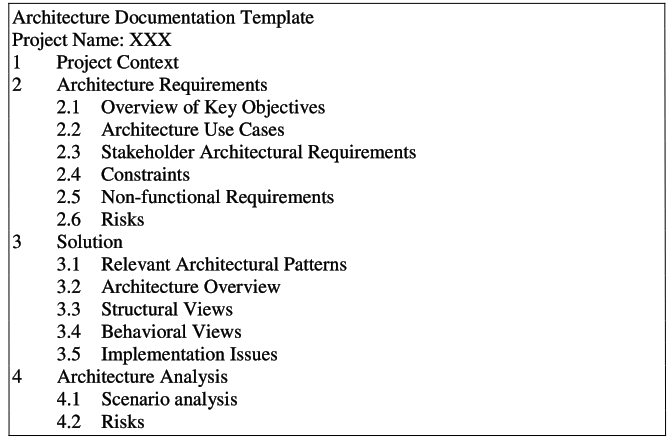
# Architecture Research Conclusion

By Abdullah Arslan. 24 February 2010

* The most common way to document an architecture is documenting its structure. But structure is not enough all the time. Other aspects of an architecture should be documented as well.
* Different stakeholders might want to look at the architecture from different viewpoints so that having different views.
* IEEE has a guideline to document architecture(Standard 1471). It recommends describing architecture by different views. The guideline contains only some example views. There are not a defined set of views. But some of them are more common than others.
* Architecture is high level design. It can contain some design issues but not much. Only important aspects of the system should be detailed. Further details should reside in Design Document.
* A candidate document structure could be like this: (*Documenting a Software Architecture Chapter from Essential Software Architecture*) 
* There is also another common approach containing predefined views. 4+1 Views approach. (Philippe Kruchten). This is also the base approach of RUP architecture document. 4 views are listed below. +1th view is the use case view that connects all others.

1. The logical view primarily supports behavioral requirements--the services the system should provide to its end users. Designers decompose the system into a set of key abstractions, taken mainly from the problem domain. These abstractions are objects or object classes that exploit the principles of abstraction, encapsulation, and inheritance. In addition to aiding functional analysis, decomposition identifies mechanisms and esign elements that are common across the system.
2. The process view addresses concurrency and distribution, system integrity, and fault-tolerance. The process view also specifies which thread of control executes each operation of each class identified in the logical view. The process view can be seen as a set of independently executing logical networks of communicating programs (“processes”) that are distributed across a set of hardware resources, which in turn are connected by a bus or local area network or wide area network.
3. The development view focuses on the organization of the actual software modules in the software-development environment. The units of this view are small chunks of software—program libraries or subsystems—that can be developed by one or more developers. The development view supports the allocation of requirements and work to teams, and supports cost evaluation, planning, monitoring of project progress, and reasoning about software reuse, portability, and security.
4. The physical view takes into account the system's requirements such as system availability, reliability (fault-tolerance), performance (throughput), and scalability. This view maps the various elements identified in the logical, process, and development views—networks, processes, tasks, and objects—onto the processing nodes.

* [A software architecture web site](http://www.rgoarchitects.com/blog/PermaLink,guid,9b5b976a-efc2-4210-ab5e-ff712f1797f2.aspx) proposes some common views as

1. Some sort layer view (usually block diagram)
2. A logical view (main classes/packages)
3. A deployment diagram (tiers, zones etc.)
4. A view to show concurrency and timing issues.
5. On SOAs there's also a service view (services, policies

**Conclusion:** IEEE suggested documenting the architecture from many different views in its standard 1471. Other authorities followed IEEE and developed different view based models. Those are SEI, 4+1(RUP), Simens and so on… All of them have different set of views. RUP’s 4+1 model is becoming more common.