SWEBOK

ÉTS

UQÀM

The Emerging Consensus on the Software Engineering Body of Knowledge

Pierre Bourque, École de technologie supérieure

Robert Dupuis, Université du Québec à Montréal

James W. Moore, The MITRE Corporation

SEPG 2002

Phoenix

February 21, 2002

Corporate Support by:

Project managed by:

Guide to the Software Engineering Body of Knowledge (SWEBOK)

- Began as a collaboration among IEEE CS, ACM and the Université du Québec à Montréal
- International participation from industry, professional societies, standards bodies, academia, authors
- By the time the project is finished literally thousands of individuals will have touched it
- Has completed the middle of three phases with the release of the *Trial Version*

List of Knowledge Areas

- Software Requirements
- Software Design
- Software Construction
- Software Testing
- Software Maintenance
- Software Configuration Management
- Software Quality
- Software Engineering Tools & Methods
- Software Engineering Process
- Software Engineering Management

Tutorial Objectives

- Give an overview of the emerging international consensus on the “core body of knowledge” of software engineering
- Explain how you can leverage the SWEBOK Guide within your organization
- Present the evolution of the SWEBOK Guide, the next steps and identify how you can participate

Presentation Plan

- **Project background**
- Project scope, objectives, audience and plan
- Contents of the Guide
- How you can leverage the Guide within your organization
- Future plans
- Class exercise in applying the Guide
- Conclusion

What is Software Engineering?

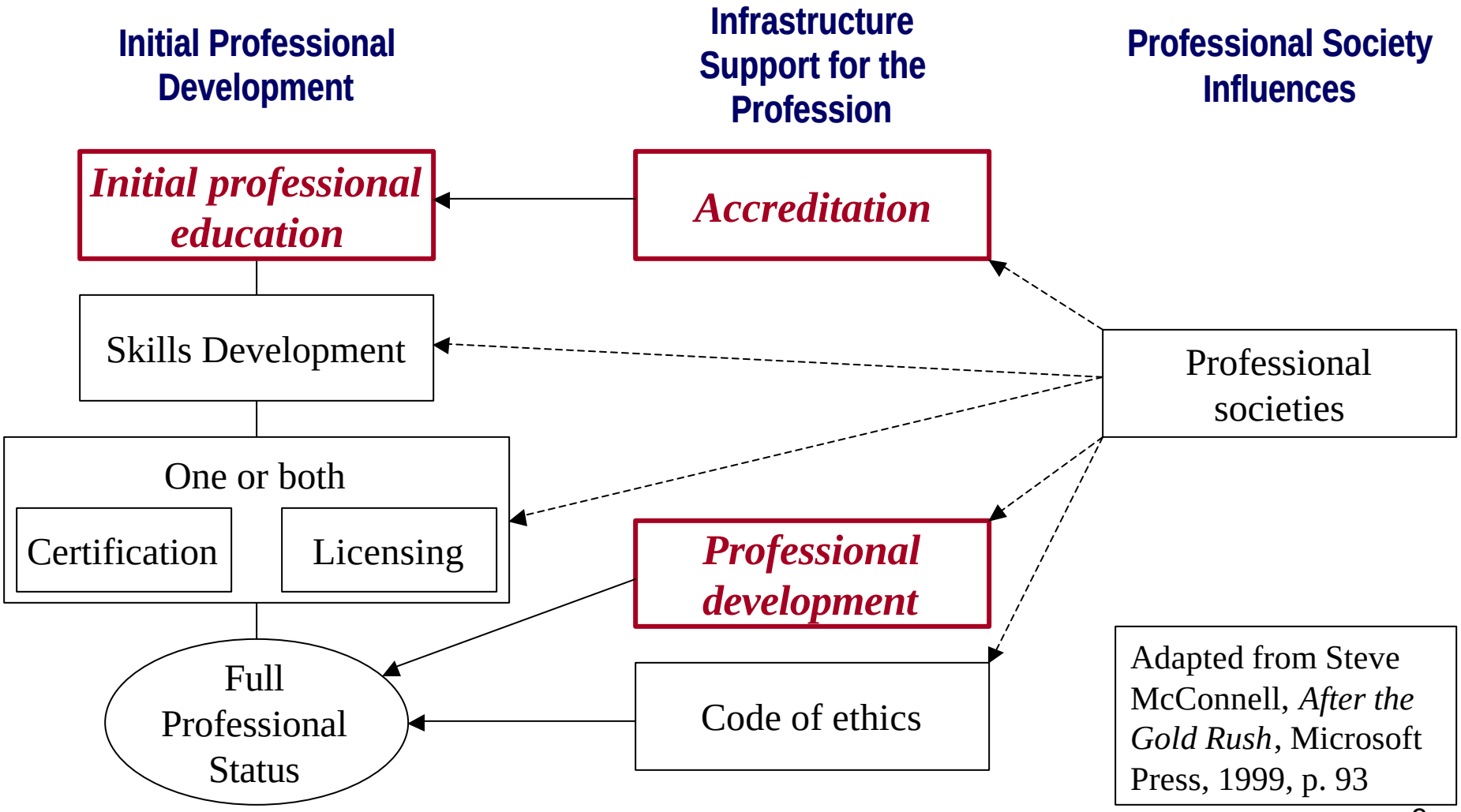
- IEEE 610.12:
  - "(1) The application of a *systematic, disciplined, quantifiable approach* to the development, operation, and maintenance of software; that is, the application of engineering to software.
  - (2) The study of approaches as in (1)."

Recognized Profession?

- Starr*:
  - Knowledge and competence validated by the community of peers
  - Consensually validated knowledge rests on rational, scientific grounds
  - Judgment and advice oriented toward a set of substantive values

* P. Starr, *The Social Transformation of American Medicine*: BasicBooks, 1982.

Professional Development



Increasing Interest in Software Engineering as an Engineering Profession?

- ACM / IEEE-CS Code of Ethics
- Texas Board of Professional Engineers
- Computer Science Curriculum 2001
- Many universities/engineering schools are offering undergraduate degrees in Software Engineering

Increasing Interest in Software Engineering as an Engineering Profession?

- CSAB & ABET are cooperating on accreditation
- Increased interest in the establishment of a profession (*After the Gold Rush* was #752 on Amazon.com)
- Continuing focus on organizational engineering capability (ISO 9000, CMM)

Presentation Plan

- Project background
- **Project scope, objectives, audience and plan**
- Contents of the Guide
- How you can leverage the Guide within your organization
- Future plans
- Class exercise in applying the Guide
- Conclusions

Project Objectives

- Characterize the contents of the Software Engineering Body of Knowledge
- Provide a topical access to the Software Engineering Body of Knowledge
- Promote a consistent view of software engineering worldwide

Project Objectives

- Clarify the place of, and set the boundary of, software engineering with respect to other disciplines (computer science, project management, computer engineering, mathematics, etc.)
- Provide a foundation for curriculum development and individual certification and licensing material

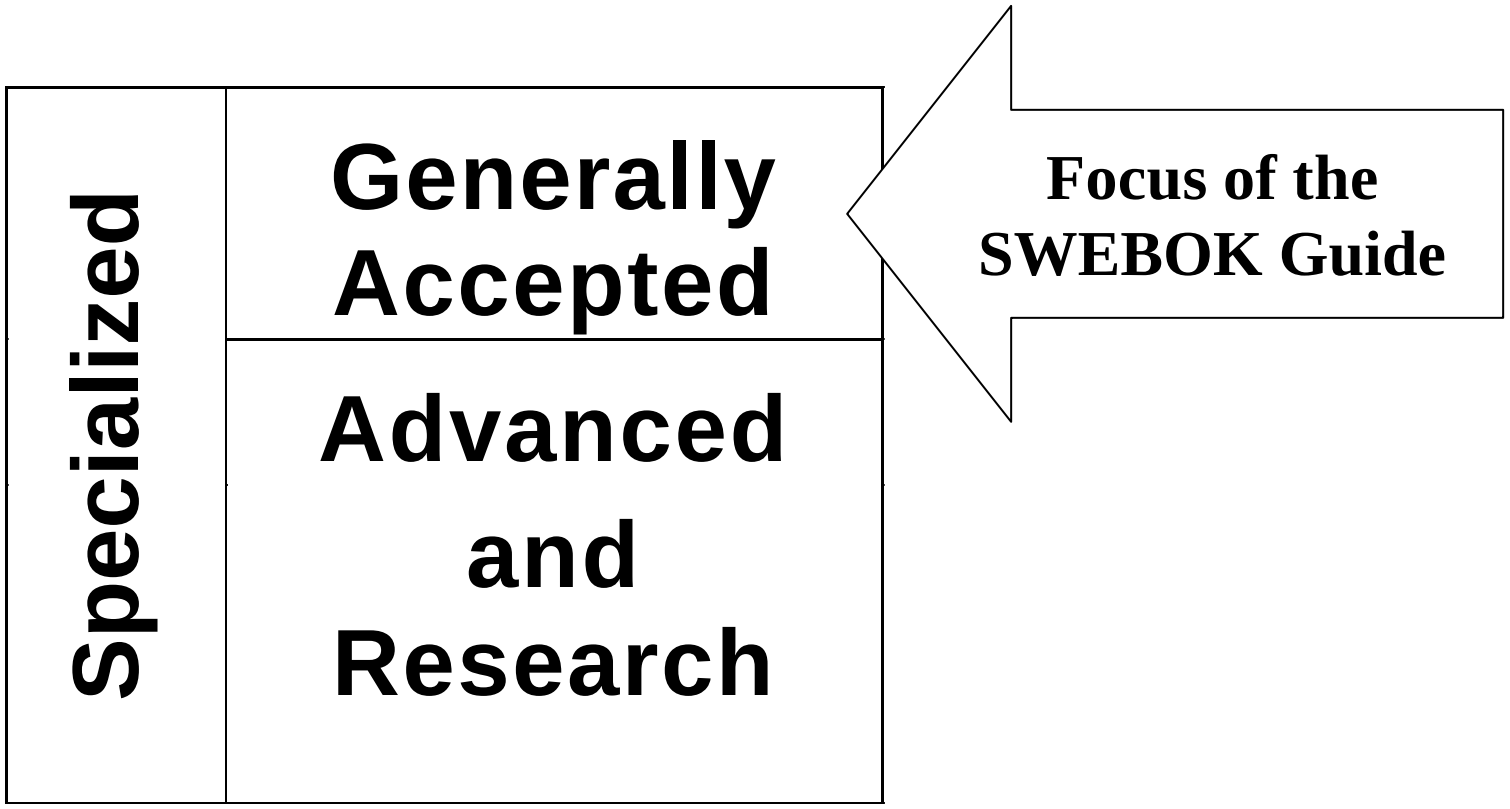
Intended Audience

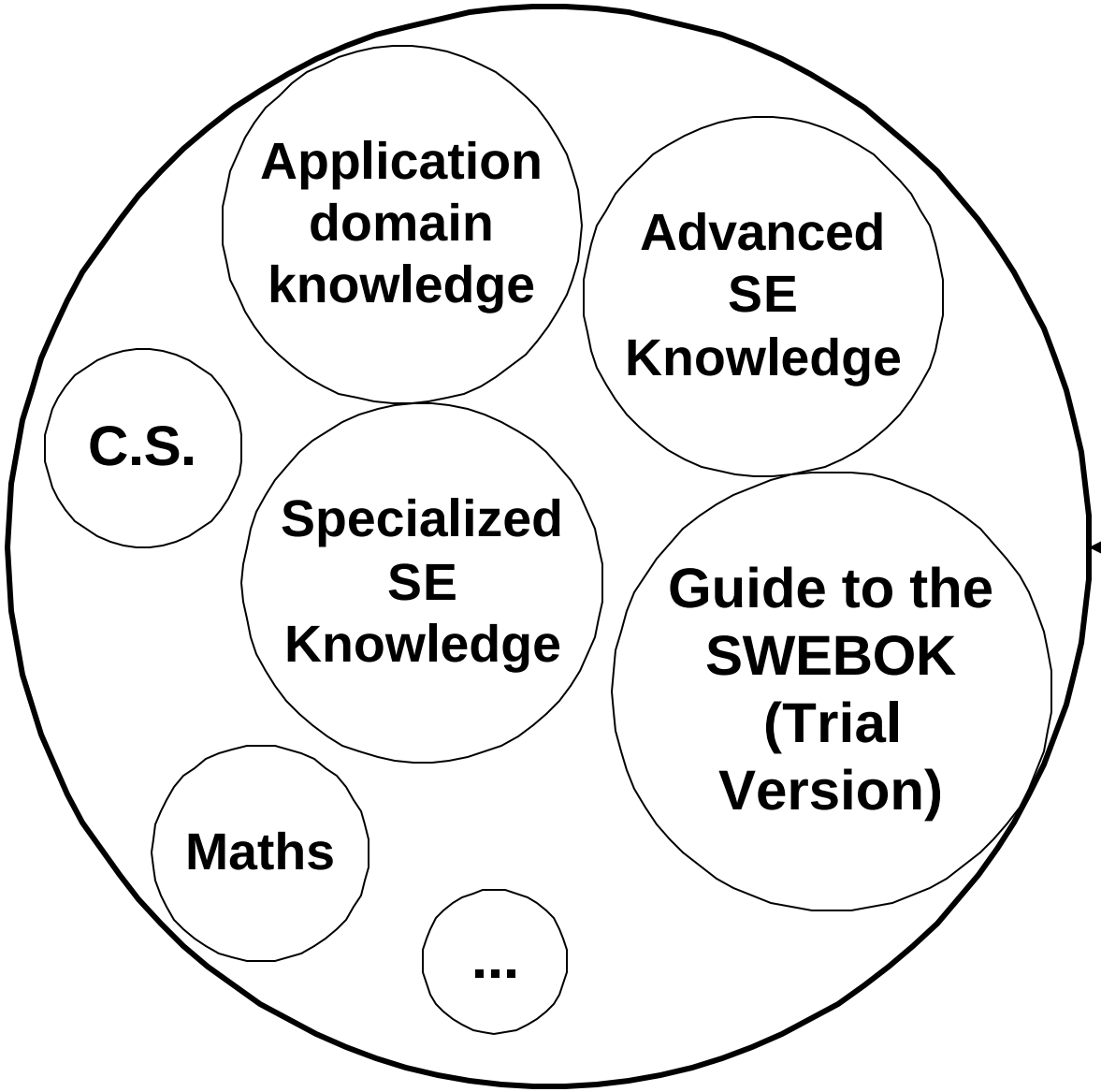
- Public and private organizations
- Practicing software engineers
- Makers of public policy
- Professional societies
- Software engineering students
- Educators and trainers

What Are we Not Trying to Accomplish?

- Not a curriculum development effort!
- Not an all-inclusive description of the sum of knowledge in the field
- Not all categories of knowledge

Categories of Knowledge in the SWEBOK Guide





Knowledge of a Software Engineer

Three Underlying Principles of the Project

- ***Transparency***: the development process is itself published and fully documented
- ***Consensus-building***: the development process is designed to build, over time, consensus in industry, among professional societies and standards-setting bodies and in academia
- Available ***free*** on the web

Project Team

- Editorial team
- Industrial Advisory Board
- Knowledge Area Specialists
- Reviewers

Editorial Team

- Project “Champion”:
  - Leonard Tripp, 1999 President, IEEE Computer Society
  - Chairman, Professional Practices Committee
- Executive Editors:
  - Alain Abran, École de technologie supérieure
  - James W. Moore, The MITRE Corp.
- Editors:
  - Pierre Bourque, École de technologie supérieure
  - Robert Dupuis, Université du Québec à Montréal

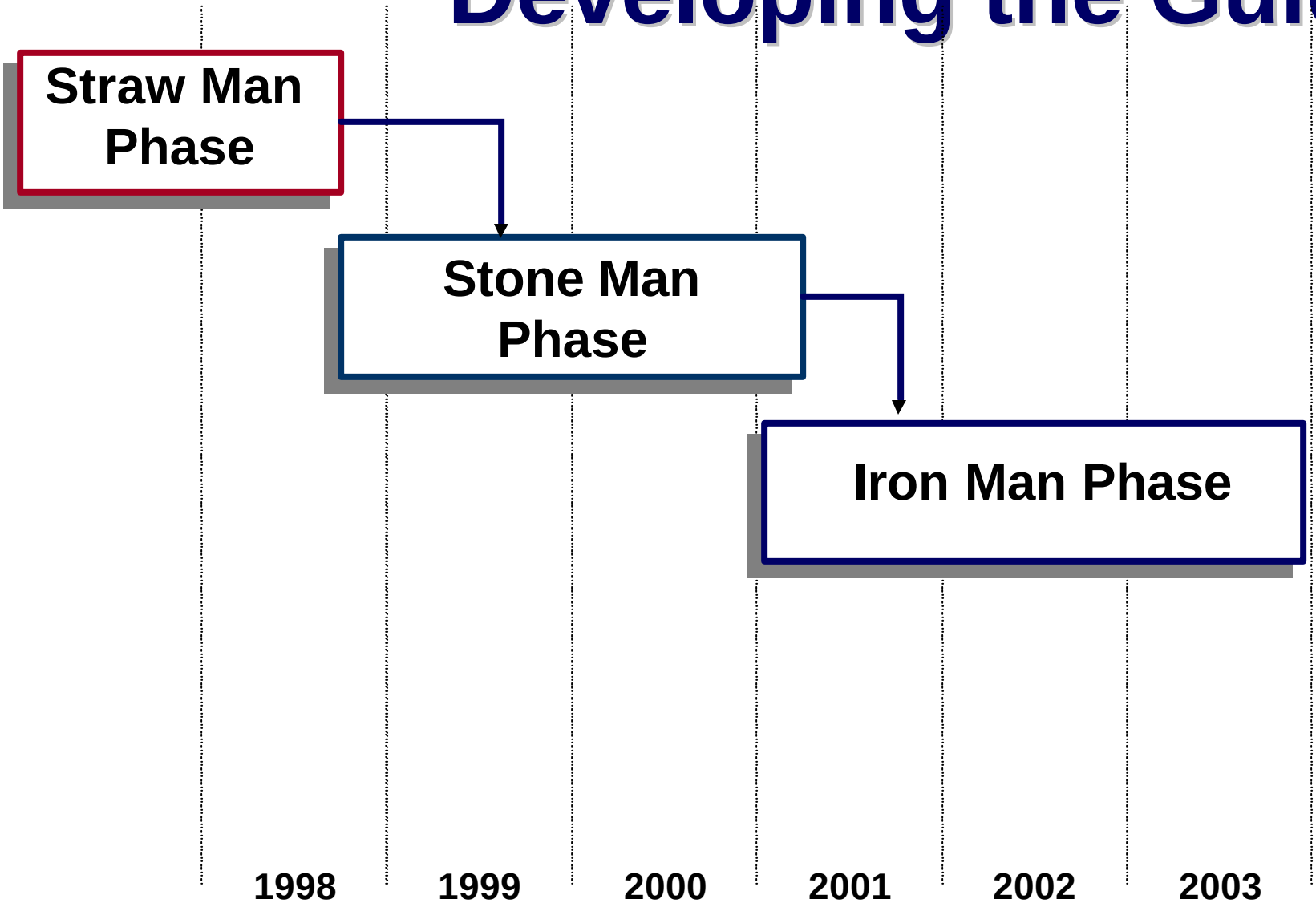
Industrial Advisory Board

- Mario R. Barbacci, SEI, representing the IEEE Computer Society
- Carl Chang, Auburn University, representing Computing Curricula 2001
- François Coallier, Bell Canada, representing ISO/IEC JTC 1 / SC7
- Chuck Howell, The MITRE Corporation
- Anatol Kark, National Research Council of Canada
- Philippe Kruchten, Rational Software Corp.
- Laure Le Bars, SAP Labs (Canada)
- Steve McConnell, Construx Software
- Dan Nash, Raytheon Systems Company
- Fred Otto, Canadian Council of Professional Engineers
- Bryan Pflug, The Boeing Company
- Larry Reeker, National Institute of Standards and Technology

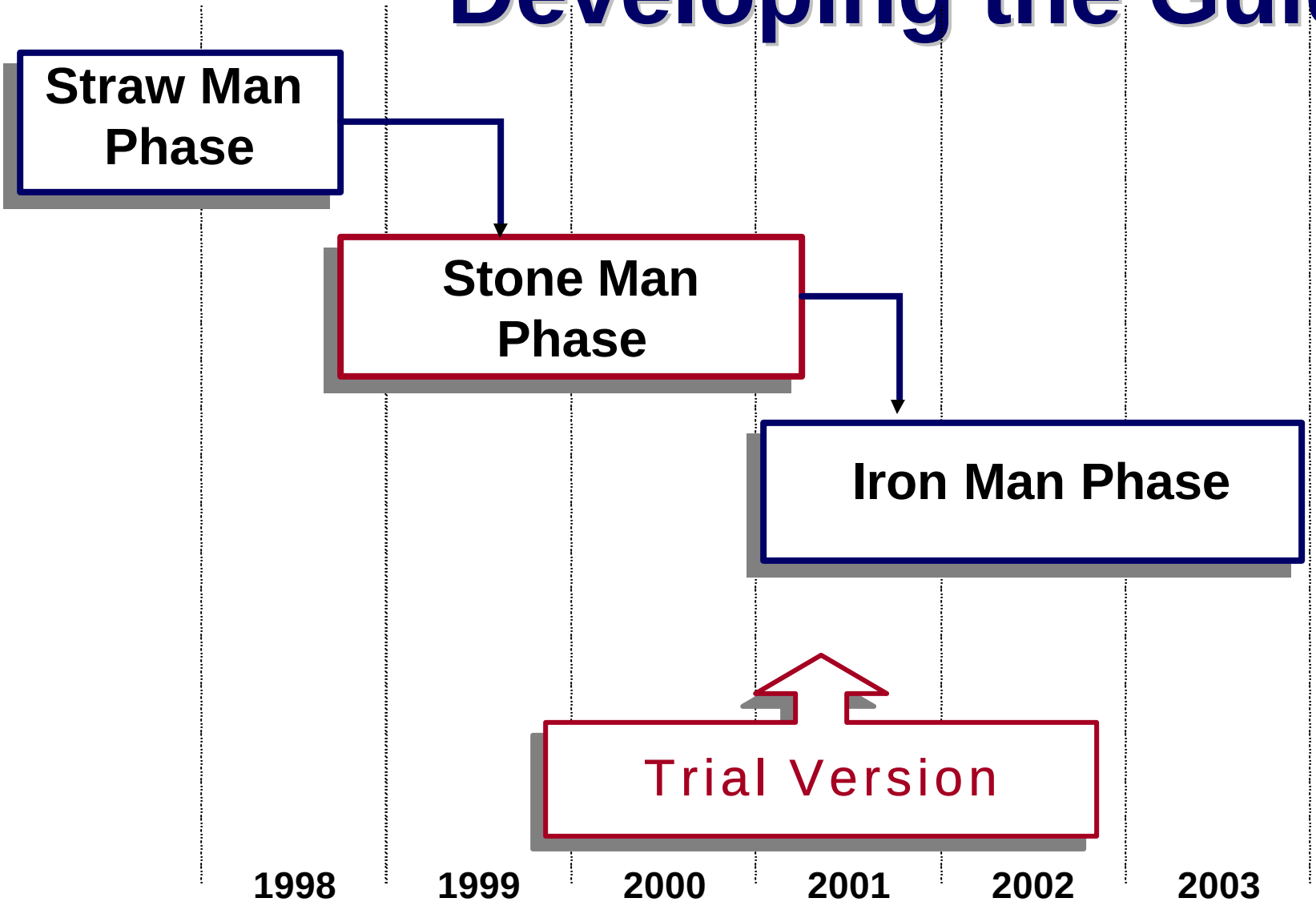
Roles of the Industrial Advisory Board

- Provide input to ensure relevance to various audiences
- Review and approve strategy and deliverables
- Oversee the development process
- Assist in promoting the Guide
- Lend credibility to the project

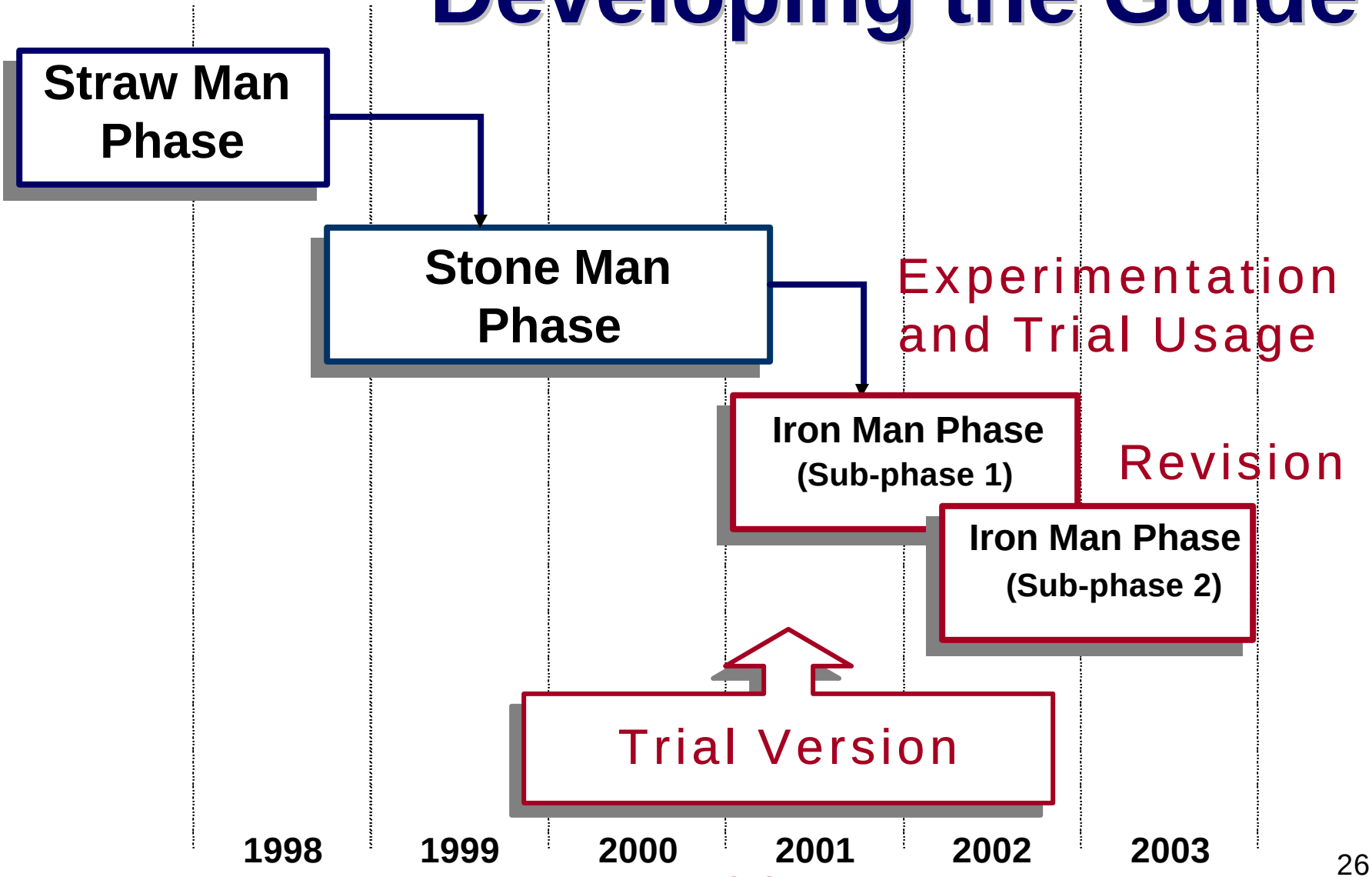
A Three-Phase Approach for Developing the Guide



A Three-Phase Approach for Developing the Guide



A Three-Phase Approach for Developing the Guide



List of Knowledge Areas

- Requirements
- Design
- Construction
- Testing
- Maintenance
- Configuration Management
- Quality
- Engineering Tools & Methods
- Engineering Process
- Engineering Management

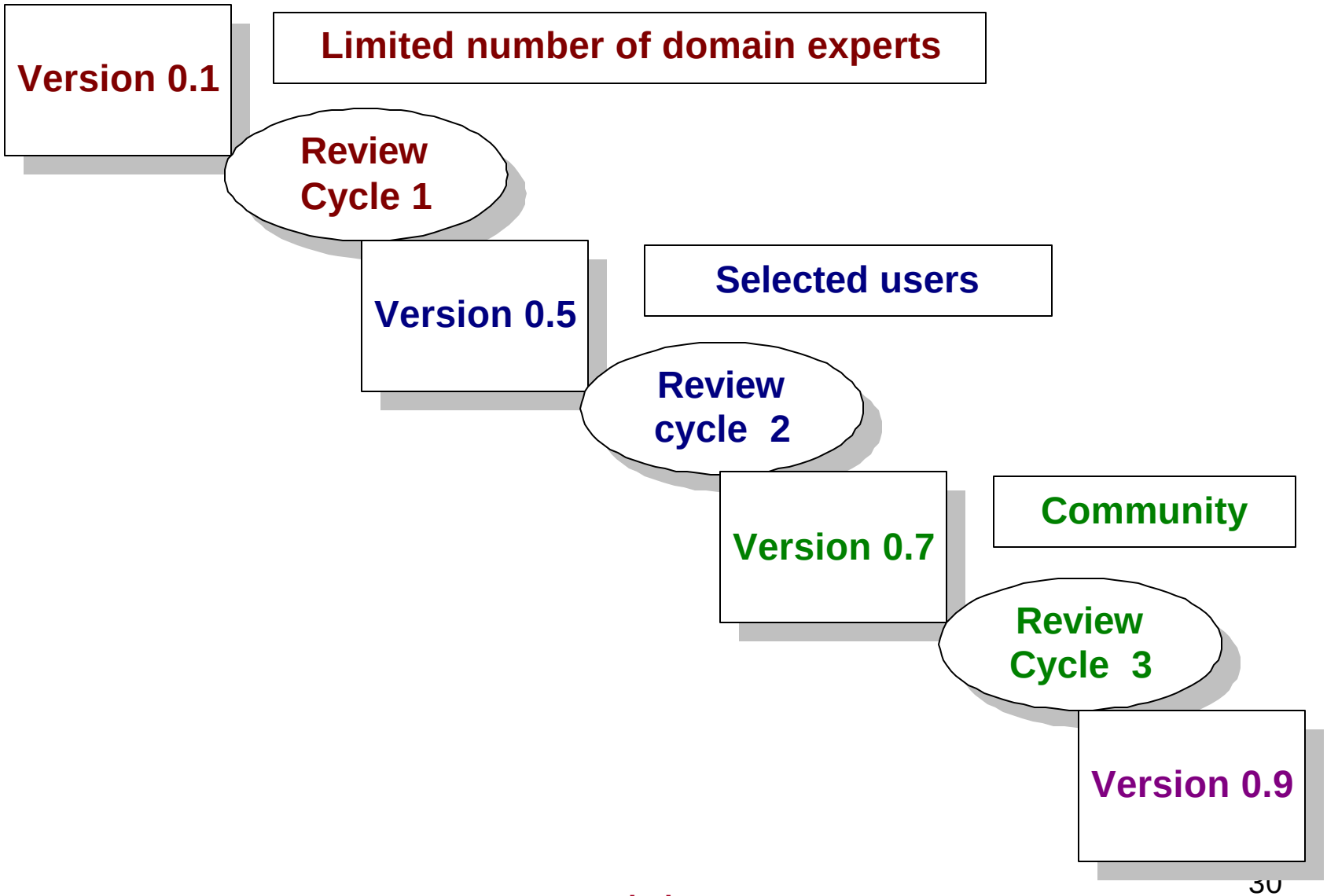
List of Related Disciplines

- Computer Science (CC2001)
- Mathematics (CC2001)
- Project Management (PMBOK)
- Computer Engineering
- Cognitive Sciences and Human Factors
- Systems Engineering
- Management and Management Science

Knowledge Area Specialists

- *Requirements:* Pete Sawyer; Gerald Kotonya, UK
- *Design:* Guy Tremblay, Canada
- *Construction:* Terry Bollinger, USA; Philippe Gabrini, Louis Martin, Canada
- *Testing:* Antonia Bertolino, Italy
- *Maintenance:* Thomas Pigoski, USA
- *Configuration Management:* John Scott; David Nisse, USA
- *Quality:* Dolores Wallace; Larry Reeker, USA
- *Tools and Methods:* Dave Carrington, Australia
- *Process:* Khaled El Emam, Canada
- *Management:* Stephen MacDonell; Andrew Gray, NZ

Phase 2: Stone Man Review Process



Stone Man Review Process

- Transparency and consensus-building
  - All intermediate versions of documents are published and archived on www.swebok.org
  - All comments are made public as well as the identity of the reviewers
  - Detailed comment disposition reports are produced
  - In all, roughly 8000 comments from 500 reviewers in 42 countries

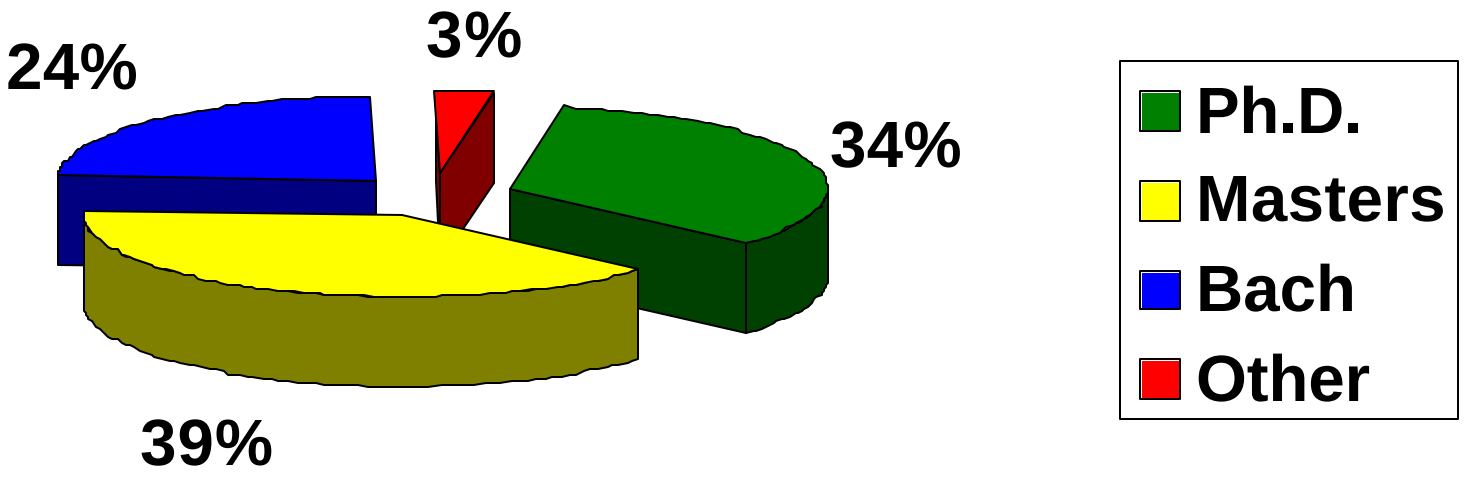
Data on reviewers

- Version 0,1: 33
- Version 0,5: 195
- Version 0,7: 378
  - + ISO reviews from 5 countries

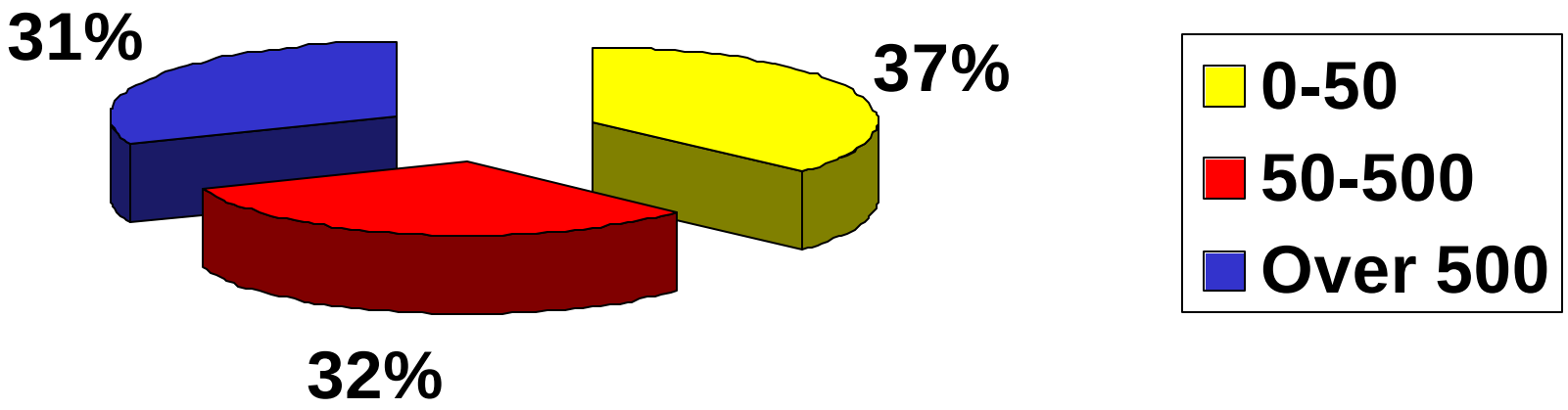
Geographic Distribution of Reviewers (Version 0,7)

- USA: 55%
- Europe: 18%
  - 90 reviewers from 25 countries
- Canada: 10%
- Australia: 5%
- Asia: 5%
- Latin America: 4%

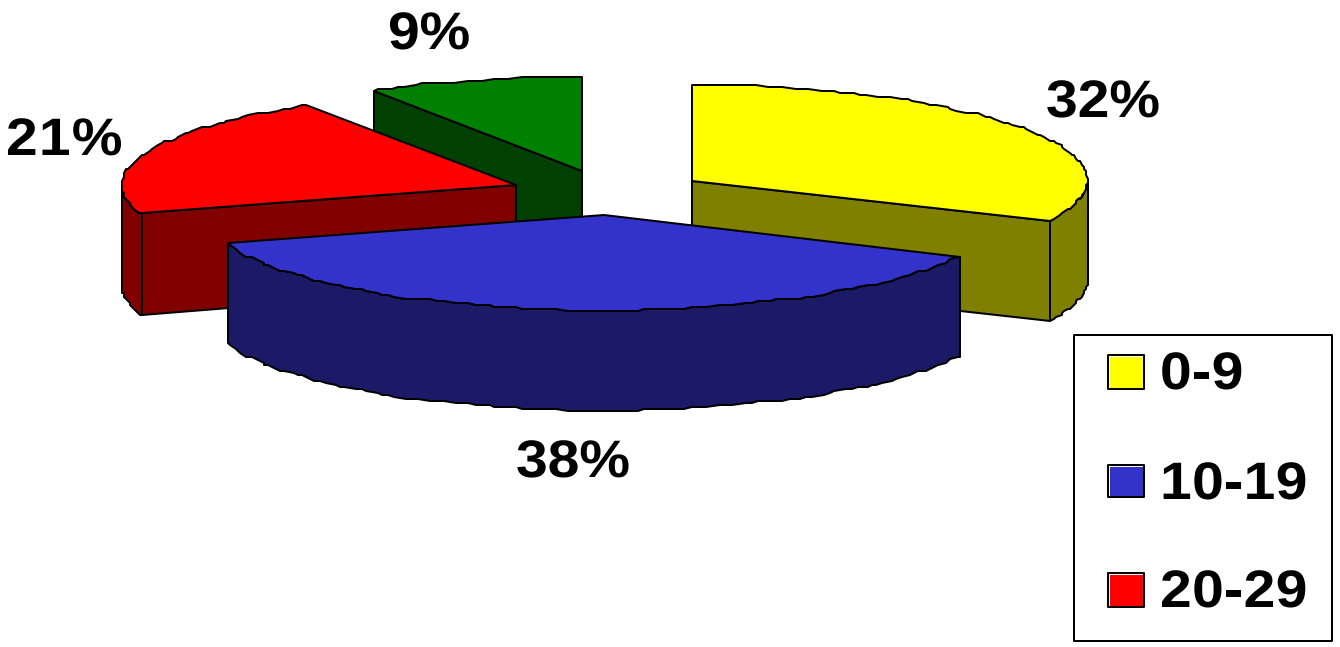
Education level of reviewers (Version 0,7)



Number of employees at reviewer location (Version 0,7)



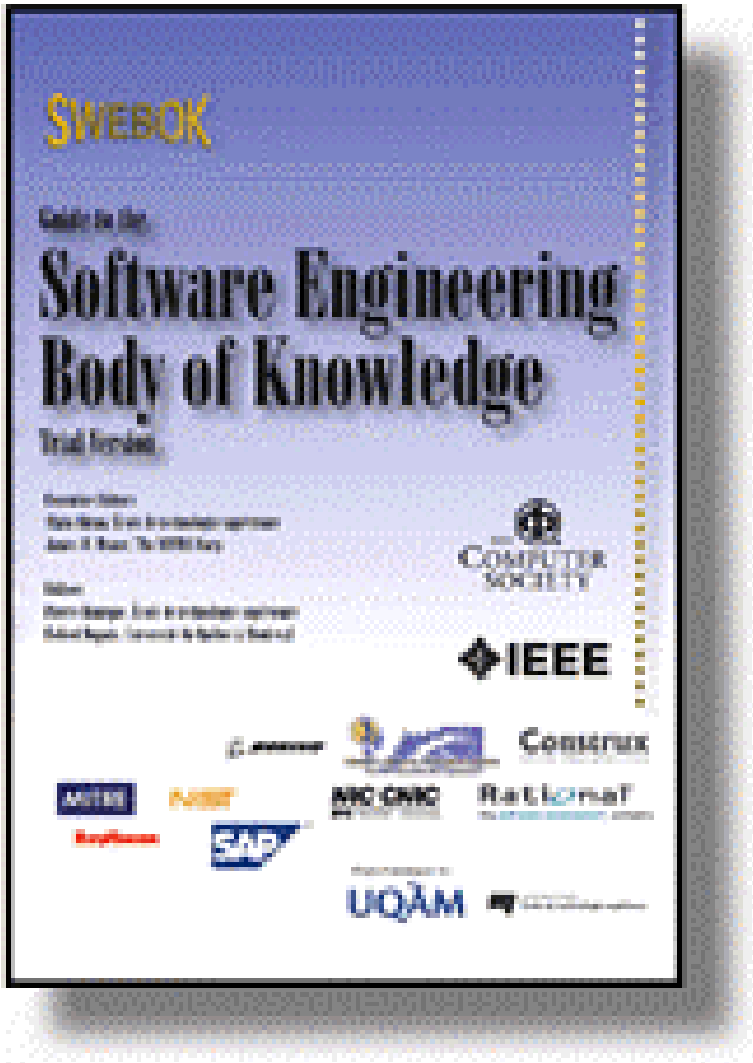
Number of years of practical experience (Version 0,7)



Stone Man Deliverables

- *Trial Version* is available for free on the web
- *Trial Version* was released in hard copy by the IEEE Computer Society Press in December 2001 (for a price...).

Trial version



www.swebok.org

Formal resolutions (Spring 2001)

- Industrial Advisory Board
- CS Board of Governors
  - *"The Board of Governors of the IEEE Computer Society accepts the Guide to the Software Engineering Body of Knowledge (Trial Version) as fulfilling its development requirements and is ready for field trials for a period of two years"*

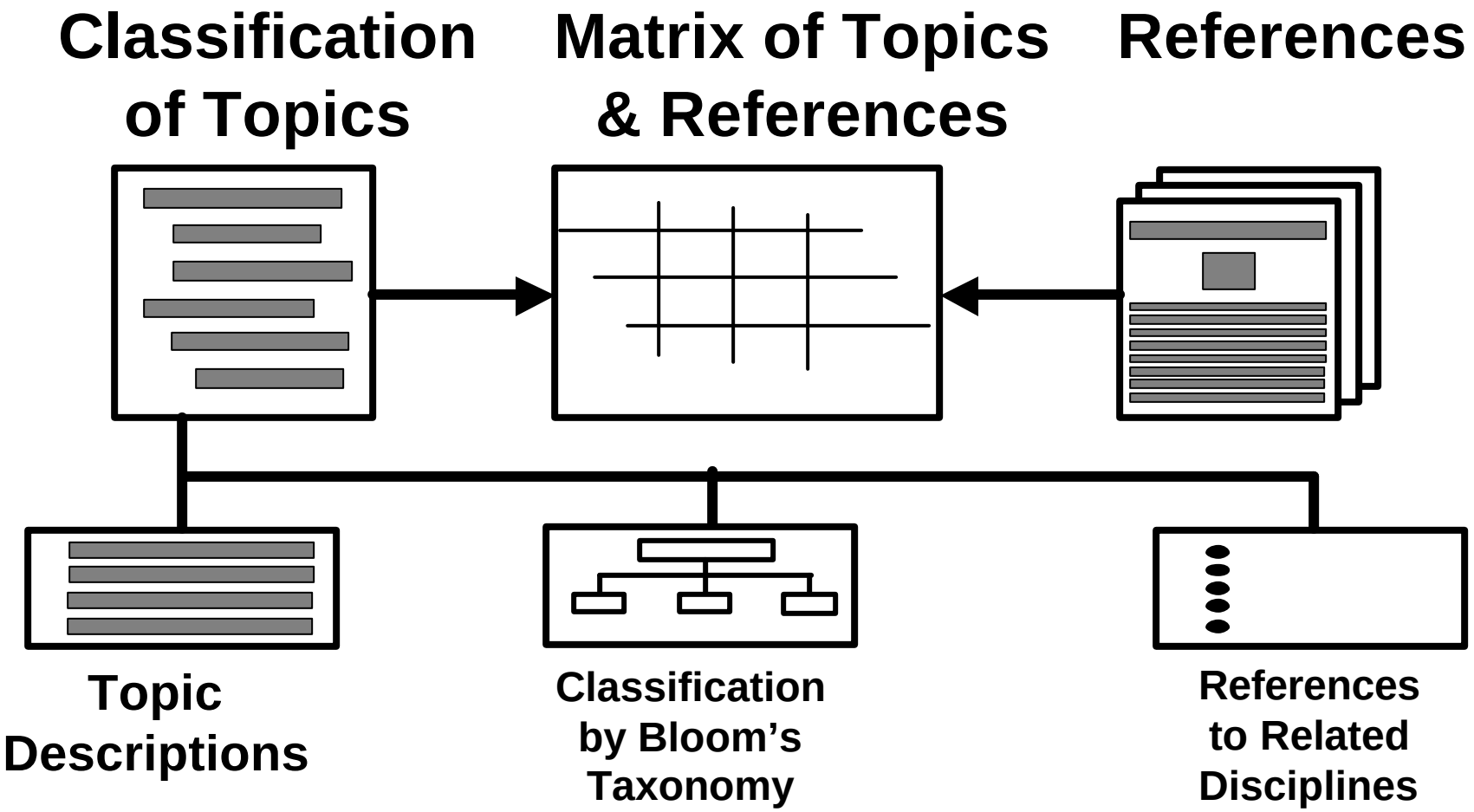
Presentation Plan

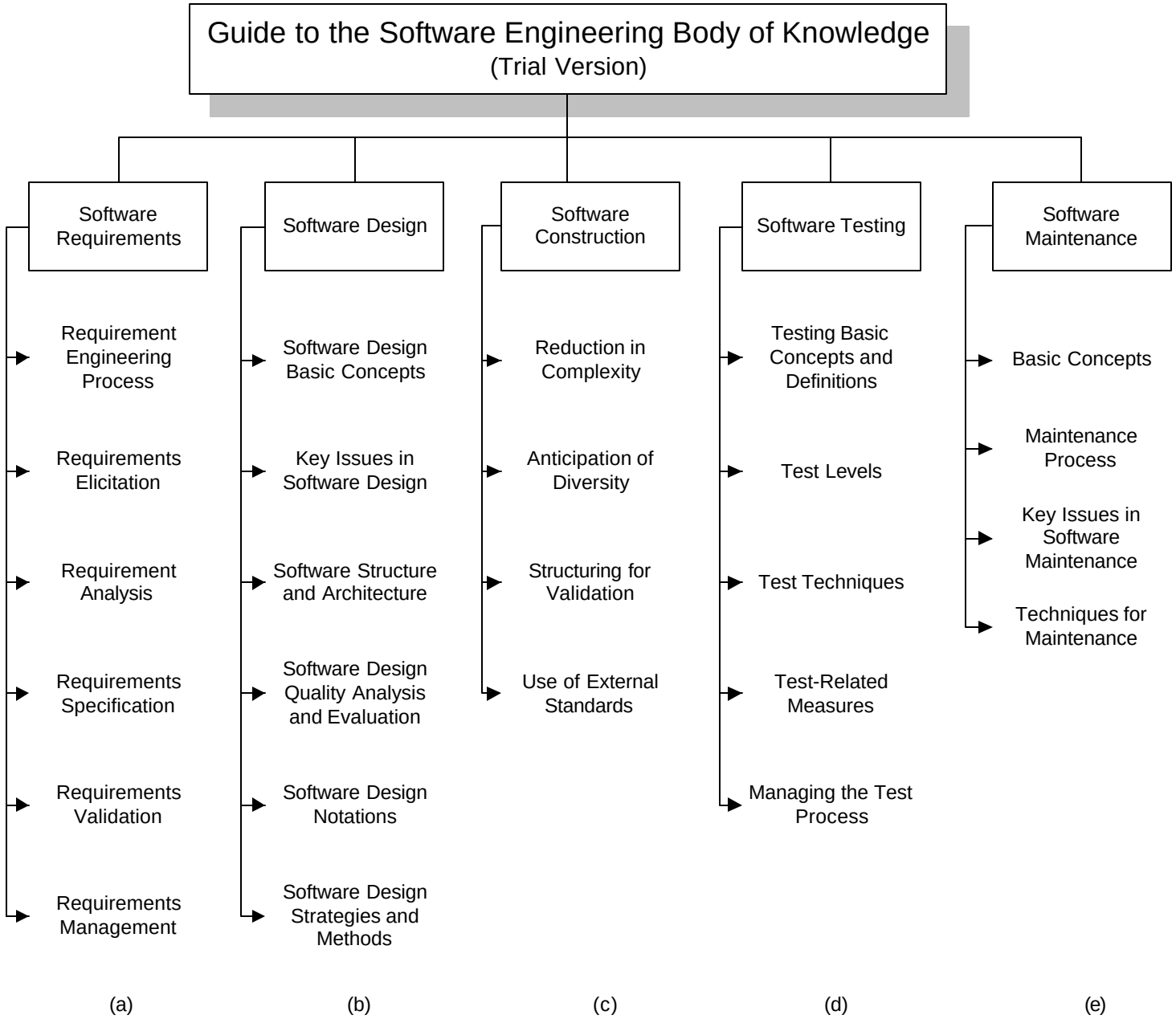
- Project background
- Project scope, objectives, audience and plan
- **Contents of the Guide**
- How you can leverage the Guide within your organization
- Future plans
- Class exercise in applying the Guide
- Conclusions

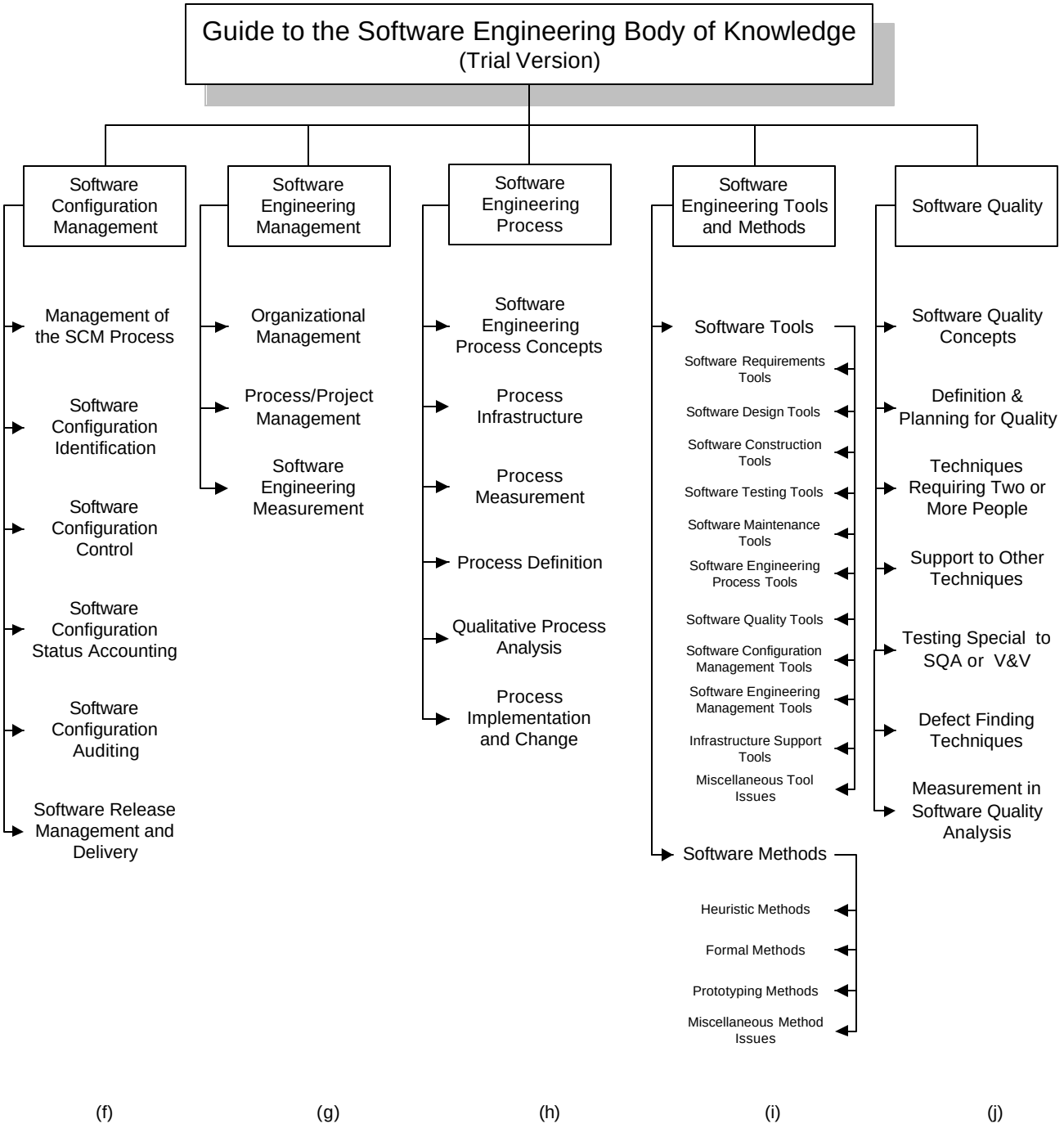
Stone Man Results

- **Consensus** on a list of Knowledge Areas
- **Consensus** on a list of **topics and relevant reference materials** for each Knowledge Area
- **Consensus** on a list of Related Disciplines

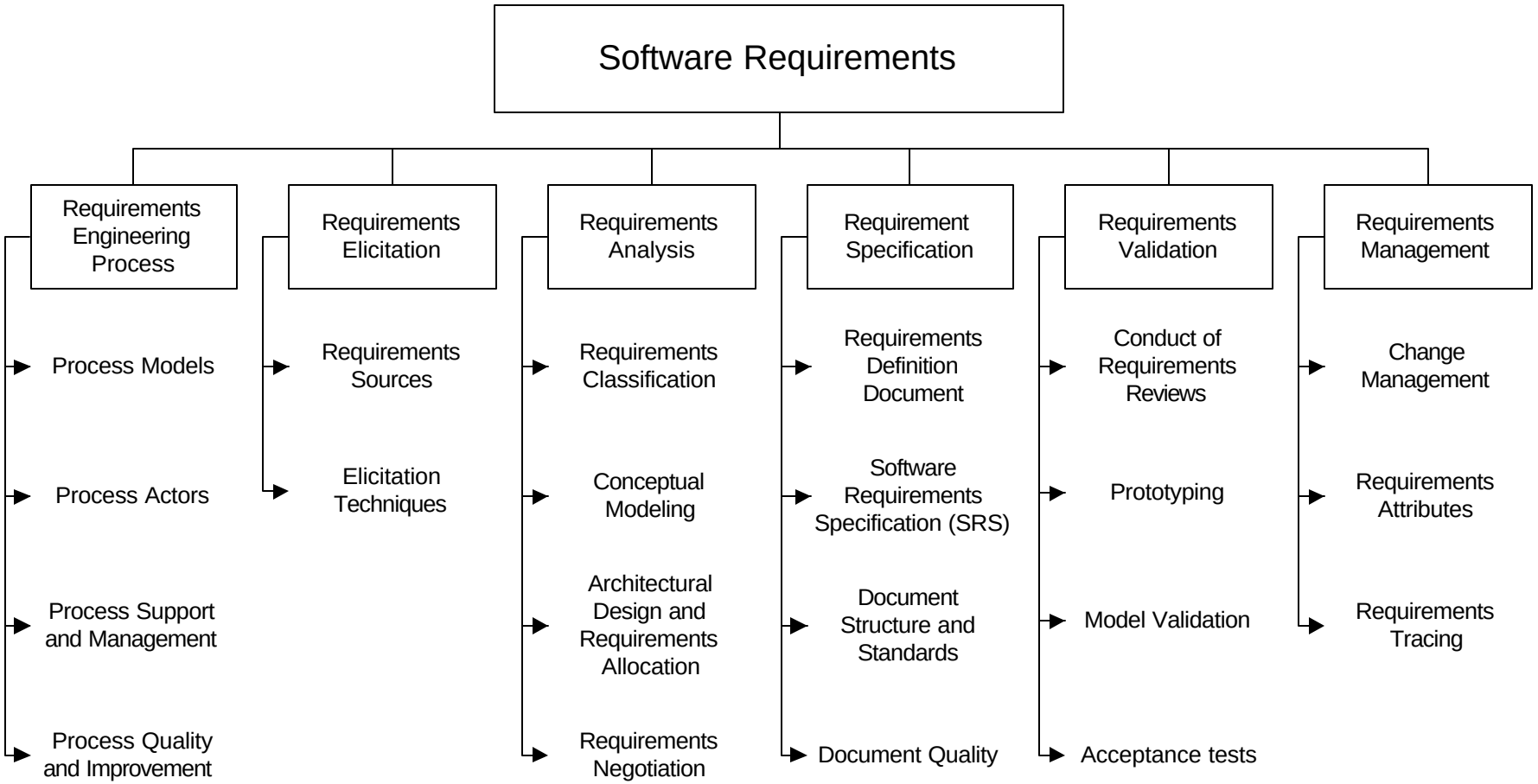
Knowledge Area Description



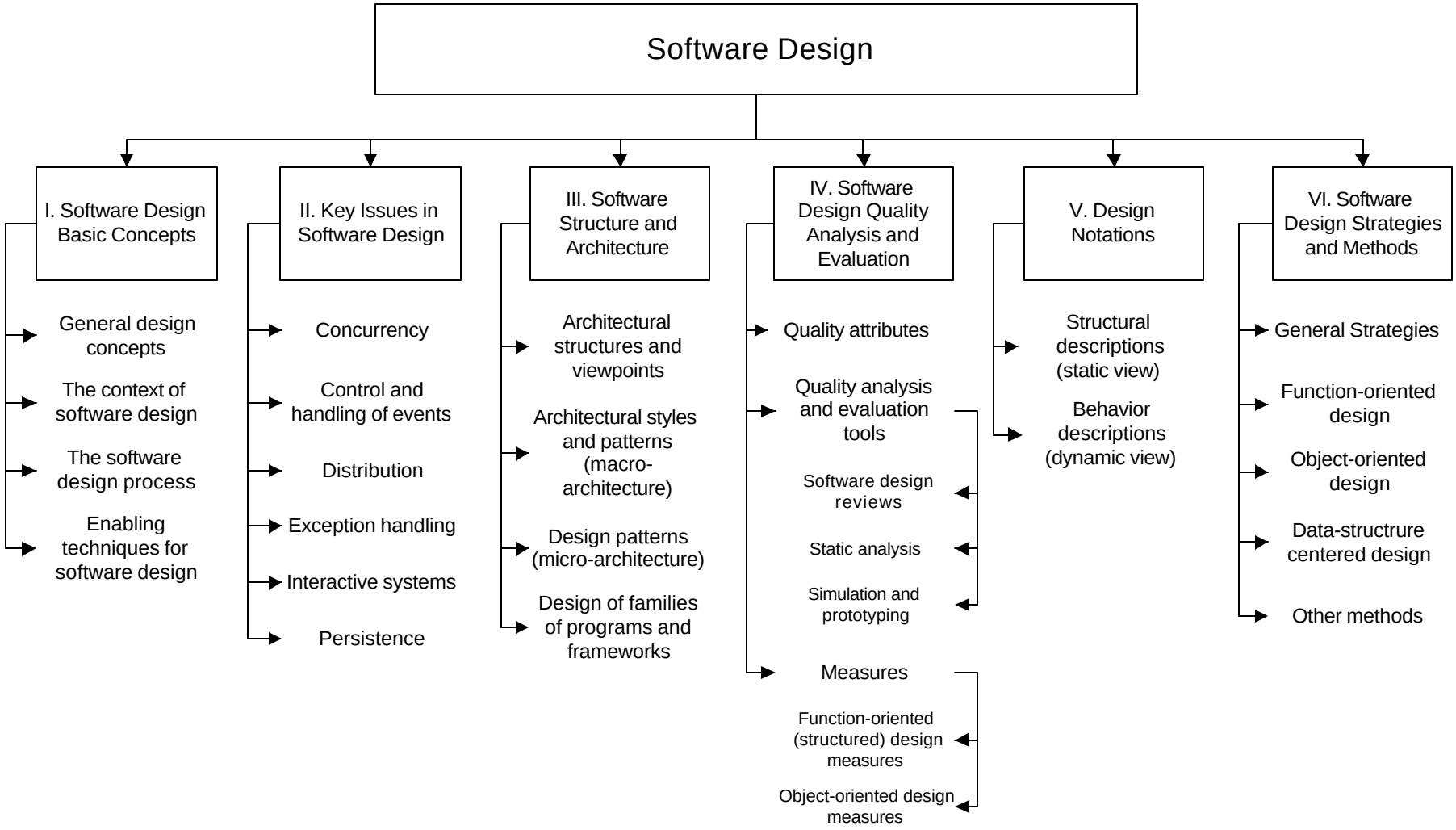

Guide to the Software Engineering Body of Knowledge
(Trial Version)
Software Requirements
Software Design
Software Construction
Software Testing
Software Maintenance
Requirement Engineering Process
Requirements Elicitation
Requirement Analysis
Requirements Specification
Requirements Validation
Requirements Management
Software Design Basic Concepts
Key Issues in Software Design
Software Structure and Architecture
Software Design Quality Analysis and Evaluation
Software Design Notations
Software Design Strategies and Methods
Reduction in Complexity
Anticipation of Diversity
Structuring for Validation
Use of External Standards
Testing Basic Concepts and Definitions
Test Levels
Test Techniques
Test-Related Measures
Managing the Test Process
Basic Concepts
Maintenance Process
Key Issues in Software Maintenance
Techniques for Maintenance
(a)
(b)
(c)
(d)
(e)


Guide to the Software Engineering Body of Knowledge
(Trial Version)
Software Configuration Management
Software Engineering Management
Software Engineering Process
Software Engineering Tools and Methods
Software Quality
Management of the SCM Process
Software Configuration Identification
Software Configuration Control
Software Configuration Status Accounting
Software Configuration Auditing
Software Release Management and Delivery
Organizational Management
Process/Project Management
Software Engineering Measurement
Software Engineering Process Concepts
Process Infrastructure
Process Measurement
Process Definition
Qualitative Process Analysis
Process Implementation and Change
Software Tools
Software Requirements Tools
Software Design Tools
Software Construction Tools
Software Testing Tools
Software Maintenance Tools
Software Engineering Process Tools
Software Quality Tools
Software Configuration Management Tools
Software Engineering Management Tools
Infrastructure Support Tools
Miscellaneous Tool Issues
Software Methods
Heuristic Methods
Formal Methods
Prototyping Methods
Miscellaneous Method Issues
Software Quality Concepts
Definition & Planning for Quality
Techniques Requiring Two or More People
Support to Other Techniques
Testing Special to SQA or V&V
Defect Finding Techniques
Measurement in Software Quality Analysis
(f)
(g)
(h)
(i)
(j)

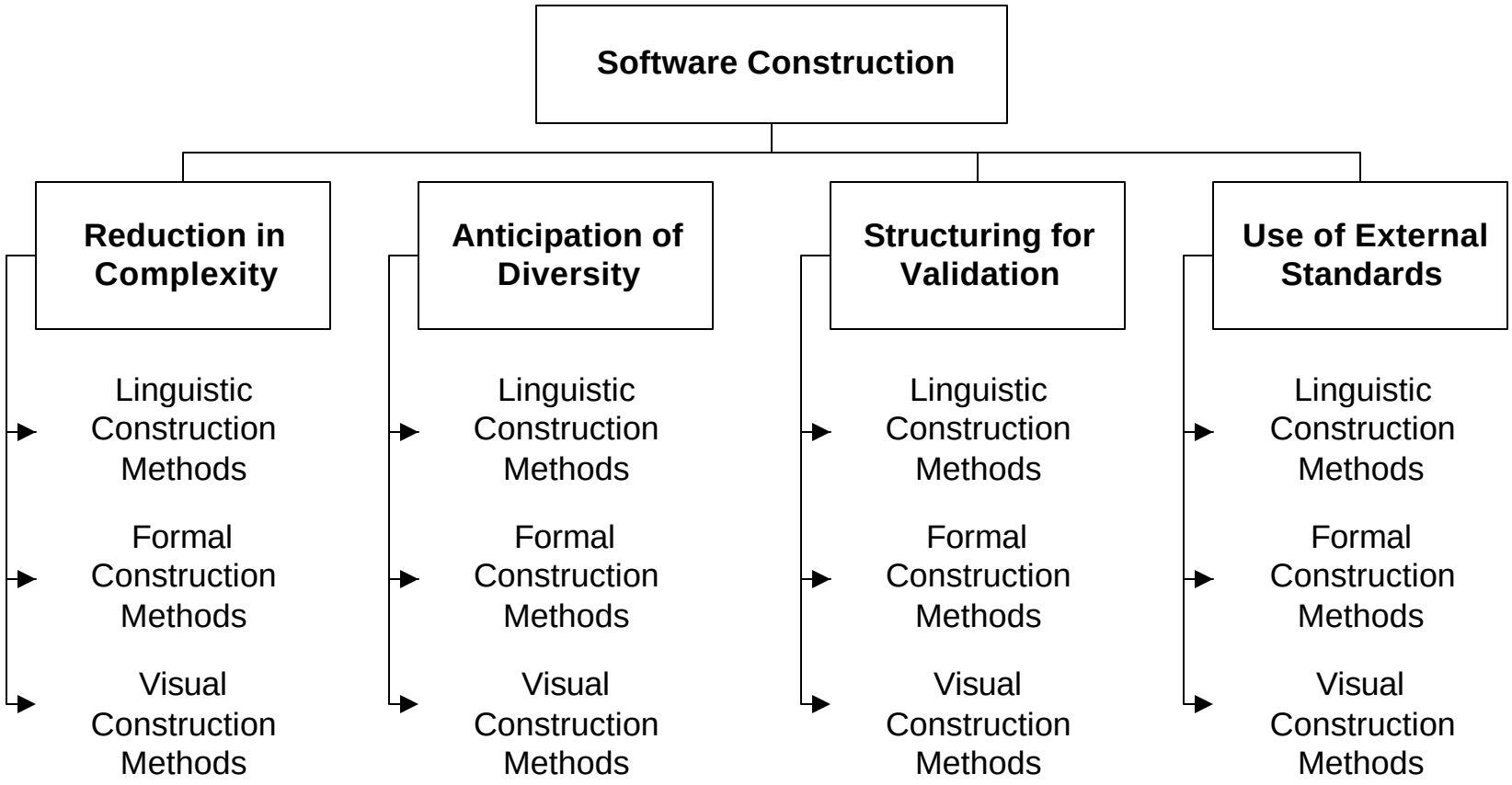
Software Requirements



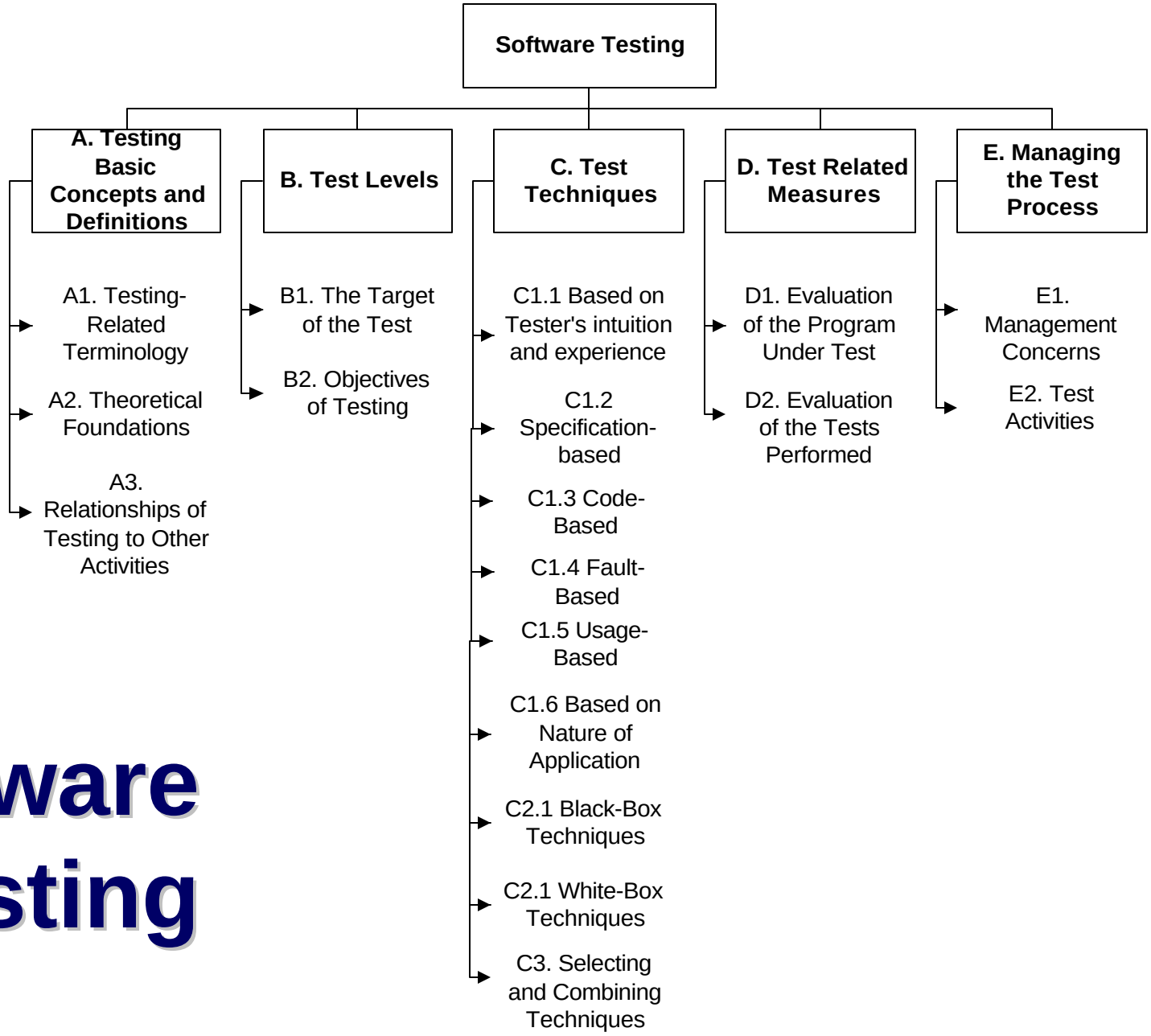
Software Design



Software Construction



Software Testing



Software Testing

- Software testing consists of the **dynamic** verification of the behavior of a program on a **finite** set of test cases, suitably **selected** from the usually infinite executions domain, against the specified **expected** behavior.

Breakdown of Software Testing

- Testing Basic Concepts and Definitions
- Test Levels
- Test Techniques
- Test Related Measures
- Managing the Test Process

A. Testing Basic Concepts and Definitions

A1. Testing-related terminology	Definitions of testing and related terminology
	Faults vs. Failures
A2. Theoretical foundations	Test selection criteria/Test adequacy criteria (or stopping rules)
	Testing effectiveness/Objectives for testing
	Testing for defect removal
	The oracle problem
	Theoretical and practical limitations of testing
	The problem of infeasible paths
	Testability
A3. Relationships of testing to other activities	Testing vs. Static Analysis Techniques
	Testing vs. Correctness Proofs and Formal Verification
	Testing vs. Debugging
	Testing vs. Programming
	Testing within SQA
	Testing within Cleanroom
	Testing and Certification

B. Test Levels

B1. The target of the test	Unit testing
	Integration testing
	System testing
B2. Objectives of testing	Acceptance/qualification testing
	Installation testing
	Alpha and Beta testing
	Conformance testing/ Functional testing/ Correctness testing
	Reliability achievement and evaluation by testing
	Regression testing
	Performance testing
	Stress testing
	Back-to-back testing
	Recovery testing
	Configuration testing
	Usability testing

C. Test Techniques

C1: (criterion "base on which tests are generated")	
C1.1 Based on tester's intuition and experience	Ad hoc
C1.2 Specification-based	Equivalence partitioning
	Boundary-value analysis
	Decision table
	Finite-state machine-based
	Testing from formal specifications
	Random testing
C1.3 Code-based	Reference models for code-based testing (flow graph, call graph)
	Control flow-based criteria
	Data flow-based criteria
C1.4 Fault-based	Error guessing
	Mutation testing
C1.5 Usage-based	Operational profile
	SRET
C1.6 Based on nature of application	Object-oriented testing
	Component-based testing
	Web-based testing
	GUI testing
	Testing of concurrent programs
	Protocol conformance testing
	Testing of distributed systems
	Testing of real-time systems
	Testing of scientific software

C. Test Techniques (Suite)

C2: (criterion "ignorance or knowledge of implementation")	
C2.1 Black-box techniques	Equivalence partitioning
	Boundary-value analysis
	Decision table
	Finite-state machine-based
	Testing from formal specifications
	Error guessing
	Random testing
	Operational profile
	SRET
C2.2 White-box techniques	Reference models for code-based testing (flow graph, call graph)
	Control flow-based criteria
	Data flow-based criteria
	Mutation testing
C3 Selecting and combining techniques	Functional and structural
	Coverage and operational/Saturation effect

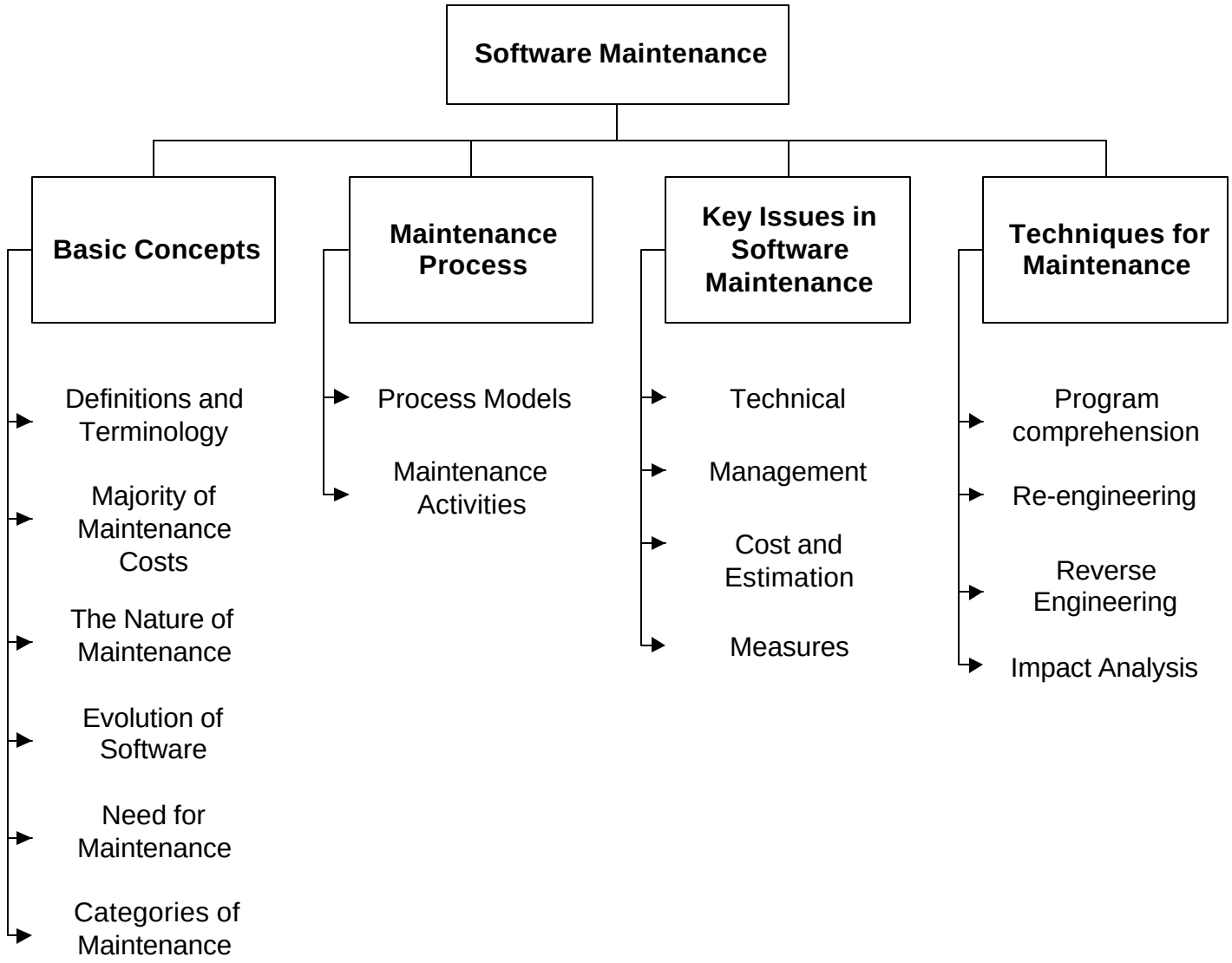
D. Test Related Measures

D.1 Evaluation of the program under test	Program measurements to aid in planning and designing testing
	Types, classification and statistics of faults
	Remaining number of defects/Fault density
	Life test, reliability evaluation
	Reliability growth models
D.2 Evaluation of the tests performed	Coverage/thoroughness measures
	Fault seeding
	Mutation score
	Comparison and relative effectiveness of different techniques

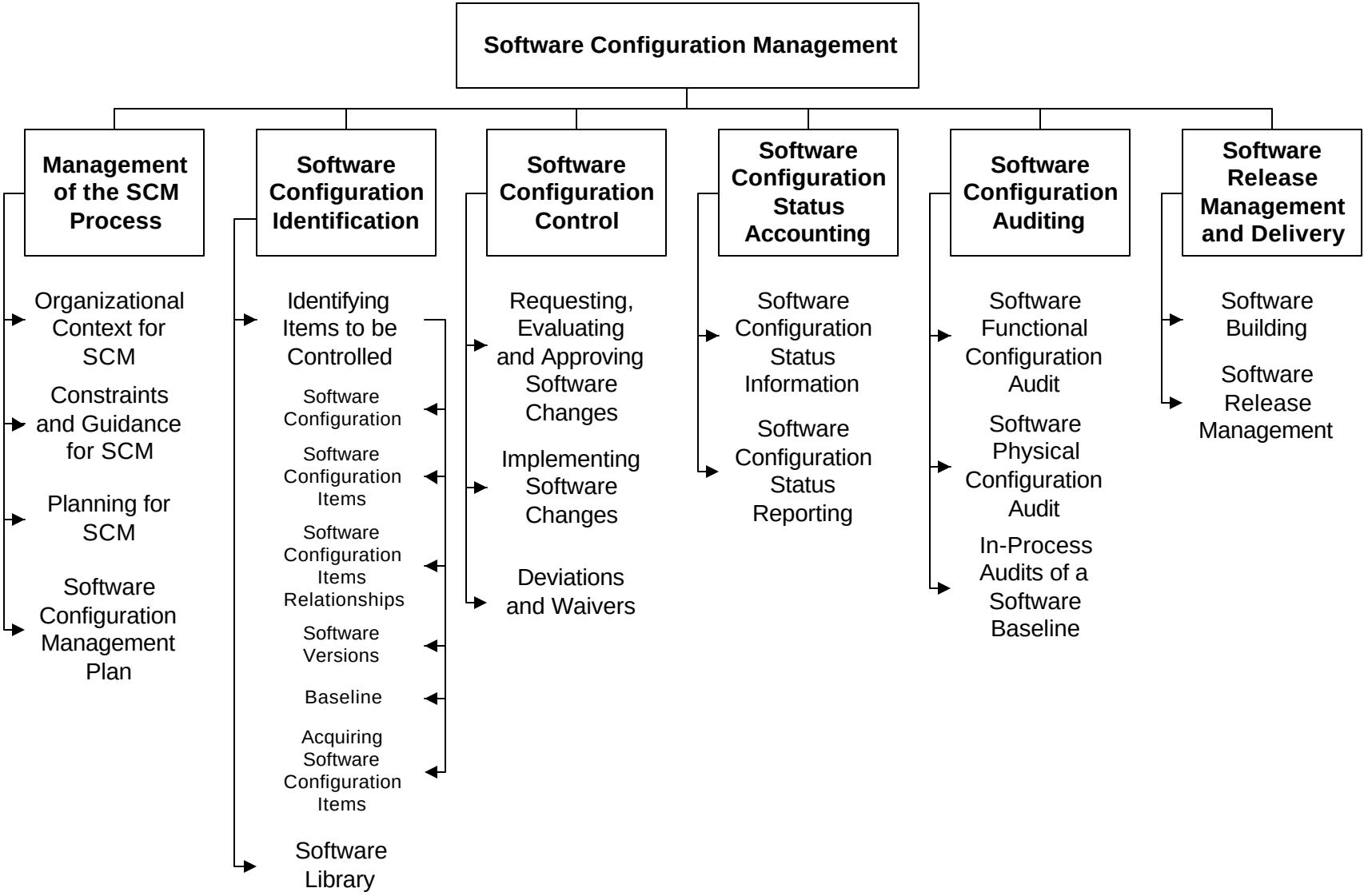
E. Managing the Test Process

E.1 Management concerns	Attitudes/Egoless programming
	Test process
	Test documentation and workproducts
	Internal vs. independent test team
	Cost/effort estimation and other process measures
	Termination
	Test reuse and test patterns
E.2 Test activities	Planning
	Test case generation
	Test environment development
	Execution
	Test results evaluation
	Problem reporting/Test log
	Defect tracking

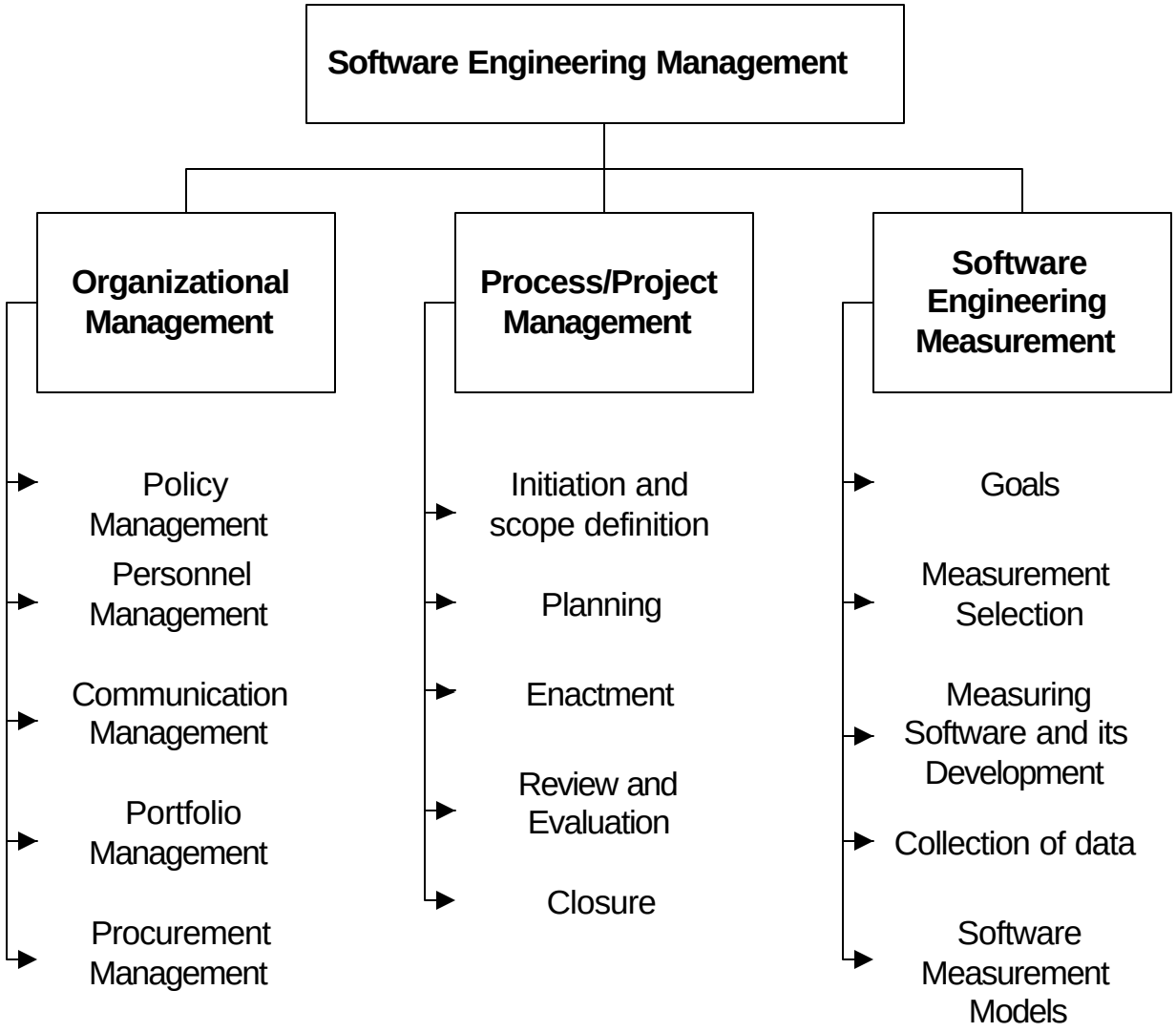
Software Maintenance



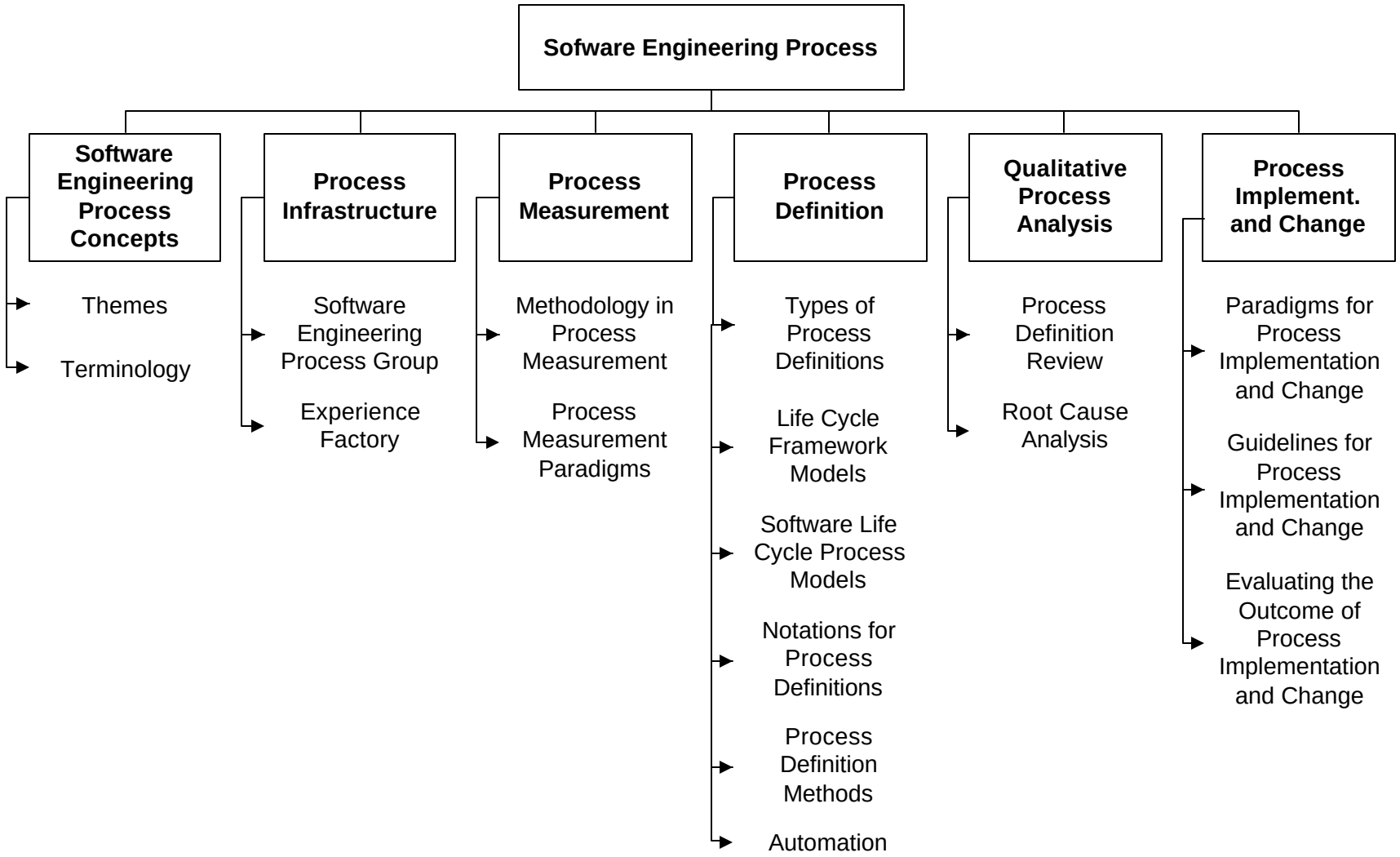
Software Configuration Management



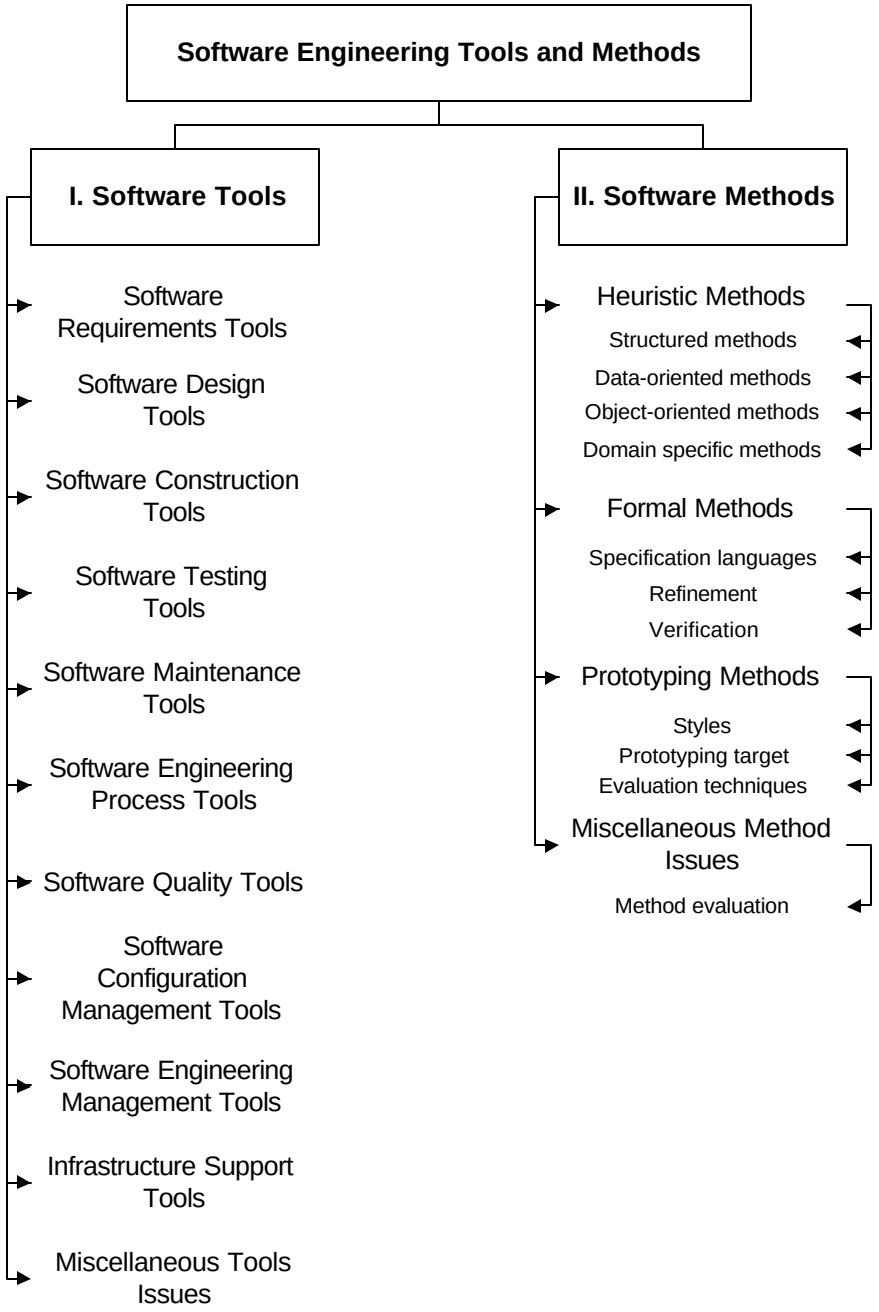
Software Engineering Management



Software Engineering Process

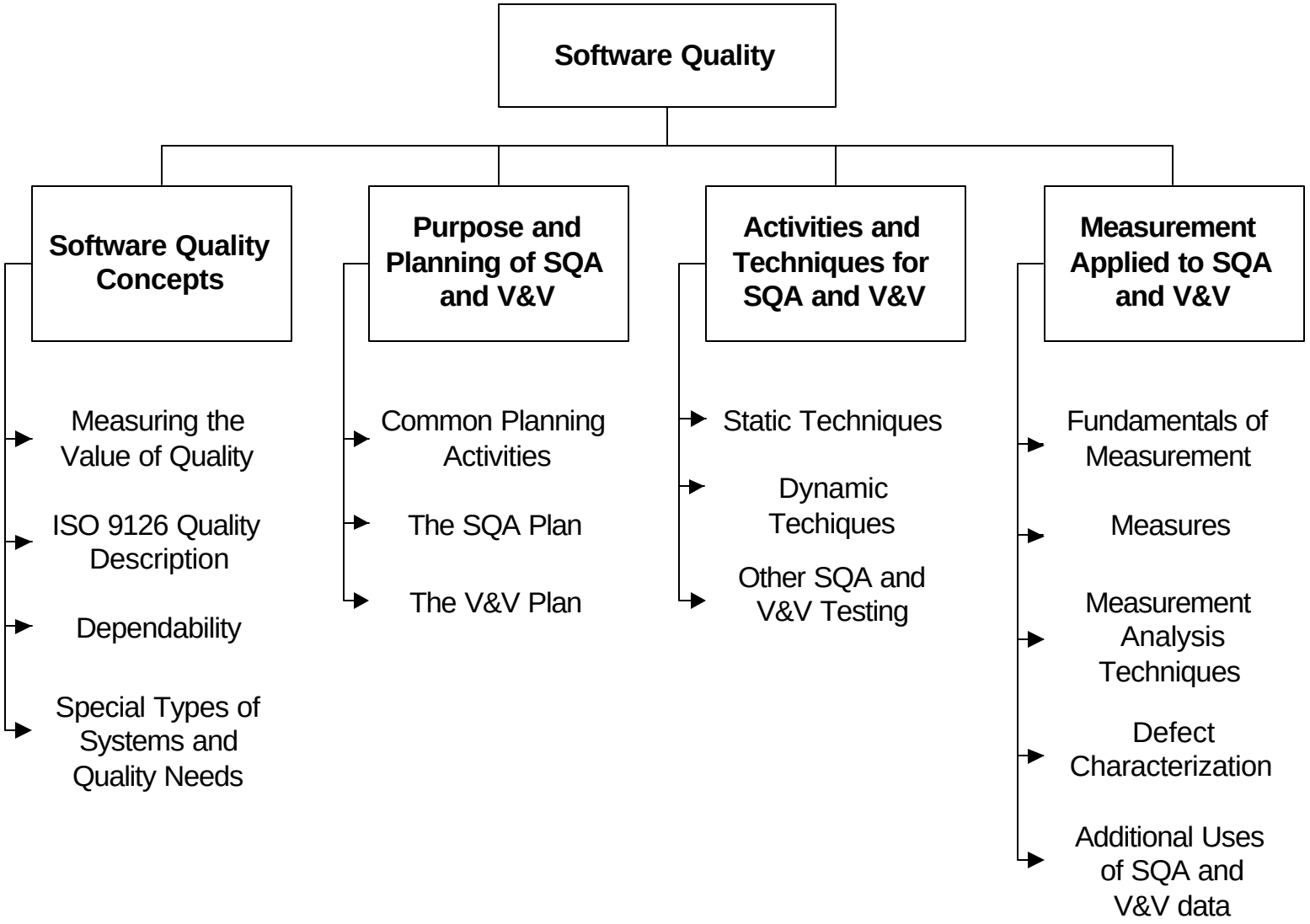


Software Engineering Tools and Methods



www.swebok.org

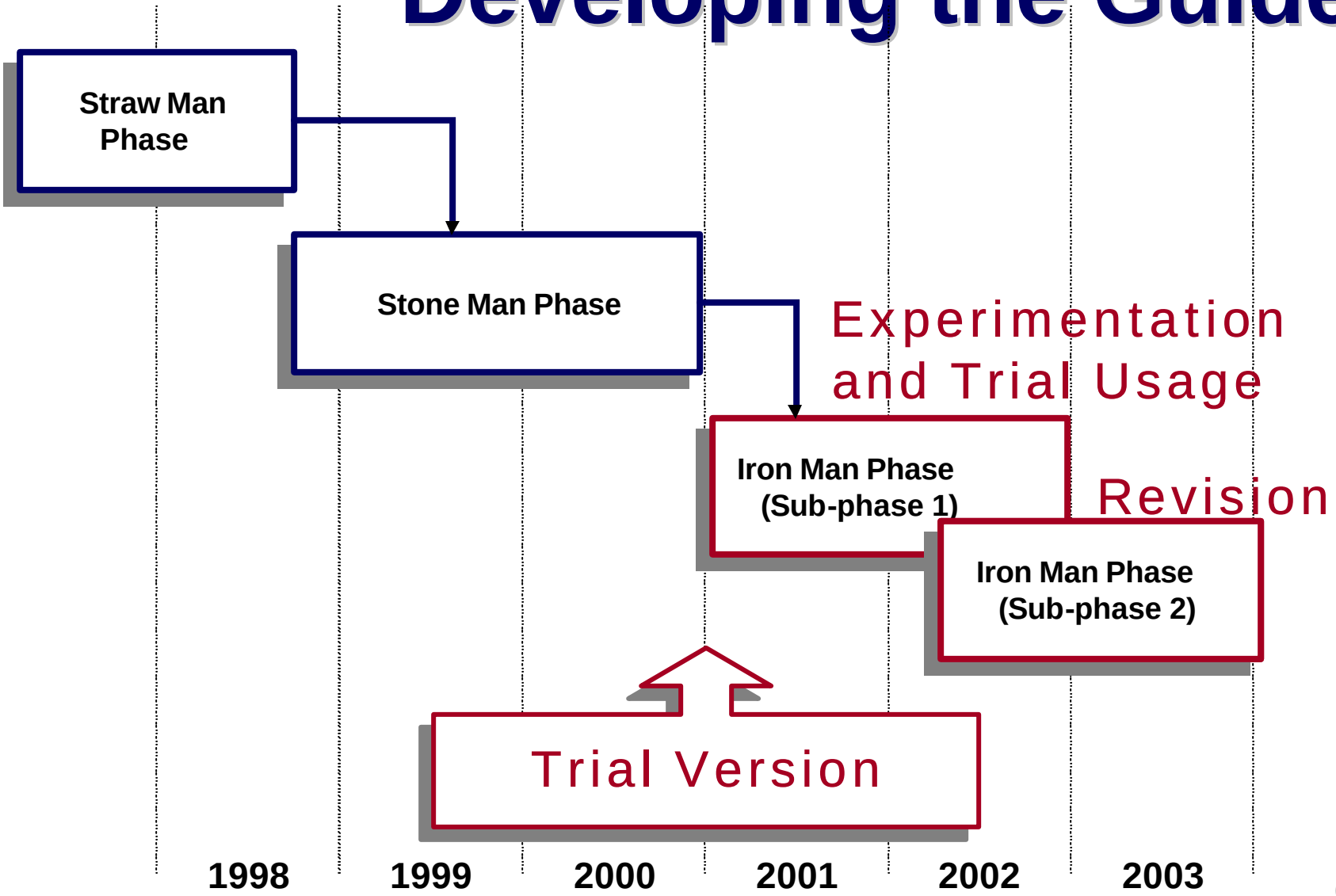
Software Quality



Project Overview Presentation Plan

- Project background
- Project scope, objectives, audience and plan
- Contents of the Guide
- **How you can leverage the Guide within your organization**
- Future plans
- Class exercise in applying the Guide
- Conclusion

A Three-Phase Approach for Developing the Guide



Leveraging the Guide

- Public and Private Organizations
- Makers of Public Policy
- Educators and Trainers
- Researchers

Public and Private Organizations

- Human Resource Management
  - Job descriptions, hiring criteria, project staffing criteria, career planning, contractual staffing criteria, etc.
  - Boeing
  - Lockheed-Martin
  - Brazilian bank

Public and Private Organizations

- Process Models, Policies:
  - Construx
  - Brazilian Bank
  - Major Manufacturing Company

Public and Private Organizations

- Professional Development
  - In-house training
  - Self-evaluation
  - Self-training
  - Example: Construx, "Financial Software Company"

Makers of Public Policy

- Ordre des ingénieurs du Québec
- Canadian Council of Professional Engineers
- Public Policy:
  - Turkish Society for Quality

A01. Yazilim Gereksinimleri

Yazilim gereksinimleri, bir yazilim ürününün amaçlarini basarmak için tasimasi istenen özelliklerdir.

1. **Gereksinim belirleme** tekniklerinden *hangilerini* kullaniyorsunuz?
a. Müsteri iletisim kanallari (yüzyüze görüsmeler, e-posta, çagri merkezi, vb.)
b. Kullanim senaryolari tanimlama
c. Prototip gelistirme
d. Beyin firtinasi toplantilari
e. Kullanim ortaminda gözlem yapma
f. Benzer ürünleri inceleme
g. Diger

2. **Gereksinim tanimlama** asamasinda asagidaki tekniklerden *hangilerini* kullaniyorsunuz?
a. Numaralandirma
b. Müsterinin verdigi öneme göre önceliklendirme
c. Teknik zorluk veya karmasiklik derecesine göre siniflandirma
d. Gerçeklestirme maliyetine göre siniflandirma
e. Degiskenlik derecesine göre siniflandirma
f. Türe göre siniflandirma
g. Ilgili oldugu is sürecini / süreçlerini kaydetme
h. Gereksinimin kaynagi olan kisiyi kaydetme
i. Gereksinimin tanimlandigi tarihi kaydetme
j. Gereksinimler arasindaki bagimliliklari kaydetme
k. Gereksinimin revizyonlarini saklama
l. Gereksinimleri yazilim ürününün planlanan sürümlerine atama
m. Diger

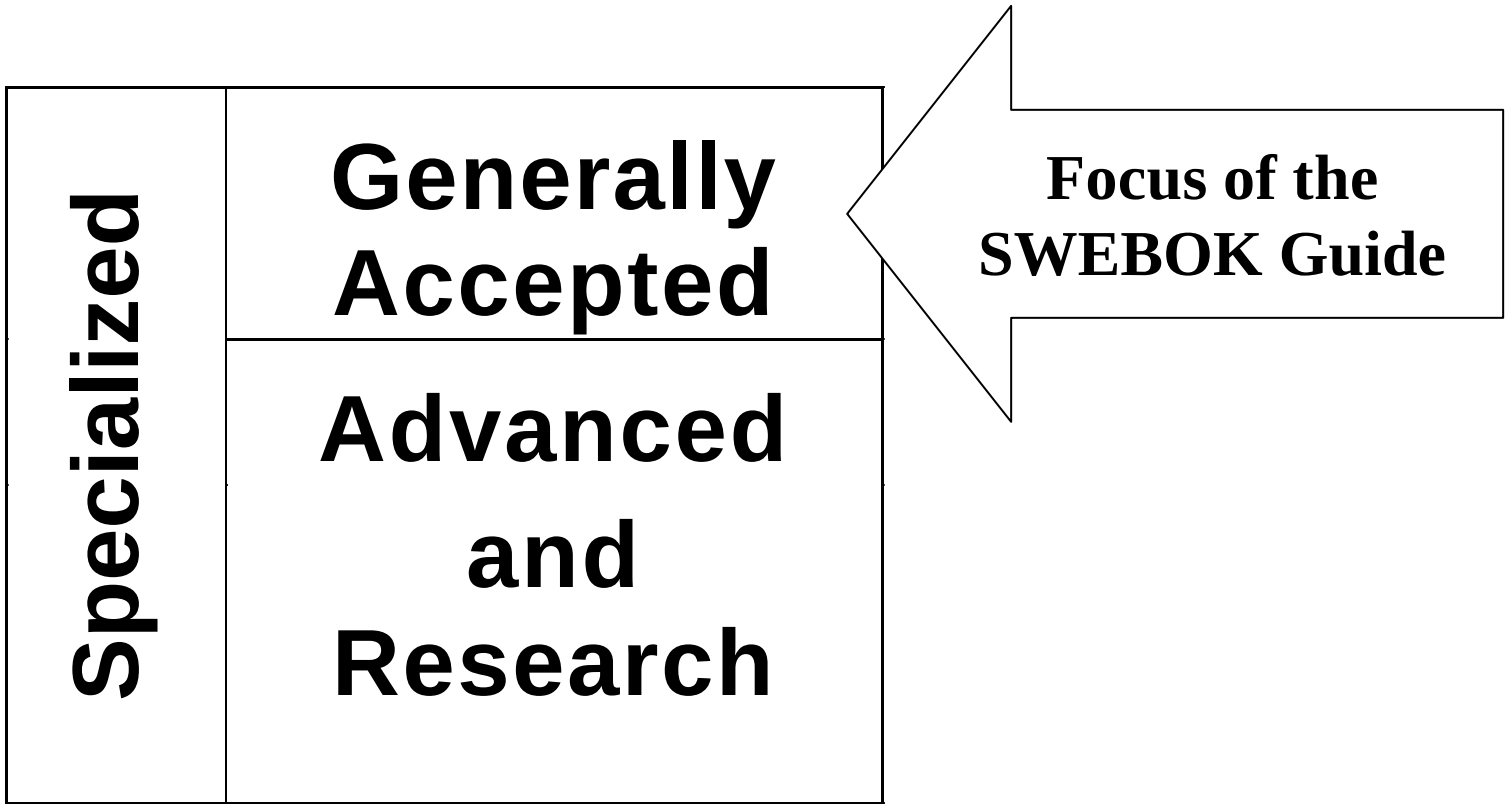
Educators and Trainers

- Course Design/Assessment:
  - Arizona State, etc.
- Program Design/Assessment:
  - University of Iceland
  - Southern Methodist University
  - Stevens Institute of Technology (NJ)
  - National Technological University

Educators and Trainers

- Program Accreditation:
  - Being evaluated in Japan

Categories of Knowledge in the SWEBOK



Researchers: Advanced and Research Topics

- What topics should be monitored as the most likely to become *generally accepted* in the near future ?
- What mechanisms should be used to monitor these and other topics ?

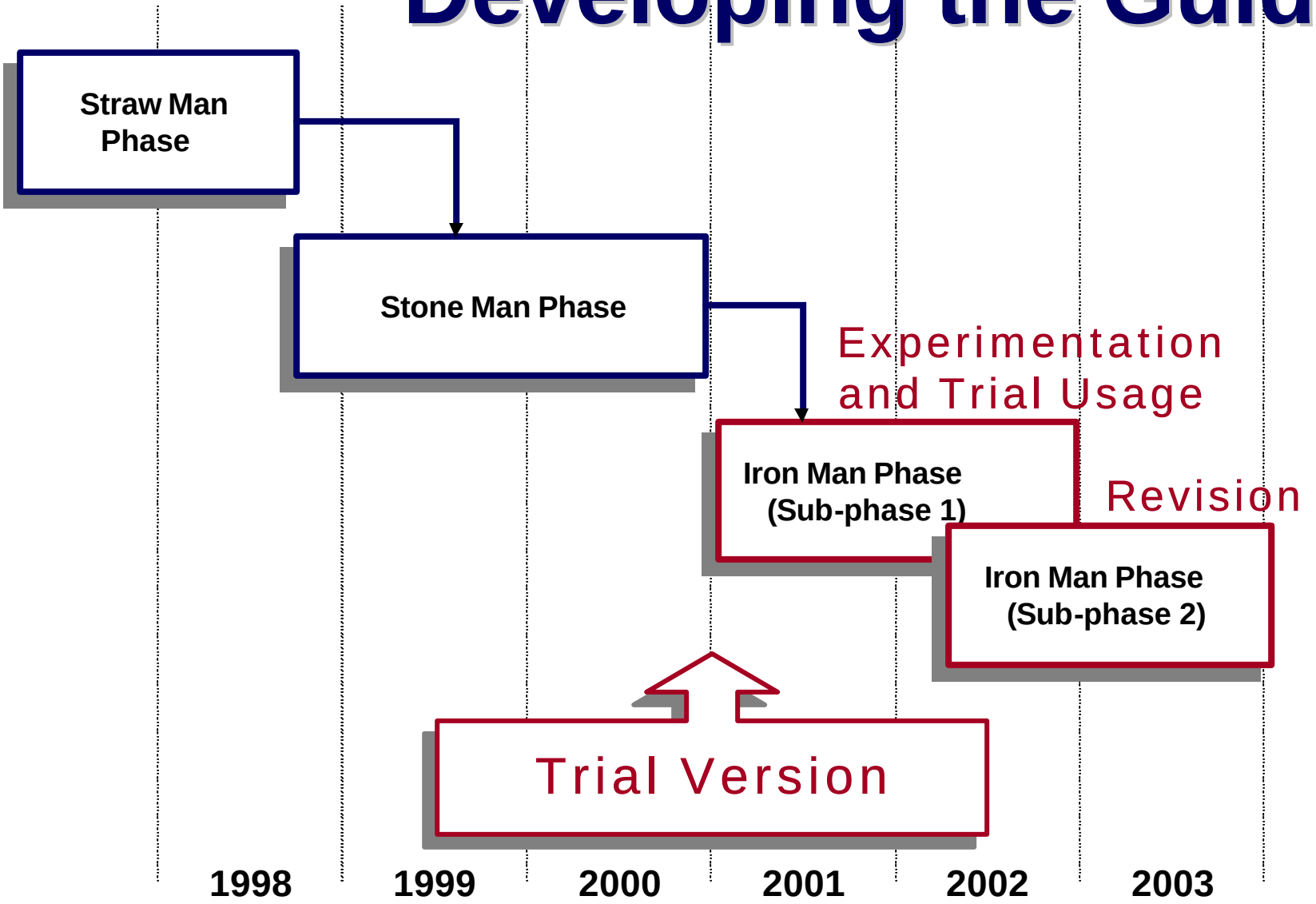
Researchers: Specialized Domains

- What are the most important domains for which knowledge should be included in “extended versions” of the Guide ?
- What characteristics make each of these domains different from the core of software engineering ?
- Do we need additional criteria for recognizing the *generally accepted* knowledge in each of these domains ?

Presentation Plan

- Project background
- Project scope, objectives, audience and plan
- Contents of the Guide
- How you can leverage the Guide
- **Future plans**
- Class exercise in applying the Guide
- Conclusions

A Three-Phase Approach for Developing the Guide



Feedback from Trial Usages

- How to gather feedback from trial usages
- How to structure and make public the gathered feedback

Coordination/Mapping with other projects

- IEEE-CS certification initiative
- CC 2001 curriculum project
- Adoption as an ISO Technical Report
- Canadian accreditation activities
- CMMI/CMM
- ISO 12207
- Task and Performance Norms Project
- …

Ironman sub-phase 2

- Updating the Guide based on gathered feedback and a review process
- Defining an ongoing mechanism for the evolution of the Guide

Presentation Plan

- Project background
- Project scope, objectives, audience and plan
- Contents of the Guide
- How you can leverage the Guide
- Future plans
- **Class exercise in applying the Guide**
- Conclusions

Bloom's Taxonomy

- What level of "knowledge"?
- A hierarchy of educational objectives
- Easy to understand and widely used
- Six levels: knowledge, comprehension, application, analysis, synthesis and evaluation

Bloom's Taxonomy

- **Knowledge**
  - observation and recall of information
  - knowledge of major ideas
- *Typical exam questions to measure this level*:
  - list...
  - define...
  - describe...

Bloom's Taxonomy

- **Comprehension**
  - grasp meaning
  - translate into new context
  - order, group, predict consequences
- *Typical questions:*
  - summarize...
  - contrast...

Bloom's Taxonomy

- **Application**
  - use information
  - use theories in new contexts
  - solve problems
- *Typical questions:*
  - demonstrate...
  - complete...

Bloom's Taxonomy

- **Analysis**
  - see patterns
  - recognize hidden meanings
  - identify components
- *Typical questions:*
  - explain...
  - compare...

Bloom's Taxonomy

- **Synthesis**
  - use old ideas to create new ones
  - generalize from given facts
  - predict
- *Typical questions:*
  - combine...
  - design...

Bloom's Taxonomy

- **Evaluation**
  - compare and discriminate
  - assess value of theories
  - make choices
- *Typical questions:*
  - summarize...
  - grade...

Class Exercise

- Define the knowledge level required for each topic of *Software Testing* for two job descriptions in a contractor-subcontractor situation.
- The job descriptions will be distributed during the tutorial

Class Exercise

- Such a taxonomy could be used for :
  - Improving or writing job descriptions
  - Providing clear objectives for a career development program
  - Identifying training needs
  - Developing training programs and courses
  - …

Presentation Plan

- Project background
- Project scope, objectives, audience and plan
- Contents of the Guide
- How you can leverage the Guide
- Future plans
- Class exercise in applying the Guide
- **Conclusions**

Tutorial Objectives

- Give an overview of the emerging international consensus on the “core body of knowledge” of software engineering
- Explain how you can leverage the SWEBOK Guide within your organization
- Present the evolution of the SWEBOK Guide, the next steps and identify how you can participate

Concluding Remarks

- Consensus on the core body of knowledge is key in all disciplines
- Participation of all communities is important

www.swebok.org

Coordinates

Pierre Bourque

Professor

Ecole de technologie supérieure

Departement de genie electrique

1100, rue Notre-Dame Ouest

Montréal (Québec)

Canada H3C 1K3

Phone: (1) 514-396-8623

Fax : (1) 514-396-8684

pbourque@ele.etsmtl.ca

Robert Dupuis

Université du Québec à Montréal

Computer Science Dept.

C.P. 8888, Succ. Centre-Ville

Montréal, Québec

H3C 3P8 Canada

Tel.: (514) 987-3000 ext. 3479

Fax: (514) 987-8477

dupuis.robert@uqam.ca

Coordinates

James W. Moore

The MITRE Corporation

7515 Colshire Drive

McLean, VA

22102-7508

Tel: 703 883-7396

Fax: 703 883-5432

James.W.Moore@ieee.org