

A new guide for the development and assessment of measurement software

Norbert Greif

norbert.greif@ptb.de

Physikalisch-Technische Bundesanstalt (PTB), Germany

Graeme Parkin

graeme.parkin@npl.co.uk

National Physical Laboratory (NPL), UK



The aim of this talk is:

- To outline the purpose and rationale for a new software guide for measurement software.
- To describe progress on constructing the new guide.

The work so far is being done in collaboration with the Physikalisch-Technische Bundesanstalt (PTB) and the National Physical Laboratory (NPL).

- Why a new guide?
- Purpose of the guide.
- Main aims of the guide.
- Risk-based approach.
- Conclusion / Way forward.

Why a new guide?



- Metrology, including on its top level, relies on software.
- However, reliability of software is more or less an individual matter, although there is no error-free software.
- Same situation with software as with measurement uncertainty decades ago: no common basis for assessment and comparison.
- There does not exist a comprehensive international software guide for measurement scientists and practitioners. At least NMIs would like one.

A matter of metrology



A measurement software guide is a matter of metrology since

- The focus is the reliability of metrological systems.
- Requirements are metrology-driven.
- Metrology must decide about its quality standards and acceptable risks.
- Software technology may provide tools but not the metrological contents.

- Current support:
 - NPL's SSfM Best Practice Guide 1, Validation of Software in Measurement Systems;
 - PTB's system of guidance documents for developing and assessing software
- Current drawbacks (of NPL/PTB guides):
 - Are not international guides/standards
 - Do not cover all aspects of the software lifecycle
 - Have become inconsistent over various revisions

- The purpose of the guide is to enable:
 - developers of measurement software to know what they have to do to produce fit-for-purpose software; and
 - assessors of measurement software to confirm that the developed software is fit-for-purpose.

Fit-for-purpose software: Software that meets domain-specific measurement standards, relevant software quality standards and best software engineering practice.

Purpose of the software guide II



The guide will also

- **Include a glossary** of software terms to provide a common understanding of terminology in software engineering.
- **Give descriptions** of appropriate techniques to be used in the development and assessment of software.
- **Provide risk categories** with corresponding techniques to be used for each risk level.
- **Provide checklists** for developers and assessors.
- **Provide examples.**

- **Who is the guide for?**
 - Includes at least measurement scientists, instrument manufacturers, and testing/calibration laboratories.
- **Structure and type of the guide:**
 - Not just one guide. Will be supplementary guides to cover things like static analysis.
 - Main guide will be developed first using a risk-based approach.
 - Guide will be practical, short, and self contained.

- **Types of measurement software covered:**
 - All types of measurement software including COTS, embedded software, control of instruments, mathematical processing and GUI.
- **Relationship with international standards:**
 - Will relate to these where necessary.
 - Tracing back procedures, descriptions, requirements, recommendations.
 - Examples: ISO/IEC 12207 (15504), IEC 61508, ISO/IEC 25000 series, ISO/IEC 27005.

- **Process view versus product view?**
 - Process view: Gathering evidence during software development via preventive process audits.
 - Product view: Analytical testing of the final (or some intermediate) software product / system.
 - Both aspects are inseparable, they supplement each other.
 - Guide will consider both aspects, and will concentrate on the process of providing evidence that the software product is fit-for-purpose.

- **Software lifecycle:**

- To serve as a base for the guide, a software process reference model is being derived from ISO/IEC 12207.
- Only essential key process areas are considered.

Essential software processes



Requirements Analysis

Software Design

Software Implementation

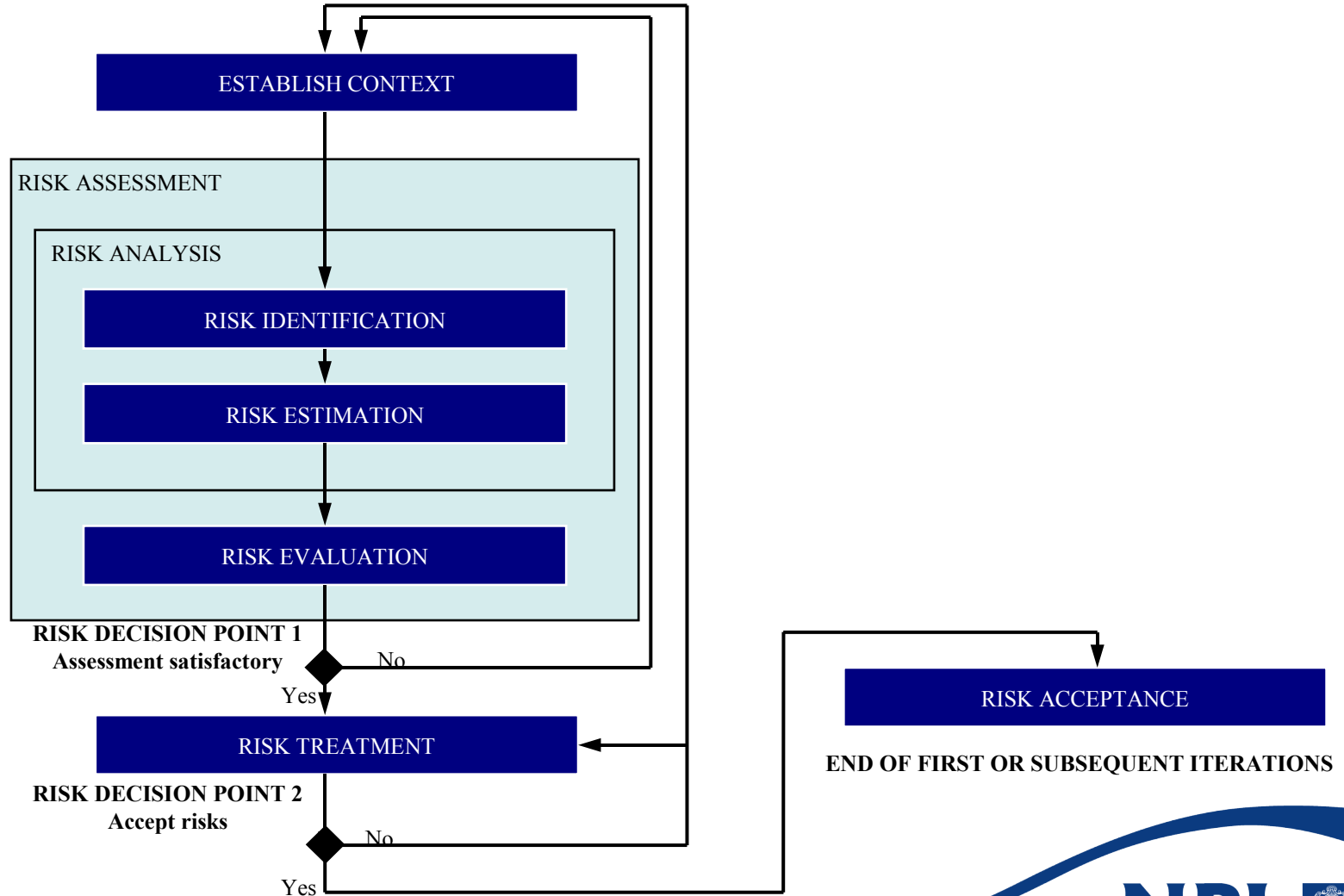
Software Testing

Operation/Maintenance

- **Software lifecycle continued:**
 - Structuring the software lifecycle helps to categorise the diversity of recommended techniques and process requirements.
 - Will simplify the selection of recommended techniques and process requirements.

- Guide will provide a risk assessment procedure based on ISO/IEC 16085 and ISO/IEC 27005
 - Widely accepted approach
 - To define risk categories: risk factors, risk levels
 - To define measures to minimise the risks

Risk management process related to ISO/IEC 27005



- Will define risk categories based on measurement software specific risk factors with appropriate risk levels.
- Will provide measurement software oriented characteristics for each risk category.
- Risk factors are mapped to a unified risk index: Measurement Software Index (MSI).
- For each risk category (MSI level) say what techniques should be used and also level of activity for each process of the software lifecycle.

- Risk factors (restricted to 3):
 - Level of **control complexity**
 - Level of **processing complexity**
 - Level of **system integrity** (safety, security, and environmental issues)
 - Can be expanded by further domain-specific aspects if needed.
- Risk levels:
 - Technical assumption: Number of risk levels for each basic risk factor is restricted to 4.

- To determine the relevant risk category of a software product, measurement oriented characteristics have to be defined for each risk factor and for each risk level.
- Examples for risk factor “control complexity”:
 - Impact of software control functions on measurement processes.
 - Influence of the software on measurement results.
 - Number and complexity of software interactions with other software/hardware subsystems.

- To keep simple the risk management procedure, the risk factors are mapped to a unified risk index → MSI
- Technical assumption: Number of risk level for the general MSI is restricted to 5 (0 to 4).

Measurement software index (MSI)



System Integrity	Processing Complexity	Control Complexity			
		VL	L	H	VH
Very Low (VL)	VL	?			
	L				
	H				
	VH				
Low (L)	VL		?		
	L				
	H				
	VH				
High (H)	VL			?	
	L				
	H				
	VH				
Very High (VH)	VL				?
	L				
	H				
	VH				

- The guide will provide assignments of appropriate techniques and process requirements to be used to the MSI levels for each of the selected lifecycle processes.
- Are being developed:
 - a list of practical development and assessment techniques.
 - a list of appropriate process requirements.
 - assignments of the selected items.

Recommended techniques: Design

Techniques	MSI Levels				
	0	1	2	3	4
Data Flow Modeling	X				
Control Flow Mod.			X		
Entity Relationship			X		
UML				X	
Z					X
State Transitions					X
Petri Nets					X
...					...

?

?

Essential software processes



Requirements Analysis

Software Design

Software Implementation

Software Testing

Operation/Maintenance

Recommended techniques: Testing

Techniques	MSI Levels				
	0	1	2	3	4
Walkthrough			X		
Review		X			
Formal Inspection				X	
Static Analysis			X		
Black-box Testing			X		
White-box Testing				X	
Formal Verification					X
...					...

?

?

- Risk factors and corresponding risk levels have been proposed.
- For each risk factor/risk level, a set of measurement software oriented characteristics has been drafted.
- Proposal of MSI levels for all risk categories has been elaborated.
- Assignments of techniques to be used to the MSI levels are being developed.

- There is the need for an international software guide for metrologists or measurement scientists.
- Fundamental assumptions have been agreed between PTB and NPL and are being proposed.
- The concept of the guide is being jointly developed by PTB and NPL.
- Want to consult as widely as possible and get it accepted as an international guide if possible through BIPM.