

common-lisp-config.lisp

Erik Winkels

August 13, 2009

Contents

1	Variables	1
---	-----------	---

1 Variables

`*LISP-OUT-DIR*` defines what the root directory of the generated Common Lisp bindings should be. I only run the bindings generator from Slime after having done a “`slime-cd`” to the “bindings-generator” directory, so keep this in mind!

Also, “okra-bindings.asd” is actually placed at “(mkstr *lisp-out-dir* “..””, so keep that in mind as well :-)

```
(defparameter *lisp-out-dir* "generated/src-bindings/")

(defparameter *lisp-files* nil) ; for ASDF
(defparameter *lisp-symbols* nil) ; for package.lisp
```

I’ve declared a lot of crap here as a :pointer of which I don’t even know what it is. (Generally camelCaps names that don’t end with a ‘*’ or ‘Ptr’.)

Also, with the advent of support for overloaded functions and the additional mess that comes with it, all `cdr`’s should become function calls or `plist` or whatever so the mapping for overloaded types can be merged with this table AND so we can do away with a lot of checks in the code that checks whether the `cdr` is either a string or a symbol.

I could really shorten this list if I don’t lookup simple types and handle everything not in this list as a `cl-pointer`.

```
(defparameter *lisp-type-mapping*
  '(;; standard types
    ("void" . cl-void)
    ("bool" . cl-boolean)
    ("double" . cl-simple-type)
    ("float" . cl-simple-type)
    ("int" . cl-simple-type)
    ("long" . cl-simple-type)
    ("size_t" . cl-size-t)
    ("std::string" . cl-string)
    ("uchar" . cl-simple-type))
```

```

(uint" . cl-simple-type)
(uint8" . cl-simple-type)
(uint16" . cl-simple-type)
(uint32" . cl-simple-type)
(ushort" . cl-simple-type)
(unsigned char" . cl-simple-type)
(unsigned int" . cl-unsigned-int)
(unsigned long" . cl-simple-type)
(unsigned long long" . cl-simple-type)
(unsigned short" . cl-simple-type)
("time_t" . cl-long) ; at least on my machine :-)
;; Ogre types
("AxisAlignedBox" . cl-array6) ; does this compute with cpp-config.lisp?
("AnimableValuePtr" . cl-pointer)
("AnimationIterator" . cl-pointer)
("AnimationStateIterator" . cl-pointer)
("CameraIterator" . cl-pointer)
("ChildNodeIterator" . cl-pointer)
("ChildObjectListIterator" . cl-pointer)
("ConstChildNodeIterator" . cl-pointer)
("ConstObjectIterator" . cl-pointer)
("ConstShadowTextureConfigIterator" . cl-pointer)
("DataStreamListPtr" . cl-pointer)
("DataStreamPtr" . cl-pointer)
("FileInfoListPtr" . cl-pointer)
("Light::LightTypes" . cl-light-types)
("Matrix3" . cl-array9)
("MeshPtr" . cl-pointer)
("SceneManagerEnumerator::MetaDataIterator" . cl-pointer)
("MovableObjectFactoryIterator" . cl-pointer)
("MovableObjectIterator" . cl-pointer)
("ObjectIterator" . cl-pointer)
("Overlay2DElementsIterator" . cl-pointer)
("OverlayMapIterator" . cl-pointer)
("PlaneBoundedVolume" . cl-pointer)
("Ray" . cl-array6)
("Real" . cl-okra-real)
("RenderOperation::OperationType" . cl-operation-type)
("RenderSystemList*" . cl-render-system-list)
("ResourceDeclarationList" . cl-pointer)
("ResourceManagerIterator" . cl-pointer)
("SceneManagerEnumerator::SceneManagerIterator" . cl-pointer)
("ShadowRenderableListIterator" . cl-pointer)
("String" . cl-string)
("StringVector" . cl-pointer)
("StringVectorPtr" . cl-pointer)
("TemplateIterator" . cl-pointer)
("TransformSpace" . cl-pointer)
("UTFString" . cl-string)
("Vector2" . cl-array2)

```

```
("Vector3" . cl-array3)
("Vector4" . cl-array4)
("ViewPoint" . cl-pointer)
("const String&" . cl-string)
("const AxisAlignedBox&" . cl-array6)
("const ColourValue&" . cl-array4)
("const DisplayString&" . cl-string)
("const Matrix3&" . cl-array9)
("const Matrix4&" . cl-array16)
("const Quaternion&" . cl-array4)
("const Plane&" . cl-plane)
("const Radian&" . cl-okra-real)
("const Sphere&" . cl-array4)
("const String&" . cl-string)
("const Vector2&" . cl-array2)
("const Vector3&" . cl-array3)
("const Vector4&" . cl-array4)
("const Vector3*" . cl-array3)))
```