Add make_proxy for the Pointer-Semantics-Based Polymorphism Library - Proxy

Project: ISO/IEC 14882 Programming Languages — C++, ISO/IEC JTC1/SC22/WG21
Authors: Tian Liao, Mingxin Wang
Reply-to: Tian Liao <tilia@microsoft.com>
Audience: LEWG

Abstract: *Proxy* is a new library feature that is being proposed to delegate general pointer types with the type-erased technique to support non-intrusive polymorphism programming in C++. The focus of paper is on a utility function – *make_proxy*, which used to be included in the *proxy* proposal as a sub-feature. We believe it's a useful tool to help allocate *proxy* instances properly.

1 Introduction

Paper 3086 proposed a *Pointer-Semantics-Based Polymorphism Library*, which is designed to help people build extendable and efficient polymorphic programs with better abstractions and less intrusive code. This paper is proposing the utility part that is separated from early versions of paper 3086 and paper 0957.

More specifically, we are eager to add function template *make_proxy* together with the *proxy* library into the standard as a library feature. *make_proxy*'s syntax is similar to the constuctors of `std::any`. It is designed to provide simple ways to construct *proxy* instances from values. With *make_proxy*, SBO (small buffer optimization) may implicitly apply, depending on library implementation, to reduce the potential overheads that may come from invoking *proxy*'s constructors directly.

2 Motivation

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

3 Technical specification

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.