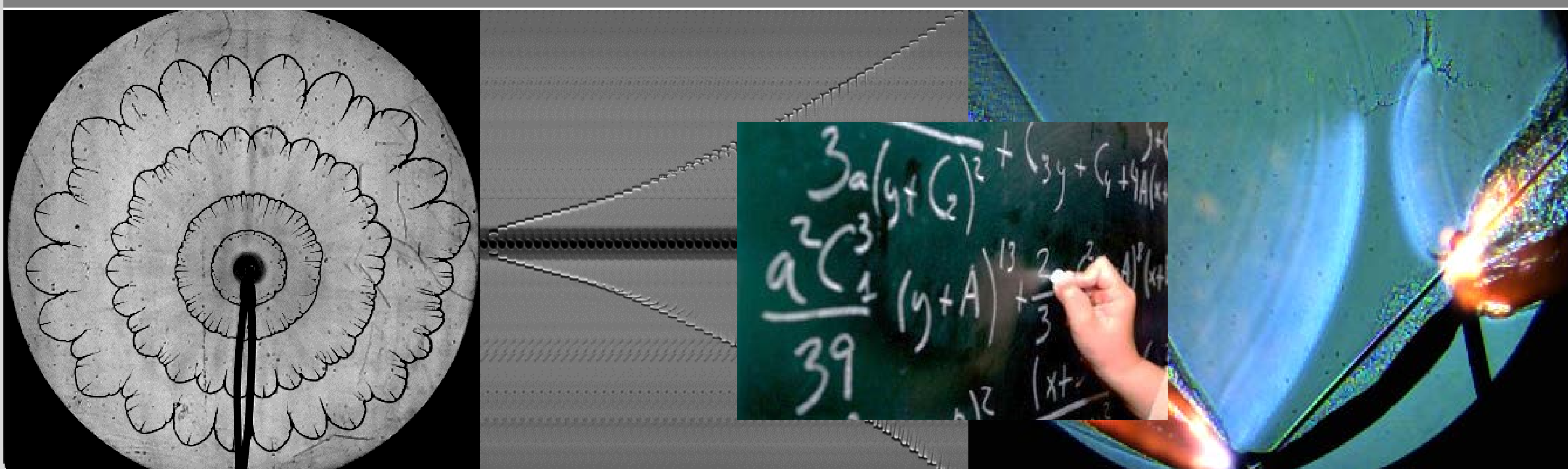
Karlsruhe Institute of Technology

# Update on hazard assessment toolkit project from KIT

**T. Jordan**

Hydrogen Group
Institute for Nuclear and Energy Technologies (IKET)



# Basic Concept

- Free and open set/collection of tools for risk assessment
- Based on published engineering correlations
- Highly modular „dispatched“ design
- Easy and safe to use, on- and offline
- Fast response times
- Well documented, quality assured
- Commonly defined and developed

⇨

- **Based on IEA HIA spirit**
- **Real IEA HIA product**

# Current Status

- At least 4 different implementations with different characteristics and 3 different drivers on the way:
  - Canadian activity (Benard) based on Seaside
  - US DoE activity (Groth) Sandia QRA Toolkit based on C#
  - EC H2FC activity (diverse) based (partially) on SAGE and Model FX

⇨

- **Threat of Fragmentation**
- **Too small user community and international character require agreement on common unified approach**

# Basic Concept for SAGE implementation: Model

Input → Model → Output / Results

**Input**

- Initial and boundary conditions
- Geometry (confinement, blockage)
- Inventory,…

**Model**

- Event, Phenomenon
- Physics
- Simplifications
- Implementation (Formula)
- Limitations
- Reference
- Test

**Output / Results**

Probabilistics

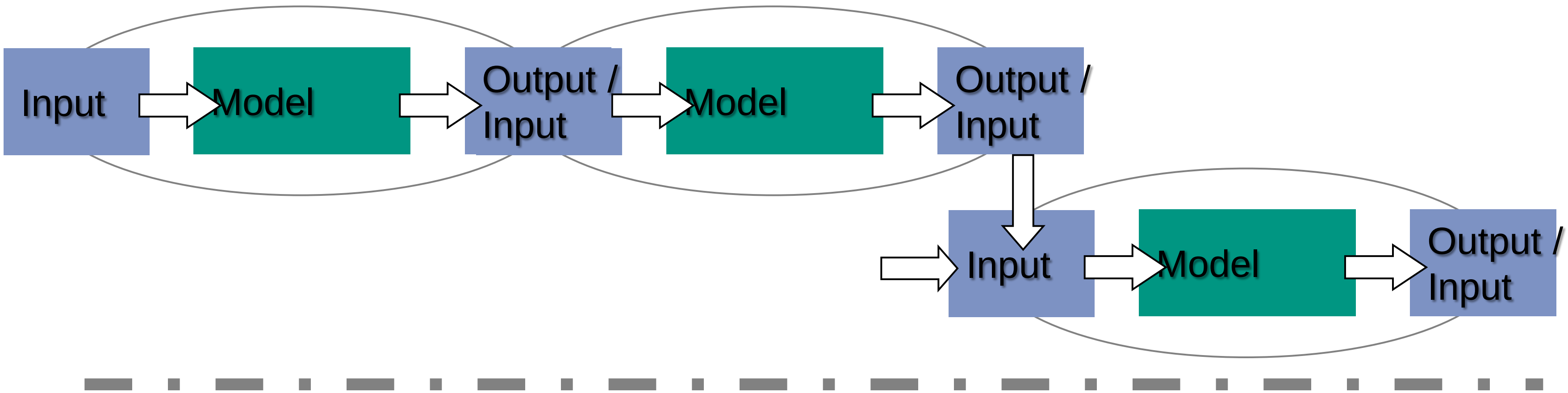
Consequence

Input

Constants: c1, c2,…. Variables: i1, i2, …, o1, o2, …

# Basic Concept: Super-Model

- Every model provides a function:
  Output = Model(Input) where
  with „Output“ and „Input“ are sets of variables
- „Super-Models“ are composed of several models →
- Recursion




Constants: c1, c2,…. Variables: i1, i2, …, o1, o2, …

# Phenomena Addressed

| Major Phenomena | Phenomena | Sub-Phenomena | Further Simplifications |
|---|---|---|---|
| **Thermodynamics / State** | | | |
| | Equilibrium of State | | ideal gas simplifications |
| | Equilibrium of State | | ideal gas simplifications |
| | Equilibrium of State | | ideal gas simplifications |
| | Equilibrium of State | | real gas model |
| | Equilibrium of State | | real gas model |
| | Equilibrium of State | | real gas model |
| **Release / Mixing** | | | |
| | Permeation | | |
| | Diffusion | | |
| | Loss of Containment | | |
| | Loss of Containment | Pipe failure | |
| | Loss of Containment | Failure of sealing | |
| | Inventory | Premixed Gas Cloud | |
| | Inventory | Liquid Pool Size | |
| | LH2 Pool Evaporation | | |
| | Release into Secondary Vessel (Refuelling) | | |
| | Release into Free Environment | | |
| | Jet Release | | |
| | Jet Release | Free Jet | |
| | Jet Release | Buoyant Jet | |
| | Jet Release | Wall Attached Jet | |
| | Jet Release | Impinging Jet | |
| | Release into Unventilated Box | Natural Ventilation | |
| | Release into Unventilated Box | Forced Ventilation | |
| **Ignition /** | | | |

# Phenomena Addressed

| | | |
|---|---|---|
| **Ignition / Combustion** | | |
| | Ignition | Minimum Ignition Energy |
| | Ignition | Ignition Location |
| | Ignition | Ignition Time |
| | Ignition | Ignition Probability |
| | Flammability Limits | |
| | Flammability Limits | |
| | Laminar Flame Speed | |
| | Diffusion/Jet Flame | |
| | Diffusion/Jet Flame | |
| | Diffusion/Jet Flame | |
| | Diffusion/Jet Flame | |
| | Flame Acceleration | |
| | Flame Acceleration | |
| | Flame Acceleration | |
| | Flame Acceleration | |
| | DDT | |
| | DDT | Detonation Cell Size |
| | | |
| | | |
| **Structural Response / Damage** | | |
| | Thermal Loads | Human Limits |
| | Thermal Loads | Structural Limits |
| | Pressure Loads | Human Limits |
| | Pressure Loads | Structural Limits |
| | | |
| | | |

# H2FC-Sage-Framework (H2FC-Sage-FX)

H2FC – Sage Framework

- Free open source mathematics software framework (based on Python)
- We will use it as a kind of "open innovation": You only have to register
- User-definable models without translation into Java code by integration SAGE math server models
- You can examine existing experiments and also create and share models by yourself on our website: http://sage.h2fc.eu/pub/

# Sage – Further information

- Further information
  - Tutorials

    http://www.sagemath.org/doc/tutorial/

    http://modular.math.washington.edu/msri06/refs/sage_tutorial.pdf

    http://www.mathematik.uni-marburg.de/~weich/Analysis2/Blaetter/Sage_Tutorial.pdf

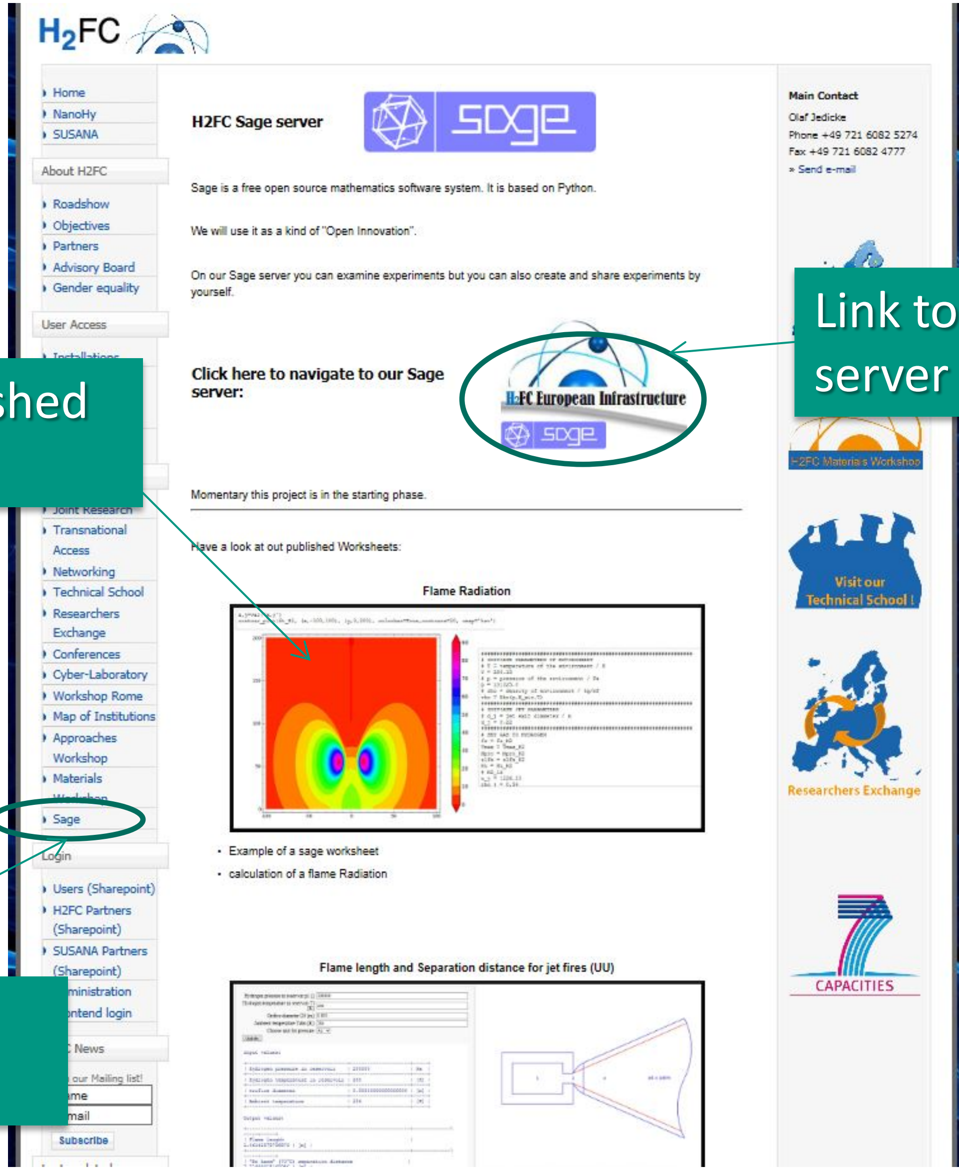
  - Link to other Sage Server Cloud solutions (not specific for hydrogen and fuel cells)

    https://cloud.sagemath.com/

  - See other examples

    http://wiki.sagemath.org/interact/

# H2FC-Sage-Framework

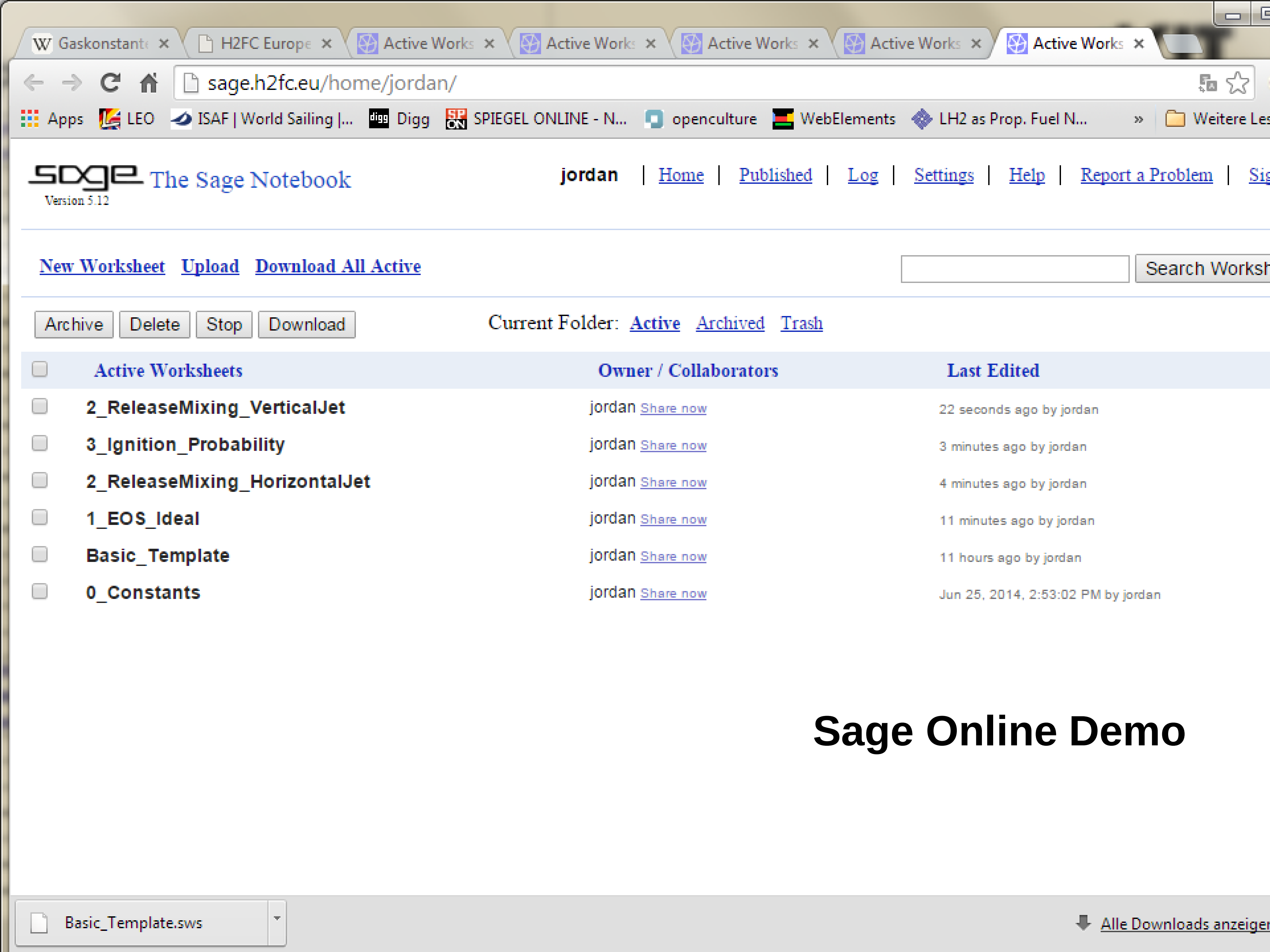



Link to the H2FC Sage server

Link to our published worksheets

Access on H2FC website

# Sage Online Demo



sage.h2fc.eu/home/jordan/5/

## Rho_Ideal : density / (g/m^3); P_Ideal : pressure / Pa; M_Ideal : mass / mol / g; T_Ideal : temperature / K

### Event, Phenomenon

This model implements the ideal gas equation of state (and serves as a template for further model definitions).

### Physics

Simplest idealised state of a gas. Related definitions are real gas equation of state like Van-der-Wals oder Nobel-Abel.

### Simplifications

Gas particles do not provide own volume and do not have interacting, attractive or disattractive forces. This implies that there is no critical state nor Joule-Thompson effect.

### Implementation

**Uses**

- Constants

**Input**

- lp : pressure / Pa
- lM : mass / mol / g
- lT : temperature / K
- lRho : density / (g/m^3)

**Output (see header)**

**Formula:**

```
Rho_Ideal(lP, lM, lT) = lP * lM / ( R * lT )
T_Ideal(lP, lM, lRho) = lP * lM / ( R * lRho )
P_Ideal(lT, lRho, lM) = R * lT * lRho / lM
M_Ideal(lT, lRho, lP) = R * lT * lRho / lP
```

### Limitations

Input variables have to be greater than 0

### Reference

*Kautz, Christian H., et al. "Student understanding of the ideal gas law, Part I: A macroscopic perspective." American Journal of Physics 73.11 (2005): 1055-1063.*

### Tests:

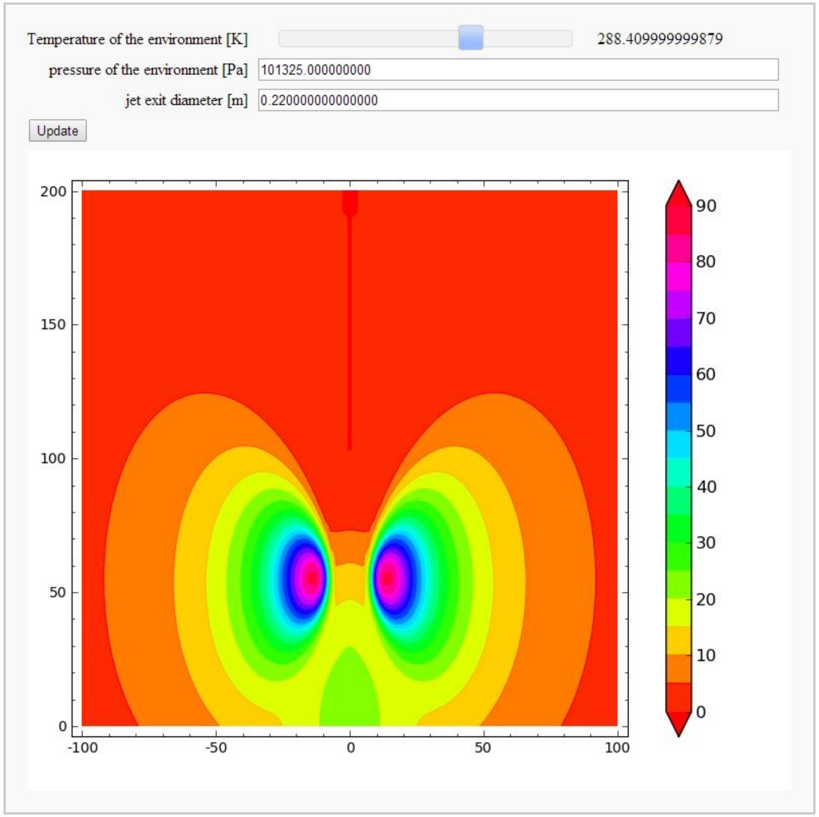
```
print T_Ideal(101325,2.01588,89.944)
    273.133476728015
print P_Ideal(293,(14*89), 28)
    108408.114090850
print Rho_Ideal(101325,2.01588,273.15)
    89.9385591463465
```

# Templated Content Implementation in Sage
# Simple Example: Ideal Gas EOS

# Complex Content Implementation in Sage
# Example: Flame Radiation (Houf, Hankinson,…)

| | |
|---|---|
| **Ignition / Combustion** | |
| | Ignition |
| | Ignition |
| | Ignition |
| | Ignition |
| | Flammability Limits |
| | Flammability Limits |
| | Laminar Flame Speed |
| | Diffusion/Jet Flame |
| | Diffusion/Jet Flame |
| | Diffusion/Jet Flame |
| | Diffusion/Jet Flame |
| | Flame Acceleration |
| | Flame Acceleration |
| | Flame Acceleration |
| | Flame Acceleration |
| | DDT |
| | DDT |
| | |
| | |
| **Structural Response / Damage** | |
| | Thermal Loads |
| | Thermal Loads |
| | Pressure Loads |
| | Pressure Loads |
| | |
| | |

**Radiative loads from standing flame**



# FlameRadiation – An interactive worksheet Example for a „Super-Model“

```
#
# STEP 1: CALCULATE DENSITY OF AIR AND FLAME FROUDE NUMBER
#

  Fr_f = (u_j * fs**1.5) / (rho_j / rho)**0.25 / ((Tmax/T - 1.0) * g * d_j)**0.5

#
# STEP 2: DETERMINE DIMENSIONLESS (SCALED) VISISBLE LENGTH  L_star/1
#
  if ( Fr_f < 5 ) L_star = 13.5 * Fr_f**0.4 / ( 1.0 + 0.07 * Fr_f**2)**0.2; else L_star = 23

#
# STEP 3: DETERMINE VISIBLE FLAME LENGTH  L_vis/m
#
  L_vis_H2 = L_star * d_j * sqrt(rho_j / rho) / fs

#
# STEP 4: DETERMINE VISIBLE FLAME WIDTH  W_f/m
#
  W_f_H2 = 0.17 * L_vis_H2

#
# STEP 5: DETERMINE RESIDENCE TIME t_f/s
#
  rho_f = Rho(p, Mpro, Tmax)
  t_f = rho_f * W_f_H2**2 * L_vis_H2 * fs / ( 3.0 * rho_j * d_j**2 * u_j )

#
# STEP 6: DETERMINE RADIANT FRACTION (ESCAPING THE FLAME)
#
  X_r_H2 = X_r(t_f, alfa_H2, Tmax_H2)
  S_rad_H2 = X_r_H2 * mfuel * Hu
```

Ideal Gas EOS

Froude

L*

L_vis

W_f

Rho_f

S_rad

# FlameRadiation – An interactive worksheet
# Online demo….

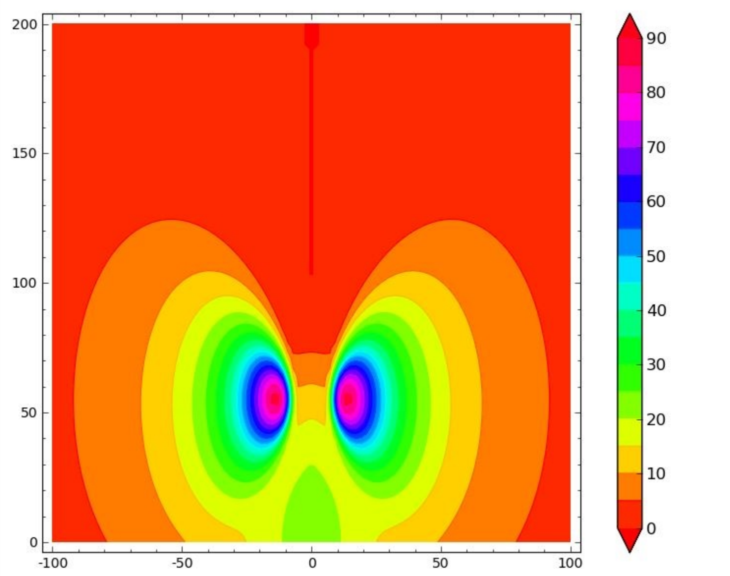
The calculation can last for a few seconds

Temperature of the environment [K] 288.409999999879

pressure of the environment [Pa] 101325.000000000

jet exit diameter [m] 0.220000000000000

Update




```
    rho_f = Rho(p, Mpro, Tmax)
    #print "rho_f = ", rho_f, " kg/m3"
    t_f = rho_f * W_f**2 * L_vis * fs / ( 3.0 * rho_j * d_j**2 * u_j)
    #print "t_f = ", t_f, " s"

    X_r = X_r(t_f,alfa_NG,Tmax_NG)
    #print "X_r = ", X_r

    S_rad = X_r * mfuel * Hu
    #print "S_rad = ", S_rad, " kW"

    def fr_H2(x,y) :
        return fr(x,y,L_vis,S_rad,Tmax_H2)
    def fr_H2_y2m(x) :
        return fr_H2(x,2.0)
    def fz_H2(x,y) :
        return fz(x,y,L_vis,S_rad,Tmax_H2)
    def fz_H2_y2m(x) :
        return fz_H2(x,2.0)
    def ft_H2(x,y):
        return sqrt((fr_H2(x,y))**2 + (fz_H2(x,y))**2)
    def ft_H2_y2m(x) :
        return sqrt((fr_H2_y2m(x))**2 + (fz_H2_y2m(x))**2)


    x,y=var('x,y')
    plot1=contour_plot(ft_H2, (x,-100,100),
(y,0,200),colorbar=True,contours=20,cmap='hsv')
    show(plot1)

print "The calculation can last for a few seconds"
@interact
def enter(out1=slider(80,400,0.01,288.15,label="Temperature of the environment
[K]"),out2=input_box(101325.0,label="pressure of the environment
[Pa]"),out3=input_box(0.22,label="jet exit diameter [m]"),auto_update=false):
    calc(out1,out2,out3)
```

evaluate

Changeable parameters

Figure of the calculation

Evaluate Sage code

# Comparison of H2FC Sage FX vs H2FC Model FX

| Category | H2FC Sage FX | H2FC Model FX |
|---|---|---|
| Structure | Open source mathematics software system (free MATLAB alternative)<br>- Expandable<br>- accident sensitiv (some functions process slow or don't seem to work reliable) | - Play! framework web application<br>- Proprietary solution |
| Interface | - Same interface for administration and user<br>- Easy scripting (PYTHON)<br>- Good collaboration platform<br>- Limited possibilities for „style“ of text and pictures in normal Sage worksheets<br>- Great graphics capabilities included | - User-friendly<br><br>- User can't create their own models |
| Availability | - No need for installation on your computer, only internet connection | - No need for installation on your computer, only internet connection |

# Current Status in a typical procedure

1. Definition of roles and development team (possibly in the new IEA HIA safety task)
2. Common definition of the system – Requirements
3. Use cases
4. Detailed Specification
5. Selection of **one** implementation framework
6. Implementation of initial content
7. Test
8. Release

prematurely without having addressed points 1. to 4. seriously

# Current Status

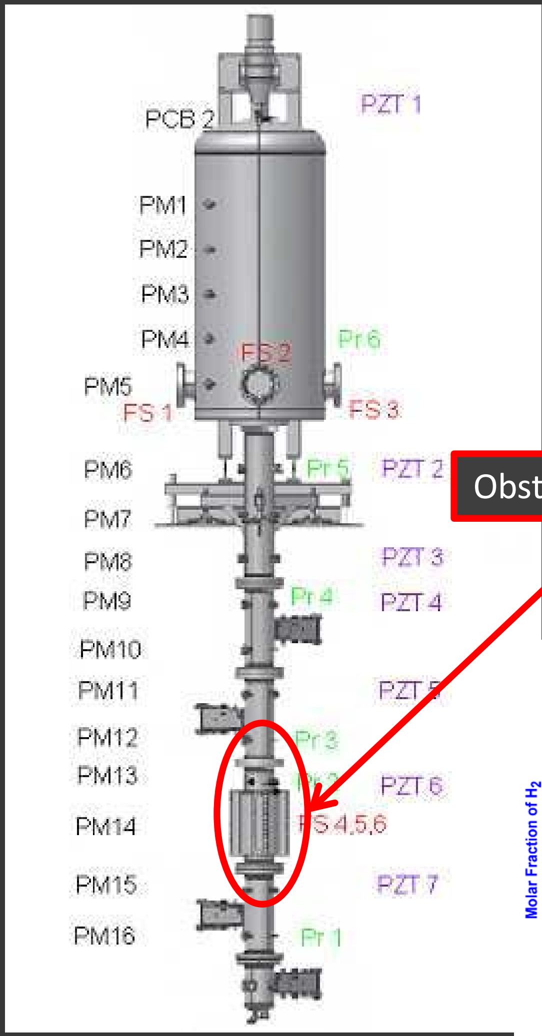
- At least 4 different implementations with different characteristics and maturity with 3 different drivers on the way:
  - Canadian activity (Benard) based on Seaside
  - US DoE activity (Groth) Sandia QRA Toolkit based on C#
  - EC H2FC activity (diverse) based (partially) on SAGE

⇨

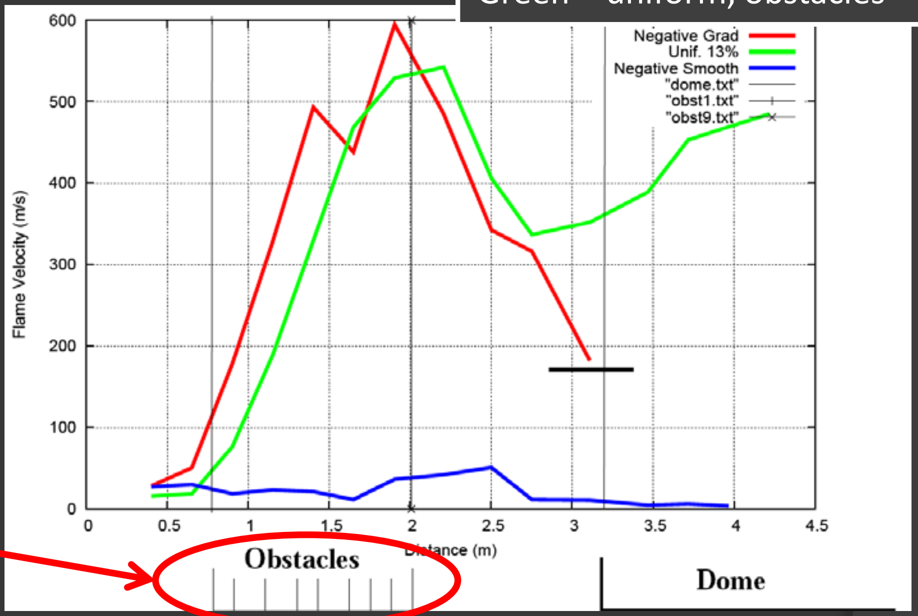
- **Threat of Fragmentation**
- **Too small user community and international character require agreement on common unified approach**

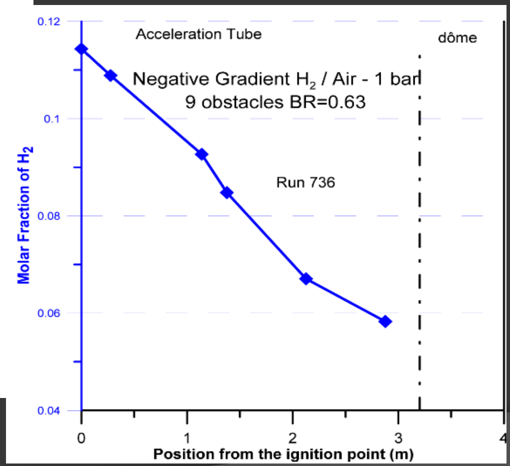
# Quenching after FA

Red – gradient, obstacles
Blue – gradient, no obstacles
Green – uniform, obstacles




Obstructed section




**Challenge for CFD:**

- **Ostacles influence**
- **Low concentrations**
- **Non-uniform mixture**

# Other demands

- Combustion of cryogenic mixtures
  - CFD: EOS for cryogenic substances
  - CFD: Multi-phase modeling (liquid and solid)
  - Laminar flame speed and other properties at $T < 100$ K
- Dust combustion
  - Turbulence -> particles / Particles –> turbulence interaction
  - Phenomenological models for such system
- Combustion with open end
  - In long open channels (tunnel) hydrodynamic resistance can hinder exhaust of products and promote FA
- Curvilinear combustion
  - Possibility of the flame acceleration /deceleration connected with the flame surface change due to combustion channel curvature

# Theoretical / CFD KG

- For premixed cases approach which uses St(u´, …) appears to be most effective and provides generally good quality, including non-uniform mixtures and vented deflagrations, however specific modeling requires for accounting of flame instabilities
- Phenomenological models for the accounting of the specifics of the congestion are highly needed

# $H_2$ Safety Research Needs

Article "How safe is Hydrogen?" by J. Hord: pp 615 Symposium Papers of the "Hydrogen for Energy Distribution„, Lyon, France, July 24-28, **1978!**

- Experimentally verify detonation in open air detonable clouds. (Evaluate strong initiator and the possibility of transition from deflagration to detonation in the absence of turbulence inducers).
- Confinement: (What constitutes sufficient confinement to sustain a detonation or higher order explosion?). Determine the effects of weak walls, elastic curtains, etc. on the transition to detionation, relief of deflagrations, etc.
- Model and study the effects of piping complex and turbulence-inducing appurtenances, for example, subdivisions, trees, buildings, etc. on transition to detonation