

**git\_comments:****git\_commits:**

1. **summary:** DISPATCH-1656: fix condition check error found by coverity  
**message:** DISPATCH-1656: fix condition check error found by coverity

**github\_issues:****github\_issues\_comments:****github\_pulls:****github\_pulls\_comments:****github\_pulls\_reviews:****jira\_issues:**

1. **summary:** Allow user-configured connection properties in the Open performative  
**description:** This feature would add a new "openProperties" map attribute to the org.apache.qpid.dispatch.connector and org.apache.qpid.dispatch.listener management entities. This new attribute is optional. The contents of the map are expected to be specified in JSON. The keys must be restricted to ASCII characters only in order to be encoded as AMQP 1.0 Symbol types on the wire. Key values that start with the prefixes \*qd.\* or \*x-opt-qd.\* are reserved and must not be used. In addition, the following keys are reserved and must not be used: product version failover-server-list network-host port scheme hostname The openProperties keys and values will be merged into the connection properties field of the outgoing Open performative when establishing or accepting connections. The openProperties attribute is NOT allowed on connector or listener entities with roles of either "inter-router" or "edge". The openProperties attribute is NOT allowed on listener entities than enable http, i.e. http: true. Example:
 

```

{{connector {}} {{ name: broker}} {{ role: route-container}} {{ host: 127.0.0.1}} {{ port: 22180}}
{{ saslMechanisms: ANONYMOUS}} {{ *openProperties: {}} {{   *foo: "bar",*}} {{
*"integer": 7,*}} {{*"list": ["a", 1, "b", -9, true],*}} {{*"map": \{"key1": null, "key2": [1, 2, 3]\},*}}
{{*}*}} {{ cost: 10}} {{}}}} The resulting connection properties in the outgoing Open performative
would include the following map entries: {{Symbol(foo): String(bar),}} {{Symbol(integer): Int(7),}}
{{Symbol(list): List(String(a), Int(1), String(b), Int(-9), Boolean),}} {{Symbol(map): Map(String(key1):
Null, String(key2): List(Int(1), Int(2), Int(3)))}}

```
2. **summary:** Allow user-configured connection properties in the Open performative  
**description:** This feature would add a new "openProperties" map attribute to the org.apache.qpid.dispatch.connector and org.apache.qpid.dispatch.listener management entities. This new attribute is optional. The contents of the map are expected to be specified in JSON. The keys must be restricted to ASCII characters only in order to be encoded as AMQP 1.0 Symbol types on the wire. Key values that start with the prefixes \*qd.\* or \*x-opt-qd.\* are reserved and must not be used. In addition, the following keys are reserved and must not be used: product version failover-server-list network-host port scheme hostname The openProperties keys and values will be merged into the connection properties field of the outgoing Open performative when establishing or accepting connections. The openProperties attribute is NOT allowed on connector or listener entities with roles of either "inter-router" or "edge". The openProperties attribute is NOT allowed on listener entities than enable http, i.e. http: true. Example:
 

```

{{connector {}} {{ name: broker}} {{ role: route-container}} {{ host: 127.0.0.1}} {{ port: 22180}}
{{ saslMechanisms: ANONYMOUS}} {{ *openProperties: {}} {{   *foo: "bar",*}} {{
*"integer": 7,*}} {{*"list": ["a", 1, "b", -9, true],*}} {{*"map": \{"key1": null, "key2": [1, 2, 3]\},*}}
{{*}*}} {{ cost: 10}} {{}}}} The resulting connection properties in the outgoing Open performative
would include the following map entries: {{Symbol(foo): String(bar),}} {{Symbol(integer): Int(7),}}
{{Symbol(list): List(String(a), Int(1), String(b), Int(-9), Boolean),}} {{Symbol(map): Map(String(key1):
Null, String(key2): List(Int(1), Int(2), Int(3)))}}

```

**label:** code-design
3. **summary:** Allow user-configured connection properties in the Open performative

**description:** This feature would add a new "openProperties" map attribute to the org.apache.qpid.dispatch.connector and org.apache.qpid.dispatch.listener management entities. This new attribute is optional. The contents of the map are expected to be specified in JSON. The keys must be restricted to ASCII characters only in order to be encoded as AMQP 1.0 Symbol types on the wire. Key values that start with the prefixes \*qd.\* or \*x-opt-qd.\* are reserved and must not be used. In addition, the following keys are reserved and must not be used: product version failover-server-list network-host port scheme hostname The openProperties keys and values will be merged into the connection properties field of the outgoing Open performative when establishing or accepting connections. The openProperties attribute is NOT allowed on connector or listener entities with roles of either "inter-router" or "edge". The openProperties attribute is NOT allowed on listener entities than enable http, i.e. http: true. Example: {{connector {}} {{ name: broker}} {{ role: route-container}} {{ host: 127.0.0.1}} {{ port: 22180}} {{ saslMechanisms: ANONYMOUS}} {{ \*openProperties: {}} {{ \*foo: "bar",\*}} {{ \*integer: 7,\*}} {{\*"list": ["a", 1, "b", -9, true],\*}} {{\*"map": \{"key1": null, "key2": [1, 2, 3]\},\*}} {{\*}\*}} {{ cost: 10}} {{}}}} The resulting connection properties in the outgoing Open performative would include the following map entries: {{Symbol(foo): String(bar),}} {{Symbol(integer): Int(7),}} {{Symbol(list): List(String(a), Int(1), String(b), Int(-9), Boolean),}} {{Symbol(map): Map(String(key1): Null, String(key2): List(Int(1), Int(2), Int(3)))}}

4. **summary:** Allow user-configured connection properties in the Open performative

**description:** This feature would add a new "openProperties" map attribute to the org.apache.qpid.dispatch.connector and org.apache.qpid.dispatch.listener management entities. This new attribute is optional. The contents of the map are expected to be specified in JSON. The keys must be restricted to ASCII characters only in order to be encoded as AMQP 1.0 Symbol types on the wire. Key values that start with the prefixes \*qd.\* or \*x-opt-qd.\* are reserved and must not be used. In addition, the following keys are reserved and must not be used: product version failover-server-list network-host port scheme hostname The openProperties keys and values will be merged into the connection properties field of the outgoing Open performative when establishing or accepting connections. The openProperties attribute is NOT allowed on connector or listener entities with roles of either "inter-router" or "edge". The openProperties attribute is NOT allowed on listener entities than enable http, i.e. http: true. Example: {{connector {}} {{ name: broker}} {{ role: route-container}} {{ host: 127.0.0.1}} {{ port: 22180}} {{ saslMechanisms: ANONYMOUS}} {{ \*openProperties: {}} {{ \*foo: "bar",\*}} {{ \*integer: 7,\*}} {{\*"list": ["a", 1, "b", -9, true],\*}} {{\*"map": \{"key1": null, "key2": [1, 2, 3]\},\*}} {{\*}\*}} {{ cost: 10}} {{}}}} The resulting connection properties in the outgoing Open performative would include the following map entries: {{Symbol(foo): String(bar),}} {{Symbol(integer): Int(7),}} {{Symbol(list): List(String(a), Int(1), String(b), Int(-9), Boolean),}} {{Symbol(map): Map(String(key1): Null, String(key2): List(Int(1), Int(2), Int(3)))}}

5. **summary:** Allow user-configured connection properties in the Open performative

**description:** This feature would add a new "openProperties" map attribute to the org.apache.qpid.dispatch.connector and org.apache.qpid.dispatch.listener management entities. This new attribute is optional. The contents of the map are expected to be specified in JSON. The keys must be restricted to ASCII characters only in order to be encoded as AMQP 1.0 Symbol types on the wire. Key values that start with the prefixes \*qd.\* or \*x-opt-qd.\* are reserved and must not be used. In addition, the following keys are reserved and must not be used: product version failover-server-list network-host port scheme hostname The openProperties keys and values will be merged into the connection properties field of the outgoing Open performative when establishing or accepting connections. The openProperties attribute is NOT allowed on connector or listener entities with roles of either "inter-router" or "edge". The openProperties attribute is NOT allowed on listener entities than enable http, i.e. http: true. Example: {{connector {}} {{ name: broker}} {{ role: route-container}} {{ host: 127.0.0.1}} {{ port: 22180}} {{ saslMechanisms: ANONYMOUS}} {{ \*openProperties: {}} {{ \*foo: "bar",\*}} {{ \*integer: 7,\*}} {{\*"list": ["a", 1, "b", -9, true],\*}} {{\*"map": \{"key1": null, "key2": [1, 2, 3]\},\*}} {{\*}\*}} {{ cost: 10}} {{}}}} The resulting connection properties in the outgoing Open performative would include the following map entries: {{Symbol(foo): String(bar),}} {{Symbol(integer): Int(7),}} {{Symbol(list): List(String(a), Int(1), String(b), Int(-9), Boolean),}} {{Symbol(map): Map(String(key1): Null, String(key2): List(Int(1), Int(2), Int(3)))}}

6. **summary:** Allow user-configured connection properties in the Open performative

**description:** This feature would add a new "openProperties" map attribute to the org.apache.qpid.dispatch.connector and org.apache.qpid.dispatch.listener management entities. This new attribute is optional. The contents of the map are expected to be specified in JSON. The keys must be restricted to ASCII characters only in order to be encoded as AMQP 1.0 Symbol types on the wire. Key values that start with the prefixes \*qd.\* or \*x-opt-qd.\* are reserved and must not be used. In addition, the

following keys are reserved and must not be used: product version failover-server-list network-host port scheme hostname The openProperties keys and values will be merged into the connection properties field of the outgoing Open performative when establishing or accepting connections. The openProperties attribute is NOT allowed on connector or listener entities with roles of either "inter-router" or "edge". The openProperties attribute is NOT allowed on listener entities than enable http, i.e. http: true. Example: `{{connector {}} {{ name: broker}} {{ role: route-container}} {{ host: 127.0.0.1}} {{ port: 22180}} {{ saslMechanisms: ANONYMOUS}} {{ *openProperties: {}} {{ *foo: "bar",*}} {{ *integer: 7,*}} {{*list: ["a", 1, "b", -9, true],*}} {{*map: \{"key1": null, "key2": [1, 2, 3]\},*}} {{*}} {{ cost: 10}} {{}}` The resulting connection properties in the outgoing Open performative would include the following map entries: `{{Symbol(foo): String(bar),}} {{Symbol(integer): Int(7),}} {{Symbol(list): List(String(a), Int(1), String(b), Int(-9), Boolean),}} {{Symbol(map): Map(String(key1): Null, String(key2): List(Int(1), Int(2), Int(3)))}}`

7. **summary:** Allow user-configured connection properties in the Open performative

**description:** This feature would add a new "openProperties" map attribute to the org.apache.qpid.dispatch.connector and org.apache.qpid.dispatch.listener management entities. This new attribute is optional. The contents of the map are expected to be specified in JSON. The keys must be restricted to ASCII characters only in order to be encoded as AMQP 1.0 Symbol types on the wire. Key values that start with the prefixes \*qd.\* or \*x-opt-qd.\* are reserved and must not be used. In addition, the following keys are reserved and must not be used: product version failover-server-list network-host port scheme hostname The openProperties keys and values will be merged into the connection properties field of the outgoing Open performative when establishing or accepting connections. The openProperties attribute is NOT allowed on connector or listener entities with roles of either "inter-router" or "edge". The openProperties attribute is NOT allowed on listener entities than enable http, i.e. http: true. Example: `{{connector {}} {{ name: broker}} {{ role: route-container}} {{ host: 127.0.0.1}} {{ port: 22180}} {{ saslMechanisms: ANONYMOUS}} {{ *openProperties: {}} {{ *foo: "bar",*}} {{ *integer: 7,*}} {{*list: ["a", 1, "b", -9, true],*}} {{*map: \{"key1": null, "key2": [1, 2, 3]\},*}} {{*}} {{ cost: 10}} {{}}` The resulting connection properties in the outgoing Open performative would include the following map entries: `{{Symbol(foo): String(bar),}} {{Symbol(integer): Int(7),}} {{Symbol(list): List(String(a), Int(1), String(b), Int(-9), Boolean),}} {{Symbol(map): Map(String(key1): Null, String(key2): List(Int(1), Int(2), Int(3)))}}`

8. **summary:** Allow user-configured connection properties in the Open performative

**description:** This feature would add a new "openProperties" map attribute to the org.apache.qpid.dispatch.connector and org.apache.qpid.dispatch.listener management entities. This new attribute is optional. The contents of the map are expected to be specified in JSON. The keys must be restricted to ASCII characters only in order to be encoded as AMQP 1.0 Symbol types on the wire. Key values that start with the prefixes \*qd.\* or \*x-opt-qd.\* are reserved and must not be used. In addition, the following keys are reserved and must not be used: product version failover-server-list network-host port scheme hostname The openProperties keys and values will be merged into the connection properties field of the outgoing Open performative when establishing or accepting connections. The openProperties attribute is NOT allowed on connector or listener entities with roles of either "inter-router" or "edge". The openProperties attribute is NOT allowed on listener entities than enable http, i.e. http: true. Example: `{{connector {}} {{ name: broker}} {{ role: route-container}} {{ host: 127.0.0.1}} {{ port: 22180}} {{ saslMechanisms: ANONYMOUS}} {{ *openProperties: {}} {{ *foo: "bar",*}} {{ *integer: 7,*}} {{*list: ["a", 1, "b", -9, true],*}} {{*map: \{"key1": null, "key2": [1, 2, 3]\},*}} {{*}} {{ cost: 10}} {{}}` The resulting connection properties in the outgoing Open performative would include the following map entries: `{{Symbol(foo): String(bar),}} {{Symbol(integer): Int(7),}} {{Symbol(list): List(String(a), Int(1), String(b), Int(-9), Boolean),}} {{Symbol(map): Map(String(key1): Null, String(key2): List(Int(1), Int(2), Int(3)))}}`

9. **summary:** Allow user-configured connection properties in the Open performative

**description:** This feature would add a new "openProperties" map attribute to the org.apache.qpid.dispatch.connector and org.apache.qpid.dispatch.listener management entities. This new attribute is optional. The contents of the map are expected to be specified in JSON. The keys must be restricted to ASCII characters only in order to be encoded as AMQP 1.0 Symbol types on the wire. Key values that start with the prefixes \*qd.\* or \*x-opt-qd.\* are reserved and must not be used. In addition, the following keys are reserved and must not be used: product version failover-server-list network-host port scheme hostname The openProperties keys and values will be merged into the connection properties field of the outgoing Open performative when establishing or accepting connections. The openProperties attribute is NOT allowed on connector or listener entities with roles of either "inter-router" or "edge". The openProperties attribute is NOT allowed on listener entities than enable http, i.e. http: true. Example:

```

{{connector {{}} {{ name: broker}} {{ role: route-container}} {{ host: 127.0.0.1}} {{ port: 22180}}
{{ saslMechanisms: ANONYMOUS}} {{ *openProperties: {}} {{   *foo: "bar",*}} {{
*"integer": 7,*}} {{*"list": ["a", 1, "b", -9, true],*}} {{*"map": \{"key1": null, "key2": [1, 2, 3]\},*}}
{{*}*}} {{ cost: 10}} {{}}}} The resulting connection properties in the outgoing Open performative
would include the following map entries: {{Symbol(foo): String(bar),}} {{Symbol(integer): Int(7),}}
{{Symbol(list): List(String(a), Int(1), String(b), Int(-9), Boolean),}} {{Symbol(map): Map(String(key1):
Null, String(key2): List(Int(1), Int(2), Int(3)))}}

```

10. **summary:** Allow user-configured connection properties in the Open performative

**description:** This feature would add a new "openProperties" map attribute to the org.apache.qpid.dispatch.connector and org.apache.qpid.dispatch.listener management entities. This new attribute is optional. The contents of the map are expected to be specified in JSON. The keys must be restricted to ASCII characters only in order to be encoded as AMQP 1.0 Symbol types on the wire. Key values that start with the prefixes \*qd.\* or \*x-opt-qd.\* are reserved and must not be used. In addition, the following keys are reserved and must not be used: product version failover-server-list network-host port scheme hostname The openProperties keys and values will be merged into the connection properties field of the outgoing Open performative when establishing or accepting connections. The openProperties attribute is NOT allowed on connector or listener entities with roles of either "inter-router" or "edge". The openProperties attribute is NOT allowed on listener entities than enable http, i.e. http: true. Example:

```

{{connector {{}} {{ name: broker}} {{ role: route-container}} {{ host: 127.0.0.1}} {{ port: 22180}}
{{ saslMechanisms: ANONYMOUS}} {{ *openProperties: {}} {{   *foo: "bar",*}} {{
*"integer": 7,*}} {{*"list": ["a", 1, "b", -9, true],*}} {{*"map": \{"key1": null, "key2": [1, 2, 3]\},*}}
{{*}*}} {{ cost: 10}} {{}}}} The resulting connection properties in the outgoing Open performative
would include the following map entries: {{Symbol(foo): String(bar),}} {{Symbol(integer): Int(7),}}
{{Symbol(list): List(String(a), Int(1), String(b), Int(-9), Boolean),}} {{Symbol(map): Map(String(key1):
Null, String(key2): List(Int(1), Int(2), Int(3)))}}

```

11. **summary:** Allow user-configured connection properties in the Open performative

**description:** This feature would add a new "openProperties" map attribute to the org.apache.qpid.dispatch.connector and org.apache.qpid.dispatch.listener management entities. This new attribute is optional. The contents of the map are expected to be specified in JSON. The keys must be restricted to ASCII characters only in order to be encoded as AMQP 1.0 Symbol types on the wire. Key values that start with the prefixes \*qd.\* or \*x-opt-qd.\* are reserved and must not be used. In addition, the following keys are reserved and must not be used: product version failover-server-list network-host port scheme hostname The openProperties keys and values will be merged into the connection properties field of the outgoing Open performative when establishing or accepting connections. The openProperties attribute is NOT allowed on connector or listener entities with roles of either "inter-router" or "edge". The openProperties attribute is NOT allowed on listener entities than enable http, i.e. http: true. Example:

```

{{connector {{}} {{ name: broker}} {{ role: route-container}} {{ host: 127.0.0.1}} {{ port: 22180}}
{{ saslMechanisms: ANONYMOUS}} {{ *openProperties: {}} {{   *foo: "bar",*}} {{
*"integer": 7,*}} {{*"list": ["a", 1, "b", -9, true],*}} {{*"map": \{"key1": null, "key2": [1, 2, 3]\},*}}
{{*}*}} {{ cost: 10}} {{}}}} The resulting connection properties in the outgoing Open performative
would include the following map entries: {{Symbol(foo): String(bar),}} {{Symbol(integer): Int(7),}}
{{Symbol(list): List(String(a), Int(1), String(b), Int(-9), Boolean),}} {{Symbol(map): Map(String(key1):
Null, String(key2): List(Int(1), Int(2), Int(3)))}}

```

12. **summary:** Allow user-configured connection properties in the Open performative

**description:** This feature would add a new "openProperties" map attribute to the org.apache.qpid.dispatch.connector and org.apache.qpid.dispatch.listener management entities. This new attribute is optional. The contents of the map are expected to be specified in JSON. The keys must be restricted to ASCII characters only in order to be encoded as AMQP 1.0 Symbol types on the wire. Key values that start with the prefixes \*qd.\* or \*x-opt-qd.\* are reserved and must not be used. In addition, the following keys are reserved and must not be used: product version failover-server-list network-host port scheme hostname The openProperties keys and values will be merged into the connection properties field of the outgoing Open performative when establishing or accepting connections. The openProperties attribute is NOT allowed on connector or listener entities with roles of either "inter-router" or "edge". The openProperties attribute is NOT allowed on listener entities than enable http, i.e. http: true. Example:

```

{{connector {{}} {{ name: broker}} {{ role: route-container}} {{ host: 127.0.0.1}} {{ port: 22180}}
{{ saslMechanisms: ANONYMOUS}} {{ *openProperties: {}} {{   *foo: "bar",*}} {{
*"integer": 7,*}} {{*"list": ["a", 1, "b", -9, true],*}} {{*"map": \{"key1": null, "key2": [1, 2, 3]\},*}}
{{*}*}} {{ cost: 10}} {{}}}} The resulting connection properties in the outgoing Open performative
would include the following map entries: {{Symbol(foo): String(bar),}} {{Symbol(integer): Int(7),}}

```

```
{{Symbol(list): List(String(a), Int(1), String(b), Int(-9), Boolean),}} {{Symbol(map): Map(String(key1): Null, String(key2): List(Int(1), Int(2), Int(3)))}}
```

13. **summary:** Allow user-configured connection properties in the Open performative

**description:** This feature would add a new "openProperties" map attribute to the org.apache.qpid.dispatch.connector and org.apache.qpid.dispatch.listener management entities. This new attribute is optional. The contents of the map are expected to be specified in JSON. The keys must be restricted to ASCII characters only in order to be encoded as AMQP 1.0 Symbol types on the wire. Key values that start with the prefixes \*qd.\* or \*x-opt-qd.\* are reserved and must not be used. In addition, the following keys are reserved and must not be used: product version failover-server-list network-host port scheme hostname The openProperties keys and values will be merged into the connection properties field of the outgoing Open performative when establishing or accepting connections. The openProperties attribute is NOT allowed on connector or listener entities with roles of either "inter-router" or "edge". The openProperties attribute is NOT allowed on listener entities than enable http, i.e. http: true. Example: {{connector {}} {{ name: broker}} {{ role: route-container}} {{ host: 127.0.0.1}} {{ port: 22180}} {{ saslMechanisms: ANONYMOUS}} {{ \*openProperties: {\*}} {{ \*foo: "bar",\*}} {{ \*integer: 7,\*}} {{\*"list": ["a", 1, "b", -9, true],\*}} {{\*"map": {"key1": null, "key2": [1, 2, 3]},\*}} {{\*}\*}} {{ cost: 10}} {{}} The resulting connection properties in the outgoing Open performative would include the following map entries: {{Symbol(foo): String(bar),}} {{Symbol(integer): Int(7),}} {{Symbol(list): List(String(a), Int(1), String(b), Int(-9), Boolean),}} {{Symbol(map): Map(String(key1): Null, String(key2): List(Int(1), Int(2), Int(3)))}}

**jira\_issues\_comments:**

1. As the AMQP open field properties is of type fields[1] it feels odd that the router configuration property restrict it to only string pairings. Would it be possible if the openProperties string was stringified JSON that must be a map? That way the user would be free to make use of other types. It would mean you could lean on the well defined JSON parsing rule, escape syntax (rather than having Dispatch having special parsing rules for this configuration item). That said, I am anticipating the EnMasse use-case will be name/string value pairs, so the proposed solution would suffice. [~lulf] any thoughts?  
[1] [http://docs.oasis-open.org/amqp/core/v1.0/os/amqp-core-transport-v1.0-os.html#type-open]
2. **body:** +1 to use JSON, it makes it a lot easier to generate the router config than writing a generator specifically for this format.  
**label:** code-design
3. JSON it is!
4. JSON makes good sense for this bit, and embedding it here is a good choice for expediency. (Tangent alert.) But! Why not just make the whole config file JSON? Then you can have nested data of this sort - in a standard format that every language can already generate and parse. Then you don't end up introducing \*special mini-syntaxes\* for element values. (IMO, this whole make-up-your-own-syntax-for-config-files thing is nonsense.)
5. (Tangent response :D) [~jross] you can already use JSON for the whole router config file. The format is a bit ... special IMHO, but it works. Example: {code:java} [ ["listener", { "host": "0.0.0.0", "port": 56721, "authenticatePeer": false, "saslMechanisms": "ANONYMOUS", "multiTenant": true }], ["policy", {"enableVhostPolicy": true, "defaultVhost": "\$default"}], ] {code} [~kgiusti] In this case though, will the config element be a JSON string within this struct, or a JSON "native" object?
6. kgiusti opened a new pull request #746: URL: <https://github.com/apache/qpid-dispatch/pull/746> -----  
----- This is an automated message from the Apache Git Service.  
To respond to the message, please log on to GitHub and use the URL above to go to the specific comment. For queries about this service, please contact Infrastructure at: [users@infra.apache.org](mailto:users@infra.apache.org)
7. codecov-commenter commented on pull request #746: URL: <https://github.com/apache/qpid-dispatch/pull/746#issuecomment-634197090> # [Codecov](<https://codecov.io/gh/apache/qpid-dispatch/pull/746?src=pr&el=h1>) Report > Merging [#746](<https://codecov.io/gh/apache/qpid-dispatch/pull/746?src=pr&el=desc>) into [master](<https://codecov.io/gh/apache/qpid-dispatch/commit/be42638a2da2d9fec0fac98f3b1c1be1606002c0&el=desc>) will \*\*decrease\*\* coverage by `0.32%`. > The diff coverage is `50.24%`. [![Impacted file tree graph](<https://codecov.io/gh/apache/qpid-dispatch/pull/746/graphs/tree.svg?width=650&height=150&src=pr&token=rk2Cgd27pP>)](<https://codecov.io/gh/apache/qpid-dispatch/pull/746?src=pr&el=tree>) ``diff @@ Coverage Diff @@ ## master #746 +/- ##  
===== - Coverage 86.95% 86.63% -0.33%  
===== Files 96 96 Lines 21728 21930 +202

===== + Hits 18894 18999 +105 - Misses 2834 2931  
+97 `` | [Impacted Files](https://codecov.io/gh/apache/qpid-dispatch/pull/746?src=pr&el=tree) |  
Coverage Δ | |---|---|---| | [src/compose.c](https://codecov.io/gh/apache/qpid-dispatch/pull/746/diff?src=pr&el=tree#diff-c3JlL2NvbXBvc2UuYWw=) | `88.02% <0.00%> (-1.58%)` | :arrow\_down: | |  
[src/proton\_utils.c](https://codecov.io/gh/apache/qpid-dispatch/pull/746/diff?src=pr&el=tree#diff-c3JlL3Byb3Rvbl91dGlscy5j) | `48.96% <38.29%> (-19.67%)` | :arrow\_down: | | [src/entity.c]  
(https://codecov.io/gh/apache/qpid-dispatch/pull/746/diff?src=pr&el=tree#diff-c3JlL2VudGl0eS5j) |  
`56.75% <57.14%> (+0.06%)` | :arrow\_up: | | [src/python\_embedded.c]  
(https://codecov.io/gh/apache/qpid-dispatch/pull/746/diff?src=pr&el=tree#diff-c3JlL3B5dGhvbGl9bWJlZGRlZC5j) | `75.22% <57.35%> (-2.44%)` | :arrow\_down: | |  
[src/connection\_manager.c](https://codecov.io/gh/apache/qpid-dispatch/pull/746/diff?src=pr&el=tree#diff-c3JlL2Nvbml5Y3Rpb25fbWVuYXdlci5j) | `91.26% <100.00%> (+0.01%)` |  
:arrow\_up: | | [src/server.c](https://codecov.io/gh/apache/qpid-dispatch/pull/746/diff?src=pr&el=tree#diff-c3JlL3NlcnZlci5j) | `86.36% <100.00%> (+0.20%)` | :arrow\_up: | | [src/container.c]  
(https://codecov.io/gh/apache/qpid-dispatch/pull/746/diff?src=pr&el=tree#diff-c3JlL2Nvbml5Y3Rpb25fbWVuYXdlci5j) |  
`82.31% <0.00%> (-0.17%)` | :arrow\_down: | | [src/router\_core/connections.c]  
(https://codecov.io/gh/apache/qpid-dispatch/pull/746/diff?src=pr&el=tree#diff-c3JlL3JvdXRlcl9jb3JlL2Nvbml5Y3Rpb25zLmM=) | `91.68% <0.00%> (-0.11%)` | :arrow\_down: | |  
[src/router\_node.c](https://codecov.io/gh/apache/qpid-dispatch/pull/746/diff?src=pr&el=tree#diff-c3JlL3JvdXRlcl9ub2RlLmM=) | `93.47% <0.00%> (+0.10%)` | :arrow\_up: | | ... and [1 more]  
(https://codecov.io/gh/apache/qpid-dispatch/pull/746/diff?src=pr&el=tree-more) | | ----- [Continue to  
review full report at Codecov](https://codecov.io/gh/apache/qpid-dispatch/pull/746?src=pr&el=continue).  
> **Legend** - [Click here to learn more](https://docs.codecov.io/docs/codecov-delta) > `Δ` = absolute  
<relative> (impact), `∅` = not affected, `?` = missing data > Powered by [Codecov]  
(https://codecov.io/gh/apache/qpid-dispatch/pull/746?src=pr&el=footer). Last update [be42638...0bff301]  
(https://codecov.io/gh/apache/qpid-dispatch/pull/746?src=pr&el=lastupdated). Read the [comment docs]  
(https://docs.codecov.io/docs/pull-request-comments). -----

--- This is an automated message from the Apache Git Service. To respond to the message, please log on to GitHub and use the URL above to go to the specific comment. For queries about this service, please contact Infrastructure at: users@infra.apache.org

8. [~lulf] , [~kwall] I've updated the feature description and have a pull request on github that is functionally complete. In terms of config file syntax - the key and values that appear between the braces of openProperties maps are specified in JSON syntax. Specifically: \* map keys are quoted ASCII strings \* map values can be any JSON type - quoted string, integer, null, true, false - and allow list and map nesting. Let me know what you think.
9. Commit a1e59d27e00d4902935421e6788a2d65d55f3ce8 in qpid-dispatch's branch refs/heads/master from Ken Giusti [ https://gitbox.apache.org/repos/asf?p=qpid-dispatch.git;h=a1e59d2 ] DISPATCH-1656: Add openProperties attribute to listener and connector This closes #746
10. asfgit closed pull request #746: URL: https://github.com/apache/qpid-dispatch/pull/746 -----  
----- This is an automated message from the Apache Git Service. To respond to the message, please log on to GitHub and use the URL above to go to the specific comment. For queries about this service, please contact Infrastructure at: users@infra.apache.org
11. Commit a350f7ba869885d51fecf31878676b7c633a7200 in qpid-dispatch's branch refs/heads/master from Ken Giusti [ https://gitbox.apache.org/repos/asf?p=qpid-dispatch.git;h=a350f7b ] DISPATCH-1656: fix test to support older json version
12. I think it looks great [~kgiusti] !
13. Commit 0d2e54ee7c7f6baf440de8a123253916dfb34ed5 in qpid-dispatch's branch refs/heads/master from Ken Giusti [ https://gitbox.apache.org/repos/asf?p=qpid-dispatch.git;h=0d2e54e ] DISPATCH-1656: do not allow openProperties on connections to router
14. Commit d8e09c4390fe452cec197fe860305ae4e8858925 in qpid-dispatch's branch refs/heads/master from Ken Giusti [ https://gitbox.apache.org/repos/asf?p=qpid-dispatch.git;h=d8e09c4 ] DISPATCH-1656: fix condition check error found by coverity