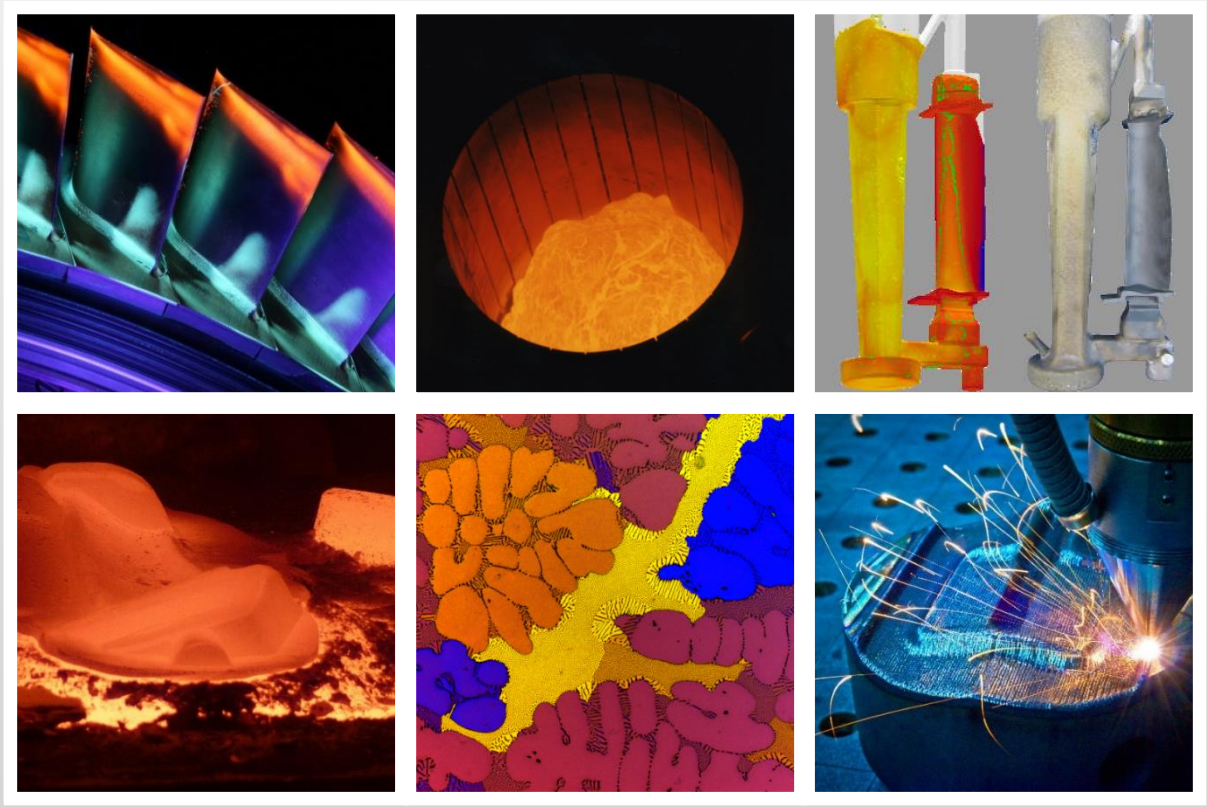


Tutorial
Importing Initial Microstructures

Outline

- Import of initial microstructures from
  - ASCII text files (2D only)
  - VTK (2D and 3D) files
- Adding properties
  - Explicit description
  - Concentration fields (2D)
- Using MICRESS Restart files

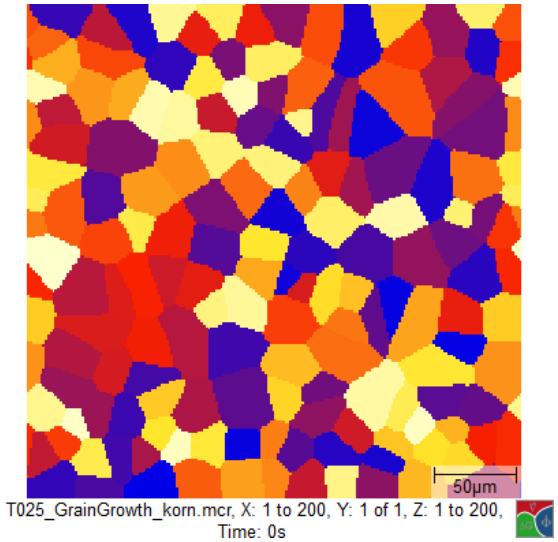
Outline

- **Import of initial microstructures from**
  - **ASCII text files (2D only)**
  - **VTK (2D and 3D) files**
- Adding properties
  - Explicit description
  - Concentration fields (2D)
- Using MICRESS Restart files

MICRESS: 2D ASCII Text Format

Format description

- ASCII text file with UTF-8 coding
- Raw data values
  - Each value as an ASCII string
  - Space separated
  - Row-wise data arrangement



Colored Grain IDs in DP_MICRESS

t025_korn_ASCII_1.txt

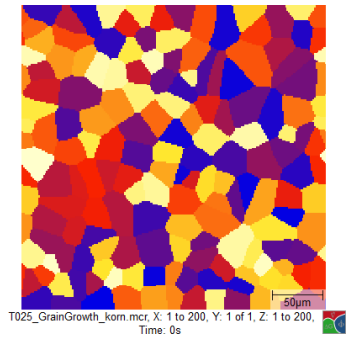
```
1  17 17 17 17 17 17 17 17 17 1
2  17 17 17 17 17 17 17 17 17 1
3  17 17 17 17 17 17 17 17 17 1
4  17 17 17 17 17 17 17 17 17 1
5  17 17 17 17 17 17 17 17 17 1
6  17 17 17 17 17 17 17 17 17 1
7  17 17 17 17 17 17 17 17 17 1
8  17 17 17 17 17 17 17 17 17 1
9  83 63 63 17 17 17 17 17 17 1
10 63 63 63 63 63 17 17 17 17 1
11 63 63 63 63 63 63 17 17 17 1
12 63 63 63 63 63 63 63 63 17 1
13 63 63 63 63 63 63 63 63 63 6
14 83 63 63 63 63 63 63 63 63 6
15 83 63 63 63 63 63 63 63 63 6
16 83 63 63 63 63 63 63 63 63 6
17 83 63 63 63 63 63 63 63 63 6
18 83 63 63 63 63 63 63 63 63 6
19 83 63 63 63 63 63 63 63 63 6
```

ASCII representation exported with DP_MICRESS (# Geometry Header removed)

MICRESS: 2D and 3D VTK Format

Format description

- VTK Header (ASCII UTF-8)
  - Options in line 3: ASCII or BINARY
  - Dimensions: Number of Nodes in each dimension
  - Origin : not used
  - Spacing in micrometers: cubic voxels !
  - Cell data: number of voxels
- Data Header (ASCII UTF-8)
  - Scalar values, field name, data type
  - Standard lookup table : not used
- Data
  - ASCII or binary
  - Order: loop over Z,Y,X
  - Multiple fields possible



Header

```
1  # vtk DataFile Version 3.0
2  t = 0 s
3  ASCII
4
5  DATASET STRUCTURED_POINTS
6  DIMENSIONS 201 2 201
7  ORIGIN 0 0 0
8  SPACING 1.5 1.5 1.5
9
10 CELL_DATA 40000
```

Data Header

```
12 SCALARS korn int
13 LOOKUP_TABLE default
```

Data

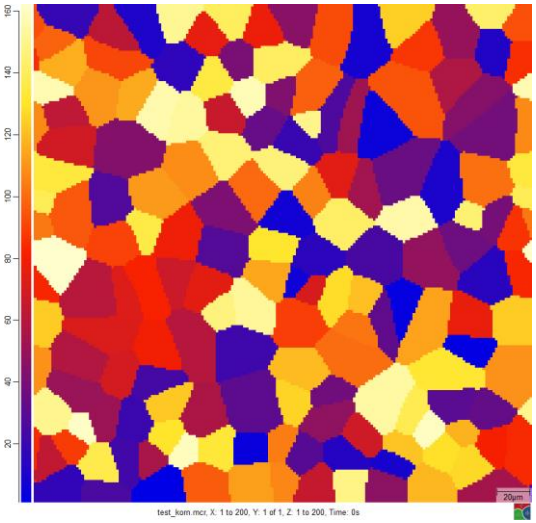
```
14  17 17 17 17
15  17 17 17 17
16  17 17 17 17
17  17 17 17 17
18  17 17 17 25
19  25 25 25 25
20  25 25 25 25
21  25 25 25 25
22  25 25 25 25
23  25 25 25 1
```

ASCII VTK representation exported with DP_MICRESS (# Geometry Header removed)

Read ASCII 2D text file

```
# Initial Microstructure
# ======================
# Type of grain positioning?
# Options: deterministic random [deterministic_infile] from_file
from_file
# Filename of initial grain/phase structure [VTK_identifier (default=korn)] ?
T025_korn_ASCII_1.txt
# Treatment of data?
# (n: none, 1: 1D, f: flip (bottom<->top), t: transpose,
# or p: 'phase to grains transformation')
n
# CellsX for initial microstructure?
200
# CellsZ for initial microstructure?
200
# Number of grains at the beginning?
# (Set to less than 1 for the number of grain to be read from the input data,
# with optionally a minimal size, in cells)
-1
```

- **Data treatment**
  none: Grain IDs are read
- **Geometry** like in the ASCII file
  to read correct number of values
  in X and Z direction
- **Number of grains at the beginning**:
  automatic recognition (-1)



Read ASCII 2D text file: options

Data treatment

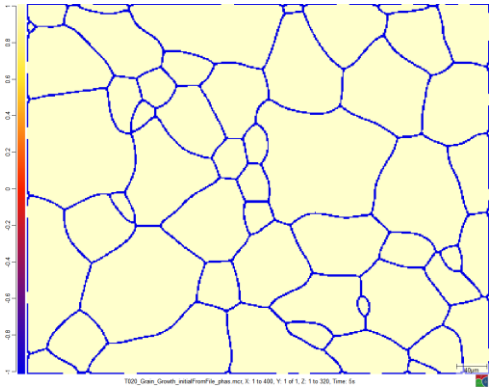
- 1D : read 1D data for a pure 1D simulation
- Flip : make bottom to top (Z dimension)
- Transpose
  - transposes x- and z- columns/rows of the picture.
    This operation is meaningful only for square-pictures or
    if columns and rows are switched by other operations before
- Phase to grain (see next slide)
  - assigns an individual grain number to each closed region
  - Grain interfaces are marked with a minimal value (typically -1)
    See example T020_Grain_Growth_initialFromFile_dri and input file T020_Microstructure.txt

Number of grains at the beginning

- Only possible value: automatic (-1)
  - Deprecated option: if not -1, number of grains have to match highest grain number
- Minimal grain size (in cells)
  - Only if data treatment is not none
  - Smaller grains will be added to neighbours

Phase to grain transformation (Example: T020)

```
# Initial Microstructure
# ======================
# Type of grain positioning?
# Options: deterministic random [deterministic_infile] from_file
from_file
# Filename of initial grain/phase structure [VTK_identifier
      (default=korn)] ?
T020_Grain_Growth_Microstructure.txt
# Treatment of data?
# (n: none, 1: 1D, f: flip (bottom<->top), t: transpose,
# or p: 'phase to grains transformation')
fp
# CellsX for initial microstructure?
500
# CellsZ for initial microstructure?
400
# Number of grains at the beginning?
# (Set to less than 1 for the number of grain to be read …
# with optionally a minimal size, in cells)
-1
...
```



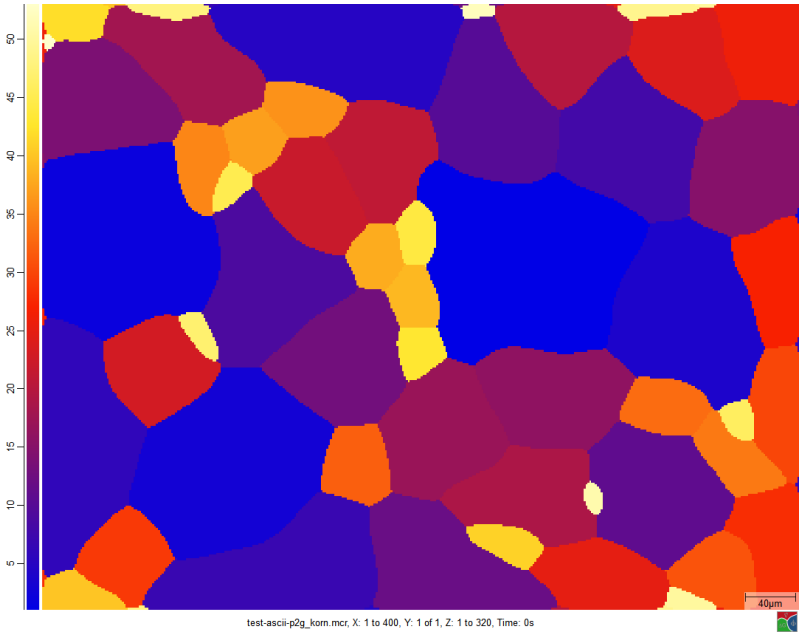
Automatic grain recognition

- By closed areas
- Lowest value denotes interface (here: 0)
- All grains 1 … 53 with same bulk value 765
  - ➢ This information can be used for block assignment of properties later (useful for multiple phases)

```
# Number of grains at the beginning?
# (Set to less than 1 for the number of grain to
#  with optionally a minimal size, in cells)
-1
# value 765: grains 1 ... 53
# Number of grains found in input data: 53
```

Read from VTK file

```
# Initial Microstructure
# ======================
# Type of grain positioning?
# Options: deterministic random [deterministic_infile] from_file
from_file
# Filename of initial grain/phase structure [VTK_identifier (default=korn)] ?
test-vtk-ascii-grainids.vtk  myGrainIds
# Treatment of data?
# (n: none, 1: 1D, x: rotate Clockwise along x-axis, y, z,
# or p: 'phase to grains transformation')
n
# Number of grains at the beginning?
# (Set to less than 1 for the number of grain to be read from the input
    data,
# with optionally a minimal size, in cells)
-1
…
```



Differences to text file input:

- New option for data treatment
  - Rotate clockwise along x-,y-, or z-axis
- Geometry is read from VTK header

Outline

- Import of initial microstructures from
  - ASCII text files (2D only)
  - VTK (2D and 3D) files
- **Adding properties**
  - **Explicit description**
  - **Concentration fields (2D)**
- Using MICRESS Restart files

Adding properties: Explicit description

Property block

- Like explicit grain description
  - Phase ID
  - Orientation
  - …

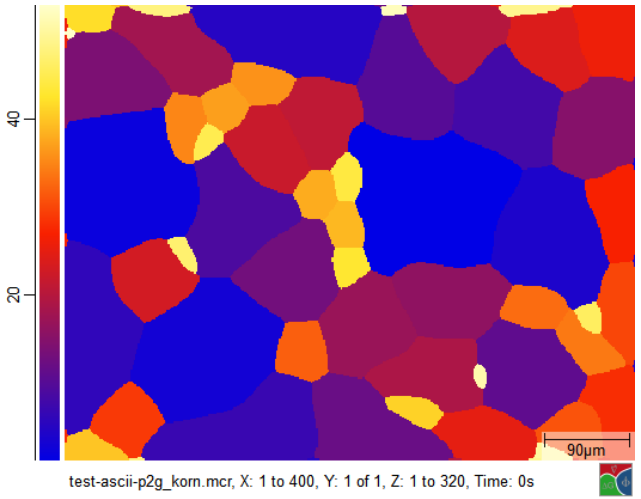
Input type options

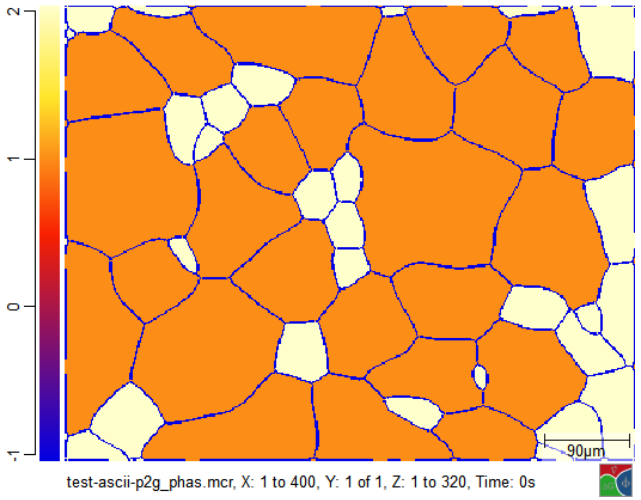
- input
  - A property block for each grain
- from_file
  - As input but deterministic description is read from a file
- identical
  - Same properties for all grains
- blocks
  - A property block for each defined grain ID range

```
# Initial Microstructure
# ======================
…
# Read grain properties from a file?
# Options: input   from_file   identical   blocks
identical
# Phase number? (integer)
1
# 'Non-geometric' data for grains -1 to 53 identical
…
```

Add properties blockwise

```
# Number of grains found in input data: 53
# Read grain properties from a file?
# Options: input from_file identical blocks
blocks
# Last grain of this 'block'?
25
# Grain number 1
# Phase number? (integer)
1
# 'Non-geometric' data for grains 1 to 25 identical
# Last grain of this 'block'?
53
# Grain number 26
# Phase number? (integer)
2
# 'Non-geometric' data for grains 26 to 53 identical
```





Initial Concentration: Reading 2D concentration files

Format

- Same as 2D ASCII files
- Local concentration values for one component of one phase in a file
- Concentration fields must exactly match to the corresponding grain structure, otherwise problems are very probable!

Remarks

- Getting experimental information on concentration distribution is not easy!
- Measured concentration distributions typically have a lot of noise and do non comply to the nominal alloy composition!

Anyway, we can read them!

```
# Initial Concentrations
# ======================
# How shall initial concentrations be set?
# Options: input equilibrium from_file [phase number]
from_file
# Concentration of component 1 (CU) in phase 0 (LIQUID) ? [at%]
    (for initialisation of TQ)
3.0000
# Filename of initial concentration profile of component 1?
    [VTK_identifier (default=conc)]
initial_copper_fcc.txt
# Treatment of data?
# (n: none, 1: 1D, f: flip (bottom<->top), or t: transpose)?
n
# CellsX for initial concentration field of component 1 ?
500
# CellsZ for initial concentration field of component 1 ?
500
# Temperature at which the initial equilibrium
# will be calculated? [K]
915.00
```

Components: AL, CU - Phases: Liquid, FCC

Outline

- Import of initial microstructures from
  - ASCII text files (2D only)
  - VTK (2D and 3D) files
- Adding properties
  - Explicit description
  - Concentration fields (2D)
- **Using MICRESS Restart files**

Using restart files

Normal usage of restart files (.rest)

- ***restart***
  - restarts simulation in the status where it was stopped
  - only a limited number of parameters can be changed

- ***restart reset_time***
  - restarts simulation, but from time 0
  - all timers are reset
  - as a consequence, more parameters can be changed!

- ***restart with_flow***
  - If flow coupling is planned and the restart was made without flow (may be obsolete in future by automatic recognition)

```
# Restart
# =======
# Restart using old results?
# Options: new restart [reset_time | with_flow]
restart
# Name of restart file (without extension)?
restart_file_name
#
```

Restart file: Reading the initial structure

Using the restart file for setting the initial structure

- only takes microstructure without any numerical conditions
- allows combination of several microstructures
- allows furthermore shifts, rotations and (simple) zoom

```
# Restart
# =======
# Restart using old results?
# Options: new restart [reset_time | with_flow]
new
```

Setting up container grains

- Define initial grain structure as usual (explicit description is recommended)

```
# Initial Microstructure
# ======================
# Type of grain positioning?
# Options: deterministic random [deterministic_infile] from_file
deterministic
# NB: the origin of coordinate system is the bottom left-hand corner,
# all points within the simulation domain having positive coordinates.
# Number of grains at the beginning?
3
…
```

Initial structure from restart file: Container Grains

…

- **Shape, position and size define the targeted region**
- **Target region will be completely replaced !**
- **Properties dummies**
  - Necessary for syntactical reason, same input routines in MICRESS
  - do not have a meaning here

```
# Grain number 1
# --------------
# Geometry?
# Options: round rectangular elliptic round_inverse
round
# Center x,z coordinates [micrometers], grain number 1?
30.000
30.000
# Grain radius? [micrometers]
1500.0
# Shall grain 1 be stabilized or shall
# an analytical curvature description be applied?
# Options: stabilisation analytical_curvature
stabilisation
# Should the Voronoi criterion be applied?
# Options: voronoi no_voronoi
no_voronoi
# Phase number? (integer)
1
…
```

Initial structure from restart file: Import

Filling the container grains

- Activate restart from file
- How many containers should be filled
- For each container
  - 1st input line
    - Container / grain ID
    - Position shift relative to RVE origin
    - Optional zoom factors (integer)
    - Optional rotation
  - 2nd input line
    - The restart file name (without extension)

```
# Structure from restart file
# -------------------------
# Shall grain(s) be replaced by initial structure(s) from a restart file(s) ?
# Options: restart_file | no_restart_file
restart_file
# How many restart files shall be read?
3
# For each restart file a grain number and (optionally)
# shift (in grid cells) and zoom factor for all 3 dimensions
# as well as a character for rotation options must be specified:
# grain number [shift X (int) shift Y (int) shift Z (int)
# zoom X (int) zoom Y (int) zoom Z (int) rot(string)] ?
# Rotation options: "xz+90" "xz-90" "xz180" "xy+90" ... "yz180"
3 10 0 65
# Name of restart file?
Results/Restart_1
…
```

Example with nice pictures in an advanced training course!

Thank you for your attention!

Short break for questions
before continuing with the next …

Challenge

- Import a Microstructure