EPUB 스크립트 구성요소(Scriptable Components)

초안 표준 2015년 7월 6일

현재 버전:

http://www.idpf.org/epub/sc/api/sc-api-20150706.html

최신 버전:

http://www.idpf.org/epub/sc/api

이전 버전:

http://www.idpf.org/epub/sc/api/sc-api-20150115.html

Copyright © 2014-2015 International Digital Publishing Forum™

All rights reserved. 이 문서는 미국 법전 제17편(Title 17 of the United States Code)의 보호를 받는다. 이 문서의 수정 복제와 배포는 International Digital Publishing Forum (IDPF)의 서면 허가(written permission)를 제외하고 금지된다.

EPUB은 IDPF(International Digital Publishing Forum)의 등록된 상표이다.

편집자

Will Manis, Invited Expert

Brady Duga, Google

Darryl Lehmann, Imagineeringart.com Inc.

Mike Chen, Inkling

한국어 편집자

- 총괄 : 임순범(숙명여대)
- 번역 및 편집: 김성중(교보문고)
- 검수 : 박웅영(토네이도미디어그룹/심야책방), 은종민(모두의지혜), 인용인(UHI)

문서의 상태

이 표준은 진행중인 문서이고 변경의 대상이다.

이 표준의 모든 코멘트는 Google Code에서 서비스 되는 EPUB 3 WG Issue Tracker에 제출될 수 있다. 메모: 당신은 이슈를 제출하기 위해서 구글 계정으로 로그인해야만 된다. "Type-ReviewComment" 그리고 "Spec-EPUBSC" 표기로 표준 코멘트를 제출해 주세요. EPUB WG의 회원은 mailing list으로 질문을 제출할 수도 있다.

이 초안에서 공개 이슈

third bullet in s. 2.2은 어떻게 리딩 시스템이 스파인 레벨 스크립트을 지원하지 않지만 ESC-인식하거나 ESC 인식 못하지만 스파인 레벨 스크립트를 지원하는 상황을 다루는지에 대해 지시한다. 제3의 케이스는 리딩 시스템은 스파인 레벨 스크립트를 지원하고 ESC 인식하는 것이 확인된다. 기대가 이 경우인지, 나중 연구가 실제 시스템에서 필요할지는 분명하지 않다. 나중 가이드가 미래 초안에 추가될 수도 있다.

이 절은 문서가 초안 상태에서 바뀔 때 제거될 예정이다.

목차

1. 개요

1.1. 목표와 범위

1.1.1 보안

1.1.2 리딩 시스템 통신에서 유저 에이전트

1.1.3 개인정보보호

1.2. 용어집

1.3 적합성 표현법

1.4. 표기 규약

2. 적합성

2.1 콘텐츠 적합성

2.2 리딩 시스템 적합성

3. 스크립트 구성요소

3.1 통합

3.2 보안

4. 구성요소간 통신

4.1 소개

4.2 PubSub 프로토콜

4.2.1 PubSub 통신 브릿지

4.2.2 메세지 구조

4.2.3 타이밍

4.3 함수

4.3.1 구현

4.3.2 구독 ("epubsc_subscribe")

4.3.3 구독해지 ("epubsc_unsubscribe")

4.3.4 발행 ("epubsc_publish")

4.4 토픽

4.4.1 통신

4.4.2 예약 토픽

4.4.2.1 준비 ("epubsc_ready")

4.4.2.2 일시정지 ("epubsc_pause")

4.4.2.3 재시작 ("epubsc_resume")

4.4.2.4 로드 ("epubsc_load")

4.4.2.5 언로드 ("epubsc_unload")

4.4.2.6 이벤트 ("epubsc_event")

4.4.3 토픽 이름

4.5 초기화

5. 구성요소 부모 이벤트 통신

5.1 소개

5.2 이벤트 처리와 전파

5.3 필수 이벤트

부록 A. 메시지 스키마

부록 B. ComponentID 샘플 코드

부록 C. 유스케이스 예제

부록 D. 감사의 글과 기여자

참고자료

규범적 참고자료

정보 제공 참고자료

1. 개요

1.1 목적과 범위

본 절은 정보 제공을 위한 것이다.

이 표준(EPUB 스크립트 구성요소, EPUB Scriptable Components)은 인터랙티브 콘텐츠가 EPUB 출판물에서 생성되고 협력될 수 있게 상호운영적인 배포와 구독 (pubsub) 패턴을 제공한다. 이 인터렉티브 콘텐츠는 이 표준의 요구사항에 적합한 어떤 리딩 시스템에서도 표현될 수 있다.

이 표준은 [ContentDocs301]에서 정의된 스크립트 아키텍처에 대한 대체품이 아니다. 그리고 API와 프로세스를 구현 또는 사용하는 것은 표준에서 정의된 요구사항 그리고 제한사항, 모범사례를 대신하지 않는다.

스크립트 구성요소는 단지 콘테이너-제한 스크립트를 지원하거나 전체 스파인-레벨 스크립트를 지원하는 리딩 시스템에서 사용될 수 있다.

1.1.1 보안

이 표준과 [ESCPackage]의 목표는 제3자 스크립트 구성요소 (위젯)이 안전하게 XHTML 콘텐츠 문서에 통합될 수 있게 허용한다. 이 표준은 스크립트 구성요소가 [HTML5] iframe 요소를 사용해서 임베디드되는 것을 요구한다. EPUB 출판물 내의 XHTML 콘텐츠 문서 또는 다른 스크립트 구성요소에 임베디드된다.

이 표준은 스크립트 구성요소 사이에 영역 분리(domain isolation)가 존재하는 것을 가정한다. 그러나 리딩 시스템은 iframe 요소에 참조되는 문서에서 영역 분리를 강제 또는 강제하지 않을 수 있다. 만약 영역 분리가 상대적으로 격리의 높은 수준으로 강제된다면 보안이 얻어진다. 영역 분리가 강제되지 않는

다면 보안이 문제가 될 수 있다 (예, 스크립트 구성요소는 EPUB 출판물 그리고(또는) 리딩 시스템과 타협해야 할 수 있다).

이 표준에 대한 준수는 어떤 특정한 보안 레벨을 보장하지 않는다. 리딩 시스템 구현자는 제3자 스크립트로 보안을 얻을 수 있는 프레임워크를 제공하려는 의도이다.

1.1.2 리딩 시스템 통신에서 유저 에이전트

모바일 디바이스에서 일부 리딩 시스템은 콘텐츠를 렌더링하기 위해 시스템에서 제공되는 웹 구성요소(종종 WebView로 참조되는)를 이용하여 구현된다. 이 구성요소들은 리딩 시스템의 나머지와 통신 수준을 제공하는게 일반적이다. 예를 들면, 콘텐츠 로딩과 진행사항을 리포팅하기 위해서이다. 이런 호출은 보안과 개인정보보호 영향을 끼칠 수 있다. 예를 들어, 사용자의 이름과 사용자 라이브러리에서 도서 목록을 알아내는 것이 가능할 수 있다. 또한 이런 통신은 리딩 시스템에서 버퍼 오버플로우와 같은 취약점을 드러내기 위해 사용될 수 있다.

이 표준은 이들 이슈사항을 직접적으로 해결하지 않는다. WebView와 앱 통신의 보안과 개인정보보호 영향에 대해 책임지는 것은 리딩 시스템 구현자의 책임이다.

1.1.3 개인정보보호

스크립트 구성요소는 발전함에 따라, 이들은 원격(클라우드) 데이터(민감한 개인 정보를 포함할 수 있는 데이터)를 찾고 저장하기 위해 통신 자원을 불변적으로 사용하게 된다.

이 정보의 개인정보보호에 대해 확인하고 강화하는 것은 이 표준의 범위 밖이다. 리딩 시스템은 네트워크를 가로질러 통신하는 것을 제한하거나 제거하기 위해 또는 쿠키와 로컬 스토리지를 제약하거나 허용하지 않기 위해 기준을

둘 수 있다.

1.2 용어집

이 문서에서 사용된 EPUB 특정 용어집의 정의에 대해 EPUB 표준을 참고하라.

구성요소 부모

스크립트 구성요소를 주최하는 부모, 다음 중 하나일 수 있다:

- [HTML5] iframe에서 참조된다면 다른 스크립트 구성요소, 또는
- 최상위 콘텐츠 문서에서 iframe에서 참조된다면 리딩 시스템

PubSub 통신 브릿지(Communication Bridge)

메시지 통신을 수용하기 위해 4.2.1 PubSub 통신 브릿지에 정의된 발행자/구독자 관계

스크립트 구성요소

다른 EPUB 출판물에서 사용하기 위해 배포되는 스크립트 배포 가능 객체(Distributable Object). 스크립트 구성요소는 이 표준과 [ESCPackage]에서 모든 콘텐츠 요구사항에 적합하다.

1.3 적합성 표현법

이 문서의 키워드 필수사항(MUST, MUST NOT, REQUIRED), 권장사항(SHALL, SHALL NOT, SHOULD, SHOULD NOT,RECOMMENDED), 선택사항(MAY, OPTIONAL)는 [RFC2119]에서 설명된 것처럼 해석된다.

이 표준의 모든 절은 "본 섹션은 정보 제공을 위한 것이다" 정보 제공 상태 표시 확인한 곳을 제외하고 규범적이다. 절에 대한 정보 제공 상태의 적용은 포함될 수 있는 모든 자식 콘텐츠와 부절에 적용한다.

이 표준의 모든 예제는 정보 제공을 위한 것이다.

1.4 표기 규약

본 표준 문서의 원문에 정의된 표기 규약은 웹 출판을 위한 것으로 종이인쇄용으로는 적합하지 않아 생략한다.

2. 적합성

2.1 콘텐츠 적합성

스크립트 구성요소를 지원하는 EPUB 콘텐츠 문서는 다음 기준을 반드시 만족해야 한다:

- 3. 스크립트 구성요소(Scriptable Components)의 모든 콘텐츠와 보안 요구사항에 반드시 적합해야 한다.
- 4. 구성요소간 통신(Inter-Component Communication)에 정의된 pub/sub 프로토콜을 지원하는 EPUB 스크립트 구성요소 코드를 반드시 포함해야 한다.

EPUB 스크립트 구성요소를 포함하는 EPUB 출판물은 반드시 [ESCPackage]와 적합하게 패키지되어야 한다.

2.2 리딩 시스템 적합성

스크립트 구성요소를 지원하는 EPUB 리딩 시스템은 다음 기준점을 만족해야만 한다:

- EPUB 표준(특히 스크립트에 대한 요구사항 [ContentDocs301])에 의해 리딩 시스템에 위치되는 모든 적합성 요구사항을 반드시 만족해야 한다.
- 컨테이너 제한 스크립트 [ContentDocs301]는 [HTML5] iframe 요소를 반드시 지원해야 한다 .
- 스파인 레벨 스크립트 [ContentDocs301]가 지원되지 않는다면, 리딩 시

스템은 최상위 수준 콘텐츠 문서에 직접적으로 임베디드된 스크립트 구성요소에 대해 4.2.1 PubSub 통신 브릿지에서 정의된 pubsub 프로토콜을 반드시 지원해야 한다(즉, 최상위 수준에서 통신을 보장하기 위해서). 그렇지 않다면, 리딩 시스템은 스파인 수준 스크립트가 지원되고 그것이 허용되면, 스파인 수준 스크립트 구성요소는 이미 구현되었기 때문에 pubsub 프로토콜을 절대로 구현하면 안된다.

3. 스크립트 구성요소

3.1 통합

모든 스크립트 구성요소는 반드시 [HTML5] iframe 요소에서 서비스되어야 한다.

원격 서비스 (예, 오디오/비디오)가 허용되지 않는다면, 스크립트 구성요소에 의해 사용되는 모든 자원은 반드시 EPUB 콘테이너에서 포함되어야 한다.

3.2 보안

스크립트 구성요소는 반드시 EPUB 콘텐츠 문서에서 분리된 영역에 존재하는 것을 가정해야 한다. 리딩 시스템은 스크립트 구성요소의 영역 분리를 강제할 수 있다.

EPUB 콘텐츠 문서와 포함된 스크립트 구성요소 사이에 통신하는 유일한 방법은 반드시 EPUB 스크립트 구성요소 프로토콜을 이용해야 한다. 스크립트 구성요소 경계를 넘어서는 직접적인 접근 허용하지 않는다; window.parent [HTML5]에 포함된 특성은 절대로 접근되면 안된다.

4. 구성요소간 통신

4.1 소개

본 섹션은 정보 제공을 위한 것이다.

스크립트 구성요소는 HTML5 postMessage API [HTML5WebMsg]의 구현에 의해 내부적으로 주도되는 프로토콜 층을 통해 통신한다.

스크립트 구성요소는 이 모델에서 구독자 그리고 구독되는 것에 의해 유비쿼터스적으로 작용한다. 본 배포된 모델은 각각의 스크립트 구성요소가 차례로 직접 종속된 자손과 그 부모에 메세지 프로토콜을 중계하는 것에 대해 책임이 있다는 것을 의미한다.

본 프로토콜은 스크립트 구성요소가 통신하는 방법을 정의하고 통신을 위한 메시지 토픽의 집합을 선정의한다.

4.2 PubSub 프로토콜

4.2.1 PubSub 통신 브릿지

window.postMessage() API [HTML5WebMsg]는 다른 영역에서 두 개의 스크립트 구성요소간 안전한 통신 브릿지를 제공하고, 타겟 윈도우(구독자)에 메시지를 발행하기 위해 스크립트 구성요소에 의해 사용된다.

스크립트 구성요소는 타겟 윈도에 메시지를 발행하기 위해 PubSub 통신 브릿지로 참조되는 window.postMessage() 방법 [HTML5WebMsg]을 반드시 사용해야 한다.

구독자는 메시지를 구독하기 위해 이벤트 핸들러 이벤트 타입 "message"

[HTML5WebMsg]에서 이벤트 리스너를 차례로 반드시 등록해야 한다.

4.2.2 메시지 구조

스크립트 구성요소는 발생하거나 전달해야 됨에 따라 토픽과 이벤트를 중계하기 위해 PubSub 통신 브릿지로 메시지를 발행한다.

각 메시지는 부록 A에서 스키마에 적합한 JSON 객체 [RFC4627]로 반드시 직렬화되어야만 한다.

- componentId - 필수사항(REQUIRED) componentId 특성은 세션 주기 동안 스크립트 구성요소를 유일하게 식별하는 문자열 식별자를 포함한다 (예, GUID [RFC4122]). 예제로 부록 B을 확인해라.
- messageId - 필수사항(REQUIRED) messageId 특성은 메시지를 유일하게 식별하는 문자열 식별자를 포함한다. 이 표준은 식별자에 대해 포맷을 요구하지 않는다. 식별자는 스크립트 구성요소 식별자 또는 단순 증가 값 또는 유일함을 보장하는 그 어떤 값의 조합이 될 수 있다.
- timestamp - 필수사항(REQUIRED) timestamp 특성은 메시지가 발생된 시간을 식별한다. 값은 리딩 시스템에서 구현된 것처럼 반드시 자바스크립트 Date.now()함수에서 반환된 값이어야 한다.
- type - 필수사항(REQUIRED) type 특성은 객체가 스크립트 구성요소에서 발생된 메시지라는 것을 나타내는 것을 식별한다. 값은 반드시 문자열 "epubsc_message"이어야 한다.
- method - 필수사항(REQUIRED) method 특성은 호출된 event method 이벤트 함수을 선언한 문자열 식별자를 포함한다. 값은 반드시 다음 중 하나이어야 한다: "epubsc_subscribe" 또는 "epubsc_unsubscribe", "epubsc_publish".
- topic - 필수사항(REQUIRED) topic 특성은 메시지가 발행되는 채널의 이름을 식별한다. 값은 반드시 4.4.2 토픽 이름 그리고 4.4.3 예약 토

픽 이름에서 정의된 것처럼 토픽 이름이어야 한다(MUST).

- topicData - 선택사항(OPTIONAL) topicData 특성은 topic이 추가적인 데이타가 필요할 때 메시지에 포함된다. 존재한다면, 값은 반드시 JSON 객체 [RFC4627]이어야 한다.

다른 스크립트 구성요소에 온도 정보를 전달하는 메시지의 예제.

```
{
        "componentId": "b33ef720-556a-11e4-8ed6-0800200c9a66",
        "messageId": "b33ef720-556a-11e4-8ed6-0800200c9a66+1",
        "timestamp": 1413488477617,
        "type": "epubsc_message",
        "method": "epubsc_publish",
        "topic": "current_temperature",
        "topicData": {
                "location": "Toronto, Ontario",
                "currentTemp": 21,
                "tempFormat": "Celsius"
        }
};
```

4.2.3 타이밍

이 절에서 정의된 프로토콜의 배포라는 본질때문에, 메시지가 구독자에 수신되는 순서는 보장되지 않는다.

스크립트 구성요소는 결과적으로 다른 구성요소에서 메시지의 순차적인 도착에 의존하도록 설계되면 안된다. 스크립트 구성요소는 구성요소의 유일한 인

스턴스에서 메시지의 순서에 의존해야만 한다. 메시지 타이밍 정보는 timestamp 특성에서 추출되어야만 한다.

4.3 함수

4.3.1 구현

스크립트 구성요소는 PubSub 통신 브릿지 상에서 통신을 수용하기 위해 이 절에서 정의된 함수를 반드시 구현해야 한다.

4.3.2 구독 ("epubsc_subscribe")

epubsc_subscribe 함수는 스크립트 구성요소에 대해 토픽을 구독하기 위한 매커니즘을 제공한다.

이 함수는 청취를 시작하기 위해 토픽에 대해 인식하는 것을 요구한다.

4.3.3 구독해지 ("epubsc_unsubscribe")

epubsc_unsubscribe 함수는 스크립트 구성요소에 대해 특정 토픽을 청취하는 것을 종료하기 위한 매커니즘을 제공한다.

이 함수는 검색과 제거를 수행하기 위해 토픽 문자열 값에 대해 인식하는 것을 요구한다.

4.3.4 발행 ("epubsc_publish")

epubsc_publish 함수는 준비 스크립트 구성요소을 이전 구독 토픽에 대해 메시지를 발행하는 것을 허용한다.

이 함수는 토픽 이름과 전달되는 메시지 객체를 인식하는 것을 요구한다.

4.4 토픽

4.4.1 통신

모든 토픽 구독과 발행은 구독과 상관없이 구성요소 부모에 반드시 전파되어야 한다. 구성요소는 모든 구독한 자식에게 발행된 토픽을 반드시 전송해야 한다. 전송자가 또한 구독되어 있다면, 또한 토픽을 반드시 수신해야 한다.

구성요소는 여러 번 토픽을 구독할 수 있다.

구성요소는 구체적인 처리자를 제거하는 것으로 토픽을 구독해지할 수 있다.

4.4.2 토픽 이름

토픽 이름은 구체적인 발생 요구에 의해 전달되는 데이타가 무엇인지를 확인하기 위해 사용된다.

토픽 이름은 반드시 알파벳숫자 문자으로 시작해야 하고, 다음에 정의된 화이트 스페이스 문자를 절대로 포함해서는 안된다:

탭 문자 (U+0009)

라인 피드 (U+000A)

폼 피드 (U+000C)

리턴 키 (U+000D)

스페이스 (U+0020)

구현시 자신의 토픽 이름을 정의할 수 있다, 그러나 이름은 절대로 "epubsc" 글자로 시작하면 안된다. 이 접두어는 이 표준에서 정의한 토픽을 위해 예약되어 있다. 콜론 문자 (":", U+003A)는 앞으로 사용을 위해 이 표준에서 예약되고 맞춤 토픽 이름에서 절대로 사용되면 안된다. 그러나, 문자의 포함은 이 표준의 버전을 구현한 리딩 시스템에 의해 절대로 오류로 제시되면 안된다.

다른 예약 이름보다, 이 표준은 어떤 토픽 이름의 의미를 정의하지 않는다.

4.4.3 예약된 토픽 이름

4.4.3.1 준비 ("epubsc_ready")

토픽 "epubsc_ready"는 구성요소가 준비 상태로 진입하고 메시지를 수신할 수 있다는 것을 지시하기 위해 예약된다.

시작할 때, 스크립트 구성요소는 epubsc_ready 토픽으로 메시지를 발행하는 것으로 메시지(토픽 구독과 발행)를 처리하기 위해 준비되었다는 것을 반드시 알려야 한다.

스크립트 구성요소는 또한 저장 가능하고 구성요소 ID을 사용하여 참여하는 스크립트 구성요소 목록을 유지해야만 한다.

4.4.3.2 일시정지 ("epubsc_pause")

토픽 "epubsc_pause"는 구성요소가 일시정지 상태로 진입하는 것을 지시하기 위해 예약된다 (예를 들면, 보이는 페이지가 더 이상 아닐 때이거나 리딩 시스템이 최상위 애플리케이션이 아닐 때).

4.4.3.3 재시작 ("epubsc_resume")

토픽 "epubsc_resume"는 구성요소가 준비 상태를 재시작하는 것을 지시하기 위해 예약된다.

4.4.3.4 로드 ("epubsc_load")

토픽 "epubsc_load"는 구성요소가 로딩 시작하는 것을 지시하기 위해 예약된

다.

4.4.3.5 언로드 ("epubsc_unload")

토픽 "epubsc_unload"는 구성요소가 언로딩 시작하는 것을 지시하기 위해 예약된다.

4.4.3.6 이벤트 ("epubsc_event")

토픽 "epubsc_event"는 자바스크립트 이벤트 통신에 대해 예약된다.

4.5 초기화

메시지를 발송하고 수신하기 위해 준비하기 위해, 스크립트 구성요소는 로드 프로세스 동안에 다음 스텝을 반드시 초기화해야 한다:

- 다음 사항을 요구하는 PubSub 통신 브릿지를 반드시 시작해야 한다:

  - 모든 스크립트 구성요소는 PubSub 통신 브릿지에 메시지를 반드시 발행할 수 있어야 한다.
  - 자식 구성요소를 포함하거나 PubSub 통신 브릿지에 통신이 필요한 스크립트 구성요소는 반드시 메시지를 구독할 수 있어야 한다.

- 4..2..2 메시지 구조에 정의된 것처럼 PubSub 통신 브릿지는 구체적인 함수 타입의 메시지를 전달하는데 사용된다.
- 4..3 함수에서 통신하기 위해 사용하는 함수가 설명되었다. 이들 함수는 구체적인 처리를 위해 통신 채널을 정의한다.
- 로드 프로세스에서 다른 맞춤 스텝을 완료한 후, 현재 활성화 상태이라고 모든 구독자에 알릴 수 있게 "epubsc_ready" 토픽을 반드시 발행해야

한다 (더 많은 정보는 4.4.1 통신을 참조). 동시에, 자신의 업데이트를 받을 필요가 있다면 구성요소는 자신의 "epubsc_ready" 토픽을 구독할 수 있다.

이 단계에서, 구성요소는 PubSub 통신 브릿지를 통해 토픽을 구독하고 구독해지하고, 발행하고, 전달한다.

5. 구성요소 부모 이벤트 통신

5.1 소개

본 섹션은 정보 제공을 위한 것이다.

스크립트 구성요소는 수행에서 의미가 있거나 의미가 없는 자바스크립트 이벤트를 받을 수 있다.

예를 들면, 스크립트 구성요소의 입력 필드에 키 입력은 전통적으로 구성요소에 의해 처리될 필요가 있다. 그러나 오른쪽 화살표 키는 처리할 필요가 없다. 스크립트 구성요소가 부모 구성요소에 이벤트에 대한 정보를 전달할 것인지는(이벤트를 응답할 기회를 부모에게 주는 것) 이벤트의 본질에 의존한다.

이 절은 어떻게 스크립트 구성요소가 부모가 처리할지 여부를 포함해서 부모에게 이벤트를 발행하는 지를 정의한다.

5.2 이벤트 처리와 전파

스크립트 구성요소는 PubSub 통신 브릿지를 통해 부모 구성요소에 [UIEvents] 이벤트를 반드시 전달해야 한다.

이벤트의 통신은 이벤트가 스크립트 구성요소에 처리될지 여부는 필수사항이다. 그리고 메시지는 예약 토픽 "epubsc_event"으로 epubsc_publish 함수를 반드시 사용해야 한다.

이벤트가 처리된다면, 메시지의 topicData은 boolean 특성 handled을 반드시 포함하는 것을 지시한다. 구성요소 부모는 이 특성이 true라면 이벤트에 응답하면 안된다.

topicData는 원본 이벤트(즉, altKey, button, ctrlKey, type)에서 모든 적용가능한 속성 [UIEvents]의 복사본으로 반드시 존재해야만 한다.

다음 예제는 스크립트 구성요소가 클릭 이벤트에 대해 전달해야 하는 epubsc_event 메시지가 처리되지 않는 것을 보여준다.


```
{
        "componentId": "1e5c4271-3ad2-43be-91d3-6df78a095047",
        "messageId": "1e5c4271-3ad2-43be-91d3-6df78a095047+1",
        "timestamp": 1431014848368,
        "type": "epubsc_message",
        "method": "epubsc_publish",
        "topic": "epubsc_event",
        "topicData": {
                "handled": false,
                "type": "click",
                "screenY": 438,
"screenX": 1227,
"offsetY": 13,
"offsetX": 64,
"clientY": 13,
"clientX": 64,
"button": 0,
"buttons": 1,
"shiftKey": false,
"ctrlKey": false,
```

"cancelable": true,

"bubbles": true,

"altKey": false,

"timeStamp": 1432258647479,

"relatedTarget": null,

"metaKey": false,

"eventPhase": 3

}

}

모든 이벤트 메시지는 구성요소 부모에 전달되어야만 한다 (MUST). 구성요소는 어떤 자식에게 이벤트 메시지를 전달하면 안된다.

부록 A. 메세지 스키마

메시지에 대한 JSON 표현 스키마:


```
{
"id": "http://www.idpf.org/epub/sc/1.0/schema.json",
"$schema": "http://json-schema.org/draft-04/schema#",
"type": "object",
"title": "EPUB Scriptable Components Message Schema",
"description": "Schema validation for the message body",
"properties": {
"componentId": { "type": "string" },
"messageId": { "type": "string" },
"timestamp": { "type": "number" },
"type": {
"type": "string",
"enum": ["epubsc_message"]
},
"method": {
"type": "string",
"enum": ["epubsc_subscribe", "epubsc_unsubscribe", "epubsc_publish"]
},
"topic": { "type": "string" },
"topicData": { "type": "object" }
},
```

"required": ["componentId", "messageId", "timestamp", "method", "topic"],

"additionalProperties": false

}

스키마를 사용하는 유효성은 [JSONSchema]을 지원하는 프로세서를 요구한다.

부록 B. ComponentID 샘플 코드

본 섹션은 정보 제공을 위한 것이다.

세션에서 구성요소 ID의 유일성은 스크립트 구성요소의 행위를 조정하는 것이이 치명적으로 중요하다. [RFC4122] 구현은 충분한 유일성을 제공한다.

예제는 다음을 포함한다:

Node-UUID - https://github.com/broofa/node-uuid

UUID - https://github.com/LiosK/UUID.js

다른 예제와 다른 논의에 대해 http://stackoverflow.com/questions/105034/create-guid-uuid-in-javascript를 참고해라

또는 다음은 세션(또는 더큰 범위)에서 유일할 수 있게 ID를 생성하는 샘플 코드이다.

```
function generateComponentID_() {
    var d = new Date().getTime();
    var template = "xxxxxxxx-xxxx-Zxxx-yxxx-xxxxxxxxxxxx";
    return template.replace(/[xy]/g, function (c) {
        var r = (d + Math.random() * 16) % 16 | 0;
        d = Math.floor(d/16);
```

```
        return (c == "x" ? r : (r & 0x7 | 0x8)).toString(16);
    });
}
```

부록 C. 유스케이스 예제

본 섹션은 정보 제공을 위한 것이다.

두 개의 스크립트 구성요소가 통신하는 것을 허용해라. 하나의 구성요소에서 UI 제어는 두 번째 구성요소에서 시각 변화를 연출할 수도 있다.

임베디드된 스크립트 구성요소는 실행 비디오를 포함할 수도 있다. EPUB 구성요소는 페이지 구분되거나 스크롤될 때 리딩 시스템은 스크립트 구성요소가 비디오를 일시정지하는 것을 지시할 것이다.

IFrames에서 발생하는 UI 이벤트는 부모 윈도우에 발산되지 않는다. 스크립트 구성요소는 이벤트 전파와 부모 윈도와 리딩 시스템에 의해 처리를 가능하게 된다.

부록 D. 감사의 글과 기여자

본 섹션은 정보 제공을 위한 것이다.

EPUB은 출판사 그리고 판매자, 소프트웨어 개발자, 연관 표준에 대한 전문가를 소집하여 협력 작업으로 IDPF에서 개발되어 왔다.

EPUB 스크립트 구성요소 패키징과 통합 1.0 표준은 다음 분의 지도하에 수행한 IDPF WG에 의해 준비되었다.

- Conboy, Garth (Google) Co-chair
- Manis, Will Co-chair

출판물의 WG의 활동 회원은 다음과 같다:

IDPF 회원들

- Albert, Paolo (PubCoder)
- Baker, Mike (Houghton Mifflin Harcourt)
- Chen, Mike (Inkling)
- Conboy, Garth (Google)
- Davis, Greg (Pearson)
- DeMeglio, Marisa (DAISY)
- Duga, Brady (Google)
- Gylling, Markus (IDPF)
- Haas, Matthew (Pearson)
- Kaplansky, Jean (Aptara)
- Keating, Patrick (Bluefire)
- Kleinfeld, Sanders (O'Reilly)
- Lehmann, Darryl (Imagineeringart.com Inc.)
- Levantovsky, Vladimir (Monotype Imaging)
- Severdia, Ron (Metrodigi)
- Weck, Daniel (Readium)
- Wright, Ric (Readium)

초대 전문가

- Costa, John
- Garrish, Matt

- Manis, Will

한국어 편집자

- 총괄 : 임순범(숙명여대)
- 번역 및 편집: 김성중(교보문고)
- 검수 : 박웅영(토네이도미디어그룹/심야책방), 은종민(모두의지혜), 인용인(UHI)
-

참고자료

규범적 참고자료

[ContentDocs301] *EPUB 콘텐츠 문서 3.0.1.*

[ESCPackage] EPUB 스크립트 구성요소 패키징과 통합.

[HTML5] HTML5: HTML과 XHTML에 대한 어휘와 관계된 API.

[HTML5WebMsg] HTML5 Web Messaging 웹 메시징.

[JSONSchema] JSON 스키마: 코어 정의와 용어집.

[Publications301] EPUB Publications 출판물 3.0.1.

[RFC2119] 요구사항 수준을 지시하기 위한 RFC에서 사용되는 키워드 (RFC 2119). March 1997.

[RFC4627] JSON(JavaScript Object Notation)에 대한 application/json 미디어 타입 (RFC 4627). D. Crockford. July 2006.

[UIEvents] UI 이벤트.

정보 제공 참고자료

[RFC4122] UUID(Universally Unique IDentifier) URN 네임스페이스 (RFC 4122). P. Leach, et al. July 2005.