

버전 6.1.0



## WebSphere Process Server 모니터링



버전 6.1.0



WebSphere Process Server 모니터링

주:

이 정보와 이 정보가 지원하는 제품을 사용하기 전에, 이 문서의 맨 끝에 있는 주의사항 섹션의 일반 정보를 읽으십시오.

**2008년 2월 1일**

이 개정판은 새 개정판에서 별도로 명시하지 않는 한 멀티플랫폼용 WebSphere Process Server의 버전 6, 릴리스 1, 수정판 0(제품 번호: 5724-L01) 및 모든 후속 릴리스와 수정판에 적용됩니다.

이 문서에 대한 의견을 보내려면 [doc-comments@us.ibm.com](mailto:doc-comments@us.ibm.com)으로 전자 우편 메시지를 전송하십시오. 여러분의 의견을 기대하고 있습니다.

IBM에 정보를 보내는 경우, IBM은 귀하의 권리를 침해하지 않는 범위 내에서 적절하다고 생각하는 방식으로 귀하가 제공한 정보를 사용하거나 배포할 수 있습니다.

# 목차

제 1 장 서비스 컴포넌트 모니터링 개요 . . . . . 1	서비스 컴포넌트 이벤트 로그 파일 보기 및 해석 . . 45
CEI(Common Event Infrastructure) . . . . . 1	공통 기본 이벤트 브라우저로 이벤트 보기 . . . . 47
공통 기본 이벤트 모델 . . . . . 3	보려는 이벤트 지정. . . . . 47
모니터하는 이유 . . . . . 4	이벤트 브라우저에서 리턴된 이벤트에 대한 작업 48
모니터 대상. . . . . 5	공통 기본 이벤트 브라우저 문제점 해결. . . . 49
모니터링 사용 방법 . . . . . 6	
제 2 장 서비스 컴포넌트 모니터링 사용 및 구성 . . 9	제 4 장 이벤트 카탈로그. . . . . 51
성능 모니터링 . . . . . 9	공통 기본 이벤트 표준 요소. . . . . 51
PMI(Performance Monitoring Infrastructure) 통	이벤트의 비즈니스 오브젝트. . . . . 52
계 . . . . . 10	Business Process Choreographer 이벤트 . . . . 53
SCA(Service Component Architecture)에 대한	비즈니스 프로세스 이벤트 모니터링 . . . . . 53
응용프로그램 응답 조치 통계 . . . . . 15	휴먼 타스크 이벤트 모니터링 . . . . . 73
서비스 컴포넌트 이벤트 모니터링 . . . . . 28	프로세스 서버 이벤트 . . . . . 82
비즈니스 프로세스 및 휴먼 타스크 이벤트의 모니	자원 어댑터 이벤트. . . . . 82
터링 사용. . . . . 29	비즈니스 규칙 이벤트 . . . . . 84
서비스 컴포넌트 이벤트에 대한 로깅 구성 . . . 30	비즈니스 상태 시스템 이벤트 . . . . . 84
CEI(Common Event Infrastructure) 서버로 서비	맵 이벤트. . . . . 86
스 컴포넌트 모니터링 . . . . . 36	중개 이벤트 . . . . . 86
세션 모니터링 . . . . . 41	복구 이벤트 . . . . . 87
제 3 장 모니터링한 이벤트 보기 . . . . . 43	SCA(Service Component Architecture) 이벤트 88
Tivoli Performance Viewer로 성능 매트릭스 보기 43	선택기 이벤트 . . . . . 89
	주의사항 . . . . . 91



---

## 제 1 장 서비스 컴포넌트 모니터링 개요

프로세스 서버에서 서비스 컴포넌트를 모니터링하는 이유에 대한 개념 개요, 모니터링하기 위해 선택하는 서비스 컴포넌트 내의 이벤트 지점 및 시스템에 모니터링을 구성하는 방법을 설명합니다.

WebSphere® Process Server는 성능 조정 및 문제점 해결과 같은 시스템 관리 기능을 보장할 서비스 컴포넌트 모니터링을 위한 기능을 제공합니다. 시스템에 전개되는 응용 프로그램 내에서 서비스 컴포넌트의 처리를 계속해서 모니터링하기 위해 정보 기술 전문가가 아닌 사용자를 위한 기능도 제공해서 전형적인 기능 이상의 범위를 다룹니다. 상호 연결된 컴포넌트의 전반적인 처리 플로우를 감독해서 사용자의 예상대로 시스템이 생성하도록 할 수 있습니다.

WebSphere Process Server는 WebSphere Application Server 설치의 맨 위에서 작동하며 이로 인해 시스템 성능 및 문제점 해결을 모니터링하기 위해 Application Server 하부 구조의 많은 기능을 사용합니다. 프로세스 서버 서비스 컴포넌트를 모니터링하기 위해 특별히 고안된 일부 추가 기능도 포함합니다. WebSphere Process Server Information Center의 이 섹션은 프로세스 서버 특정 서비스 컴포넌트를 모니터링하는 방법을 중점적으로 설명합니다. 이는 WebSphere Application Server, 버전 6.1 Information Center의 모니터링 및 문제점 해결 주제를 보충합니다. 따라서 결합된 제품의 기타 모니터링 기능에 대한 자세한 내용은 이 문서를 참조해야 합니다.

---

### CEI(Common Event Infrastructure)

CEI(Common Event Infrastructure)는 기본 이벤트 관리 서비스를 필요로 하는 응용 프로그램에 이 서비스를 제공하도록 계획된 임베드 가능한 기술입니다.

이 이벤트 하부 구조는 여러 이기종 소스에서 원시 이벤트를 통합하고 지속시키며 이러한 이벤트를 이벤트 고객에 분배하기 위한 통합 위치를 제공합니다. 이벤트는 이벤트의 구조를 정의하는 표준 XML 기반 형식인 공통 기본 이벤트 모델을 사용하여 표시됩니다. 자세한 정보는 공통 기본 이벤트 모델 하위 주제를 참조하십시오.

이 공통 하부 구조를 사용하여 서로 긴밀하게 결합되지 않은 다양한 제품이 이벤트 관리를 통합해서 엔터프라이즈 자원의 단말간 보기를 제공하고 도메인 경계에서 이벤트를 상관시킵니다. 예를 들어, 네트워크 모니터링 응용프로그램이 생성한 이벤트를 보안 응용프로그램이 생성한 이벤트와 상관시킬 수 있습니다. 각 제품이 자체의 방법으로 이벤트를 관리하는 경우에는 이러한 상관이 어려울 수 있습니다.

CEI(Common Event Infrastructure)는 이벤트를 생성, 전파, 지속 및 이용하기 위한 기능을 제공하지만 이벤트 자체를 정의하지는 않습니다. 대신에 응용프로그램 개발자와 관리자가 이벤트 유형, 이벤트 그룹, 필터링 및 상관을 정의합니다.

## CEI(Common Event Infrastructure) 컴포넌트

CEI(Common Event Infrastructure)는 다음 주요 컴포넌트로 구성됩니다.

### 공통 기본 이벤트

공통 기본 이벤트 컴포넌트는 이벤트의 작성 및 특성 데이터에 대한 액세스를 지원합니다. 이벤트 소스는 공통 기본 이벤트 API를 사용하여 공통 기본 이벤트 모델에 따라 새 이벤트를 작성합니다. 이벤트 고객은 API를 사용하여 수신된 이벤트의 특성 데이터를 읽습니다. 응용프로그램은 다른 도구와의 교환을 지원해서 XML 텍스트 형식으로(부터) 이벤트를 변환할 수도 있습니다. 공통 기본 이벤트 컴포넌트는 Eclipse TPTP(Eclipse Test and Performance Tools Platform)의 일부입니다.

**이미터** 이미터 컴포넌트는 이벤트의 전송을 지원합니다. 이벤트 소스가 이벤트를 작성하고 데이터로 채우면 이벤트 소스는 이벤트를 이미터에 제출합니다. 이미터는 선택적으로 자동 콘텐츠 완료를 수행한 다음 이벤트가 공통 기본 이벤트 스펙을 준수하는지 검증합니다. 구성 가능한 필터 기준에 이벤트를 비교하기도 합니다. 이벤트가 유효하며 필터 기준에 맞으면 이미터는 이벤트를 이벤트 서비스로 전송합니다. 이미터는 이벤트를 이벤트 서비스에 동기식으로(Enterprise JavaBeans™ 호출 사용) 또는 비동기로(Java™ Message Service 대기열 사용) 전송할 수 있습니다.

### 이벤트 서비스

이벤트 서비스는 이벤트 소스와 이벤트 고객 사이의 콘딧입니다. 이벤트 서비스는 이미터가 이벤트 소스를 통해 제출한 이벤트를 수신합니다. 지속 데이터 스토어에 이벤트를 저장한 다음 등록된 이벤트 고객에 비동기로 이 이벤트를 분배합니다. 또한 이벤트 서비스는 지속적 저장에서 히스토리 이벤트의 동기 조회를 지원합니다.

### 이벤트 카탈로그

이벤트 카탈로그는 이벤트 메타데이터의 저장소입니다. 응용프로그램은 이벤트 카탈로그를 사용하여 이벤트의 클래스 및 허용되는 콘텐츠에 대한 정보를 검색합니다.

CEI(Common Event Infrastructure)를 사용하는 솔루션 또는 응용프로그램은 다음 컴포넌트(하부 구조 자체의 일부가 아님)도 포함할 수 있습니다.

### 이벤트 소스

이벤트 소스는 이미터를 사용하여 이벤트 서비스에 이벤트를 전송하는 응용프로그램입니다.



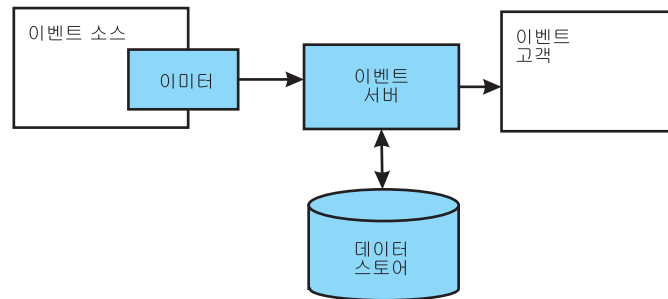
## 이벤트 고객

이벤트 고객은 이벤트 서비스로부터 이벤트를 수신하는 응용프로그램입니다.

## 이벤트 카탈로그 응용프로그램

이벤트 카탈로그 응용프로그램은 이벤트 카탈로그에서 이벤트 메타데이터를 저장하거나 검색하는 응용프로그램입니다. 이는 관리 또는 개발 도구이거나 이벤트 소스 또는 이벤트 고객일 수도 있습니다.

다음 다이어그램은 CEI(Common Event Infrastructure)를 사용하는 이벤트 소스에서 이벤트 고객으로의 일반 이벤트 플로우를 보여줍니다.



## 공통 기본 이벤트 모델

공통 기본 이벤트 모델은 엔터프라이즈 관리 및 비즈니스 응용프로그램에 사용하도록 계획된 이벤트의 공통 표시를 정의하는 표준입니다. IBM® Autonomic Computing Architecture Board에서 개발된 이 표준은 로깅 인코딩, 추적, 관리 및 공통 XML 기반 형식을 사용하는 비즈니스 이벤트를 지원해서 여러 다른 응용프로그램의 상이한 유형의 이벤트를 상관시킬 수 있게 합니다. 공통 기본 이벤트 모델은 IBM Autonomic Computing 툴킷의 일부입니다. 자세한 정보는 <http://www.ibm.com/autonomic>를 참조하십시오.

CEI(Common Event Infrastructure)는 현재 버전 1.0.1 스펙을 지원합니다.

공통 기본 이벤트 모델의 기본 개념은 상황입니다. 상황은 컴퓨팅 하부 구조의 어디에서나 발생하는 것으로 예를 들어, 서버 종료, 디스크 드라이브 장애 또는 실패한 사용자 로그인입니다. 공통 기본 이벤트 모델은 발생할 수 있는 대부분의 상황을 포함하는 표준 상황 유형 세트를 정의합니다(예: StartSituation 및 CreateSituation).

이벤트는 상황에 관련된 정보를 보고하는 구조화된 공고입니다. 이벤트는 세 가지 유형의 정보를 보고합니다.

- 상황 자체(발생한 사항)
- 영향이 미친 컴포넌트의 ID(예: 시스템 종료된 서버)
- 상황을 보고 중인 컴포넌트의 ID(영향을 받은 컴포넌트와 동일할 수도 있음)

공통 기본 이벤트 스펙은 세 가지 유형의 정보를 모두 제공하는 특성이 포함된 XML 요소로 이벤트를 정의합니다. 이 특성은 루트 요소, `CommonBaseEvent`의 하위 요소 및 속성으로 인코딩됩니다.

공통 기본 이벤트 형식은 확장할 수 있습니다. 표준 이벤트 특성 외에 이벤트는 상황에 관련된 임의의 요소의 정보를 포함할 수 있는 응용프로그램 특정 요소인 확장 데이터 요소를 포함할 수도 있습니다. `extensionName` 속성은 선택적 분류 이름(이벤트 클래스)으로 이벤트에 레이블을 붙이며 이는 예상하는 확장 데이터 요소의 종류를 응용프로그램에 표시합니다. 이벤트 카탈로그는 이 이벤트 클래스 및 허용되는 콘텐츠를 설명하는 이벤트 정의를 저장합니다.

공통 기본 이벤트 형식에 대한 완전한 세부사항은 IBM Autonomic Computing Toolkit에 포함된 XSD 스키마 및 스펙 문서를 참조하십시오.

---

## 모니터하는 이유

WebSphere Process Server 내의 서비스 컴포넌트를 모니터링하여 성능을 평가하고, 문제점을 해결하며, 시스템에 전개된 응용프로그램을 구성하는 서비스 컴포넌트의 전반적인 처리 진행 상태를 평가합니다.

서비스 컴포넌트는 WebSphere Process Server로 통합되는 필수 불가결한 기능으로, 이를 통해 엔터프라이즈에 사용되는 프로세스를 미러링하는 시스템의 응용프로그램을 쉽게 작성하고 전개할 수 있습니다. 따라서 프로세스 서버에서 수행하려는 작업을 관리하기 위해서는 서비스 컴포넌트를 효과적으로 모니터링할 필요가 있습니다. 프로세스 서버의 서비스 컴포넌트를 모니터링할 필요가 있는 세 가지 주요 이유가 있습니다.

### 문제점 해결

WebSphere Process Server 아래에 있는, WebSphere Application Server에서 제공하는 로깅 및 추적 기능을 사용하여 특정 오류를 진단할 수 있습니다. 예를 들어, 특정 응용프로그램이 예상되는 결과를 산출하지 않는 경우 이 응용프로그램을 이루는 서비스 컴포넌트의 처리를 모니터링하도록 로거를 설정할 수 있습니다. 파일에 공개된 로그 출력이 있을 수 있으며 그러면 이를 검토해서 문제점의 원인을 정확히 지적할 수 있습니다. 문제점 해결은 시스템 관리자 및 시스템 하드웨어와 소프트웨어의 유지보수에 관련된 기타 사용자에게 중요한 task입니다.

### 성능 조정

대부분의 프로세스 서버 특정 서비스 컴포넌트가 생성하는 특정 성능 통계를 모니터링할 수 있습니다. 이 정보를 사용하여 시스템 성능 상태를 유지보수 및 조정하고 응용프로그램이 최적 상태에서 효율적으로 조정되게 하십시오. 하나 이상의 서비스가 빈약한 레벨에서 수행되고 있는 상황(기타 프로그램이 시스템에 있음을 나타내는 것일 수 있음)을 분별할 수도 있습니다. 문제점 판별과 마찬가지로 성능 조정은 정보 기술 전문가가 일반적으로 수행하는 task입니다.

## 서비스 컴포넌트의 처리 평가

문제점 판별 및 성능 조정은 특정 문제를 해결하기 위해 단기간에 수행하는 타스스크입니다. 시스템에 전개된 응용프로그램으로 통합된 서비스 컴포넌트를 계속해서 모니터하도록 프로세스 서버를 설정할 수도 있습니다. 이 유형의 서비스 컴포넌트 모니터링은 프로세스가 설계 목표에 도달하도록 하며 설계와 구현을 담당하는 사용자에게 중요하며 반드시 정보 기술 전문가가 아닌 사용자도 이를 수행할 수 있습니다.

---

## 모니터 대상

서비스 컴포넌트 이벤트가 처리 중 도달하는 특정 지점을 선택해서 WebSphere Process Server의 서비스 컴포넌트 이벤트를 모니터할 수 있습니다. 각 서비스 컴포넌트는 응용 프로그램이 제공된 이 지점을 처리할 때 이벤트를 "가동"하는 이벤트 지점을 정의합니다. 서비스 컴포넌트 이벤트에 대한 성능 통계를 모니터할 수도 있습니다.

서비스 컴포넌트에 대해 수행하려는 모니터링 유형(문제점 판별, 성능 조정 또는 프로세스 모니터링)과 무관하게 실제로는 이 컴포넌트 처리 중 도달하는 특정 지점을 모니터하게 됩니다. 이 지점을 이벤트 지점이라 하며 모니터 시 이 지점을 선택합니다. 각 이벤트 지점은 서비스 컴포넌트 종류 태그, 선택적 요소 종류(서비스 컴포넌트 유형의 특정 함수) 및 이벤트의 네이처를 캡슐화합니다. 이 모든 요인이 모니터링에 가동된 이벤트의 유형을 결정합니다.

이벤트 네이처는 서비스 컴포넌트 처리 중 이벤트를 생성(또는 "가동")하는 데 필요한 상황을 설명합니다. 이 네이처는 본래 모니터하도록 선택하는 서비스 컴포넌트의 논리 구조에서 핵심 지점입니다. 서비스 컴포넌트 이벤트의 가장 일반적인 네이처는 ENTRY, EXIT 및 FAILURE이지만 특정 컴포넌트 및 요소에 따라 여러 다른 네이처가 있습니다. 지정된 서비스 컴포넌트를 포함하는 응용프로그램이 연속으로 호출될 때마다 서비스 컴포넌트의 처리 시 이벤트 네이처에 해당하는 지점을 지나가면 언제나 이벤트가 가동됩니다.

서비스 컴포넌트 종류에 대해 이벤트가 정의되는 방식의 예를 들면 MAP 서비스 컴포넌트 종류는 ENTRY, EXIT 및 FAILURE 네이처가 있는 이벤트를 직접 가동합니다. MAP 컴포넌트 종류 내에서 특정 유형의 기능을 정의하는 Transformation이라는 요소 종류도 포함합니다. 이 요소도 ENTRY, EXIT 및 FAILURE 네이처가 있는 이벤트를 가동합니다. 결과적으로 MAP 서비스 컴포넌트 종류는 지정한 네이처와 요소의 조합에 따라 최대 6개의 다른 이벤트를 가동할 수 있습니다. 모든 서비스 컴포넌트, 요소 및 이벤트 네이처의 목록이 이벤트 카탈로그에 포함되어 있습니다.

모니터링은 응용프로그램 처리의 맨 위에 위치하며 서비스 컴포넌트의 처리와 인터페이스하지 않는 별도의 기능 레이어입니다. 모니터링은 지정된 이벤트 지점에서 활동을 감지하는 경우에만 서비스 컴포넌트 처리에 관여합니다. 이러한 경우에는 아래에 설명된

대로 수행 중인 모니터링의 유형에 기초하여 이벤트가 전송된 위치 및 이 이벤트에 포함된 데이터를 판별하는 이벤트가 모니터링을 위해 가동됩니다.

### 성능 매트릭스

성능 매트릭스를 집계하기 위해 서비스 컴포넌트를 모니터 중인 경우 PMI(Performance Monitoring Infrastructure)로 경량의 이벤트가 가동됩니다. 프로세스 서버 특정 서버 컴포넌트에 대해 생성된 세 가지 성능 통계 중 하나 이상의 통계를 모니터하도록 선택할 수 있습니다.

- 각 EXIT 이벤트 네이처에 대한 카운터 - 성공한 수치를 계산합니다.
- 각 FAILURE 이벤트 네이처에 대한 카운터 - 실패한 수치를 계산합니다.
- 해당 ENTRY 및 EXIT 이벤트 간에 계산된 처리 지속 기간(동기 수치만).

ARM(Application Response Measurement) 통계를 사용하여 SCA(Service Component Architecture) 레벨에서 응용프로그램의 성능을 모니터할 수도 있습니다. 이를 통해 기타 서비스 컴포넌트 이벤트에서 다르게 사용할 수 있는 것보다 응용프로그램 내의 훨씬 더 정교한 레벨에서 응용프로그램의 모니터가 가능합니다. 이 통계를 사용하여 초기 응용프로그램 호출 및 서비스 응답(SCA를 사용할 때) 사이의 여러 다른 지점을 모니터할 수 있습니다.

### 비즈니스 오브젝트가 있는 서비스 컴포넌트 이벤트

서비스 컴포넌트의 지정된 이벤트 지점에서 모니터링으로 가동된 데이터를 캡처하려면 이벤트 및 공통 기본 이벤트 형식으로 인코딩할 데이터를 생성하도록 프로세스 서버를 구성합니다. 각 서비스 컴포넌트 이벤트에서 캡처할 비즈니스 오브젝트 데이터의 세부사항 레벨을 지정할 수 있습니다. 특별히 구성된 CEI 서버 데이터베이스에 출력을 지정하는 CEI(Common Event Infrastructure) 버스 또는 로거로 이벤트를 공개할 수 있습니다.

---

## 모니터링 사용 방법

프로세스 서버에서 수행하려 계획 중인 모니터링의 유형에 따라 모니터할 서비스 컴포넌트 이벤트 지점을 지정하는 데 사용할 수 있는 여러 메소드가 있습니다.

### 성능 통계

PMI(Performance Monitoring Infrastructure) 통계의 경우 관리 콘솔을 사용하여 모니터하려는 특정 이벤트 지점 및 연관된 성능 조치를 지정하십시오. 모니터링 서비스 컴포넌트 성능을 시작하면 생성된 통계는 일정한 간격으로 Tivoli® Performance Viewer에 공개됩니다. 이 표시기를 사용하여 시스템에 발생하는 결과를 보고 선택적으로 동일한 표시기 내에서 나중에 보고 분석할 수 있는 파일에 결과를 로깅할 수 있습니다.

ARM(Application Response Measurement) 통계의 경우는 관리 콘솔 요청 매트릭스 섹션을 사용하여 모니터링하려는 통계를 지정하십시오. 요청 매트릭스에 대한 자세한 내용과 지시사항은 WebSphere Application Server, 버전 6.1 Information Center를 참조하십시오.

#### 문제점 판별 및 비즈니스 프로세스 모니터링을 위한 공통 기본 이벤트

응용프로그램을 작성할 때 실행 중인 서버에 응용프로그램이 전개된 후 이벤트의 특정 세부사항 레벨과 함께 서비스 컴포넌트 이벤트 지점을 모니터링하도록 지정할 수 있습니다. 프로세스 서버에서 응용프로그램이 전개되고 이벤트를 최소 한번은 호출한 후 모니터링 이벤트 지점을 선택할 수도 있습니다. 두 경우 모두 모니터링으로 생성된 이벤트는 CEI(Common Event Infrastructure) 버스에서 가동됩니다. 이 이벤트를 로그 파일에 공개하거나 구성된 CEI 서버 데이터베이스에 공개할 수 있습니다. WebSphere Process Server는 문제점 판별 및 비즈니스 프로세스 모니터링에 대한 두 가지 유형의 공통 기본 이벤트 사용을 지원합니다.

**정적** 응용프로그램 내의 특정 이벤트 지점 및 세부사항 레벨에 WebSphere Integration Developer 도구를 사용하여 모니터링을 위한 태그를 붙일 수 있습니다. 선택사항은 계속해서 모니터링하고 프로세스 서버 응용프로그램과 함께 분배 및 전개되는 확장자가 .mon인 파일에 저장될 이벤트 지점을 나타냅니다. 프로세스 서버가 CEI 서버를 사용하도록 구성되면 모니터링 함수는 지정된 서비스가 호출될 때마다 CEI 서버에 서비스 컴포넌트 이벤트를 가동하기 시작합니다. 응용프로그램이 프로세스 서버에 전개되는 한, .mon 파일에 지정된 서비스 컴포넌트 이벤트 지점은 응용프로그램이 중지될 때까지 지속적으로 모니터링됩니다. 실행 중인 응용프로그램에서 추가 이벤트를 모니터링하도록 지정하고 이미 모니터링한 이벤트 지점의 세부사항 레벨을 증가시킬 수 있습니다. 그러나 이 응용프로그램이 활성화된 동안에는 전개된 응용프로그램의 .mon에 지정된 모니터링 이벤트 지점을 중지하거나 해당 세부사항 레벨을 낮출 수 없습니다.

**동적** 응용프로그램의 처리 중 서버를 종료하지 않고 추가 이벤트 지점을 모니터링할 필요가 있으면 동적 모니터링을 사용할 수 있습니다. 관리 콘솔을 사용하여 모니터링할 서비스 컴포넌트 이벤트 지점을 지정하고 공통 기본 이벤트에 포함될 페이로드에 대한 세부사항 레벨을 설정하십시오. 프로세스 서버가 시작된 후 처리된 서비스 컴포넌트가 도달한 이벤트 지점으로 목록이 컴파일됩니다. 이 목록에서 모니터링할 개별 이벤트 지점 또는 이벤트 지점 그룹을 선택하고 서비스 컴포넌트 이벤트를 로거 또는 CEI 서버 데이터베이스로 지정하십시오.

동적 인에이블먼트의 기본 용도는 로그에 공개되는 상관된 서비스 컴포넌트 이벤트를 작성해서 서비스에 대한 문제점 판별을 수행하는 것

입니다. 서비스 컴포넌트 이벤트는 요청되고 있는 데이터의 용량에 따라 커질 수 있으며 CEI 서버에 이벤트를 전송하도록 선택하는 경우 많은 데이터베이스 자원을 사용할 수 있습니다. 결과적으로 이벤트의 비즈니스 데이터를 읽어야 하는 경우 또는 그렇지 않으면 이벤트의 데이터베이스 레코드를 보존해야 하는 경우에만 모니터링 이벤트를 동적으로 CEI 서버에 공개해야 합니다. 그러나 특정 세션을 모니터링하고 있으면 CEI 서버 데이터베이스를 사용하여 이 세션에 관련된 서비스 컴포넌트 이벤트에 액세스합니다.



---

## 제 2 장 서비스 컴포넌트 모니터링 사용 및 구성

프로세스 서버에서 서비스 컴포넌트를 모니터링하려면 먼저 모니터링 기능을 사용 가능하게 해야 합니다. 그런 다음 모니터링하려는 이벤트, 이벤트에서 캡처하려는 정보 및 결과를 공개하는 데 사용되는 메소드를 지정해야 합니다.

---

### 성능 모니터링

성능 조치는 서비스 컴포넌트 이벤트 지점에 사용 가능하며 PMI(Performance Monitoring Infrastructure)를 통해 처리됩니다. 서비스 컴포넌트 이벤트 지점에서 성능 메트릭을 수집하도록 프로세스 서버를 구성합니다. 응용프로그램의 서비스 호출에서 직접 SCA(Service Component Architecture) 특정 성능 통계를 수집할 수도 있습니다.

최적의 효율성 또는 저조한 성능 진단을 위해 WebSphere Process Server 서비스 컴포넌트를 사용 중이든 아니든, 성능 Perspective에서 얼마나 다양한 런타임 및 응용프로그램 자원이 작동하고 있는지 이해하는 것이 중요합니다. PMI(Performance Monitoring Infrastructure)는 런타임 및 응용프로그램 자원 작동을 설명하는 포괄적인 데이터 세트를 제공합니다. PMI 데이터를 사용하여 Application Server의 성능 병목 현상을 식별하고 수정할 수 있습니다. Application Server의 성능 상태를 모니터링하는 데에도 PMI 데이터를 사용할 수 있습니다.

PMI는 기본 WebSphere Application Server 설치에 포함됩니다. 이 섹션은 WebSphere Process Server에 특정한 서비스 컴포넌트에 관련되어 있어서 성능 모니터링에 대한 보충 정보만을 제공합니다. 따라서 전체 제품의 기타 파트에 관한 PMI 사용에 대해서는 WebSphere Application Server 문서의 정보를 참조하십시오.

PMI에서 모니터링할 수 있는 WebSphere Process Server에 특정한 서비스 컴포넌트 이벤트 지점은 일반적으로 이벤트 네이처가 ENTRY, EXIT 및 FAILURE입니다. 이 패턴에 따라 정의되지 않은 이벤트 소스는 지원되지 않습니다. 지원되는 이벤트에는 측정할 수 있는 세 가지 유형의 성능 통계가 있습니다.

- 성공적 호출
- 실패한 호출
- 이벤트 완료 경과 시간

ARM(Application Response Measurement) 통계를 사용하여 응용프로그램의 서비스 호출에서 파생된 성능 통계를 모니터링할 수도 있습니다. 이 통계는 엔터프라이즈 응용프로그램을 구성하는 프로세스 서버 서비스 컴포넌트 이벤트 아래에 있는 실제 런타임 프로세스를 측정합니다. 이 통계를 사용하여 응용프로그램의 처리를 위한 다양한 성능 조치를 이끌어낼 수 있습니다.

## PMI(Performance Monitoring Infrastructure) 통계

PMI(Performance Monitoring Infrastructure)를 사용하여 세 가지 유형의 성능 통계 즉, 성공한 호출 수, 장애 수 및 이벤트 완료 경과 시간을 모니터링할 수 있습니다. 이 통계는 ENTRY, EXIT 및 FAILURE 이벤트 네이처가 있는 이벤트에만 사용할 수 있습니다.

### 관리 콘솔을 통한 PMI 사용

성능 데이터를 모니터링하려면 먼저 서버에서 PMI(Performance Monitoring Infrastructure)를 사용 가능하게 해야 합니다.

#### 타스크 정보

관리 콘솔을 통해 PMI를 사용 가능하게 할 수 있습니다.

#### 프로시저

1. 관리 콘솔을 여십시오.
2. 콘솔 탐색 트리에서 서버 > **Application Server**를 클릭하십시오.
3. *server\_name*을 클릭하십시오.

주: 관리 콘솔에서 모니터링 및 조정 > PMI > *server\_name*을 클릭하여 동일한 패널을 여십시오.

4. 구성 탭을 클릭하십시오.
5. **PMI(Performance Monitoring Infrastructure) 사용** 선택란을 선택하십시오.
6. 옵션: 순차 카운터 갱신 사용 선택란을 선택해서 세밀한 통계 갱신을 사용 가능하게 하십시오.
7. 서버 이름 링크를 클릭해서 서버 PMI 구성 페이지로 돌아가십시오.
8. 적용 또는 확인을 클릭하십시오.
9. 저장을 클릭하십시오.
10. 서버를 재시작하십시오.

#### 다음에 수행할 작업

변경사항은 서버를 다시 시작해야 적용됩니다.

### 이벤트 성능 통계

성능 모니터링 통계는 대부분의 서버 이벤트에 대해 사용 가능합니다. 호출 요청(양호하고 잘못된)의 계수 및 이벤트 완료에 소요된 시간을 모니터링할 수 있습니다.

PMI(Performance Monitoring Infrastructure)를 사용하여 다음 표에 표시된 대로 특정 서버 이벤트에서 생성된 세 가지 성능 통계를 모니터링할 수 있습니다.



표 1. 이벤트의 PMI 통계

통계 이름	유형	설명
<b>BadRequests</b>	카운터	실패한 이벤트 호출 수
<b>GoodRequests</b>	카운터	성공한 이벤트 호출 수
<b>ResponseTime</b>	타이머	이벤트 완료 경과 시간

이 통계는 ENTRY, EXIT 및 FAILURE 네이처가 있는 요소의 서비스 컴포넌트 이벤트로 제한됩니다. 응용프로그램에서 제공된 서버 이벤트 유형의 단일 이벤트에 대해 각 통계가 작성됩니다. 모든 성능 조치는 카운터(제공된 이벤트 지점의 누적 가동 수) 또는 타이머(두 이벤트 지점의 가동 간 밀리초 단위 지속 기간)입니다. 모니터링할 수 있는 각 이벤트 종류(및 관련 요소)가 아래에 나옵니다.

표 2. 이벤트 성능 통계를 생성할 수 있는 이벤트 유형 및 요소

이벤트 유형	요소
비즈니스 프로세스	Process Invoke Staff Receive Wait Compensate Pick Scope
휴먼 태스크	Task
비즈니스 규칙	Operation
비즈니스 상태 시스템	Transition Guard Action EntryAction ExitAction
선택기	Operation
맵	Map Transformation
중개	OperationBinding ParameterMediation
자원 어댑터	InboundEventRetrieval InboundEventDelivery Outbound

## 모니터할 성능 통계 지정

관리 콘솔을 사용하여 PMI(Performance Monitoring Infrastructure)를 통해 단일 통계, 복수 통계 또는 관련된 통계 그룹을 모니터링하도록 지정할 수 있습니다.

시작하기 전에

이 작업을 수행하기 전에 성능 모니터링을 사용 가능하게 했는지 및 모니터링하려는 이벤트를 적어도 한번은 호출했는지 확인하십시오.

### 프로시저

1. 관리 콘솔을 여십시오.
2. 모니터링 및 조정 > **PMI(Performance Monitoring Infrastructure)**를 선택하십시오.
3. 모니터링하려는 이벤트 지점을 포함하는 Node Agent 또는 서버를 선택하십시오.

주: 클러스터의 통계를 모니터링하도록 선택할 수는 없습니다. 특정 서버나 노드의 통계만을 모니터링할 수 있습니다.

4. **WBISStats.RootGroup** 또는 **Enterprise Bean**과 같은 일부 그룹을 펼치십시오. 모니터링할 수 있는 모든 통계는 나열된 그룹 내에 있습니다. 일부 통계는 서버가 마지막으로 시작된 이후에 호출되지 않았기 때문에 나열되지 않을 수도 있습니다.
5. 패널의 왼쪽에 있는 트리 내에서 모니터링하려는 통계를 선택한 다음 오른쪽에서 수집하려는 통계를 선택하고 **사용 가능**을 클릭하십시오. 모니터링하려는 모든 통계에 대해 이를 반복하십시오.
6. 서버 이름 링크를 클릭해서 서버 PMI 구성 페이지로 돌아가십시오.
7. **적용** 또는 **확인**을 클릭하십시오.
8. **저장**을 클릭하십시오.

### 결과

이제 Tivoli Performance Viewer에서 선택한 통계의 성능 모니터링을 시작할 수 있습니다.

주: 이 통계를 볼 때 카운터 유형 통계와 지속 기간 유형 통계를 혼합해선 안됩니다. 카운터는 누적되며 그래프로 표시되는 비율이 응용프로그램에 따라 빠르게 올라갈 수 있습니다. 이와 반대로 지속 기간 통계는 시스템이 각 이벤트를 처리하는 데 걸리는 평균 시간을 나타내기 때문에 일정 범위 내에 머무릅니다. 결과적으로 통계와 상대 배율 간의 불일치로 인해 하나 또는 기타 유형의 통계가 표시기 그래프에서 비뚤어지게 표시될 수 있습니다.

### 학습서: 서비스 컴포넌트 성능 모니터링

이 학습서는 PMI(Performance Monitoring Infrastructure)에 공개되는 서비스 컴포넌트 이벤트 모니터를 설정하는 방법 및 TPV(Tivoli Performance Viewer)에서 결과적인 성능 통계를 보는 방법에 대한 예제를 제공합니다. CEI(Common Event Infrastructure) 서버 및 로거를 사용한 모니터링 및 서비스 컴포넌트 이벤트 지점의 성능 모니터링이 얼마나 다른지 보여줍니다. 유의해야 할 주요 차이점은 성능 모니터링의 경우 특정 네이처가 있는 개별 이벤트 대신 전체 서비스 컴포넌트 요소를 선택한다는

접입니다. WebSphere Process Server는 ENTRY, EXIT 및 FAILURE 네이처가 있는 서비스 컴포넌트 요소에 대해서만 성능을 모니터링할 수 있기 때문에 이러한 유형의 서비스 컴포넌트 요소만을 모니터링하도록 선택할 수 있습니다.

서비스 컴포넌트 이벤트 지점 ENTRY, EXIT 및 FAILURE는 모든 모니터링 유형에 동일한 반면 프로세스 서버 성능 모니터링 함수는 CEI 이벤트에 포함된 정보가 모두 포함되지는 않은 "최소화" 이벤트를 가동합니다. 이 이벤트는 해당 이벤트 세트의 성능 통계를 계산하는 PMI에 전송됩니다.

- 성공적인 호출 — 해당 ENTRY 이벤트 후 EXIT 네이처 유형의 이벤트 가동
- 실패한 호출 — 해당 ENTRY 이벤트 후 FAILURE 네이처가 있는 이벤트 가동
- 성공적인 완료 시간 — ENTRY 이벤트 가동과 해당 EXIT 이벤트 지점 가동 사이의 경과 시간

PMI는 성공 및 실패한 호출 수의 누적 카운터와 완료 응답 시간의 실행 평균을 표시하는 TPV에 통계를 공개합니다.

## 이 학습서의 목표

이 학습서를 완료한 후에는 다음을 수행할 수 있습니다.

- 모니터링하려는 서비스 컴포넌트 요소의 성능 통계를 선택합니다.
- 결과적인 성능 통계를 보고 해석합니다.

## 이 학습을 완료하는 데 필요한 시간

이 학습서를 완료하려면 약 15 - 20분이 소요됩니다.

## 전제조건

이 학습서를 수행하려면 다음을 완료했어야 합니다.

- 프로세스 서버를 구성하고 시작했습니다.
- 프로세스 서버에서 PMI를 사용 가능하게 했습니다.
- 프로세스 서버에 샘플 갤러리 응용프로그램을 설치하고 시작했습니다.
- 프로세스 서버에 비즈니스 규칙 샘플 응용프로그램을 설치하고 시작했습니다. 샘플 갤러리 페이지의 지시사항에 따라 비즈니스 규칙 샘플 응용프로그램을 설정하고 실행하십시오.

이 모든 전제조건이 완료된 후 학습서를 진행하기 전에 샘플 갤러리에서 비즈니스 규칙 샘플 응용프로그램을 적어도 한번은 실행하십시오.

## 예제: 서비스 컴포넌트 성능 모니터링: 태스크 정보

이 시나리오의 비즈니스 규칙 샘플 응용프로그램을 사용하여 세 가지 성능 통계 즉, 성공, 실패 및 응답 시간을 모두 모니터링합니다. 웹 페이지에 이 응용프로그램이 이미 열

려 있어야 합니다. 모니터링을 시작한 후 샘플을 여러 번 실행할 예정이므로 응용프로그램을 계속 열어 놓으십시오. 샘플을 최소 한번은 이미 실행했는지 확인하십시오. 실행한 경우 모니터하려고 선택할 수 있는 함수 목록에 샘플이 나타납니다.

### 프로시저

1. 관리 콘솔을 여십시오.
2. 클러스터를 모니터하려면 콘솔 탐색 트리에서 서버 > 클러스터 > *cluster\_name* 을 클릭하십시오. 단일 프로세스 서버를 모니터하려면 콘솔 탐색 트리에서 서버 > **Application Server** > *server\_name* 을 클릭하십시오.
3. 런타임 탭을 클릭하십시오.
4. 성능 아래에서 **PMI(Performance Monitoring Infrastructure)**를 클릭하십시오.
5. 사용자 정의를 선택하십시오.
6. **WBISStats.RootGroup > BR > brsample\_module.DiscountRuleGroup > 조**작을 펼치고 **\_calculateDiscount**를 선택하십시오.
7. **BadRequests, GoodRequests** 및 **ResponseTime** 옆의 선택란을 선택하고 사용가능을 누르십시오.
8. 관리 콘솔 탐색 트리에서 모니터링 및 조정 > 성능 표시기 > 현재 활동을 클릭하십시오.
9. *server\_name* 옆의 선택란을 선택하고 모니터링 시작을 누르십시오.
10. *server\_name*을 클릭하십시오.
11. **WBISStats.RootGroup > BR > brsample\_module.DiscountRuleGroup > 조**작을 펼치고 **\_calculateDiscount** 옆의 선택란을 선택하십시오.

### 결과

이제 빈 그래프와 이 그래프 아래에 세 가지 통계의 이름 및 값이 표시되어야 합니다. 통계 이름 옆의 선택란이 아직 선택되지 않은 경우 선택하십시오. 이제 PMI는 선택된 이벤트의 성능 데이터를 공개할 준비가 되었으며 TPV는 결과를 표시할 준비가 되었습니다.

비즈니스 규칙 샘플 응용프로그램을 여러 번 실행한 다음 주기적으로 새로 고쳐지는 성능 표시기를 지켜 보십시오. 이제 그래프에 성공한 요청의 누적 수와 성공한 각 요청에 대한 평균 응답 시간을 나타내는 행이 있어야 함에 유의하십시오. 그래프 아래에 각 통계의 이름 옆에서 값도 볼 수 있습니다. 성공 수를 나타내는 행은 추가로 샘플을 호출함에 따라 계속해서 올라가야 하는 반면 응답 시간 행은 몇 번의 새로 고치기 후 낮아져야 합니다.

이 작업을 완료하면 WebSphere Process Server가 서비스 컴포넌트의 성능 모니터링을 구현하는 방법을 이해해야 합니다. 이제 모니터할 서비스 컴포넌트를 선택하는 방

법 및 성능 통계를 계산하는 방법을 학습했습니다. 성능 모니터를 시작하고 사용 중인 응용프로그램에 대한 성능 조치를 볼 수도 있습니다.

성능 모니터링은 많은 시스템 자원을 소모할 수 있습니다. 따라서 이 작업을 완료한 후에는 모니터를 중지해야 합니다. 중지하려면 간단히 Tivoli Performance Viewer 링크를 클릭하고 노드와 서버를 모두 선택한 다음 모니터링 중지를 누르면 됩니다.

## SCA(Service Component Architecture)에 대한 응용프로그램 응답 조치 통계

SCA(Service Component Architecture) 레벨에서 모니터링할 수 있는 성능 통계는 25가지가 있습니다. 카운터 또는 타이머인 응용프로그램 응답 조치 통계를 사용하여 서비스 호출 및 응답을 다양한 패턴으로 측정할 수 있습니다.

다음 표에 표시된 ARM(Application Response Measurement) 통계는 SCA(Service Component Architecture) 레이어에 대한 호출자 호출의 시간 및 계수 조치와 서비스에서 리턴된 결과입니다(매우 간단한 방식의). 실제로 지연된 응답의 동기 및 비동기 구현 사이의 다양한 여러 서비스 호출 패턴, 결과 검색, 콜백 및 단방향 호출이 있습니다. 그러나 이 모든 것은 호출자 호출과 서비스, 서비스로부터의 응답 또는 일부 경우 SCA 레이어가 사이에 있는 데이터소스 사이에 있습니다.

관리 콘솔에서 모니터링 및 조정 > 요청 매트릭스 패널을 열어서 모니터링하려는 ARM 통계를 지정할 수 있습니다. 요청 매트릭스 정보는 나중에 저장 및 분석할 수 있도록 로그 파일에 저장되거나 ARM 에이전트에 송신되거나 또는 두 가지 모두입니다. WebSphere Process Server는 ARM 에이전트를 제공하지 않지만 ARM 4.0에 맞는 에이전트의 사용은 지원됩니다. 자신만의 ARM 구현 프로바이더를 선택해서 ARM 구현 라이브러리를 얻을 수 있습니다. ARM 프로바이더의 지시사항에 따르면 ARM 프로바이더에 있는 ARM API Java 아카이브(JAR) 파일이 클래스 경로에 있는지 확인해서 WebSphere Process Server가 필요한 클래스를 로드할 수 있게 하십시오. 그런 다음 서버를 다시 시작하기 전에 관리 콘솔에서 **Application Server > server\_name > 프로세스 정의 > JVM(Java Virtual Machine) > 사용자 정의 특성**을 선택해서 각 서버에 대한 시스템 특성에 다음 항목을 추가해야 합니다.

- Arm40.ArmMetricFactory — ARM 구현 프로바이더의 매트릭스 팩토리의 완전한 Java 클래스 이름
- Arm40.ArmTranReportFactory — ARM 구현 프로바이더의 트랜잭션 보고서 팩토리의 완전한 Java 클래스 이름
- Arm40.ArmTransactionFactory — ARM 구현 프로바이더의 트랜잭션 팩토리의 완전한 Java 클래스 이름

ARM 통계를 수집하도록 서버를 구성하는 방법에 대한 자세한 내용은 WebSphere Application Server 문서를 참조하십시오.

표 3. ARM 통계를 생성할 수 있는 이벤트 유형 및 요소

이벤트 유형	요소
비즈니스 프로세스	Process
휴먼 태스크	Task
비즈니스 규칙	Operation
비즈니스 상태 시스템	Transition Guard Action EntryAction ExitAction
선택기	Operation
맵	Map Transformation
중개	OperationBinding ParameterMediation
자원 어댑터	InboundEventRetrieval InboundEventDelivery Outbound

표 4. 공통 사항. 다음 통계는 모든 서비스 호출 패턴에 공통됩니다.

통계 이름	종류	설명
<b>GoodRequests</b>	카운터	예외가 발생하지 않은 서버 호출 수
<b>BadRequests</b>	카운터	예외가 발생한 서버 호출 수
<b>ResponseTime</b>	타이머	요청 수신과 결과 계산 사이의 서버측에서 측정된 지속 기간
<b>TotalResponseTime</b>	타이머	호출자가 서비스를 요청한 시간부터 호출자에 대한 결과가 사용 가능한 시간까지 호출자측에서 측정된 지속 기간. 호출자의 결과 처리는 포함하지 않습니다.
<b>RequestDeliveryTime</b>	타이머	호출자가 서비스를 요청한 시간부터 서버측 구현으로 요청이 인계되는 시간까지 호출자측에서 측정된 지속 기간. 분산 환경에서 이 조치의 품질은 시스템 시계의 동기 품질에 따라 다릅니다.
<b>ResponseDeliveryTime</b>	타이머	결과가 클라이언트에 사용 가능하도록 하는 데 필요한 시간. 지연된 응답의 경우 결과 검색 시간은 이 시간에 포함되지 않습니다. 분산 환경에서 이 조치의 품질은 시스템 시계의 동기 품질에 따라 다릅니다.

표 5. 참조. 이러한 통계는 호출자가 서비스로부터의 응답 없이 SCA 레이어 또는 데이터소스에 호출할 때 발생합니다.

통계 이름	유형	설명
<b>GoodRefRequests</b>	카운터	예외가 발생하지 않은 SCA 레이어에 대한 호출자 호출 수
<b>BadRefRequests</b>	카운터	예외가 발생한 SCA 레이어에 대한 호출자 호출 수
<b>RefResponseTime</b>	타이머	호출자가 SCA 레이어를 요청한 시간부터 이 호출의 결과가 호출자에 리턴된 시간까지 호출자측에서 측정된 지속 기간

표 5. 참조 (계속). 이러한 통계는 호출자가 서비스로부터의 응답 없이 SCA 레이어 또는 데이터소스에 호출할 때 발생합니다.

통계 이름	유형	설명
<b>BadRetrieveResult</b>	카운터	예외가 발생한 데이터소스에 대한 호출자 호출 수
<b>GoodRetrieveResult</b>	카운터	예외가 발생하지 않은 데이터소스에 대한 호출자 호출 수
<b>RetrieveResultResponseTime</b>	타이머	호출자가 SCA 레이어를 요청한 시간부터 데이터소스 응답이 호출자에 리턴된 시간까지 호출자측에서 측정된 지속 기간
<b>RetrieveResultWaitTime</b>	타이머	제한시간 초과가 발생하는 경우 호출자측에서 측정된 지속 기간

표 6. 대상. 이 통계는 서비스와 SCA 또는 데이터소스 간에 나온 요청이 있을 때 발생합니다.

통계 이름	종류	설명
<b>GoodTargetSubmit</b>	카운터	예외가 발생하지 않은 서비스에 대한 SCA 호출 수
<b>BadTargetSubmit</b>	카운터	예외가 발생한 서비스에 대한 SCA 호출 수
<b>TargetSubmitTime</b>	타이머	SCA가 서비스를 요청한 시간부터 이 호출의 결과가 SCA에 리턴된 시간까지 서버측에서 측정된 지속 기간
<b>GoodResultSubmit</b>	카운터	예외가 발생하지 않은 데이터소스에 대한 서비스 호출 수
<b>BadResultSubmit</b>	카운터	예외가 발생한 데이터소스에 대한 서비스 호출 수
<b>ResultSubmitTime</b>	타이머	서비스가 데이터소스를 요청한 시간부터 결과가 서비스에 리턴된 시간까지 서버측에서 측정된 지속 기간

표 7. 콜백. 이 통계는 호출자에 콜백(원래 호출의 "형제")이 있을 때 발생합니다.

통계 이름	종류	설명
<b>GoodCB</b>	카운터	예외가 발생하지 않은 콜백에 대한 SCA 호출 수
<b>BadCB</b>	카운터	예외가 발생한 콜백에 대한 SCA 호출 수
<b>CBTime</b>	타이머	SCA가 콜백을 요청한 시간부터 콜백의 결과가 SCA에 리턴된 시간까지의 지속 기간
<b>GoodCBSubmit</b>	카운터	서비스에서 콜백을 핸들하는 SCA로의 예외가 발생하지 않은 호출 수
<b>BadCBSubmit</b>	카운터	서비스에서 콜백을 핸들하는 SCA로의 예외가 발생한 호출 수
<b>CBSubmitTime</b>	타이머	서비스가 콜백을 핸들하는 SCA를 요청한 시간부터 SCA의 결과가 서비스에 리턴된 시간까지의 지속 기간


다음 주제는 다양한 구현에서 이 통계가 사용되는 방식을 보여줍니다.


## 동기 호출

서비스에 대한 단순 SCA 호출 및 서비스의 응답에서 얻을 수 있는 ARM 성능 통계를 설명합니다.



## 매개변수

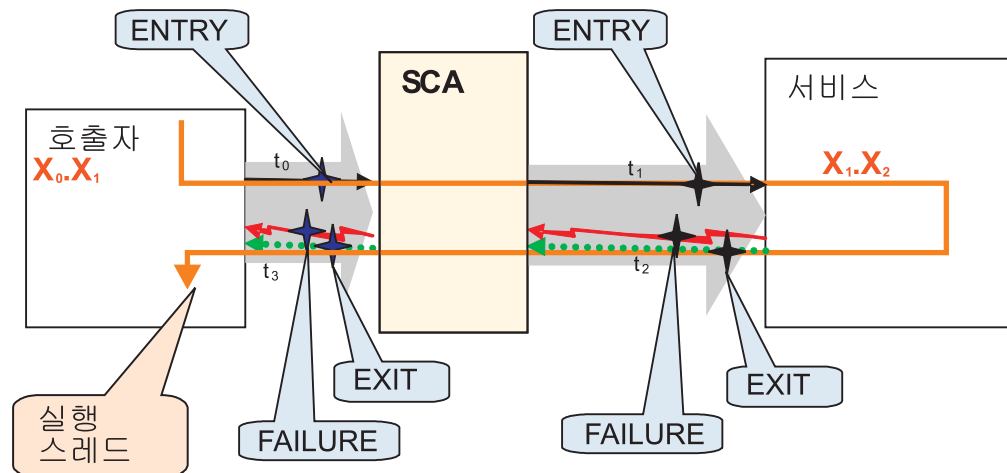
SCA 컴포넌트에 대한 이벤트 모니터링은 검은색  로 표시되는 이벤트 지점을 포

합하는 반면 파란색  로 표시되는 이벤트 지점은 파일 PMI/ARM 통계를 계산하는 데에만 사용됩니다.

아래의 표와 다이어그램에서 "현재" ARM 트랜잭션( $X_1$ 로 표시됨)은 호출 서비스 컴포넌트가 처음으로 호출되었을 때 작성됩니다. 호출자가 서비스 컴포넌트가 아닌 경우에는 현재 암(Arm) 트랜잭션이 사용되거나 새로 작성됩니다. 시작 트랜잭션이 아니면 상위가 있습니다. 이는 다음 표와 다이어그램에서  $X_n, X_{n+1}$ 로 표시됩니다. 이들은 트랜잭션 계보를 표시하는 데 사용됩니다. 모든 SCA 호출은 새 트랜잭션을 시작하며 호출자의 현재 트랜잭션이 이 트랜잭션의 상위가 됩니다. 새 트랜잭션을 작성할 수 있으며 현재 트랜잭션에 액세스할 수 있지만 SCA 트랜잭션 계보가 수정되지는 않습니다.

표 8. SCA의 동기 호출에 대한 암(Arm) 통계

통계	공식	ARM 트랜잭션
TotalResponseTime	$t_3 - t_0$	$X_0, X_1$
RequestDeliveryTime	$t_1 - t_0$	$X_1, X_2$
ResponseDeliveryTime	$t_3 - t_2$	
GoodRequests	Count <sub>EXIT</sub>	
BadRequests	Count <sub>FAILURE</sub>	
ProcessTime	$t_2 - t_1$	




## 동기 구현의 지연 응답

요청의 동기 호출로 얻을 수 있는 ARM 통계 및 데이터 스토어에 출력으로 전송되는 리턴된 결과가 비동기 구현에 대해 표시됩니다.



## 매개변수

SCA 컴포넌트에 대한 이벤트 모니터링은 검은색  로 표시되는 이벤트 지점을 포

합하는 반면 파란색  로 표시되는 이벤트 지점은 파일 PMI/ARM 통계를 계산하는 데에만 사용됩니다.

아래의 표와 다이어그램에서 "현재" ARM 트랜잭션( $X_1$ 로 표시됨)은 호출 서비스 컴포넌트가 처음으로 호출되었을 때 작성됩니다. 호출자가 서비스 컴포넌트가 아닌 경우에는 현재 암(Arm) 트랜잭션이 사용되거나 새로 작성됩니다. 시작 트랜잭션이 아니면 상위가 있습니다. 이는 다음 표와 다이어그램에서  $X_n, X_{n+1}$ 로 표시됩니다. 이들은 트랜잭션 계보를 표시하는 데 사용됩니다. 모든 SCA 호출은 새 트랜잭션을 시작하며 호출자의 현재 트랜잭션이 이 트랜잭션의 상위가 됩니다. 새 트랜잭션을 작성할 수 있으며 현재 트랜잭션에 액세스할 수 있지만 SCA 트랜잭션 계보가 수정되지는 않습니다.

표 9. 요청 및 리턴 결과 호출

유형	통계	공식	ARM 트랜잭션
공용	TotalResponseTime	$t_3 - t_0$	$X_0, X_1$
	RequestDeliveryTime	$t'_0 - t_0$	$X_1, X_2$
	ResponseDeliveryTime	해당 없음	해당 없음
	GoodRequests	$Count_{EXIT}$	$X_1, X_2$
	BadRequests	$Count_{FAILURE}$	
	ResponseTime	$t'_1 - t'_0$	
참조 A	GoodRefRequest	$Count_{EXIT}$	$X_1, X_2$
	BadRefRequests	$Count_{FAILURE}$	
	RefResponseTime	$t_1 - t_0$	

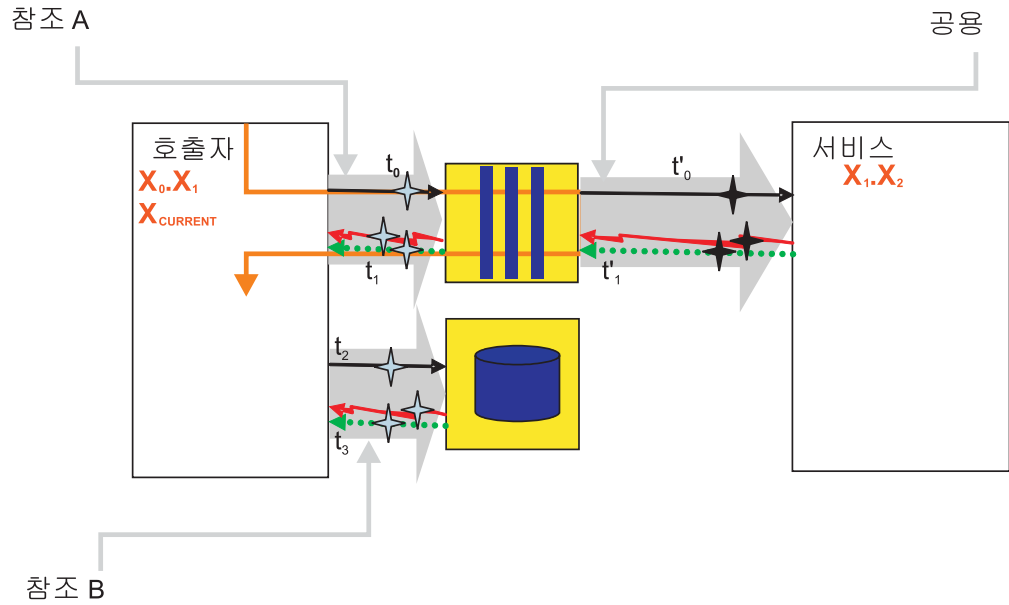



표 10. 데이터 소스에 대한 출력 호출


유형	통계	공식	ARM 트랜잭션
참조 B	GoodRetrieveResult	Count <sub>EXIT</sub>	X <sub>1</sub> .X <sub>2</sub>
	BadRetrieveResult	Count <sub>FAILURE</sub>	
	ResultRetrieveResponseTime	$\sum t_3 - t_2$	
	ResultRetrieveWaitTime	$\sum \text{timeout}$	

## 비동기 구현의 지연 응답

서비스 및 리턴 결과가 호출되지만 결과 출력이 서비스 대상에서 데이터 스토어로 전송되는 ARM 통계를 보여줍니다.

## 매개변수

SCA 컴포넌트에 대한 이벤트 모니터링은 검은색  로 표시되는 이벤트 지점을 포

함하는 반면 파란색  로 표시되는 이벤트 지점은 파일 PMI/ARM 통계를 계산하는 데에만 사용됩니다.

아래의 표와 다이어그램에서 "현재" ARM 트랜잭션(X<sub>1</sub>로 표시됨)은 호출 서비스 컴포넌트가 처음으로 호출되었을 때 작성됩니다. 호출자가 서비스 컴포넌트가 아닌 경우에는 현재 암(Arm) 트랜잭션이 사용되거나 새로 작성됩니다. 시작 트랜잭션이 아니면 상위가 있습니다. 이는 다음 표와 다이어그램에서 X<sub>n</sub>.X<sub>n+1</sub>로 표시됩니다. 이들은 트랜잭션 계보를 표시하는 데 사용됩니다. 모든 SCA 호출은 새 트랜잭션을 시작하며 호출자의

현재 트랜잭션이 이 트랜잭션의 상위가 됩니다. 새 트랜잭션을 작성할 수 있으며 현재 트랜잭션에 액세스할 수 있지만 SCA 트랜잭션 계보가 수정되지는 않습니다.

표 11. 요청 및 리턴 결과 호출

유형	통계	공식	ARM 트랜잭션
공용	TotalResponseTime	$t_3 - t_0$	$X_0 \cdot X_1$
	RequestDeliveryTime	$t'_0 - t_0$	$X_1 \cdot X_2$
	ResponseDeliveryTime	$t'_{03} - t'_2$	
	GoodRequests	$Count_{EXIT}$	
	BadRequests	$Count_{FAILURE}$	
	ResponseTime	$t'_3 - t'_0$	
참조 A	GoodRefRequest	$Count_{EXIT}$	
	BadRefRequests	$Count_{FAILURE}$	
	RefResponseTime	$t_1 - t_0$	
대상 A	GoodTargetSubmit	$Count_{EXIT}$	$X_1 \cdot X_2$
	BadTargetSubmit	$Count_{FAILURE}$	
	TargetSubmitTime	$t'_1 - t'_0$	

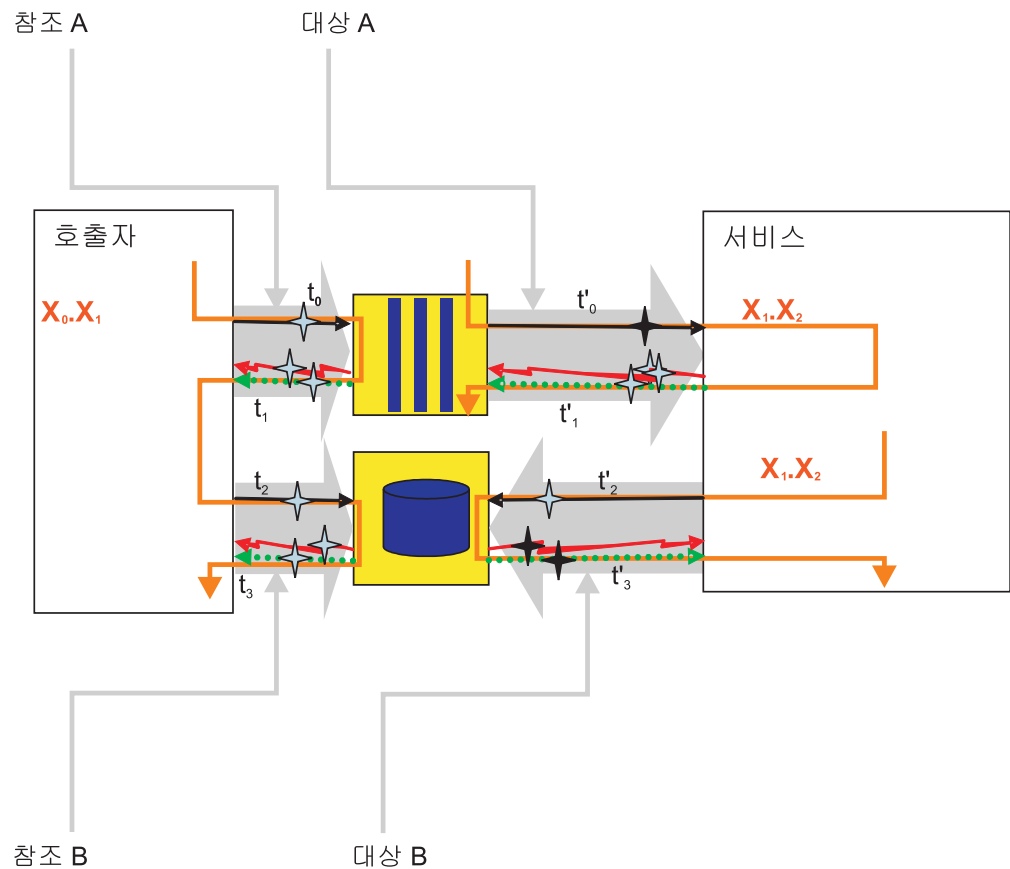




표 12. 데이터 스토어로의 리턴 결과 호출

유형	통계	공식	ARM 트랜잭션
참조 B	GoodResultSubmit	Count <sub>EXIT</sub>	$X_0 \cdot X_1$
	BadResultSubmit	Count <sub>FAILURE</sub>	
	ResultResponseTime	$t'_3 - t'_2$	
대상 B	GoodResultRetrieve	Count <sub>EXIT</sub>	$X_1 \cdot X_2$
	BadResultRetrieve	Count <sub>FAILURE</sub>	
	ResultRetrieveResponseTime	$\sum t_3 - t_2$	
	ResultRetrieveWaitTime	$\sum \text{timeout}$	

### 비동기 결과 검색의 지연 응답

$X_{\text{PARENT-1}}$  및  $X_{\text{PARENT-2}}$ 에 공통 조상 트랜잭션이 있는 경우에만 ARM 트랜잭션을 사용하여 ResultRetrieve ARM 통계를 일부 원래 요청과 상관시킬 수 있습니다. 요청 호출 및 결과 검색은 다른 스레드에서 발생합니다.

### 매개변수

SCA 컴포넌트에 대한 이벤트 모니터링은 검은색  로 표시되는 이벤트 지점을 포함하는 반면 파란색  로 표시되는 이벤트 지점은 파일 PMI/ARM 통계를 계산하는 데에만 사용됩니다.

아래의 표와 다이어그램에서 "현재" ARM 트랜잭션( $X_1$ 로 표시됨)은 호출 서비스 컴포넌트가 처음으로 호출되었을 때 작성됩니다. 호출자가 서비스 컴포넌트가 아닌 경우에는 현재 암(Arm) 트랜잭션이 사용되거나 새로 작성됩니다. 시작 트랜잭션이 아니면 상위가 있습니다. 이는 다음 표와 다이어그램에서  $X_n \cdot X_{n+1}$ 로 표시됩니다. 이들은 트랜잭션 계보를 표시하는 데 사용됩니다. 모든 SCA 호출은 새 트랜잭션을 시작하며 호출자의 현재 트랜잭션이 이 트랜잭션의 상위가 됩니다. 새 트랜잭션을 작성할 수 있으며 현재 트랜잭션에 액세스할 수 있지만 SCA 트랜잭션 계보가 수정되지는 않습니다.

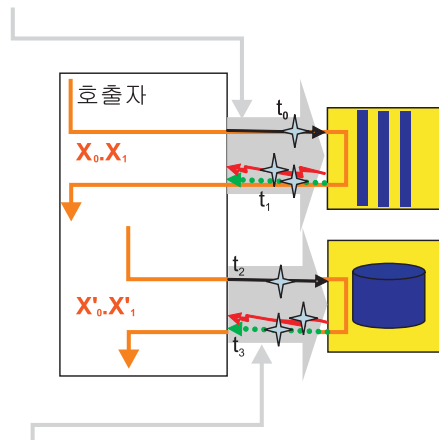
표 13. 요청 및 리턴 결과 호출

유형	통계	공식	ARM 트랜잭션
공용	TotalResponseTime	$t_3 - t_0$	$X_0 \cdot X_1$
	RequestDeliveryTime	$t'_0 - t_0$	$X_1 \cdot X_2$
	ResponseDeliveryTime	해당 없음	해당 없음
	GoodRequests	Count <sub>EXIT</sub>	$X_1 \cdot X_2$
	BadRequests	Count <sub>FAILURE</sub>	
	ResponseTime	특정 다이어그램을 보십시오.	

표 13. 요청 및 리턴 결과 호출 (계속)

유형	통계	공식	ARM 트랜잭션
참조 A	GoodReferenceRequest	Count <sub>EXIT</sub>	X <sub>1</sub> .X <sub>2</sub>
	BadReferenceRequests	Count <sub>FAILURE</sub>	
	ReferenceResponseTime	t <sub>1</sub> - t <sub>0</sub>	

참조 A



다이어그램의 오른쪽의 경우, "동기/동기 구현의 지연 응답" 주제의 다이어그램을 참조하십시오.  
요청 수신 시간 t'0은 두 경우 모두 동일합니다.

참조 B

표 14. 요청 및 리턴 결과 호출

유형	통계	공식	ARM 트랜잭션
참조 B	GoodRetrieveResult	Count <sub>EXIT</sub>	X' <sub>0</sub> .X' <sub>1</sub>
	BadRetrieveResult	Count <sub>FAILURE</sub>	
	RetrieveResultResponseTime	∑ t <sub>3</sub> - t <sub>2</sub>	
	RetrieveResultWaitTime	∑ timeout	

### 동기 구현의 비동기 콜백

ARM 통계는 콜백 요청 및 콜백 실행이 동기 구현에서 서로 다른 스레드를 사용할 때 사용 가능합니다.

### 매개변수

SCA 컴포넌트에 대한 이벤트 모니터링은 검은색 ✨ 로 표시되는 이벤트 지점을 포함하는 반면 파란색 ✨ 로 표시되는 이벤트 지점은 파일 PMI/ARM 통계를 계산하는 데에만 사용됩니다.

아래의 표와 다이어그램에서 "현재" ARM 트랜잭션(X<sub>1</sub>로 표시됨)은 호출 서비스 컴포넌트가 처음으로 호출되었을 때 작성됩니다. 호출자가 서비스 컴포넌트가 아닌 경우에

은 현재 암(Arm) 트랜잭션이 사용되거나 새로 작성됩니다. 시작 트랜잭션이 아니면 상위가 있습니다. 이는 다음 표와 다이어그램에서  $X_n, X_{n+1}$ 로 표시됩니다. 이들은 트랜잭션 계보를 표시하는 데 사용됩니다. 모든 SCA 호출은 새 트랜잭션을 시작하며 호출자의 현재 트랜잭션이 이 트랜잭션의 상위가 됩니다. 새 트랜잭션을 작성할 수 있으며 현재 트랜잭션에 액세스할 수 있지만 SCA 트랜잭션 계보가 수정되지는 않습니다.

표 15. 요청 및 리턴 결과 호출

유형	통계	공식	ARM 트랜잭션
공용	TotalResponseTime	$t_2 - t_0$	$X_0 \cdot X_1$
	RequestDeliveryTime	$t'_0 - t_0$	$X_1 \cdot X_2$
	ResponseDeliveryTime	$t_2 - t'_1$	
	GoodRequests	$Count_{EXIT}$	
	BadRequests	$Count_{FAILURE}$	
	ResponseTime	$t_3 - t_2$	
참조	GoodRefRequest	$Count_{EXIT}$	$X_1 \cdot X_2$
	BadRefRequests	$Count_{FAILURE}$	
	RefResponseTime	$t'_1 - t'_0$	

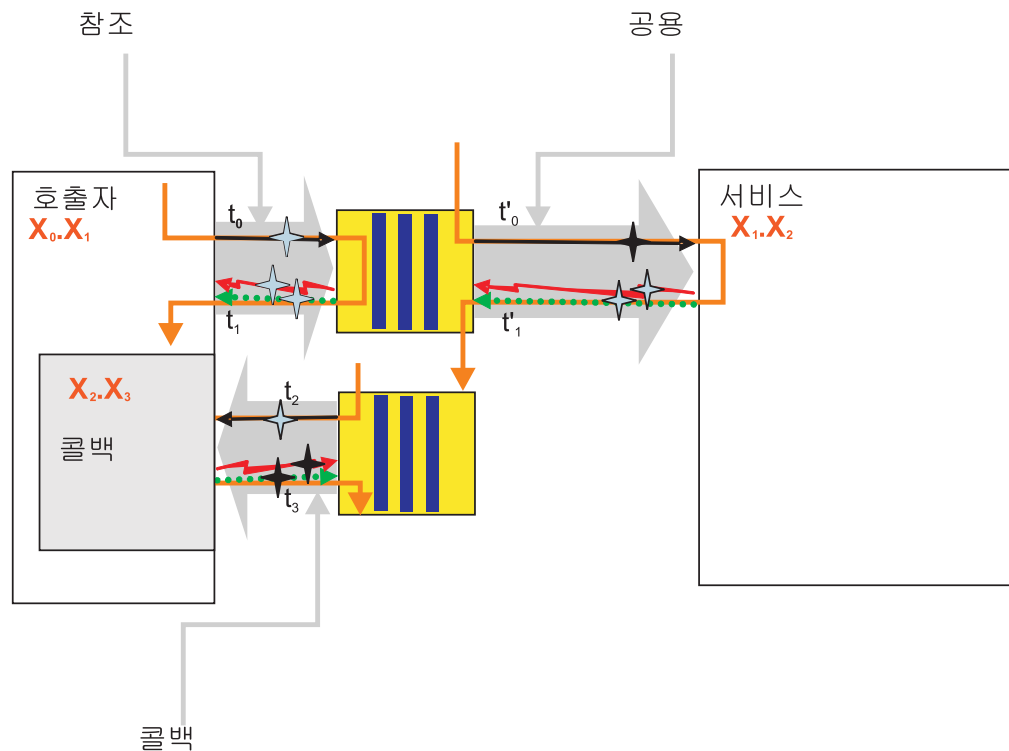



표 16. 콜백의 호출


유형	통계	공식	ARM 트랜잭션
콜백	GoodCB	$Count_{EXIT}$	$X_1 \cdot X_3$
	BadCB	$Count_{FAILURE}$	
	CBTime	$t_3 - t_2$	

## 비동기 구현의 비동기 콜백

비동기 구현의 다른 스레드를 사용한 콜백 요청 및 콜백 실행에 사용 가능한 ARM 통계

### 매개변수

SCA 컴포넌트에 대한 이벤트 모니터링은 검은색  로 표시되는 이벤트 지점을 포

합하는 반면 파란색  로 표시되는 이벤트 지점은 파일 PMI/ARM 통계를 계산하는 데에만 사용됩니다.

아래의 표와 다이어그램에서 "현재" ARM 트랜잭션( $X_1$ 로 표시됨)은 호출 서비스 컴포넌트가 처음으로 호출되었을 때 작성됩니다. 호출자가 서비스 컴포넌트가 아닌 경우에는 현재 암(Arm) 트랜잭션이 사용되거나 새로 작성됩니다. 시작 트랜잭션이 아니면 상위가 있습니다. 이는 다음 표와 다이어그램에서  $X_n, X_{n+1}$ 로 표시됩니다. 이들은 트랜잭션 계보를 표시하는 데 사용됩니다. 모든 SCA 호출은 새 트랜잭션을 시작하며 호출자의 현재 트랜잭션이 이 트랜잭션의 상위가 됩니다. 새 트랜잭션을 작성할 수 있으며 현재 트랜잭션에 액세스할 수 있지만 SCA 트랜잭션 계보가 수정되지는 않습니다.

표 17. 요청 및 리턴 결과 호출

유형	통계	공식	ARM 트랜잭션
공용	TotalResponseTime	$t_2 - t_0$	$X_0 \cdot X_1$
	RequestDeliveryTime	$t'_0 - t_0$	$X_1 \cdot X_2$
	ResponseDeliveryTime	$t_2 - t'_2$	
	GoodRequests	Count <sub>EXIT</sub>	
	BadRequests	Count <sub>FAILURE</sub>	
	ResponseTime	$t'_3 - t'_0$	
참조 A	GoodRefRequest	Count <sub>EXIT</sub>	$X_0 \cdot X_1$
	BadRefRequests	Count <sub>FAILURE</sub>	
	RefResponseTime	$t_1 - t_0$	
대상 A	GoodTargetSubmit	Count <sub>EXIT</sub>	$X_1 \cdot X_2$
	BadTargetSubmit	Count <sub>FAILURE</sub>	
	TargetSubmitTime	$t'_1 - t'_0$	

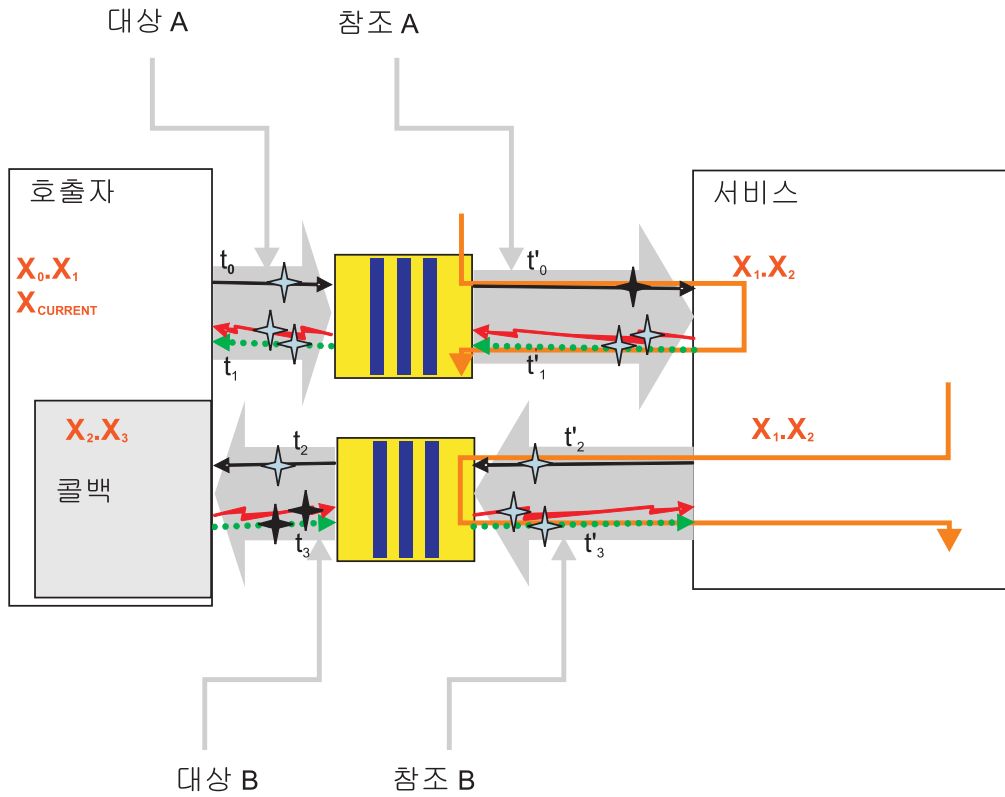




표 18. 콜백의 호출

유형	통계	공식	ARM 트랜잭션
참조 B	GoodCBSubmit	$Count_{EXIT}$	$X_1.X_2$
	BadCBSubmit	$Count_{FAILURE}$	
	CBSubmitTime	$t'_3 - t'_2$	
대상 B	GoodCB	$Count_{EXIT}$	$X_0.X_1$
	BadCB	$Count_{FAILURE}$	
	CBTime	$t_3 - t_2$	

### 동기 구현의 비동기 단방향

동기 구현으로 호출이 제출(가동 후 일임)된 경우의 ARM 통계

### 매개변수

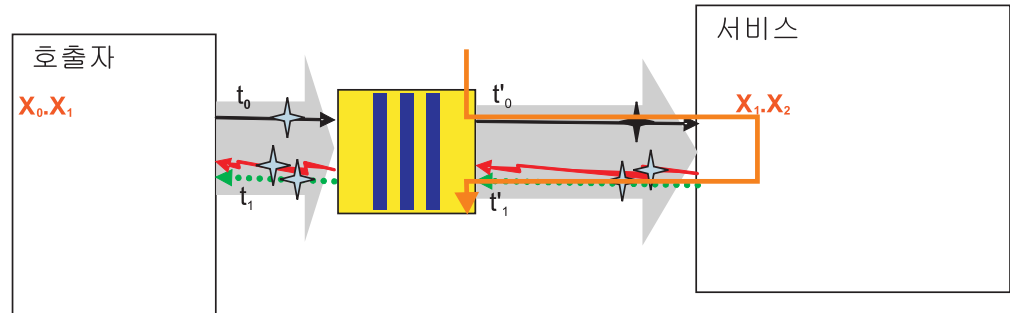
SCA 컴포넌트에 대한 이벤트 모니터링은 검은색  로 표시되는 이벤트 지점을 포함하는 반면 파란색  로 표시되는 이벤트 지점은 파일 PMI/ARM 통계를 계산하는 데에만 사용됩니다.



아래의 표와 다이어그램에서 "현재" ARM 트랜잭션( $X_1$ 로 표시됨)은 호출 서비스 컴포넌트가 처음으로 호출되었을 때 작성됩니다. 호출자가 서비스 컴포넌트가 아닌 경우에는 현재 압(Arm) 트랜잭션이 사용되거나 새로 작성됩니다. 시작 트랜잭션이 아니면 상위가 있습니다. 이는 다음 표와 다이어그램에서  $X_n, X_{n+1}$ 로 표시됩니다. 이들은 트랜잭션 계보를 표시하는 데 사용됩니다. 모든 SCA 호출은 새 트랜잭션을 시작하며 호출자의 현재 트랜잭션이 이 트랜잭션의 상위이 됩니다. 새 트랜잭션을 작성할 수 있으며 현재 트랜잭션에 액세스할 수 있지만 SCA 트랜잭션 계보가 수정되지는 않습니다.

표 19. 요청 및 리턴 결과 호출


유형	통계	공식	ARM 트랜잭션
공용	TotalResponseTime	$t_1 - t_0$	$X_0, X_1$
	RequestDeliveryTime	$t'_0 - t_0$	$X_1, X_2$
	ResponseDeliveryTime	해당 없음	해당 없음
	GoodRequests	$Count_{EXIT}$	$X_1, X_2$
	BadRequests	$Count_{FAILURE}$	
	ResponseTime	$t'_1 - t'_0$	




### 비동기 구현의 비동기 단방향

비동기 구현으로 호출이 제출(가동 후 일임)된 경우의 ARM 통계

### 매개변수

SCA 컴포넌트에 대한 이벤트 모니터링은 검은색  로 표시되는 이벤트 지점을 포

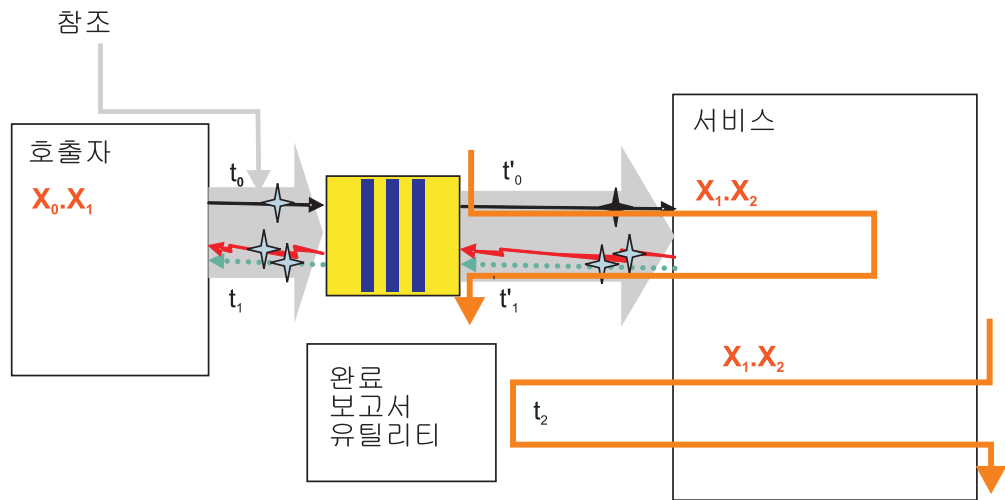
함하는 반면 파란색  로 표시되는 이벤트 지점은 파일 PMI/ARM 통계를 계산하는 데에만 사용됩니다.

아래의 표와 다이어그램에서 "현재" ARM 트랜잭션( $X_1$ 로 표시됨)은 호출 서비스 컴포넌트가 처음으로 호출되었을 때 작성됩니다. 호출자가 서비스 컴포넌트가 아닌 경우에는 현재 압(Arm) 트랜잭션이 사용되거나 새로 작성됩니다. 시작 트랜잭션이 아니면 상위가 있습니다. 이는 다음 표와 다이어그램에서  $X_n, X_{n+1}$ 로 표시됩니다. 이들은 트랜잭션

계보를 표시하는 데 사용됩니다. 모든 SCA 호출은 새 트랜잭션을 시작하며 호출자의 현재 트랜잭션이 이 트랜잭션의 상위가 됩니다. 새 트랜잭션을 작성할 수 있으며 현재 트랜잭션에 액세스할 수 있지만 SCA 트랜잭션 계보가 수정되지는 않습니다.

표 20. 요청 및 리턴 결과 호출

유형	통계	공식	ARM 트랜잭션
공용	TotalResponseTime	$t_1 - t_0$	$X_{PARENT} \cdot X_{REQUEST}$
	RequestDeliveryTime	$t'_0 - t_0$	$X_{REQUEST} \cdot X_{PROCESS}$
	ResponseDeliveryTime	해당 없음	해당 없음
	GoodRequests	$Count_{EXIT}$	$X_{REQUEST} \cdot X_{PROCESS}$
	BadRequests	$Count_{FAILURE}$	
	ResponseTime	$t'_2 - t_0$	
참조	GoodRefRequest	$Count_{EXIT}$	$X_{PARENT} \cdot X_{REQUEST}$
	BadRefRequest	$Count_{FAILURE}$	
	RefResponseDuration	$t_1 - t_0$	



## 서비스 컴포넌트 이벤트 모니터링

WebSphere Process Server 모니터링은 특정 이벤트 지점에서 서비스 컴포넌트의 데이터를 캡처할 수 있습니다. 로그 파일의 각 이벤트를 보거나 CEI(Common Event Infrastructure) 서버의 보다 융통성 있는 모니터링 기능을 사용할 수 있습니다.

프로세스 서버에 전개되는 응용프로그램은 응용프로그램이 실행하는 한 모니터링 서비스 컴포넌트 이벤트의 스펙을 포함할 수 있습니다. WebSphere Integration Developer를 사용하여 응용프로그램을 개발했으면 계속해서 모니터링 서비스 컴포넌트 이벤트를 지정할 수 있습니다. 이 스펙은 응용프로그램의 일부로 포함되며 응용프로그램이 전개될 때 프로세스 서버가 읽으며 확장자가 .mon인 파일 양식을 사용합니다. 응용프로그램이 시작된 후에는 .mon 파일에 지정된 서비스 컴포넌트의 모니터링을 끌 수 없게 됩

니다. WebSphere Process Server의 문서에서는 이 유형의 연속 모니터링을 다루지 않습니다. 이 주제에 대한 자세한 정보는 WebSphere Integration Developer 문서를 참조하십시오.

WebSphere Process Server를 사용하여 응용프로그램의 .mon 파일에 아직 지정되지 않은 서비스 컴포넌트 이벤트를 모니터링할 수 있습니다. 이벤트 모니터의 출력이 로그 파일이나 CEI(Common Event Infrastructure) 서버 데이터베이스로 지정되도록 프로세스 서버를 구성할 수 있습니다. 모니터링되는 이벤트는 공통 기본 이벤트 표준을 사용하여 형식화되지만 각 이벤트에 포함되는 정보의 양을 규제할 수 있습니다. WebSphere Process Server의 모니터링 기능을 사용하여 문제점을 진단하거나, 응용프로그램의 프로세스 플로우를 분석하거나 또는 응용프로그램이 사용되는 방식을 감사하십시오.

## 비즈니스 프로세스 및 휴먼 타스크 이벤트의 모니터링 사용

비즈니스 프로세스 및 휴먼 타스크 서비스 컴포넌트의 실제 모니터링을 수행하기 전에 이러한 종류의 서비스 컴포넌트 모니터링을 지원하도록 WebSphere Process Server를 구성해야 합니다.

### 시작하기 전에

비즈니스 프로세스 컨테이너 및 휴먼 타스크 컨테이너를 프로세스 서버에 이전에 작성했어야 합니다.

### 타스크 정보

이 타스크를 수행하여 WebSphere Process Server의 CEI(Common Event Infrastructure) 모니터링 지원을 사용 가능하게 하십시오.

### 프로시저

1. 관리 콘솔을 여십시오.
2. Business Process Choreographer가 단일 서버에 구성된 경우 다음 단계를 완료하여 서버에서 비즈니스 프로세스 이벤트를 생성할 수 있게 하십시오.
  - a. 왼쪽 프레임에서 서버를 펼치고 **Application Server** > *server\_name*을 클릭 하십시오.
  - b. 컨테이너 설정 > **Business Process Choreographer** 컨테이너 설정 > **Business Process Choreographer** 컨테이너를 선택하십시오.
  - c. 상태 업저버를 클릭하십시오.
  - d. 감사 로깅 및 **CEI(Common Event Infrastructure)** 로깅 선택란이 비즈니스 플로우 관리자 및 휴먼 타스크 관리자 모두에 대해 선택되었는지 확인하십시오. 선택란이 선택되지 않은 경우에는 먼저 선택란을 선택하고 서버를 다시 시작해야 합니다.

3. Business Process Choreographer가 클러스터에 구성된 경우 다음 단계를 완료하여 클러스터에서 비즈니스 프로세스 이벤트를 생성할 수 있게 하십시오.
  - a. 클러스터 > *cluster\_name*을 선택하십시오.
  - b. **Business Process Choreographer** 컨테이너 설정 > **Business Process Choreographer** 컨테이너를 선택하십시오.
  - c. 상태 옵저버를 클릭하십시오.
  - d. 감사 로깅 및 CEI(Common Event Infrastructure) 로깅 선택란이 비즈니스 플로우 관리자 및 휴먼 태스크 관리자 모두에 대해 선택되었는지 확인하십시오. 선택란이 선택되지 않은 경우에는 먼저 선택란을 선택하고 클러스터를 다시 시작해야 합니다.

다음에 수행할 작업

선택란을 선택해야 한 경우에는 변경사항이 적용되도록 서버나 클러스터를 다시 시작해야 합니다.

## 서비스 컴포넌트 이벤트에 대한 로깅 구성

프로세스 서버 모니터링으로 가동된 서비스 컴포넌트 이벤트를 캡처하는 WebSphere Application Server의 로깅 기능을 사용하도록 선택할 수 있습니다. 로거를 사용하여 응용프로그램 처리에 대한 문제점을 진단할 때 이벤트의 데이터를 보십시오.

WebSphere Process Server는 기본WebSphere Application Server의 광대한 로깅 기능을 사용하여 서비스 컴포넌트 이벤트 지점에서 프로세스 서버 모니터링으로 가동된 이벤트를 캡처할 수 있습니다. 관리 콘솔을 사용하여 모니터링하려는 특정 서비스 컴포넌트 이벤트 지점, 결과 서비스 컴포넌트 이벤트에 포함된 페이로드 세부사항의 양 및 결과를 공개하는 데 사용된 메소드(예: 특정 형식의 파일로 또는 콘솔에 직접)를 지정할 수 있습니다. 모니터 로그는 공통 기본 이벤트 형식으로 인코딩된 이벤트를 포함하며 이벤트 요소에 포함된 정보를 사용하여 서비스 컴포넌트 처리에 대한 문제점을 추적할 수 있습니다.

WebSphere Application Server 로깅 및 추적 기능은 로깅 및 추적이 전체 제품 내에서 사용되는 방식에 대한 완전한 설명과 함께 WebSphere Application Server 문서에서 상당히 자세하게 설명합니다. 이 섹션은 WebSphere Process Server에 특정한 서비스 컴포넌트에 관련되어 있어서 로깅에 대한 보충 정보만을 제공합니다. 전체 제품의 기타 컴포넌트에 관한 로깅 및 추적 사용에 대해서는 WebSphere Application Server 문서의 정보를 참조하십시오.

### 진단 추적 서비스 사용

이 태스크를 사용하여 서비스 컴포넌트 이벤트에 포함된 세부사항의 양을 관리할 수 있는 로깅 서비스인 진단 추적 서비스를 사용 가능하게 하십시오.

## 시작하기 전에

CEI(Common Event Infrastructure) 로깅 및 감사 로깅을 허용하도록 비즈니스 프로세스 및 휴먼 타스크 컨테이너를 구성해야 합니다.

## 타스크 정보

진단 추적 서비스는 서비스 컴포넌트 이벤트의 요소에 포함된 세부사항을 캡처하는 데 필요한 세부사항 레벨을 제공할 수 있는 유일한 로거 유형입니다. 이벤트를 로깅하려면 프로세스 서버를 시작하기 전에 진단 추적 서비스를 사용 가능하게 해야 합니다. 관리 콘솔을 통해 CEI 서버를 사용하여 모니터링하기 위한 서비스 컴포넌트 이벤트 지점을 선택하는 경우에도 서비스가 사용 가능해야 합니다.

## 프로시저

1. 탐색 패널에서 서버 > **Application Server**를 클릭하십시오.
2. 작업하려는 서버 이름을 클릭하십시오.
3. 문제점 해결 아래에서 진단 추적 서비스를 클릭하십시오.
4. 구성 탭에서 **로그 사용**을 선택하십시오.
5. 적용을 클릭한 다음 저장을 클릭하십시오.
6. 확인을 클릭하십시오.

## 다음에 수행할 작업

프로세스 서버가 이미 시작되었으면 변경사항이 적용되도록 서버를 다시 시작해야 합니다.

## 관리 콘솔을 사용한 로깅 특성 구성

이 작업을 사용하여 모니터링 함수가 서비스 컴포넌트 이벤트를 로거 파일에 공개하도록 지정하십시오.

## 타스크 정보

WebSphere Process Server 응용프로그램이 모니터링 이벤트를 로깅하려면 모니터링하려는 서비스 컴포넌트 이벤트 지점, 각 이벤트에 필요한 세부사항 레벨 및 로그에 이벤트를 공개하는 데 사용되는 출력 형식을 지정해야 합니다. 관리 콘솔을 사용하여 다음을 수행할 수 있습니다.

- 특정 이벤트 로그를 사용 가능 또는 불가능으로 설정합니다.
- 로그의 세부사항 레벨을 지정합니다.
- 로그 파일이 저장되는 위치, 보관되는 로그 파일 수 및 로그 출력의 형식을 지정합니다.

로그 구성을 정적 또는 동적으로 변경할 수 있습니다. 정적 구성 변경은 Application Server를 시작 또는 다시 시작할 때 응용프로그램에 영향을 줍니다. 동적 또는 런타임 구성 변경은 즉시 적용됩니다.

로그가 작성되면 이 로그의 레벨 값이 구성 데이터로부터 설정됩니다. 특정 로그 이름으로 사용할 구성 데이터가 없으면 로그의 상위에서 이 로그의 레벨을 얻습니다. 상위 로그에 대한 구성 데이터가 존재하지 않는 경우에는 레벨 값이 널(null)이 아닌 로그를 찾을 때까지 트리의 위쪽으로 이 로그의 상위를 검토합니다. 로그 레벨을 변경하면 로그의 하위에 변경사항이 전파되고 필요에 따라 반복적으로 하위에 변경사항을 전파합니다.

### 프로시저

1. 로깅을 사용 가능하게 하고 로그의 출력 특성을 설정하십시오.
2. 탐색 패널에서 서버 > **Application Server**를 클릭하십시오.
3. 작업하려는 서버 이름을 클릭하십시오.
4. 문제점 해결 아래에서 로깅 및 추적을 클릭하십시오.
5. 로그 세부사항 레벨 변경을 클릭하십시오.
6. 컴포넌트, 패키지 및 그룹의 목록은 실행 중인 서버에 현재 등록된 모든 컴포넌트를 표시합니다. 적어도 한번은 호출된 프로세스 서버 이벤트만이 이 목록에 나타납니다. 로그될 수 있는 모드 프로세스 서버 컴포넌트는 **WBILocationMonitor.LOG** 이름으로 시작하는 컴포넌트 중 한 컴포넌트 아래에 나열됩니다.
  - 구성을 정적으로 변경할 이벤트를 선택하려면 구성 탭을 클릭하십시오.
  - 구성을 동적으로 변경할 이벤트를 선택하려면 런타임 탭을 클릭하십시오.
7. 로깅하려는 이벤트 또는 이벤트 그룹을 선택하십시오.
8. 각 이벤트 또는 이벤트 그룹에 대한 로깅 레벨을 설정하십시오.

주: FINE, FINER 및 FINEST 레벨만이 CEI 이벤트에 유효합니다.
9. 적용을 클릭하십시오.
10. 확인을 클릭하십시오.
11. 정적 구성 변경사항을 적용하려면 프로세스 서버를 중지한 다음 다시 시작하십시오.

### 결과

기본적으로 로거는 *install\_root/profiles/profile\_name/logs/server\_name* 폴더에 있는 *trace.log*라는 파일에 출력을 공개합니다.

## 학습서: 서비스 컴포넌트 이벤트 로깅

이 학습서는 로거에 공개되는 서비스 컴포넌트 이벤트 모니터링을 설정하는 방법 및 로그 파일의 이벤트를 보는 방법에 대한 예제를 제공합니다. 이 예제의 시나리오는 이미 개발되어 프로세스 서버에서 실행 중인 응용프로그램에서 모니터링할 서비스 컴포넌트 이벤트 지점을 선택하는 방법을 보여줍니다. 응용프로그램의 처리가 이 이벤트 지점 중 하나에 도달할 때마다 모니터링 함수가 이벤트를 가동하는 방식을 보게 됩니다. 가동된 각 이벤트는 XML 문자열로 로그 파일에 직접 공개되는 표준화된 공통 기본 이벤트 양식을 사용합니다.

### 이 학습서의 목표

이 학습서를 완료한 후에는 다음을 수행할 수 있습니다.

- 프로세스 서버 로거에 출력이 공개된 모니터링할 서비스 컴포넌트 이벤트 지점을 선택합니다.
- 로그 파일의 저장된 이벤트를 봅니다.

### 이 학습을 완료하는 데 필요한 시간

이 학습서를 완료하려면 약 15 - 20분이 소요됩니다.

### 전제조건

이 학습서를 수행하려면 다음을 완료했어야 합니다.

- 프로세스 서버를 구성하고 시작했습니다.
- CEI(Common Event Infrastructure)를 구성했습니다.
- 프로세스 서버의 진단 추적 서비스를 사용 가능하게 했습니다.
- 프로세스 서버에 샘플 갤러리 응용프로그램을 설치하고 시작했습니다.
- 프로세스 서버에 비즈니스 규칙 샘플 응용프로그램을 설치하고 시작했습니다. 샘플 갤러리 페이지의 지시사항에 따라 비즈니스 규칙 샘플 응용프로그램을 설정하고 실행하십시오.

이 모든 전제조건이 완료된 후 학습서를 진행하기 전에 샘플 갤러리에서 비즈니스 규칙 샘플 응용프로그램을 적어도 한번은 실행하십시오.

### 예제: 로거의 이벤트 모니터링: **타스크 정보**

이 시나리오의 비즈니스 규칙 샘플 응용프로그램을 사용하므로 웹 페이지에 이 응용프로그램이 이미 열려 있어야 합니다. 모니터링 매개변수를 지정한 후 샘플을 실행해야 하므로 응용프로그램을 계속 열어 놓으십시오. 샘플을 최소 한번은 이미 실행했는지 확인하십시오. 실행한 경우 모니터링하려고 선택할 수 있는 함수 목록에 샘플이 나타납니다.

### 프로시저



1. 관리 콘솔을 여십시오.
2. 탐색 패널에서 서버 > **Application Server**를 클릭하십시오.
3. *server\_name*을 클릭하십시오.
4. 문제점 해결 아래에서 로깅 및 추적을 클릭하십시오.
5. 로그 세부사항 레벨 변경을 클릭하십시오.
6. 런타임 탭을 선택하십시오.
7. **WBILocationMonitor.LOG.BR**에 대한 트리를 펼치십시오. 그러면 **WBILocationMonitor.LOG.BR.brsample.\*** 요소 아래에 7개의 이벤트 유형이 나타납니다. 첫 번째 이벤트는 **WBILocationMonitor.LOG.BR.brsample\_module.DiscountRuleGroup**이며 다음 네이처가 있는 **Operation.\_calculateDiscount**라는 단일 함수를 포함합니다.
  - ENTRY
  - EXIT
  - FAILURE
  - SelectionKeyExtracted
  - TargetFound
8. 각 이벤트를 클릭하고 **finest**를 선택하십시오.
9. 확인을 클릭하십시오.
10. 비즈니스 규칙 샘플 응용프로그램 페이지를 전환하고 응용프로그램을 한번 실행하십시오.
11. 문서 편집기를 사용하여 시스템의 *profile\_root/logs/server\_name* 폴더에 있는 *trace.log* 파일을 여십시오.

## 결과

샘플 응용프로그램을 실행했을 때 모니터가 가동한 비즈니스 규칙 이벤트가 포함된 로그의 행이 표시되어야 합니다. 유의해야 할 주요 사항은 공통 기본 이벤트 표준에 따른 구문 분석되지 않은 긴 XML 문자열로 출력이 이루어진다는 사실입니다. ENTRY 및 EXIT 이벤트를 검토하십시오. 그러면 **finest** 세부사항 레벨을 선택했기 때문에 포함된 비즈니스 오브젝트가 16진 형식으로 인코딩된 것을 알 수 있습니다. XML을 읽기 쉬운 테이블로 구문 분석하고 비즈니스 오브젝트 데이터를 읽기 쉬운 형식으로 해독하는, CEI(Common Event Infrastructure) 서버에 공개된 이벤트와 이 출력을 비교해보십시오. 이 학습을 거슬러 올라가서 세부사항 레벨을 **finest**에서 **fine** 또는 **finer**로 변경한 후 이벤트 간의 차이점을 비교해 볼 수 있습니다.

학습을 완료한 후에는 로그에 모니터할 서비스 컴포넌트 이벤트 지점을 선택하는 방법을 이해해야 합니다. 이 유형의 모니터링에 가동된 이벤트가 표준 형식이며 결과는 로



그 파일에 직접 원시 XML 형식의 문자열로 공개되는 것을 보았습니다. 공개된 이벤트를 보는 것은 문서 편집기에서 로그 파일을 열고 개별 이벤트의 콘텐츠를 해독하는 간단한 문제입니다.

#### 다음에 수행할 작업

비즈니스 규칙 샘플 응용프로그램을 더 이상 모니터링하지 않으려면 여기에 설명된 단계를 거슬러 올라가서 샘플 이벤트의 세부사항 레벨을 **info**로 재설정할 수 있습니다.

#### 비즈니스 규칙 및 선택기에 대한 감사 로깅

비즈니스 규칙 및 선택기에 대한 변경사항을 자동으로 로깅하고 변경하도록 WebSphere Process Server를 설정할 수 있습니다.

비즈니스 규칙 및 선택기가 변경되는 시기를 자동으로 감지해서 로그 파일에 변경사항이 자세히 설명된 항목을 작성하도록 서버를 구성할 수 있습니다. 표준 JVM SystemOut.log 파일이나 선택한 사용자 정의 감사 로그 파일에 로그 항목을 쓰도록 선택할 수 있습니다. 변경이 수행된 방식에 따라 각 비즈니스 규칙 또는 선택기 변경이 수행되는 프로세스 서버가 다음을 로깅합니다.

- 변경을 수행한 사용자의 이름
- 변경 요청이 발생한 위치
- 이전 비즈니스 규칙 또는 선택기 오브젝트
- 이전 오브젝트를 바꾸는 새 비즈니스 규칙 또는 선택기

비즈니스 규칙 및 선택기 오브젝트는 완전한 비즈니스 규칙 세트, 결정 테이블, 비즈니스 규칙 그룹 또는 교체되는 비즈니스 규칙 및 선택기와 이를 바꾼 새 버전 모두에 대한 선택기입니다. 로그를 확인해서(감사 출력은 CEI(Common Event Infrastructure) 데이터베이스에 지정할 수 없음) 이전 및 새 비즈니스 규칙 또는 선택기를 비교하여 변경사항을 판별해야 합니다. 다음 시나리오는 로깅이 발생하는(구성된 경우) 상황 및 로그 항목의 콘텐츠를 설명합니다.

시나리오	결과	로그 항목 콘텐츠
비즈니스 규칙 관리자를 사용하여 비즈니스 규칙을 공개함	요청	사용자 ID, 서버 이름(적용 가능한 경우 셀과 노드 포함), 이전 비즈니스 규칙 규칙 세트, 새 규칙 세트
	장애	사용자 ID, 서버 이름(적용 가능한 경우 셀과 노드 포함), 이전 비즈니스 규칙 규칙 세트, 새 규칙 세트
저장소 데이터베이스 갱신 및 확장 (비즈니스 규칙 관리자를 사용한 공개 시도에서)	성공	사용자 ID, 이전 규칙 세트, 새 규칙 세트
	장애	사용자 ID, 새 규칙 세트

시나리오	결과	로그 항목 콘텐츠
선택기 또는 비즈니스 규칙 그룹 내보내기	요청	사용자 ID, 선택기 또는 비즈니스 규칙 그룹 이름
	성공	사용자 ID, 서버 이름(적용 가능한 경우 셀과 노드 포함), 내보낸 선택기 또는 비즈니스 규칙 그룹의 사본
	장애	사용자 ID, 서버 이름(적용 가능한 경우 셀과 노드 포함), 선택기 또는 비즈니스 규칙 그룹 이름
선택기 또는 비즈니스 규칙 그룹 가져오기	요청	사용자 ID, 새 선택기 또는 비즈니스 규칙 그룹의 사본
	성공	사용자 ID, 서버 이름(적용 가능한 경우 셀과 노드 포함), 가져온 선택기 또는 비즈니스 규칙 그룹의 사본, 가져온 버전으로 바꾼 선택기 또는 비즈니스 규칙 그룹의 사본
	장애	사용자 ID, 서버 이름(적용 가능한 경우 셀과 노드 포함), 가져오려 한 선택기 또는 비즈니스 규칙 그룹의 사본
응용프로그램 설치	성공	사용자 ID, 서버 이름(적용 가능한 경우 셀과 노드 포함), 선택기 또는 비즈니스 규칙 그룹 이름
	장애	사용자 ID, 서버 이름(적용 가능한 경우 셀과 노드 포함), 선택기 또는 비즈니스 규칙 그룹 이름
응용프로그램 갱신(관리 콘솔 또는 wsadmin 명령을 통해)	성공	사용자 ID, 서버 이름(적용 가능한 경우 셀과 노드 포함), 새 선택기 또는 비즈니스 규칙 그룹의 사본, 이전 선택기 또는 비즈니스 규칙 그룹의 사본
	장애	사용자 ID, 서버 이름(적용 가능한 경우 셀과 노드 포함), 새 선택기 또는 비즈니스 규칙 그룹의 사본
기존 비즈니스 규칙, 선택기 또는 둘 모두가 있는 이전에 전개된 응용프로그램이 시작됨	성공	서버 이름(적용 가능한 경우 셀과 노드 포함), 선택기 또는 비즈니스 규칙 그룹의 사본
	장애	서버 이름(적용 가능한 경우 셀과 노드 포함), 선택기 또는 비즈니스 규칙 그룹의 사본

## CEI(Common Event Infrastructure) 서버로 서비스 컴포넌트 모니터링

서비스 컴포넌트 모니터링 결과를 CEI(Common Event Infrastructure) 서버에 공개하도록 선택할 수 있습니다. CEI(Common Event Infrastructure) 서버로 모니터링하는 데

사용할 서비스 컴포넌트 이벤트 지점을 응용프로그램 플로우를 보고 관리하는 것에 대해 영구적으로 또는 문제점을 해결하는 것에 대해서는 임시로 지정할 수 있습니다.

WebSphere Process Server 모니터링을 사용하여 CEI(Common Event Infrastructure) 버스에 가동된 서비스 컴포넌트 이벤트 내 서비스 컴포넌트 이벤트 지점의 데이터를 공개할 수 있습니다. 이 모니터링 접근법은 시스템의 서비스 컴포넌트 활동을 보다 유연하게 분석하도록 합니다. 프로세스 서버에 포함된 공통 기본 이벤트 브라우저처럼 CEI 이벤트에 최적화된 브라우저를 사용할 수도 있습니다.

이벤트는 로거에 전송된 이벤트와 동일하게 구조화되지만 서비스 컴포넌트 이벤트를 분석하기 위해 특별히 정의된 표시기로 액세스할 수 있는 데이터베이스에 저장됩니다. 응용프로그램이 서버에 전개되어 실행된 이후 항상 계속해서 모니터링하도록("정적" 모니터링으로 알려진 메소드) 응용프로그램이 작성될 때 응용프로그램 내에 서비스 컴포넌트 이벤트 지점을 지정할 수 있습니다. 시스템에서 컴포넌트 처리의 적절한 플로우에서 특별히 중요한 서비스 컴포넌트 이벤트 지점에 대한 정적 모니터링을 수행합니다. 이 정보로 시스템에서 실행하는 서비스 컴포넌트 프로세스 간의 상호작용 및 전반적인 조치를 쉽게 감독할 수 있습니다. 서비스 컴포넌트가 적절하게 작동 중이 아님을 나타내는 것일 수 있는 이 프로세스의 정상 플로우 이탈을 빠르게 발견할 수도 있습니다.

서비스 컴포넌트의 정적 모니터링을 구성하려면 WebSphere Integration Developer를 사용하여 응용프로그램에서 프로세스 서버에 전개될 서비스 컴포넌트 이벤트 지점을 선택합니다. 선택사항은 응용프로그램과 함께 전개될 확장자가 .mon인 XML 파일 양식으로 지정됩니다. 실행 중인 서버에 전개된 후에는 응용프로그램의 .mon 파일에 지정된 이벤트에 대한 모니터링의 세부사항 레벨을 끄거나 낮출 수 없습니다. 이러한 종류의 모니터링을 중지하려면 서버를 중지하고 응용프로그램을 전개 취소해야 합니다. .mon 파일이 있는 응용프로그램의 작성 및 전개에 대한 자세한 내용은 WebSphere Integration Developer Information Center를 참조하십시오.

실행 중인 서버에 이미 전개된 응용프로그램에서 사용 가능 및 불가능으로 설정할 수 있는 "동적" 모니터링에 대한 서비스 컴포넌트 이벤트 지점을 선택할 수도 있습니다. CEI 서버를 사용한 동적 모니터링 수행 이론은 본질적으로 시스템의 문제점을 진단해서 해결하는 로깅에 대한 이론과 동일합니다. CEI 버스에 가동된 각 이벤트에 대한 구조를 이루는 공통 기본 이벤트 요소와 함께 출력은 본래 로거에 공개된 출력과 동일합니다. 또한 로깅 데이터와 마찬가지로 세부사항 레벨의 차이는 페이로드가 이벤트 내에 인코딩되는 정도에만 영향을 줍니다.

## 관리 콘솔을 사용한 서비스 컴포넌트 이벤트 모니터링 구성

관리 콘솔을 사용하여 CEI(Common Event Infrastructure) 서버에 서비스 컴포넌트 이벤트를 공개할 모니터링 함수를 동적으로 지정하십시오.

시작하기 전에

로거의 경우처럼 진단 추적 서비스를 사용 가능하게 해야 합니다. 서버를 다시 시작한 후 모니터링하려는 이벤트를 한번 호출합니다. 그러면 모니터링할 수 있는 이벤트의 목록이 나타납니다.

### 태스크 정보

모니터할 이벤트를 선택하는 이 메소드는 프로세스 서버에 이미 전개된 응용프로그램에 사용됩니다. 프로세스 서버의 응용프로그램으로 전개되는 .mon 파일에 지정된 이벤트는 여기에서 수행한 변경사항과 무관하게 CEI(Common Event Infrastructure) 데이터베이스에서 모니터링됩니다. 이러한 이벤트의 경우 캡처하여 CEI 데이터베이스에 공개할 보다 높은 레벨의 세부사항만을 지정할 수 있습니다. CEI 데이터베이스에 공개되는 출력은 로거가 공개하는 출력과 매우 유사합니다.

### 프로시저

1. 관리 콘솔에서 문제점 해결 > 로깅 및 추적을 클릭하십시오.
2. 로그 세부사항 레벨 변경을 클릭하십시오.
3. 컴포넌트, 패키지 및 그룹의 목록은 실행 중인 서버에 현재 등록된 모든 컴포넌트를 표시합니다. 적어도 한번은 호출된 프로세스 서버 이벤트만이 이 목록에 나타납니다. 로그될 수 있는 모드 프로세스 서버 이벤트는 **WBILocationMonitor.CEI** 이름으로 시작하는 컴포넌트 중 한 컴포넌트 아래에 나열됩니다.
  - 구성에 대한 정적 변경을 수행하려면 구성 탭을 클릭하십시오.
  - 구성을 동적으로 변경하려면 런타임 탭을 클릭하십시오.
4. 모니터링할 이벤트 또는 이벤트 그룹을 선택하십시오.
5. 각 이벤트에 대해 캡처하려는 세부사항 레벨을 클릭하십시오.

주: FINE, FINER 및 FINEST 레벨만이 CEI 이벤트에 유효합니다.
6. 적용을 클릭한 다음 저장을 클릭하십시오.
7. 확인을 클릭하십시오.
8. 구성을 정적으로 변경한 경우 변경사항을 적용하려면 프로세스 서버를 다시 시작해야 합니다.

### 결과

공통 기본 이벤트 브라우저에서 모니터링한 이벤트 결과를 볼 수 있습니다.

### 학습서: 이벤트 모니터링에 CEI(Common Event Infrastructure) 서버 사용

이 학습서는 CEI(Common Event Infrastructure) 서버에 공개되는 서비스 컴포넌트 이벤트 모니터링을 설정하는 방법 및 공통 기본 이벤트 브라우저에서 저장된 이벤트를 보는 방법에 대한 예제를 제공합니다. 이 시나리오에 사용할 예제는 정적 모니터링을 포

합하지 않으므로 .mon 파일로 전개된 응용프로그램은 계속해서 특정 서비스 컴포넌트 이벤트 지점을 모니터링합니다. 정적 모니터링을 수행하는 방법에 대한 정보는 IBM WebSphere Integration Developer Information Center를 참조하십시오.

대신에 이 예제의 시나리오는 프로세스 서버에서 실행 중이며 이미 전개된 응용프로그램에서 서비스 컴포넌트의 모니터링 이벤트 지점을 선택하는 방법을 보여줍니다. 응용프로그램의 처리가 이 이벤트 지점 중 하나에 도달할 때마다 모니터링 함수가 이벤트를 가동하는 방식을 보게 됩니다. 가동된 각 이벤트는 데이터베이스에 대한 이벤트 정보를 저장하는 CEI 서버에 공개됩니다. 공통 기본 이벤트 브라우저를 사용하여 이벤트를 봅니다.

## 이 학습서의 목표

이 학습서를 완료한 후에는 다음을 수행할 수 있습니다.

- CEI 서버에 이벤트가 공개된 모니터링 서비스 컴포넌트 이벤트 지점을 선택합니다.
- 공통 기본 이벤트 브라우저로 저장된 이벤트를 봅니다.

## 이 학습을 완료하는 데 필요한 시간

이 학습서를 완료하려면 약 15 - 20분이 소요됩니다.

## 전제조건

이 학습서를 수행하려면 다음을 완료했어야 합니다.

- 프로세스 서버를 구성하고 시작했습니다.
- CEI 및 데이터베이스를 구성했습니다.
- 프로세스 서버의 진단 추적 서비스를 사용 가능하게 했습니다.
- 프로세스 서버에 샘플 갤러리 응용프로그램을 설치하고 시작했습니다.
- 프로세스 서버에 비즈니스 규칙 샘플 응용프로그램을 설치하고 시작했습니다. 샘플 갤러리 페이지의 지시사항에 따라 비즈니스 규칙 샘플 응용프로그램을 설정하고 실행하십시오.

이 모든 전제조건이 완료된 후 학습서를 진행하기 전에 샘플 갤러리에서 비즈니스 규칙 샘플 응용프로그램을 적어도 한번은 실행하십시오.

## 예제: CEI(Common Event Infrastructure) 서버로 모니터링: 태스크 정보

이 시나리오의 비즈니스 규칙 샘플 응용프로그램을 사용하므로 웹 페이지에 이 응용프로그램이 이미 열려 있어야 합니다. 모니터링 매개변수를 지정한 후 샘플을 실행해야 하므로 응용프로그램을 계속 열어 놓으십시오. 샘플을 최소 한번은 이미 실행했는지 확인하십시오. 실행한 경우 모니터링하려고 선택할 수 있는 함수 목록에 샘플이 나타납니다.

## 프로시저

1. 관리 콘솔을 여십시오.
2. 탐색 패널에서 서버 > **Application Server**를 클릭하십시오.
3. *server\_name*을 클릭하십시오.
4. 문제점 해결 아래에서 로깅 및 추적을 클릭하십시오.
5. 로그 세부사항 레벨 변경을 클릭하십시오.
6. 런타임 탭을 선택하십시오.
7. **WBILocationMonitor.CEI.BR**에 대한 트리를 펼치십시오. 그러면 **WBILocationMonitor.CEI.BR.brsample.\*** 요소 아래에 7가지 이벤트 유형이 나타납니다. 각 이벤트 유형은 **Operation.\_calculateDiscount** 함수로 추가된 **WBILocationMonitor.CEI.BR.brsample\_module.DiscountRuleGroup** 이름 및 다음 네이처를 포함합니다.
  - ENTRY
  - EXIT
  - FAILURE
  - SelectionKeyExtracted
  - TargetFound
8. 각 이벤트를 클릭하고 **finest**를 선택하십시오.
9. 확인을 클릭하십시오.
10. 비즈니스 규칙 샘플 응용프로그램 페이지를 전환하고 응용프로그램을 한번 실행하십시오.
11. 관리 콘솔로 돌아가서 탐색 패널에서 통합 응용프로그램 > 공통 기본 이벤트 브라우저를 선택하십시오.
12. Network Deployment 환경 내의 노드에서 서버를 실행 중이면 서버 및 노드의 이름을 포함하도록 **Event Data Store** 필드를 수정해야 할 수 있습니다. 다음 양식으로 문자열을 입력하십시오. ‘cell/nodes/node\_name/servers/server\_name/ejb/com/ibm/events/access/EventAccess’.
13. 이벤트 가져오기를 누르십시오.

## 결과

이제 샘플 응용프로그램을 실행했을 때 CEI 서버에 공개된 네 가지 비즈니스 규칙 이벤트의 공통 기본 이벤트 브라우저의 위쪽 분할창에 목록이 표시되어야 합니다. 이벤트 중 하나를 선택하십시오. 그러면 아래쪽 분할창에 이벤트의 콘텐츠가 표시됩니다. 로거에 공개된 이벤트와 이 이벤트를 비교해보십시오. 브라우저는 CEI 서버에 공개된 원래 XML 문자열을 구문 분석했으며 ENTRY 및 EXIT 이벤트의 비즈니스 오브젝트 코드



가 원래의 16진 형식에서 읽기 쉬운 XML로 변환되었음에 유의하십시오. 이 학습을 거슬러 올라가서 세부사항 레벨을 **finest**에서 **fine** 또는 **finer**로 변경한 후 이벤트 간의 차이점을 비교해 볼 수 있습니다.

학습을 완료한 후에는 CEI 서버를 사용하여 모니터할 서비스 컴포넌트 이벤트 지점을 선택하는 방법을 이해해야 합니다. 이 유형의 모니터링에 가동된 이벤트가 표준 형식이며 데이터베이스에 결과가 공개되는 것을 보았습니다. 공통 기본 이벤트 브라우저를 사용하여 데이터베이스에서 이벤트를 검색하고 개별 이벤트에 대한 정보를 브라우저의 구분 분석된 테이블 형식으로 볼 수 있어야 합니다.

#### 다음에 수행할 작업

비즈니스 규칙 샘플 응용프로그램을 더 이상 모니터하지 않으려면 여기에 설명된 단계를 거슬러 올라가서 샘플 이벤트의 세부사항 레벨을 **info**로 재설정할 수 있습니다.

## 세션 모니터링

공통 기본 이벤트 브라우저를 사용하여 동일한 세션 ID 속성을 포함하는 CEI(Common Event Infrastructure) 데이터베이스에서 모든 이벤트를 찾아서 동일한 세션의 일부인 여러 이벤트를 모니터할 수 있습니다.

WebSphere Process Server에는 단일 세션의 일부인 모든 서비스 컴포넌트 이벤트를 식별할 수 있는 향상된 기능이 있습니다. 공통 기본 이벤트의 표준 요소는 contextDataElement 요소 아래에 WBISessionID라는 속성을 포함합니다. 세션의 일부였던 모든 서비스 컴포넌트에 대해 개별 세션의 고유 ID는 이 속성에 저장됩니다. 공통 기본 이벤트 브라우저의 **SessionID** 필드를 사용하여 CEI(Common Event Infrastructure) 데이터베이스에 저장된 이벤트 중 지정한 세션 ID에 일치하는 이벤트를 찾을 수 있습니다. 이 기능을 통해 모든 서비스 컴포넌트 이벤트의 콘텐츠 및 프로세스 플로우를 쉽게 검토할 수 있습니다. 이 정보를 사용하여 응용프로그램의 효율성을 평가하고 특정 상황에서만 발생하는 문제점을 진단할 수 있습니다.

공통 기본 이벤트 브라우저에서 이벤트 및 연관된 콘텐츠의 리턴된 목록을 볼 수 있습니다. 모든 이벤트 보기를 클릭하면 **실패 및 비즈니스 프로세스**라는 두 가지 열이 표시됩니다. 특정 이벤트에 실패 열의 링크가 있는 경우 이 링크를 클릭하여 실패한 이벤트에 대한 내용을 보다 자세히 볼 수 있습니다. 이와 유사하게 특정 이벤트와 연관된 비즈니스 프로세스의 링크가 있으면 이 링크를 클릭하여 비즈니스 프로세스 탐색기를 열고 비즈니스 프로세스 또는 휴먼 타스크 이벤트에 대한 추가 정보를 볼 수 있습니다.





---

## 제 3 장 모니터링한 이벤트 보기

사용 중인 모니터링의 유형에 따라 모니터링한 이벤트의 공개된 결과를 볼 수 있는 여러 방법이 있습니다. 이 섹션은 CEI(Common Event Infrastructure) 데이터베이스에 저장된 성능 데이터, 이벤트 로그 및 서비스 컴포넌트 이벤트를 볼 수 있는 메소드를 제시합니다.

---

### Tivoli Performance Viewer로 성능 매트릭스 보기

이 주제는 Tivoli Performance Viewer를 사용하여 성능 모니터링을 시작 및 중지하고, PMI(Performance Monitoring Infrastructure) 데이터가 시스템에 발생할 때 이를 도표나 테이블 양식으로 보며, 선택적으로 나중에 동일한 표시기에서 검토할 수 있는 파일에 데이터를 로깅할 수 있는 방법을 설명합니다.

#### 시작하기 전에

여기에서는 하나 이상의 서버가 노드에 작성되어 실행 중이고, PMI(Performance Monitoring Infrastructure)가 사용 가능하며, 모니터링하려는 서비스 컴포넌트 이벤트 지점이 표시기 내에서 선택될 수 있도록 적어도 한번은 호출되었다고 가정합니다.

#### 타스크 정보

TPV(Tivoli Performance Viewer)는 프로세스 서버의 성능을 모든 측면에서 다양하고 자세히 볼 수 있게 하는 강력한 응용프로그램입니다. "Tivoli Performance Viewer로 성능 모니터링" 섹션은 이 도구를 다양한 용도로 사용하는 방법에 대한 자세한 내용을 포함하며, 이 프로그램 사용에 대한 전체 지시사항을 보려면 이 자원을 참조해야 합니다. 이 섹션에서는 WebSphere Process Server 특정 이벤트에 대한 성능 데이터를 보는 방법만을 설명합니다.

성능 표시기를 사용하면 관리자와 프로그래머가 WebSphere Process Server의 현재 성능 상태를 모니터링할 수 있습니다. 데이터를 수집하고 보는 것은 프로세스 서버에서 발생하기 때문에 성능에 영향이 미칩니다. 성능 영향을 최소화하려면 활동을 모니터링하려는 서버만을 모니터링하십시오.

주: 이 통계를 볼 때 카운터 유형 통계와 지속 기간 유형 통계를 혼합해선 안됩니다. 카운터는 누적되며 그래프로 표시되는 비율이 응용프로그램에 따라 빠르게 올라갈 수 있습니다. 이와 반대로 지속 기간 통계는 시스템이 각 이벤트를 처리하는 데 걸리는 평균 시간을 나타내기 때문에 일정 범위 내에 머무릅니다. 결과적으로 통계와 상대 배율 간의 불일치로 인해 하나 또는 기타 유형의 통계가 표시기 그래프에서 비뚤어지게 표시될 수 있습니다.

- 현재 성능 활동 보기

1. 관리 콘솔 탐색 트리에서 **모니터링 및 조정 > 성능 표시기 > 현재 활동**을 클릭하십시오.
2. 서버를 선택하고 활동을 모니터링하려는 서버의 이름을 클릭하십시오. 활동을 모니터링하려는 서버에 대한 선택란을 선택하고 **모니터링 시작**을 클릭하는 방법도 있습니다. 동시에 여러 서버에 대한 모니터링을 시작하려면 서버를 선택하고 **모니터링 시작**을 클릭하십시오.
3. 성능 모듈을 선택하십시오.
4. 보려는 각 성능 모듈의 이름 옆에 있는 선택란을 선택하십시오. 성능 통계를 발행하며 최소 한번은 호출된 적이 있는 모든 프로세스 서버 특정 이벤트가 **WBISStats.RootGroup** 계층 구조 아래에 나열됩니다. 노드 옆의 +를 클릭해서 트리를 펼치고 노드 옆의 -를 클릭해서 노드를 접으십시오.
5. 모듈 보기를 클릭하십시오. 요청한 데이터가 제공된 도표나 테이블이 페이지의 오른쪽에 표시됩니다. 도표는 기본적으로 표시됩니다.

각 모듈에는 모듈과 연관된 여러 카운터가 있습니다. 이 카운터는 데이터 도표 또는 테이블 아래의 테이블에 표시됩니다. 선택한 카운터는 도표나 테이블에 표시됩니다. 도표나 테이블 옆의 선택란을 선택하거나 지워서 도표 또는 테이블에서 카운터를 추가하거나 제거할 수 있습니다. 기본적으로 각 모듈의 처음 세 개의 카운터가 표시됩니다.

최대 20개의 카운터를 선택해서 TPV에 현재® 활동 모드로 표시할 수 있습니다.

6. 옵션: 도표나 테이블에서 모듈을 제거하려면 모듈 옆의 선택란을 지우고 모듈 보기를 다시 클릭하십시오.
7. 옵션: 테이블의 데이터를 보려면 카운터 선택 테이블에서 **테이블 보기**를 클릭하십시오. 도표로 다시 토글하려면 **그래프 보기**를 클릭하십시오.
8. 옵션: 도표의 범례를 보려면 **범례 표시**를 클릭하십시오. 범례를 숨기려면 **범례 숨기기**를 클릭하십시오.
9. 이벤트의 성능 모니터링이 완료되면 **Tivoli Performance Viewer**를 클릭하고 모니터 중인 서버를 선택한 다음 **모니터링 중지**를 클릭하십시오.

- 성능 통계 로깅

모니터링이 서버에서 활성화된 동안 현재 사용 가능한 모든 PMI 카운터의 데이터를 로깅하고 결과를 TPV 로그 파일에 기록할 수 있습니다. 최대 20가지 카운터의 여러 조합을 매번 다르게 선택해서 특정 기간에 대한 TPV 로그 파일을 여러 번 볼 수 있습니다. 특정 기간 동안 서버의 여러 다른 성능 측정 간 관계를 유연하게 관찰합니다.

1. 성능 모듈 또는 요약 보고서를 볼 때 **로깅 시작**을 클릭하십시오.

2. 완료되면 로깅 중지를 클릭하십시오. 기본적으로 로그 파일은 서버가 실행 중인 노드의 `profile_root/logs/tpv` 디렉토리에 저장됩니다. TPV는 공간을 보존하기 위해 로그 파일에 쓰기가 완료되면 로그 파일을 자동으로 압축합니다. 이 때 압축된 각 파일에는 단일 로그 파일만이 있어야 하며 압축된 파일과 이름이 동일해야 합니다.
3. 관리 콘솔 탐색 트리에서 **모니터링 및 조정 > 성능 표시기 > 로그 보기**를 클릭해서 로그를 보십시오.

## 서비스 컴포넌트 이벤트 로그 파일 보기 및 해석

이 주제는 서비스 컴포넌트 모니터링으로 생성된 로그 파일의 정보를 해석하는 방법을 논의합니다. 관리 콘솔의 로그 표시기에서 또는 선택한 별도의 텍스트 파일 편집기에서 로그 파일을 볼 수 있습니다.

서비스 컴포넌트 모니터링으로 로거에 가동된 이벤트는 공통 기본 이벤트 형식으로 인코딩됩니다. 이벤트는 로그 파일에 공개될 때 태그가 붙은 XML 형식의 긴 단일 텍스트 행으로 포함되며 아래에 표시된 대로 여기에는 여러 로거 특정 필드도 포함됩니다. 로깅된 이벤트의 공통 기본 이벤트 코딩 해독에 대한 자세한 내용은 이 문서의 이벤트 카탈로그 섹션을 참조해야 합니다. 이 섹션을 사용하여 로그 파일의 각 항목에 포함된 다른 필드 및 로거를 구성할 때 로그 파일에 대해 선택한 형식이 구조화된 방식을 이해하십시오.

### 기본 및 고급 형식 필드

프로세스 서버에서는 로깅 출력을 파일 또는 인메모리 순환 버퍼로 지정할 수 있습니다. 추적 출력이 인메모리 순환 버퍼로 지정된 경우 이 출력을 보려면 파일에 출력을 덤프해야 합니다. 출력은 사용자가 지정한 기본, 고급 또는 로그 분석기 형식의 일반 텍스트로 생성됩니다. 출력의 기본 및 고급 형식은 메시지 로그에 사용 가능한 기본 및 고급 형식과 유사합니다. 기본 및 고급 형식은 여러 동일한 필드 및 형식화 기법을 사용합니다. 이 형식으로 사용할 수 있는 필드는 다음과 같습니다.

#### TimeStamp

시간소인은 형식화되는 프로세스의 로케일을 사용하여 형식화됩니다. 밀리초 단위의 정밀도와 시간대가 있는 완전한 날짜(YYMMDD)를 포함합니다.

#### ThreadId

추적 이벤트를 발행한 스레드의 해시 코드에서 생성된 8자의 16진 값

#### ThreadName

메시지 또는 추적 이벤트를 발행한 Java 스레드의 이름

#### ShortName

추적 이벤트를 발행한 로깅 컴포넌트의 축약 이름. 일반적으로 WebSphere Process Server 내부 컴포넌트의 클래스 이름이지만 사용자 응용프로그램의 일부 다른 ID일 수도 있습니다.

### LongName

추적 이벤트를 발행한 로깅 컴포넌트의 전체 이름. 일반적으로 WebSphere Process Server 내부 컴포넌트의 완전한 클래스 이름이지만 사용자 응용프로그램의 일부 다른 ID일 수도 있습니다.

### EventType

추적 이벤트의 유형을 나타내는 1자 필드. 추적 유형은 소문자입니다. 다음과 같은 값이 가능합니다.

- 1 이벤트 또는 fine 유형의 추적 항목
- 2 finer 유형의 추적 항목
- 3 덤프, 디버그 또는 finest 유형의 추적 항목
- Z 추적 유형이 인식되지 않았음을 나타내는 플래이스홀더

### ClassName

메시지나 추적 이벤트를 발행한 클래스

### MethodName

메시지나 추적 이벤트를 발행한 메소드

### Organization

메시지나 추적 이벤트를 발행한 응용프로그램을 소유하는 조직

### Product

메시지나 추적 이벤트를 발행한 제품

### 구성요소

메시지나 추적 이벤트를 발행한 제품 내의 컴포넌트

## 기본 형식

기본 형식으로 표시되는 추적 이벤트는 다음 형식을 사용합니다.

```
<timestamp><threadId><shortName><eventType>[className][methodName]<textmessage>  
    [parameter 1]  
    [parameter 2]
```

## 고급 형식

고급 형식으로 표시되는 추적 이벤트는 다음 형식을 사용합니다.

```
<timestamp><threadId><eventType><UOW><source=longName>[className][methodName]  
<Organization><Product><Component>[thread=threadName]  
<textMessage>[parameter 1=parameterValue][parameter 2=parameterValue]
```

## 로그 분석기 형식

로그 분석기 형식을 지정하면 WebSphere Application Server에 포함된 응용프로그램 인 로그 분석기 도구를 사용하여 추적 출력을 열 수 있습니다. 이 도구를 통해 로그 분석기의 병합 기능을 사용할 수 있으므로 두 가지 다른 서버 프로세스의 추적을 상관시 키려는 경우에 유용합니다.

---

## 공통 기본 이벤트 브라우저로 이벤트 보기

공통 기본 이벤트 브라우저를 사용하여 이벤트를 선택하고 정렬하며 보십시오.

### 시작하기 전에

이 타스크에서는 사용자가 서버 관리 콘솔에 로그인했다고 가정합니다.

### 타스크 정보

이벤트 브라우저는 이벤트 액세스 인터페이스를 사용하여 이벤트 데이터를 조회합니다. 조회 결과는 브라우저에 표시됩니다.

### 프로시저

1. 이벤트 브라우저를 열어 시작하십시오. 관리 콘솔의 탐색 패널에서 통합 응용프로그램을 클릭한 다음 공통 기본 이벤트 브라우저를 클릭하십시오.
2. 보려는 이벤트를 지정하십시오.
3. 리턴된 이벤트의 보기를 선택하십시오.
4. 임의의 브라우저 패널에서 검색 또는 정렬 기준 선택을 완료한 다음, 브라우저 패널의 맨 아래에 있는 이벤트 가져오기 단추를 클릭하여 이벤트를 표시하십시오.

## 보려는 이벤트 지정

공통 기본 이벤트 브라우저를 사용하여 이벤트 데이터베이스의 이벤트를 조회하기 위한 검색 기준을 지정하는 방법입니다.

### 시작하기 전에

이 타스크는 사용자가 이미 이벤트 브라우저를 열어 이벤트 가져오기 패널을 보고 있다고 가정합니다.

이벤트 데이터 스토어 특성 필드를 채워야 합니다. 이벤트 필터 특성 필드는 선택적이며, 시간, 날짜, 서버 이름, 하위 컴포넌트 이름 및 이벤트 심각도 매개변수에 기반하여 이벤트 검색을 세분화할 수 있습니다.

### 프로시저

1. 필수: 검색할 이벤트 데이터 스토어를 지정하십시오.

이 필드는 관리 콘솔에서 구성될 수 있는 JNDI(Java Naming and Directory Interface) 이름, EJB(Enterprise JavaBeans) 참조입니다. 서버 기본값은 `java:comp/env/eventsaccess`이지만 Network Deployment 환경에서 작업 중인 경우 다른 이름을 지정해야 할 수도 있습니다(JNDI 이름 지정에 대한 자세한 정보는 WebSphere Application Server Network Deployment, 버전 6.1 문서 참조).

2. 필수: 검색할 이벤트 그룹을 지정하십시오.

이는 이벤트가 검색되는 이벤트 그룹입니다. 기본 그룹은 모든 이벤트입니다.

3. 필수: 검색할 이벤트의 수를 지정하십시오.

검색할 최대 이벤트 수는 500입니다.

4. 옵션: 보고서의 작성 날짜(달력 기간)를 지정하십시오.

Enter 및 종료 날짜를 입력하십시오.

5. 옵션: 보고서의 작성 날짜(시간 기간)를 지정하십시오.

시작 및 종료 시간을 입력하십시오.

6. 옵션: 서버 이름을 지정하십시오.

7. 옵션: 하위 컴포넌트 이름을 지정하십시오(적용 가능한 경우).

8. 옵션: 이벤트의 우선순위를 지정하십시오. 검색할 이벤트 우선순위의 범위는 0(최저 우선순위)에서 100(최고 우선순위)입니다.

9. 옵션: 이벤트의 심각도를 지정하십시오.

검색할 이벤트 심각도의 범위는 0(최저 심각도)에서 70(최고 심각도)입니다.

10. 이벤트 가져오기를 클릭하십시오.

## 결과

검색 기준과 일치하는 공통 기본 이벤트 수가 표시됩니다. 조회 결과가 표시되지 않으면 문제점 해결 및 지원 PDF 파일의 "공통 기본 이벤트 브라우저 문제점 해결" 주제를 참조하십시오.

리턴된 이벤트를 보려면 탐색줄에서 보기를 선택하십시오. 모든 이벤트, BPEL 프로세스 이벤트, 사용자 데이터 이벤트 또는 서버 이벤트를 클릭할 수 있습니다. 이벤트 데이터를 보는 경우, 이벤트 가져오기를 클릭하여 검색 기준을 언제든지 변경할 수 있습니다.

## 다음에 수행할 작업

이벤트가 리턴되면 이에 대해 작업하여 다양한 레벨의 이벤트 세부사항을 가져올 수 있습니다.

## 이벤트 브라우저에서 리턴된 이벤트에 대한 작업

이벤트 브라우저를 사용하여 조회에서 리턴된 이벤트를 봅니다.

### 시작하기 전에

이 타스크는 지정된 조회에서 리턴된 데이터에 대해 보려는 이벤트 지정 주제에서 설명된 대로 작업합니다.

## 타스크 정보

조회는 기준에 맞는 모든 이벤트를 리턴합니다.

## 프로시저

1. 탐색줄에서 보기를 클릭하십시오.

탐색줄은 리턴된 조회의 다음 보기를 제공합니다.

### 모든 이벤트

리턴된 모든 이벤트

### BPEL 프로세스 이벤트

특정 프로세스 인스턴스에 대한 Business Process Choreographer 이벤트

### 사용자 데이터 이벤트

확장자 이름이 ECS:UserDataEvent인 이벤트. 이 이벤트 유형은 ECSEmitter 클래스의 addUserDataEvent 메소드로 작성됩니다.

### 서버 이벤트

특정 서버의 이벤트

2. 다음 조치 중 하나를 수행하십시오.

- 1단계에서 **BPEL 프로세스 이벤트**를 클릭한 경우, 프로세스 템플릿을 클릭한 다음 프로세스 인스턴스를 클릭해야 합니다.
- 1단계에서 **서버 이벤트**를 클릭한 경우, 서버를 클릭해야 합니다.

3. 이벤트를 클릭하여 브라우저 창의 맨 아래에 있는 분할창에 이벤트 데이터를 표시하십시오.

## 공통 기본 이벤트 브라우저 문제점 해결

공통 기본 이벤트 브라우저에 액세스할 수 없는 네 가지 기본 조건이 있습니다.

### 조건

#### “서버를 찾을 수 없음”

서버를 사용할 수 없습니다. 이벤트 브라우저 URI를 실행하려고 하면, 서버를 사용할 수 없음을 표시하는 『서버를 찾을 수 없음』 브라우저 페이지가 리턴됩니다. 이 경우 서버 관리자에게 문의하여 문제점의 원인을 확인하십시오.

#### “파일을 찾을 수 없음”

서버를 사용할 수 있지만 이벤트 브라우저 응용프로그램이 서버에서 설치 또는 시작되지 않습니다. 이벤트 브라우저 URI를 실행하려고 하면, 서버를 사용할 수 있지만 해당 서버에서 URI를 사용할 수 없음을 표시하는 『파일을 찾을 수 없음』 브라우저 페이지가 리턴됩니다. 이 경우 서버 관리자에게 문의하여 문제점의 원인을 확인하십시오.

### 로그온 패널 표시

서버 및 이벤트 브라우저를 사용할 수 있지만 이벤트 브라우저에 액세스할 수 있는 올바른 역할로 맵핑되지 않았습니다. 로그인 패널이 프롬프트됩니다. 사용자 ID 및 암호를 입력하여 로그인하려고 시도하면 로그인에 실패합니다. 이 경우 서버 관리자에게 문의하여 이벤트 브라우저를 실행할 수 있는 적합한 권한을 얻어야 합니다.

### “이벤트 데이터 가져오기” 패널의 오류 메시지

서버 및 이벤트 브라우저를 사용할 수 있으며 액세스할 수 있는 적합한 권한이 있지만, CEI(Common Event Infrastructure) 서버를 사용할 수 없습니다. 이벤트 가져오기 단추를 누르면 이벤트 브라우저의 이벤트 가져오기 패널에 오류 메시지가 표시됩니다. 메시지 로그에 오류 정보가 로그됩니다.



## 제 4 장 이벤트 카탈로그

이벤트 카탈로그는 각 이벤트가 생성하는 연관된 공통 기본 이벤트 확장 데이터 요소 뿐만 아니라 각 서비스 컴포넌트 유형에 대해 모니터링할 수 있는 모든 이벤트의 스펙을 포함합니다. 이 섹션에 제시된 정보를 참조 자료로 사용하여 개별 이벤트가 구조화된 방식을 이해하십시오. 각 이벤트가 생성하는 상대적인 대용량 데이터에서 필요한 정보를 빠르게 식별할 수 있도록 각 이벤트에 포함된 정보를 해독하는 데 도움이 될 것입니다. 이 섹션은 공통 기본 이벤트의 구조 및 표준 요소, Business Process Choreographer 서비스 컴포넌트 및 WebSphere Process Server 특정 서비스 컴포넌트에 대한 이벤트 목록과 각 이벤트 유형에 고유한 공통 기본 이벤트에 대한 확장을 다룹니다. 서비스 컴포넌트가 처리할 수 있는 비즈니스 오브젝트가 서비스 컴포넌트 이벤트에 캡처되는 방식도 논의합니다.

제공된 유형의 이벤트는 CEI(Common Event Infrastructure) 버스에서 CEI 서버 또는 로거로 가동될 때 본래 이벤트 카탈로그 스펙에 따라 작성된 이벤트 요소의 XML 캡슐화인 공통 기본 이벤트의 양식을 사용합니다. 공통 기본 이벤트는 표준 요소 세트, 프로세스 서버 컴포넌트 식별 요소, 이벤트 상관 스피어 ID 및 각 이벤트 유형에 고유한 추가 요소를 포함합니다. 이 모든 요소는 한 가지 경우를 제외하면 서비스 컴포넌트 모니터에서 이벤트가 가동될 때마다 CEI 서버 또는 로거에 전달됩니다. 이벤트가 페이로드 내에 비즈니스 오브젝트 코드를 포함하는 경우에는 이벤트에 포함하려는 비즈니스 오브젝트 데이터의 용량을 지정할 수 있습니다.

### 공통 기본 이벤트 표준 요소

서비스 컴포넌트 모니터링에서 가동된 모든 이벤트에 포함되는 공통 기본 이벤트의 요소가 여기에 나열됩니다.

속성	설명
version	1.0.1로 설정됩니다.
creationTime	이벤트가 UTC로 작성된 시간
globalInstanceId	공통 기본 이벤트 인스턴스의 ID. 이 ID는 자동으로 생성됩니다.
localInstanceId	이 ID는 자동으로 생성됩니다(공백일 수 있음).
severity	이벤트가 비즈니스 프로세스 또는 휴먼 태스크에 미치는 영향. 이 속성은 10(정보)으로 설정됩니다. 그렇지 않으면 사용되지 않습니다.
priority	사용되지 않음
reporterComponentId	사용되지 않음
locationType	Hostname으로 설정하십시오.
location	실행 중인 서버의 호스트 이름으로 설정됩니다.
application	사용되지 않음

속성	설명
executionEnvironment	운영 체제를 식별하는 문자열
구성요소	프로세스 서버 버전. 비즈니스 프로세스 및 휴먼 태스크의 경우: WPS# 다음 현재 플랫폼 ID 및 기본 소프트웨어 스택 버전
componentType	Apache QName 형식에 기반한 컴포넌트 QName  비즈니스 프로세스:  www.ibm.com/namespaces/autonomic/Workflow_Engine  휴먼 태스크:  www.ibm.com/xmlns/prod/websphere/scdl/human-task
subComponent	관찰 가능한 요소 이름  비즈니스 프로세스의 경우 BFM으로 설정되며, 휴먼 태스크의 경우 HTM으로 설정됩니다.
componentIdType	ProductName으로 설정하십시오.
instanceId	서버 ID. 이 ID의 형식은 <i>cell_name/node_name/server_name</i> 입니다. 분리문자는 운영 체제에 따라 다릅니다.
processId	운영 체제의 프로세스 ID
threadId	JVM(Java virtual machine)의 스레드 ID.
Situation Type	이벤트의 보고를 유발한 상황의 유형. 프로세스 서버 특정 컴포넌트의 경우 ReportSituation으로 설정하십시오.
Situation Category	이벤트의 보고를 유발한 상황 유형의 카테고리. 프로세스 서버 특정 컴포넌트의 경우 STATUS로 설정하십시오.
Situation Reasoning Scope	보고된 상황의 영향 범위. 프로세스 서버 특정 컴포넌트의 경우 EXTERNAL로 설정하십시오.
ECSCurrentID	현재 이벤트 상관 스피어 ID의 값
ECSParentID	상위 이벤트 상관 스피어 ID의 값
WBISessionID	현재 세션 ID의 값
extensionName	이벤트 이름으로 설정하십시오.

## 이벤트의 비즈니스 오브젝트

비즈니스 오브젝트 데이터는 버전 6.1에서부터 이벤트 내에서 XML 형식으로 전달됩니다. 공통 기본 이벤트 형식은 비즈니스 오브젝트 페이로드를 XML 요소에 캡슐화하는 `xs:any` 스키마를 포함합니다.

서비스 컴포넌트 이벤트에 캡처할 비즈니스 오브젝트 세부사항의 레벨을 지정합니다. 이 세부사항 레벨은 이벤트에 전달될 비즈니스 오브젝트 코드의 양에만 영향을 미칩니다. 다른 모든 공통 기본 이벤트 요소(표준 및 이벤트 특정 모두)는 이벤트에 공개됩니다. 서비스 컴포넌트 이벤트에 적용 가능한 세부사항 레벨의 이름은 WebSphere Integration Developer를 사용하여 정적 모니터를 작성했는지 또는 관리 콘솔에서 동적 모니터를 작성했는지 여부에 따라 다르며 아래 테이블에 표시됩니다.

관리 콘솔 세부사항 레벨	공통 기본 이벤트/WebSphere Integration Developer 세부사항 레벨	공개된 페이로드 정보
FINE	EMPTY	없음
FINER	DIGEST	페이로드 설명만
FINEST	FULL	모든 페이로드

세부사항 레벨은 이벤트 인스턴스 데이터의 일부인 PayloadType 요소로 지정됩니다. 실제 비즈니스 오브젝트 데이터는 모니터가 레코드 FULL/FINEST 세부사항으로 설정된 경우에만 이벤트에 포함됩니다. 비즈니스 오브젝트 데이터 자체는 xsd:any 스키마 아래의 공통 기본 이벤트에 포함됩니다. 실제로는 루트 요소가 wbi:event인 프로세스 서버 비즈니스 오브젝트 페이로드를 보게 됩니다. 이벤트 출력을 로거에 공개 중이면 로그 파일을 볼 때 출력이 표시됩니다. 이벤트가 CEI 서버에 공개된 경우에는 공통 기본 이벤트 브라우저를 사용하여 이벤트를 볼 수 있습니다. 그런 다음 wbi:event 링크를 클릭해서 비즈니스 오브젝트 데이터를 볼 수 있습니다.

## Business Process Choreographer 이벤트

WebSphere Process Server는 비즈니스 프로세스 및 휴먼 타스크에 대한 Business Process Choreographer 서비스 컴포넌트를 통합합니다. 이 컴포넌트에서 모니터링할 수 있는 이벤트 지점을 이 섹션에서 설명합니다.

### 비즈니스 프로세스 이벤트 모니터링

비즈니스 프로세스를 대신해서 생성되는 이벤트는 상황 독립적인 데이터와 비즈니스 프로세스 이벤트에 특정한 데이터로 구성됩니다. 비즈니스 프로세스 이벤트에 특정한 속성 및 요소를 기술합니다.

비즈니스 프로세스 이벤트에는 다음과 같은 범주의 이벤트 콘텐츠가 있을 수 있습니다.

#### 비즈니스 프로세스 특정 이벤트 데이터

비즈니스 프로세스에서 이벤트는 프로세스, 활동, 범위, 링크 및 변수와 관련됩니다. 각각의 해당 이벤트 유형의 오브젝트 특정 콘텐츠를 설명합니다.

Business Process Choreographer 버전 6.1에는 두 가지 이벤트 형식이 발생할 수 있습니다.

#### WebSphere Business Monitor 6.0.2 형식

WebSphere Business Monitor 6.0.2 형식 이벤트는 WebSphere Integration Developer 6.0.2 내에 모델화된 프로세스가 있을 때나 WebSphere Business Monitor 6.0.2 형식 모드가 WebSphere Integration Developer 6.1 내에서 사용 가능할 때 발생합니다. 다르게 지정하지 않으면 이 이벤트에 대한 오브젝트 특정 콘텐츠가 유형 문자열의 *extendedDataElement* XML 요소로 기록됩니다.

## WebSphere Business Monitor 6.1 형식

WebSphere Business Monitor 6.1 형식 이벤트는 WebSphere Integration Developer 6.1 내에 모델화된 프로세스가 있고 WebSphere Business Monitor 6.1 형식 모드가 사용 가능할 때 발생합니다. 이 이벤트에 대한 오브젝트 특정 콘텐츠는 공통 기본 이벤트의 eventPointData 폴더에 있는 xs:any 슬롯의 XML 요소로 기록되고 페이로드 메시지는 *applicationData* 섹션에 기록됩니다. XML의 구조는 XSD(XML Schema Definition) 파일 BFMEvents.xsd에 정의됩니다. 이 파일은 *install\_root#ProcessChoreographer#client* 디렉토리에서 찾을 수 있습니다.

## 프로세스

프로세스 인스턴스 이벤트는 다음과 같은 오브젝트 특정 이벤트 콘텐츠를 갖습니다.

속성	설명
processTemplateName	인스턴스가 유래된 프로세스 템플릿의 이름
processTemplateValidFrom	템플릿이 유효하기 시작한 날짜
processTemplateId	프로세스 템플릿의 ID
processInstanceDescription	선택적: 프로세스 인스턴스 설명
processInstanceExecutionState	프로세스 상태를 나타내는 문자열 값. 형식: <i>state number-state description</i> . 이 속성은 다음 값 중 하나를 가집니다.  1 - STATE_READY 2 - STATE_RUNNING 3 - STATE_FINISHED 4 - STATE_COMPENSATING 5 - STATE_FAILED 6 - STATE_TERMINATED 7 - STATE_COMPENSATED 8 - STATE_TERMINATING 9 - STATE_FAILING 10 - STATE_INDOUBT 11 - STATE_SUSPENDED 12 - STATE_COMPENSATION_FAILED
principal	현재 실행 단계가 실행되는 프린시펄. 일반적으로 프로세스의 시작기입니다.
PayloadType	페이로드 유형. 문자열 값은 none, digest 또는 full 중 하나일 수 있습니다. 값은 WebSphere Integration Developer의 설정에 따라 다릅니다. 사용 가능한 명시 모니터링 정의가 없으면(예를 들어, 기본 이벤트 사용을 선택한 경우) 기본 페이로드 유형은 full입니다.

## 활동 및 범위

활동 및 범위는 다음과 같은 오브젝트 특정 이벤트 콘텐츠를 갖습니다.

속성	설명
processTemplateName	인스턴스가 유래된 프로세스 템플릿의 이름
processTemplateValidFrom	템플릿이 유효하기 시작한 날짜
activityTemplateName	선택적: 인스턴스가 유래된 활동 템플릿의 이름
activityInstanceDescription	선택적: 활동 인스턴스에 대한 설명
activityKind	<p>활동 종류를 식별하는 문자열 값. 형식: <i>kind number-kind description</i>. 이 속성은 다음 값 중 하나를 가집니다.</p> <ul style="list-style-type: none"> <li>3 - KIND_EMPTY</li> <li>21 - KIND_INVOKE</li> <li>23 - KIND_RECEIVE</li> <li>24 - KIND_REPLY</li> <li>25 - KIND_THROW</li> <li>26 - KIND_TERMINATE</li> <li>27 - KIND_WAIT</li> <li>29 - KIND_COMPENSATE</li> <li>30 - KIND_SEQUENCE</li> <li>32 - KIND_SWITCH</li> <li>34 - KIND_WHILE</li> <li>36 - KIND_PICK</li> <li>38 - KIND_FLOW</li> <li>40 - KIND_SCOPE</li> <li>42 - KIND_SCRIPT</li> <li>43 - KIND_STAFF</li> <li>44 - KIND_ASSIGN</li> <li>45 - KIND_CUSTOM</li> <li>46 - KIND_RETHROW</li> <li>47 - KIND_FOR_EACH_SERIAL</li> <li>48 - KIND_FOR_EACH_PARALLEL</li> <li>1000 - SQLSnippet</li> <li>1001 - RetrieveSet</li> <li>1002 - InvokeInformationService</li> <li>1003 - AtomicSQLSnippetSequence</li> </ul>

속성	설명
상태	<p>활동 상태를 나타내는 문자열 값. 형식: <i>state number-state description</i>. 활동의 상태 코드는 프로세스에 사용된 상태 코드와 다릅니다. 이 속성은 다음 값 중 하나를 가집니다.</p> <p>1 - STATE_INACTIVE  2 - STATE_READY  3 - STATE_RUNNING  4 - STATE_SKIPPED  5 - STATE_FINISHED  6 - STATE_FAILED  7 - STATE_TERMINATED  8 - STATE_CLAIMED  9 - STATE_TERMINATING  10 - STATE_FAILING  11 - STATE_WAITING  12 - STATE_EXPIRED  13 - STATE_STOPPED</p>
bpelId	활동의 wpc:id 속성을 나타내는 문자열 값
PayloadType	페이로드 유형. 문자열 값은 none, digest 또는 full 중 하나일 수 있습니다. 값은 WebSphere Integration Developer의 설정에 따라 다릅니다. 사용 가능한 명시 모니터링 정의가 없으면(예를 들어, 기본 이벤트 사용을 선택한 경우) 기본 페이로드 유형은 full입니다.

## 링크

링크는 다음과 같은 오브젝트 특정 이벤트 콘텐츠를 갖습니다.

속성	설명
processTemplateName	인스턴스가 유래된 프로세스 템플릿의 이름
processTemplateValidFrom	템플릿이 유효하기 시작한 날짜
flowBpelId	링크를 포함하는 플로우 활동의 wpc:id 속성을 나타내는 문자열 값
elementName	평가된 링크의 이름
description	링크에 대한 설명. 이 속성은 프로세스 모델에서 지정된 경우에만 포함됩니다.
PayloadType	페이로드 유형. 문자열 값은 none, digest 또는 full 중 하나일 수 있습니다. 값은 WebSphere Integration Developer의 설정에 따라 다릅니다. 사용 가능한 명시 모니터링 정의가 없으면(예를 들어, 기본 이벤트 사용을 선택한 경우) 기본 페이로드 유형은 full입니다.

## 변수

변수는 다음과 같은 오브젝트 특정 이벤트 콘텐츠를 갖습니다.

속성	설명
processTemplateName	인스턴스가 유래된 프로세스 템플릿의 이름

속성	설명
processTemplateValidFrom	템플릿이 유효하기 시작한 날짜
variableName	변경된 변수 이름
variableData	<p>WBI 모니터 호환 가능 이벤트 요청 시 생성됩니다. 변수의 콘텐츠에 대한 XML 표시. 데이터 오브젝트의 각 특성은 중첩 확장 데이터 요소 양식으로 보고됩니다. 요소 유형은 적절한 값을 갖는 'boolean' 또는 'string' 유형 중 하나일 수 있습니다. <i>variableName</i> 변수가 초기화되지 않으면 <i>variableData</i> 요소는 없습니다.</p> <p>이 속성은 WebSphere Business Monitor 6.0.2 형식 이벤트에만 사용됩니다. WebSphere Business Monitor 6.1 형식 이벤트의 경우에는 변수의 이름으로 설정된 이름이 있는 하나의 콘텐츠 요소를 포함하는 <i>applicationData</i> 섹션에 변수의 콘텐츠가 기록됩니다.</p>
variableData_BO	<p>비WBI 모니터 호환 가능 이벤트 요청 시 생성됩니다. 이 요소는 'noValue' 유형이고 변수 내용에 대한 XML 표시를 포함합니다. 데이터 오브젝트의 각 특성은 중첩 확장 데이터 요소 양식으로 보고됩니다. <i>variableName</i> 변수가 초기화되지 않으면 <i>VariableData_BO</i> 요소는 없습니다.</p> <p>이 속성은 WebSphere Business Monitor 6.0.2 형식 이벤트에만 사용됩니다. WebSphere Business Monitor 6.1 형식 이벤트의 경우에는 변수의 이름으로 설정된 이름이 있는 하나의 콘텐츠 요소를 포함하는 <i>applicationData</i> 섹션에 변수의 콘텐츠가 기록됩니다.</p>
bpelId	활동의 wpc:id 속성을 나타내는 문자열 값
PayloadType	페이로드 유형. 문자열 값은 none, digest 또는 full 중 하나일 수 있습니다. 값은 WebSphere Integration Developer의 설정에 따라 다릅니다. 사용 가능한 명시 모니터링 정의가 없으면(예를 들어, 기본 이벤트 사용을 선택한 경우) 기본 페이로드 유형은 full입니다.

## 비즈니스 프로세스 이벤트의 확장자 이름

확장자 이름은 이벤트의 페이로드를 나타냅니다. 비즈니스 프로세스 이벤트 및 해당 페이로드에 대한 모든 확장 이름의 목록을 여기에서 찾을 수 있습니다.

확장자 이름은 공통 기본 이벤트의 *extensionName* 속성 값으로 사용되는 문자열 값을 포함합니다. 이 이름은 이벤트에 대한 추가 데이터를 제공하는 XML 요소의 이름이기도 합니다. 이벤트 요소의 이름은 대문자이며(예: BPC.BFM.BASE) XML 요소의 이름은 대소문자가 혼합됩니다(예: BPCEventCode). 표시한 경우를 제외하고 모든 데이터 요소가 string 유형입니다.

다음 확장자 이름이 비즈니스 프로세스 이벤트에 사용 가능합니다.

- 58 페이지의 『BPC.BFM.BASE』
- 59 페이지의 『BPC.BFM.PROCESS.BASE』
- 59 페이지의 『BPC.BFM.PROCESS.STATUS』
- 59 페이지의 『BPC.BFM.PROCESS.START』
- 60 페이지의 『BPC.BFM.PROCESS.FAILURE』

- 60 페이지의 『BPC.BFM.PROCESS.CORREL』
- 60 페이지의 『BPC.BFM.PROCESS.WISTATUS』
- 60 페이지의 『BPC.BFM.PROCESS.WITRANSFER』
- 61 페이지의 『BPC.BFM.PROCESS.ESCALATED』
- 61 페이지의 『BPC.BFM.PROCESS.EVENT』
- 61 페이지의 『BPC.BFM.PROCESS.PARTNER』
- 62 페이지의 『BPC.BFM.ACTIVITY.BASE』
- 63 페이지의 『BPC.BFM.ACTIVITY.STATUS』
- 63 페이지의 『BPC.BFM.ACTIVITY.FAILURE』
- 64 페이지의 『BPC.BFM.ACTIVITY.MESSAGE』
- 64 페이지의 『BPC.BFM.ACTIVITY.CLAIM』
- 64 페이지의 『BPC.BFM.ACTIVITY.WISTATUS』
- 64 페이지의 『BPC.BFM.ACTIVITY.WITRANSFER』
- 65 페이지의 『BPC.BFM.ACTIVITY.FOREACH』
- 65 페이지의 『BPC.BFM.ACTIVITY.ESCALATED』
- 65 페이지의 『BPC.BFM.ACTIVITY.EVENT』
- 65 페이지의 『BPC.BFM.LINK.STATUS』
- 66 페이지의 『BPC.BFM.VARIABLE.STATUS』

## BPC.BFM.BASE

BPC.BFM.BASE는 WBIMonitoringEvent로부터 XML 요소를 상속합니다.

표 21. BPC.BFM.BASE의 XML 요소

XML 요소	설명
<i>BPCEventCode</i>	이벤트 네이처를 식별하는 Business Process Choreographer 이벤트 코드
<i>processTemplateName</i>	프로세스 템플릿의 이름. 이 이름은 표시 이름과 다를 수 있습니다.
<i>processTemplateValidFrom</i>	프로세스 템플릿의 유효한 from 속성
<i>eventLocalCounter</i>	로컬 카운터는 동일한 트랜잭션에서 발생하는 두 이벤트의 순서를 찾아내는 데 사용됩니다. 마이크로플로우 인스턴스의 경우 이 카운터는 발행된 모든 이벤트의 순서를 재구성합니다. 장기 실행 프로세스의 경우 로컬 카운터는 현재 탐색 트랜잭션의 순서를 나타냅니다.



## BPC.BFM.PROCESS.BASE

BPC.BFM.PROCESS.STATUS는 58 페이지의 『BPC.BFM.BASE』에서 XML 요소를 상속합니다.

표 22. BPC.BFM.PROCESS.BASE의 XML 요소

XML 요소	설명
<i>processInstanceExecutionState</i>	<p>다음 형식으로 된 프로세스의 현재 실행 상태. &lt;state code&gt;-&lt;state name&gt;. 이 속성은 다음 값 중 하나를 가집니다.</p> <ul style="list-style-type: none"> <li>1 - STATE_READY</li> <li>2 - STATE_RUNNING</li> <li>3 - STATE_FINISHED</li> <li>4 - STATE_COMPENSATING</li> <li>5 - STATE_FAILED</li> <li>6 - STATE_TERMINATED</li> <li>7 - STATE_COMPENSATED</li> <li>8 - STATE_TERMINATING</li> <li>9 - STATE_FAILING</li> <li>10 - STATE_INDOUBT</li> <li>11 - STATE_SUSPENDED</li> <li>12 - STATE_COMPENSATION_FAILED</li> </ul>

## BPC.BFM.PROCESS.STATUS

BPC.BFM.PROCESS.STATUS는 『BPC.BFM.PROCESS.BASE』에서 XML 요소를 상속합니다.

표 23. BPC.BFM.PROCESS.STATUS의 XML 요소

XML 요소	설명
<i>processTemplateId</i>	프로세스 템플릿 ID
<i>processInstanceDescription</i>	프로세스 인스턴스 설명
<i>principal</i>	이 이벤트와 연관된 사용자의 이름

## BPC.BFM.PROCESS.START

BPC.BFM.PROCESS.START는 『BPC.BFM.PROCESS.STATUS』에서 XML 요소를 상속합니다.

표 24. BPC.BFM.PROCESS.START의 XML 요소

XML 요소	설명
<i>username</i>	BPC.BFM.PROCESS.START의 경우에는 프로세스의 시작 또는 재시작을 요청한 사용자의 이름

## BPC.BFM.PROCESS.FAILURE

BPC.BFM.PROCESS.FAILURE는 59 페이지의 『BPC.BFM.PROCESS.STATUS』에서 XML 요소를 상속합니다.

표 25. BPC.BFM.PROCESS.FAILURE의 XML 요소

XML 요소	설명
<i>processFailedException</i>	프로세스 실패를 리드하는 예외 메시지

## BPC.BFM.PROCESS.CORREL

BPC.BFM.PROCESS.CORREL은 59 페이지의 『BPC.BFM.PROCESS.STATUS』에서 XML 요소를 상속합니다.

표 26. BPC.BFM.PROCESS.CORREL의 XML 요소

XML 요소	설명
<i>correlationSet</i>	다음 형식의 상관 세트 인스턴스: <pre>&lt;?xml version="1.0"?&gt; &lt;correlationSet name="correlation set name"&gt;   &lt;property name="property name"     value="property value"/&gt;* &lt;/correlationSet&gt;</pre>

## BPC.BFM.PROCESS.WISTATUS

BPC.BFM.PROCESS.WISTATUS는 59 페이지의 『BPC.BFM.PROCESS.STATUS』에서 XML 요소를 상속합니다.

표 27. BPC.BFM.PROCESS.WISTATUS의 XML 요소

XML 요소	설명
<i>username</i>	BPC.BFM.PROCESS.WISTATUS의 경우에는 작성되거나 삭제된 작업 항목이 있는 사용자 목록

## BPC.BFM.PROCESS.WITRANSFER

BPC.BFM.PROCESS.WITRANSFER은 59 페이지의 『BPC.BFM.PROCESS.STATUS』에서 XML 요소를 상속합니다.

표 28. BPC.BFM.PROCESS.WITRANSFER의 XML 요소

XML 요소	설명
<i>current</i>	작업 항목의 현재 소유자의 사용자 이름. 작업 항목이 누군가에게 전송된 사용자입니다.
<i>target</i>	작업 항목의 새 소유자의 사용자 이름

## BPC.BFM.PROCESS.ESCALATED

BPC.BFM.PROCESS.ESCALATED는 59 페이지의 『BPC.BFM.PROCESS.STATUS』에서 XML 요소를 상속합니다.

표 29. BPC.BFM.PROCESS.ESCALATED의 XML 요소

XML 요소	설명
<i>escalationName</i>	에스컬레이션 이름
<i>operation</i>	BPC.BFM.PROCESS.ESCALATED의 경우 인라인 호출 타스크가 에스컬레이트된 이벤트 핸들러와 연관된 조작
<i>portTypeName</i>	인라인 호출 타스크가 에스컬레이트된 이벤트 핸들러와 연관된 조작의 포트 유형 이름
<i>portTypeNamespace</i>	인라인 호출 타스크가 에스컬레이트된 이벤트 핸들러와 연관된 조작의 포트 유형 네임 스페이스

## BPC.BFM.PROCESS.EVENT

BPC.BFM.PROCESS.EVENT는 59 페이지의 『BPC.BFM.PROCESS.STATUS』에서 XML 요소를 상속합니다.

표 30. BPC.BFM.PROCESS.EVENT의 XML 요소

XML 요소	설명
<i>message</i> 또는 <i>message_BO</i>	문자열 또는 비즈니스 오브젝트(BO) 표시로 서비스에 대한 입력 또는 출력 메시지. WebSphere Integration Developer의 이벤트 모니터 탭에서 호환 가능 이벤트 모니터 옵션을 선택했는지 여부에 따라 형식이 다릅니다.  이 속성은 WebSphere Business Monitor 6.0.2 형식 이벤트에만 사용됩니다. WebSphere Business Monitor 6.1 형식 이벤트의 경우에는 메시지의 이름으로 설정된 이름이 있는 하나의 콘텐츠 요소를 포함하는 <i>applicationData</i> 섹션에 메시지의 콘텐츠가 기록됩니다.
<i>operation</i>	수신된 이벤트에 대한 조작의 이름
<i>portTypeName</i>	이벤트 핸들러와 연관된 조작의 포트 유형 이름
<i>portTypeNamespace</i>	이벤트 핸들러와 연관된 조작의 포트 유형 네임 스페이스

## BPC.BFM.PROCESS.PARTNER

BPC.BFM.PROCESS.PARTNER는 59 페이지의 『BPC.BFM.PROCESS.STATUS』에서 XML 요소를 상속합니다.

표 31. BPC.BFM.PROCESS.PARTNER의 XML 요소

XML 요소	설명
<i>partnerLinkName</i>	상대 링크의 이름

## BPC.BFM.ACTIVITY.BASE

BPC.BFM.ACTIVITY.BASE는 58 페이지의 『BPC.BFM.BASE』에서 XML 요소를 상속합니다.

표 32. BPC.BFM.ACTIVITY.BASE의 XML 요소

XML 요소	설명
<i>activityKind</i>	<p>활동 종류(예: 순서 또는 호출). 형식은 &lt;kind code&gt;-&lt;kind name&gt;입니다. 이 속성은 다음 값 중 하나를 가집니다.</p> <p>3 - KIND_EMPTY                  21 - KIND_INVOKE                  23 - KIND_RECEIVE                  24 - KIND_REPLY                  25 - KIND_THROW                  26 - KIND_TERMINATE                  27 - KIND_WAIT                  29 - KIND_COMPENSATE                  30 - KIND_SEQUENCE                  32 - KIND_SWITCH                  34 - KIND_WHILE                  36 - KIND_PICK                  38 - KIND_FLOW                  40 - KIND_SCOPE                  42 - KIND_SCRIPT                  43 - KIND_STAFF                  44 - KIND_ASSIGN                  45 - KIND_CUSTOM                  46 - KIND_RETHROW                  47 - KIND_FOR_EACH_SERIAL                  48 - KIND_FOR_EACH_PARALLEL                  1000 - SQLSnippet                  1001 - RetrieveSet                  1002 - InvokeInformationService                  1003 - AtomicSQLSnippetSequence</p>

표 32. BPC.BFM.ACTIVITY.BASE의 XML 요소 (계속)

XML 요소	설명
<i>state</i>	<p>다음 형식의 활동 인스턴스에 대한 현재 상태: &lt;state code&gt;-&lt;state name&gt;이 속성은 다음 값 중 하나를 가집니다.</p> <p>1 - STATE_INACTIVE                  2 - STATE_READY                  3 - STATE_RUNNING                  4 - STATE_SKIPPED                  5 - STATE_FINISHED                  6 - STATE_FAILED                  7 - STATE_TERMINATED                  8 - STATE_CLAIMED                  9 - STATE_TERMINATING                  10 - STATE_FAILING                  11 - STATE_WAITING                  12 - STATE_EXPIRED                  13 - STATE_STOPPED</p>
<i>bpelId</i>	BPEL 파일에 있는 활동의 wpc:id 속성. 프로세스 모델 내부 활동에 고유합니다.

## BPC.BFM.ACTIVITY.STATUS

BPC.BFM.ACTIVITY.STATUS는 62 페이지의 『BPC.BFM.ACTIVITY.BASE』에서 XML 요소를 상속합니다.

표 33. BPC.BFM.ACTIVITY.STATUS의 XML 요소

XML 요소	설명
<i>activityTemplateName</i>	활동 템플릿의 이름. 표시 이름과 다를 수 있습니다.
<i>activityTemplateId</i>	프로세스 템플릿의 내부 ID
<i>activityInstanceDescription</i>	활동 인스턴스에 대한 설명
<i>principal</i>	활동을 청구한 사용자의 이름

## BPC.BFM.ACTIVITY.FAILURE

BPC.BFM.ACTIVITY.FAILURE는 『BPC.BFM.ACTIVITY.STATUS』에서 XML 요소를 상속합니다.

표 34. BPC.BFM.ACTIVITY.FAILURE의 XML 요소

XML 요소	설명
<i>activityFailedException</i>	활동이 실패할 때 발생한 예외

## BPC.BFM.ACTIVITY.MESSAGE

BPC.BFM.ACTIVITY.MESSAGE는 63 페이지의 『BPC.BFM.ACTIVITY.STATUS』에서 XML 요소를 상속합니다.

표 35. BPC.BFM.ACTIVITY.MESSAGE의 XML 요소

XML 요소	설명
<i>message</i> 또는 <i>message_BO</i>	문자열 또는 비즈니스 오브젝트(BO) 표시로 서비스에 대한 입력 또는 출력 메시지. WebSphere Integration Developer의 이벤트 모니터 탭에서 호환 가능 이벤트 모니터 옵션을 선택했는지 여부에 따라 형식이 다릅니다.  이 속성은 WebSphere Business Monitor 6.0.2 형식 이벤트에만 사용됩니다. WebSphere Business Monitor 6.1 형식 이벤트의 경우에는 메시지의 이름으로 설정된 이름이 있는 하나의 콘텐츠 요소를 포함하는 <i>applicationData</i> 섹션에 메시지의 콘텐츠가 기록됩니다.

## BPC.BFM.ACTIVITY.CLAIM

BPC.BFM.ACTIVITY.CLAIM은 63 페이지의 『BPC.BFM.ACTIVITY.STATUS』에서 XML 요소를 상속합니다.

표 36. BPC.BFM.ACTIVITY.CLAIM의 XML 요소

XML 요소	설명
<i>username</i>	BPC.BFM.ACTIVITY.CLAIM의 경우 타스크를 청구한 사용자

## BPC.BFM.ACTIVITY.WISTATUS

BPC.BFM.ACTIVITY.WISTATUS는 63 페이지의 『BPC.BFM.ACTIVITY.STATUS』에서 XML 요소를 상속합니다.

표 37. BPC.BFM.ACTIVITY.WISTATUS의 XML 요소

XML 요소	설명
<i>username</i>	BPC.BFM.ACTIVITY.WISTATUS의 경우 작업 항목과 연관된 사용자 목록

## BPC.BFM.ACTIVITY.WITRANSFER

BPC.BFM.ACTIVITY.WITRANSFER는 63 페이지의 『BPC.BFM.ACTIVITY.STATUS』에서 XML 요소를 상속합니다.

표 38. BPC.BFM.ACTIVITY.WITRANSFER의 XML 요소

XML 요소	설명
<i>current</i>	작업 항목의 현재 소유자의 사용자 이름. 작업 항목이 누군가에게 전송된 사용자입니다.
<i>target</i>	작업 항목의 새 소유자의 사용자 이름

## BPC.BFM.ACTIVITY.FOREACH

BPC.BFM.ACTIVITY.FOREACH는 63 페이지의 『BPC.BFM.ACTIVITY.STATUS』에서 XML 요소를 상속합니다.

표 39. BPC.BFM.ACTIVITY.FOREACH의 XML 요소

XML 요소	설명
<i>parallelBranchesStarted</i>	시작된 분기의 수

## BPC.BFM.ACTIVITY.ESCALATED

BPC.BFM.ACTIVITY.ESCALATED는 63 페이지의 『BPC.BFM.ACTIVITY.STATUS』에서 XML 요소를 상속합니다.

표 40. BPC.BFM.ACTIVITY.ESCALATEDFOREACH의 XML 요소

XML 요소	설명
<i>escalationName</i>	에스컬레이션 이름
<i>operation</i>	BPC.BFM.PROCESS.ESCALATED의 경우 인라인 호출 태스크가 에스컬레이트된 이벤트 핸들러와 연관된 조작

## BPC.BFM.ACTIVITY.EVENT

BPC.BFM.ACTIVITY.EVENT는 64 페이지의 『BPC.BFM.ACTIVITY.MESSAGE』에서 XML 요소를 상속합니다.

표 41. BPC.BFM.ACTIVITY.EVENT의 XML 요소

XML 요소	설명
<i>operation</i>	수신된 이벤트에 대한 조작의 이름

## BPC.BFM.LINK.STATUS

BPC.BFM.LINK.STATUS는 58 페이지의 『BPC.BFM.BASE』에서 XML 요소를 상속합니다.

표 42. BPC.BFM.LINK.STATUS의 XML 요소

XML 요소	설명
<i>elementName</i>	링크 이름
<i>description</i>	링크에 대한 설명

표 42. BPC.BFM.LINK.STATUS의 XML 요소 (계속)

XML 요소	설명
<i>flowBpellId</i>	링크가 정의된 플로우 활동의 ID

## BPC.BFM.VARIABLE.STATUS

BPC.BFM.VARIABLE.STATUS는 58 페이지의 『BPC.BFM.BASE』에서 XML 요소를 상속합니다.

표 43. BPC.BFM.VARIABLE.STATUS의 XML 요소

XML 요소	설명
<i>variableName</i>	변수 이름
<i>variableData</i> 또는 <i>variableData_BO</i>	<i>variableName</i> 변수가 초기화되지 않으면 <i>variableData</i> 또는 <i>VariableData_BO</i> 요소는 없습니다. 변수의 데이터는 문자열 또는 비즈니스 오브젝트(BO)로 표시됩니다. WebSphere Integration Developer의 이벤트 모니터 탭에서 호환 가능 이벤트 모니터 옵션을 선택했는지 여부에 따라 형식이 다릅니다.  이 속성은 WebSphere Business Monitor 6.0.2 형식 이벤트에만 사용됩니다. WebSphere Business Monitor 6.1 형식 이벤트의 경우에는 변수의 이름으로 설정된 이름이 있는 하나의 콘텐츠 요소를 포함하는 <i>applicationData</i> 섹션에 변수의 콘텐츠가 기록됩니다.
<i>bpellId</i>	변수에 대한 Business Process Choreographer ID
<i>principal</i>	변수를 갱신한 사용자의 이름

## 비즈니스 프로세스 이벤트

WebSphere Integration Developer의 비즈니스 프로세스 요소에 대한 모니터링이 요청되면 비즈니스 프로세스 이벤트가 전송됩니다. 비즈니스 프로세스에서 생성할 수 있는 모든 이벤트의 목록을 여기서 찾을 수 있습니다.

이벤트는 프로세스 또는 활동의 상태가 변경될 때 상태가 방출됩니다. 다음과 같은 이벤트 유형이 비즈니스 프로세스에 의해 발생할 수 있습니다.

- 67 페이지의 『프로세스 이벤트』
- 69 페이지의 『활동 이벤트』
- 71 페이지의 『활동 범위 이벤트』
- 72 페이지의 『링크 이벤트』
- 72 페이지의 『변수 이벤트』



## XML 스키마 정의(XSD) 파일

이벤트 구조는 XML 스키마 정의(XSD) 파일 BFMEvents.xsd에 설명되어 있습니다. 이 파일은 `install_root#ProcessChoreographer#client` 디렉토리에서 찾을 수 있습니다.

## 테이블 컬럼 키

다음 테이블의 열은 다음과 같습니다.

**코드** 이벤트의 번호를 포함합니다. WebSphere Business Monitor 6.0.2 형식 이벤트의 경우 이름이 `BPCEventCode`인 확장 데이터 요소로 공통 기본 이벤트에 값이 기록됩니다. WebSphere Business Monitor 6.1 형식 이벤트의 경우에는 공통 기본 이벤트의 `xs:any` 슬롯에 값이 기록됩니다.

### 확장자 이름

`extensionName`은 공통 기본 이벤트에 포함된 이벤트 특정 정보를 정의하는 문자열 값을 포함합니다. 이 이름은 이벤트에 대한 추가 데이터를 제공하는 XML 요소의 이름이기도 합니다. 확장자 이름에 대한 자세한 정보는 57 페이지의 『비즈니스 프로세스 이벤트의 확장자 이름』을 참조하십시오.

**상황** 비즈니스 프로세스 이벤트의 상황 이름을 참조합니다. 상황에 대한 세부사항은 73 페이지의 『비즈니스 프로세스 이벤트의 상황』의 내용을 참조하십시오.

### 이벤트 네이처

WebSphere Integration Developer에 표시된 대로 `EventNature` 매개변수의 비즈니스 프로세스 요소에 대한 이벤트 상황을 가리키는 포인터입니다.

## 프로세스 이벤트

다음 표는 모든 프로세스 이벤트를 설명합니다.

코드	확장자 이름	상황	이벤트 네이처	설명
21000	BPC.BFM.PROCESS.START	시작	ENTRY	프로세스 시작됨
21001	BPC.BFM.PROCESS.STATUS	보고서	SUSPENDED	프로세스가 일시중단됨. 프로세스 인스턴스를 일시중단하려면 Business Process Choreographer 탐색기를 사용하십시오.
21002	BPC.BFM.PROCESS.STATUS	보고서	RESUMED	프로세스가 재개됨. 일시중단된 프로세스만 재개할 수 있습니다. 프로세스 인스턴스를 재개하려면 Business Process Choreographer 탐색기를 사용하십시오.

코드	확장자 이름	상황	이벤트 네이처	설명
21004	BPC.BFM.PROCESS.STATUS	중지	EXIT	프로세스 완료됨
21005	BPC.BFM.PROCESS.STATUS	중지	TERMINATED	프로세스가 종료됨. 프로세스 인스턴스를 종료하려면 Business Process Choreographer 탐색기를 사용하십시오.
21019	BPC.BFM.PROCESS.START	보고서	RESTARTED	프로세스 다시 시작됨
21020	BPC.BFM.PROCESS.STATUS	삭제	DELETED	프로세스 삭제됨
42001	BPC.BFM.PROCESS. FAILURE	실패	FAILED	프로세스 실패함
42003	BPC.BFM.PROCESS.STATUS	보고서	COMPENSATING	프로세스 보상 중. 프로세스 인스턴스를 보상하려면 Business Process Choreographer 탐색기를 사용하십시오.
42004	BPC.BFM.PROCESS.STATUS	중지	COMPENSATED	프로세스 보상됨
42009	BPC.BFM.PROCESS.STATUS	보고서	TERMINATING	프로세스 종료 중
42010	BPC.BFM.PROCESS.STATUS	보고서	FAILING	프로세스 실패 중
42027	BPC.BFM.PROCESS.CORREL	보고서	CORRELATION	상관 설정이 초기화됨. 프로세스 인스턴스에 대한 새 상관 설정이 초기화될 때 발행됩니다. 예를 들어, 시작 상관 설정이 있는 receive 활동이 메시지를 수신하는 경우입니다.
42041	BPC.BFM.PROCESS. WISTATUS	보고서	WI_DELETED	프로세스 작업 항목 삭제됨
42042	BPC.BFM.PROCESS. WISTATUS	보고서	WI_CREATED	프로세스 작업 항목 작성됨
42046	BPC.BFM.PROCESS.STATUS	실패	COMPFAILED	프로세스 보상에 실패함
42047	BPC.BFM.PROCESS.EVENT	보고서	EV_RECEIVED	프로세스 이벤트가 수신됨. 이벤트를 정의하려면 프로세스 인터페이스를 사용하십시오. 이벤트는 프로세스와 연관된 이벤트 핸들러가 활성화될 때 생성됩니다.

코드	확장자 이름	상황	이벤트 네이처	설명
42049	BPC.BFM.PROCESS.ESCALATED	보고서	EV_ESCALATED	프로세스 이벤트가 에스컬레이트됨. 이 이벤트는 프로세스 레벨에 정의되어 있으며 onEvent 이벤트 핸들러와 연관된 인라인 호출 태스크가 에스컬레이트될 때 생성됩니다.
42056	BPC.BFM.PROCESS. WITRANSFER	보고서	WI_TRANSFERRED	프로세스 작업 항목 전송됨
42058	BPC.BFM.PROCESS.PARTNER	보고서	PA_CHANGE	프로세스 상대가 변경됨. 이 이벤트는 새 엔드포인트 참조가 상대 링크에 지정될 때 생성됩니다.

프로세스 이벤트의 경우 이벤트 상관 스피어 ID의 콘텐츠는 다음과 같습니다.

- ECSCurrentID는 프로세스 인스턴스 ID를 제공합니다.
- ECSParentID는 프로세스 인스턴스가 현재 프로세스 이벤트를 시작하기 이전 ECSCurrentID 값을 제공합니다.

### 활동 이벤트

다음 표는 모든 활동 이벤트를 설명합니다.

코드	확장자 이름	상황	이벤트 네이처	설명
21006	BPC.BFM.ACTIVITY.MESSAGE	시작	CREATED	활동이 준비됨. 이 이벤트는 휴먼 태스크 활동이 시작될 때 생성됩니다.
21007	호출 활동: BPC.BFM.ACTIVITY. MESSAGE. 모든 기타 활동 유형: BPC.BFM.ACTIVITY.STATUS	시작	ENTRY	활동이 시작됨. Invoke 활동의 경우 비즈니스 오브젝트 페이로드가 사용 가능합니다.
21011	invoke, 휴먼 태스크, receive 및 reply 활동: BPC.BFM.ACTIVITY. MESSAGE. pick 활동: BPC.BFM.ACTIVITY.EVENT. 모든 기타 활동 유형: BPC.BFM.ACTIVITY.STATUS	중지	EXIT	활동이 완료됨. invoke, 휴먼 태스크, receive 및 reply 활동의 경우 비즈니스 오브젝트 페이로드가 사용 가능합니다.
21021	BPC.BFM.ACTIVITY.STATUS	보고서	DEASSIGNED	청구가 취소됨. 이 이벤트는 휴먼 태스크 활동에 대한 청구가 취소될 때 생성됩니다.

코드	확장자 이름	상황	이벤트 네이처	설명
21022	BPC.BFM.ACTIVITY.CLAIM	보고서	ASSIGNED	활동이 청구됨. 이 이벤트는 휴먼 타스크 활동이 청구될 때 생성됩니다.
21027	BPC.BFM.ACTIVITY.STATUS	중지	TERMINATED	활동이 종료됨. 활동이 지정된 프로세스 또는 범위의 결합 핸들링으로 인해 장기 실행 활동이 종료될 수 있습니다.
21080	BPC.BFM.ACTIVITY.FAILURE	실패함	FAILED	활동 실패함
21081	BPC.BFM.ACTIVITY.STATUS	보고서	EXPIRED	활동이 만기됨. invoke 및 인라인 휴먼 타스크 활동에 대한 만기를 정의할 수 있습니다.
42005	BPC.BFM.ACTIVITY.STATUS	보고서	SKIPPED	활동이 생략됨. 이 이벤트는 결합 작동이 정의된 활동에만 적용할 수 있습니다. 결합 작동이 거짓으로 평가되면 활동을 건너뛰며 건너뛴 이벤트가 발행됩니다.
42012	BPC.BFM.ACTIVITY.MESSAGE	보고서	OUTPUTSET	활동 출력 메시지가 설정됨. 비즈니스 오브젝트 페이로드가 사용 가능합니다.
42013	BPC.BFM.ACTIVITY.MESSAGE	보고서	FAULTSET	활동 결합 메시지가 설정됨. 비즈니스 오브젝트 페이로드가 사용 가능합니다.
42015	BPC.BFM.ACTIVITY.STATUS	중지	STOPPED	활동 중지됨
42031	BPC.BFM.ACTIVITY.STATUS	보고서	FRETRIED	활동이 강제로 재시도됨. 활동을 재시도하도록 강제 실행하려면 Business Process Choreographer 탐색기를 사용하십시오.
42032	BPC.BFM.ACTIVITY.STATUS	중지	FCOMPLETED	활동이 강제로 완료됨. 활동을 완료하도록 강제 실행하려면 Business Process Choreographer 탐색기를 사용하십시오.
42036	BPC.BFM.ACTIVITY.MESSAGE	보고서	EXIT	활동에 메시지가 수신됨
42037	BPC.BFM.ACTIVITY.STATUS	보고서	CONDTRUE	TRUE 루프 조건

코드	확장자 이름	상황	이벤트 네이처	설명
42038	BPC.BFM.ACTIVITY.STATUS	보고서	CONDFALSE	FALSE 루프 조건
42039	BPC.BFM.ACTIVITY.WISTATUS	보고서	WI_DELETED	작업 항목이 삭제됨. 이 이벤트는 pick, 인라인 휴먼 태스크 및 receive 이벤트에만 적용할 수 있습니다.
42040	BPC.BFM.ACTIVITY.WISTATUS	보고서	WI_CREATED	작업 항목이 작성됨. 이 이벤트는 pick, 인라인 휴먼 태스크 및 receive 이벤트에만 적용할 수 있습니다.
42050	BPC.BFM.ACTIVITY.ESCALATED	보고서	ESCALATED	활동이 에스컬레이트됨. 이 이벤트는 pick, 인라인 휴먼 태스크 및 receive 이벤트에만 적용할 수 있습니다.
42054	BPC.BFM.ACTIVITY.WISTATUS	보고서	WI_REFRESHED	활동 작업 항목을 새로 고침. 이 이벤트는 pick, 인라인 휴먼 태스크 및 receive 이벤트에만 적용할 수 있습니다.
42055	BPC.BFM.ACTIVITY.WITRANSFER	보고서	WI_TRANSFERRED	작업 항목이 전송됨. 이 이벤트는 pick, 인라인 휴먼 태스크 및 receive 이벤트에만 적용할 수 있습니다.
42057	BPC.BFM.ACTIVITY.FOREACH	보고서	BRANCHES_STARTED	각 활동에 대해 분기가 시작됨

활동 이벤트의 경우 이벤트 상관 스피어 ID의 콘텐츠는 다음과 같습니다.

- *ECSCurrentID*는 활동 ID를 제공합니다.
- *ECSParentID*는 상위 프로세스의 ID를 제공합니다.

### 활동 범위 이벤트

다음 표는 모든 활동 범위 이벤트를 설명합니다.

코드	확장자 이름	상황	이벤트 네이처	설명
42020	BPC.BFM.ACTIVITY.STATUS	시작	ENTRY	범위가 시작됨
42021	BPC.BFM.ACTIVITY.STATUS	보고서	SKIPPED	범위가 생략됨
42022	BPC.BFM.ACTIVITY.FAILURE	실패	FAILED	범위가 실패함
42023	BPC.BFM.ACTIVITY.STATUS	보고서	FAILING	범위 종료 중
42024	BPC.BFM.ACTIVITY.STATUS	중지	TERMINATED	범위가 종료됨
42026	BPC.BFM.ACTIVITY.STATUS	중지	EXIT	범위가 완료됨

코드	확장자 이름	상황	이벤트 네이처	설명
42043	BPC.BFM.ACTIVITY.STATUS	보고서	COMPENSATING	범위 보상 중
42044	BPC.BFM.ACTIVITY.STATUS	중지	COMPENSATED	범위가 보상됨
42045	BPC.BFM.ACTIVITY.STATUS	실패	COMPFAILED	범위 보상이 실패함
42048	BPC.BFM.ACTIVITY.EVENT	보고서	EV_RECEIVED	활동 이벤트가 수신됨
42051	BPC.BFM.ACTIVITY.ESCALATED	보고서	EV_ESCALATED	범위 이벤트가 에스컬레이트됨

활동 범위 이벤트는 활동 이벤트의 유형으로, BPC.BFM.ACTIVITY.STATUS에 대하여 해당 구문을 위에서 설명합니다.

활동 범위 이벤트의 경우 이벤트 상관 스피어 ID의 콘텐츠는 다음과 같습니다.

- ECSCurrentID는 범위 ID를 제공합니다.
- ECSParentID는 상위 프로세스의 ID를 제공합니다.

### 링크 이벤트

다음 표는 모든 링크 이벤트를 설명합니다.

코드	확장자 이름	상황	이벤트 네이처	설명
21034	BPC.BFM.LINK.STATUS	보고서	CONDTRUE	TRUE 링크 평가
42000	BPC.BFM.LINK.STATUS	보고서	CONDFALSE	FALSE 링크 평가

링크 이벤트의 경우 이벤트 상관 스피어 ID의 콘텐츠는 다음과 같습니다.

- ECSCurrentID는 링크의 소스 활동 ID를 제공합니다.
- ECSParentID는 상위 프로세스의 ID를 제공합니다.

### 변수 이벤트

다음 표는 변수 이벤트를 설명합니다.

코드	확장자 이름	상황	이벤트 네이처	설명
21090	BPC.BFM.VARIABLE.STATUS	보고서	CHANGED	변수 갱신. 비즈니스 오브젝트 페이로드가 사용 가능합니다.

변수 이벤트의 경우 이벤트 상관 스피어 ID의 콘텐츠는 다음과 같습니다.

- ECSCurrentID는 포함하는 프로세스의 ID를 제공합니다.
- ECSParentID는 프로세스 인스턴스가 현재 프로세스 이벤트를 시작하기 이전 ECSCurrentID입니다.

## 비즈니스 프로세스 이벤트의 상황

비즈니스 프로세스 이벤트는 서로 다른 상황에서 생성될 수 있습니다. 해당 상황에 대한 데이터를 상황 요소에서 설명합니다.

비즈니스 프로세스는 다음과 같은 상황 요소 중 하나를 포함할 수 있습니다.

상황 이름	공통 기본 이벤트의 콘텐츠	
시작	categoryName이 StartSituation으로 설정됩니다.	
	situationType	
	유형	StartSituation
	reasoningScope	EXTERNAL
	successDisposition	SUCCESSFUL
	situationQualifier	START_COMPLETED
중지	categoryName이 StopSituation으로 설정됩니다.	
	situationType	
	유형	StopSituation
	reasoningScope	EXTERNAL
	successDisposition	SUCCESSFUL
	situationQualifier	STOP_COMPLETED
삭제	categoryName이 DestroySituation으로 설정됩니다.	
	situationType	
	유형	DestroySituation
	reasoningScope	EXTERNAL
	successDisposition	SUCCESSFUL
실패	categoryName이 StopSituation으로 설정됩니다.	
	situationType	
	유형	StopSituation
	reasoningScope	EXTERNAL
	successDisposition	UNSUCCESSFUL
	situationQualifier	STOP_COMPLETED
보고서	categoryName이 ReportSituation으로 설정됩니다.	
	situationType	
	유형	ReportSituation
	reasoningScope	EXTERNAL
	reportCategory	STATUS

## 휴먼 태스크 이벤트 모니터링

휴먼 태스크를 대신해서 생성되는 이벤트는 상황 독립적인 데이터와 비즈니스 프로세스 이벤트에 특정한 데이터로 구성됩니다. 휴먼 태스크 이벤트에 특정한 속성 및 요소를 기술합니다.

휴먼 태스크 이벤트에는 다음과 같은 범주의 이벤트 콘텐츠가 있을 수 있습니다.

## 휴먼 태스크 특정 이벤트 데이터

이벤트는 태스크 및 에스컬레이션 대신 작성됩니다. 각각의 해당 이벤트 유형의 오브젝트 특정 콘텐츠를 설명합니다.

Business Process Choreographer 버전 6.1에는 두 가지 이벤트 형식이 발생할 수 있습니다.

### WebSphere Business Monitor 6.0.2 형식

WebSphere Business Monitor 6.0.2 형식 이벤트는 WebSphere Integration Developer 6.0.2 내에 모델화된 태스크가 있을 때나 WebSphere Business Monitor 6.0.2 형식 모드가 WebSphere Integration Developer 6.1 내에서 사용 가능할 때 발생합니다. 다르게 지정하지 않으면 이 이벤트에 대한 오브젝트 특정 콘텐츠가 유형 문자열의 *extendedDataElement* XML 요소로 기록됩니다.

### WebSphere Business Monitor 6.1 형식

WebSphere Business Monitor 6.1 형식 이벤트는 WebSphere Integration Developer 6.1 내에 모델화된 태스크가 있고 WebSphere Business Monitor 6.1 형식 모드가 사용 가능할 때 발생합니다. 이 이벤트에 대한 오브젝트 특정 콘텐츠는 공통 기본 이벤트의 *eventPointData* 폴더에 있는 *xs:any* 슬롯의 XML 요소로 기록됩니다. XML의 구조는 XSD(XML Schema Definition) 파일 *HTMEvents.xsd*에 정의됩니다. 이 파일은 *install\_root#ProcessChoreographer#client* 디렉토리에서 찾을 수 있습니다.

## 태스크

태스크 이벤트는 다음과 같은 오브젝트 특정 이벤트 콘텐츠를 갖습니다.

속성	설명
<i>taskTemplateName</i>	인스턴스가 파생된 태스크 템플릿의 이름
<i>taskTemplateValidFrom</i>	템플릿이 유효하기 시작한 날짜
<i>taskTemplateId</i>	인스턴스가 파생된 태스크 템플릿의 ID
<i>taskInstanceDescription</i>	기본 로케일의 태스크 인스턴스에 대한 설명
<i>PayloadType</i>	페이로드 유형. 문자열 값은 <i>none</i> , <i>digest</i> 또는 <i>full</i> 중 하나일 수 있습니다. 값은 WebSphere Integration Developer의 설정에 따라 다릅니다. 사용 가능한 명시 모니터링 정의가 없으면 기본 페이로드 유형은 <i>full</i> 입니다.

## 에스컬레이션

에스컬레이션은 다음과 같은 오브젝트 특정 이벤트 콘텐츠를 갖습니다.

속성	설명
<i>taskTemplateName</i>	인스턴스가 파생된 태스크 템플릿의 이름



속성	설명
taskTemplateValidFrom	템플릿이 유효하기 시작한 날짜
taskTemplateId	인스턴스가 파생된 태스크 템플릿의 ID
escalationName	에스컬레이션 이름
escalationInstanceDescription	선택적: 에스컬레이션 인스턴스에 대한 설명
PayloadType	페이로드 유형. 문자열 값은 none, digest 또는 full 중 하나일 수 있습니다. 값은 WebSphere Integration Developer의 설정에 따라 다릅니다. 사용 가능한 명시 모니터링 정의가 없으면 기본 페이로드 유형은 full입니다.

## 휴먼 태스크 이벤트의 확장자 이름

확장자 이름은 휴먼 태스크 이벤트의 페이로드를 나타냅니다. 휴먼 태스크 이벤트 및 해당 페이로드에 대한 모든 확장 이름의 목록을 여기에서 찾을 수 있습니다.

확장자 이름은 공통 기본 이벤트의 *extensionName* 속성 값으로 사용되는 문자열 값을 포함합니다. 이 이름은 이벤트에 대한 추가 데이터를 제공하는 XML 요소의 이름이기도 합니다. 이벤트 요소의 이름은 대문자이며(예: BPC.HTM.BASE) XML 요소의 이름은 대소문자가 혼합됩니다(예: *HTMEventCode*). 표시한 경우를 제외하고 모든 데이터 요소가 string 유형입니다.

다음 확장자 이름이 휴먼 태스크 이벤트에 사용 가능합니다.

- 76 페이지의 『BPC.HTM.BASE』
- 76 페이지의 『BPC.HTM.TASK.BASE』
- 76 페이지의 『BPC.HTM.TASK.STATUS』
- 76 페이지의 『BPC.HTM.TASK.FOLLOW』
- 76 페이지의 『BPC.HTM.TASK.MESSAGE』
- 77 페이지의 『BPC.HTM.TASK.INTERACT』
- 77 페이지의 『BPC.HTM.TASK.FAILURE』
- 77 페이지의 『BPC.HTM.TASK.WISTATUS』
- 78 페이지의 『BPC.HTM.TASK.WITRANSFER』
- 78 페이지의 『BPC.HTM.ESCALATION.STATUS』
- 78 페이지의 『BPC.HTM.ESCALATION.WISTATUS』
- 78 페이지의 『BPC.HTM.ESCALATION.WITRANSFER』

## BPC.HTM.BASE

BPC.HTM.BASE는 WBIMonitoringEvent로부터 XML 요소를 상속합니다.

표 44. BPC.HTM.BASE의 XML 요소

XML 요소	설명
<i>HTMEventCode</i>	이벤트 유형의 수를 식별하는 Business Process Choreographer 이벤트 코드. 가능한 이벤트 코드가 다음 테이블에 나열됩니다.

## BPC.HTM.TASK.BASE

BPC.HTM.TASK.BASE는 『BPC.HTM.BASE』에서 XML 요소를 상속합니다.

표 45. BPC.HTM.TASK.BASE의 XML 요소

XML 요소	설명
<i>taskTemplateId</i>	템플릿 ID
<i>taskTemplateName</i>	태스크 템플릿의 이름. 이 이름은 표시 이름과 다를 수 있습니다.
<i>taskTemplateValidFrom</i>	태스크 템플릿이 유효한 날짜 및 시간

## BPC.HTM.TASK.STATUS

BPC.HTM.TASK.STATUS는 『BPC.HTM.TASK.BASE』에서 XML 요소를 상속합니다.

표 46. BPC.HTM.TASK.STATUS의 XML 요소

XML 요소	설명
<i>taskInstanceDescription</i>	태스크에 대한 설명

## BPC.HTM.TASK.FOLLOW

BPC.HTM.TASK.FOLLOW는 『BPC.HTM.TASK.BASE』에서 XML 요소를 상속합니다.

표 47. BPC.HTM.TASK.FOLLOW의 XML 요소

XML 요소	설명
<i>followTaskId</i>	후속 태스크로 시작된 태스크 ID

## BPC.HTM.TASK.MESSAGE

BPC.HTM.TASK.MESSAGE는 『BPC.HTM.TASK.STATUS』에서 XML 요소를 상속합니다.

표 48. BPC.HTM.TASK.FOLLOW의 XML 요소

XML 요소	설명
<i>message</i> 또는 <i>message_BO</i>	<p>입력 또는 출력 메시지를 포함하는 문자열 또는 비즈니스 오브젝트 표시. WebSphere Integration Developer의 이벤트 모니터 탭에서 호환 가능 이벤트 모니터 옵션을 선택했는지 여부에 따라 형식이 다릅니다.</p> <p>이 속성은 WebSphere Business Monitor 6.0.2 형식 이벤트에만 사용됩니다. WebSphere Business Monitor 6.1 형식 이벤트의 경우에는 메시지의 이름으로 설정된 이름이 있는 하나의 콘텐츠 요소를 포함하는 <i>applicationData</i> 섹션에 메시지의 콘텐츠가 기록됩니다.</p>

## BPC.HTM.TASK.INTERACT

BPC.HTM.TASK.INTERACT는 76 페이지의 『BPC.HTM.TASK.STATUS』에서 XML 요소를 상속합니다.

표 49. BPC.HTM.TASK.INTERACT의 XML 요소

XML 요소	설명
<i>username</i>	BPC.HTM.TASK.INTERACT의 경우 타스크와 연관된 사용자의 이름

## BPC.HTM.TASK.FAILURE

BPC.HTM.TASK.FAILURE는 76 페이지의 『BPC.HTM.TASK.STATUS』에서 XML 요소를 상속합니다.

표 50. BPC.HTM.TASK.FAILURE의 XML 요소

XML 요소	설명
<i>taskFailedException</i>	세미콜론(;)으로 분리된 <i>faultNameSpace</i> 및 <i>faultName</i> 을 포함하는 문자열

## BPC.HTM.TASK.WISTATUS

BPC.HTM.TASK.WISTATUS는 76 페이지의 『BPC.HTM.TASK.STATUS』에서 XML 요소를 상속합니다.

표 51. BPC.HTM.TASK.WISTATUS의 XML 요소

XML 요소	설명
<i>username</i>	BPC.BPC.TASK.WISTATUS의 경우에는 작성되거나 삭제된 작업 항목이 있는 사용자 목록

## BPC.HTM.TASK.WITRANSFER

BPC.HTM.TASK.WITRANSFER는 76 페이지의 『BPC.HTM.TASK.STATUS』에서 XML 요소를 상속합니다.

표 52. BPC.HTM.TASK.WITRANSFER의 XML 요소

XML 요소	설명
<i>current</i>	BPC.HTM.TASK.WITRANSFER의 경우에는 현재 사용자의 이름. 작업 항목이 다른 누군가에게 전송된 사용자입니다.
<i>target</i>	BPC.HTM.TASK.WITRANSFER의 경우에는 작업 항목 수신자의 사용자 이름

## BPC.HTM.ESCALATION.STATUS

BPC.HTM.ESCALATION.STATUS는 76 페이지의 『BPC.HTM.TASK.BASE』에서 XML 요소를 상속합니다.

표 53. BPC.HTM.ESCALATION.STATUS의 XML 요소

XML 요소	설명
<i>escalationName</i>	에스컬레이션 이름
<i>escalationInstanceDescription</i>	에스컬레이션에 대한 설명

## BPC.HTM.ESCALATION.WISTATUS

BPC.HTM.ESCALATION.WISTATUS는 『BPC.HTM.ESCALATION.STATUS』에서 XML 요소를 상속합니다.

표 54. BPC.HTM.ESCALATION.WISTATUS의 XML 요소

XML 요소	설명
<i>username</i>	BPC.HTM.ESCALATION.WISTATUS의 경우에는 에스컬레이트된 작업 항목이 있는 사용자의 목록

## BPC.HTM.ESCALATION.WITRANSFER

BPC.HTM.ESCALATION.WITRANSFER는 『BPC.HTM.ESCALATION.STATUS』에서 XML 요소를 상속합니다.

표 55. BPC.HTM.ESCALATION.WITRANSFER의 XML 요소

XML 요소	설명
<i>current</i>	BPC.HTM.ESCALATION.WITRANSFER의 경우에는 현재 사용자의 이름. 작업 항목이 다른 누군가에게 전송된 사용자입니다.
<i>target</i>	BPC.HTM.ESCALATION.WITRANSFER의 경우에는 작업 항목 수신자의 사용자 이름.

## 휴먼 태스크 이벤트

WebSphere Integration Developer의 태스크 요소에 대한 모니터링이 요청되면 휴먼 태스크 이벤트가 전송됩니다. 휴먼 태스크에서 생성할 수 있는 모든 이벤트의 목록을 여기서 찾을 수 있습니다.

이벤트는 태스크의 상태가 변경될 때 상태가 발행됩니다. 다음과 같은 이벤트 유형이 휴먼 태스크에 의해 발생할 수 있습니다.

- 80 페이지의 『태스크 이벤트』
- 80 페이지의 『에스컬레이션 이벤트』

주: 비즈니스 관련성 플래그가 태스크 모델에서 TRUE로 설정된 경우 임시 태스크에 대해서만 이벤트가 생성됩니다.

인라인 태스크에 대한 이벤트는 활동 이벤트로 발행됩니다. 이벤트의 목록을 보려면 66 페이지의 『비즈니스 프로세스 이벤트』를 참조하십시오.

## XML 스키마 정의(XSD) 파일

이벤트 구조는 XML 스키마 정의(XSD) 파일 `HTMEvents.xsd`에 설명되어 있습니다. 이 파일은 `install_root#ProcessChoreographer#client` 디렉토리에서 찾을 수 있습니다.

## 테이블 컬럼 키

다음 테이블의 열은 다음과 같습니다.

**코드** 이벤트의 번호를 포함합니다. WebSphere Business Monitor 6.0.2 형식 이벤트의 경우 이름이 `HTMEventCode`인 확장 데이터 요소로 공통 기본 이벤트에 값이 기록됩니다. WebSphere Business Monitor 6.1 형식 이벤트의 경우에는 공통 기본 이벤트의 `xs:any` 슬롯에 값이 기록됩니다.

### 확장자 이름

공통 기본 이벤트의 `extensionName` 속성 값으로 사용되는 문자열 값을 포함합니다.

WebSphere Business Integration Modeler를 사용하여 기본 태스크 모델을 작성할 경우 해시 문자(#) 뒤에 문자를 추가해서 페이로드에 메시지 데이터가 포함된 이벤트의 확장자 이름을 확장할 수 있습니다. 추가된 문자를 사용하여 다양한 메시지 오브젝트를 전달하는 공통 기본 이벤트를 구별할 수 있습니다. 메시지 데이터를 생성하는 이벤트에는 데이터 오브젝트의 내용을 보고하는 중첩 `extendedDataElements`도 포함됩니다. 자세한 정보는 WebSphere Business Integration Modeler 문서를 참조하십시오.

**상황** 휴먼 태스크 이벤트의 상황 이름을 참조합니다. 상황에 대한 세부사항은 81 페이지의 『휴먼 태스크 이벤트의 상황』의 내용을 참조하십시오.

## 이벤트 네이처

WebSphere Integration Developer에 표시된 대로 EventNature 매개변수의 비즈니스 프로세스 요소에 대한 이벤트 상황을 가리키는 포인터입니다.

## 태스크 이벤트

다음 표는 모든 태스크 이벤트를 설명합니다.

코드	확장자 이름	상황	이벤트 네이처	설명
51001	BPC.HTM.TASK.INTERACT	보고서	CREATED	태스크가 작성됨
51002	BPC.HTM.TASK.STATUS	삭제	DELETED	태스크가 삭제됨
51003	BPC.HTM.TASK.STATUS	시작	ENTRY	태스크가 시작됨
51004	BPC.HTM.TASK.STATUS	중지	EXIT	태스크가 완료됨
51005	BPC.HTM.TASK.STATUS	보고서	DEASSIGNED	청구 취소됨
51006	BPC.HTM.TASK.INTERACT	보고서	ASSIGNED	태스크가 청구됨
51007	BPC.HTM.TASK.STATUS	중지	TERMINATED	태스크가 종료됨
51008	BPC.HTM.TASK.FAILURE	실패	FAILED	태스크가 실패함
51009	BPC.HTM.TASK.STATUS	보고서	EXPIRED	태스크가 만기됨
51010	BPC.HTM.TASK.STATUS	보고서	WAITFORSUBTASK	하위 태스크 대기 중
51011	BPC.HTM.TASK.STATUS	중지	SUBTASKCOMPLETED	하위 태스크가 완료됨
51012	BPC.HTM.TASK.STATUS	보고서	RESTARTED	태스크가 재시작됨
51013	BPC.HTM.TASK.STATUS	보고서	SUSPENDED	태스크가 일시중단됨
51014	BPC.HTM.TASK.STATUS	보고서	RESUMED	태스크가 재개됨
51015	BPC.HTM.TASK.FOLLOW	보고서	COMPLETEDFOLLOW	태스크가 완료되고 후속 태스크가 시작됨
51101	BPC.HTM.TASK.STATUS	보고서	UPDATED	태스크 특성이 갱신됨
51103	BPC.HTM.TASK.MESSAGE	보고서	OUTPUTSET	출력 메시지가 갱신됨. 비즈니스 오브젝트 페이지 로드가 사용 가능합니다.
51104	BPC.HTM.TASK.MESSAGE	보고서	FAULTSET	결함 메시지가 갱신됨. 비즈니스 오브젝트 페이지 로드가 사용 가능합니다.
51201	BPC.HTM.TASK.WISTATUS	삭제	WI_DELETED	작업 항목 삭제됨
51202	BPC.HTM.TASK.WISTATUS	보고서	WI_CREATED	작업 항목 작성됨
51204	BPC.HTM.TASK.WITRANSFER	보고서	WI_TRANSFERRED	작업 항목 전송됨
51205	BPC.HTM.TASK.WISTATUS	보고서	WI_REFRESHED	작업 항목 새로 고침

태스크 이벤트의 경우 이벤트 상관 스피어 ID의 콘텐츠는 다음과 같습니다.

- ESCcurrentID는 태스크 인스턴스의 ID를 제공합니다.
- ECSParentID는 태스크 인스턴스 이벤트 이전 ECSCurrentID입니다.

## 에스컬레이션 이벤트

다음 표는 모든 태스크 에스컬레이션 이벤트를 설명합니다.

코드	확장자 이름	상황	이벤트 네이처	설명
53001	BPC.HTM.ESCALATION. STATUS	보고서	ENTRY	에스컬레이션이 시작됨
53201	BPC.HTM.ESCALATION. WISTATUS	삭제	WI_DELETED	작업 항목 삭제됨
53202	BPC.HTM.ESCALATION. WISTATUS	보고서	WI_CREATED	작업 항목 작성됨
53204	BPC.HTM.ESCALATION. WITRANSFER	보고서	WI_TRANS-FERRED	에스컬레이션이 전송됨
53205	BPC.HTM.ESCALATION. WISTATUS	보고서	WI_REFRESH-ED	작업 항목 새로 고침

태스크 이벤트의 경우 이벤트 상관 스피어 ID의 콘텐츠는 다음과 같습니다.

- ESCcurrentID는 에스컬레이션의 ID를 제공합니다.
- ECSParentID는 연관된 태스크 인스턴스의 ID를 제공합니다.

### 휴먼 태스크 이벤트의 상황

비즈니스 프로세스 이벤트는 서로 다른 상황에서 생성될 수 있습니다. 해당 상황에 대한 데이터를 상황 요소에서 설명합니다.

비즈니스 프로세스는 다음과 같은 상황 요소 중 하나를 포함할 수 있습니다.

상황 이름	공통 기본 이벤트의 콘텐츠	
시작	categoryName이 StartSituation으로 설정됩니다.	
	situationType	
	유형	StartSituation
	reasoningScope	EXTERNAL
	successDisposition	SUCCESSFUL
	situationQualifier	START_COMPLETED
중지	categoryName이 StopSituation으로 설정됩니다.	
	situationType	
	유형	StopSituation
	reasoningScope	EXTERNAL
	successDisposition	SUCCESSFUL
	situationQualifier	STOP_COMPLETED
삭제	categoryName이 DestroySituation으로 설정됩니다.	
	situationType	
	유형	DestroySituation
	reasoningScope	EXTERNAL
	successDisposition	SUCCESSFUL

상황 이름	공통 기본 이벤트의 콘텐츠	
실패	categoryName이 StopSituation으로 설정됩니다.	
	situationType	
	유형	StopSituation
	reasoningScope	EXTERNAL
	successDisposition	UNSUCCESSFUL
	situationQualifier	STOP_COMPLETED
보고서	categoryName이 ReportSituation으로 설정됩니다.	
	situationType	
	유형	ReportSituation
	reasoningScope	EXTERNAL
	reportCategory	STATUS

## 프로세스 서버 이벤트

WebSphere Process Server는 자체 서비스 컴포넌트가 주요 특징을 이루며 각 컴포넌트에는 모니터링할 수 있는 자체적인 이벤트 지점 세트가 있습니다. 서비스 컴포넌트는 하나 이상의 요소를 포함하며 이들 요소는 각 서비스 컴포넌트에서 처리되는 다른 단계 세트입니다. 각 요소에는 서비스 컴포넌트 요소를 처리할 때 도달하는 핵심 지점인 자체 이벤트 네이처 세트가 있습니다. 모든 서비스 컴포넌트, 요소 및 연관된 이벤트 네이처와 각 이벤트에 고유한 확장 데이터 요소가 나열됩니다.

## 자원 어댑터 이벤트

각 이벤트에 고유한 확장 데이터 요소, 이름 및 연관된 이벤트 네이처와 함께 모니터링할 수 있는 자원 어댑터 컴포넌트의 요소(기본 이름은 eis:WBI.JCAAdapter)가 여기에 나열됩니다.

이벤트 이름	이벤트 네이처	이벤트 콘텐츠	유형
<b>InboundEventRetrieval 요소</b>			
eis:WBI.JCAAdapter. InboundEventRetrieval. ENTRY	ENTRY	pollQuantity	int
		status	int
		eventTypeFilters	문자열
eis:WBI.JCAAdapter. InboundEventRetrieval. EXIT	EXIT	해당 없음	
eis:WBI.JCAAdapter. InboundEventRetrieval. FAILURE	FAILURE	FailureReason	예외
<b>InboundEventDelivery 요소</b>			
eis:WBI.JCAAdapter. InboundEventDelivery.ENTRY	ENTRY	해당 없음	
eis:WBI.JCAAdapter. InboundEventDelivery.EXIT	EXIT	해당 없음	



이벤트 이름	이벤트 네이처	이벤트 콘텐츠	유형
eis:WBI.JCAAdapter. InboundEventDelivery.FAILURE	FAILURE	FailureReason	예외
<b>Outbound 요소</b>			
eis:WBI.JCAAdapter. Outbound.ENTRY	ENTRY	해당 없음	
eis:WBI.JCAAdapter. Outbound.EXIT	EXIT	해당 없음	
eis:WBI.JCAAdapter. Outbound.FAILURE	FAILURE	FailureReason	예외
<b>InboundCallbackAsyncDeliverEvent 요소</b>			
eis:WBI.JCAAdapter. InboundCallbackAsyncDeliverEvent. ENTRY	ENTRY	해당 없음	
eis:WBI.JCAAdapter. InboundCallbackAsyncDeliverEvent. EXIT	EXIT	해당 없음	
eis:WBI.JCAAdapter. InboundCallbackAsyncDeliverEvent. FAILURE	FAILURE	FailureReason	예외
<b>InboundCallbackSyncDeliverEvent 요소</b>			
eis:WBI.JCAAdapter. InboundCallbackSyncDeliverEvent. ENTRY	ENTRY	해당 없음	
eis:WBI.JCAAdapter. InboundCallbackSyncDeliverEvent. EXIT	EXIT	해당 없음	
eis:WBI.JCAAdapter. InboundCallbackSyncDeliverEvent. FAILURE	FAILURE	FailureReason	예외
<b>Polling 요소</b>			
eis:WBI.JCAAdapter. Polling.STARTED	STARTED	PollFrequency	int
		PollQuantity	int
eis:WBI.JCAAdapter. Polling.STOPPED	STOPPED	해당 없음	
<b>Delivery 요소</b>			
eis:WBI.JCAAdapter. Delivery.EXIT	EXIT	해당 없음	
eis:WBI.JCAAdapter. Delivery.FAILURE	FAILURE	EventID	문자열
		FailureReason	예외
<b>Retrieval 요소</b>			
eis:WBI.JCAAdapter. Retrieval.FAILURE	FAILURE	EventID	문자열
		FailureReason	예외
<b>Endpoint 요소</b>			
eis:WBI.JCAAdapter. Endpoint.FAILURE	FAILURE	FailureReason	예외
<b>Recovery 요소</b>			
eis:WBI.JCAAdapter. Recovery.EXIT	EXIT	해당 없음	
eis:WBI.JCAAdapter. Recovery.FAILURE	FAILURE	FailureReason	예외
<b>EventFailure 요소</b>			

이벤트 이름	이벤트 네이처	이벤트 콘텐츠	유형
eis:WBI.JCAAdapter. EventFailure.FAILURE	FAILURE	FailureReason	예외
<b>Connection 요소</b>			
eis:WBI.JCAAdapter. Connection.FAILURE	FAILURE	FailureReason	예외

## 비즈니스 규칙 이벤트

비즈니스 규칙 컴포넌트(기본 이름 br:WBI.BR)는 모니터할 수 있는 단일 요소를 포함합니다. 각 이벤트에 고유한 확장 데이터 요소, 이벤트 이름 및 연관된 이벤트 네이처와 함께 이 요소에 대한 모든 이벤트 유형이 여기에 나열됩니다.

이벤트 이름	이벤트 네이처	이벤트 콘텐츠	유형
br:WBI.BR.ENTRY	ENTRY	operationName	문자열
br:WBI.BR.EXIT	EXIT	operationName	문자열
br:WBI.BR.FAILURE	FAILURE	ErrorReport	예외
		operationName	문자열
WBI.BR. br:SelectionKeyExtracted	SelectionKeyExtracted	operationName	문자열
br:WBI.BR.TargetFound	TargetFound	operationName	문자열
		target	문자열

## 비즈니스 상태 시스템 이벤트

각 이벤트에 고유한 모든 확장 데이터 요소, 이벤트 이름 및 연관된 이벤트 네이처와 함께 모니터할 수 있는 비즈니스 상태 시스템 컴포넌트의 요소(기본 이름은 bsm:WBI.BSM)가 여기에 나열됩니다.

이벤트 이름	이벤트 네이처	이벤트 콘텐츠	유형
<b>StateMachineDefinition 요소</b>			
bsm:WBI.BSM.StateMachineDefinition. ALLOCATED	ALLOCATED	instanceID	문자열
bsm:WBI.BSM.StateMachineDefinition. RELEASED	RELEASED	instanceID	문자열
<b>Transition 요소</b>			
bsm:WBI.BSM.Transition.ENTRY	ENTRY	instanceID	문자열
		name	문자열
bsm:WBI.BSM.Transition.EXIT	EXIT	instanceID	문자열
		name	문자열
bsm:WBI.BSM.Transition.FAILURE	FAILURE	ErrorReport	예외
		instanceID	문자열
		name	문자열

이벤트 이름	이벤트 네이처	이벤트 콘텐츠	유형
<b>State 요소</b>			
bsm:WBI.BSM.State.ENTRY	ENTRY	instanceID	문자열
		name	문자열
bsm:WBI.BSM.State.EXIT	EXIT	instanceID	문자열
		name	문자열
bsm:WBI.BSM.State.FAILURE	FAILURE	ErrorReport	예외
		instanceID	문자열
		name	문자열
<b>Guard 요소</b>			
bsm:WBI.BSM.Guard.ENTRY	ENTRY	instanceID	문자열
		name	문자열
bsm:WBI.BSM.Guard.EXIT	EXIT	instanceID	문자열
		name	문자열
		결과	boolean
bsm:WBI.BSM.Guard.FAILURE	FAILURE	ErrorReport	예외
		instanceID	문자열
		name	문자열
<b>Action 요소</b>			
bsm:WBI.BSM.Action.ENTRY	ENTRY	instanceID	문자열
		name	문자열
bsm:WBI.BSM.Action.EXIT	EXIT	instanceID	문자열
		name	문자열
bsm:WBI.BSM.Action.FAILURE	FAILURE	ErrorReport	예외
		instanceID	문자열
		name	문자열
<b>EntryAction 요소</b>			
bsm:WBI.BSM.EntryAction. ENTRY	ENTRY	instanceID	문자열
		name	문자열
bsm:WBI.BSM.EntryAction. EXIT	EXIT	instanceID	문자열
		name	문자열
bsm:WBI.BSM.EntryAction. FAILURE	FAILURE	ErrorReport	예외
		instanceID	문자열
		name	문자열
<b>ExitAction 요소</b>			
bsm:WBI.BSM.ExitAction.ENTRY	ENTRY	instanceID	문자열
		name	문자열
bsm:WBI.BSM.ExitAction.EXIT	EXIT	instanceID	문자열
		name	문자열
bsm:WBI.BSM.ExitAction. FAILURE	FAILURE	ErrorReport	예외
		instanceID	문자열
		name	문자열

이벤트 이름	이벤트 네이처	이벤트 콘텐츠	유형
<b>Timer 요소</b>			
bsm:WBI.BSM.Timer.START	START	instanceID	문자열
		name	문자열
		duration	문자열
bsm:WBI.BSM.Timer.STOPPED	STOPPED	instanceID	문자열
		name	문자열
		duration	문자열

## 맵 이벤트

각 이벤트에 고유한 모든 확장 데이터 요소, 이벤트 이름 및 이벤트 네이처와 함께 모니터링할 수 있는 맵 컴포넌트의 요소(기본 이름은 map:WBI.MAP)가 여기에 나열됩니다.

표 56. 기본 요소

이벤트 이름	이벤트 네이처	이벤트 콘텐츠	유형
map:WBI.MAP.ENTRY	ENTRY	해당 없음	해당 없음
map:WBI.MAP.EXIT	EXIT	해당 없음	해당 없음
map:WBI.MAP.FAILURE	FAILURE	FailureReason	예외
<b>Transformation 요소</b>			
map:WBI.MAP.Transformation. ENTRY	ENTRY	해당 없음	해당 없음
map:WBI.MAP.Transformation. EXIT	EXIT	해당 없음	해당 없음
map:WBI.MAP.Transformation. FAILURE	FAILURE	FailureReason	예외

## 중개 이벤트

각 이벤트에 고유한 모든 확장 데이터 요소, 이름 및 연관된 이벤트 네이처와 함께 모니터링할 수 있는 중개 컴포넌트의 요소(기본 이름은 ifm:WBI.MEDIATION)가 여기에 나열됩니다.

이벤트 이름	이벤트 네이처	이벤트 콘텐츠	유형
<b>OperationBinding element</b>			
ifm:WBI.MEDIATION. OperationBinding.ENTRY	ENTRY	InteractionType	문자열
		TicketID	문자열
		Source	문자열
		Target	문자열
ifm:WBI.MEDIATION. OperationBinding.EXIT	EXIT	InteractionType	문자열
		TicketID	문자열
		Source	문자열
		Target	문자열

이벤트 이름	이벤트 네이처	이벤트 콘텐츠	유형
ifm:WBI.MEDIATION. OperationBinding.FAILURE	FAILURE	InteractionType	문자열
		TicketID	문자열
		Source	문자열
		Target	문자열
		ErrorReport	예외
<b>ParameterMediation</b> 요소			
ifm:WBI.MEDIATION. ParameterMediation. ENTRY	ENTRY	종류	문자열
		TransformName	문자열
WBI.MEDIATION. ParameterMediation. EXIT	EXIT	종류	문자열
		TransformName	문자열
ifm:WBI.MEDIATION. ParameterMediation. FAILURE	FAILURE	종류	문자열
		TransformName	문자열
		ErrorReport	예외

## 복구 이벤트

복구 컴포넌트(기본 이름 `recovery:WBI.Recovery`)는 모니터링할 수 있는 단일 요소를 포함합니다. 각 이벤트에 고유한 확장 데이터 요소, 이벤트 이름 및 연관된 이벤트 네이처와 함께 이 요소에 대한 모든 이벤트 유형이 여기에 나열됩니다.

이벤트 이름	이벤트 네이처	이벤트 콘텐츠	유형
recovery:WBI.Recovery. FAILURE	FAILURE	MsgId	문자열
		DestModuleName	문자열
		DestComponentName	문자열
		DestMethodName	문자열
		SourceModuleName	문자열
		SourceComponentName	문자열
		ResubmitDestination	문자열
		ExceptionDetails	문자열
		SessionId	문자열
		FailureTime	dateTime
		ExpirationTime	dateTime
		Status	int
		MessageBody	byteArray
recovery:WBI.Recovery. DEADLOOP	DEADLOOP	DeadloopMsgId	문자열
		SIBusName	문자열
		QueueName	문자열
		Reason	문자열

이벤트 이름	이벤트 네이처	이벤트 콘텐츠	유형
recovery:WBI.Recovery. RESUBMIT	RESUBMIT	MsgId	문자열
		OriginalMesId	문자열
		ResubmitCount	int
		Description	문자열
recovery:WBI.Recovery. DELETE	DELETE	MsgId	문자열
		deleteTime	dateTime
		Description	문자열

## SCA(Service Component Architecture) 이벤트

SCA(Service Component Architecture)는 기본 이름이 sca:WBI.SCA.MethodInvocation인 단일 요소를 포함합니다. 각 이벤트에 고유한 모든 확장 데이터 요소와 함께 이 요소의 모든 이벤트 및 연관된 네이처가 여기에 나열됩니다.

주: 이 이벤트를 SCA 특정 ARM 성능 통계와 혼동해선 안됩니다.

이벤트 이름	이벤트 네이처	이벤트 콘텐츠	유형
WBI.SCA. MethodInvocation. ENTRY	ENTRY	SOURCE COMPONENT	문자열
		SOURCE INTERFACE	문자열
		SOURCE METHOD	문자열
		SOURCE MODULE	문자열
		SOURCE REFERENCE	문자열
		TARGET COMPONENT	문자열
		TARGET INTERFACE	문자열
		TARGET METHOD	문자열
WBI.SCA. MethodInvocation. EXIT	EXIT	SOURCE COMPONENT	문자열
		SOURCE INTERFACE	문자열
		SOURCE METHOD	문자열
		SOURCE MODULE	문자열
		SOURCE REFERENCE	문자열
		TARGET COMPONENT	문자열
		TARGET INTERFACE	문자열
		TARGET METHOD	문자열
TARGET MODULE	문자열		

이벤트 이름	이벤트 네이처	이벤트 콘텐츠	유형
WBI.SCA. MethodInvocation. FAILURE	FAILURE	SOURCE COMPONENT	문자열
		SOURCE INTERFACE	문자열
		SOURCE METHOD	문자열
		SOURCE MODULE	문자열
		SOURCE REFERENCE	문자열
		TARGET COMPONENT	문자열
		TARGET INTERFACE	문자열
		TARGET METHOD	문자열
		TARGET MODULE	문자열
		예외	문자열

## 선택기 이벤트

선택기 컴포넌트는 모니터링할 수 있는 단일 요소를 포함합니다. 각 이벤트에 고유한 확장 데이터 요소, 이벤트 이름 및 연관된 이벤트 네이처와 함께 이 요소에 대한 모든 이벤트 유형이 여기에 나열됩니다. 모든 선택기 이벤트에는 기본 이름, sel:WBI.SEL이 있습니다.

이벤트 이름	이벤트 네이처	이벤트 콘텐츠	유형
sel:WBI.SEL.ENTRY	ENTRY	operationName	문자열
sel:WBI.SEL.EXIT	EXIT	operationName	문자열
sel:WBI.SEL.FAILURE	FAILURE	ErrorReport	예외
		operationName	문자열
sel:WBI.SEL.SelectionKeyExtracted	SelectionKeyExtracted	operationName	문자열
sel:WBI.SEL.TargetFound	TargetFound	operationName	문자열
		target	문자열





---

## 주의사항

이 정보는 미국에서 제공되는 제품 및 서비스용으로 작성된 것입니다.

IBM은 다른 국가에서 이 책에 기술된 제품, 서비스 또는 기능을 제공하지 않을 수도 있습니다. 현재 사용할 수 있는 제품 및 서비스에 대한 정보는 한국 IBM 담당자에게 문의하십시오. IBM 제품, 프로그램 또는 서비스에 대한 모든 언급은 IBM 제품, 프로그램 또는 서비스를 사용할 수 있음을 의미하거나 암시하지 않습니다. IBM의 지적 재산권을 침해하지 않는 한, 기능상으로 동등한 모든 제품, 프로그램 또는 서비스를 대신 사용할 수도 있습니다. 그러나 비IBM 제품, 프로그램 또는 서비스의 작동을 평가 및 검증은 사용자의 책임입니다.

IBM이 이 책에서 다루고 있는 특정 내용에 대해 특허를 보유하고 있거나 현재 특허 출원 중일 수 있습니다. 이 책을 제공한다고 해서 특허에 대한 라이선스까지 부여하는 것은 아닙니다. 라이선스에 대한 의문사항은 다음으로 문의하십시오.

135-270

서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩

한국 아이.비.엠 주식회사

고객만족센터

전화번호: 080-023-8080

2바이트(DBCS) 정보에 관한 라이선스 문의는 한국 IBM 고객만족센터에 문의하거나 다음 주소로 서면 문의하시기 바랍니다.

*IBM World Trade Asia Corporation*

*Licensing*

*2-31 Roppongi 3-chome, Minato-ku*

*Tokyo 106-0032, Japan*

다음 단락은 현지법과 상충하는 영국이나 기타 국가에서는 적용되지 않습니다. IBM은 타인의 권리 비침해, 상품성 및 특정 목적에의 적합성에 대한 묵시적 보증을 포함하여(단, 이에 한하지 않음) 묵시적이든 명시적이든 어떠한 종류의 보증없이 이 책을 “있는 그대로” 제공합니다. 일부 국가에서는 특정 거래에서 명시적 또는 묵시적 보증의 면책사항을 허용하지 않으므로, 이 사항이 적용되지 않을 수도 있습니다.

이 정보에는 기술적으로 부정확한 내용이나 인쇄상의 오류가 있을 수 있습니다. 이 정보는 주기적으로 변경되며, 변경된 사항은 최신판에 통합됩니다. IBM은 이 책에서 설명한 제품 및/또는 프로그램을 사전 통지 없이 언제든지 개선 및/또는 변경할 수 있습니다.

이 정보에서 언급되는 비IBM의 웹 사이트는 단지 편의상 제공된 것으로, 어떤 방식으로든 이들 웹 사이트를 옹호하고자 하는 것은 아닙니다. 해당 웹 사이트의 자료는 본 IBM 제품 자료의 일부가 아니므로 해당 웹 사이트 사용으로 인한 위험은 사용자 본인이 감수해야 합니다.

IBM은 귀하의 권리를 침해하지 않는 범위 내에서 적절하다고 생각하는 방식으로 귀하가 제공한 정보를 사용하거나 배포할 수 있습니다.

(i) 독립적으로 작성된 프로그램과 기타 프로그램(본 프로그램 포함) 간의 정보 교환 및  
ii) 교환된 정보의 상호 이용을 목적으로 본 프로그램에 관한 정보를 얻고자 하는 라이선스 사용자는 다음 주소로 문의해야 합니다.

135-270

서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩

한국 아이.비.엠 주식회사

고객만족센터

이러한 정보는 해당 조건(예를 들어, 사용료 지불 등)하에서 사용될 수 있습니다.

이 정보에 기술된 라이선스가 부여된 프로그램 및 이 프로그램에 대해 사용 가능한 모든 라이선스가 부여된 자료는 IBM이 IBM 기본 계약, IBM 프로그램 라이선스 계약 또는 모든 동등한 계약 하에서 제공한 것입니다.

본 문서에 포함된 모든 성능 데이터는 제한된 환경에서 산출된 것입니다. 따라서 다른 운영 환경에서 얻어진 결과는 상당히 다를 수 있습니다. 일부 성능은 개발 단계의 시스템에서 측정되었을 수 있으므로 이러한 측정치가 일반적으로 사용되고 있는 시스템에서도 동일하게 나타날 것이라고는 보증할 수 없습니다. 또한 일부 성능은 추정을 통해 추측되었을 수도 있으므로 실제 결과는 다를 수 있습니다. 이 책의 사용자는 해당 데이터를 본인의 특정 환경에서 검증해야 합니다.

비IBM 제품에 관한 정보는 해당 제품의 공급업체, 공개 자료 또는 기타 범용 소스로부터 얻은 것입니다. IBM에서는 이러한 비IBM 제품을 반드시 테스트하지 않았으므로, 이들 제품과 관련된, 성능의 정확성, 호환성 또는 기타 주장에 대해서는 확인할 수 없습니다. 비IBM 제품의 성능에 대한 의문사항은 해당 제품의 공급업체에 문의하십시오.

IBM이 제시하는 방향 또는 의도에 관한 모든 언급은 특별한 통지 없이 변경될 수 있습니다.

이 정보에는 일상의 비즈니스 운영에서 사용되는 자료 및 보고서에 대한 예제가 들어 있습니다. 이들 예제에는 개념을 가능한 완벽하게 설명하기 위하여 개인, 회사, 상표 및 제품의 이름이 사용될 수 있습니다. 이들 이름은 모두 가공의 것이며 실제 기업의 이름 및 주소와 유사하더라도 이는 전적으로 우연입니다.

저작권 라이선스:

이 정보에는 여러 가지 운영 플랫폼에서의 프로그래밍 기법을 보여주는 원어로 된 샘플 응용프로그램이 들어 있습니다. 귀하는 이러한 샘플 프로그램의 작성 기준이 된 운영 플랫폼의 응용프로그램 프로그래밍 인터페이스(API)에 부합하는 응용프로그램을 개발, 사용, 판매 또는 배포할 목적으로 추가 비용 없이 이들 샘플 프로그램을 어떠한 형태로든 복사, 수정 및 배포할 수 있습니다. 이러한 샘플 프로그램은 모든 조건 하에서 완전히 테스트된 것은 아닙니다. 따라서 IBM은 이러한 프로그램의 신뢰성, 서비스 기능성 또는 기능을 보증하거나 진술하지 않습니다.

이러한 샘플 프로그램 또는 파생 제품의 각 사본이나 그 일부에는 반드시 다음과 같은 저작권 표시가 포함되어야 합니다. (c) (회사명) (연도). 이 코드의 일부는 IBM Corp. 의 샘플 프로그램에서 파생됩니다. (c) Copyright IBM Corp. \_연도\_. All rights reserved.

이 정보를 소프트카피로 확인하는 경우에는 사진과 컬러 삽화가 제대로 나타나지 않을 수도 있습니다.

## 프로그래밍 인터페이스 정보

프로그래밍 인터페이스 정보는 본 프로그램을 사용하는 응용프로그램 소프트웨어 작성을 돕기 위해 제공됩니다.

귀하는 범용 프로그래밍 인터페이스를 통해 본 프로그램 툴의 서비스를 제공하는 응용 프로그램 소프트웨어를 작성할 수 있습니다.

그러나 본 정보에는 진단, 수정 및 성능 조정 정보도 포함되어 있습니다. 진단, 수정 및 성능 조정 정보는 응용프로그램 소프트웨어의 디버그를 돕기 위해 제공된 것입니다.

**경고:** 본 진단, 수정 및 조정 정보는 변경될 수 있으므로 프로그래밍 인터페이스로서 사용될 수 없습니다.

## 상표 및 서비스표

IBM, IBM 로고, developerWorks, Tivoli, 및 WebSphere는 미국 또는 기타 국가에서 사용되는 IBM Corporation의 등록상표입니다.

Adobe는 미국 또는 기타 국가에서 사용되는 Adobe Systems Incorporated의 등록상표입니다.

Java 및 모든 Java 기반 상표는 미국 또는 기타 국가에서 사용되는 Sun Microsystems, Inc.의 상표입니다.

기타 회사, 제품 및 서비스 이름은 해당 회사의 상표 또는 서비스표입니다.

본 제품에는 Eclipse 프로젝트에서 개발한 소프트웨어가 포함되어 있습니다. (<http://www.eclipse.org> 웹 사이트 참조)



멀티플랫폼용 IBM WebSphere Process Server, 버전 6.1.0



**IBM**