

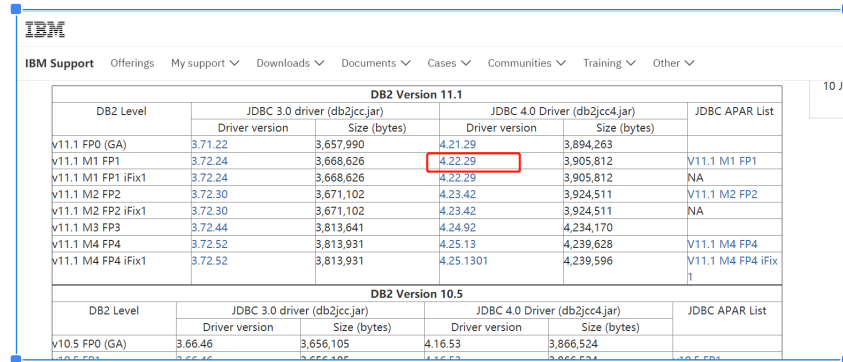
开发skywalking的db2插件

1、仿照skywalking的Oracle插件开发db2插件

Oracle插件地址: <https://github.com/SkyAPM/java-plugin-extensions>

db2 jdbc驱动包下载地址: <https://www.ibm.com/support/pages/db2-jdbc-driver-versions-and-downloads>

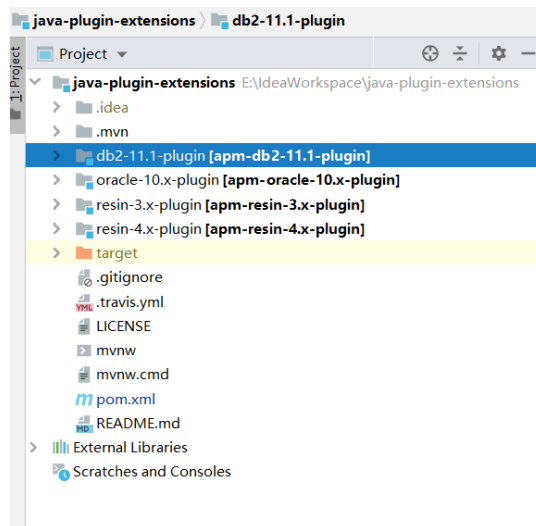
驱动包选用版本:



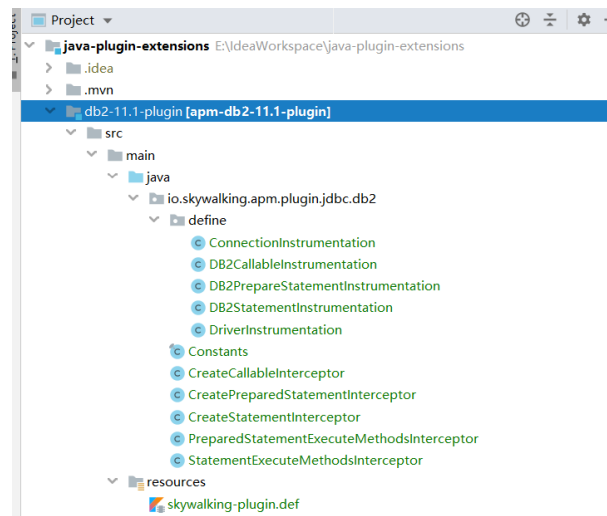
DB2 Version 11.1					
DB2 Level	JDBC 3.0 driver (db2jcc.jar)		JDBC 4.0 Driver (db2jcc4.jar)		JDBC APAR List
	Driver version	Size (bytes)	Driver version	Size (bytes)	
v11.1 FP0 (GA)	3.71.22	3,657,990	4.21.29	3,894,263	
v11.1 M1 FP1	3.72.24	3,668,626	4.22.29	3,905,812	V11.1 M1 FP1
v11.1 M1 FP1 iFix1	3.72.24	3,668,626	4.22.29	3,905,812	NA
v11.1 M2 FP2	3.72.30	3,671,102	4.23.42	3,924,511	V11.1 M2 FP2
v11.1 M2 FP2 iFix1	3.72.30	3,671,102	4.23.42	3,924,511	NA
v11.1 M3 FP3	3.72.44	3,813,641	4.24.92	4,234,170	
v11.1 M4 FP4	3.72.52	3,813,931	4.25.13	4,239,628	V11.1 M4 FP4
v11.1 M4 FP4 iFix1	3.72.52	3,813,931	4.25.1301	4,239,596	V11.1 M4 FP4 iFix1

DB2 Version 10.5					
DB2 Level	JDBC 3.0 driver (db2jcc.jar)		JDBC 4.0 Driver (db2jcc4.jar)		JDBC APAR List
	Driver version	Size (bytes)	Driver version	Size (bytes)	
v10.5 FP0 (GA)	3.66.46	3,656,105	4.16.53	3,866,524	

项目结构:



文件结构:



2、在skywalking中编写DB2URLParser.java文件

位置：skywalking\apm-sniffer\apm-sdk-plugin\jdbc-

commons\src\main\java\org\apache\skywalking\apm\plugin\jdbc\connectionurl\parser\DB2URLParser.java

格式要求：按照PostgreSQLURLParser.java格式进行编写

DB2URLParser.java

```
1 package org.apache.skywalking.apm.plugin.jdbc.connectionurl.parser;
2
3 import org.apache.skywalking.apm.network.trace.component.ComponentsDefine;
4 import org.apache.skywalking.apm.plugin.jdbc.trace.ConnectionInfo;
5
6 /**
7  * {@link DB2URLParser} parse connection url of db2.
8  *
9  */
10 public class DB2URLParser extends AbstractURLParser {
11
12     private static final int DEFAULT_PORT = 50000;
13     private static final String DB_TYPE = "DB2";
14
15     public DB2URLParser(String url) {
16         super(url);
17     }
18
19     @Override
20     protected URLLocation fetchDatabaseHostsIndexRange() {
21         int hostLabelStartIndex = url.indexOf("//");
22         int hostLabelEndIndex = url.indexOf("/", hostLabelStartIndex + 2);
23         if (hostLabelEndIndex == -1) {
24             hostLabelEndIndex = url.indexOf(":", hostLabelStartIndex + 2);
25         }
26         return new URLLocation(hostLabelStartIndex + 2, hostLabelEndIndex);
27     }
28
29     @Override
30     protected URLLocation fetchDatabaseNameIndexRange() {
31         int databaseStartTag = url.lastIndexOf("/");
32         int databaseEndTag = url.indexOf(":", databaseStartTag);
33         if (databaseEndTag == -1) {
34             databaseEndTag = url.length();
35         }
36         return new URLLocation(databaseStartTag + 1, databaseEndTag);
37     }
38
39     @Override
40     public ConnectionInfo parse() {
41         URLLocation location = fetchDatabaseHostsIndexRange();
42         String hosts = url.substring(location.startIndex(), location.endIndex());
43         String[] hostSegment = hosts.split(",");
44         if (hostSegment.length > 1) {
45             StringBuilder sb = new StringBuilder();
46             for (String host : hostSegment) {
47                 if (host.split(":").length == 1) {
48                     sb.append(host + ":" + DEFAULT_PORT + ",");
49                 }
50             }
51             return new ConnectionInfo(hostSegment[0], sb.toString());
52         }
53         return new ConnectionInfo(hostSegment[0], "");
54     }
55 }
```

```

49         } else {
50             sb.append(host + ",");
51         }
52     }
53     return new ConnectionInfo(ComponentsDefine.DB2_DRIVER, DB_TYPE, sb.toString(), fetchDat
54 } else {
55     String[] hostAndPort = hostSegment[0].split(":");
56     if (hostAndPort.length != 1) {
57         return new ConnectionInfo(ComponentsDefine.DB2_DRIVER, DB_TYPE, hostAndPort[0], Int
58     } else {
59         return new ConnectionInfo(ComponentsDefine.DB2_DRIVER, DB_TYPE, hostAndPort[0], DEF
60     }
61 }
62 }
63 }

```

3、在skywalking中更改URLParser

位置：和DB2URLParser.java同一目录

添加：

```
private static final String DB2_JDBC_URL_PREFIX = "jdbc:db2";
```

和

```
else if (lowerCaseUrl.startsWith(DB2_JDBC_URL_PREFIX)){
    parser = new DB2URLParser(url);
}

```

```

5 private static final String DB2_JDBC_URL_PREFIX = "jdbc:db2";
6
7
8 @
9 public static ConnectionInfo parser(String url) {
10     ConnectionUrlParser parser = null;
11     String lowerCaseUrl = url.toLowerCase();
12     if (lowerCaseUrl.startsWith(MYSQL_JDBC_URL_PREFIX)) {
13         parser = new MySQLUrlParser(url);
14     } else if (lowerCaseUrl.startsWith(ORACLE_JDBC_URL_PREFIX)) {
15         parser = new OracleUrlParser(url);
16     } else if (lowerCaseUrl.startsWith(H2_JDBC_URL_PREFIX)) {
17         parser = new H2UrlParser(url);
18     } else if (lowerCaseUrl.startsWith(POSTGRESQL_JDBC_URL_PREFIX)) {
19         parser = new PostgreSQLUrlParser(url);
20     } else if (lowerCaseUrl.startsWith(DB2_JDBC_URL_PREFIX)){
21         parser = new DB2URLParser(url);
22     }
23     return parser.parse();
24 }
25 }

```

4、在ComponentsDefine.java中添加属性

位置：apm-protocol\apm-

network\src\main\java\org\apache\skywalking\apm\network\trace\component\ComponentsDefine.java

文件末尾添加：

```
public static final OfficialComponent DB2_DRIVER = new OfficialComponent(69, "db2jcc4");
```

5、在component-libraries.yml中添加属性

位置：oap-server\server-starter\src\main\resources\component-libraries.yml

263行添加：

```

db2jcc4:
  id: 69
  languages: Java

```

6、在skywalking中关闭checkStyle和test

位置：主项目Pom文件

搜索：maven-checkstyle-plugin

在<configuration>中添加：<skip>true</skip>

搜索：maven-surefire-plugin

在<configuration>中添加：<skip>true</skip>

7、将写db2插件jar包放到skywalking-agent/plugins文件夹中

db2插件源码地址：<https://github.com/zml400/skywalking-java-plugin-extensions>

8、准备测试spring boot web项目

9、运行测试

- (1) 启动 SkyWalking OAP Server
- (2) 启动 SkyWalking UI
- (3) 配置springboot web项目jvm参数
- (4) 运行该web项目的Application.java
- (5) 输入ip地址查看效果

skywalking UI: <http://localhost:8080/>

springboot web项目: <http://localhost:9999/>

10、监控远程应用

将skywalking解压文件夹中的agent文件夹拷贝到需要监控的应用所在的服务器，更改./agent/config下的配置文件，重点添加如下配置

collector.servers=192.168.101.161:10800

上面192.168.101.161是skywalking主程序中的地址

正常启动java应用，带上如下参数：

-javaagent:/path/to/agent/skywalking-agent.jar