

```
std::type_identity< std::variant< Ts... > >
```

```
std::type_identity< std::variant< Ts... > >
```

```
templa::convert::convert_to_variant< From< Ts... > >
```

```
graph BT; A[templa::convert::convert_to_variant< From< Ts... > >] --> B[std::type_identity< std::variant< Ts... > >]; A --> C[std::type_identity< std::variant< Ts... > >];
```

The diagram illustrates a conversion from a templated variant to a standard variant. A box at the bottom contains the code `templa::convert::convert_to_variant< From< Ts... > >`. Two arrows originate from this box: one points to the left box containing `std::type_identity< std::variant< Ts... > >`, and the other points to the right box containing `std::type_identity< std::variant< Ts... > >`. This suggests that the templated function can be used to convert a templated variant into a standard `std::variant` type.