

Core issue 374: Explicit specialization outside a template's parent (revision 1)

Notes

The changes are against N3035.

Wording Changes

Change 7.3.1.2 [] paragraph 2 as follows:

- 2 Members ~~(including explicit specializations of templates (14.8.3))~~ of a named namespace can also be defined outside that namespace by explicit qualification (3.4.3.2) of the name being defined, provided that the entity being defined was already declared in the namespace and the definition appears after the point of declaration in a namespace that encloses the declaration's namespace. ...

In 8.3 [dcl.meaning] paragraph 1, change the following sentences as indicated:

... A *declarator-id* shall not be qualified except for the definition of a member function (9.3) or static data member (9.4) outside of its class, the definition or explicit instantiation of a function or variable member of a namespace outside of its namespace, or the definition of a ~~previously declared~~ explicit specialization outside of its namespace, or the declaration of a friend function that is a member of another class or namespace (11.4). When the *declarator-id* is qualified, the declaration shall refer to a previously declared member of the class or namespace to which the qualifier refers (or of an inline namespace within that scope (7.3.1)) ~~or to a specialization thereof~~, and the member shall not have been introduced by a *using-declaration* in the scope of the class or namespace nominated by the *nested-name-specifier* of the *declarator-id*.

Change 14.8.3 [temp.expl.spec] paragraph 2 as follows:

- 2 An explicit specialization shall be declared in a namespace enclosing the specialized template. An explicit specialization whose *declarator-id* is not qualified shall be declared in the nearest enclosing namespace of the template, or, if the namespace is inline (7.3.1), any namespace from its enclosing namespace set. Such a declaration may also be a definition. If the declaration is not a definition, the specialization may be defined later (7.3.1.2).

Change 14.8.3 [temp.expl.spec] paragraph 3 as follows:

- 3 A declaration of a function template or class template being explicitly specialized shall ~~be in scope at the point of~~ precede the declaration of ~~an~~ the explicit specialization. [*Note:* a declaration, but not a definition of the template is required. — *end note*] The definition of a class or class template shall ~~be in scope at the point of~~ precede the declaration of an explicit specialization for a member template of the class or class template.

Change 14.8.3 [temp.expl.spec] paragraph 4 as follows:

- 4 A member function, a member class or a static data member of a class template may be explicitly specialized for a class specialization that is implicitly instantiated; in this case, the definition of the class template shall ~~be in scope at the point of declaration~~ precede the explicit specialization for the member of the class template. If such an explicit specialization for the member of a class template names an implicitly-declared special member function (Clause 12), the program is ill-formed.