

# 하위 호환성을 염두한 크로스브라우저에 대한 단상

---



## We recommend the following browsers.

To make use of rich web applications in MobileMe, we recommend one of the following supported browsers:

- [Safari 3.1 or later](#) (Mac / PC)
- [Firefox 2 or later](#) (Mac / PC)

Update your browser now to access your email, calendar, contacts, and more at [me.com](#)

Questions? Go to [MobileMe Support](#)

Don't show again

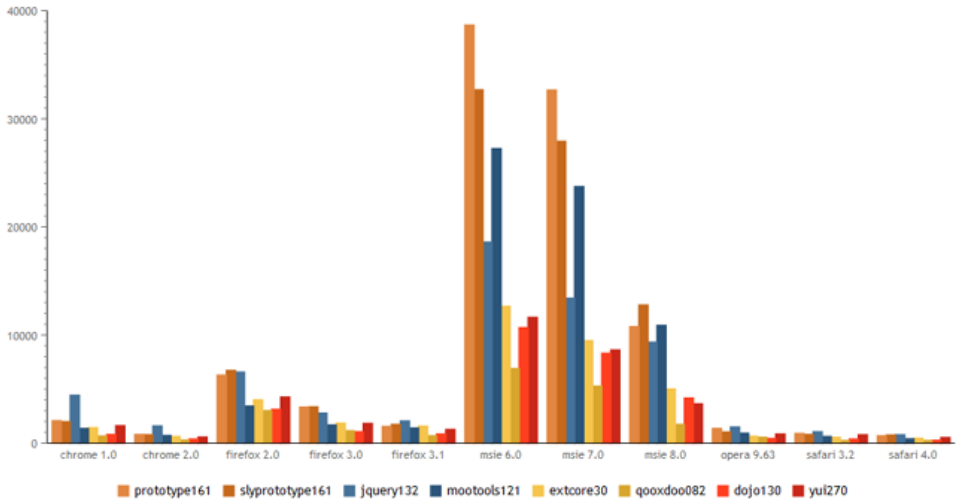
Continue

© Copyright 2009 Apple Inc.



요즘 새로이 출시되는 웹 서비스들 중에는 과감하게 IE6을 지원 대상에서 배제하고 개발한 경우를 종종 볼 수 있습니다. 서비스 성격에 따라 여러가지 이유들이 있겠지만, 그 이유들 중에 분명한 하나는 개발 장애 요인을 무수히 내포하고 있기 때문일 것입니다. IE6을 서비스 지원 범위에 포함하면서 부터 웹 개발자는 생지옥을 경험하게 되니까요. 여담입니다만, [IE의 attachEvent 문제](#) 하나로 파생된 이슈는 십만여 건에 달하고 있으며, 이로 인하여 개발자들의 한숨에서 배출되는 CO2는 연간 100만 톤에 이르는 것으로도 알려져 있습니다. 상식의 범주를 넘어선 지랄맞은 이슈들이 발생하고 그로 인한 마이너 패치들은 순수 기능 자체의 성능을 저하시키기 일수여서 비 IE계열 브라우저들이 피해를 보거나 기능 자체를 아예 제공하지 않도록 결정해 버리는 등 황당무계한 상황들이 전개되는 것을 쉽사리 목격할 수 있었습니다.

제가 경험한 바에 따르면, IE6과 7을 지원하기 위해 소모하는 리소스 비율은 프로젝트 단위로 최소 20%에서 40% 이상을 차지했던 것으로 기억하고 있습니다. 이슈 트래커(맨티스, 트랙)의 기록에만 근거해서 말이죠. 자바스크립트 사용량이 많아질 수록 작업량은 더욱 증가합니다. 작업 리소스가 40%이상으로 할당 될 것이 예상되면 IE에서 보여지는 웹페이지는 개별적으로 디자인(설계) 하는 것이 여러모로 이롭다고 생각하게 되었습니다. 크로스브라우저를 위한 자바스크립트 라이브러리를 들고 있을 필요도 없고, IE만을 위한 스타일시트를 구분하여 로드할 필요도 없거니와 IE의 문서 표현영역을 고려할 필요도 없으며, 웹 애플리케이션은 각 브라우저의 환경에서 최고의 성능을 낼 수 있는 구조로 구축할 수 있게 되겠죠. 프론트엔드 엔지니어의 입장에서는 IE류와 비 IE류로 구분하여 분할하고 병렬로 작업을 진행 할 수도 있겠군요.



브라우저가 웹을 표현하는 영역이 다양해 지고 처리 성능이 급증함에 따라

**꿈처럼 생각하던 일들**이 현실로 나타나고 있습니다. 자바스크립트 해 보겠다고 달려든 지도 벌써 4년째에 접어들었네요. 그동안은 이러한 문제를 오로지 시간만이 해결 해 줄 것으로 굳게 믿고 있었습디만, 관행처럼 이어져 온 것들이기 때문에 앞으로도 적잖은 시간이 소요될 것임을 스스로 깨닫고 있습니다. 이제는 조금씩 인식을 바꾸어야 할 때가 아닐까요? 성능 차이가 40배 이상 벌어진 이 마당에 크로스브라우저를 운운하며 생 지랄을 쳐 까는 짓은 그만 두자는 것이죠, 진보적인 웹 개발에 걸림돌이 될 것이 불보듯 뻔하니까요. 그 실천의 일환으로 **IE6 사용자들에게 업그레이드 경고**를 보여주고 있습니다. 찾아보니 **그럴싸 한 라이브러리**도 있더군요. 그 다음 단계는 브라우저 단위 버저닝입니다. 구린 브라우저 사용자에게는 구린 사용자 경험(UX)을 제공하는 수 밖에 없다는 것이 제 결론입니다.