



QSA scripting for (semi-)automatic valve train simulation

**Dr. Carsten Träbing
Jürgen Butzbacher**

Schaeffler Technologies GmbH & Co. KG

SIMPACK User Meeting, May 19, 2011

QSA scripting for (semi-)automatic valve train simulation

Overview



- Schaeffler Group
- Background
- QSA tool
- Programming details
- Summary

QSA scripting for (semi-)automatic valve train simulation

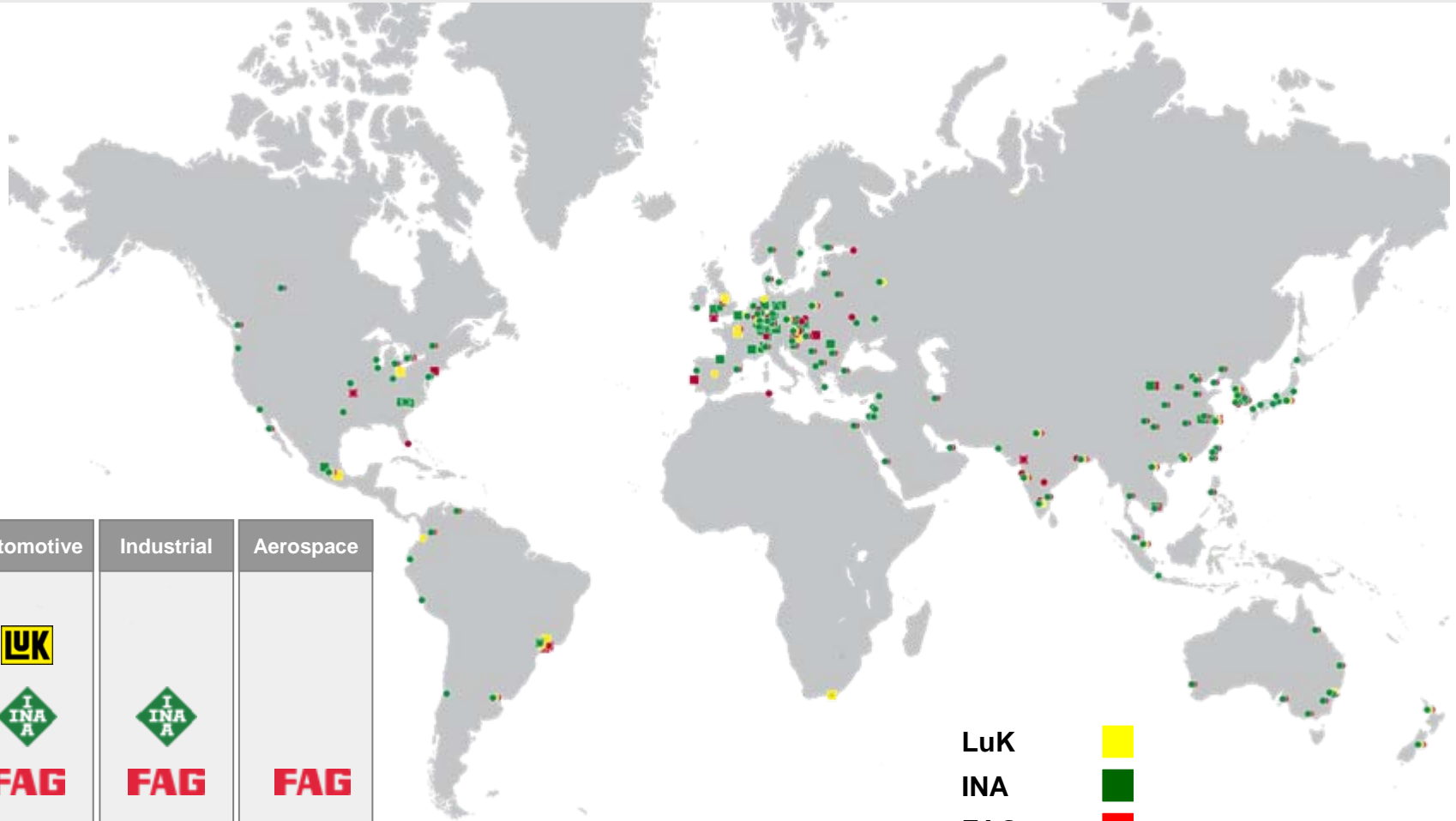
Schaeffler Group



Employees
worldwide: approx. 67,500

Sales 2010
worldwide: 9.5 bn euros

180 locations
in more than 50 countries



Automotive	Industrial	Aerospace

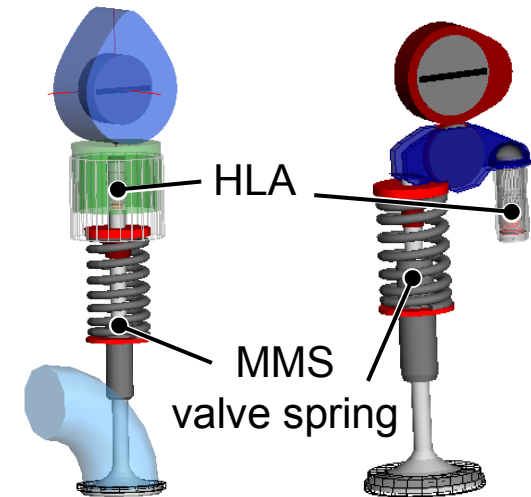
LuK
INA
FAG

QSA scripting for (semi-)automatic valve train simulation

Background

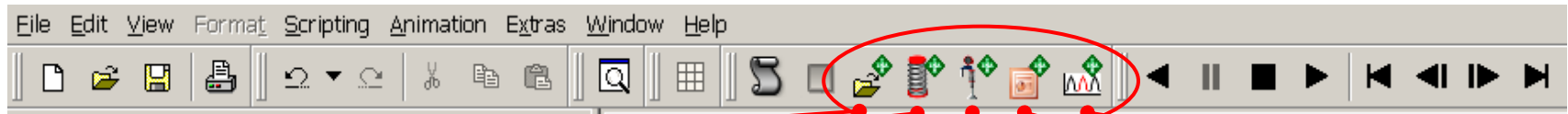


- Dynamic simulation of valve trains since 1994 at Schaeffler
- Semi-automatic in-house DOS tool exists since 1998
 - Advantages:
 - easy, comprehensible input (one text file)
 - post processing integrated
 - Disadvantages:
 - simplified MMS valve spring (one body/coil)
 - no enhancements of model
- Requirements for similar SIMPACK tool:
 - as easy to use as previous tool
 - fast and comprehensible
 - one input data sheet
 - automatic post processing
 - up-to-date spring modeling
 - prepared for RFF, tappet, type V, rocker arms, complete camshaft, ...
 - handling of different versions of database (model, scripts)
 - ...



QSA scripting for (semi-)automatic valve train simulation

QSA tool



Project
handling

Create/verify
valve spring

Single valve train simulation
incl. pre/post processing

PPT
export

Set working
cycle

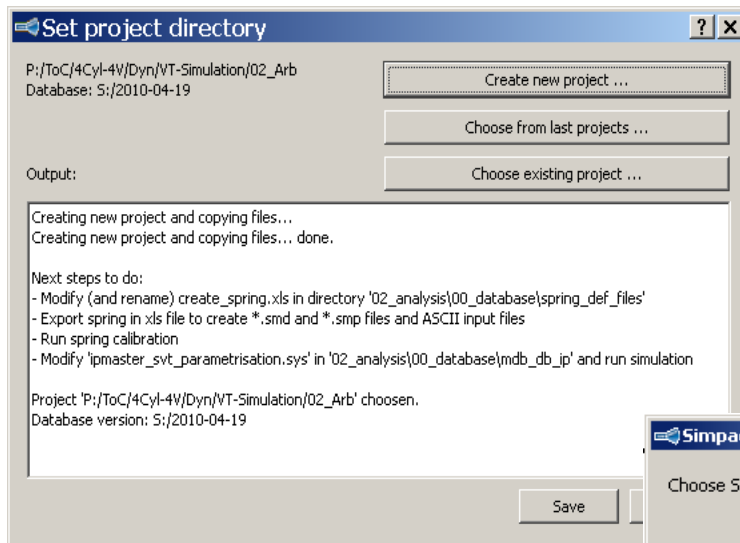
- QSA chosen for realization
- tools are accessed from post processor
- access by additional icons

QSA scripting for (semi-)automatic valve train simulation

QSA tool

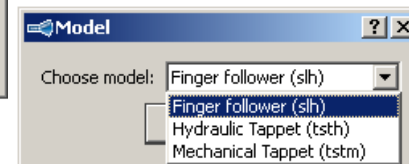
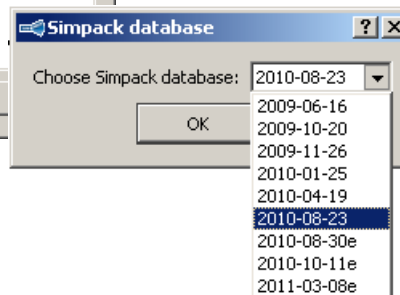


Project handling



Tasks:

- create new project / choose existing project
- set current project
- create directory structure of new project

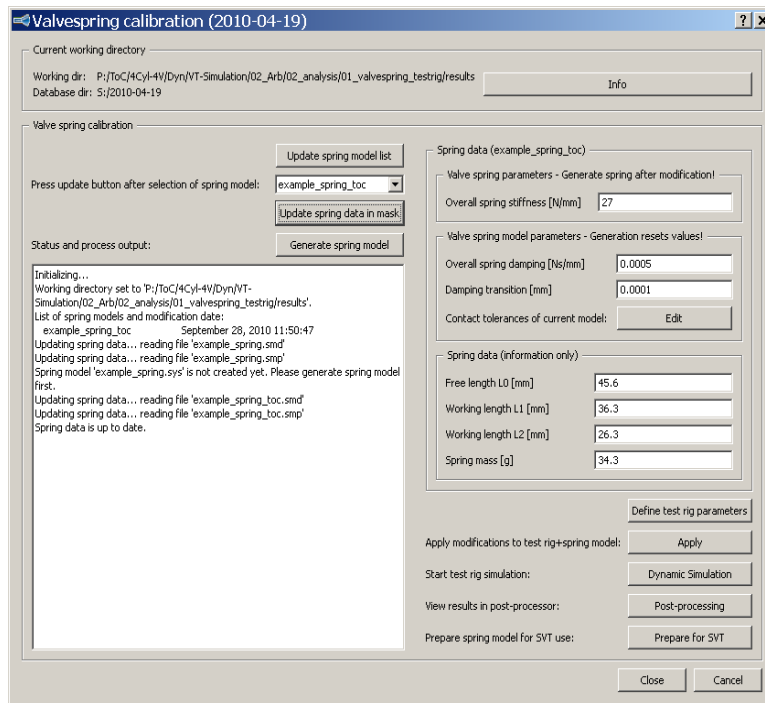


QSA scripting for (semi-)automatic valve train simulation

QSA tool

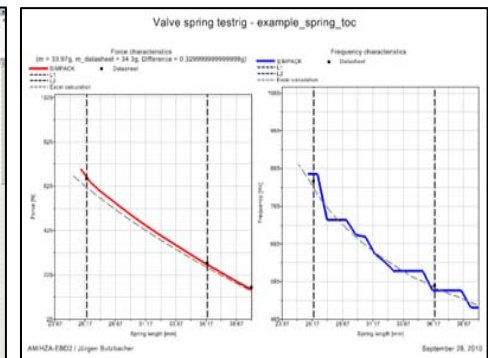
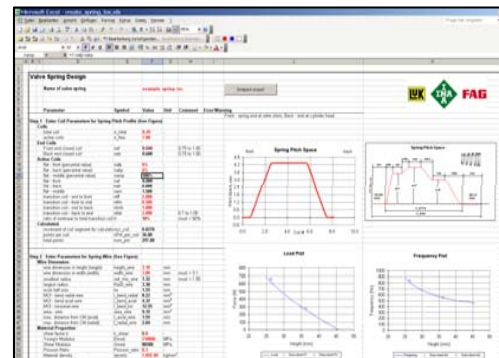


Create/verify
valve spring



Tasks:

- create valve spring from center line
- run dynamic simulation
- automated post processing (force, frequency)
- prepare for simulation (set to installation length)



QSA scripting for (semi-)automatic valve train simulation

Programming details



Scripts written in cooperation with Intec GmbH/SIMPACK AG

Complexity

- 11 files
- 140 functions
- 12.000 lines of code

Challenges

- new programming language QSA
- handling of different database versions
- feedback on progress of simulation

QSA scripting for (semi-)automatic valve train simulation

Programming details



New programming language QSA

- QT Script for Applications (invented by Trolltech)
- simple and easy (comparable to Python)
- good help system
- complete set of control statements (including exception handling)
- nearly everything is possible or can be done by calling a shell command (e.g. for registry access or DDE commands)
- GUI capability is limited/inconsistent
 - no pictures
 - arrangement of widgets limited
 - in some widgets HTML syntax (font, size, color) of text is enabled (e.g. `MessageBox`, `PushButton.label`), in others it isn't (`PushButton.text`, `GroupBox.title`)
 - some widgets allow event triggered actions (e.g. `LineEdit`), others don't (`ComboBox`)
- inconsistent set of commands
 - e.g. `match()`, `find()`, `search()`, `startsWith()`, `endsWith()` can be used for same purpose
 - array indices can be defined by strings, e.g. `array["text"] = 1`, but a list of the indices is not provided

QSA scripting for (semi-)automatic valve train simulation

Programming details

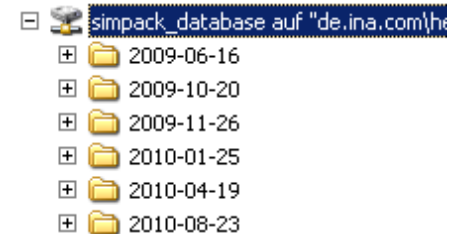


Handling of different database versions

- Requirement: Always ensure access to existing projects
- Solution: Save database version of project and access QSA scripts of related database using a wrapper. SIMPACK script directory contains only one file "wrapper.qs" to manage database handling.

```
function WRAPPER_svt() { // called by GUI
    // get database version
    dbVersion = WRAPPER_readDBFile();
    // set global variables
    WRAPPER_setGlobalVars(dbVersion);
    // read scripts of db version of project
    WRAPPER_readScripts(dbVersion);
    // call function in database, function
    // needs to be defined in each database
    SVT_main();
}
function WRAPPER_readScripts(dbVersion) {
    // define paths where to find scripts
    Script.resetSearchPath();
    Script.addSearchDir("C:"+dbVersion)
    // read all scripts
    Script.include("script1.qs");
    Script.include("script2.qs");
    ...
}

// define global vars of all databases,
// otherwise post processor crashes
var globalVar1;
var globalVar2;
...
function WRAPPER_setGlobalVars(dbVersion) {
    switch (dbVersion) {
        case "2009-06-16":
            globalVar1 = 1;
            globalVar2 = 100;
            ...
            break;
        case "2009-10-20":
            globalVar1 = 2;
            globalVar2 = 300;
            ...
            break;
        ...
    }
}
```



QSA scripting for (semi-)automatic valve train simulation

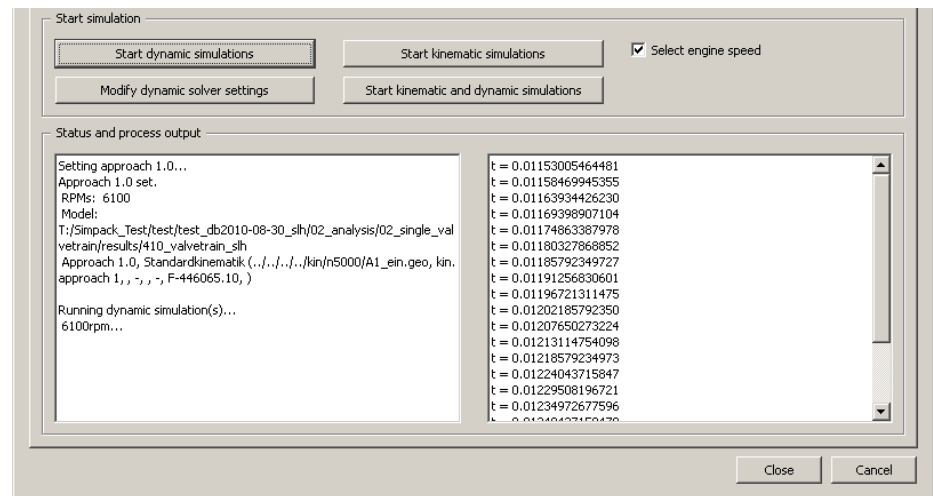
Programming details



Feedback on progress of simulation

- Requirement: User feedback while simulation is running in background
- Solution: Run SIMPACK in asynchronous mode

```
function callSimpackAsync(command) {
    var p = new Process;
    var out = "";
    var cmd = System.getenv("SIMPACK") + "/simpack/sys/win32/msys/bin/sh -t -c '"+command+"'";
    p.arguments = cmd.split(" ");
    p.start();
    while (p.running) {
        for (var i=0; i<100000; i++) {} // sleep
        out += p.readStdout(); // get output from stdout
        // print last 20 lines of 'out' (otherwise just beginning of text is shown)
        textEdit_UserInfo.text = last20LinesOfText(out);
    }
}
```



QSA scripting for (semi-)automatic valve train simulation

Summary



Resume after nearly two years of usage

A tool as easy to use as the former tool was created using QSA with the following additional advantages:



- user-optimized handling (GUI)
- effort to adapt tool to new version of SIMPACK 8.9xx takes 5 minutes
- enhancements of model (e.g. complete system) are possible
- better model of valve spring
- duration of a standard project was reduced by 20%

Planned enhancements

- type V and rocker arm models
- support to build up camshaft with associated valve trains

SIMPACT requirements

- revision of QSA language, e.g. triggered actions for ComboBox widget, consistent set of commands
- faster handling of result files in post processor

Together we move the world.
So you can move the world.



Thank you for your attention.