



Celebrating 125 Years
of Engineering the Future



The Role of SCM in Robust, Reliable, and Resilient Software Engineering

Chuck Walrad
cwalrad@daven.com

Agenda

1. Overview
2. What SCM Is - Standards View
3. What SCM Is – Practitioner's View
4. How They Come Together: Elements of SCM Implementation
5. Summary



Celebrating 125 Years
of Engineering the Future



OVERVIEW

April /2009

Davenport Consulting, Inc.

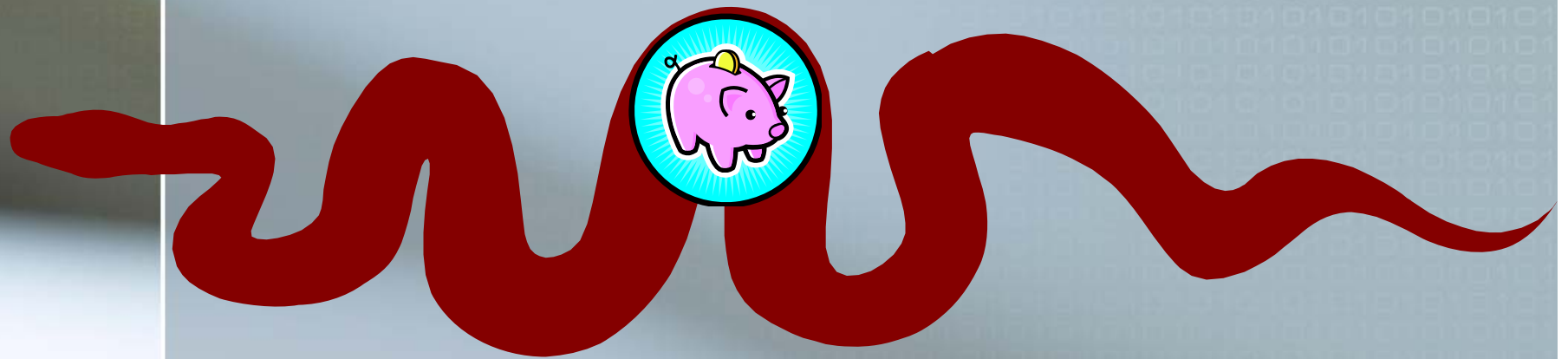
3

The Centrality of SCM

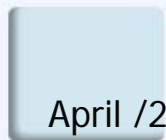
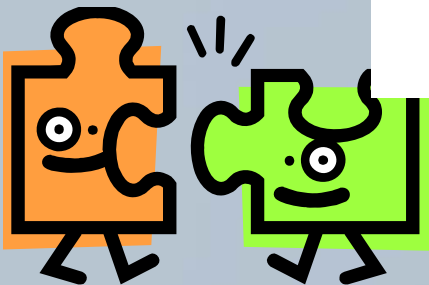
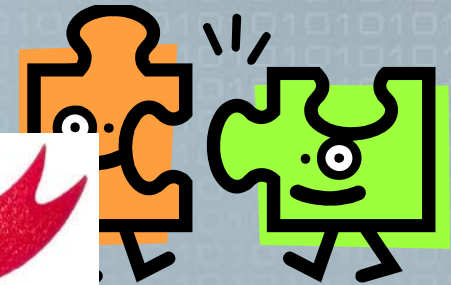
Although Software Configuration Management is traditionally classified as a “supporting” process in the SW Lifecycle, the emergence of a disciplined Agile approach to software development has made it clear that successful Agile development of robust and reliable software depends on asserting SCM as central to software success.

Getting the Pig thru the Python

- Was one way to deliver software ...



But Agility Requires More



April /2009

Goal of SCM

The primary goal of SCM is to provide full visibility of the [emerging] product's present configuration and the status of achievement of its requirements so that *everyone* working on the product -- at any time in its lifecycle – will have access to [the same] correct, accurate, and consistent information.

Software 101

- SW products evolve
- Their individual components evolve
- They evolve during the initial development effort.
- They evolve over the lifetime of the product.

Change Happens!

SCM's Value

- Effective SCM facilitates change.
- It keeps information (code, docs, etc.) associated with any change.
- It clearly identifies the latest (current) version of each changed item.
- It facilitates speed and integrity of the change process.

A Change Process Can't Be Fast and Efficient if

- What is being changed is not properly identified
- There is no owner for the change
- There is insufficient linkage other items that may be affected.
- The change process is not a closed loop.

The Cost of IN-effective SCM

The Institute of Configuration Management estimates that the cost of resources for intervention necessitated by poor CM ranges from 40% to 60% of product cost.

<http://www.icmhq.com/configuration-management-about-cmii.html>

Pitfalls -- without SCM



- Bad builds -- the wrong code --are delivered
- Fixes are lost
- Patches are lost track of, bad patches are distributed
- Schedules are delayed
- Costs go up

April /2009

12

The Heart of the Matter



- A large part of my work is project reviews.
- Very consistently, it turns out that the lack of well-defined and well-understood software configuration management, build and release processes is the **heart** of the inability to get the *right product* out the door -- at the *right time*, to the *right people*.

Effective SCM prevents frustration, extra costs and schedule delays

- Provides a known basis for making changes.
- Enables decisions based on correct, current information
- Avoids the costly errors of ad hoc or misapplied changes, both during development and in operation.
- Assures production repeatability

Effective SCM Makes the Whole Team More Effective

- Enables support of concurrent versions in the field
- Facilitates distributed development
- Facilitates parallel development

April /2009



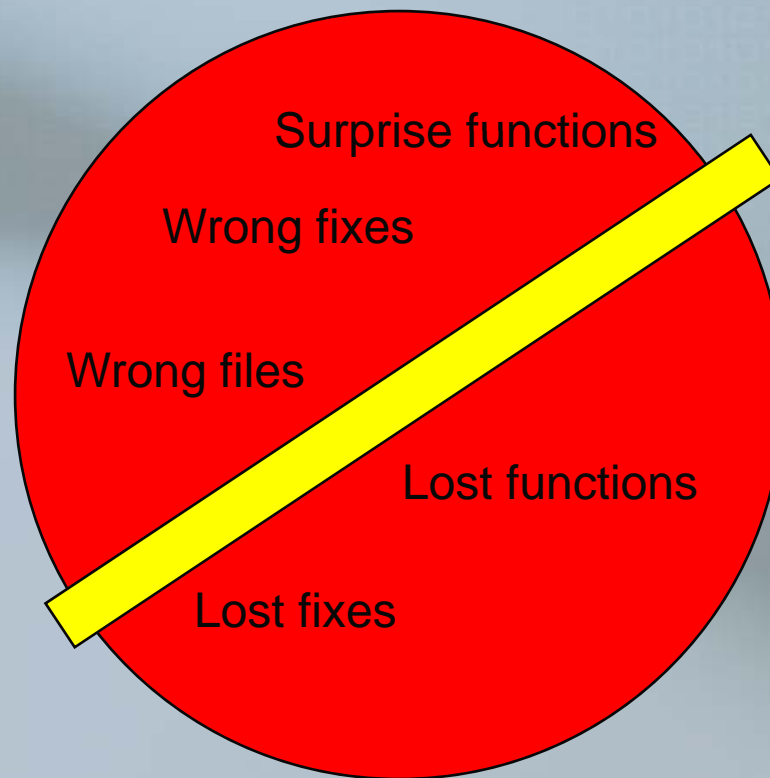
Effective SCM includes...

- The selection of an appropriate branching model
- Identification of all [and nothing but] items in the configuration (CIs)
- Development and enforcement of appropriate policies for managing those CIs
- Managing changes to CIs
- Build & release management

Attributes of Effective SCM

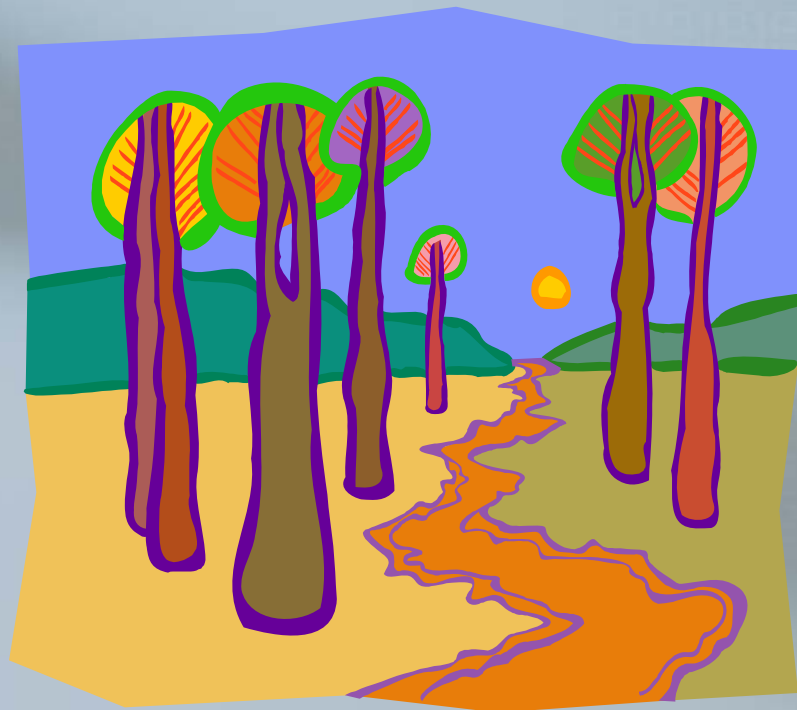
- Safety
- Stability
- Control
- Auditability
- Reproducibility
- Traceability
- Scalability

In other words ...



That's Great!

- But how do we get there?



The "Law" of the Excluded Middle

- Software development and maintenance, all too often, represent a state of barely controlled (or nearly controlled) chaos.



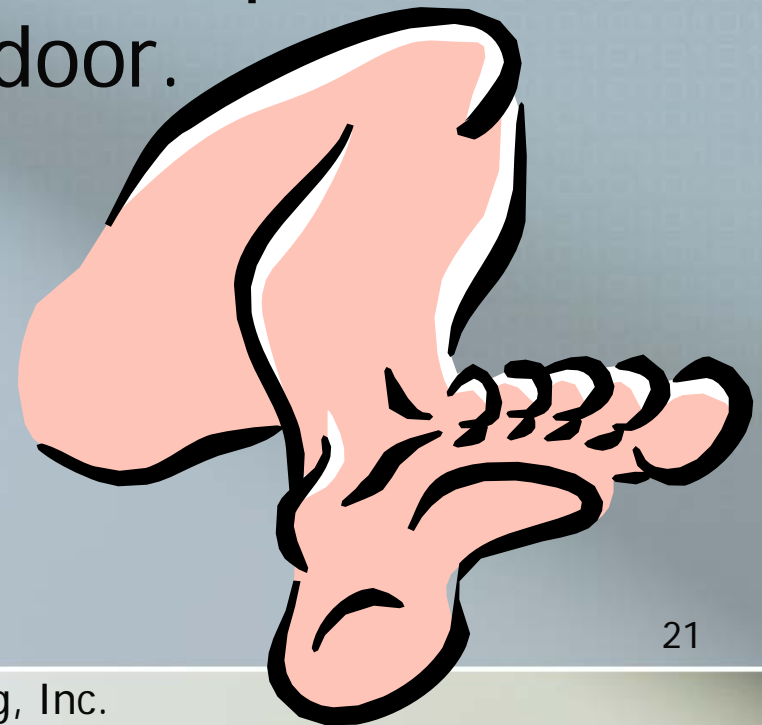
- Developers often fend off "process" as if life itself were at stake.



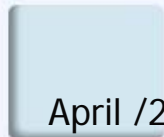
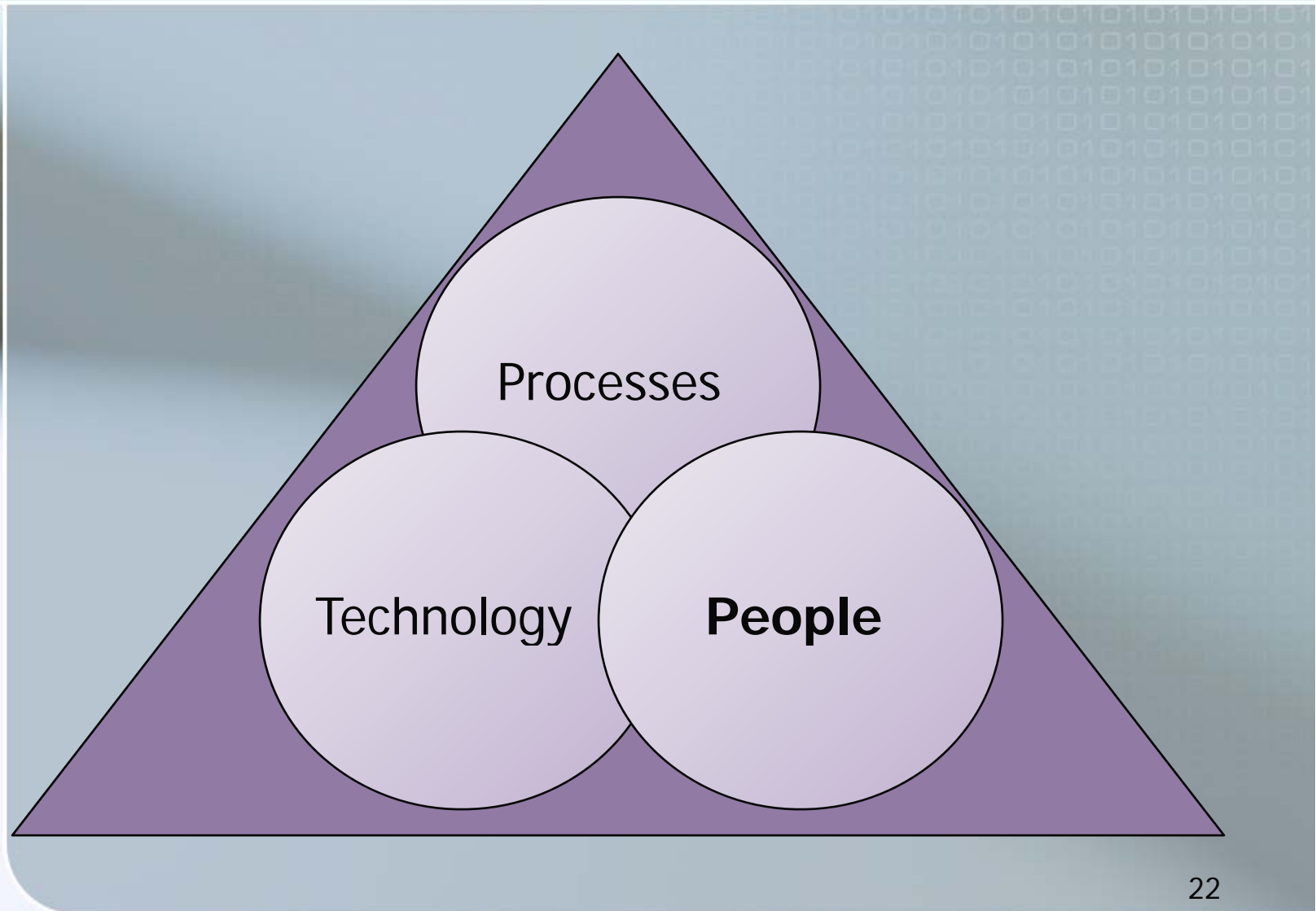
April /2009

Getting a Toe in the Door

- We embrace each new type of tool as the true silver bullet.
- Fortunately, the ready availability of SCM tools helps the discipline of SCM get its foot in the door.



The Whole Answer has to address the Whole Picture



April /2009



Celebrating 125 Years
of Engineering the Future

WHAT SCM IS – STANDARDS VIEW

April /2009

Davenport Consulting, Inc.

23

Relevant IEEE Standards

- IEEE Std. 828 -2005: SCM Plans
- IEEE/EIA Std. 12207*: SWLC Processes
- IEEE/EIA Std. 15288: System Lifecycle Processes
- IEEE SWEBoK – Chapter 7

*Note: According to 12207, SW is always part of a system. 12207 provides the minimum *systems engineering activities* needed for a *SW project*.

Where SCM Fits in IEEE Std 12207

- “Technical Processes” build the product: requirements, design, construction, integration, test ...
- “Supporting Processes” facilitate and augment the technical processes: doc, SCM, SQA, V &V ...

Standards-based SCM Activities

- Planning and managing the SCM process
- Identifying SW Configuration Items (CIs)
- Controlling SW configurations
- Collecting and reporting SW configuration status
- Auditing/evaluating SW configurations
- Managing SW release and delivery

SCM Planning Must Address

- What SW product (release) the plan is for
- Responsibilities and authorities for carrying out the SCM activities
- Description of the SCM activities to be performed
- SCM schedules
- SCM resources (H/W, SW and people)
- How the plan will be kept current

*IEEE Std. 828 Defines SCM Plan Contents*₂₇

April /2009



Celebrating 125 Years
of Engineering the Future

WHAT SCM IS – PRACTITIONER'S VIEW

April /2009

Davenport Consulting, Inc.

28

The Typical Job Descriptions





- “Responsibilities include **configuration management, change control, release management, build creation, build promotion** and related activities “

The Typical Job Descriptions


- “Develops, analyzes and maintains tools that support and automate **processes for software product releases**. Write **installation** scripts and programs for installation of products. Works with product teams to determine an appropriate build schedule and then initiates the **build and packaging process**. Establish, document and maintain the **build process** for each product. Facilitates the **Software Configuration Control Board meetings** for each product. **Assist developers or other product personnel with any questions or problems** related to the version control tool, defect tracking tool, or general **Configuration Management process**. Works with other development and product team members to analyze, evaluate and design alternatives and improvements to processes in support of XP development methodology. **Champions the effort to define and work through build problems**. Demonstrates good judgment in selecting methods and techniques for **obtaining solutions to CM situations**. Interacts well with **all levels of personnel**. Performs **software deficiency and change request updates and reporting**; **Maintains the product repository for source code and script files**; **Tracks and maintains the product Software Change Requests**; **Measures and reports on releases and Software Change Requests trends**; **Supports management with technical performance data from the CM process** ”



Celebrating 125 Years
of Engineering the Future



HOW THEY COME TOGETHER— IMPLEMENTATION



April /2009

Davenport Consulting, Inc.

31

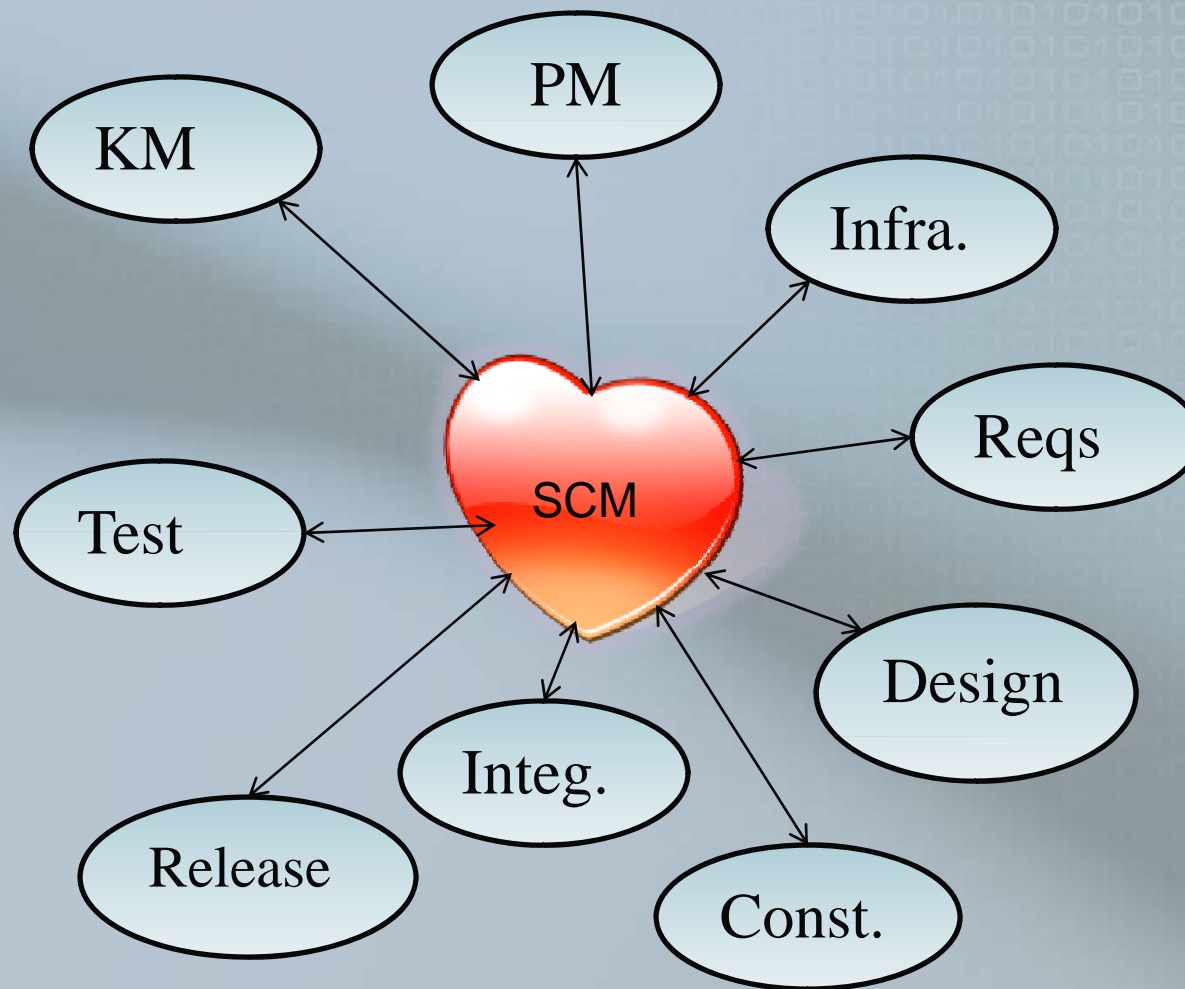
The Expanding Purview of SCM in Industry Is Driven By

- The increasing adoption of iterative, incremental development processes
- The need for parallel development for shorter cycle times
- The need for distributed development
- The greater availability of tools that support branching and merging

As a Result

- SCM has been pushed upstream in the development process.
- SCM has become an integral part of code development
- SCM kicks in early because developers need it
- “SCM” is typically seen as encompassing code integration, not just promotion of builds.

The Centrality of SCM



SCM Standards

- Because SCM tools are becoming widespread, it is important for SW organizations to recognize the importance of first defining adequate processes for the tools to support
- This enables getting the best ROI for the tool and allows the project team to use it effectively
- Existing standards can help a lot in defining SCM processes!

April /2009

35



Celebrating 125 Years
of Engineering the Future

SUMMARY

April /2009

Davenport Consulting, Inc.

36

Times Are Changing



- Software product companies recognize SCM as a best practice.
- IT organizations are increasingly aware of the need for SCM as governance requirements have come to the forefront.
- A wide variety of increasingly sophisticated tools are available in a variety of acquisition options (free, fee-based, open source, proprietary)

April /2009

Times Are Changing



- SCM has moved from a trailing function to become a central function
- Especially apparent in Development best practices, with nightly or continuous builds that take the source code out of the SCM repository to assure that everyone has equal access to the best & most current files.
- Voila – speed and integrity!

Times Are Changing



- Software Lifecycle Standards call out the importance of SCM
- Software standards describe the key aspects of SCM
- IT Operations standards are raising the visibility and importance of SCM.



Celebrating 125 Years
of Engineering the Future



QUESTIONS?

cwalrad@daven.com

www.daven.com

650-593-1833

April /2009

Davenport Consulting, Inc.

40