Did Your Quality Initiative Fail Again?

Daniel Navarro

Presentation

F4

Friday 23rd November, 2001

International Conference On

Software Testing, Analysis & Review

November 19 - 23 Stockholm, Sweden

Did your SQA initiative fail again ?

Try Software Risk Management jargon for a change !

By Daniel Navarro

EuroSTAR 2001 -Stockholm, Sweden

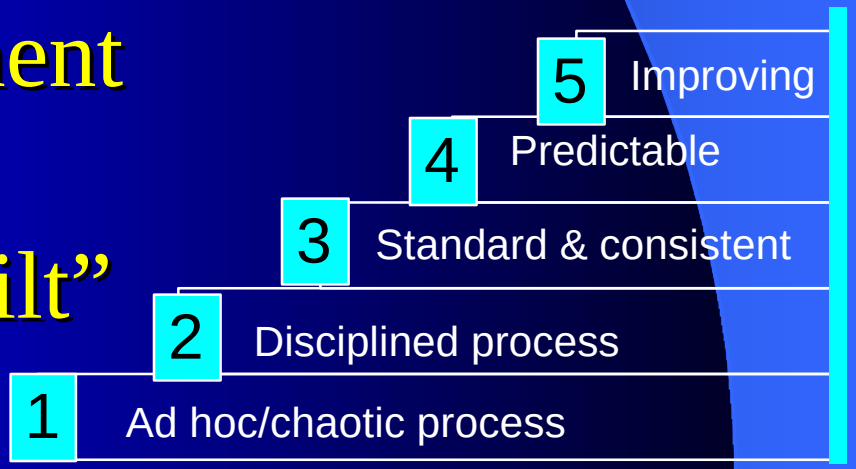
November 23, 2001

Agenda

- What is Software Quality Assurance (SQA) ?
- Why do SQA initiatives often fail ?
- What is Software Risk Management (SRM) ?
- Why do SRM initiatives work ?
- How to integrate SQA and SRM into a comprehensive framework
- Examples & Resources
- Conclusions
- Q & A

Software Quality Assurance (SQA) according to CMM

"SQA's purpose is to provide MANAGEMENT with appropriate VISIBILITY into the PROCESS being used by the software development PROJECT and of the PRODUCTS being built"



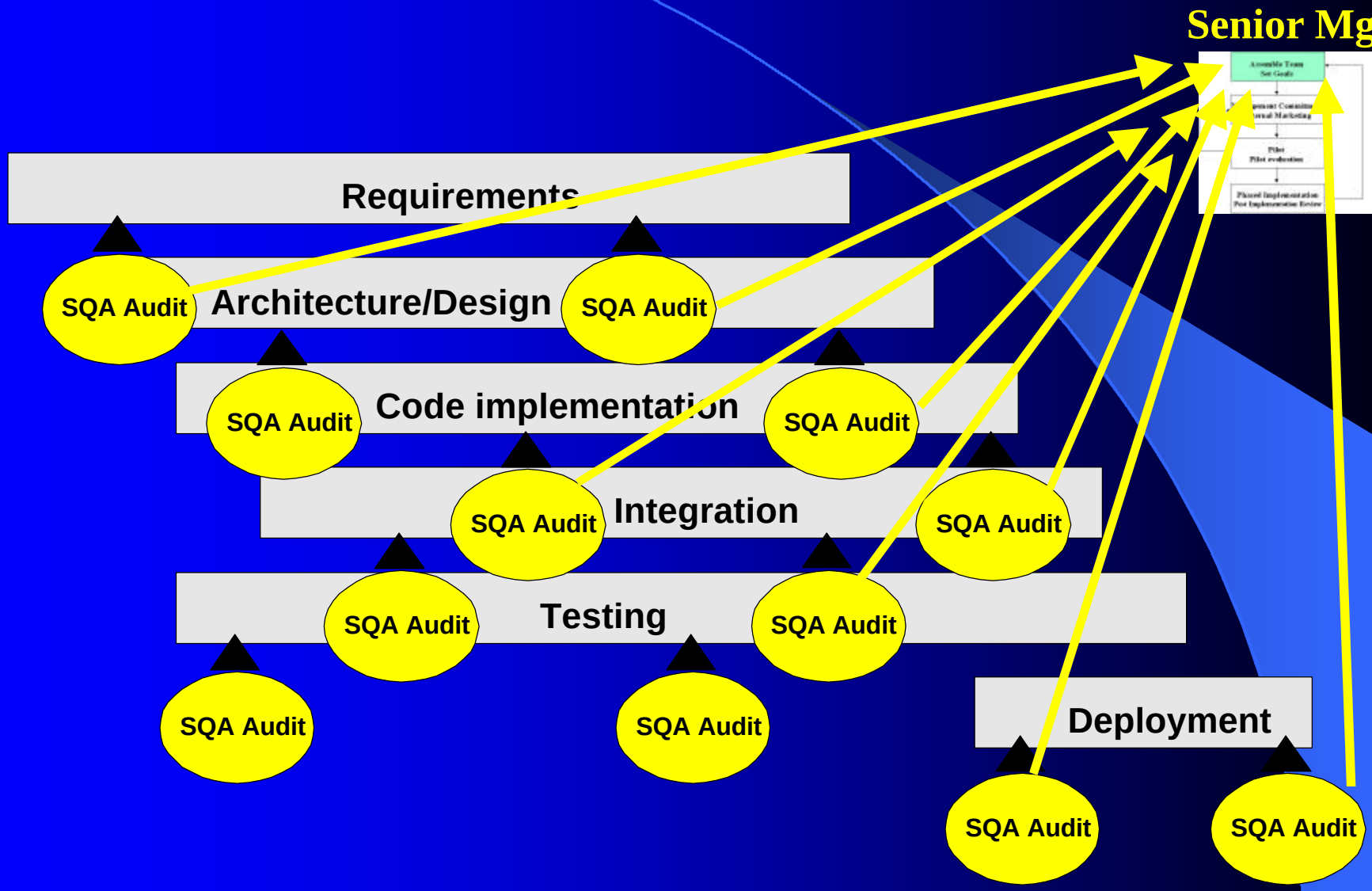
SQA's commitment to perform according to CMM

- **The project follows a written organizational policy for implementing SQA**

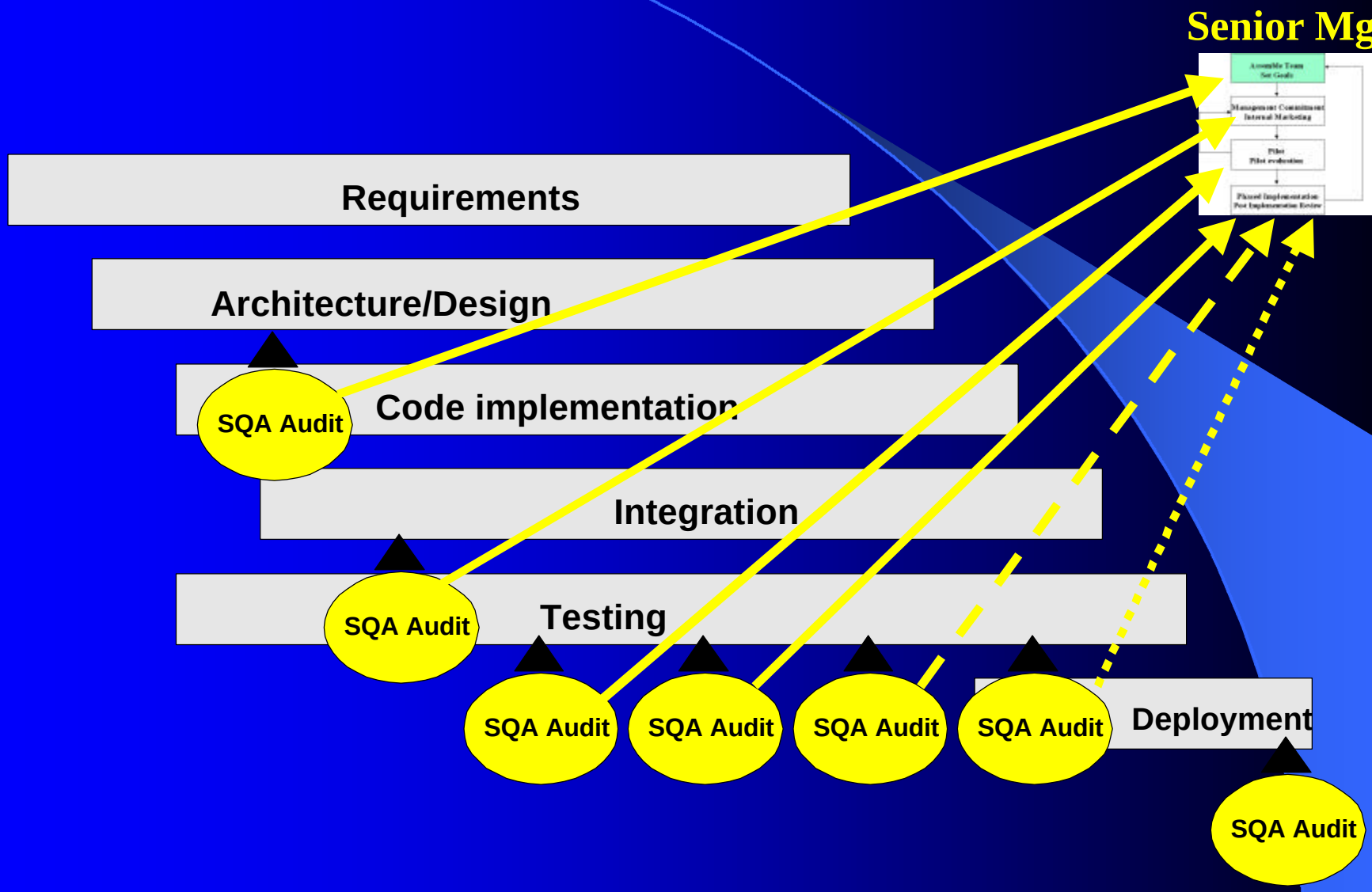
- **The SQA group has a reporting channel to senior management that is independent of:**
  - **The project manager**
  - **The project's software engineering group, and**
  - **The other software related groups**

- **Senior management periodically reviews the SQA activities and results**

The way the process should work



The way it actually works



A typical “SQA” Report

From: SQA Manager
To: Project Manager

I regret to inform you that our latest product TP v.2.3 should not be released to production tomorrow because of the following reasons:

- **Failure to adhere to software development process: 17 times**
- **Issues found during testing: 1,345**
- **Programs that do not comply with naming standards: 24**
- **Black box testing (code coverage): 46%**

As you can see, we would be in high risk if we decide to release it as scheduled.

Sincerely,
Bobo

Problems are :

- **It does not provide a business oriented focus**
- **It does not reach the right audience**
- **It does not provide enough visibility on the quality of process nor product**
- **It shows up too late**
- **It provides a partial view of Software Testing (detection) rather than SQA (prevention)**

Software Development's Irony

"When companies rush to get a product into consumers' hands, they often sacrifice product quality, which ultimately costs them time and money"

Anup Ghosh

Director of Security Research at Cigital, Inc.

What is Software Risk Management?

Software Risk Management is the means of identifying and managing the risks that software brings to a business

- Software presents real business risk that must be mitigated
- Mitigation strategies should be driven by business requirements
- All software problems are related

Software-induced business risks

- **Loss of revenue**
  - **Software fails**
  - **Key information is stolen or compromised**
- **Brand damage and Severe market impact**
  - **Software does not work as advertised**
  - **Security vulnerability impacts customer trust**
- **Liability Costs / Negligence**
- **Loss of Productivity**
  - **Software malfunctions**
  - **Software ceases to function altogether**

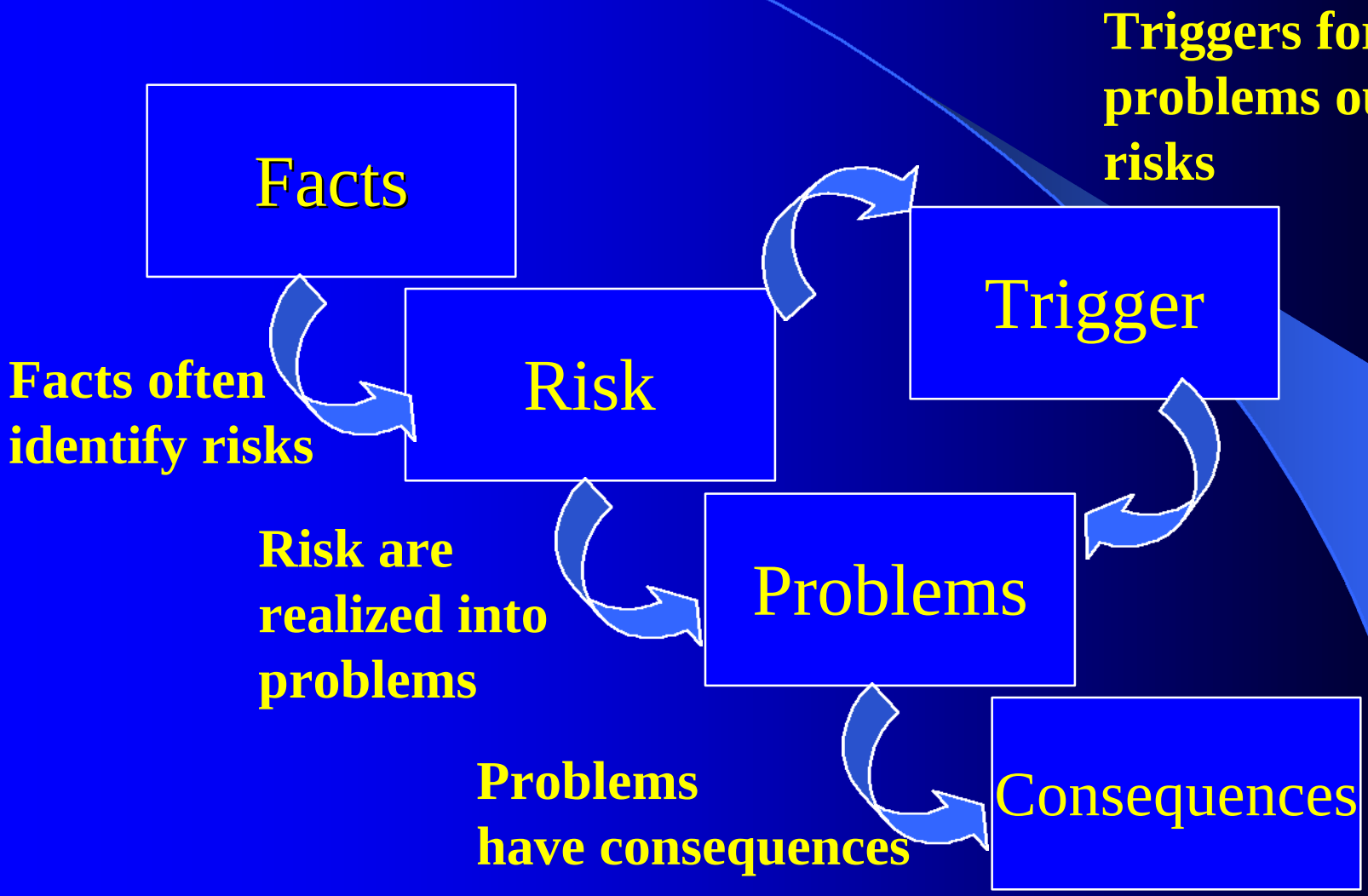
Technology issues behind SRM

- Product risks
  - Poor reliability (it crashes or is unavailable)
  - Security vulnerabilities (it can be hacked)
  - Safety issues (it causes systems to crash and kill people)
- Project risks
  - Project risk management
  - Skills issues
  - Processes – configuration management, bug tracking, build management, development processes, test facility management

Why use SRM jargon ?

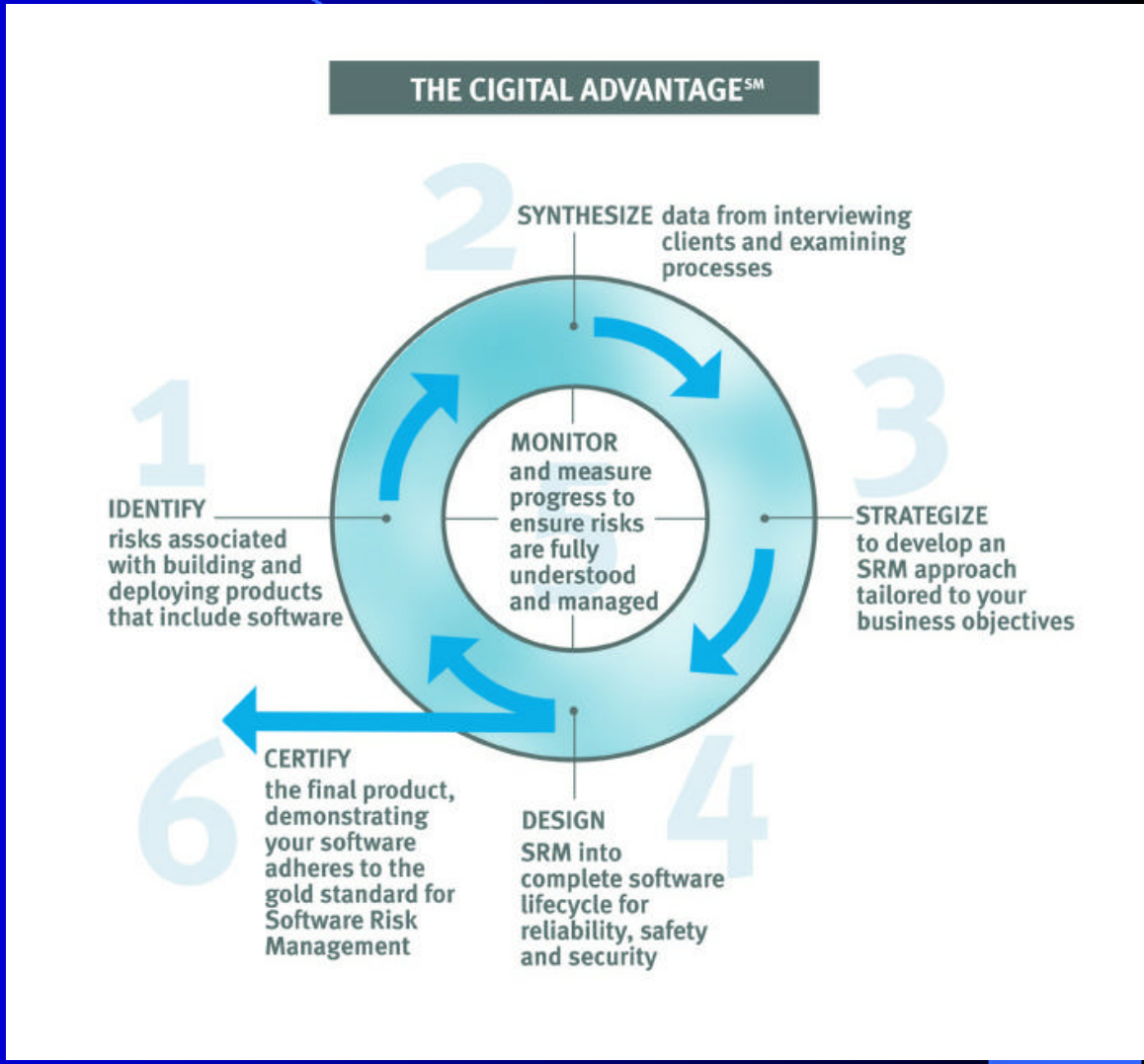
- **Business people and executives understand how to manage risk – they make calculated decisions when given the right data about software behavior**
- **Issues can be framed in terms of potential payoff and required investment**
- **The key is to understand the business impact of technical software risk**
- **Software Risk Management efforts must be driven by business impact determination**
- **Risks must be Identified, Ranked in order of Severity and Addressed with mitigation techniques.**

Definition relationships



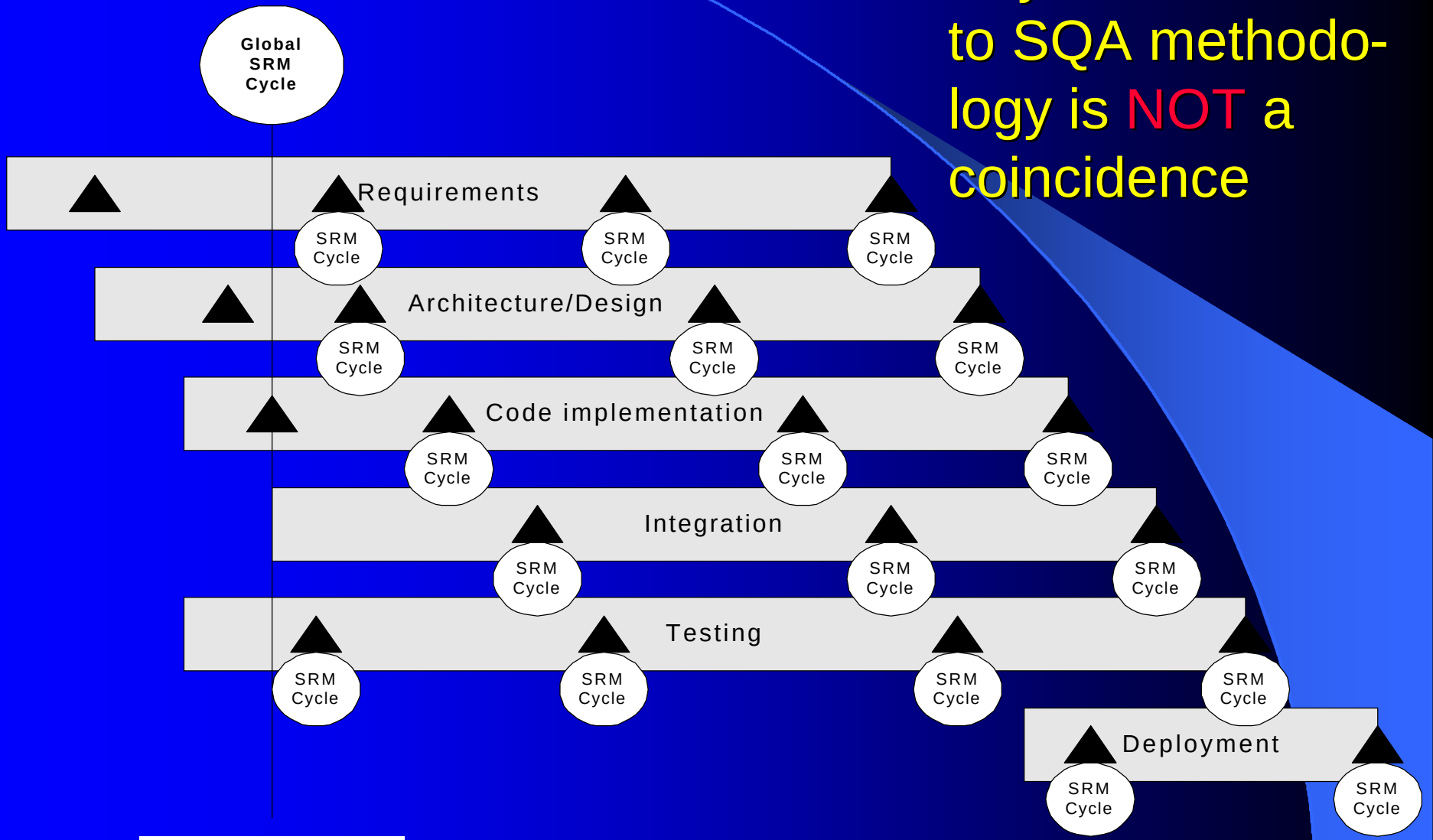
SRM Methodology based on business-risk analysis

- **Identify software-induced business risks**
- **Synthesize information relevant to product use**
- **Create a strategy to determine critical trade-offs**
- **Implement the SRM strategy by designing, measuring, monitoring and testing software against identified business risks**



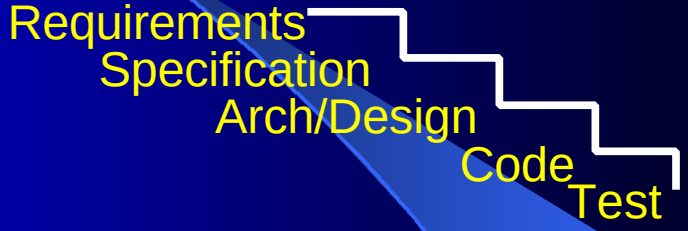
Applying the SRM model

Any resemblance to SQA methodology is NOT a coincidence

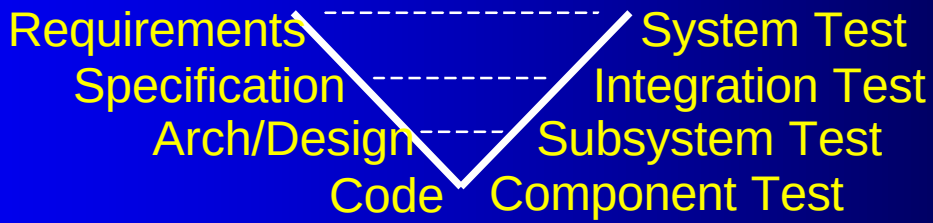


Software development life-cycle models

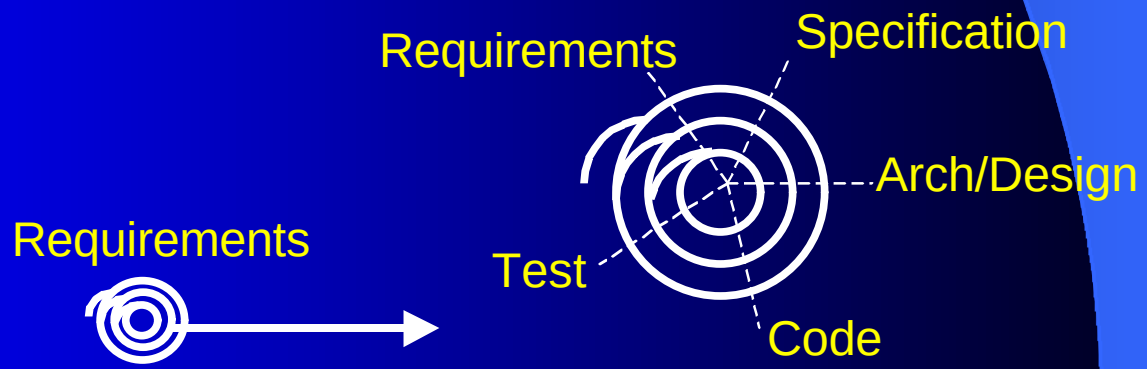
- Waterfall



- V-model



- Spiral model



Why do SRM initiatives work ?

- **Primary considerations:**
  - Cost
  - Time-to-market
  - Quality
- **Additional considerations:**
  - Security
  - Protecting consumer privacy
  - Your own ghosts

Why do SRM initiatives work ?

- They let the business guide their activities – Risk identification is a dynamic process

- They create a strategy for releasing a product on time, meeting both functional and assurance requirements

- They think ahead (prescriptive) and analyze the impact beyond the software; develop strategies to mitigate software induced risks

How to integrate SQA and SRM into a comprehensive framework

SQA and SRM have a lot in common:

- **Both collect facts about a product, project and their corresponding software development processes**
- **The reports of both target the same audience**
- **Both identify the business consequences of unreliable software**

Plus some of SRM's features:

- **Developing a plan that includes all facts and reliability risks and a tie to the consequences of such risks**
- **Creating a mitigation and monitoring strategy to reduce risks**

Reporting SQA facts with an SRM perspective

- **Collect facts about a product and its project to assess the reliability risks your company is taking**
- **Identify the business consequences of unreliable software**
- **Develop a plan that includes:**
  - **a statement of all facts and associated risks**
  - **a tie to the consequences of these risks (magnitude)**
  - **a mitigation and monitoring strategy to reduce these risks.**

A revised “SQA” Report

From: SQA Director
To: Vice President of Product Engineering

Upon completion of our assessment of the design phase of product TP v.2.3 we have found the following business risks:

- **Fact:**

We found 17 violations to screen format standards, including the use of our former company’s logo and the lack of Copyright legends for third-party software.

- **Business Risks:**

Brand Damage and Liability/Lawsuits if compliance with copyright laws is not appropriate

- **Mitigation Strategy:**

Re-Design all 17 screen formats and verify the compliance with copyright laws. Estimated effort: 40 person/hours.

Sincerely,
Maxwell Smart

It really works !!!

When has SQA and Sw Testing had more focus than during

- **Y2K bug**
- **Euro conversion ?**

Additional Resources on SRM

Books

- Hall, E. (1998). Managing Risk: Methods for Software Systems Development. Software Engineering Institute Series.
- Heiman, R. (2001). Software Risk Management: The Cigital Solution. IDC.
- Jones, C. (1994). Assessment and Control of Software Risks. Yourdon Press.
- Voas, J. (1995). Software Assessment: Reliability, Safety, Testability. John Wiley and Sons

Articles/Magazines

- Ghosh, A. (2000). Software Risk Management in E-business: Balancing Market-Driven Needs with Security and Privacy. SC Magazine. Available on line: http://www.scmagazine.com/scmagazine/sc-online/2000/srm/article.html
- Software Risk Management Magazine. http://www.srmmagazine.com

Any questions ?

Daniel Navarro
Vastera
45025 Aviation drive
Suite 200
Dulles, VA 20166 USA
dnavarro@mailbanamex.com

Friday 23 November 2001

F4

Did Your Quality Initiative Fail Again?

Daniel Navarro

Daniel Navarro has studied, worked, lectured and researched in the software area since 1983. He holds a Bachelors degree on Computer Science, a Masters degree on Business Administration and several post-graduate diplomas on Total Quality Management and Software Quality. He is currently a doctoral student at the George Washington University.

For six years he was in charge of the Quality Assurance effort at the 5th Largest Bank in Latin America. He has also been a consultant on Software Testing and Software Risk Management.

He is a founding member of the Mexican Association for Quality on Software Engineering (AMCIS, 1999) and has been a speaker at EuroSTAR, STAREAST in the USA, and LatinSTAR and other Software Testing conferences in Mexico. He has translated some English materials on Testing and Test Process Improvement into Spanish. Currently he is the QA Manager at Vastera in the United States.