Implementing an Enterprise GIS Project

Lana Tylka
Jennifer Prather
Craig Venker

July 9, 2017
Introductions

Lana Tylka

Jennifer Prather

Craig Venker
The **purpose** of this workshop is to provide you with an **understanding** of the Enterprise GIS implementation **process** and its **components**.
## Agenda

<table>
<thead>
<tr>
<th>Time</th>
<th>Session Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>8:30 – 9:00</td>
<td>Opening and Introductions</td>
</tr>
<tr>
<td>9:00 – 9:30</td>
<td>Case Study</td>
</tr>
<tr>
<td>9:30 – 10:30</td>
<td>Vision, Goals, and Objectives</td>
</tr>
<tr>
<td>10:30 – 10:45</td>
<td>Morning Break</td>
</tr>
<tr>
<td>10:45 – 11:30</td>
<td>Assess and Plan</td>
</tr>
<tr>
<td>11:30 – 1:00</td>
<td>Lunch Break</td>
</tr>
<tr>
<td>1:00 – 2:00</td>
<td>Building IOC</td>
</tr>
<tr>
<td>2:00 – 2:15</td>
<td>Afternoon Break</td>
</tr>
<tr>
<td>2:15 – 3:30</td>
<td>Building IOC</td>
</tr>
<tr>
<td>3:30 – 3:45</td>
<td>Afternoon Break</td>
</tr>
<tr>
<td>3:45 – 4:30</td>
<td>Building IOC</td>
</tr>
<tr>
<td>4:30 – 5:00</td>
<td>Evaluate, Review, and Plan Ahead</td>
</tr>
</tbody>
</table>
• Sign up sheet
• Class materials
• Questions
• Break and lunch times
• Bathrooms
• Evacuation procedures

Please!
Turn OFF cell phones and paging devices
Course Scenario

https://www.youtube.com/watch?v=3MyyiuPAqE
Assess the Following Components

Vision, Goals, Objectives
Assess and Plan
Building an IOC
Evaluate, Review, and Plan Ahead
How to Start…

1. Conduct kickoff meeting
2. Discuss similar industries
3. Assess workflows
4. Prioritize workflows
5. Choose a life cycle
6. Create a plan

Launching your Location Platform Guide: www.esri.com/LaunchGuide
Implementing the Platform
Rapid Deployment of Initial Operating Capability

Foundation Apps
Essential Information Products

Portal

“GIS Ready” Data

Desktop  Web  Device

Server  Online Content and Services

...COTS configuration for quick results
Implementing the Platform
Realization of an Integrated Enterprise System

...Reaching across the enterprise with lasting success

Extended / Custom Apps
Integration with business systems
Advanced integrated workflows
Content Production Systems Data modernization
Patterns of Implementation

Initial Operating Capability

- Foundation Apps
- Essential Information Products

Integrated Enterprise System

- Extended / Custom Apps
- Integration with business systems
- Advanced integrated workflows
- Content Production Systems
- Data modernization

“GIS Ready” Data

Server

Online Content and Services

Portal
Beck’s Platform Vision

- Customer Engagement
- Agronomic Analysis
- Sharing and Collaboration
- In-field Management/Decision Making
- Integrated Imagery Management
- Customer-driven Reporting

- ArcGIS Server
- Image Server
- 3rd Party Data Integration

Beck’s Platform Strategy
Exercise <Your Agency Name> Platform Vision
GIS is really about the data

What already exists? What’s next?

Who are the potential end-users of your data?

What are your data retention policies?

How does growth of the project affect your data?
Geodata Management Strategies

- Separate production from publication environments!
- Make Data Owners responsible for (and proud of) their data.
- Structure data to optimize for your information products
- Consider your deployment environment (Web? Desktop? Device?)

ArcGIS for Server

Data Creation Workflows

Versions Replicas

Production Geodatabase

Tuned for Editing

Tuned for Services

Online Content and Services
Geodata Management Best Practices

- Design and group data according to maintenance and security policies, not the application using it
- Feature datasets are for topologies and networks, not to make data look pretty in ArcCatalog
- Don’t overthink your QA and versioning to avoid paralysis by analysis
- Consider a default-open strategy
- Don’t recreate or re-host data from external providers if possible
- Host heavily-used data and services in a cloud environment to reduce performance impacts
Base your Infrastructure Decisions on Facts

- Who are your users?
- How much usage will there be?
- What are the existing limitations of your systems?
- Are you ready for/interested in the cloud?
- What happens in the future?
Things to Think About

- Cloud Hosting vs. On-Premises
- Security: Access, Authentication, Auditing, Standards
- Network Access
- Reliability
- Mobile Device Access
- Contractors, 3rd Party and Public
- Existing users, apps, infrastructure
- Databases and storage types
- Physical vs. Virtualization
Using the Cloud vs. “Your Own” Infrastructure

- Cloud Deployments
  - Reduced capital expenditure
  - Reduced maintenance
  - Rapidly scalable
  - Many ready to use resources
  - Many deployment options
  - Trend is to the Cloud

- Managing your Own Environment
  - More control, but also more management
  - Security: Inside firewall (but the cloud is inside a firewall too)
  - Network: LAN is faster than WAN
  - Reliability: Not affected by Internet outages
Cloud Deployment Models

Source: trust.arcgis.com

Implementing an Enterprise GIS Project 2017
• Security
  - Confidence that your customer data and organizational data is safe

• Authentication
  - The GIS needs to ‘know’ who you are, and trust that you are who you say you are

• Authorization
  - Separation of roles and responsibilities
  - Least Privileges Model
Who are your Users?

- Internal Users
  - Usually your first tier of support
  - Everything ‘works’ in the office
- 3rd Parties
  - Contractors, outside consultants, neighboring organizations
- External users
  - How does the public interface with your GIS? How do they make use of your data and services (if at all)?
• Where is your data now?
  - In an enterprise geodatabase?
  - On a highly-available file share?
  - In an existing business system?
  - On someone’s laptop?
  - External Hard Drives?
  - DVDs?

• Priorities
  - Quality
  - Availability
  - Redundancy
• What data is needed to support the key business needs
• What are easy GIS data sets that can be stood up quickly or are already accessible
• Who is responsible for/owns which GIS layers?
• What are the data exchange protocols?
• What is the format of the GIS data?
• Are there any GIS data standards in your organization?
• What security is associated to your data?
• Web Services and Map Services are essential to Web GIS and modern mapping workflows
• Design services around data, not around application: reusable, flexible, optimized
• Consider services the primary data access method
• Make data application-agnostic and easily understood
• Be ready to leverage outside data and services in your information products
System Architecture:

Leverage best practices: ex. Environment Isolation

https://www.esri.com/~/media/Files/Pdfs/products/arcgis-platform/architecting-the-arcgis-platform
Exercise <Your Agency Name> Platform Vision

Update with what you have learned…
Assess the Following Components

- Vision, Goals, Objectives
- Assess and Plan
- Building an IOC
- Evaluate, Review, and Plan Ahead
Assess and Plan

- Supporting Team
- Project Lifecycle
- Contracts
- Risks
Capacity of Human Resources, organization and staff

Historic usage and expected future growth

Monitoring and Testing

SLAs and the impact of over-capacity
• Does your organization have enterprise software implementation experience?
• Does your organization have GIS implementation experience?
• Does your organization have the necessary talent with knowledge to meet the roles and responsibilities?
• Hardware is cheap, experience is expensive
• Servers are easy, network is hard
• Beware of key components that you don’t control (NAS, firewalls, web accelerators…)
• Understand your vendor relationships
• Trust anecdotal information but verify
• Who will be impacted by Enterprise GIS changes?
• How will they be impacted and how will they respond?
• What mitigation options are available for negative impacts?
• Plan your strategies carefully & engage
• An Agile approach can be a massive transition
• An Agile approach can yield earlier successes
• What roles are needed to implement the Enterprise GIS?
• What would be their responsibilities?
• Identify in-house or contractor resources to potentially support the project as overseers or implementers
• Document roles, their responsibilities and candidates
• List your assumptions
• List roles, responsibilities, potential resources and gaps
• List HW, SW and implementation gaps
• Identify mitigation options

Known Gaps Can Drive Requirements
• Test your GIS for performance (More this afternoon)
  - So you know when it’s out of compliance

• Monitor system metrics and availability
  - So you know about problems before your customers do

• Create 1/3/5 year plans
  - To ease staff augmentation, hardware and software acquisition

• Stay current on software versions
  - Operating System, ArcGIS Platform, Database instances, 3rd party applications

Known Gaps Can Drive Requirements
Assess and Plan

- Supporting Team
- Project Lifecycle
- Contracts
- Risks
Build for Value

Requirements evolve over time

Standish Group Study Reported at XP2002 by Jim Johnson, Chairman

Implementing an Enterprise GIS Project 2017
How Requirements, Technology Affect your Choice of Project Lifecycle

Waterfall

Agile

Certainty

Disaster

Implementing an Enterprise GIS Project 2017
Project Lifecycle Types

Waterfall
- Sequential Steps
- Single Release

Staged
- Upfront requirements, design
- Prototypes
- Incremental deliveries

Agile/Scrum
- Just in time requirements, design
- Short duration sprints
- Continued releasable product
Agile Manifesto

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

**Individuals and interactions** over processes and tools
**Working software** over comprehensive documentation
**Customer collaboration** over contract negotiation
**Responding to change** over following a plan

That is, while there is value in the items on the right, we value the items on the left more.
Agile Development Build Cycle

Define, Prioritize, Plan

Product Backlog
- Workflows
- Maps
- Apps

Sprint Plan
- Priorities
- Tasks

Sprint Backlog

Build

Daily Stand-up

Design

Develop

Test

Deploy

Sprint

Use

Review

Potentially Shippable Product Increment
An Agile Sprint Lifecycle

Product Backlog

1
2
3
4
5
6
7

Sprint Backlog

1
2
3

Sprint Cycle <30 Days

Daily Review

Tasks

Working Increment
# When Do These Models Work Best?

<table>
<thead>
<tr>
<th>Scope, Technology, Contract</th>
<th>Waterfall</th>
<th>Staged Delivery</th>
<th>Agile</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Clear requirements</td>
<td>• Several applications</td>
<td>• Flexible scope, deliverables</td>
<td></td>
</tr>
<tr>
<td>• Fixed deliverables</td>
<td>• Fixed deliverables</td>
<td>• One or several applications</td>
<td></td>
</tr>
<tr>
<td>• Single application</td>
<td>• Single application</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Size, Duration</th>
<th>Waterfall</th>
<th>Staged Delivery</th>
<th>Agile</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Small size, short duration project</td>
<td>• Small size, short duration project</td>
<td>• Medium or large size, mid to long duration</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Medium or large size, mid to long duration</td>
<td>• Any size or duration project</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Capacity, Capabilities, Environment</th>
<th>Waterfall</th>
<th>Staged Delivery</th>
<th>Agile</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Limited capacity, resources, and environment</td>
<td>• Limited capacity, resources, and environment</td>
<td>• Capacity, resources, and environment to support multiple releases</td>
<td></td>
</tr>
<tr>
<td>• Frequent turnover on project team</td>
<td>• Frequent turnover on project team</td>
<td>• Capacity, resources, and environment to support multiple releases</td>
<td>• Customer EXPECTS collaboration</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Capacity, resources, and environment to support multiple releases</td>
<td>• Stable, experienced project team</td>
</tr>
</tbody>
</table>
Exercise: Waterfall vs Agile
<table>
<thead>
<tr>
<th></th>
<th>Waterfall</th>
<th>Agile</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scope</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Planning</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Priorities</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Customer Participation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Testing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Development Team</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Project Management</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Waterfall</td>
<td>Agile</td>
</tr>
<tr>
<td>------------------</td>
<td>---------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Scope</strong></td>
<td>Fixed and well-defined</td>
<td>Flexible, continuous refining</td>
</tr>
<tr>
<td><strong>Planning</strong></td>
<td>Extensive at the beginning allowing for faster development</td>
<td>Continuous and MUST be part of each iteration</td>
</tr>
<tr>
<td><strong>Priorities</strong></td>
<td>Set at the beginning</td>
<td>Constantly changing</td>
</tr>
<tr>
<td>**Customer</td>
<td>Can be minimal</td>
<td>Must be substantial</td>
</tr>
<tr>
<td><strong>Participation</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Testing</strong></td>
<td>Not until the end (more bugs)</td>
<td>Constant (helps reduce bugs)</td>
</tr>
<tr>
<td>**Development</td>
<td>Replacing resources is easier due to robust documentation</td>
<td>Must have strong skillsets and be adaptable</td>
</tr>
<tr>
<td><strong>Team</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>**Project</td>
<td>Documentation and planning make managing “easier”</td>
<td>Strong PM skills required for scope management</td>
</tr>
<tr>
<td><strong>Management</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Key Takeaways

1. All software development projects complete similar tasks.
2. Select a life cycle that fits your project (one size does not fit all).
3. Evaluate – scope, organizational skills, project size, duration.
Assess and Plan

- Supporting Team
- Project Lifecycle
- Contracts
- Project Planning
Assess and Plan

Contract Types

Writing a Proposal

Estimating
• **Firm Fixed Price (FFP)**
  - Contractors will bear the risk

• **Time & Materials (T&M)**
  - You bear all the risk

<table>
<thead>
<tr>
<th>Contract questions</th>
<th>FFP</th>
<th>T&amp;M</th>
</tr>
</thead>
<tbody>
<tr>
<td>Do you know what you want?</td>
<td>A lot</td>
<td>Little</td>
</tr>
<tr>
<td>How Much is Managing Risk Worth to You?</td>
<td>A lot</td>
<td>Little</td>
</tr>
<tr>
<td>How Important is Flexibility to You?</td>
<td>Little</td>
<td>A lot</td>
</tr>
</tbody>
</table>
• Are you going to engage Contractors?
• If so what would be the Contract type
  - Firm Fixed Price (FFP); or
  - Time and Material (T&M)
What should you include in an RFP?

Not Just Technical Details!

Current State and Needs
Implementation Considerations
Administrative Details
Technical Requirements

Good RFPs Strike a Balance
RFP Process (from the Vendor perspective!)

THE REQUEST WE GOT FOR A QUOTE IS VAGUE, AND THE DEADLINE FOR OUR RESPONSE IS TOMORROW.

IF I ASK FOR CLARITY, WE'LL MISS THE DEADLINE. IF I DON'T, OUR BID WILL EITHER BE BELOW OUR COST OR TOO HIGH TO WIN.

WHICH PATH OF CERTAIN FAILURE DO YOU PREFER?

I LIKE THE ONE THAT MAKES YOU WORK THE HARDEST.
So What Should Be In Your Enterprise Plan?

- Vision statement
- Key workflows
- Approach – life cycle, budget and staffing
- Key things you need from the rest of the organization
- Schedule – overview of key milestone target dates
- Assumptions, dependencies and risks
- Appendices – WBS, roles, key points of contact, etc.
Assess and Plan

Contract Types

Writing a Proposal

Estimating

Implementing an Enterprise GIS Project 2017
How to turn information into an initial plan

- Identify Initial operating capability
- Identify low hanging fruit – quick wins
- Identify high priorities
- Identify time frames
- Identify resources
- Create Work Breakdown Structure (WBS)
Proposal Considerations

- Compliance
- Ask the right questions early
- Have a process and an owner
- Focus on the Executive Summary

- Put on the advertising hat
- Dedicate a team
- Do your homework

Focus on the Key Areas…

• Executive Summary
• Solution Overview
• Scope of Work
• Schedule
Assess and Plan

- Contract Types
- Writing a Proposal
- Estimating
**Story point** is an arbitrary measure used by Scrum teams. This is used to measure the effort required to implement a story. In simple terms, it's a number that tells the team how hard the story is. Hard could be related to complexity, unknowns, and effort. In most cases, a story point range is 1, 2, 3, 5, 8, 13, 21, 34, 45.
## Estimating Sheet

<table>
<thead>
<tr>
<th>Activity</th>
<th>Scrum Master</th>
<th>Product Owner</th>
<th>Developer</th>
<th>Analyst</th>
<th>System Admin</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Architecture</td>
<td>16</td>
<td>16</td>
<td>0</td>
<td>16</td>
<td>120</td>
<td>168</td>
</tr>
<tr>
<td>Geodatabase Design</td>
<td>24</td>
<td>24</td>
<td>0</td>
<td>184</td>
<td>40</td>
<td>272</td>
</tr>
<tr>
<td>Widget 1</td>
<td>24</td>
<td>24</td>
<td>176</td>
<td>48</td>
<td>0</td>
<td>272</td>
</tr>
<tr>
<td>Widget 2</td>
<td>20</td>
<td>20</td>
<td>240</td>
<td>80</td>
<td>0</td>
<td>360</td>
</tr>
<tr>
<td>Application Hardening</td>
<td>40</td>
<td>16</td>
<td>84</td>
<td>24</td>
<td>4</td>
<td>168</td>
</tr>
</tbody>
</table>
Implementing an Enterprise GIS Project 2017
Assess and Plan

- Supporting Team
- Project Lifecycle
- Contracts
- Project Planning
Why is a Project Plan Critical?

• Defines the Project Execution Roadmap
  - What you are building
  - Key Timing, sequence of events
  - Resources
  - Strategy and Communications

• Defines when you are done
  - Quality expectations
  - Acceptance Criteria
• Group and list priorities:
  - Key workflows
  - Key data
  - Key IT needs
  - Staffing needs
  - Schedule

• Rank importance
• High level functional requirements
  - Business processes
  - Output
  - Communication
• Non-Functional
  - Technology
  - Capacity
  - Maintenance and support
  - Policies and procedures
### Project size and complexity

- Project complexity increases with size
- Large projects will require more dedicated roles

### Conflict of interest, separation of duties

- If you are the Developer, should you be testing?
- If you are the Project Manager, should you be the Release Manager?

### Skills and expertise required for the task

- Do you have domain knowledge to do requirements?
- Do you have the qualifications needed for testing or release management?

### Best use of time

- Should a senior manager test or focus on other things?
• Work Breakdown Structure (WBS) is aligned with the project plan to support project tracking and monitoring
• WBS is decomposing the project into phases, deliverables and work packages
  – Manageable work elements
• Derived from the Statement of Work (SOW)
• All branches in the WBS should be well defined otherwise they represents a risk in scope
A Good WBS...

Accuracy of charging
Too few
Ease of reporting

Ease of charging
Too many
Complexity
Start Here

Develop Baseline Schedule

Build a schedule built on WBS

Add detailed activities

Define dependencies between activities

Assign durations

Assign resources

Identify Milestones & deliverables

Implementing an Enterprise GIS Project 2017
Validate estimates by considering relative level of effort.

Waterfall Approach

Tempo of an Enterprise GIS Project

- Initiation, Planning & Requirements Phase
- Development Phase
- Design Phase
- Deployment Phase

Legend:
- Proj Planning & Mgt
- Requirements
- Design
- Development
- QA/Release/Deploy
- Release Management

Hours

Months

Implementing an Enterprise GIS Project 2017
Agile Approach

Chart Title

- Plan
- Requirements
- Design
- Develop
- Test
- Deploy

Implementing an Enterprise GIS Project 2017
Implementing an Enterprise GIS Project 2017

### Estimated Accrual as of Feb 28, 2017

- **Planned**: $120,000.00
- **Actual**: $40,000.00
- **Accrual**: $80,000.00
- **Spent**: $20,000.00

### Task Progress

<table>
<thead>
<tr>
<th>Task</th>
<th>Proposed Start</th>
<th>Proposed Finish</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setup EMCS environment</td>
<td>4/10/17</td>
<td>4/14/17</td>
<td>Complete</td>
</tr>
<tr>
<td>Configure Portal</td>
<td>4/14/17</td>
<td>4/28/17</td>
<td>Complete</td>
</tr>
<tr>
<td>Configure Surveys</td>
<td>4/14/17</td>
<td>4/28/17</td>
<td>Complete</td>
</tr>
<tr>
<td>Survey tie-in</td>
<td>4/24/17</td>
<td>5/19/17</td>
<td>Complete</td>
</tr>
<tr>
<td>Onsite visit</td>
<td>5/9/17</td>
<td>5/11/17</td>
<td>Complete</td>
</tr>
<tr>
<td>Finishing touches</td>
<td>5/15/17</td>
<td>5/19/17</td>
<td>Complete</td>
</tr>
<tr>
<td>Pilot</td>
<td>After 5/22</td>
<td></td>
<td>Complete</td>
</tr>
</tbody>
</table>
• Watch the video

• Think about:
  - What went wrong?
  - What would you do differently?

• There will be a test!

https://www.youtube.com/watch?v=Wac3aGn5twc
Video Review

• What went wrong?

  - Execution
    - No business rhythm
    - Poor communication
    - Over allocated resources

  - Control
    - Changing scope
    - No schedule
    - Undefined budget
    - No risk management

Let’s look at this more closely…
What happens when there is no risk management?
What is the purpose of risk management?

Identify

Reduce

Purpose of Risk Management
What are sources of risk?

Examples:
- Impossible schedule
- Loss of funding
- Natural disasters - e.g. Hurricane Katrina 2005

Take a piece of paper
Write down one idea of your own
Take five minutes
What are sources of risk?

No project management
  Too many projects
Not enough resources
  Too many changes

Change in government or regulation
  Legal issue
Currency rate fluctuations
  Technology changes

And many more…
Assess the Following Components

- Vision, Goals, Objectives
- Assess and Plan
- Building an IOC
- Evaluate, Review, and Plan Ahead
Lunch Break
11:30am - 1:00pm
Assess the Following Components

- Vision, Goals, Objectives
- Assess and Plan
- Building an IOC
- Evaluate, Review, and Plan Ahead
Scrum Sprint Cycle

Product Owner

Sprint Planning

Sprint Backlog

The team

Daily Scrum

Product Owner

Scrum Master

2 - 4 Week Sprint

Product Backlog

Stakeholders

Potentially Shippable Product Increment

Retrospective

Implementing an Enterprise GIS Project 2017
**KanBan Approach (Still Agile, just not Scrum)**

- No defined iterations
- No defined roles
- Direct communication with customer
- Limit your work-in-progress
- Visualize your work
- Ever-changing backlog with on-the-fly prioritization

<table>
<thead>
<tr>
<th></th>
<th>Pending</th>
<th>Analysis</th>
<th>Development</th>
<th>Test</th>
<th>Deploy</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>Doing</td>
<td>Done</td>
<td>Doing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td>Doing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
User Stories, Use Cases, Requirements…

How do we relate them?

User Stories

- Narrative texts that describe an interaction of the user and the system, focusing on the value

Use Cases

- Criteria to which the system or business must adhere

Requirements

- Series of interactions by the user with the system and the response of the system

Conceptual level

Functional requirement

Technical requirement
Types of Requirements

Business, Functional/Technical, COTS

Business Level
- Streamline processes
- Reduce costs
- Increase productivity
- Improve communication between user groups

Functional Level
- Enhance capabilities to accomplish work tasks
- Store metadata to support historical metrics
- Use location information to harness the power of GIS

Performance/Quality Level
- Must be faster than the old system
- Must be on software that is still supported
- Must be built with future growth allowable

Identifying opportunities of improvement
A good user story uses the “INVEST” model:

**Independent.** Reduced dependencies = easier to plan

**Negotiable.** Details added via collaboration

**Valuable.** Provides value to the customer

**Estimate-able.** Too big or too vague = not estimate-able

**Small.** Can be done in less than a week

**Testable.** Good acceptance criteria
As a [role], I can [feature] so that [benefit]
As a field representative, I want to collect information offline so that data can be collected in remote locations.
Use stories to facilitate a **conversation** with the team and with the users…

Hey! Something to look at, talk about, comment on, react to…
Progressively decompose your stories…

As a real estate manager, I need to interact with a map to explore new locations.

As an analyst, I need the ability to create a map with pop-ups to build easy-use-maps for management.

As a real estate manager, I need to be able to see a pie chart of surrounding demographics so that I can match consumer demand with product.

As a real estate manager, I need interactive map capability on my iPad so that I can travel minimalistic during site visits.

As a real estate manager, I need to receive information by clicking on potential sites to better understand the property.
Consider grouping user stories into themes
Some things are not about a [user]

As a user interface, I need to look like the following image so that I can be intuitive to use.
As a Farmer, I need to be able to overlay existing data on top of my basemap in order to see different data while working in the Field.

Use **paper cards**

Write out **concise** user stories

Define **acceptance criteria** for each user story
Keep your stories visible
Watch out for the ‘Gotchas’

Things to avoid

- Avoid long lists of acceptance criteria on a single user story
- Prepare for conflicting requirements
- Avoid requirements that are ambiguous
- Avoid requirements that describe HOW
- Requirements must have a “reason”
- Avoid moving forward on development until after the customer has reviewed the design
- Don’t forget to prioritize
Exercise: Creating User Stories
Managing Software Development in the Agile World

Development Team
- Developers
- GIS Database Specialists
- Testers

Daily Builds
Check In Work, Unit Test & Update Status

Work Items
- Daily Scrum/Stand-up Meeting
- Reporting

Product Backlog

Sprint Plan

Scrum Master/Technical Lead

Development Artifacts
- Source Code
- Configuration Files
- Wireframes
- GIS Data

Release Testing

…a Formal Process
…but unique to each team
Example: Agile Daily Stand Up Meeting for Visibility

15 Minutes, 3 Important Questions

- What did you complete yesterday?
- What have you planned for today?
- Are you facing any obstacles?

Defined | In Progress | Completed | Accepted

https://www.rallydev.com/
Implementing an Enterprise GIS Project 2017
Sprint Backlog
Using Agile in a Consulting Project
Using Agile in a Consulting Project
Using Agile in a Consulting Project

Method

Waterfall

Agile

Time

Implementing an Enterprise GIS Project 2017
Using Agile in a Consulting Project

Waterfall

Agile

Time
Using Agile in a Consulting Project

Method

Waterfall

Agile

Time
Using Agile in a Consulting Project

Waterfall

Agile

Time

Final Release

Implementing an Enterprise GIS Project 2017
Managing Resources

Plan A
Your Project

Plan B
Sprint
100%
50%
75%
100%
75%
50%

Plan Z
Sprint
50%
100%
50%
50%
75%
100%
75%
50%
Tools
Tool to assist in Release/Sprint Planning

RealTimeBoard
Use Tools to Manage Requirements

Microsoft Team Foundation Server (TFS)

Rally Agile Implementation Tool

https://www.rallydev.com/
Using Trello

https://www.rallydev.com/
Using GitHub

What content goes on doc.arcgis.com?

1. Home pages and help systems for the ArcGIS Online platform & its connected apps and data. These pages are presented in the same UI as ArcGIS.com.

Note: The links below go to the content under development, not the current released version.

- AppStudio for ArcGIS
- ArcGIS.com Help
- ArcGIS for Smartphones & Tablets
- BAO
- Collector for ArcGIS
- Community Analyst
- Community Maps

Home

Marka Vertanen edited this page 21 days ago. Add revisions.

This wiki is about the http://doc.arcgis.com/... sub-domain. It includes info on how content on this sub-domain is produced and edited.

Implementing an Enterprise GIS Project 2017
Making a Decision

<table>
<thead>
<tr>
<th>Project Considerations</th>
<th>Trello</th>
<th>GitHub</th>
<th>TFS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Requirements are Proprietary</td>
<td>⚫</td>
<td>⚫</td>
<td>⚫</td>
</tr>
<tr>
<td>Mobile App</td>
<td>⚫</td>
<td>⚫</td>
<td>⚫</td>
</tr>
<tr>
<td>Easy to setup</td>
<td>⚫</td>
<td>⚫</td>
<td>⚫</td>
</tr>
<tr>
<td>Estimation tools</td>
<td>⚫</td>
<td>⚫</td>
<td>⚫</td>
</tr>
<tr>
<td>Scheduling tools</td>
<td>⚫</td>
<td>⚫</td>
<td>⚫</td>
</tr>
<tr>
<td>Automated Burndown chart</td>
<td>⚫</td>
<td>⚫</td>
<td>⚫</td>
</tr>
<tr>
<td>Easily integrated with Visual Studio for Code Repository</td>
<td>⚫</td>
<td>⚫</td>
<td>⚫</td>
</tr>
<tr>
<td>Capacity Planning</td>
<td>⚫</td>
<td>⚫</td>
<td>⚫</td>
</tr>
<tr>
<td>Exports to MPP and Excel</td>
<td>⚫</td>
<td>⚫</td>
<td>⚫</td>
</tr>
</tbody>
</table>
Avoiding Design Nightmares

https://www.youtube.com/watch?v=BKorP55Aqvg
Design

- Workflows
  - Storyboards
  - Process description
  - GUI
  - Application

- System
  - Architecture
  - Security

- Data
  - Concepts
  - Tools

- Documentation
  - Business cases
  - Data
  - System
What needs to get designed?

The Front
- User Experience
- Information Products (Maps, Reports, Models)

The Middle
- Business Logic
- Integration Points (Between components, to enterprise systems)

The Back
- Databases, content, services
- Architecture (→ Foundational Decisions)
Design Components Together

Interoperable

- Hardware
- Data
- Software
- Apps
- Governance
- Security
- Convert user stories into storyboards visually describing the workflow process
- A picture is worth more than a 100 words
- Describe user and system actions for each storyboard frame
• Available tools
  - Visio
  - Balsamiq
  - Power Point / Keynote
  - Photo software – PSP/Corel…
• Process identification
• Actions by the user
  - Inputs
  - Format
  - Limitations

• System action
  - System process description – properties
  - Calculation formula’s
  - Rules for processing
  - Data and domain values
  - Test cases with input values and results
Application Design Themes

- Immediate capabilities
- Simple, focused apps
- Traceability to requirements
- Engage the customer
- Involve developers and end users
- Focus on mitigating risk
- Prototype

Use ArcGIS Online and Resource Centers
http://www.arcgis.com/features/
http://resources.arcgis.com/en/home/
http://github.com/esri

Example Templates for Water Leak Management

- Web Isolation Trace
- Operations Dashboard for ArcGIS
- Mobile Leak Inspector
- Field Asset Inspections
- Scaffolding tools such as Yeoman
- Existing application templates published by Esri and others
- Existing styling frameworks such as Bootstrap may guide design
- Rebuild from existing templates
Design Information Products

✓ Consider what’s available already
✓ Consider your deployment platform!

Maps, reports, models

For Web/mobile apps… Design maps as Web Maps
Design Access to Content

- **Applications → Web services → Data**
  - Web/Mobile – no direct database access
  - Requires deliberate & thoughtful authoring
  - Different security model

For Web Apps use the Web GIS Information Model
Design the User Experience – How to Evolve an Interface

- Wireframes (sketch ideas and workflows, validate assumptions)
- Review Wireframes and conduct Usability Testing
- Look and Feel of the GUI
System: Application Design

Design the UX for the common user
- Avoid adding superfluous features
- Easier to add in the future than remove

More Functionality Isn’t always better

Unless the users are experts
Design

Workflows
- Storyboards
- Process description
- GUI
- Application

System
- Architecture
- Security

Data
- Concepts
- Tools

Documentation
- Business cases
- Data
- System
System Architecture: Factors

Non-functional requirements

- User Load
- User Location
- Usage Patterns
- Availability
- COTS/Apps
- Security
- Data
- Policies

Reality

- Network Capacity (WAN, Subnets, Virtualization)
- Servers (CPU, Memory, Count)
- Redundancy (Backup, Disaster Recovery)
- Security Solution (Firewalls, Authentication, Encryption etc.)
- Disk Space
- Software Stack
Architecture: Tools & Methodology

- Gathering requirements
- Design tools
- Capacity: CPU, Network, Memory
- Reporting on Performance

ArcGIS System Designer: https://www.arcgis.com/home/item.html?id=8ff490eef2794f428bde25b561226bda

ArcGIS System Monitor available through Esri Professional Services
• Engineering for High Availability adds complexity and cost
  - Load balancing
  - Duplication / redundancy
  - Adequate testing, monitoring

• How many “9’s” do you really need?
  - Evaluate costs of downtime against costs of availability

Architecture: High Availability

<table>
<thead>
<tr>
<th>9</th>
<th>9</th>
<th>9</th>
<th>9</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>35 days</td>
<td>4 days</td>
<td>9 hours</td>
<td>50 mins</td>
<td>9 mins</td>
</tr>
</tbody>
</table>

(MySQL Reference Manual)
Solutions/Best Practices

**Access Control**
- Who is allowed to access the GIS
- What is acceptable system usage

**Audit, Logging**
- Monitoring system health and activities
- Minimize performance burden on GIS

**Encryption**
- Protecting the confidentiality of your data

**Identity Management**
- Uniquely identifying / authenticating users
- Managing user roles

**Intrusion Detection**
- How to prevent threats
- How to manage vulnerabilities

- Use rules of behavior
- Site banner with acceptable use policy
- Configure tools to strike the right balance
- Use HTTPS/TLS
- Consider encrypting data at rest
- ArcGIS built in or your own Provider?
- Single factor or Multi-factor?
- What is your Password Policy?
- Use your Firewall, don’t work around it
- Train developers in security practices!
- Implement patch management

Examples – Identity Management

**Implement Password Policy**
- At least 6 characters
- Combination of letters, numbers, characters
- Cannot include parts of user name
- Expire after 60 days, cannot be re-used

<table>
<thead>
<tr>
<th>RANK</th>
<th>PASSWORD</th>
<th>CHANGE FROM 2014</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>123456</td>
<td>Unchanged</td>
</tr>
<tr>
<td>2</td>
<td>password</td>
<td>Unchanged</td>
</tr>
<tr>
<td>3</td>
<td>12345678</td>
<td>1 ↤</td>
</tr>
<tr>
<td>4</td>
<td>qwerthy</td>
<td>1 ↤</td>
</tr>
<tr>
<td>5</td>
<td>12345</td>
<td>2 ↘</td>
</tr>
<tr>
<td>6</td>
<td>123456789</td>
<td>Unchanged</td>
</tr>
<tr>
<td>7</td>
<td>football</td>
<td>3 ↤</td>
</tr>
<tr>
<td>8</td>
<td>1234</td>
<td>1 ↘</td>
</tr>
<tr>
<td>9</td>
<td>1234567</td>
<td>2 ↤</td>
</tr>
<tr>
<td>10</td>
<td>baseball</td>
<td>2 ↘</td>
</tr>
</tbody>
</table>

**Built-in or 3rd Party Identity Management**

- **GIS Administrator Managed**
  - Add, configure users
  - add roles, grant privileges
  - through web interface

  ...*With a small system*

- **IT Managed**
  - LDAP Server
  - AD Server
  - SAML 2.0 Compliant
  - Identity Provider

  ...*With a large system*
• Is your system mission critical?
• In case of a catastrophic failure
  - How much time will it take to restore (RTO)
  - How much data loss can you afford (RPO)
• Develop a plan and test it regularly
  - Roles and responsibilities
  - Succession
  - Backup strategy
  - System restoration

The lower your RPO and RTO, the more redundancy you will have to build into your system!

How many of you have a real RTO/RPO or DR Plan?

Source: Disaster Recovery Preparedness (drbenchmark.org)
Design

System
- Architecture
- Security
- Application

Workflows
- Storyboards
- Process description
- GUI

Data
- Concepts
- Tools

Documentation
- Business cases
- Data
- System
Design the GIS Database

- Identify data requirements for information products
- Collect data inventory
- Identify gaps and how to resolve them
- Create the data model

Focus on data you will use and maintain!
Design to the normal not the exception!
Data Design - Tools

Geoprocessing Models; DDL Scripts
- Repeatable maintenance process
- Initial investment to create the model

ArcGIS Diagrammer, XRay
- Schema Creation and Documentation Functions
- GDB centric

Visio, UML
- Simple; tool is well known
- Limited support for GDB features
- No schema creation or reverse engineering (any more)

Enterprise Architect, UML
- Full UML Support
- ArcGIS Extension: Import XML Workspace Exports
- Currently better for documentation than design
Exercise: List Data
Requirements
Exercise – List Data Requirements for Information Products

Example: Water Leak Management

- Water Department Asset Map
- Leak Trace
- Map with Service Requests location by status
- Map showing field crews
- Map showing customers
Design

System
• Architecture
• Security
• Application

Workflows
• Storyboards
• Process description
• GUI

Data
• Concepts
• Tools

Documentation
• Business cases
• Data
• System
How much is enough?
How Much is Enough? *(for the customer)*

- Enough to articulate how the business processes will be realized
  - Solution architecture
  - Concept of Operations (CONOPS)
  - User Experience
  - User Stories, Scenarios
  - Data Layers and Attributes
  - Maps, Reports
How Much is Enough? (for technical team)

- Enough to determine and articulate (where necessary) how the solution will be developed
  - System architecture
  - Detailed component breakdown
  - Design Specifications for apps and data
  - Services
  - Prototyping and/or design models for…
    - High risk areas
    - Integration points
    - Key points that are fail points or are non-traditional
    - Prototype ➔ Prove ➔ implement ➔ review cycle
Configure and Develop your System

- Development Principles
- Application Development
  - Tracking, Managing
  - Tools
- Data Development
  - Content development options
  - Process

https://www.youtube.com/watch?v=X5SkW7K0e3Y
• Build to your design!
  - If change is necessary, inform and explain it to the team, use change management
• COTS and configuration first, customization later or when 100% necessary
• Track development progress continuously
  - Use tools and metrics
  - Keep everything visible and review frequently
• Use existing and trusted tools for:
  - Source control - change control - issue tracking - build automation
Apps and Information Products | How to Implement Them

- Use COTS and GIS-ready data to deploy Initial Operating Capability (IOC)

1. Deploy Foundation Apps
2. Configure Apps and Workflows
3. Customize for specific needs (only if necessary)

Presenting IOCs instead of finished products is a big switch!
Development Principles

Tools are your Friends
- Work Item Management
- Source Code Repository
- Build System
- IDEs

Use Coding Standards
- Logging
- Exception handling
- Naming conventions
- Code Reviews

Enforce Unit Testing
- By developers
- Use automation
- Test design
- Test robustness of code

Configure and Develop your System

Development Principles

Application Development
• Tracking, Managing
• Tools

Data Development
• Content development options
• Process
Application Development - Tools

Source Control
- TFS
- git/GitHub
- Subversion
- Mercurial

Build
- TFS,
- JIRA
- Travis
- Ant
- Maven
- Grunt
- Less

Work Item Management, Issue / Change Tracking
- TFS,
- GitHub
- JIRA
- Rally, Trello
- Bugzilla
- OnTime
- Remedy
- Lighthouse

Consider a complete lifecycle tool (integrates source control, build, issue tracking)

IDEs
- Sublime Text
- Visual Studio
- Eclipse
- Technology Stack dependent

Development Environment
• Options:
  - Microsoft .NET
  - Java
  - Node.js
  - Also:
    - Ruby on Rails, Django, …
• JavaScript is the best bet for native-looking apps
• Python is not only for geoprocessing but for web services, hosting web apps with Flask and other cases
• Mix and match languages to suit the need and suit your team’s experience
• With web services, you can use any language / any technology stack to access the services once they exist.
• Leverage existing APIs or wrapper libraries
  - ArcREST
  - arcgis-node
  - node-geoservices
Configure and Develop Your System

Development Principles

Application Development
• Tracking, Managing
• Tools

Data Development
• Content development options
• Process
What are the Options for Developing Data Content?

**Migration – Move data between GIS Platforms**
- AutoCAD, Microstation
- to ArcGIS
- Shapefile to GDB

**Conversion – Automate new geospatial data**
- Hardcopy Maps to GDB
- Imagery to vectors

**Collection – New data from remote platforms**
- Imagery
- GPS
- Surveys

---

Appendix A Afternoon Page 29

Data Migration Procedures Document
For best results:
Let your data evolve with your Apps and information products!

Data Development

Tools
- ArcCatalog
- ArcGIS Data Interoperability Extension
- Production Mapping Data Loader
- ArcGIS Data Reviewer
- Open Source data production
- Relational Database Geometry storage

Coordinated Approach
- Incremental deliveries
- Synchronized with development

Pilot
- Test Design
- Real data for developers

Tools and Procedures
• Formal process is key
• Strive for transparency in the process
• Engage the leadership team and sponsors
• Coordinate database and application development
• Keep developers developing and testers testing
• Stay engaged with customer or project owner
• Test early, release as often as is possible and prudent.
• Code Reviews
Test your system

- Testing Process
- Test Plan
- Test Cases
- How to Test
Testing Themes

• Test the right things at the right time
• Focus on your deliverables
• Develop good test cases
  - Prove requirements are met
  - Fit the importance of the requirement
  - Reflect workflows
• Use an (approved) test plan
• Tests need to be realistic and fair

https://www.youtube.com/watch?v=mafRRWKSwu0
Testing Works Best as an Integrated Process!

Testers, Designers, Developers
Collaborate during test case development (design stage)

Developers
Develop and unit test single components

Testers
Stakeholders, End Users
Test in a separate test environment
Testing Process

- Release for testing
- Test team training
- Perform testing
- Fix bugs
- Review results and log bugs
- Testing criteria met
- Ready to client verification and validation
Sample Design Artifacts for a Leak Management Isolation Trace Tool

User Stories

As a Field Crew Member
I want to run an leak isolation trace
So I can identify affected service area, customers, and assets

Develop Design Artifacts Specific to the Sprint

Acceptance Criteria

Configure Trace Tool
Execute Trace
Save Trace

User Interface Prototype & Information Products Design

Summary Report
Valve Report
Hydrant Report
Customer Report
Trace Map

GIS Data Layers

Feature Classes

Pipes
Valves
Hydrants
Service Locations
Work Order Locations
Basemap Layers
Test your system
Typical Test Plan Outline:

- Your testing strategy including:
  - Overview of the solution
  - Objective with the testing
  - Scope of the testing
  - Sequence of testing
- Acceptance procedure and criteria
- Test environment to be used
- Test data and test area to be used

The test plan is your ‘battle plan’ for acceptance
Test your system

- Testing Process
- Test Plan
- Test Cases
- How to Test
• Test cases are procedures for verifying requirements
• Address all requirements
  – Functional
  – Non-Functional
• Good Test Cases:
  – State the purpose of the test with reference to requirement
  – State pre-conditions
  – Clearly and objectively outline the steps to verify/validate a specific outcome
  – Unambiguously list the expected result or outcome of the test determining a pass/fail of the test

Test Case Template

Appendix A
Afternoon
Pages 37 to 39

Test Case Template
Test Script

Implementing an Enterprise GIS Project 2017 172
User Stories and Testing

...Think back to the invest model... Think Traceability

User Story Template

As a [User Role]
I want to [Desired Feature]
So that [Value/benefit]

Testing Template

Given [Some Context]
[When] some action is carried out
[Then] a particular set of observable results should be obtained

Focus on
• Pre-conditions, Post-conditions
• Step-wise descriptions
• Testable outcomes

Independent
Negotiable
Valid
Estimable
Small
Testable
Example: Test Scenario for a Water Leak Isolation Trace User Story

User Stories

Isolation Trace

Locate Leak

Locate Leak on Map

Acceptance Conditions

As a Field Crew Member
I want to run a leak isolation trace starting from the leak location
So I can identify affected service areas, customers, and assets

As a field crew member I need to locate the leak on the map so I can determine the starting point of the trace

Given - User is logged in
Mobile app is open with map display and list of active work orders
Then -
1. User selects an active work order from list
2. Selected work order highlights
3. User picks “locate”
4. Map display zooms to work order location
5. Location is highlighted with marker
6. Service point nearest to the location is highlighted

• After user login, map displays based on default of user’s profile
• Active work orders display in side panel
• User can select a single work order from the list
• Clicking “locate” zooms map display to the location of selected work order and closest service point is highlighted on the map
• Clicking “cancel” clears the selection

Test Cases Template
Test Scripts/Scenarios

Appendix A
Afternoon
Pages 37-39
Test your System

Testing Process  Test Plan  Test Cases  How to Test
Acceptance Testing Process

- Your site or client site?
- How much time?
- When to train?
- What happens after the test?
- Is on-site support necessary?
- How to track and manage defects?
How Do You Evaluate Software Quality?

Testing process

Measure and analyze against acceptance criteria
- Defect metrics
- Status meetings

Anecdotal evidence from developers and end users

Implementing an Enterprise GIS Project 2017
Sample Test Manager Reports

- Are you on or off track?

**Scenario 1**

- Defect rate does not decline
- Large number of active defects at UAT
- Acceptance unlikely

**Scenario 2**

- Defect rate declining steadily over time
- Few active defects when entering UAT
- Acceptance likely
Deploy
Typical Sequence of Events

1. **Pre-UAT Training**
2. **Test**
3. **Review / Retrospective**
4. **Fix**
5. **Deploy Solution**
6. **Stand-up Production**
7. **Production Checklist**
8. **Acceptance Issued**
9. **Move to Production**
10. **Deployment**
11. **Production Staff Training**
12. **Assess**
13. **Rollout**
14. **Support**

- **Releasable Product**
Plan Before you Go….

- Confirm the customer is ready
- Communicate all Hardware/Software pre-requisites
- Define resources, tasks, assignments
- Develop a detailed checklist
- Develop a detailed schedule
Deploy: Typical Challenges

- Use an installation package
- Plan sequencing with customer

- Itemized list of components
- Detailed checklist

- Separate environments
- Move to operations in OFF hours

- Targeted training program
Tips and Techniques

- Swap staging and production
- Swap passive and active leg (for high availability)
- When upgrading, you must be able to fall back to previous configuration
- Down-time because of installation and upgrade still counts as down-time!
Training

- Build training curriculum supporting GIS business processes
  - Targeted to the audience – not generic
  - Differentiate casual from advanced users
  - Help user visualize success in the production environment

- Sequencing and timing is essential
  - COTS before custom
  - Just-in-time is best

- Continue to grow GIS knowledge and capacity
Assess the Following Components

- Vision, Goals, Objectives
- Assess and Plan
- Building an IOC
- Evaluate, Review, and Plan Ahead
Performance Monitoring, Tuning

- Monitor, evaluate
  - How is the system REALLY used?
  - Are there unanticipated bottlenecks?
  - Can performance be improved?

- Tuning can improve performance
  - RDBMS database, ArcGIS Server
  - Versioning workflows
Production Rollout / Transitioning / Operations

- Users assume ownership
- A critical stage
  - Encourage use of the system
  - Avoid operational failures
  - Avoid abandonment of the system
- Plan to provide adequate support
  - Performance monitoring, tuning
  - Resolving issues (e.g. Help Desk)
  - Review and follow-up
Version Updates

- Evaluate options
- Upgrade in test environment first
- Includes Software Patches and OS Updates
- Communicate with IT
- Investigate new tools and functionality
- Get ready for rollback
- Is it worth it?
Adopting Emerging Technologies

- Risk/Reward
- Prototype new technology
- Demonstrate functionality to yourselves
- Demonstrate functionality to customer
- Consider Integration impacts
Measuring Benefits, Path Forward

- **Operational review**
  - Are existing expectations met?
  - What are future expectations?
  - Are there new requirements?
  - What benefits were realized?
  - Are there short comings?

- **Assess the happiness of your stakeholders**

- **Make recommendations**
  - Priorities for new data, services, apps
  - Organizational considerations
  - System tuning, upgrades

- **Plan for Growth**
Implementing an Enterprise GIS Project 2017

Increase organizational enablement

- New apps, data, services
- Further integration of organizational units

Plan Ahead

Desktop

Web

Device

Portal

Server

Online Content and Services
Moving to an Integrated Enterprise System

Initial Operating Capability

- Foundation apps
- Essential Information Products

- "GIS Ready" data

Integrated Enterprise System

- Apps
  - Extended / custom apps

- Integration with business systems

- Advanced workflows

- Content production systems
  - Data modernization

Portal

Desktop

Web

Device

Server

Online Content and Services

Access & Identity

Implementing an Enterprise GIS Project 2017
Questions?