

Robotics Systems Commons - Tasks #127

Create a plugin mechanism

09/13/2010 05:24 PM - J. Wienke

Status:	Resolved	Start date:	09/29/2012
Priority:	Normal	Due date:	
Assignee:	J. Moringen	% Done:	100%
Category:	Plugins	Estimated time:	0.00 hour
Target version:	rsb-0.9		
Description			
This mechanism must allow to dynamically load extensions into previously defined registries from a directory of shared libraries.			
Subtasks:			
Tasks # 1176: Implement plugin mechanism for Windows			Resolved
Tasks # 1177: Implement plugin mechanism for MacOS			Resolved
Tasks # 1178: Add unit tests for plugin system			Resolved
Related issues:			
Blocks Robotics Service Bus - Feature # 305: Implement plugin mechanism		Resolved	05/20/2011
Blocked by Robotics Systems Commons - Tasks # 1420: Simplify plugin initiali...		Resolved	02/14/2013

Associated revisions

Revision dbea211d - 09/29/2012 03:24 PM - J. Moringen

Added a plugin system in `src/rsc/plugins/*.{h,cpp}`

refs #127

The entry point of the plugin system is the Manager class which maintains a plugin loadpath and a list of plugins which Plugin objects. Plugin object basically have a name, a load() and a unload() method.

NOTE: the current implementation is for Linux and is not portable.

- `src/rsc/plugins/Provider.h`: new file; API for plugin implementors
- `src/rsc/plugins/Plugin.{h,cpp}`: new files; contain Plugin class
- `src/rsc/plugins/Manager.{h,cpp}`: new files; contain Manager class, a singleton, which manages the plugin load path and plugin list
- `src/rsc/plugins/Configurator.{h,cpp}`: new files; contain PluginConfigurator helper class which is used to configure the plugin load path and load certain plugins according to the configuration mechanism
- `src/CMakeLists.txt`: added files `src/rsc/plugins/*.{h,cpp}`; link against `libdl`

Revision 8ffc839e - 09/29/2012 03:40 PM - J. Moringen

Added a plugin system in `src/rsc/plugins/*.{h,cpp}`

refs #127

The entry point of the plugin system is the Manager class which maintains a plugin loadpath and a list of plugins which Plugin objects. Plugin object basically have a name, a load() and a unload() method.

NOTE: the current implementation is for Linux and is not portable.

- src/rsc/plugins/Provider.h: new file; API for plugin implementors
- src/rsc/plugins/Plugin.{h,cpp}: new files; contain Plugin class
- src/rsc/plugins/Manager.{h,cpp}: new files; contain Manager class, a singleton, which manages the plugin load path and plugin list
- src/rsc/plugins/Configurator.{h,cpp}: new files; contain PluginConfigurator helper class which is used to configure the plugin load path and load certain plugins according to the configuration mechanism
- src/CMakeLists.txt: added files src/rsc/plugins/*.{h,cpp}; link against libdl

Revision 1c00f507 - 09/29/2012 04:09 PM - J. Moringen

Added a plugin system in src/rsc/plugins/*.{h,cpp}

refs #127

The entry point of the plugin system is the Manager class which maintains a plugin loadpath and a list of plugins which Plugin objects. Plugin object basically have a name, a load() and a unload() method.

NOTE: the current implementation is for Linux and is not portable.

- src/rsc/plugins/Provider.h: new file; API for plugin implementors
- src/rsc/plugins/Plugin.{h,cpp}: new files; contain Plugin class
- src/rsc/plugins/Manager.{h,cpp}: new files; contain Manager class, a singleton, which manages the plugin load path and plugin list
- src/rsc/plugins/Configurator.{h,cpp}: new files; contain PluginConfigurator helper class which is used to configure the plugin load path and load certain plugins according to the configuration mechanism
- src/CMakeLists.txt: added files src/rsc/plugins/*.{h,cpp}; link against libdl

Revision 1f3f4b49 - 09/29/2012 05:01 PM - J. Moringen

Added a plugin system in src/rsc/plugins/*.{h,cpp}

refs #127

The entry point of the plugin system is the Manager class which maintains a plugin loadpath and a list of plugins which Plugin objects. Plugin object basically have a name, a load() and a unload()

method.

NOTE: the current implementation is for Linux and is not portable.

- src/rsc/plugins/Provider.h: new file; API for plugin implementors
- src/rsc/plugins/Plugin.{h,cpp}: new files; contain Plugin class
- src/rsc/plugins/Manager.{h,cpp}: new files; contain Manager class, a singleton, which manages the plugin load path and plugin list
- src/rsc/plugins/Configurator.{h,cpp}: new files; contain PluginConfigurator helper class which is used to configure the plugin load path and load certain plugins according to the configuration mechanism
- src/CMakeLists.txt: added files src/rsc/plugins/*.{h,cpp}; link against libdl

History

#1 - 03/13/2012 05:18 PM - J. Wienke

- Assignee deleted (J. Wienke)

#2 - 09/13/2012 12:16 PM - J. Moringen

- Status changed from New to In Progress
- Assignee set to J. Moringen
- Target version set to rsb-0.9

#3 - 09/16/2012 03:09 PM - J. Moringen

- File 0001-Added-a-plugin-system-in-src-rsc-plugins-.h-cpp.patch added
- Status changed from In Progress to Feedback
- % Done changed from 0 to 70

Any feedback (patch is attached in case my email did not get through)?

#4 - 09/29/2012 04:13 PM - J. Moringen

- Category changed from Registry to Plugins
- % Done changed from 70 to 90

#5 - 06/05/2013 09:38 PM - J. Moringen

- Status changed from Feedback to Resolved

Files

0001-Added-a-plugin-system-in-src-rsc-plugins-.h-cpp.patch	27 KB	09/16/2012	J. Moringen
--	-------	------------	-------------