

Database - Support #4550

eliminate the need to place the p2jspi.jar in the JVM extension directory

02/19/2020 11:30 AM - Greg Shah

Status:	Closed	Start date:	
Priority:	Normal	Due date:	
Assignee:		% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:		case_num:	
billable:	No		
vendor_id:	GCD		
Description			
Related issues:			
Related to Runtime Infrastructure - Support #4549: reduce/eliminate installat...			New
Related to Database - Feature #5491: custom locale/collation implementations			WIP

History

#1 - 02/19/2020 11:37 AM - Greg Shah

In Java 8 it is supposed to be possible to load service providers from the application classpath. See [ServiceLoader Java 8 Javadoc](#).

In Java 9, it is no longer even possible to load service providers from the JVM extension location. See [Locale Sensitive Services SPIs](#) and the [LocaleServiceProvider Java 9 Javadoc](#).

The p2jspi.jar should be eliminated (and the SPI placed in p2j.jar) OR it should be loaded from the application classpath. Either way there should be no requirement to place it anywhere in the OS as part of the install, ever. It should come along as one of the jars needed for FWD, optimally in the p2j.jar itself.

#2 - 02/19/2020 02:11 PM - Greg Shah

- Related to Support #4549: reduce/eliminate installation dependencies added

#3 - 02/26/2021 07:53 AM - Greg Shah

We also need to consider that the effective codepage for H2 temp-tables must be something that can be overridden. In other words, customers using something other than 8859-1 should not be using the current p2jspi.jar. If I understand correctly, the p2jspi.jar is hard coded to the equivalent of the 4GL 8859-1. This is a massive "bug" hiding in plain sight.

#4 - 02/26/2021 02:28 PM - Eric Faulhaber

Greg Shah wrote:

If I understand correctly, the p2jspi.jar is hard coded to the equivalent of the 4GL 8859-1.

No, it is very much not hard-coded, but currently p2jspi.jar only contains a collator implementation for 8859-1:

```
Archive:  p2jspi.jar
  Length      Date    Time    Name
-----
         0  2021-02-25  13:20  META-INF/
```

```

215  2021-02-25 13:20  META-INF/MANIFEST.MF
0    2021-02-25 13:18  META-INF/services/
43   2021-02-25 13:18  META-INF/services/java.text.spi.CollatorProvider
0    2021-02-25 13:18  com/
0    2021-02-25 13:20  com/goldencode/
0    2021-02-25 13:19  com/goldencode/p2j/
1097 2021-02-25 13:19  com/goldencode/p2j/SpiVersion.class
0    2021-02-25 13:20  com/goldencode/p2j/spi/
2879 2021-02-25 13:19  com/goldencode/p2j/spi/ISO_8859_1Collator.class
2463 2021-02-25 13:20  com/goldencode/p2j/spi/P2JCollatorProvider.class
-----
6697                                11 files

```

Other collation implementations can be added to the jar as needed, and a Locale mapping for each must be added to the P2jCollatorProvider.collatorClasses map. The list of locales supported must be exposed to Java's SPI architecture by implementing the java.text.spi.CollatorProvider interface, which P2jCollatorProvider does. Using Java's SPI architecture to expose custom collations allows H2 to know what to do with a SET COLLATION statement.

The collator to be set at server startup is read from the directory (from the database/<database_name>/p2j/embedded-collation path). If it is not specified, it defaults to en_US_P2J, which is mapped to the com.goldencode.p2j.spi.ISO_8859_1Collator class. It is set within H2 using a set collation <name> statement, using that default name, or the one read from the directory.

This is a massive "bug" hiding in plain sight.

Only to the degree that a customer uses a different collation, which has not yet been added to the jar and specified in the directory.

#5 - 02/26/2021 02:47 PM - Greg Shah

It is good to know that the bug is just a configuration issue. At a minimum we need to improve our documentation. For example, if you read the embedded-collation portions of [Database Configuration](#) you will get a different idea of how it works as compared to your comments above.

Also, some of the text has FWD instead of P2J...

#6 - 07/28/2021 09:47 AM - Greg Shah

- Related to Feature #5491: custom locale/collation implementations added

#7 - 05/10/2022 12:28 PM - Greg Shah

As part of the shift to Java 11 ([#5567](#)), this task should be completed.

#8 - 06/08/2023 05:13 AM - Greg Shah

- % Done changed from 0 to 100

- Status changed from New to Closed