# 오픈소스 빅데이터 개발 비즈니스 추진 사례

빅데이터 플랫폼 Flamingo를 통해 알아보는 성공적인 오픈소스 비즈니스 비법

빅데이터개발본부 | 김병곤 상무

# 한컴, 오픈소스 라이선스 위반으로 국제소송 휘말려

BY 심재석 *on* 2017년 5월 15일 • ( 0 )


한글과컴퓨터가 국제 소송에 휘말렸다. 미국 온라인 매체 쿼츠는 지난 11일(미국시각) 미국의 문서 소프트웨어 아티펙스(Arttifex)가 한컴을 일반공중라이선스(GPL) 위반으로 고소했다고 보도했다.

한컴은 지난 2013년부터 아티펙스의 오픈소스 기반 PDF 인터프리터 고스트스크립트(Ghostscript)를 한컴 오피스에 내장했다. 이는 한컴 오피스로 작성한 문서를 PDF 파일로 쉽게 변환할 수 있는 컴포넌트다.

문제는 고스트스크립트가 GPL을 따르고 있다는 점이다. GPL은 오픈소스 소프트웨어 라이선스의 한 종류로, GPL 소프트웨어를 사용하면 그것과 연결된 소프트웨어 소소코드도 오픈소스로 공개해야 하는 것이 특징이다. 대표적으로 리눅스 커널이 GPL을 따른다. 이 때문에 리눅스커널을 사용한 모든 리눅스 배포판은 소스코드가 공개돼 있다.

이 때문에 '고스트스크립트'를 사용한 한컴 오피스는 전체 소스코드를 공개해야 한다. 한컴 오피스가 오픈소스 소프트웨어가 되는 셈이다.

다만 아티펙스는 듀얼라이선스 제도를 마련해 두고 있었다. '고스트스크립트'를 유료로 구매한 회사는 GPL을 따르지 않아도 된다.

보도에 따르면, 한컴은 GPL을 따르지도, 비용을 지불하지도 않았다. 이 때문에 아티펙스는 2016년 말 캘리포니아주 북부 지방법원에 소송을 제기했다. 아티펙스는 한컴이 고스트스크립트 사용을 중단하고, 로열티를 지불할 것을 요구했다.

이에 대해 한컴 측은 〈바이라인네트워크〉에 "GPL을 위반하지 않았다"고 반박했다. 한컴 관계자는 "한컴은 GPL 의무사항에 대해 성실히 이행하였으나, 이행의 정도에 대해 양사(한컴과 아티펙스)간 해석차이가 있다"고 설명했다.

그러나 기자가 GPL에 따라 공개한 소스코드를 보여달라고 요청하자, "현재 소송이 진행중인 사안이라 불필요한 논란을 낳을 수 있다"면서 거부했다.

글. 바이라인네트워크
〈심재석 기자〉shimsky@byline.network

- GPL 라이선스인 Ghostscript를 한컴 오피스에 내장
- GPL 라이선스 위반
- Ghostscript 개발사인 Artifex가 소송
- 소송에서 패소 (협의만 남음)
- 여전히 한컴은 소스코드를 공개하지 않음

*“오픈 소스(open source)는 소프트웨어의 제작자의 권리를 지키면서 원시 코드를 누구나 열람할 수 있도록 한 소프트웨어 혹은 오픈 소스 라이선스에 준하는 모든 통칭을 일컫는다.”*

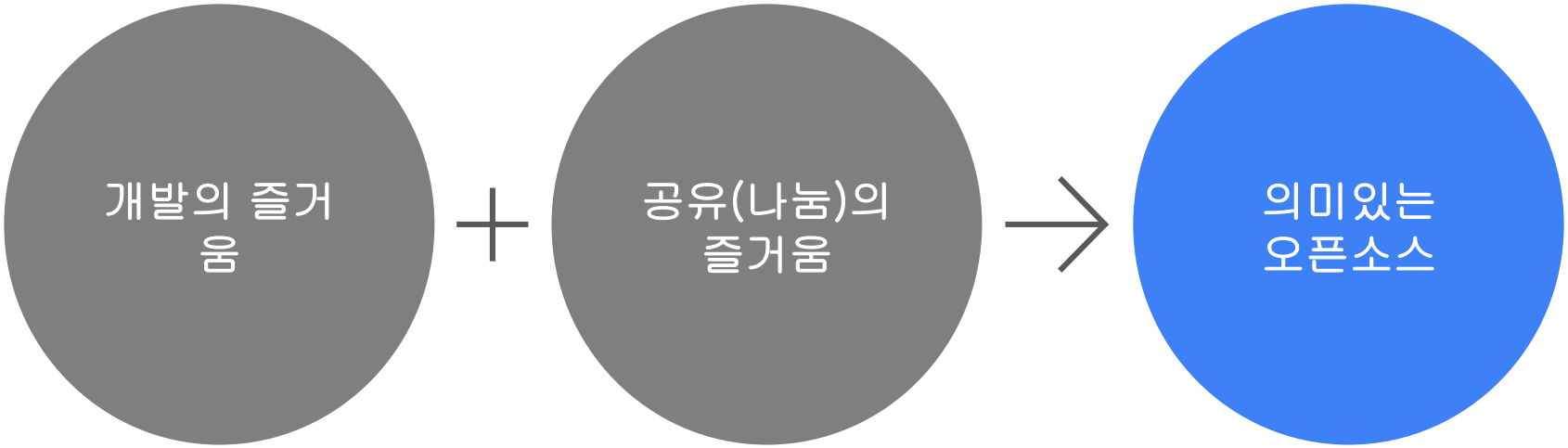
*– Wikipedia*

*http://korea.gnu.org/documents/copyleft/osd-korean.html*

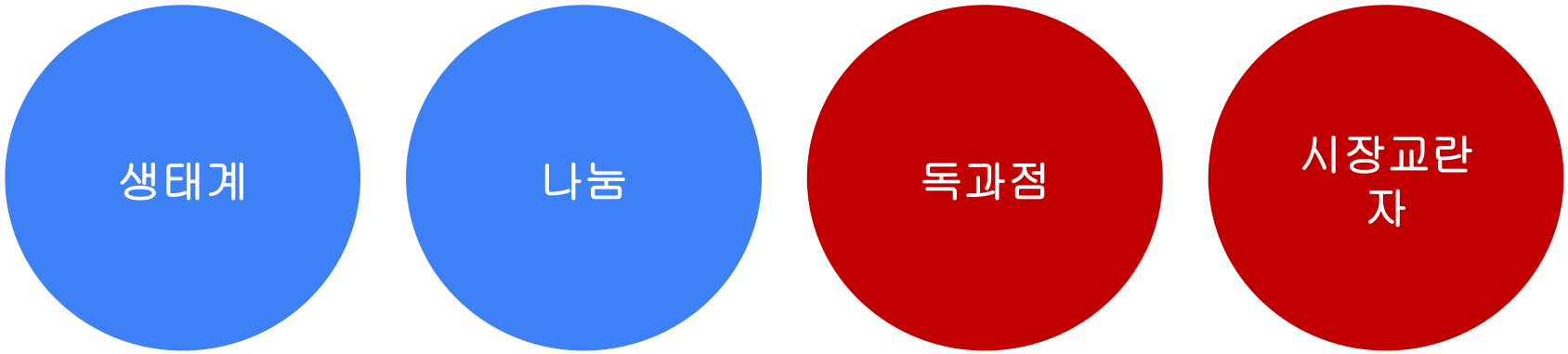
애플리케이션 및 서비스 등을  개발할 때

사용할 수 있는 무료 라이브러리!!

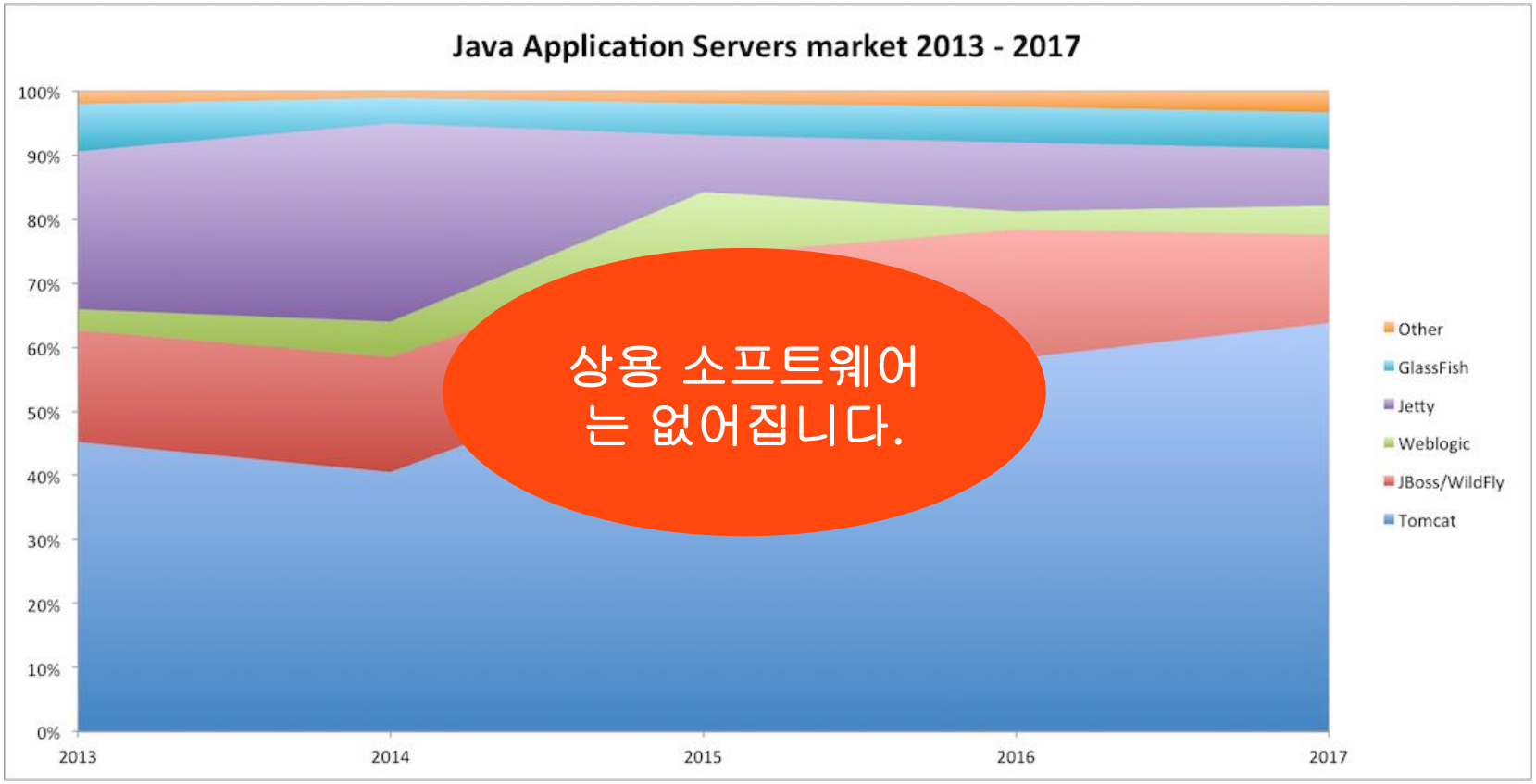
# 저작자 입장에서 오픈소스란 무엇인가?



# 오픈소스를 좀더 다른 방향에서 본다면?



# 시간이 지날 수록 해당 분야의 기술이 OLD 해지면 오픈소스가 그 자리를 채운다!




Plumbr – Most popular Java application servers: 2017 edition

# 오픈소스에 대한 다른 관점에서 생각해보기

- 오픈소스로 소스코드를 모두 공개하면 누구나 복제해서 사용할텐데 어떻게 먹고 살지?
  - 내가 만든 소스코드를 그대로 복제해서 이름만 바꾼다면?
- 오픈소스 라이선스의 규정에 따라서 사용자들은 어떤 감정을 느낄까?
  - Apache, GPL, LGPL, AGPL 등등
- 소스코드가 공개되어 있으니 커스터마이징이 자유로운데 이것이 문제가 될까?
  - 고객사에 소스코드를 주면서 기능을 자유롭게 추가해서 수정했을때…
- 상황에 따라서 다른 제약조건을 적용할 수 없을까?
  - 비영리, 학교 등의 조직에서는 무료, 영리 조직에서 영리적 목적의 경우 유료
  - 유료인 경우 과연 돈을 내기는 할까?
- 오픈소스는 꼭 코드가 대단히 잘 작성해야 하나?

# 오픈소스 라이선스에 대해서 잠깐 알아보기

| | 무료<br>이용가능 | 배포<br>허용가능 | 소스코드<br>취득가능 | 소스코드<br>수정가능 | 2차적 저작물<br>재공개 의무 | 독점SW와<br>결합가능 |
|---|---|---|---|---|---|---|
| GPL | O | O | O | O | O | X |
| LGPL | O | O | O | O | O | O |
| MPL | O | O | O | O | O | O |
| BSD license | O | O | O | O | X | O |
| Apache lilcense | O | O | O | O | X | O |

| LICENSE | BRANCHES | PERCENTAGE |
|---|---|---|
| GNU General Public License (GPL) | 30,374 | 65.84% |
| GNU Lesser General Public License (LGPL) | 3,017 | 6.54% |
| BSD License (original) | 1,354 | 2.94% |
| BSD License (revised) | 1,310 | 2.84% |
| Freeware | 1,079 | 2.34% |
| Freely Distributable | 980 | 2.12% |

# 오픈소스 프로젝트 시작시 개발팀 내부의 고민거리

Commercial Software를 개발하는 회사가 OpenSource를 추진했을 때 개발팀은 오픈소스 라이선스를 기반으로 프로젝트를 수행함에 있어서 다양한 고민거리가 발생합니다.

| | | | |
|---|---|---|---|
| 공개시기 | 개발부터 공개하고 할 것인가?<br>개발 완료후 공개할 것인가? | 커머셜 정책 | Subscription vs. On-Premise<br>부분 유료화, 전체 유료화<br>에디션<br>양심 |
| 라이센스 | 내가 비즈니스를 한다면 어떤 라이선스를 적용하는 것이 좋을가? | 소스코드 관리와 품질 | 더 많은 개발 조직<br>오픈소스 버전과 상용 버전의 관리<br>표준화 |

# 오픈소스 비즈니스의 핵심

# 오픈소스 라이선스 정책 정하기 (사례; Sencha Ext JS)

## Commercial Software License

This is the appropriate option if you want to use Ext JS to develop commercial applications whose source code you want to keep proprietary.

View our commercial SDK license agreement
Read more about our commercial software license

## Commercial OEM License

This is the appropriate option if you want to use Ext JS to create your own commercially licensed SDK, or web application builder. Since use cases vary widely, Commercial OEM licenses are customized for each customer.

Contact us to learn more

## Open Source License

Sencha is an avid supporter of open source software. Our open source license is the appropriate option if you are creating an open source application under a license compatible with the GNU GPL license v3. Although the GPLv3 has many terms, the most important is that you must provide the source code of your application to your users so they can be free to modify your application for their own needs.

Open Source FAQ
View the license terms

If you would like to use the GPLv3 version of Ext JS with your non-GPLv3 open source project, the following FLOSS (Free, Libre and Open Source) exceptions are available:

Open Source License Exception for Applications
Open Source License Exception for Development

# 오픈소스 라이선스 정책 정하기 (사례; MongoDB)

## MongoDB Licensing

### Software and Documentation

#### MongoDB Database Server and Tools

- Free Software Foundation's GNU AGPL v3.0.
- Commercial licenses are also available from 10gen, including free evaluation licenses.

#### Drivers

- mongodb.org supported drivers: Apache License v2.0.
- Third parties have created drivers too; licenses will vary there.

#### Documentation

- Documentation: Creative Commons.

# 오픈소스 라이선스 정책 정하기 (사례; Spring Framework)

Spring is modular in design, allowing for incremental adoption of individual parts such as the core container or the JDBC support. While all Spring services are a perfect fit for the Spring core container, many services can also be used in a programmatic fashion outside of the container.

Supported deployment platforms range from standalone applications to Tomcat and Java EE servers such as WebSphere. Spring is also a first-class citizen on major cloud platforms with Java support, e.g. on Heroku, Google App Engine, Amazon Elastic Beanstalk and VMware's Cloud Foundry.

The Spring Framework serves as the foundation for the wider family of Spring open source projects, including:

- Spring Security
- Spring Integration
- Spring Batch
- Spring Data
- Spring Web Flow
- Spring Web Services
- Spring Mobile
- Spring Social
- Spring Android

See the spring projects page for a full listing.

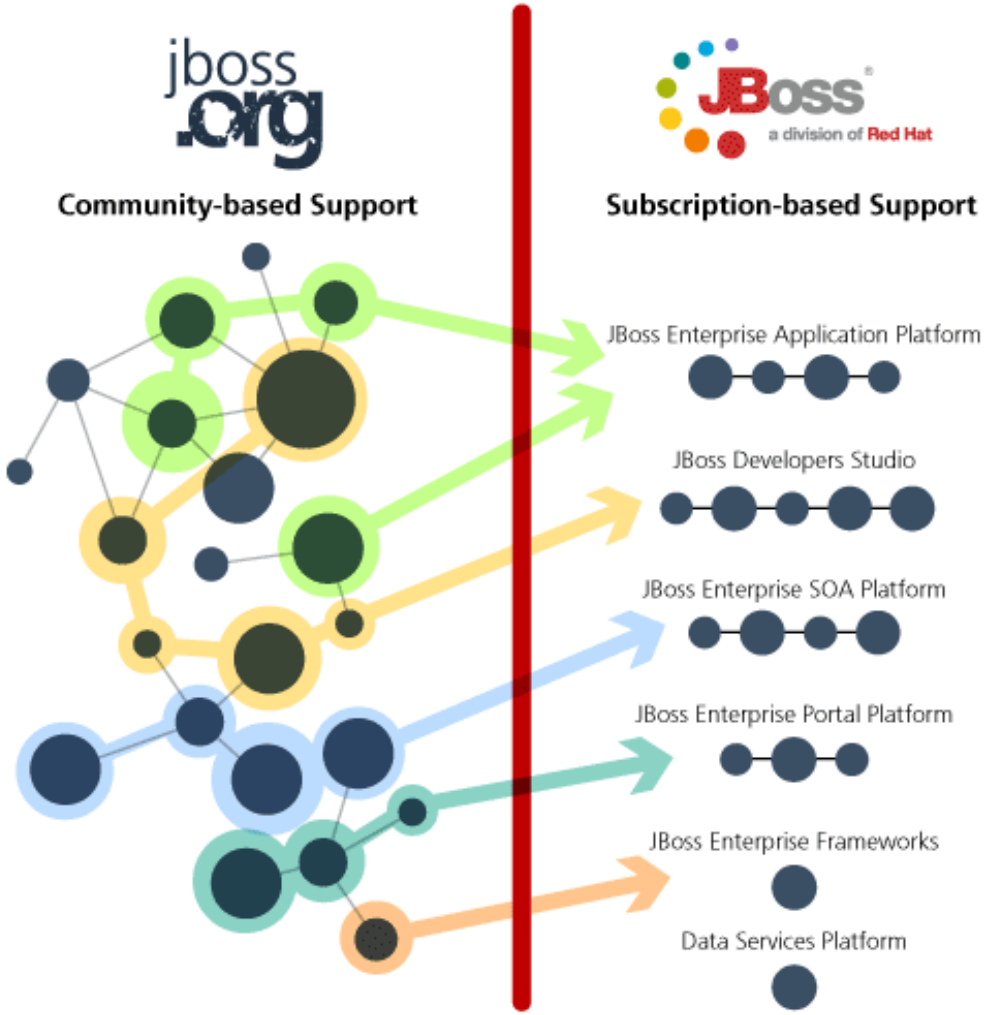
The Spring Framework is released under version 2.0 of the Apache License.

# 오픈소스를 팔기 위한 전략 수립 (고객에게 어려운 점이 바로 비즈니스 핵심 포인트)

각 분야별로 고객이 어려워 할 수 있는 것을 도출하고 그것을 오픈소스와 엮에서 비즈니스로 발전시킬 수 있습니다.

| | | | |
|---|---|---|---|
| UI | 기술문서, 예제, 버그에 대한 대응<br>확산시 사용자 수<br>강력한 기능성 | Framework | 아키텍처<br>개발 편의성 및 API<br>표준의 지원<br>개발도구 |
| Infra & Platform | 복잡한 구성<br>아키텍처<br>장애 및 유지보수 | DevTool | 다양한 개발 환경 지원<br>생산성<br>확성을 위한 플러그인 |

# 오픈소스의 파는 방법 정하기 (고객에게 어려운 점이 바로 비즈니스 핵심 포인트)



# 오픈소스의 파는 방법 정하기 (사례; Edition)

| JBoss Community | JBoss Enterprise |
| --- | --- |
| ● 업계 최고의 혁신적인 미들웨어 기술을 개발<br>● WIKI 또는 Forum을 이용한 지원<br>● 안정적인 배포판의 신속한 릴리즈<br>● 개인 사용자 대상(비용을 지불하지 않고 사용하고자 하는 개인, 교육용, 규모가 작은 사업자 등등)<br>● 최신 기술의 적용<br>● Red Hat/JBoss의 지원 | ● Bug Fix와 Patch를 적용한 안정적인 바이너리 제공(3개월 누적 패치 사이클)<br>● 5년 동안 지원<br>● 기업 고객 대상<br>● 기술지원<br>● 운영 수준의 품질을 보장하는 배포판<br>● 다양한 환경에서 테스트하여 성능과 안정성을 보장(17 OS, 5 DB, 다양한 JVM) |

# 오픈소스의 파는 방법 정하기 (사례; Edition)

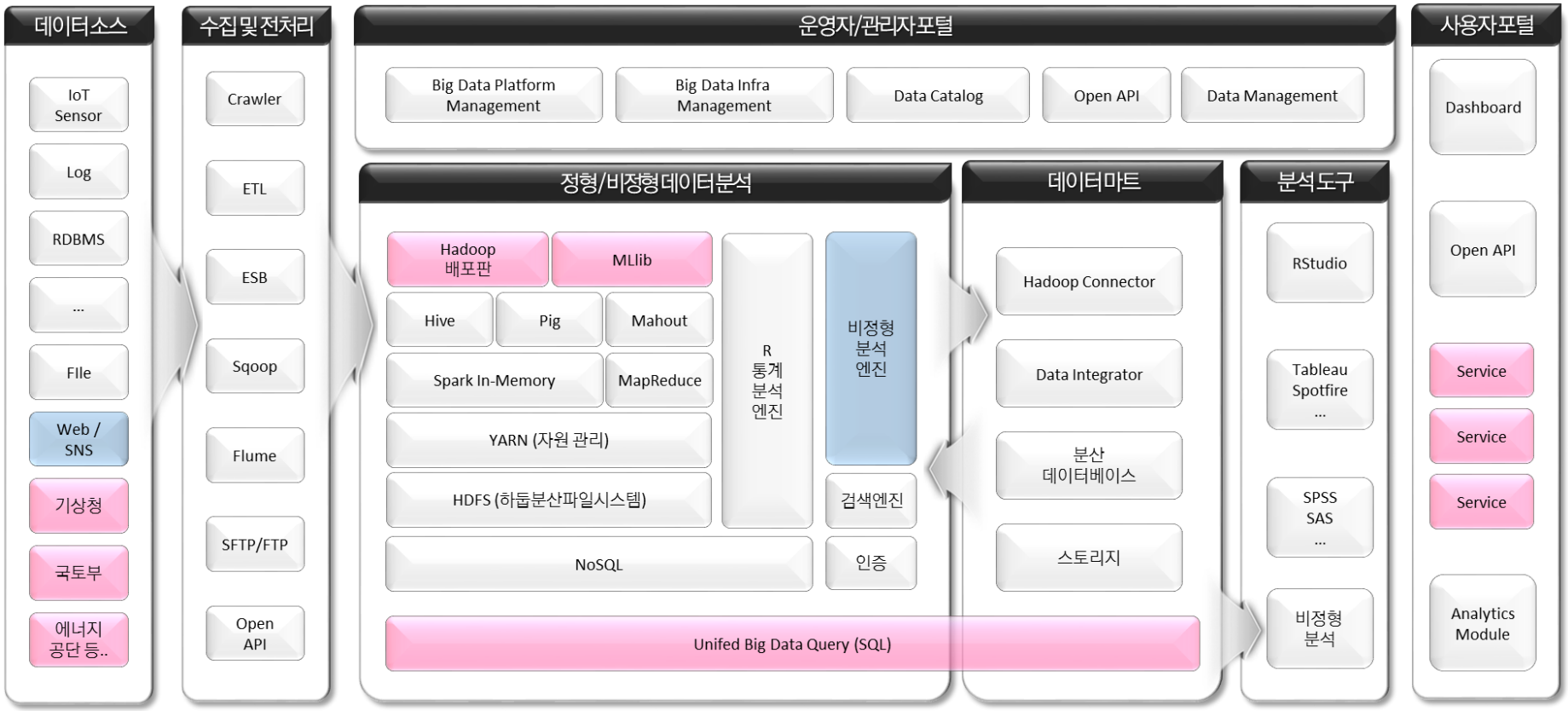
| 서비스 | JBoss Community | JBoss Enterprise |
|---|---|---|
| 오픈 소스 | X | X |
| 전세계 커뮤니티에 의한 테스트의 이점 | X | X |
| 패치 업데이트 및 서비스 팩 프로그램 | | X |
| 보안 Errata 프로그램 | | X |
| Hot Fix 프로그램 | | X |
| 자동 소프트웨어 업데이트 및 경고 서비스 | | X |
| 버그 에스컬레이션(Escalation) 프로세스 | | X |
| 24x7(연중무휴, 일일 24시간) 프로덕션 지원 및 상담 서비스 | | X |
| 플랫폼 인증 및 교육 인증 | | X |
| 정의된 지원 SLA 및 End-of-Life 정책 | | X |
| 엔터프라이즈용 Out-of-the-Box 구성 | | X |
| JBoss ON(Operations Network) 포함 | | X |
| JBoss ON 모니터링 가능 | | X |
| 광범위한 용도의 내/외부 테스트 | | X |
| 패치가 적용된 JBoss 배포판의 재배포 | | X |

# 오픈소스의 파는 방법 정하기 (서비스 레벨)

| 구분 | Developer Professional | Developer Enterprise | Production Standard | Production Premium |
|---|---|---|---|---|
| 지원시간 | 월요일~금요일 9AM~5PM | 24X7 | 월요일~금요일 9AM~5PM | 24X7 |
| 응답시간 | 2일 | 4일 | 4시간 | 1시간 |
| 지원방법 | 웹/전화지원 | 웹/전화지원 | 웹/전화지원 | 웹/전화지원 |

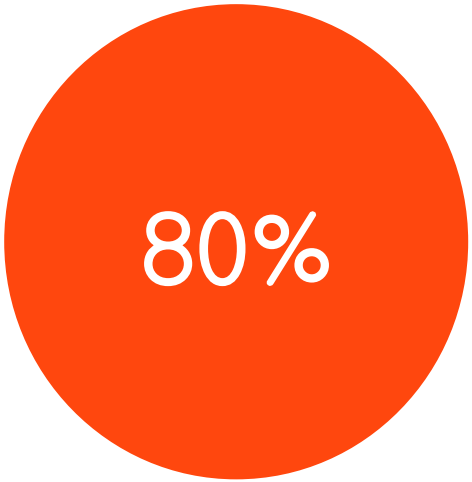
# 복잡한 빅데이터 플랫폼 아키텍처

- 데이터 수집/저장/전처리/분석/시각화/관리 등의 일련의 생명주기를 다루는 빅데이터 플랫폼은 상당히 다양한 소프트웨어와 복잡한 시스템 구성을 하게 됨



# 국내 빅데이터의 약 80%이상이 Hadoop, R 등의 오픈소스 기반

국내 빅데이터의 대부분은 오픈소스인 R, Hadoop EcoSystem 등을 기반으로 구축이 되어 있으며 이를 운영, 관리, 분석하기 위한 빅데이터 플랫폼을 구축하는 사업이 상당수임



# 빅데이터 플랫폼 관리시 어려운 점

빅데이터 플랫폼은 오픈소스인 Hadoop, Spark, Hive, R 및 상용 솔루션의 통합으로 구성되어 있어서 상대적으로 과거의 시스템 대비 복잡합니다.

| | | | |
|---|---|---|---|
| 플랫폼의 복잡도 | 다양한 오픈소스와 시스템으로 구성되어 복잡도가 매우 높아서 유지보수가 쉽지 않음 | 많은 수의 서버로 구성된 클러스터 | 많은 수의 서버로 구성되므로 운영자가 관리하는데 어려움을 느낌 |
| 분산 처리 | 분석 작업이 여러 대의 서버에서 나누어서 분산 처리를 하므로 동작 상황을 파악하기 어려움 | 다양한 오픈소스의 활용 | R, Hadoop, Spark, Hive 등 다양한 오픈소스가 포함되어 있으며 에러 메시지에 대한 파악이 어려움 |

# 빅데이터 플랫폼 관리를 단순화 하기 위한 방안

빅데이터 플랫폼은 오픈소스인 Hadoop, Spark, Hive, R 및 상용 솔루션의 통합으로 구성되어 있어서 상대적으로 과거의 시스템 대비 복잡하므로 이를 위한 최적화된 솔루션을 제공해야 합니다.

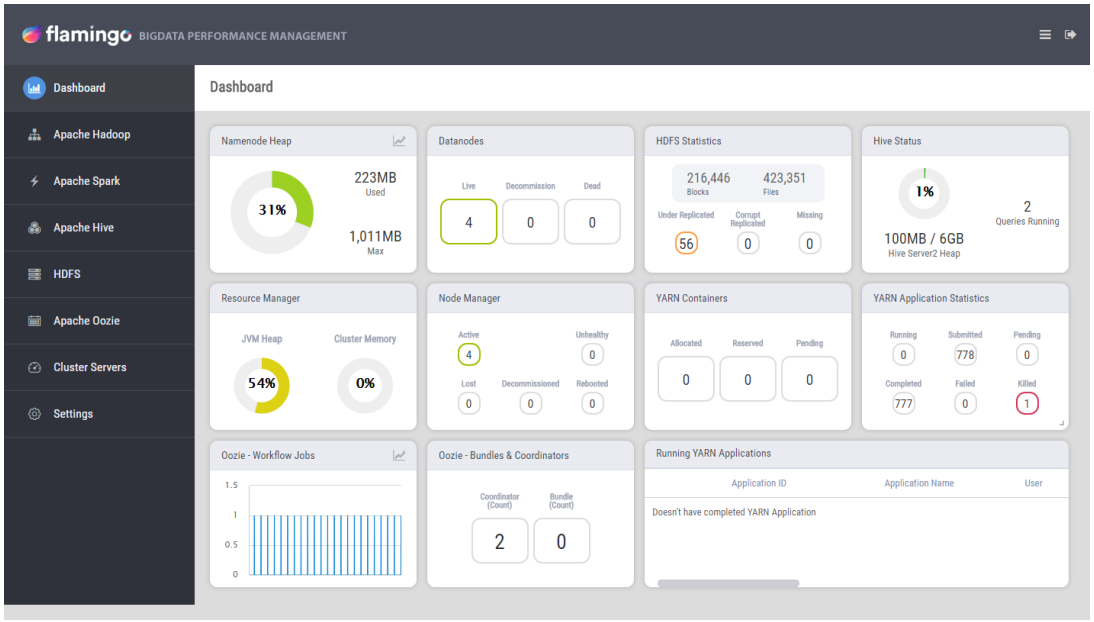
| | | | |
|---|---|---|---|
| 플랫폼의 복잡도 | 운영자를 위한 강력한 관리 기능을 제공해야 함 | 많은 수의 서버로 구성된 클러스터 | 모든 노드의 정보를 한눈에 볼 수 있도록 하고 요약 정보를 제공해야 함 |
| 분산 처리 | 분산 처리를 하는 분석 애플리케이션의 사용 현황을 손쉽게 관리할 수 있도록 해야 함 | 다양한 오픈소스의 활용 | 관점에 따라서 다른 뷰를 제공해야 함 |

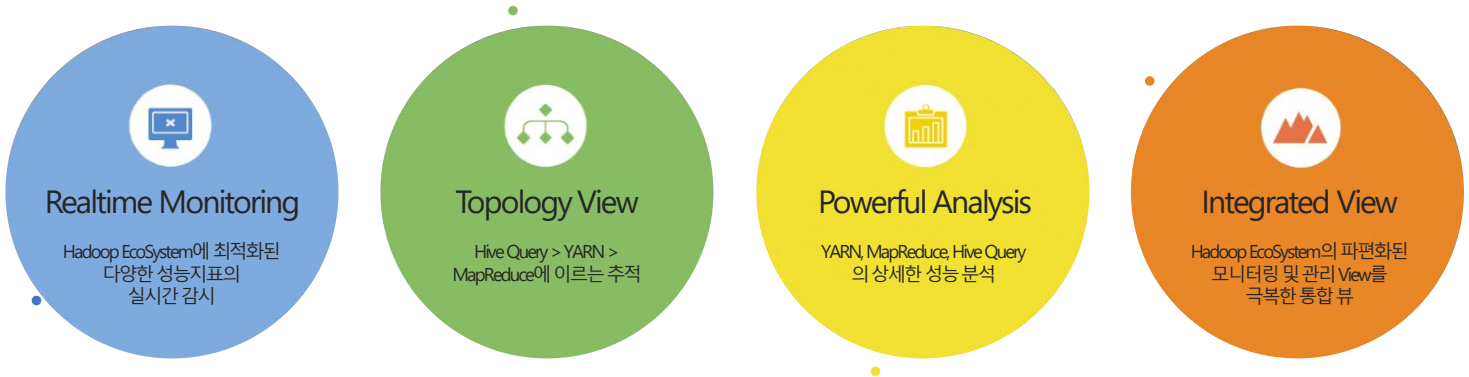
# Flamingo Big Data Platform의 오픈소스 비즈니스 전략

| | | |
|---|---|---|
| Big Data Market | Hadoop EcoSytem | Big Data Experience |
| UI Usability | All In One | Dual License |

# 빅데이터 플랫폼의 분석 및 성능 관리를 해결하는 솔루션 – FLAMINGO

Flamingo는 빅데이터 플랫폼의 분석 및 성능 관리 문제를 해결하는 유일한 솔루션입니다.



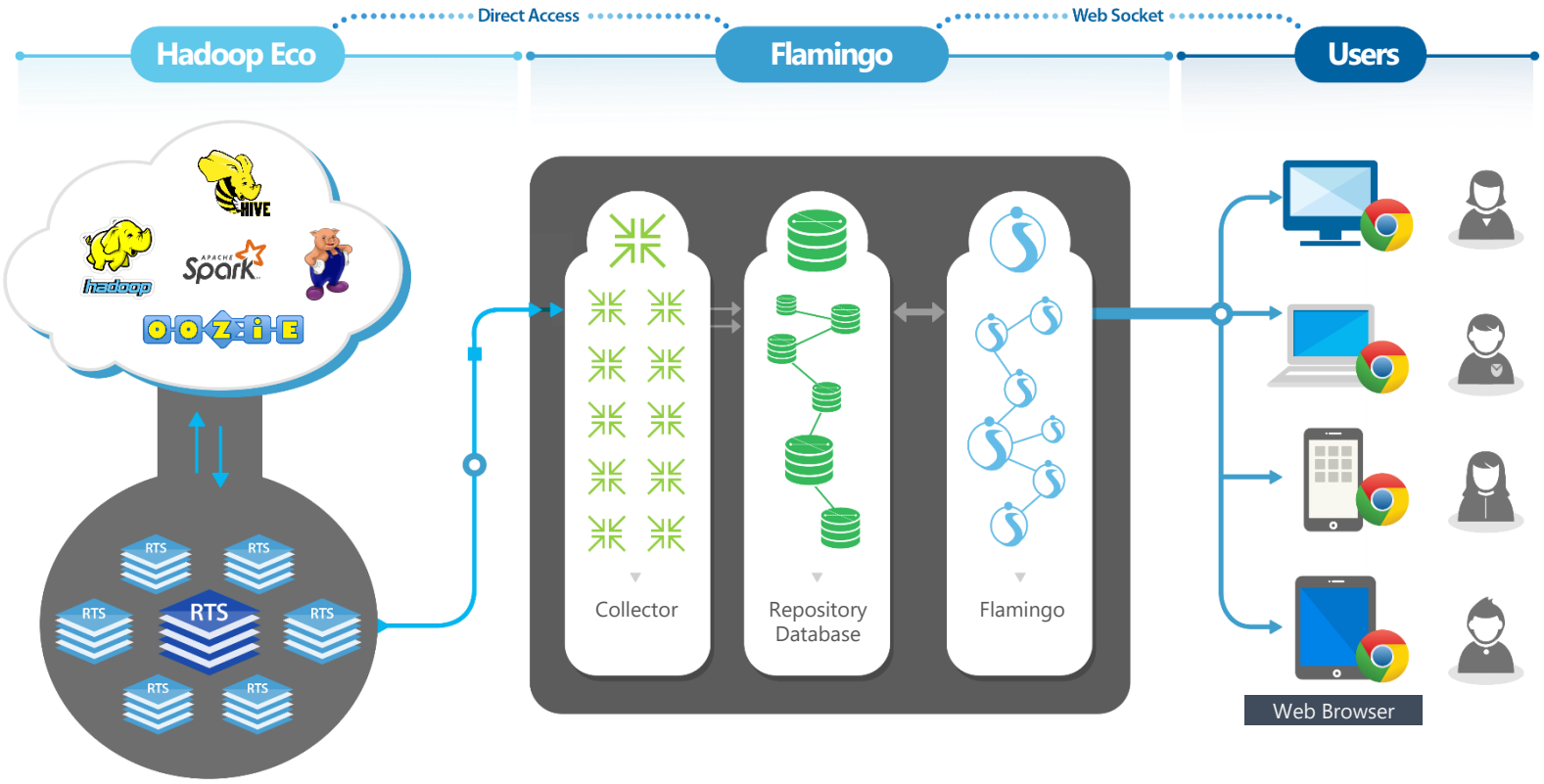



# Flamingo는 Apache License 2 라이선스를 준수하는 오픈소스

Flamingo는 성능 관리 기능과 분석 기능이 통합되어 3.0으로 다시 세상에 오픈소스로 공개 됩니다.

# Flamingo의 Architecture

Flamingo는 빅데이터 플랫폼의 분석 및 성능 관리 문제를 해결하는 유일한 솔루션입니다.



# 다양한 Hadoop 배포판 지원

Flamingo는 다양한 Hadoop 배포판을 지원하여 호환성이 뛰어납니다.


cloudera


Pivotal™

# Flamingo의 레퍼런스

오픈소스이지만 국내 많은 공공/민간 기업들이 빅데이터 플랫폼 구축에 활용하고 있습니다.





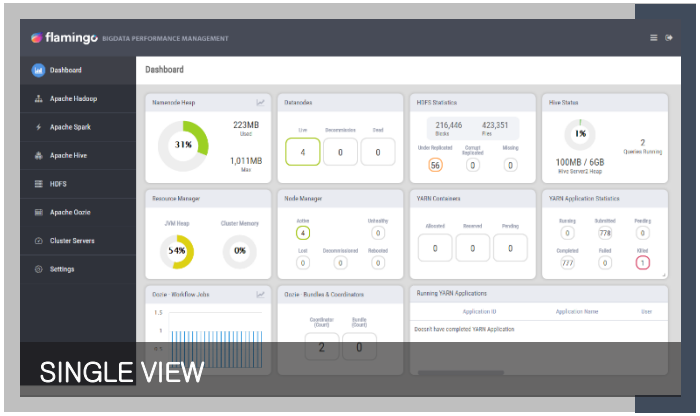




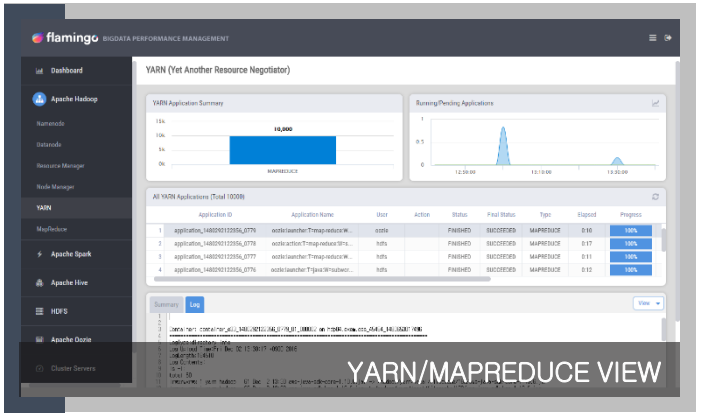



# 다양한 관점에 최적화된 뷰 제공

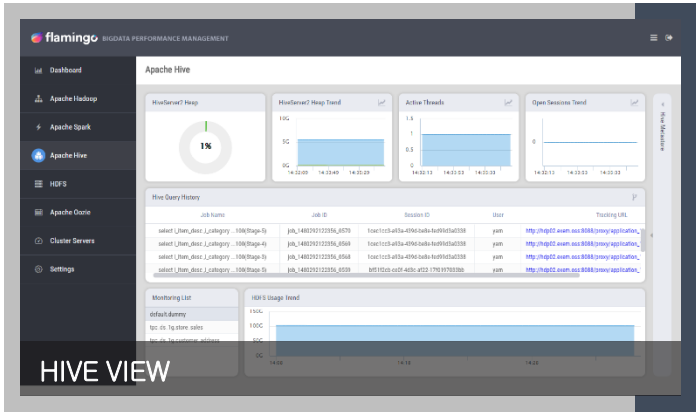
다양한 사용자 관점에 따라서, 모니터링 관점에 따라서 최적화된 뷰를 제공합니다.



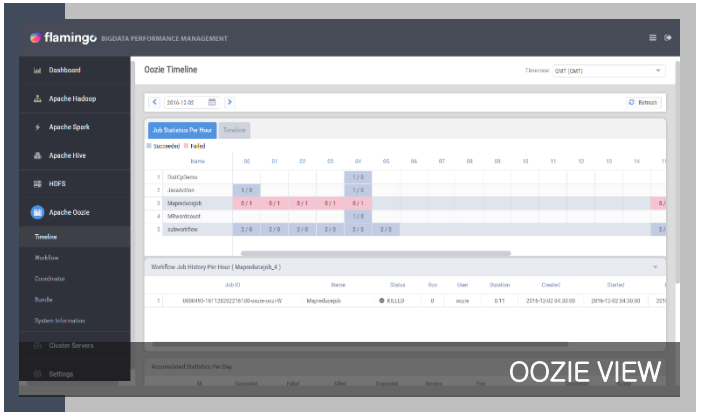

Hadoop Cluster 핵심 모니터링 지표에 대한 집중적인 모니터링




YARN/MAPREDUCE 핵심 관리 항목의 모니터링에 최적화




HIVE 핵심 관리 항목의 모니터링에 최적화

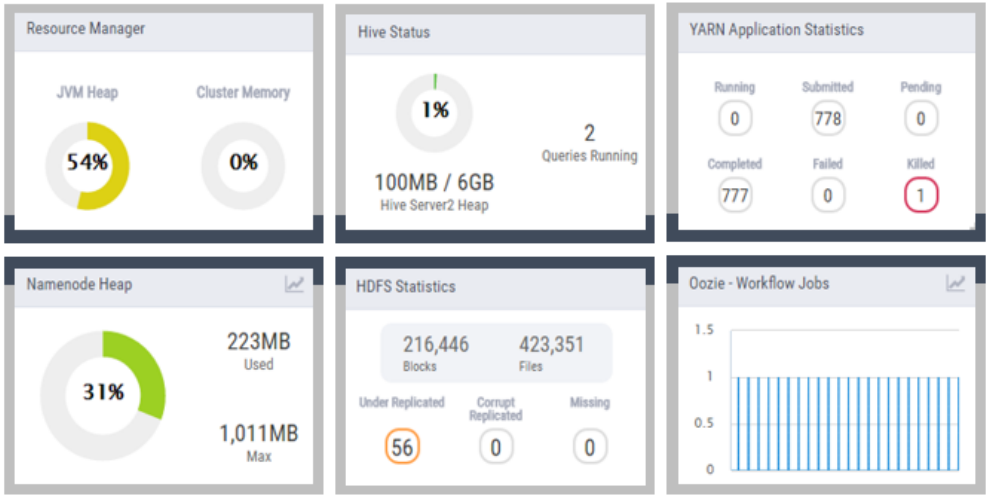



OOZIE 핵심 관리 항목의 모니터링에 최적화

# Hadoop Cluster 종합 감시를 위한 전용 화면 제공

Flamingo는 Hadoop Cluster의 중요한 각종 지표를 한눈에 알아볼 수 있도록 전용 화면을 제공합니다. 모니터링 지표들을 통해 운영자는 빠르고 쉽게 문제를 해결해 나갈 수 있습니다.

제품 화면 상세




View Type

## DASHBOARD VIEW

DASHBOARD View를 통해 Hadoop Cluster의 핵심 지표(**Resource Manager, Namenode, HDFS, Hive, YARN, Node Manager, Oozie** 등) 모니터링에 대해서 쉽게 파악할 수 있습니다.

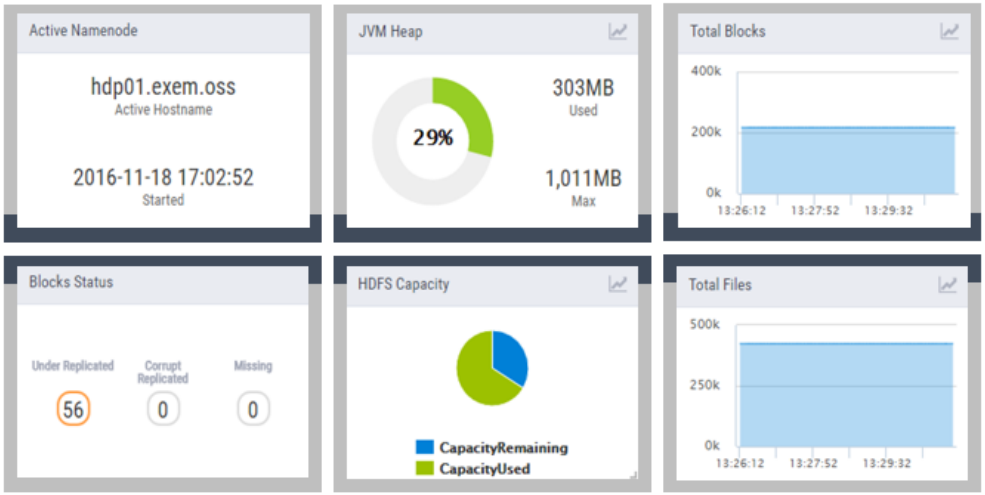
**Feature**

- 10초 주기 실시간 모니터링 정보 수집 (Polling이 아닌 Pushing 데이터 수집을 통한 정확성 확보 및 누락 데이터 방지)
- Namenode 메모리, Datanode 상태, HDFS 상태
- Hive Server 상태
- Resource Manager 및 Node Manager 상태
- YARN Container 및 YARN Application 상태
- Oozie Workflow 및 Coordinator 상태

# 대용량 데이터를 관리하는 Namenode의 핵심 지표를 모니터링

Namenode는 Hadoop Cluster의 분산 파일 시스템인 HDFS를 관리하는 핵심 서비스로서 Flamingo는 Namenode의 장애와 관련 지표를 실시간 모니터링합니다.

제품 화면 상세




View Type

## NAMENODE VIEW

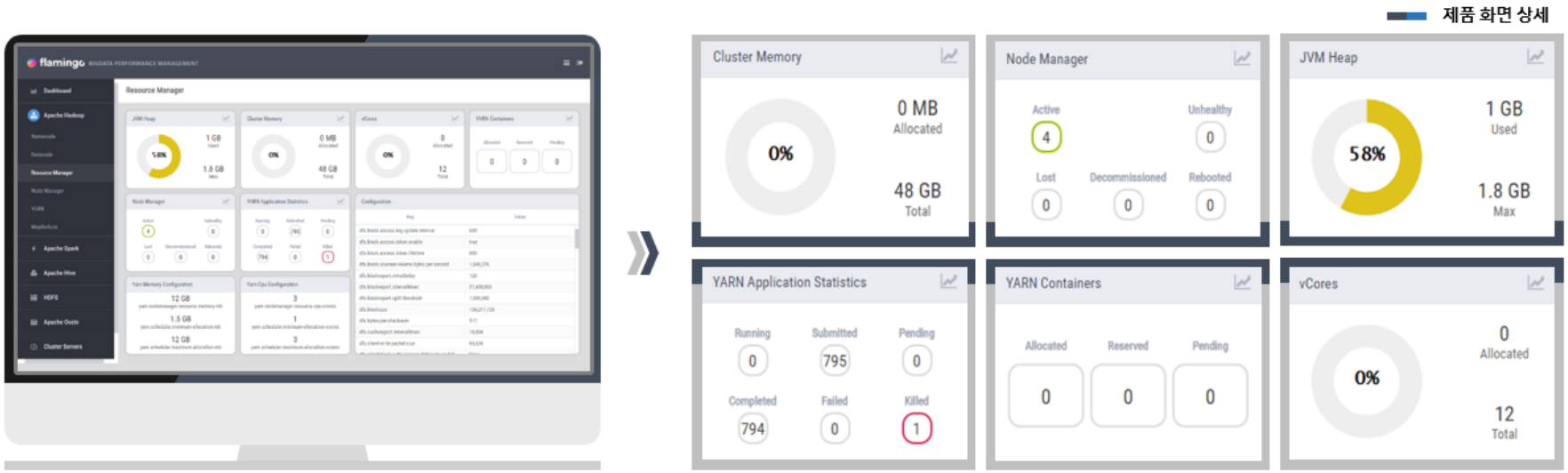
분산 파일 시스템을 관리하는 Namenode에 대해서 가장 중요한 성능 지표 (**JVM Heap, Active Namenode, HDFS Capacity, Block Status**) 모니터링에 대해서 쉽게 파악할 수 있습니다.

**Feature**

- Hadoop Namenode에서 Collector로 실시간 Pushing을 통해 수집하므로 데이터의 정확성 및 실시간성 보장
- Namenode HA시 Active Namenode에 상태 표시
- HDFS Metadata 수 증가시 JVM Heap 부족으로 인한 장애 모니터링
- Block 상태 모니터링을 통해 운영자가 HDFS에 파일 깨짐에 대응
- HDFS의 용량 및 Block/File 개수의 실시간 모니터링

# 분석 애플리케이션을 관리하는 Resource Manager의 핵심 지표를 모니터링

Resource Manager는 Hadoop Cluster에서 분석 작업을 수행하기 위해서 각 서버의 자원을 종합 관리하는 핵심 서비스입니다.




View Type

## RESOURCE MANAGER VIEW

RESOURCE MANAGER는 YARN Cluster의 전체 리소스를 관리하고 NODE MANAGER를 관리하는 주체입니다. Flamingo는 RESOURCE MANAGER에 핵심 모니터링 지표를 모니터링하여 관리자가 애플리케이션 튜닝을 위한 정보를 손쉽게 파악할 수 있습니다.

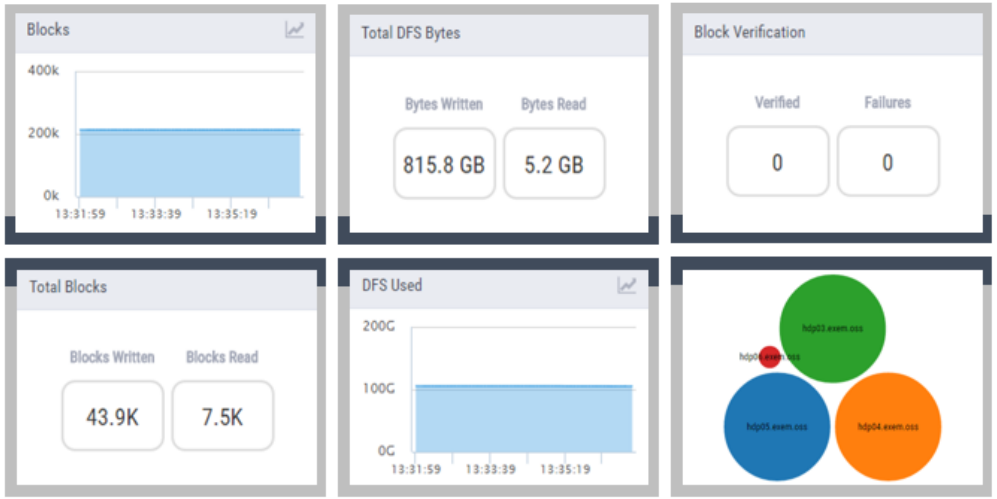
**Feature**

- 10초 주기 실시간 모니터링 정보 수집 (Polling이 아닌 Pushing 데이터 수집을 통한 정확성 확보 및 누락 데이터 방지)
- YARN Cluster 메모리 사용량
- vCore, Container 사용량
- Node Manager 정상 동작 여부
- YARN Application 상태
- YARN 설정 정보

# 각 서버의 데이터를 관리하는 데이터 노드의 핵심 지표를 모니터링

Datanode는 Hadoop Cluster에서 실제로 데이터를 저장하고 관리하고 분석 작업을 실행하는 핵심 노드입니다.

제품 화면 상세




View Type

## DATANODE VIEW

다수의 노드로 구성되는 DATANODE는 파일의 블록을 저장하고 분석 작업을 수행하는 노드로써 Flamingo에서는 각 노드별 지표를 확인할 수 있도록 지원하여 특정 데이터 노드의 문제점을 파악할 수 있습니다.

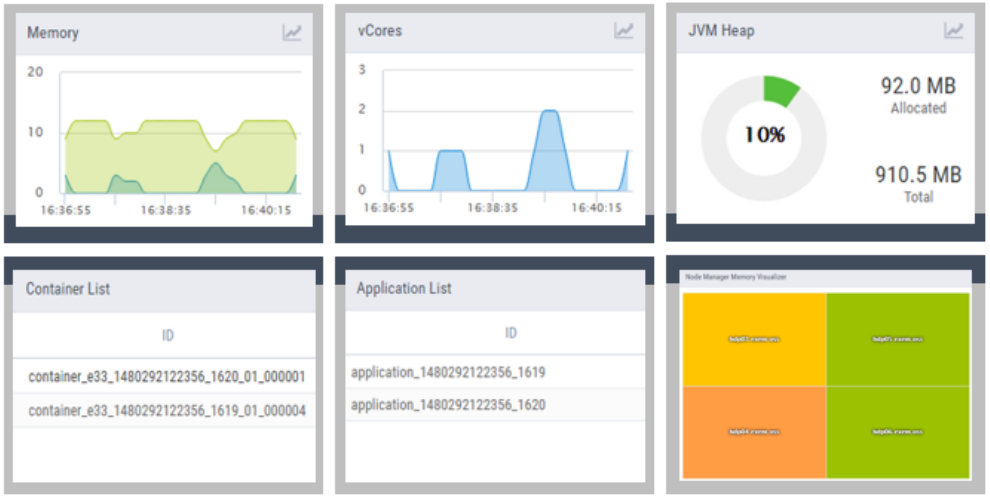
**Feature**

- 10초 주기 실시간 모니터링 정보 수집 (Polling이 아닌 Pushing 데이터 수집을 통한 정확성 확보 및 누락 데이터 방지)
- 데이터 노드의 블록, 분산 파일 시스템 사용량에 대한 분포도
- 노드별 블록 개수
- DFS 크기, DFS 사용량 표시
- Namenode와 통신 상태

# 각 노드의 Core, RAM 등의 자원을 관리하는 Node Manager의 핵심 지표를 모니터링

Node Manager는 각 노드의 자원을 관리하는 핵심 서비스로 Flamingo는 분석 애플리케이션 실행시 각 노드에 배정된 메모리 및 코어를 모니터링할 수 있도록 하여 자원의 최적화 분배 상황을 실시간으로 파악할 수 있게 해줍니다.

제품 화면 상세


View Type

## NODE MANAGER VIEW

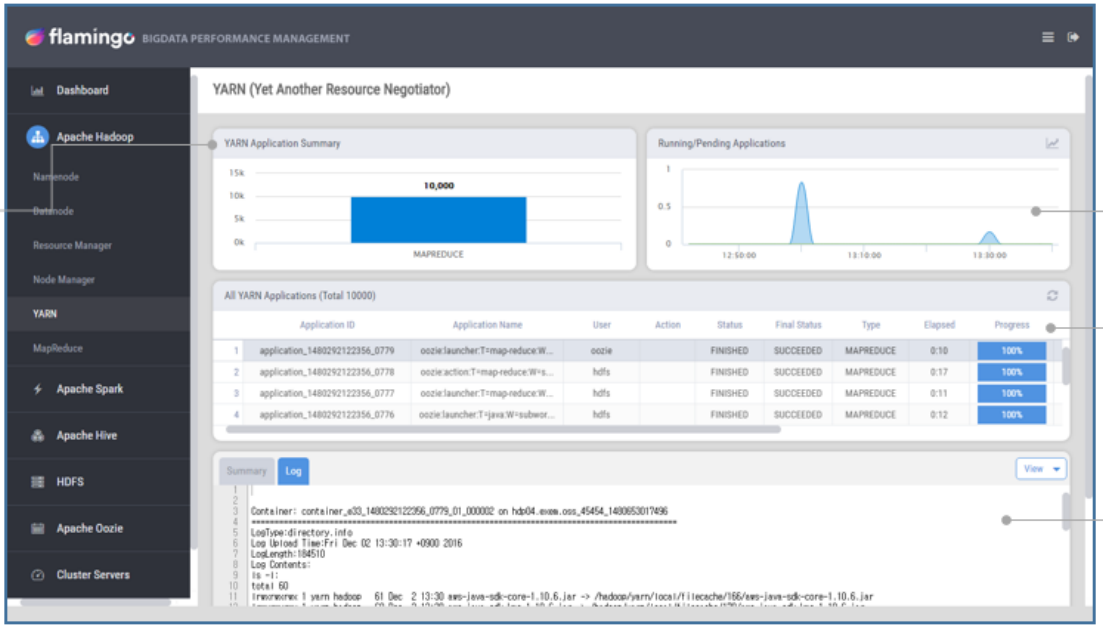
NODE MANAGER는 RESOURCE MANAGER의 요청에 따라서 컨테이너를 실행하고 vCore, RAM을 할당하여 작업을 모니터링합니다. FLAMINGO에서는 각 노드의 자원을 노드별로 사용량을 확인하고

**Feature**

- 10초 주기 실시간 모니터링 정보 수집 (Polling이 아닌 Pushing 데이터 수집을 통한 정확성 확보 및 누락 데이터 방지)
- 각 노드별 YARN Application 자원 분배 현황 확인
- 각 노드별 Application, Container 목록 확인
- vCore, RAM 사용량 확인
- JVM Heap Size 확인

# YARN의 분석 애플리케이션 모니터링

YARN은 분석 애플리케이션의 자원을 분배하고 모니터링하는 주요 서비스로 Flamingo에서는 관리자가 수작업으로 관리해야하는 대부분의 작업을 UI를 통해서 관리할 수 있도록 해주어 생산성을 향상시키고 모니터링을 가시화 시킵니다.




View Type

## YARN VIEW

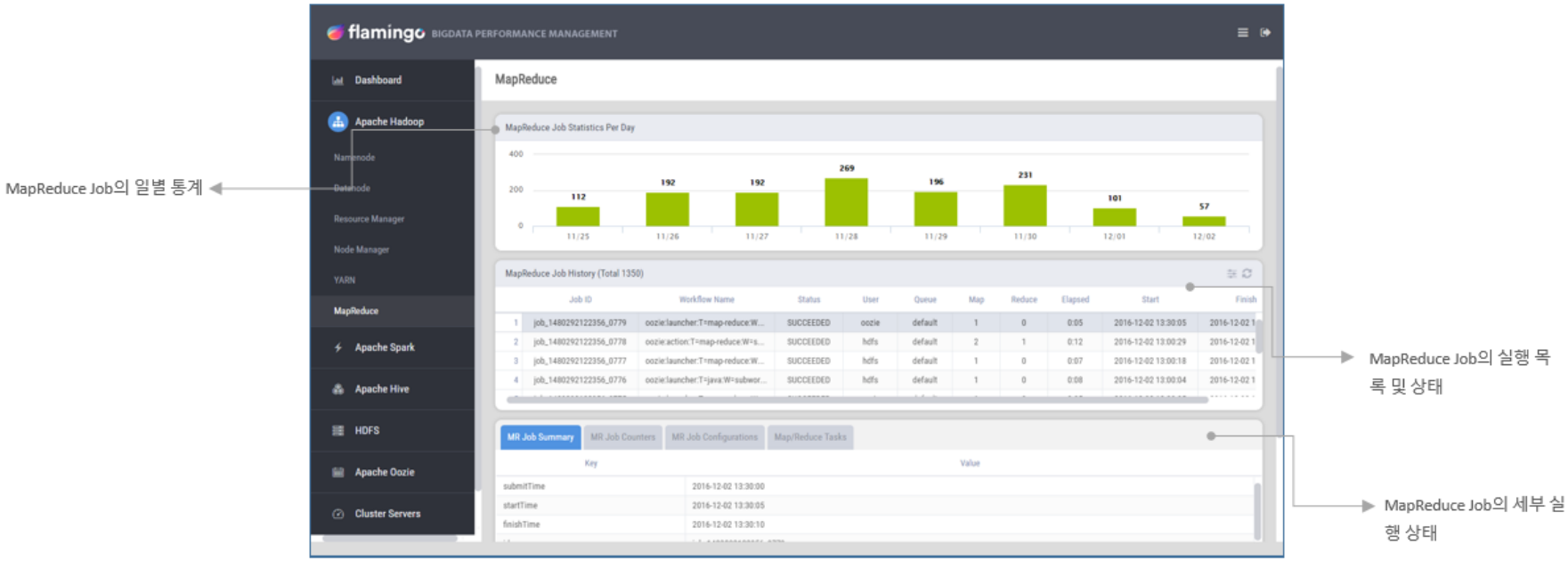
YARN은 Hadoop 2에서 자원을 관리하는 핵심 서비스로 FLAMINGO에서는 YARN을 관리하고 개발하고 분석하는데 있어서 가장 중요한 로그를 볼 수 있도록 하며, YARN 실행 이력을 표시하여 관리자들이 애플리케이션을 손쉽게 모니터링할 수 있습니다.

**Feature**

- 애플리케이션 유형별 통계 (예; MAPREDUCE, TEZ, SPARK 등)
- 실행중/지연중인 YARN Application 추이 그래프
- 실행중 및 완료된 YARN 애플리케이션 목록 및 각종 정보
- YARN 애플리케이션의 실행 로그
- YARN 애플리케이션의 실행에 대한 요약정보

# MapReduce Job 모니터링

MapReduce는 분산/병렬 처리 프레임워크로 Flamingo에서는 실행 통계, MapReduce Job 상세 모니터링을 제공하여 MapReduce Job의 문제를 파악하고 이를 시각화 합니다.




View Type

## MAPREDUCE VIEW

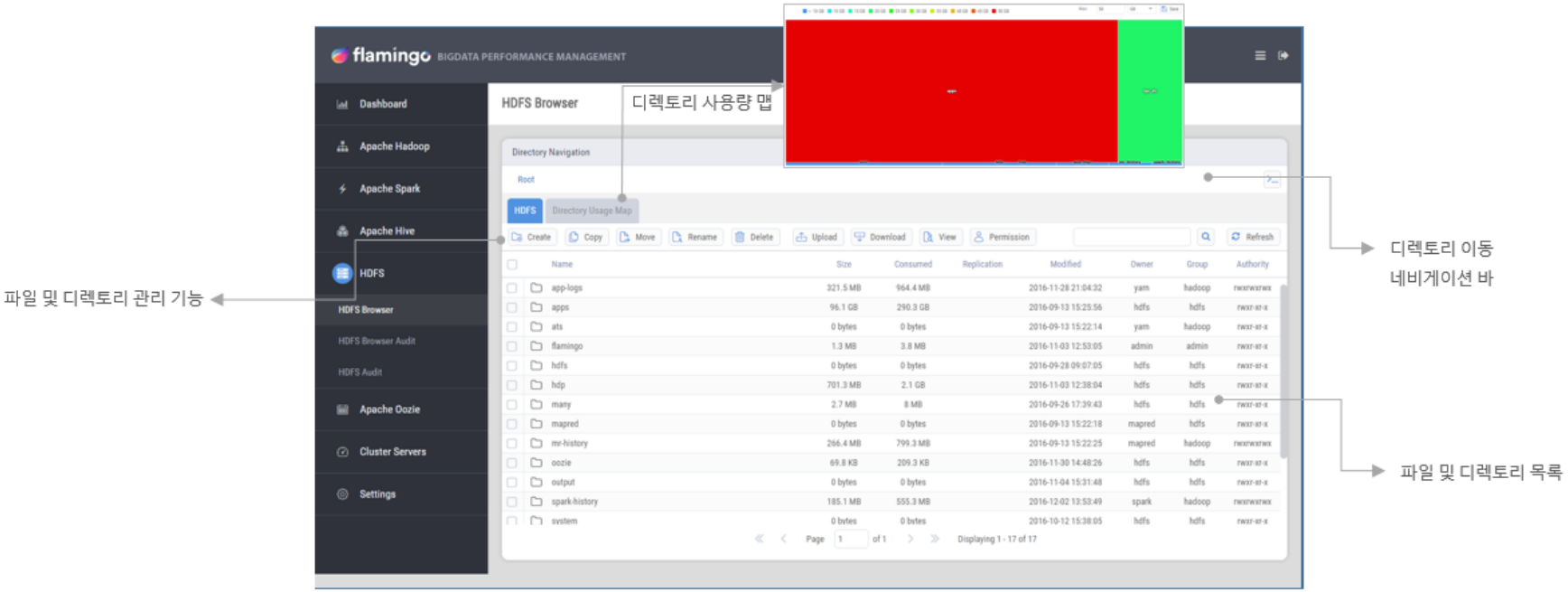
MAPREDUCE는 Map과 Reduce로 구성되어 있으며 분석 작업을 구현하는 프레임워크입니다. FLAMINGO는 Map, Reduce Task에 대한 로그 정보에서 부터 Timeline에 이르는 등, 관리자가 MapReduce Job 모니터링을 위해서 필요한 핵심 지표를 모니터링 할 수 있도록 지원합니다.

**Feature**

- MapReduce Job의 일별 통계
- MapReduce Job의 실행 이력
- MapReduce Job의 Summary 정보 (실행 시간, 구간별 처리 시간 상태, MR Task 개수등)
- MapReduce Job Counter Tree
- MapReduce Job Configuration
- MapReduce Job Task 목록 및 Task별 실행 로그
- MapReduce Job Task의 Timeline

# 분산 파일 시스템 관리

Hadoop의 HDFS는 분산 파일 시스템으로서 Flamingo에서는 UI를 통해 디렉토리 및 파일을 한번에 관리할 수 있는 강력한 HDFS 브라우저를 제공합니다.




View Type

## HDFS BROWSER VIEW

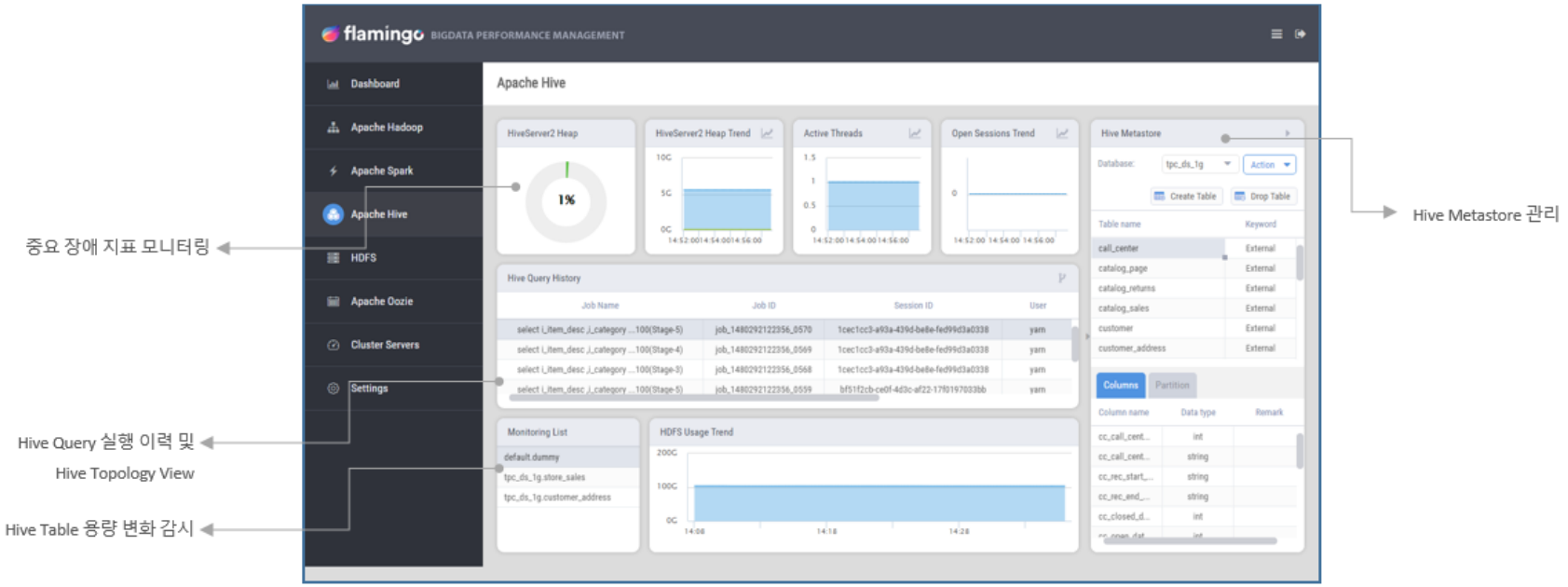
HDFS 브라우저는 HDFS 상의 디렉토리 및 파일을 관리하는 핵심 기능으로 FLAMINGO의 핵심 기능입니다. 파일을 업로드하고 다운로드할 수 있으며, 수십만개의 파일도 거뜬히 표시합니다.

**Feature**

- HDFS의 파일 및 디렉토리 관리
- 하나의 디렉토리에 수십만개의 파일이 있더라도 UI 문제없이 페이징
- 디렉토리 및 파일의 권한 관리
- 파일 내용 보기 및 파일 업로드/다운로드
- 문자열 기반 디렉토리 경로 입력시 해당 디렉토리로 이동
- 디렉토리별 사용량 표시 맵

# Hive 모니터링 및 테이블 관리

Hive는 SQL을 기반으로 데이터를 분석할 수 있는 Hadoop EcoSystem의 핵심 분석 도구로써 Flamingo에서는 테이블 용량, 메모리, 테이블/데이터베이스를 관리할 수 있는 핵심 기능을 제공합니다.




View Type

## HIVE VIEW

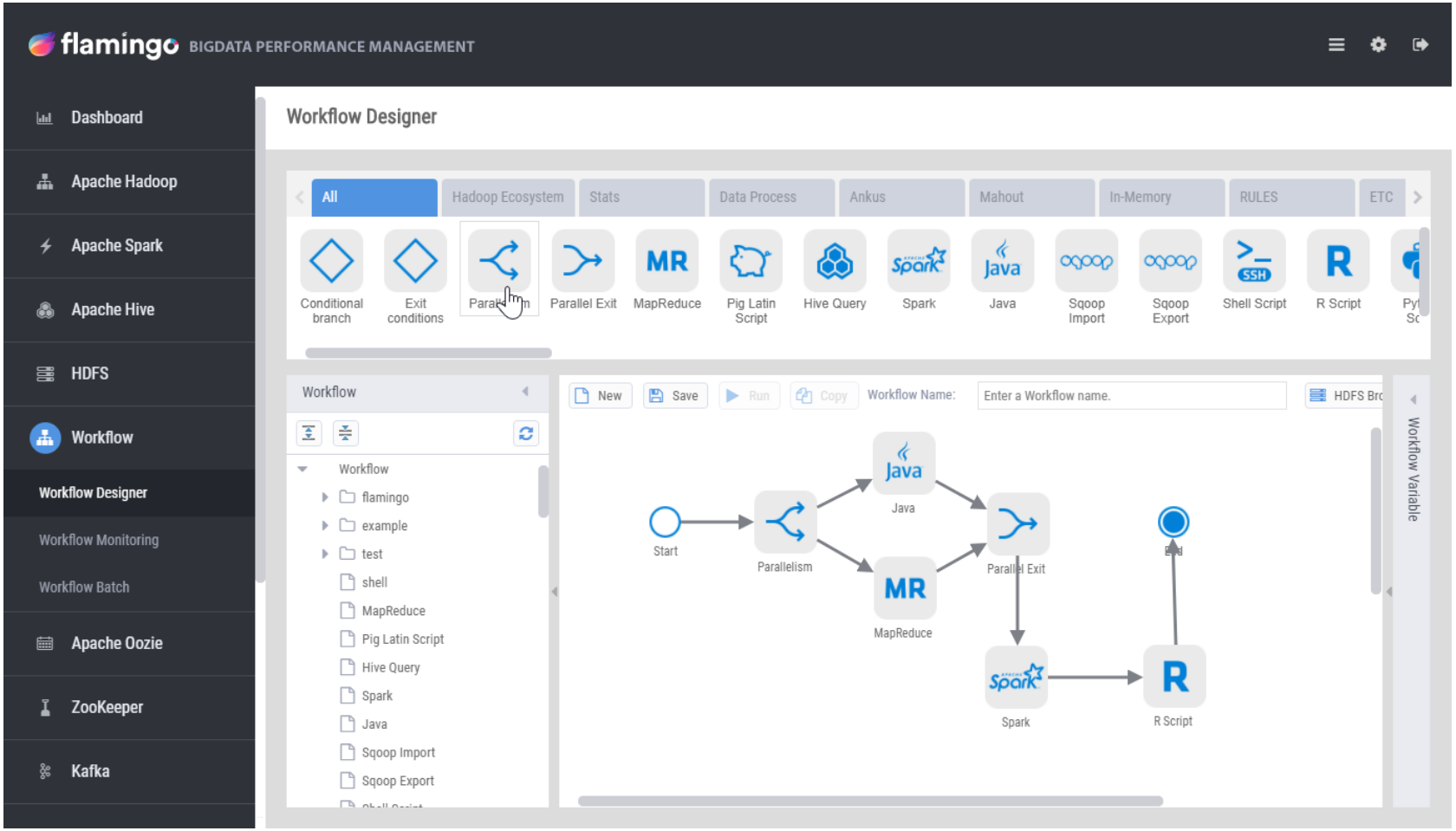
FLAMINGO는 HIVE SERVER 및 HIVE JOB을 모니터링 하기 위한 Total 솔루션입니다. Hive Query는 다수의 YARN, MapReduce로 동작하여 모니터링이 어렵지만 FLAMINGO는 Topology View를 통해서 그래프로 추적합니다. 또한 Hive Server의 주요 장애 지표를 모니터링합니다.

**Feature**

- 10초 주기 실시간 모니터링 정보 수집 (Polling이 아닌 Pushing 데이터 수집을 통한 정확성 확보 및 누락 데이터 방지)
- Hive Server 2의 JVM Heap Size 모니터링을 통한 장애 대응
- Hive Server 2의 Open Session 모니터링을 통한 접속 사용자수 확인
- Hive Metastore 관리기능을 이용한 테이블 및 데이터베이스 관리
- Hive Query Topology View (Hive Query Plan)
- Hive Table 용량 모니터링

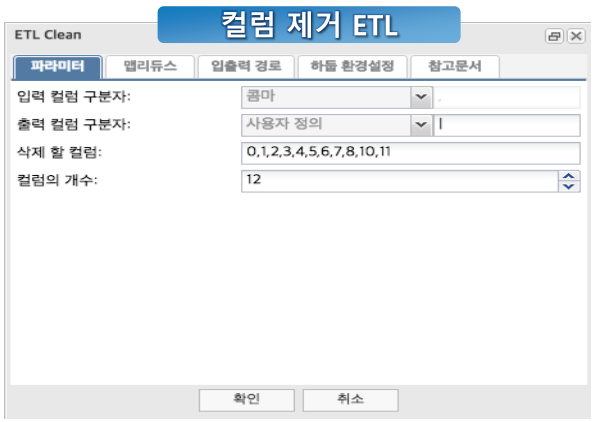
# 워크플로우 디자이너

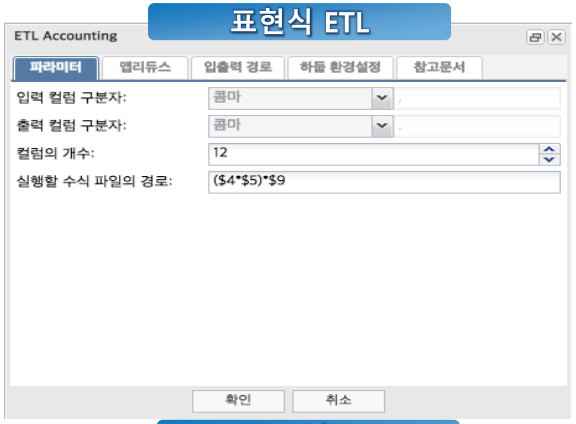
Workflow Designer는 머신러닝 알고리즘, ETL, R, Spark, Hadoop MapReduce, Hive, Python 등의 프로그램을 하나로 묶어서 데이터 분석 흐름을 구성하는 디자이너입니다.

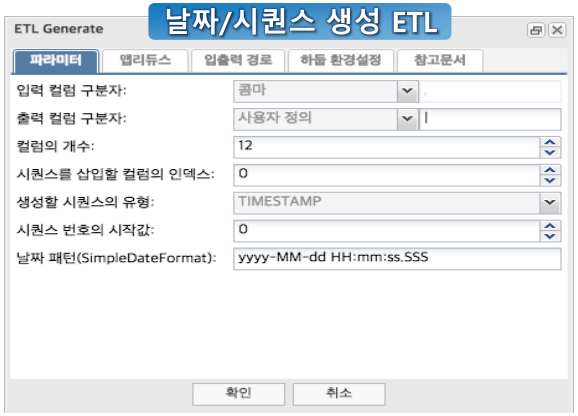


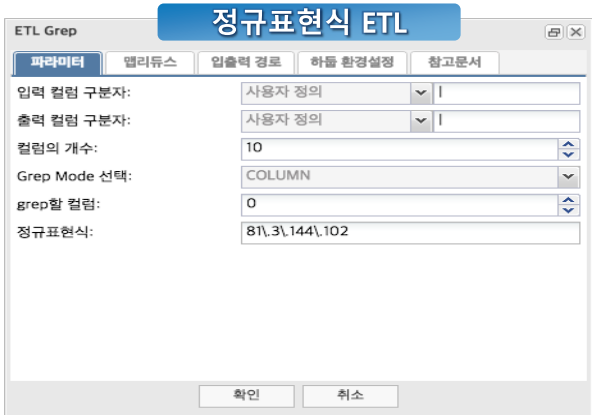
# 워크플로우 디자이너 – 데이터 전처리 모듈 탑재

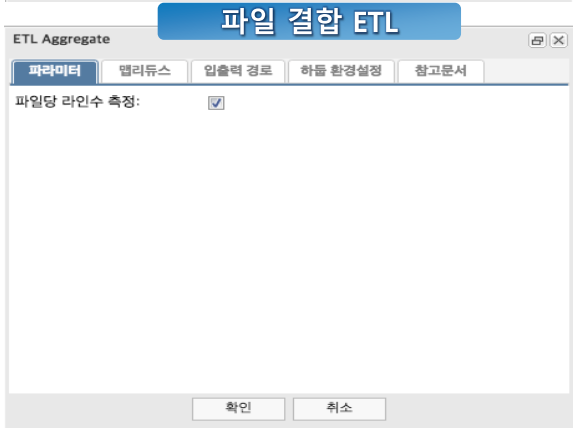
데이터 분석 알고리즘을 통합하여 제공하는 Workflow Designer는 새로운 디자인이 적용되어 곧 Flamingo 3.0에 포함될 예정입니다.

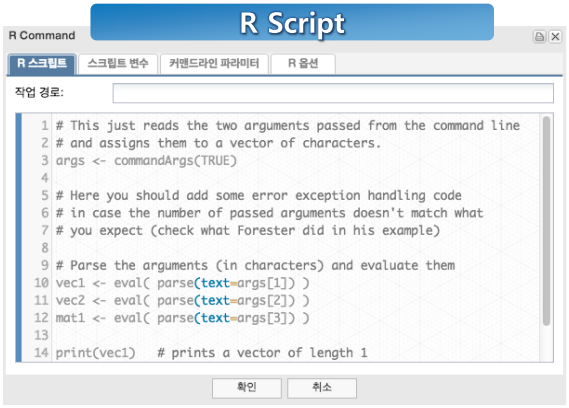



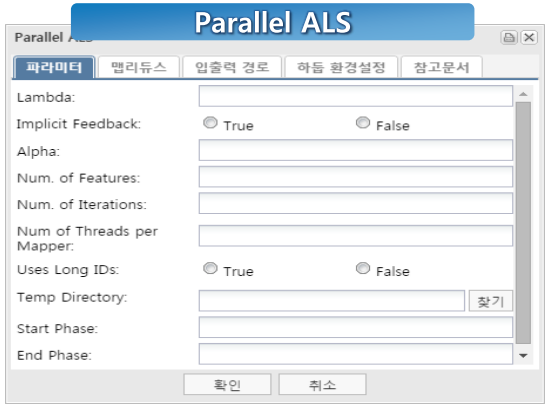



# 워크플로우 디자이너 – 머신러닝 알고리즘 모듈 탑재

데이터 분석 알고리즘을 통합하여 제공하는 Workflow Designer는 새로운 디자인이 적용되어 곧 Flamingo 3.0에 포함될 예정입니다.

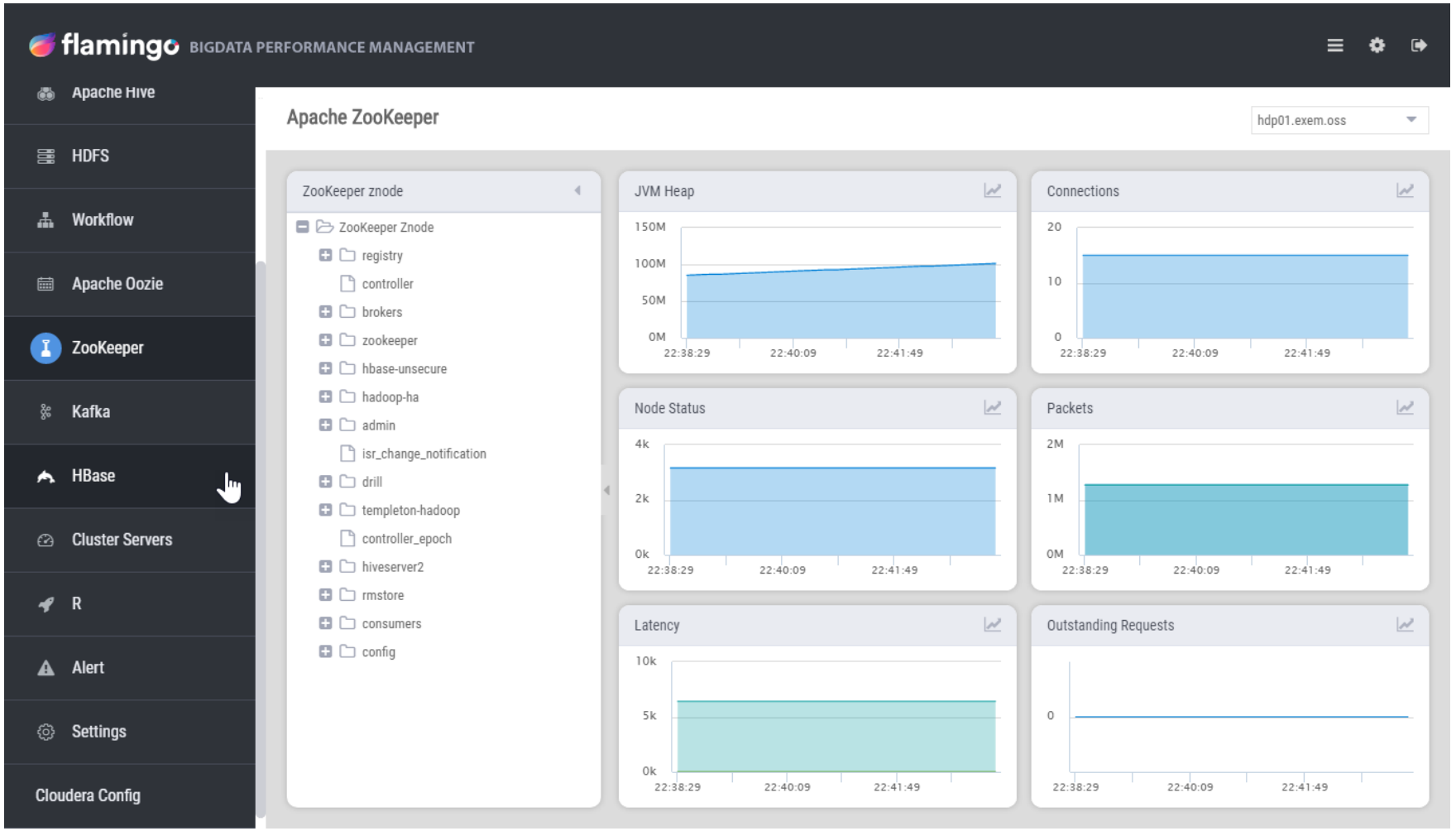




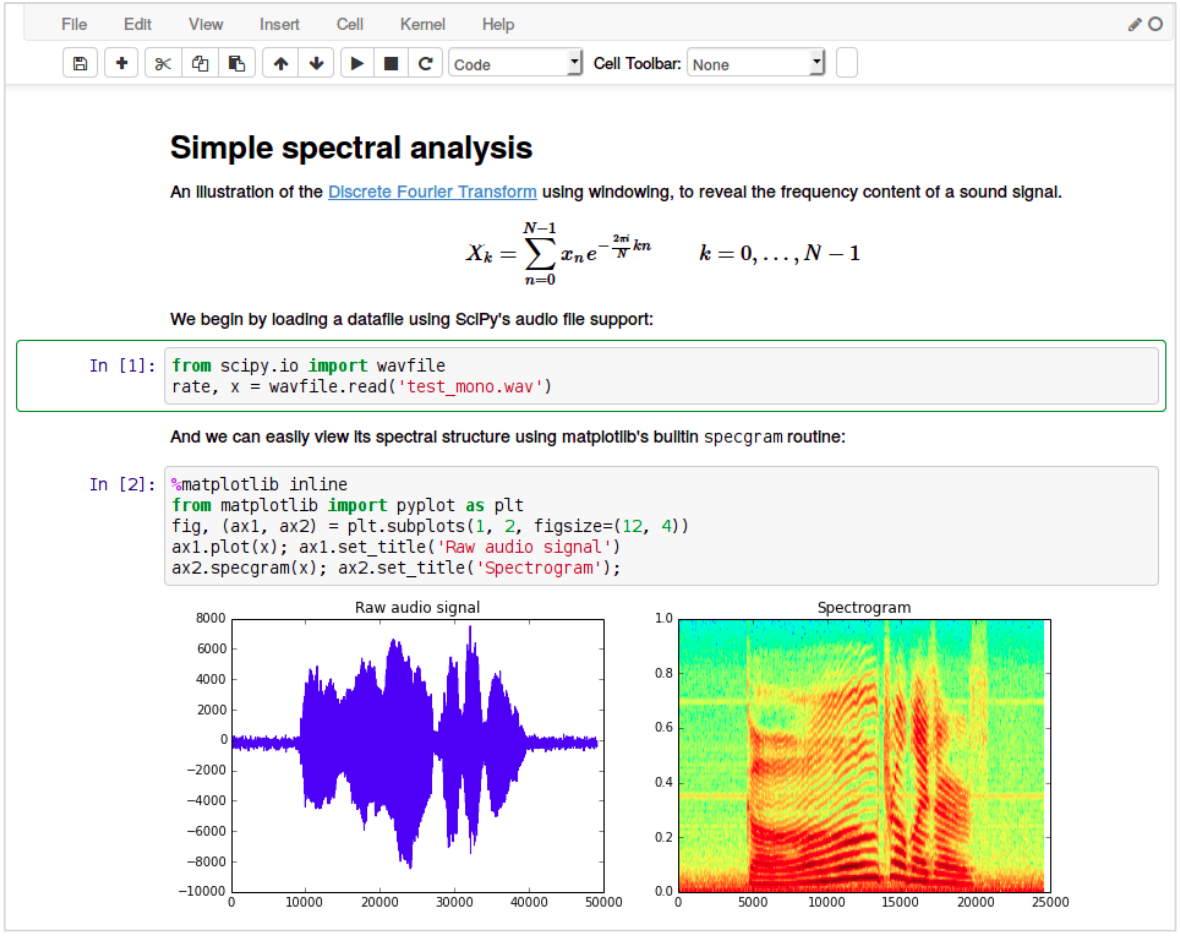


# Apache ZooKeeper 모니터링



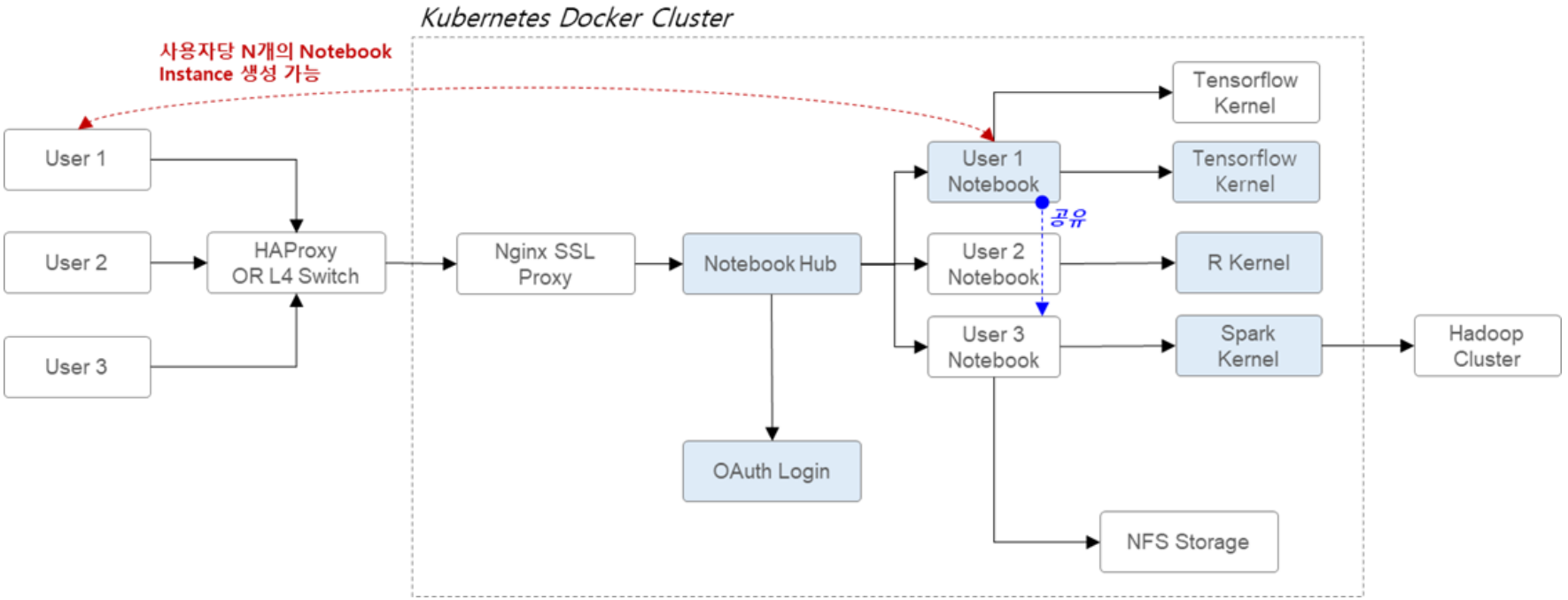
# Notebook

Flamingo에서 지원하는 Notebook은 데이터 분석가들이 웹 기반으로 Python, R, Spark을 이용하여 데이터를 분석하고 관리하는 통합 분석 도구입니다.



# Apache ZooKeeper 모니터링

Flamingo에서 지원하는 Notebook은 다수의 분석가들이 독립적인 분석 환경을 구성하고자 하기 위해서 Docker 기반의 애플리케이션 가상화를 구현합니다. 이를 통해 Spark, R, Python, Tensorflow 등의 다양한 환경을 동시에 분석 시스템을 활용할 수 있도록 지원합니다.



# 감사합니다

빅데이터본부 | 김병곤 본부장