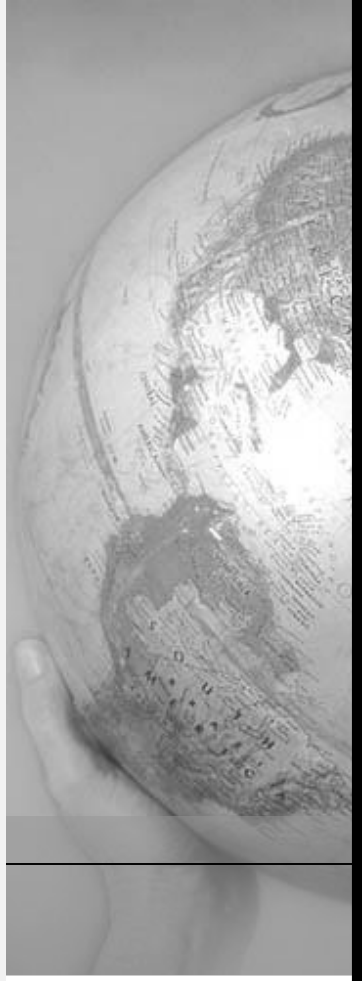
RELEASE MANAGEMENT

Release Management: A Software Delivery Methodology

Seshagiri Sriram

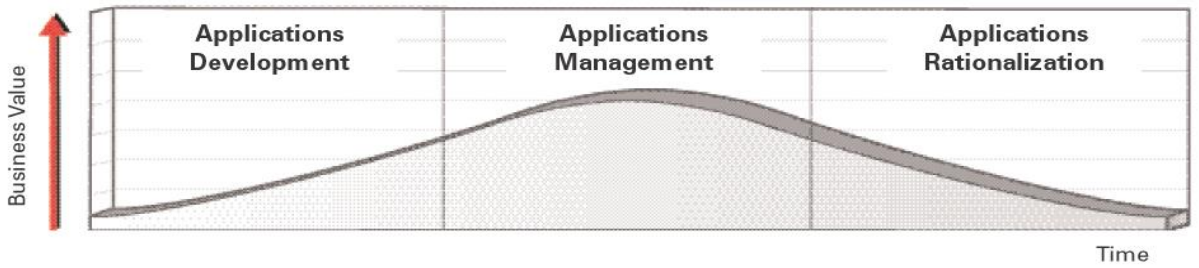
Feb 2017



“…Predictability is one of IT's most elusive goals…”

Name: Laurent Séraphin
Title: EMEA Product Director
Company: Borland

SYSTEMS LIFECYCLE



Applications Development	Applications Management	Applications Rationalization
Goals	**Goals**	**Goals**
High strategic impact	High operational efficiency	Proactively replace, retire, or upgrade applications providing declining business value
Aligned with business	Optimally supports needs of business	Continuously streamline and standardize portfolio costs and increase flexibility
On-time	Flexible	
Within budget	Scalable	
	Reliable	
	Cost-effective	
Challenges	**Challenges**	**Challenges**
No budget	Inefficiencies	Redundant applications
70% development projects fail (Standish Group study)	Escalating costs	Disparate applications
Poor project management	Poor process, management disciplines	End-of-life applications
Aligning IT investment with business strategy	Poor performance metrics	Non-core technologies
	Reactive	Inflexible systems

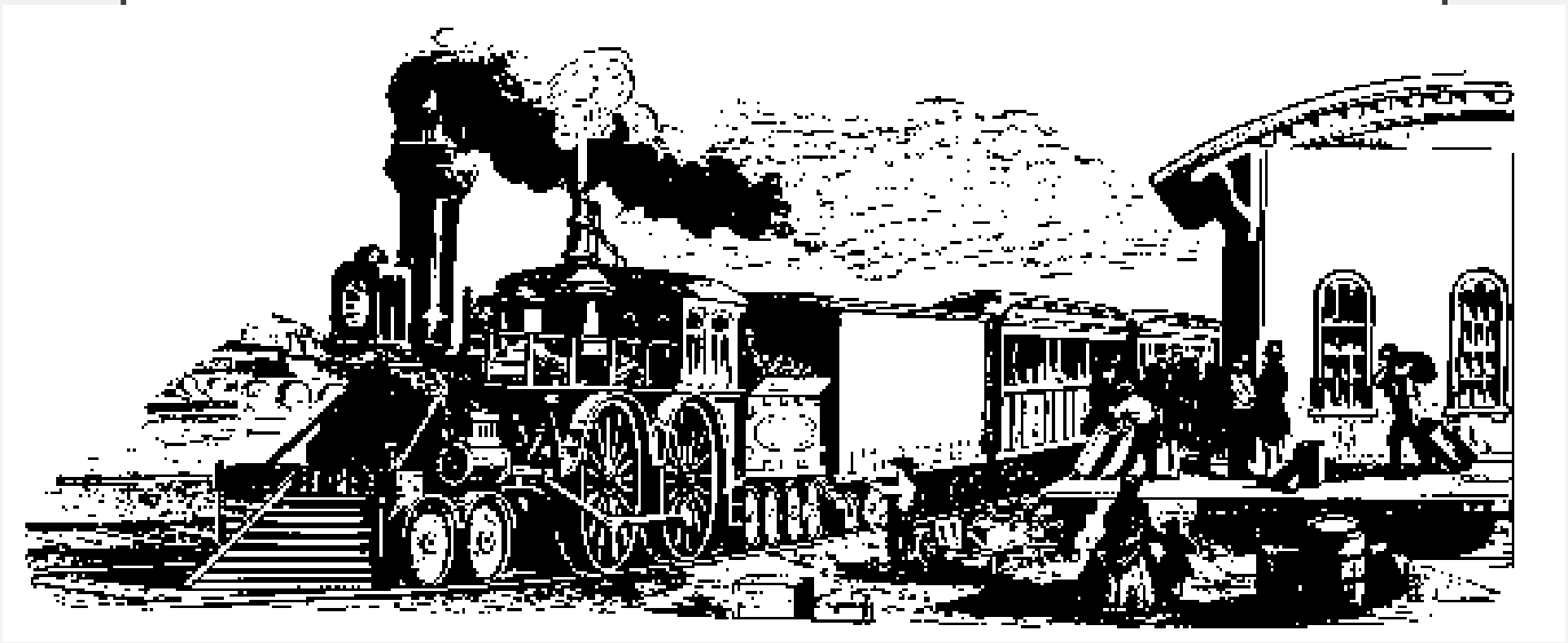
APPLICATION MANAGEMENT - TYPICAL PROBLEMS

- Inability to track and manage multiple projects affecting the same or multiple systems with multiple end dates and multiple business owners.
- Lack of predictability in delivery timeframes, costs, and support requirements.
  - IT: “You will get it next phase…” Business: “Yeah right…”
- Escalation management paradigm <squeaky wheel syndrome followed by whiplash syndrome>.
- Total breakdown in business owner confidence and trust (credibility) in IT’s ability to deliver.
- Risk to application uptime.

A SOLUTION!!!!

Release Management

A MODEL OF PREDICTABILITY



WHAT IS RELEASE MANAGEMENT?

- **Release Management (aka Release Train) can be defined as a methodology for planning and implementing an integrated set of functional components and processes in a controlled manner.**
  - **Date driven; releases are scoped in order to meet pre-specified delivery dates, the project management "Iron Triangle" balanced on the schedule apex.**
  - **Reversed planned; start with your target implementation dates and work backwards.**
  - **Mechanized; process should try to emulate a typical factory operation.**
  - **Forecasted schedules as well as functionality; force an organization to strategize and plan in advance.**
  - **Integrated and predictable; many business needs folded into one release, and everyone knows the schedule.**
  - **Uses standard system development lifecycle (SDLC) and project management methodologies (PMBOK) and best practices**

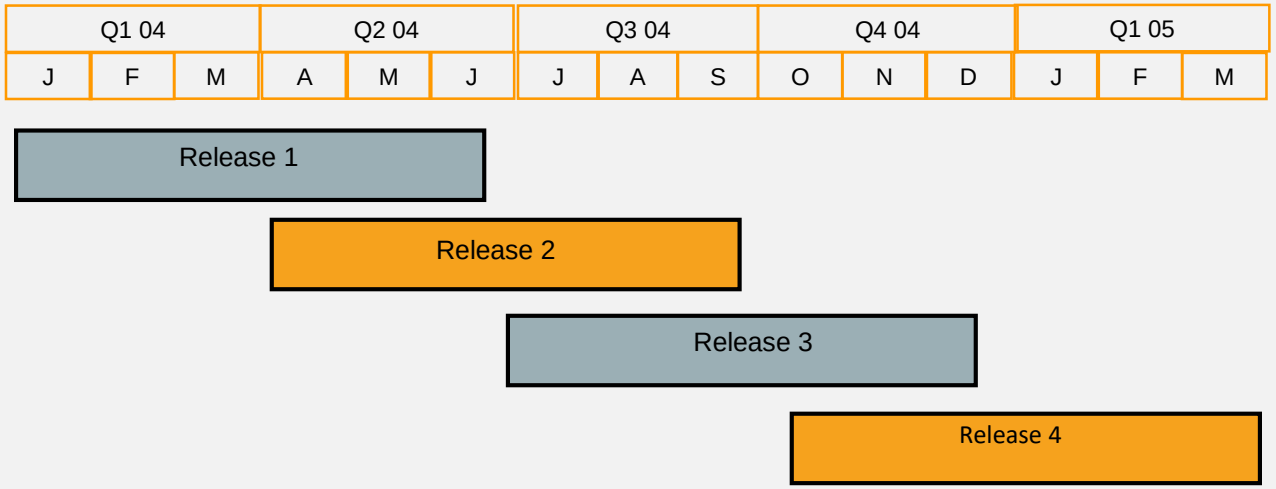
BENEFITS OF RELEASE MANAGEMENT

- Provides for an integrated (and transparent) view of both business and IT plans
  - Open planning can provide a clear view of what is being developed, and when the key milestone will be achieved.
- Results in a more stable production system
  - The introduction of an integrated release early in the development cycle allows for more careful analysis and testing of impact to normal operations.
- Creates predictability in delivery timeframes, costs, and support requirements
  - Release Management provides all corporate entities with a clear view of the functionality being delivered and release scheduling, both in the short and long run.
- Allows for the utilization of corporate resources consistent with corporate priorities
- Used by many large organizations such as Cisco, Sun, etc.

RELEASE MANAGEMENT

RELEASE PLANNING

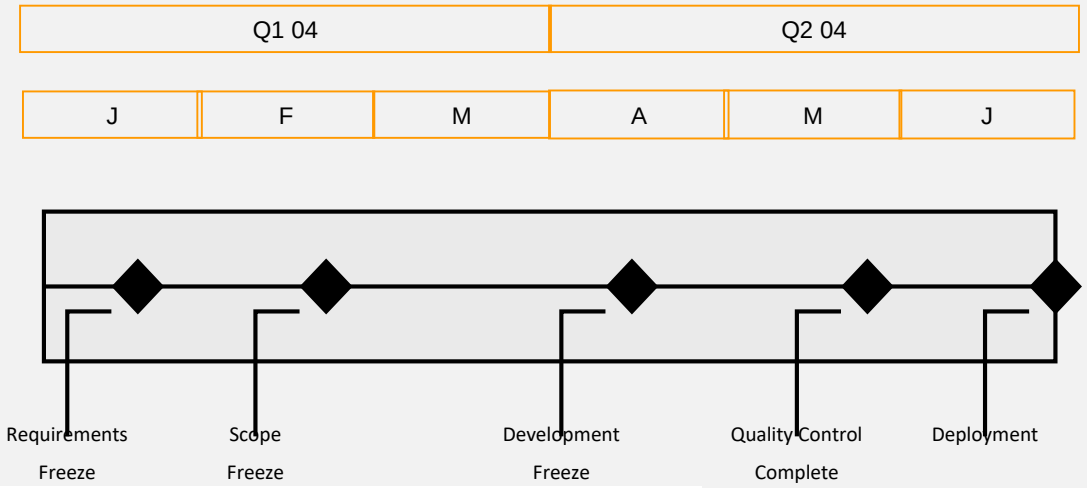
- What are the systems? Is there a grouping of systems?
- How many releases? When to release?
- How much overlap?



RELEASE MANAGEMENT

RELEASE LIFECYCLE

- Each release will have a lifecycle, with phases.
  - i.e. Initiation, Planning, Build, Deployment and Close Out
- Identify within each phase a key milestone(s).
- Manage each release to these milestones.



RELEASE MANAGEMENT

RELEASE SCHEDULING

- Need to determine the duration of each release.
  - i.e. 4 releases in a year, therefore each release is 3 months.
- Determine the percentage of time each phase will occupy.
- Choose your release date(s) and reverse plan the release.

	Initiation	Planning	Build	Deployment
Percentage	10%	20%	45%	25%
Key Milestones	Requriements Freeze	Release Scope Freeze	Development Freeze	Quality Control Complete/Deployment
Duration (months) = 6	0	1.2	2.7	1.5
Schedule	26-Mar-05	7-May-05	30-Jul-05	15-Sep-05

RELEASE MANAGEMENT

RIGIDITY VERSUS FLEXIBILITY

- Rigidity of the release schedule is dependent upon the organization implementing Release Management.
- Typically, there is a process for introducing “Hot Fixes” in between release deployment dates.
  - Constrained to only include Priority 1 system bugs, or “Business Critical” enhancements.
  - Exceptions and not the norm, otherwise the organization risks losing the benefits.
  - Determine, well in advance, the number and type of Hot Fixes the RM team can absorb without putting the release in jeopardy.
- To allow for some level of flexibility, the release dates can be given a plus or minus factor.
  - Allow the team to either expand or contract the release schedule depending on different constraints.
  - Changes to release dates will cascade down to subsequent releases, therefore Change Management is critical.

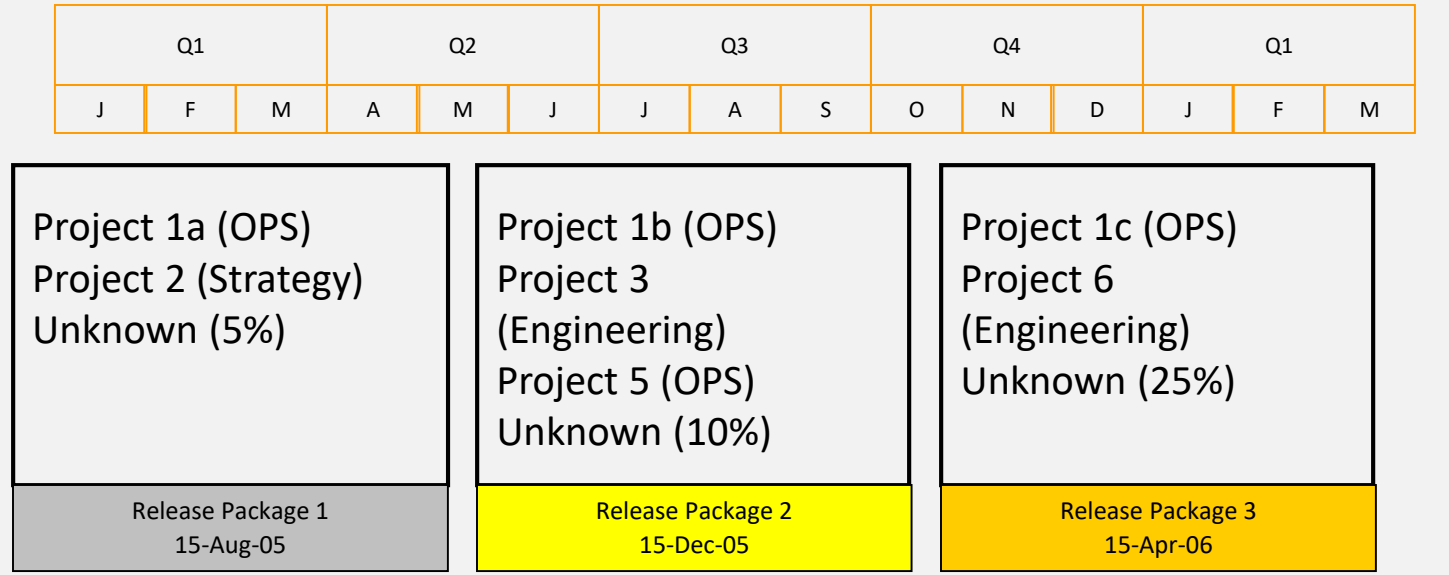
RELEASE MANAGEMENT

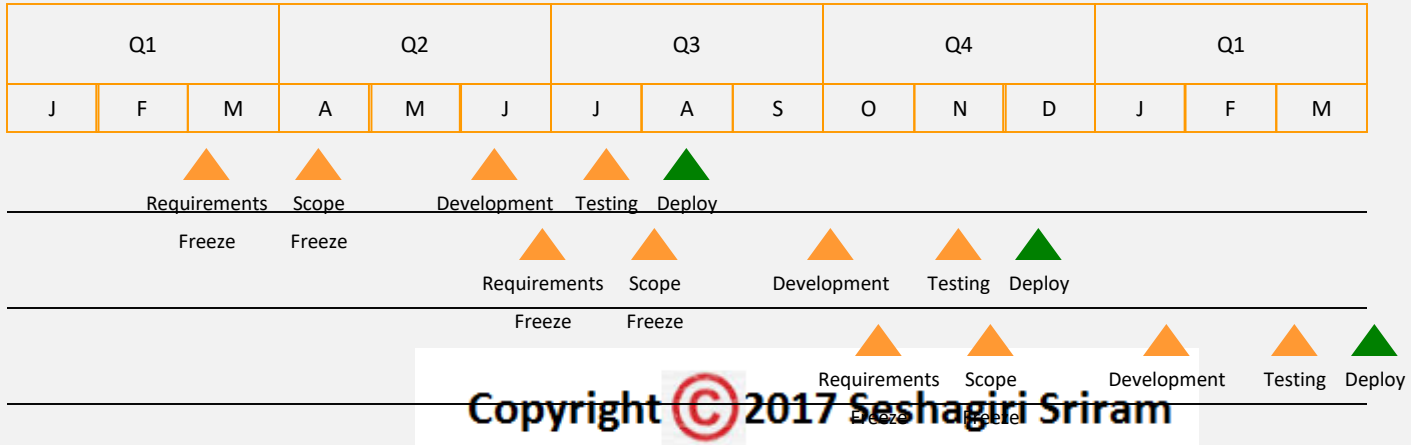
MULTIPLE SYSTEMS/MULTIPLE BUSINESS REQUESTS

- Typically with an enterprise, there are a series of systems that may be either tightly or loosely coupled.
  - Groupings of systems; i.e. Provisioning Systems versus Operations Systems
  - Upstream versus downstream
- There may be any number of organizations with business requests resulting in development on one or more of the systems.
  - May or may not have shared needs or competing needs
  - Need must be integrated
- Release planning sessions that include all appropriate organizations from the lines of business and IT.
  - Determine the critical scope or “Anchor Functionality” and the reserve for unknown functionality.
  - Rolling 12 month view of the release plan, therefore meet regularly.

RELEASE MANAGEMENT

RELEASE ROADMAP

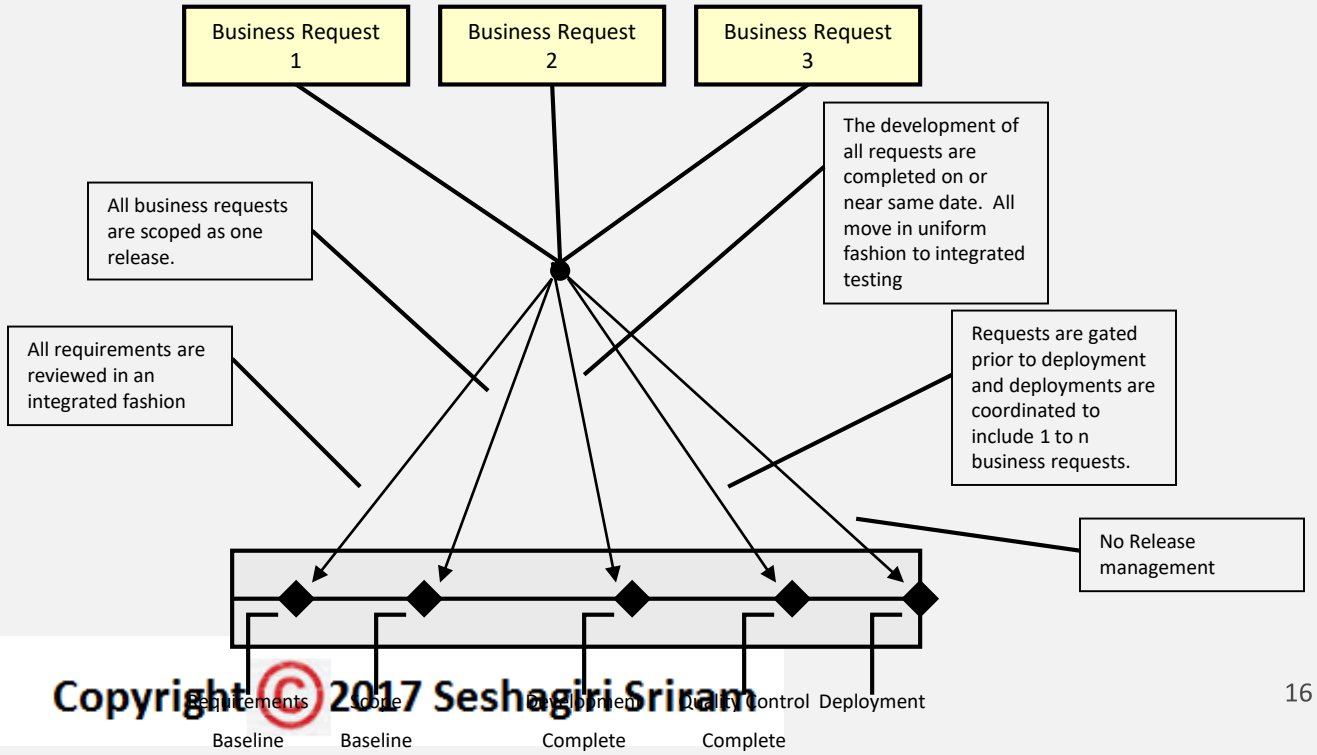




RELEASE MANAGEMENT

INTEGRATION POINTS

- Need to determine where an integration point should be planned within the RM lifecycle.
- May be a temptation to only concentrate on those functionalities that appear, during analysis phase, to be interrelated.



RELEASE MANAGEMENT

MAKE IF OFFICIAL

- Use System Development Lifecycles (SDLC) such as the Waterfall or Rational Unified Process (RUP) approaches.
- Use Project Management best practices, i.e. PMBOK.
  - Each release could be viewed as a project.
- Formally document the policies and operating principles, such as:
  - Scope Management
  - Metrics Management
  - Quality Management
  - Change Management
  - Strategic Release Planning
  - Integrated Testing
  - Risk Management

RELEASE MANAGEMENT
ROLES

- Release Manager
  - A project manager whom manages release.
    - The release, the whole release and nothing but the release.
  - Must be thick skinned!
- Business Project Manager
  - Focused on the business needs.
  - Natural tension with Release Manager.
- Development Manager
- Configuration Manager
- Environments Manager
- Testing Manager

RELEASE MANAGEMENT

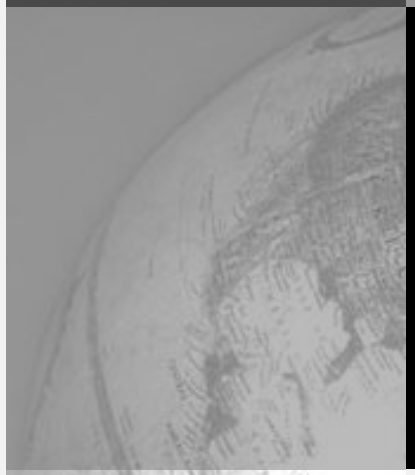
INFORMATION MANAGEMENT TOOLS

- Release Management Planning and Deployment tool (RMPD)
  - Tracking of multiple business requests, the associated software deliverables of those requests, as well as release planning and scheduling, and the association of software deliverables contained within a release.
  - Open-view of planned releases and the functionality for each release.
  - Automate additional Project Management tasks geared towards the release.
  - Integrated with other IT tools, such as Defect Tracking, Budgeting, etc.
- There is not a strong suite of tools currently available for managing the release management process.
  - Maybe the Rational Suite
  - MS Excel
  - MS Project
  - Homegrown

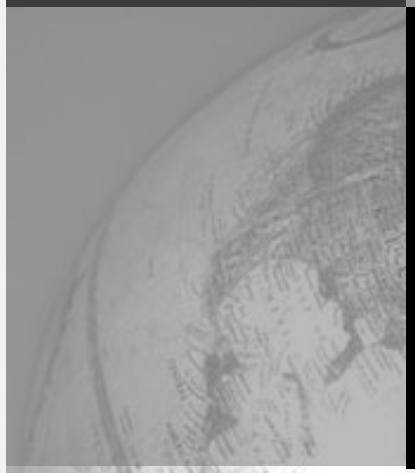
RELEASE MANAGEMENT

SUMMARY

- IT organizations are losing credibility due to their inability to provide predictability in software delivery timeframes, costs, and support requirements. This is particularly evident during the Application Management phase of a systems lifecycle.
- Release Management is a methodology that provides predictability, stability and transparency to software delivery.
- Planning, planning, planning.
  - A release in June might start in January
- Scope takes a back seat to schedule.
- Not a silver bullet, the implementing organization must truly understanding the business objectives and the tradeoffs.



Questions and Answers.



Thanks for listening!!!

Seshagiri Sriram
Email: SeshagiriSriram@gmail.com