

## 2. SVGRoot::loadFromFile() (Tải và Chuẩn bị)

---

- Mở tệp "sample.svg" bằng ifstream.
- Đọc toàn bộ nội dung tệp vào một std::vector<char> (gọi là buffer).
- Thêm một ký tự \0 (null-terminator) vào cuối buffer.
- Tạo một đối tượng rapidxml::xml\_document<>doc;.
- Gọi doc.parse<...>(buffer[0]). Lúc này, RapidXML đã xây dựng xong một cây các xml\_node trong bộ nhớ.
- Tìm node gốc của tài liệu: xml\_node<>\* svgNode = doc.first\_node("svg");.

## 3. Phân tích Thẻ <svg> Gốc (trong loadFromFile)

---

- Hàm lặp qua các thuộc tính (attributes) chỉ của svgNode.
- Nó trích xuất width, height, viewBox và lưu vào các biến thành viên của SVGRoot (ví dụ: this->width = ...).

## 4. SVGRoot::parseNode() (Quản lý & Đệ quy)

---

- Hàm loadFromFile gọi parseNode(svgNode, this, nullptr); để bắt đầu xử lý các con của thẻ <svg>.
- parseNode lặp qua từng node con trực tiếp của svgNode (ví dụ: thẻ <path>, thẻ <g>, thẻ <rect>...).
- VỚI MỌI NODE CON (ví dụ: một thẻ <path>):
  - c. (Factory) Gọi createElement(childNode). Hàm này xem tên node ("path") và trả về một đối tượng mới: new SVGPath() (hoặc make\_unique<SVGPath>()).
  - c. (Điền dữ liệu) Ra lệnh cho đối tượng mới: newElement->parseAttributes(childNode);.
  - c. (Lưu trữ) Thêm đối tượng newElement (nay đã có đủ dữ liệu) vào danh sách this->elements.push\_back(newElement).

## 5. SVGELEMENT::parseAttributes() (Trích xuất Dữ liệu)

---

- Đây là hàm bị parseNode gọi.
- Nó chạy một vòng lặp nhỏ chỉ duyệt các thuộc tính (ví dụ: id, fill, stroke, d...) của một node XML cụ thể.
- Nó gán các giá trị đọc được vào các biến thành viên của chính nó (ví dụ: this->id = "myPath";, this->fill = "none";).
- Khi hàm này kết thúc, đối tượng C++ (ví dụ SVGPath) đã được "lắp đầy" dữ liệu.