



JBUILDER® 2008

	TURBO	PROFESSIONAL	ENTERPRISE
인증된 Eclipse 번들			
통합 설치	>>	>>	>>
샘플 응용 프로그램	>>	>>	>>
WTP 2.0이 포함된 Eclipse 3.3.2(Europa) Framework:			
BIRT(Business Intelligence and Reporting Tools)	>>	>>	>>
CVS 플러그 인	>>	>>	>>
Dali Java 지속성 도구(WTP의 일부)	>>	>>	>>
DTP(Data Tools Platform)	>>	>>	>>
EMF(Eclipse Modeling Framework)	>>	>>	>>
GEF(Graphical Editing Framework)	>>	>>	>>
JST(J2EE Standard Tools)	>>	>>	>>
JDT(Java Development Tools)	>>	>>	>>
Mylyn(이전 Mylar)	>>	>>	>>
PDE(Plug-in Development Environment)	>>	>>	>>
RCP(Rich Client Platform)	>>	>>	>>
TPTP(Test and Performance Tools Platform)	>>	>>	>>
VE(Visual Editor)	>>	>>	>>
WST(Web Standard Tools)	>>	>>	>>
추가 Eclipse 및 써드파티 플러그인(일부 목록):			
Apache	>>	>>	>>
Apache Ant	>>	>>	>>
Apache Maven 2.0 Integration	>>	>>	>>
Apache Xerces	>>	>>	>>
AJDT(Eclipse AspectJ Development Tools)	>>	>>	>>
ECF(Eclipse Communication Framework)	>>	>>	>>
Eclipse GMF(Graphical Modeling Framework)	>>	>>	>>
Eclipse JSF(JavaServer Face Tools)	>>	>>	>>
EMFT(Eclipse Modeling Framework Technology)	>>	>>	>>
Eclipse UML2	>>	>>	>>
Eclipse XSD(Eclipse MDT의 일부)	>>	>>	>>
Hibernate	>>	>>	>>
JUnit	>>	>>	>>
mySQL	>>	>>	>>
Spring IDE	>>	>>	>>
Java EE 지원			
EJB 2.1 프로젝트의 LiveSource 그래픽 뷰		>>	>>
EJB 3.0 프로젝트의 LiveSource 그래픽 뷰		>>	>>
JPA(Hibernate/TopLink) 프로젝트의 LiveSource 그래픽 뷰		>>	>>
엔티티, 세션 및 메시지 bean을 시각적으로 설계할 수 있는 포인트 앤 클릭 양방향 디자이너. 클래스 및 주석의 양방향 생성		>>	>>
시각적으로 CMP 2.x 관계 설계 및 데이터베이스 매핑 구성		>>	>>
시각적 방식의 EJB 3.0/JPA 엔티티 관계 설계 및 데이터베이스 매핑 구성		>>	>>

	TURBO	PROFESSIONAL	ENTERPRISE
JBUILDER에서 지원되는 서버 런타임 XDoclet 작업 속성 구성		>>	>>
시각적으로 EJB 3.0/JPA의 지속성 속성 구성		>>	>>
전체 빌드/배포/재배포 기능		>>	>>
시각적으로 세션 bean 작성		>>	>>
설계 창에서 bean 자동 배열		>>	>>
여러 설계 창을 작성하여 EJB 그룹에서 bean의 논리적 그룹화 가능		>>	>>
필터링 조정으로 생성된 클래스를 숨기거나 표시		>>	>>
공급업체별 설명자를 포함하여 지원되는 모든 서버에 대한 EJB 설명자 가져오기		>>	>>
시각적으로 메시지 중심 bean 작성		>>	>>
시각적으로 엔티티 bean 작성		>>	>>
설명자 기반 소스를 XDoclet 또는 Java EE 5 주석으로 변환		>>	>>
다중 모듈 Java EE 프로젝트를 이전 버전 JBUILDER에서 가져와서 네이티브 Eclipse 프로젝트로 변환		>>	>>
기존 데이터베이스 스키마에서 EJB 2.x 엔티티 bean 생성		>>	>>
기존 데이터베이스 스키마에서 EJB 3.x 엔티티 bean 생성		>>	>>
디자이너 및 소스에서 시각적 EJB 구성 요소 즉시 탐색		>>	>>
시각적으로 OCL 제약 조건 작성		>>	>>
Eclipse 3.3 Framework 및 Eclipse WTP 2.0과 완벽하게 통합		>>	>>
VisiBroker Builder - VisiBroker 도구에 통합		>>	>>
메소드와 작업을 포함하는 시각적 Java/EJB 표현		>>	>>
지원되는 응용 프로그램 서버			
Apache Geronimo 1.1.1	>>	>>	>>
Apache Geronimo 2.0	>>	>>	>>
Apache Tomcat 6.0	>>	>>	>>
BEA WebLogic Application Server 10.0	>>	>>	>>
BEA WebLogic Application Server 8.1	>>	>>	>>
BEA WebLogic Application Server 9.2	>>	>>	>>
Borland Enterprise Server 6.7	>>	>>	>>
IBM WebSphere 6.0	>>	>>	>>
IBM WebSphere 6.1(EJB 3 기능 팩 포함)	>>	>>	>>
JBoss 3.2.3	>>	>>	>>
JBoss 4.0.5	>>	>>	>>
JBoss 4.2.2	>>	>>	>>
Oracle Application Server 10.1.3.2	>>	>>	>>
Oracle Containers for Java(OC4J) 10.1.3.2	>>	>>	>>
Sun Glassfish 1.0	>>	>>	>>
Sun Glassfish 1.1	>>	>>	>>
Sun Glassfish 2.0	>>	>>	>>
웹 서비스 지원			
Web Services Designer			
Axis 기반 웹 서비스를 사용할 수 있는 응용 프로그램을 만들기 위한 시각적 양방향 디자이너		>>	>>
단순화된 사용자 인터페이스 및 마법사		>>	>>
메소드와 작업을 포함하는 시각적 WSDL 표현		>>	>>
디자이너에서 구성 요소를 제거하지 않고 서비스 활성화 또는 비활성화		>>	>>
디자이너 및 소스에서 시각적 웹 서비스 구성 요소 즉시 탐색		>>	>>

	TURBO	PROFESSIONAL	ENTERPRISE
웹 서비스 프로젝트를 이전 버전의 Jbuilder에서 가져와서 네이티브 Eclipse 프로젝트로 변환		>>	>>
EJB를 웹 서비스로 배포		>>	>>
웹 서비스 빌드, 디버그 및 실행			
Axis 배포 환경 작성		>>	>>
프로젝트 빌드 중에 웹 서비스 구성 요소 동적 생성 및 배포		>>	>>
Tomcat, JBoss, WebLogic, WebSphere, Geronimo, Glassfish, Oracle에 Axis 런타임 배포		>>	>>
ProjectAssist 및 TeamInsight를 이용한 공동 작업 및 팀 개발			
새 팀 프로젝트 환경 정의 작성			
새 배포 모델 작성 마법사			>>
기존 서비스를 다양한 수준으로 검색하는 옵션			>>
소스 코드 관리를 위한 기존 설치 동화 또는 새 설치 작성			>>
결함 추적을 위한 기존 설치 동화 또는 새 설치 작성			>>
요구 사항 관리 및 프로젝트 계획을 위한 기존 설치 동화 또는 새 설치 작성			>>
연속 통합 빌드를 위한 기존 설치 동화 또는 새 설치 작성			>>
모든 서비스에서 새 서버 또는 기존 서버 배포 시 새 프로젝트 작성			>>
새 사용자 작성, 새 프로젝트 또는 기존 프로젝트에 사용자 추가			>>
Mylar(OSS)	>>	>>	>>
XPlanner Mylar Connector		>>	>>
StarTeam Mylar Connector			>>
새 Teaminsight 스택 설치			
대상 서버에서 다양한 수준으로 잠재적 충돌 검색			>>
모든 서비스의 전체 배포 및 자동 구성			>>
배포를 자동으로 테스트하여 시스템 작동 확인			>>
Teaminsight 지원 도구			
Bugzilla			>>
Continuum			>>
CVS			>>
StarTeam			>>
Subversion			>>
XPlanner			>>
프로젝트 포털			
프로젝트에 대한 소스 코드 저장소에서 활동 모니터링, 최근 체크인 추적			>>
심각도, 제품 영역, 소유자, 최신 버그 및 버그 찾기/수정율별 버그 테이블을 포함하여 품질 메트릭을 모니터링.			>>
실시간 차트를 통해 팀 속도 모니터링			>>
커밋된 기능에 대해 기능별 팀 진행 상황 모니터링			>>
연속 통합 빌드 모니터링, 실패한 빌드를 추적하여 근본 원인 식별			>>
IDE 통합			
CVS에서 프로젝트를 가져오도록 IDE 자동 구성			>>
Subversion에서 프로젝트를 가져오도록 IDE 자동 구성			>>
개별 프로젝트 책임에 대한 단일 창 뷰: 할당된 작업, 소유한 요구 사항, 추적한 요구 사항, 할당된 버그, 보고된 버그, 할 일 코드			>>
IDE에서 통합된 탭 브라우저를 통해 모든 서비스 및 프로젝트 포털 사용 가능			>>
버그 보고서 통합 작성 및 편집			>>
요구 사항 및 작업 통합 작성 및 편집			>>

	TURBO	PROFESSIONAL	ENTERPRISE
자동 동기화를 통해 요구 사항, 작업, 결합에 지속적으로 액세스할 수 있는 완전 오프라인 모드			>>
서버를 사용하지 않는 LAN 피어 검색 및 채팅	>>	>>	>>
피어 코드 리뷰			>>
Open API를 사용하여 Perforce, ClearCase, ClearQuest 등 널리 사용되는 다른 버전 관리 시스템과 통합 가능		>>	>>
추가 생산성 기능			
RMI Builder		>>	>>
JNI Builder		>>	>>
JBUILDER 2006 프로젝트 가져오기	>>	>>	>>
품질 보증: 코드 감사 및 메트릭			>>
코드 템플릿 설계 및 재사용을 통한 설계 패턴 지원(GoF 패턴 포함)		>>	>>
Optimizeit™ Technology에서 제공하는 코드 프로파일 및 성능 조정 기능			
JBUILDER와 완벽하게 통합		>>	>>
코드 수준의 성능 문제 식별 및 해결을 위한 완전한 코드 프로파일 및 성능 관리 기능		>>	>>
다음과 비롯한 모든 Java 코드의 성능 및 안정성 향상: Java 응용 프로그램, Java EE 응용 프로그램, 서블릿, 애플릿, EJB, JavaBeans, JSP 응용 프로그램 및 Java 태그 라이브러리		>>	>>
원격 Java 프로세스에 손쉽게 연결하여 JBUILDER 외부에서 실행된 프로그램이나 다른 컴퓨터에서 실행 중인 프로그램 테스트		>>	>>
OPTIMIZEIT에서 제공하는 메모리 및 Cpu 프로파일			
성능 문제가 CPU나 메모리, 또는 두 가지 모두와 관련되어 있는지 여부를 확인할 수 있는 높은 수준의 성능 관련 데이터가 실시간으로 표시됩니다.		>>	>>
Automatic Memory Leak Detector(자동 메모리 누수 탐지기)가 잠재적인 메모리 누수 가능성을 즉시 식별할 수 있도록 시간 경과에 따른 메모리 사용량의 변화를 모니터링합니다.		>>	>>
실시간 개체 할당 모니터링을 통해 프로파일된 프로그램의 가상 컴퓨터 메모리 사용에 대해 이해할 수 있으며 Allocation Backtrace View(할당 역추적 뷰)를 사용하여 인스턴스 할당을 담당하는 프로그램의 코드 또는 부분을 식별할 수 있습니다.		>>	>>
Object Size Display(개체 크기 표시)는 클래스의 모든 인스턴스에서 사용 중인 메모리의 양을 자동으로 계산하여 실시간으로 표시하고, 메모리를 가장 많이 사용하는 개체부터 나열되도록 개체 크기별로 정렬합니다.		>>	>>
축소된 참조 그래프에서 전체 참조 그래프의 이행적 폐쇄를 제공하여 가비지 컬렉션 개체를 비우기 위해 제거해야 하는 참조만 표시합니다.		>>	>>
CPU Profiler(CPU 프로파일러)는 샘플링 기반 프로파일러 또는 계측 기반 프로파일러 사용 옵션을 통해 프로파일 세션 중 순수한 CPU 또는 시간 사용을 측정합니다.		>>	>>
프로파일 세션 중에 사용된 스레드를 색상으로 강조 표시하고 스레드 및 스레드 그룹별 프로파일 정보를 표시합니다.		>>	>>
HotSpot Display(핫스팟 표시)로 가장 많은 시간이 소비된 메소드를 나열하여 단일 메소드로 인한 병목 현상을 식별할 수 있습니다.		>>	>>
확장 가능한 호출 그래프로 중요 코드를 시각적으로 고립시킵니다. 문자열 할당을 선택하고 메소드 호출 흐름에 강조 표시하면 사용 중인 메모리 및 소비된 시간을 확인할 수 있습니다.		>>	>>
Automatic Application Quality Analyzer(응용 프로그램 품질 자동 분석기)는 VM 수준의 성능 버그를 자동으로 검색하여 성능 오류 방지 및 코딩 표준을 지원합니다.		>>	>>
XML, HTML 및 CSV 형식으로 뷰를 내보낼 수 있습니다.		>>	>>
OPTIMIZEIT에서 제공하는 스레드 디버거			
가상 컴퓨터에서 실행 중인 모든 스레드의 진행 상황을 실시간으로 표시		>>	>>
경합 뷰의 세부 정보 패널을 사용하여 모니터에 대한 스레드 경합 확인		>>	>>
대기 상태 모니터링을 통해 Waiting View(대기 뷰) 및 I/O Waiting View(I/O 대기 뷰)에서 스레드가 진행되지 않는 원인 확인		>>	>>
Monitor Enter View(모니터 시작 뷰) 보고서를 사용하여 스레드에서 모니터를 시작하고 유지하는 초과 잠금을 식별한 후 수정		>>	>>
스레드와 모니터 간의 관계 그래픽 뷰를 통해 교착 상태를 분석하여 교착 상태를 신속하게 파악		>>	>>

	TURBO	PROFESSIONAL	ENTERPRISE
잠금 명령 경고, 잠금 및 대기 경고, 잠금 및 I/O 대기 경고 등과 같이 교착 상태와 성능 병목 현상을 초래할 수 있는 모든 경고 및 오류 목록을 생성하는 Monitor Usage Analyzer(모니터 사용 분석기)를 사용하여 교착 상태를 예측합니다.	>>	>>	>>
OPTIMIZEIT에서 제공하는 코드 적용 범위			
실시간 Class Coverage View(클래스 적용 범위 뷰)를 사용하여 각 클래스의 적용 범위를 빠르게 확인하고 완전히 적용되지 않는 클래스를 식별할 수 있습니다.	>>	>>	>>
테스트 대상 프로그램에서 사용되는 모든 클래스와 인터페이스를 실시간으로 표시하고 클래스당 포함된 줄의 실시간 비율을 표시합니다.	>>	>>	>>
Method Coverage View(메소드 적용 범위 뷰)는 선택한 클래스에 대해 사용되지 않은 코드 줄 및 메소드를 표시하므로 개발자가 모든 코드 영역을 포함하도록 테스트 계획을 수정할 수 있습니다.	>>	>>	>>
Source Code Viewer(소스 코드 뷰어)는 비활성 코드를 쉽게 확인할 수 있도록 실행된 적이 없는 코드 줄을 표시합니다.	>>	>>	>>
가상 컴퓨터에서 로드된 인터페이스와 로드되지 않은 인터페이스를 표시하는 옵션이 제공됩니다.	>>	>>	>>
배치 모드 테스트 프로세스에서 코드 적용 범위를 쉽게 포함하도록 배치 모드를 지원합니다.	>>	>>	>>
OPTIMIZEIT에서 제공하는 리퀘스트 분석기			
JDBC, JSP, JNDI, Enterprise JavaBeans 및 JMS 컨테이너와 같은 Java EE 구성 요소 전체에서 Java EE 응용 프로그램 코드의 성능 동작을 프로파일합니다.	>>	>>	>>
Java EE 구성 요소에 대한 드릴다운 성능 정보를 사용하여 개발 초기에 Java EE 관련 응용 프로그램 코드의 성능 및 안정성을 향상시킬 수 있습니다.	>>	>>	>>
그래픽 표현을 사용한 시각적 인터페이스로 Java EE 응용 프로그램 상호 작용의 복잡도를 줄입니다.	>>	>>	>>
System Dashboard(시스템 대시보드) 뷰에는 Java EE 구성 요소에서 소비된 응용 프로그램 시간과 총 리퀘스트 수가 그래픽으로 표시됩니다. 각 서버 모듈의 사용률을 표시하여 주요 구성 요소 수준의 성능 문제를 신속하게 감지합니다.	>>	>>	>>
System Composite(시스템 복합) 뷰는 응용 프로그램에서 발생한 모든 Java EE 이벤트를 적절한 계층 구조로 실시간으로 표시합니다. 계층 구조에는 생성된 이벤트 간의 관계가 표시됩니다.	>>	>>	>>
Together® Technology에서 제공하는 UML 모델링 및 코드 고고학			
HTML 포털 문서 생성	>>	>>	>>
템플릿을 이용한 문서 생성	>>	>>	>>
UML 다이어그램에서 이미지 파일 생성	>>	>>	>>
시퀀스 다이어그램 자동 생성	>>	>>	>>
유스케이스 다이어그램			>>
활동 다이어그램	>>	>>	>>
시퀀스 다이어그램	>>	>>	>>
클래스 다이어그램	>>	>>	>>
구성 요소 다이어그램			>>
배포 다이어그램			>>
상태 시스템 다이어그램			>>
복합 구조 다이어그램			>>
통신 다이어그램			>>
웹 서비스 다이어그램	>>	>>	>>
JBUILDER 응용 프로그램 팩터리를 이용한 개발자 생산성 및 코드 재사용			
시각적으로 코드를 구성하고, 변경 내용을 추적하고, 변경 내용을 작업에 연결하고, 과거의 작업을 데이터 마이닝하고, 원하는 사용자 사례 또는 작업의 컨텍스트에서 모든 프로젝트 아티팩트를 연결하는 기능			>>
응용 프로그램 모듈 편집기			>>
응용 프로그램 모듈 런타임			>>
응용 프로그램 팩터리 탐색기			>>

JBUILDER 2008

TURBO PROFESSIONAL ENTERPRISE

레시피 편집기			>>
Struts 2, Spring MVC, JSF용 데이터 인식 웹 응용 프로그램 모듈			>>
전자 상거래 응용 프로그램 모듈			>>
애견센터 및 서점용 템플릿 응용 프로그램 모듈			>>
세 가지 사용자 역할 지원: 소비자 역할, 생산자 역할, 임시 개발자 역할			>>
응용 프로그램 모듈 스크립트를 통해 템플릿 코드를 생성할 수 있게 해주는 코드 생성/템플릿 메커니즘			>>
JBuilder Swing Designer			
코드 생성 옵션			
양방향 코드 생성	>>	>>	>>
거의 모든 스타일 읽기 및 쓰기	>>	>>	>>
차단 모드(기본값)	>>	>>	>>
일반 모드	>>	>>	>>
접두사 구성 요소 만들기		>>	>>
변수 공유		>>	>>
제어 변수 선언		>>	>>
이벤트 처리기 스텝 만들기		>>	>>
기본 변수 이름 정의		>>	>>
기본값을 사용하여 선택한 위젯 필드 만들기		>>	>>
기본값을 사용하여 모든 위젯 필드 만들기		>>	>>
이벤트 처리기(익명 클래스)		>>	>>
이벤트 처리기(내부 클래스)		>>	>>
부모 클래스에서 수신기 인터페이스 구현		>>	>>
특수 기능			
수동 작성 코드 리버스 엔지니어링	>>	>>	>>
사용자 코드 리팩터링 지원	>>	>>	>>
자유 형식 코드 편집	>>	>>	>>
시각적 상속		>>	>>
사용자 정의 위젯 지원		>>	>>
팔레트 관리자		>>	>>
임베디드 합성 지원		>>	>>
사용자 정의 위젯 템플릿		>>	>>
위젯 모핑		>>	>>
레이아웃 도우미		>>	>>
그래픽 탭 순서 편집		>>	>>
다중 위젯 선택 및 편집		>>	>>
움직이는 텍스트 선택		>>	>>
다중 위젯 정렬		>>	>>
너비 및 높이 복제		>>	>>
CLabels용 그라데이션 편집기		>>	>>
무제한 실행 취소/다시 실행	>>	>>	>>
잘라내기/복사/붙여넣기	>>	>>	>>
다중 위젯 추가를 위한 고정 모드	>>	>>	>>
외부 문자열 표시	>>	>>	>>
여러 로깅 레이블 표시		>>	>>

JBUILDER 2008



	TURBO	PROFESSIONAL	ENTERPRISE
국가별 도구 및 마법사		>>	>>
리소스 번들 자동 업데이트		>>	>>
편집기 레이아웃 제어	>>	>>	>>
SWT 특수 기능			
SWT_AWT		>>	>>
합성 추출 리팩터링		>>	>>
CLabels용 그라데이션 편집기		>>	>>
간단하게 인쇄 모드	>>	>>	>>
SWING 마법사			
Swing JFrame	>>	>>	>>
Swing JDialog		>>	>>
Swing JPanel		>>	>>
Swing JApplet		>>	>>
Swing JInternalFrame		>>	>>
Swing Application		>>	>>
SWING 컨테이너			
JPanel	>>	>>	>>
JScrollPane	>>	>>	>>
JSplitPane	>>	>>	>>
JTabbedPane	>>	>>	>>
JToolBar	>>	>>	>>
JDesktopPane		>>	>>
JInternalFrame		>>	>>
SWING 레이아웃			
그래픽 레이아웃 피드백	>>	>>	>>
Null/절대	>>	>>	>>
FlowLayout	>>	>>	>>
BorderLayout	>>	>>	>>
GridLayout	>>	>>	>>
CardLayout		>>	>>
GridBagLayout		>>	>>
BoxLayout/Struts 및 Glue		>>	>>
SpringLayout		>>	>>
JGoodies FormLayout		>>	>>
Java 6 GroupLayout		>>	>>
SWING 컨트롤			
사용자 정의/써드파티 위젯 지원		>>	>>
JButton	>>	>>	>>
JCheckBox	>>	>>	>>
JRadioButton	>>	>>	>>
JToggleButton	>>	>>	>>
JLabel	>>	>>	>>
JTextField	>>	>>	>>
JPasswordField	>>	>>	>>
JTextArea	>>	>>	>>
JFormattedTextField		>>	>>
JSlider		>>	>>
JScrollBar		>>	>>

JBUILDER 2008

	TURBO	PROFESSIONAL	ENTERPRISE
JList	>>	>>	>>
JComboBox	>>	>>	>>
JSpinner		>>	>>
JTree	>>	>>	>>
JTable	>>	>>	>>
JToolBar.Separator	>>	>>	>>
JSeparator		>>	>>
JProgressBar		>>	>>
JOptionPane		>>	>>
JTestPane		>>	>>
JEditorPane		>>	>>
작업		>>	>>
ButtonGroup		>>	>>
SWING 메뉴			
그래픽 메뉴 편집		>>	>>
JMenuBar		>>	>>
JMenu		>>	>>
JPopupMenu		>>	>>
JMenuItem		>>	>>
JCheckBoxMenuItem		>>	>>
JRadioButtonMenuItem		>>	>>
메뉴 구분 기호		>>	>>
AWT 위젯			
패널		>>	>>
스크롤 창		>>	>>
버튼		>>	>>
레이블		>>	>>
확인란		>>	>>
선택		>>	>>
목록		>>	>>
스크롤 막대		>>	>>
텍스트 필드		>>	>>
텍스트 영역		>>	>>
SWING 모양 및 느낌			
Windows	>>	>>	>>
CDE/테마	>>	>>	>>
금속	>>	>>	>>
JGoodies 플라스틱		>>	>>
JGoodies 플라스틱 3D		>>	>>
JGoodies 플라스틱 XP		>>	>>
JGoodies Windows		>>	>>
역체		>>	>>
Kunststoff		>>	>>

CODEGEAR 정보

CodeGear의 목표는 컴퓨팅 환경에 관계 없이 개발자들에게 응용 프로그램 구축 작업을 단순화시키는 최첨단 도구를 제공하는 것입니다. CodeGear의 IDE 솔루션을 통해 개발자들은 특정 공급업체의 기술에 의존할 필요 없이 추가 프로그래밍 시간을 들이지 않고도 원하는 대로 플랫폼을 자유롭게 개발할 수 있습니다.

Copyright© 2008 CodeGear™ All rights reserved. 모든 CodeGear 브랜드 및 제품 이름은 미국 및 다른 국가에서 Borland Software Corporation의 서비스 표시, 상표 또는 등록 상표입니다. 다른 모든 표시는 해당 소유자의 자산입니다.