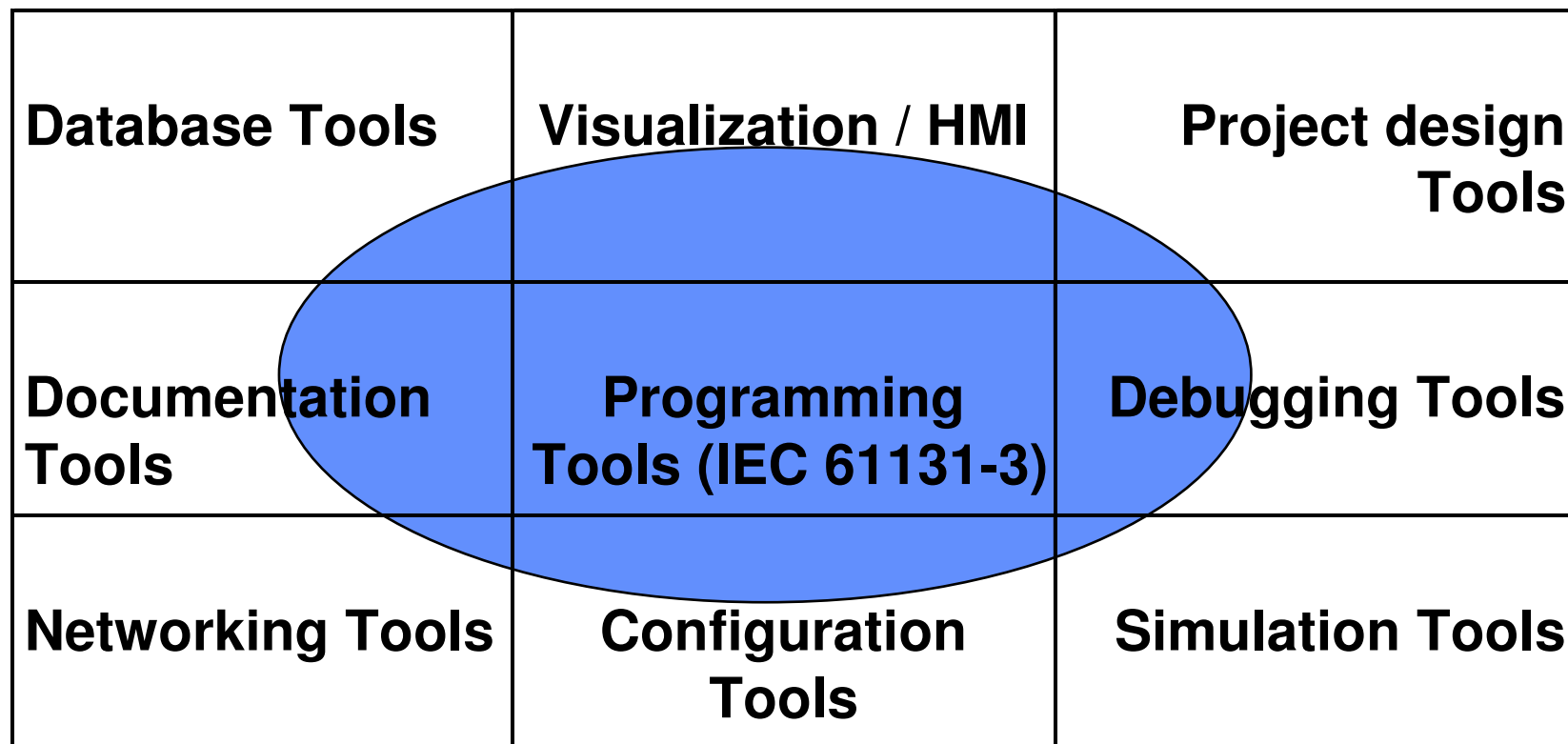
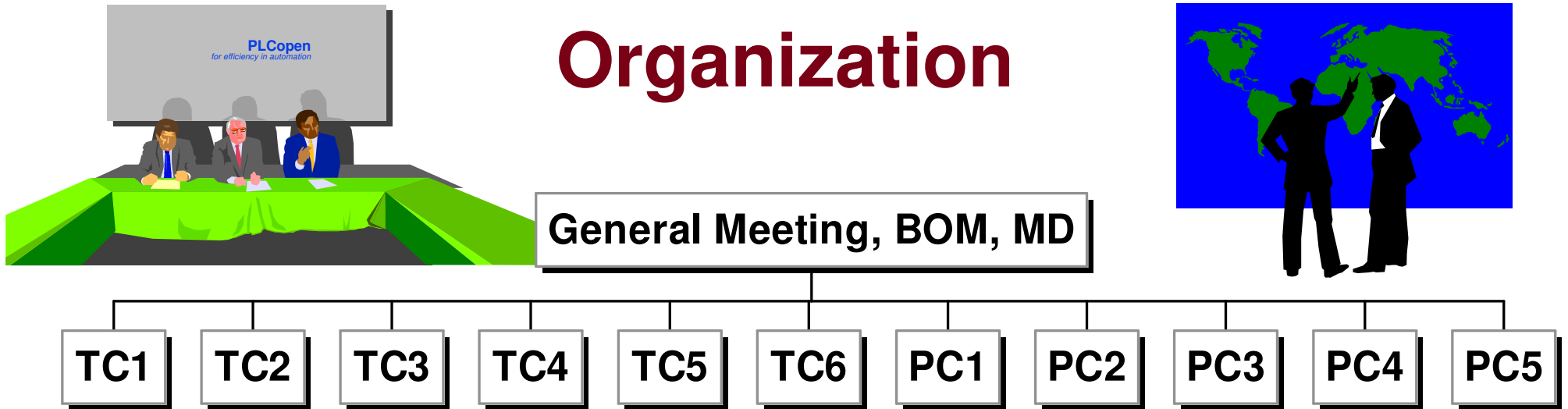


Welcome
at the
PLCopen presentation (short)

Programming in its environment



Organization



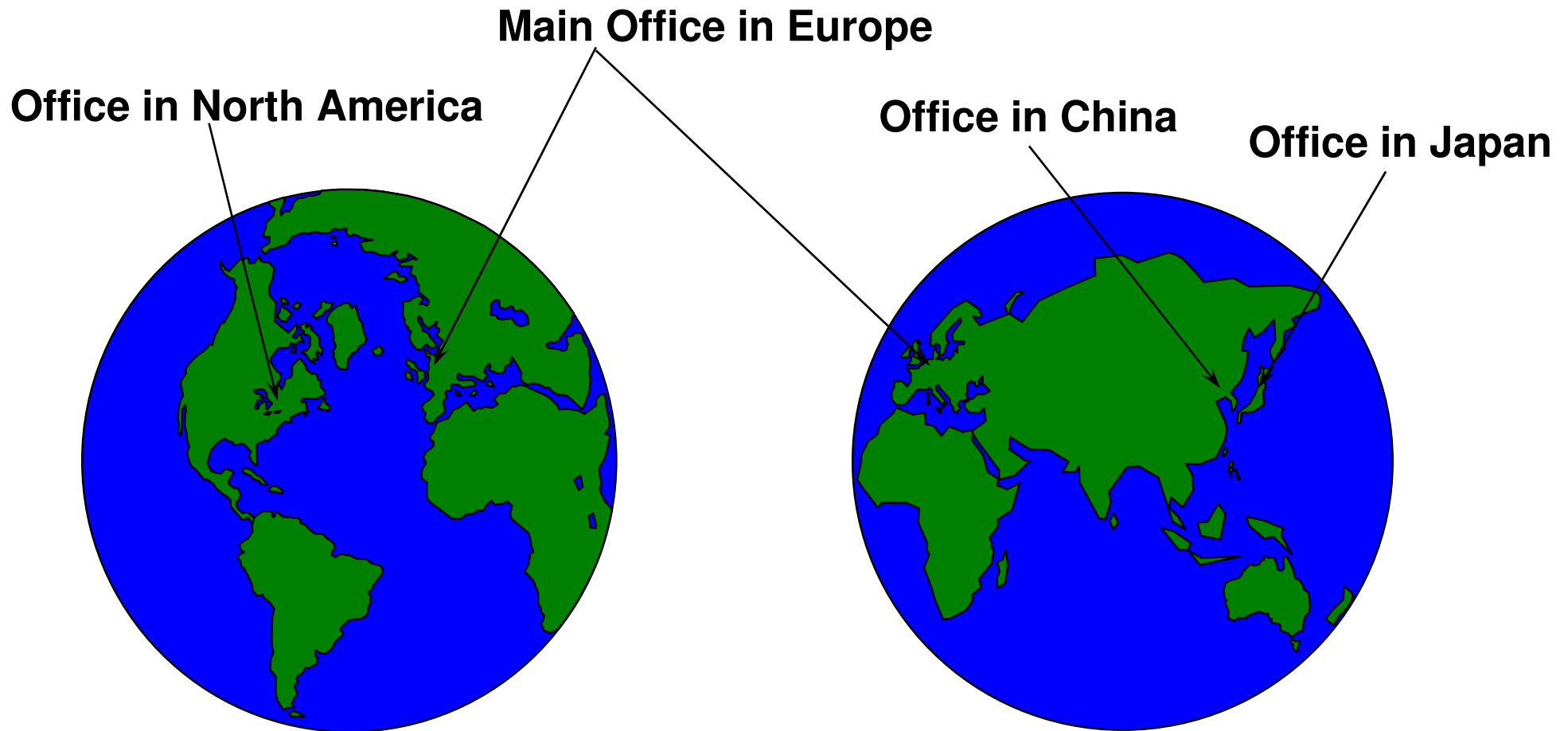
TECHNICAL

PROMOTIONAL

COMMITTEES



PLCopen as a World-wide association



General Promotion

- **PC1: General Promotion**
- **PC3: Promotion North America**
- **PC4: Promotion Japan**
- **PC5: Promotion China**

PC2: Common training program

- **The effect of training is often underestimated**
- **Standardization can be very useful and provide a better interface between study and reality**
- **PC2 defined common basics for training..**
- **.. for instance: a IEC 61131-3 training guideline is published**
- **Training facilities fulfilling basic requirements can be certified and listed / referenced to (see website for listing)**

TC1: Standards

- **IEC 61131-3 is enhanced with Corrigendum & Amendments**
- **Development of joint PLCopen position for IEC**
- **Communication of information from IEC to PLCopen**
- **Improvement proposals**
- **Focused to update, published 2003**

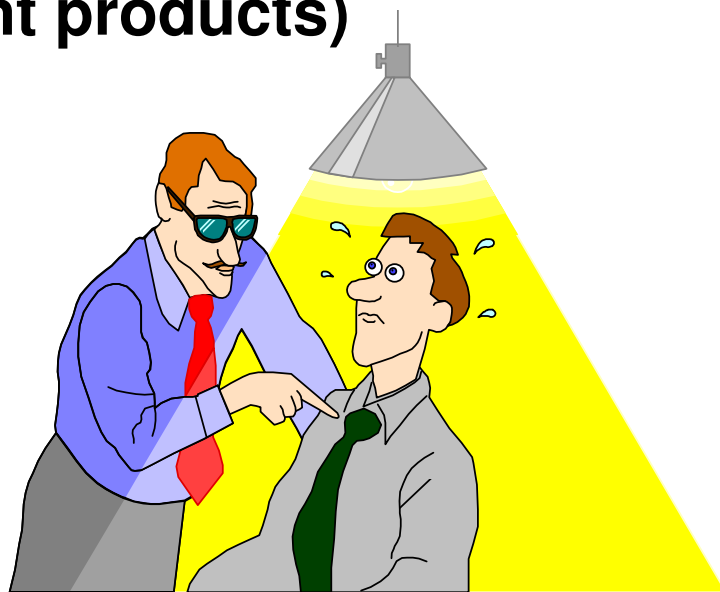
TC2: Functions

- **Definition of Function Block libraries & calling conventions**
- **... for example...**
- **Motion Control Library: the integration of different technologies: logic and motion**

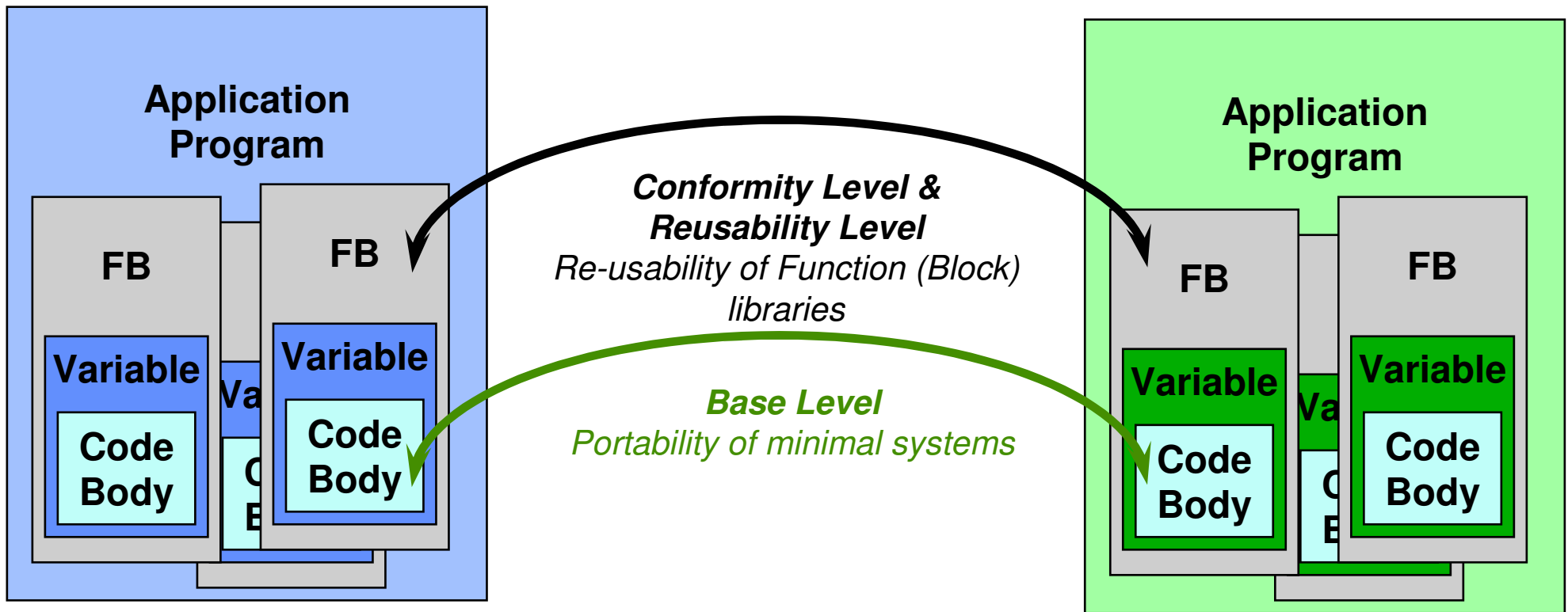
TC3 : Certification

...without testing there is no standard...

- **The IEC 61131 standard only gives basic rules for compliance**
- **Certification gives guidance for users towards real IEC 61131-3 programming systems (e.g. PLCopen certified list shows compliant products)**



TC3: PLCopen Compliance Levels



TC4: Communication

- **Communications interfaces**
- **Interfaces to add-on packages**
- **Application interchange format**
- **Mapping of Profibus has been done**
- **Mapping of CANopen has been done**

TC5: Safety

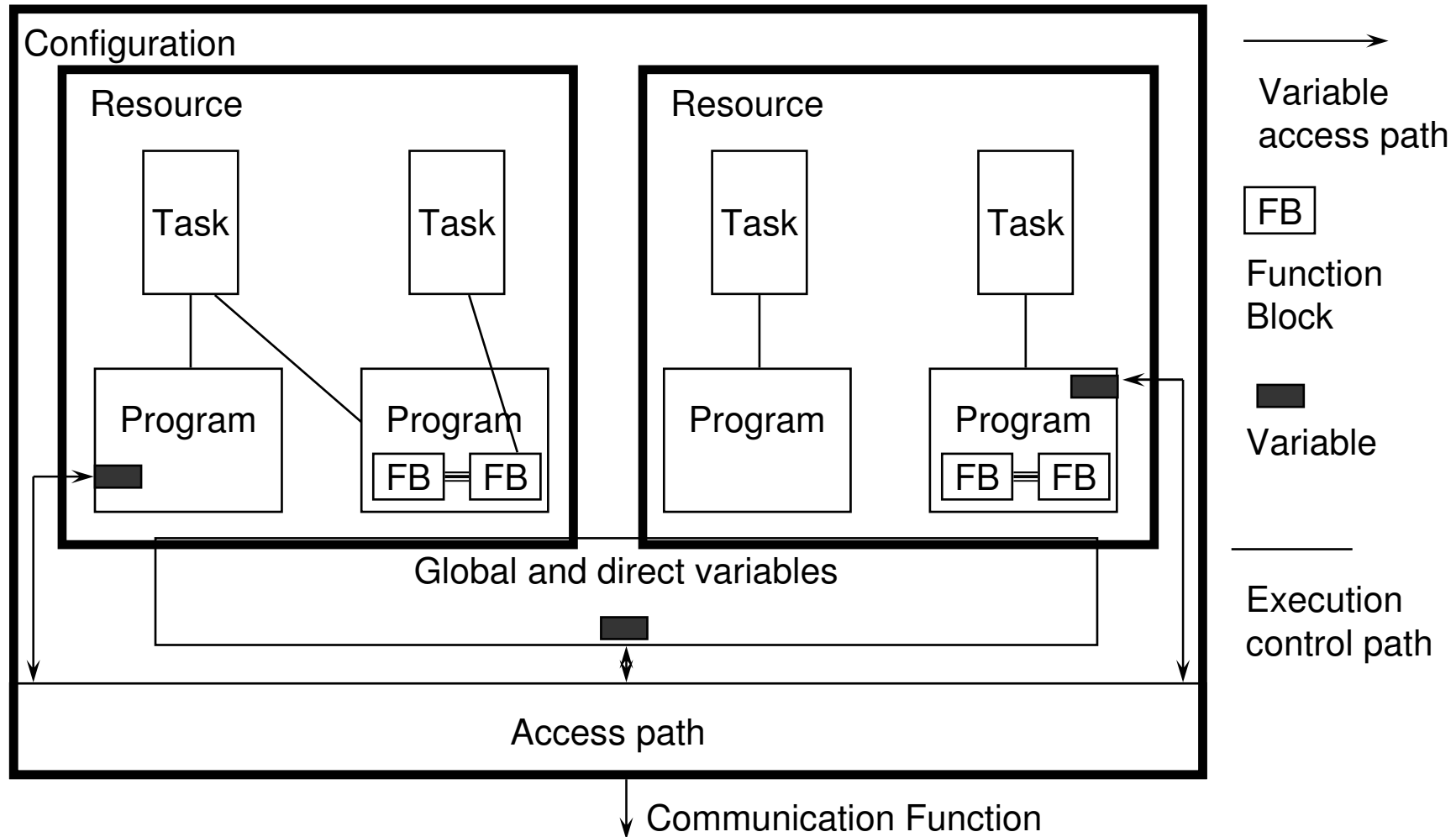
- **Support for safe programming techniques**
- **Focus to IEC 61508 “Functional Safety of Safety Related systems”**
- **Guidelines for the use of the IEC standard**
- **Basis for easier commissioning**
- **In combination with the Function Blocks**
- **First results published April 2005**



TC6: XML

- **Definition of XML schemes for all the IEC languages**
- **Representation of graphical information**
- **Interface to other tools**
- **Basis for distribution of Function Block libraries**
- **Released as V. 1.0 – official release, in June 2005**
- **Consists of a technical doc, the XML schema, and explanation.**

IEC 61131-3 Software Model



The IEC 61131-3 Programming Languages

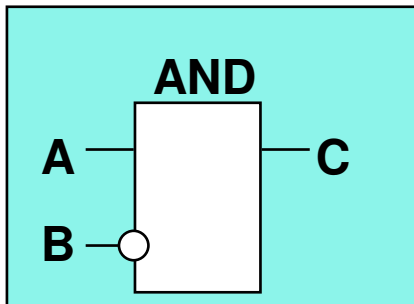
Instruction List

```
LD    A
ANDN  B
ST    C
```

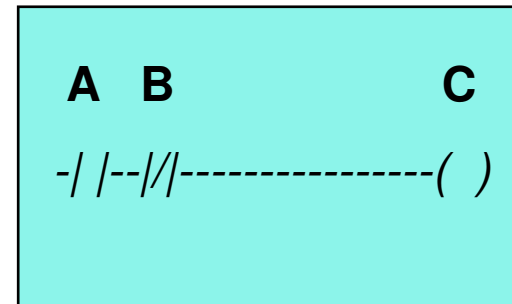
Structured Text

```
C := A AND NOT B
```

Function Block Diagram

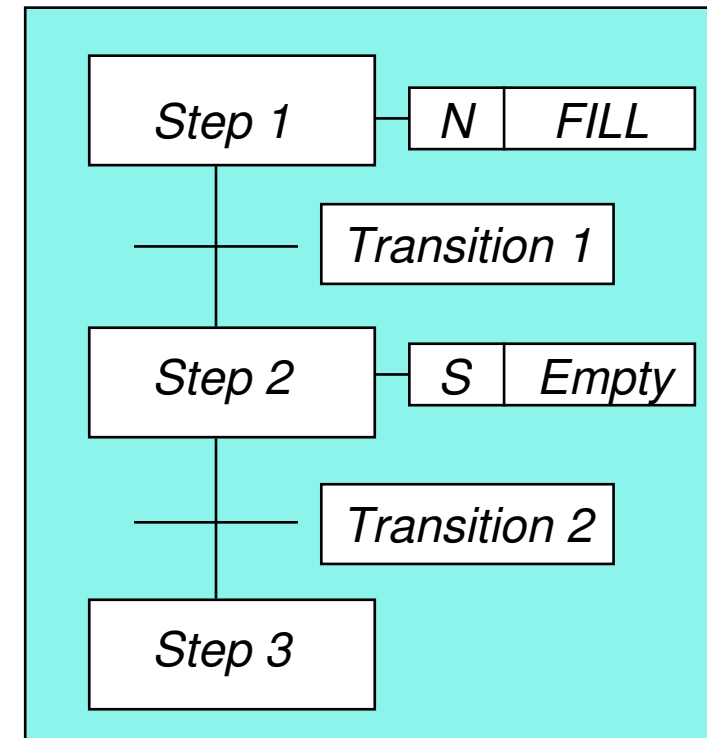


Ladder Diagram



Sequential Function Chart

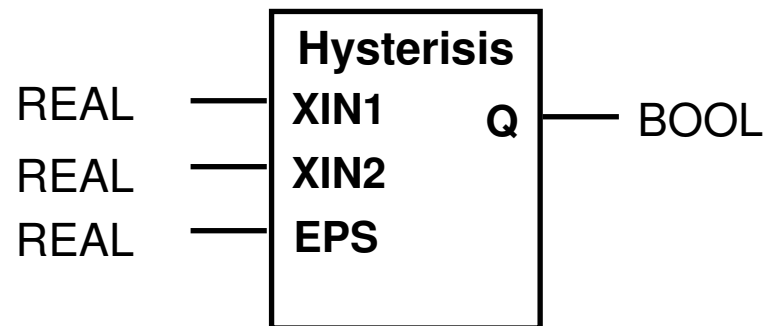
- Powerful graphical technique for describing the sequential behaviour of a control program
- Used to partition a control problem
- Shows overview, also suitable for rapid diagnostics
- The basic elements are **STEPS** with **ACTION BLOCKS** and **TRANSITIONS**
- Support for alternative and parallel sequences



Functions & Function Blocks

- highly re-usable in same program, different programs or project

▪ e. g.



- standard
- additionally supplied (PLC vendor)
- own definitions (vendor or project specific)

Function Block example (declaration)

FUNCTION_BLOCK CTU_INT

VAR_INPUT

CU: BOOL;

R: BOOL;

PV: INT;

END_VAR

VAR

PVmax: INT := 32767;

END_VAR

VAR_OUTPUT

Q: BOOL;

CV: INT;

END_VAR

IF R THEN

CV := 0;

ELSIF CU AND (CV < PVmax) THEN

CV := CV + 1;

END_IF ;

Q := (CV >= PV);

END_FUNCTION_BLOCK

Function Block example (instantiation and usage)

```
PROGRAM MyTestProgram
```

```
VAR_INPUT
```

```
Signal: BOOL;
```

```
Signal2: BOOL;
```

```
END_VAR
```

```
VAR
```

```
MyCounter: CTU_INT;
```

```
MyCounter2: CTU_INT;
```

```
END_VAR
```

```
VAR_TEMP
```

```
QTemp: BOOL;
```

```
CVTemp: INT;
```

```
END_VAR
```

```
MyCounter(CU := Signal, R := FALSE, PV := 24);
```

```
QTemp := MyCounter.Q; // FALSE
```

```
CVTemp := MyCounter.CV; // 11
```

```
MyCounter2(CU := Signal2, R := FALSE, PV := 19);
```

```
QTemp := MyCounter2.Q; // TRUE
```

```
CVTemp := MyCounter2.CV; // 74
```

```
END_PROGRAM
```

More Information...

www.plcopen.org

- **Free-of-Charge electronic Newsletter 'PLCopening' (in English)**

email: evdwal@plcopen.org