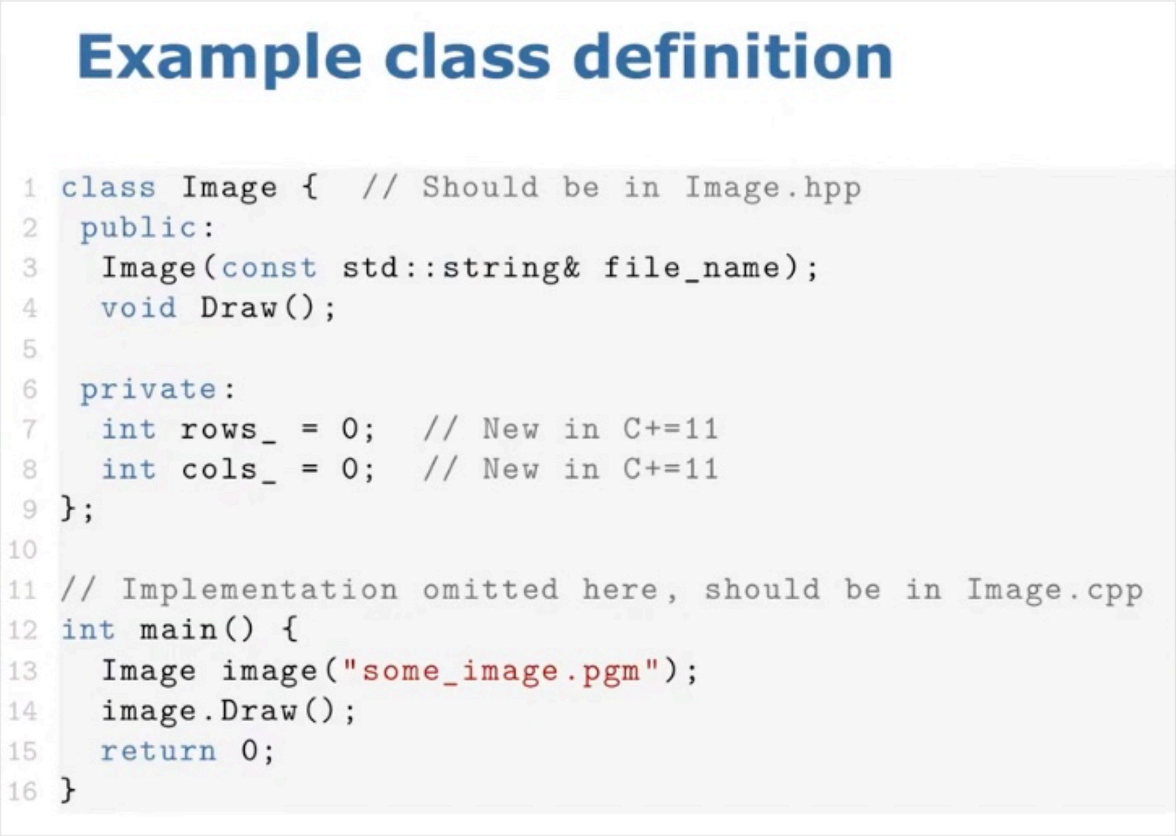
Class Basics

C++ Classes are a tools for creating **new types** that can be used as conveniently as the built-in types. In addition, derived classes allow the programmer to express relationships among classes and to take advantage of such relationships.

A type is a concrete representation of a concept. A program, that provides types that closely match the concepts of the application tends to be easier to understand, easer to reason about and easier to modify than program the does not.

- A class is a user-defined type.
- A class consists of a set of members (member data and member functions).
- Members can be accessed using . (dot) for objects and -> (arrow) got pointers.
- Operators, such as +, !, and [] can be defined for a class.
- A class is a namespace containing its members.
- A struct is a class where all members are public.



Example class definition

```cpp
class Image {  // Should be in Image.hpp
 public:
  Image(const std::string& file_name);
  void Draw();

 private:
  int rows_ = 0;   // New in C+=11
  int cols_ = 0;   // New in C+=11
};

// Implementation omitted here, should be in Image.cpp
int main() {
  Image image("some_image.pgm");
  image.Draw();
  return 0;
}
```

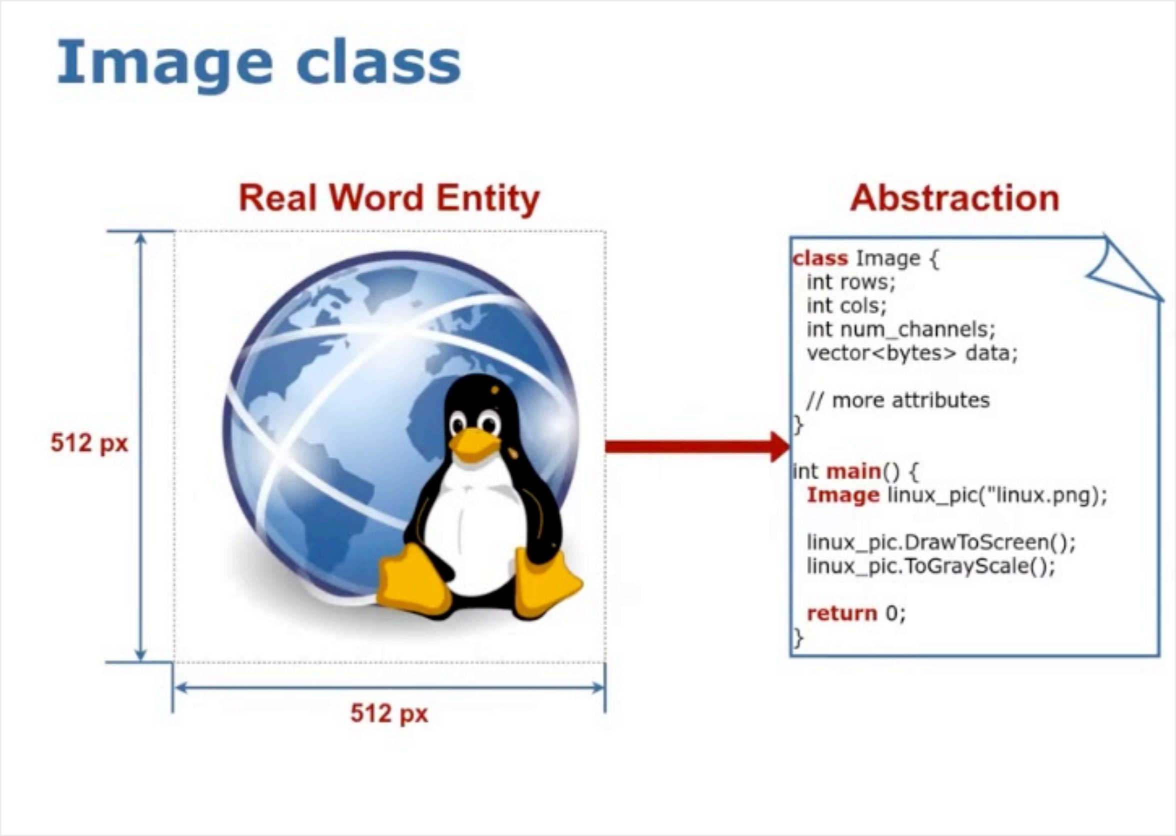


Image class

Real Word Entity

512 px
512 px

Abstraction

```
class Image {
  int rows;
  int cols;
  int num_channels;
  vector<bytes> data;

  // more attributes
}

int main() {
  Image linux_pic("linux.png");

  linux_pic.DrawToScreen();
  linux_pic.ToGrayScale();

  return 0;
}
```