Runtime Infrastructure - Feature #1842

implement the multi-source directory approach

10/30/2012 01:58 PM - Greg Shah

Status:	New	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:		**Estimated time:**	160.00 hours
Target version:	Deployment and Management Improvements		
billable:	No	**vendor_id:**	GCD

Description

Related issues:

Related to Runtime Infrastructure - Feature #2316: improve directory XML enco...	**New**

History

#1 - 10/30/2012 02:05 PM - Greg Shah

Change the current directory approach from the reading of a single database to having a runtime directory representation that is a combined "database" that is potentially created from multiple sources. There is a precedence order to the sources, such that conflicting nodes get resolved based on the higher priority sources.

Precedence (1 to 6 where 1 is highest priority and 6 is lowest):

1. command line (syntax for the server command line that specifies the node path, type and value)
2. bootstrap config (syntax for the config file that specifies the node path, type and value)
3. external directory (an application-specific directory that is explicitly specified for the server)
4. application jar directory (directory values in a directory XML file that is found in the application jar)
5. default jar directory (default directory values in a directory XML file that found in the P2J jar)
6. default directory values in the P2J code (put better defaults in and expand the amount of places with good defaults so that this minimizes what has to be put in the other places; this is a big effort on its own)

The result is a virtual directory used at runtime. The idea is that this is significantly easier to manage since it allows the maximum amount of standard stuff to be hidden away and only the stuff that is different must be specified. The various means of specification gives great flexibility to implementers.

#2 - 10/30/2012 02:09 PM - Greg Shah

- Estimated time changed from 120.00 to 160.00

#3 - 10/31/2012 11:51 AM - Greg Shah

- Target version set to Deployment and Management Improvements

#4 - 04/21/2016 07:42 AM - Paul E

I've not come across this issue before - I like it and have been having similar (less well-structured) thoughts.

#5 - 04/21/2016 07:53 AM - Greg Shah

It should really minimize per-site/per-system configuration and allow most things to be configured at the most appropriate level, including quite a bit that can come directly from the jar.

If you have any ideas to enhance this, we are certainly open to hearing them. Considering that you have done quite a bit of deployment work with P2J, you qualify as one of leading players in that exclusive community. :)

#6 - 09/21/2016 09:03 AM - Greg Shah

An interesting question was posed by Igor:

```
-------- Forwarded Message --------
Subject:     Command line arguments and DirectoryService
Date:     Wed, 21 Sep 2016 08:29:41 -0400
From:     Igor
To:     ...

Gentlemen,
I have a question about the relation between command line arguments and
Directory service.
Imagine that I need to obtain a configuration value for the
"net/xml/compress" configuration parameter at the server side. As far as
I can see this can be done in two ways:
1. Specifynet:xml:compress argument in the command line and call a
corresponding method of the BootstrapConfig.
2. Add <net> container under the <standard/runtime/default> in the
directory.xml and use Utils.getDirectoryNodeBoolean call.

Are these two ways "orthogonal"? More specifically:
1. Is it possible to override the directory.xml settings from the
command line?
2. Is it possible to access settings specified in the command line with
Utils.getDirectoryNodeXXX call?

Thank you,
Igor.
```

The answer:

No the current design is not orthogonal.

The current implementation cannot override directory settings from the command line, but we want to provide this capability (see note 1 above).

The bootstrap configuration was originally designed to provide the minimum initial configuration to allow the server to start up. This means the values that are there are supposed to be ones that must be known before the directory has been read. Mostly it will include how to access the directory, passwords for keystores, truststores and some low level networking configuration. The client-side bootstrap config has the same idea except it is mostly about how to access the server, which client driver to use, some client-specific configuration values that must be known before the server can be contacted (in addition to the passwords for keystores, truststores).

#7 - 11/16/2016 01:06 PM - Greg Shah

- Target version changed from Deployment and Management Improvements to Deployment and Management Improvements

#8 - 10/07/2019 06:22 AM - Greg Shah

- Related to Feature #2316: improve directory XML encoding added

#9 - 12/06/2019 04:54 PM - Greg Shah

One customer mentioned to me that he would like to see the security sections of the directory as a separate XML file. The plan above may accommodate this idea though it is designed to be more general purpose (not just security portions).

He also really wanted a more condensed format for the contents of the directory, which we have planned as #2316.

#10 - 12/28/2020 06:33 PM - Greg Shah

One approach to implement this is to provide a syntax for "include" directives. This would allow a complex multi-file "tree" of configuration sources.