Carnegie Mellon University/Software Engineering Institute: Software Solutions Symposium

A PERSPECTIVE ON MILITARY SOFTWARE NEEDS

The Honorable Heidi Shyu

heidi.shyu@yahoo.com

March 21, 2017

AGENDA

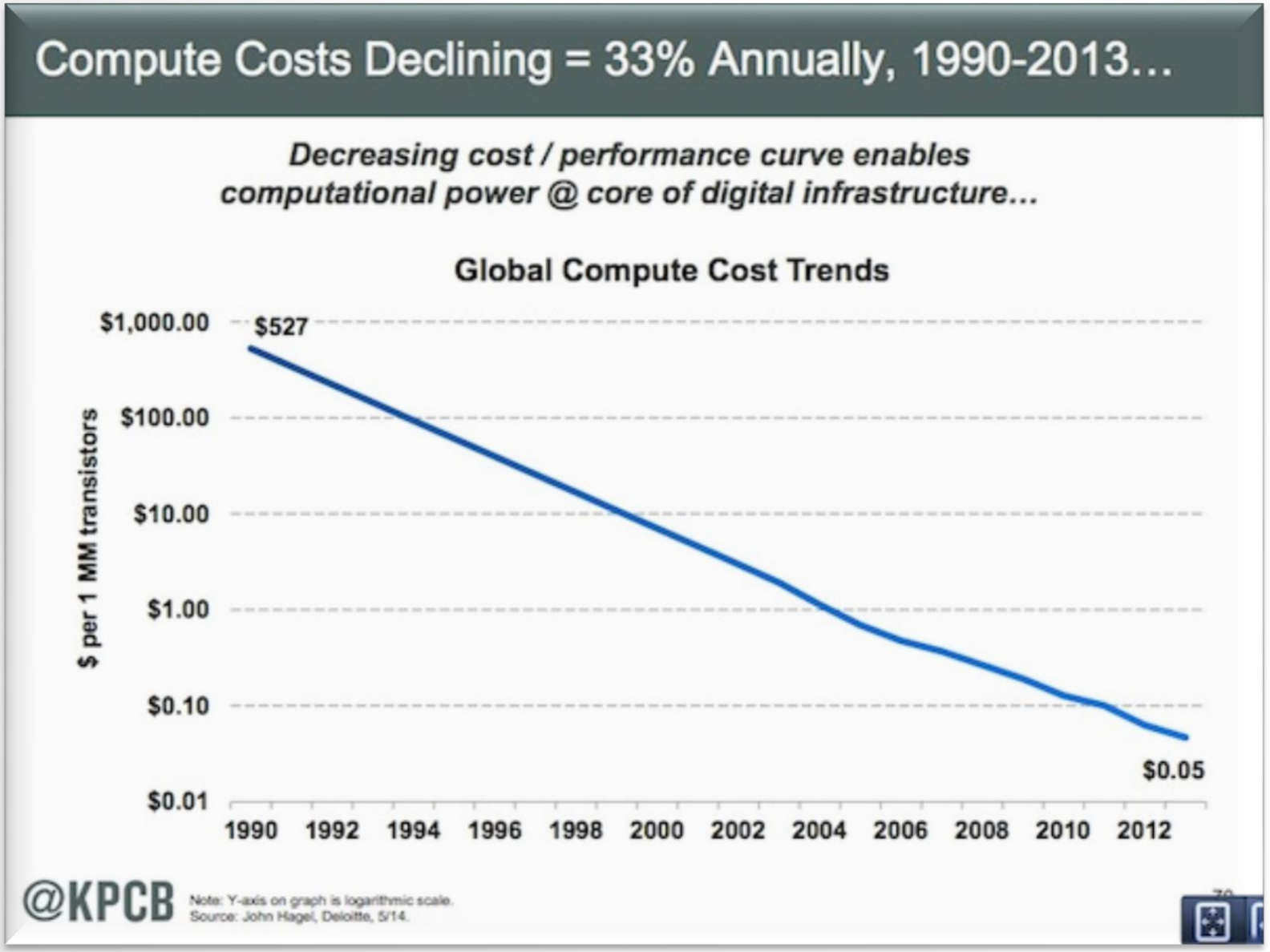
- Examples of Software Advances in 2016
- Dramatic Decrease in Cost of Computing Power
- Emerging Software Capabilities Changing the Future of Warfare
- Diverse Needs of Military Software Systems
- Software Challenges of Military Systems
- Challenges for Software Developers of Military Systems

EXAMPLES: EXCITING ADVANCES IN SOFTWARE IN 2016

- **Tesla's Autopilot** brings man with blood clot to hospital
- Jan 2016: **Google's AlphaGo** beats world champion at the game Go
- **Microsoft's AI** can now understand speech better than humans
- AI improves cancer diagnosis **– IBM Watson** detected leukemia in a woman in Japan that had been previously missed

Source: TechRepublic - The 6 most exciting AI advances of 2016

DRAMATIC DECREASE IN COMPUTE COST



EMERGING SOFTWARE CAPABILITIES ARE CHANGING THE FUTURE OF WARFARE

- **Rapid growth of synchronized autonomous systems**

Oct. 2016: 3 F/A-18 Super Hornets released 103 Perdix self-communicating drones

Summer 2016: Marine swarming UAVs demo

Electronics miniaturization combined with software advancement is creating disruptive technologies

EMERGING UNMANNED GROUND VEHICLES WILL DRAMATICALLY CHANGE THE GROUND FORCES

Photos: courtesy of Roboteam

EMERGING CAPABILITIES WILL FUNDAMENTALLY CHANGE TRAINING

Advancements in:

- Emergence of augmented reality
- Virtual reality gaming
- Machine learning advancement
- Natural language processing

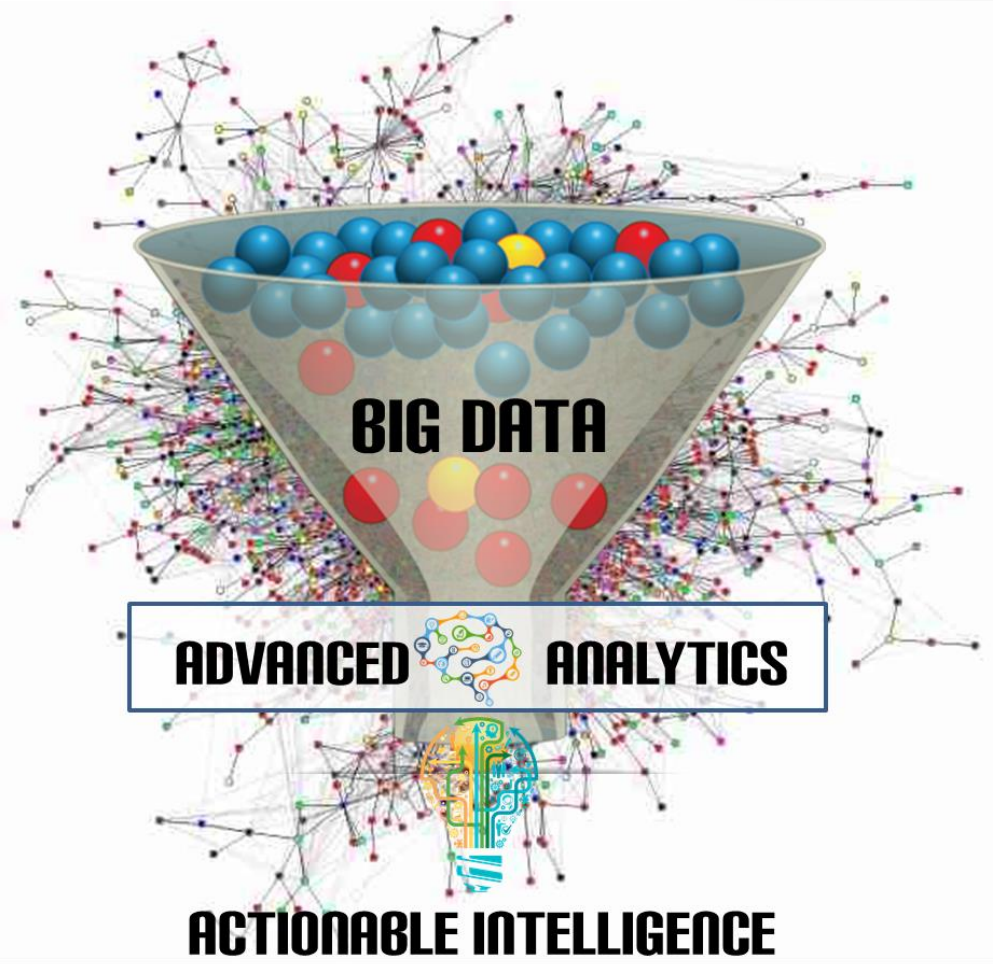
Vuzix Augmented Reality glasses

Monitor vital functions

Wireless earbud with smart trainer

BIG DATA ANALYTICS IS TRANSFORMING ISR

- **Enabling discovery of hidden patterns, unknown correlations with unprecedented speed**



Graphics from enterra solutions

CHALLENGES

- **How do we leverage the explosive growth in software for military applications?**
- **What are the software development challenges that's unique to the military?**

DIVERSE NEEDS OF MILITARY SOFTWARE SYSTEMS

- **Highly complex safety-critical mission systems:** fighters, bombers, helicopters, tanks, ships, …
- **Command & Control systems:** control satellite launch, missile launch, rocket launch, tasking UAVs, …
- **Communication systems:** operate in challenging environment & interoperate
- **Radar systems:** provide accurate all-weather real-time target detection & identification
- **Logistics systems:** tracks supply & inventory, supply chain
- **Enterprise Resource Planning (ERPs):** Military personnel & pay systems, property management, contracting …

SOFTWARE CHALLENGES OF MILITARY SYSTEMS

- **Embedded software in complex weapon systems: multiple contractors for a multitude of different systems that must interoperate in real time in challenging environment**
  - Safety critical systems
  - Interoperability at systems-of-systems level
  - Systems are bought at different times – different acquisition programs
  - Security against evolving multiple attacks (cyber, physical, RF, etc.)
  - Anti-tamper requirements
- **Extensive amount of independent operational testing is required**

COMPLEXITIES OF MILITARY PROGRAMS

- Can't buy things that you don't have a requirement for
- ~2 years to formulate, develop and approve requirements
- Have to plan 2 years in advance to get into the POM
- Vague requirements, unattainable requirements contributing to issues down-stream
- Evolving threats impact the stability of requirements
- Affordability & budget availability creates "add-on" capabilities not compatible to the architecture: EP, AT
- Budget instability result in changing requirements

AT – Anti-Tamper, EP- Electronic Protection

COMPLEXITIES OF MILITARY PROGRAMS (CONTINUED)

- To initiate a program…you need a crystal ball into development, procurement, and Operational & Sustainment cost up-front
- Government Must-do's: Market survey, Analysis-of-Alternatives (AoA), Industry Day, determine contract type, affordability, not-to-exceed budget, draft RFPs, RFP, source-selection
- DoD programs are highly risk-adverse with many stakeholders to satisfy

COMPARISON: BETWEEN COMMERCIAL VS MILITARY SOFTWARE

	Commercial Software	Military Software
Product Life Cycle	Months to a couple of years	Decades
Software Language	Evolves rapidly	Must be supported for decades due to legacy processor
Interoperability	Limited	Interoperability with legacy systems that lasts for decades
Updates	Frequent	Infrequent
Testing	In-house	In-house plus independent operational testing
Training	Current software version	Multiple configurations

CHALLENGING SOFTWARE ENVIRONMENT FOR DEVELOPERS OF MILITARY SYSTEMS

- **Architecture must last for <u>decades</u>** but software & hardware evolution is much faster…resulting in clunky patchwork
- **Weapon systems have much longer update cycle-time than technology life-cycle** (e.g. computers refreshed every 5 yrs)
- **Evolving threats outpace system updates**
- **Interoperability with legacy systems is critical**
- **Training of operators is difficult when multiple configurations exists in operation**
- **Trust in system, encryption, secure data, lineage of data**

SOFTWARE CAPABILITIES NEEDED

- **Rapid transformation of multiple legacy systems into intuitive, ease-of-use applications**
- **Rapid transformation of training materials and logistics manuals into ease-of-use apps**
- **Rapid determination of "trusted" systems**
- **Examples of needed capabilities:**
  - Real-time integration of multi-modal, multi-functional, multi-sensor capabilities to enable automated detection, discrimination, identification and targeting
  - Higher order human-machine interface enabling communication thru gesture, voice, etc.

EXAMPLES OF SOFTWARE SOLUTIONS NEEDED

- **Flexible, modular architecture that enables rapid add-on**
- **Self-checking, self-testing modular code**
- **Assurance of trusted code**
- **Intuitive, easy-to-use software applications**
- **Automated transformation of trusted assured code in higher level language to lower-level language**

QUESTIONS?

Curtesy of Sean Heritage