Moto Dynamic Loader Re-Factor

Design Goals

 Refactor the extension loader so it can be used outside of the moto context to load moto extensions. Specifically it should not be dependent on MotoEnv

Design Decisions

 MX should become MotoExtension to provide the sort of readability improvement we are looking for.

 The Dynamic Loader is extremely dependant on MotoEnv right now.

1) In allocates MX records in the moto env->mpool.
2) It adds the includes needed by the moto c code generator to env->includes
3) It checks for the registration of types with moto_getType and if they are not registered it adds them with moto_addType
4) It adds MotoFuntions to env->ftable and tracks function memory in env->mpool

Tracking of memory in env->mpool

There are two ways to deal with this. First we could pass a pool directly to moto_loadLibrary. Second, we could add a pool (or reference to the pool) in ftable and pass in the ftable directly

Adding of Includes

First of all this is just going to be one of many similar activities that will be required by the new, more powerful mxc. We will need to be able to collect up InlcudePaths, LibraryPaths, libraries and archives as well as includes for loaded extensions.

Registration of Types

Registration of Functions and Methods

 moto_use in env.c is the only function to call moto_loadLibrary. This is where the cut should be made. All env interaction should occur here and loadLibrary should return a MotoExtension* only.

Potential Designs

MotoExtensionLoader
/* Returns the path on disk of where the shared library for a moto extension of the form foo.bar.maka is located */
String find(String usename)

/* Returns a MotoExtension object for this
MotoExtension load(String usename)

MotoExtension
/* Returns an Enumeration of String types that this extension defines */
Enumeration mext_getTypes(MotoExtension* mx)

```
/* Returns an Enumeration of MotoFunction *s this extension defines */
Enumeration mext_getFunctions(MotoExtension* mx)

/* Returns an Enumeration of String includes that this extension defines */
Enumeration mext_getIncludes(MotoExtension* mx)

/* Reurns an Enumeration of IncludePaths that are required to compile a generated C file
that requires this extension */
Enumeration mext_getIncludePaths(MotoExtension* mx)

/* Returns an Enumeration of LibraryPaths that are required to link with this extension */
Enumeration mext_getLibraryPaths(MotoExtension* mx)

/* Returns an Enumeration of .a files that are requires to link with this extension */
Enumeration mext_getArchives(MotoExtension* mx)

/* Returns an Enumeration of shared libraries that this Extension depends upon */
Enumeration mext_getLibraries(MotoExtension* mx)

/* Returns the path on disk to this Extension */
String mext_getPath(MotoExtension* mx)
```