Core issue 743: `decltype(...)` name qualifiers

Notes
The wording changes proposed in this paper address national body comment JP 8 (Core issue 743). The changes are against N2914.

Wording Changes
In 3.4.3 [basic.lookup.qual] paragraph 1 change the first two sentences

The name of a class, concept map (but not a concept), or namespace member or enumerator can be referred to after the `::` scope resolution operator (5.1) applied to a `nested-name-specifier` that nominates its class, concept map, namespace, or enumeration. During the lookup for a name preceding the `::` scope resolution operator, object, function, and enumerator names are ignored.

To

The name of a class, concept map (but not a concept), or namespace member or enumerator can be referred to after the `::` scope resolution operator that denotes its class, concept map, namespace, or enumeration. If a `::` scope resolution operator in a `nested-name-specifier` is not preceded by a `decltype-specifier`, lookup of the name preceding that `::` considers only namespaces, types, and templates whose specializations are types.

Add a production to the grammar rule for `nested-name-specifier` in 5.1.1 [expr.prim.general] paragraph 6 as follows

6 ... `nested-name-specifier:
dcltype-specifier ::
type-name ::
namespace-name ::
nested-name-specifier identifier ::
nested-name-specifier templateopt simple-template-id ::
nested-name-specifieropt concept-id ::`

Change the first sentence following this grammar rule from

A `nested-name-specifier` that names a class, optionally followed by the keyword `template` ...
to

A nested-name-specifier that denotes a class, optionally followed by the keyword template ...

In 5.1.1 [expr.prim.general] paragraph 8 change the first sentence from

8 A nested-name-specifier that names an enumeration ...

to

8 A nested-name-specifier that denotes an enumeration ...

In 7.1.6.2 [dcl.type.simple] paragraph 1 replace the production

    simple-type-specifier:
         ...
         decltype ( expression )

by

    simple-type-specifier:
         ...
         decltype-specifier

and add the following rule:

    decltype-specifier:
         decltype ( expression )

In 8.3.3 [dcl.mptr] paragraph 1 change the phrase

    the nested-name-specifier names a class

by

    the nested-name-specifier denotes a class

(one occurrence).

In 11.2 [class.access.base] paragraph 5 change the phrase

    class named by the nested-name-specifier

by

    class denoted by the nested-name-specifier

(one occurrence).

In 11.5 [class.protected] paragraph 1 change the phrase
the `nested-name-specifier` shall name
by
the `nested-name-specifier` shall denote
(one occurrence).

In 12.9 `class.inhctor` paragraph 8 change the phrase
the base class named in the `nested-name-specifier`
by
the base class denoted by the `nested-name-specifier`
(one occurrence).

In 14.7.2.4 `temp.dep.temp` change paragraph 4

4 A template `template-argument` is dependent if it names a `template-parameter` or is a
qualified-id with a `nested-name-specifier` which contains a `class-name` that names a
dependent type.

4 A template `template-argument` is dependent if it names a `template-parameter` or is a
qualified-id with a `nested-name-specifier` which contains a `class-name` or a `decltype-
specifier` that denotes a dependent type.