Use Case #3 - Distributed Autonomy

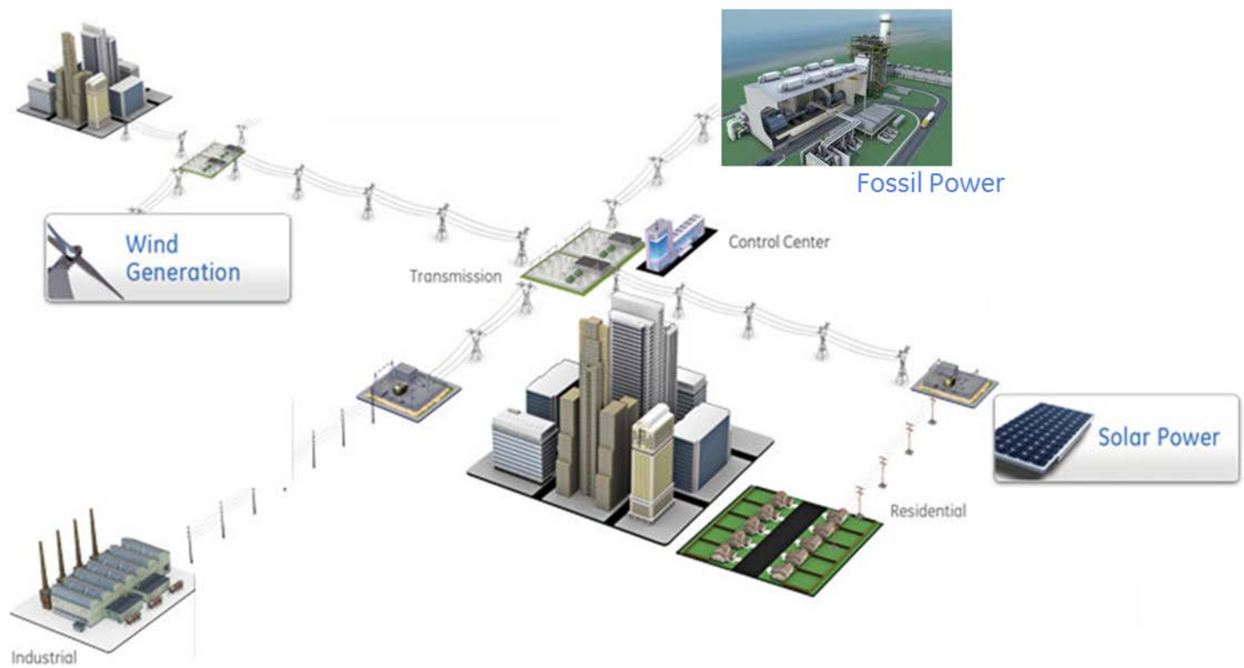
Resiliency and trust in the face of autonomy - issues on fault tolerance, reliability, connectivity.

Technology Working Group

Michael Dell'Anno
GE Global Research
michael.dellanno@ge.com

Industrial Internet Consortium

This document is an example use case defining a future potential industry need. It can be utilized to identify technology gaps and architectural requirements.



Technology/Power Generation/UC003: Distributed Control of Autonomous Systems	
Vignette	The power grid is a delicate balance of economic power generation and electric load. Each of these dynamically change as a function of many factors, some that are predictable and some that are not. Today this balance is achieved by dispatching assets to generate fixed energy levels (base load), load following, and spinning reserve. Satisfying this balance is becoming more challenging as generation becomes more distributed with a high fraction being renewables that are subject to uncontrolled environmental (wind, solar irradiance) variation. Optimum economic generation using renewables is difficult to achieve.
Actor	Power plant
Event	Power plant needs to optimally set power generation level

Pre-/Post-Conditions	**Pre:** ISO (Independent Systems Operator) and plant owner presets power generation levels. These fixed set points are followed throughout the day unless and event occurs requiring a centralized reallocation of power generation dispatch. **Post:** ISO posts required generation requirements. Plant owners accept requirements and dispatch their fleet of generation assets. Assets autonomously optimize in local clusters based upon cost, machine health and local grid demand. The assets continuously reallocate power generation requirements as changes in energy availability (solar, wind, faults), cost of energy, and grid load occur.
Technology	**Available:** • Autonomous algorithms and computation hardware are available at each power generation source. • Sensing technologies to measure machine health, generation capability, generation cost, and grid demand are available. • Data transport technologies between power generation facilities and between sensors and power generation facilities exists. **Gaps:** • Resilient autonomous power plant control algorithms that do not require local human supervision. • Reliable low latency data transport technologies to support distributed wide area autonomous power plant control. • Secure trusted communication sufficient to minimize risk of only remote human supervision. • Sensors to provide complete situational awareness for remote human supervision.
Fit Criteria	Communication Latency – • Grid Load Data **-** < 1.0 minute • Power set points from neighboring plants **-** < 1.0 minute Communication Reliability – • Outage frequency – Reliability >99.9% of an outage duration >= 10% of the maximum fault tolerance of autonomy control algorithm. Communication Security – • NERC CIP requirements • NISTIR-7628 Guidelines for Smart Grid Cyber security
Scenario	1. Power plant assesses its health, energy availability, generation cost, local grid demand 2. ISO dispatches power generation requirements 3. Plant owner accepts generation requirements from ISO 4. Plant owner dispatches power generation requirements to its fleet of assets 5. Each plant using a distributed control algorithm defines its optimum power generation level to satisfy the owner/ISO total generation requirements. 6. Each plant continues to assess its capability, cost, grid load, etc. and informs the local cluster of power generation assets. 7. The fleet of power generation assets continuously updates their optimum power generation levels.

1. Definition of Elements

This section provides two paragraphs for each element of the template. The first describes what is meant by the element. The second relates the element to the sources in CMAS Operational Scenarios for Taxonomy Validation and CMAS Mission Thread (and others, no doubt, as we refine the document).

Technology/<Domain>/UC<ID>: <Name>	
Vignette	This is a short story that encapsulates the purpose of the use case. Drawn from Operational Scenarios 'Purpose' and Mission Thread 'Vignette'.
Actor	The 'Actor', as per most texts. Drawn from Operational Scenarios 'Originator' and Mission Thread 'Nodes/Actors'.
Event	That which causes the use case to execute. Sometimes called 'Trigger'. Drawn from Operational Scenarios 'Event'.
Pre-/Post-Conditions	A more precise statement of what must be true before and after the use case executes. Because our use cases are at a rather abstract level, this covers some of what is to be found in Mission Thread 'Assumptions'.
Technology	A statement of what technology is required for the use case to work. We should focus this on the Architecture (i.e. Architectural Technology.) Drawn from Operational Scenarios 'Technology'.
Fit Criterion	An objective measurement of whether the use case met the requirement. In our context, this will often be related to performance and security. Drawn from Operational Scenarios 'Recipient'.
Scenario	A numbered sequence of steps. Drawn from Operational Scenarios 'Response Target' and Mission Thread 'Mission Steps'.

- The header includes our WG name, the domain whence the use case came, a unique ID (UC001 is reserved for Brad :), and a name.
- I recommend short vignettes, just enough to capture the purpose.
- Stick figures add nothing, but it *may* be helpful to identify the actor.
- The event should be an abstraction what, in-the-world causes the use case to execute.
- I'm big on pre- and post-conditions. They say what the desired end state is very compactly, and they hint at other use cases. For example, the example above says the airline must have paid up. What if they haven't? Do you upload anyway? This helps make our assumptions precise.
- 'Technology' is not a common element of use cases, but it will help us identify what requirements we place on the architecture.
- I am also big on fit criteria[1]. "[A] fit criterion is an objective measure of the requirement's meaning; it is the criterion for evaluating whether or not a given solution fits the requirement," from [the leaders in requirements engineering.]
- The scenario is the usual thing. I recommend we do not worry too much about alternatives and error analysis, but if we must we make a new use case.

[1] If your Greek is weak, it's one criterion, two criteria and three critters.