



# Overview

- 1 Modern Civil Infrastructure Systems
  - Industrial Revolution
  - Post- Industrial Revolution
  - Transition to Information Era
- 2 Near-Term Challenges (2020-2060)
  - Crisis in US Infrastructure Investment
  - Urbanization and Sustainable Cities
- 3 Features of Modern Computing
- 4 Urban and Global Applications
- 5 Summary (Connections to Scientific Computing)





# Modern Civil Infrastructure Systems

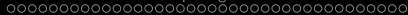
Various Sources (Google, ScienceDirect):

- **Civil Infrastructure Systems** provide for **human activity**, ranging in scale from **buildings to cities**.
- Includes supporting infrastructure: **water supply** networks; **energy** networks; **transportation** systems, **communication** systems.

Support Human Needs:

- Basic: Access to **clean air** and **clean water**.
- Health: Access to good **medical services**.
- Economic: Affordable low maintenance **housing**.
- Security: Protections against **crime**, **environmental attack**.





# Framing the Opportunity

## Metrics of Good Engineering Design:

- A good engineering design **works correctly**, has **good performance**, and is **economical**.
- Functionality and performance are resilient to uncertainties.
- System can be **easily upgraded** to take advantage of **new technologies**.

## Metrics of Good Systems Operation:

A well-run system has **situational awareness** and handles unexpected events:

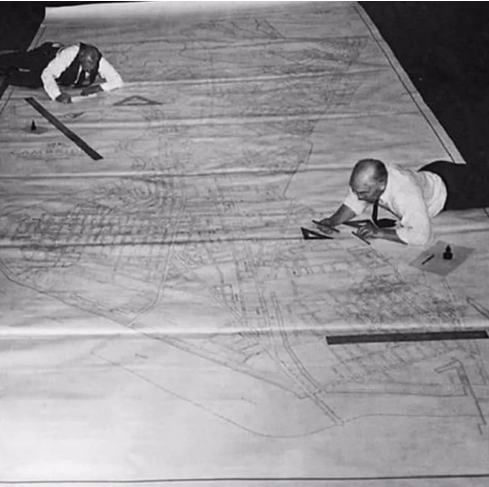
- **Sense the system state** and **surrounding environment**,
- **Look ahead** and anticipate **events**, and
- Take action to control **system behavior**.





# Framing the Opportunity

## ENCE 200: Architecture Development before CAD (pre- 1980):













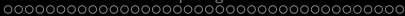




# Industrial Revolution

## Industrial Revolution Actually Changed the World!

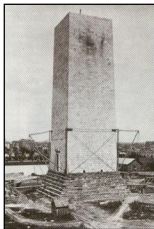
<b>Characteristics</b>	<b>Stage 1 Mechanical Era</b>	<b>Stage 2 Electrical Era</b>
Onset in the U.S.	Late 1700s.	Late 1800s.
Economic Focus	Agriculture/Mining	<b>Manufacturing</b>
Productivity Focus	Farming	Factory
Underlying Technologies	<b>Mechanical Tools</b>	<b>ElectroMechanical</b>
Product Lifecycle	Decades	Years
Human Contribution	Muscle Power	Muscle/Brain Power
Living Standard	Subsistence	<b>Quality of Goods</b>
Geographical	Family/Locale	Regional/National



# Industrial Revolution

## Advances in Civil Engineering

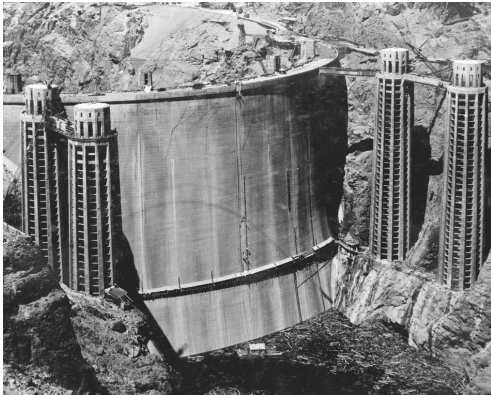
Year	Milestone
1854	Bessemer invents steel converter.
1849	Monier develops reinforced concrete.
1863	Siemens-Martin makes steel available in bulk.





# Post- Industrial Revolution

## Dams: Construction of the Hoover Dam (1931-1935)



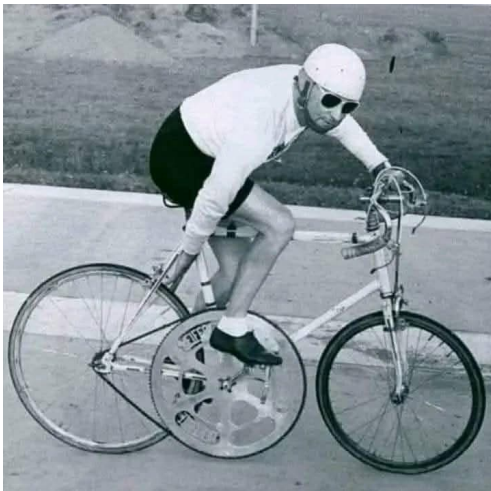
Hydroelectric power for Arizona, Nevada, Southern California;  
controls floods; provide irrigation water.





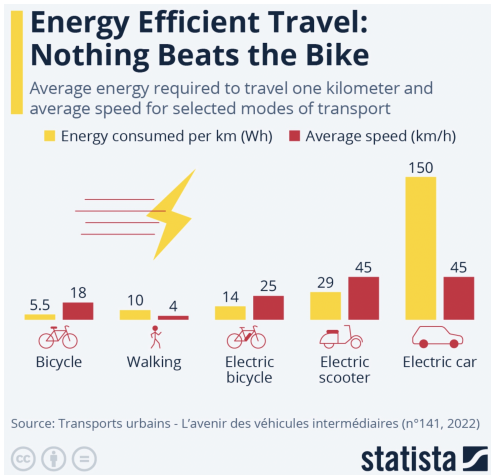
# Post- Industrial Revolution

**Bicycles:** Optimized for speed ...



# Post-Industrial Revolution

## Bicycles: Locomotion efficiency ...



# Post- Industrial Revolution

**Bicycles → Electric Cars → Flying Cars:**



From an **energy consumption standpoint**, **improvements** in **transportation performance** and **convenience** come at a **huge cost**.



# Transition to Information Era

**Motivation (Why?):** State-of-the-art systems – planes, trains and automobiles – rely on **human involvement** as a means for **sensing and controlling behavior**, e.g.,

- Driving a car,
- Manual collection of road tolls,
- Traffic controllers at an airport,
- Manual focus of a camera.

Systems work, but:

- Humans are slow.
- Humans make mistakes.
- They also easily tire.



# Transition to Information Era

## Emergence of Automation:

- New types of systems where **human involvement** for management of system functionality is **replaced** (or partially replaced) by **software automation**.

## Civil Engineering Applications:

- Automated road toll collection (Rt. 200), baggage handling systems at airports.
- Self-driving cars, smart traffic intersections/cities, etc ...



# Crisis in US Infrastructure Investment

## Exemplars of Work from the 1800s and 1900s

<b>From the 1800s</b>	<b>From the 1900s</b>
Erie Canal (1825)	New York City Subway (1904)
Transcontinental Railroad (1869)	The Panama Canal (1914)
Brooklyn Bridge (1883)	Holland Tunnel (1927)
Washington Monument (1884)	Empire State Building (1931).
	Hoover Dam (1936).
	Golden Gate Bridge (1937)
	Interstate Highway System (1956)

Source: Celebrating the Greatest Profession, Magazine of the American Society of Civil Engineers, Vol. 72, No. 11, 2002.

# Crisis in US Infrastructure Investment

## Universal Observations:

- Aging infrastructure becomes expensive to maintain.
- New (replacement) infrastructure is very expensive.
- Politicians are eager to talk up Infrastructure Investment , but slow to deliver ....

## Bottom line:

- Critical infrastructure is taken for granted and not a national priority (ASCE, IEEE).

Delay, delay, delay ....



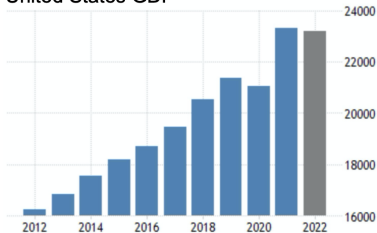
Bangkok, Thailand

# Crisis in US Infrastructure Investment

## Statistics:

- US: Post World-War II (1950-1970): 3% of Gross Domestic Product (GDP)
- US: 1980-present: 2% of GDP.
- China: 5% GDP.
- India: 9% GDP.

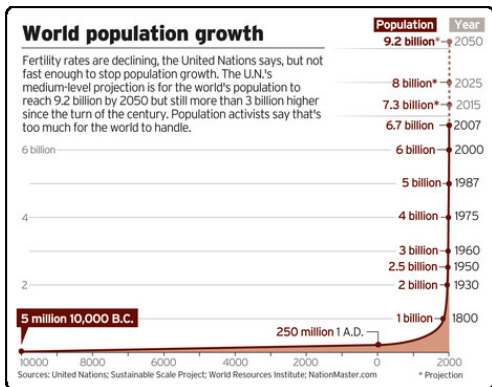
United States GDP



## Infrastructure Investment and Jobs Act (2021).

- Invest \$1.2T over 10 years.
- Sounds like a lot – but is it too low, too high?
- Increases investment by 0.5% of GDP.

# World Population Forecasts



Increasing Population → Increased Demand on Limited Resources  
 → Increasing need for **Improvements to System Efficiency.**

# Urbanization and Sustainable Cities

## Urbanization in America:

- In 2010, 82 percent of Americans lived in cities.
- By 2050 it will be 90 percent.

## Cities are responsible for:

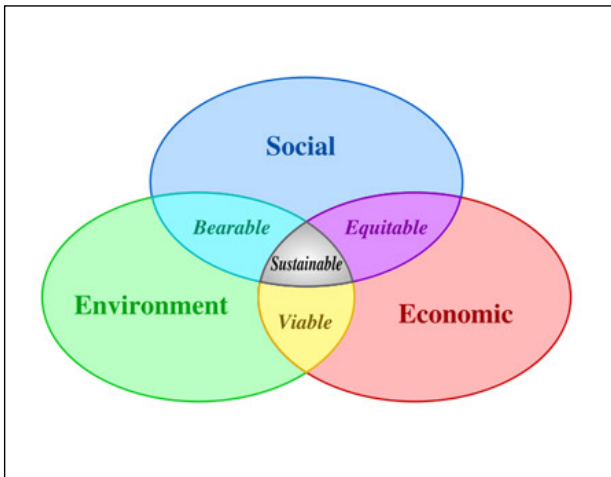
- Two thirds of the energy used,
- 60 percent of all water consumed, and
- 70 percent of all greenhouse gases produced worldwide.

## Sustainable cities (SIEMENS, Sustainable Cities, USA):

- Environmentally friendly infrastructures;
- Improved quality of life for residents;
- Good economics.

# Sustainable Urban Systems

Sustainability involves **physical**, **organizational** and **social** systems.



# Sustainable Urban Systems

Urban systems are like plants in your garden:

- Cities are defined by their **emergent properties** (e.g., beautiful flower  $\Leftrightarrow$  New York City Skyline).
- Cities **grow and flourish** based on societal and economic stimulus, and **fall into decay** when stimulus is absent.

But sustainability is a tough problem:

- Many of the world's large urban areas – so-called **mega-cities** – are in **poor economic shape**.

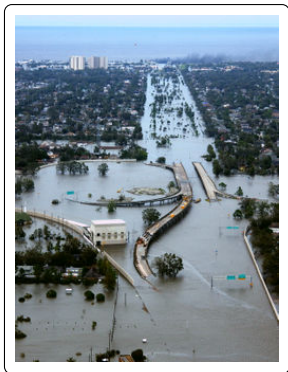
Cities are **system of systems**:

- Subsystems have a preference to **operate** as **independently as possible** from the other subsystems.
- Strategic **collaborations needed** to raise levels of **attainable performance** and **limit cascading failures**.

# Urban Infrastructure Protection and Recovery

## Case Study. Cascading Failures in Hurricane Katrina (2005)

- Hurricane Katrina caused a storm surge which, in turn, resulted in the failure of levees around New Orleans.
- This is a failure in the waterway network.
- A more conservative (expensive) design might have prevented this failure.
- But the failure didn't stop there.



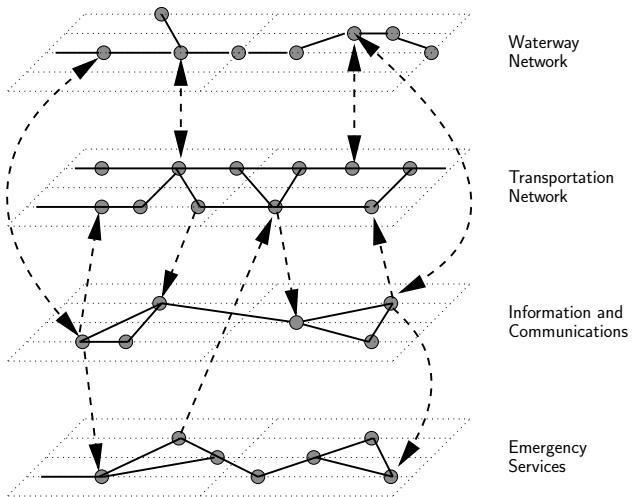
# Urban Infrastructure Protection and Recovery

## Cascading Failures in Hurricane Katrina:

- **Waterway system failure.** The levees were insufficient to resist the storm surge.
- **Highway and electrical power system failures.** Flooding resulted in failure of the electrical power and highway systems.
- **Federal emergency failures.** Inhabitants had to flee their homes, but few plans were in place for their orderly evacuation.
- **Social network failures.** After the inhabitants left their homes, looters stole property from evacuated properties.
- **Political system failures.** ...

# Urban Infrastructure Protection and Recovery

## Dependencies Among Urban Networks:



# Urban Infrastructure Protection and Recovery

## Lessons Learned:

Cascading failures of this type indicate that:

- There is a need to **understand** and **manage interactions** among **infrastructure networks** and **organizational** and **societal factors**.

## Basic Questions

- What kinds of dependencies exist between the networks?
- How will a failure in one network impact other networks?  
These are so-called **cascading failures**.
- What parts of a system are most vulnerable?

We need to look at **interactions between network models**.



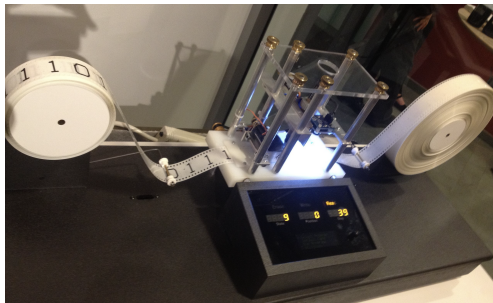




# Early Models of Computing

## Turing Machine Model: 1930s ...

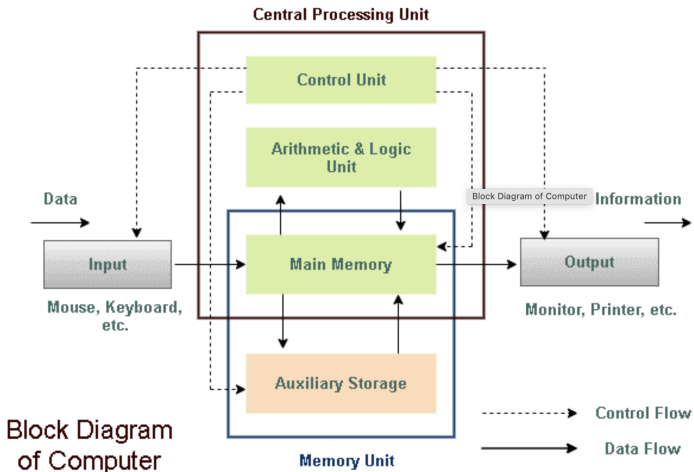
- Alan Turing (1936) created the **Turing machine** that included the **idea** of a **computer program**.



- Turing showed that you can **compute anything** using only **6 primitives**: right, left, print, scan, erase, nothing.

# Early Models of Computing

## Block Diagram of a Computer: 1980s ...



# State-of-the-Art Computing

## What does a modern computer do?

- Performs calculations – **billions** (sometimes even trillions) of **calculations per second**.
- Remembers results – **gigabytes** and terabytes of **storage**.

## Modern Programming Languages

- **Modern programming languages** have a **more convenient set of primitives**.
- Can abstract methods to create new primitives (e.g., user-defined objects).
- **Anything computable** in **one language** is **computable** in **any other programming language**.

# Man and Machine (Traditional View)

Man	Machine
<ul style="list-style-type: none"><li>● Good at formulating solutions to problems.</li><li>● Can work with incomplete data and information.</li><li>● Creative.</li><li>● Reasons logically, but very slow.</li><li>● Performance is static.</li><li>● Humans break the rules.</li></ul>	<ul style="list-style-type: none"><li>● Manipulates Os and 1s.</li><li>● Very specific abilities.</li><li>● Requires precise descriptions of problem solving procedures.</li><li>● Dumb, but very fast.</li><li>● Performance doubles every 18-24 months.</li><li>● Machines will follow the rules.</li></ul>

# Sensible Problem Solving Strategy

Let engineers and computers play to their strengths:

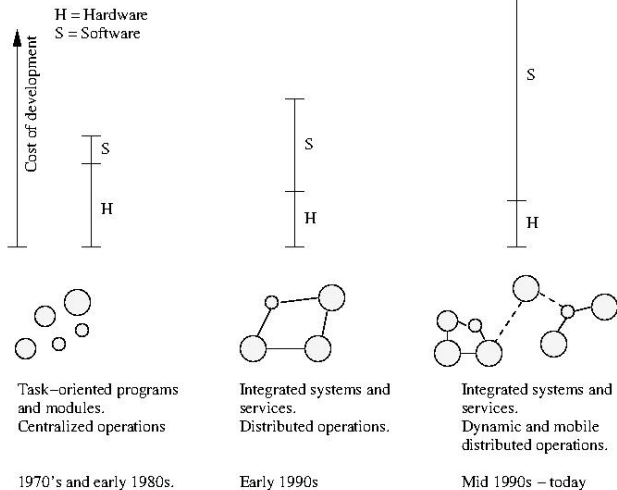
- Accelerates the solution procedure.
- Enables the analysis of problems having size and complexity beyond manual examination.
- Adds value in areas that will lead to long-term economic growth.

**Getting things to work** We need to:

- Describe to the computer solution procedures that are completely unambiguous.
- Look at data, organization and manipulation of data, and formal languages.

# Expanding Expectations of Computing

## Economics of computing and systems development





# Evolution of Computer Languages

**Computer Languages.** Formal description – [precise grammar](#) – for how a problem can be solved.

**Evolution.** It takes about a decade for significant advances in computing to occur:

Capability	1970s	1980s	1990s
Users	Specialists	Individuals	Groups
Usage	Numerical computations	Desktop computing	E-mail, web, file transfer.
Interaction	Type at keyboard	Screen and mouse	audio/voice.
Languages	Fortran, C	MATLAB	Python, Java

# Popular Computer Languages

Tend to be **designed** for a **specific set of purposes**:

- FORTRAN (1950s – today). Stands for formula translation.
- C (early 1970s – today). New operating systems.
- C++ (early 1970s – today). Object-oriented version of C.
- MATLAB (mid 1980s – today). Stands for matrix laboratory.
- Python (1990s – today). Object-oriented scripting language.
- HTML (1990s – today). Layout of web-page content.
- Java (1994 – today). Object-Oriented language for network-based computing.
- XML (late 1990s – today). Description of data on the Web.

# Post- 2000 Era

**Imagine:** What if COVID-19 had arrived in 2000?

- No iPhone, No iPad, No iTunes.
- No Facebook, No Instagram, No WhatsApp.
- No Google Maps, No Google Streetview.
- No Dropbox, No Zoom.

## **Recent Advances in Technology:**

- Average internet speeds: In 2000, 0.07 Mbs; In 2009, 5-7 Mbs; In 2020, 100-200 Mbs; 5G, 1000-2000 Mbs.
- Cloud-based data storage and computational services (AWS).
- New languages: [Swift](#) → App development on [iPhone/iPad](#).
- Many new types of [sensors](#) and [methods of data collection](#).

# Post-2010 Era → Emergence of AI

State-of-the-Art Implementation (2020, Google, Siemens, IBM)

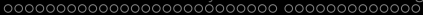
- AI and ML will be **deeply embedded** in new **software and algorithms**.

Artificial Intelligence:

- **Knowledge representation** and **reasoning** with ontologies and rules. Semantic graphs. Executable **event-based processing**.

Machine Learning:

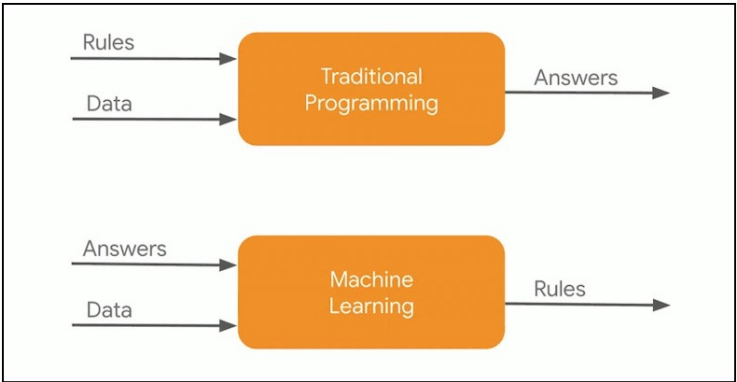
- Modern neural networks. Input-to-output prediction.
- Data mining.
- Identify **objects**, **events**, and **anomalies**.
- Learn structure and sequence. **Remember stuff**.



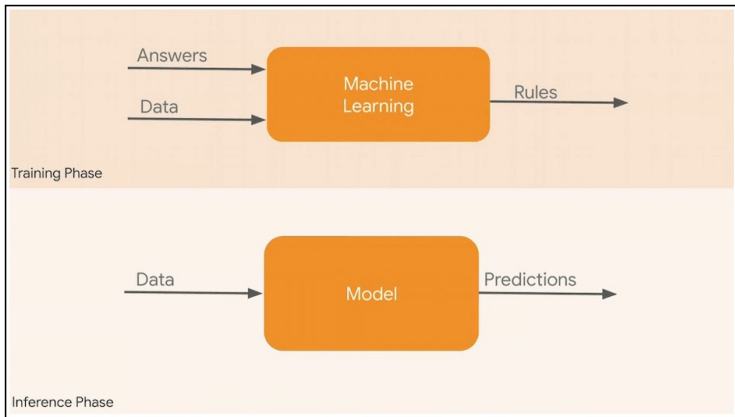
# Man and Machine (AI-ML View)

Man	AI-ML Machine
<ul style="list-style-type: none"><li>● Good at formulating solutions to problems.</li><li>● Can work with incomplete data and information.</li><li>● Creative.</li><li>● Reasons logically, but very slow. Forgetful.</li><li>● Performance is static.</li><li>● Humans make the rules, then they break them.</li></ul>	<ul style="list-style-type: none"><li>● Manipulates 0s and 1s.</li><li>● Can work with incomplete data and information.</li><li>● Creative.</li><li>● Fast logical reasoning.</li><li>● Performance doubles every 18-24 months.</li><li>● Data mining can discover the rules.</li></ul>

# Traditional Programming vs AI-ML Workflow

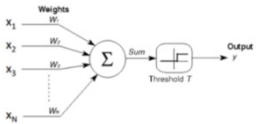


# Traditional Programming vs AI-ML Workflow



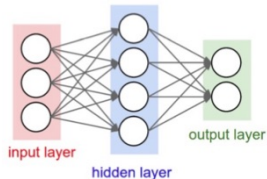
# Machine Learning Capabilities (1980-1990)

## Expressive Power of a Neural Network

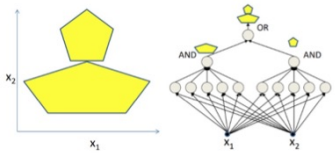
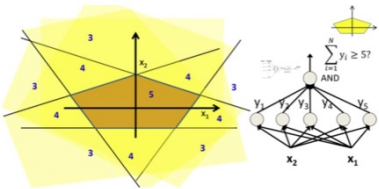


$$y = \begin{cases} 1 & \text{if } \sum_{i=1}^d w_i x_i \geq T \\ 0 & \text{else} \end{cases}$$

## Neural Network with Single Hidden Layer



## Approximation of Functions / Boolean Logic



# Machine Learning Capabilities (1997-2014)

## Learning Streams of Text

- Download complete works of Shakespeare (5.4 million characters)
- Train machine to remember text.
- Write new Shakespeare!

```

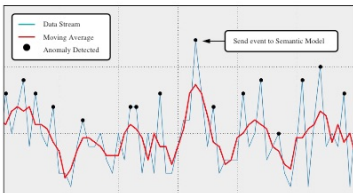
Sample after 1000 Iterations
-----
OVIERS. Allow up, who, Will a moutice,
Resson thot boy Skcents allien; comn iugsius,
And encarsenover heth last;)
TARTOR. When no; devires at thogh me beligo jody?
NANCHELO. Trom juthar and bur itnot spock,
That as take hava wendisho

Sample after 2000 Iterations
-----
VIBOTARD. Walk this boy and door as am 'stone!
NTRURLEN. Being entainure af Saton.
Els that by much that, I mion's now, who make foil the Kid!!
CLOVE. Why which Ham'd?
QUICKLY. And stand quast of I; this Fi

Sample after 2500 Iterations
-----
FORD. Nay, You're, excoun: and now did yet.
PAROLLES. Take DOTRY
This who is begin Cnoban a bows; but yet which that have be,
Oll, thou stan, and me not ready withered gids
And he in the pleases or pardon us.
Mer. I pray you, how can, and tu

Sample after 3000 Iterations
-----
Zume, 'tigoning, deaz?
My las, an hour, chork'd more in my grain,
I an thing forent innoion, madem! I thank I do speak you?
ARBESTIKO. But his bosines, giving to know; foward to
the distvemail.' The to you well know yea, my lowi.
SECOND CMENOR. He needs, for the refords are;
    
```

## Time Series Anomaly Detection



## Time Series Prediction

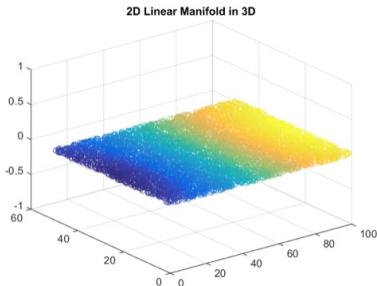
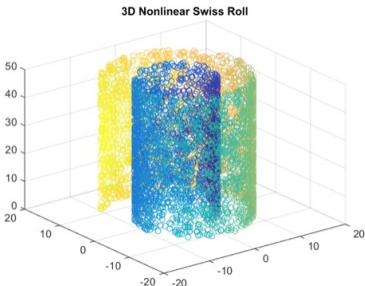


# Classification of Machine Learning Problems

## Dimensionality Reduction

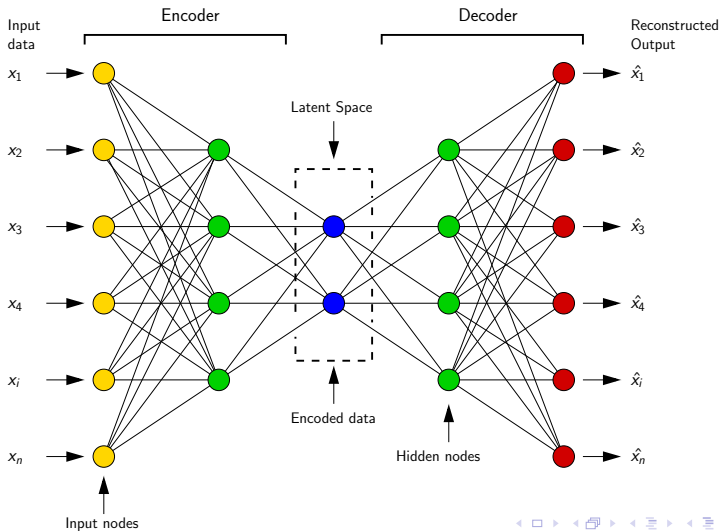
Strategies of **dimensionality reduction** involve transformation of data to new (lower) dimension in such a way that some of the dimensions can be **discarded without a loss of information**.

**Example:** Projection of Swiss Roll data in 3D to 2D ...



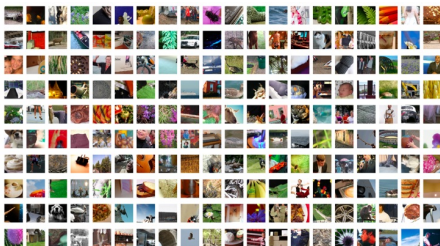
# Classification of Machine Learning Problems

## AutoEncoder (Encoder-Decoder-Reconstruction)



# Classification of Machine Learning Problems

## ImageNet and Deep Learning (2009-present)



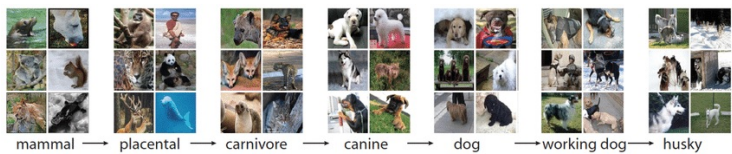
## Indexed Database of 14.2 million Images

- Project initiated by Fei Fei Li in 2006
- Image annotation process crowd sourced via Amazon's Mechanical Turk. Categories derived from WordNet.
- Well organized → supervised machine learning.

# Classification of Machine Learning Problems

## ImageNet and Deep Learning Capabilities:

- Identify objects in an image.
- 27 high-level categories; 21,800 sub-categories.



# ImageNet and Deep Learning

## Capabilities (2018):

- Identify relationship among multiple objects in a image.

**Example.** Dog riding skateboard











# ImageNet and Deep Learning

Captions generated by a neural network:

cs.stanford.edu/people/karpathy/deepimagesent/

40 maps that explain Amazon Web Services | Primer | Math | Pro | deeplearning.net/tu/ | Deep Learning Tutor | deep learning | PHILIPS - Golden Ears | Language Technology | MyDCare - Dashbo... | Other bookmarks

 <p>"man in black shirt is playing guitar."</p>	 <p>"construction worker in orange safety vest is working on road."</p>	 <p>"two young girls are playing with lego toy."</p>	 <p>"boy is doing backflip on wakeboard."</p>
 <p>"girl in pink dress is jumping in air."</p>	 <p>"black and white dog jumps over bar."</p>	 <p>"young girl in pink shirt is swinging on swing."</p>	 <p>"man in blue wetsuit is surfing on wave."</p>

Navigation icons: back, forward, search, refresh, etc.

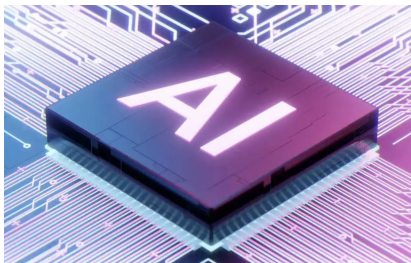
# Machine Learning at Scale

Object-recognition module:

- 24 million nodes; 140 million parameters; 15 billion connections.

Source: Fei Fei Li, TEDTalk, YouTube 2015.

**AI Chips:** Nvidia, Google TPUs, etc ...



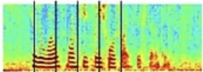
# Machine Learning at Scale (Transformers)

Input

Pixels:



Audio:



"Hello, how are you?"

Pixels:



Output

"leopard"

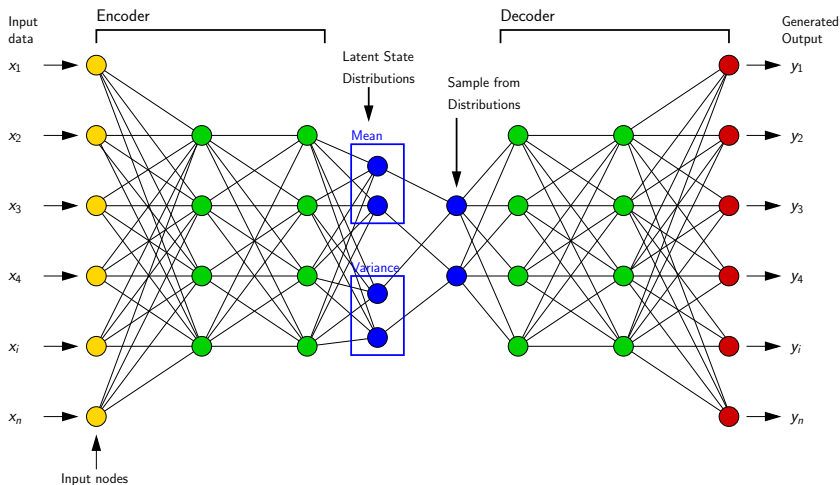
"How cold is it outside?"

"Bonjour, comment allez-vous?"

"A cheetah lying on top of a car"

## Post-2020 Era → Explosion of Generative AI

## Variational AutoEncoders (Generative Models)

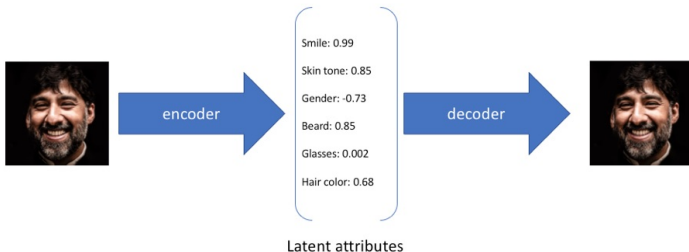


# Post-2020 Era → Explosion of Generative AI

Standard Autoencoders vs. Variational Autoencoders:

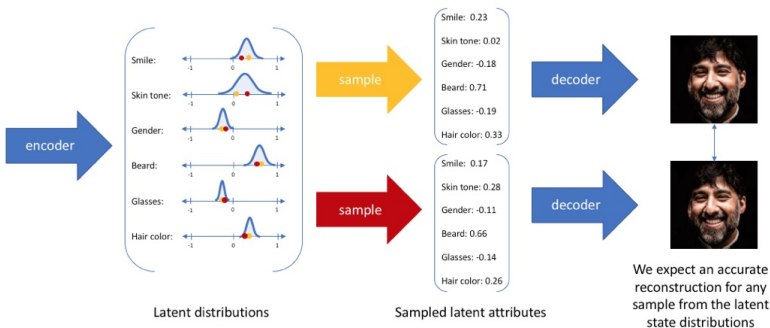
- A **standard autoencoder** outputs a **single value** for each **encoding dimension**.
- **Variational autoencoders** provide a **probability distribution** for each latent attribute.

**Example:** Single value representations for latent attributes:



# Post-2020 Era → Explosion of Generative AI

## Image Reconstruction: sampled from latent distributions ...



Source: Jordan J., Variational Autoencoders, Data Science, March 2018.

# Post-2020 Era → ChatGPT

## ChatGPT: Generative Pre-Trained Transformer

AU

write paragraph on civil engineering



Civil engineering is a fundamental discipline that shapes and transforms the physical infrastructure of our modern world. It encompasses the design, construction, and maintenance of a vast array of structures and systems, ranging from buildings, bridges, and roads to water supply networks, sewage systems, and environmental protection facilities. Civil engineers are responsible for ensuring the safety, functionality, and sustainability of these essential structures that support and enhance our daily lives. Their work involves a deep understanding of materials, mechanics, geotechnical engineering, structural analysis, and urban planning. By integrating scientific principles with innovative technologies, civil engineers address societal needs, address challenges posed by urbanization, and contribute to the creation of resilient and efficient urban environments. Whether it's developing eco-friendly infrastructure, earthquake-resistant buildings, or efficient transportation systems, civil engineering remains at the forefront of shaping a better and more connected world.

↻ Regenerate

Free Research Preview. ChatGPT may produce inaccurate information about people, places, or facts. [ChatGPT August 3 Version](#)

# Post-2020 Era → ChatGPT

AU

integrate  $\sin(3x) \cdot \sin(2x) dx$ 

To integrate the expression  $\int \sin(3x) \cdot \sin(2x) dx$ , we can use the product-to-sum trigonometric identity, which states that:



$$\sin(A) \cdot \sin(B) = (1/2) \cdot [\cos(A - B) - \cos(A + B)].$$

Using this identity, we can rewrite the integral as:

$$\begin{aligned}\int \sin(3x) \cdot \sin(2x) dx &= (1/2) \int [\cos(3x - 2x) - \cos(3x + 2x)] dx \\ &= (1/2) \int [\cos(x) - \cos(5x)] dx.\end{aligned}$$

Now, we can integrate each term separately:

$$\int \cos(x) dx = \sin(x) + C_1,$$

$$\int \cos(5x) dx = (1/5)\sin(5x) + C_2.$$

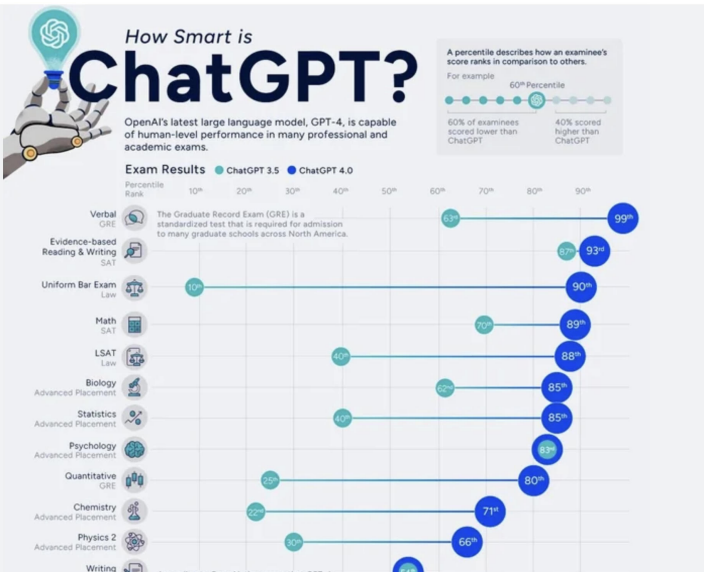
Putting it all together:

$$\begin{aligned}\int \sin(3x) \cdot \sin(2x) dx &= (1/2) \cdot [\sin(x) + C_1 - (1/5)\sin(5x) - C_2] \\ &= (1/2) \cdot [\sin(x) - (1/5)\sin(5x)] + (C_1/2 - C_2).\end{aligned}$$

So, the integral of  $\sin(3x) \cdot \sin(2x)$  with respect to  $x$  is:

$$\begin{aligned}\int \sin(3x) \cdot \sin(2x) dx &= (1/2) \cdot [\sin(x) - (1/5)\sin(5x)] + C, \\ \text{where } C &= C_1/2 - C_2 \text{ is the constant of integration.}\end{aligned}$$

# Post-2020 Era → ChatGPT



# Post-2020 Era → AI Generated Architecture

## Convergence: Engineering-Architecture-AI

AI-generated art ...



AI-generated building architecture





# AI and Machine Learning in CEE

# Machine Learning in CEE

## Opportunities for Machine Learning in CEE:

- Predicting system response and performance.
- Interpreting data and formulating models to predict component and subsystem-level properties.
- Information retrieval from images and text.
- Recognizing patterns in streams of sensed data.

## Economic Considerations (CS Perspective):

- Urban infrastructure is permanent/semi-permanent and very expensive to build and maintain.
- Prioritize improvements to efficiency by identifying and removing bottlenecks in performance.
- Use AI-ML to identify events, cause-and-effect relationships, and design of actions that enhance system performance.

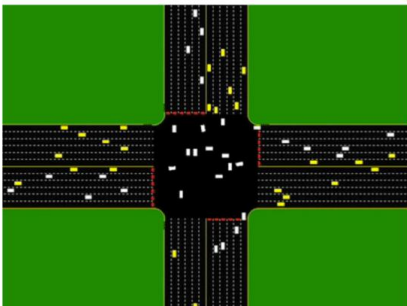
# AI-ML Enabled Decision Making

**AI Traffic System:** Reduced congestion and accident rates:



# AI-ML Enabled Decision Making (Self-Driving Cars)

**Goal.** Improve performance by removing bottlenecks → **no human driver; no traffic lights.**



**Remark:** 95% of the requirements are for the system software.

**Source:** ISR visitor from GM Research.

**Remark:** Tesla will produce self-driving cars by 2016.

**Source:** Elon Musk.

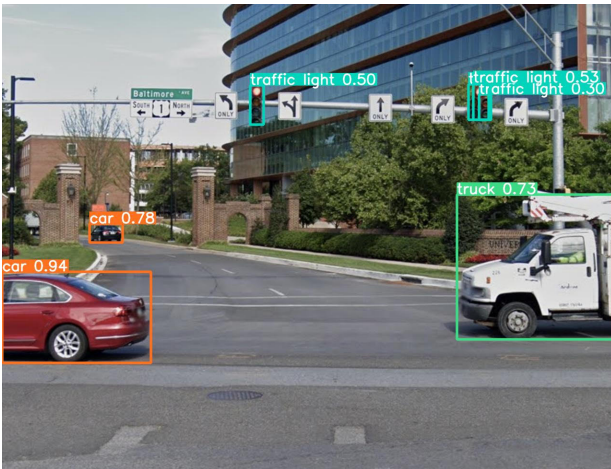
Stop signs and **traffic lights** are replaced by **mechanisms for vehicle-to-vehicle communication** (Adapted from <http://citylab.com>).





# AI-ML Enabled Decision Making (Self-Driving Cars)

Ainur's Experiments with Computer Vision (OpenCV):





# AI-ML Enabled Decision Making (Self-Driving Cars)

## Navigating a Busy Traffic Intersection:

### How the car operates

- Any object the vehicle's sensors spot is interpreted by software to determine if it's a pedestrian, cyclist, vehicle or something else.
- Using what it's learned from previous driving, the software makes predictions about what objects will do next.
- The software analyzes the information to decide whether it is safe to accelerate, turn or hit the brakes.

Source: Google  
Graphic: Tribune News Service



### How the car sees the world

This computerized image is what Google researchers monitoring sensor data see as they ride in the vehicle.

-  Other vehicle
-  Pedestrian
-  Cyclist
-  Objects that warrant caution
-  A crosswalk, indicating the car needs to stop
-  A traffic signal, warning of upcoming railroad tracks
-  Path where Google's car intends to go

- Identify various kinds of objects (e.g., vehicles, crosswalk).
- Predict what objects will do next.
- Conduct safety assessment.
- Take action.

# Google DeepMind (2018-2020)

## Teach Self-Driving Cars to Navigate a City without a Map



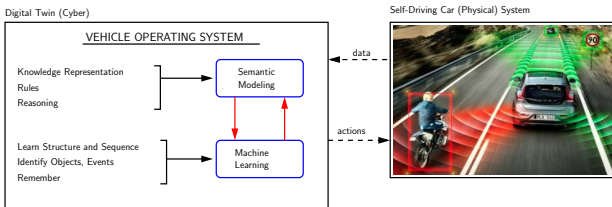
**Test Cities:** London, Paris, New York.

# Digital Twin Systems

New Computing Infrastructure → New System Abstractions

# Digital Twins (2000-today)

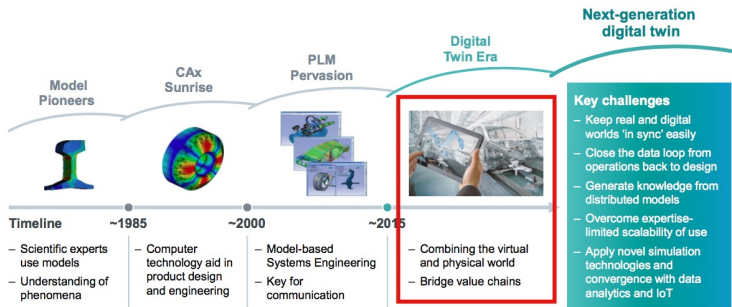
**Definition.** Virtual representation of a physical object or system that operates across the system lifecycle (not just the front end).



## Required Functionality

- Mirror implementation of physical world through real-time monitoring and synchronization of data with events.
- Provide algorithms and software for observation, reasoning, and physical systems control.

# Digital Twins (Business Case + Applications)



## Many Applications

- NASA Spacecraft
- Manufacturing processes
- Building operations
- Personalized medicine
- Smart Cities
- ... etc.

# Digital Twins (Technical Implementation)

## Technical Implementation (2023, Google, Siemens, IBM)

- AI and ML will be **deeply embedded** in new **software and algorithms**.

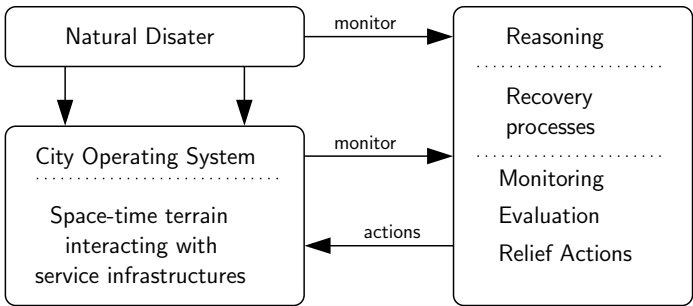
### Artificial Intelligence:

- **Knowledge representation** and **reasoning** with ontologies and rules. Semantic graphs. Executable **event-based processing**.

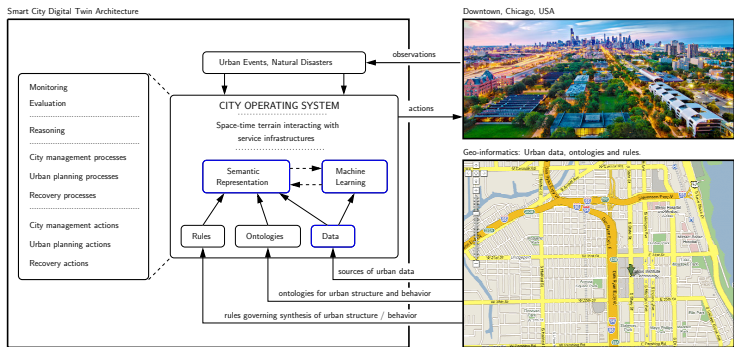
### Machine Learning:

- Modern neural networks. Input-to-output prediction.
- Data mining.
- Identify **objects**, **events**, and **anomalies**.
- Learn structure and sequence. **Remember stuff**.

# Digital Twin: City Operating Systems



# Smart City Digital Twins (2018-2019)

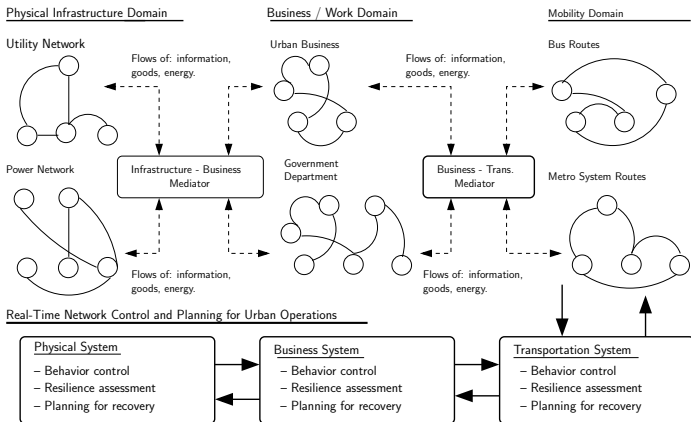


**Required Capability.** Monitoring and control of urban processes.

**Complications.** Potentially, a very large number of digital twins.

Distributed decision making.

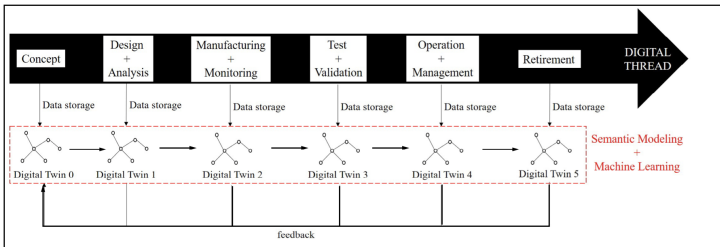
# Smart City Digital Twins (2018-2019)



**Requirements.** Support for digital twin **individuals** and digital twin **communities**.

# Digital Thread Systems

## Digital Threads: (Cradle-to-Grave Lifecycle Support) ...

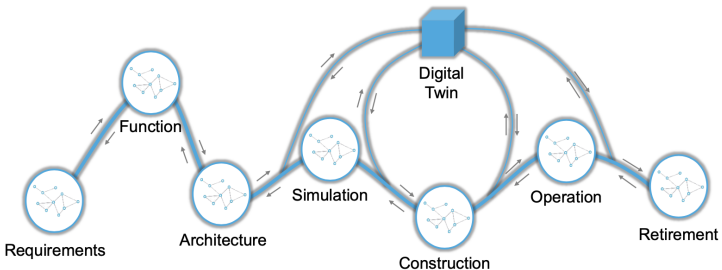


## Graph-based Approach

A lot of **model-centric engineering** boils down to representation of systems as graphs and sequences of graph transformations punctuated by decision making and work/actions.

# Digital Thread Systems

## Digital Thread System at INL: (Conceptual Model) ...



**Def'n:** A digital thread is an **interconnected software data exchange** used to enable **digital engineering** and **digital twinning systems** ...





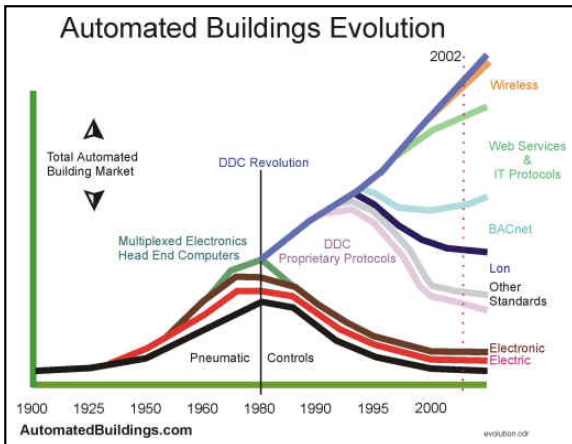
# Modern Buildings (Key Features)

Modern buildings are:

- Advanced, self-contained and tightly controlled environments design to provide services (e.g., transportation, lighting, etc).
- Large size (e.g., 30,000 occupants, thousands of points of sensing and control for air quality and fire protection).
- Many stakeholders; highly multi-disciplinary.
- Buildings have networks for: arrangement of spaces; fixed circulatory systems (power, hvac); dynamic circulatory systems (flows of energy).
- Many sources of heterogeneous data.
- Necessity of performance-based design and real-time management.
- System functionality controlled by software!

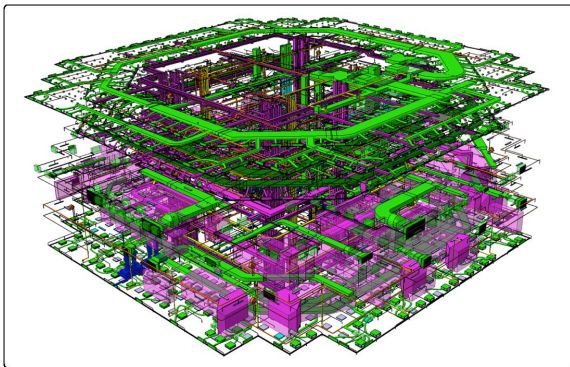
# Modern Buildings (Key Features)

Large-scale building systems are packed with automation:



# Modern Buildings (Key Features)

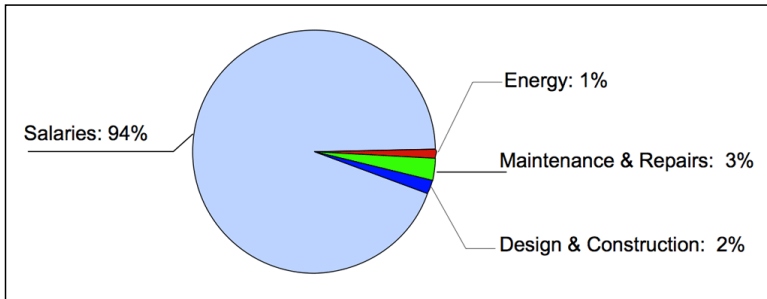
Large-scale building systems are intertwined networks of networks:



Understanding the **relationships among the networks** and their combined behaviors can be **very challenging**.

# Modern Buildings (Economics)

Lifecycle costs in office buildings over a 30-Year period:



Energy systems have a huge impact on building occupant comfort and **indoor air quality** which, in turn, **affects salary performance**.

Source: United Technologies Research Center, 2009.

# Modern Buildings (Integrated Energy Systems)

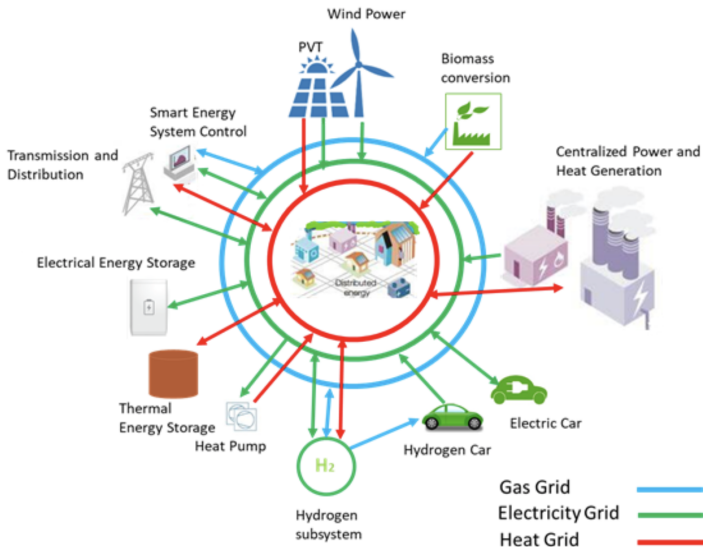
## Trend toward Integrated Energy Systems:

- Commercial and residential buildings consume 1/3 of the world's energy.
- And by 2025, buildings will consume more energy than the transportation and industrial sectors combined.
- **Standard models** of building operation rely on **centrally produced power** as a source of high-grade energy.
- Advances in technology allow for consideration of alternatives, such as **local production of power**.

## Examples:

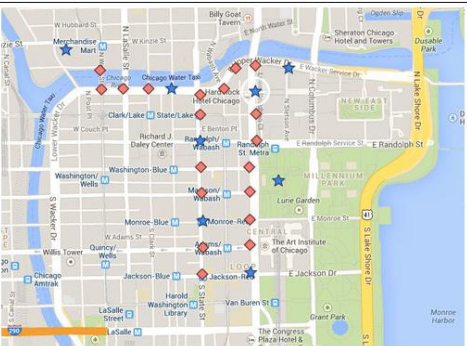
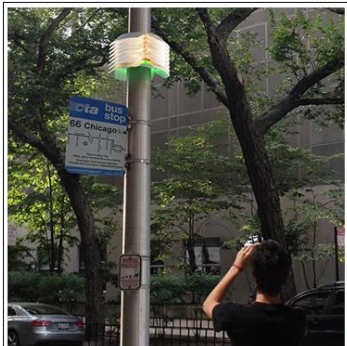
- Solar power; small-scale combined heat and power systems.
- **Electricity production** through use of **ducted wind turbines**.

# Integrated Energy Systems (Proposed)

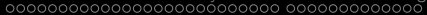


# Smart Cities: Urban Sensing in Chicago

**Array of Things, Chicago (EOL 2022).** Modular sensor boxes will collect real-time data on the city's environment, infrastructure and activity.



**Basic Questions.** How is the city used? What is going on?



# Smart Cities: Urban Sensing in Chicago

## What Data is Collected?

The nodes will initially measure temperature, barometric pressure, light, vibration, carbon monoxide, nitrogen dioxide, sulfur dioxide, ozone, ambient sound intensity, pedestrian and vehicle traffic, and surface temperature. Continued research and development will help create sensors to monitor other urban factors of interest such as flooding and standing water, precipitation, wind, and pollutants.

Array of Things is interested in monitoring the city's environment and activity, not individuals. In fact, the technology and policy have been designed to specifically avoid any potential collection of data about individuals, so privacy protection is built into the design of the sensors and into the operating policies. Array of Things will not collect any personal or private information.



# Smart Cities: Urban Sensing in Chicago

## What does an AoT "node" Measure?



### Environment

- Ambient, UV, IR light
- Visibility
- Magnetic Field
- Vibration
- Sound pressure
- Temperature
- Relative humidity
- Barometric pressure

### Air Quality

- PM 1, 2.5, 10, 40
- Carbon monoxide
- Ozone
- Sulfur dioxide
- Nitrogen dioxide
- Hydrogen sulfide
- Total reducing gases
- Total oxidizing gases

### Edge Computing: Remotely programmable AI

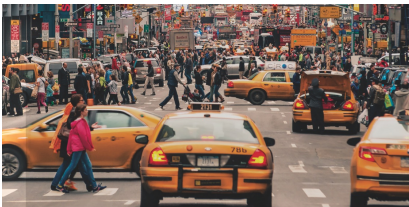
Computer Vision: Flooding, traffic flow, safety (bike helmet use, pedestrian patterns...), use patterns of public spaces, cloud cover

Computer Audio: Noise components, sound events



# SONYC: Sounds of New York City

**SONYC.** A system for monitoring, analysis and mitigation of urban noise pollution.



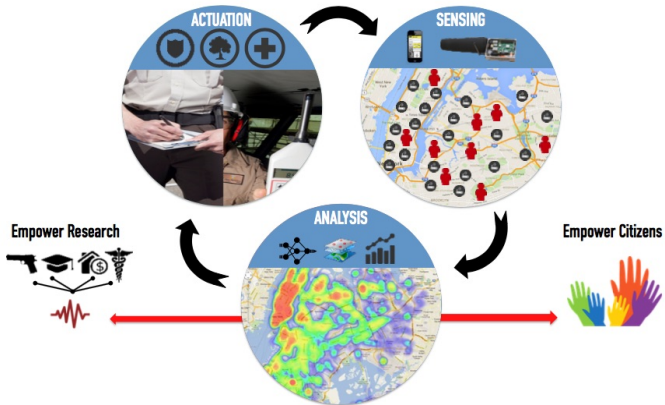
**Motivation.** Over 70 million people in US are exposed to noise levels beyond the limit of EPA considers to be harmful.

**Short-term Problems.** Sleep disruption.

**Long-term Problems.** Hypertension, heart disease, hearing loss.

# SONYC: Sounds of New York City

**Complaints.** NYC authorities receive more than 800 noise-related complaints per day!





# Summary

# Recurring Themes and Key Points

## Recurring Themes

- Information-age systems offer enhanced functionality and better performance, but their design is more difficult than in the past.
- Physical systems and computational systems fail in completely different ways.
- **Sensor networks** will form the **eyes and ears** of complex control and information systems.
- As system complexity increases, **more and more of the functionality** will be **managed by software!**

# Key Points for Building Better Systems

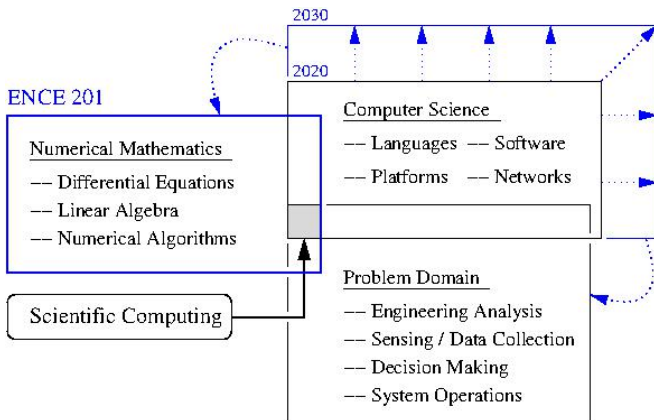
## Looking Forward

Use sensing and software to build better systems:

- Improve **situational awareness** – to understand what is actually happening a building or city?
- Connect **sensor measurements** to short- and long-term **urban needs** (e.g., decisions on a bus stop; longer term urban planning).
- Capture the **spatial**, **temporal**, and **intensity** aspects of environmental phenomena (e.g., fires, flooding) and their **impact** on natural (e.g., air quality) and **man-made systems** (e.g., transportation networks, food chains).
- **Look ahead** and **forecast future states** of the system?

# Central Role of Scientific Computing

Scientific computing lies at the **intersection** of **computer science**, **numerical mathematics**, and domain-specific **problem solving**.



# Central Role of Scientific Computing

## Computer Science and Software:

- Very fast computations.
- Mass collection of data.
- Rapidly growing importance of data sciences.
- Artificial Intelligence and Data Mining.
- Machine Learning.

## Numerical and Applied Mathematics:

- Differential Equations
- Numerical Analysis and Linear Algebra.

# Central Role of Scientific Computing

## Large-Scale Simulation:

- Improved protection of buildings from extreme environmental loadings (e.g., earthquakes, fire, tsunamis, blast).
- To understand how consequences of global warming (e.g., sea-level rise; wild fires) will impact cities.

## Improved Management of Urban Processes:

- Network systems analysis and optimization.
- New strategies for data-driven management of interdependent urban networks.
- Prevention of cascading failures.

# Computer Language for ENCE 201

**Getting Started.** We need learn to walk before we can run:

Capability	1970s	1980s	1990s
Languages	Fortran, C	MATLAB	Python, Java

## Why Python?

- Not too difficult – it's a reasonable place to start learning.
- Good support for data analysis and data analytics.
- Good support for numerical calculations.
- Provides a stepping stone to other languages.



# Appendices

# How do Physical Systems Fail?

## Physical System Concern

- Design success corresponds to notions of **enhanced performance**, **resilience** and **reliability**.
- Behavior is constrained by conservation laws (e.g., conservation of mass, conservation of momentum, conservation of energy, etc..).
- Behavior often described by families of **differential equations**.
- Behavior tends to be continuous – usually there will be **warning** of **imminent failure**.
- Behavior may not be deterministic – this aspect of physical systems leads to the need for **reliability analysis**.
- For design purposes, **uncertainties** in behavior are often **handled** through the use of **safety factors**.

# How do Software Systems Fail?

## Software System Concerns

- Design success corresponds to notions of correctness of functionality and timeliness of computation.
- Computational systems are **discrete** and **inherently logical**.  
Notions of energy conservation ...etc... and differential equations do not apply.
- Does not make sense to apply a safety factor. If a computational strategy is logically incorrect, then “saying it louder” will not fix anything.
- The main benefit of software is that **functionality can be programmed** and then **re-programmed at a later date**.
- A **small logical error** can result in a **system-wide failure**.