

Recall cost & Complexity !



RESORT for C(Pro*C) RESORT for Java(JSP)

소프트웨어 품질 및
생산성의 향상을 위한
소프트웨어 품질 관리 도구

RESORT for C(Pro*C) – SW 품질 도구

소프트웨어 품질 도구(SW Quality Tool)는 소프트웨어 품질 측정 및 평가를 통해서 소프트웨어 품질 향상 및 개선을 지원하는 솔루션입니다.

소프트웨어 품질 관리 도구(Software Metrics)는 코드의 최적화, 크기, 구조, 복잡도 등의 소프트웨어 품질 목표에 따라 잠재적 문제점을 식별, 진단, 해결할 뿐만 아니라 전체적인 소프트웨어 품질의 이해성, 유지보수성, 성능성 등에 대한 품질 향상/개선을 극대화할 수 있습니다.

역공학 도구(Reverse Engineering)는 소스 코드로부터 설계 정보를 추출하여 다양한 설계 문서 생성과 절차지향 소프트웨어 품질 측정을 지원하는 솔루션입니다.

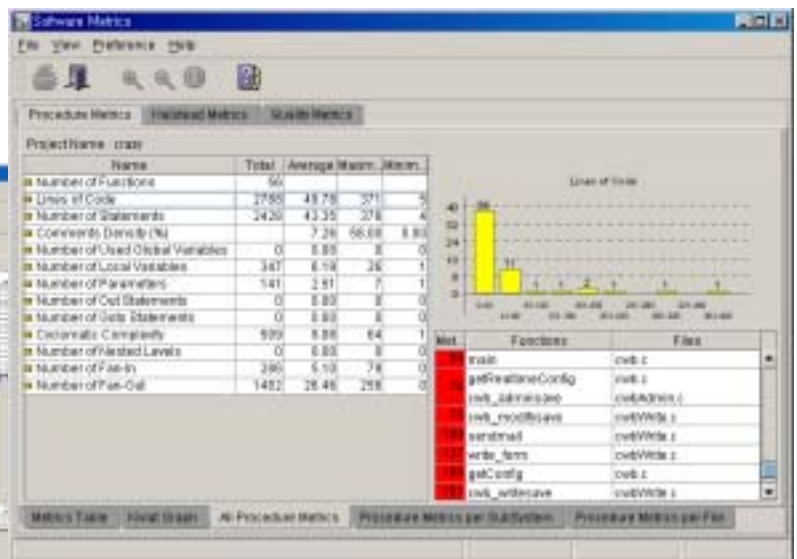
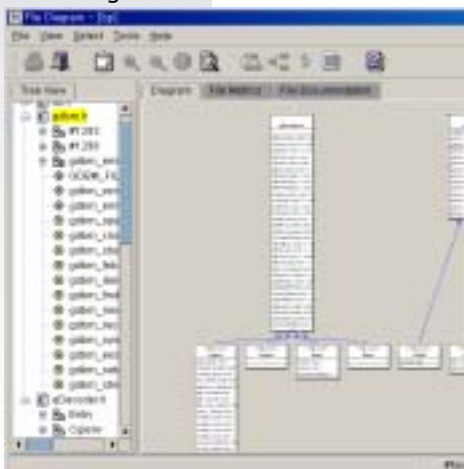
□ 주요기능

- UML 다이어그램/그래프와 절차지향 Metrics를 결합한 역공학 기법
- File에서 Function까지 완벽한 UML 다이어그램/그래프를 자동 생성
- 소프트웨어의 크기와 구조, 절차지향 특성, Halstead Metrics 등을 포함한 90개 이상의 Metrics 제공(라인 크기, 주석 비율, 내부 복잡도, 결합도 등등)
- 절차지향 특성, Halstead Metrics 및 ISO 9126-3의 유지보수성 측정 및 평가
- 다이어그램/그래프와 코드 간의 실시간 매핑
- 프로젝트 관리를 위한 다단계 통계 품질 보고서

□ 장점

- File에서 Function까지 소프트웨어 시스템의 상세 설계 및 아키텍처 이해
- 소프트웨어 개발과정에서 소프트웨어 잠재적 에러 조기 발견
- 소프트웨어 품질 목표에 따른 품질 진단
- 소프트웨어 이해성, 유지 보수성, 성능성 향상
- IT&QA 관리자의 소프트웨어 품질 관리 용이
- 소프트웨어 개발, 테스트 및 유지보수 비용 절감

File Diagram



Procedure Metrics –
Multi-level Statistical Metrics

□ Software Quality Solutions

- Multi-level Quality Statistical Metrics
 - Project
 - Subsystem
 - File
- Multi-level Automatic Visualization
 - Subsystem
 - File
 - Function

□ Visualization (Quality Evaluation)

- Procedure Metrics
- Halstead Metrics
- Quality Metrics

□ Visualization (Reverse Engineering)

- File Diagram
- Call Graph
- Variable Reference Graph
- Control Flow Graph
- Source Code Browser

RESORT for C(Pro*C) - 코드 인스펙션 도구

Code Checker Solutions

- Multi-level Code Audit
 - File
 - Function
 - MISRA-C:2004
- Multi-level Statistical Violation
 - Project
 - Subsystem
 - File
 - Function

Visualization (Inspection Monitoring)

- All Audit
- File Audit
- Function Audit
- MISRA Audit

코드 인스펙션 도구(Code Checker Tool)는 단지 소스 코드만으로 정적 분석함으로써 실제 Run-time에 발생할 수 있는 코딩 오류를 검출하고, 그리고 고급 코드를 작성하기 위한 코딩 표준을 지원하는 솔루션입니다.

코드 인스펙션 도구는 폭 넓게 적용되는 코딩 표준과 MISRA-C:2004 표준을 기반으로 코드 에러, 성능 저하, Hungarian 표기법, Memory 누수, Compile 에러, Run-time 에러 등의 위배 코드들을 검출할 뿐만 아니라 마치 한 명의 개발자가 전체 시스템의 소스 코드를 작성한 것처럼 일관된 코드의 구조와 스타일을 유지해 줍니다. 향후, 유지보수 및 테스트 비용을 획기적으로 절감하는 효과를 얻을 수 있습니다.

MISRA-C: 2004 코딩 가이드 라인은 고 품질 코드를 생산하기 위한 141개의 프로그래밍 규칙을 제시하고 있으며, 자동차, 전자, 우주 항공, 통신, 의료, 국방 등 제조 분야에서 산업 표준으로 많이 사용하고 있는 코딩 표준이며, 또한 임베디드 코딩 표준으로 널리 사용하고 있습니다.

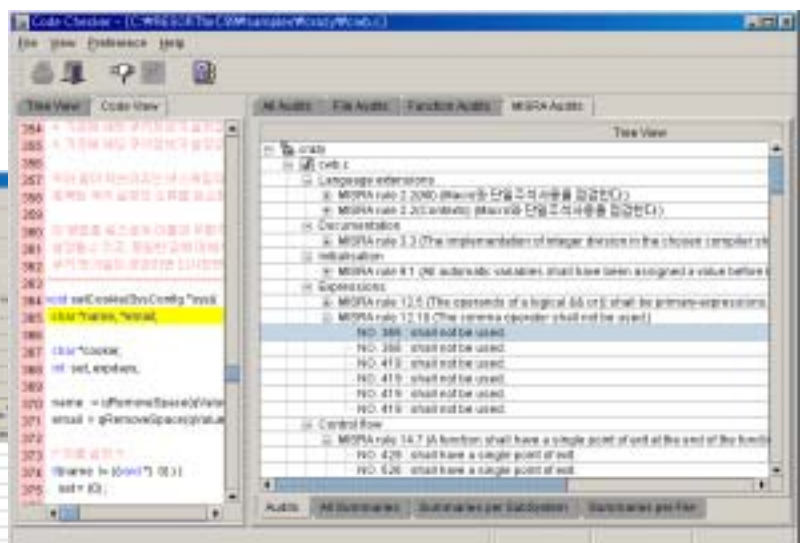
주요기능

- MISRA-C:2004(141 규칙), Null Pointer, Hungarian 표기법 등을 포함한 C 코드의 에러 및 오용된 패턴 검사
- 200개 이상의 코드 규칙 검사
- 검출된 오류 내용과 코드 간의 실시간 매핑
- 코딩 규칙 기준 설정 및 확장 지원
- 프로젝트 관리를 위한 다단계 통계 품질 보고서

장점

- 소프트웨어 개발 초기에 소스 코드 에러를 조기 발견 및 예방
- Compile, Run-time, 프로그래밍 에러 등 예방
- 코드의 가독성, 유지 보수성, 성능성, 안정성 향상
- 테스팅 단계로 전파되는 에러를 70-90% 발견
- 프로젝트의 프로그래밍 표준화 및 일관된 코딩 구조 유지
- 개발자의 C 프로그램 개발 능력을 향상
- IT&QA 관리자의 코드 품질 관리 용이
- 소프트웨어 개발, 테스트 및 유지보수 비용 절감

Code Checker - Statistical Violation Reporting



Code Checker - Code Violation Detection and Highlighting

RESORT for Java(JSP) – SW 품질 도구

□ Software Quality Solutions

• Multi-level Quality Statistical Metrics

- Project
- Package
- Class

• Multi-level Automatic Visualization

- Package
- Class
- Method

□ Visualization (Quality Evaluation)

- OO Metrics
- Package Metrics
- Halstead Metrics
- Quality Metrics
- System Metrics

□ Visualization (Reverse Engineering)

- Class(Package) Diagram
- Sequence(Collaboration) Diagram
- Control Flow Graph
- Data Flow Graph
- Source Code Browser

소프트웨어 품질 도구(SW Quality Tool)는 객체지향 소프트웨어 품질 측정 및 평가를 통해서 소프트웨어 품질 향상 및 개선을 지원하는 솔루션입니다.

소프트웨어 품질 관리 도구(Software Metrics)는 코드의 최적화, 크기, 구조, 복잡도, 객체지향 특성 등의 소프트웨어 품질 목표에 따라 잠재적 문제점을 식별, 진단, 해결할 뿐만 아니라 전체적인 소프트웨어 품질의 이해성, 유지보수성, 성능성 등에 대한 품질 향상/개선을 극대화할 수 있습니다.

역공학 도구(Reverse Engineering)는 소스 코드로부터 설계 정보를 추출하여 다양한 설계 문서 생성과 객체지향 소프트웨어 품질 측정을 지원하는 솔루션입니다.

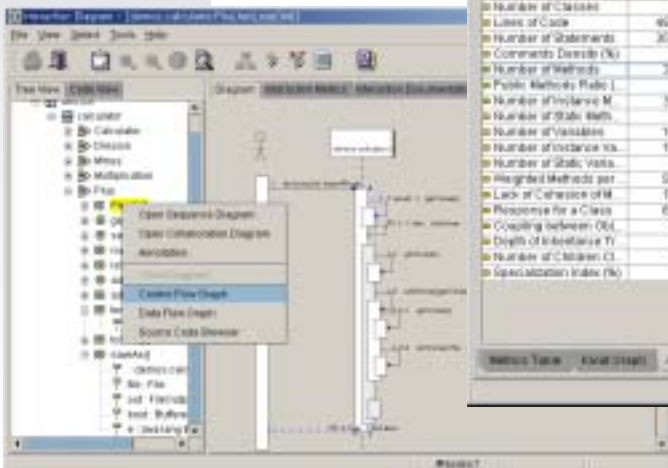
□ 주요기능

- UML 다이어그램/그래프와 객체지향 Metrics를 결합한 역공학 기법
- Package에서 Method까지 완벽한 UML 다이어그램/그래프를 자동 생성
- Class와 Table간의 영향 분석
- 소프트웨어의 크기와 구조, 객체지향 특성, Halstead Metrics Metrics 등을 포함한 100개 이상의 Metrics 제공(라인 크기, 주석 비율, 내부/외부 복잡도, 사용된 객체 수, 결합도, 재사용 비율 등등)
- 객체지향 특성, Halstead 및 Package Metrics 그리고 ISO 9126-3의 유지보수성 측정 및 평가
- 다이어그램/그래프와 코드 간의 실시간 매핑
- 프로젝트 관리를 위한 다단계 통계 품질 보고서

□ 장점

- Package에서 Method까지 소프트웨어 시스템의 상세 설계 및 아키텍처 이해
- 소프트웨어 개발과정에서 소프트웨어 잠재적 에러 조기 발견
- 소프트웨어 품질 목표에 따른 품질 진단
- 소프트웨어 이해성, 유지 보수성, 성능성 향상
- IT&QA 관리자의 소프트웨어 품질 관리 용이
- 소프트웨어 개발, 테스트 및 유지보수 비용 절감

Sequence Diagram



OO Metrics –
Multi-level Statistical Metrics

RESORT for Java(JSP) - 코드 인스펙션 도구

□ Code Checker Solutions

- Multi-level Code Audit
 - File
 - Class
 - Method
- Multi-level Statistical Violation
 - Project
 - Package
 - File
 - Class
 - Method

□ Visualization (Inspection Monitoring)

- All Audit
- File Audit
- Class Audit
- Method Audit

코드 인스펙션 도구(Code Checker Tool)는 단지 소스 코드만으로 정적 분석함으로써 실제 Run-time에 발생할 수 있는 코딩 오류를 검출하고, 그리고 고급 코드를 작성하기 위한 코딩 표준을 지원하는 솔루션입니다.

코드 인스펙션 도구는 폭 넓게 적용되는 Java(JSP)코딩 표준과 DB 인터페이스 가이드라인을 기반으로 코드 에러, 성능 저하, Hungarian 표기법, Memory 누수, Run-time 에러 등의 위배 코드들을 발견할 뿐만 아니라 마치 한 명의 개발자가 전체 시스템의 소스 코드를 작성한 것처럼 일관된 코드의 구조와 스타일을 유지해 줍니다. 향후, 유지보수 및 테스트 비용을 획기적으로 절감하는 효과를 얻을 수 있습니다.

DB 인터페이스 코딩 가이드 라인은 메모리 누수의 원인으로 WAS 또는 BDMS 중지를 발생할 수 있는 중요한 DB 자원 해제에 프로그래밍 규칙이다.

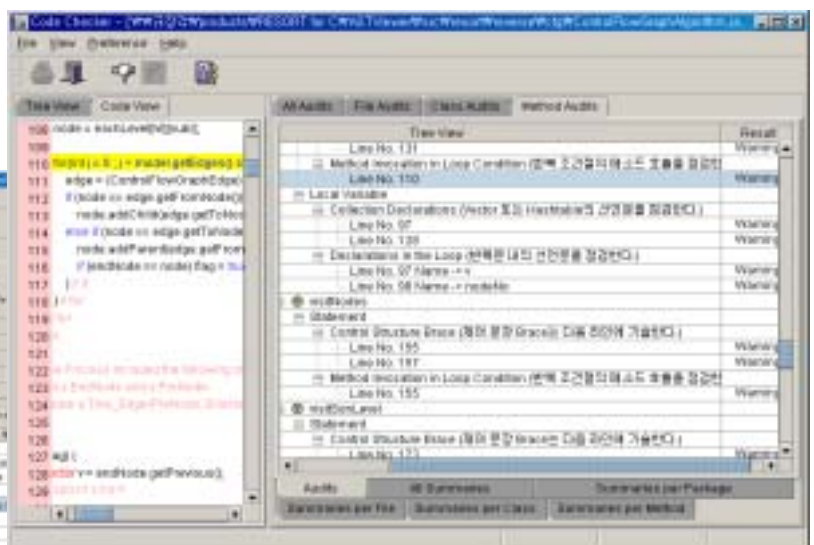
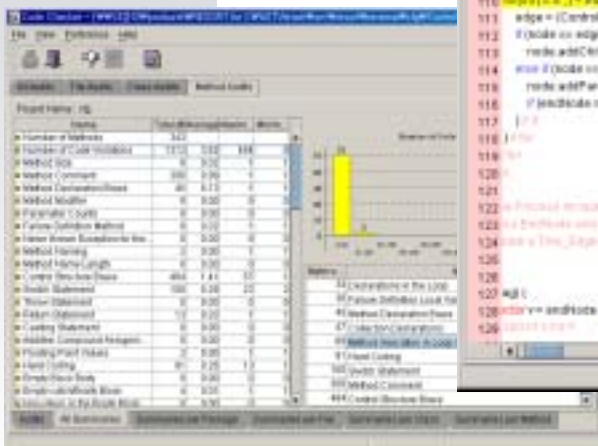
□ 주요기능

- EJB, JDBC, BC4J 등을 포함한 Java(JSP) 코드의 에러 및 오용된 패턴 검사
- 120개 이상의 코드 규칙 검사
- 검사된 코드 검사 항목과 코드 간의 실시간 매핑
- 코딩 규칙 기준 설정 및 확장 지원
- 프로젝트 관리를 위한 다단계 통계 품질 보고서

□ 장점

- 소프트웨어 개발 초기에 소스 코드 에러를 조기 발견 및 예방
- Compile, Run-time , 프로그래밍 에러 예방
- 코드의 가독성, 유지 보수성, 성능성, 안정성 향상
- 테스트 단계로 전파되는 에러를 70~90% 발견
- 프로젝트의 프로그래밍 표준화 및 일관된 코딩 구조 유지
- 개발자의 Java 프로그램 개발 능력을 향상
- IT&QA 관리자의 코드 품질 관리 용이
- 소프트웨어 개발, 테스트 및 유지보수 비용 절감

Code Checker – Statistical Violation Reporting



Code Checker – Code Violation Detection and Highlighting



RESORT for Java - 단위 및 통합 테스트 도구

□ Testing Solutions

- Test Plan
- Test Design
- Test Implementation
- Test Evaluation
- Multi-level Statistical Coverage
 - Project
 - Package
 - Class

□ Visualization

- Test Case Design
- Unit-level Monitoring
 - Control Flow Testing
 - Data Flow Testing
- Integration-level Monitoring
 - Sequence Testing

단위 및 통합 테스트 도구(Unit/Integration Testing Tool)는 테스트 관리, 테스트 설계, 테스트 실행, 테스트 평가 등에 대한 통합 테스트 프로세스를 실행할 수 있도록 코드 기반 White-box 및 Black-box 테스트 솔루션을 지원합니다.

Java 또는 Web 애플리케이션 개발 시, 소프트웨어 에러를 신속하게 점검하고, 진단하여, 해결할 수 있도록 지원할 뿐만 아니라, 소프트웨어 성능을 강화, 그리고 최적의 테스트 커버리지를 확보할 수 있도록 지원합니다.

□ 구성 요소

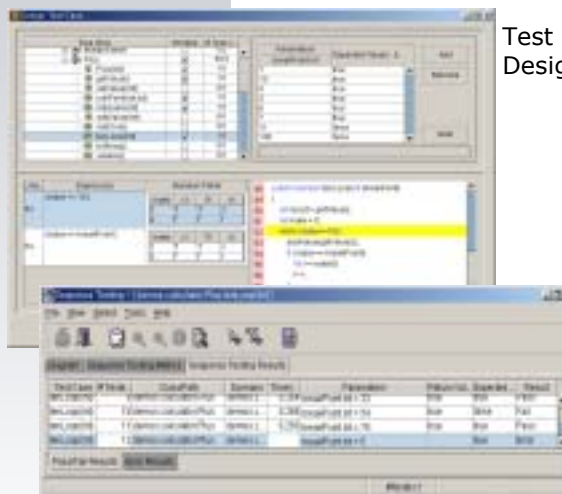
- 테스트 계획 및 테스트 케이스 설계
 - 단위 테스트를 위한 기본 경로(Basis path)
 - 통합 테스트를 위한 메시지 경로(ME(message execution) path)
 - Method의 테스트 케이스 설계, Class/Package/Project의 시나리오 설계
 - Code 및 Interface 커버리지 테스트
- 테스트 결과 분석
 - 테스트 케이스의 결과 분석(pass/fail/error message)
 - 30개 이상의 포괄적인 테스트 커버리지 및 성능 분석

□ 주요기능

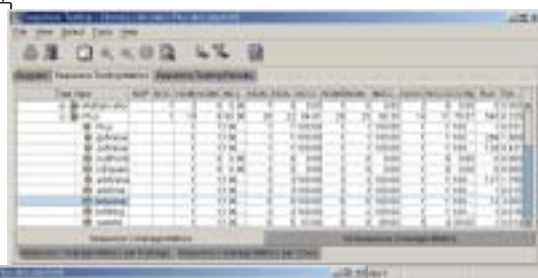
- UML 다이어그램/그래프와 테스트 솔루션을 결합한 테스트 기법
- 소스 코드로부터 테스트 케이스의 템플릿 자동 생성
- 테스트 케이스, 스위트, 시나리오 생성 및 관리
- 무장애(error-free) 테스트 드라이버 및 JUnit 드라이버 자동 생성
- 선택된 테스트 케이스 결과에 대한 그래픽 모니터링 및 테스트된 코드 식별
- 테스트 실행 중 발생된 에러(logic/interface)를 그래프 상에서 추적 및 분석
- 병목 지점 분석
- 시스템 관리를 위한 다단계 통계 커버리지 보고서
- 다이어그램/그래프들간의 이동
- 다이어그램/그래프와 코드 간의 실시간 매핑

□ 장점

- 간단한 테스트 계획, 테스트 케이스 설계, 그리고 유연한 테스트 프로세스 제공
- 소프트웨어 신뢰성 향상
- 객체지향 소프트웨어 품질 향상
- 테스트 및 디버깅의 시간/노력 단축



Test Case Design



Test Coverage-Sequence Coverage



Interface Error Monitoring

Test Case Analysis

RESORT for JavaFP - 기능 점수 산정 도구

기능 점수 산정 도구(Function Point Tool)는 Java 소스 코드로부터 사용자에게 제공하는 기능(데이터 및 트랜잭션 기능)을 논리적인 관점에서 식별하여 S/W 개발 또는 유지보수의 비용, 그리고 생산성을 객관적이고 정량적으로 측정할 수 있는 자동화 도구입니다.

이 솔루션은 IFPUG Counting Practices Manual 4.2와 소프트웨어사업대가기준(정보통신부 고시 제2006-18호)의 표준을 지원합니다.

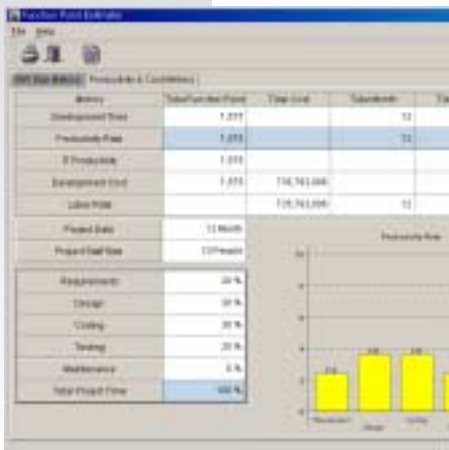
□ 주요기능

- UML 다이어그램, 객체지향 Metrics, 그리고 기능 점수 솔루션을 결합한 기능 점수 산정 기법
- 코드 및 요구 사항 기반 기능 점수 자동 측정
- 개발 프로젝트, 개선 프로젝트, 애플리케이션의 기능 점수 측정
- 소프트웨어 개발비 및 유지보수 비용 산정
- 소프트웨어 생산성 및 비용 분석
- 시스템 관리를 위한 다단계 통계 품질 보고서 및 최종 기능 문서화
- 언어별 SLOC(Source Lines of Code) 예측
- 다이어그램과 코드 간의 실시간 매핑

□ 장점

- 논리적 관점에서 소프트웨어 규모 측정
- 소프트웨어 전 생명주기 적용
- 소프트웨어 기능 분석 및 이해
- 소프트웨어 개발 변경 관리 및 개발비 사후 정산 용이
- 소프트웨어 공정별 비용 산정 용이
- 프로젝트 및 조직 관리 향상
- 소프트웨어 프로세스 개선 - CMMI 3 레벨 이상에서 필수
- 기능 점수 산정을 위한 시간/노력 단축

Function Point Estimator - Productivity & Cost Metrics



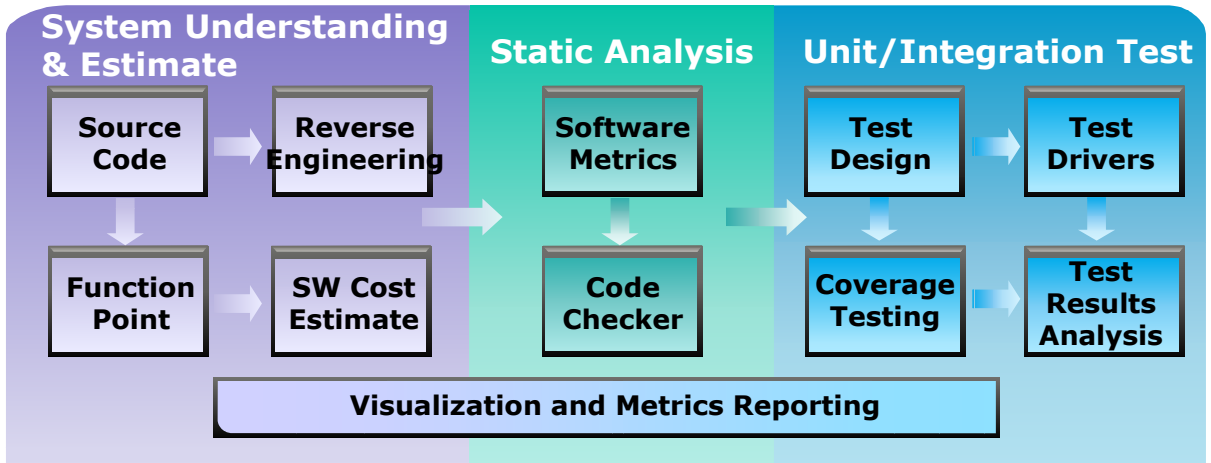
Function Point Estimator - S/W Size Metrics

Soft4Soft Solution Map

□ 소프트웨어 품질 및 기능 점수 산정을 위한 소프트4소프트 솔루션

RESORT는 Java, JSP, C, C++, C# 등의 언어로 작성된 소프트웨어를 구현초기부터 테스트까지의 소프트웨어 개발 및 유지 보수를 할 수 있도록 소프트웨어 품질과 프로세스를 지원하는 새로운 통합 소프트웨어 품질 솔루션 도구입니다. RESORT는 귀사의 time-to-market에 대한 대처 능력 및 전반적인 개발 비용을 절감시켜 드릴뿐만 아니라, 소프트웨어 품질을 향상시켜 드립니다.

- ✓ 기존 시스템의 구조 분석 및 이해
- ✓ 소프트웨어의 품질을 초기에 관리
- ✓ 품질 프로세스 기반 품질보증 솔루션
- ✓ 단위 및 통합 테스트를 위한 통합 테스트 프로세스
- ✓ 소프트웨어 개발비 및 유지보수 비용 산정
- ✓ 소프트웨어 생산성 및 비용 평가
- ✓ IT&QA 관리자와 개발자간의 대화 용이



□ Soft4Soft 제품군

- ✓ RESORT for Java(JSP), JavaFP
- ✓ RESORT for C(Pro*C)
- ✓ RESORT for C# (출시 예정)
- ✓ RESORT for C++ (출시 예정)

□ 플랫폼 및 시스템 사양

- ✓ Java 플랫폼 : Windows 2000/XP/NT, Linux (Red Hat 7.3 higher), UNIX
- ✓ 시스템 요구 사항 : SUN JDK 1.4.x or higher

□ 적용 분야

- ✓ 소프트웨어 개발 및 감리 업체, 기업 IT 부분
- ✓ 자동차, 전자, 우주 항공, 통신, 의료, 국방 등 제조 분야 (MISRA-C: 2004)
- ✓ CMMI, ISO 9126-3, or 9001 등의 국제 표준의 프로세스를 채택한 업체
- ✓ 소프트웨어 품질 교육, IT 교육 센터
- ✓ 소프트웨어 외주 관리(프로그램 검수)

www.soft4soft.com

대전광역시 유성구 문지동 103-6, ICU 창업보육센터 T215호
전화 042-866-6632~3
팩스 042-866-6626
Sales{Info}@soft4soft.com

(주) Soft4Soft

Soft4Soft는 구현 초기부터 테스트까지의 소프트웨어 개발 및 유지보수를 지원하는 역공학, 품질 보증, 테스트, 기능점수 등의 소프트웨어 품질 도구를 연구하고 개발하는 전문 품질 솔루션 공급 업체입니다.

Soft4Soft Solutions

Soft4Soft는 소프트웨어 Metrics 기반으로 소프트웨어 분석 및 이해, 품질, 테스트, 기능 점수 산정 등을 지원하는 통합 소프트웨어 품질 솔루션을 RESORT 브랜드로 제공하고 있습니다.

Customer Support & Services

Soft4Soft는 품질 솔루션에 대한 전문적인 지식을 제공함으로써, 고객이 RESORT 제품을 사용하여 단기간 내에 성공적인 결과를 거두실 수 있도록 최상의 서비스를 지원합니다.

- ✓ 전화 및 E-mail 지원
- ✓ 업그레이드 서비스
- ✓ 교육
- ✓ 테스트 및 품질 컨설팅

RESORT 제품에 대한 자세한 정보를 원하시면, 저희 홈페이지 www.soft4soft.com을 방문하여 주시길 바랍니다.

www.soft4soft.com

대전광역시 유성구 문지동 103-6, ICU 창업보육센터 T215호
전화 042-866-6632~3
팩스 042-866-6626
Sales{Info}@soft4soft.com