
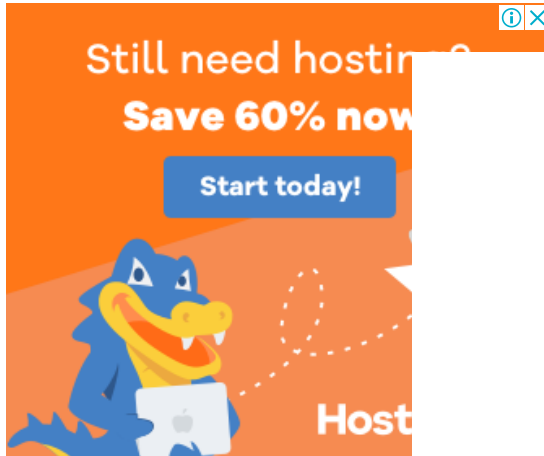
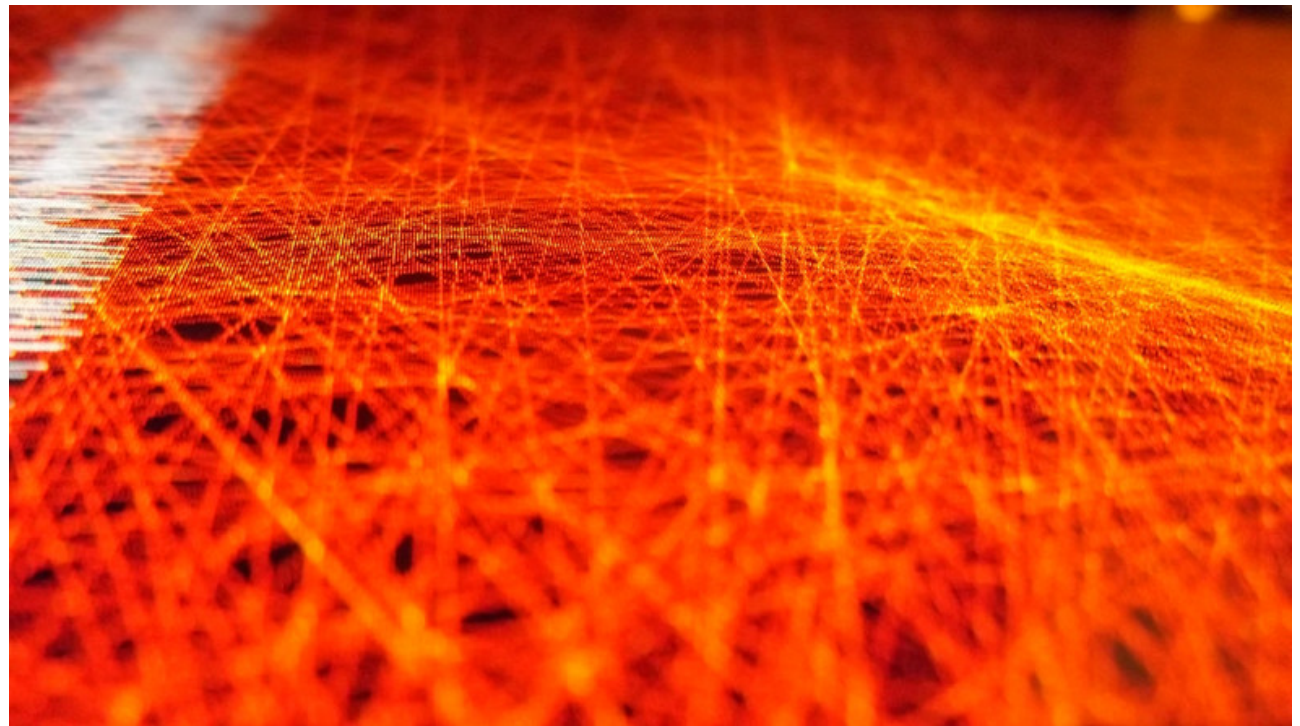


Simplicable

[A-Z](#) [Popular](#) [Coding](#) [Search »](#) Simplicable[A-Z](#) [Popular](#) [Coding](#) [Search »](#)[Coding](#)[top](#) » [design](#) » [software design](#) » [separation of concerns](#)

What is Separation Of Concerns?

posted by [John Spacey](#), December 27, 2016

Separation of concerns, or **SoC**, is a principle of [software design](#) that code be separated into layers and components that each have distinct functionality with as little overlap as possible. It is a fundamental principle that is widely observed.

Value

Separation of concerns reduces complex problems into a series of manageable layers and components. It tends to reduce risks as changes are often isolated to a single component as opposed to intermingled throughout a large and complex code base.

دهانات الجزيرة
Jazeera Paints

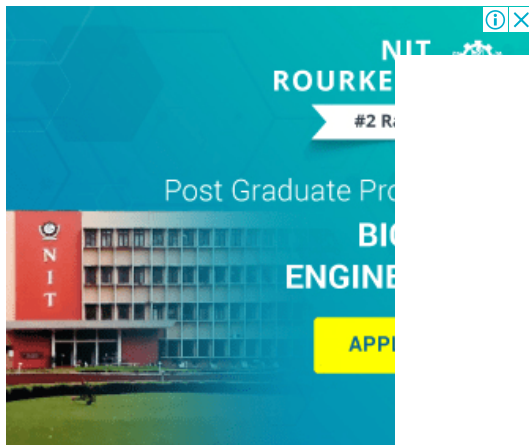


نوفل ..
ك أثر



Simpllicable Guide

[A-Z](#)
[Popular](#)
[Business](#)
[Design](#)
[Tech](#)
[More ...](#)



Related Guides

[Analytics](#)
[Automation](#)
[Coding](#)
[Controls](#)
[Data](#)
[Ethics](#)

Encapsulation

Separation of concerns is implemented by encapsulating functionality in components that offer a well-defined interface. Components [hide complexity](#) such as user interfaces, [business logic](#), data access and transaction execution from the rest of the code. When something changes, the interface often isn't impacted meaning that the change is isolated to a component.

Overview: Separation Of Concerns

| | |
|------------------|--|
| Type | Software Design |
| Definition | The principle that code be structured into layers and components that each have distinct functionality with as little overlap as possible. |
| Value | Reduces complexity Makes change more manageable |
| Also Known As | SoC |
| Related Concepts | Software Design Principle Of Least Astonishment Coding |

[Cite »](#)

Coding

| | | |
|---------------------------------------|-----------------------------------|----------------------------------|
| Abstraction | Algorithms | API |
| Bootstrapping | Caching | Code Refactoring |
| Code Smell | Complexity Hiding | Components |
| Deep Magic | Edge Case | Event Processing |
| Forward Compatibility | Hardcoded | Layers |
| Microservices | Negative Code | Precomputation |
| Proof Of Work | Pseudorandom | Reusability |
| Scalability | Software Design | More ... |

Coding

A list of coding considerations and techniques.

[30+ Coding Terms »](#)

[10+ Types of Algorithm »](#)

[8 Types of Code Reuse »](#)

[20+ Coding Basics »](#)



[Coding](#)

[Algorithms](#)

[Architectural Patterns](#)

[Artificial Intelligence](#)

دهانات الجزيرة
Jazeera Paints

نوفل ..
بات فائق للألوان



Emergence vs Big Design Up Front

The difference between emergence and big-design-up-front.



[Emergence vs Big Design Up Front »](#)



Deep Magic

An overview of deep magic, a technology term.



What is Deep Magic? »

Principle Of Least Astonishment

An overview of the Principle Of Least Astonishment.

Design: Principle Of Least Astonishment »



[Pull vs Push](#)

The difference between pull and push technology.



[Pull vs Push Technology](#) »

[Binary vs Hexadecimal](#)

A comparison of binary and hexadecimal.

[Binary vs Hexadecimal](#) »



End-User Computing

An overview of end-user computing.



[10 Types of End-User Computing »](#)

Library vs API

The difference between a library and API explained.

[Library vs API »](#)



Code Reuse

The common types of code reuse.



[8 Types of Code Reuse »](#)

Code Freeze

The common types of code freeze.

[3 Types of Code Freeze »](#)



Algorithms

A few types of algorithms.



[10+ Types of Algorithm »](#)

[What is a Brute Force Algorithm? »](#)

[What is Edit Distance? »](#)

[What is Forward Chaining? »](#)

Algorithms vs Code

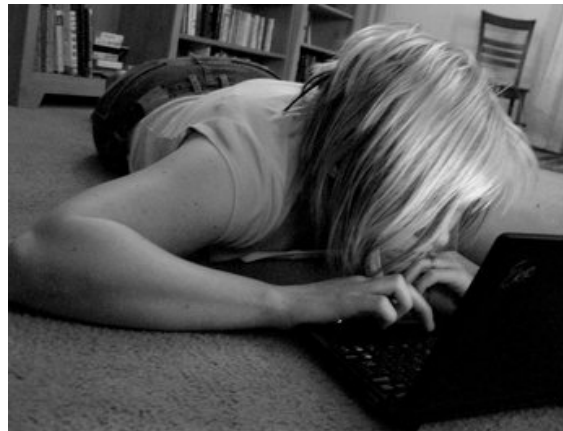
The difference between algorithms and code.

[Algorithms vs Code »](#)



Edit Distance

An overview of edit distance.

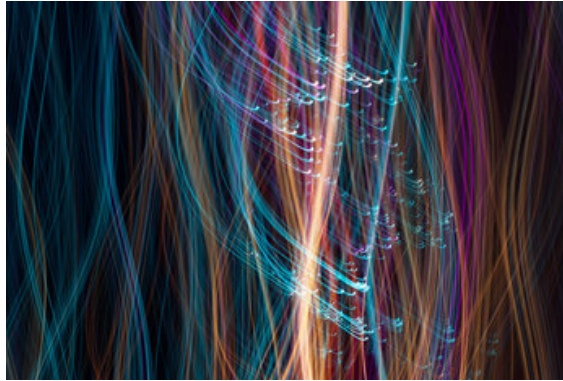


[What is Edit Distance? »](#)

Random Seed

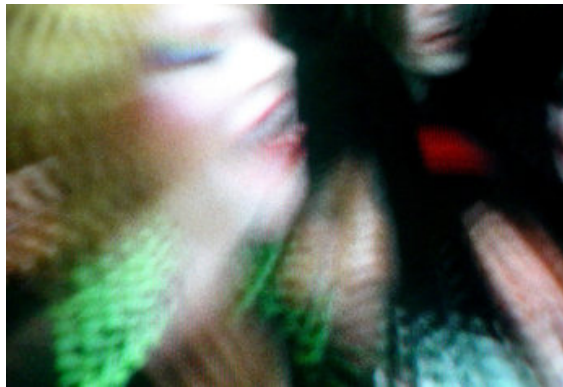
The definition of random seed with examples.

[7 Examples of a Random Seed »](#)



[Soft Computing](#)

The definition of soft computing with examples.



[4 Types of Soft Computing »](#)

[Trending](#)

The most popular articles on Simplicable in the past day.

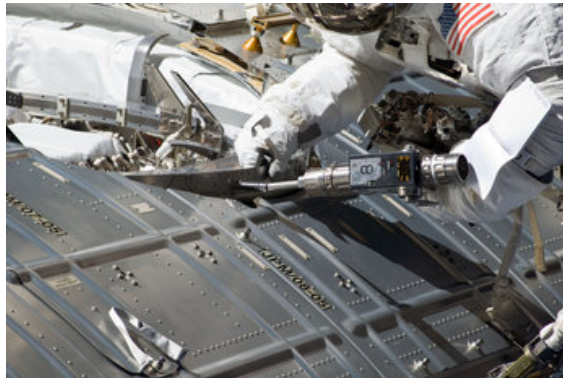
[52 Design Considerations](#)

A list of common design considerations.



[20 Examples Of Assets](#)

A definition of asset with a few examples.



[4 Examples of Problem Analysis](#)

Common problem analysis techniques.

[47 Examples of the Human Experience](#)

The definition of human experience with examples.



New Articles

Recent posts or updates on Simpllicable.



Art vs Design

The difference between art and design.

33 Types of Business Communication

The definition of business communication with examples.



[131 Countries by Income Distribution](#)
A list of countries by income distribution.



[World Income Distribution Rankings](#)
A simple metric of income distribution with data.

[more »](#)

© 2010-2020 Simplicable. All Rights Reserved. Reproduction of materials found on this site, in any form, without explicit permission is prohibited.

View [credits & copyrights](#) or [citation information](#) for this page.

[About](#) [Privacy Policy](#) [Cookies](#) [Terms of Use](#) [Contact Us](#) [Sitemap](#)

Cookies help us deliver our site. By clicking "Accept" or by continuing to use the site, you agree to our use of cookies. Visit our [privacy policy](#), [cookie policy](#) and [consent tool](#) to learn more.

Copyright 2002-2020 Simplicable. All rights reserved. This material may not be published, broadcast, rewritten, redistributed or translated. Report violations [here](#).

PAGE PROTECTED BY **COPYSCAPE** DO NOT COPY