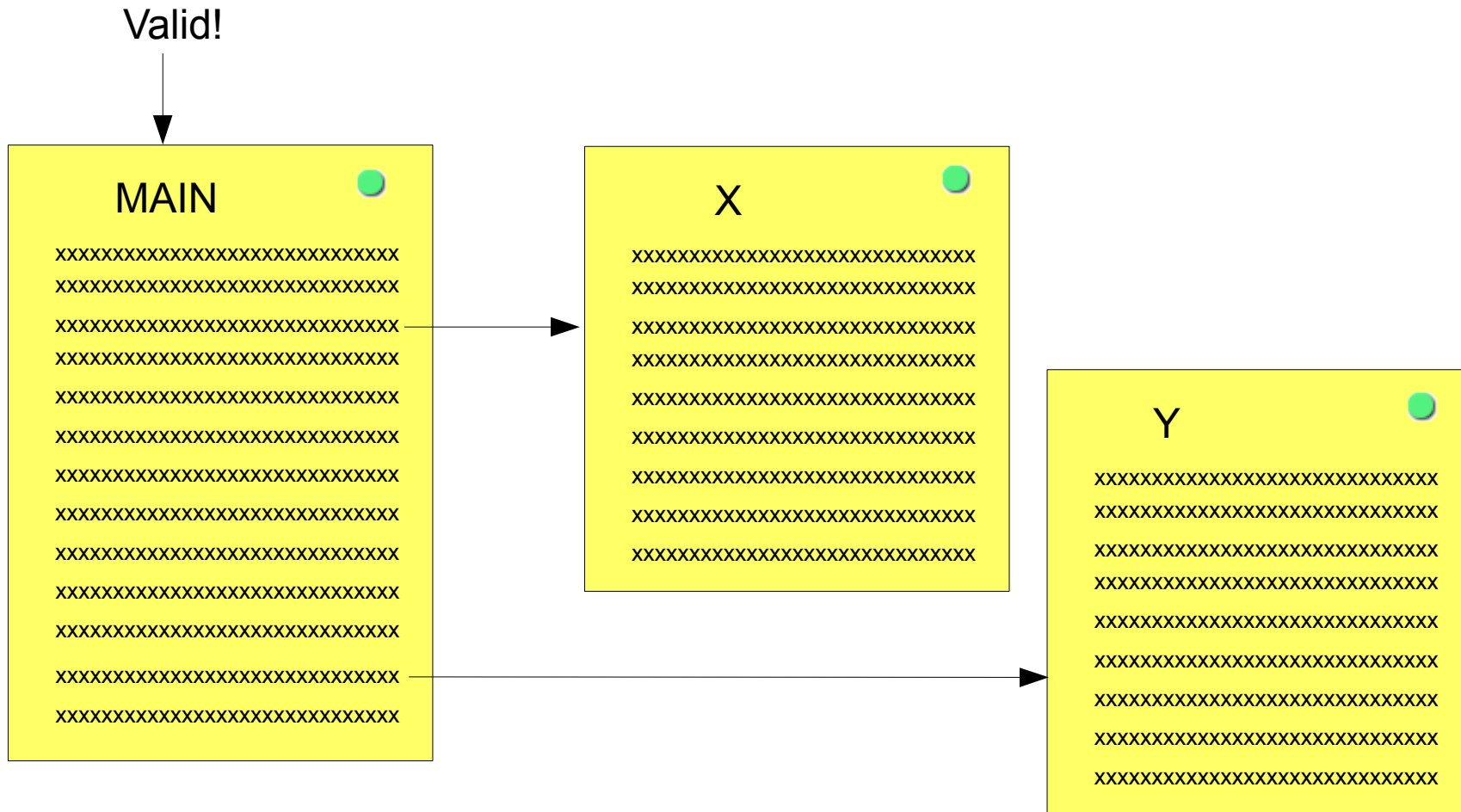


Java development

- Sources
- Libraries
- Output folder
- Other resources

XML development

Modules



Modules

DocBook XSL example [demo]

Modules

- XSLT
- XML Schema
- Relax NG
- XQuery
- XML

Modules – oXygen solution

Validation scenarios

a set of validation units, each can specify

- the file to be validated
- the engine used for validation
- other parameters (extension libraries, etc.)

Modules

DocBook XSL example in oXygen using validation scenarios [demo]

Modules

- Validation in context
- Editing in context
 - Show proposals also from other files, starting from the main file
 - Show all the components in the Outline view starting from the main file

What happens when we have more main files?

Search and refactoring

- Search
 - Declarations
 - References
- Refactoring
 - Rename – the most common refactoring action
 - Extract as...
 - Etc.

What files we should look into?

→ We need to define a scope

Hierarchy and dependencies

- Hierarchy
 - What components does the current component refers?
- Dependencies
 - What components have references to the current component?

Where do we look for references to the current component?

→ We need to define a scope

oXygen solution

- Default scope
 - All project files
- Allow the user to specify a scope
 - Define working sets
 - Select from the defined working sets
 - Use editor variables for dynamic working sets

Scope usage

Component dependencies [demo]

- Show dependencies of admon.graphics from admon.xsl

Rename an XSLT variable [demo]

- Rename admon.graphics from admon.xsl

Issues with the current situation

Solves the problems but...

- Not dynamic enough
 - Import a new file, that needs to have a scenario that refers to the master file
- Not that user-friendly
- oXygen specific – ideally we should have tool neutral development concepts, as in Java case
 - I am not sure if we can remove this in the short term

A possible XML Project

There are mainly 4 categories of resources:

- Sources
 - Main files
 - Modules
- Libraries
- Other resources

A possible XML project

This organization allows:

- Automatically determine the validation scenarios and editing context for any given file
- Automatically determine the scope for search, refactoring and dependencies operations

Sources vs libraries help on refactoring – you do not want to change a library

A possible XML project

Challenge:

Determine a user-friendly way to specify these resource categories!

Some conclusions

- We started to work on these some time ago and we are working actively also right now
- Your feedback is welcome!
- The current support (validation scenarios, scope) will be the base upon which we plan to implement the XML project support
- The best thing will be to have a common (tool neutral) set of concepts in the area of XML development

Thank you!

Questions?

<oxygen/> XML Editor
<http://www.oxygenxml.com>

@georgebina
george@oxygenxml.com