GMLC 1.2.1 Grid Architecture High Resilience Track

Key Architecture Principles v3

This document contains a list of the key architectural principles to be used in developing the high resilience reference architecture views for GMLC 1.2.1.

Specific to Resilience

1. Grid resilience is the ability of the grid to avoid or withstand grid stresses without suffering operational compromise or to adapt to and compensate for the resultant strains so as to minimize compromise via graceful degradation. Grid resilience should be understood in terms of grid vulnerability, not in terms of hypothetical large rare events. Resilience elements comprise countermeasures to grid vulnerabilities.
2. The resilience domain has three regimes: stress avoidance, stress resistance, and strain adjustment. The scope of resilience includes more than just device/system failures and large exogenous events; it includes stressors related to operational issues and non-circuit exogenous forces.[1,2]
3. Grid resilience architectures use the following as foundational elements:
    - Situational awareness (monitoring, forecasting, extended grid state)
    - Planning/Design for resilience
    - Interoperability & standards
4. Grids are Ultra Large Scale systems, subject to normal failures.[3] Consequently, architecture should not treat failures as exception conditions but as part of normal operating modes.
5. The architecture views should employ redundancy/critical component backup to eliminate single points of failure and to provide options for strain adjustment.
6. The architecture views should incorporate buffering for the purpose of decoupling volatilities that may be induced by stochastic generation sources and the behaviors of controllable loads.
7. The architecture views should make use of modularity principles, namely module strength and module (de)coupling, to limit propagation of stresses and strains and therefore cascading failures.
8. Dependencies should be minimized to limit system brittleness (anti-resilience).[4] This may involve decoupling or may involve structures that provide some form of compensation for dependency, such as resilience loops in gas-electric systems.[5]

---

[1] JD Taft, Electric Grid Resilience and Reliability for Grid Architecture, PNNL, March 2018, available online: https://gridarchitecture.pnnl.gov/media/advanced/Electric_Grid_Resilience_and_Reliability_v4.pdf

[2] S Widergren, A Shankar, B Kelley, R Melton and J Taft, Toward a Practical Theory of Grid Resilience, GMLC, April 2018, available online: https://gridarchitecture.pnnl.gov/media/advanced/Theory_of_Grid_Resilience_final_GMLC.pdf

[3] Mark Klein, Linda Northrop, et. al, Ultra-Large-Scale Systems, Software Engineering Institute, Carnegie-Mellon University, 2006.

[4] A brittle system is a system characterized by a sudden and steep decline in performance as the system state changes. A system is brittle if it is unable to effectively absorb stress-induced shock.

[5] See for example JD Taft and A Becker-Dippmann, Grid Architecture, PNNL, January 2015, Section 6.2 Electricity/Gas Interaction, available online: https://gridarchitecture.pnnl.gov/media/white-papers/Grid%20Architecture%20%20-%20DOE%20QER.pdf

9. The architecture views should use structures that enable agility, adaptation, and configurability (including energy resource flexibility) to support strain adjustments. This is related to redundancy and modularity.
10. Architectural views should provide structures that can support or enable graceful degradation[6] in the event of grid strains.
11. Architectural views should provide structures that can support or enable fault tolerance[7] in the event of stress-induced failures.
12. Known resilient structures (distributed systems, layering, platforms, etc.) are to be preferred where appropriate.

General Architecture Principles

1. A good architecture is one that meets the needs of the stakeholders (especially the users) to their satisfaction, does not violate established principles of system architecture, and takes into account the relevant qualities and properties as the stakeholders require.
2. Good architectures have conceptual integrity (adhering to a set of core principles, clean of unnecessary complexities or 'exceptions,' similar problems are solved in similar ways, etc.).
3. Conceptual integrity is best achieved by a small cohesive team of like-minded architects. Architecture should be the product of a single architect or small team with an identified leader.
4. Essential functionality drives complexity, not architectural "elegance."
5. Architectural structures should have formal bases where possible to minimize ad hoc configurations with unknown properties.
6. Architecture should not depend on a particular commercial product or tool.
7. Architecture should produce enforceable key constraints.
8. The architect must be cognizant of the global system when optimizing subsystems.
9. Stakeholders should be involved in the process as much as possible, giving frequent and honest feedback on all aspects of the system architecture.
10. Each component should be responsible for only a specific feature or functionality, or aggregation of cohesive functionality. Components should be coupled only through explicit structure, avoiding hidden coupling where possible.
11. Reference grid architectures should inform interfaces.
12. The system architect is not a generalist, but rather a specialist in managing complexity.

[6] Graceful degradation is the ability of a system or network to maintain limited functionality even when a portion of it has been rendered inoperative. In graceful degradation, the operating efficiency or speed declines gradually as an increasing number of components fail. It is somewhat analogous to ductile failure in materials.
[7] Fault tolerance is the property that enables a system to continue operating properly in the event of the failure of (or one or more faults within) some of its components. A highly fault-tolerant system may continue at the same level of performance even though one or more components have failed.