

StepWise Refinement

This example is taken from Horstmann and Budd's textbook, "Big C++".

One of the most powerful strategies for problem solving is the process of **Stepwise Refinement**. Stepwise refinement simply says that to solve a difficult problem, break it down into a set of simpler tasks. Then keep breaking down each of these simpler tasks into even simpler ones, until you end up with tasks that you know how to solve. These smaller, simpler tasks are usually excellent candidates for methods. Such methods are well focused, small, and do one thing well.

Consider this problem from real life. You get up in the morning and simply must **get some orange juice**. How do you **get orange juice**? You can see if someone else can bring you some. If this fails, then you have to **make orange juice**. How do you **make orange juice**? If there is some frozen orange juice in the freezer, then you can **make frozen orange juice**. How do you **make frozen orange juice**? Simply **thaw frozen orange juice** and add water. How do you **thaw frozen orange juice**? Just **open the can** and **put can in the microwave** for a few minutes. What if you don't have any frozen orange juice? Then you will have to **make fresh orange juice**. How do you **make fresh orange juice**? Simply **slice an orange in half**, then **squeeze out the juice**.

The solution to the orange juice problem breaks down tasks in two different ways, with **decisions** and **refinements**. You are familiar with decisions. **If** you have frozen orange juice, use it, **else** use fresh oranges. Decisions are implemented with **if** and **if/else** statements. A refinement names a composite task and later breaks that task down into smaller pieces: **thaw frozen orange juice**: **open the can ... put can in microwave**.

This process is illustrated in the diagram below:

