
# Head in the Clouds

## Tips, Tricks, and Lessons Learned from Building Ground Systems in the Cloud

Jon Erdman, Sean Kelley

02/25/2014

# Who we are

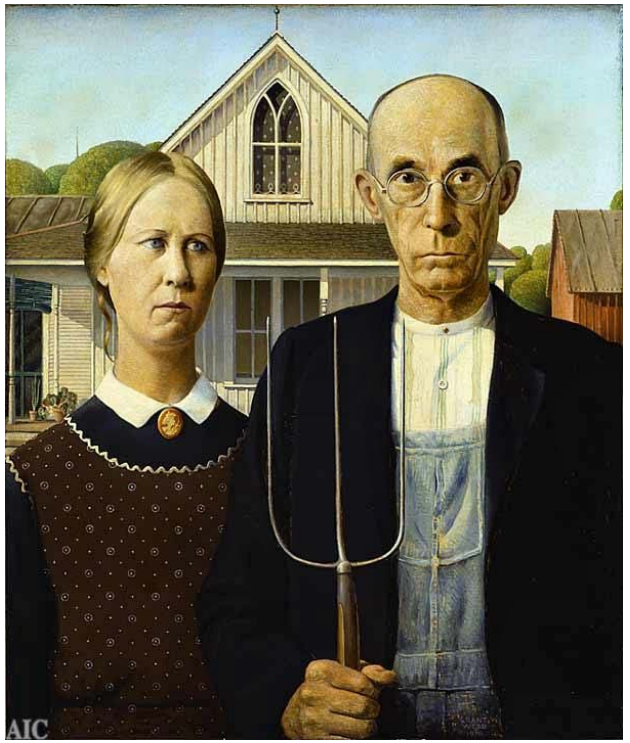
- **Jon Erdman - Chief Architect/Intelligence & Information Systems - Solers, Inc.**
- **Sean Kelley – Chief Architect/Intelligence Systems - Solers, Inc.**
- **Ground System Experience:**
  - SOA-based ground systems for NOAA focused on shared services and infrastructure
  - Ground systems for the Intelligence Community (IC) focused on large-scale distributed processing
  - Commercial ground systems to support advanced image processing and value-added products
- **Cloud Experience:**
  - Created a secure, multi-tenant private cloud implementation for the IC
    - Accredited under ICD 503
    - Supported integration of new and legacy programs into the cloud
  - Multiple prototypes of distributed processing applications in AWS

# What is the “cloud”?

- **“Cloud” in this presentation refers to an Infrastructure as a Service (IaaS) cloud unless otherwise specified**
- **What does an IaaS cloud offer?**
  - Virtualized infrastructure providing processing, networking, and storage
  - Services for monitoring and dynamic scaling of application services
  - Infrastructure-level security support
    - Secure virtualization, Private VLAN, Isolated storage, Firewall, etc.
- **Two Broad Categories**
  - Stateful (i.e. “Virtual Datacenter”)
    - Typically long-running persistent instances
  - Stateless (i.e. “Dynamic Processing Cloud”)
    - Expects a combination of long-running and short-lived instances

# What is a “Tenant”?

- **“Tenant” refers to an entity running within the cloud infrastructure, potentially consisting of many individual applications and services**
- **What does a Tenant bring?**
  - The applications!
    - Generally in the form of virtual machine images
  - Resource Requirements
    - Compute, network, storage
    - Part of the Tenant Agreement (aka Service Level Agreement)
  - Infrastructure Security Accreditation

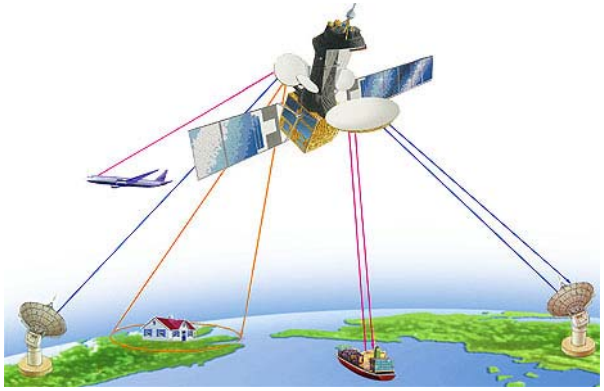


# Lessons Learned Countdown

**Lesson 5** | **Choose Appropriate Deployment Model**

# When to use the cloud

- **Satellite ground systems have many software components**
  - Tasking and requirements management
  - Command and Control (C2)
  - Data processing
  - Advanced data exploitation and analytics
  - Product dissemination



- **Each has its own unique requirements that may make it more or less suitable for a cloud environment**
- **Cloud does not have to be all-or-nothing!**
  - A proper design should be able to support portions of the system both within and outside a cloud

**Tip: Have a good understanding of your system components and requirements before making any infrastructure decisions**

SOLERS
An Employee Owned Company

# Cloud Benefits and Limitations

## Clouds are best at:

- Large-scale distributed data processing
- "Burst" processing
- Hosting web-based services and interfaces
- Making data available to a large community
- Simplifying backup and update procedures

## Clouds may struggle with:

- "Hard" real-time processing
- Fulfilling custom hardware requirements
- Moving very large amounts of data efficiently
- Standalone thick-client user interfaces

SOLERS
An Employee Owned Company

# Lessons Learned Countdown

| | |
|---|---|
| **Lesson 5** | **Choose Appropriate Deployment Model** |
| **Lesson 4** | **Understand your cloud's security model** |

SOLERS
An Employee Owned Company

# Security is key

- **Leverage IaaS provider's accreditation**
- **Take advantage of IaaS provider's security features**
  - Virtual Private Network/Virtual Private Cloud support
  - Network Address Translation, Access Control Lists, Distributed Firewalls
  - Secure tenant storage
  - Intrusion detection, monitoring, audit archiving and filtering
- **Automate configuration management**
  - Use tools such as Red Hat Satellite, Puppet, or Chef to enforce configuration of tenant virtual machines
- **Know the risks**
  - IaaS administrators could gain access to internal application data
  - IaaS providers are a tempting target for attack
    - Present a larger target and successful penetration could mean compromises across numerous programs

**Tip: There are many misconceptions about cloud security – make sure that your customer is informed and knowledgeable**

SOLERS
An Employee Owned Company

# Lessons Learned Countdown

| | |
|---|---|
| Lesson 5 | Choose Appropriate Deployment Model |
| Lesson 4 | Understand your cloud's security model |
| Lesson 3 | Architect for the cloud |

SOLERS
An Employee Owned Company

# Creating new Cloud Applications

- **New applications targeted for a cloud environment should be designed for the cloud from the start**
  - Efficient execution and communication between distributed processes requires a distributed management framework
    - Hadoop, Storm, Mesos, Hystrix, Zookeeper, etc.
  - How can distributed processing be monitored effectively?
    - Are my processes starting and stopping correctly?
    - Are there processing limits enforced by the cloud?
  - What is the COTS licensing structures for dynamic processing?
    - Example: MATLAB compiler/runtime vs parallel toolkit

**Tip: Cloud technology is evolving rapidly. Try to avoid vendor lock-in, use standards, and stay aware of emerging trends**

SOLERS
An Employee Owned Company

# Migrating Legacy Applications

- **Migrating legacy applications to a cloud requires careful consideration**
  - What is the cost to rebuild your legacy application?
    - Specific installation and configuration settings are often undocumented
    - Databases may need to be deployed differently for a virtualized environment
  - Can the Cloud Hypervisor support the legacy hardware requirements?
    - Limits may exist for maximum number of cores, storage, and/or memory on a single instance
  - Are my current COTS/GOTS/OSS solutions compatible with the cloud enviornment?
  - Stateful clouds will often be a faster & easier migration path

# Whatever can go wrong …

- **There are many cloud-specific issues that may arise during the normal course of operations**
  - Cloud failures during dynamic provisioning attempts
  - Process preemption by someone else's higher priority process
  - Network bottlenecks from other misbehaved tenants
  - Network latency between geographically distributed cloud sites
- **Extensive testing is required to ensure that the applications can handle failures**

**Tip: Design & Test for failure!**

SOLERS
An Employee Owned Company

# Lessons Learned Countdown

| | |
|---|---|
| Lesson 5 | Choose Appropriate Deployment Model |
| Lesson 4 | Understand your cloud's security model |
| Lesson 3 | Architect for the cloud |
| Lesson 2 | Prototype early and often |

SOLERS
An Employee Owned Company

# Prototype, Prototype, Prototype

- **Prototypes will help ensure that the cloud performance meets your requirements**
  - Are advertised network and data transfer rates consistently reproducible?
  - How long does provisioning new instances take?
  - Do all of my COTS/GOTS/Open Source software packages function correctly?
- **Try different system configurations to improve performance**
  - Will using a larger instance help?
  - Do different operating systems have different performance?
  - Can we be smarter about how we store and process our data to prevent excessive transfers?
    - Frameworks including Gluster/OrangeFS or Hadoop/HDFS may help

**Tip: Get to know your cloud provider – they can often help if you aren't seeing the performance that you expect**

SOLERS
An Employee Owned Company

# Lessons Learned Countdown

| | |
|---|---|
| Lesson 5 | Choose Appropriate Deployment Model |
| Lesson 4 | Understand your cloud's security model |
| Lesson 3 | Architect for the cloud |
| Lesson 2 | Prototype early and often |
| Lesson 1 | Manage Expectations |

SOLERS
An Employee Owned Company

# Bottom Line

- **Your program, or at least some portion of your program, could probably benefit from the cloud**
  - However, being forced to blindly "migrate to the cloud" may not be appropriate for all legacy programs
- **The longer a program goes without incorporating cloud into the design, the fewer the benefits and less chance of success**
  - Include cloud requirements from the beginning and scope the work appropriately (including prototypes!)
- **Define your measures of success**
  - What are the goals of using a cloud and how can those goals be met?
- **Clearly define your roles and responsibilities**
  - Infrastructure provider vs application provider
- **May not initially result in cost savings, especially in legacy systems**
  - Software may require significant changes to account for cloud limitations
  - COTS/GOTS/OSS components may need to be replaced
  - Additional expertise may be necessary if the current team is not cloud-savvy

**Tip: Open and honest communication between all stakeholders is critical to success**

SOLERS
An Employee Owned Company

# Questions?

- **Contact Info:**

Jon Erdman

jon.erdman@solers.com

Sean Kelley

sean.kelley@solers.com

SOLERS
An Employee Owned Company