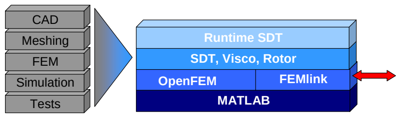
# SDT & OpenFEM
# A technical overview

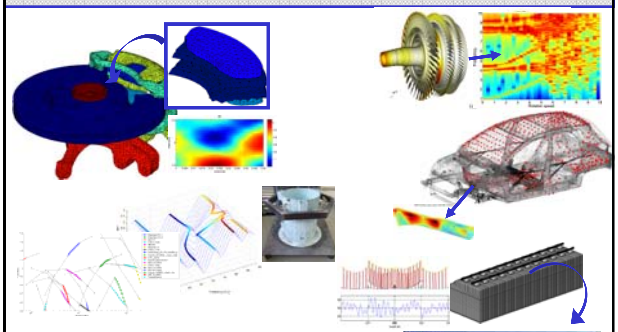

Etienne Balmes
SDTools


## Structural Dynamics Toolbox

- Structural Dynamics Toolbox offers :
  - 3D Finite Element Modeling
  - Experimental Modal Analysis
  - Test / Analysis correlation
- With a modular approach
  - MATLAB environment
  - OpenFEM : Core software for Finite Element Modeling (co-developed with INRIA)
  - FEMlink : Import / export industrial modules
  - Runtime SDT : Customized and standalone compiled applications




## Current SDTools activities



Funding Bosch (squeal), SNECMA (full rotor), PSA (damping design), SNCF (train/track), VALEO (joint non linearities), …

## Why MATLAB ?

Requirements for a long term development environment

| | Matlab | Scilab | Python | Java |
|---|---|---|---|---|
| Ease of development | | | | |
| – Interactive operation on data structures | ☺ | ☺ | ☺ | x |
| – Prior declaration not required (automated memory management) but pre-allocation possible | ☺ | ☺ | 😐 | x |
| – Interactive debugger | ☺ | ☹ | 😐 | ? |
| – Profiler | ☺ | ☹ | 😐 | ? |
| 90 % the code fully portable ⇒ | | | | |
| – virtual machine | ☺ | ☺ | ☺ | ☺ |
| – Performance of interpretation (in-line compiler) | ☺ | ☹ | ? | x |
| Native support of math libraries | ☺ | ☺ | 😐 | x |
| Native support of sparse libraries | ☺ | ☺ | ☹ | x |
| Allow links of few specific libraries to C, C++, Fortran, Java (compile only the small part that needs it) | ☺ | ☺ | ☺ | ☺ |

- MATLAB best (SDT/OpenFEM)
- Python only alternative (ABAQUS, Code_Aster)

Problems

- Memory handling : argument passing without copy often critical (pre-allocation) but user need to pay attention
- MATLAB not perceived as capable of handling large jobs
- NOT FREE : cheap licenses but many users because interactive
- FREE deployment of MATLAB Runtime

## Application areas

Mechanics/dynamics problems are at the interface of many scientific/software areas

| | |
|---|---|
| CATIA, IDEAS, ProEngineer, … | CAD |
| PATRAN, … | Meshing |
| NASTRAN, ABAQUS, ANSYS,... | FEM |
| Adams, Simulink,... | Simulation |
| CADA-X, IDEAS Test, … | Testing |

Simulation ↓ ↑ Validation

## FEM (3) : OpenFEM / SDT architecture

**Preprocessing**
- Mesh manipulations
- Structured meshing
- Property/boundary condition setting

**Import**
- Modulef, GMSH, GID
- NASTRAN, IDEAS, ANSYS, PERMAS, SAMCEF, ABAQUS, MISS, GEFDYN

**FEM core**
- Shape function utilities
- Element functions
- Matrix and load assembly
- Factored matrix object (dynamic selection of sparse library, additional solvers)
- Linear static and time response (linear and non linear)
- Real eigenvalues
- Optimized solvers for large problems, superelements, and system dynamics, model reduction and optimization, vibroacoustics, active control, …
- Drive other software (NASTRAN, MISS)

**Postprocessing**
- Stress computations
- Signal processing
- 3D visualization (major extension , optimized, object based)

**Export**
- VTK, MEDIT
- NASTRAN, IDEAS, SAMCEF
- Ensight, MISS3D, Gefdyn

OpenFEM, SDTools, MSSMat

## OpenFEM

OpenFEM toolbox (Collaboration with INRIA, LGPL license )

- Element library (2-D & 3-D, linear & non-linear, mechanics, acoustics, heat, …), load generation, stress evaluations
- Time and non-linear solvers
- Pre- and post-processing tools

Objectives for the partners:

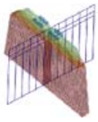
- augment use of general purpose FEM in the MATLAB environment
- INRIA : prototyping environment, demonstrate viability of MATLAB/Scilab type environment for FEM computations
- SDTools : share development costs for non core applications (2-D, non-linear mechanics, …)

## Meshing 1: unstructured

Meshing is a serious business that needs to be integrated in a CAD environment.

OpenFEM is a computing environment.

- IMPORT (MODULEF, GMSH, GID, NASTRAN, ANSYS, SAMCEF, PERMAS, IDEAS)
- Structured meshing
- Run meshing software : GMSH Driver
- 2D quad meshing, 2D Delaunay

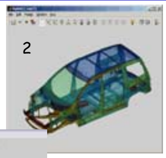


OpenFEM, SDTools

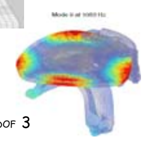
## SDT/FEMLink

- NASTRAN : read/write/drive
- ABAQUS read/write model and results (.fil)
- ANSYS : read model and element matrices, partial write model
- UFF : read/write model
- PERMAS : read/write model and matrices
- SAMCEF read model and matrices, write topology …

- Focus on very large model and matrix support







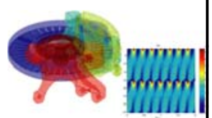
1. ESA ESTEC, 2e6 DOF, 8 GB modal base
2. PSA Peugeot Citroen : body with partial trim 1.2 MDOF
3. Bosch : brake model with contact for squeal

3

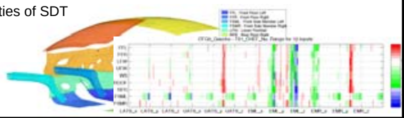
## Why NASTRAN, ANSYS … and SDT ?

Typical reasons to use your FEM and SDT

- Post-processing
  ⇒ extract response from full model output, generate state-space models, visualize, …
- Pre-processing
  ⇒ generate parts of your job, for example in a shape optimization process
- Serve as base for your in-house developments (example Bosch time simulation of squeal)
- Integrate pre- and post-processing in an optimization process (example PSA topology optimization for vibroacoustic response)



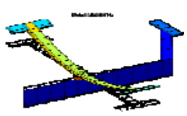
- Model reduction capabilities of SDT



## Meshing 1 : structured example

```
femesh

FEelt=[];
FEnode = [1 0 0 0  0 0 0;2 0 0 0  0 0 .15;
        3 0 0 0 0.4 1.0 .176;4 0 0 0 0.4 0.9 0.176];
% fuselage
femesh('objectbeamline 1 2');
femesh('extrude 0  1.0 0.0 0.0', …
  [linspace(0,.55,5) linspace(.65,1.4,6) 1.5]);
femesh('addsel;');

% vertical tail
femesh('objectbeamline',femesh('findnode z==.15 & x>=1.4'));
femesh(';extrude 3 0 0 .1;addsel;');
% vertical horizontal tail
femesh('objectbeamline',femesh('findnode z==.45'));
femesh('extrude 0  0.0 0.2 0.0',[-1 -.5 0 .5 1]);
femesh('addsel;');

% right drum
femesh(';objectbeamline 3 4;extrude 1 .4 0 0');
femesh('divide',[0 2/40 15/40 25/40 1],[0 .7 1]);
femesh('addsel;');

% left drum
femesh(';symsel 1 0 1 0;addsel;');
```

- Structured meshing
- Mapped divisions
- Objects (beam, circle, tube, …)



```
  Node: [144x7 double]
   Elt: [100x9 double]
    pl: [2x6 double]
    il: [4x6 double]
   bas: []
 Stack: {}
```

## Meshing 2 : femesh/feutil

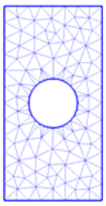
- Add FEeli FEelj, AddSel
- AddNode [,New] [, From i]
- AddTest [,NodeShift,Merge]
- Divide div1 div2 div3
- DivideInGroups
- DivideGroup i ElementSelectors
- EltId
- Extrude nRep tx ty tz
- FindDof ElementSelectors
- GetDof
- Find [Elt,El0] ElementSelectors
- FindNode Selectors
- GetEdge[Line,Patch]
- GetElemF
- GetLine
- GetNode Selectors
- GetNormal[Elt,Node][,Map]
- GetPatch
- Info [ ,FEeli, Nodei]
- Join [,el0] [group i, EName]
- model [,0]
- Matid,ProId,MPID

- ObjectBeamLine i, ObjectMass i
- ObjectHoleInPlate
- Object[Quad,Beam,Hexa] MatId ProId
- Object[Circle,Cylinder,Disk]
- Optim [Model, NodeNum, EltCheck]
- Orient, Orient i [ , n nx ny nz] [,-neg]
- Plot [Elt, El0]
- Quad2Tria, quad42quadb, etc.
- RefineBeam
- Remove[Elt,El0] ElementSelectors
- Renumber
- RepeatSel nITE tx ty tz
- Rev nDiv OrigID Ang nx ny nz
- RotateSel OrigID Ang nx ny nz
- Sel [Elt,El0] ElementSelectors
- SelGroup i, SelNode i
- SetGroup [i,name] [Mat j, Pro k, EGID e, Name s]
- StringDOF
- SymSel OrigID nx ny nz
- TransSel tx ty tz
- UnJoin Gp1 Gp2

Generation, Selection, …

## Meshing 4: fe_gmsh

```
FEnode = [1 0 0 0  0 0 0; 2 0 0 0  1 0 0; 3 0 0 0  0 2 0];
femesh('objectholeinplate 1 2 3 .5 .5 3 4 4');
FEelt=FEel0;femesh('selelt seledge');model=femesh('model0');
model.Node=feutil('getnode groupall',model);

model=fe_gmsh('addline',model,'groupall');
mo1=fe_gmsh('write temp.msh -lc .3 -run -2 -v 0',model);
feplot(mo1); delete('temp.msh')
```



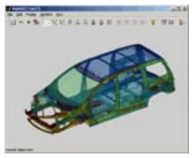
Default element size
GMSH options

- Good functionality for 2D and 3D
- Limited handling of complex surfaces
- Extensions for TETGEN and NETGEN

## Meshing 5: selection

Recursive node and element selections

| | |
|---|---|
| GID ~=i | EltId i |
| Group ~=i | EltInd i |
| Groupa i | EltName ~=s |
| InElt{sel} | EGID == i |
| NodeId > i | Facing > cos x y z |
| NotIn{sel} | Group i |
| Plane == i nx ny nz | InNode i |
| rad <=r x y z | MatId i |
| cyl <=*r i nx ny nz* | ProId i |
| Setname name | SelEdge type |
| x >a | SelFace type |
| x y z | Setname name |
| | WithNode i |
| | WithoutNode i |



## FEM (3) : OpenFEM / SDT architecture

**Preprocessing**
- Mesh manipulations
- Structured meshing
- Property/boundary condition setting

**Import**
- Modulef, GMSH, NetGen, TetGen, GID
- NASTRAN, IDEAS, ANSYS, PERMAS, SAMCEF, ABAQUS, MISS, GEFDYN

**FEM core**
- Shape function utilities
- Element functions
- Matrix and load assembly
- Factored matrix object (dynamic selection of sparse library, additional solvers)
- Linear static and time response (linear and non linear)
- Real eigenvalues
- Optimized solvers for large problems, superelements, and system dynamics, model reduction and optimization, vibroacoustics, active control, ...
- Drive other software (NASTRAN, MISS)

**Postprocessing**
- Stress computations
- Signal processing
- 3D visualization (major extension , optimized, object based)

**Export**
- VTK, MEDIT
- NASTRAN, IDEAS, SAMCEF
- Ensight, MISS3D, Gefdyn

OpenFEM, SDTools, MSSMat

## FEM (1) : why MATLAB ?

MATLAB can be an efficient driver for all of FEM applications
- Very useful versatility in manipulating input to FEM
- Many steps can be performed in MATLAB itself
- MATLAB can be efficiently linked against external libraries (compile only the small part that needs it)
- Debugger and profiler allow quality and optimization

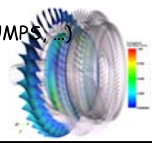
- Easy to manipulate
- Lower development costs
- No loss of performance

But
- Element level computations need to be compiled
- Argument passing without copy often critical
- MATLAB not perceived as capable of handling large jobs
- A different pricing strategy : cheap licenses but many users because interactive (need for SDT Runtime)

## FEM (2) : How big ?

- 64 bit OS no particular limitation : 2000e6 (int32) nodes/DOFs, typical failure for very large dataset : matrix factor (no out-of-core solver), very long time response, …
- 32 bit OS : models that have run 400.000 DOF car body, 150.000 engine blade, generally fails due to memory segmentation (largest block becomes small)

Current SDT trends out-of-core routines
- matrix read/multiply 52 bit (4096 TB)
- storage of datasets (v_handle pointer to data in file) for matrices, time responses, …
- coupling with external solvers (NASTRAN, MUMPS, …)
- Java3D rendering considered



*Multistage rotor with 21e6 nodes, SNECMA*

## ofact : gateway to sparse libraries

Kq=F is central to most FEM problems. Optimal is case/machine dependent. ofact object allows library independent code.

- Method : dynamic selection of method (OpenFEM, SDTools)

```
      lu : MATLAB sparse LU solver
    chol : MATLAB sparse Cholesky solver
 pardiso : PARDISO sparse solver
*umfpack : UMFPACK solver (NOT AVAILABLE ON THIS MACHINE)
->  spfmex : SDT sparse LDLt solver
  mtaucs : TAUCS sparse solver
 sp_util : SDT skyline solver
 *psldlt : SGI sparse solver (NOT AVAILABLE ON THIS MACHINE)
```

- Symfact : symbolic factorization (renumbering, allocation)
- Fact : numeric factorization (possibly multiple for single symfact)
- Solve : forward backward solve (possibly multiple for single fact)
- Clear : free memory

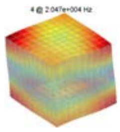
- Not tried : MUMPS, BCS-Lib, …

## ofact : performance test

| 10x10x100 elt<br>36 663 DOF | 10x20x100 elt<br>69 993 DOF | 10x40x100 elt<br>136 653 DOF | |
|---|---|---|---|
| 83 (0.8) | 363 (2.6) | 1706 (5.8) | SPOOLES, PIII 1 Ghz, Linux |
| 10 (0.2) | 90 (2.3) | 262 (6.0) | TAUCS snll + metis, PIII 1GHz Linux |
| | 39 (2.6) | | SPOOLES, AMD 64 4000+ Linux |
| 28 (0.17) | 99 (0.4) | | SPOOLES, Xeon 2.6 GHz, Windows |
| 6.8 (0.48) | 16 (1.1) | | MKL-Pardiso, Xeon 2.6 GHz, Windows |
| 32 (0.64) | | | CHOL Matlab 7.1 (R13SP3) Xeon 2.6 GHz, Windows |
| 56 (0.69) | | | LU Matlab 7.1 (R13SP3) Xeon 2.6 GHz, Windows |

*Fact (solve) CPU seconds*

- *All libraries can be accessible (OpenFEM, SDTools), best is application/machine dependent.*
- *Memory usage and fragmentation is another issue that may drive library selection*




## OpenFEM history

Start in 2001 from

- *Structural dynamics toolbox* .m file elements limited library
- *MODULEF* large library but no longer a convenient prototyping environment

Phase I -> OpenFEM 1.0 & 2.0

- Port of MODULEF elements (2D and 3D volumes, MITC4)
- Translation to SCILAB (stopped in 2006)

Phase II (current)

- Efficient non-linear operation (generic compiled elements, geometric non-linear mechanics, … )

## Design criteria

- Be a toolbox (easy to develop, debug, optimization only should take time)
- Optimize ability to be extended by users
- Performance identical to good fully compiled code
- Solve very general multi-physics FE problems
- Be suitable for application deployment

## Easy user extensions

- Object oriented concepts (specify data structures and methods)
- But non typed data structures (avoid need to declare inheritance properties)

Example user element

- Element name of .m file (beam1.m)
- Must provide basic methods (node, DOF, face, parent, …)
- Self provide calling format. Eg : beam1('call')
  [k1,m1]=beam1(nodeE, elt(cEGI(jElt),:), pointers(:,jElt), integ, constit, elmap, node);

Other self extensions

- Material functions
- Property functions (problem formulations)
- Non-standard topology definitions

## Shape function utilities (integrules)

Supported topologies are

- node1 (0 d topology)
- bar1 (1D linear)
- beam1,beam3 (1D cubic)
- quad4 (2D bi-linear), quadb (2d quadratic)
- tria3 (2D affine), tria6 (2D quadratic)
- tetra4, tetra10
- penta6, penta15
- hexa8, 20, 21, 27

Standard quadrature rules

```
» integrules('hexa8',3)
          N: [27x8 double]
         Nr: [27x8 double]
         Ns: [27x8 double]
         Nt: [27x8 double]
         Nw: 27
        NDN: [8x108 double]
  NDNLabels: {'' ',x' ',y' ',z'}
       jdet: [27x1 double]
          w: [27x4 double]
      Nnode: 8
         xi: [8x3 double]
       type: 'hexa8'
```

```
>> integrules('gauss q2d')
    [ -3]               []    'order default'
    [ -2]    [ 0x1 double]    'node'
    [ -1]    [ 1x4 double]    'center'
    [102]    [ 4x4 double]    'gefdyn 2x2'
    [  2]    [ 4x4 double]    'standard 2x2'
    [109]    [ 9x4 double]    'Q4WT'
    [  9]    [ 9x4 double]    '9 point'
    [  3]    [ 9x4 double]    'standard 3x3'
    [  2]    [ 4x4 double]    'standard 2x2'
    [ 13]    [13x4 double]    '2x2 and 3x3'
```

## Formulation library

Generic compiled elements

- Support any linear element without recompilation
- 3D elasticity with full anisotropy, 2D plane stress/strain
- 2 & 3D acoustic fluids, fluid/structure coupling
- Heat equation
- Piezo-electric volumes, poro-elasticity
- Gyroscopic effects

Other compiled families

- geometrically non-linear mechanics, mechanical or thermal pre-stress
- Hyper-elasticity, follower pressure

*Other elements*

- Shells. Laminated plate theory. Piezoelectric shells.
- 3D lines/points : Bar, Beam, Pre-stressed beam, spring, bush, mass

The OpenFEM specification is designed for multiphysics applications

99 DOF/node
999 internal DOF/element

SDT

# User elements


```
elseif  comstr(Cam,'node');  out = [1 2];
elseif  comstr(Cam,'prop');  out = [3 4 5];
elseif  comstr(Cam,'dof');   out=[1.01 1.02 1.03 2.01 2.02 2.03]';
elseif  comstr(Cam,'line');  out = [1 2];
elseif  comstr(Cam,'face');  out =[];
elseif  comstr(Cam,'sci_face'); out = [1 2 2];
elseif  comstr(Cam,'edge');  out =[1 2];
elseif  comstr(Cam,'patch');  out = [1 2];
elseif  comstr(Cam,'parent');   out = 'beam1';
```

```
[k1,m1]=beam1(nodeE, elt(cEGI(jElt),:),
pointers(:,jElt), integ, constit, elmap, node);
```

```
[ID,pl,il]=deal(varargin);
pe=pe(find(pe(:,1)==ID(1),3:end); % material properties
ie=ie(find(ie(:,1)==ID(2),3:end);

%          E*A nu      eta  rho*A       A lump
constit = [pe(1)*ie(4)  0    pe(3)*ie(4) ie(4) ie(7)];
integ=ID;%matid proid
Elmap=[];

out=constit(:); out1=integ(:); out2=ElMap;
```

# Generic compiled elements

Objective ease implementation of

- linear element families
- arbitrary multi-physic
- good compiled speed
- provisions for non linear extensions

Assumptions

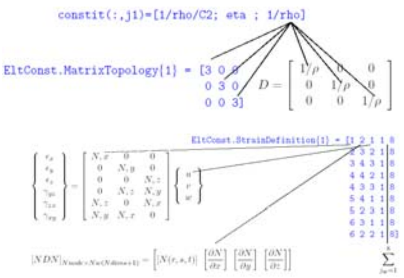
- Strain ε=[B]{q} linear function of N and ∇N
- Element matrix quadratic function of strain

$$k^{(e)} = \sum_{ji,jj} \sum_{jw} \left[ \{B_{ji}\} D_{ji\,jk}(w(jw)) \{B_{jj}\}^T \right] J(w(jw)) W((jw))$$

# Generic compiled elements

During assembly init define

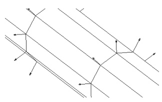
- D ↔ constit
- ε ↔ EltConst.NDN
- $K_e$ ↔ D,e (EltConst. MatrixIntegrationRule built in integrules MatrixRule)




EltConst=p_solid('constsolid','hexa8',[],[])
p_solid('constsolid','hexa8',[],[])
p_solid('constfluid','hexa8',[],[])

# OpenFEM current developments

- Support shape function tricks (MITC, average strain, …)
- Extend FieldAtNode interpolation
- 3D material orientation maps
- Better support of analytic expressions for loads



Waiting for real need

- Improve ease of development of NL constitutive laws
- Extend parallel operations beyond assembly

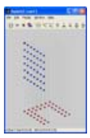
*INRIA MACS Simplified modeling of cardiac contraction with OpenFEM*

# Boundary conditions

Cases define :

boundary conditions, point and distributed loads, physical parameters, ...

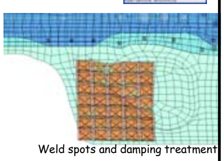
```
data=struct('sel','x==-.5', ...
          'eltsel','withnode {z>1.25}', ...
          'def',1,'DOF',.19);
model = fe_case(femesh('testubeam'),...
     'FSurf','Pressure load',data, ...
     'FixDof','Fixed boundary condition','x==0');
```



Supported boundary conditions

- KeepDOF, FixDOF
- Rigid
- MPC, Un=0

Typical handling by elimination



Weld spots and damping treatment

$$\left[Ms^2 + Cs + K\right]\{q(s)\} = [b]\{u(s)\}$$
$$\{y(s)\} = [c]\{q(s)\}$$
$$[c_{int}]\{q(s)\} = 0$$

$$\left[T^T M T s^2 + T^T C T s + T^T K T\right]\{q_R(s)\} = \left[T^T b\right]\{u(s)\}$$
$$\{y(s)\} = [cT]\{q_R(s)\}$$

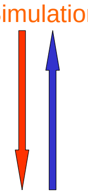
$$\text{range}([T]_{N\times(N-NC)}) = \ker([c_{int}]_{NS\times N})$$

# Application areas

Mechanics/dynamics problems are at the interface of many scientific/software areas

| | |
|---|---|
| CATIA, IDEAS, ProEngineer, … | CAD |
| PATRAN, … | Meshing |
| NASTRAN, ABAQUS, ANSYS,... | FEM |
| Adams, Simulink,... | Simulation |
| CADA-X, IDEAS Test, … | Testing |

Simulation

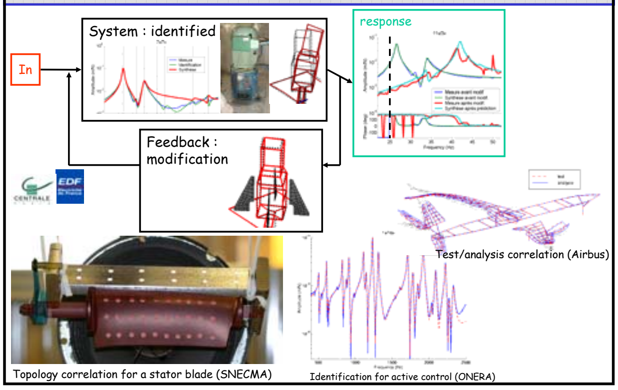


Validation

## SDT & not OpenFEM

- Large model optimization
- feplot (interactive mesh viewer), iiplot (curve viewer)
- Model reduction & superelements
- Sensors
- Experimental modal analysis
- Cyclic symmetry
- Active control with piezoelectric
- Parametric model analysis
- Non-conform mesh matching

## SDT & Vibration testing/EMA



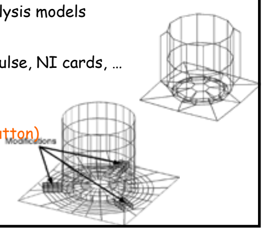

## SDT & Vibration testing/EMA

MATLAB is perfectly adapted for Experimental Modal Analysis

SDT provides
Frequency domain identification
Test / analysis correlation (topology, expansion, correlation, …)
GUI based ODS analysis

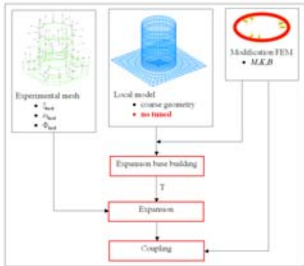
Current development in the following areas
Structural modification, hybrid test/analysis models
Model updating
In operation identification
Improved links with acquisition Photon, Pulse, NI cards, …

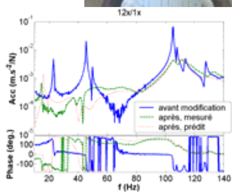
Main limitation (target audience)
Matlab based Toolbox
rather than GUI application (full push-button)




## SDM, operational loads

- Objective predict the effect of highly dissipative modifications on structures known from experiment
- In operation loads (injectors for PSA)



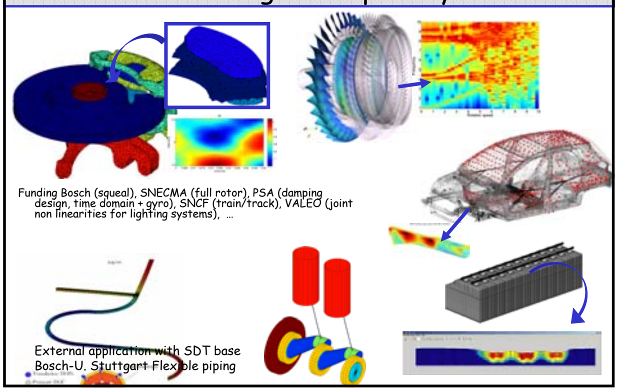




## SDT & Superelements

Superelement :
- group of elements identified by a name
- stored as a sub-model
- possibly reduced $\{q\}=[T]\{q_r\}$
- Possibly reused (sectors, slices, …)

SDT provided utilities
- Select in model (and dispatch constraints)
- Display full and partial
- Partial recovery for reduced, support reduced sensors
- Reduction (Craig-bampton, free modes, many variants, possibly parameterized)
- Node renumbering

## Substructuring & complex systems



# Model reduction

Substructuring and complex structures

Established
- Multi-level substructuring, Parametric analysis

Current trends
- Component design within whole
- Time domain, contact/friction
- Thermo-elastic coupling
- Time domain damping, self-heating, …

- Stabilize procedures for several million DOFs

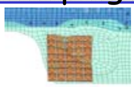
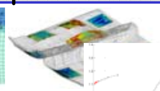
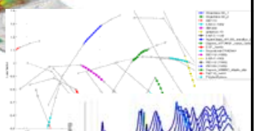
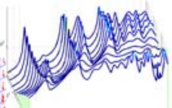
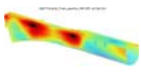
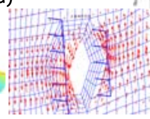
Vibroacoustics

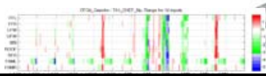
**Established techniques**
- Acoustic FEM modeling
- Coupling matrices
- Coupled predictions
- Reduction methodology (critical for heavy fluids)

Current trends
- Lining materials (poroelasticity)
- Design & robustness
- Industrial process automation

# Damping procedures

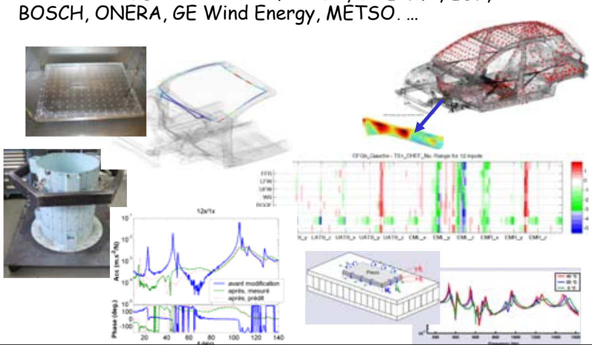
- Meshing
- Material handling
- Response predictions (huge challenge, but well established)
- Analysis tools
- Vibroacoustic optimization



# Damping projects

- In this area SDTools worked for PSA, SNECMA, EDF, BOSCH, ONERA, GE Wind Energy, METSO. …



# Damping issues

Established techniques undergoing refinement
- Damping device (re)-meshing
- Reduction methods for direct FRF & modal

Opened issues
- Device placement & shape optimization techniques
- Design methodologies (initial evaluation & validation)
- Viscoelastic behavior & non linearities (time domain simulation, brake lining, engine supports, suspensions, …).
- Thermally challenging environments & self heating, thermoelasticity
- Rethinking the use of piezoelectric shunts to damp vibrations above 150C

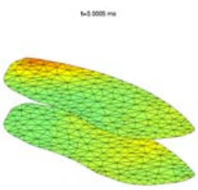
- Improve material knowledge
- Help building convincing prototypes

# Vibration & surface contact

Established
- Optimized time integration of small displacement contact/friction coupled with superelement reduction

Current trends
- Design methodologies for time based simulations (application : squeal with BOSCH)
- Equivalent linear design parameters for systems with gaps & friction (probable EDF)
- Weld spot, rivet, bolt, … dissipation (unfunded)

Possible (seen as related)
- Lubricated joints, high velocity abrasion or machining,



# Thanks to …

AD SANDVIK COROMANT , AEROSPATIALE MATRA AIRBUS, AEROSPATIALE MATRA LANCEURS, AFIT ENY, ALCANTAR & BROWN , Agency for Defence Development, Alenia Aeronautica, Finmeccanica, Arnold Air Force Base, Aérospatiale Matra Lanceurs, BAE Systems, BBN Technologies, BOFORS SA , BOSCH SYSTEMES DE FREINAGE S.A.S, Bassin d'Essai des Carènes, Boeing Space and Communications, Bridgestone/Firestone, Bruel & Kjaer, CAMBRIDGE CONTROL Ltd, CCS Informationssysteme GmbH, CERN, CETIM, CIRA, CNAM, CNR - ITESRE, CNR ITIA, CNRS - GDR Vibroacoustique, CNRS - L.M.A., CNRS LULI, CSIR - defencetek, Case Western R. University. , Ceanet, Centro Ricerche Fiat, Centro de Investigacicon Cientifica de Yuca..., Chalmers University of Technology, Cybernet Systems Co., Ltd., Czech TU, D.I.C.E Tokyo Denki University , DALHOUSIE UNIVERSITY, DISAT Università dell'Aquila, DISTART, DLR, DLR e.V., DORNIER LUFTFAHRT, DORNIER SATELLITENSYSTEME Gmbh, Daewoo Shipbuilding & Marine Engineering Co., Ltd., DaimlerChrysler AG, Deh Yu College of Nursing & Mgmt, Denel Aviation, Det Norske Veritas, Technical Consulting, Deutsche Bahn AG, EADS LV, ECOLE CENTRALE DE LYON, ECP, EDF, EDF , EDF - POLE INDUSTRIE, EMBRAER, ENSAM - CER FRANCO - ALLEMAND , ENSIB, ENSIM- UNIVERSITE DU MAINE, ENSTA, Ebara Research Co, Ltd, Ecole Centrale de Lille, Ecole Centrale de Lyon, Eindhoven University of Technology, Elliptec Resonant Actuator AG, Elliptec co. Siemens, Etablissement Technique de Bourges, F.G. UNIVERSIDAD POLITECNICA MADRID, FH AALEN, FH Wilhelmshafen, FORSCHUNGSZENTRUM KARLSRUHE , FUNDACION GENERAL DE LA UNIVERSIDAD POLITECNICA, Fachhochschule Aalen, Fachhochschule Braunschweig / Wolfenb|ttel, Fairchild Dornier GmbH, Ford Motor Company, Fugro Geotechnical Services (HK) Ltd, Hutchinson SA, IABG gmbh, IFMA, INGEMANSSON TECHNOLOGY , INRETS, INRIA, INSA Lyon, INSA Rouen, INSA de Lyon, INSTITUT PASTEUR, IPSE, IRCAM, ISVR, Imperial College, Institute for Defense Analyses, Institute of Advanced Computing , Instituto Tecnológico de Aeronáutica-CTA, JPL, JRC ISPRA, KAIST, KOYO SEIKO CO.,LTD., KUL, KUL - BWK, Kaist University, Kanagawa Institute of Technology, Kanazawa Univ., Kurashiki Kako Co., Ltd., Kwangju Institute of Science and Technology, Kyoto Univ., Kyoto Univ. , L.C.P.C. NANTES, LASPI, LCPC, LEIBNIZ-RECHENZENTRUM , LMT, LNE, LOCKHEED MARTIN MISSILE & SPACE CO, LRBA, LTH, Lund University, Laboratory for Experimental Audiology, Lawrence Livermore national Laboratory, Lockheed Martin Aeronautics Co, Lucent Technologies - Bell Laboratories, MDI, MPI für Mathematik, Magnecomp Corp., Meiji University, School of Science and Tec..., Military University of Technology, Mitsubishi Electric Corporation, Morgan State University, NASA Ames Research Center, NASA Goddard Space Flight Center, NASA Langley Research Center, NASA Marshall Space Flight Center, NEC, NSWCCD, Nanjing University of Aeronautics & Astronautics, Nanyang Technological University, National Technical University of Athens, Naval Air Warfare Center Aircraft Division, Nihon Univ., Northwestern University, ONERA, ONT, OPGRAMAMOWANIE NAUKOWO-TECHNICZNE, ORBITAL, ORBITAL SCIENCES CORPORATION , Oak Ridge National Laboratory, Ono Sokki Co., Ltd., POLISH ACADEMY OF SCIENCES, PRODERA, PSA Peugeot Citroën, Pennsylvania State University, Pirelli Informatica Sistemi Informativi Ricerca e Sviluppo, Politecnico di Bari, Progettazione e Produzione, Politecnico di Torino, Politecnico di bari Vie e Trasporti, Politecnico di torino, Pratt & Whitney, Pratt & Whitney Aircraft, RENAULT VI, RENSSELAER POLYTECHNIC INSTITUTE, RWTH Aachen, Renault Sport, Rolls-Royce Marine Power, Ruhr- Universität Bochum, SANDIA NATIONAL LABORATORIES, SCHENCK PEGASUS GMBH, SCHOOL OF ELECTRONIC & ELECTRICAL , SECOND SOURCE ELECTRONIC INC, SEGIME, SEP EC, **SERAM - ENSAM**, SILESIAN TECHNICAL UNIVERSITY, SKF E&R Center, SMC Corporation, SNAMProgetti, SNCF, SNECMA, SSPA Sweden AB, STANDARD PRODUCTIONS LIMITED, Saarland University, Sandia National Lab, Sandia National Laboratories, School of Mechanical, Materials, Manufacturing Engineering and Management, Shneider Electric, Siemens A&D, Siemens PL, Sony Corp., Southern Illinois University, TECHNICAL UNIVERSITY BRATISLAVA , TECHNICAL UNIVERSITY OF ZIELONA GORA , TNIT, TNO TPD - TU Delft, TNO-Building and Constructions Research, TRW Lucas Aerospace, TU Berlin, TU Braunschweig, TU Braunschweig, IWF, TU Chemnitz , TU Cottbus, TU Darmstadt, TU München, Takaoka Electric MFG Co, Ltd, Technical University of Szczecin, Technion, Technische Universitaet Clausthal, Technische Universitaet Hamburg-Harburg, The MathWorks GmbH, The MathWorks, Inc., Tokyo Electric Power Svcs Co Ltd, Toshiba Corp., UDS Ancona, UDS Catania, UDS Ferrara, UDS Genova, UDS Lecce, UDS Modena e Reggio Emilia, UDS Padova, UDS Pisa, UDS Roma, UDS Roma, Aerospaziale, UDS Trento, UDS Trieste, UDS Trieste, Energetica, UNIV. CLAUDE BERNARD LYON 1, UNIVERSIDAD DE OVIEDO, UNIVERSITE DE REIMS , UNIVERSITE DE TOURS, UNIVERSITY OF BIRMINGHAM, UNSW, UPMC Paris 6, UTRC, Univ. of Shiga Prefecture; The, Universidad Politecnica de Cartagena, Universidad del País Vasco, Universidade de Aveiro, Universitaet Bremen, Universitaet Stuttgart, Universitaet Wuppertal, Universite De Marne La Vallee, Universite Libre de Bruxelles, University of Bristol, University of Central Florida, University of Liverpool; The, University of Mississippi, University of Missouri Rolla, University of Pretoria, University of Queensland, University of Sheffield, University of Sheffield; The, University of Southampton, University of Stellenbosch, University of Thessaly, University of Tokyo; The, University of Wales Swansea, Università di Roma La Sapienza, Universität GH Siegen, Universität Hannover, Universität Kassel, Universität Weimar, Université de Reims Champagne Ardenne. Université de Valenciennes, Université du Littoral Côte d'Opale, VALEO SYSTEME D'ESSUYAGE, VATTENFALL Data AB, VTT Manufacturing Technology, Vi & Sound Solutions Limited, Virginia Tech, Volvo Trucks of North America, Inc., WICHITA STATE UNIVERSITY, Yamaha Corporation, ZVVZ a.s., enea, takum kenthiku kouzou kenkyusyo