T7.3 (Dependable Avionic Systems)
Lead Partner: Selex Galileo

Partners: Alfatroll (Movation), UNIGE, SESM, S-LAB, HAI, Selex Galileo

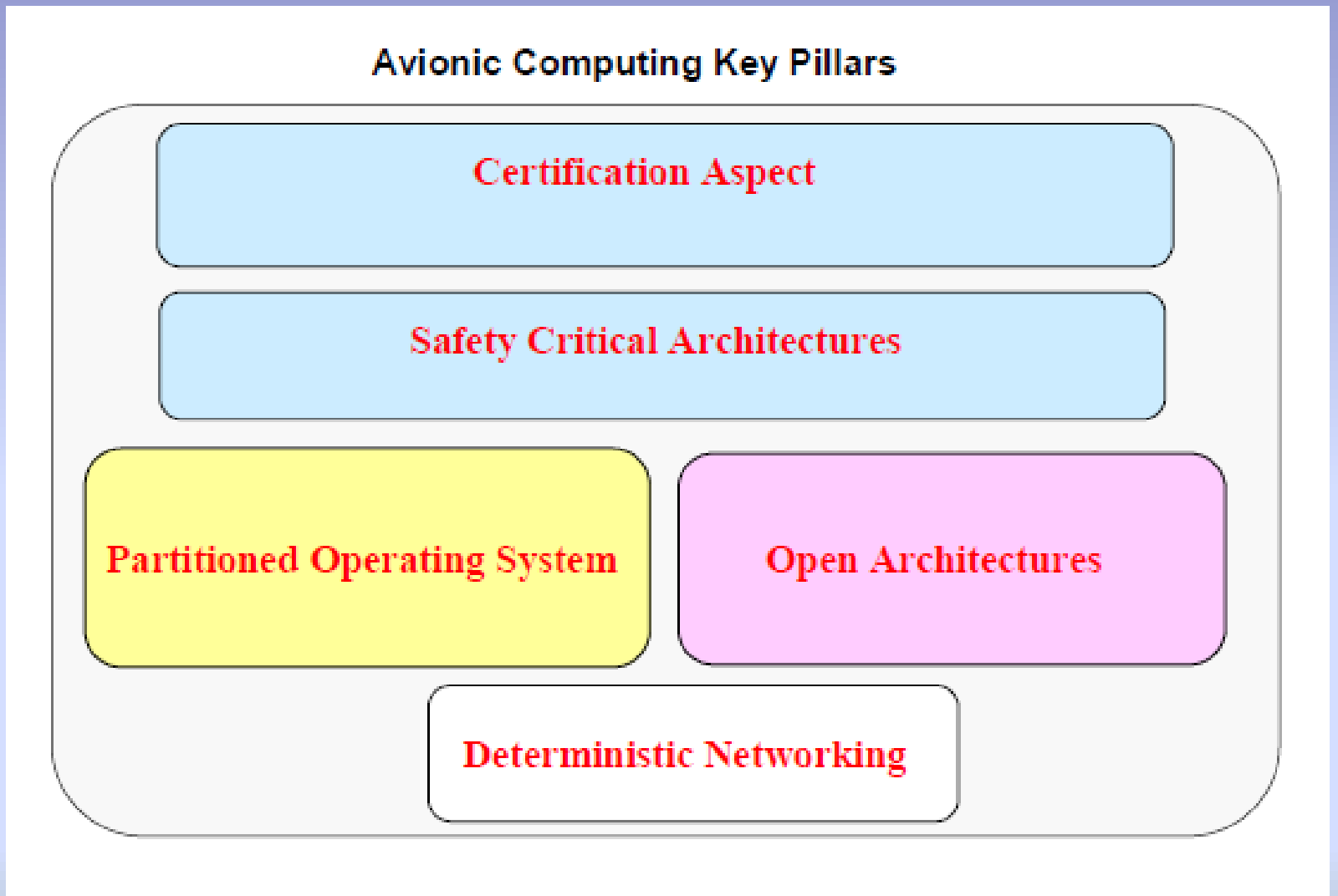
About me..


Kristen Nygård
Ole Johan Dahl
Inventors of object oriented programming: Simula
Bård Sørbye
R&D Norsk Data
Robert Cailliau
www co-inventor
Dave Walden
Internet/BBN
Tor Olav Steine
R&D Norsk Data
www.norsk-data.no

T7.3 (Dependable Avionic Systems) nSHIELD Objective

- A dependable avionics system need to include the following attribute:
  - Reliability
  - Availability
  - Safety
  - Confidentiality
  - Integrity
  - Maintainability
- Focus areas: ***Security, Privacy and Dependability***

From Annex B.2 – Dependable Avionic Computer



Todays situation:

"The demand for complex hardware & software systems has increased more rapidly than the ability to design, implement, test, and maintain them. …"

Michael Lyu

Handbook of Software Reliability Engineering, 1996

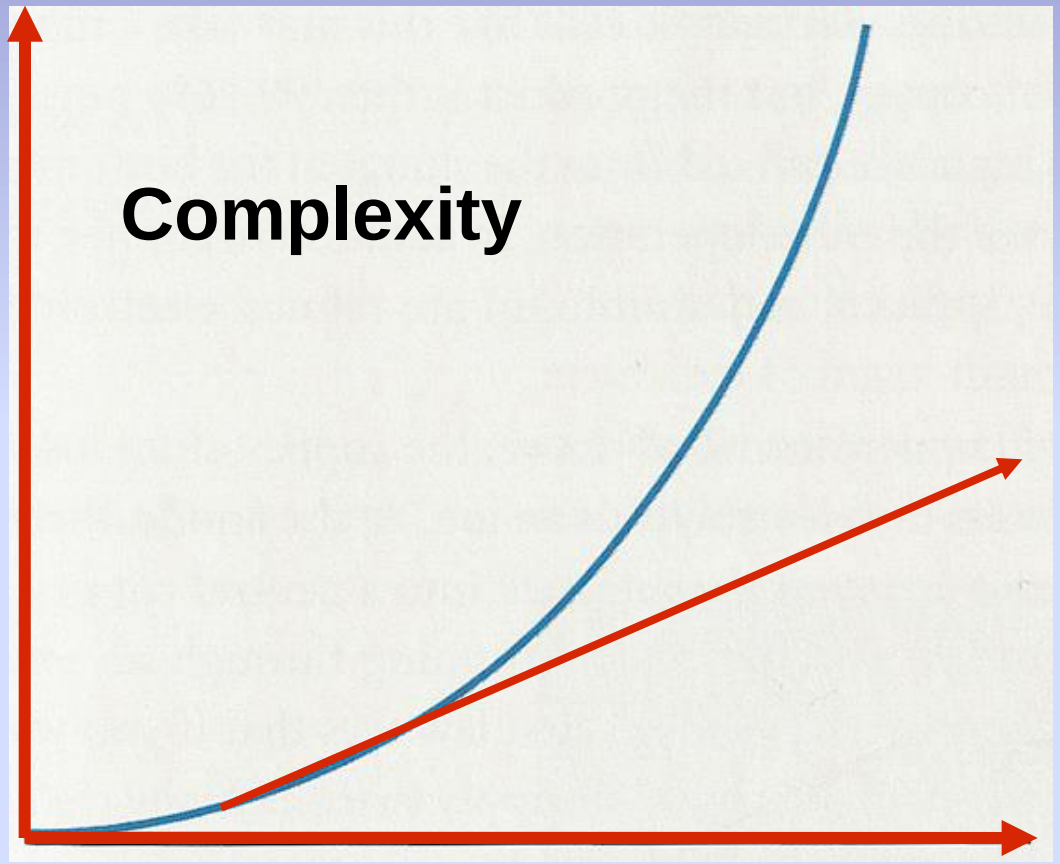
Today's situation:

Software complexity **seems to be *THE* most significant challenge**

What are desireable properties of Embedded Systems?

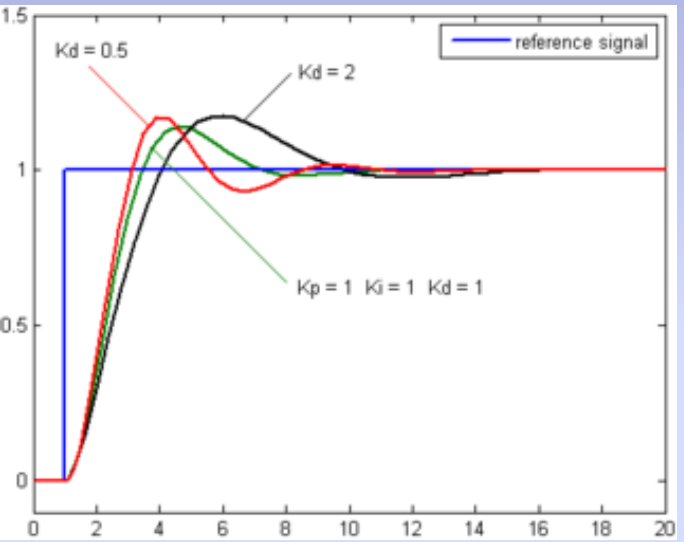
a. Fast and predictable response in all situations
b. Always deterministic results, situation based
c. Complies with strict QA standards, e.g. DO178B/C
d. Low development costs, yet adaptable to quick changes in requirements
e. Low runtime footprint and software complexity.
f. Certifiable in independent steps to save costs.
g. Full scaleability in both functionality and performance, yet maintaining all of the above.

Is exponential growth in complexity necessary?



Reduction of complexity with IQ_Engine

Can IQ_Egine e.g. replace the well known PID regulators?



- On PID regulators: http://enwikipedia.org/wiki/PID_controller
- **We claim that IQ_Engine can be equipped with all the properties of a PID regulator, yet outperforming it with respect to fast repositioning.**
- This example shows control movements of an acro pilot using large movements and quick contra to avoid the PID overshoot (e.g. 2:45 onwards): http://www.youtube.com/watch?v=J3NyptGJzLo
- IQ_Engine control signals can be adjusted for:
  - P,I and D factors
  - Control surface positioning latency
  - Sensitivity of the individual control surfaces
  - General properties of the airframe (weight, maneuverability, etc).

Today's situation:

Software certification
is a significant challenge

A certified version is required

The established standards for development of avionics are these:
Software: DO-178B (USA)/ED-12 (Europe) and
Hardware: DO-254 (USA)/ED-80 (Europe)

ED12B-C/DO-178B *Level A* is governing situations where a defect would result in catastrophic accident.

Certification is expensive, costing approx. $40-50 per line of code *)
One manufacturer estimates full UAV/RPA code of its new UAV to be 65mill lines of code… (50% more than Windows)
***) Source: http://www.windriver.com/solutions/aerospace-defense/**

DO 178B/ED 12 Software restrictions

- **Compilers are limited to certified ADA or C,**
- **Operating system must be certified RTOS***
- **Choice of hardware is limited (ARINC)**
- **Fully deterministic behaviour is required ****
- **No surplus code allowed, only net usable code (including subsystems) *****

* Real Time Operatins System
** Excludes Neural Networks
*** Excludes ordinary DBMS systems

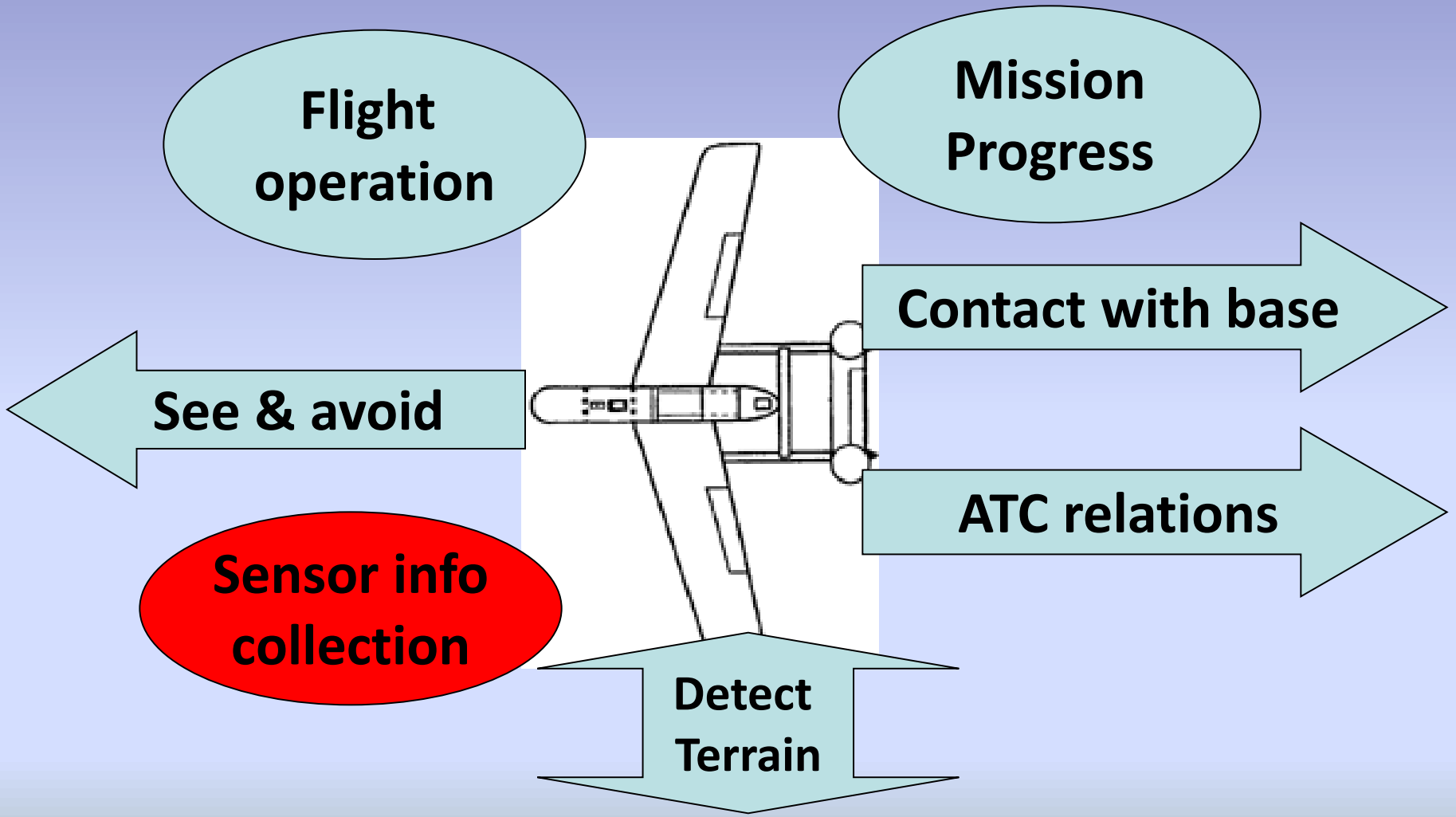
DO-178B/EB 12 software projects
involve this documentation:

1. *Plan for Software Aspects of Certification (PSAC)*
2. *Software Development Plan (SDP)*
3. *Top-level Design Document for the RTOS (Arinc653)*
4. *Detailed Design Document for (Arinc653)*
5. *Tested software system executable as standalone system*
6. *Test suite for acceptance test and Q&S verification*
7. *Manuals & Certification Evidence*
8. *Regular Reporting of progress*

Today's situation:

The functions needed
are a significant challenge

IQMotor – a task structure example



Today's situation:

The proposed solution: USE a knowledge based system: IQ_Engine

IQ_Engine System Structure

Knowledge def.
Knowledge structure
Variable def.
Mission def.

IQ_Edit
Database
import/export

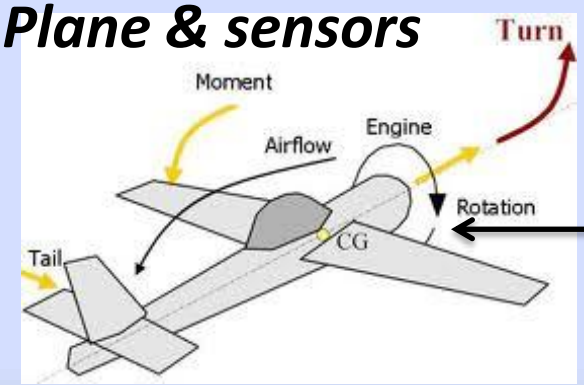
Off-line
database

Off-line system

*Embedded /
On-line system*

Knowledge
database

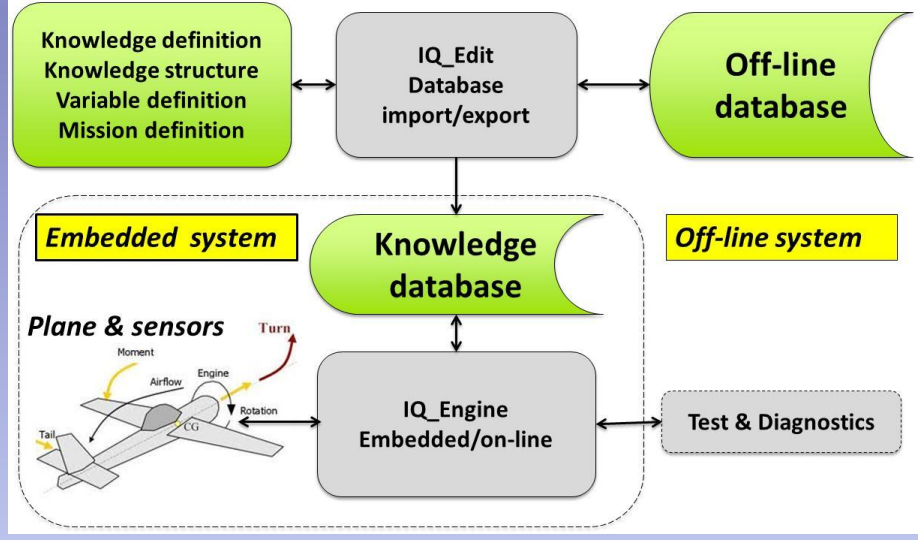
Plane & sensors



IQ_Engine
Embedded/on-line
Reasoning Engine

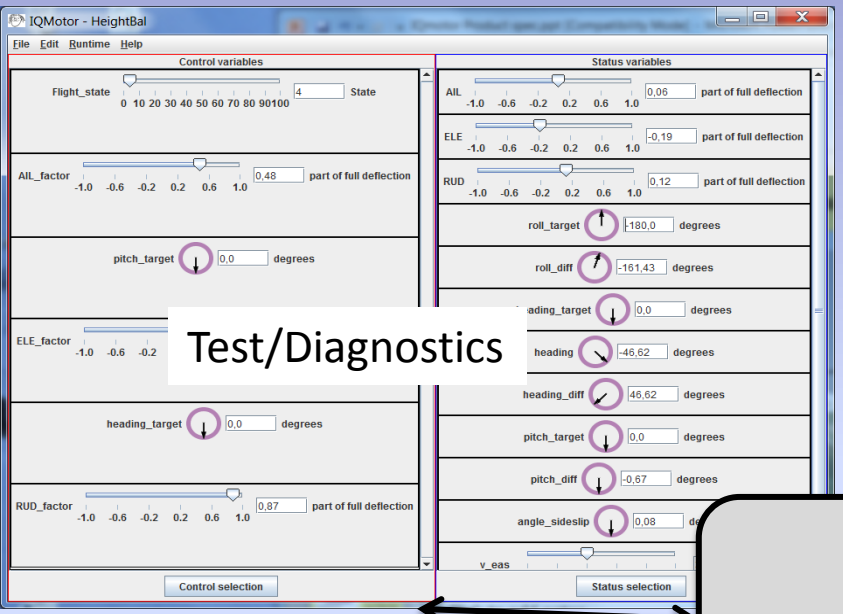
Test & Diagnostics

IQ_Engine System Structure



- Knowledge can be made manually or automated on the ground
- The Knowledge is fragmented into maneageable pieces
- Off-line DB is used to create the online Database
- Search Engine gets input from real sensors and ground control station
- Search Engine enquiries the Knowledge database
- On-board Search Engine SW has to be certified
- On-bord DB has to be certified
- First certification has same cost of traditional SW certification
- Advantage: after a change, only incremental certification is required. Not necessary re-certify the full Search Engine (and DB)

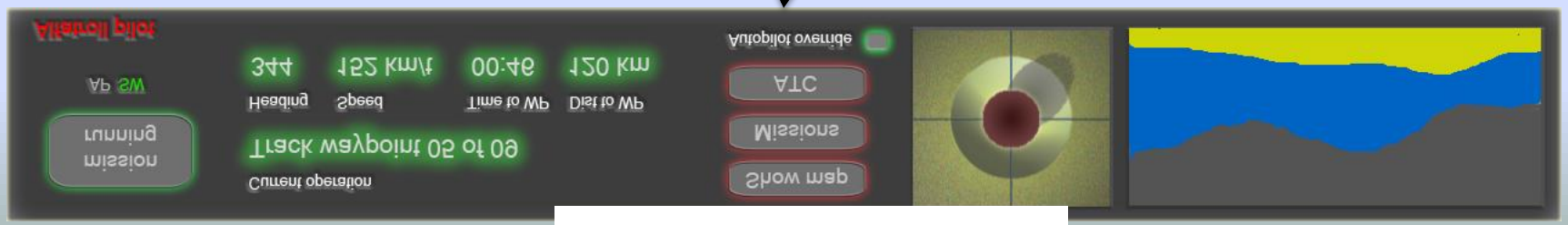
Full Demonstrator



Test/Diagnostics

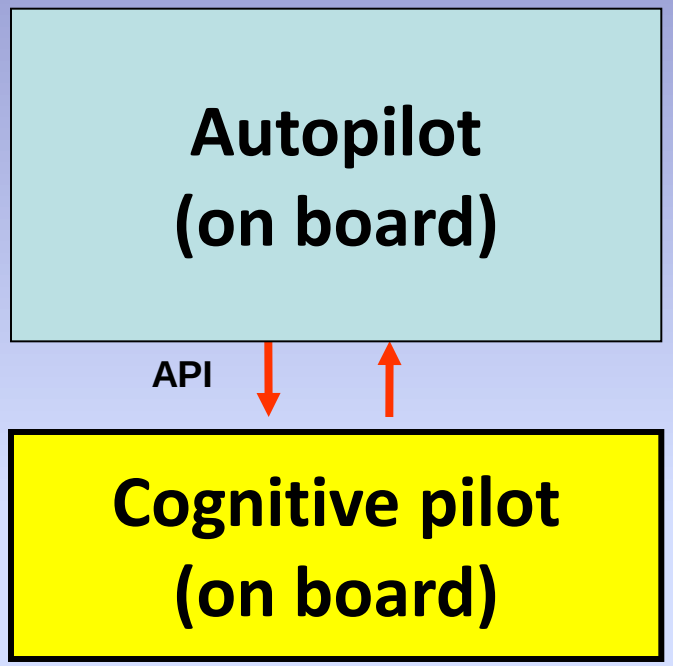
IQ_Engine Embedded/on-line

Knowledge database



Ground Control Dashboard

More than an Autopilot

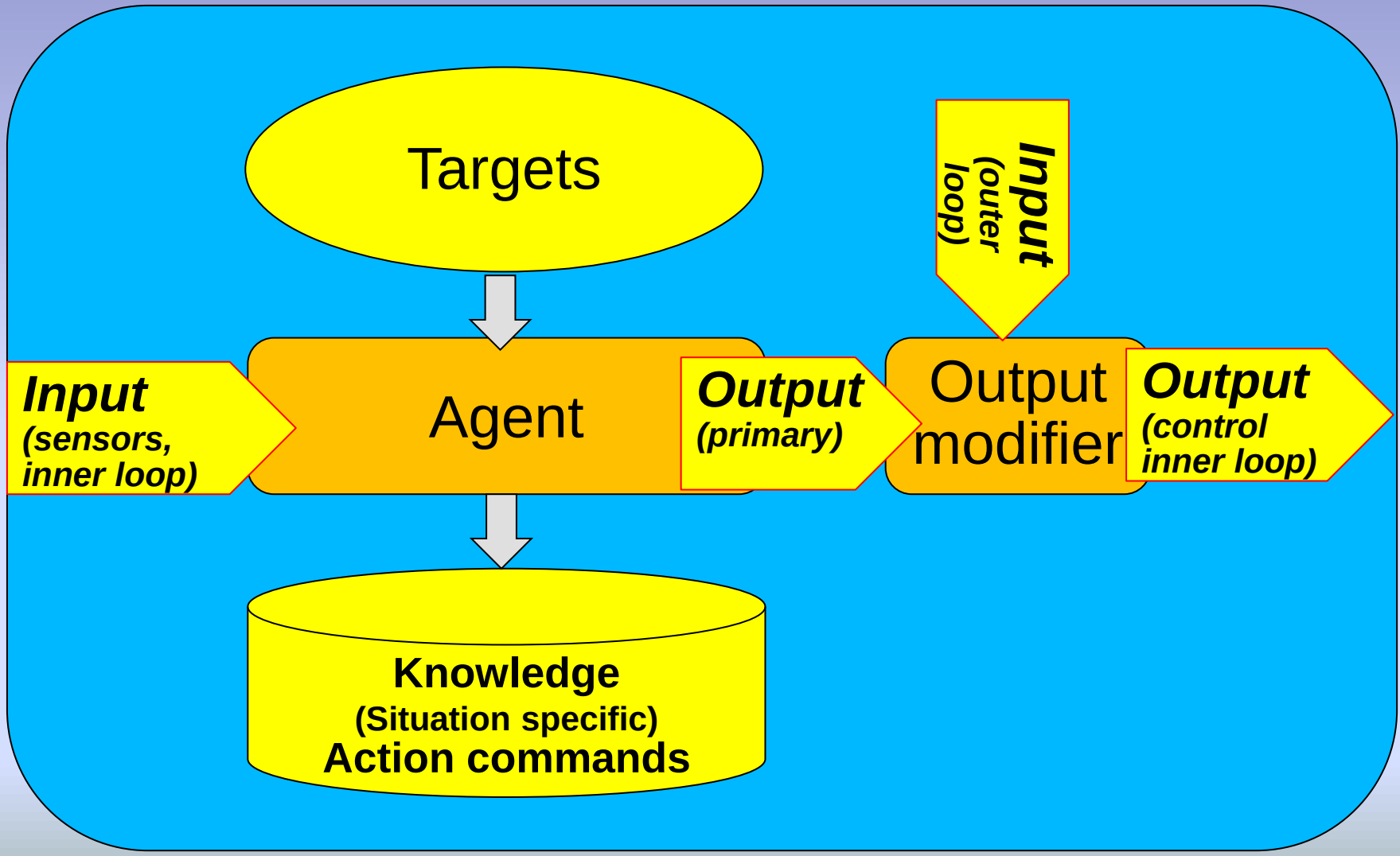


Only low level decisions
- Follows order
- Bone marrow driven reflexes
- Inner loop commands (ca50#/sec)

High level decisions
- Gives the autopilot orders
- Makes decisions based upon all information avilable.
-Outer loop commands (ca 1#/sec)

…while maintaining simplicity!

IQ_Engine Agent

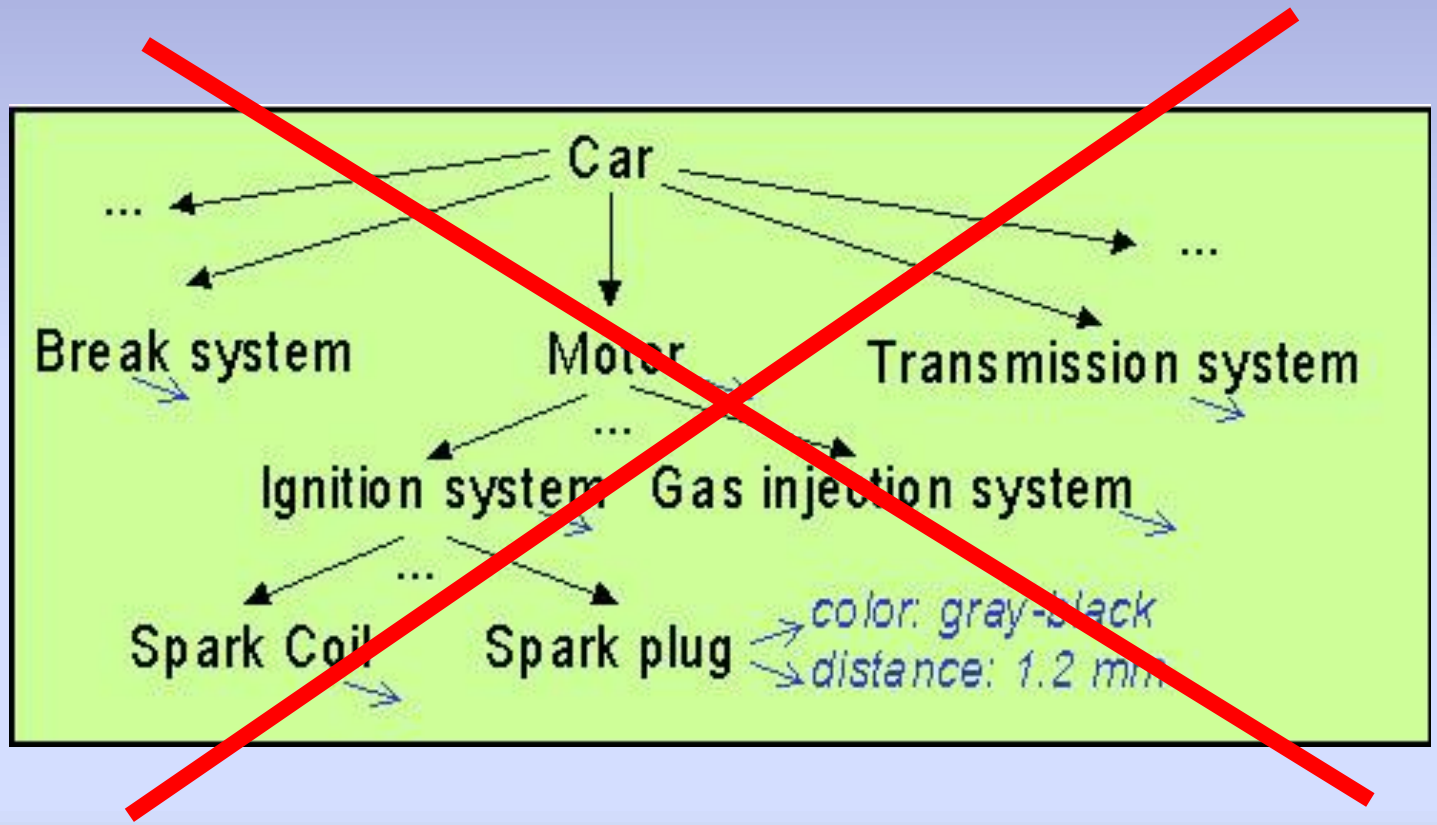


Alfatroll's IQ_Engine claims:

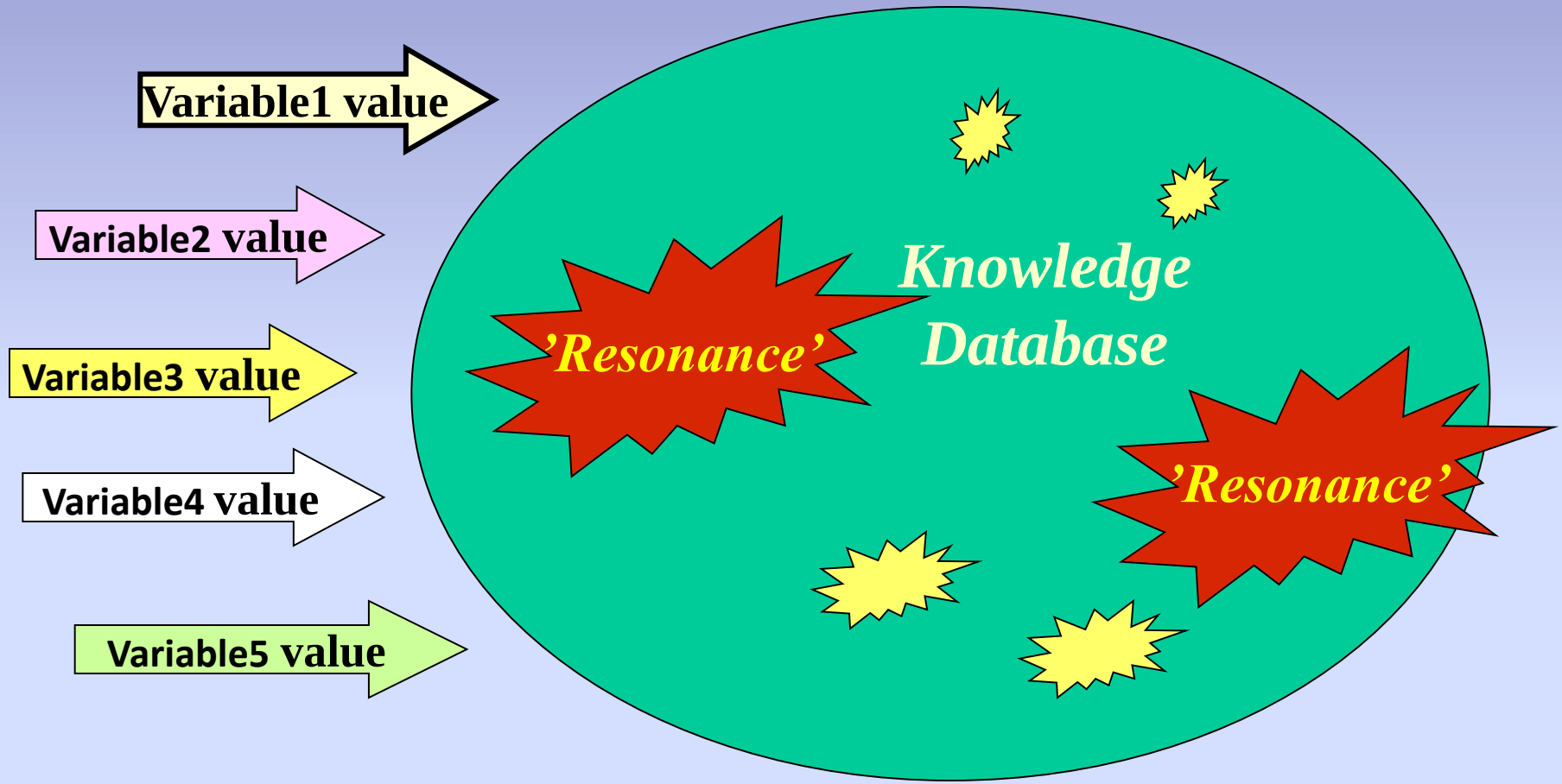
a. **Fast and predictable response in all situations**
b. **Always deterministic results, situation based**
c. **Complies with strict QA standards, e.g. DO178B/C**
d. **Low development costs, yet adaptable to quick changes in requirements**
e. **Low runtime footprint and software complexity.**
f. **Certifiable in independent steps to save costs.**
g. **Full scaleability in both functionality and performance, yet maintaining all of the above.**

The Knowledge Base:

…not like this (trad. Case Based Rasoning):

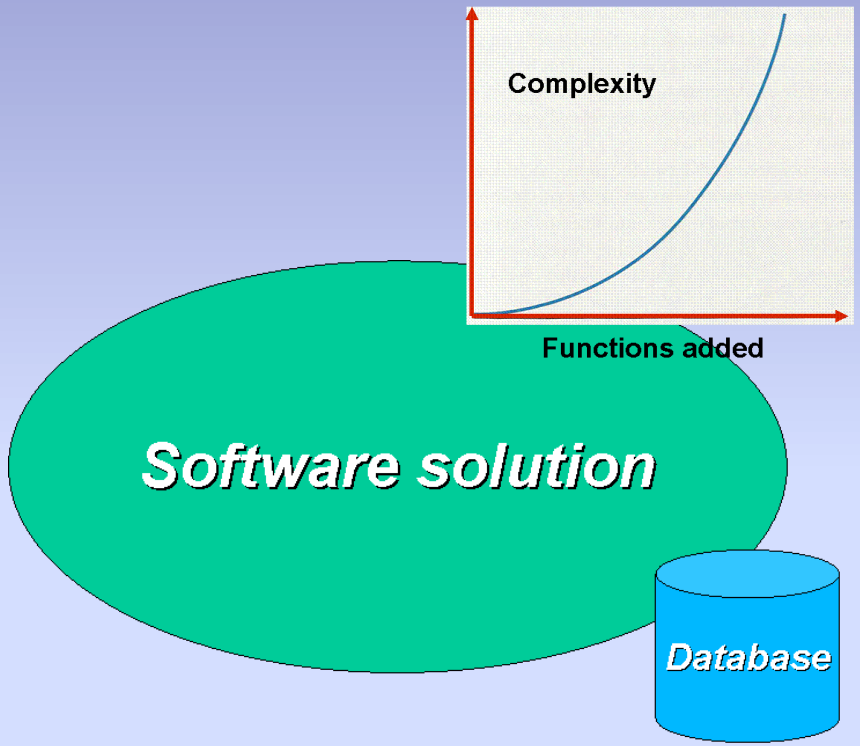


IQMotor
direct recognition by 'resonance' [1)]

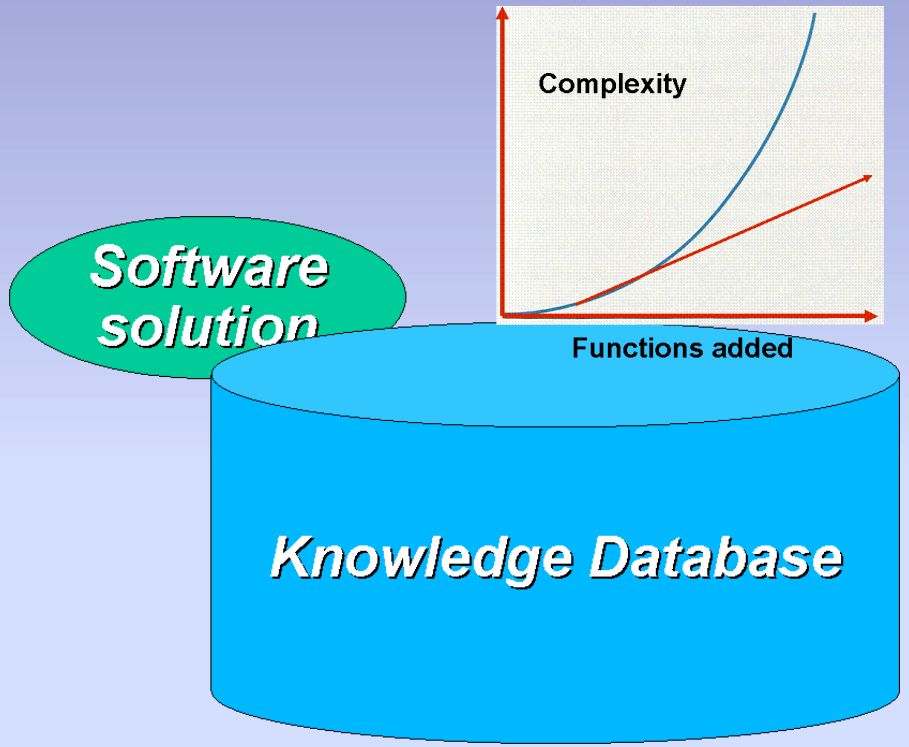


[1)] Pat. granted

IQMotor vs traditional



Traditional



IQMotor

The nSHIELD project a staged approach

Phase 1
Basic Software setup

Phase 2
Test functionality in simulator

A total budget of Euro 150’ does not cover the full development of IQ_Engine

IQ_Engine Cost reduction
when used in advanced solutions

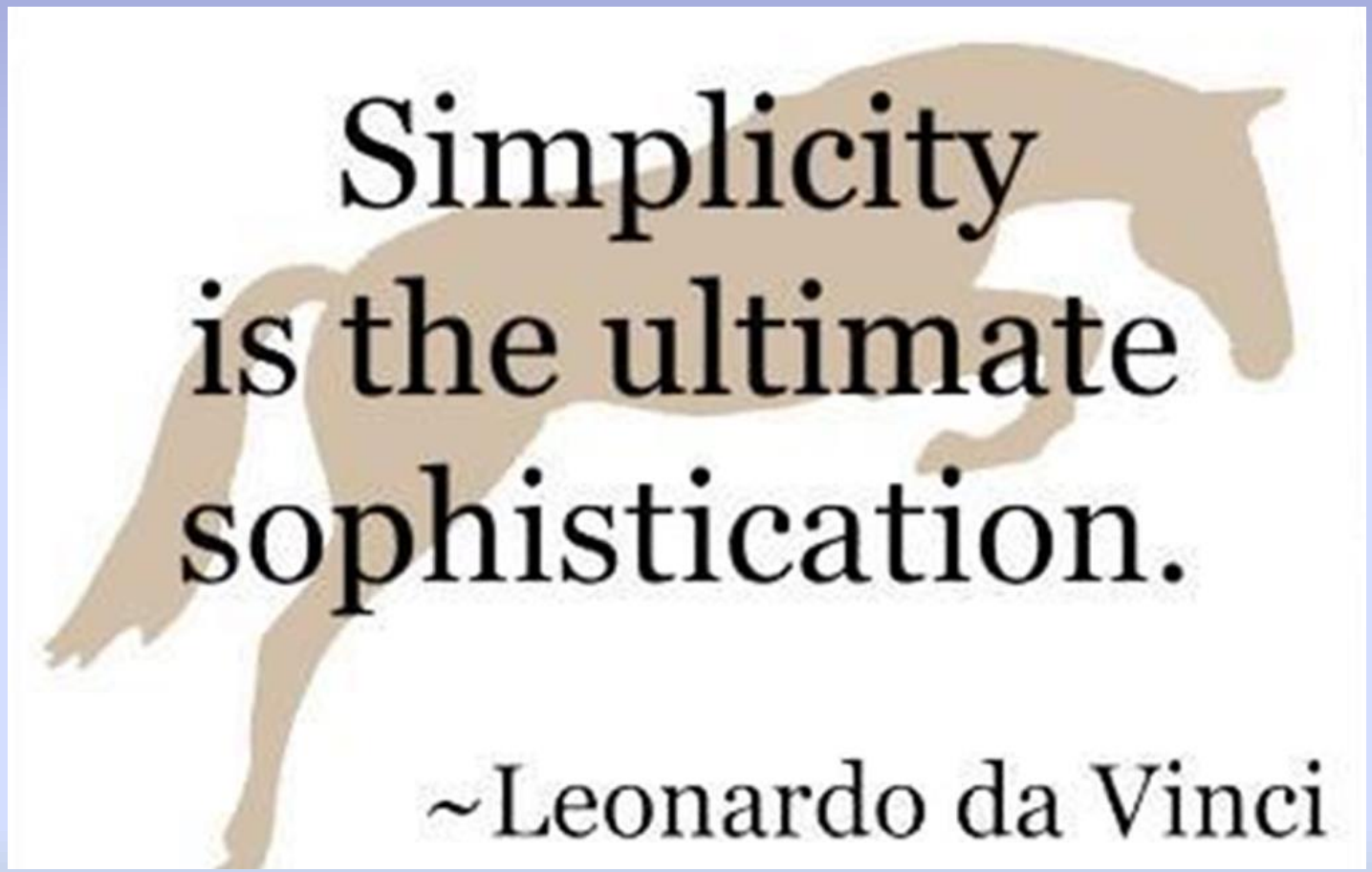
Area	Software, standard development	IQ_E Knowledge Based development
Prototype module	100	120
Convert to embedded version	250	160
Production module, verified	350	240
Production module, certified	600	450
Updated version, certified	1200	650
Estimated lifetime costs	3600	1050
In percent:	100%	29%

Estimated cost savings: 71%

For systems without certification, estimated cost reduction: 50%

The key to solving complex systems

Tor Olav Steine - tos@alfatroll.com



Source NASA