

LECTURER: TAI LE QUY

INTRODUCTION TO COMPUTER SCIENCE

Basic Concepts of Data Processing

1

Information Representation

2

Algorithms and Data Structures

3

Propositional Logic, Boolean Algebra and Circuit Design

4

Hardware and Computer Architectures

5

Networks and the Internet

6

Software

7

Computer Science as a Discipline

8

Unit 7

SOFTWARE



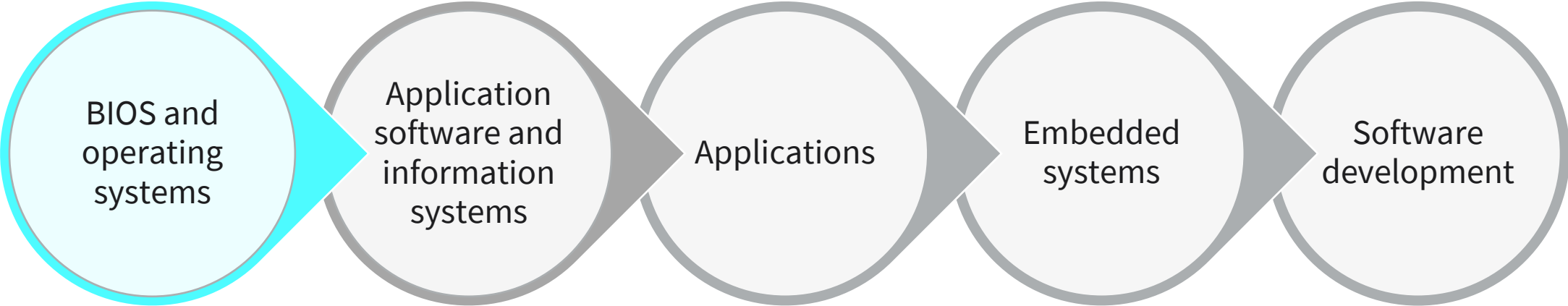
On completion of this unit, you will have learned ...

- ... how BIOS works.
- ... about modern file systems.
- ... about application software for operating systems.
- ... about the concept of embedded systems.
- ... about the process of software development.
- ... what the functions of an operating system are.



1. What is the difference between the motherboard and the BIOS chip?
2. What is OSS Open Source Software?
3. What are user requirements?

SOFTWARE

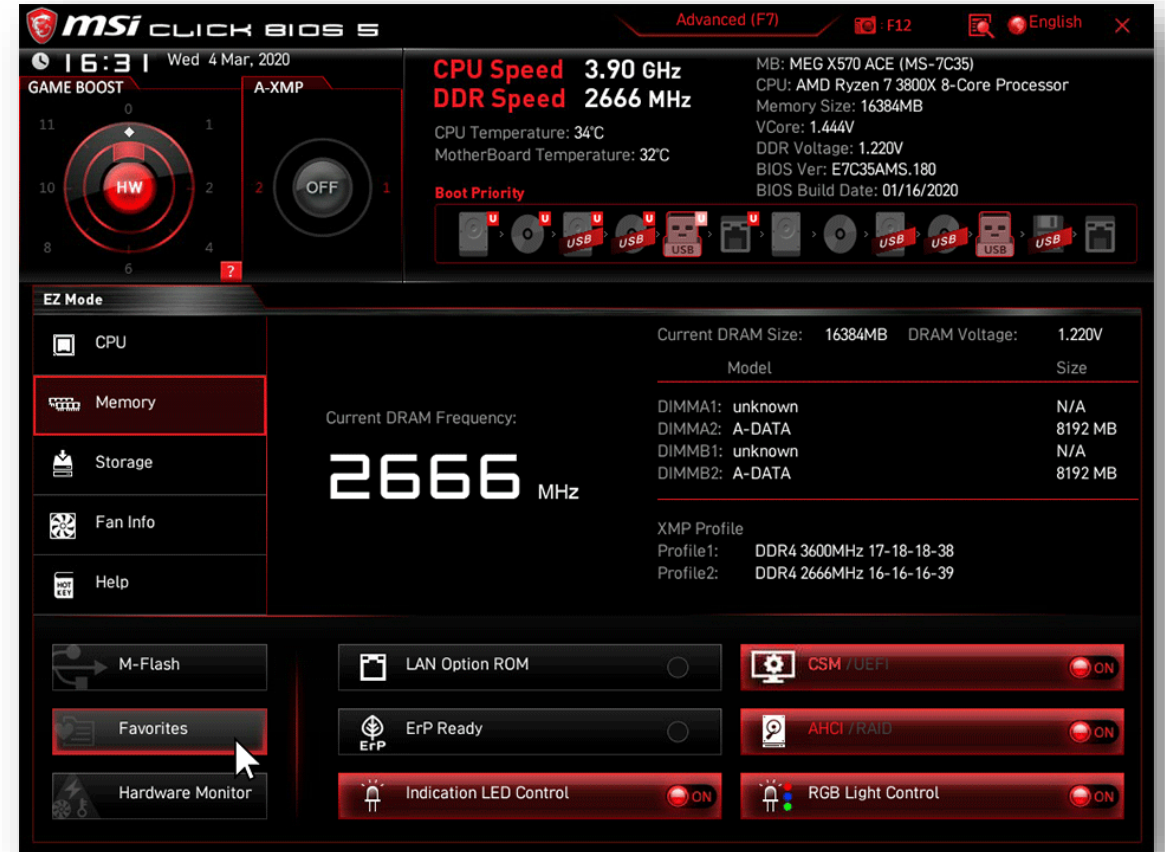


PC'S HYPOTHALAMUS - CPU'S BRAIN

When you first turn on your PC, your PC requires information:

- to detect PC components
- to find the operating system(floppy disk, hard drive, USB, network or a CD-ROM)

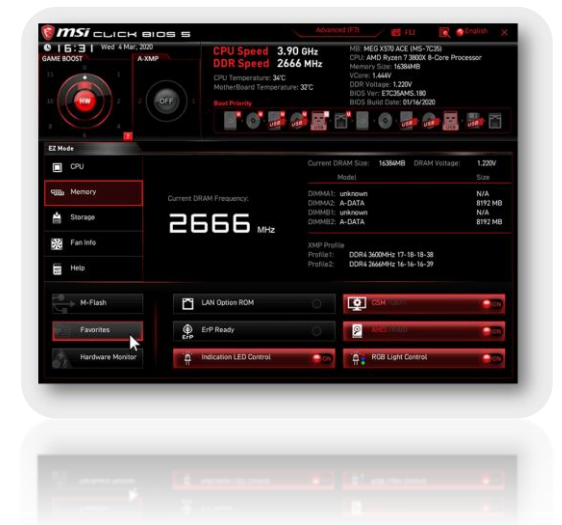
This information is stored in the **BIOS**.



BIOS

BIOS is mostly for IBM-compatible computers.

- It is a boot firmware containing an essential set of routines , designed to be the first code run by a PC when switched on.
- Its main purpose is to prepare the machine into a known state, so that software stored on compatible media can be loaded, executed, and given control of the PC.
- It also provides basic software drivers for all peripheral technologies.



BIOS FUNCTIONS

Power-on Self Test (POST)

Takes place right after you switch the PC on. It will test computer hardware, ensuring hardware is properly functioning before starting the process of loading the operating system.

If the POST is successful, BIOS calls **INT 19 (Interrupt 19)** and then proceeds to look for devices attached to the motherboard.

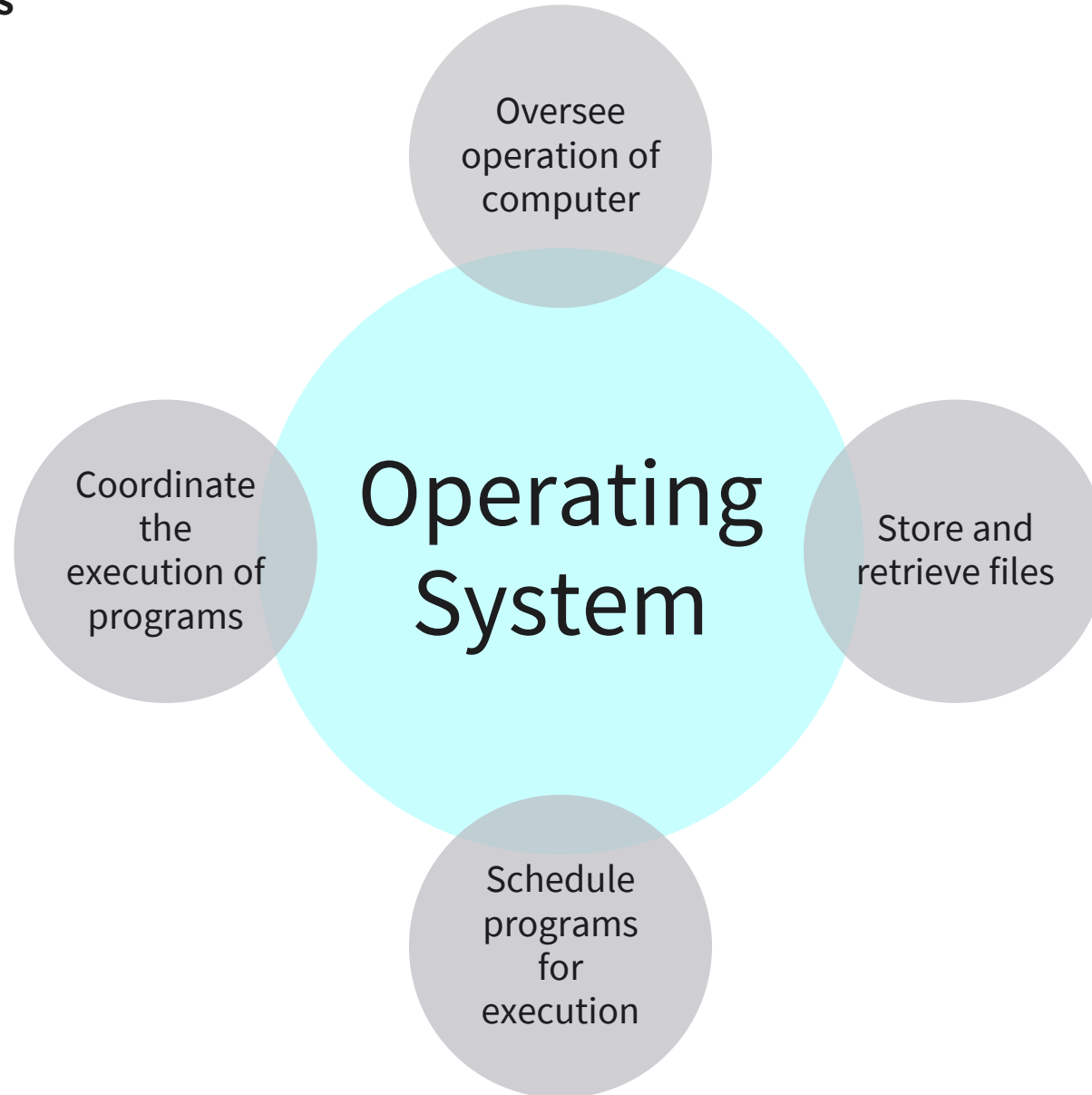
Bootstrap Loader

Process of locating the operating system.

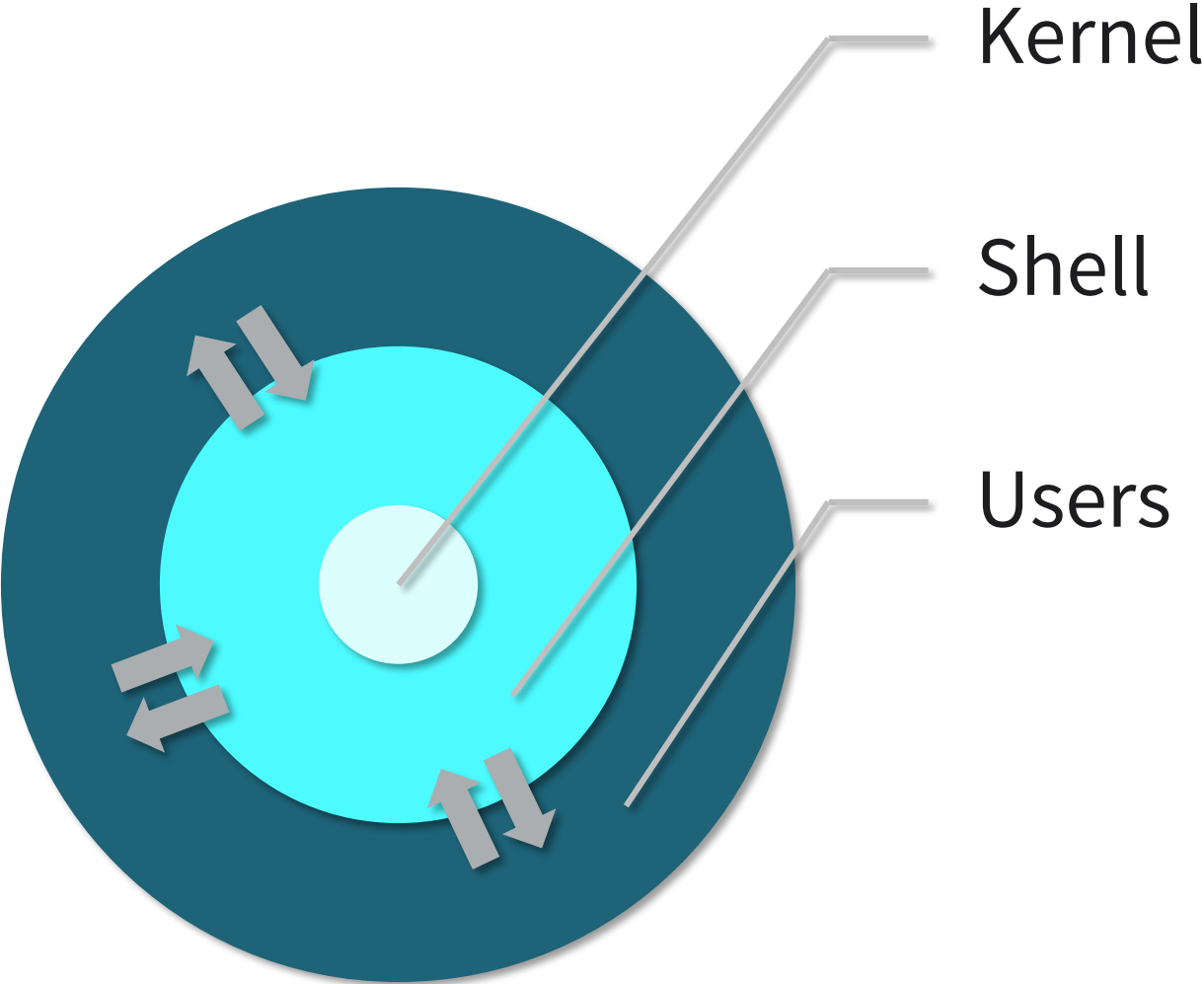
If a capable operating system is located, BIOS will pass control to it.



FUNCTIONS OF OPERATING SYSTEMS



THE SHELL AS AN INTERFACE BETWEEN USERS AND THE OPERATING SYSTEM

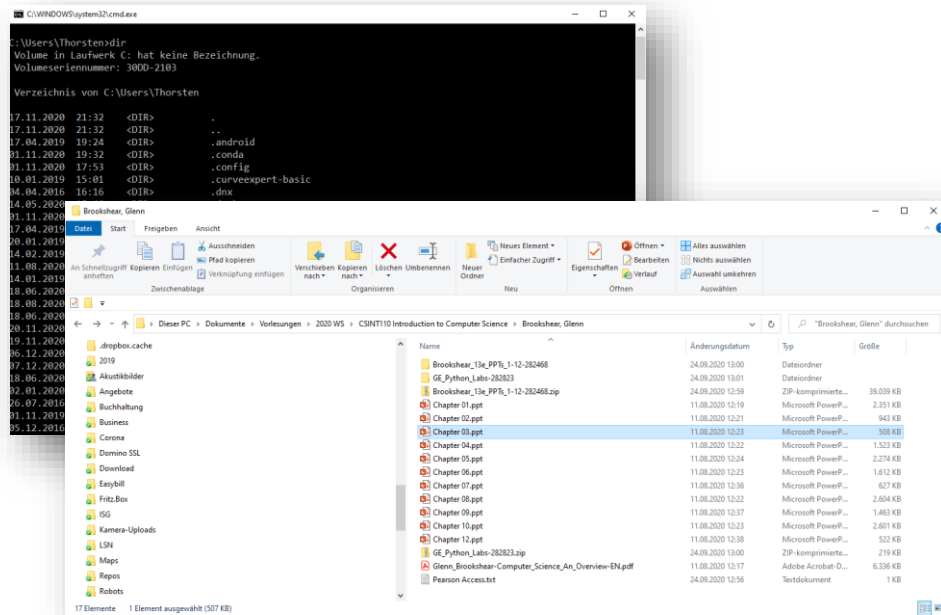


OPERATING SYSTEM COMPONENTS

Shell

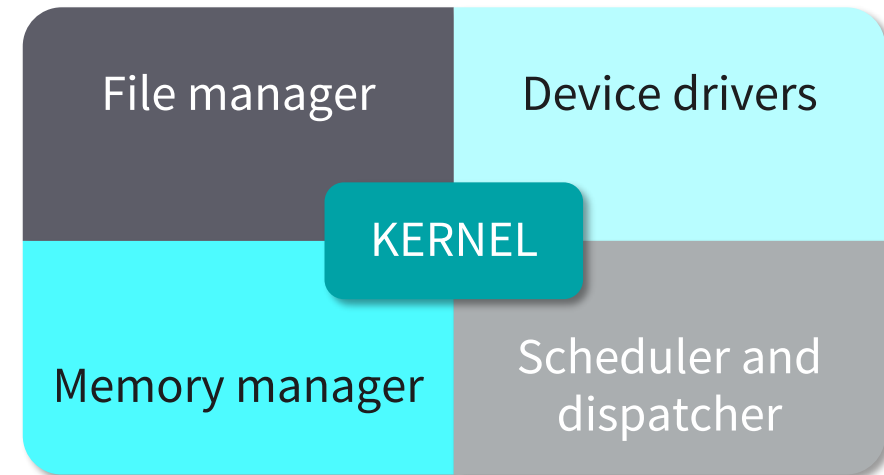
Communicates with users

- text-based
- Graphical User Interface (GUI)

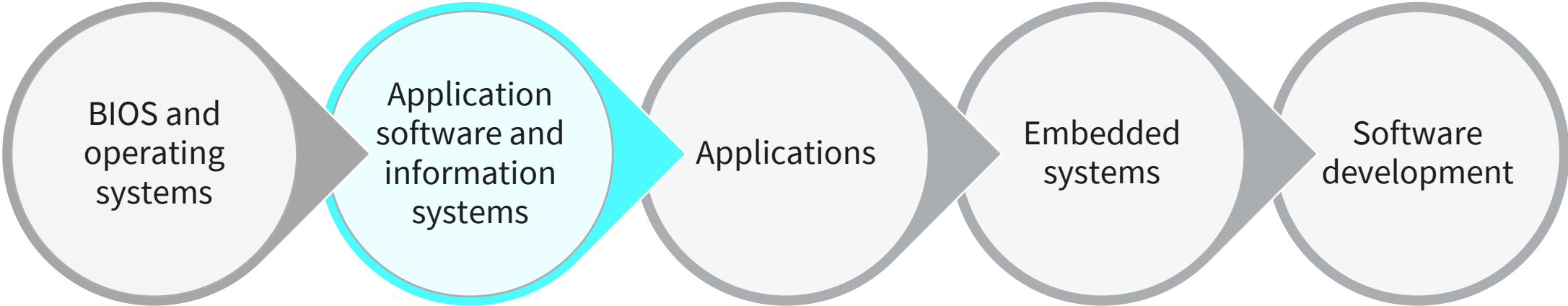


Kernel

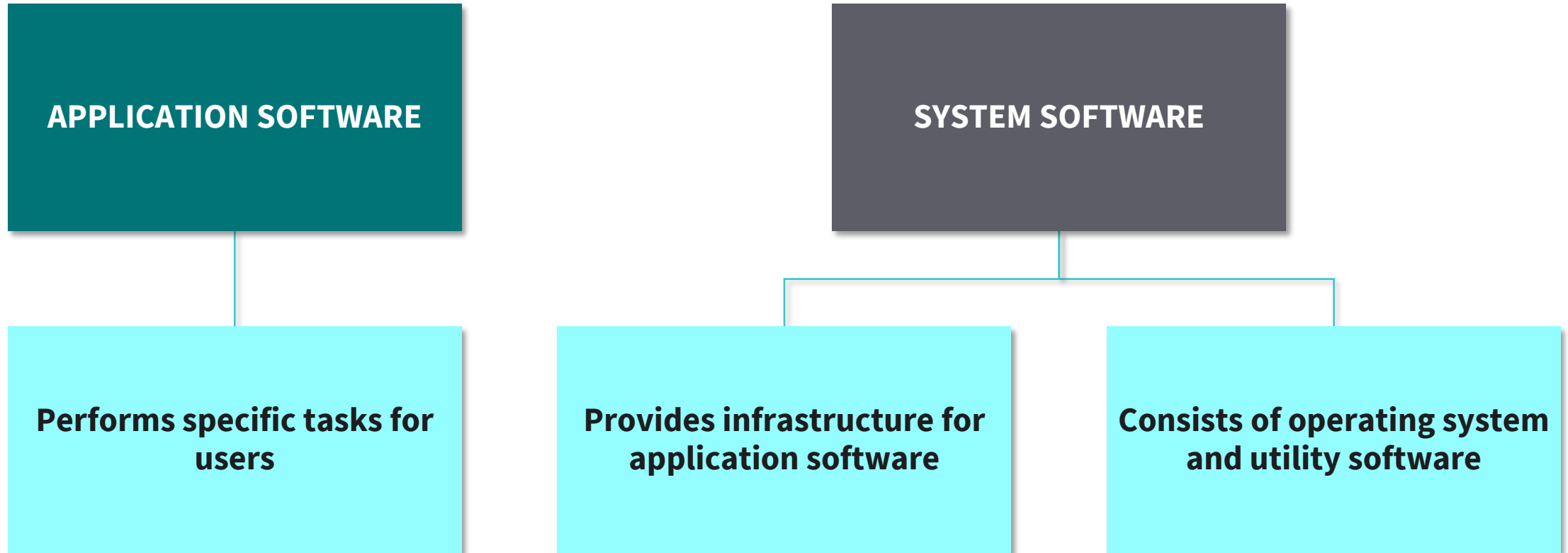
Performs basic required functions



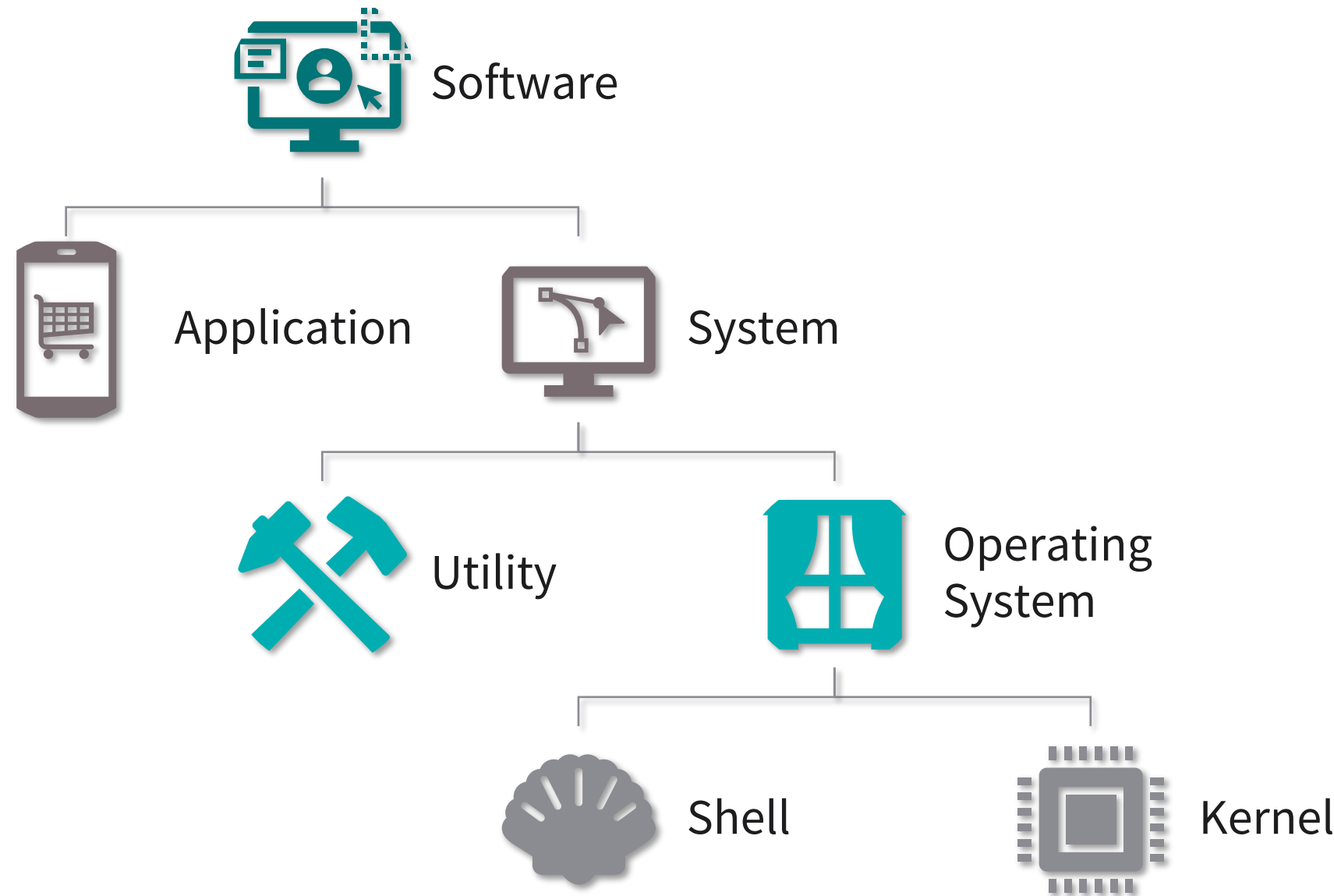
SOFTWARE



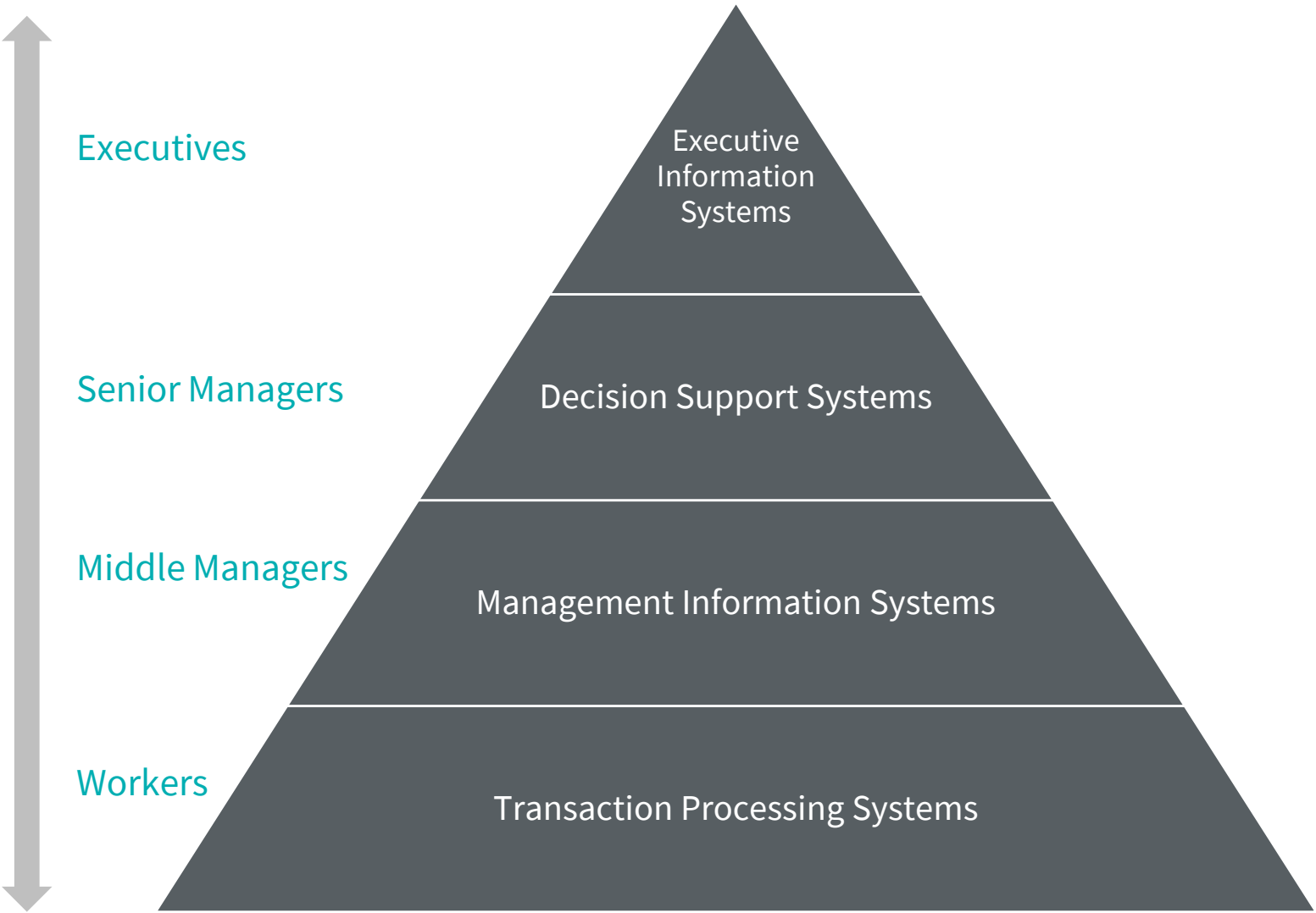
TYPES OF SOFTWARE



SOFTWARE CLASSIFICATION

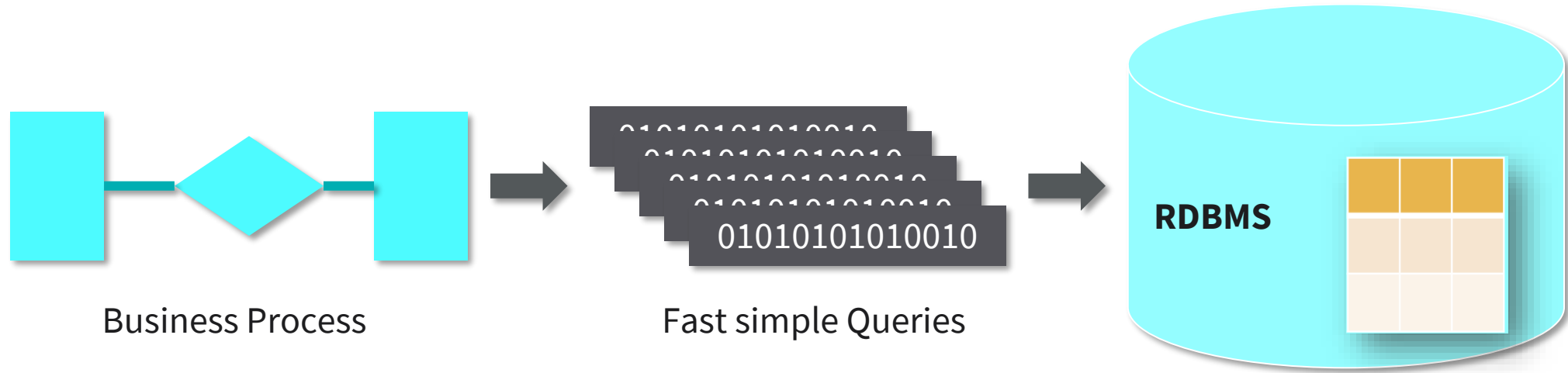


INFORMATION SYSTEM LEVELS



Source of the graphic: Kimble, n.d.

ONLINE TRANSACTION PROCESSING (OLTP)



OLTP systems perform simple database operations to provide sub-second response times.

The queries supported by OLTP systems are comprised of simple insert, delete and update operations with sub-second response times.

Examples include ticket reservation systems, banking and point of sale systems.

ONLINE ANALYTICAL PROCESSING (OLAP)

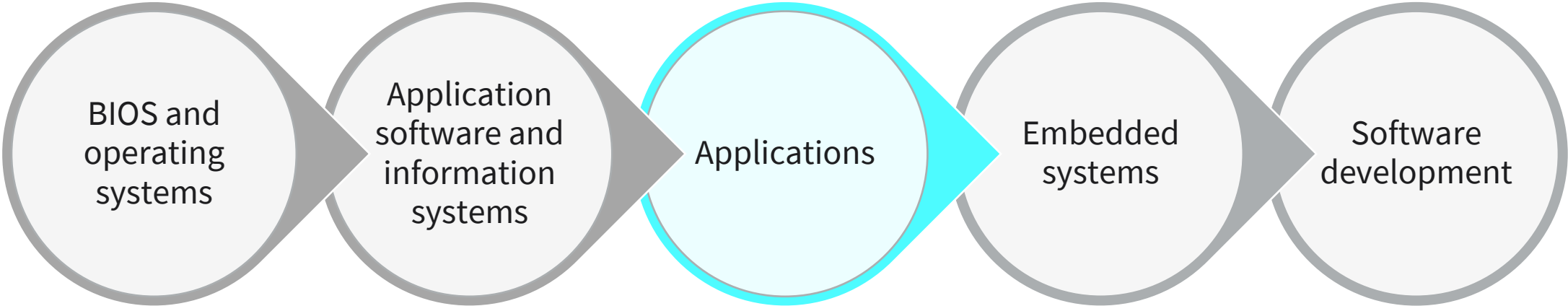


OLAP systems use **multidimensional databases**.

OLAP systems store historical data that is aggregated and denormalized to support fast reporting capability.

They further use databases that store historical data in multidimensional structures and can answer complex queries based on the relationships between multiple aspects of the data.

SOFTWARE



APPLICATION TYPES



Standalone applications

- Application systems that run on a local computer, such as a PC.
- Include all necessary functionality and do not need to be connected to a network.

Interactive transaction-based applications

- Applications that execute on a remote computer and are accessed by users from their own PCs or terminals.
- These include web applications such as e-commerce applications.

Embedded control systems

- Software control systems that control and manage hardware devices.
- Numerically, there are probably more embedded systems than any other type of system.

Batch processing systems

- Business systems that are designed to process data in large batches.
- They process large numbers of individual inputs to create corresponding outputs.



Entertainment systems

- Systems that are primarily for personal use, and which are intended to entertain the user.

Systems for modeling and simulation

- Systems that are developed by scientists and engineers to model physical processes or situations, which include many, separate, interacting objects.

Data collection systems

- Systems that collect data from their environment using a set of sensors and send that data to other systems for processing.

Systems of systems

- These are systems that are composed of a number of other software systems.



Freedom to run, copy, distribute, study, change and improve the software.

- freedom to run the program, for any purpose
- freedom to study how the program works and adapt it to your needs
- freedom to redistribute copies so you can help others
- freedom to improve the program and release your improvements to the public

Access to the source code is needed to realise most of the above.



The term **open source** refers to something people can modify and share because its design is publicly accessible.

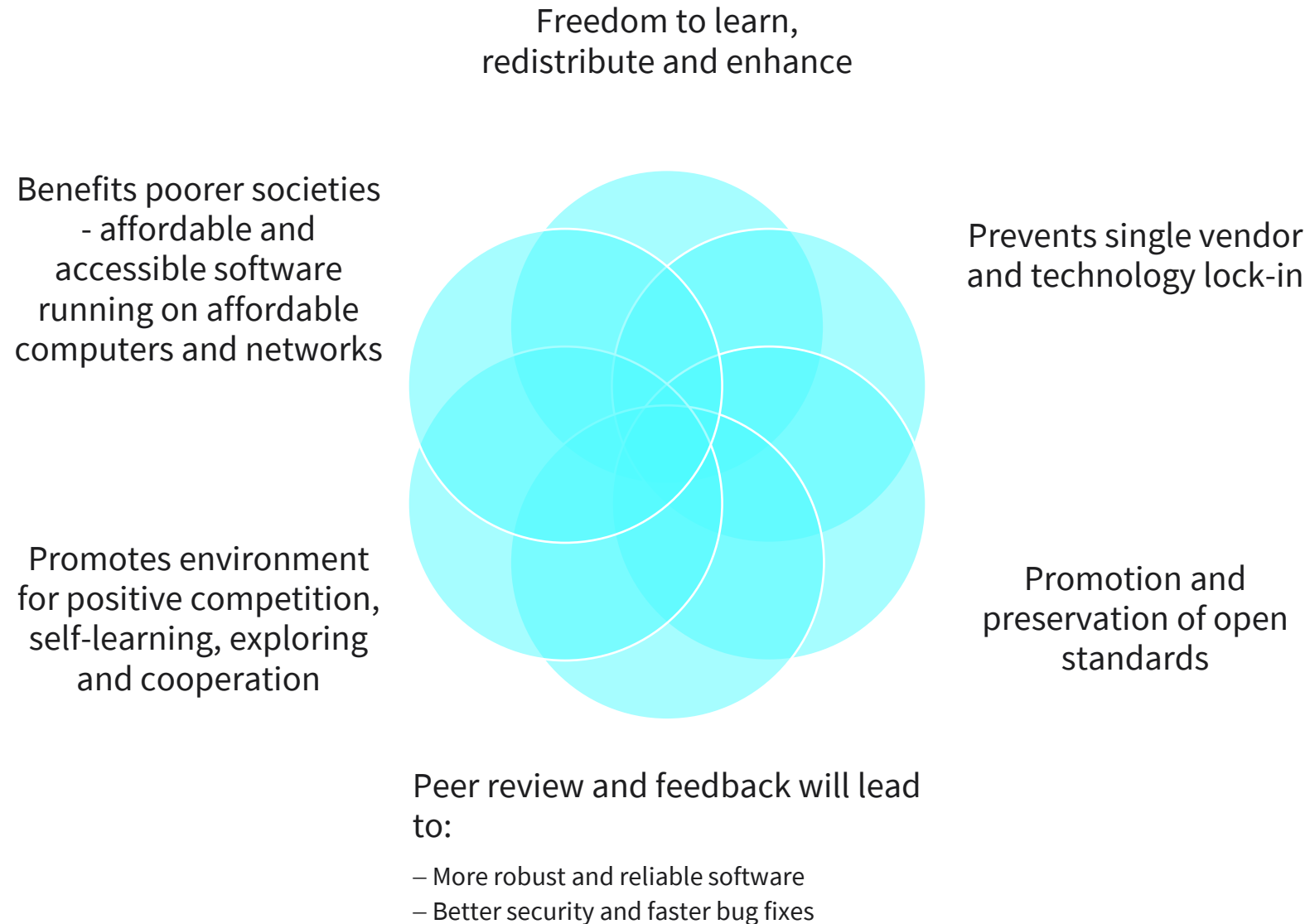
The term originated in the context of software development to designate a specific approach to creating computer programs.

The term “free” can lead to negative marketing connotations especially to business and corporate users.

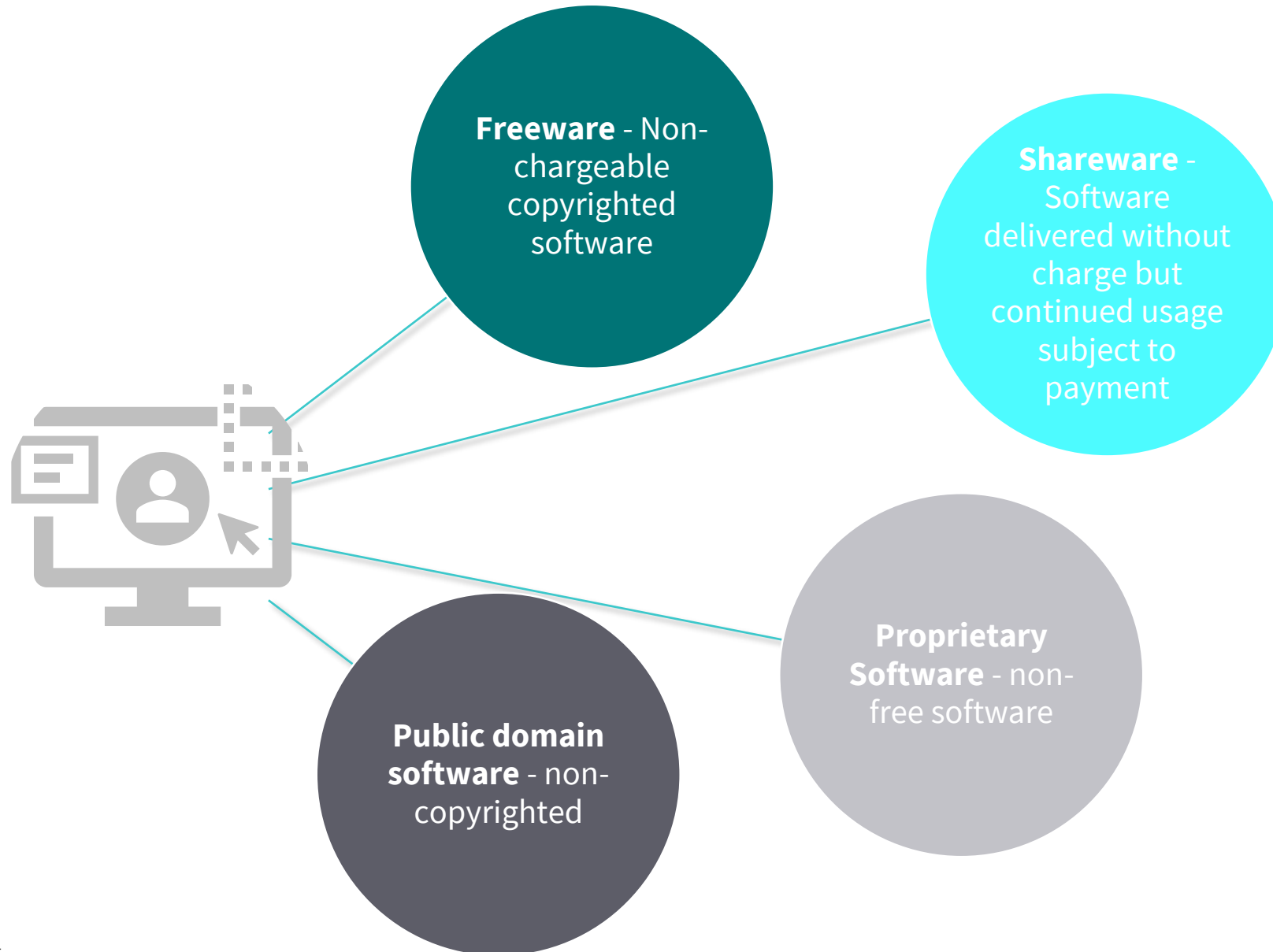
How can anything be good if it is free (i.e. has no monetary value to it)?

The term open-source software was coined to refer to free software.

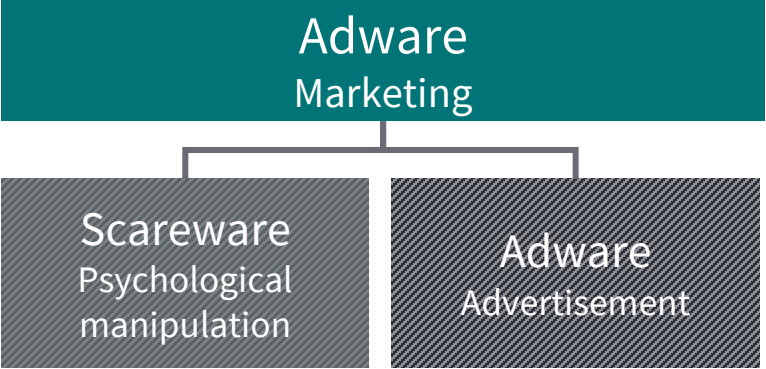
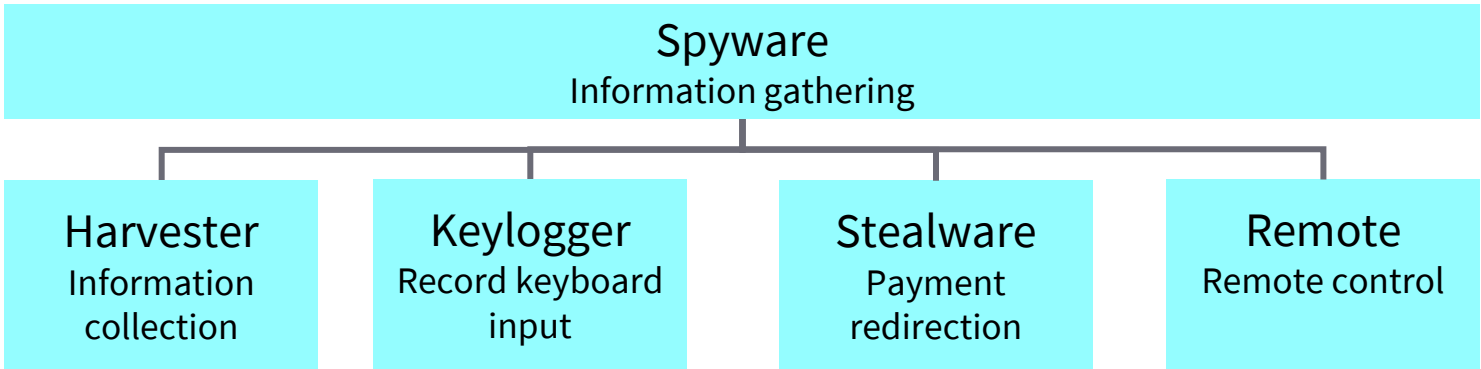
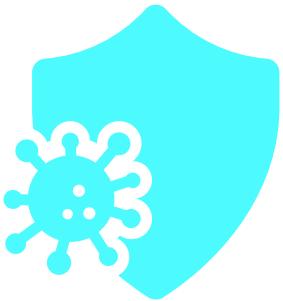
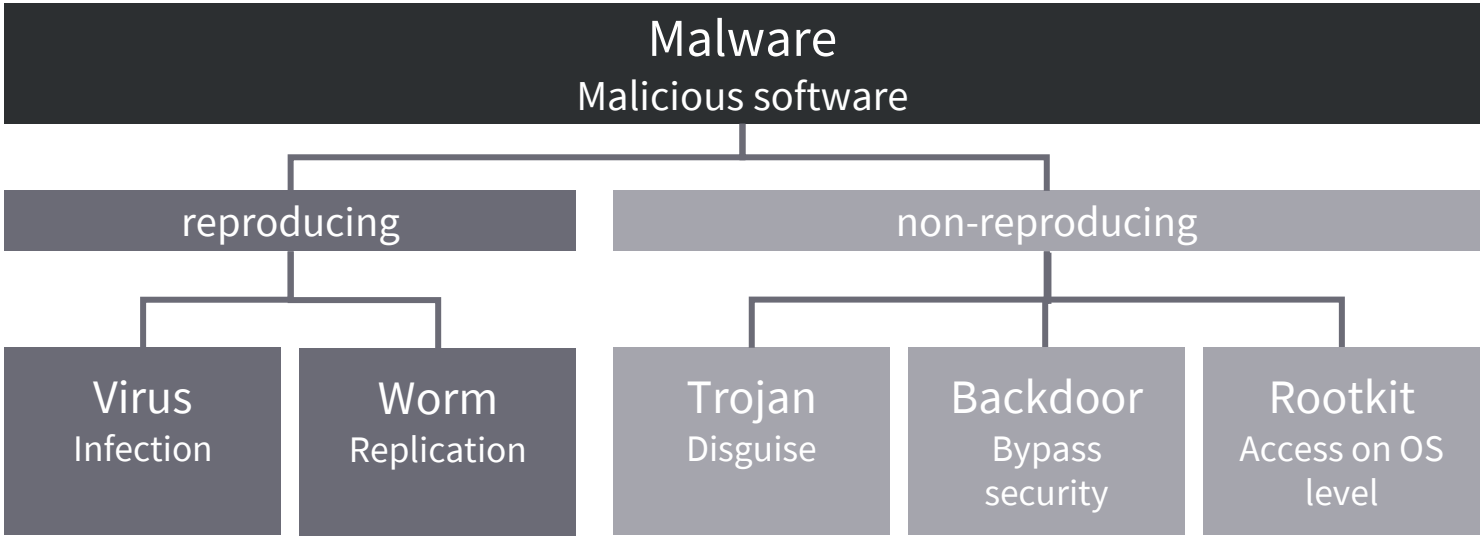
IMPORTANCE OF OPEN SOURCE



OTHER CATEGORIES OF SOFTWARE



MALWARE – SPYWARE – ADWARE



Bacteriophage

RNA

- Contains data for replication
- Suppresses simultaneous replication
- Ensures dominance against DNA of host cell

Mutation ensures adaptation of RNA

Cytopathic Effect destroys functional part of host cell

Cell lysis destroys cell membrane

Long tail fiber

Ensures adsorption

Spikes react on cell membranes only

Computer virus

Replicator: contains commands for distribution

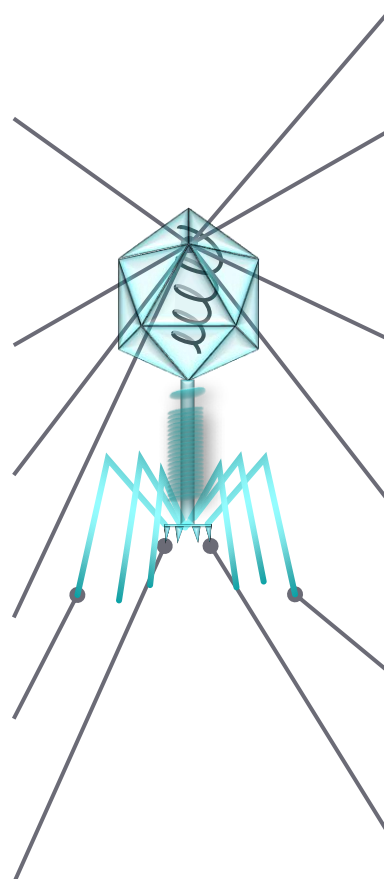
Identifier: checks, whether the file has been infected already

Concealer: conceals the virus against anti virus software

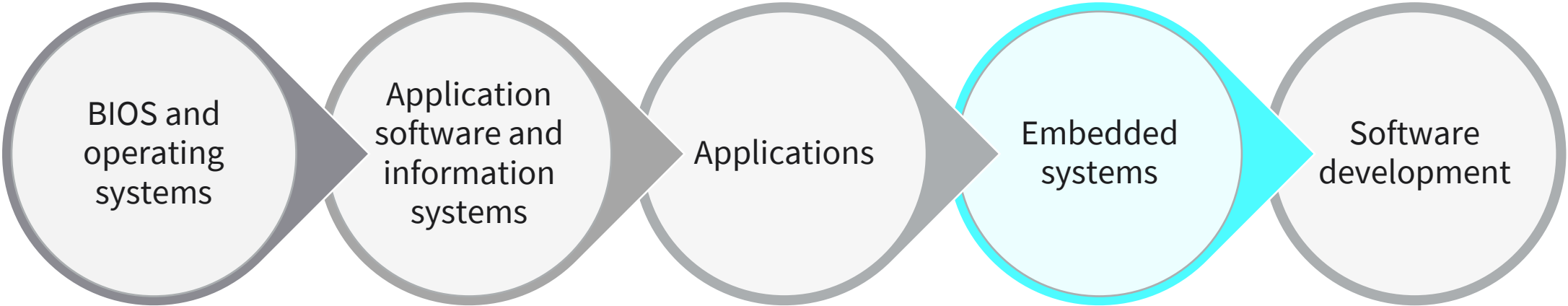
Destructor (Payload): does damage

Activator: activates the virus

Conditioner: ensures that activation takes place only under certain conditions



SOFTWARE



WHAT'S AN EMBEDDED SYSTEM?

Embedded systems

information processing systems embedded into a larger product

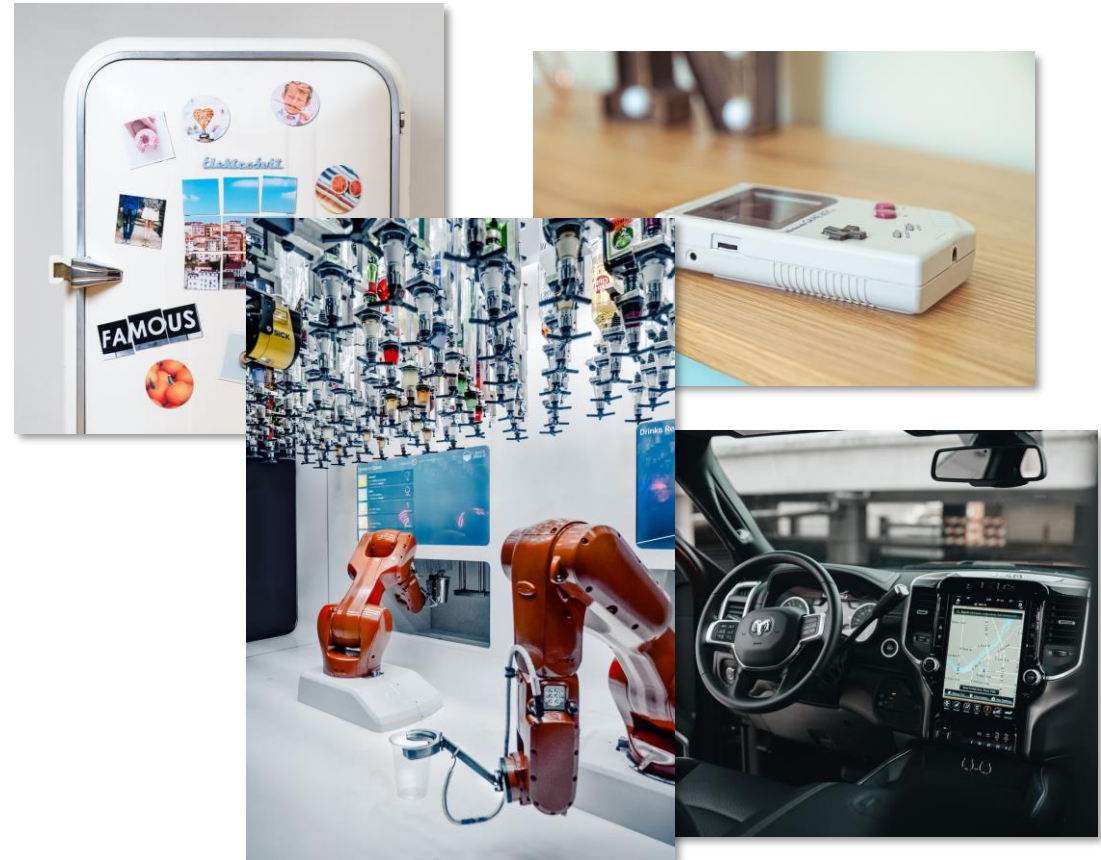
Two Types of Computing

Desktop – produced millions/year

Embedded – billions/year

Non-Embedded Systems

PCs, servers, and notebooks



COMPONENTS OF EMBEDDED SYSTEMS

Analog Components

- sensors, actuators, controllers, ...

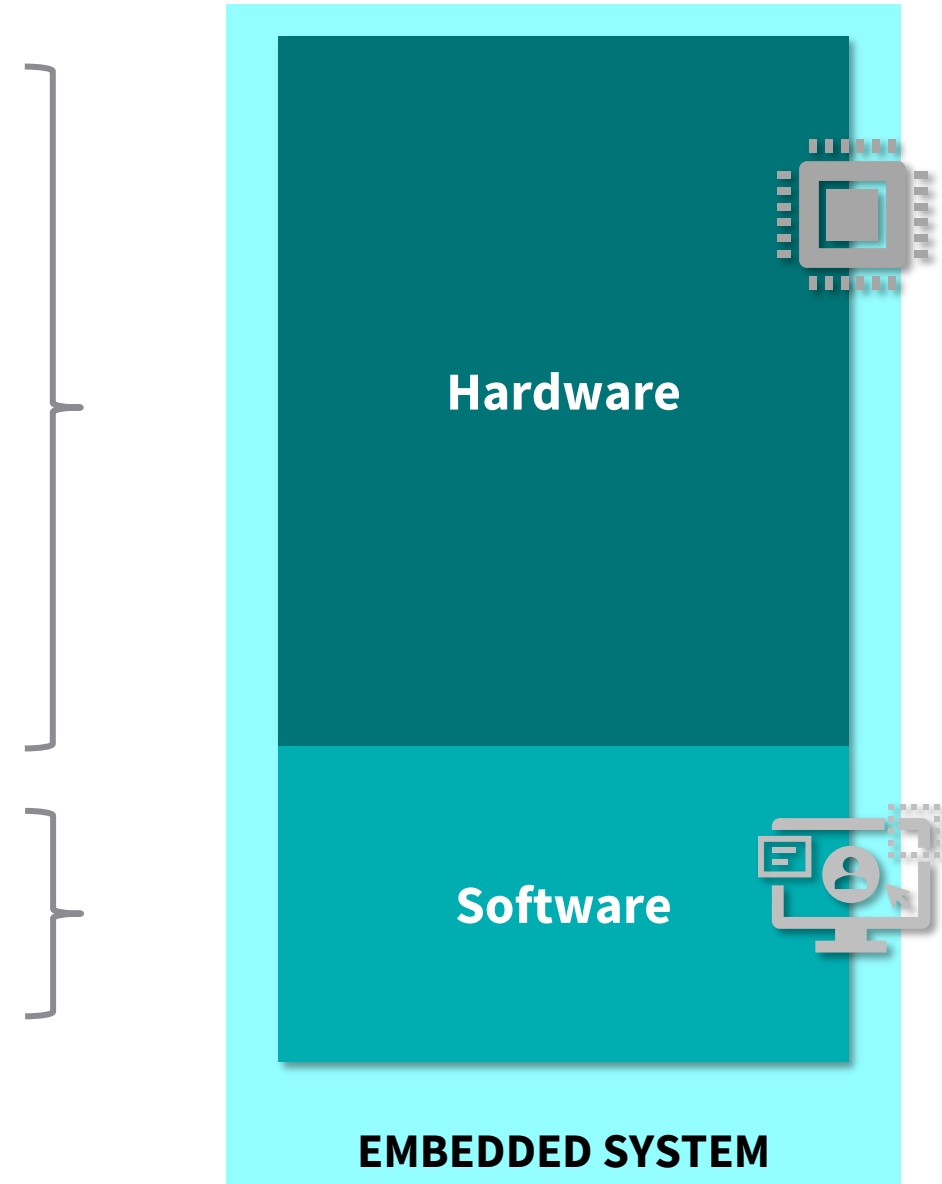
Digital Components

- processor, coprocessors
- memories
- controllers, buses
- Application Specific Integrated Circuits (ASIC)

Converters – A2D, D2A, ...

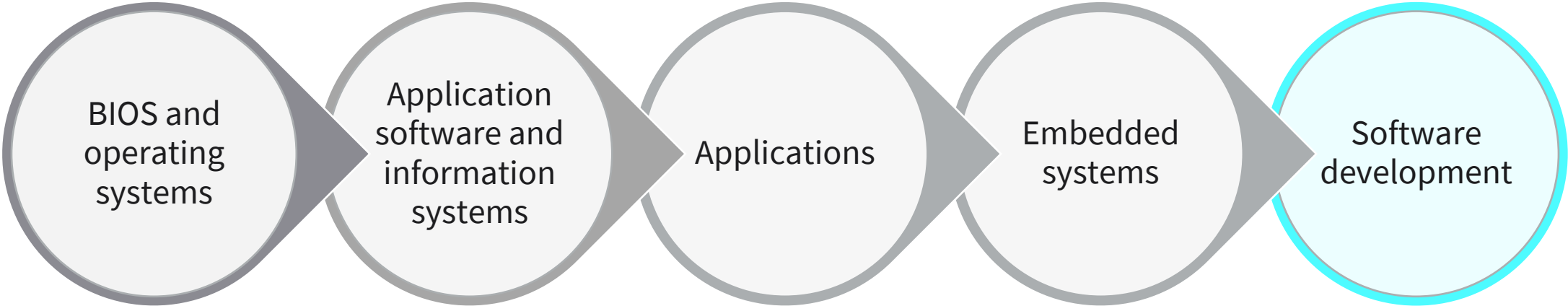
Software

- application programs
- exception handlers





SOFTWARE



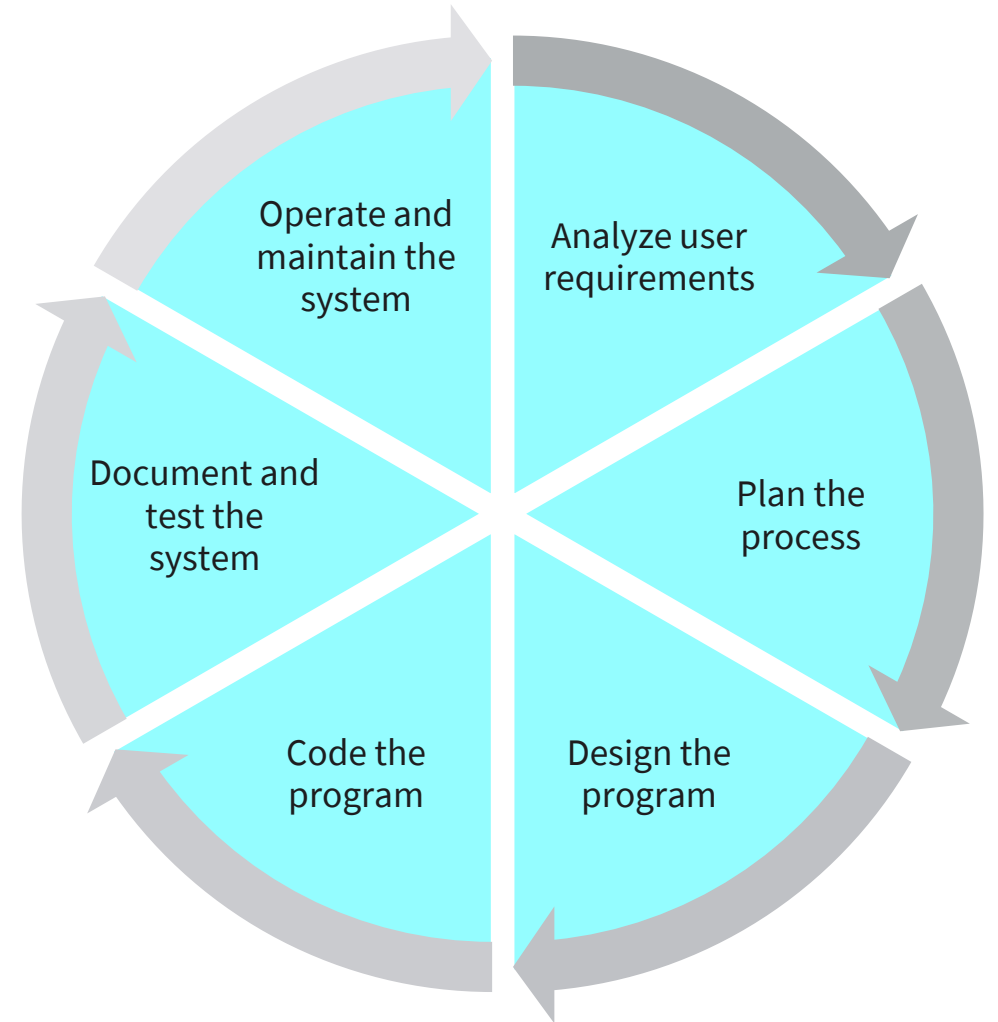
SOFTWARE DEVELOPMENT LIFE CYCLE

What is the Software Development Life Cycle (SDLC)?

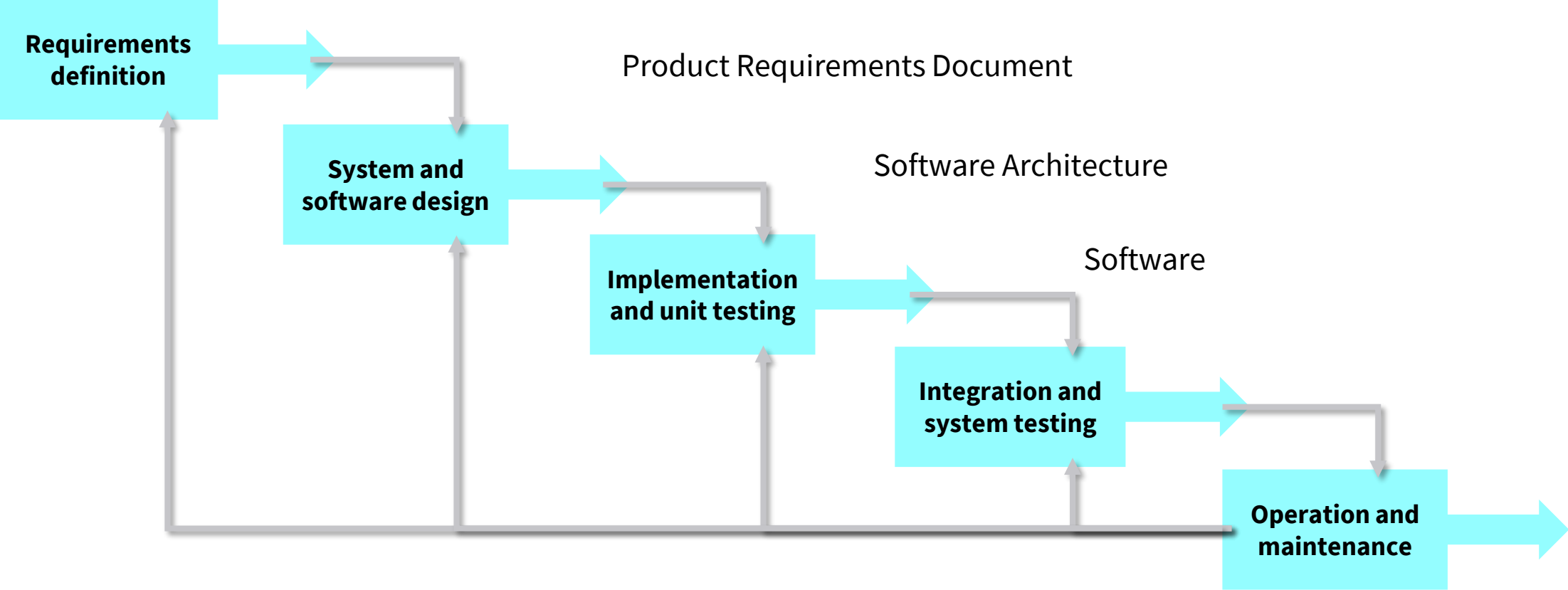
The Software Development Life Cycle is a process that ensures good software is built.

Each phase in the life cycle has its own process and deliverables that feed into the next phase.

There are typically 5 to 6 phases starting with the analysis and requirements gathering and ending with the implementation.



THE WATERFALL MODEL



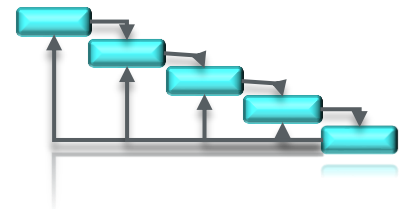
WATERFALL MODEL PHASES

There are separate identified phases in the waterfall model:

- Requirements analysis and definition
- System and software design
- Implementation and unit testing
- Integration and system testing
- Operation and maintenance

The main drawback of the waterfall model is the difficulty of accommodating change after the process is underway.

In principle, a phase has to be complete before moving on to the next phase.



WATERFALL MODEL PROBLEMS

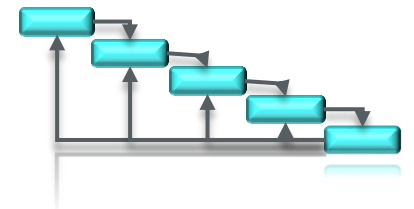
Inflexible partitioning of the project into distinct stages makes it difficult to respond to changing customer requirements.

Therefore, this model is only appropriate when the requirements are well-understood and changes will be fairly limited during the design process.

Few business systems have stable requirements.

The waterfall model is mostly used for large systems engineering projects where a system is developed at several sites.

In those circumstances, the plan-driven nature of the waterfall model helps coordinate the work.



VARIOUS SDLC MODELS

Agile Model

Big Bang
Model

Conceptual
Model

Extreme
Programming

Iterative Model

Kaizen Model

Kanban Model

Rapid
Application
Development

Rational
Unified
Process

Scaled Agile
Framework

Scrum

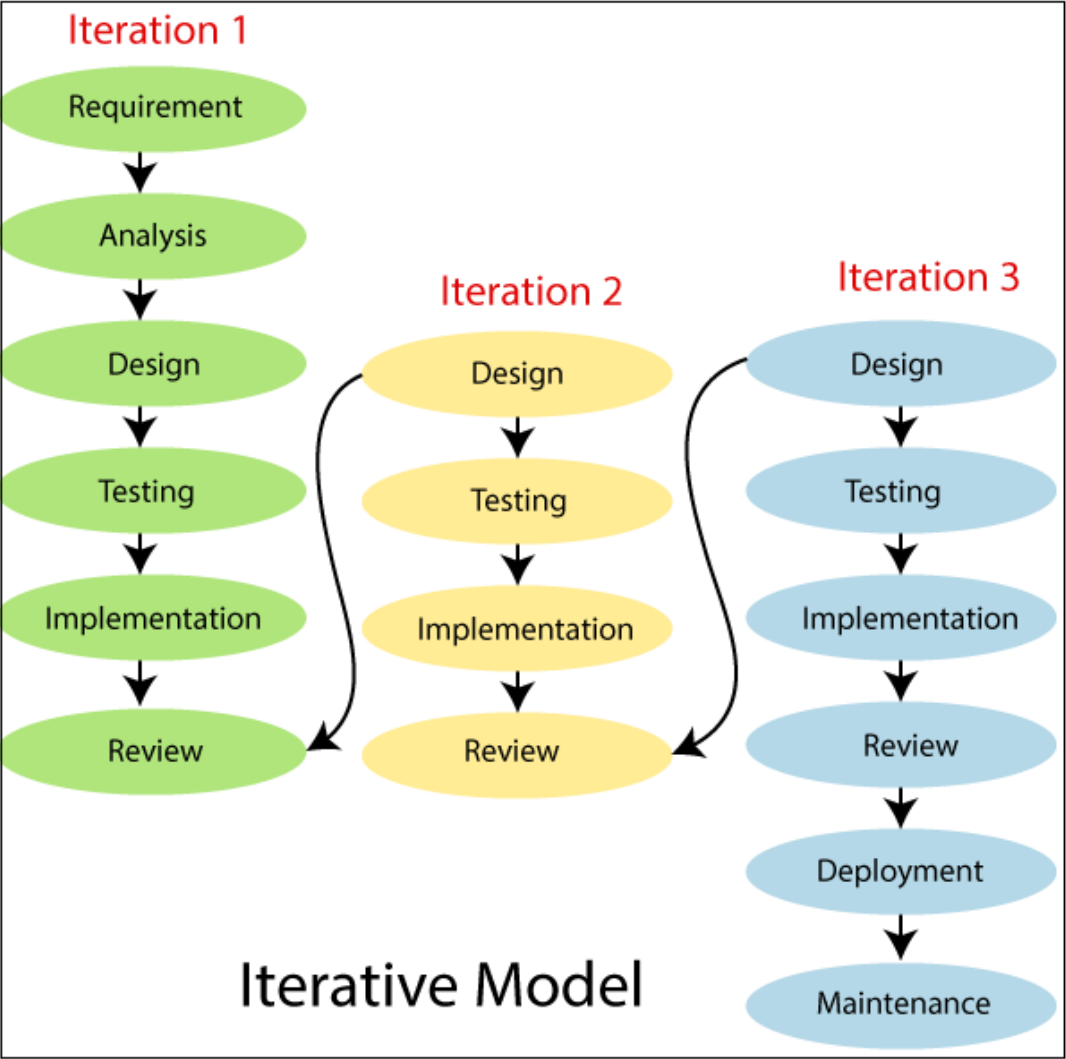
Spiral Model

Test Driven
Development

V Model

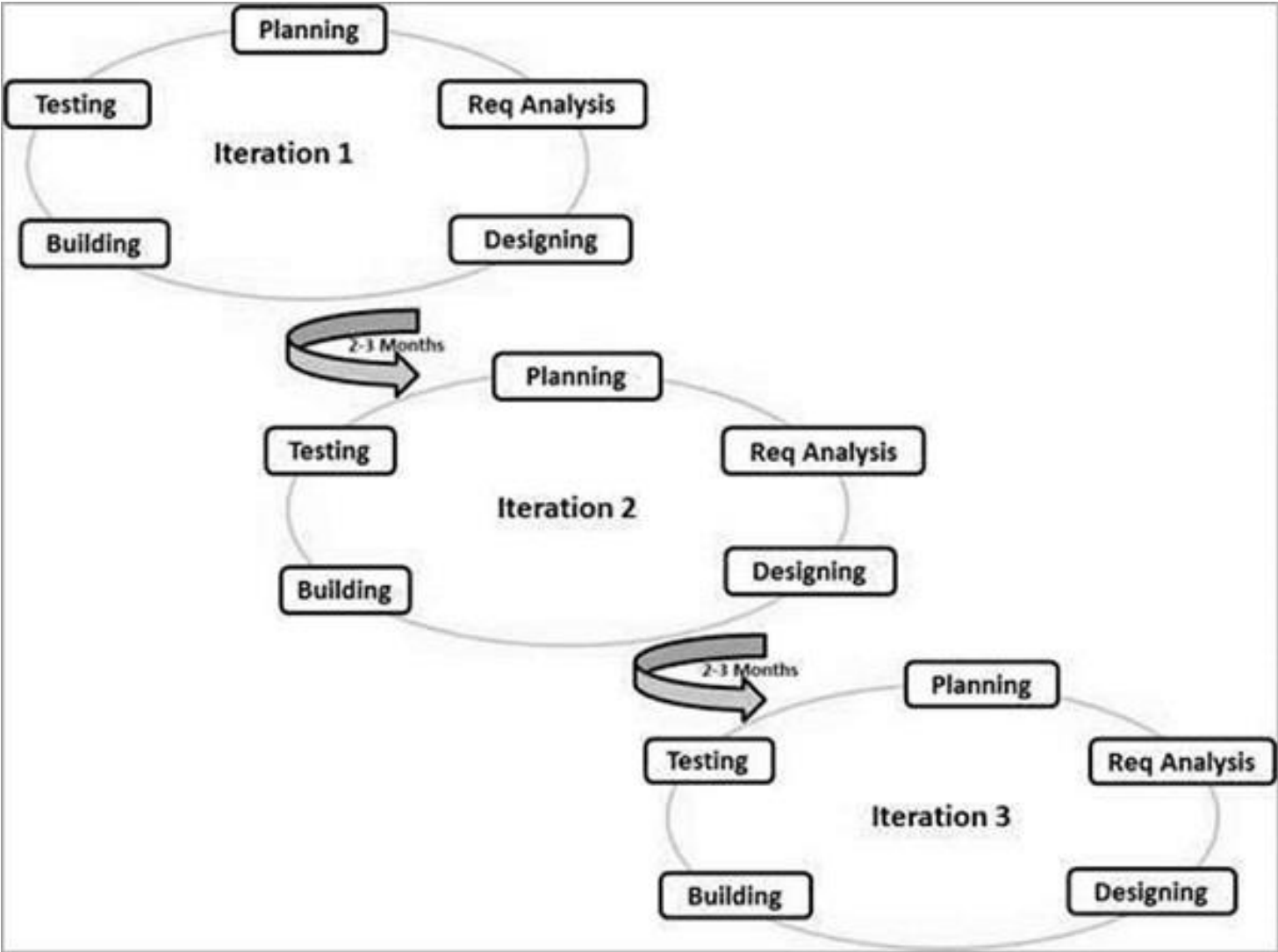
Waterfall
Model

ITERATIVE MODEL

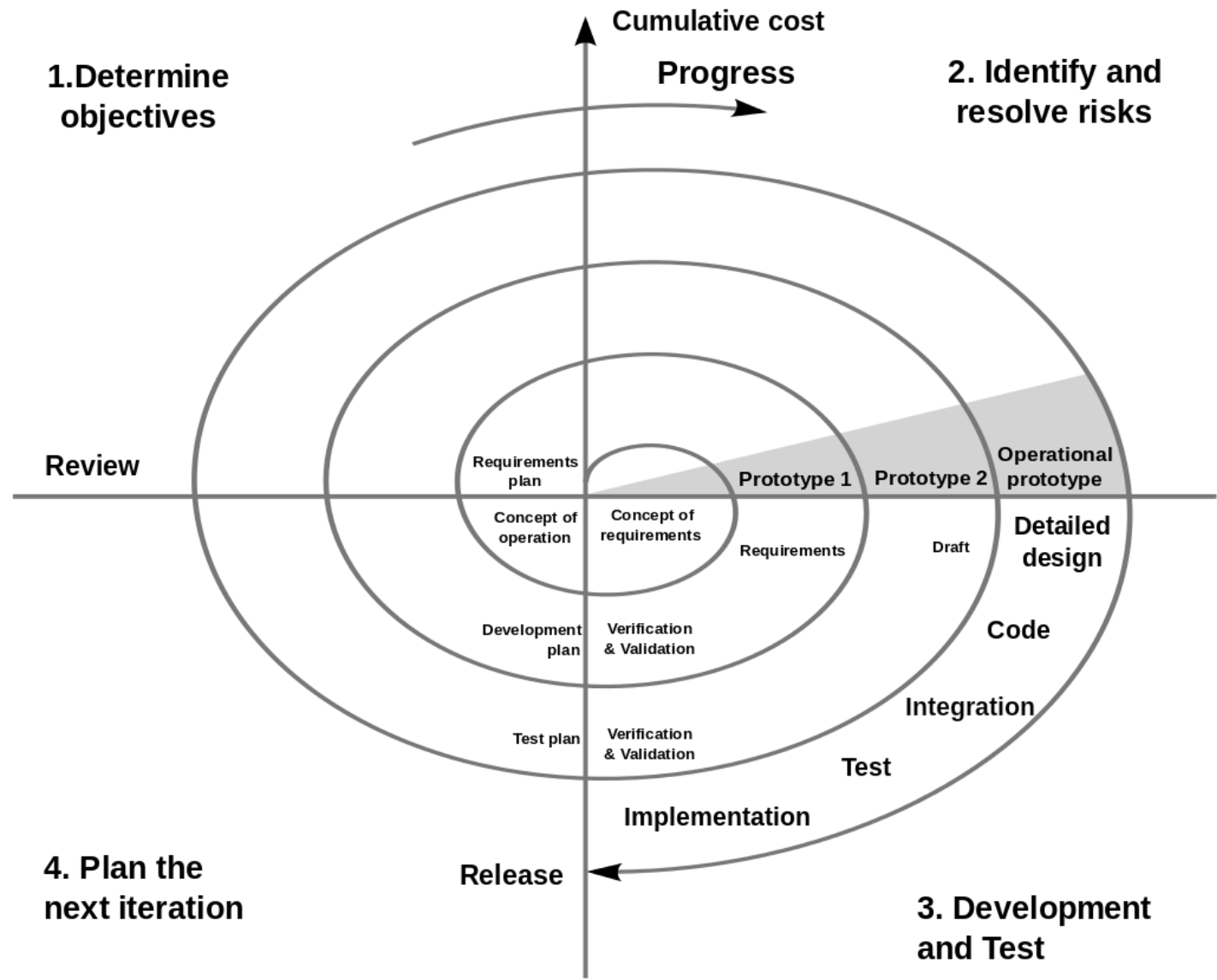


Srouce of image: <https://laptrinhx.com/iterative-model-in-software-engineering-1408694444/>

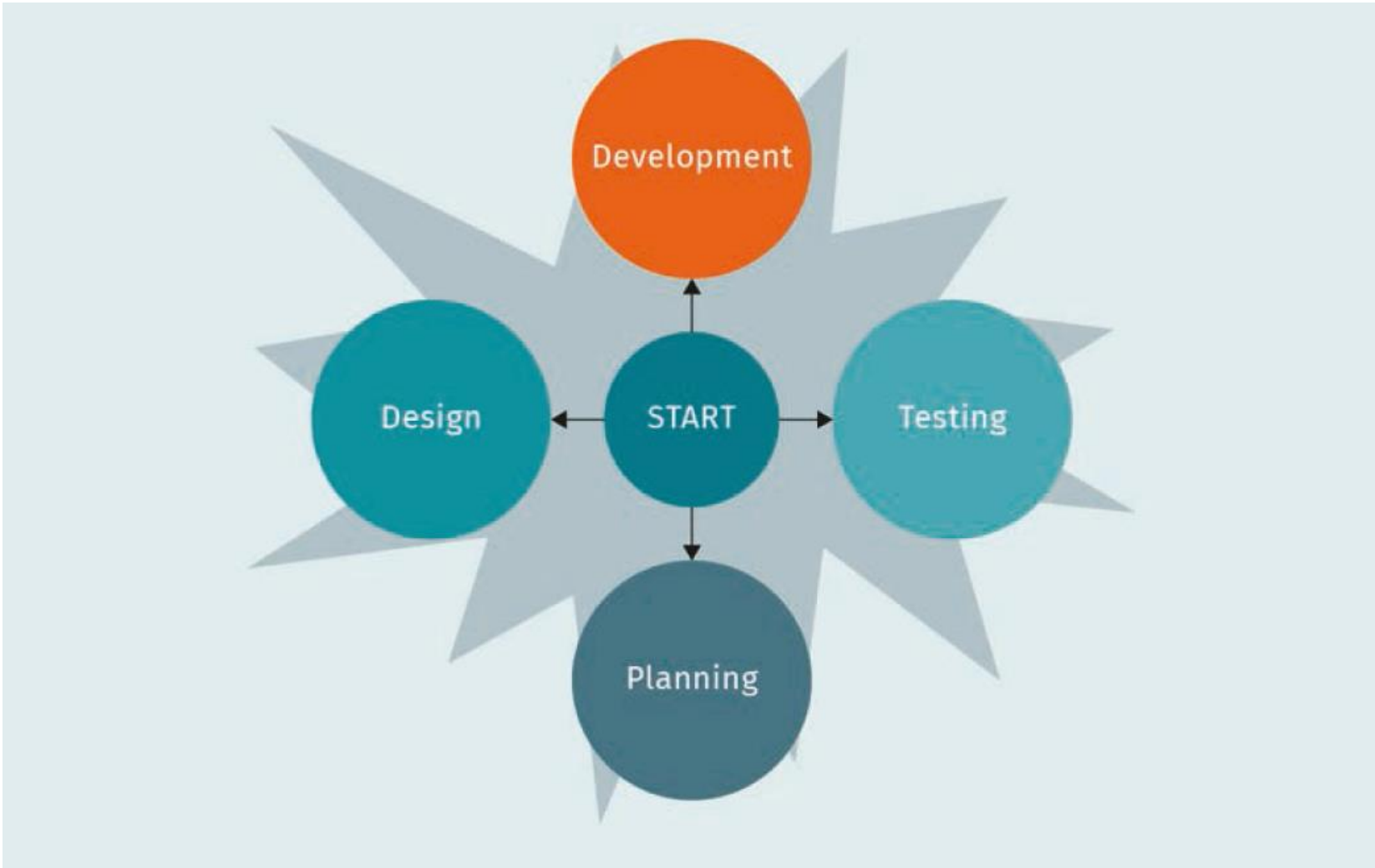
AGILE MODEL



SPIRAL MODEL



BIG BANG MODEL





You have learned ...

- ... how BIOS works.
- ... about modern file systems.
- ... about application software for operating systems.
- ... about the concept of embedded systems.
- ... about the process of software development.
- ... what the functions of an operating system are.

SESSION 5

TRANSFER TASK



Types of Mobile Apps

There are 4 different types of mobile Apps.

- 1. Native apps**
- 2. Web apps**
- 3. Hybrid apps**
- 4. Progressive Web apps**

Select one type and create a presentation explaining the following:

- Characteristics
- Architecture
- Advantages
- Disadvantages
- Examples

TRANSFER TASK
PRESENTATION OF THE RESULTS

Please present your
results.

The results will be
discussed in
plenary.





1. The startup process of a desktop computer is handled by the?
 - a) CPU
 - b) BIOS
 - c) RAM
 - d) SATA port



2. To talk to installed devices, this software is needed?
- a) driver
 - b) application
 - c) hard disk
 - d) motherboard



3. An example of an embedded system would be?
- a) mainframe
 - b) desktop
 - c) fire alarm
 - d) web server



4. When you download an app, it will only:
- a) work for one user
 - b) install on your C: drive
 - c) reset the BIOS
 - d) run on one OS



5. In the Waterfall software lifecycle model, what is the first step?
- a) design
 - b) requirements
 - c) develop
 - d) testing

LIST OF SOURCES

Brookshear, G. & Bylow, D. (2011). *Computer Science: An Overview* (11th ed.). Pearson.

Dale, N. & Lewis, J. (2020). *Computer Science Illuminated* (7th ed.). Jones & Bartlett Learning.

Downey, A. B. & Mayfield, C. (2020). *Think Java: How to Think Like a Computer Scientist*. O'Reilly.

Filho, W. F. (2018). *Computer Science Distilled: Learn the Art of Solving Computational Problems*. Code Energy LLC.

Kimble, C. (n.d.). *Information systems and strategy, session 1: Types of information system and the classic pyramid model*. Chris-Kimble.com. <http://www.chris-kimble.com/Courses/index.html>

Petzold, C. (2000). *Code: The Hidden Language of Computer Hardware and Software*. Microsoft Press.

Sommerville, I. (2015). *Software Engineering* (10th ed.). Pearson.

Weizenbaum, J. (1966). *ELIZA—a computer program for the study of natural language communication between man and machine*. Communications of the ACM, 1966. dl.acm.org

Whittington, J. (2016). *A Machine Made this Book: Ten Sketches of Computer Science*. Coherent Press.