

Chapter 1

Topic Documentation

1.1 User-defined type support class library

Classes

- class **lfc1::udtsup::CManipBase**< T1 >

This class template provides exception/error handling for single argument manipulators of user-defined types.

Functions

- template<typename T2>
void **lfc1::udtsup::gvDoManip** (std::basic_ios< char > &arIos, const **CManipBase**< T2 > &arManipBase)
This template function calls the user-defined manipulator function and handles exceptions from it.
- template<typename T3>
std::istream & **lfc1::udtsup::operator**>> (std::istream &arStream, const **CManipBase**< T3 > &arManipBase)
This template function invokes a user-defined manipulator for an input stream.
- template<typename T3>
std::ostream & **lfc1::udtsup::operator**<< (std::ostream &arStream, const **CManipBase**< T3 > &arManipBase)
This template function invokes a user-defined manipulator for an output stream.
- template<typename UDT>
std::istream & **lfc1::udtsup::gvInput** (std::istream &arStream, UDT &arUdt)
This template function is meant to be the only function called by the extractor of a user-defined type. This function provides exception/error handling for the extractor of a user-defined type.
- template<typename UDT>
std::istream & **lfc1::udtsup::gvInputGuarded** (std::istream &arStream, UDT &arUdt)
This template function is meant to be the only function called by the extractor of a user-defined type. This function provides exception/error handling for the extractor of a user-defined type.
- template<typename UDT>
std::ostream & **lfc1::udtsup::gvOutput** (std::ostream &arStream, const UDT &arUdt)
This template function is meant to be the only function called by the inserter of a user-defined type. This function provides exception/error handling for the extractor of a user-defined type.
- template<typename UDT>
std::ostream & **lfc1::udtsup::gvOutputGuarded** (std::ostream &arStream, const UDT &arUdt)
This template function is meant to be the only function called by the inserter of a user-defined type. This function provides exception/error handling for the extractor of a user-defined type.

1.1.1 Detailed Description

This library contains a set of templates which provides exception/error handling for inserters, extractors and manipulators of user-defined types. These templates handle exceptions derived from `std::bad_alloc`, `std::exception` and unknown exceptions.

1.1.2 Function Documentation

1.1.2.1 `gvDoManip()`

```
template<typename T2>
void lfcl::udtsup::gvDoManip (
    std::basic_ios< char > & arIos,
    const CManipBase< T2 > & arManipBase)
```

This template function calls the user-defined manipulator function and handles exceptions from it.

Template Parameters

<i>T2</i>	The user-defined manipulator function argument type.
-----------	--

Parameters

<i>arIos</i>	The stream's state object.
<i>arManipBase</i>	The user-defined manipulator function container.

1.1.2.2 `gvInput()`

```
template<typename UDT>
std::istream & lfcl::udtsup::gvInput (
    std::istream & arStream,
    UDT & arUdt)
```

This template function is meant to be the only function called by the extractor of a user-defined type. This function provides exception/error handling for the extractor of a user-defined type.

This template was derived from Section 3.1.5 of the book *Standard C++ IOSTreams and Locales*. The exception handling sets the stream state and allows the handled exception to propagate if the stream allows it.

The user-defined type is expected to contain a function with the following signature:

```
@code
std::ios_base::iostate mvInput(std::istream&)
@endcode
```

The `mvInput()` function can call extractors for built-in types thus this function does not instantiate a stream sentry object.

Template Parameters

<i>UDT</i>	The user-defined type.
------------	------------------------

Parameters

in	<i>arStream</i>	The source stream.
out	<i>arUdt</i>	The user-defined type.

Returns

The source stream.

1.1.2.3 `gvInputGuarded()`

```
template<typename UDT>
std::istream & lfc1::udtsup::gvInputGuarded (
    std::istream & arStream,
    UDT & arUdt)
```

This template function is meant to be the only function called by the extractor of a user-defined type. This function provides exception/error handling for the extractor of a user-defined type.

This template was derived from Section 3.1.5 of the book *Standard C++ IOStreams and Locales*. The exception handling sets the stream state and allows the handled exception to propagate if the stream allows it.

The user-defined type is expected to contain a function with the following signature:

```
@code
std::ios_base::iostate mvInput(std::istream&)
@endcode
```

The `mvInput()` function must not call extractors for built-in types because this function instantiates a stream sentry object.

Template Parameters

<i>UDT</i>	The user-defined type.
------------	------------------------

Parameters

in	<i>arStream</i>	The source stream.
out	<i>arUdt</i>	The user-defined type.

Returns

The source stream.

1.1.2.4 gvOutput()

```
template<typename UDT>
std::ostream & lfc1::udtsup::gvOutput (
    std::ostream & arStream,
    const UDT & arUdt)
```

This template function is meant to be the only function called by the inserter of a user-defined type. This function provides exception/error handling for the extractor of a user-defined type.

This template was derived from Section 3.1.5 of the book *Standard C++ IOStreams and Locales*. The exception handling sets the stream state and allows the handled exception to propagate if the stream allows it.

The user-defined type is expected to contain a function with the following signature:

```
@code
std::ios_base::iostate mvOutput(std::istream&)
@endcode
```

The mvOutput() function can call inserters for built-in types thus this function does not instantiate a stream sentry object.

Template Parameters

<i>UDT</i>	The user-defined type.
------------	------------------------

Parameters

in	<i>arStream</i>	The destination stream.
out	<i>arUdt</i>	The user-defined type.

Returns

The destination stream.

1.1.2.5 gvOutputGuarded()

```
template<typename UDT>
std::ostream & lfc1::udtsup::gvOutputGuarded (
    std::ostream & arStream,
    const UDT & arUdt)
```

This template function is meant to be the only function called by the inserter of a user-defined type. This function provides exception/error handling for the extractor of a user-defined type.

This template was derived from Section 3.1.5 of the book *Standard C++ IOStreams and Locales*. The exception handling sets the stream state and allows the handled exception to propagate if the stream allows it.

The user-defined type is expected to contain a function with the following signature:

```
@code
std::ios_base::iostate mvOutput(std::istream&)
@endcode
```

The mvOutput() function must not call inserters for built-in types because this function instantiates a stream sentry object.

Template Parameters

<i>UDT</i>	The user-defined type.
------------	------------------------

Parameters

<i>in</i>	<i>arStream</i>	The destination stream.
-----------	-----------------	-------------------------

out	<i>arUdt</i>	The user-defined type.
-----	--------------	------------------------

Returns

The destination stream.

1.1.2.6 operator<<()

```
template<typename T3>
std::ostream & lfc1::udtsup::operator<< (
    std::ostream & arStream,
    const CManipBase< T3 > & arManipBase)
```

This template function invokes a user-defined manipulator for an output stream.