Deutsches Zentrum für Luft- und Raumfahrt e.V.

Block Structure and Grid Logic, Part 1:

Blocks and Segments

Jochen Raddatz

Institute of Aerodynamics and Flow Technology, DLR Braunschweig

DLR

Overview:

What is a block ?

Block Faces

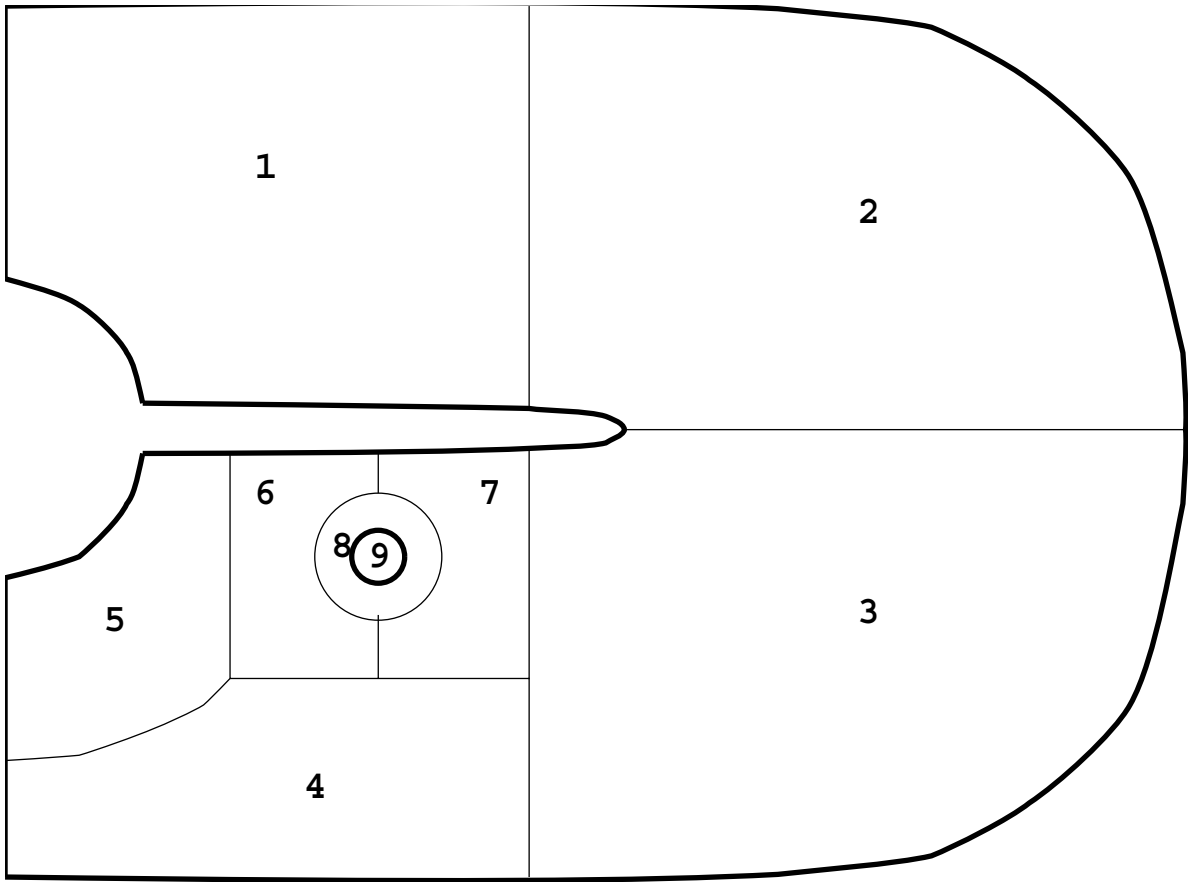
- Numbering of Block Faces
- Indexing of Variables

Segments

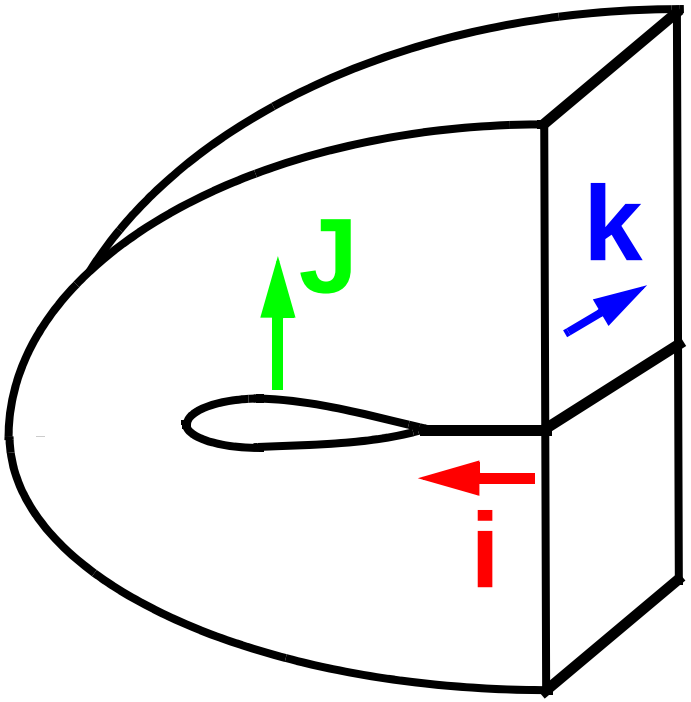
- Definition and Indexing
- L1, L2 Direction

What is a block ?

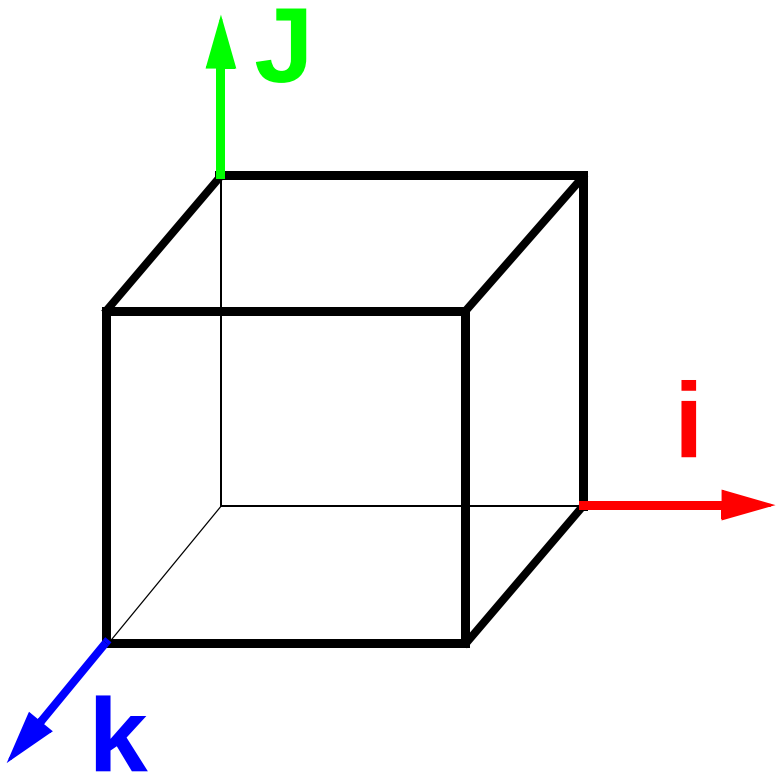
Using multiple blocks (multiblock)
allows the generation of structured grids around complex geometries, such as a generic transport aircraft.



DLR

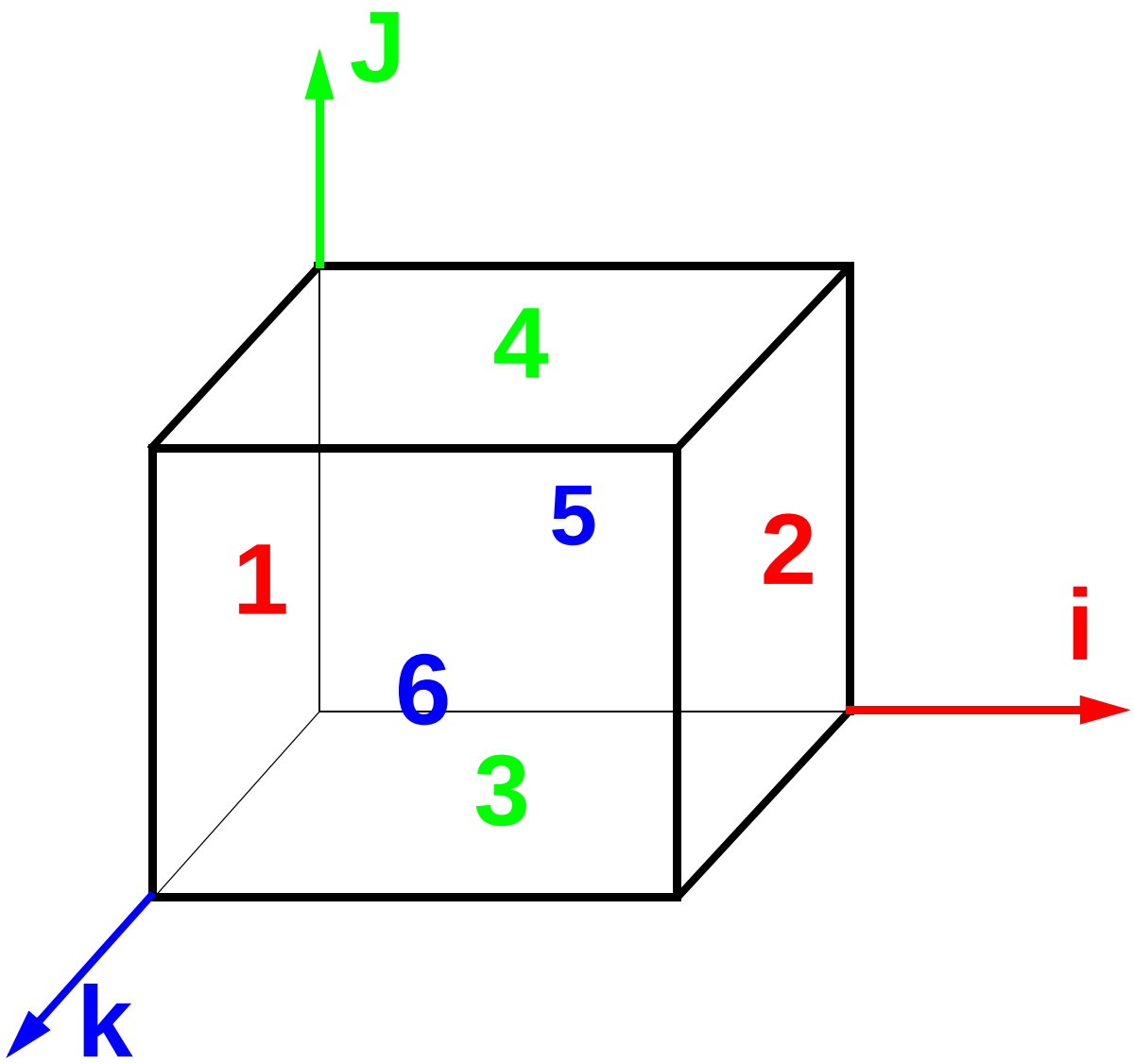


"physical" space



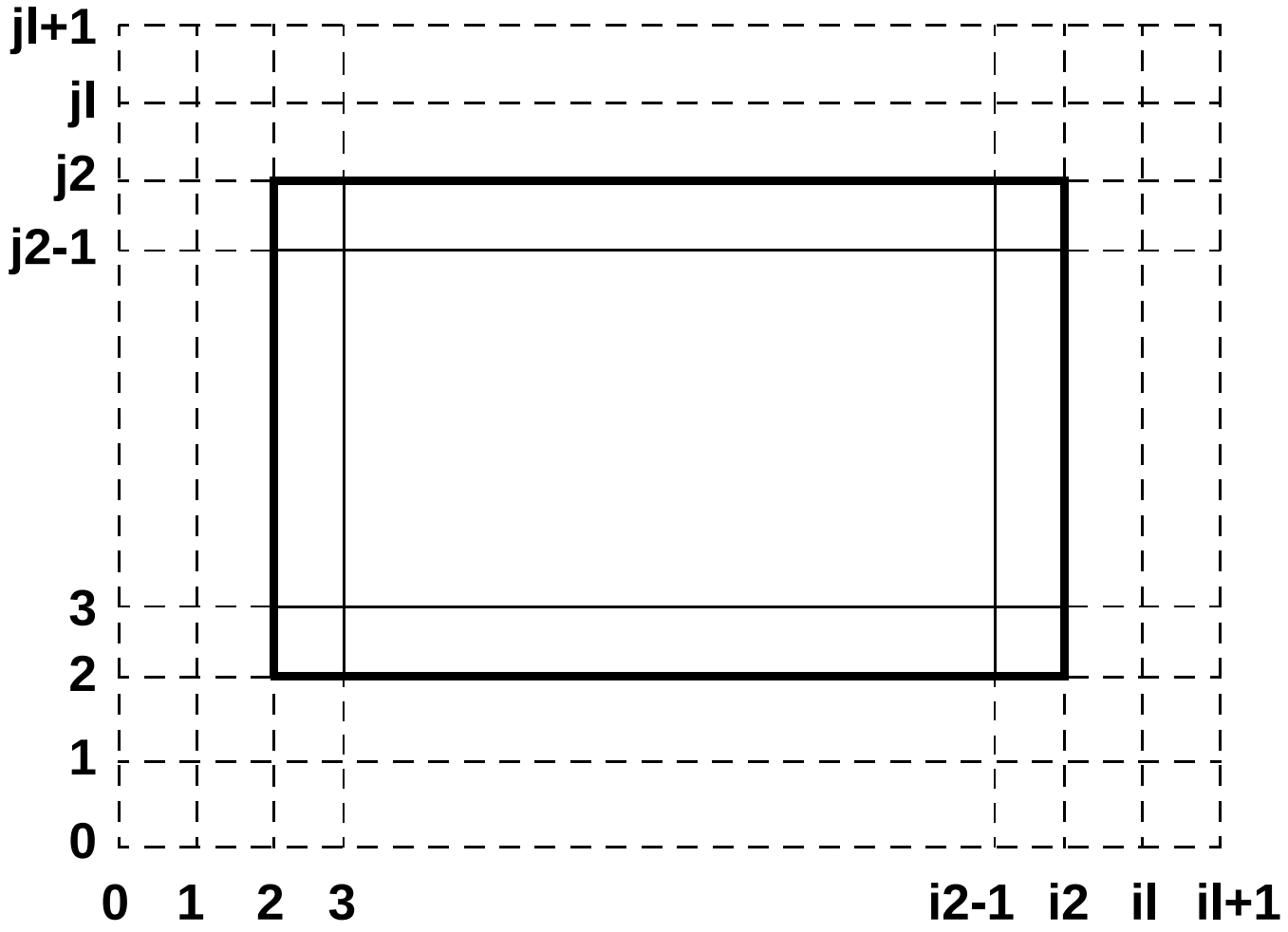
index space

Block Faces: Numbering of Faces



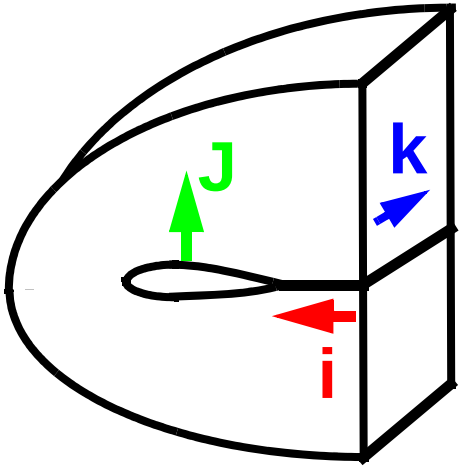
DLR

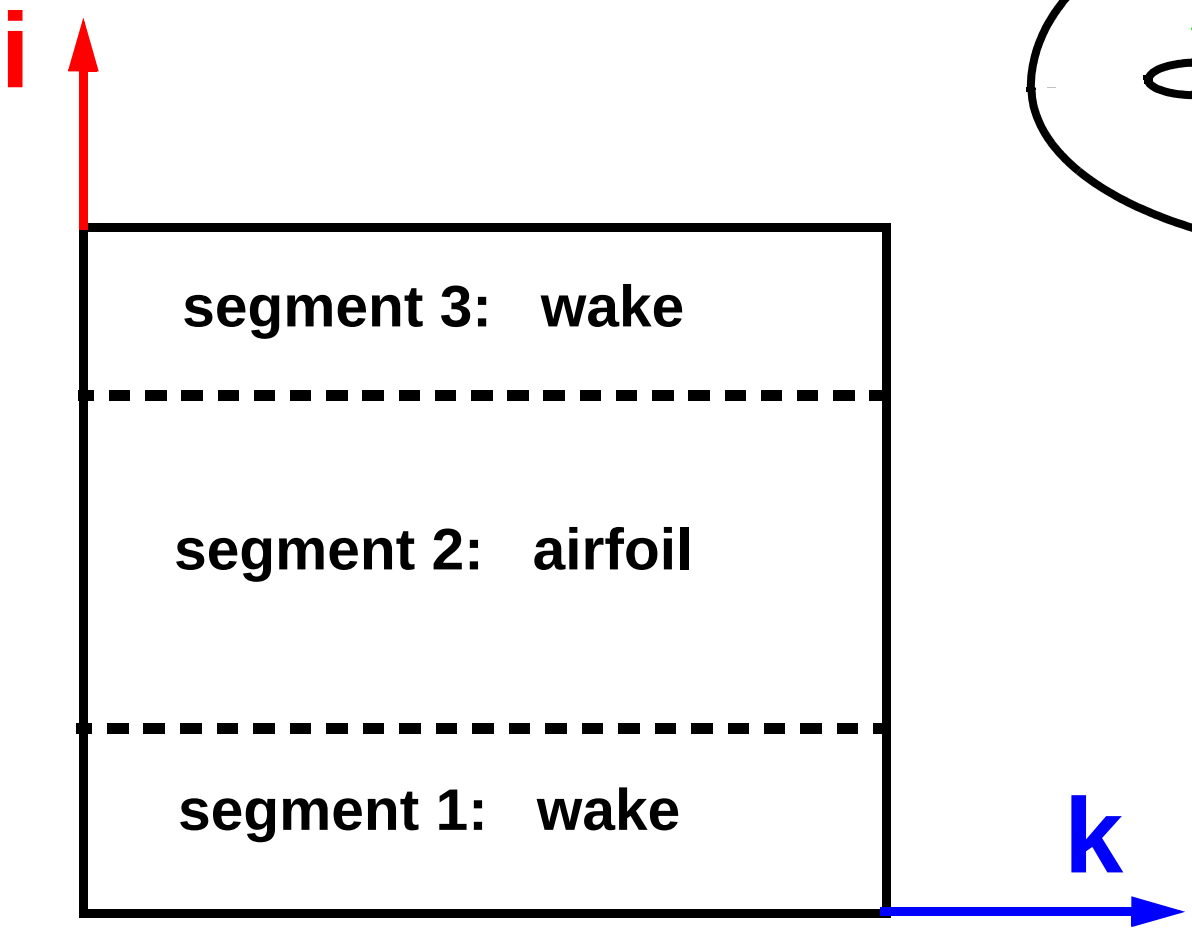
Block Faces: Indexing of grid points



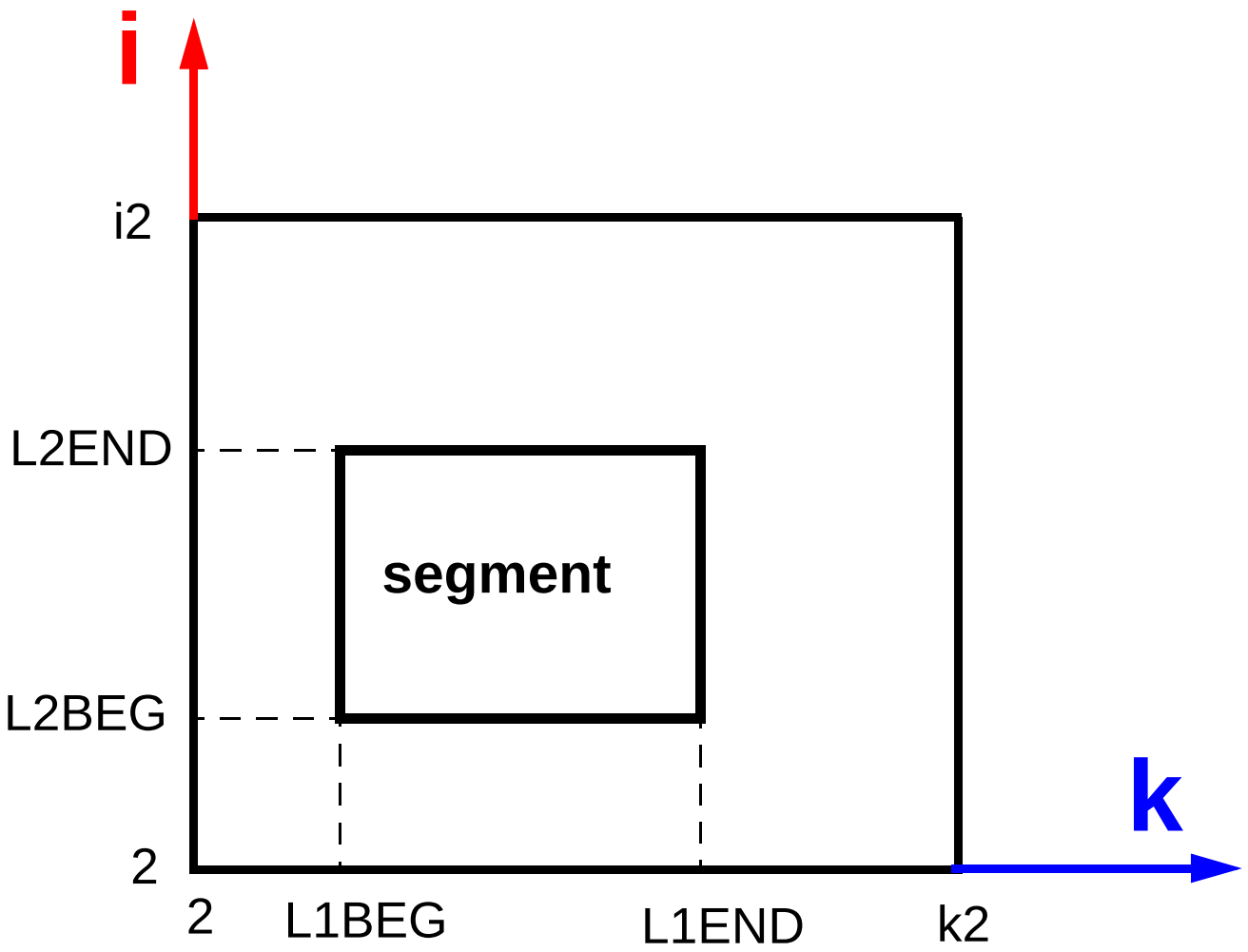
DLR

Segments: Definition





Segments: Indexing



DLR

Segments: L1, L2 Direction

In a right handed coordinate system:

On Face		L1	L2
i= const:	1, 2	j	k
j=const:	3, 4	k	i
k=const:	5, 6	i	j

Block Structure and Grid Logic, Part 2:

Blocks and Segment Boundaries

DLR

Overview:

Boundary Conditions

- "Physical" Boundary Conditions
- Cut Segments
- Singularities

Boundary Management

- Cell Vertex Scheme

Boundary Conditions: "Physical" Boundary Conditions

Solid Walls: ITYP = 1, ..., 10

ITYP = 1 Euler / Slip Wall

ITYP = 2 Navier-Stokes / No Slip Wall

Far-Field Conditions: ITYP = 11, ..., 20

ITYP = 12 Variable (Inflow / Outflow)

ITYP = 13 Free Stream (Inflow)

Boundary Conditions: "Physical" Boundary Conditions

Symmetry Conditions: ITYP = 21, ..., 30

ITYP = 21 Symmetry in X-direction

ITYP = 22 Symmetry in Y-direction

ITYP = 23 Symmetry in Z-direction

ITYP = 25 Arbitrary Symmetry plane
(not finally validated)

Boundary Conditions: "Physical" Boundary Conditions

Engine Conditions: ITYP = 31, ..., 40

ITYP = 31	Engine Inlet (only for Euler) (only usable for Euler)
ITYP = 51	Outflow, P=const. (usable for Engine Inlet)
ITYP = 32	Engine Outlet
ITYP = 33	Engine Core Outlet

Boundary Conditions: "Physical" Boundary Conditions

Internal Flow - Inflow Conditions: ITYP = 41, ..., 50

ITYP = 41 Inflow, P-Extrapolation

ITYP = 42 Inflow, U-Extrapolation

ITYP = 43 Inflow, Characteristics

ITYP = 44 Subsonic Inflow, Characteristics
(under development, only for CC)

Internal Flow - Outflow Conditions: ITYP = 51, ..., 60

ITYP = 51 Outflow, P-Extrapolation

ITYP = 54 Subsonic Outflow, Characteristics
(under development, only for CC)

Boundary Conditions: Singularities

ITYP = 101 Singular Line / Singular Point
0th order extrapolation
(treatment similar to Euler Wall)

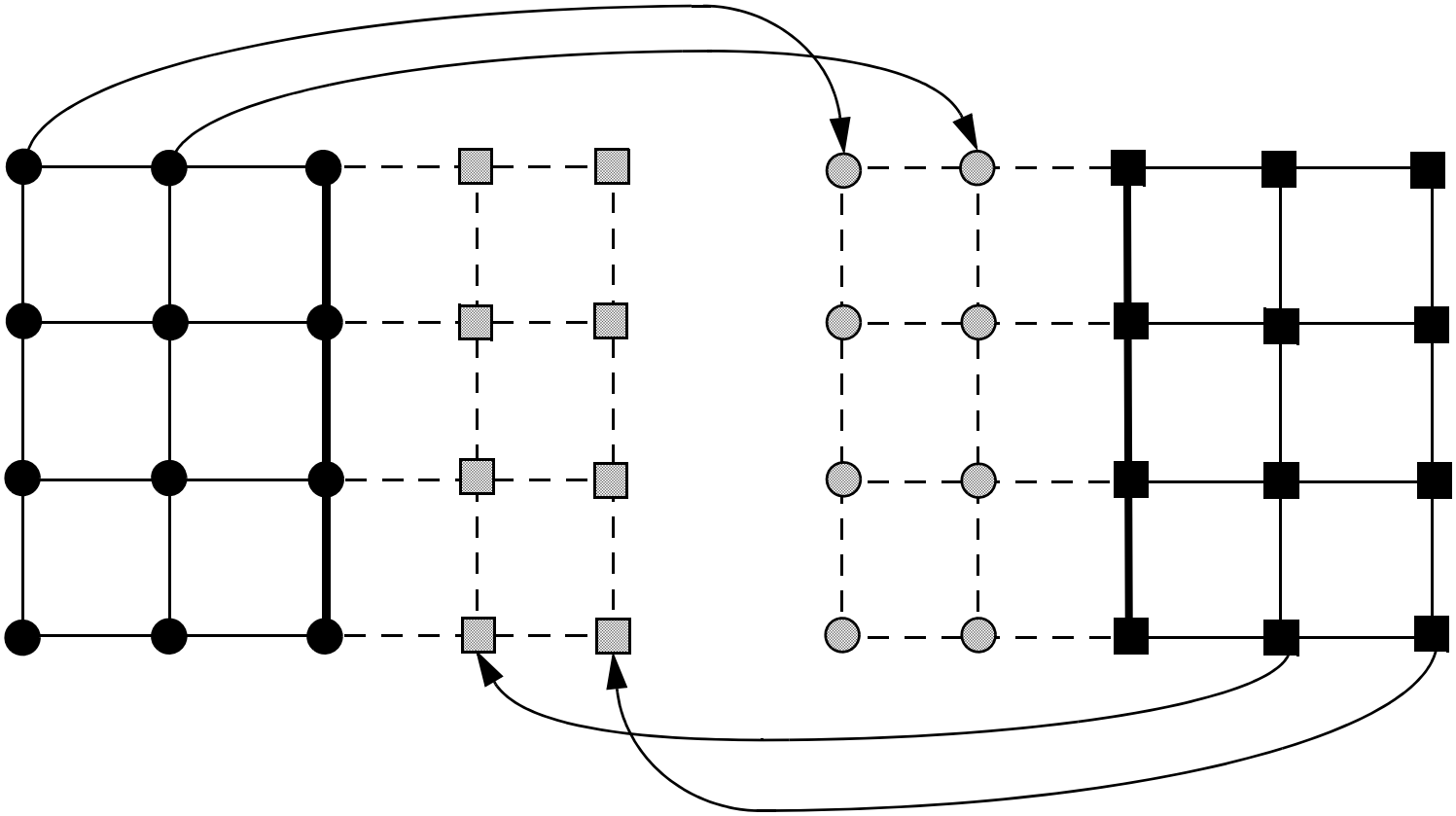
ITYP = 102 Singular Line plus averaging of all boundary data
ITYP = 103 using 0th (ityp=102) or 1st order (ityp=103)
extrapolation

ITYP = -101 Singular Cut
not supported any longer

Boundary Conditions: Cut Segments (ITYP $\leq$ 0)

ITYP = 0 Internal Cut / Cut to the same block

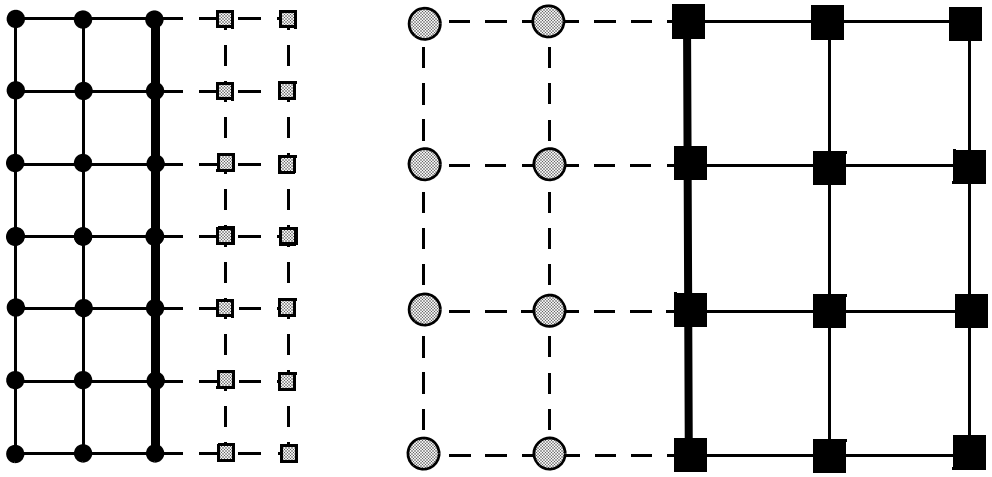
ITYP = -1 Cut to annother block



DLR

Boundary Conditions: Cut Segments (ITYP $\leq$ 0)

ITYP = -2 Cut on discontinuous block interfaces
with fixed stride relationship
(conservative treatment for CC metric implemented)



ITYP = -3 Cut on discontinuous block interfaces
with arbitrary node distributions
(no conservative treatment available)

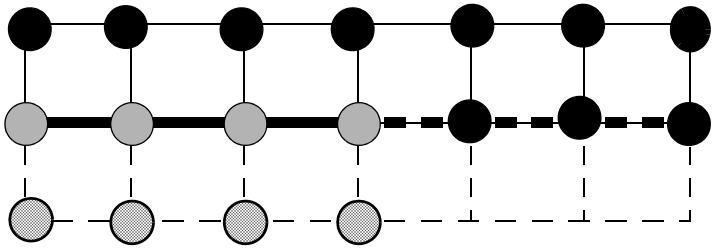
Boundary Management in case of Cell Vertex Metric

- **Upate in the order of appearance in the Grid Logic file**

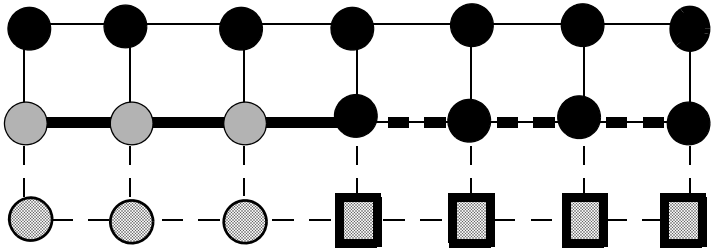
- **Recommended Order:**
  1. **Farfield & Engine**
  2. **Walls**
  3. **Symmetry**
  4. **Cuts**

- **Example:**

1st step:
No slip B.C.



2nd step:
Cut B.C.



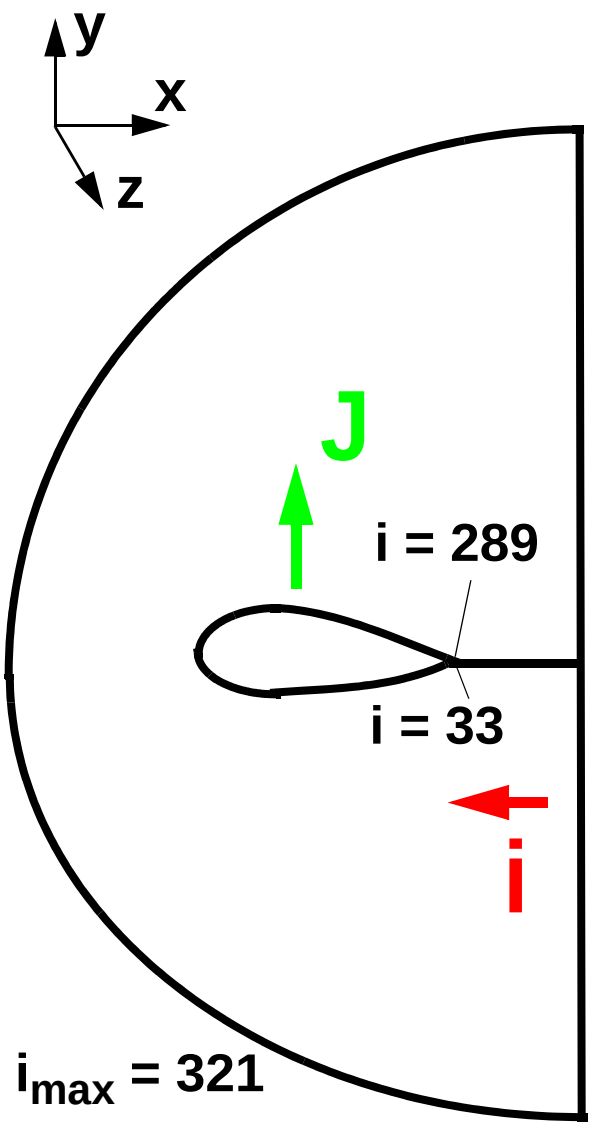
DLR

Block Structure and Grid Logic, Part 3:

Creation of a "Grid Logic File"

- Tombstone Logic, with Example
- Manual Creation of a simple "Grid Logic File"

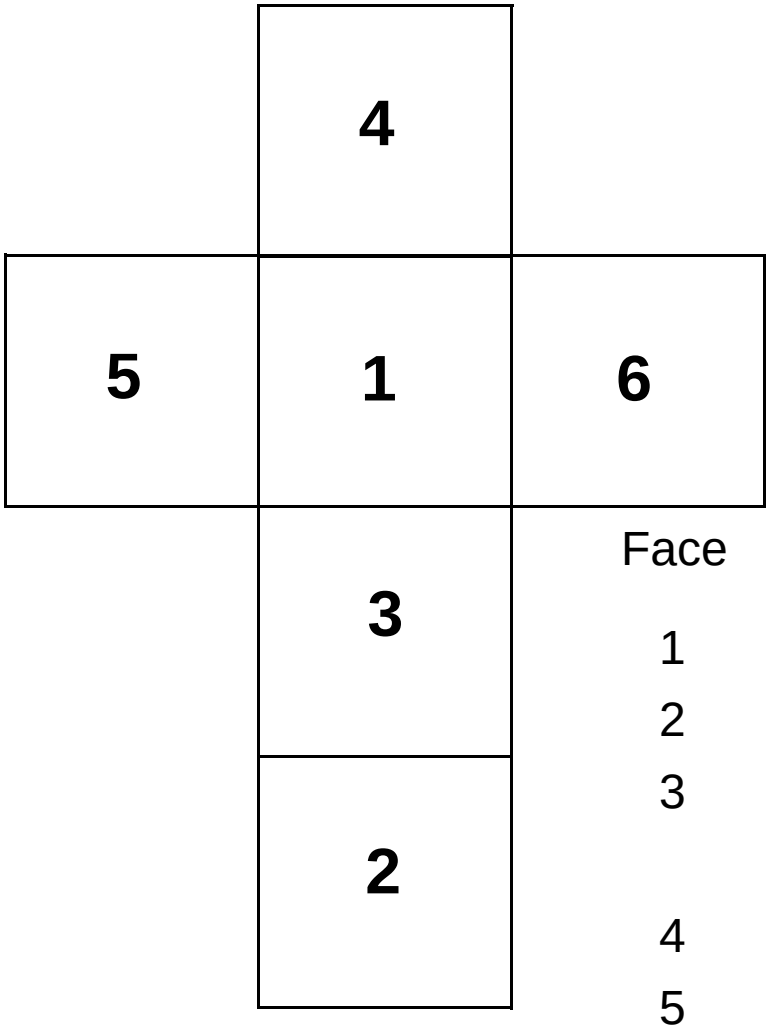
Grid Logic: Tombstone Logic, with Example



$i_{max} = 321$

$j_{max} = 65$

$k_{max} = 1$



Face	L1	L2
1		
2		
3		
4		
5		
6		

DLR

Manual Creation of a simple "Grid Logic File"

```
$$   format for integers in grid point file

$$   format for reals in grid point file

$$   nblock   imax   jmax   kmax   ijkmax   icoord

$$
$$   topology of block no.   1
$$   -------------------------
$$   iblock   nseg(1)  nseg(2)  nseg(3)  nseg(4)  nseg(5)  nseg(6)  isolve

$$    ibeg     iend     jbeg     jend     kbeg     kend  (physical
boundaries)

$$
```

```
$$
$$   segments:
$$
$$   ityp lb l1beg l1end l2beg l2end mbls lbs l1begs l1ends l2begs l2ends
icomp
$$
$$   far field: characteristic variables
      12   1
$$   far field: characteristic variables
      12   2
$$   far field: characteristic variables
      12   4
$$   no-slip wall
       2   3
$$   symmetry to plane z = const.
      23   5
$$   symmetry to plane z = const.
      23   6
$$   cut to another block
       0   3
$$   cut to another block
       0   3
$$
```

Grid Logic: Answer Key

```
$$   format for integers in grid point file
(*)
$$   format for reals in grid point file
(*)
$$   nblock   imax   jmax   kmax   ijkmax   icoord
       1      -99    -99    -99     -99        1
$$
$$   topology of block no.   1
$$   -------------------------
$$   iblock   nseg(1)  nseg(2)  nseg(3)  nseg(4)  nseg(5)  nseg(6)  isolve
       1         1        1        3        1        1        1        1
$$    ibeg     iend     jbeg     jend     kbeg     kend  (physical boundaries)
       2       322        2       66        2        2
$$
$$   segments:
$$
$$   ityp lb l1beg l1end l2beg l2end mbls lbs l1begs l1ends l2begs l2ends
icomp
$$
$$   far field: characteristic variables
      12   1    2    66     2     2    0   0      0      0      0      0     0
$$   far field: characteristic variables
      12   2    2    66     2     2    0   0      0      0      0      0     0
$$   far field: characteristic variables
      12   4    2     2     2   322    0   0      0      0      0      0     0
$$   no-slip wall
       2   4    2     2    34   290    0   0      0      0      0      0     0
$$   symmetry to plane z = const.
      23   5    2   322     2     2    0   0      0      0      0      0     0
$$   symmetry to plane z = const.
      23   6    2   322     2     2    0   0      0      0      0      0     0
$$   cut to another block
       0   3    2     2     2    34    1   3      2      2    322    290     0
$$   cut to another block
       0   3    2     2   290   322    1   3      2      2     34      2     0
$$
```