



Developing a PMC for membrane based sub-sea gas separation

Olaf T. Berglihn, Magne Lysberg, Inna Kim
SINTEF Industry
Trondheim, Norway

co LaN

CAPE-OPEN

2023 Annual Meeting

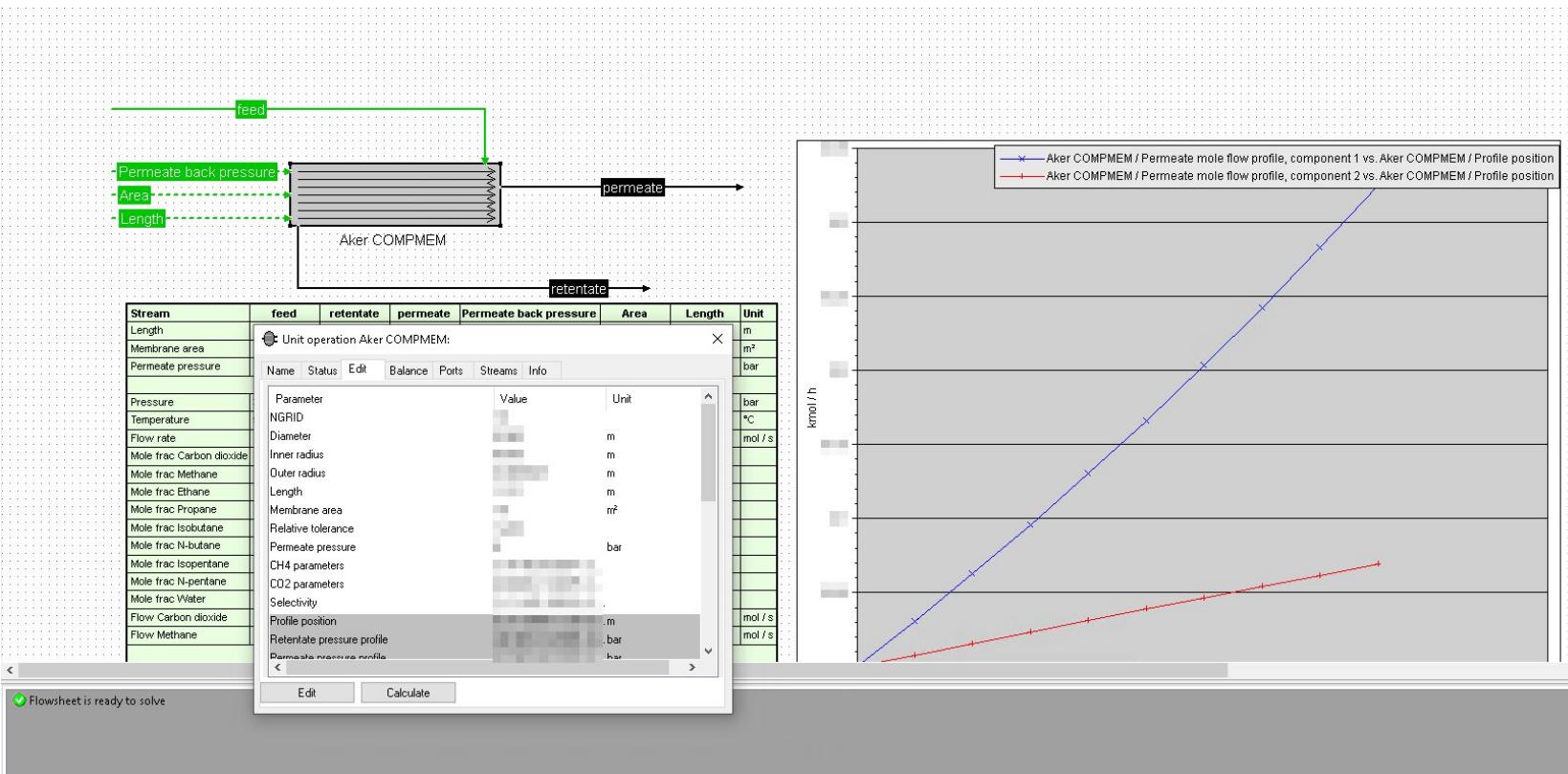
October 11 - 12 | Nancy, France



SINTEF

Outline

- The COMPMEM project
- Development of a COBIA PMC for COMPMEM partners
- Derivatives wrt. mole numbers - is there some confusion here?



COMP MEM

Qualifying membranes for sub-sea gas separation

CLIMIT Demo project 617322,
May 2019 – September 2022

- Micro-scale (lab scale) gas permeation test
- Material compatibility test
- Pilot scale test with real natural gas
- Confirm and compare the membrane performance
- Establish heat mass transport correlations/models

Budget – 3.5 M€

Project link <https://www.sintef.no/en/projects/2019/compmem-subsea-gas-separation-using-membranes/>

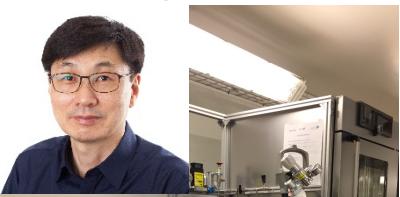


COMPMEEM team

Taek Joong Kim (WP1 lead)



Kai Hjarbo



Extra-scope



Thijs Petters
(tests with H₂S)



Magne Lysberg
(WP3 lead:)



Olaf Trygve Berglihn

Membrane performance testing

Materials compatibility testing

Modelling

Pilot design and testing

Administration

Inna Kim, Project Manager
Karl Anders Hoff, QA & SC member
Kirsti Blomsøy, coordinator



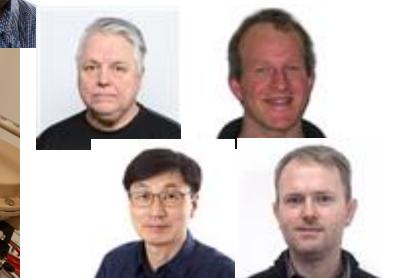
Rune Gaarder (WP2 lead),



Torstein Lange, Ann-Karin Kvernbråten (HC exposure)



Are Lund (WP4 lead),
Tor Erling Unander
(test rig design, building)
Actor Chikukwa (HAZOP)

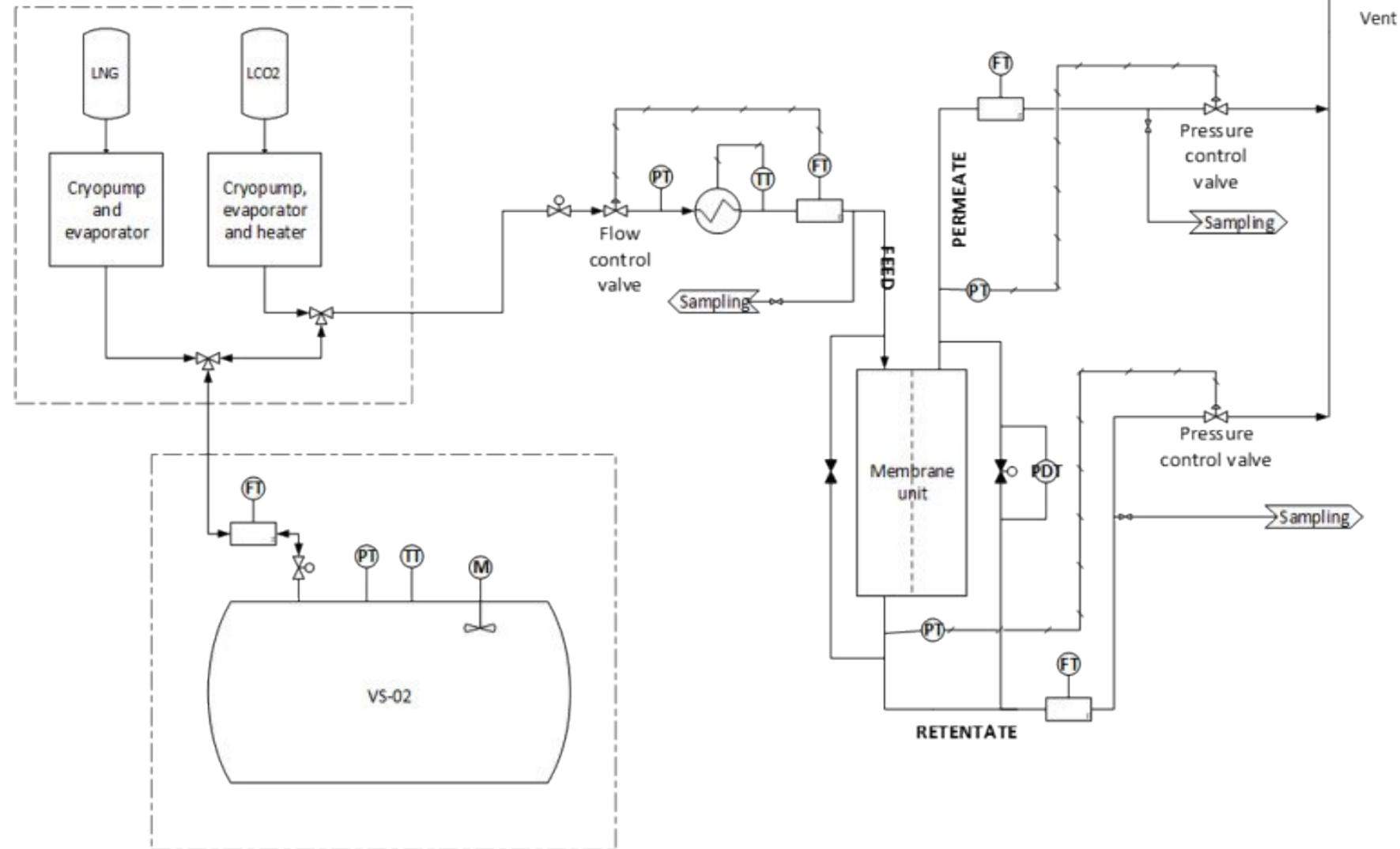


Taek Joong Kim, Terje Øyangen, Arne Erik Rekkebo, Tor Erling Unander (operation)

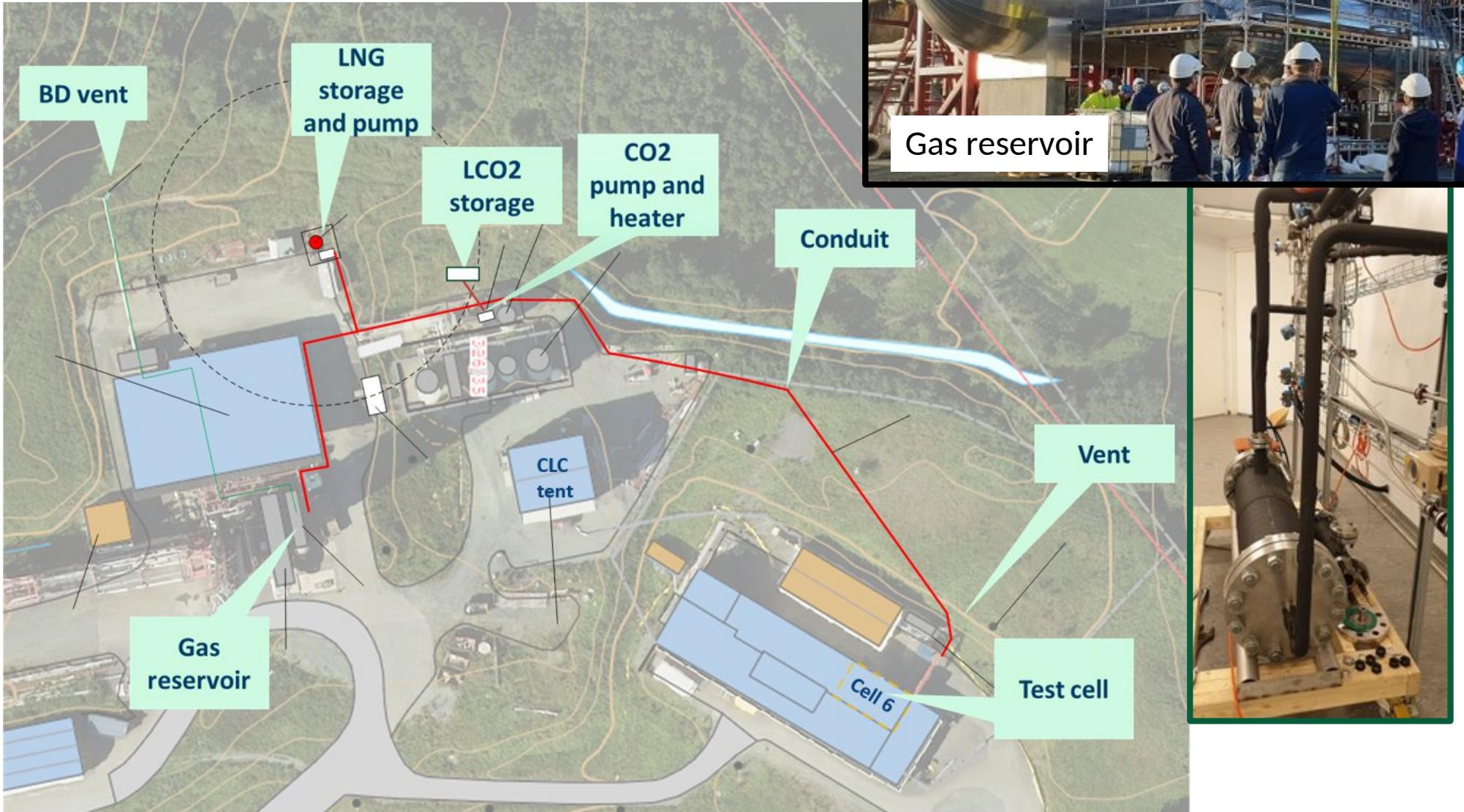


SINTEF

The screening pilot



The pilot - overview

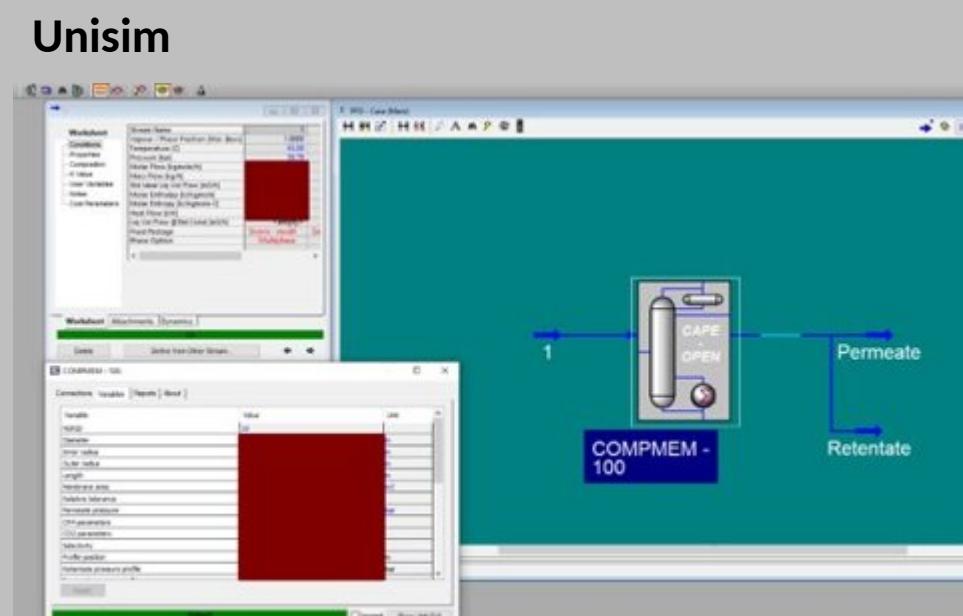


Model development and interface - first attempt

FORTRAN
BVP-problem
Collocation method



UNISIM plug-in
Native UNISIM C++ API



- Licences
- Unfamiliar API
- A lot of boiler-plate code for structures
- GUI?

Model development and interface - refined

FORTRAN
BVP-problem
Collocation method



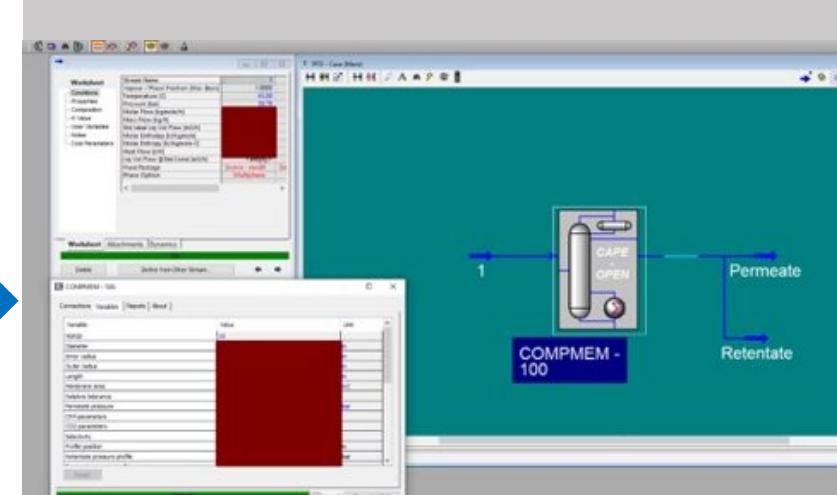
C++ PMC using COBIA



COBIA

CAPE-OPEN

Unisim



- Enabled testing with open/alternative tools
- Standardized API
- Use GUI of choice
- Testing in Linux - or any POSIX platform



SINTEF

On properties

7.5.5 Non-constant single-phase mixture properties

See section 7.5.7 for more information on entries in this table.

Identifier	Meaning	type of property	dimensionality	units	basis	overall
enthalpy	Enthalpy (may or may not include the enthalpy of formation)	E		J	mole/mass	Y

CAPE-OPEN Thermo v.1.1 facimilie

```
propList.resize(2);
propList[0] = COBIATEXT("enthalpy.Dmoles");
propList[1] = COBIATEXT("enthalpy");
```

```
stream.GetSinglePhaseProp(ConstCapeString(COBIATEXT("enthalpy")),
phaseName, ConstCapeString(COBIATEXT("mole")), propValues);
```

propValues <- molar enthalpy [J/mol]

enthalpy.Dmoles

property.Dmoles

derivatives of property with respect to mole number keeping pressure and temperature and other mole numbers fixed for a mixture containing a total of one mole of material. For some property H the ith element of derivative is

$$H.Dmoles_i = \bar{h}_i = \left(\frac{\partial H}{\partial n_i} \right)_{p,T,n_{j \neq i}}$$

For a two-phase property the mole number derivatives are evaluated independently for each phase by keeping the temperature and pressure of both phases and the mole numbers in the other phase fixed

[property]/mol

enthalpy.Dmoles

```
.GetSinglePhaseProp(ConstCapeString(COBIATEXT("enthalpy")),
phaseName, ConstCapeString(COBIATEXT("mole")), propValues);
```

$$\text{enthalpy: } \frac{H}{\sum n_i} \text{ [J/mol]}$$

```
.GetSinglePhaseProp(ConstCapeString(COBIATEXT("enthalpy.Dmoles")),
phaseName, ConstCapeString(COBIATEXT("mole")),
CapeArrayReal(partialMolarEnthalpies));
```

$$\text{enthalpy.Dmoles: } \left(\frac{\partial H}{\partial n_i} \right)_{T,p,n_j} \text{ [J/mol]}$$



SINTEF

<property>.Dmoles – fails in some simulator

property	type	d[]/dn[ETHANE]	d[]/dn[ETHANE](perturbed)	d[]/dn[PROPANE]	d[]/dn[PROPANE](perturbed)
fugacityCoefficient(Vapor)	intensive	-0.0231774	-0.0231727	-0.0563937	-0.0563758
logFugacityCoefficient(Vapor)	intensive	-0.0264349	-0.0264295	-0.0643195	-0.0642993
fugacity(Vapor)	intensive	-973298	-973195	-1.00922e+06	-1.0091e+06
enthalpy(Vapor)	extensive	0.0901916	-84196.7	0.0688533	-105533
enthalpyF(Vapor)	extensive	0.0901916	-84196.7	0.0688533	-105533
enthalpyNF(Vapor)	extensive	-0.000489496	-367.622	-0.000967767	-845.965
entropy(Vapor)	extensive	-9.6106e-05	-182.667	-0.000171951	-258.552
entropyF(Vapor)	extensive	-9.6106e-05	-182.667	-0.000171951	-258.552
entropyNF(Vapor)	extensive	1.17227e-05	-8.60455	3.10985e-05	10.7221
gibbsEnergy(Vapor)	extensive	0.121248	-25167.8	0.124419	-21982
volume(Vapor)	extensive	-8.00437e-11	0.00062498	-1.70181e-10	0.00053484
viscosity(Vapor)	intensive	-3.85138e-06	-3.85084e-06	-5.84622e-06	-5.84515e-06
thermalConductivity(Vapor)	intensive	-0.00925663	-0.00925505	-0.0230221	-0.0230175

Incorrectly interpreted as $\left(\frac{\partial(H/\sum n_i)}{\partial n_i} \right)_{T,p,n_{j \neq i}}$?

Resolution: check if Euler homogeneity, $H = \sum \left(\frac{\partial H}{\partial n_i} \right)_{T,p,n_{j \neq i}} n_i$

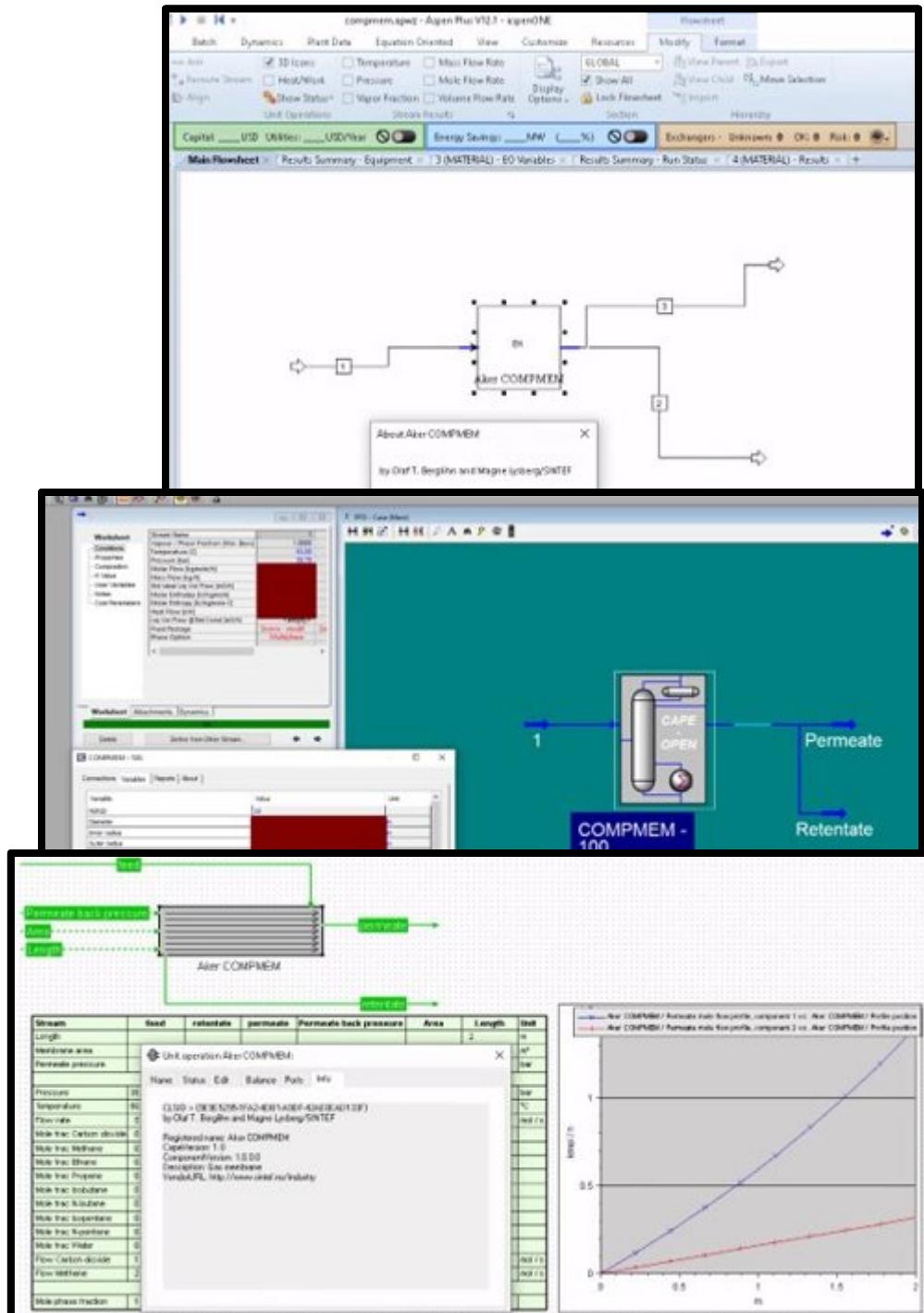
Not OK? Perturb H wrt n_i through manipulating (all) x_i



SINTEF

Overall results

- Shipped PMC to partner based on testing in alternative simulator - it worked flawlessly
- Array-parameters are not supported by some major vendors - crashes or fails to calculate
- Some vendors implement derivatives of extensive quantities, “.Dmoles”, incorrectly



Acknowledgement

Many thanks to **Jasper van Baaten and Amsterchem** for technical support and guidance.

The COMPMEM project has been performed under the Norwegian CLIMIT Demo program (project number 617322).

The authors acknowledge the following partners for their contributions: Aker Carbon Capture (project coordinator), Equinor, Pertamina, Carbon Capture project (CCP) consortia, and the Research Council of Norway.



SINTEF

Technology for a
better society

The pilot

