

레거시 웹 호환을 위한 크로스 브라우징 솔루션 구조 및 개발

위성준, 김양중*

한국공학대학교 소프트웨어융합공학과
{wisj82, *zeroplus}@tukorea.ac.kr

Implementation and Architectural Model of Cross Browsing Solutions for Legacy Web Compatibility

Sungjun Wi, Yangjung Kim*
Tech University of Korea

요 약

웹 표준 기술의 비약적 발전에도 불구하고, 국내외 기업과 공공기관에서는 여전히 인터넷 익스플로러 및 ActiveX 기반의 레거시 웹 시스템을 운영하고 있다. 이에 따라 최신 브라우저 환경에서의 호환성 문제가 심각하게 대두되고 있으며, 업무 연속성 확보와 시스템 전환 비용 절감의 필요성이 커지고 있다. 따라서, 브라우저 확장, 네이티브 메시징 및 내장 IE 엔진의 연동 구조를 통해 IE 전용 웹 서비스를 다양한 브라우저에서도 서비스 연속성을 갖게 하는 시스템 구조와 동작 원리를 체계적으로 분석해 봄으로써, 실제 활용시에 효용성, 기술적 한계, 보안 위험, 그리고 최신 웹 표준 및 모바일 환경과의 호환성, 또한 분석하여 크로스 브라우징 솔루션의 구현 시 참조모델 구조와 관련 메시지 타입을 정의해 이를 제안하고자 한다.

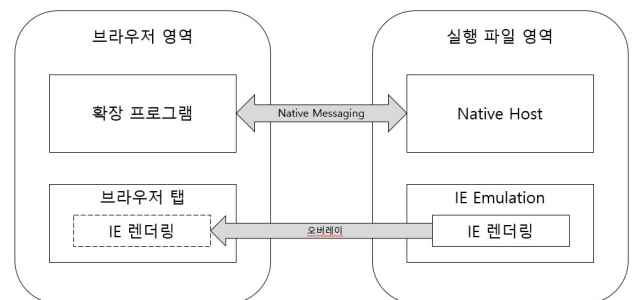
I. 서 론

HTML5[3], CSS3, JavaScript 및 WebAssembly[4] 등 웹 표준 기술들은 웹 생태계의 발전을 이끌어 왔고 이용자로 하여금 사용성을 높이는데 기여하고 있다. 그러나 레거시 시스템은 여전히 기존의 IE (Internet Explorer) 및 ActiveX 을 기반으로 하고 있음으로 호환성과 보안적 위협의 대상이 되고 있다. 2022 년 마이크로소프트의 IE 지원 종료 이후[2], 더욱 기존 시스템의 접근성 저하와 보안 위협 그리고 유지보수 비용 증가 등 우려된 문제가 속출하고 있다. 여전히 ActiveX 와 같은 비표준 기술을 사용하는 서비스들은 최신 브라우저 및 모바일 환경과의 호환이 되지 못해 서비스를 원활히 제공하지 못하는 상황이다[1]. 이에, 레거시 시스템의 사용성을 보장하면서도 점진적으로 웹 표준으로의 전환을 지원할 수 있는 기술적 대안인 크로스 브라우징 솔루션의 도입되고 있는 바, 현실적인 시스템 요구사항과 이벤트 처리 프로시저를 정의하여 이 문제를 해결한 결과를 제공한다.

II. 본 론

1. 크로스 브라우징 솔루션의 시스템 구조와 동작 원리

크로스 브라우징 솔루션은 웹 페이지 또는 웹 애플리케이션이 IE, Chrome, Edge 등 다양한 최신 브라우저 환경에서도 원활하게 동작할 수 있도록 구성된 기술로, 각 브라우저의 HTML, CSS 및 JavaScript 처리 방식의 차이로 인해 발생하는 호환성 문제를 해결하는데 목적이 있다. 본 시스템은 특히 IE 및 ActiveX 에 의존적인 레거시 시스템을 최신 브라우저 환경에서도 유지할 수 있도록 설계되었다. 그림 1 은 본 논문에서 제안하는 크로스 브라우징 솔루션의 전체적인 시스템 구조를 나타낸 것이다[5].

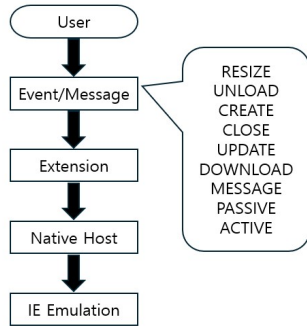


<그림 1. 크로스 브라우징을 위한 시스템 구조도>

시스템은 브라우저 영역과 실행 파일 영역으로 구성되며, 두 영역은 Native Messaging 기반의 메시지 통신을 통해 유기적으로 연동된다. Native Messaging 은 브라우저 확장 프로그램이 로컬 시스템에 설치된 실행 파일과 JSON 형식의 메시지를 통해 안전하게 통신할 수 있도록 하는 구조로, 브라우저 샌드박스 밖의 기능과 연동이 필요한 상황에서 주로 사용된다.

브라우저 영역에는 Chrome 확장 프로그램이 설치되어 있으며, 사용자의 이벤트를 감지하여 실행 파일과 통신하는 역할을 수행한다. 사용자가 보는 화면에는 IE 기반 콘텐츠가 마치 브라우저 내부에서 실행되는 것처럼 시각적으로 표시되지만, 실제로는 외부 실행 파일에서 구동된 IE 엔진의 렌더링 결과가 브라우저 탭 위치에 맞춰 오버레이되어 출력된다. 이러한 구조는 확장 프로그램과 실행 파일 간의 실행 권한을 분리함으로써, 보안적으로 민감한 렌더링 처리와 시스템 자원 접근을 브라우저 외부에서 통제할 수 있어 안정성과 보안성을 동시에 확보할 수 있다.

실행 파일 영역은 확장 프로그램으로부터 메시지를 수신하는 Native Host 와 IWebBrowser2 기반 COM 인터페이스를 통해 IE 엔진을 구동하는 IE Emulation 으로 구성된다. 사용자의 명령 또는 이벤트가 확장 모듈을 통해 전달되면, Native Host 는 이를 해석하고 IE Emulation 에 명령을 전달하여 해당 웹 페이지를 렌더링하게 된다. 이때 렌더링된 결과는 브라우저 탭 위치에 시각적으로 오버레이되어 표시된다. 그림 2 는 이러한 동작 과정을 이벤트 및 메시지 흐름 중심으로 도식화한 것이다[5].



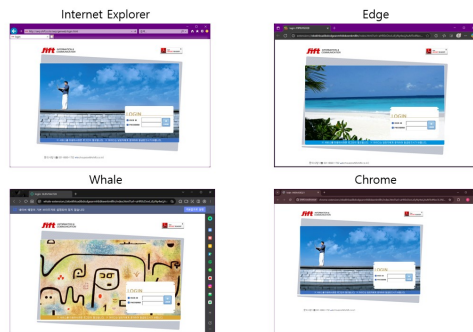
<그림 2. 크로스 브라우징 솔루션의 이벤트/메시지 프로시저>

사용자가 브라우저를 실행하고 확장 모듈이 활성화되면, 브라우저 환경 분석 결과에 따라 Native Host 에 정보가 전달되고, IE Emulation 이 구동되어 웹 페이지를 로딩한다. 시스템은 사용자 인터랙션, 창 크기 변경, 브라우저 종료 등 다양한 상황에 따라 RESIZE, UNLOAD, CREATE, CLOSE, UPDATE, DOWNLOAD, PASSIVE, ACTIVE 등의 메시지를 주고받으며 각 모듈이 동작하도록 한다[5].

이러한 구조를 통해 사용자는 ActiveX 와 같은 비표준 기술을 사용하는 기존 웹 서비스도 최신 브라우저 환경에서 별도의 IE 실행 없이 사용할 수 있으며, 브라우저와 IE 엔진을 물리적으로 분리한 구조를 통해 보안 위험을 최소화하면서도 사용자 경험을 유지할 수 있다.

2. 크로스 브라우징 솔루션의 적용 결과 및 사례

제안한 크로스 브라우징 솔루션을 기반으로 PMS 및 MES 시스템 구축을 가정하고, ActiveX 기반 UI 컴포넌트를 활용하는 웹페이지에 적용한 결과는 그림 3 에 나타나 있다.



<그림 3. 크로스 브라우징 솔루션 적용 사례>

테스트 대상은 Grid, ComboBox, Report, Chart, Dataset 등 총 643 개의 ActiveX 컴포넌트를 포함하고 있으며, 다양한 브라우저 환경에서도 기존 IE 환경과 동일한 화면과 기능이 정상적으로 동작하는 것이 확인되었다.

특히 Chrome, Edge, Whale 등 주요 브라우저에서 별도의 코드 수정 없이 기존 시스템과 동일한 사용자 경험을 제공하였으며, IE 환경뿐만 아니라 타 브라우저에서도 오류 없이 안정적으로 콘텐츠가 출력되어 100% 호환성을 유지하는 것으로 나타났다.

이러한 결과는 웹 표준 전환이 어려운 레거시 환경에서도 기존 시스템 자산을 유지하며 최신 브라우저 기반의 서비스 제공이 가능함을 보여준다.

III. 결론

본 논문에서는 IE 및 ActiveX 기반 레거시 웹 시스템의 한계를 극복하기 위해, 최신 브라우저 환경에서도 호환성을 유지할 수 있는 크로스 브라우징 솔루션의 구조와 동작 절차를 정의하고 이를 구현하였다. 제안된 시스템은 다양한 ActiveX 기반 UI 컴포넌트에 적용 가능하며, Chrome, Edge 등 최신 브라우저 환경에서도 안정적인 호환성과 사용자 경험을 제공하는 것으로 확인되었다.

향후에는 WebAssembly 모듈 기반의 시스템 확장 방안을 모색하고, 크로스 브라우징 환경에서의 보안성 및 성능 향상을 위한 추가 연구를 지속적으로 진행할 예정이다.

ACKNOWLEDGMENT

본 연구는 고용노동부 및 한국산업인력공단의 ‘2025 년 고속런 마이스터 사업’의 지원을 받음.

참 고 문 헌

- [1] 행정안전부, “공공 웹사이트 플러그인 제거 가이드라인,” 2018.
- [2] Microsoft, "Lifecycle FAQ - Internet Explorer and Edge," 2022.
- [3] W3C, "HTML5 Specification," 2023.
- [4] MDN Web Docs, "WebAssembly Concepts," 2024.
- [5] 최홍규, 박인복, 위성준, "웹브라우저에 이중 웹브라우저의 프레임을 표시하는 장치," 대한민국 특허출원 10-2015-0164649, 2015.