Since the introduction of the STAR-CCM+ solver and an automatic polyhedral mesher by CD-adapco, we are asked one question more than any other: why use polyhedral rather than tetrahedral meshes? In this article we provide some explanations and a demonstration providing some answers to this question.

Tetrahedra are the simplest volume elements; their faces are plane segments, so both face and volume centroid locations are well defined. Tetrahedral meshes are also relatively easy to generate automatically; their use is often standard practice in automatic mesh generation and they are used by all major CFD software tools. On the negative side, tetrahedra cannot be stretched too much, so in order to achieve a reasonable accuracy in boundary layers, long channels or small gaps, a much larger number of control volumes is needed than if structured (hexahedral) meshes are used. These problems are partly overcome by using prism layers along walls.

Tetrahedral control volumes have only four neighbors, and computing gradients at cell centers using standard approximations (linear shape functions) can be problematic. One problem is linked with the spatial position of neighbor nodes, because they may all lie in nearly one plane, making it impossible to evaluate the gradient in the direction normal to that plane. Another problem is encountered for cells next to boundaries: even when only one face is a boundary face, the remaining three neighbors may be unfavorably distributed, moreover along edges or at corners, one may end up with only one or two neighbor cell, which can cause serious numerical problems, let alone reduced accuracy.

In order to achieve accurate solutions and good convergence properties on tetrahedral meshes, one needs special discretization techniques and a large number of cells. None of these remedies is optimal; the first makes the code more complicated, more difficult to extend and maintain, while the second increases the memory and computing time requirements.

Polyhedral cells are especially beneficial for handling recirculating flows. Tests have shown that, for example, in the cubic lid-driven cavity flow, many fewer polyhedra are needed to achieve a specified accuracy than even Cartesian hexahedra (which one would expect to be optimal for rectangular solution domains). This can be explained by...
the fact that, for a hexahedral cell, there are three optimal flow
directions which lead to the maximum accuracy (normal to each of the
three sets of parallel faces); for a polyhedron with 12 faces, there are
six optimal directions which, together with the larger number of
neighbors, leads to a more accurate solution with a lower cell count.

A more detailed analysis of properties of various mesh types and some
results from test cases are published in an article by Peric (2004) [1].

Comparisons in many practical tests have verified that with polyhedral
meshes, one needs about four times fewer cells, half the memory and
a tenth to fifth of computing time compared to tetrahedral meshes to
reach solutions of the same accuracy. In addition, convergence
properties are much better in computations on polyhedral meshes,
where the default solver parameters usually do not need to be
adjusted. An example is presented below to demonstrate this.

The test case represents a water jacket of an engine; the pressure
drop between inlet and outlet is the relevant engineering quantity that
is monitored. Computations were performed on six polyhedral and six
tetrahedral grids, with numbers of cells ranging between 21872 and
593888 (polyhedra) and between 39587 and 2322106 (tetrahedra).
Prismatic layers along walls were generated in all cases. Figure 01
shows one polyhedral mesh.

Figure 02 shows computed pressure distribution on the finest
tetrahedral and polyhedral mesh. Since these meshes were very fine,
the results are also very similar. In order to determine the mesh
dependency of the solutions, the pressure drops computed on all grids
are compared in Figure 03. This figure shows that the results from
both grid types are indeed – as required – converging against a grid-
independent value. In all cases, the same discretization and solution
method was used (second-order upwind differencing scheme for
convective fluxes)

It is important to note that the results obtained on any polyhedral
mesh are more accurate than the results obtained on a tetrahedral
mesh with a comparable number of cells. Actually, the result from
polyhedral mesh with 65513 cells is slightly more accurate than the
result from a tetrahedral mesh with 393273 cells (about 6 times
more). The computing time on this polyhedral mesh is less than one
tenth of the computing time for the tetrahedral mesh that would
deliver the same accuracy. Similar conclusions resulted from all
comparative applications performed so far, verifying that the decision
by CD-adapco to develop this new technology was the right one. With
the availability of CD-adapco’s fully-automatic mesh generation tools
for polyhedral control volumes, polyhedral meshes are expected to
soon become the standard in engineering applications.

Reference
[1] M. Peric: Flow simulation using control volumes of arbitrary

Figures
01: One polyhedral mesh for the computation of flow through a water jacket of
an engine.
02: Pressure distribution on the walls of the water jacket as predicted on the
finest tetrahedral mesh (left) and the finest polyhedral mesh (right).
03: Pressure drop between inlet and outlet of the water jacket, as predicted on
different tetrahedral and polyhedral meshes.