**대용량 데이터 유통 블록체인 서비스 API**

**(MediaBlockChain Service API)**

Version 1.0

**㈜ 마크애니**

**목차**

[1 개요 3](#_Toc48312143)

[1.1 문서의 범위 3](#_Toc48312144)

[2 서비스 설계 개념 4](#_Toc48312145)

[3 서비스 API 5](#_Toc48312146)

[3.1 접속 로그인 5](#_Toc48312147)

[3.2 업로드 파일 선택 6](#_Toc48312148)

[3.3 업로드 폴더 선택 7](#_Toc48312149)

[3.4 대표이미지 단일파일 선택 9](#_Toc48312150)

[3.5 대표이미지 다중파일 선택 11](#_Toc48312151)

[3.6 Form 콘텐츠 업로드 13](#_Toc48312152)

[3.7 BDChainUp 프로세스 15](#_Toc48312153)

[3.8 콘텐츠 다운로드(BCMC) 15](#_Toc48312154)

[3.9 BDChainDown 절차 17](#_Toc48312155)

[3.10 콘텐츠 다운로드(Circler) 19](#_Toc48312156)

[3.11 CirclerDown 절차 21](#_Toc48312157)

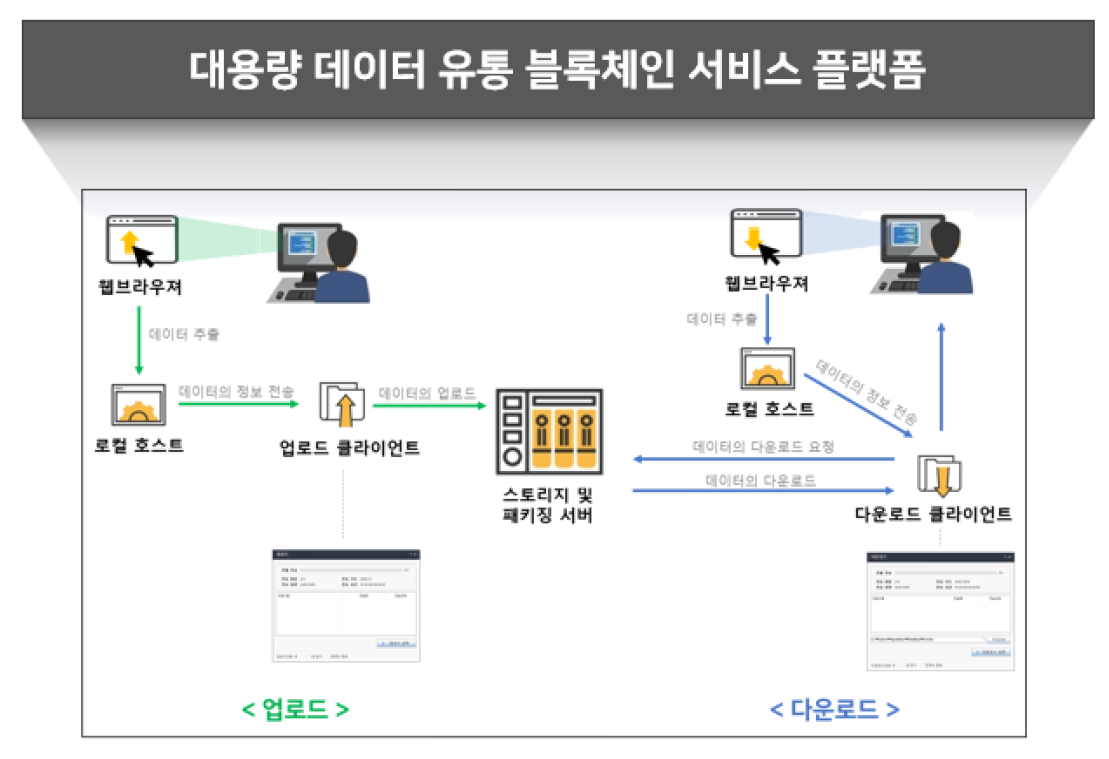
# 개요

데이터 유통 블록체인 서비스는 대용량 데이터에 대한 실시간 유통을 가능하게 하는 블록체인 서비스이고, 이 서비스를 위한 API 규격을 정의하는 문서이다.

## 문서의 범위

블록체인 서비스로는 대용량 데이터 패키징 서비스인 BCMC와 디지털 콘텐츠 유통 서비스인Circler가 있으며, API는 두 서비스에 모두 적용 가능하다.

# 서비스 설계 개념

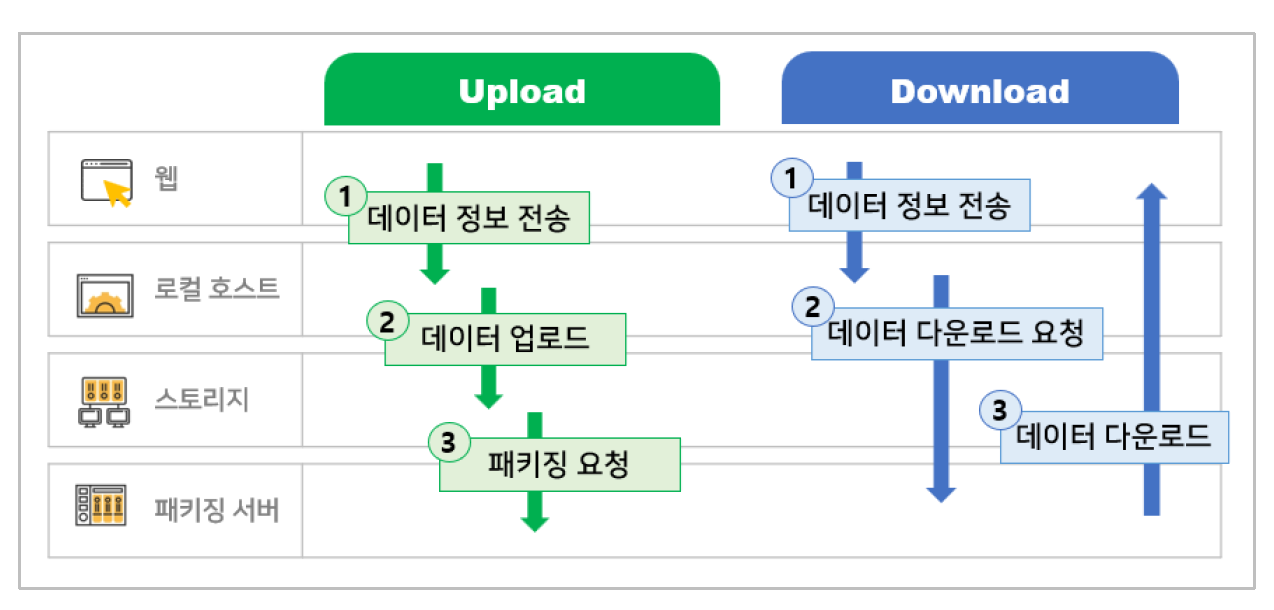


**그림 1. 대용량 데이터 유통 블록체인 서비스 응용 프로그램 전체 개념도**

* 대용량 데이터 유통 블록체인 서비스 응용 프로그램 시스템

-. 실시간 대용량 데이터 유통 온-오프 하이브리드 블록체인 기술이 적용된 전체 시스템 중 대용량 데이터 유통 블록체인 서비스 플랫폼의 연동을 위한 응용 프로그램 시스템에 대해서 그림1에 나타낸다. 이 시스템은 실제 블록체인 서비스를 준비하기 위해서 필요한 요소기술을 제공하는 역할을 수행한다.

-. 그림 2에서는 서비스 플랫폼의 GUI를 통해서 파일을 등록하고 업로드 후 패키징 및 다운로드에 이르기까지의 시스템 동작 흐름을 나타내고 있다. 대용량 데이터를 유통시키기 위해 어떻게 1차적인 데이터 처리가 유기적으로 연결되는지 보여주고 있다.



**그림 2. 대용량 데이터 유통 블록체인 서비스 응용프로그램 동작 흐름도**

# 서비스 API

제공되는 API는 HTTP 서비스를 지원하는 웹서비스 프로그램으로 제공되며, 사용 URL에 따라 별도 기능이 제공된다. 서버에서 API 구분을 통한 Route 처리를 하지 않고, 모든 API는 <http://127.0.0.1:54777>로 POST/GET 방식으로 전송하며 BODY에 채워진 내용을 기반으로 기능 분기하여 처리한다.

## 접속 로그인

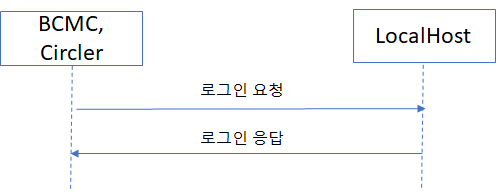
|  |  |
| --- | --- |
| **구분** | **설명** |
| 전송URL | http://127.0.0.1:54777 |
| 전송방식 | POST 방식 |
| 전송파라메터 | k=address ,127.0.0.1,55442,password   |  |  |  | | --- | --- | --- | | Name | 구분 | Value | | address | 필수 | ethereum account 정보 | | password | 필수 | 사용자 로그인 password | |
| 응답코드 | Http 응답코드 사용 |
| 응답값 | 성공 : 1  실패 : 0 |
| 비고 | Body의 내용을 ipfs config 함수에 그대로 전달함. |
| 대상 | BCMC, Circler |

[예]

(전송)

k:asdfasfasdasp3asdklj;as,127.0.0.1,55442,p@ssw0rd

[시퀀스]



## 업로드 파일 선택

|  |  |
| --- | --- |
| **구분** | **설명** |
| 전송URL | http://127.0.0.1:54777 |
| 전송방식 | GET 방식 |
| 전송파라메터 | f=c |
| 응답코드 | Http 응답코드 사용 |
| 응답값 | Json format을 사용   |  |  |  | | --- | --- | --- | | Name | 구분 | Value | | result | 필수 | 결과 코드  • 정상 : S  • 오류 : E | | filepath | 필수 | 파일 경로(성공시) | | filesize | 필수 | 파일 크기(성공시) | |
| 대상 | BCMC |

[예]

(전송)

f=c

(응답)

-. 성공시 :

{

"result":"S",

"filepath":"E:\\contents\\20190904\\1567685394.jpg",

"filesize":"254254"

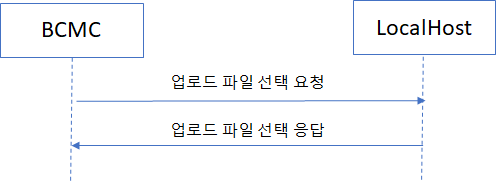
}

-. 실패시 :

{

“result” : “E”

}



## 업로드 폴더 선택

|  |  |
| --- | --- |
| **구분** | **설명** |
| 전송URL | http://127.0.0.1:54777 |
| 전송방식 | GET 방식 |
| 전송파라메터 | f=f |
| 응답코드 | Http 응답코드 사용 |
| 응답값 | Json format을 사용   |  |  |  | | --- | --- | --- | | Name | 구분 | Value | | result | 필수 | 결과 코드  • 정상 : S  • 오류 : E | | folderpath | 필수 | 폴더명 | | fileinfo | 필수 | 요청 목록   |  |  | | --- | --- | | Name | Value | | filepath | 파일 경로(성공시) | | filesize | 파일 크기(성공시) | | |
| 대상 | BCMC |

[예]

(전송)

f=f

(응답)

-. 성공시 :

{

"result":"S",

"folderpath":"E:\\contents\\test",

"fileinfo":[{

"filepath":"E:\\contents\\test\\1568005627.jpg",

"filesize":"706293"

},

{

"filepath":"E:\\contents\\test\\1568017286.jpg",

"filesize":"293572"

}]

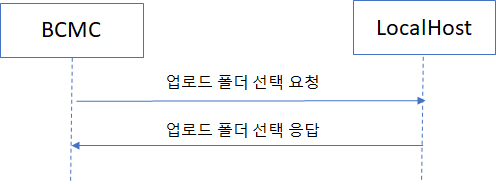
}

-. 실패시 :

{

“result” : “E”

}



## 대표이미지 단일파일 선택

|  |  |
| --- | --- |
| **구분** | **설명** |
| 전송URL | http://127.0.0.1:54777 |
| 전송방식 | GET, POST(파일전송) 방식 |
| 전송파라메터 | i=s |
| 응답코드 | Http 응답코드 사용 |
| 응답값 | Json format을 사용   |  |  |  | | --- | --- | --- | | Name | 구분 | Value | | result | 필수 | 결과 코드  • 정상 : S  • 오류 : E | | filepath | 필수 | 파일 경로(성공시) | | fileinfo | 필수 | 파일 정보(성공시) | | filesize | 필수 | 파일 크기(성공시) | |
| 대상 | BCMC |

[예]

(전송)

i=s

(응답)

-. 성공시 :

{

"result":"S",

"filepath":"E:\\contents\\20190904\\1567685394.jpg",

”fileinfo”:”320,240”,

"filesize":"254254“

}

-. 실패시 :

{

“result” : “E”

}



## 대표이미지 다중파일 선택

|  |  |
| --- | --- |
| **구분** | **설명** |
| 전송URL | http://127.0.0.1:54777 |
| 전송방식 | GET, POST(파일전송) 방식 |
| 전송파라메터 | i=m |
| 응답코드 | Http 응답코드 사용 |
| 응답값 | Json format을 사용   |  |  |  | | --- | --- | --- | | Name | 구분 | Value | | result | 필수 | 결과 코드  • 정상 : S  • 오류 : E | | fileinfo | 필수 | (성공시)   |  |  | | --- | --- | | Name | Value | | filepath | 파일 경로 | | filename | 파일 이름 | | fileinfo | 파일 정보 | | filesize | 파일 크기 | | |
| 대상 | BCMC |

[예]

(전송)

i=m

(응답)

-. 성공시 :

{

"result":"S",

"fileinfo":[

{

"filepath":"E:\\contents\\20190904\\1567685394.jpg",

"filename":"/data/20200610/1592066303.jpg",

"fileinfo":"1018,684",

"filesize":"254254"

},

{

"filepath":"E:\\contents\\20190904\\1567729055.jpeg",

"filename":"/data/20200610/1592006579.jpeg",

"fileinfo":"1768,1262",

"filesize":"1214390"

}]

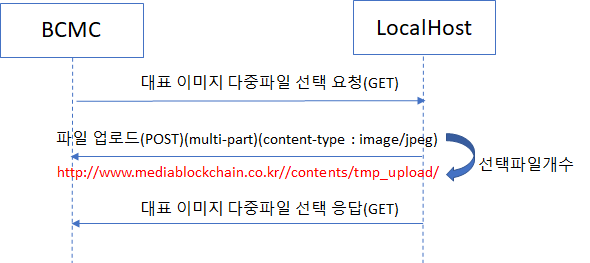
}

-. 실패시 :

{

“result” : “E”

}



## Form 콘텐츠 업로드

|  |  |
| --- | --- |
| **구분** | **설명** |
| 전송URL | http://127.0.0.1:54777 |
| 전송방식 | POST 방식 |
| 전송파라메터 | u={콘텐츠정보}   |  |  |  | | --- | --- | --- | | Name | 구분 | Value | | content\_no | 필수 | 복합 콘텐츠 번호 | | contents\_file\_no | 필수 | 콘텐츠 번호 | | user\_id | 필수 | 사용자 아이디 | | next\_val | 필수 | 업로드에 사용할 검증용 코드값  (패키징 진행 상태 및 파일의 다운로드시 사용되는 정보) | | c\_name | 필수 | 콘텐츠 이름 | | file\_no | 필수 | 현재 업로드하는 파일의 순번  (전체 파일 중 n번째 파일을 업로드 하겠다는 표기) | | file\_size | 필수 | 파일 크기 | | file\_path | 필수 | 파일 경로 | | c\_type | 필수 | 콘텐츠 형식 | | file | 필수 |  | | dev | 필수 | 개발진행버전여부 | |
| 응답코드 | Http 응답코드 사용 |
| 응답값 | Json format을 사용   |  |  |  | | --- | --- | --- | | Name | 구분 | Value | | result | 필수 | 결과 코드  • 정상 : 1  • 오류 : 0 | |
| 대상 | BCMC |

[예]

(전송)

u={

"contents\_no":"812",

"contents\_file\_no":"1480",

"user\_id":"9bcecd9085fae8fa787ac3f3bd3c2f25a90e0610",

"next\_val":"9610a54a-3497-4d9d-be71-00597bcc7735",

"c\_name":"aaaa",

"file\_no":"1",

"file\_size":"1214390",

"file\_path":"E:\\contents\\20190904\\1567729055.jpeg",

"c\_type":"v",

"file":"",

"dev":"0"

}

(응답)

-. 성공시 :

{

"result":"1"

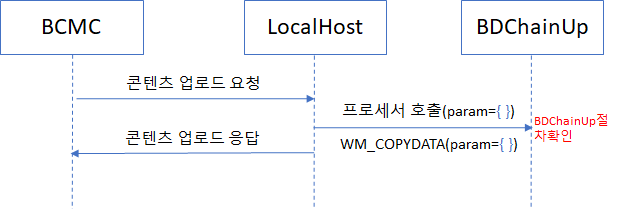
}

-. 실패시 :

{

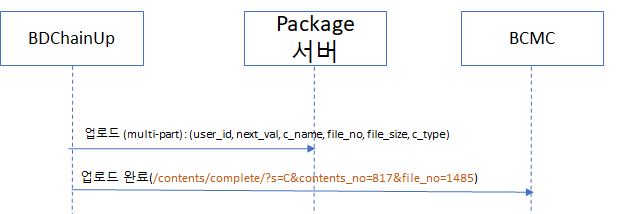
“result” : “0”

}



## BDChainUp 프로세스

|  |  |
| --- | --- |
| **구분** | **설명** |
| 전송URL | http://127.0.0.1:54777 |
| 전송방식 | POST 방식 |
| 전송파라메터 | |  |  |  | | --- | --- | --- | | Name | 구분 | Value | | user\_id | 필수 | 사용자 아이디 | | next\_val | 필수 | 업로드에 사용할 검증용 코드값  (패키징 진행 상태 및 파일의 다운로드시 사용되는 정보) | | c\_name | 필수 | 콘텐츠 이름 | | file\_no | 필수 | 현재 업로드하는 파일의 순번  (전체 파일 중 n번째 파일을 업로드 하겠다는 표기) | | file\_size | 필수 | 파일 크기 | | c\_type | 필수 | 콘텐츠 형식 |  * 업로드 요청시 * 업로드 완료시   /contents/complete/?s=C&contents\_no={콘텐츠번호}&file\_no={파일번호} |
| 응답코드 | Http 응답코드 사용 |
| 응답값 |  |
| 대상 | BDChainUp |



## 콘텐츠 다운로드(BCMC)

|  |  |
| --- | --- |
| **구분** | **설명** |
| 전송URL | http://127.0.0.1:54777 |
| 전송방식 | POST 방식 |
| 전송파라메터 | d={콘텐츠정보}   |  |  |  | | --- | --- | --- | | Name | 구분 | Value | | content\_no | 필수 | 복합 콘텐츠 번호 | | contents\_file\_no | 필수 | 콘텐츠 번호 | | user\_id | 필수 | 사용자 아이디 | | next\_val | 필수 | 업로드에 사용할 검증용 코드값  (패키징 진행 상태 및 파일의 다운로드시 사용되는 정보) | | file\_name | 필수 | 파일 이름 | | file\_no | 필수 | 현재 업로드하는 파일의 순번  (전체 파일 중 n번째 파일을 업로드 하겠다는 표기) | | file\_size | 필수 | 파일 크기 | | del\_yn | 필수 | 다운로드 후 삭제 여부  y : 즉시 삭제처리 (추가 다운로드 불가능)  n : 삭제하지 않음(1주일 후 삭제 처리, 재 다운로드 가능) | | total\_cnt | 필수 | 다운로드 받을 콘텐츠 개수 | | dev | 필수 | 개발진행버전여부 | |
| 응답코드 | Http 응답코드 사용 |
| 응답값 | Json format을 사용   |  |  |  | | --- | --- | --- | | Name | 구분 | Value | | result | 필수 | 결과 코드  • 정상 : 1  • 오류 : 0 | |
| 대상 | BCMC |

[예]

(전송)

d={

"contents\_no":"813",

"contents\_file\_no":"1481",

"file\_name":"1567755741.jpg",

"file\_size":"706293",

"user\_id":"9bcecd9085fae8fa787ac3f3bd3c2f25a90e0610",

"next\_val":"e16e5c91-c3db-484f-b0d6-b642ad590d74",

"file\_no":"1",

"del\_yn":"N",

"total\_cnt":"1",

"dev":"0"

}

(응답)

-. 성공시 :

{

"result":"1"

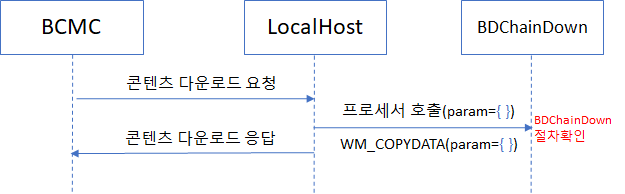
}

-. 실패시 :

{

“result” : “0”

}



## BDChainDown 절차

|  |  |
| --- | --- |
| **구분** | **설명** |
| 전송URL | 프로세스통신(LocalHost-> BDChainDown) |
| 전송방식 | WM\_COPY 방식 |
| 전송파라메터 | * WM\_COPY   3.8. 콘텐츠 다운로드의 전송 파라미터를 그대로 BDChainDown이 그대로 전송 받음**(3.8의 콘텐츠다운로드 전송파라미터 참고)**   * 다운로드 요청시  |  |  |  | | --- | --- | --- | | Name | 구분 | Value | | user\_id | 필수 | 사용자 아이디 | | next\_val | 필수 | 업로드에 사용할 검증용 코드값  (패키징 진행 상태 및 파일의 다운로드시 사용되는 정보) | | file\_no | 필수 | 현재 업로드하는 파일의 순번  (전체 파일 중 n번째 파일을 업로드 하겠다는 표기) | | del\_yn | 필수 | 다운로드 후 삭제 여부  y : 즉시 삭제처리 (추가 다운로드 불가능)  n : 삭제하지 않음(1주일 후 삭제 처리, 재 다운로드 가능) |  * 콘텐츠 정보 요청   /contents/getContentsJson/?s=C&contents\_no={콘텐츠번호}&save\_path={저장디렉토리명}   * 콘텐츠 CCID 등록 요청   /contents/setContentsJson/contents\_no={콘텐츠번호}&result={아래시퀀스 참조} |
| 응답코드 | Http 응답코드 사용 |
| 응답값 | Json format을 사용(콘텐츠 정보요청, CCID 등록 후 응답값)   |  |  |  | | --- | --- | --- | | Name | 구분 | Value | | result | 필수 | 결과 코드  • 정상 : 1  • 오류 : 0 | |
| 대상 | BCMC |



## 콘텐츠 다운로드(Circler)

|  |  |
| --- | --- |
| **구분** | **설명** |
| 전송URL | http://127.0.0.1:54777 |
| 전송방식 | POST 방식 |
| 전송파라메터 | c={콘텐츠정보}   |  |  |  | | --- | --- | --- | | Name | 구분 | Value | | content\_ccid | 필수 | 상품 CCID | | contents\_version | 필수 | 상품 CCID version | | account\_id | 필수 | 사용자 아이디 | | product\_id | 필수 | 제작자 아이디 | | sub\_path | 필수 | 사용안함 | | payment\_id | 필수 | 사용안함 | | file\_path | 필수 | 파일 경로 | | file\_size | 필수 | 파일 크기 | | file\_type | 필수 | 파일 형식 | |
| 응답코드 | Http 응답코드 사용 |
| 응답값 | Json format을 사용   |  |  |  | | --- | --- | --- | | Name | 구분 | Value | | result | 필수 | 결과 코드  • 정상 : 1  • 오류 : 0 | |
| 대상 | Circler |

[예]

(전송)

c={

"contents\_ccid":"---","contents\_version":"---","account\_id":"---","product\_id":"---","sub\_path":"","payment\_id":"---","file\_path":"---","file\_size":"---","file\_type":"T"

}

(응답)

-. 성공시 :

{

"result":"1"

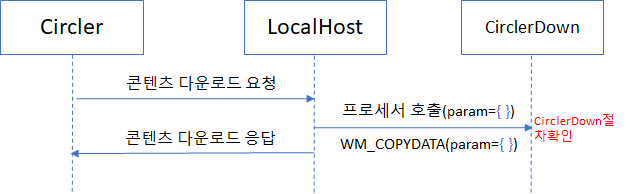
}

-. 실패시 :

{

“result” : “0”

}



## CirclerDown 절차

|  |  |
| --- | --- |
| **구분** | **설명** |
| 전송URL | 프로세스통신(LocalHost-> CirclerDown) |
| 전송방식 | WM\_COPY 방식 |
| 전송파라메터 | * WM\_COPY   3.10. 콘텐츠 다운로드의 전송 파라미터를 그대로 CirclerDown이 그대로 전송 받음**(3.10의 콘텐츠다운로드 전송파라미터 참고)**   * 라이선스 확인**(onChainAPI 참조)**   -. <http://onchain_server:55442/dsa/sign>  -. 전송 파라미터  inData=contents\_ccid\_val(-----)  -. 응답  {  "resultCode" : 0,  "signature": "b/wsU3rnQM4GqpAvzmiWkocx7RMC0HkTcXIaDUegjMdfyOMjt5UXVYyKRaGMImCfCKWfirjIQWk93kTxCWeCCiIxMTEi"  }   * 파일 다운로드 요청**(IPFS API 참고)**   (/ccfs/%s/%s/ -p %s –o %s) , contents\_ccid, contents\_version, payment\_id, FILEPATH |
| 대상 | Circler |

