

Shared Object in Micromedia Flash 9 (action script 2)

# Shared Object

# Index

- Introduce Shared Object
- Defferences
- Characteristics
- Object Outline
- Javascript Library
- Usage

# Introduce

- **Shared Object**는 클라이언트에 저장된 데이터를 사용합니다.
- **Shared Object**는 기능은 웹 브라우저에서 생성하는 쿠키와 유사하지만 저장방식과 로딩방식은 다릅니다.

# Defference

- Browser cookies는 Browser 에 의해 제어되고 Shared-Objects 는 Flash player 에 의해서 제어된다.
- Browser cookies 나 Shared-Objects 모두 사용자 PC에 저장되지만 Browser cookies 의 경우 대부분의 사용자가 지우는 방법을 알고 있고 많은 유틸리티도 제공되고 있다.  
하지만 Shared-Objects 는 위치가 명백하지 않으며 삭제도 쉽지 않다. 그래도 좋다, 나쁘다는 보는 이의 관점에 따라 달라집니다.
- Local Shared Objects 는 Browser cookies 대안으로 좋습니다. 사용도 더 쉽고 간단합니다.
- Shared-Object는 “expire” 메커니즘이 없습니다. 하지만 Browser cookies 에는 있습니다.
- Browser cookies 는 사용자가 볼 수 있도록 저장되지만 Shared Objects 는 binary 데이터로 저장됩니다.

# Characteristics

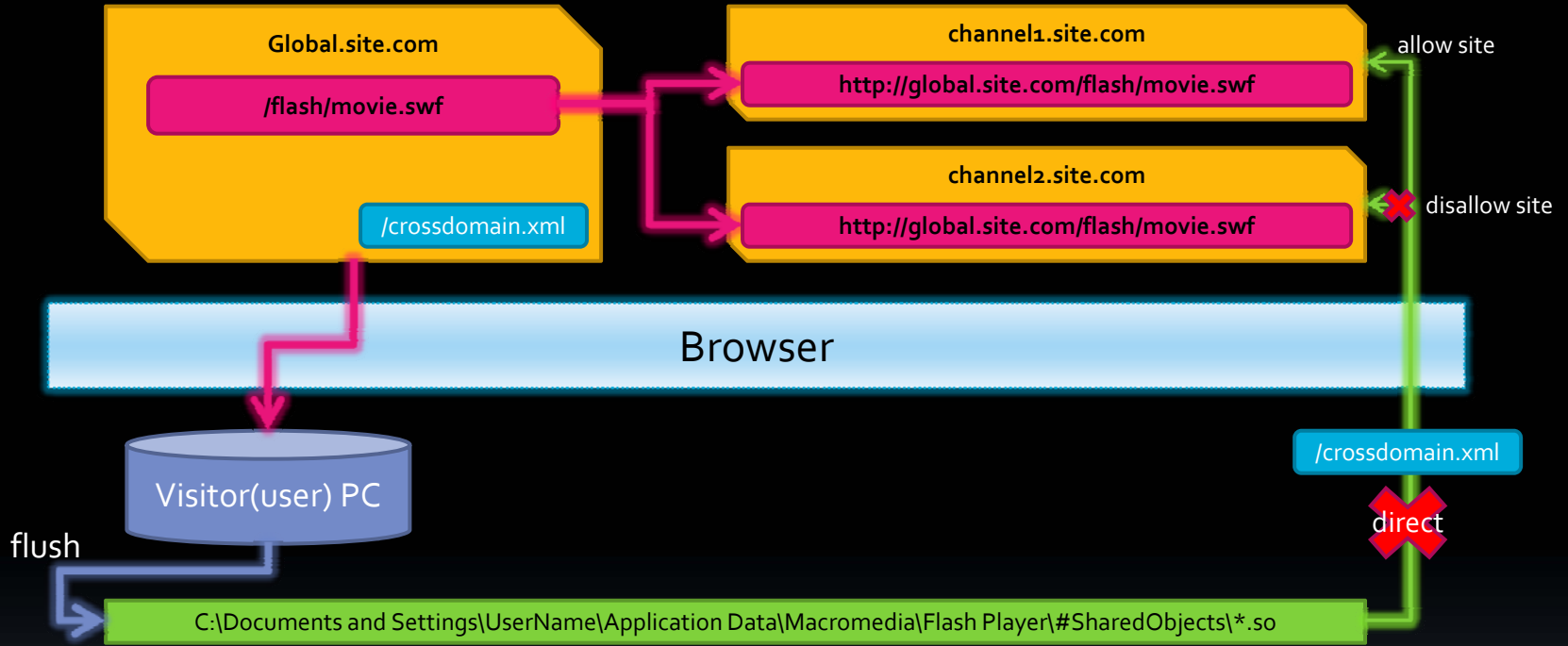
- Local Shared Object 는 Shared Object를 사용(발생)한 동일 도메인 에 있는 Flash Movie에 의해서 읽혀집니다. 즉 mysite.com 의 Flash Movie 가 생성한 Shared Object를 yoursite.com 에 있는 Movie가 읽을 수 없습니다.
- 현재 사용하는 Local Shared Object는 실제로 데이터를 썼던 Flash Movie가 이를 지시하는 유일한 방법입니다.
- Local Shared Object 는 사용자 PC에 저장됩니다. 그래서 사용자에게 의해 의도적 혹은 실수로 삭제가 가능합니다. 예를 들어 포맷하거나 지정된 위치를 삭제해 버린다면 사라집니다. 윈도우에서는 C:\Documents and Settings\UserName\Application Data\Macromedia\Flash Player\#SharedObjects\ 에 저장됩니다.
- Macromedia 에 문서에 따르면 Shared Object의 파일 확장자가 \*.so 라고 되어있지만 window XP에서는 \*.sol 파일로 생성됩니다.

# Characteristics

- Local Shared Object 를 사용하지 못할 수도 있습니다. 특정 사이트를 방문하여 사용자 디스크에 쓰는 것을 허가하지 않기 때문입니다. 이것은 Macromedia Player Setting window를 통해서 수정 할 수 있습니다.
- Flash Player 6 혹은 그 이상에서만 Shared Object 를지원하고 있습니다.
- Shared Object는 Flash Movie 1개당 1개의 Shared Object만 국한되지 않습니다.
- Cross-Domain 도 가능합니다. 예를 들어 global.site.com 에서 생성 된 Shared Object를 my.site.com 에서 사용하고자 할때 Flash Movie가 존재하는 Domain 의 document root 에 crossdomain.xml 파일이 존재하여야 합니다. 이를 통해 Allow 할 도메인을 선별해 공유해 사용할 수 있습니다.



# Object Outline



crossdomain.xml



```
<?xml version="1.0" encoding="utf-8" ?>
<!-- http://www.pandora.tv/crossdomain.xml -->
<-cross-domain-policy>
<allow-access-from domain="*" />
</cross-domain-policy>
```

# Javascript library

- API

method	Static Class TSharedObject
arguments	none
description	Returns a singleton instance of this class
return	Return <i>object</i> Object
usage	<b>var</b> instance = TSharedObject. <a href="#">getInstance()</a> ;



# Javascript library

- API

method	Static Method Instance.setup()
arguments	none
description	Flash Object Dynamic creational
return	none
usage	<pre>var instance = TSharedObject.getInstance(); instance.setup();</pre>

# Javascript library

- API

method	Static Method Instance.setItem()
arguments	setItem( <i>string</i> key, <i>string</i> value[, <i>string</i> so])
description	Shared Object data save method
return	Return <i>boolean</i>
usage	<pre>var instance = TSharedObject.getInstance(); instance.setItem('name', 'rhio.kim', 'channel'); instance.setItem('age', '16', 'channel'); instance.setItem('movies.1', '{ list : [a,b,c]}', 'video'); instance.setItem('movies.2', '{ list : [d,e,f]}', 'video');</pre>

# Javascript library

- API

method	Static Method Instance.getItem()
arguments	getItem( <i>string</i> key [, <i>string</i> so])
description	Shared Object data get method
return	Return <i>string</i>
usage	<pre><b>var</b> instance = TSharedObject.getInstance(); instance.getItem('name', 'channel');</pre>

# Javascript library

- API

method	Static Method Instance.removeItem()
arguments	removeItem( <i>string</i> key[, <i>string</i> so])
description	Shared Object data remove method
return	Return <i>boolean</i>
usage	<pre><b>var</b> instance = TSharedObject.getInstance(); instance.removeItem('name', 'channel');</pre>

# Usage

- Using powerful cookie interface
- Dynamic user interface config
- Cross-browser Data Sharring

The end