Tallinn, 26th October 2020

Language concepts
in old and recent programming languages

Joachim Raue

Agenda

itestra
be excellent

Motivation

- Why "old" programming languages are still relevant

Selected programming languages

- RPG, COBOL, PL/I, C

Selected language concepts

- Parameterization und Reuse
- Program generators, proprietary languages, and click tools

Conclusion

itestra GmbH

- **itestra GmbH** = **i**nformation **te**chnology and **stra**tegy

- Founded in 2004

- Currently approx.
  100 employees and 25 students

- 10 locations across Europe

- Headquarters in Munich

- In Tallinn since 2008

- Acclaimed specialists for large, business-critical, individual software systems

München

Service Portfolio

We ensure and reestablish IT efficiency.

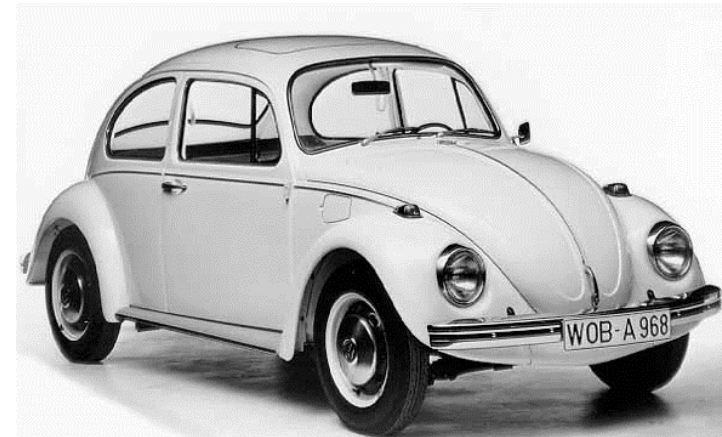| Governance & Renovation	- Measurement of software quality and costs - Systems renovation and optimization - Reduced costs and increased value
Solution Engineering	Planning, design and implementation of cost effective, high performance, mission critical IT solutions

Zoo of technologies

itestra
be excellent

	BUSINESS	MAINFRAME	MID & SMALL	S.PURPOSE	Web & Mobile
Languages	Java, C#, C++, PL/SQL , ABAP	COBOL, PL/I, ASM, NATURAL, RPG, COOLGEN, SQL	RPG, VB/A Python, PHP, Tcl/Tk, Gupta	C, Ada, Lisp, MATLAB, Pearl	ObjectiveC, Java, JavaScript
Frameworks	Java EE, .net, SAP ERP (MM, SD, PP(-PI), LE, FI/CO), Spring		Typo3, vTiger, openCMS/ERP	MP/PVM, SOLR, JavaFX	Struts GWT, PrimeFaces, MyFaces Angular, AngularJS, Node.JS, BackboneJS, EmberJS, ReactJS, Apache Cordova
OR-Mapping	Hibernate, TopLink				HTML5 Storage, MongoDB
Databases	Oracle, DB2, PostgreSQL, SQL Server	DB2, IMS/DB, VSAM	MySQL, Access, MariaDB	SQLite	
Platforms	Linux, Solaris, Windows SAP Web-AS 6, SAP NetWeaver7	MQ CICS, IMS/DC z/OS, Linux	Windows, Linux	LynxOS, Windows RT	Android, iOS, Windows Mobile

SOFTWARE ENGINEERING
Agile, Scrum, Use Cases, OO Design, UML, Inspections, CM, KM, etc.

We get to see interesting systems ...

- **1968**
  - Large bank, stock order handling
  - COBOL ~ 50.000 LoC
    (GO TO, no „PERFORM", yet)
- 1970
  - Life Insurance
  - **5 Mio LoC Assembler**
- 1980+
  - Diverse systems in proprietary „4 GLs"
  - Coolgen, VaGen, Gupta, PowerBuilder, ...
- 2000 ~ PHP
- 2012
  - General insurance
  - 1 Mio LoC Java → very messy and extremely high costs ;-)

Motivation

- A significant amount of all **business critical software systems** is implemented in programming languages, which

  - are not being taught anymore / are partly unknown,

  - do not offer concepts of modern languages,

  - have limited tool-support,

  - yet still need to be mastered in order to be able to maintain, run and develop the extremely important systems built with them

COBOL

Volume:

220 billion lines of COBOL are in use today (65% of the total software). This represents a **$2 trillion dollar total investment**. – Gartner Group

5 billion lines of new COBOL are developed **every year**. – Gartner Group.

COBOL applications **process over 80% of all daily business transactions** and mainframe platforms store 70% of all the data

Source: Future of COBOL, LegacyJ Corporation. 2003. http://www.legacyj.com/cobol/FutureOfCobol.pdf

Updated number: Micro Focus, 2019

Your bank account
Your insurance
...

... and it is still growing!

===> _____

Major developments 1960 – 2010s

- **Hardware**
  - Punch cards → magnetic tape → disks (i.e. from bytes to terabytes)
  - Memory very expensive and rare → Memory almost unlimited
  - Closed systems → Internet → mobile Computing & Cloud
  - Performance of the CPUs, Multi-Core
- **Infrastructural software**
  - OS: Single-User & Task → Scheduling, parallelism
  - DB: Files → hierarchical databases → relational databases → distributed DB
  - Compiler: text replacement → attributed parse trees
- **Software Engineering**
  - Separation of Concerns, etc.
- **User requirements**
  - Personal talk at the bank / insurance / ... and written communication
    → 7x24 online, complete transparency, willingness to switch, social networks

The Mainframe meanwhile

- Mainframes (vast majority by IBM) still are the vital core of most long established enterprises
  - 2013: 95% of the 200 biggest companies in Germany use one
- Conceived in the 1960ies, IBM still guarantees that all code written after **1965** runs **unmodified and without recompilation** on modern systems.
  - Remarkable hardware abstraction!
- Reknown for
  - reliability (minimal downtimes)
  - IO throughput
- Extremely powerful
  - Up to 100 CPUs
  - Hundreds of hard disks / SSDs
  - Magnetic tape robots
- IBM business model
  - Cannot buy, just lease by the CPU hour

Sun Fire E25K Server

The new flagship of the industry.

Get It   From $1.023.047,00 (US)

Upgrade now and get over 5x performance gains within the same chassis.

Conclusion

Conclusion

- Older programming languages are in practice still very important **(and exciting!)** for computer scientists

- These languages often miss essential concepts from today's point of view

- Particular relevance is attributed to the concepts of:
  - **Composition** and **(parameterized) reuse** of components
  - Definition of **visibilities**/scopes
  - **Type safety** and other approaches for **error avoidance** (e.g. Garbage Collection)

- **„Modern" products and technologies (rule engines, MDA, etc.) should be carefully examined to this effect!**

Requirements

Successfully dealing with existing, large software systems in older languages

- is a **new (!) challenge** (growing its importance daily) and

- poses **high requirements** to quality, qualifications and processes, e.g.:
  - Understanding (old) technologies
  - Accepting unfamiliar thoughts... often from a different time!
  - State of the Art knowledge
  - Economic thinking und doing
  - Power to persuade (effect change)
  - Courage (critical systems!)
  - Diligence ~ 0 error target
  - Application of new tools (e.g. visualizing code structures), ...

Qualification!

Much of what you learn is extraordinarily important...

- Abstraction (technical as well as business)

- Transactions

- Parallel processing

- Inheritance, OO

- Data normalization

- Algorithmic complexity, efficient data structures

...and at the same time by no means standard in the industry!

(Software) Quality means cost reductions.

MORE INTEREST?

Computer scientists wanted!

- **Full-time employment**

- **Working student**

- **Internship**

- **Final thesis** (Bachelor, Master)

Apply now! **jobs@itestra.com**

QUESTIONS?

raue@itestra.com
www.itestra.com