Understanding NVDL

The Anatomy of an Open Source XProc/XSLT implementation of NVDL

George Bina

george@oxygenxml.com
@georgebina

NVDL

Copyright @ Syncro Soft, 2012. All rights reserved.

NVDL

Namespace-based

Validation and

Dispatching

Language

ISO Standard (like Relax NG and Schematron)

Copyright @ Syncro Soft, 2012. All rights reserved.

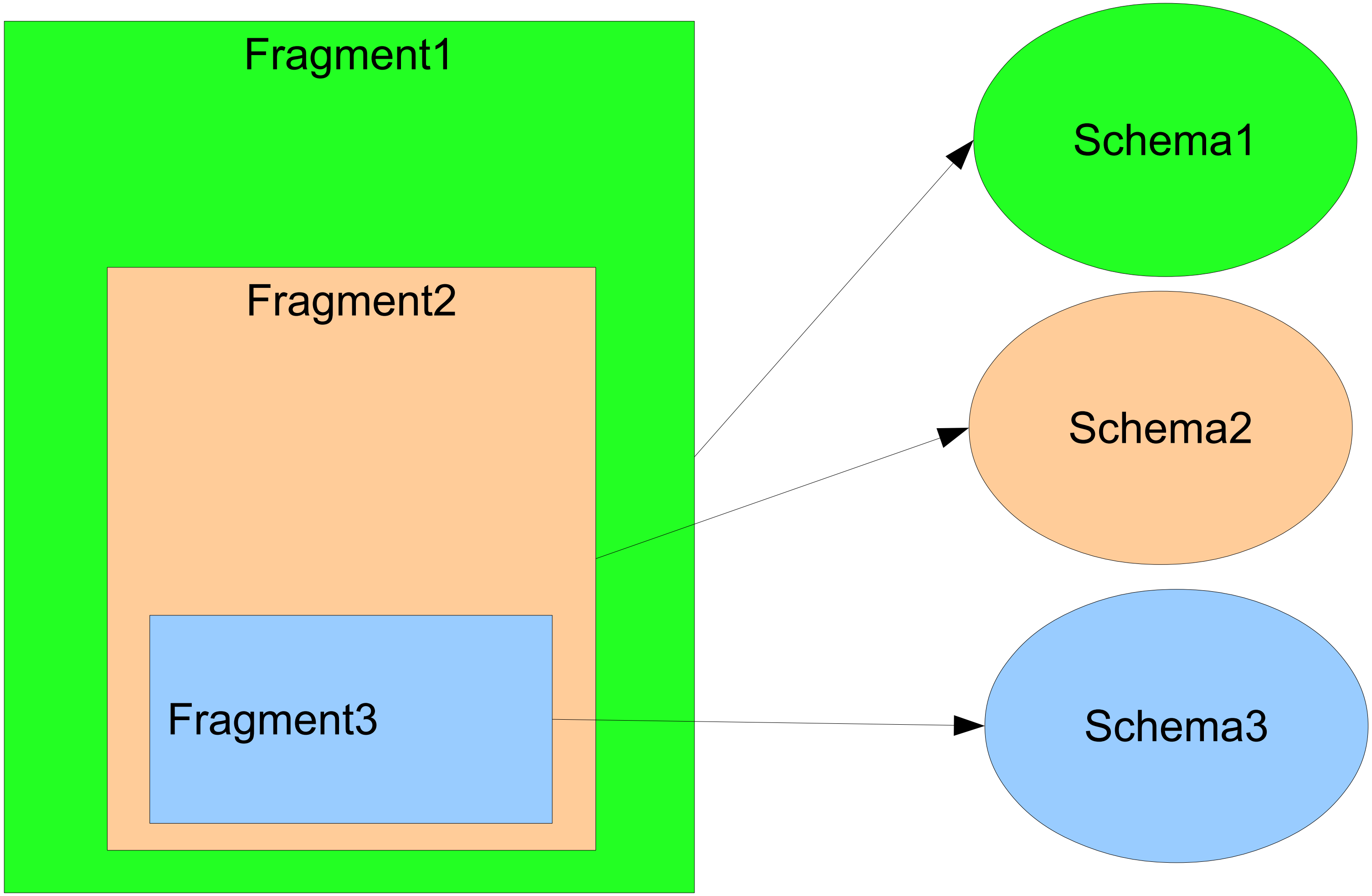
NVDL

A solution for validating compound documents

Copyright @ Syncro Soft, 2012. All rights reserved.

Problem

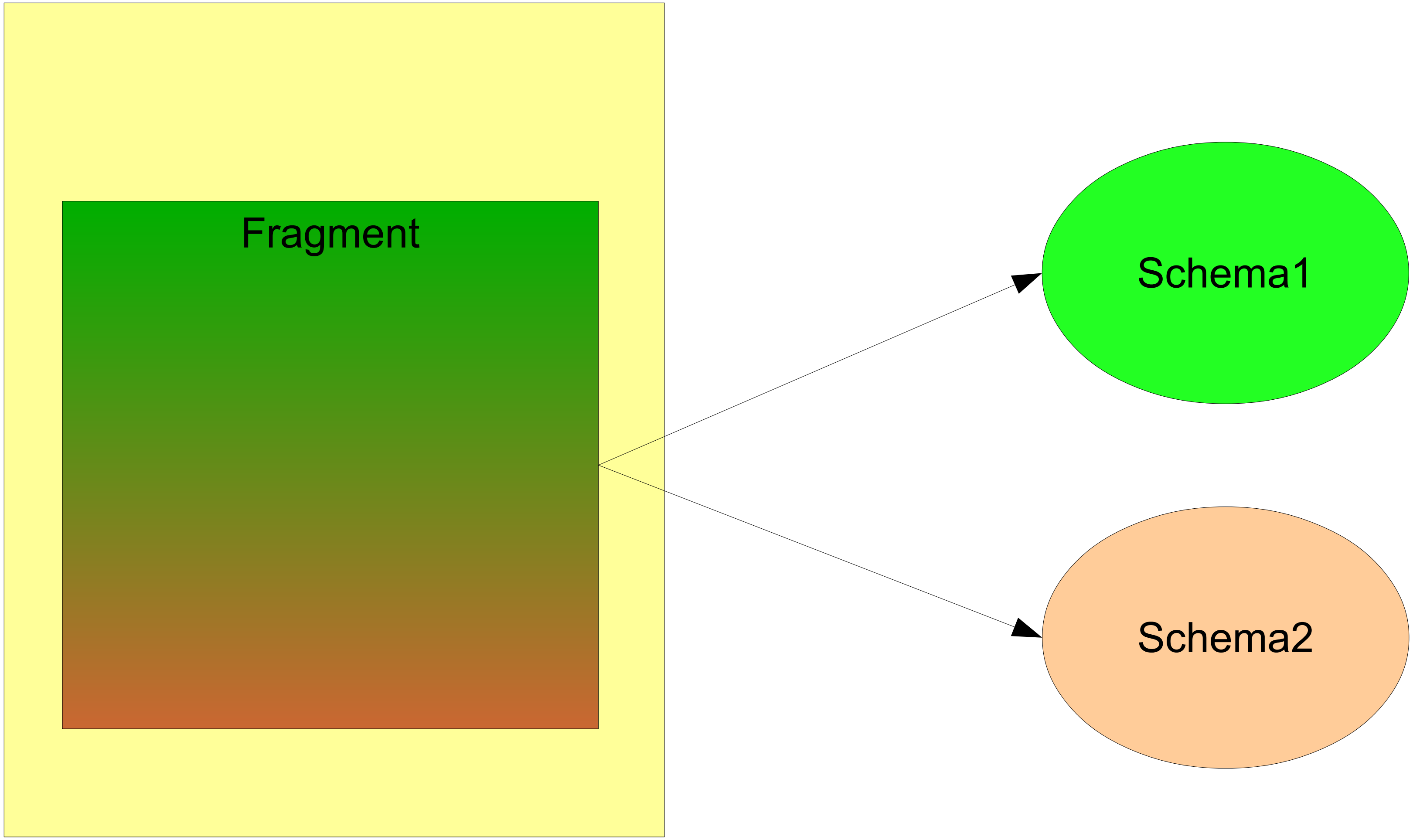
Validate different fragments with different schemas



Copyright @ Syncro Soft, 2012. All rights reserved.

Problem

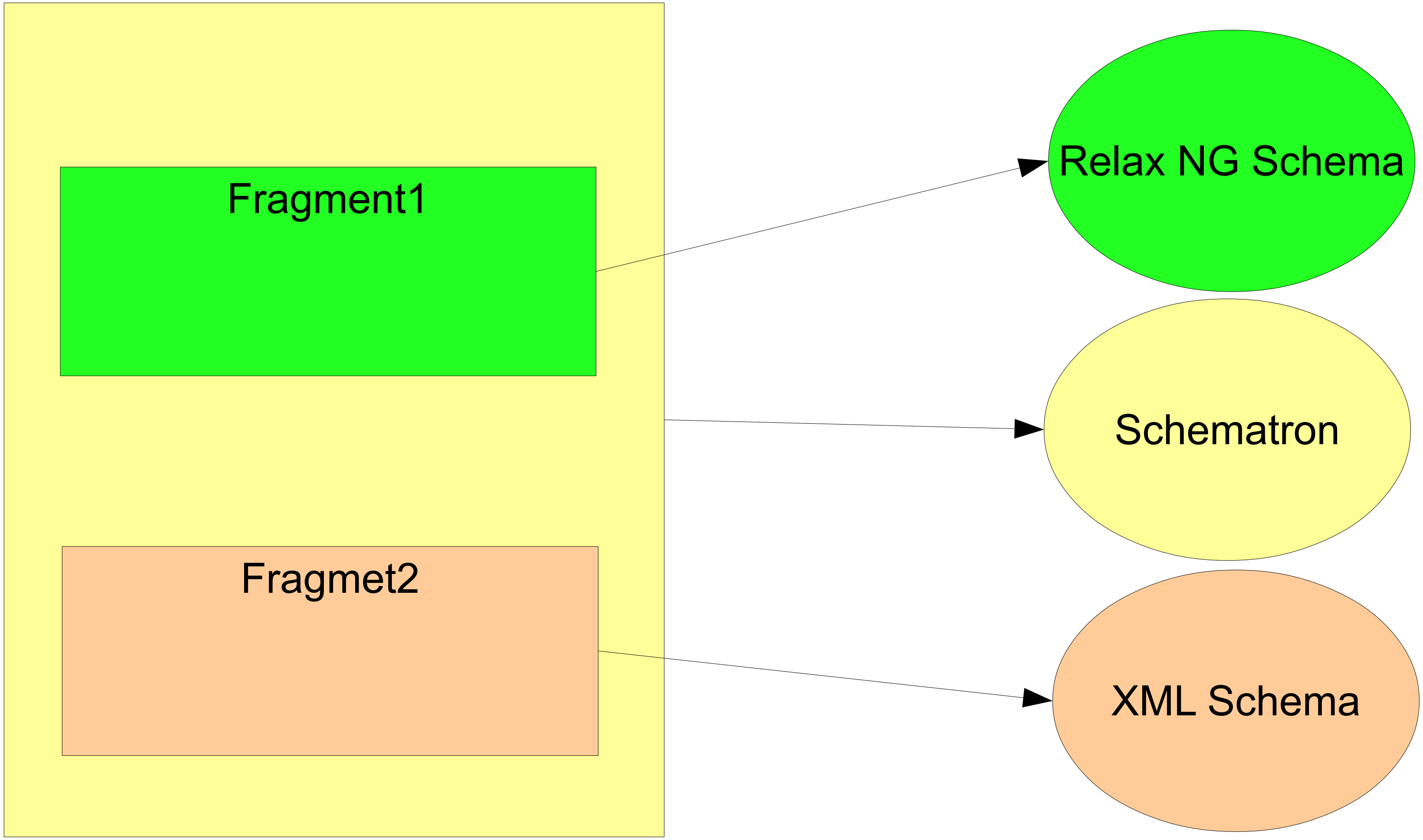
Validate concurrently against multiple schemas



Copyright @ Syncro Soft, 2012. All rights reserved.

Problem

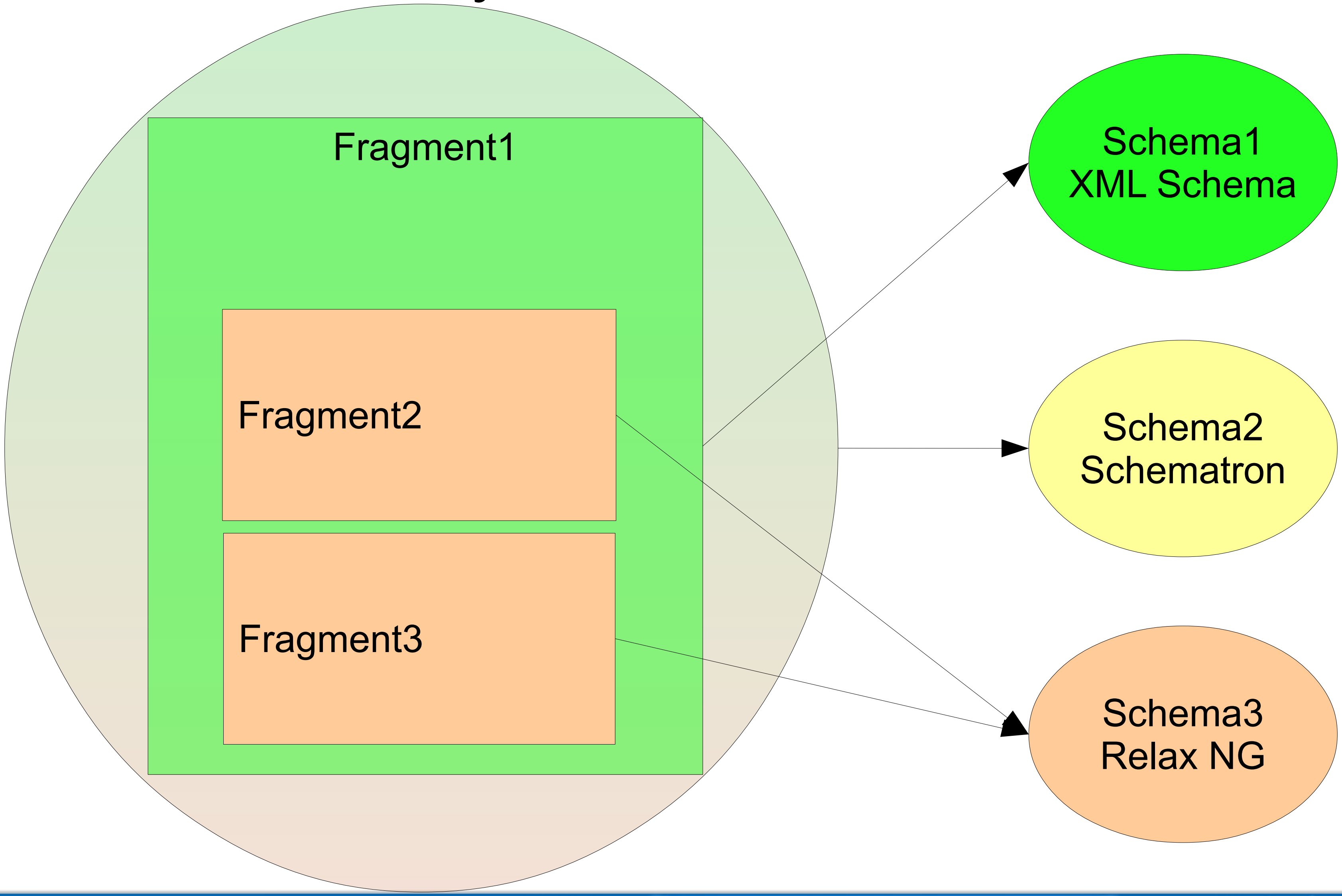
Validate against different schema types



Copyright @ Syncro Soft, 2012. All rights reserved.

Problem

Any combination of the above



Copyright @ Syncro Soft, 2012. All rights reserved.

NVDL

Provides a declarative language to easily specify:

- how the document is split into fragments (dispatching)
- how each fragment is validated (validation)

Copyright @ Syncro Soft, 2012. All rights reserved.

NVDL Processing

- Dispatching
- Validation

Copyright @ Syncro Soft, 2012. All rights reserved.

NVDL Dispatching

- Split the document into element and attribute sections
- Specify how these sections are combined to form the document fragments to be validated

Copyright @ Syncro Soft, 2012. All rights reserved.

NVDL Validation

Take each document fragment and perform the specified validate action

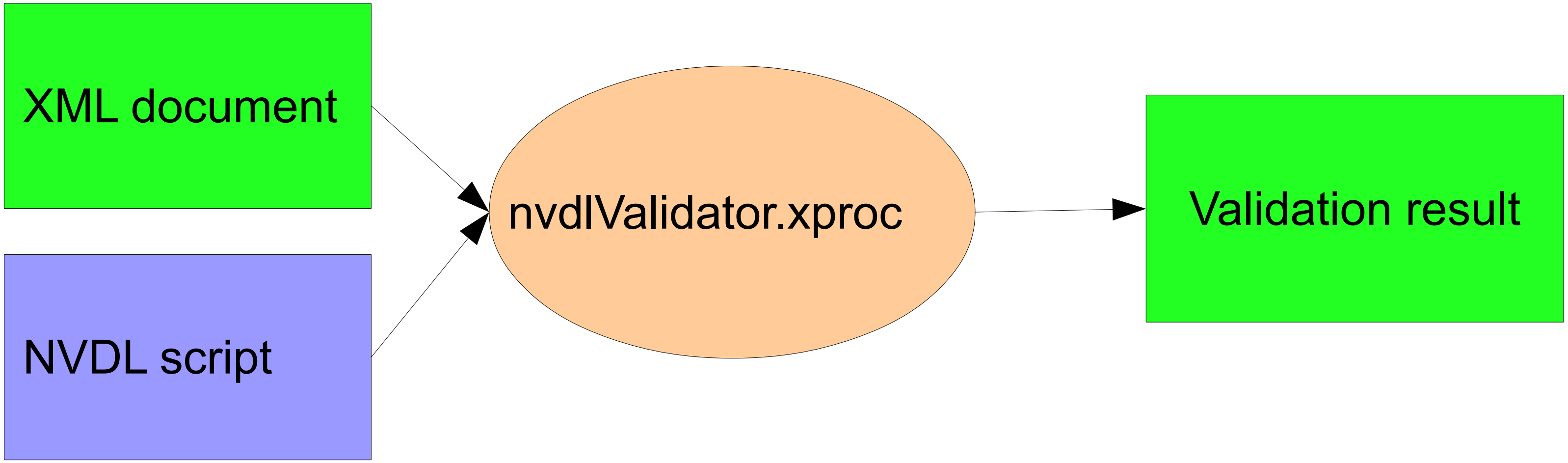
Copyright @ Syncro Soft, 2012. All rights reserved.

XSLT/XProc NVDL Implementation

- Inspired by Schematron skeleton
- XProc is a perfect choice for orchestrating:
  - XSLT transformations
  - Multiple validations
  - Other XML processing tasks

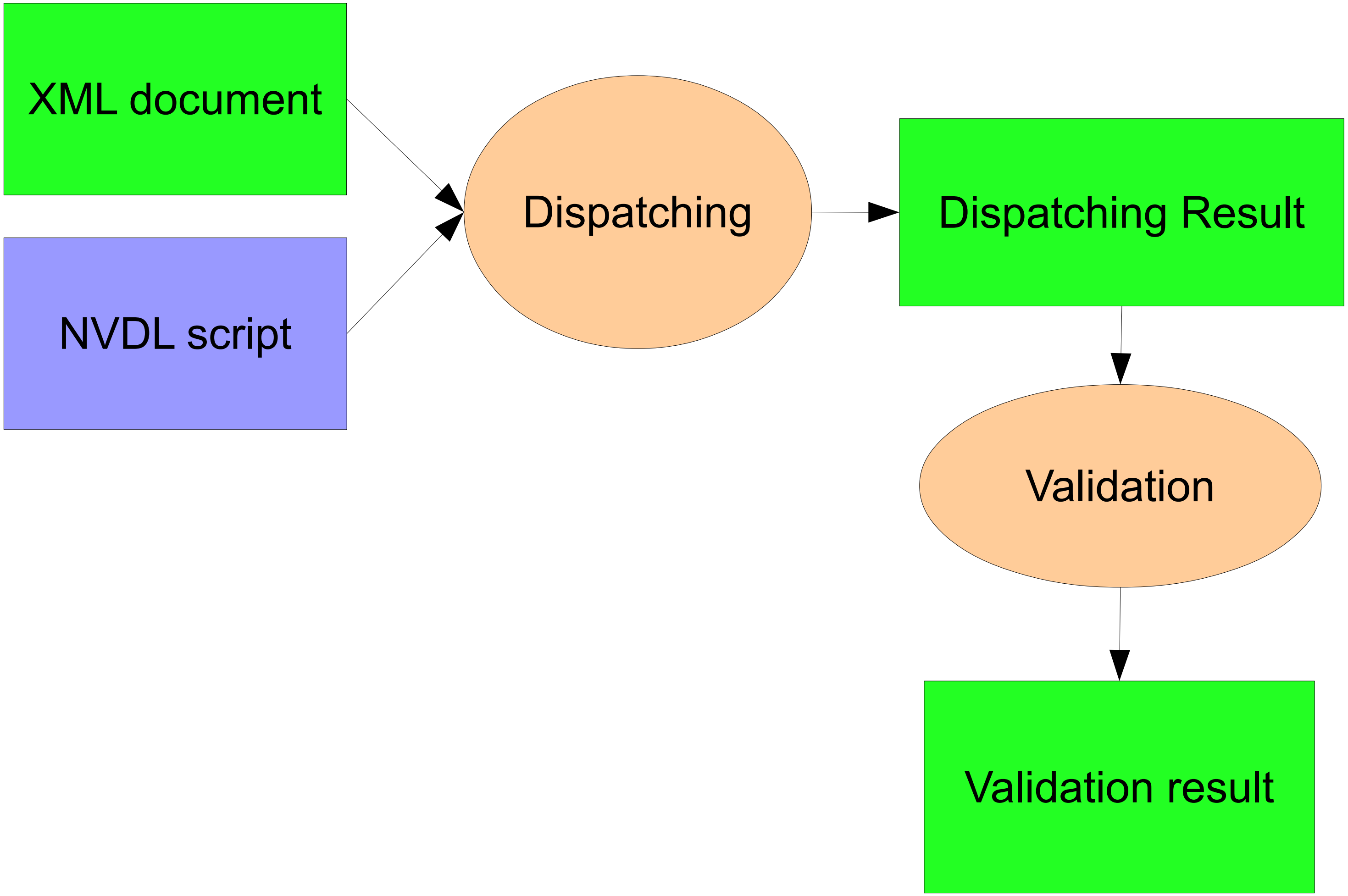
Copyright @ Syncro Soft, 2012. All rights reserved.

Processing Workflow



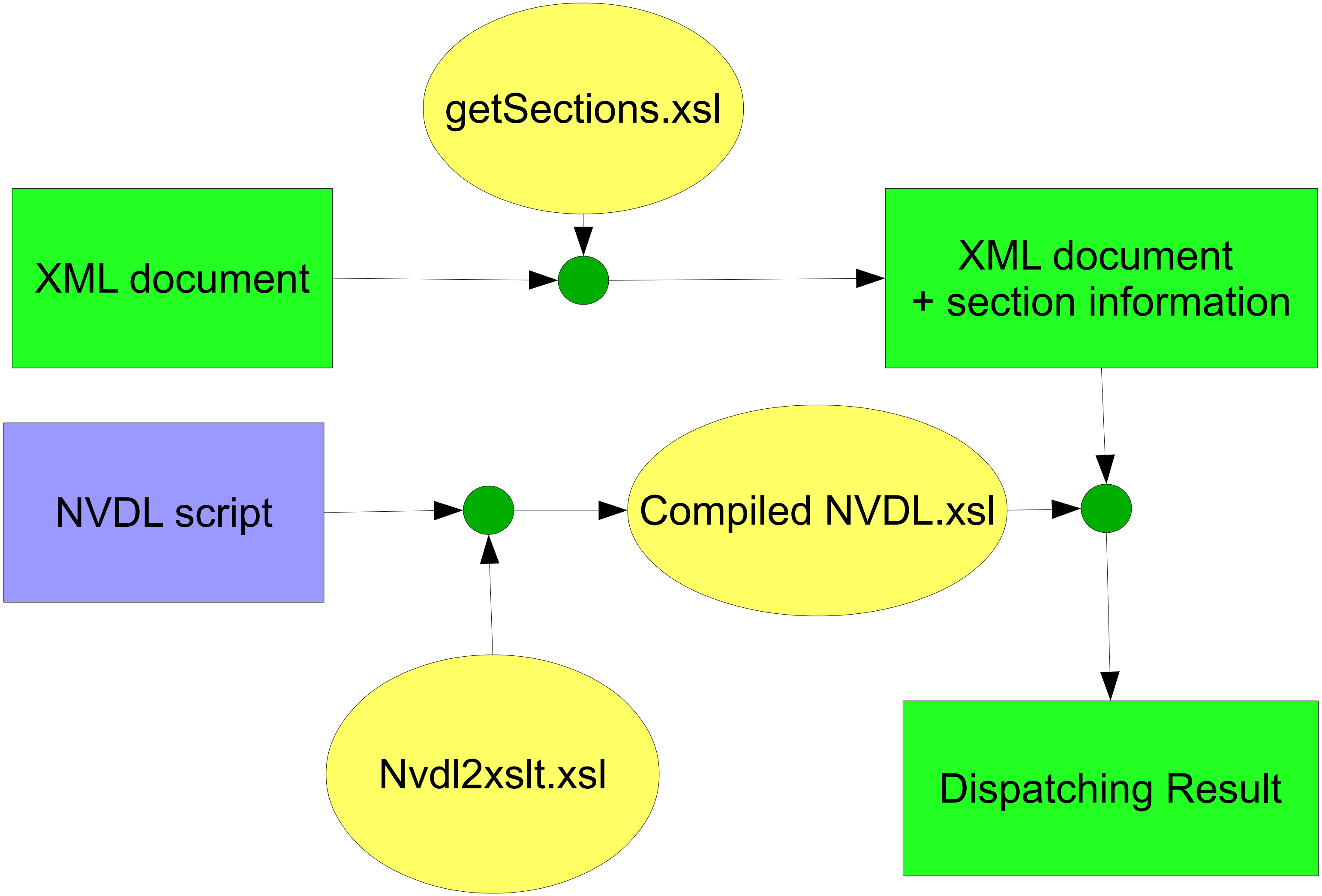
Copyright @ Syncro Soft, 2012. All rights reserved.

Processing Workflow



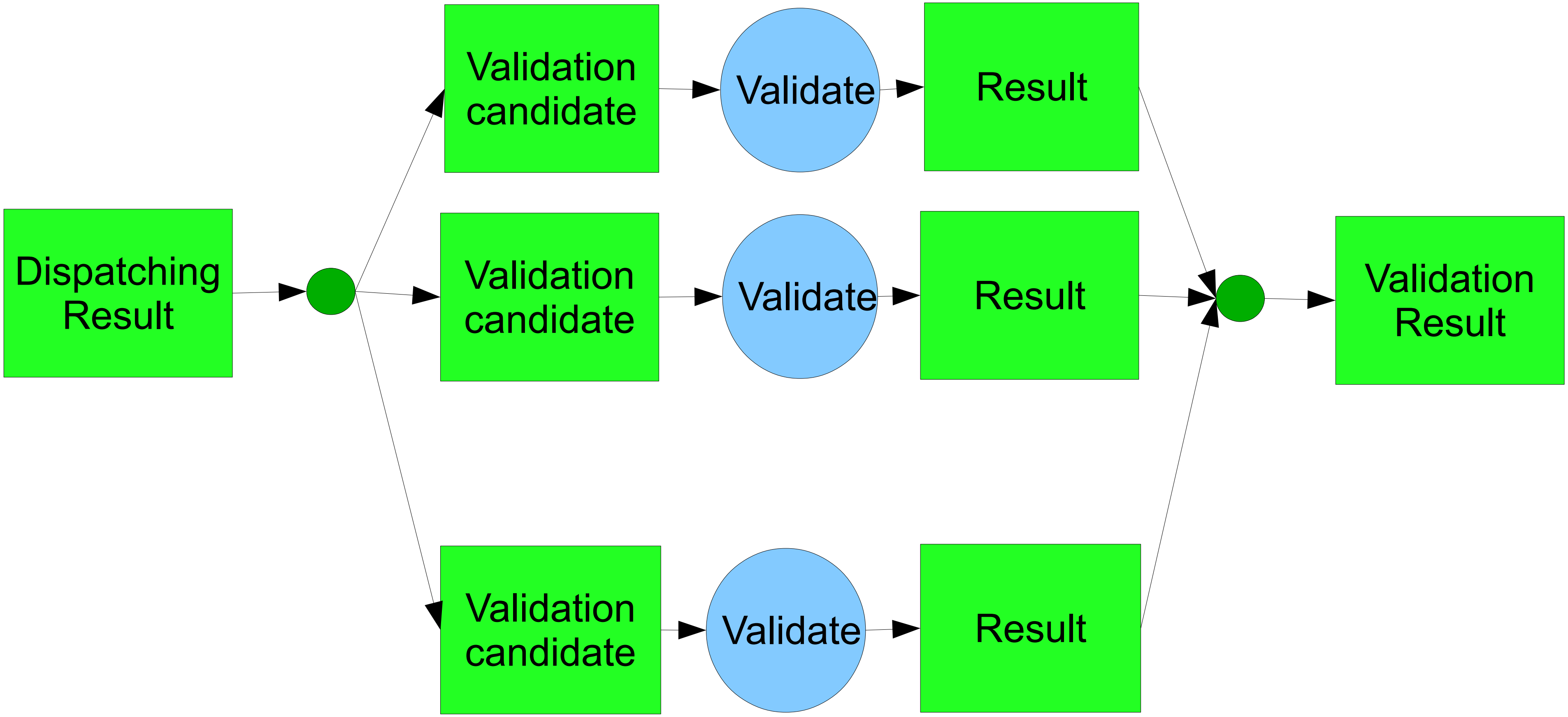
Copyright @ Syncro Soft, 2012. All rights reserved.

Dispatching



Copyright @ Syncro Soft, 2012. All rights reserved.

Validation



Copyright @ Syncro Soft, 2012. All rights reserved.

NVDL and XSLT

NVDL and XSLT

• Element sections	• Elements
• Attribute sections	• Attributes
• Modes	• Modes
• Rules	• Templates
• Built-in rules	• Built-in templates
• Actions	• Instructions

Sections

- Continuous content from the same namespace
- Element sections
  - The parent is in the same namespace
  - Can be split by triggers that specify the start of a section (see XHTML + XForms demo)
- Attribute sections
  - Group attributes from the same namespace

Copyright @ Syncro Soft, 2012. All rights reserved.

Encoding Sections in XML

```
<env:Envelope xmlns="http://www.w3.org/1999/xhtml"
  xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Body>
    <html>
      <head>
        <title>Document 1</title>
      </head>
      <body>
        <h1></h1>
        <p>...</p>
      </body>
    </html>

    <html>
      <head>
        <title>Document 2</title>
      </head>
      <body>
        <p>...</p>
      </body>
    </html>
  </env:Body>
</env:Envelope>
```

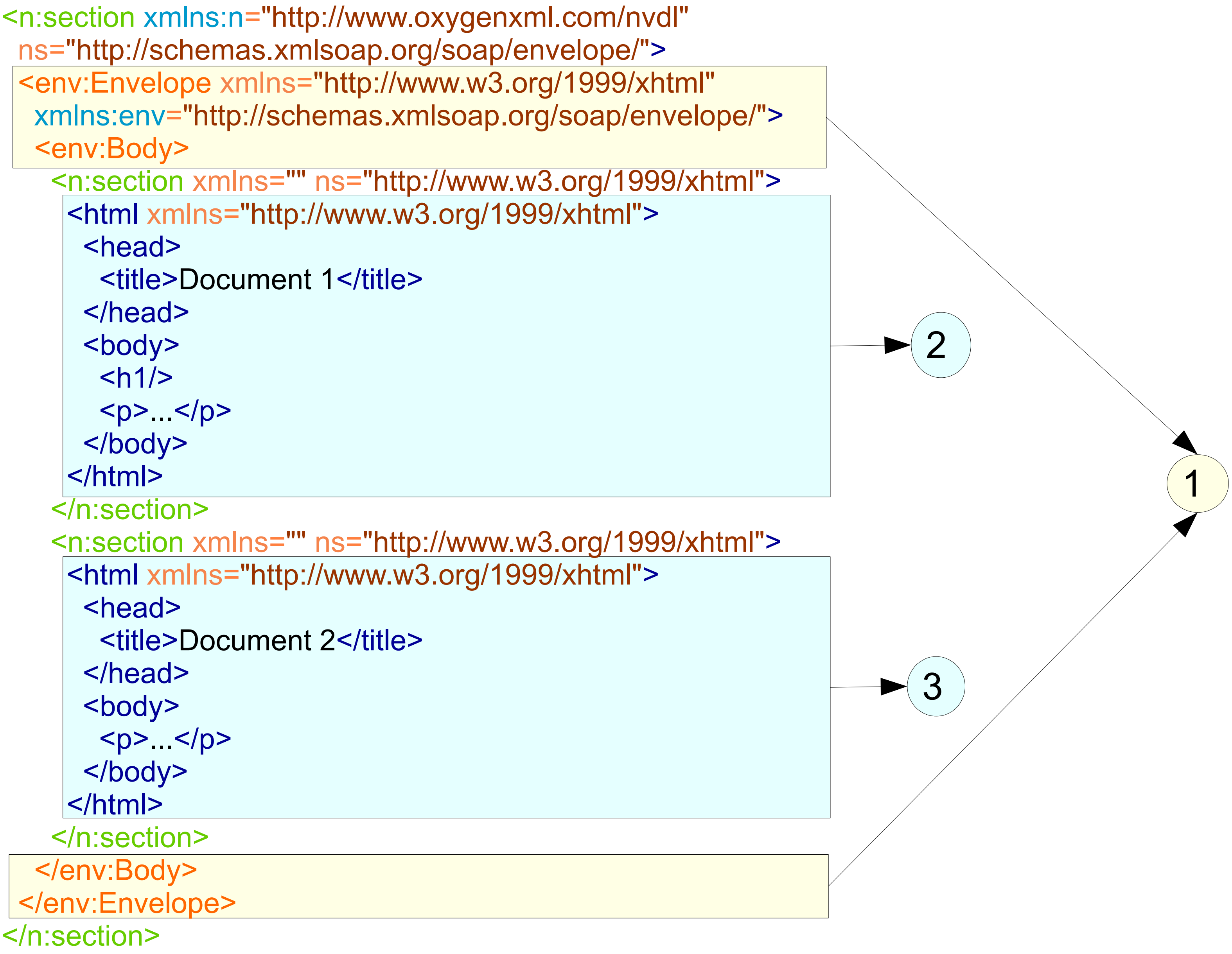
Copyright @ Syncro Soft, 2012. All rights reserved.

Encoding Sections in XML

```
<n:section xmlns:n="http://www.oxygenxml.com/nvdl"
 ns="http://schemas.xmlsoap.org/soap/envelope/">
 <env:Envelope xmlns="http://www.w3.org/1999/xhtml"
  xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
  <env:Body>
   <n:section xmlns="" ns="http://www.w3.org/1999/xhtml">
    <html xmlns="http://www.w3.org/1999/xhtml">
     <head>
      <title>Document 1</title>
     </head>
     <body>
      <h1/>
      <p>...</p>
     </body>
    </html>
   </n:section>
   <n:section xmlns="" ns="http://www.w3.org/1999/xhtml">
    <html xmlns="http://www.w3.org/1999/xhtml">
     <head>
      <title>Document 2</title>
     </head>
     <body>
      <p>...</p>
     </body>
    </html>
   </n:section>
  </env:Body>
 </env:Envelope>
</n:section>
```

Copyright @ Syncro Soft, 2012. All rights reserved.

Encoding Sections in XML



```
<n:section xmlns:n="http://www.oxygenxml.com/nvdl"
  ns="http://schemas.xmlsoap.org/soap/envelope/">
 <env:Envelope xmlns="http://www.w3.org/1999/xhtml"
   xmlns:env="http://schemas.xmlsoap.org/soap/envelope/">
   <env:Body>
     <n:section xmlns="" ns="http://www.w3.org/1999/xhtml">
      <html xmlns="http://www.w3.org/1999/xhtml">
        <head>
          <title>Document 1</title>
        </head>
        <body>
          <h1/>
          <p>...</p>
        </body>
      </html>
     </n:section>
     <n:section xmlns="" ns="http://www.w3.org/1999/xhtml">
      <html xmlns="http://www.w3.org/1999/xhtml">
        <head>
          <title>Document 2</title>
        </head>
        <body>
          <p>...</p>
        </body>
      </html>
     </n:section>
   </env:Body>
 </env:Envelope>
</n:section>
```

Copyright @ Syncro Soft, 2012. All rights reserved.

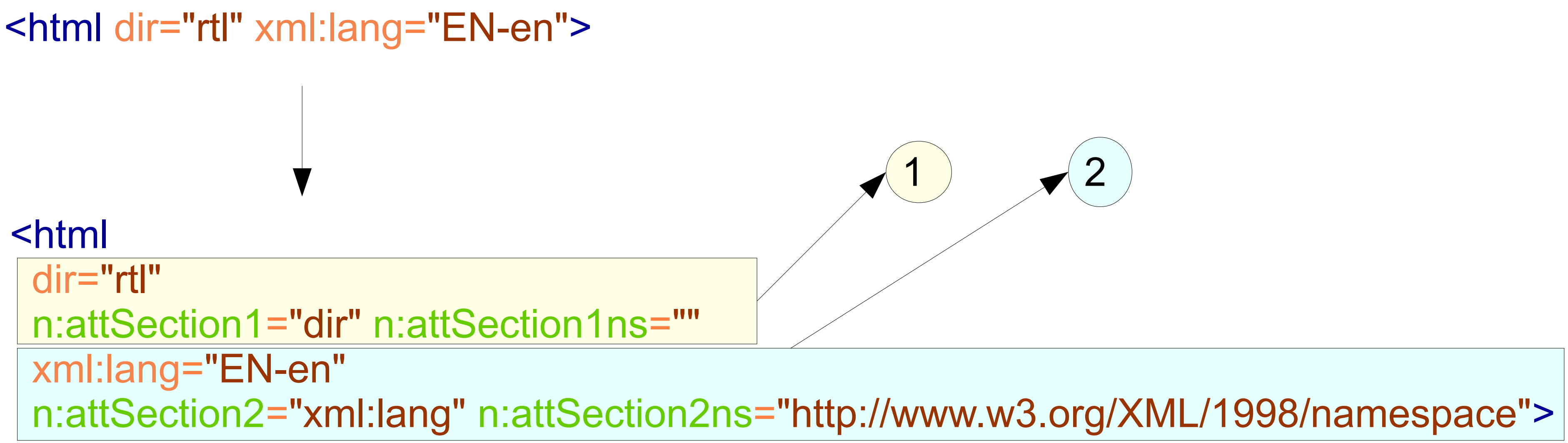
Encoding Sections in XML

```
<html dir="rtl" xml:lang="EN-en">
```

```
<html
  dir="rtl"
  n:attSection1="dir" n:attSection1ns=""
  xml:lang="EN-en"
  n:attSection2="xml:lang" n:attSection2ns="http://www.w3.org/XML/1998/namespace">
```

Copyright @ Syncro Soft, 2012. All rights reserved.

Encoding Sections in XML



```
<html dir="rtl" xml:lang="EN-en">
```

```
<html
 dir="rtl"
 n:attSection1="dir" n:attSection1ns=""
 xml:lang="EN-en"
 n:attSection2="xml:lang" n:attSection2ns="http://www.w3.org/XML/1998/namespace">
```

NVDL Structure

- Trigger*
- Mode+
  - Rule*
    - Action* (use-mode?)
      - Context? (use-mode?)

Triggers

Split further sections if one of the trigger elements is encountered inside a section

```
<trigger
  ns="http://www.w3.org/1999/xhtml"
  nameList="head body"/>
```

Copyright @ Syncro Soft, 2012. All rights reserved.

Triggers

```
<n:section xmlns:n="http://www.oxygenxml.com/nvdl" ns="http://www.w3.org/1999/xhtml">
  <html xmlns="http://www.w3.org/1999/xhtml" xmlns:xforms="http://www.w3.org/2002/xforms">
      <head xmlns="http://www.w3.org/1999/xhtml">
        <title>Sample</title>
      </head>
      <body xmlns="http://www.w3.org/1999/xhtml">
        <h1>XForms sample</h1>
        <p>Input</p>
        <p>Submit:</p>
      </body>
  </html>
</n:section>
```

+

```
<trigger
  ns="http://www.w3.org/1999/xhtml"
  nameList="head body"/>
```

Copyright @ Syncro Soft, 2012. All rights reserved.

Triggers

```
<n:section xmlns:n="http://www.oxygenxml.com/nvdl" ns="http://www.w3.org/1999/xhtml">
 <html xmlns="http://www.w3.org/1999/xhtml" xmlns:xforms="http://www.w3.org/2002/xforms">
  <n:section xmlns="" ns="http://www.w3.org/1999/xhtml">
   <head xmlns="http://www.w3.org/1999/xhtml">
    <title>Sample</title>
   </head>
  </n:section>
  <n:section xmlns="" ns="http://www.w3.org/1999/xhtml">
   <body xmlns="http://www.w3.org/1999/xhtml">
    <h1>XForms sample</h1>
    <p>Input</p>
    <p>Submit:</p>
   </body>
  </n:section>
 </html>
</n:section>
```

Copyright @ Syncro Soft, 2012. All rights reserved.

Modes

- Group a set of rules

  Similar with the XSLT modes that group a set of templates

- Can be named (and referred)
- If no mode is explicitly defined there is an implicit default mode

Copyright @ Syncro Soft, 2012. All rights reserved.

Rules

Equivalent with XSLT templates

- Namespace rule

```
<namespace ns="http://docbook.org/ns/docbook">
  ...actions...
</namespace>
```

- anyNamespace rule

```
<anyNamespace>
  ...actions...
</anyNamespace>
```

Copyright @ Syncro Soft, 2012. All rights reserved.

Rules

- Match
  - Element sections (default)
  - Attribute sections
  - Element and attribute sections
- Support wildcards for namespace specification

Copyright @ Syncro Soft, 2012. All rights reserved.

Actions

- Result actions
  - attach (equiv. with xsl:copy + xsl:apply-templates)
  - unwrap (ignore – equiv. with xsl:apply-templates)
  - attachPlaceholder
- No-result actions
  - validate (starts a fragment – like xsl:variable)
  - allow (validate with a schema that allows anything)
  - reject (validate with a schema that rejects anything)
- Cancel action
  - cancelNestedActions

Copyright @ Syncro Soft, 2012. All rights reserved.

Use-mode

- Specifies how descendant sections are processed (what set of rules is applied)
- If not specified the same mode is used
- Can refer to a mode or specify a local mode

```
<validate schema="http://www.w3.org/1999/xhtml/xhtml.rng"
       useMode="allXHTML"/>

<validate schema="http://www.oasis-open.org/docbook/xml/5.0/rng/docbook.rng">
  <mode>
    <namespace ns="http://docbook.org/ns/docbook"><attach/></namespace>
    <anyNamespace><unwrap/></anyNamespace>
  </mode>
</validate>
```

Copyright @ Syncro Soft, 2012. All rights reserved.

Context

If a section appears in a specific place then we can process that using a specified mode

```
<mode name="original">
    <namespace ns="http://www.oxygenxml.com/ns/sample/versions">
      <unwrap useMode="original">
        <context path="added" useMode="ignore"/>
      </unwrap>
    </namespace>
    <anyNamespace>
      <attach/>
    </anyNamespace>
  </mode>
```

Copyright @ Syncro Soft, 2012. All rights reserved.

Built-in Rules

```
<anyNamespace match="elements">
      <reject/>
</anyNamespace>

<anyNamespace match="attributes">
      <attach/>
</anyNamespace>
```

Demo – Envelope + Content

Used for web services

The envelope should be validated with the envelope schema and the content with a schema for content

The schemas may use different schema languages

Copyright @ Syncro Soft, 2012. All rights reserved.

Demo – XHTML + XForms

- All XHTML content is extracted and validated
- The Xforms content is placed inside the root XHTML element and validated with a wrapper schema

  This is needed due to ID references from the XForms content, all XForms fragments need to be validated in a single validate actions

Copyright @ Syncro Soft, 2012. All rights reserved.

Change Tracking - Versions

- Mark added and deleted content inside a DocBook document
- Validate both the original and final documents, ignoring additions in the first case and ignoring content marked as removed in the second case

Copyright @ Syncro Soft, 2012. All rights reserved.

References – Spec and Tutorials

- The NVDL specification

  http://bit.ly/nvdlSpec

- Namespace Routing Language (NRL)

  http://www.thaiopensource.com/relaxng/nrl.html

- NVDL Tutorials

  http://www.xfront.com/nvdl/index.html

  http://www.dpawson.co.uk/nvdl/

  http://jnvdl.sourceforge.net/tutorial.html

Copyright @ Syncro Soft, 2012. All rights reserved.

References - Implementations

- oNVDL – Initial implementation of NVDL for Jing + this XSLT&XProc implementation

  http://sourceforge.net/projects/onvdl/
- Jing – the NVDL implementation from oNVDL is not part of Jing

  http://code.google.com/p/jing-trang/
- JNVDL – another NVDL implementation in Java

  http://sourceforge.net/projects/jnvdl/

Copyright @ Syncro Soft, 2012. All rights reserved.

Conclusions

- NVDL is simple
  - element and attribute sections
  - triggers, modes, rules, actions
- Keep in mind the analogy with XSLT to easily understand it
- It is available anywhere - it can be implemented with XSLT + (XProc or any other script that can orchestrate XSLT transformations and validations)

Copyright @ Syncro Soft, 2012. All rights reserved.

Thank You!

Questions?

<oXygen/> XML Editor
http://www.oxygenxml.com
george@oxygenxml.com
@georgebina

Copyright @ Syncro Soft, 2012. All rights reserved.