**Comparing Three different Parsers:**

Dom (Document Object Model) Parser

The first parser that we explore is the DOM Parser:

DOM - Parses the document by loading the complete contents of the document and creating its complete hierarchical tree in memory. (The DOM tree)

The whole XML will be loaded into memory

Loads the XML content into a Tree structure

Iterate through the Node and NodeList to get the content of the XML.

***Basics***:

* Memory intensive
* Read-write
* Typically used for working with documents smaller than 10 megabytes

**Advantages**

you can navigate/read to any node

you can append, delete or update a child node because data is available in the memory.

**Disadvantages**

if the XML contains a large data, then it will be very expensive memory wise

the whole XML is loaded to memory although you are looking for something particular

SAX (Simple API for XML)Parser

SAX Parses the document on event-based triggers.

Does not load the complete document into the memory.

SAX parser doesn’t load the complete XML into the memory!

It parses the XML line by line

Triggers different events as and when it encounters different elements like: opening tag, closing tag, character data

you can add your event handler depending on your requirements.

so, you register a handler (or more than one handler) and those handlers are called when an event occurs.

***Basics***:

* Memory efficient
* Read-only
* Typically used for working with documents larger than 10 megabytes

**Advantages**

you do not need to worry on memory Consumptions

if the performance is the criteria, (and if you are only reading the xml), SAX is a much better choice then DOM.

**Disadvantages**

you are not going to have a tree structure where you can require parent or child elements

StAX (Streaming API for XML) Parser

StAX Parses the document in similar fashion to SAX parser but in more efficient way.

StAX does not store anything to memory

StAX looks like parsing with SAX

in SAX, your event handler is called by SAX when an event occurs.

In StAX, you ask StAX to continue to next event

SAX parser issues events as and when certain data is encountered

the “cursor model” in StAX: next event can be required by parser.next()

XMLInputFactory, XMLStreamReader classes are used to load an XML file

We read through the XML file using XMLStreamReader

Events are generated in the form of integer values

These are then compared with the constants in XMLStreamConstants

***Basics***:

* Memory efficient
* Read-write
* Appropriate for documents of all sizes
* Easier to use than DOM and SAX