Robotics Service Bus - Enhancement #2196

Enhancement # 2195 (Resolved): Implement Spread connection pooling

Spread connection pooling for informers [Java]

03/11/2015 11:46 AM - J. Wienke

Status:	Resolved	**Start date:**	03/11/2015
Priority:	Normal	**Due date:**	
Assignee:	J. Wienke	**% Done:**	100%
Category:	Java	**Estimated time:**	0.00 hour
Target version:	rsb-0.12		

Description

Associated revisions

Revision c1f14b19 - 03/11/2015 12:19 PM - J. Wienke

Extract spread connection options into SpreadOptions

Creates a new class SpreadOptions which contains all options used to specify the properties of a connection to the spread daemon and uses this class instead of separate arguments or Properties instances.

This prepares to use the connection properties as keys in maps for later sharing of connections.

refs #2196

Revision 19367f99 - 03/11/2015 12:19 PM - J. Wienke

Make SpreadWrapper an interface

Makes SpreadWrapper and interface and moves the existing implementation to SpreadWrapperImpl. This allows to implement facade classes, which is necessary for the connection pooling.

- SpreadFactory: Use SpreadWrapperImpl instead of interface for new instances
- SpreadInPushConnector: Pass through some exceptions specified by the Activatable interface which previously had been explicitly masked by SpreadWrapperImpl
- SpreadOutConnector: likewise
- SpreadWrapper: Converted to interface
- SpreadWrapperImpl: old SpreadWrapper implementation. Also removed some unused methods.
- Utilities: Use SpreadWrapperImpl

refs #2196

Revision 9a5e604d - 03/11/2015 12:19 PM - J. Wienke

Implement connection pooling for outgoing connections

This commit adds spread connection pooling for outgoing connection by
reusing existing connections inside SpreadFactory. The effective
activation and deactivation of the underlying connection is now
implemented via reference counting of activate and deactivate calls so
that the first user of a connection activates it and the last leaving
user effectively deactivates it.

- RefCountingSpreadWrapper: new SpreadWrapper facade implementation
  which does the reference counting for
  multiple users of an underlying
  SpreadWrapper instance.
- SpreadFactory: Cache SpreadWrapper instances based on their
  SpreadOptions to reuse them for outgoing connections
- SpreadOutConnector: Decouple the activation state from the underlying
  SpreadWrapper instance since the reference
  counting might actually leave a connection open
  after a call to deactivate

refs #2196

Revision 20e3e516 - 03/31/2015 07:01 PM - J. Wienke

Extract spread connection options into SpreadOptions

Creates a new class SpreadOptions which contains all options used to
specify the properties of a connection to the spread daemon and uses
this class instead of separate arguments or Properties instances.

This prepares to use the connection properties as keys in maps for later
sharing of connections.

refs #2196

Revision be6538b0 - 03/31/2015 07:14 PM - J. Wienke

Make SpreadWrapper an interface

Makes SpreadWrapper and interface and moves the existing implementation
to SpreadWrapperImpl. This allows to implement facade classes, which is
necessary for the connection pooling.

- SpreadFactory: Use SpreadWrapperImpl instead of interface for new
  instances
- SpreadInPushConnector: Pass through some exceptions specified by the
  Activatable interface which previously had been
  explicitly masked by SpreadWrapperImpl
- SpreadOutConnector: likewise
- SpreadWrapper: Converted to interface
- SpreadWrapperImpl: old SpreadWrapper implementation. Also removed some
  unused methods.
- Utilities: Use SpreadWrapperImpl

refs #2196

Revision c7737c02 - 03/31/2015 07:14 PM - J. Wienke

Implement connection pooling for outgoing connections

This commit adds spread connection pooling for outgoing connection by
reusing existing connections inside SpreadFactory. The effective
activation and deactivation of the underlying connection is now
implemented via reference counting of activate and deactivate calls so
that the first user of a connection activates it and the last leaving
user effectively deactivates it.

- RefCountingSpreadWrapper: new SpreadWrapper facade implementation
  which does the reference counting for
  multiple users of an underlying
  SpreadWrapper instance.
- SpreadFactory: Cache SpreadWrapper instances based on their
  SpreadOptions to reuse them for outgoing connections
- SpreadOutConnector: Decouple the activation state from the underlying
  SpreadWrapper instance since the reference
  counting might actually leave a connection open
  after a call to deactivate

refs #2196

Revision acb98f27 - 04/20/2015 07:08 PM - J. Wienke

Use the same connection for all SpreadOutConnectors

refs #2196

History

#1 - 03/11/2015 02:10 PM - J. Wienke

- % Done changed from 0 to 30

Connection pooling for informers is implemented in this branch now.

#2 - 04/23/2015 11:50 AM - J. Wienke

- Subject changed from Implement java spread connection pooling to Implement java spread connection pooling for informers

#3 - 04/23/2015 11:51 AM - J. Wienke

- % Done changed from 30 to 90

#4 - 04/23/2015 11:51 AM - J. Wienke

- Status changed from In Progress to Resolved

- % Done changed from 90 to 100

#5 - 04/15/2016 11:34 AM - J. Moringen

- Subject changed from Implement java spread connection pooling for informers to Spread connection pooling for informers [Java]