

어플리케이션 영향도 분석 솔루션

『DATAWARE™ AP#』

EN-CORE Inc. is "Total Data Service Provider." We, as one of the leading data scientists groups in Korea, are offering a wide-range of services such as consulting, education, solutions, and technology information based on data ranging from planning the corporation IT systems to analyzing, designing, developing, transitioning, managing, performing governance, and data servicing.



dataware
impact
analysis

en-core co.,ltd.

04130 | 8F,163, Mapo-daero, Mapo-gu, Seoul, Republic of Korea

office +82 2 754 7301 | fax +82 2 754 7305 | web www.en-core.com



- 01. 제안 배경**
- 02. 솔루션 개요**
- 03. 솔루션 구성**
- 04. 구축사례**
- 05. 데이터 통합 관리 솔루션**

01 제안 배경

현재 TREND + 인식의 변화

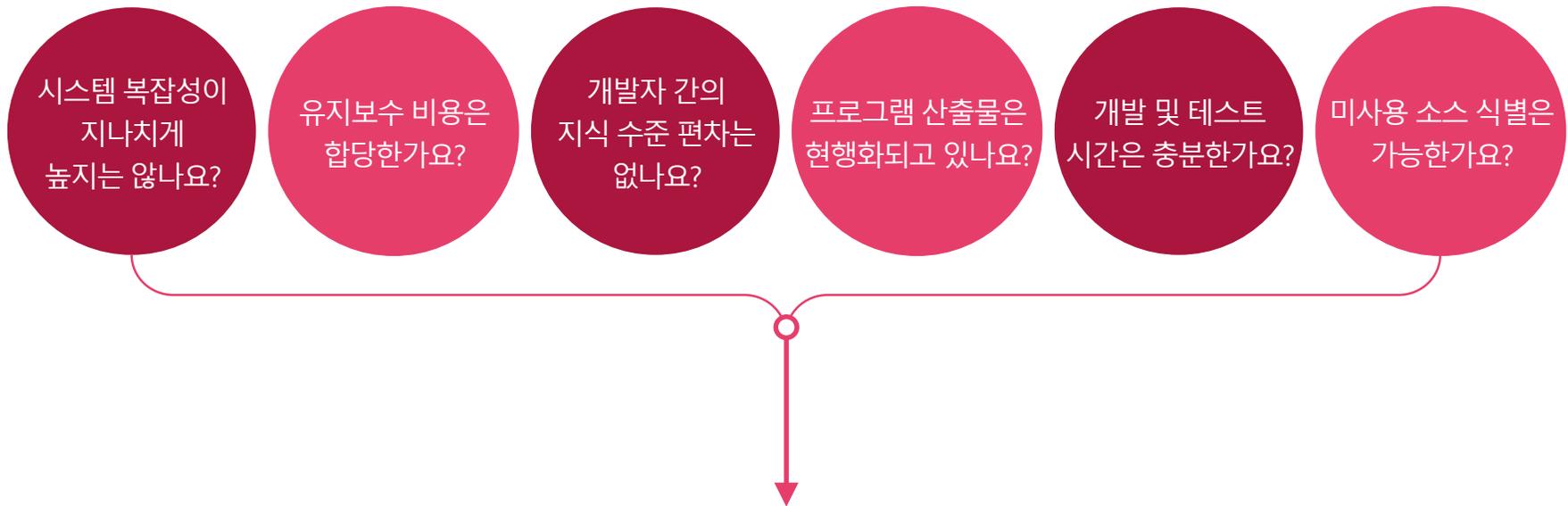


관리

+

활용

우리 기업은 어플리케이션 변경에 따른 위험성을 **최소화**하고 있습니까?

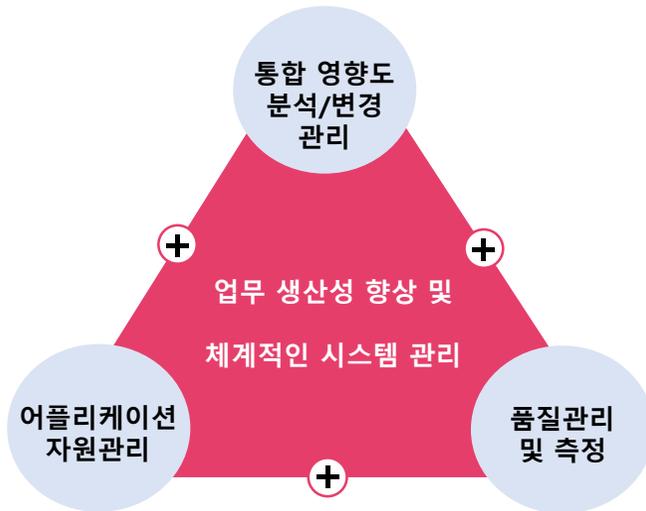


시스템 장애 예방, 소스 품질 향상, 개발 프로세스 관리를 지원하는
통합 영향도 분석 및 변경관리가 프로젝트 성공에 직결

“어플리케이션 영향도 분석 시스템 필요”

어플리케이션 영향도 분석 솔루션 구축 필요성

시스템 변경에 대한 효과적인 영향도 분석 프로세스를 수립하여 **변경 범위 사전 인지 및 장애 발생 요인을 최소화하고, IT 운영 안정성과 운영 효율성 극대화**



- 1 다양한 언어로 작성된 어플리케이션 소스와 메타데이터를 자동으로 수집 및 파싱해, 프로그램 구조정보와 상호 연관정보 시각화
- 2 특정 시점의 변경/추가/삭제된 소스 이력 정보를 조회하여 개발 및 운영 효율성 증대
- 3 프로그램 변경 시 M/M 산정이 가능하도록 코드 크기(LOC), 복잡도, 로직 Depth 등 정량적 분석 자료 제공

02

솔루션 개요

DATAWARE™ AP# - 데이터 거버넌스, 어플리케이션 영향도 분석 시스템 -



“최고의 분석 엔진 성능과 안정성,
메타정보 활용 및 확장이 용이한 제품으로
관련 업계 1위”

- 통합 Repository로 DB 카탈로그 정보와 어플리케이션 소스를 수집 및 분석
- 전체 시스템의 정확한 구조 및 상호 연관관계에 대한 정보 제공
- 어플리케이션 변경/품질 관리와 유지 보수 체계 지원



소스 및 DB 변경에 따른 위험 예방, AS-IS 분석 비용 절감, 업무 효율성 향상




특장점

01어플리케이션 소스
영향도 분석

- 소스 기본 정보, 통계 정보, 사용 객체 정보, 파일 사용도 정보, 사용된 SQL 정보, 클래스 다이어그램 정보 분석
- Include/Import 정보, 함수 영향도, 파일 영향도 분석
- 소스 내의 다양한 로직 정보를 시각화된 플로우 차트로 제공

02DB 및 어플리케이션
상호 영향 분석

- 특정 업무 및 프로그램 종류별 DB 오브젝트 검색 기능
- 소스 객체, 소스 사용 SQL, 연관 문자열, 주석 등 다양한 검색 기능
- 특정 프로그램의 호출 타겟, 피호출 프로그램 목록 검색 기능

03다양한
매트릭스 분석

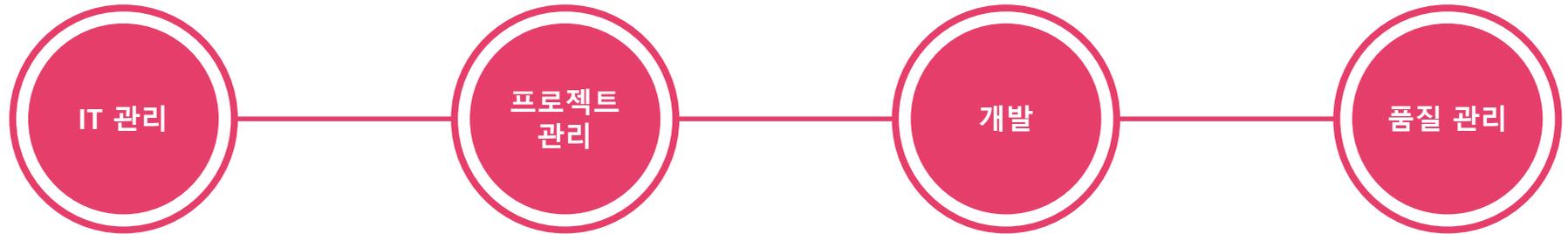
- 어플리케이션 소스 간의 포함 관계 매트릭스 정보
- 소스와 함수 간의 호출 관계 매트릭스 정보
- 클래스와 클래스 간의 상속 관계 매트릭스 정보
- 소스와 테이블 간의 CRUD (Create, Read, Update, Delete) 매트릭스 정보

04통계 분석 및
보고서 제공

- 프로그램 종류별 통계 정보
- DB 오브젝트별 통계 정보
- DB 오브젝트 목록 및 파일 명세 정보
- 파일 사용도 정보
- PDF, 엑셀 양식의 산출물 제공

도입 후 기대효과

기대효과



- 통합 영향도 분석/변경 시각화 정보 및 자동화 문서 제공으로 유지보수비용 절감
- IT 어플리케이션 개발 / 관리 정보의 유지
- Outsourcing에 대한 제어력 향상

- 업무 변화에 대한 적응력 향상
- 업무 이해도 증가
- 프로젝트 통제력 강화
- 프로그램 산출물 현행화

- 다른 개발자의 소스코드 이해
- 전체적인 어플리케이션 Road Map 이해로 개발 생산성 향상
- 개발 및 유지보수 시 발생하는 비효율적인 반복작업의 제거

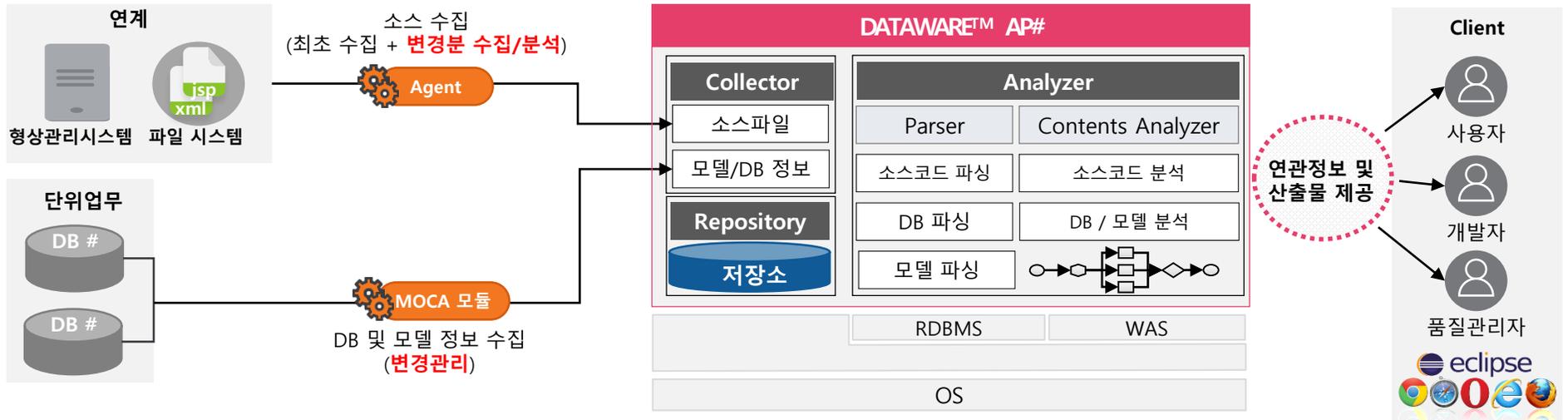
- 정의된 표준화 적용의 용이성 및 개발 표준에 대한 품질 관리 기능 강화로 어플리케이션 품질 향상
- 프로젝트의 위험요소 파악

03

솔루션 구성

시스템 구성도 및 사양

DATAWARE™ AP#은 Collector, Analyzer, Repository, Client 로 구성되어 있으며, WEB UI를 통해 보다 높은 접근성과 활용성을 제공



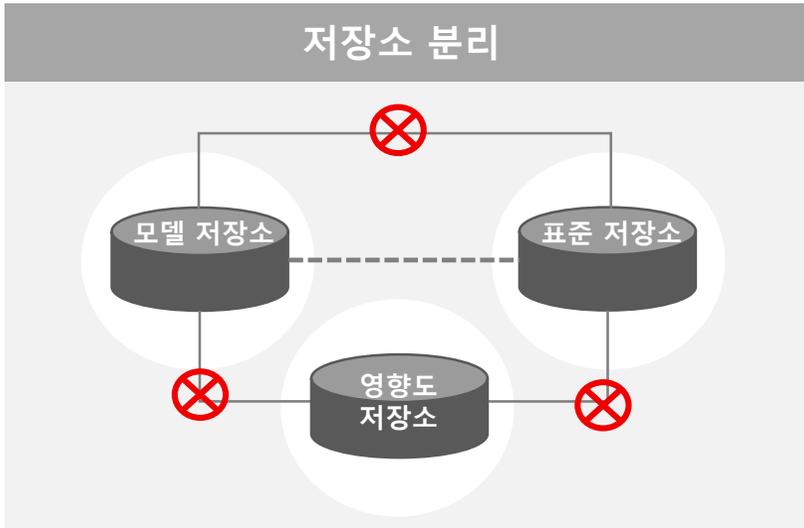
제품 구성	설명
Collector	<ul style="list-style-type: none"> 모델, 데이터베이스 정보, 소스 파일 수집 및 추출 모듈 메타 데이터 수집 배치 작업을 위한 스케줄러 엔진
Analyzer	<ul style="list-style-type: none"> 수집된 소스와 DB 정보, 모델 정보를 파싱한 분석 정보 및 연관 정보 분석
Repository	<ul style="list-style-type: none"> 메타 분석 정보 및 관리 대상 데이터를 저장하는 저장소 역할
Client	<ul style="list-style-type: none"> 분석에 따른 결과 조회 및 리포트 자동 생성 기능을 제공

구분	제안규격	
H/W	CPU/Memory	4Core 이상 / 8Gb 이상
	HDD	Portal 500Mb 이상, Repository 100 Gb 이상
S/W	OS	OS Independent, HTML5 이상 지원 브라우저 분석엔진은 Linux만 지원
	WAS	Tomcat 8.5~9.0, Weblogic 10 이상, Jeus 8 이상 지원
	JRE	JRE1.8
	RDBMS	PostgreSQL Client 14.0

특장점: 통합 Repository 구성

일반 시스템 구성

대다수 시스템은 DB모델 관리, 메타데이터 관리, 어플리케이션 영향도 분석 솔루션이 각각 독립적인 Repository로 구성

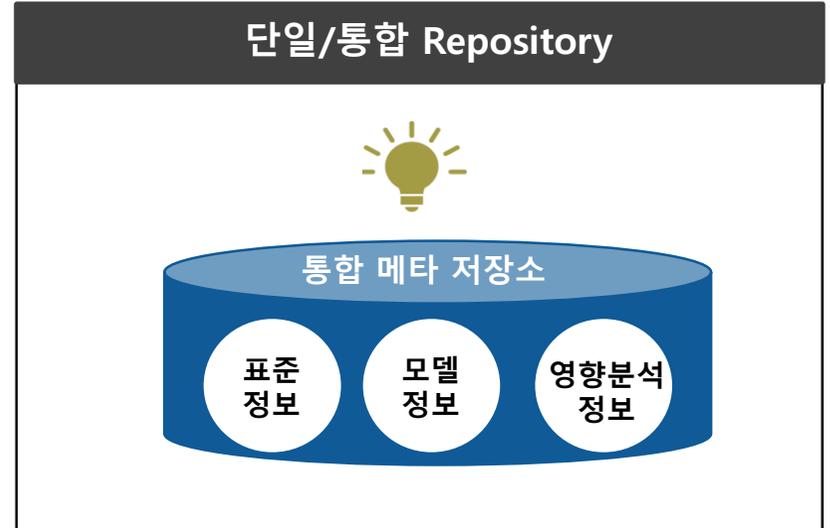


“이질적 벤더 간의 저장소 분리로 데이터 중복 관리 / 호환성 부족 / 실시간성 정보 교환 미흡 / 관리 포인트 증대”

VS

제안사 시스템 구성

제안사는 통합 Repository 기반의 DB모델 관리, 메타데이터 관리, 어플리케이션 영향도 분석 솔루션을 개발 및 제공



“단일/통합 Repository 기반으로 솔루션간 데이터 중첩 최소화 / 유지보수 점점 단일화 / 실시간성 정보교환”

특장점: 메타 데이터 영향도 시각화

통합 Repository를 기반으로 시스템 전체의 변경 영향도 정보를 시각화된 형태로 제공

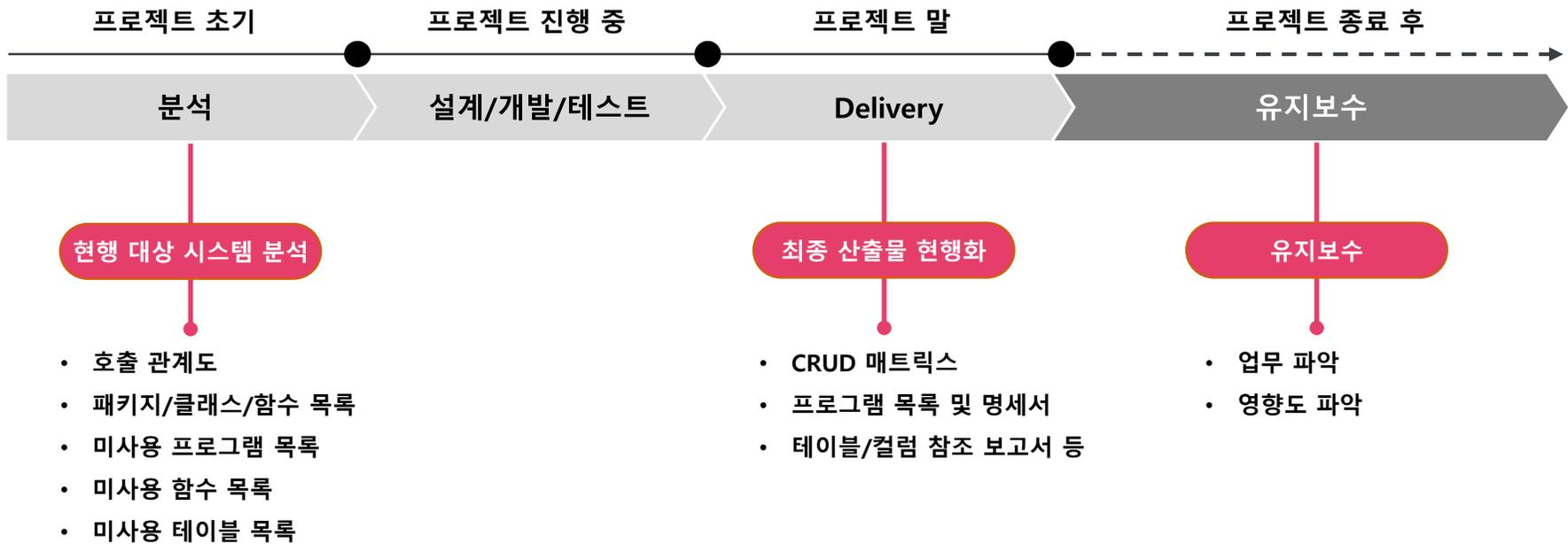


The screenshot displays a hierarchical navigation tree on the left with the following items:

- TERM (거리주소)
- MODEL (LOCATIONS : [차세대 > 공통 > HR])
- CATALOG (LOCATIONS)
- COLUMN (ORDDEPTCD)
- XML (OiRaeSuNapKioskMap.xml)
- XML (OiRaeSuNapMap.xml)
- Java (PAMCommonbizDAO.java)
- Java (PAMCommonBIZ.java)

Three main analysis windows are highlighted:

- 모델 정보 확인** (Model Information Confirmation): Shows a detailed ER diagram with tables like EMPLOYEES, DEPARTMENTS, EMP_LOG, and LOCATIONS.
- DB 구조 및 객체 확인** (DB Structure and Object Confirmation): Shows a table definition for LOCATIONS with columns: CITY, STATE_PROVINCE, COUNTRY_ID.
- 어플리케이션 흐름 확인** (Application Flow Confirmation): Shows a flow diagram of application components.
- 어플리케이션 소스 확인** (Application Source Confirmation): Shows the source code for PAMCommonbizDAO.java.



타사 제품 간 비교표

자사는 풍부한 구축경험과 높은 수준의 제품 완성도를 갖추고 있으며, 제품 개발에 직접 참여한 전문개발인력을 통해 **탁월한 문제해결능력과 유연한 커스터마이징 서비스**를 제공함

항목		DATAWARE™ AP#	A 제품	B 제품
시장점유율		▪ 중간	▪ 높음	▪ 낮음
타사 제품 대비 장점		<ul style="list-style-type: none"> ▪ 분석엔진 성능 및 안정성 ▪ 확장 및 연계 용이성 ▪ 사용편의성 	▪ 미흡	▪ 미흡
성능	분석 소요시간 (10만본 소스기준)	▪ 1시간	▪ 7시간	▪ 2시간
	분석결과 조회 소요시간	▪ 평균 2~3초 내외	▪ 평균 10초 내외	▪ 평균 10초 내외
안정성 (자동 스케줄 및 장애 시 복구)		▪ 우수	▪ 보통	▪ 보통
클라이언트 환경		▪ 국제 웹 표준 지원	▪ C/S 환경으로만 제약	▪ 사용성 복잡
기술지원		▪ 우수 (최적화 및 유지보수)	▪ 보통	▪ 미흡 (최적화 X)
유연성 (시스템 연계)		▪ 우수 (개발환경(IDE) 및 다양한 형상관리, 표준, 모델, 품질시스템 연계 경험)	▪ 보통(자사제품 연계 : 올인원 솔루션)	▪ 미흡 (외산)
라이선스 정책		▪ 서버 1식 + 분석대상 언어	▪ 상황 별 정책 변동	▪ 분석 추출 object 수

04

구축사례

구축사례

미사용 자원(테이블/프로그램) 추출 사례

[C 물류사]

- 사업명 택배 차세대 프로젝트
- 사업범위 ASIS 시스템에서 사용되지 않는 테이블/프로그램 목록을 미리 추출하여 차세대 프로젝트 분석/설계 단계에서 활용
- 적용 방안
 - ✓ 미사용 테이블/프로그램 추출
 - 어플리케이션에서 사용된 SQL을 분석하여 테이블 목록을 추출하고, 실제 DB에 존재하는 테이블과 비교하여 사용되지 않은 테이블 목록 도출
 - 참조관계(호출관계)가 없는 어플리케이션을 추출하여 미사용 프로그램 후보로 제공

미사용 테이블 보고서

스키마	테이블	애플리케이션	스키마 선택	보고서 출력
(SERVER) ORCLUTF8, (INSTANCE) ORCLUTF8, (SCHEMA) PAMDBA	AAA	KH APPS, KH sample		
	APPENDETEST	KH APPS, KH sample		
	AUTOCONV	KH APPS, KH sample		
	AUTOMAP	KH APPS, KH sample		
	A_INTEGRITY_TEMP	KH APPS, KH sample		
	A_SAVEUPDATES	KH APPS, KH sample		
	A_SEL100RAND	KH APPS, KH sample		
	A_SEL100SEQ	KH APPS, KH sample		
	A_TENPCT	KH APPS, KH sample		
	A_TINY	KH APPS, KH sample		
	A_UNIQUES	KH APPS, KH sample		
	A_UPDATES	KH APPS, KH sample		
	BESUGA_TAMP	KH APPS, KH sample		
	CARD_TEMP	KH APPS, KH sample		

선택한 스키마에서 미사용 테이블 목록 조회

미사용 프로그램/함수 보고서

응용분류	파일명	프로그램명	함수	보고서 출력
DATAWARE APPS	change_manage.c			
DATAWARE jobmgr	db_util.c	/home/impact/src/Impact/Manager/JobMgr		
DATAWARE jobbapp	deletefileinfo.c	/home/impact/src/Impact/Manager/JobMgr		
DATAWARE APPS	deletejobinfo.c	/home/impact/src/Impact/Manager/JobMgr		

업무(수집, 분석) 대상 선택

선택한 대상에서 미사용 프로그램 목록 조회

프로그램경로	프로그램명	함수
/home/impact/sample/dataware/src	Ognl.java	substring
/home/impact/sample/dataware/src	Ognl.java	length
/home/impact/sample/dataware/src	er.java	DuplicationCallChe...
/home/impact/sample/dataware/src	er.java	has
/home/impact/sample/dataware/src/encore/meta/a...	NodeExpander.java	nextLevel
/home/impact/sample/dataware/src/encore/meta/a...	NodeExpander.java	print
/home/impact/sample/dataware/src/encore/meta/a...	CommonAspect.java	pushAfterRequest
/home/impact/sample/dataware/src/encore/meta/a...	MailAspect.java	sendMailForRequest
/home/impact/sample/dataware/src/encore/meta/c...	DatabaseMessage...	get18NSqlQuery

선택한 대상에서 미사용 함수 목록 조회

구축사례

소스 복잡도 활용 사례

[H 제조사]

- **사업명** MDM 고도화 구축 프로젝트
- **사업범위** 프로젝트 수행 시 영향받는 시스템의 어플리케이션 복잡도를 도출해 정량적인 프로젝트 공수 산정
- **적용 방안**
 - ✓ **어플리케이션 복잡도 수치를 M/M 산정의 참고자료로 활용**
 - 소스 파일 내 분기/반복문의 개수에 따라 복잡도를 계산하여 파일 및 함수 복잡도를 도출
 - 코드 크기(LOC), 복잡도, 로직 Depth 등을 활용하여 개발 M/M 산정의 정량적 근거 제시

파일별 복잡도

파일경로	파일명	파일용량(...)	총라인	소스라인	주석라인	호출번호	함수개수	최대복잡도	평균복잡도	최대 Depth	평균 Depth
13 /case3/action	ByeongSikJeongBoJoh...	6	146	107	39	73	2	1	1	0	0
14 /case3/action	ByeongWeoniJeongG...	7	176	119	57	55	4	1	1	0	0
15 /case3/action	DamDangJaSuNapJoh...	10	234	171	63	109	5	1	1	0	0
16 /case3/action	ERHwanJaGwanRiAct...	9	214	75	139	43	1	1	1	0	0
17 /case3/action	ERPcrlmAction.java	4	107	75	32	30	2	1	1	0	0
18 /case3/action	EuiSaBoChungJinRyol...	3	78	48	30	17	1	1	1	0	0
19 /case3/action	EuiSaHolMasterAction...	8	185	137	48	84	5	1	1	0	0
20 /case3/action	EuiSaJinRyolJeongGw...	14	322	230	92	162	8	1	1	0	0
21 /case3/action	GalToWeonGwanRiAct...	8	218	157	61	64	5	1	1	0	0
22 /case3/action	GaeInGyeYakMisuSuN...	53	1141	905	236	660	20	5	2.8	2	0.81
23 /case3/action	GiTaSuNapAction.java	21	497	309	188	165	10	5	3	1	0.9
24 /case3/action	GongTongJoHoiAction...	28	580	351	229	257	7	9	3.43	2	0.88
25 /case3/action	HabChoiBunChoiAction...	16	366	287	79	233	7	1	1	0	0
26 /case3/action	HallnCodeGwanRiActo...	8	212	153	59	92	4	6	3.75	2	1.05
27 /case3/action	HallnCodeSuigCodeG...	7	178	126	50	50	3	1	1	0	0
28 /case3/action	HimslSampleAction.java	11	240	211	29	198	2	19	19	1	1
29 /case3/action	HomePageGwanRiAct...	20	439	396	43	255	10	1	1	0	0
30 /case3/action	HttpCorHelper.java	2	84	82	2	6	2	1	1	0	0
31 /case3/action	HwanJaGOnJeongBo...	62	1458	1106	352	826	33	1	1	0	0
32 /case3/action	HwanJaGChoJeongBo...	5	143	103	40	46	2	6	6	1	0
33 /case3/action	HwanJaSangSeAction...	11	290	206	84	82	6	5	3.33	1	0
34 /case3/action	IfsKnmcmgSalbackH...	3	83	44	39	0	7	1	1	1	0
35 /case3/action	IfsKnmcmgStub.java	357	7255	6215	1040	883	21	1	1	1	0
36 /case3/action	IpWeonDeungRokActo...	59	1251	910	341	732	26	1	1	0	0

파일별 복잡도

복잡도 정보
엑셀 다운로드

함수별 복잡도

파일경로	파일명	함수명	복잡도
101 /sample/case1	OPI_CAutoDiagGwanRi.xfm	delSpcdMainData	4
102 /sample/case1	OPI_CAutoDiagGwanRi.xfm	deptChange	0
103 /sample/case1	OPI_CAutoDiagGwanRi.xfm	diagcdJoHoi	0
104 /sample/case1	OPI_CAutoDiagGwanRi.xfm	doctChange	0
105 /sample/case1	OPI_CAutoDiagGwanRi.xfm	existsDestData	3
106 /sample/case1	OPI_CAutoDiagGwanRi.xfm	fClickAddRowBtn	3
107 /sample/case1	OPI_CAutoDiagGwanRi.xfm	fClickDeleteRowBtn	8
108 /sample/case1	OPI_CAutoDiagGwanRi.xfm	getMainData	1
109 /sample/case1	OPI_CAutoDiagGwanRi.xfm	getMainJoHoi	10
110 /sample/case1	OPI_CAutoDiagGwanRi.xfm	getOPICATSPMList	5
111 /sample/case1	OPI_CAutoDiagGwanRi.xfm	getSpcdHistoryData	1
112 /sample/case1	OPI_CAutoDiagGwanRi.xfm	getSubData	3
113 /sample/case1	OPI_CAutoDiagGwanRi.xfm	getSumCalcscorcdList	3
114 /sample/case1	OPI_CAutoDiagGwanRi.xfm	getSumDiagcdList	3
115 /sample/case1	OPI_CAutoDiagGwanRi.xfm	getSumOrddptList	2
116 /sample/case1	OPI_CAutoDiagGwanRi.xfm	saerogochim	1
117 /sample/case1	OPI_CAutoDiagGwanRi.xfm	searchDeptChange	8
118 /sample/case1	OPI_CAutoDiagGwanRi.xfm	setDataForUpdate	1
119 /sample/case1	OPI_CAutoDiagGwanRi.xfm	setFlagNewData	6
120 /sample/case1	OPI_CAutoDiagGwanRi.xfm	setGenrMemo	2
121 /sample/case1	OPI_CAutoDiagGwanRi.xfm	setHelpCaption	2
122 /sample/case1	OPI_CAutoDiagGwanRi.xfm	setMainData	20
123 /sample/case1	OPI_CAutoDiagGwanRi.xfm	setNewDataForUpdate	1
124 /sample/case1	OPI_CAutoDiagGwanRi.xfm	setOPICATSPMFForUpdate	1

함수별 복잡도

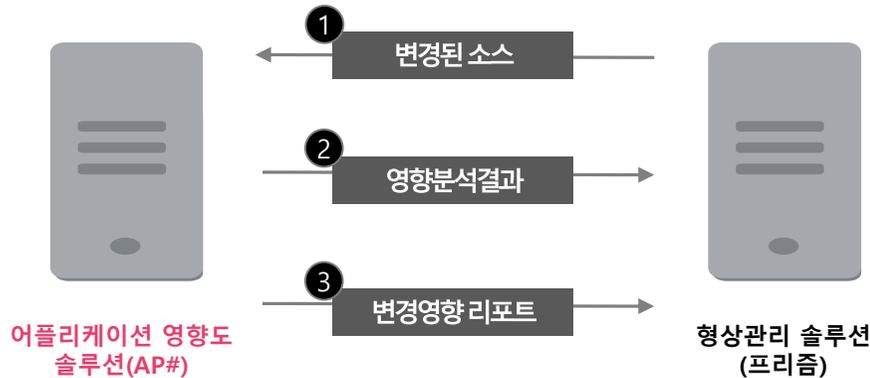
복잡도 정보
엑셀 다운로드

구축사례

빌드&배포 전 영향분석 사례

[K 은행]

- **사업명** 데이터품질관리체계 구축 프로젝트
- **사업범위** 형상관리 솔루션(프리즘)과 연계해 변경대상 소스코드 및 배포대상 프로그램에 대한 영향분석을 수행
- **적용 방안**
 - ✓ **변경대상/배포대상 파일에 대한 영향 분석 수행**
 - 소스코드를 수정하기 전에 영향 분석을 미리 수행하여, 함께 수정되어야 할 소스가 누락되지 않았는지 확인
 - 소스 수정이 완료되고 변경된 프로그램이 배포되기 전에 영향 분석을 수행하여, 함께 배포되어야 할 프로그램이 누락되지 않았는지 확인



- ① 프리즘에서 변경된 파일 목록과 파일 저장 위치를 제공한다.
AP#에서는 파일 목록을 확인하여 AP# 분석 서버로 복사한다.
- ② 프리즘에서 소스가 변경되기 전(계획단계)과 배포하기 전(적용의뢰단계)에 영향 분석을 수행한다.
AP#은 변경대상/배포대상 파일에 대한 영향분석 결과를 제공한다.
- ③ 영향분석 결과는 AP# 분석서버에 저장하며, 프리즘에서 영향분석 결과 리포트를 다운로드할 수 있도록 제공한다.



데이터 통합 관리 솔루션
DATAWARE™

엔코아는 데이터 관리 체계 전반을 지원하는 통합 관리 솔루션 DATAWARE™를 제공합니다.
DATAWARE™는 단일 플랫폼에서 데이터 품질을 극대화하고 관리 효율성을 높이는 솔루션입니다.

✓ 기업 전반의 데이터 거버넌스 체계 지원

- 통합 Repository 기반의 End-to-End 솔루션 제공
- 분석·설계·운용·관리에 이르는 데이터 라이프 사이클 지원
- On-Premise 및 Cloud 환경에 최적화된 서비스 모듈 구성

✓ 운영에 최적화된 웹 기반 통합 관리 포털 제공

- OS 및 브라우저 종속 없는 서비스 제공
- 각 솔루션은 플러그인 형태로 별도 장착 가능
- 국제 웹 표준 준수 및 전자정부 프레임워크 호환

DATAWARE™

DA
dataware
data
architecture

META
dataware
metadata
management

DQ
dataware
data
quality

DF
dataware
data flow
management

AP
dataware
impact
analysis

ETT
dataware
data
integration

SQL
dataware
data
operation

DV
dataware
data
virtualization

DP
dataware
data
portal

데이터
모델

데이터
표준

데이터
품질

데이터
흐름

어플리케이션
영향도

데이터
이행

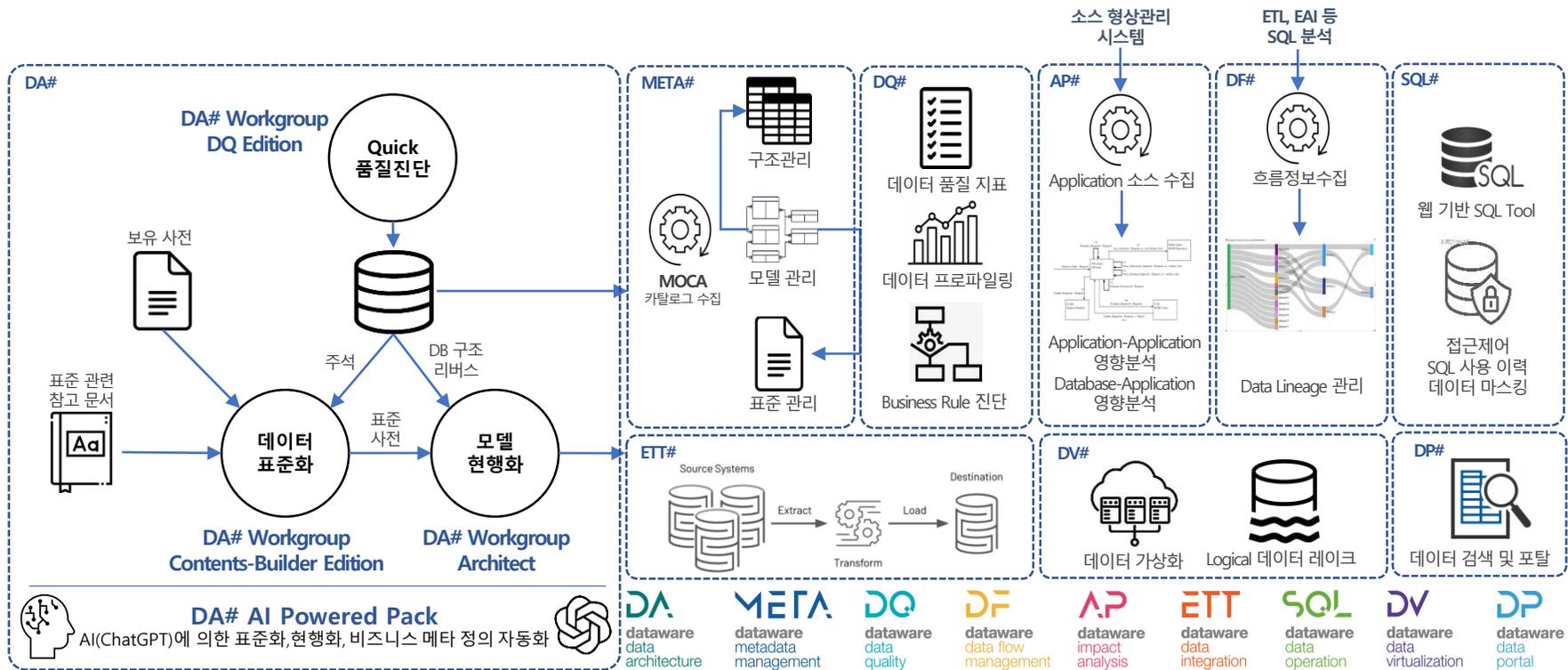
데이터
보안

분산
이기종 데이터
실시간 결합

데이터
포털

데이터 거버넌스 기반 시스템 – Full Stack 솔루션 : DATAWARE

DATAWARE에 속한 모든 제품은 데이터 관리 및 활용성의 극대화를 위해 제품간 정보교환 및 상호 연계기반을 통해 압도적인 효율성을 제공합니다



감사합니다.
Thank You!

EN-CORE Inc. is "Total Data Service Provider." We, as one of the leading data scientists groups in Korea, are offering a wide-range of services such as consulting, education, solutions, and technology information based on data ranging from planning the corporation IT systems to analyzing, designing, developing, transitioning, managing, performing governance, and data servicing.

The logo for AP dataware impact analysis, featuring the letters "AP" in a large, white, stylized font, followed by "dataware" in a smaller, white, lowercase, sans-serif font, and "impact analysis" in a smaller, white, lowercase, sans-serif font below it.