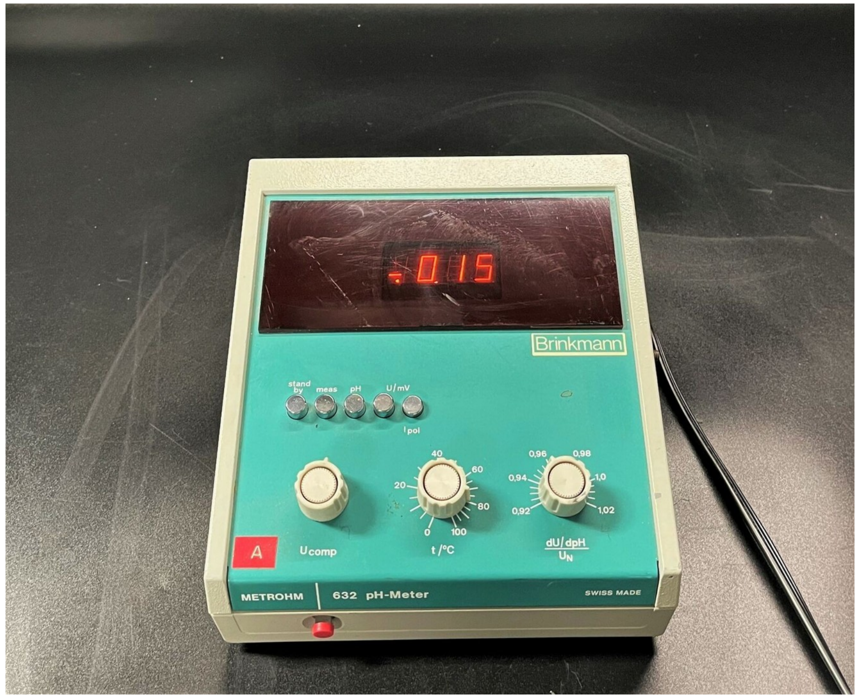
Digital measurement and control technology in the analytical sciences and its quality assurance

Ernst P. Halder
Eurachem-CH

- **Moving from analog to digital**
- **The fourth dimension**
- **Surprises**
- **How to develop software**
- **How to avoid surprises and support software development**
- **End user development (EUD)**
- **Cloud**

Moving from analog to digital

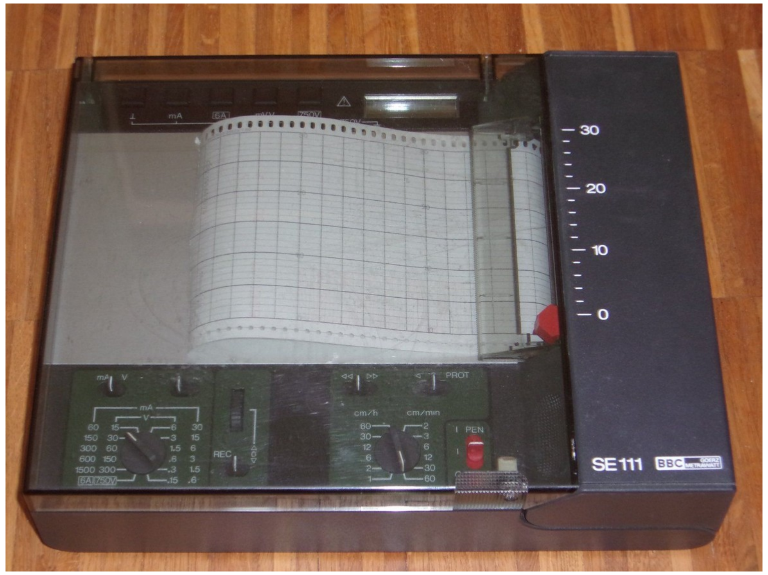


The fourth dimension

Chart recorder

- Speed
- Attenuation

•



Integrator

- Speed
- Attenuation
- Integration
- Re-Integration ???? -> «raw data»

PU 4850 video control centre – handles four channels of chromatograph control and data simultaneously.

PU 4810 computing integrator – one or two channels and many other options.

PU 4700 Autojector – the most flexible and efficient available.

PHILIPS Scientific & Analytical Equipment

Surprises

- Just when you think you have everything under control, the next surprise is waiting around the corner.

- But: Sometimes so-called surprises are predictable, you just don't want to see them coming.

Surprise I

CFR 21 Part 11, Electronic Signatures and CROMERR

- The FDA only wanted to get rid of the mountains of paper when submitting applications for approval. At the same time, the same data security should remain.
- The EPA adopted the idea for GLP (Cross-Media Electronic Reporting Rule, CROMERR).
- The FDA's ideas were immediately extended to all IT systems in the pharmaceutical industry.
- The problem was that nobody knew how to interpret the requirements.

- **Surprise II - The Y2K problem**

Surprise II - The Y2K problem

Reason for occurrence: In the 1960s and 1970s, storage space was limited and expensive. However, many programmes (and also the associated databases) were repeatedly expanded over the years, building on previous versions, without this "known shortcoming" being corrected.

Suddenly, users became aware that there was so-called "firmware" in the devices.

Can something similar happen again?

Yes, the year 2038 problem.

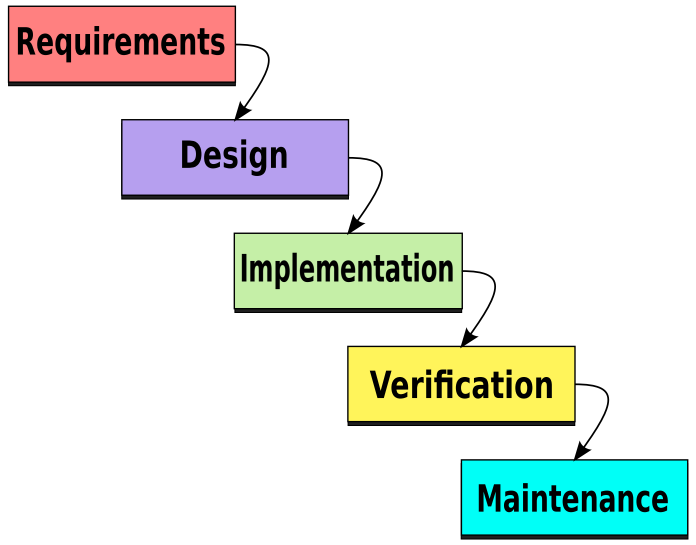
The year 2038 problem (also known as Y2038, Y2K38, or the Epochalypse) is a time formatting bug in Unix-like computer systems with representing times after 03:14:07 UTC on 19 January 2038.

Since Linux kernel 5.6 (29. March 2020) 64-bit variable time_t is also supported on 32-bit architectures.

Possible problems: Embedded systems and legacy data.

How to develop software I

Waterfall model

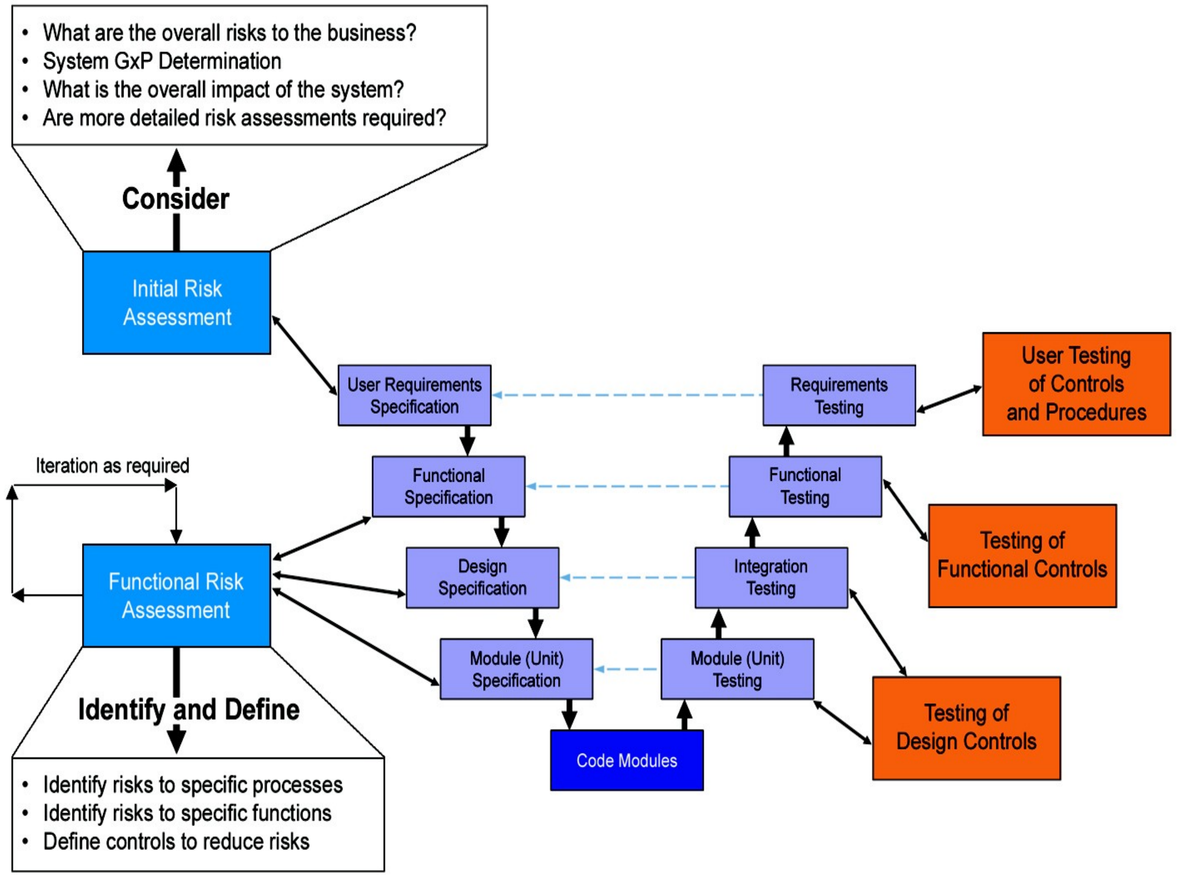


Agile software development

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

How to develop software II

GAMP 5


What are the overall risks to the business?
System GxP Determination
What is the overall impact of the system?
Are more detailed risk assessments required?
Consider
Initial Risk Assessment
User Requirements Specification
Requirements Testing
User Testing of Controls and Procedures
Iteration as required
Functional Specification
Functional Testing
Testing of Functional Controls
Functional Risk Assessment
Design Specification
Integration Testing
Module (Unit) Specification
Module (Unit) Testing
Testing of Design Controls
Identify and Define
Code Modules
Identify risks to specific processes
Identify risks to specific functions
Define controls to reduce risks

Source: Figure M3.8, GAMP 5: A Risk-Based Approach to Compliant GxP Computerized Systems, © Copyright ISPE 2008. All rights reserved. www.ISPE.org.

How to avoid surprises and support software development

(my personal experiance)

- Close cooperation between the developer/IT specialist and the future user
- First develop a prototype and test it on a “play ground”
- Based on the working prototype write the final user requirements
- This is the start of the GAMP 5 process

End user development (EUD)

End-user development (EUD) refers to activities and tools that allow end-users – people who are **not professional software developers** – to program computers and computerized equipment. People who are not professional developers can use EUD tools to create or modify software artefacts (descriptions of automated behaviour) and complex data objects without significant knowledge of a programming language.

Examples include natural language programming, spreadsheets, scripting languages (particularly in an office suite or laboratory equipment), visual programming, trigger-action programming and programming by example.

The most popular EUD tool is the spreadsheet (e.g. Excel, Libreoffice Calc).

Compared with expert programmers, end-user programmers rarely have the time or interest in systematic and disciplined software engineering activities, which makes ensuring the quality of the software artefact produced by end-user development particularly challenging.

Without adequate documentation, the maintenance and further development of programmes, especially by people other than the original developer, becomes extremely difficult.

Cloud computing

If you put **your data in the cloud**, you give it **out of your hands**. Many companies quickly forget this when the first positive effects of cloud computing appear. Therefore, it is important to ask yourself a few key questions about your own data before entrusting it to someone else.

First of all, one must be very clear that one is giving one's data out of one's hands. You must never lose sight of this fact: encryption protects data that is not currently being used and secures it in transit. In the memory for **processing**, on the other hand, data is always unencrypted. Even the best encryption is of no use, because whoever has access to the memory also has access to the data.

Only if you **encrypt the data yourself** and **no one else knows the key** the data is secure.

The data must be classified in terms of confidentiality. Then the protection is carried out quite classically according to the principles of "Need to Know" and "Least Privilege", which determine which data may be "out" and which may only be processed internally. Example: First prepared patent applications do not belong on a cloud drive!

CLOUD Act 18 US Code § 2713; The law obliges US internet companies and IT service providers to guarantee US authorities access to stored data even if the storage does not take place in the USA.

Reminder, only **you are responsible for adhering to the ALCOA+ (data integrity) principles**.