



Elkera Pty Limited

Elkera XML Print

Setting up an XML rendering application using Elkera XML Print

3 February 2006

Andrew Squire – Senior Consultant & Peter Meyer – Managing director, Elkera Pty Limited

Elkera Pty Limited ABN 68 092 447 428
Suite 701, 10 Help Street, Chatswood, NSW 2067, Australia
PO Box 5280, West Chatswood, NSW 1515
Telephone +61 2 8440 6999
Facsimile +61 2 8440 6988
www.elkera.com

Contents

Scope of this session.....	1
Background to Elkera XML Print.....	2
Common uses.....	2
Some distinguishing features of Elkera XML Print.....	2
Steps in style sheet development life cycle.....	3
Output requirements specification.....	3
Steps to prepare output requirements specification.....	3
Why the output requirements specification is important.....	3
Style sheet development in Elkera XML Print.....	3
Context rules and formatting rules.....	3
Two classes of layout properties in a rendering application.....	4
Typical style sheet development steps.....	4
Style sheet development approaches with Elkera XML Print.....	4
Conclusion – The benefits of Elkera XML Print.....	5

Setting up an XML rendering application using Elker XML Print

Scope of this session

The session will describe the operation and configuration of Elker XML Print, Elker's XML to RTF converter and document printing workflow application.

Elker XML Print is an industrial strength XML to RTF converter and print workflow application. It can produce RTF with or without named Word styles. When named styles are used, it can produce RTF documents with automatic numbering ready to be edited as a native Word document by non XML authors. Since the outputs from Elker XML Print can be so different, style sheet developers may need to use different development strategies according to the features required in the RTF.

When styling is created without named styles, Elker XML Print allows style properties to be inherited from ancestor XML contexts to simplify development effort. If style information has to be defined in the style sheet by the developer, the use of styles can add a lot of complexity to the styling because a separate style must be defined for every context with distinct layout properties.

A common problem in developing XML document publishing applications is communication of layout requirements to style sheet developers. Normally, style sheets in most rendering applications must be developed by programmers. However, programmers may not be document layout designers. How does the style sheet developer acquire the information needed to develop the style sheets?

Elker XML Print can import style definitions directly from Word templates. This has several important advantages:

- It simplifies the work involved in developing applications that generate RTF with named styles.
- It enables non programmers to maintain and adapt style sheets in response to enterprise needs.
- It also means that a Word document can be used as the basis of an outputs specification and an ongoing reference for style sheet development.

The session will discuss the key steps in planning a document publishing application from XML data and show how Elker XML Print enables designers to use Microsoft Word as a visual document layout designer and to then import document styles from a Word document into the style rules in Elker XML Print. It will show how Elker XML Print can provide high quality printed output and minimize application development and maintenance costs.

Background to Elkera XML Print

Common uses

Elkera XML Print can be used in most situations where XML documents can be published using the layout capabilities of Microsoft Word.

Particular applications include:

- Where authors of XML documents are responsible for published outputs in print or PDF and need to be able to easily set printing options at run time.
- Where XML content must be exchanged with non XML authors for collaboration and review.
- When database reports must be edited by authors or copied into Word documents.
- Where documents are generated from precedents or templates in XML based document assembly systems and require further editing by authors using word processing software.
- Where output documents require formatting adjustments that cannot be added to style sheets within production deadlines.

Some distinguishing features of Elkera XML Print

- Elkera XML Print uses its own style sheet language. The language is closely aligned with document publishing concepts to reduce the effort needed to become proficient with the application. Concepts found in XSLT and XSL-FO are quite abstract and require a steep learning curve to master.
- It provides an easily configured user interface that allows users to have extensive run time control over document publishing options without the need to set persistent values in the XML data. Through the user interface, users can control an extensive range of production features in print, RTF and PDF outputs.
- Styles properties can be imported from a Word document or template. This has two benefits. It provides the foundation for the outputs specification and it reduces the need for programmers to maintain the application after development. These features reduce cost and enable greater responsiveness to enterprise publishing needs.
- Style properties are extracted from the Word document as an XML file. This file can be formatted and included as part of the output requirements specification.

Steps in style sheet development life cycle

- Prepare the output requirements specification.
- Develop application style sheets.
- Test generated output for conformity with requirements.
- Adjust style sheets as layout requirements change.

Output requirements specification

Steps to prepare output requirements specification

- Analyse sample output documents, identify page components and agree on a common vocabulary for components.
- Analyse the DTD or Schema and identify of important contexts and how these relate to the page components.
- Specify layout of schema elements and other page components.

Why the output requirements specification is important

- A specification enables a common understanding of the output requirements between document designers and application programmers.
- It promotes a more complete analysis at an earlier stage, thus producing a more reliable application.
- It provides a reference for testing application outputs.
- It provides an ongoing reference for future maintenance developers, avoiding the creation of "Guru" systems.

Style sheet development in Elkera XML Print

Context rules and formatting rules

Context rules define the element and attribute contexts that are significant to the document layout. Once defined, context rules tend to be fixed unless the DTD or schema is changed or if there is a major re-design of the document layout.

Formatting rules define the layout properties that are to be applied to particular contexts. These may involve conditional logic.

The Elkera XML Print style language allows context rules and formatting rules to be separated. This can simplify application maintenance.

Two classes of layout properties in a rendering application

While not a technical distinction, it can be useful to divide layout properties into two classes, based on the extent to which they are likely to change during the application lifetime:

- Persistent layout properties are those which implement the basic structure of the Schema or DTD. This may include line breaks, paragraph breaks, column and table layout. These rarely change as part of normal application maintenance.
- Variable layout properties are those such as font face and font size, indents and other spacing. These may be implemented differently in related applications and the may change over time as enterprise needs evolve.

Typical style sheet development steps

- Configure page size, margins and other document wide properties.
- Configure default settings for basic components such as paragraphs and text.
- Develop context rules for the main document structures defined by the DTD or schema and the persistent layout rules for those structures.
- Develop layout rules for variable properties.
- Develop other page objects such as cover, headers and footers, etc.
- Set up run-time print options.

Style sheet development approaches with Elkera XML Print

Elkera XML Print allows styles sheets to be designed in different ways, depending on the desired functionality in the RTF and requirements for application maintenance:

- Fully coded style sheets. With this approach context rules are defined and mapped to formatting rules that are defined in the code. This approach may be used if named styles and automatic numbering is not required in the application. Context inheritance can be used to create layout properties when named styles are not required. This makes coding easier than if a full set of named styles has to be defined. However, it is the least flexible for ongoing application maintenance. This is suitable for simpler applications that are maintained by the developer.
- Coded style sheets with application variables for variable layout properties. This is more flexible than the code only option. Variables can be separated and non-programmers with some knowledge can adjust properties by changing the values of variables.

- Code with style properties extracted from Word templates. Under this approach, context rules are defined in the code and linked to imported style properties. This approach can be used even if named styles are not required in the RTF because styles can be edited by non-programmers using MS Word.

Generally, it is not desirable to mix all 3 approaches. The use of variables limits the ability to import style definitions.

Conclusion – The benefits of Elkera XML Print

- Elkera XML Print is very flexible and allows the style sheet development approach to match to the desired outputs and maintenance needs. It produces a spectrum of outputs unmatched by most alternative applications.
- Elkera XML Print encourages a good style sheet architecture that matches the layout design process. This enhances application maintainability and reduces costs.
- Elkera XML Print can extract styles from Word templates. This supports development of the outputs specification and allows non-programmers to do some of the style sheet development and maintenance. This reduces the cost of application development and minimizes delays in style sheet adjustments.
- Elkera XML Print provides a customizable graphical user interface with the ability to pass run-time print options to the application. This eliminates the need for command line options and schema changes to embed options in the XML.