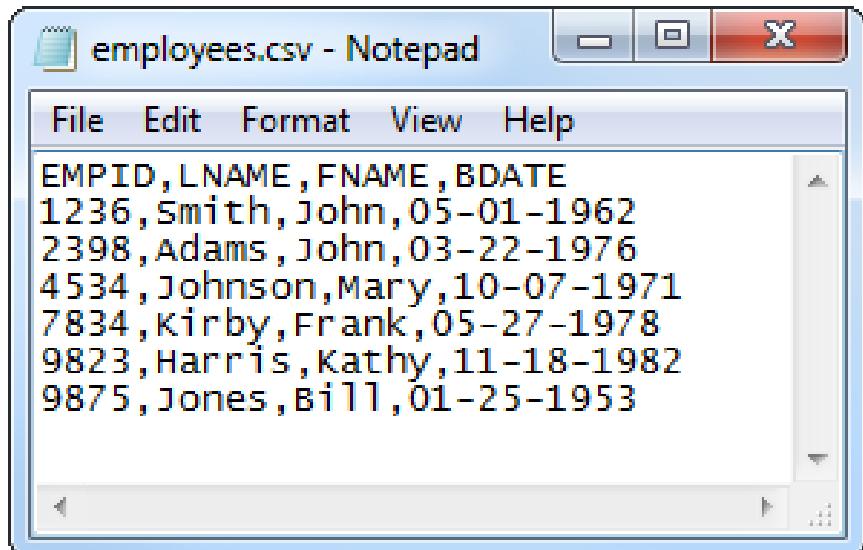


Flat-File Developer`s Guide - 10.11

- References
 - <https://www.netinbag.com/ko/internet/what-is-a-flat-file.html> (플랫 파일이란?)
 - https://documentation.softwareag.com/webmethods/integration_server/pie10-11/10-7_Flat_File_Schema_Developers_Guide.pdf (Software AG 공식문서)
 - <https://kylenoh-developer-note.tistory.com/entry/webMethods-FlatFile> (블로그)
 - <https://www.youtube.com/watch?v=ETIHrKnQt4g> (Youtube - Tech Lightning)

- Flatfile (플랫파일) 이란
 - 플랫파일은 아무런 구조적 상호관계가 없는 레코드 들이 들어 있는 파일
 - 하나의 데이터 모델을 플레인 텍스트 파일로 인코딩 하여 저장, 일종의 데이터베이스
 - 데이터베이스에 순차적으로 저장되고 액세스되는 정보 모음으로, 정보를 비 구조적 방식으로 저장하기 위해 종종 만들어 진다.

- 예를 들어 CSV (Comma Separated Value) 시트는 각 필드가 다음 필드와 쉼표로 구분
- 플랫 파일 데이터 모델을 기능적으로 사용하려면 레코드에 대한 모든 관련 정보를 동일한 파일에 저장해야 한다.



CSV 파일 예시

- 필드는 쉼표로 구분, 하나의 레코드는 한 행으로 구분
- 레코드들을 포함하긴 하지만, 한 줄의 크기가 얼마인지, 제목 또는 그 문서를 포맷하기 위해 프로그램이 사용할 수 있는 목차 등에 관한 다른 정보는 없다. (구조데이터, 즉 메타데이터를 포함하지 않는다)
- 관리적 측면에서 비효율적
- 작은 크기의 도서 관리 프로그램이나 일정 관리 프로그램, 특히 주소록 프로그램 등은 플랫 파일을 빈번히 이용하기도 함

- **Flatfile의 구조 & webMethods에서 Flatfile**

필드 (Field)	가장 작은 데이터 구조, 원자단위의 데이터 (= String) ≈ 컬럼
컴포지트 (Composite)	필드의 결합체, 하나 이상의 필드로 결합 되어 컴포지트 내 필드를 서브필드 (Subfields)를 이라 지칭 (= Document)
레코드 (Record)	Segment라고도 지칭 되며, 하나의 레코드를 뜻함 (= Document List) ≈ 테이블
구분문자	별도의 독립 영역 사이의 경계를 지정하기 위한 하나 이상의 문자, 각 필드,

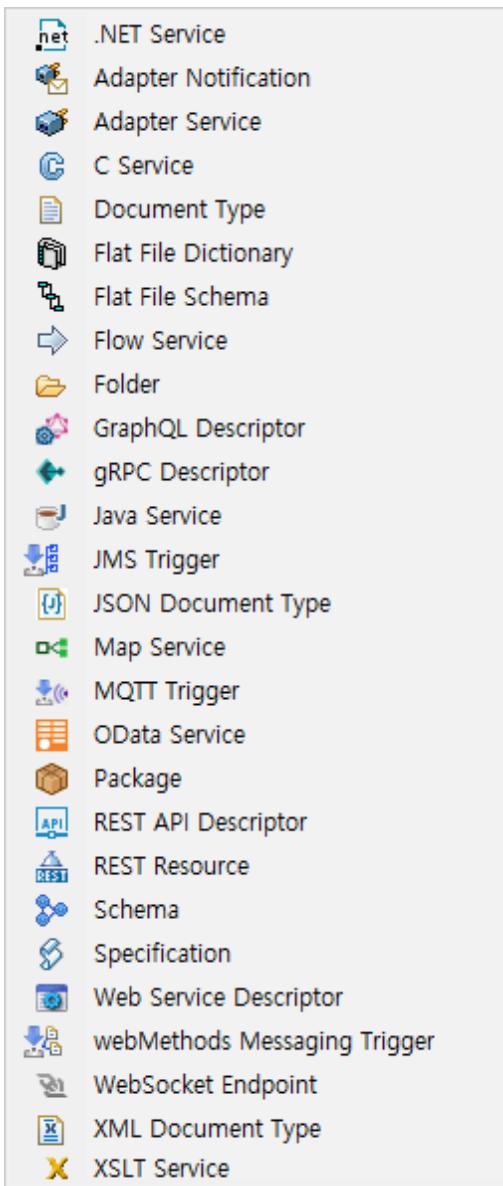
(Delimiter)

컴포지트, 레코드별 구분문자가 있음.



record delimiter = !
field delimiter = *
subfield delimiter = +

FlatFile 구조



webMethods Integration Server 10.11

- webMethods 에서는 **Flat File Dictionary**와 **Schema** 생성 기능 제공

- **Flat File Schema** 란

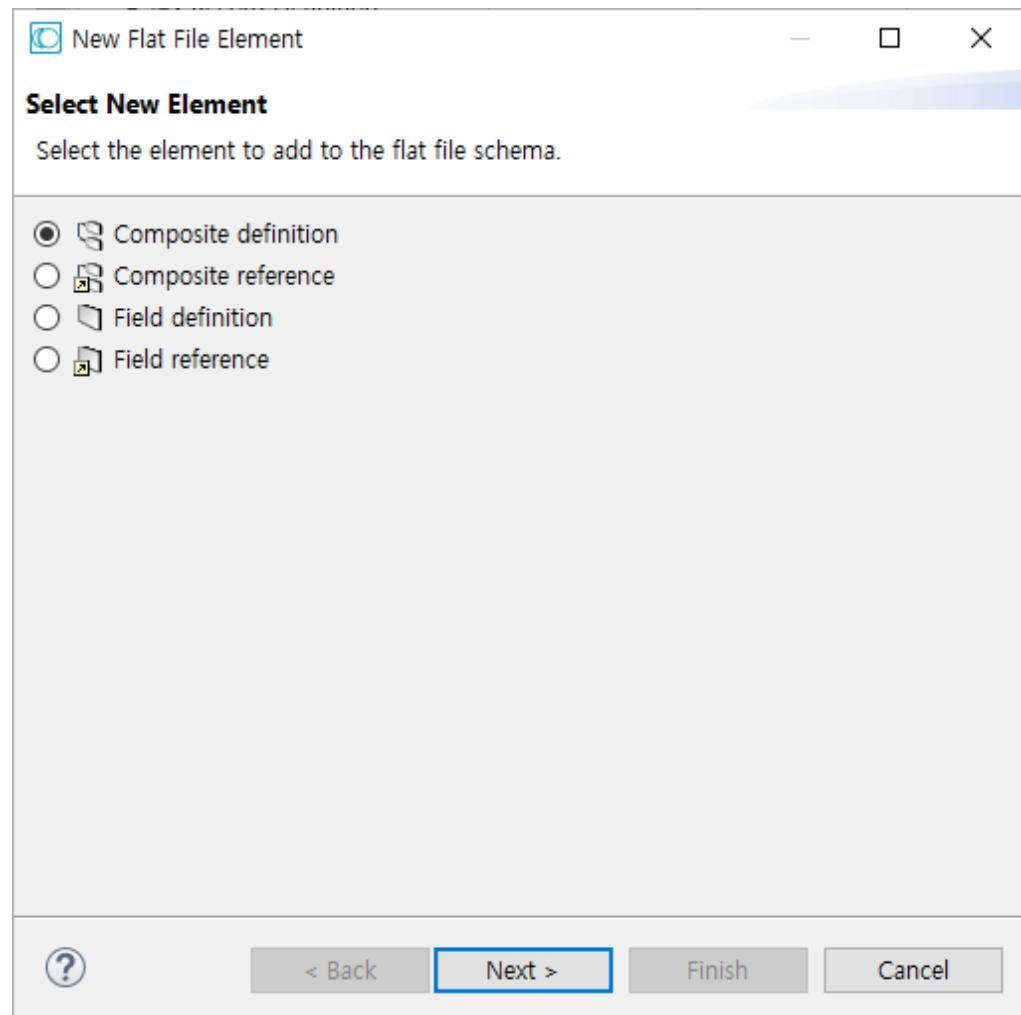
- 스키마란 데이터베이스의 구조와 제약 조건에 관한 전반적인 명세를 기술한 메타데이터의 집합
- 플랫파일 스키마는 플랫파일을 구문 분석 또는 생성 지침이 포함 된 Blueprint ≈ 스키마
- webMethods에서 플랫파일을 사용하기 위해 플랫파일 스키마를 생성하여, 레코드를 식별, 해당 레코드를 분리하여 사용



- ① Inbound : 플랫파일 스키마를 통해 인바운드 플랫파일의 구문을 분석하고, 검증
- ② Outbound : 플랫파일 스키마를 통해 Native Service (Flow Service) 내에서 처리 되는 Pipeline데이터를 Flatfile 형식으로 변환하여 전송

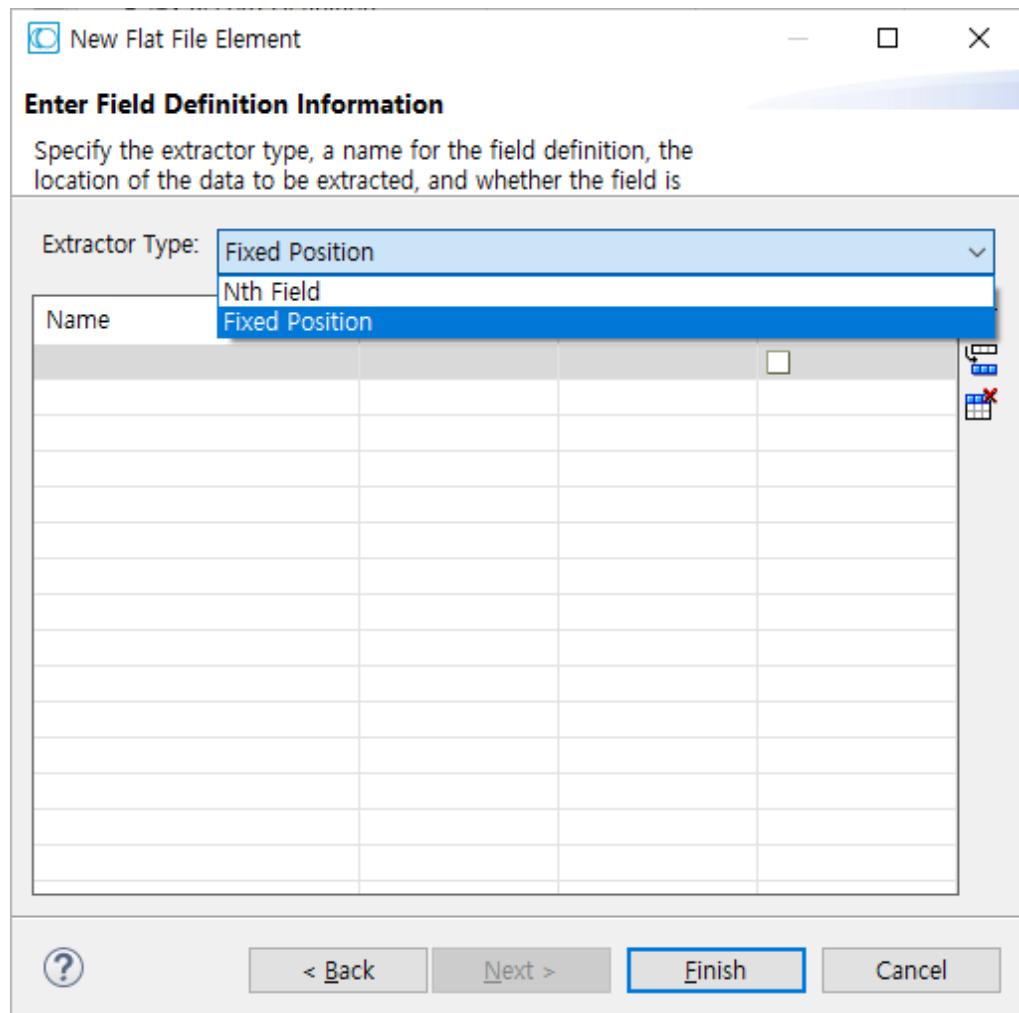
- **Flat File Dictionary** 란

- 플랫 파일 스키마의 레코드 정의를 사전에 정의해두고 하나의 개체로서 재사용성을 늘리기 위함
- 보통 Dictionary 생성 후 해당 Dictionary를 사용하여 Schema 생성
- Record Definition 부터 정의
 - Record 생성 후 Composite 또는 Field 생성



- 생성은 Definition/Reference 로 생성 가능 (정의 또는 기존 Element 재사용)

- Field 정의



- Fixed Position은 시작과 끝을 길이로 제한하는 방식
 - Nth Field는 스키마의 정의에 따라 Field를 구분하는 방식
-
- Flat File 사용 튜토리얼 1 - Dictionary 를 사용하여 Schema 생성
 - STEP 1. Dictionary 생성

Dictionary

Name	Referring To	Dictionary	Type
FlatFile_DEMO.ff:Dictionary			Dictionary
Record Definition			
EMPLOYEES			Record Definition
EMPLOYEE_ID			Field Definition
FIRST_NAME			Field Definition
LAST_NAME			Field Definition
EMAIL			Field Definition
PHONE			Field Definition
HIRE_DATE			Field Definition
MANAGER_ID			Field Definition
JOB_TITLE			Field Definition
FLAG			Field Definition
Composite Definition			
Field Definition			

Dictionary 생성 예시

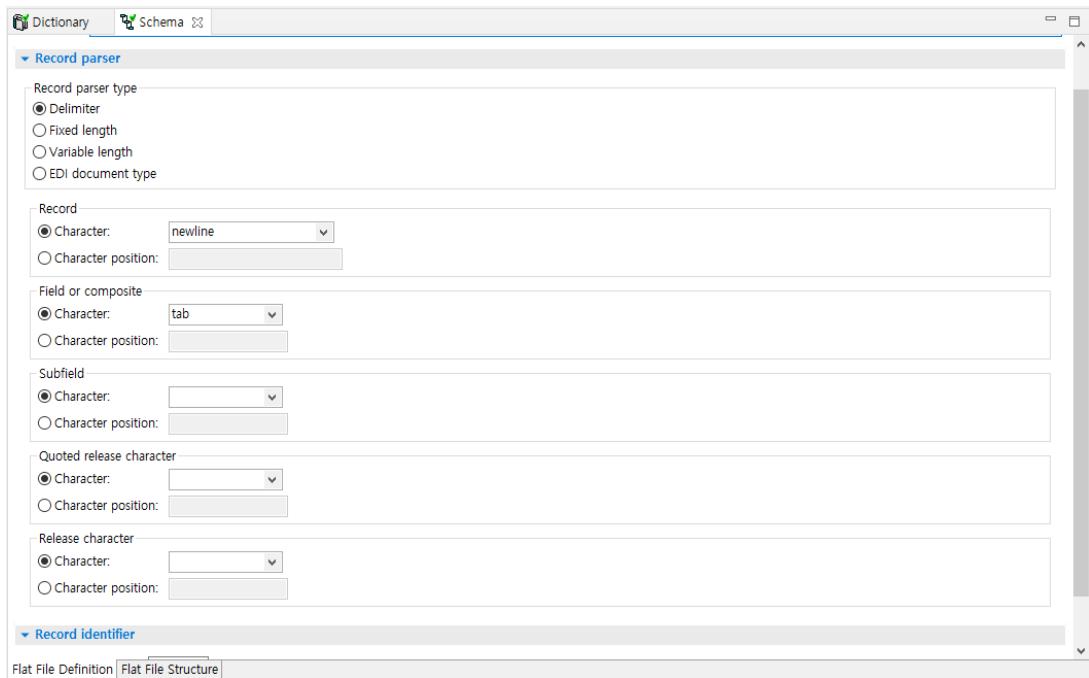
- STEP 2. Schema 생성

Dictionary Schema

Name	Referring To	Dictionary	Type
FlatFile_DEMO.ff:Sc1			Schema Definition
EMPLOYEES	EMPLOYEES	FlatFile_DEMO.ff:Diction...	Record Reference
EMPLOYEE_ID			Field Definition
FIRST_NAME			Field Definition
LAST_NAME			Field Definition
EMAIL			Field Definition
PHONE			Field Definition
HIRE_DATE			Field Definition
MANAGER_ID			Field Definition
JOB_TITLE			Field Definition
FLAG			Field Definition

Schema 생성 예시

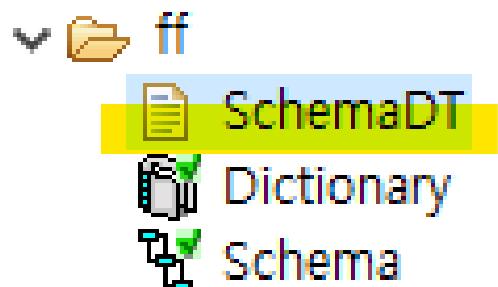
- STEP 3. Record Parser 설정



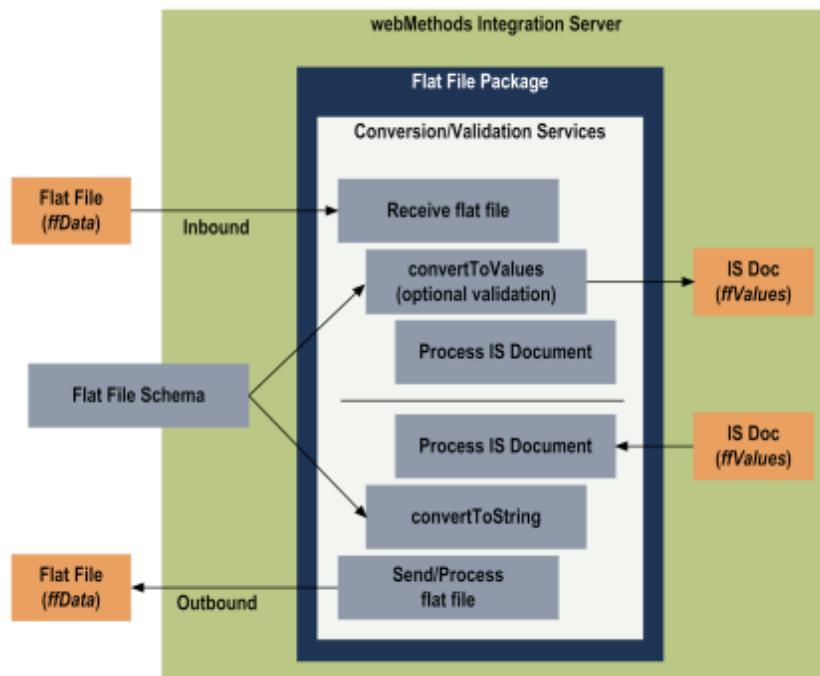
- Delimiter : 구분문자로 FlatFile 구문 분석
- Fixed length : 레코드 Length가 고정일 때 구문분석
- Variable Length : 각 레코드 앞에 2바이트의 레코드 길이를 나타내는 구문이 있을 때, 가변길이 레코드 파서 라고 함
- EDI Document Type : EDI 문서 표준일 때 사용, ANSI X12 및 UN/EDIFACT 등
 - EDI 모듈이 필요 (* webMethods Module For EDI Installation and User's Guide 참고)
- **STEP 4. Record identifier (?)** → 정확한 의미인지는 확실치 않음..
 - Start at Position : 시작 위치 지정
 - Nth field : N번째 필드부터
- **STEP 5. Document Type 생성 - Schema > Structure > 우측 위 Create Document Type 클릭**

Name	Referring To	Dictionary	Type
FlatFile_DEMO.ff:Schema			Schema Definition
EMPLOYEES	EMPLOYEES	FlatFile_DEMO.ff:Dictionary	Record Reference
EMPLOYEE_ID			Field Definition
FIRST_NAME			Field Definition
LAST_NAME			Field Definition
EMAIL			Field Definition
PHONE			Field Definition
HIRE_DATE			Field Definition
MANAGER_ID			Field Definition
JOB_TITLE			Field Definition
FLAG			Field Definition

- STEP 6. 생성 된 Document Type은 Service 의 Input / Output 등으로 활용 가능



- Flat File 사용 튜토리얼 2 - 플랫 파일 Sending / Receiving (→ getFile 사용)

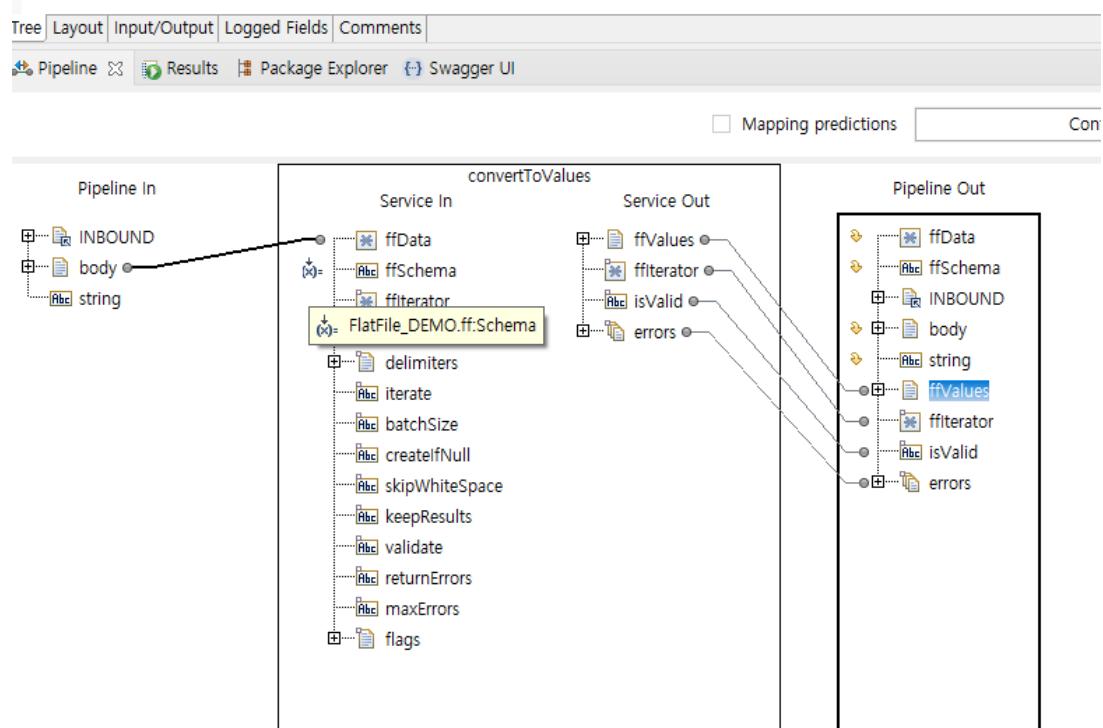


- In/Outbound 호출 방식
 - HTTP POST 호출 (Content-type : application/x-wmflatfile)
 - FTP
 - File Polling
 - E-Mail 등등
- pub.file:getFile, pub.flatFile:convertToValues 서비스를 사용한 Receive 예시

```

➡ pub.file:getFile
➡ pub.string:bytesToString
➡ pub.flatFile:convertToValues
↳ MAP

```



○ 결과 값 예시

Name	value
INBOUND	
recordWithNoID	
recordWithNoID[0]	
recordWithNoID[1]	
recordWithNoID[2]	
recordWithNoID[3]	
recordWithNoID[4]	
recordWithNoID[5]	
recordWithNoID[6]	
recordWithNoID[7]	

- Receive 된 DocumentList를 Roop로 Document로 & 세부 데이터 Handling
 - ffValues/recordWithNoID/recordWithNoID 형식으로 받아진다.