Tutorial on Distributed Computing for Decentralization of Grids and for IoT(A4)

Dr. David E. Bakken, training@naspinet.com , www.naspinet.com

January 2025

Abstract

Power grids have undergone massive changes in recent years. Deregulation of generation ~30 years ago began the move grids away from the centralized control center architecture that controlled a vertically-integrated "utility". However, recent developments including renewable integration are forcing changes even faster.

These recent evolutions are extremely disruptive**: historical design assumptions for power grids are rapidly becoming obsolete**. Edison and even Tesla would decreasingly be able to understand what is going on. Indeed, The Resnick Institute of Caltech begins a report with a bold statement[1] (**emphasis** is ours):

> The transformation occurring across the world's electrical systems **represents one of the greatest technological challenges industrialized societies have undertaken**. Reconfiguring a grid designed to carry power one way from reliable generation sources managed by few agents to a system increasingly laden with unreliable wind and solar energy while involving millions more participants using advanced technologies will introduce a high degree of uncertainty and variability into the future grid. These changes potentially **threaten reliability** of electrical supply and **must be carefully choreographed** to avoid widespread perturbations in cost, reliability and efficiency.

There is widespread agreement among power researchers that grids are getting increasingly stressed, and thus are likely less stable. However, better communications, coordination, and utilization of other distributed technologies long used in other industries can help alleviate this increasing stress.

This tutorial is intended for electric power grad students, engineers, and managers who wish to learn more about *distributed computing* (DC) technologies (i.e, **ABOVE the well-known network layers**) that have already, or likely soon will, affect their operations. It also applies to the Internet of Things (IoT). It helps them be aware of, and even gain non-superficial insight into, some of their "**unk-unks**" outside their core competencies, to avoid re-inventing the wheel. I.e., the opposite of the buzzword soup of Dilbert's pointy-haired boss.

If left to their own devices, power engineers will re-invent the wheel in computer networking, poorly[2].

— Power engineer at a NASPI meeting, circa 2015.

It's the unk-unks that will get you[3].

— Anonymous

How many of these "unk-unks" is your organization unprepared for?

Does it even know that they exist?

---

[1] *Grid 2020: Towards a Policy of Renewable and Distributed Energy Resources*, Resnick Institute Report, Caltech, September 2012.

[2] We're all good at something(s). So if the power grid were left to me to design/run, we'd be back in the stone ages, lusting for the infrastructure that Osama bin Laden had in those caves in Tora Bora! *DANGER WILL ROBINSON*: network protocols seem straightforward… but power engineers have little clue that the field of distributed computing exists, let alone how to find and apply its 40+ years of results. "A little knowledge is dangerous". – *Aldous Huxley*

[3] Unknown unknowns, i.e. unknown variables. See Mullins, John, "Discovering 'Unk-Unks': How innovators identify the critical things they don't even know that they don't know.", *MIT Sloan Management Review*, 4:4 (Summer 2007).

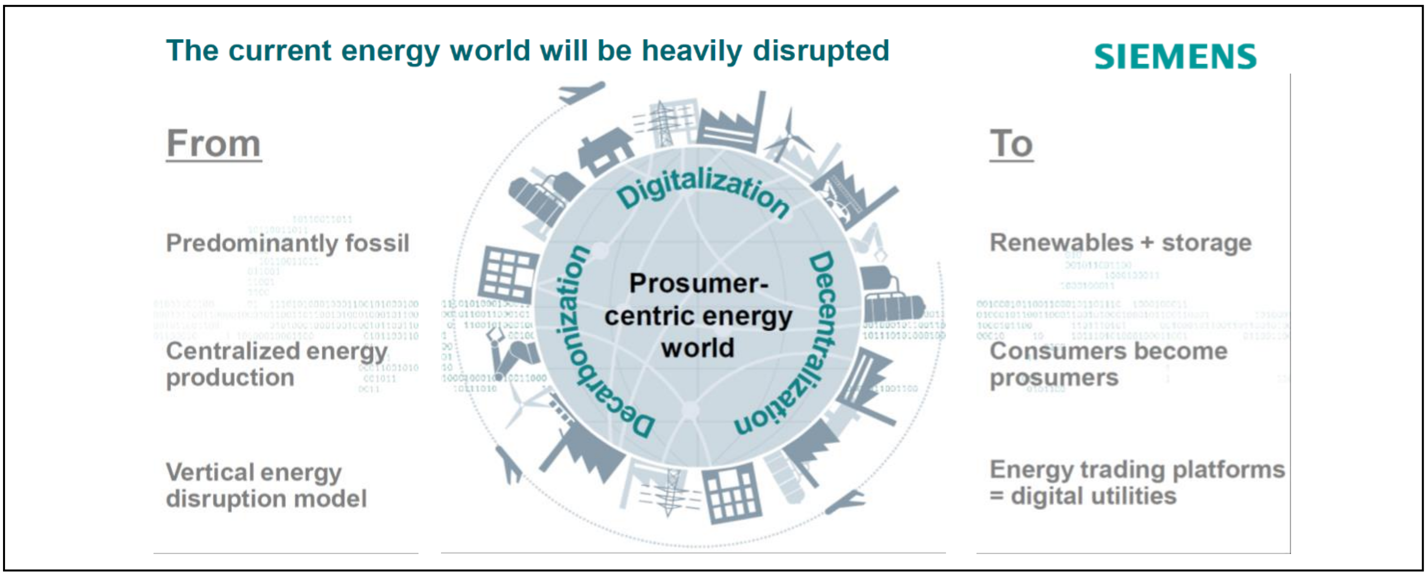
Context

So what actually IS distributed computing? *Computer networking* gets bytes of data from Point A to Point B with some statistical properties (delay, bandwidth, drop rate, …). *Distributed computing* evolved in the late 1970s to figure out how this: how, then, do we best use networks to program distributed application programs? I.e., how do we replicate, synchronize, coordinate, send data structures between computers, etc. From this arose *middleware*, consensus/agreement protocols, and many other distributed algorithms.

Grid Decentralization

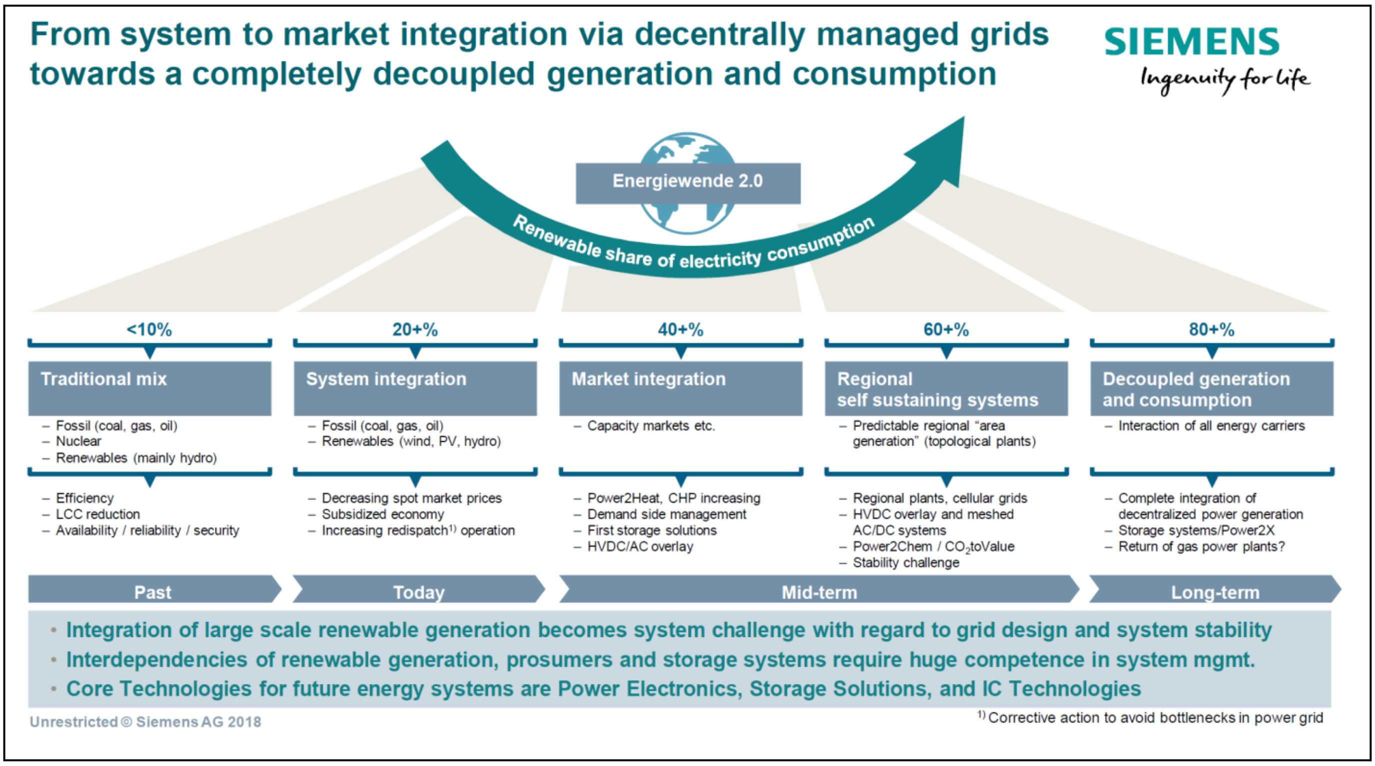
Quae ipsi loquuntur: (These (3 figures) speak for themselves) and are “Unrestricted © Siemens AG 2018”.

Driving Factors

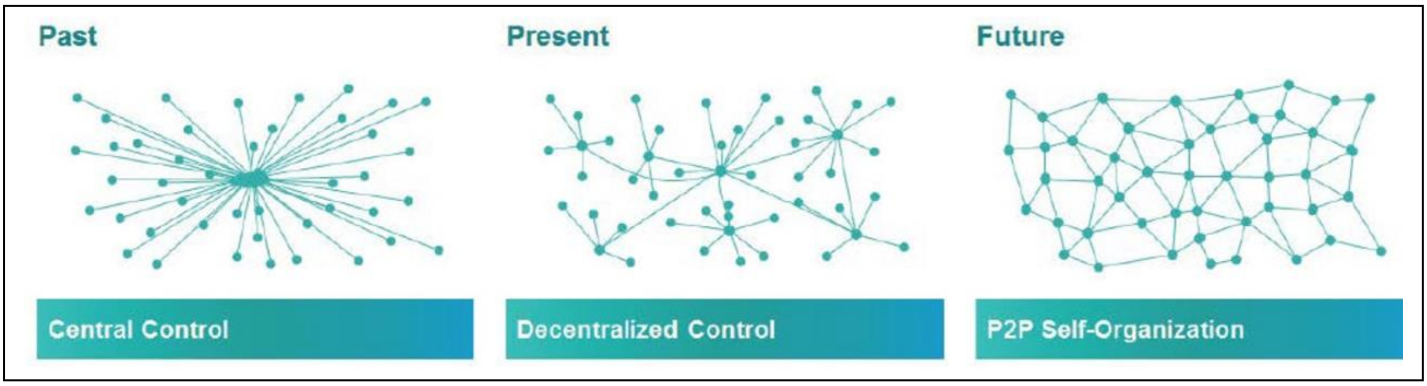


Evolution of Power Grids

Note: *Energiewende* is literally “Energy Turn”, so I interpret this to be “Energy Version 2.0”. So many driving factors and issues! Power engineers from utilities, ISO/RTO etc. and vendors would do well to study each bullet below, and share why you agree or disagree with each. There are very likely things that you were not (fully) aware of. I was unaware of many!



Evolution of Supporting ICT Structure: Centralized➔Decentralized➔Peer-to-Peer (P2P)



Note: the middle graph is a hierarchical tree emanating from the central node, and "Future" heavily requires DC.

Contents of Class

Each segment here is one 50-minute lecture with a 10-minute break. There will be a lunch break between the 4th and 5th segments. Note: these topics may be changed *slightly* as the course goes along in early 2025, based on feedback.

1. Grid decentralization & implications
    a. Changing landscape
    b. Challenges with existing practices
    c. Trends towards decentralization (e.g., see siemens figures above)
    d. Implementation challenges for decentralized grids
2. Resilience with fault tolerant computing
    a. Resilience vs fault tolerance
    b. Dependability: basic definitions, achieving, measuring, and validating it
    c. Fault assumptions & fault-tolerant computing and architectures
3. Distributed computing 1
    a. Overview of network protocols and DC
    b. Local vs. Remote invocations
    c. Resource sharing
    d. Consistency
    e. Middleware
4. Distributed computing 2
    a. Quality of Service (QoS)
    b. Cloud computing
    c. Fog and edge computing
    d. Some promising power apps for cloud, fog, and edge computing
5. Wide-area measurement systems (WAMS) & wide-area networks (WAN)
    a. Wan apps with extreme communications requirements
    b. It guidelines for achieving these requirements
    c. Implementations (GridStat)
6. Prosumers & coordinating them
    a. Prosumers and their impact on grids
    b. Peer-to-peer (P2P) computing
    c. Basics of fault-tolerant coordination (for prosumers and RAS)
    d. DCBlocks coordination platform
    e. DCBlocks coordination-based power applications (so far)
7. Miscellaneous
    a. How power apps can utilize better communications and coordination support
    b. UML, XML, and RDF; CIM (IEC 61870) & use for default comms/etc. Parameters
8. Analysis: pulling it all together
    a. Debate: middleware vs. Network level programming for power grid
    b. How middleware can inform and enable SDN usage for IEC 61850-90-5
    c. Critique of existing power-related technologies: IEC 61850-90-5, MPLS, SCTP, etc.

Bio

David E. Bakken, Ph.D. is a Professor Emeritus of Computer Science at Washington State University. He is a leading expert (arguably **the** leading one) in next-generation power grid communications and coordination. The core of his expertise is applied fault-tolerant distributed computing systems, and especially developing new kinds of software in the category of *middleware*, which makes distributed computing systems significantly easier to program. He has worked with the power grid since 1999, devoting 90% of his research time to this over the last quarter century. He has developed the *GridStat* data-delivery framework to deliver sensor data and commands with high enough availability and speed to support the fastest Remedial Action Schemes (RAS), which must operate on the order of 15 msec across dozens or hundreds of miles, sometimes with distributed coordination or two-way communications[4]. He has developed a prototype of a coordination framework, called *DCBlocks*, to support the coordination needed in today's decentralized power grids with individual prosumers and hierarchical federations of them; they also apply to RAS (its necessary consensus).

Prior to WSU, he was a research scientist at BBN, the research lab that built the first internet (the ARPANET) in 1969, and started the first research on midddleware in 1979. There he was a PI on DARPA projects developing the Quality Objects (QuO) framework, and was a lead PI collaborating with Cornell University, U. Illinois, and others. QuO was designed to provide adaptive QoS to wide-area applications in dynamic resource environments. QuO had approx. 60 person-years of BBN labor invested in it by DARPA and others. QuO has flown in Boeing experimental aircraft, been evaluated for a number of military systems, and used by Georgia Tech and others in their research.

Dr. Bakken has consulted to Amazon, Intel, Harris Corp, Real-Time Innovations, Siemens, and others. He has served on the Board of Directors of a cyber-security startup, and has also served as a voting member of doctoral committees from CMU, Georgia Tech, TU Munich, Chalmers, and others. Besides teaching this day class, he presently consults on middleware systems, architectural issues, and next-generation power grid ICT.

He received his BS degrees in Computer Science and Mathematics from WSU in 1985 and his MS and PhD degrees in Computer Science from The University of Arizona in 1990 and 1994, respectively. He worked for Boeing from 1985–1988 and developed the *Data_Flow* tool suite to help parallelize legacy FORTRAN simulations; it is still in use as of at least 2020. He not only developed it (to teach himself Unix and C) but recognized the need, lobbied for it, then just did it when nobody was looking. I.e., he takes the initiative.

Dr. Bakken has taught all the material in this course before, most of it 20+ times, and the core of it to power engineering students for 5 years in a novel interdisciplinary class at WSU[5], plus explaining parts of it in his research seminars. He has given approx. 100 invited presentations to power researchers, practitioners, and students since 1999, and is quite adept at translating between power and ICT issues and identifying the "disconnects" (misunderstandings or confusions) between them that inevitably arise in such interdisciplinary (and intradisciplinary!) engineering.

Miscellaneous Course Notes

Dr. Bakken will provide:

1. Printouts of all slides, bound in honeycomb bindings, for students to take notes on and to serve as a reference later.
2. A web page that has links to all references cited in his lecture slides; and other helpful material.

Recommended Followup Resources

1. Paulo Verissimo and Luis Rodrigues, *Distributed Systems for System Architects (Advances in Distributed Computing and Middleware)*, Kluwer, 2001.
2. Maarten van Steen and Andrew S. Tanenbaum, *Distributed Systems 4ed*, 2023, available for free from https://www.distributed-systems.net/.
3. Perry Lea, Edge Computing Simplified: Explore all aspects of edge computing for business leaders and technologies, Packt Publishing, 2024.

#1 is an outstanding but very insightful — a rare combination — explanation of DS from the top down. Most top-down approaches covering this area are inherently superficial. #2 has a lot more detail on many subjects and is from the bottom up. Ergo, my advice is to read #1 first, then skim and read #2 (and other similar textbooks) as needed. #3 is very helpful for people who are not distributed computing techies, from a Fellow at HP.

---

[4] David Bakken, Anjan Bose, Carl Hauser, David Whitehead, and Gregary Zweigle. "Smart Generation and Transmission with Coherent, Real-Time Data", Proceedings of the IEEE, 99:6(2011), 928–951. Note: Bakken was directly invited to write this paper.
[5] Anurag Srivastava, Carl Hauser, and David Bakken. "Study Buddies: Computer Geeks and Power Freaks are learning Smart Systems Together at Washington State", *IEEE Power & Energy Magazine*, 11(1), January/February 2013, 39–43.