

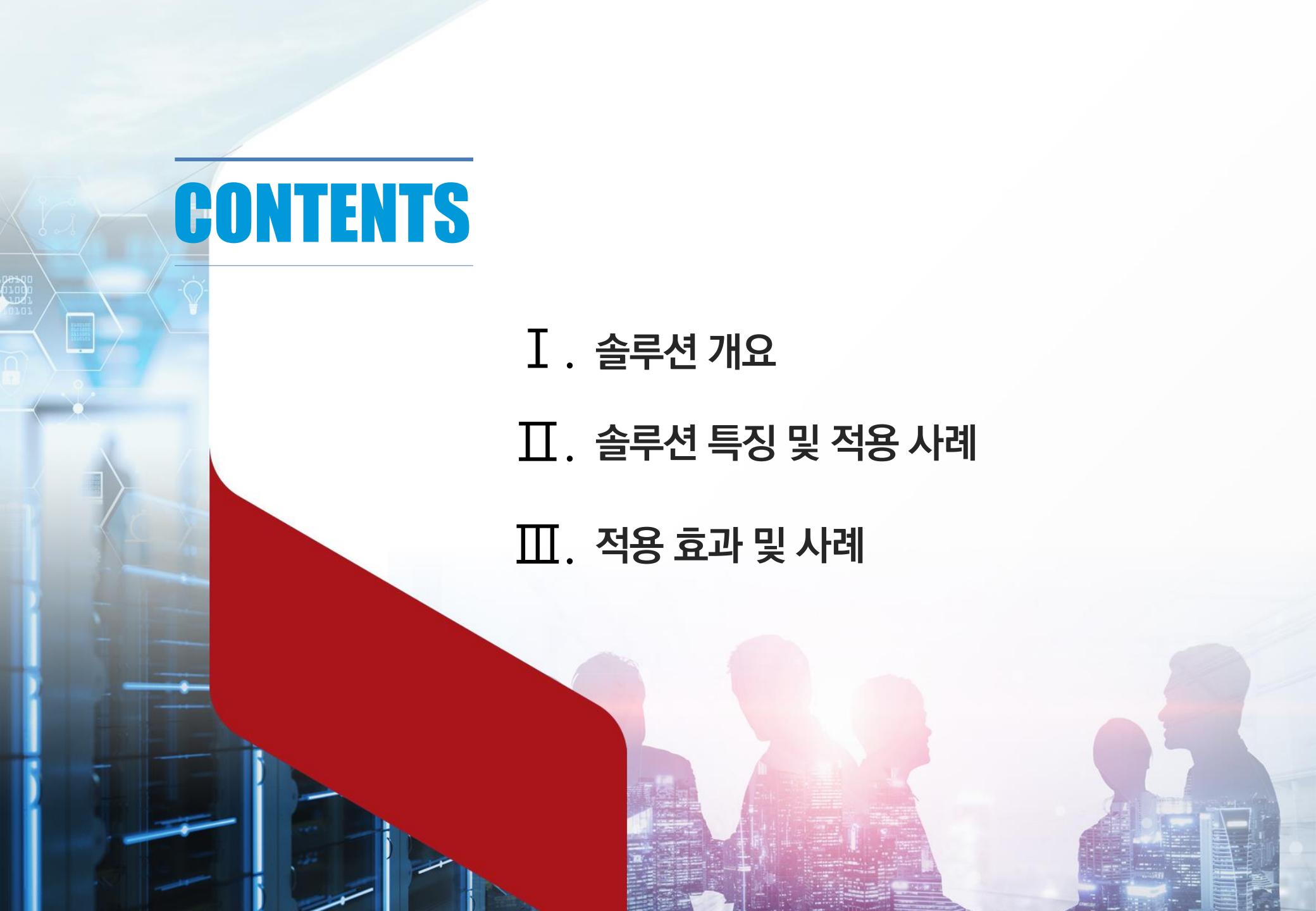
Smart Conversion Suite

# 스마트 마이그레이션 솔루션 소개서

뱅가드랩

# CONTENTS

---

- 
- I . 솔루션 개요
  - II . 솔루션 특징 및 적용 사례
  - III. 적용 효과 및 사례

# Chapter I . 솔루션 개요

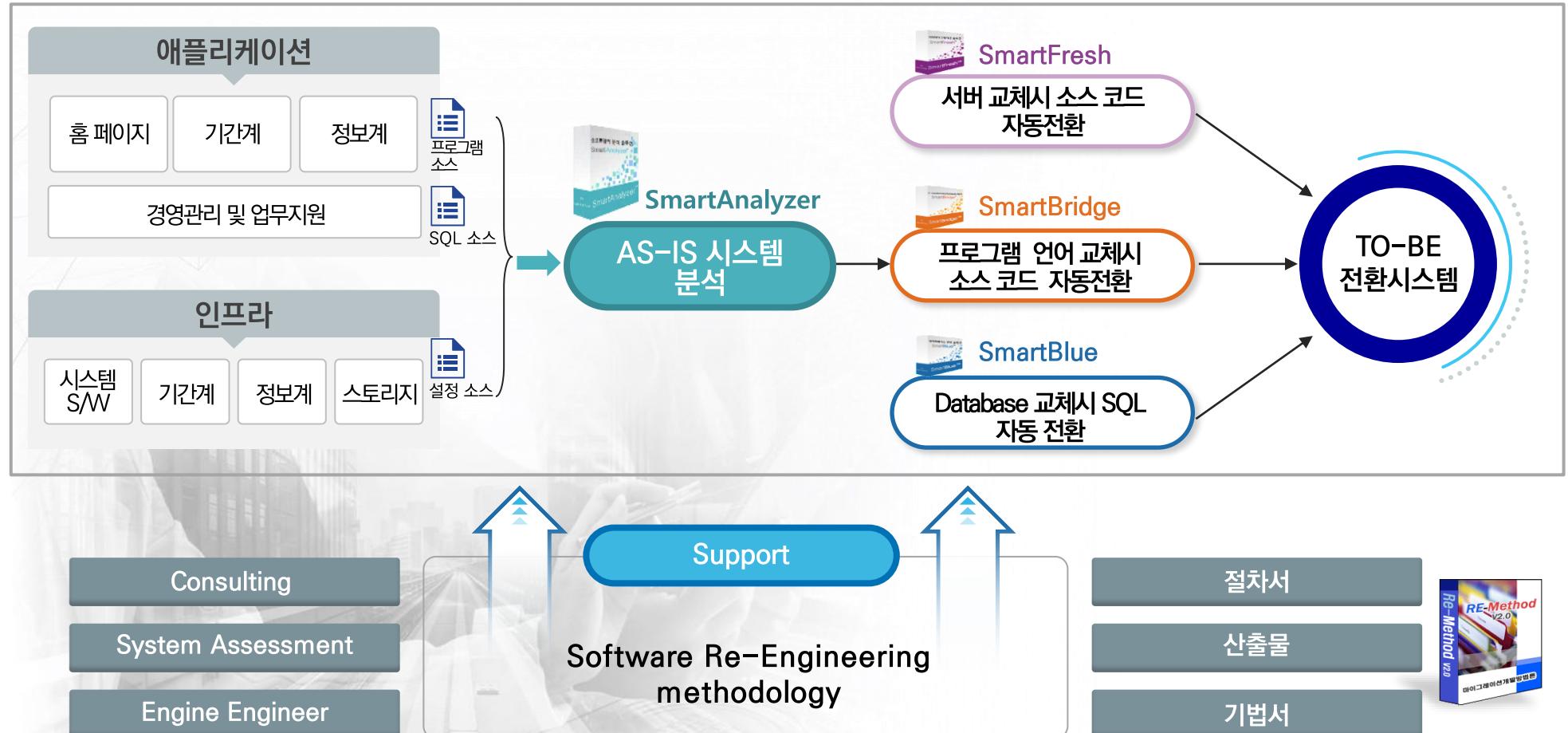
1. Smart Conversion Suite 개요
2. SmartAnalyzer – 분석 Report 생성 솔루션
3. SmartFresh – 서버 컨버전 솔루션
4. SmartBridge – 애플리케이션 컨버전 솔루션
5. SmartBlue – 데이터베이스 컨버전 솔루션



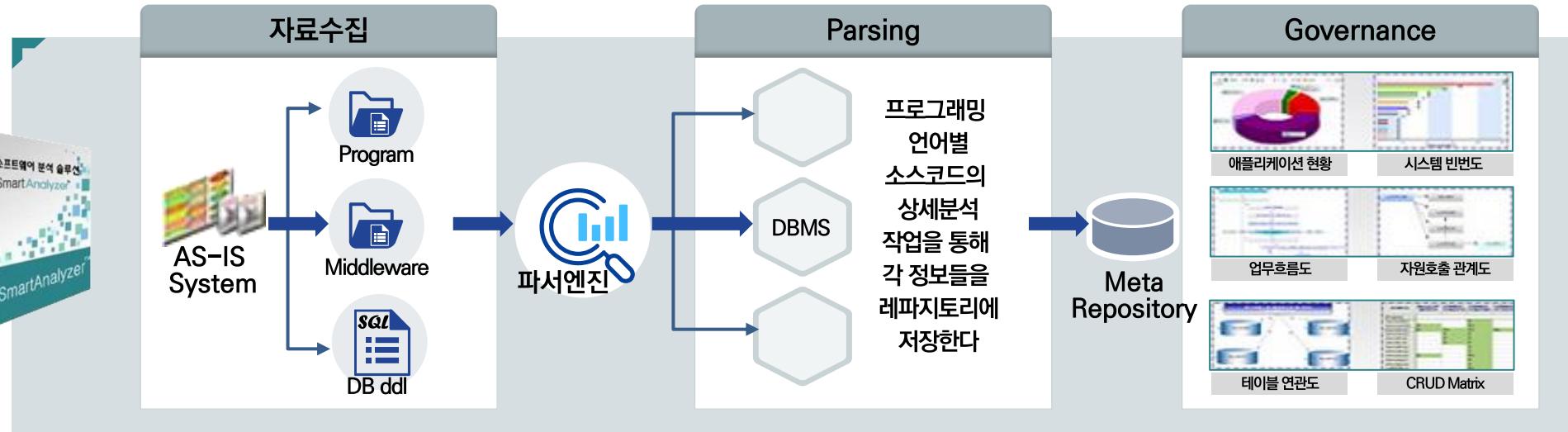
## 1

# SmartConversion Suite 개요

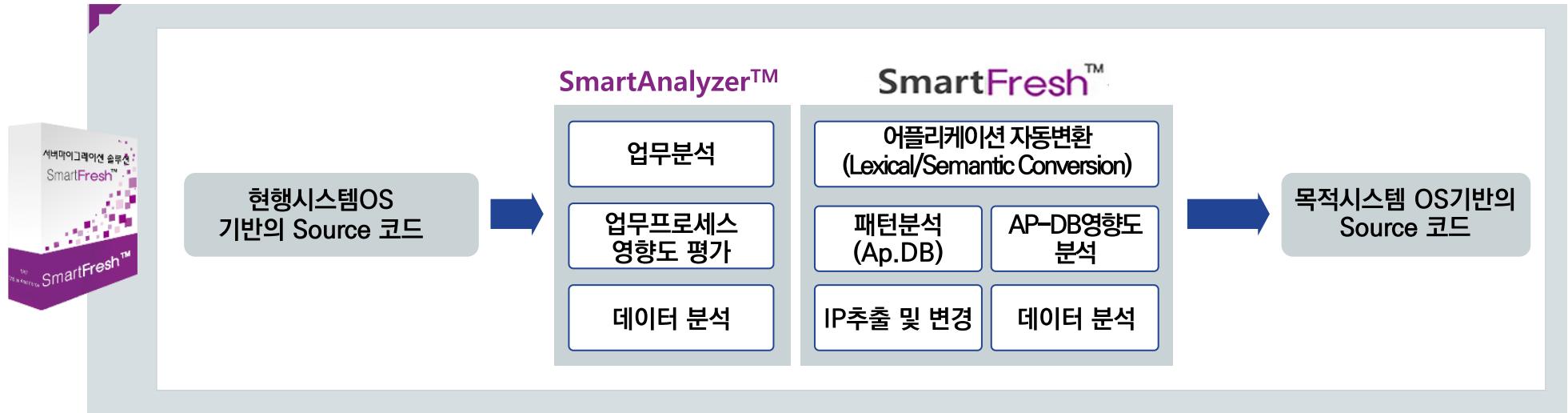
As-Is 애플리케이션 시스템 분석을 위한 **분석 Report 생성 기능과 프로그램 소스 및 SQL 코드를 자동 변환**하는 제품군으로 구성되어 있습니다.



현행 시스템 현황을 보여주는 분석 Report를 자동 생성하여 **현행 시스템 애플리케이션 자원을 상세하게 분석**할 수 있도록 지원합니다.



자동화 도구를 이용하여 서버 교체 혹은 업그레이드 시 기존 운영체제에서 실행되는 애플리케이션을 새로운 운영체제에서 실행될 수 있도록 **애플리케이션 소스를 자동으로 변환**합니다.



#### 현행시스템 분석 도구(SmartAnalyzer)

- マイグレーション을 위한 현행시스템 분석 도구
- OS 변경에 따른 마이그레이션을 위한 소스 분석을 신속하고 안정적으로 지원
- 애플리케이션 소스 코드를 스캔하여 수정해야 할 부분과 문제점, 예상기간을 리포팅

#### 자동변환 도구(SmartFresh)

- 분석 완료된 소스에 대한 자동 변환을 수행하는 도구
- SmartAnalyzer를 통해 분석한 가이드를 바탕으로 작성된 Rule을 적용하여 자동 변환

#### 자동화 도구 장점

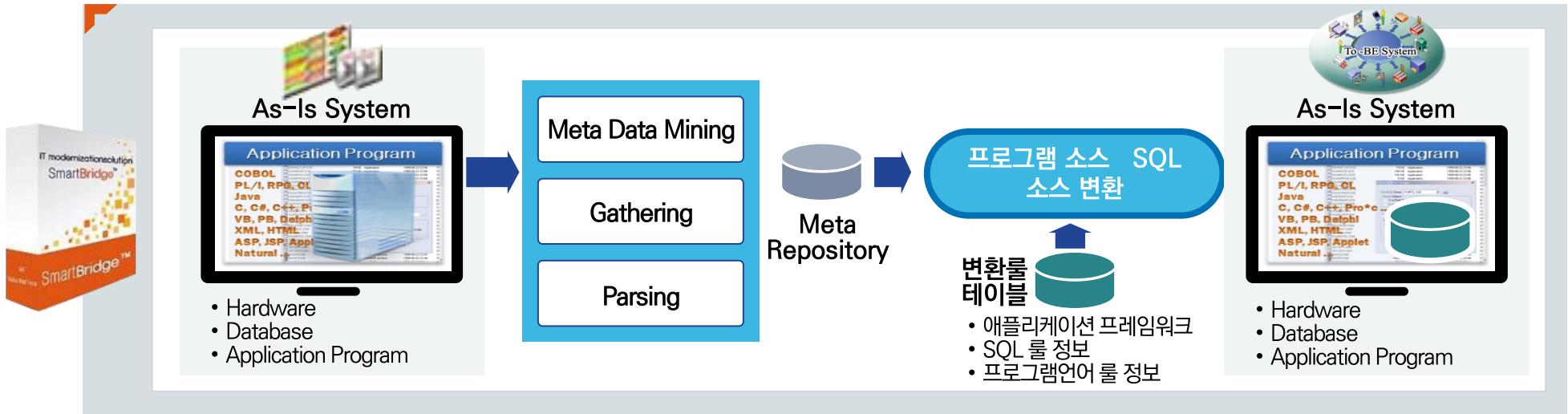
- 안정성/효율성**  
휴면 에러 감소, 소스의 통일성 및 직관성 증대로 유지관리 효율성 증대
- TCO 절감**  
マイ그레이션 시간 단축, 테스트 단계 단축으로 비용 절감
- Risk 관리**  
문제 사전 파악과 다각적 영향 분석을 통해 Risk 최소화 가능



## 4

## SmartBridge™ – 애플리케이션 프로그램 컨버전 솔루션

자동화 도구를 이용하여 **기존 프로그래밍 언어로 작성된 소스를 새로운 프로그래밍 언어로 변환**할 수 있으며, 다양한 애플리케이션 프레임워크 환경에 맞게 변환합니다.



### 지원 환경

- 지원 프로그램 언어
  - C, C++, C#, Java, Shell, COBOL, PL/I, Assembler, VisualGen, Power Builder 등
  - Stored Procedure, Oracle Forms, Trigger 등
- 지원 애플리케이션 프레임워크
  - Nexcore, DevOn, AnyFrame, BANCS, Proframe, ADF, Systemier

### 활용 분야

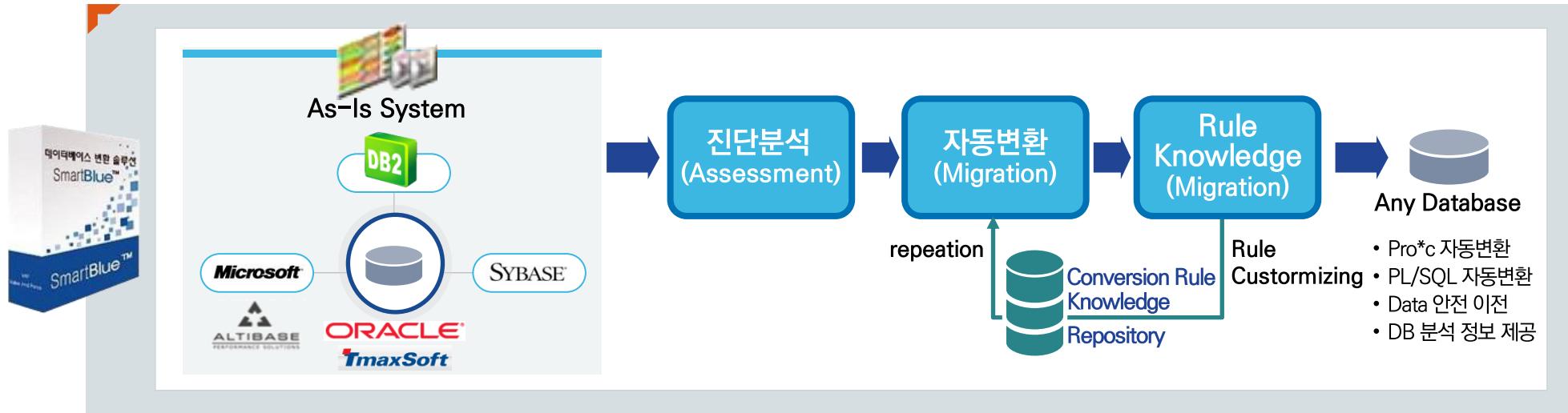
- 차세대 시스템 개발 프로젝트
- 시스템 업그레이드 프로젝트
- 비즈니스 룰 재사용 및 개발 프로젝트
- 시스템 아키텍처 및 프레임워크 관리 업무

### 활용 분야

- 정확한 현행시스템 자원 분석
- 자연언어지원, 이해도 제고
- 비즈니스 룰 계승
- 애플리케이션 변경 관리
- 신규 시스템 병행관리
- 개발 리스크 감소



데이터베이스 시스템 변경 시 **새로운 데이터베이스에 맞게 SQL 소스코드를 자동으로 변환** 합니다.



### 제품 특징

- 강력한 변환 룰 엔진
- 다양한 DBMS 지원
- DBMS 연관 영역  
(프로그램언어, WAS,F/W) 지원
- 데이터베이스 분석 정보 제공
- SQL 관리 기능 제공

### 활용 분야

- 각 기업 데이터베이스 관리자 및 개발자
- 프로젝트 개발사 및 매니저
- SI 업체 및 SW 개발업체
- 데이터베이스 공급자

### 도입 기대 효과

- 데이터베이스 변환 시 비용 및 기간 단축(40~50%)
- 데이터베이스 관리 정보를 메타화하여 유지보수에 활용
- 변환 전후 데이터베이스 및 프로그램 정보 확인

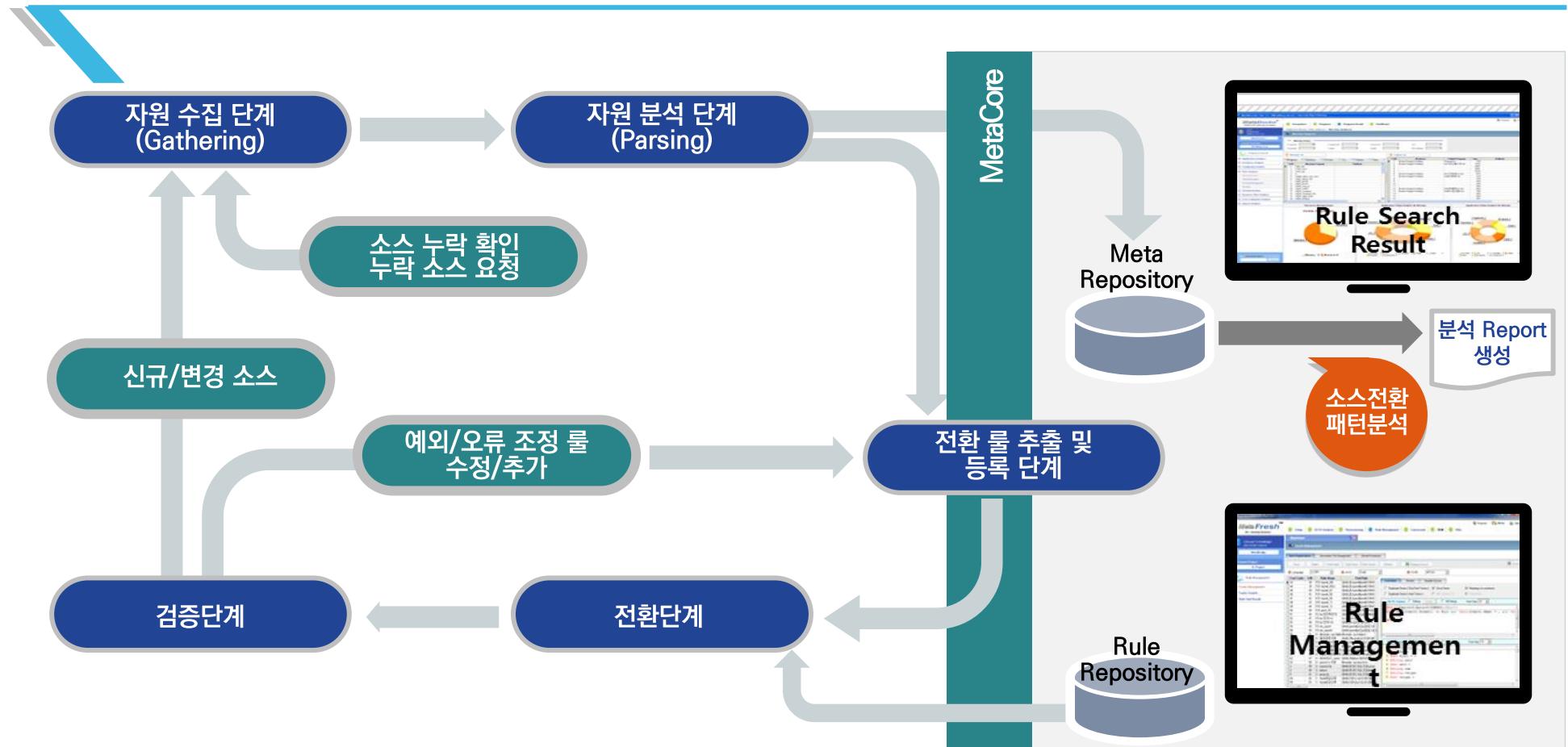


## Chapter II. 자동화 도구 특징 및 기능

1. 자동화 도구 적용 프로세스
2. 핵심 기술 – 소스코드 분해 기술 Parser
3. 핵심 기술 – 자동화 룰 엔진
4. 핵심 기술 – ABL Script 언어
5. 핵심 기능 – 분석 Report 생성
6. 핵심 기능 – 프로그램/SQL 소스 자동 컨버전



자동화 도구 활용 과정은 자원수집, 자원분석, 전환 룰 추출 및 등록, 전환 검증단계로 진행됩니다.

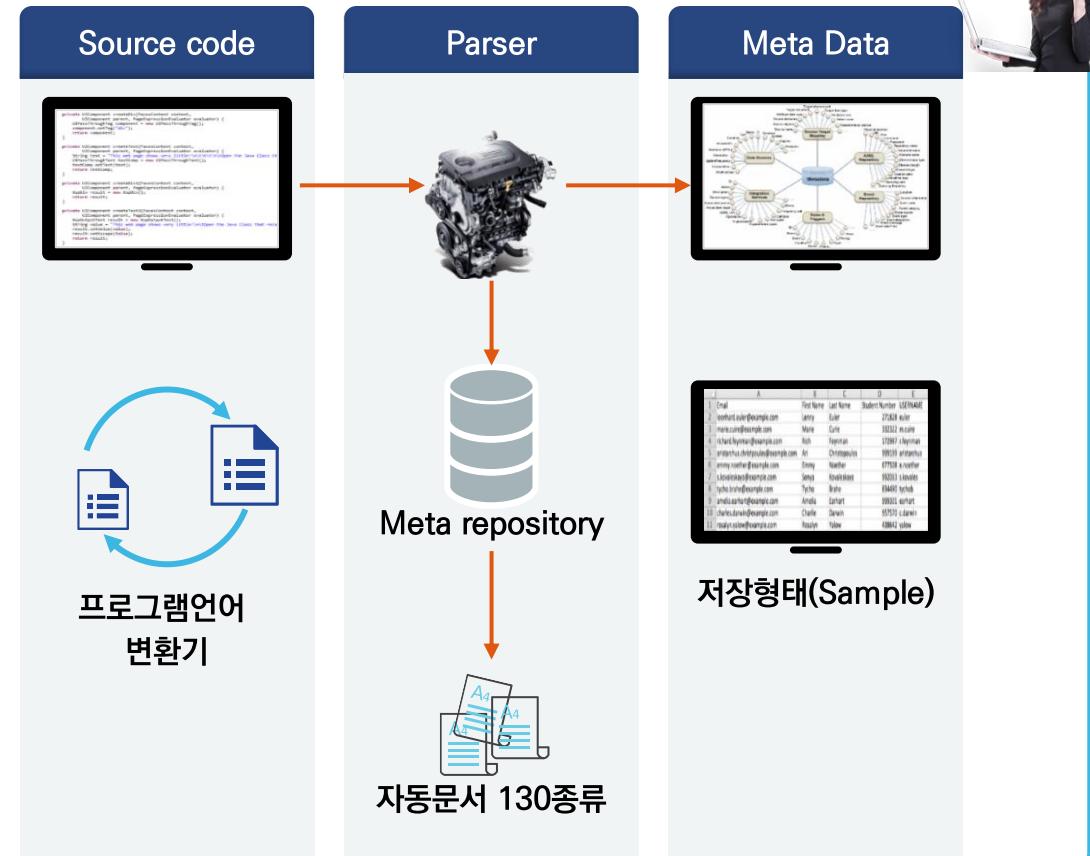


SW 재공학의 SW재사용을 위한 핵심기술인 “파서(Parser)”를 기초 기술기반으로 자체 개발한 핵심 엔진입니다.



### 소스코드 분해 엔진(Parser)

- 소스코드를 수집, 분석, 분해하는 Parser
- 특허를 보유한 자체 개발 Parser
- 45종의 프로그램 언어 지원
- Lexical(문법적)과 Semantic(의미론적) 기능 두 단계로 진행
- 분해된 정보(Meta Data)는 메타데이터 레파지토리에 메타형태로 저장
- 메타데이터는 자동문서와 언어변환의 기초데이터로 재사용



## 3

## 핵심기술-자동화 룰 엔진

기존 소스에서 파싱된 정보를 기초로 하여 새로운 프로그래밍 언어에 대한 매핑 룰을 등록하고 **프로그램 소스를 자동 변환하는 기술**이 적용된 엔진입니다.



The screenshot displays the 'MetaRule Set' configuration window for the 'Logistics - Oracle to DB2' project. Key sections include:

- Project Information:** Shows the selected Work Group as 'Oracle to DB2\Logistics\Port' and Rule Set as 'Logistics'.
- Platform Information:** Lists AS-IS Language (None), AS-IS OS (AIK), AS-IS Framework (None), AS-IS DB (Oracle), TO-BE Language (None), TO-BE OS (AIK), TO-BE Framework (None), and TO-BE DB (DB2).
- MetaRule Information:** Shows Language set to [ALL] and MetaRule Count at 27.
- MetaRule Registration:** Buttons for New, Save, Delete, Refresh, and Full Screen.
- Registration Mapping Rule:** A table showing registered rules (e.g., A001 - CREATE...), their API Name, Rule Name, Sequence, and support options (Partial Support, Not Support).
- Description:** A text area containing the rule logic: "CREATE OR REPLACE statement is deleted, the function name in the '\*\*\*' is deleted."
- AS-IS MetaRule:** A code editor showing the transformation logic for AS-IS statements, such as `@ABL(''[UCREATE] [UOR] [UREPLACE] [UFUNCTION|PROCEDURE] [a\_FuncName] [\*]'')`.
- TO-BE MetaRule:** A code editor showing the transformation logic for TO-BE statements, such as `@Data()`, `@SetInfodelete(NUMBER)`, etc.

ABL은 **룰 등록을 하기 위해 만든 Script 언어**로 다양한 프로그램언어, SQL 문장들을 스크립트로 작성한 후 통일된 메타데이터와 결합하여 프로그램 언어, SQL을 처리할 수 있는 기술을 제공합니다.

## “ABL” Manual



변경할 프로그램 혹은  
소스코드

**As Is Source Code []**

Program Name: FBB01511S\_조회.pc

Original Embedded

```

121  /* sql error마다 ChecksSql() 함수 실행 tpreturn */
122  EXEC SQL whenever sqlerror DO CheckSql();
123
124  /*EXEC SQL set isolation to dirty read;*/
125
126  if (FBB01511S_ClientProcess() != SUCC)
127      tpreturn(TPFAIL,0,(char*)trans,0,0L);
128
129  /*2006.01.23 고태연: 회계단위에 따른 토직 분리*/
130
131  if(!strcmp(Cfinan_unit,"ALL")){
132      if (FBB01511S_DatabaseProcess_ALL(&tbfa021) != SUCC)
133          tpreturn(TPFAIL,0,(char*)trans,0,0L);
134  }
135  else {
136      if (FBB01511S_DatabaseProcess(&tbfa021) != SUCC)
137          tpreturn(TPFAIL,0,(char*)trans,0,0L);

```

**To Be Source Code [Java]**

Program Name: FBB01511S\_조회Biz.java

Conversion Date: 2011-01-02 20:04:43 Make User: ks\_kim

Version: J2008.03.0.0.2(최신버전)(마님)

Conversion Code: 71009

Conversion Error:  Compile Error...  Run Error...  Parse Error...

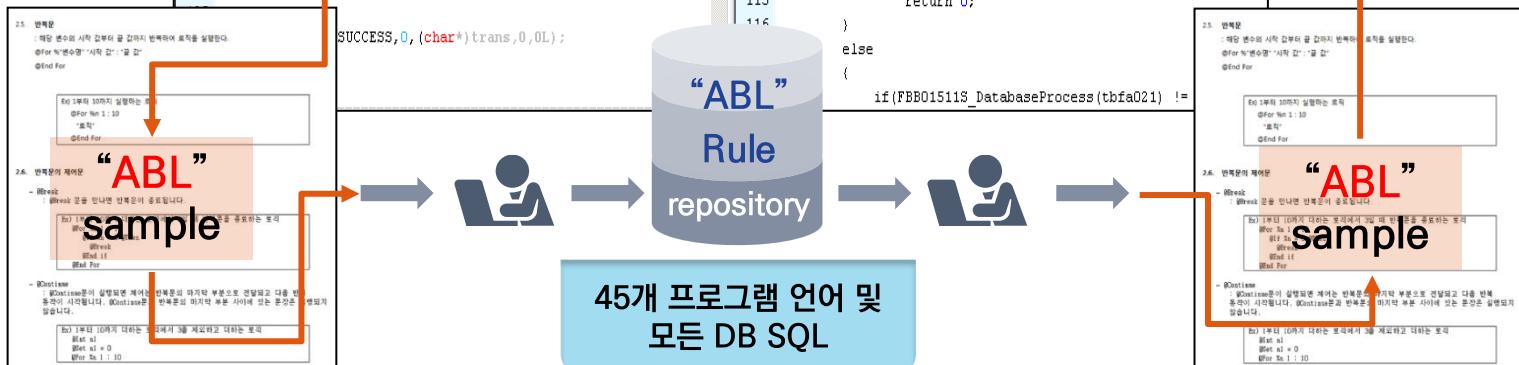
```

104      error.Procedure(TMAX, tpprno, "tprealloc error!");
105      return 0;
106  }
107  /* sql error마다 ChecksSql() 함수 실행 tpreturn */
108  /* EXEC SQL set isolation to dirty read; */
109  if(FBB01511S_ClientProcess() != SUCC)
110      return 0;
111  /* 2006.01.23 고태연: 회계단위에 따른 토직 분리 */
112  if(!getCfinan_unit().equals("ALL"))
113  {
114      if(FBB01511S_DatabaseProcess_ALL(&tbfa021) != SUCC)
115          return 0;
116  }
117  else
118  {

```



변경된 프로그램  
혹은  
소스코드

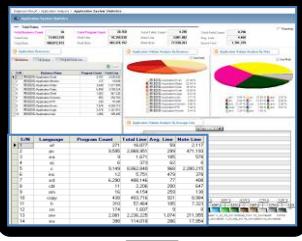


분석도구는 기존 시스템의 애플리케이션 현황을 **다양한 시각적 형태로 보여주어 분석내용을 쉽게 파악**할 수 있도록 지원합니다.

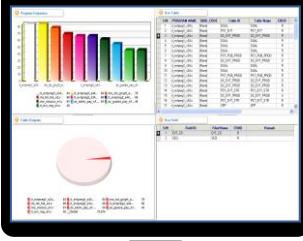
### 진단 분석 주요 산출물



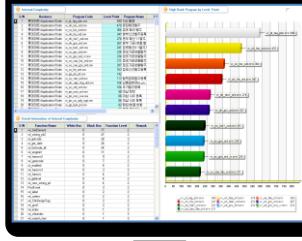
프로그램 현황



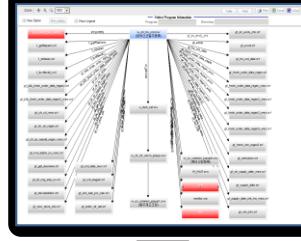
복잡도 분석



빈번도 분석

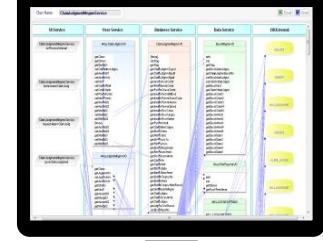


영향도 분석



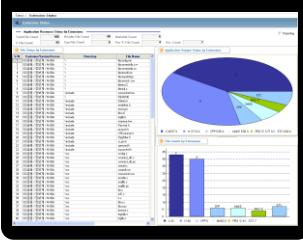
Pseudo Code

UML

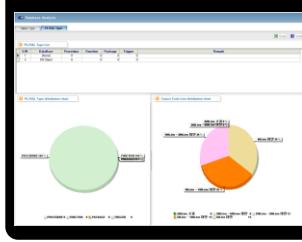


자원 Inventory

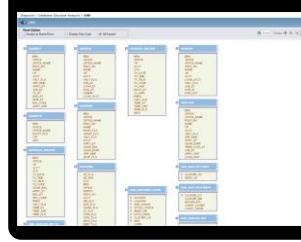
자원 형태별 현황



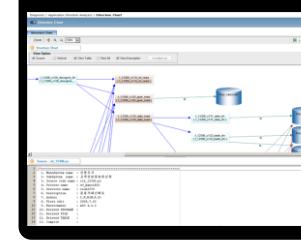
ERD



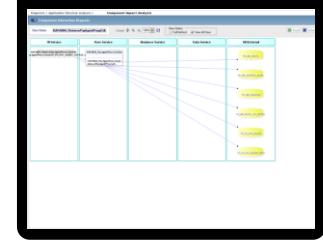
자원 속성별 현황



프로세스 연관도 분석



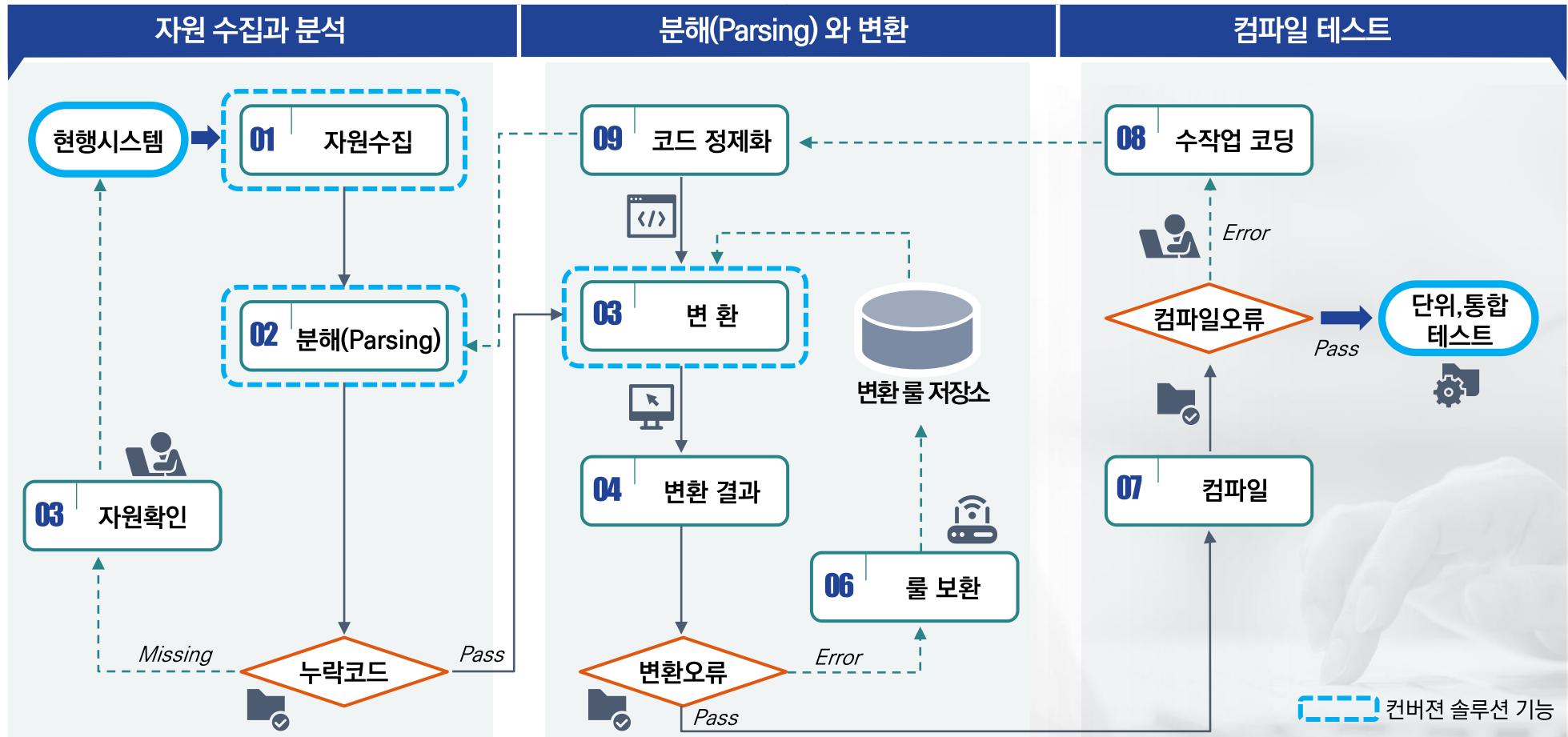
내부 프로세스 흐름 분석



## 6

## 핵심기능-프로그램 / SQL 소스 자동 컨버전

기존 프로그램 소스와 데이터베이스는 분석, 변환, 컴파일의 과정을 통해 다른 프로그램 언어와 데이터베이스로 자동적으로 전환됩니다.



## Chapter III. 적용 효과 및 사례

1. 프로젝트 단계별 활용 예시
2. 프로젝트 단계별 적용 효과
3. 프로그램 변환사례
4. 주요 고객 사례





### 수작업시

- 수작업에 의한 애플리케이션 소스 분석
- 문서화된 정보 부족으로 인한 생산성 저하
- 영향 분석 추적 및 시스템 연관 분석 불가능

- 부정확한 수작업 분석 자료 의존에 따른 품질 저하
- 영향분석 추적 불가능
- 시스템 연관 분석 불가능
- 자료관련도, 프로그램 연관도 분석에 어려움

- 문제·장애 발생시 소스 코드를 직접 분석
- 프로그램 로직 흐름을 수작업으로 파악
- 테스트 단계 파악된 대량의 변경 대상 소스를 각 개발자에 의해 수작업으로 변경

- 문제·장애 발생시 소스 코드를 직접 분석
- 프로그램 로직 흐름을 수작업으로 파악
- 이행 단계에 파악된 대량의 변경 대상 소스를 각 개발자에 의해 수작업으로 변경

### 자동화 S/W 도구 적용 시

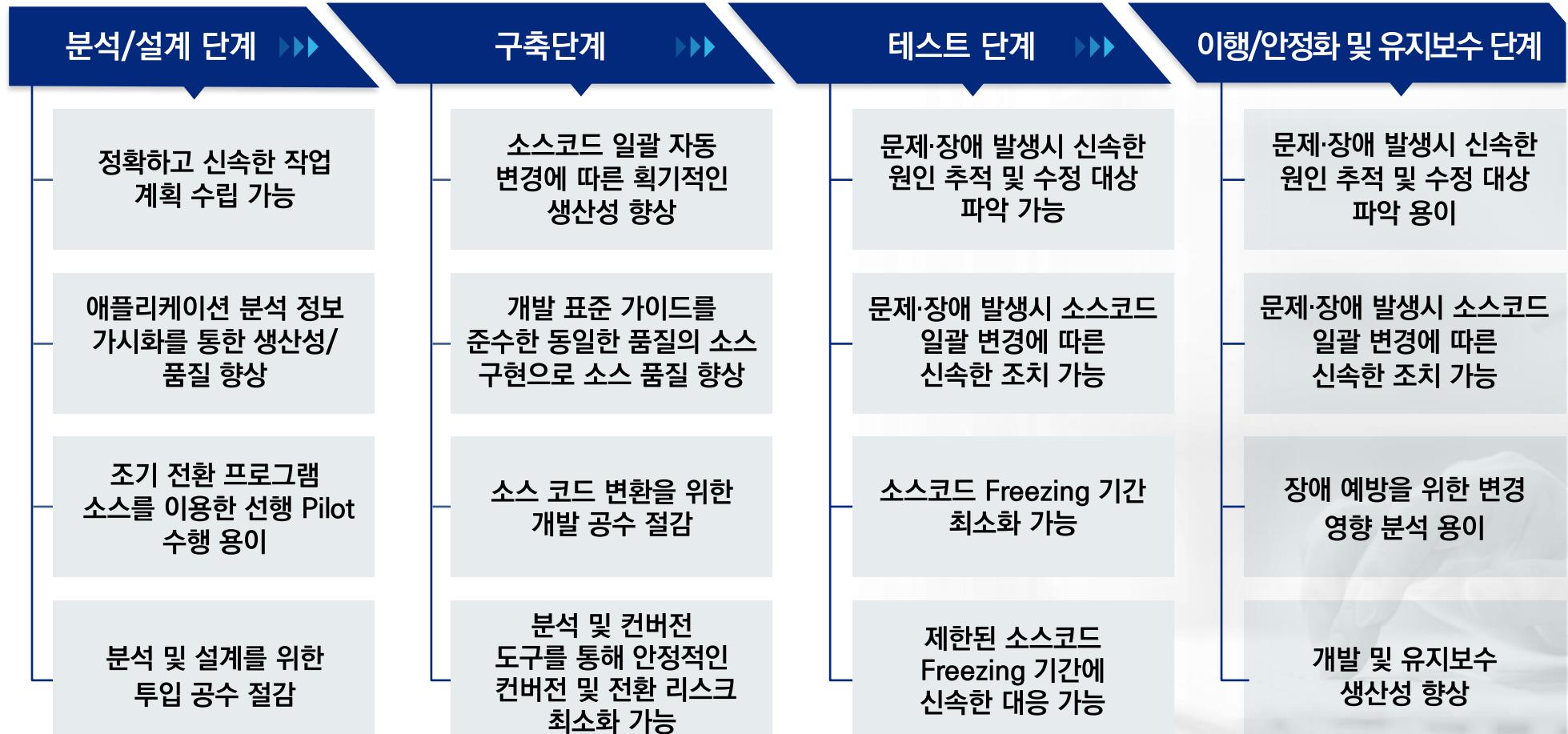
- 분석도구를 이용한 작업량 및 복잡도 분석 가능
- 프로그램 연관도, Flow Chart, 클래스 다이어그램, CRUD Matrix, ERD 등 150여종 보고서 자동 생성
- 미호출 및 존재하지 않는 프로그램/테이블 및 정보 파악 가능

- 등록된 룰 기반 프로그램 소스 자동 컨버전
- 데이터베이스 변경에 따른 SQL 소스 자동 컨버전
- 시스템 S/W 및 애플리케이션 프레임워크 변경에 따른 변경 대상 자동 전환
- OS 및 인프라 구성 변경에 따른 프로그램 소스 변경 대상 자동 전환

- 단위/통합 테스트 기간 문제·장애 발생시 각종 영향도 분석 산출물을 이용한 분석
- 문제·장애 발생시 변경 대상 소스 일괄 자동 컨버전 수행
- 성능 테스트시 성능 개선관련 변경 대상 소스 일괄 자동 컨버전 수행

- 인수 테스트, 이행 및 안정화 기간 문제·장애 발생시 각종 영향도 산출물을 이용한 분석
- 문제·장애 발생시 변경 대상 소스 일괄 자동 컨버전 수행
- 유지보수 요청(SR) 처리시 각종 영향도 분석 산출물을 이용한 분석

프로젝트 단계별 일정을 기준으로 자동 전환 도구 적용에 따른 효과는 다음과 같습니다.





- CICS/COBOL 10,000본
- Database : DB2

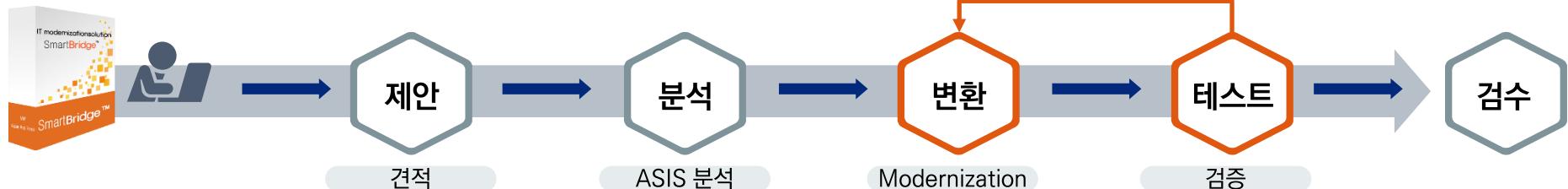


- C, JAVA
- Spring Framework
- Database : Oracle

### 삼성카드 사례

- 총괄 수주기업 : 삼성SDS
- マイグ레이션 : 벨류엔포스, 스마트브릿지, 인력3명
- 총 수주 금액 : 200억
- マイグ레이션 비용 : 20억
- 개발기간 : 7개월
- 벨류엔포스 참여단계 : 기획, 분석, 개발, 테스트단계 참여 ( 옆 그림 붉은 부분)
- 성공사례 : 삼성카드 라쿠톤 현대해상 흥국화재 등 50사이트

구분		계획 / 설계		사전 개발		개발 / 단위 테스트		통합 테스트 / 안정화	
PMO	프로젝트 관리	◆ 착수보고 ◆ 범위검증		◆ 중간보고 ◆ 어려움점 검증				◆ 이행결과보고 ◆ 개발 산출물 검증 ◆ 성능테스트 검증	◆ 테스트 평가
Operate	Application Migration	AS-IS 분석 Pilot 테스트 설계 대외 연계 현황	Pilot 테스트 설계 대외 인터페이스 설계	Application Migration 단위 테스트 DBMS Migration 및 정합성 검증 솔루션 Setup 및 연계 서비스 개발		Application Migration 단위 테스트 DBMS Migration 및 정합성 검증 솔루션 Setup 및 연계 서비스 개발		통합 테스트 데이터 이행 테스트 인터페이스 통합 테스트	인수 테스트 인정화 운영
SO	일반 업무	업무 분석 대외 인터페이스 설계	업무 모델링 및 설계 대외 인터페이스 설계			솔루션 설치 및 단위 테스트 운영 인프라 구축	업무 개발 지원 시스템 테스트		
Infra	솔루션 및 네트워크 구축	테스트 시스템 세팅							자세대 안정화 지원 서비스
IT	Server/Storage/NW	서버분석 설계	설계 시스템 이관	시스템 이관	통합 테스트	데이터 분석 및 대상 식별	모델 분석/연동 인터페이스 변경	테스트	서비스 이관
시제	기존 시스템 연동								



## 4

## 주요 고객 사례

2007년 제품을 출시한 이래 총 200여 적용 사례가 있으며, **자동 전환율 90% 이상의 효과를 활용하여 마이그레이션 품질 및 생산성 향상을** 기반으로 안정적으로 마이그레이션을 수행하였습니다.

고객명	프로젝트 유형	AS-IS 시스템 환경				TO-BE 시스템 환경				비 고
		O/S	Language	DBMS	Framework	O/S	Language	DBMS	Framework	
H 증권	프레임워크 upgrade	AIX	Java	Oracle	Nexcore 4.5	AIX	Java	Oracle	Nexcore 8.5	Java 160,000 본
S 은행	DW 교체	AIX	C	Teradata		AIX	C	Vertica		Shell 9200 본
H 은행	DW 교체	AIX	C	SybaseIQ		AIX	C	Exadata		Query 94,000 본
A 저축은행	차세대	AIX	COBOL	Oracle	BANCS		JAVA	Oracle		COBOL 22,000 본
S 은행	계정계 고도화	AIX	C / JAVA	Oracle12	BANCS	LINUX	C / JAVA	Oracle1 9c	BANCS	C 20,000본, JAVA 32,000본
K 증권	리테일증권 1/2차	HP-UX	C	Oracle	Nexcore	AIX	JAVA	Oracle	Nexcore	C 16,500본
K 은행	차세대 (TheK)	HP-UX	JAVA	Oracle	IBF	AIX	JAVA	DB2	Nexcore	인터넷뱅킹, JSP53,000, JAVA18,000, Weblogic → WebSphere
S 증권	U2L	UNIX	C / Pro*C	DB2		LINUX	JAVA	Oracle	Nexcore	Pro*C 25000본
S 증권	차세대	UNIX		Sybase		UNIX		DB2		Sybase SP 3,100 본
공공기관	차세대	AIX	EC / PL/SQL	UniSQL / Oracle		AIX	PL/SQL	Oracle		EC 5,400본, SP 4,,700본
H 화재	차세대	HP-UX	Pro*C	Oracle	Tmax Proc	HP-UX	JAVA	Oracle	AnyFrame	Pro*C 7,500본
S 카드	Vanguard PJT	Z/OS	CICS/COBOL	DB2	Systemier	HP-UX	C / Pro*C	Oracle	Systemier	CICS/COBOL 21.600본, Tuxedo
L 은행(일본)	플랫폼전환	AS/400	RPG	DB2		AIX	JAVA	DB2		RPG 1,800본



### 소프트웨어 분석 솔루션



### 서버 전환 솔루션



### 프로그램 언어 변환 솔루션



### 데이터베이스 전환 솔루션



Vanguardlab

