

# Architectural pattern

---

An **architectural pattern** is a general, reusable solution to a commonly occurring problem in software architecture within a given context.<sup>[1]</sup> The architectural patterns address various issues in software engineering, such as computer hardware performance limitations, high availability and minimization of a business risk. Some architectural patterns have been implemented within software frameworks.

The use of the word "pattern" in the software industry was influenced by similar concepts as expressed in traditional architecture, such as Christopher Alexander's *A Pattern Language* (1977) which discussed the practice in terms of establishing a pattern lexicon, prompting the practitioners of computer science to contemplate their own design lexicon.

Usage of this metaphor within the software engineering profession became commonplace after the publication of *Design Patterns* (1994) by Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides—now commonly known as the "Gang of Four"—coincident with the early years of the public Internet, marking the onset of complex software systems "eating the world"<sup>[2]</sup> and the corresponding need to codify the rapidly sprawling world of software development at the deepest possible level, while remaining flexible and adaptive.

Architectural patterns are similar to software design patterns but have a broader scope.

## Contents

---

**Definition**

**Architectural style**

**Examples**

**See also**

**References**

**Bibliography**

## Definition

---

Even though an architectural pattern conveys an image of a system, it is not an architecture. An architectural pattern is a concept that solves and delineates some essential cohesive elements of a software architecture. Countless different architectures may implement the same pattern and share the related characteristics. Patterns are often defined as "strictly described and commonly available".<sup>[3][4]</sup>

## Architectural style

---

Following traditional building architecture, a 'software architectural style' is a specific method of construction, characterized by the features that make it notable.

An architectural style defines: a family of systems in terms of a pattern of structural organization; a vocabulary of components and connectors, with constraints on how they can be combined.<sup>[5]</sup>

An architectural style is a named collection of architectural design decisions that (1) are applicable in a given development context, (2) constrain architectural design decisions that are specific to a particular system within that context, and (3) elicit beneficial qualities in each resulting system.<sup>[1]</sup>

Some treat architectural patterns and architectural styles as the same,<sup>[6]</sup> some treat styles as specializations of patterns. What they have in common is both patterns and styles are idioms for architects to use, they "provide a common language"<sup>[6]</sup> or "vocabulary"<sup>[5]</sup> with which to describe classes of systems.

The main difference is that a pattern can be seen as a solution to a problem, while a style is more general and does not require a problem to solve for its appearance.

## Examples

---

Here is a list of architecture patterns, and corresponding software design patterns and solution patterns.

| Sub-domain area                            | Architecture pattern  | Software design patterns  | Solution patterns  | Related patterns  |
|--|---|---|--|---|
| <u>Data integration/SOA</u>                | <ul style="list-style-type: none"> <li>▪ <u>ETL (data extraction transformation and loading)</u></li> </ul>   | <ul style="list-style-type: none"> <li>▪ <u>Change data capture</u></li> <li>▪ Near real-time ETL</li> <li>▪ Batch ETL</li> <li>▪ <u>Data discovery</u></li> </ul>  | <ul style="list-style-type: none"> <li>▪ <u>Error handling</u></li> <li>▪ <u>Job scheduling</u></li> <li>▪ <u>Data validation</u></li> <li>▪ <u>Slowly changing dimensions load</u></li> </ul> | <ul style="list-style-type: none"> <li>▪ <u>EAI</u></li> <li>▪ Master data hub</li> <li>▪ <u>Operational data store (ODS)</u></li> <li>▪ <u>Data mart</u></li> <li>▪ <u>Data warehouse</u></li> </ul> |
|  | <ul style="list-style-type: none"> <li>▪ <u>MFT</u></li> </ul>  |   |  |   |
|  | <ul style="list-style-type: none"> <li>▪ <u>EAI/ESB</u></li> </ul>  | <ul style="list-style-type: none"> <li>▪ <u>Publish/subscribe</u></li> <li>▪ <u>Request/reply</u></li> <li>▪ <u>Message exchange patterns</u></li> </ul>  | <ul style="list-style-type: none"> <li>▪ One-way</li> <li>▪ Synchronous request/response</li> <li>▪ Basic callback</li> <li>▪ Claim check</li> </ul>   | <ul style="list-style-type: none"> <li>▪ <u>SOA</u></li> </ul>  |
| <u>Data architecture</u>                   | <ul style="list-style-type: none"> <li>▪ <u>Transaction data stores (TDS/OLTP)</u></li> <li>▪ <u>Master data store</u></li> <li>▪ <u>Operational data store</u></li> <li>▪ <u>Data mart</u></li> <li>▪ <u>Data warehouse</u></li> </ul>   | <ul style="list-style-type: none"> <li>▪ Custom applications databases</li> <li>▪ Packaged application databases</li> </ul>   |  | <ul style="list-style-type: none"> <li>▪ ETL</li> <li>▪ EAI</li> <li>▪ SOA</li> </ul>   |
| <u>Analytics and business intelligence</u> | <ul style="list-style-type: none"> <li>▪ Transactional reporting</li> <li>▪ Operational analytics</li> <li>▪ Business analytics</li> <li>▪ Predictive analytics</li> <li>▪ Prescriptive analytics</li> <li>▪ Streaming analytics</li> <li>▪ Data science and advanced analytics</li> <li>▪ NLP</li> </ul> | <ul style="list-style-type: none"> <li>▪ Transactional reporting data access</li> <li>▪ Operational reporting data access</li> <li>▪ Analytical reporting data access</li> <li>▪ Analytical dashboard data access</li> <li>▪ Operational dashboard data access</li> <li>▪ <u>Data mining</u></li> </ul> | <ul style="list-style-type: none"> <li>▪ Real-time dashboards</li> <li>▪ In-memory analytics</li> <li>▪ Statistical analysis</li> <li>▪ <u>Predictive analytics</u></li> </ul>                 | <ul style="list-style-type: none"> <li>▪ ETL</li> <li>▪ EAI</li> <li>▪ TDS</li> <li>▪ Operational data store</li> <li>▪ Data mart</li> </ul>  |
| <u>Master data management</u>              | <ul style="list-style-type: none"> <li>▪ Master data hub</li> </ul>   | <ul style="list-style-type: none"> <li>▪ Master data replication</li> <li>▪ Master data services</li> <li>▪ Master data synchronization</li> </ul>  |  | <ul style="list-style-type: none"> <li>▪ <u>Change data capture</u></li> <li>▪ EAI</li> <li>▪ STD</li> </ul>  |
| <u>Data modeling</u>                       | <ul style="list-style-type: none"> <li>▪ <u>Dimensional data modeling</u></li> <li>▪ <u>E-R data modeling</u></li> </ul>  | <ul style="list-style-type: none"> <li>▪ Modeling standards</li> <li>▪ Naming conventions</li> </ul>  |  |   |
| <u>Artificial intelligence</u>             | <ul style="list-style-type: none"> <li>▪ Decision management</li> <li>▪ <u>Speech recognition</u></li> <li>▪ Text analytics and NLP</li> <li>▪ <u>Natural language generation</u></li> <li>▪ Classic machine learning</li> <li>▪ Deep learning</li> </ul>   | <ul style="list-style-type: none"> <li>▪</li> <li>▪</li> </ul>  |  |   |

|  |  |  |  |  |
|--|--|--|--|--|
|  | <ul style="list-style-type: none"> <li>▪ <u>Robotic process automation</u></li> <li>▪ <u>Image and video analysis</u></li> </ul> |  |  |  |
|--|--|--|--|--|

Some additional examples of architectural patterns:

- Blackboard system
- Broker pattern
- Event-driven architecture
- Implicit invocation
- Layers
- Hexagonal architecture
- Microservices
- Action-domain-responder,
  - Model-view-controller
  - Presentation-abstraction-control
  - Model-view-presenter
  - Model-view-viewmodel
- Entity component system
- Entity-control-boundary
- Multitier architecture (often three-tier or  $n$ -tier)
- Object-oriented programming
- Naked objects
- Operational data store (ODS)
- Peer-to-peer
- Pipe and filter architecture
- Service-oriented architecture
- Space-based architecture
- Distributed hash table
- Publish-subscribe pattern
- Message broker
- Hierarchical model-view-controller

## See also

---

- List of software architecture styles and patterns
- Process Driven Messaging Service
- Enterprise architecture
- Common layers in an information system logical architecture

## References

---

1. Taylor, R.N.; Medvidović, N.; Dashofy, E.M. (2009). *Software architecture: Foundations, Theory and Practice* ([https://www.google.com/books/edition/Software\\_Architecture/npB5DwAAQBAJ?hl=en&pg=PR1](https://www.google.com/books/edition/Software_Architecture/npB5DwAAQBAJ?hl=en&pg=PR1)). Wiley. ISBN 9780470167748.
2. Andreessen, Marc (20 August 2011). "Why Software Is Eating The World" (<https://www.wsj.com/articles/SB10001424053111903480904576512250915629460>). *The Wall Street Journal*. Retrieved 25 April 2020.
3. Chang, Chih-Hung; Lu, Chih-Wei; Lin, Chih-Hao; Yang, Ming-Feng; Tsai, Ching-Fu (June 2008). "An Experience of Applying Pattern-based Software Framework to Improve the Quality of Software Development: 4. The Design and Implementation of OS2F" (<https://web.archive.org/web/20110922035257/http://jses.seat.org.tw/index.php/jses/article/viewFile/41/30>). *Journal of Software Engineering Studies*, Vol. 2, No. 6. the Third Taiwan Conference on Software Engineering (TCSE07). pp. 185–194. Archived from the original (<http://jses.seat.org.tw/index.php/jses/article/viewFile/41/30>) on 2011-09-22. Retrieved 2012-05-16. "Furthermore, patterns are often defined as something

"strictly described and commonly available". For example, layered architecture is a call-and-return style, when it defines an overall style to interact."

4. "Architectural Patterns: Definition" ([https://web.archive.org/web/20120623081009/http://aahninfotech.com/arct\\_pattern.html](https://web.archive.org/web/20120623081009/http://aahninfotech.com/arct_pattern.html)). AAHN INFOTECH (INDIA) PVT. LTD. Archived from the original ([http://aahninfotech.com/arct\\_pattern.html](http://aahninfotech.com/arct_pattern.html)) on 2012-06-23. Retrieved 2012-05-16. "Even though an architectural pattern conveys an image of a system, it is not an architecture as such. An architectural pattern is rather a concept that solves and delineates some essential cohesive elements of a software architecture. Countless different architectures may implement the same pattern and thereby share the related characteristics. Furthermore, patterns are often defined as something "strictly described and commonly available"."
5. Shaw, M.; Garlan, D. (1996). *Software architecture : perspectives on an emerging discipline*. Prentice Hall. ISBN 9780131829572.
6. "Ch. 3: Architectural Patterns and Styles" (<http://msdn.microsoft.com/en-us/library/ee658117.aspx>) . *Microsoft Application Architecture Guide* (2nd ed.). Microsoft Press. October 2009. ISBN 9780735627109.

## Bibliography

---

- Avgeriou, Paris; Zdun, Uwe (2005). "Architectural patterns revisited:a pattern language" (<http://eprints.cs.univie.ac.at/2698/1/ArchPatterns.pdf>) (PDF). *10th European Conference on Pattern Languages of Programs (EuroPlop 2005), Irsee, Germany, July*. UVK Verlagsgesellschaft. pp. 1-39. CiteSeerX 10.1.1.141.7444 (<https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.141.7444>). ISBN 9783879408054.
- Buschmann F.; Meunier R.; Rohnert H.; Sommerlad P.; Stal M. (1996). *Pattern-Oriented Software Architecture: A System of Patterns* ([https://www.google.com/books/edition/Pattern\\_Oriented\\_Software\\_Architecture\\_Aj\\_ahu\\_BS3hAC?hl=en&pg=PP1](https://www.google.com/books/edition/Pattern_Oriented_Software_Architecture_Aj_ahu_BS3hAC?hl=en&pg=PP1)). Wiley. ISBN 9781118725269.
- Bass L.; Clements P.; Kazman R. (2003). *Software Architecture in Practice* ([https://www.google.com/books/edition/Software\\_Architecture\\_in\\_Practice/mdilu8Kk1WMC?hl=en&pg=PP1](https://www.google.com/books/edition/Software_Architecture_in_Practice/mdilu8Kk1WMC?hl=en&pg=PP1)). Addison-Wesley. ISBN 9780321154958.

---

Retrieved from "[https://en.wikipedia.org/w/index.php?title=Architectural\\_pattern&oldid=1085545619](https://en.wikipedia.org/w/index.php?title=Architectural_pattern&oldid=1085545619)"

---

**This page was last edited on 1 May 2022, at 03:22 (UTC).**

Text is available under the Creative Commons Attribution-ShareAlike License 3.0; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.