The Extensible Stylesheet Language Transformations APIs

Figure 4-3 shows the XSLT APIs in action.

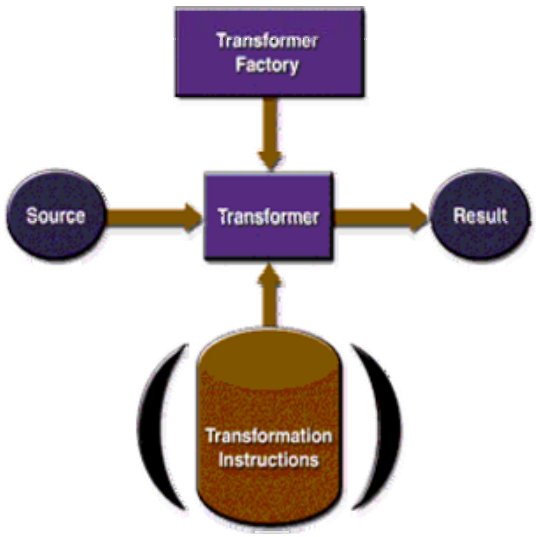

Figure 4-3 XSLT APIs

A `TransformerFactory` object is instantiated and used to create a `Transformer`. The source object is the input to the transformation process. A source object can be created from a SAX reader, from a DOM, or from an input stream.

Similarly, the result object is the result of the transformation process. That object can be a SAX event handler, a DOM, or an output stream.

When the transformer is created, it can be created from a set of transformation instructions, in which case the specified transformations are carried out. If it is created without any specific instructions, then the transformer object simply copies the source to the result.

The XSLT Packages

The XSLT APIs are defined in the packages shown in Table 4-3.

Table 4-3 XSLT Packages

Package	Description
`javax.xml.transform`	Defines the `TransformerFactory` and `Transformer` classes, which you use to get an object capable of doing transformations. After creating a transformer object, you invoke its `transform()` method, providing it with an input (source) and output (result).
`javax.xml.transform.dom`	Classes to create input (source) and output (result) objects from a DOM.
`javax.xml.transform.sax`	Classes to create input (source) objects from a SAX parser and output (result) objects from a SAX event handler.
`javax.xml.transform.stream`	Classes to create input (source) objects and output (result) objects from an I/O stream.