INTENS

A Flexible Integration Platform for Application Software

Ronald Tanner

31st January 2006



Presentation Agenda

- SEMAFOR
  - Company Profile
  - Products and Services
- INTENS
  - Goals and Concepts
  - Architecture and Features
  - Application
  - Configuration Example



Company Profile

- independent software developer, founded 1991 and situated in Basel
- employs 12 engineers with technical and scientific know-how
- provides services: development, consulting, support and maintenance
- References:
  - Industry: Bombardier Transportation, Novartis, SwissProbe
  - Finance & Insurances: UBS, National Versicherung
  - Government & Public Administration: Swiss Energy Office



Products and Services

- **Motor Challenge Assessments**:
  Energy-efficiency analysis of industrial drive systems
- **Software development, integration and consulting**:
  scientific and technical applications in multi-platform environments (Unix/Linux, Windows), proficient knowledge in various programming languages (C/C++, Java, Fortran, Mathematica, Matlab, Perl, Python ...)
- **Database design and implementation**:
  Oracle, PostgreSQL, MySQL
- **Project management**:
  quality assurance, configuration and release management
- **Training, Support and Maintenance**



Goals and Concepts

Acceleration of Throughput Times: data can be shared world-wide between applications and departments through access to a common database

Quality Improvement: enables traceability of business processes by means of version control

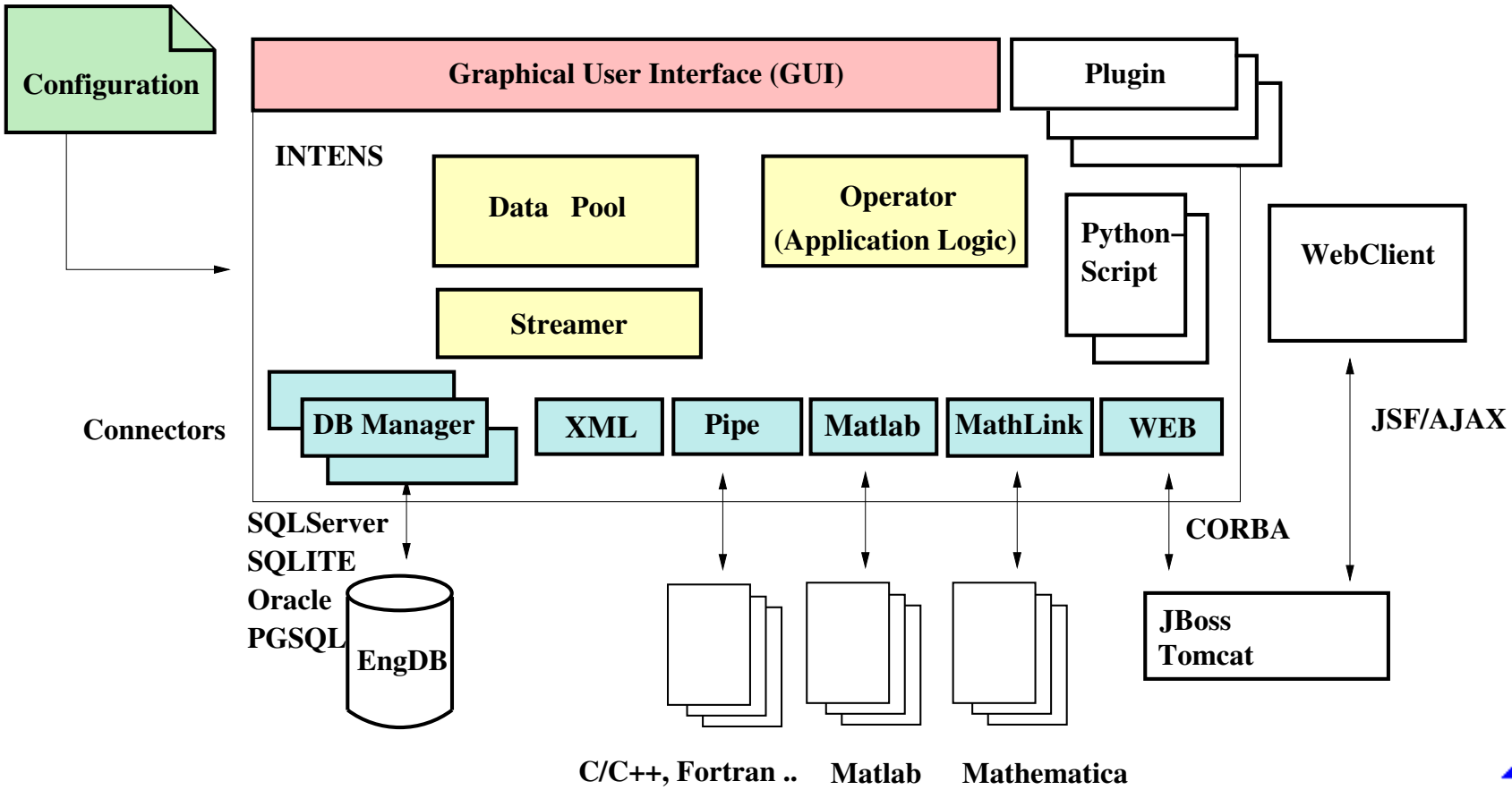
Cost Reduction: provides a set of configurable interfaces that eliminates repetitive and redundant software development

Easy to use: includes a standardized and comprehensive graphical user interface with consistent look and feel

Extensibility: simplified integration of external systems by rigorous commitment to industry standards such as XML, Matlab, Mathematica, Oracle, PostgreSQL, Python

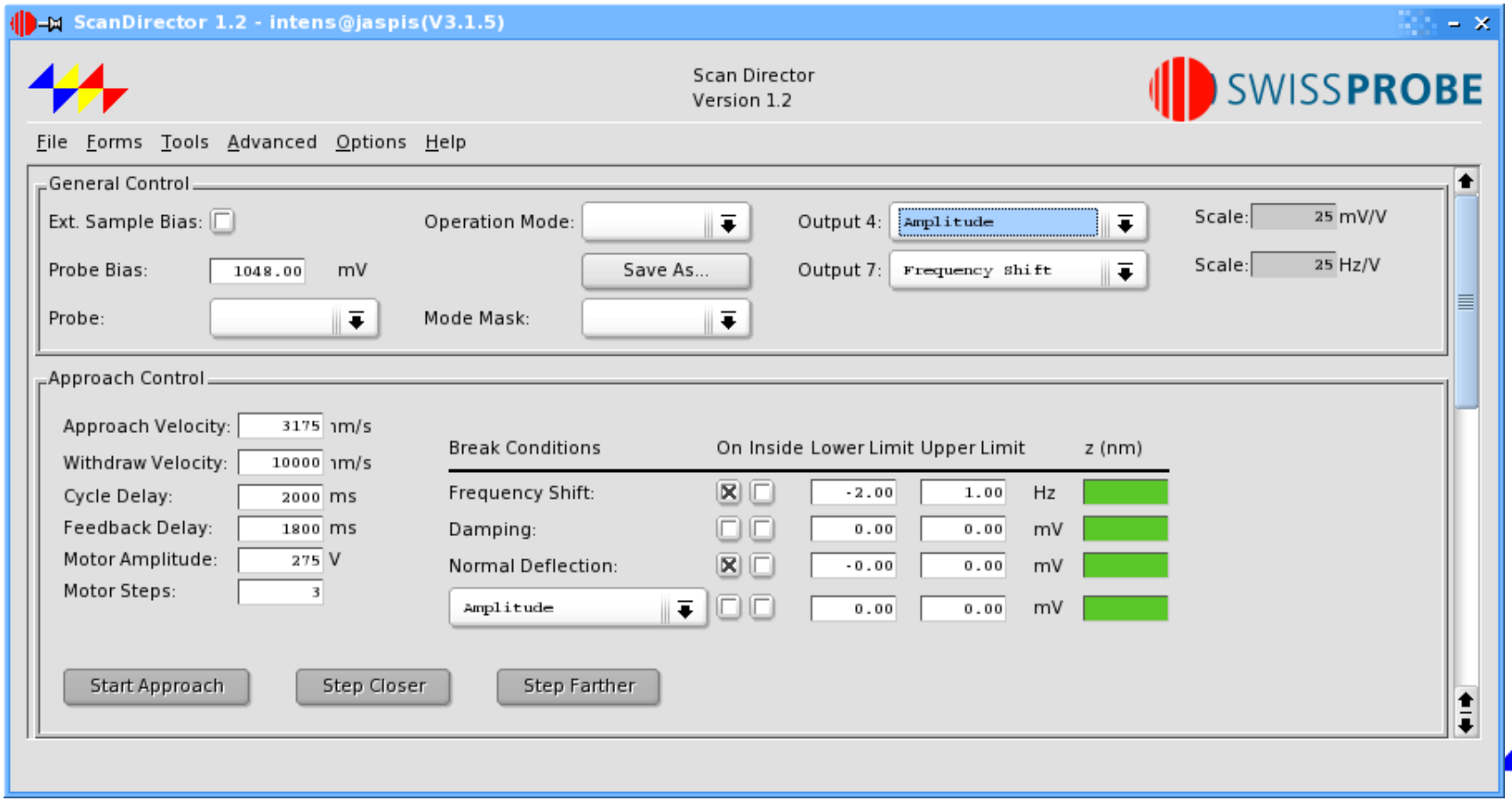


Architecture and Features

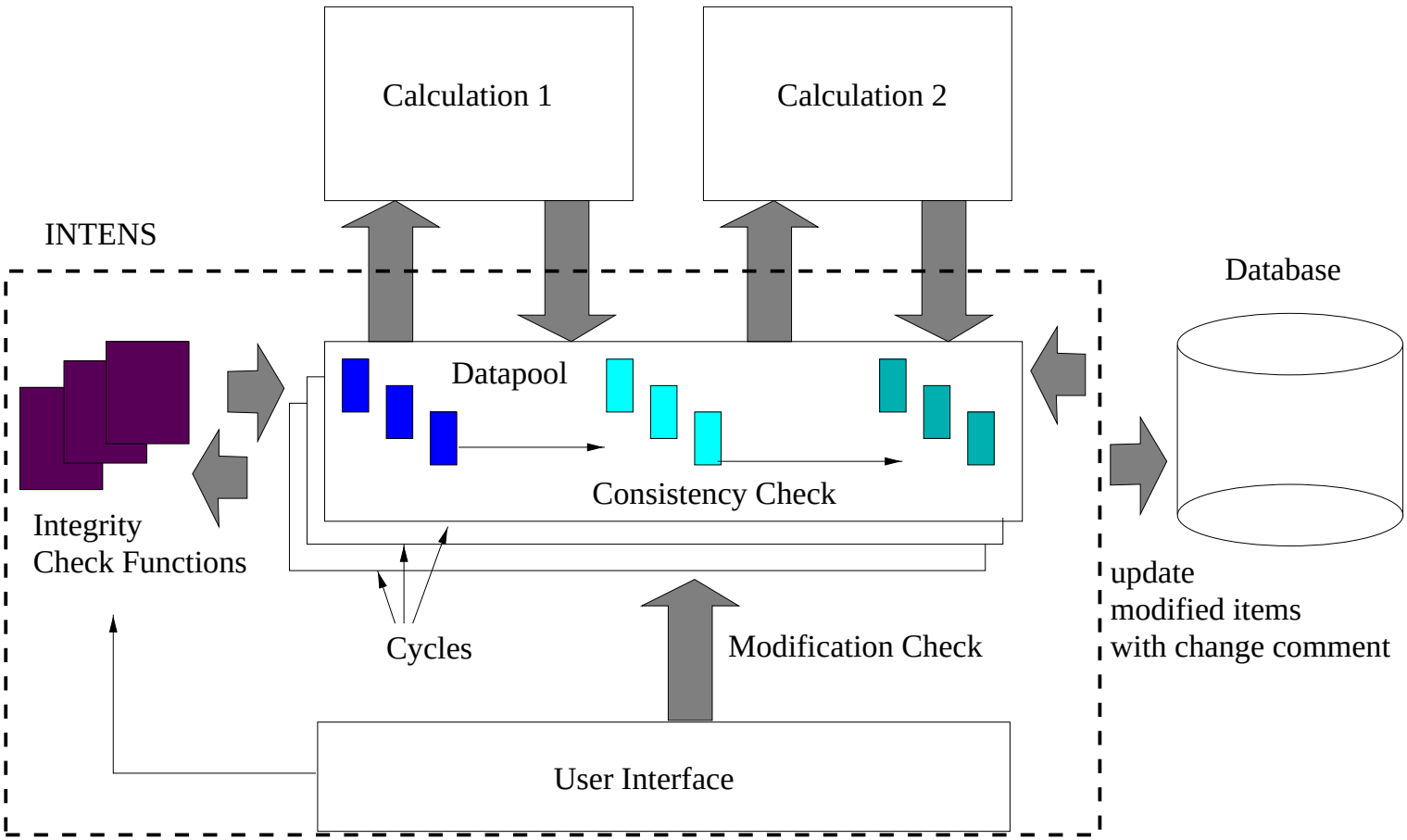




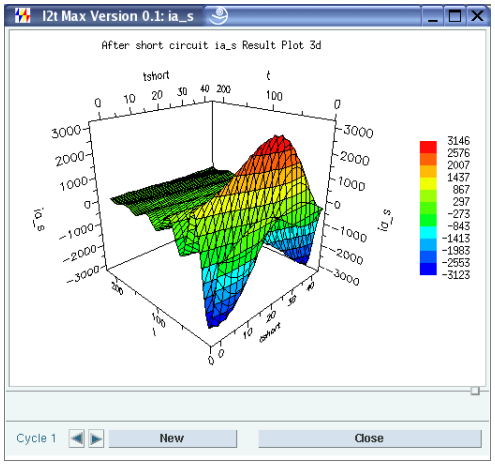
User Interface (Example)

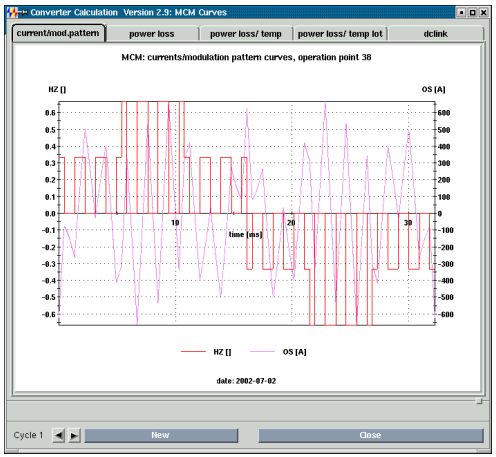


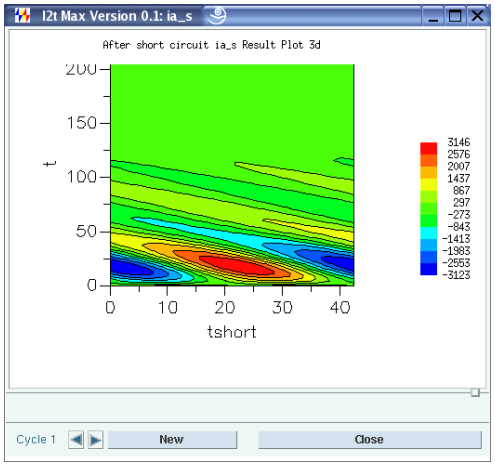
Cycles, Traceability, Integrity and Consistency

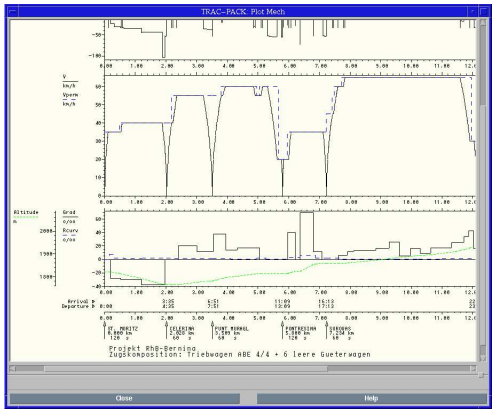


Plotting Capabilities (Examples)



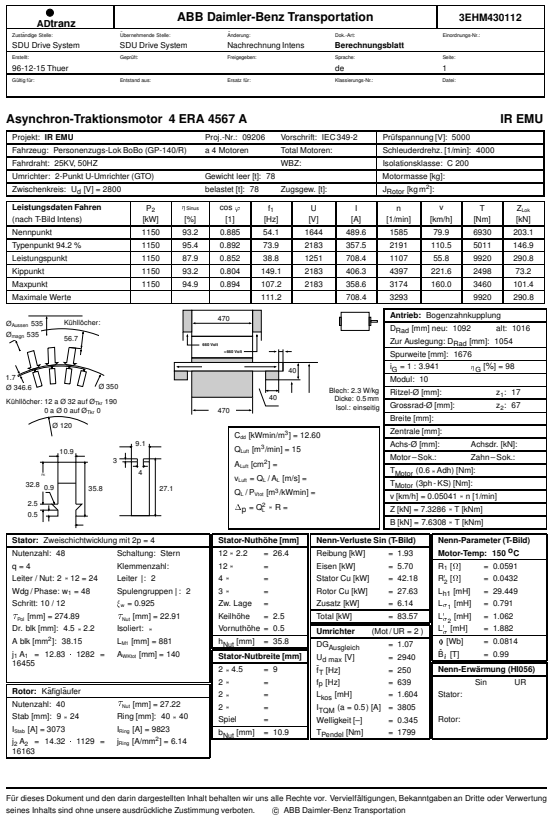






Document Generation

- Automatic Generation of high-quality reports based on LaTeX
- Flexible Configuration with Report Wizard



ADtranz | ABB Daimler-Benz Transportation | 3EHM430112

Zuständige Stelle: SDU Drive System | Übernehmende Stelle: SDU Drive System | Änderung: Nachrechnung Intens | Dok.-Art: Berechnungsblatt | Einordnungs-Nr.:
Erstellt: 96-12-15 Thuer | Geprüft: | Freigegeben: | Sprache: de | Seite: 1
Gültig für: | Ersteind durch: | Ersatz für: | Klassierungs-Nr.: | Datei:

Asynchron-Traktionsmotor 4 ERA 4567 A — **IR EMU**

Projekt: IR EMU	Proj.-Nr.: 09206	Vorschrift: IEC 349-2	Prüfspannung [V]: 5000
Fahrzeug: Personenzugs-Lok BoBo (GP-140/R)	a 4 Motoren	Total Motoren:	Schleuderdrehz. [1/min]: 4000
Fahrdraht: 25KV, 50HZ		WBZ:	Isolationsklasse: C 200
Umrichter: 2-Punkt U-Umrichter (GTO)	Gewicht leer [t]: 78		Motormasse [kg]:
Zwischenkreis: U_d [V] = 2800	belastet [t]: 78	Zugsgew. [t]:	J_{Rotor} [kg m^2]:

Leistungsdaten Fahren (nach T-Bild Intens)	P_2 [kW]	η_{Sinus} [%]	$\cos\varphi$ [1]	f_1 [Hz]	U [V]	I [A]	n [1/min]	v [km/h]	T [Nm]	Z_{Lok} [kN]
Nennpunkt	1150	93.2	0.885	54.1	1644	489.6	1585	79.9	6930	203.1
Typenpunkt 94.2 %	1150	95.4	0.892	73.9	2183	357.5	2191	110.5	5011	146.9
Leistungspunkt	1150	87.9	0.852	38.8	1251	708.4	1107	55.8	9920	290.8
Kippunkt	1150	93.2	0.804	149.1	2183	406.3	4397	221.6	2498	73.2
Maxpunkt	1150	94.9	0.894	107.2	2183	358.6	3174	160.0	3460	101.4
Maximale Werte				111.2		708.4	3293		9920	290.8

Antrieb: Bogenzahnkupplung

D_{Rad} [mm] neu: 1092	alt: 1016
Zur Auslegung: D_{Rad} [mm]: 1054	
Spurweite [mm]: 1676	
i_G = 1 : 3.941	η_G [%] = 98
Modul: 10	
Ritzel-Ø [mm]:	z_1: 17
Grossrad-Ø [mm]:	z_2: 67
Breite [mm]:	
Zentrale [mm]:	
Achs-Ø [mm]:	Achsdr. [kN]:
Motor – Sok.:	Zahn – Sok.:
T_{Motor} (0.6 · Adh) [Nm]:	
T_{Motor} (3ph · KS) [Nm]:	
v [km/h] = 0.05041 · n [1/min]	
Z [kN] = 7.3286 · T [kNm]	
B [kN] = 7.6308 · T [kNm]	

C_{det} [kWmin/m^3] = 12.60
$Q_{Lüft}$ [m^3/min] = 15
$A_{Lüft}$ [cm^2] =
$v_{Lüft} = Q_L / A_L$ [m/s] =
Q_L / P_{Vist} [m^3/kWmin] =
$\Delta_p = Q_L^2 \cdot R$ =

Stator: Zweischichtwicklung mit 2p = 4

Nutenzahl: 48	Schaltung: Stern
q = 4	Klemmenzahl:
Leiter / Nut: 2 · 12 = 24	Leiter |: 2
Wdg / Phase: w_1 = 48	Spulengruppen |: 2
Schritt: 10 / 12	ζ_w = 0.925
τ_{Nu} [mm] = 274.89	τ_{Nut} [mm] = 22.91
Dr. blk [mm]: 4.5 · 2.2	Isoliert: -
A blk [mm^2]: 38.15	L_{Wi} [mm] = 881
j_1 A_1 = 12.83 · 1282 = 16455	A_{Aktkup} [mm] = 140

Rotor: Käfigläufer

Nutenzahl: 40	τ_{Nut} [mm] = 27.22
Stab [mm]: 9 · 24	Ring [mm]: 40 · 40
I_{Stab} [A] = 3073	I_{Ring} [A] = 9823
j_2 A_2 = 14.32 · 1129 = 16163	j_{Ring} [A/mm^2] = 6.14

Stator-Nuthöhe [mm]

12 · 2.2	= 26.4
12 ·	=
4 ·	=
3 ·	=
Zw. Lage	=
Keilhöhe	= 2.5
Vornuthöhe	= 0.5
h_{Nut} [mm]	= 35.8

Stator-Nutbreite [mm]

2 · 4.5	= 9
2 ·	=
2 ·	=
2 ·	=
Spiel	=
b_{Nut} [mm]	= 10.9

Nenn-Verluste Sin (T-Bild)

Reibung [kW]	= 1.93
Eisen [kW]	= 5.70
Stator Cu [kW]	= 42.18
Rotor Cu [kW]	= 27.63
Zusatz [kW]	= 6.14
Total [kW]	= 83.57

Umrichter (Mot / UR = 2)

$DG_{Ausgleich}$	= 1.07
$U_{d\,max}$ [V]	= 2940
f_T [Hz]	= 250
f_D [Hz]	= 639
L_{sos} [mH]	= 1.604
I_{TGM} (a = 0.5) [A]	= 3805
Welligkeit [–]	= 0.345
T_{Pendel} [Nm]	= 1799

Nenn-Parameter (T-Bild)

Motor-Temp: 150 °C

R_1 [Ω]	= 0.0591
R'_2 [Ω]	= 0.0432
L_{h1} [mH]	= 29.449
$L_{\sigma 1}$ [mH]	= 0.791
$L_{\sigma 2}$ [mH]	= 1.062
L'_σ [mH]	= 1.882
ø [Wb]	= 0.0814
$\hat{B}_1$ [T]	= 0.99

Nenn-Erwärmung (Hl056)

	Sin	UR
Stator:		
Rotor:		

Für dieses Dokument und den darin dargestellten Inhalt behalten wir uns alle Rechte vor. Vervielfältigungen, Bekanntgaben an Dritte oder Verwertung seines Inhalts sind ohne unsere ausdrückliche Zustimmung verboten. © ABB Daimler-Benz Transportation



Report Wizard

Report-Wizard

Name: Reportwizard-Ex. Version: 1 Status: Preliminary Select... Store/Modify

Type	Name/Label	Parameter
Reportwizard-Ex.		
Textfield		
Textfield	Abstract: This reportwizard is a powerful tool to store your reports structered to database.	
Table Scalars: label / value		
Title	Title Table 1	
Caption	Caption Table 1	
Row	Train Mass [t]	mlt: Mass [t]
Row	Train Lenght [m]	llt: Length [m]
Row	Gear ration	kv: Gear ratio (>1)
Plot		
Title	Test Plot	
X-Axis	x [km]	t_sat: Running distance [
Y-Axis	Train Speed [km/h]	t_vt: Train speed [km/h]
X-Axis		t_sat: Running distance [
Y-Axis	Speed Limit [km/h]	Track: Speed Limit Input



Integration Prerequisites

External programs can be integrated without source code modification provided that:

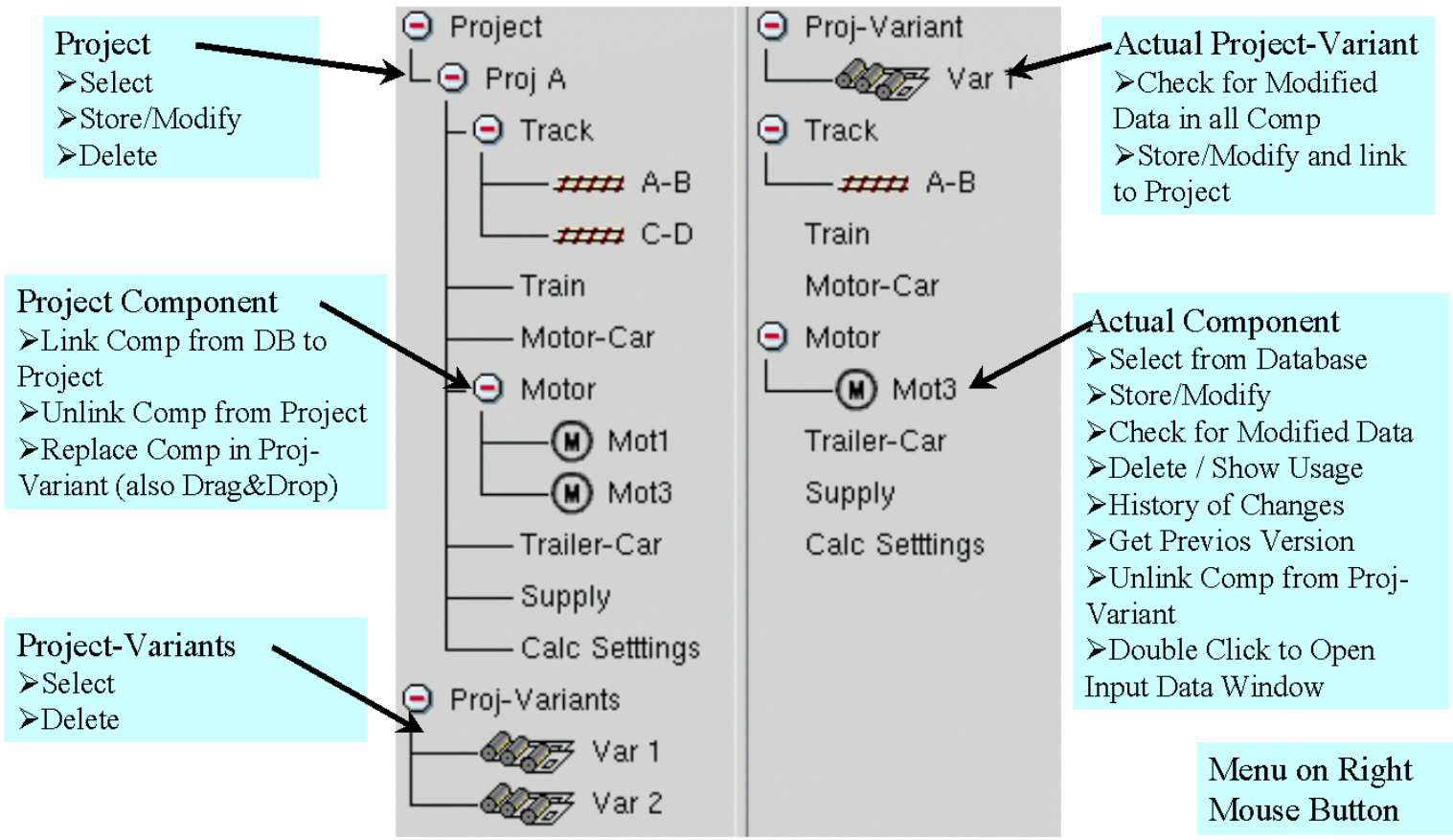
- their input/output operations are file based or
- they are programmed as Mathematica, Matlab or Python functions

with no user interaction and

- the data types of the exchanged items are structured values, scalars, one- or multi-dimensional lists (both numeric or string).

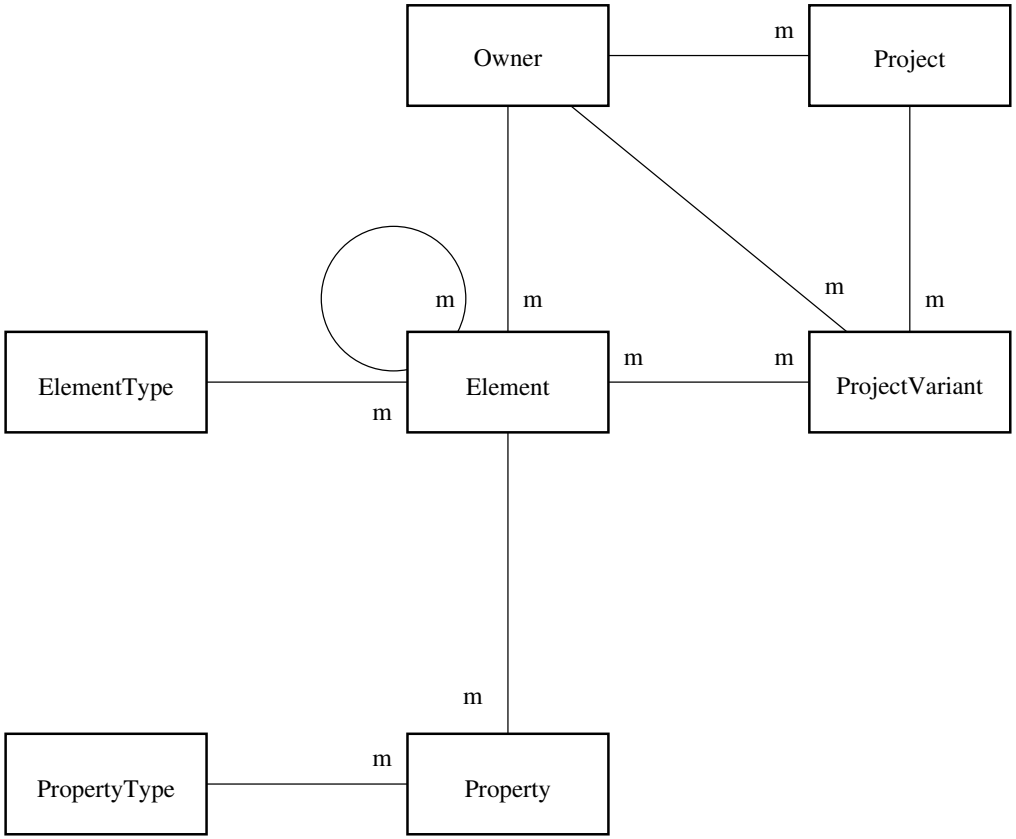


Application Example: Vehicle Engineering





Engineering Database



Configuration Examples: Datapool

Provides a temporary storage for all data values

- allocates/frees memory dynamically
- maintains data consistency/integrity
- records changes and modifications

Each data item has

- a unique name,
- a type: STRUCT, INTEGER, REAL, STRING, COMPLEX,
- a modification time and other attributes: LABEL, UNIT, HELPTEXT ...

and a dynamic, multi-dimensional size



```
DATAPOOL
   cool_med_set = SET OF (
      "Air"    = 1, "Water" = 2, "Oil"    = 3 );

 STRUCT Cooler {
  STRING  {EDITABLE}  name;
  INTEGER {EDITABLE,
           SET=cool_med_set}     CoolingMedium;
  REAL    AirPressure, CoolingFlux, CoolingSpeed;
  REAL   {EDITABLE, FUNC=test_humidity}   Humidity;
  REAL                  Temperature;
 };

 Cooler cooler;
END DATAPOOL;

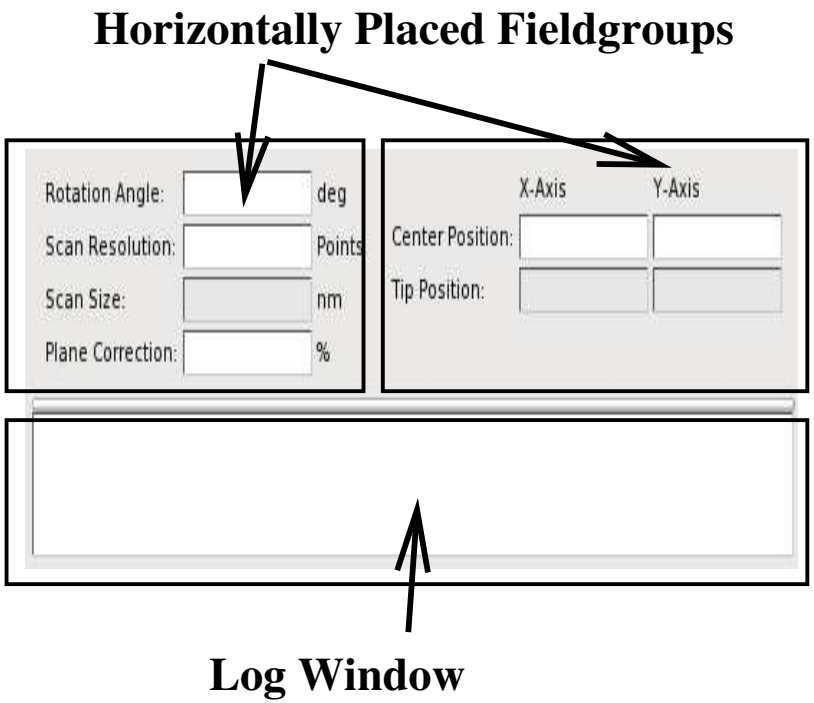
FUNCTIONS
  FUNC test_humidity {
    IF( VALID(THIS.Humidity) {
      IF( ! (THIS.Humidity > 0) && (THIS.Humidity < 1) ){
        SET_ERROR "Impossible value!";
      }
    }
  };
END FUNCTIONS;
```



Creates the requested dialog elements: Menubars, buttons, text fields, labels, plot areas and tables and groups them in form windows.

```
UI_MANAGER
  FIELDGROUP img_fg (
   "Rotation Angle:"
                area.phi:8:1 "deg",
   "Scan Resolution:"
                area.resol "Points",
   "Scan Size:"  area.size   "nm",
   "Plane Correction:" area.corr "%");

FORM main_form{MAIN}(
   (img_fg, pos_fg),
   (LOG_WINDOW{ SIZE=3*50 } )
  );
END UI_MANAGER;
```





Collects the requested input and output streams.
Streams are sequences of data items and strings. They are used for transfering data from and to:

- Calculation (or other external) programs
- document generators (LaTeX)
- Data files
- Printer

```
STREAMER
cin_stream (
    cooler.CoolingMedium,
    cooler.AirPressure
);

cout_stream(
    cooler.Temperature
);

spread_sheet(
  ( Current[]:12, Voltage[]:8, \n )
);

doc_stream (
 "\documentclass{article}
  \usepackage{helvetica,mathfont}
  \docid{", DocId,"}
  \responsibleDpmt{BES3}
  \createdBy{", user.name,"}
    ..."
  );
END STREAMER;
```



Controls the execution of the calculation programs and communicates with the operating system (file I/O and printing) and the external processes.

Processes can be implemented as:

- Mathematica functions,
- Matlab functions,
- executable batch programs.

They are combined in process groups.

```
OPERATOR
  PROCESS
    AsmChar: MATHEMATICA,
    Calc: BATCH {"/local/calc"}
    ;
  PROCESSGROUP
    MathProg {"Mathematica Run"}
      ( mma_out = AsmChar( mma_in );
    )
    BatchProg {"Batch Run"}
      ( bat_out = Calc( bat_in );
    );

  TASK
    BatchT {"Optimize"}{
      Sn = 0;
      WHILE( Sn < SnMax ){
        RUN BatchProg;
    };
END OPERATOR;
```

