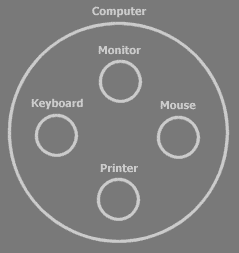
|  |
| --- |
| Circle Language Spec |

## Introduction

This chapter aims to introduce the basics of Circle language. Circle language is an idea for a computer programming language, that is mostly a diagram notation for visualizing computer code.

### Splitting up Ideas

Something that might play a role in making software, is splitting up a larger idea into smaller ideas.



Here you can see the idea of *the computer*, split up into four sub ideas: the monitor, keyboard, mouse and printer.

In Circle language, the main symbol for an idea might be a *circle*:

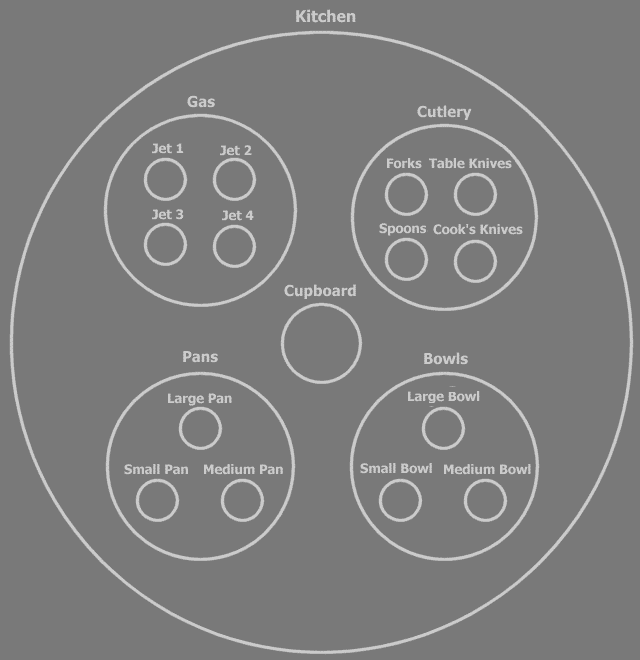


Each sub idea could work more or less independently of the othes. **Monitor** does something, **Keyboard** does something, and so on. It's the larger idea that could tie the sub ideas together. Meaning: the computer could make the link between monitor, keyboard, mouse and printer. The larger idea might combine the sub ideas and could manage the communication between the sub ideas.

In computer programming, ideas, so both larger ideas and sub ideas, might be called *objects*. Each object can be responsible for its own part of the system.



Each sub idea can be split up in sub ideas itself and you can go on and on splitting up ideas.



But to see the general point of the system, you might only need to see the top layers of the split up.



This can make object oriented programming an nice way to keep overview over a system as it grows. The understandability of the design could be helped by a clear the split up into ideas is, perhaps the top layers most of all. Depending on who you ask, a subdivision in objects might be the most important thing in software. So splitting up an idea into sub ideas, something that most people are probably able to do, might actually be the most important thing about programming.