

Handling very large XML documents in an editing application

Presenter:

Radu Coravu
radu_coravu@oxygenxml.com
@radu_coravu



Bytes and characters

- The byte is a unit of digital information that most commonly consists of eight bits. Used for storage and low-level communication between computers.

<https://en.wikipedia.org/wiki/Byte>

- A character is a sign or symbol. Like the alphabet symbols. End users interact with application by reading and editing characters.

<https://en.wikipedia.org/wiki/Character>

Character Encoding

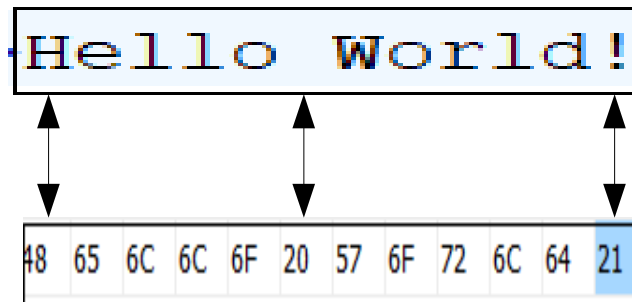
Used to translate backwards and forwards between:

- Bytes on disk (numeric values 0-255).
- Characters in application (usually stored on two bytes).

https://en.wikipedia.org/wiki/Character_encoding

ASCII Encoding

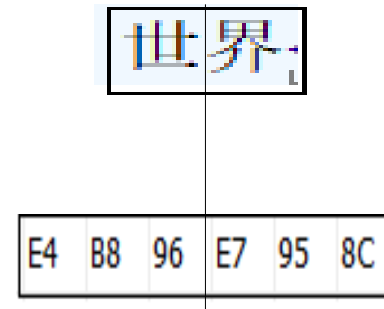
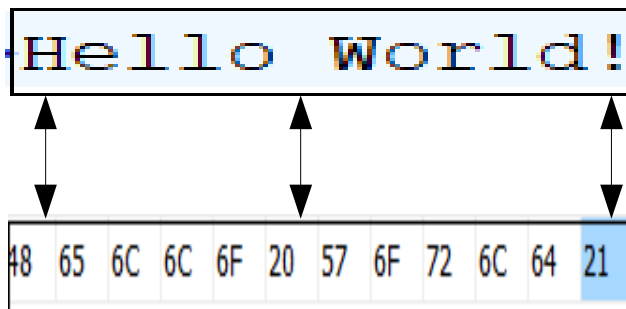
- Encodes about 127 different characters.
- One to one correspondence between bytes and characters.



<https://en.wikipedia.org/wiki/ASCII>

UTF-8 Encoding

- Capable of encoding all 1,112,064 valid code points in Unicode.
- Variable width encoding.
- Many to one correspondence between bytes and characters.



<https://en.wikipedia.org/wiki/UTF-8>

UTF-16 Encoding

- Capable of encoding all valid code points in Unicode.
- Usually two to one correspondence between bytes and characters.



<https://en.wikipedia.org/wiki/UTF-16>

Detecting the encoding of an XML document

<https://www.w3.org/TR/xml/#sec-guessing>

- Look at Xml encoding declaration in header:

```
<?xml version="1.0" encoding="UTF-8"?>
```

- Fallback to UTF-8 or UTF-16 (default XML encoding is UTF-8).
- The encoding can be detected without reading the entire file contents.

What encoding should I use for my XML documents?

- UTF-8 “Hello” => 5 bytes on disk.
- UTF-16 “Hello” => 10 bytes on disk.
- UTF-8 “ 世界 ” => 6 bytes on disk.
- UTF-16 “ 世界 ” => 4 bytes on disk.

What encoding should I use for my XML documents?

- UTF-8 is default XML encoding and is supported by any processor.
- For non-European languages UTF-16 provides more compact file sizes.
- XML character entities allow us to save all ranges of characters to a restrictive encoding => larger file sizes.

XML file sizes

- Standard (0 bytes → 30 MBs on disk).
- Large (30 Mbs → 300 MBs).
- Huge (300 Mbs → Gygabytes).

Standard-size XML memory footprint

Memory footprints for each edit mode:

- **Text** $\approx 10 * \text{FILE_SIZE_ON_DISK}$
- **Grid** $\approx 9 * \text{FILE_SIZE_ON_DISK}$
- **Author** $\approx 20 * \text{FILE_SIZE_ON_DISK}$

Text Page Memory Footprint

- Characters in memory: $2 * \text{FILE_SIZE}$
- Visual editing support.
- Outline.
- Syntax highlight: depends on number of lines.
- Automatic validation: Usually streaming.
- Format and indent: $5 * \text{FILE_SIZE}$
- Xpath: $5 * \text{FILE_SIZE}$

Large-size XML documents (30 Mbs – 300 MBs)

Usually generated as database export or dump.

- **Text** mode: `8 * FILE_SIZE_ON_DISK`
 - XML characters are stored in separate buffer file on disk.
 - Format and indent: Streaming.
 - Xpath: Streaming.

Huge-size XML editing problems

- Generated as database export or dump (most users are not interested in editing).
- Loading the entire XML content from disk in the application is not a viable option (not enough computer internal memory).

Huge-size XML editing (1)

Tools-> Large File Viewer.

- Takes very little memory for editing files up to 1GB.
 - No Outline.
 - No Syntax highlight.
 - No validation.
 - Copy/Paste and Find/Replace.
 - Allows scrolling through entire document.

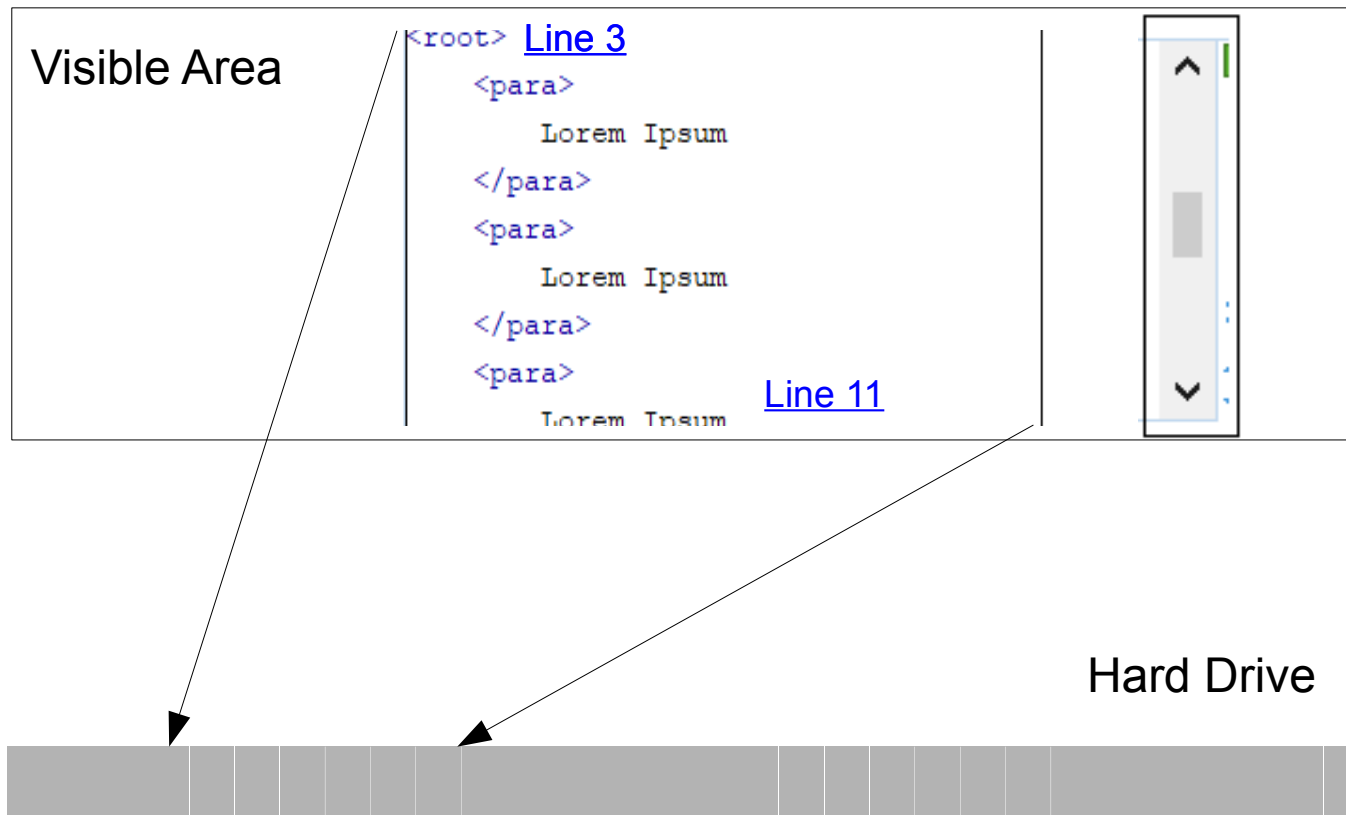
Huge-size XML editing (1)

Implementation details:

- Detects line ranges directly from file on disk.
- As vertical scrolling is done, content is directly drawn by looking in file on disk.

Huge-size XML editing (1)

Scrolling in Large File Viewer:

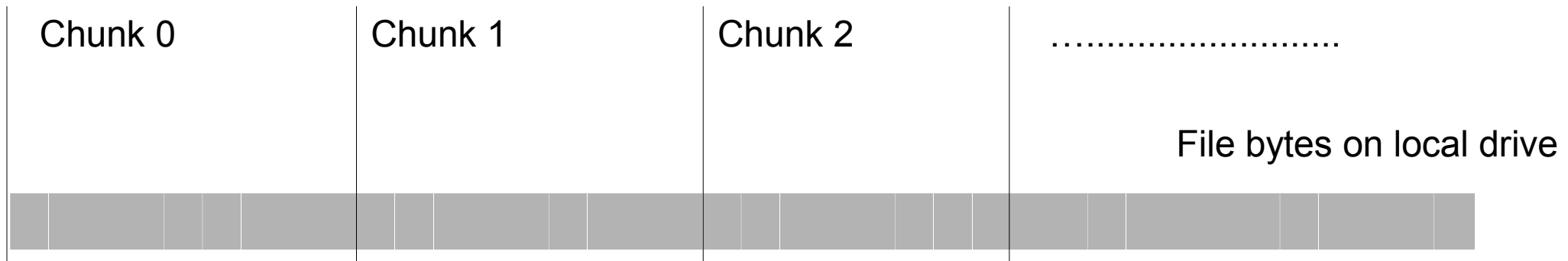


Huge-size XML editing – New solution (2)

- Instantly open any file of any size having encoding ASCII, UTF-8 or UTF-16.
- Edit/Save.
- Find/replace
- Split edited content in chunks/parts.
- Syntax highlight
- No outline.
- No automatic validation (manual validation works).
- No format and indent.

Huge-size XML editing

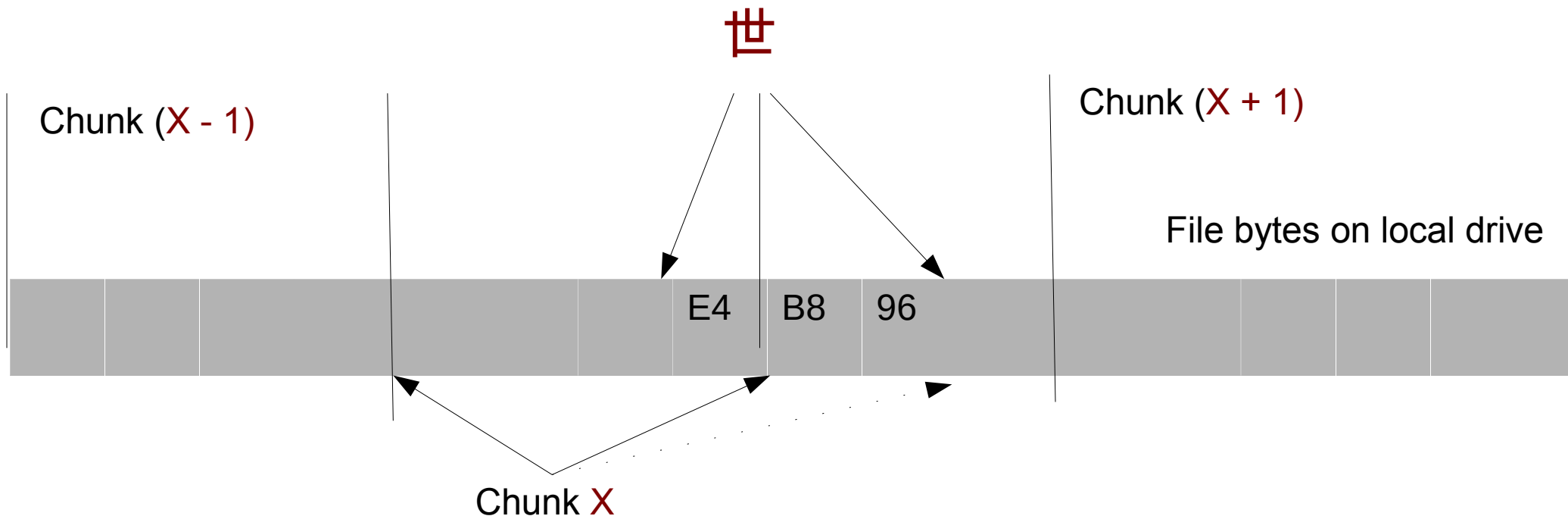
How does it work?



Chunk size = 1 million bytes

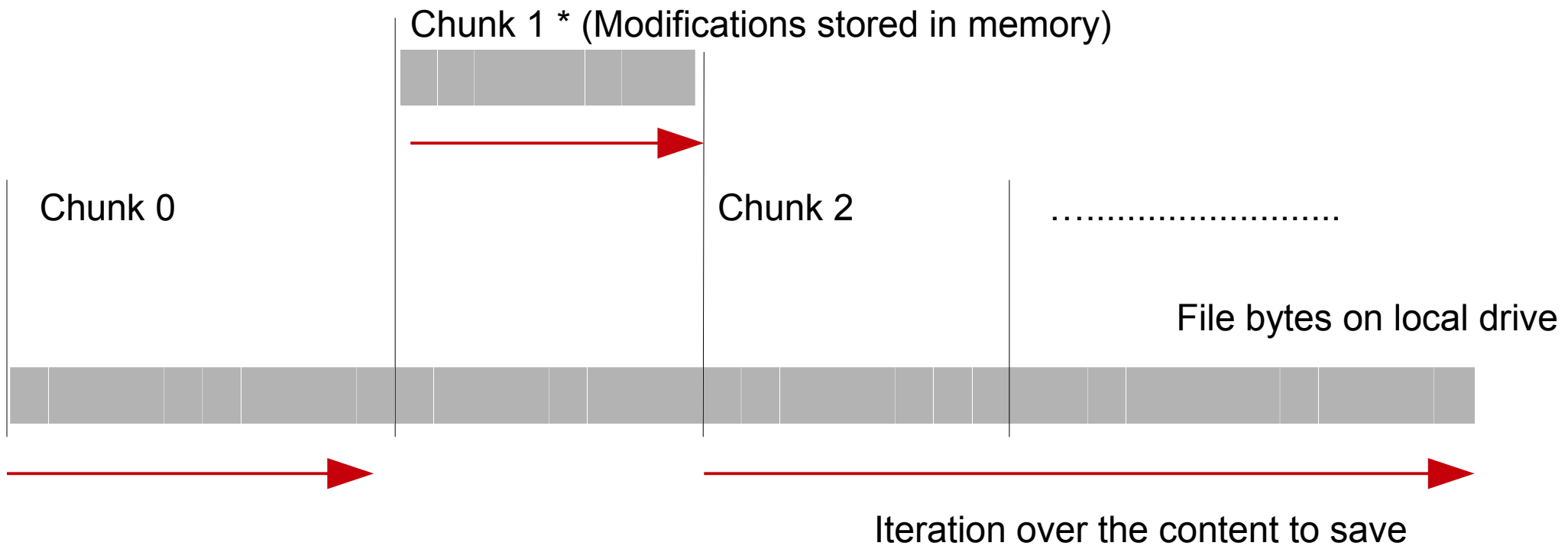
Huge-size XML editing

Repositioning chunks to include entire characters:



Huge-size XML editing

Editing content, searching and saving to disk:



Thank You!

Questions?

Radu Coravu
radu_coravu@oxygenxml.com
@radu_coravu