

局部容差的定义与应用

——精度、容差和局部容差

Carl

20240730

背景

- ZW3D当前有大量和容差与精度的问题
- 部分是能力问题，部分是原则性问题，目前已经混淆

定义

- 精度：数值求解过程中对 0.0 的定义
- API容差：建模层级提供给用户的控制值，含义视功能而定
- 局部容差：Brep表达下，为弥补边界处不水密而采取的补救措施

精度、API容差——曲线插值

- 接口形式: `int CrvInterpolating(points, result);`
- 接口行为: 曲线经过所有的点, $\text{Dist}(\text{pnt}, \text{crv}) < \text{delta}$
- 没有 API 容差, delta为精度

精度、API容差——曲线拟合

- 接口形式: `int CrvFitting(points, tol, result);`
- 接口行为: 可理解为优化问题, 用户希望最终曲线与拟合点的偏差小于 `tol`
($\text{Dist}(\text{result1}, \text{result2}) < \text{tol} + \text{delta}$)
- `tol` 为 API 容差, `delta` 为精度

精度、API容差——距离

- 接口形式: `int DistGeomGeom(geom1, geom1, tol, result);`
- 接口行为: 多项式的极值求解, $\text{Dist}(\text{result1}, \text{result2}) < \text{minDist} + \text{delta}$, 原则上不需要 `tol`
- 出于效率因素, 用户不需要精确解, 只需要比真实值的偏差小于 `tol` 即可
($\text{Dist}(\text{result1}, \text{result2}) < \text{minDist} + \text{API} + \text{delta}$)
- `tol` 为 API 容差, `delta` 为精度
- 问题: `tol` 是否应该存在

精度、API容差——曲线投影

- 接口形式: `CrvProjectToSurface(curve, surface, tol, result)`
- 接口行为:
 - 理论解: 曲线所有点在曲面求最小距离, 显然不实际
 - 用户能接受: 投影结果的形态与原始曲线类似, 一般解法为: 采样 + March + 拟合
 - 涉及求值精度 (δ)、距离计算精度 ($\text{tol1} + \delta$)、拟合精度 ($\text{tol2} + \delta$)
 - 此外, 为了避免采样点过于接近, 还需要控制相邻采样点的距离 (tol3)
- 问题: 入参为何只有一个tol? 合理吗?

精度、API容差——曲线投影

- tol 为 API 容差, delta为精度
- 容差控制体系
 - 依据tol调整tol1（距离计算容差）、拟合容差（tol2）、临近点判断距离（tol3）的值,
 - 容差控制没有理论解, 只有工程经验
- delta不可控, 且会叠加, 所以浮点精度为 $1e-15$, 但保证精度（有效精度）一般都较低
 - $1e-12$ (ZW3D)、 $1e-11$ (PK)
 - 影响界面用户的输入值范围（box的长度为10.0或9.9999999999的极限位数)

精度、API容差——线线求交

- 接口形式: `int CrvlsectCrv(crv1, crv2, tol, result);`
- 令曲线 $C(u)$ 和 $S(v)$, 曲线的交等价于 $C(u)-S(v)=0$ 的解 (多项式求根)

序号	场景	tol的含义
1	精确的结果	无需dTol, 迭代的终止精度设置为 $\text{dist}(p1,p2)<\text{delta}$
2	与真值有一定偏差	需要tol, 结果需满足 $\text{dist}(p1,p2)\leq\text{tol}+\text{delta}$
3	有交点时获得准确交点, 没有交点时以tol作为容差判定条件	

- 问题: 线线重合与tol有关吗? tol是否必要?

精度、API容差——线线求交

- tol 为 API 容差, delta为精度
- 容差控制体系
 - 依据tol调整tol1（距离计算容差）、拟合容差（tol2）、临近点判断距离（tol3）的值,
 - 容差控制没有理论解, 只有工程经验
- delta不可控, 且会叠加, 所以浮点精度为 $1e-15$, 但保证精度（有效精度）一般都较低
 - $1e-12$ (ZW3D)、 $1e-11$ (PK)
 - 影响界面用户的输入值范围（box的长度为10.0或9.9999999999的极限位数)

精度、API容差——其他接口

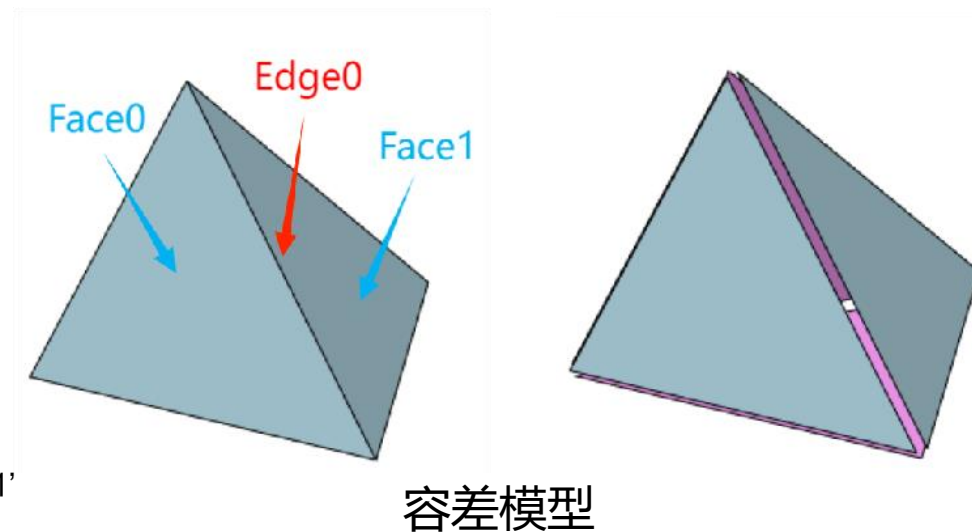
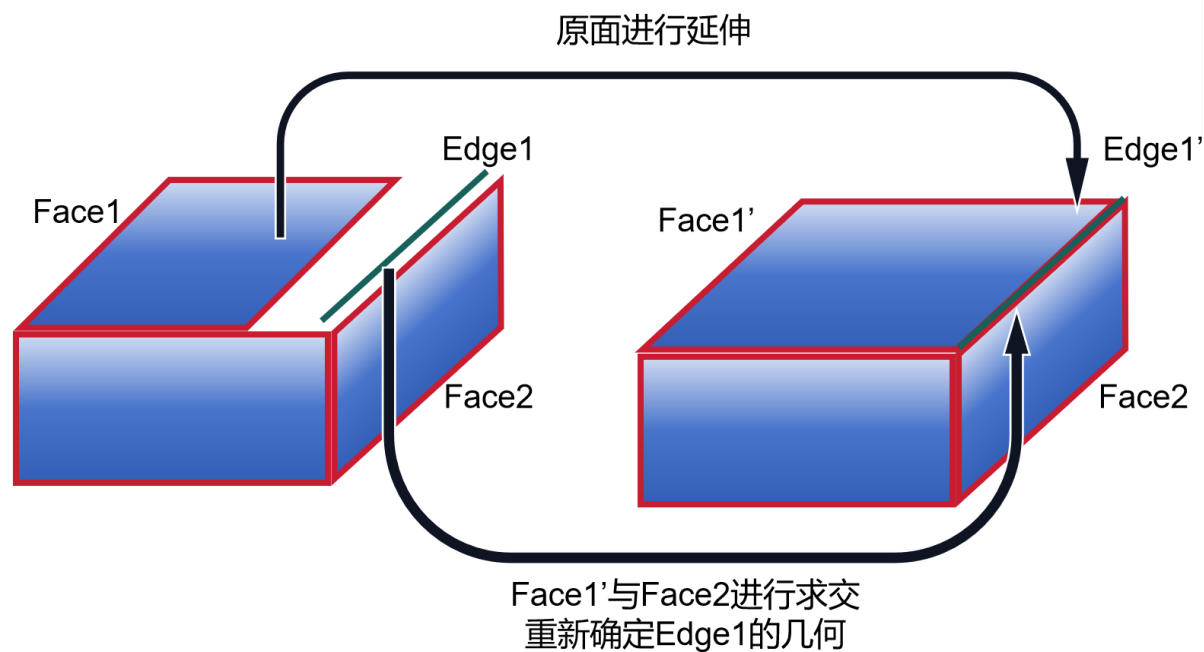
- 拉伸: Extrude(origin, result)
- 布尔: Boolean(obj1, obj2, operation, tol)
- 边圆角: Fillet(edge, parameter, tol)
- NOTE:
 - 每个接口是否应该有API容差是需要思考的
 - 每个API容差应当有其明确意义, 其必须明确的告知所有的调用者
 - 当前内核Overdrive的接口设计并不总是合理的, 开发过程需要多思考, 多交流

局部容差——产生的原因

- 在CAX系统中，应用边界面表达法（B-Rep）来描述一个三维物体。 n 维拓扑Cell是依赖于 n 维拓扑domain的边界来进行定义的，而 n -domain又由 $n-1$ -Cell来进行定义；比如3-Cell（region）由3-domain（shell）定义，而shell由2-Cell（face）来定义。
- 但在拓扑操作、上层建模算法应用、模型导入导出以及渲染等情形下，拓扑结构又会以常用的solid（3维拓扑）-face（2维拓扑）-edge（1维拓扑）-vertex（0维拓扑）来进行定义。
- 这两套拓扑定义同时存在就会不可避免地造成1维拓扑edge的几何和B-Rep中1-Cell（Fin）的几何不同，从而产生几何与拓扑不相容的问题。

局部容差——作用

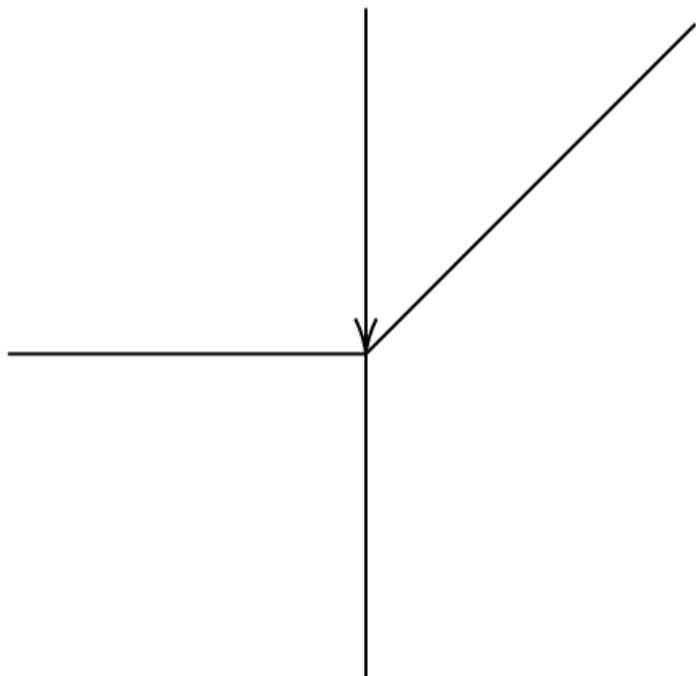
1. 解决几何与拓扑不相容的问题
2. 简化几何计算



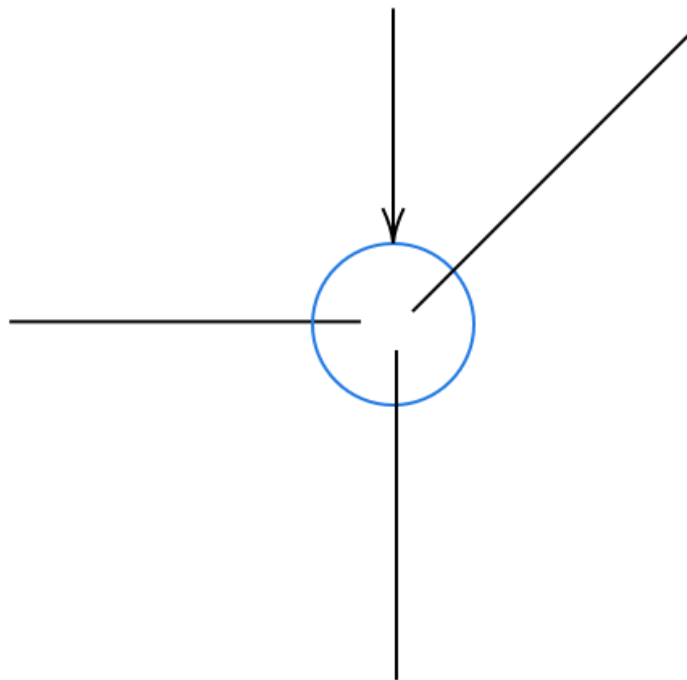
局部容差——分类

- 容差顶点 (ϵ_{vertex})

Vertex with global tolerance



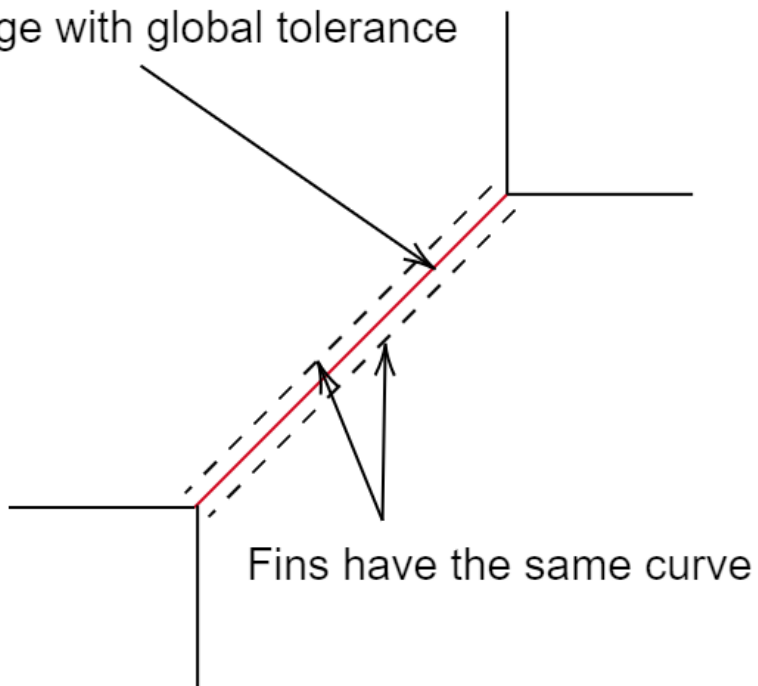
Vertex with local tolerance



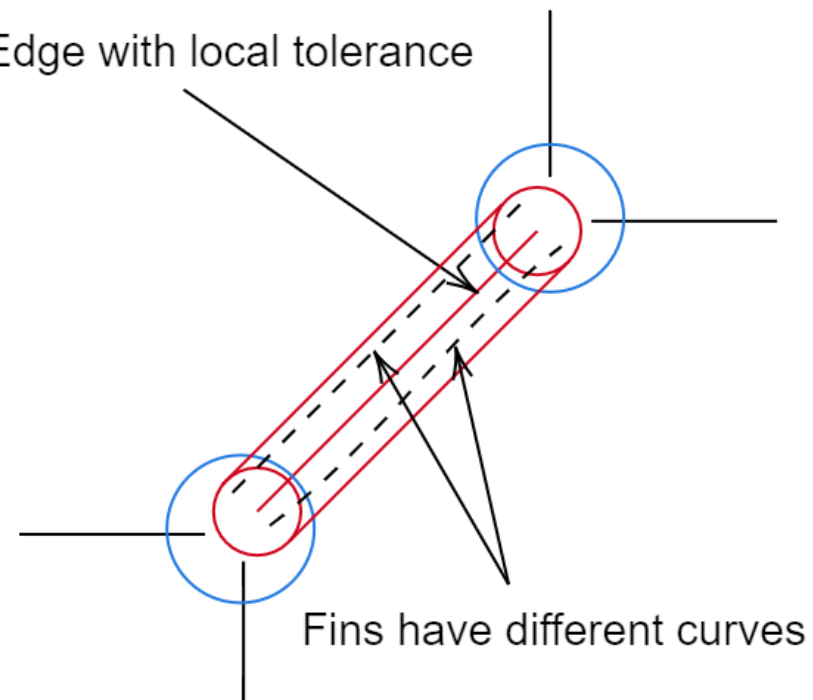
局部容差——分类

- 容差边 (ϵ_{edge})

Edge with global tolerance

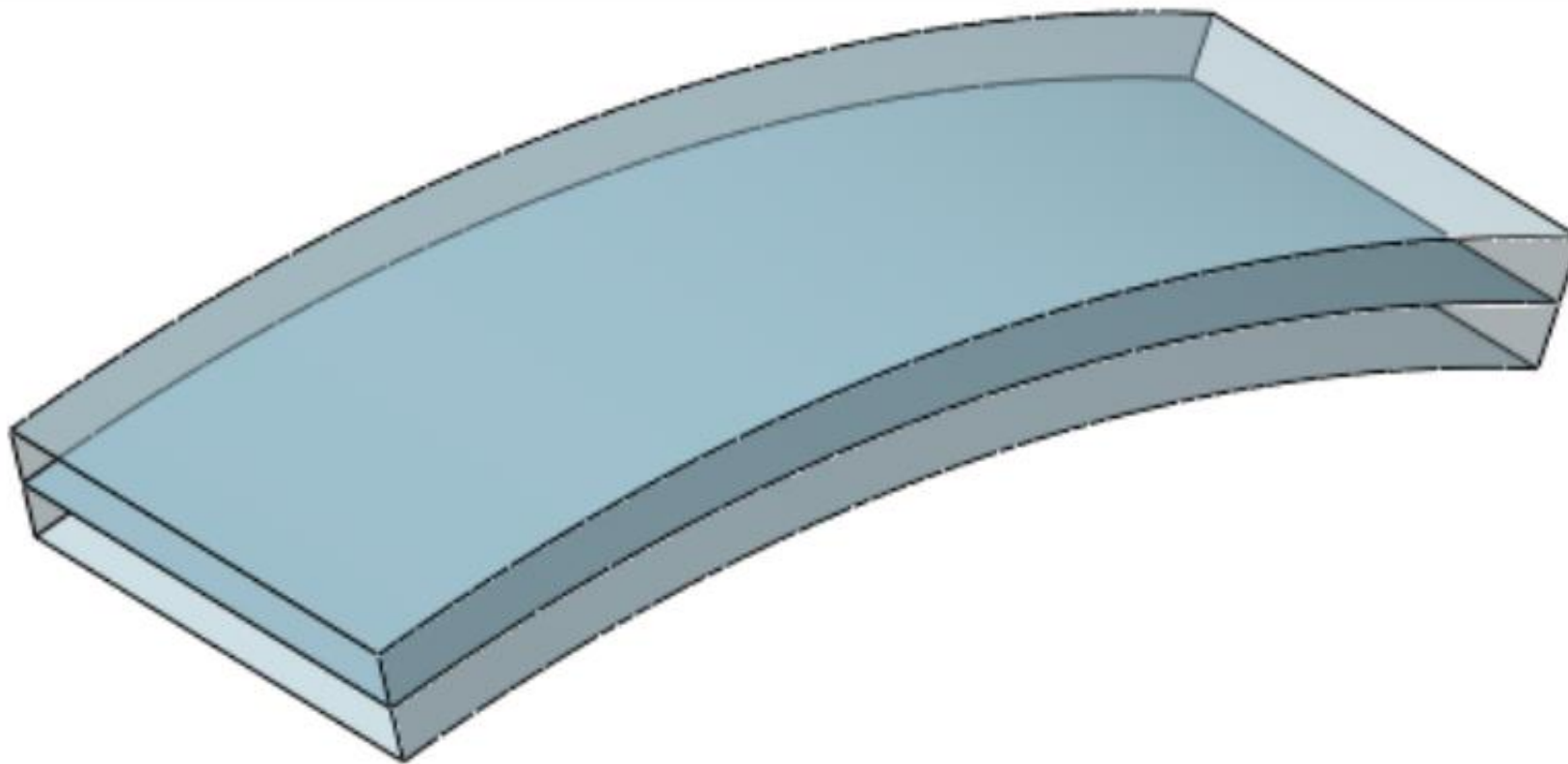


Edge with local tolerance



局部容差——分类

- 容差面 (ϵ_{face})



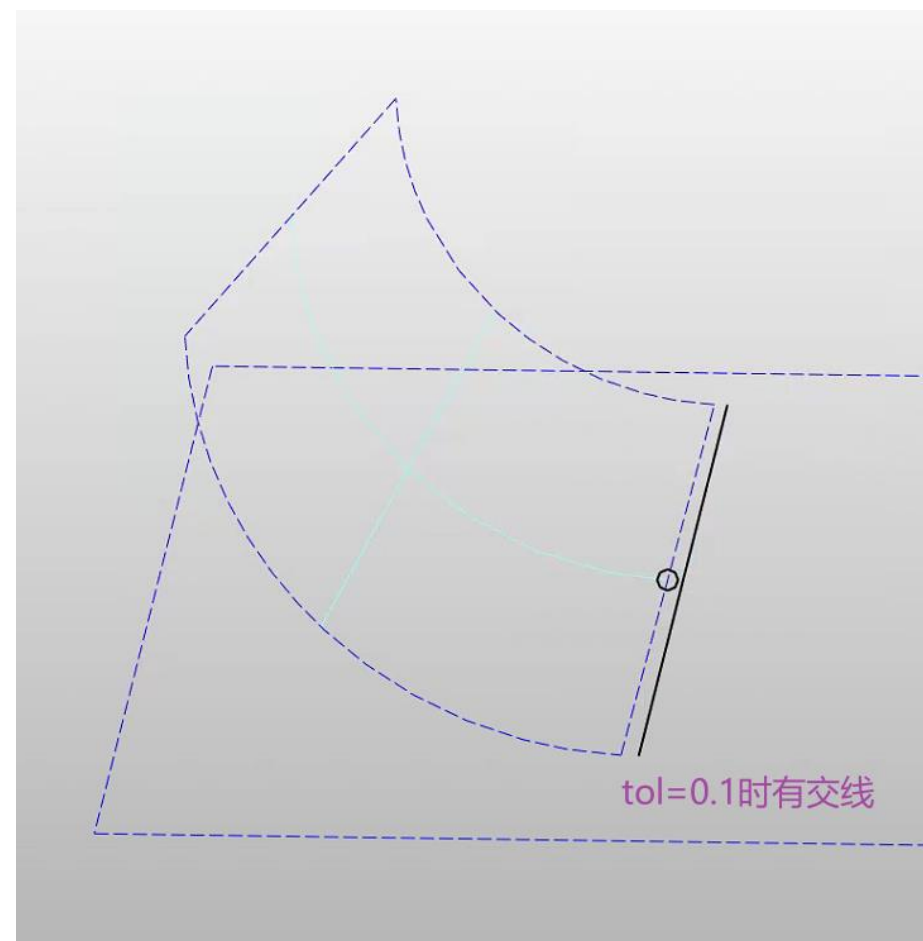
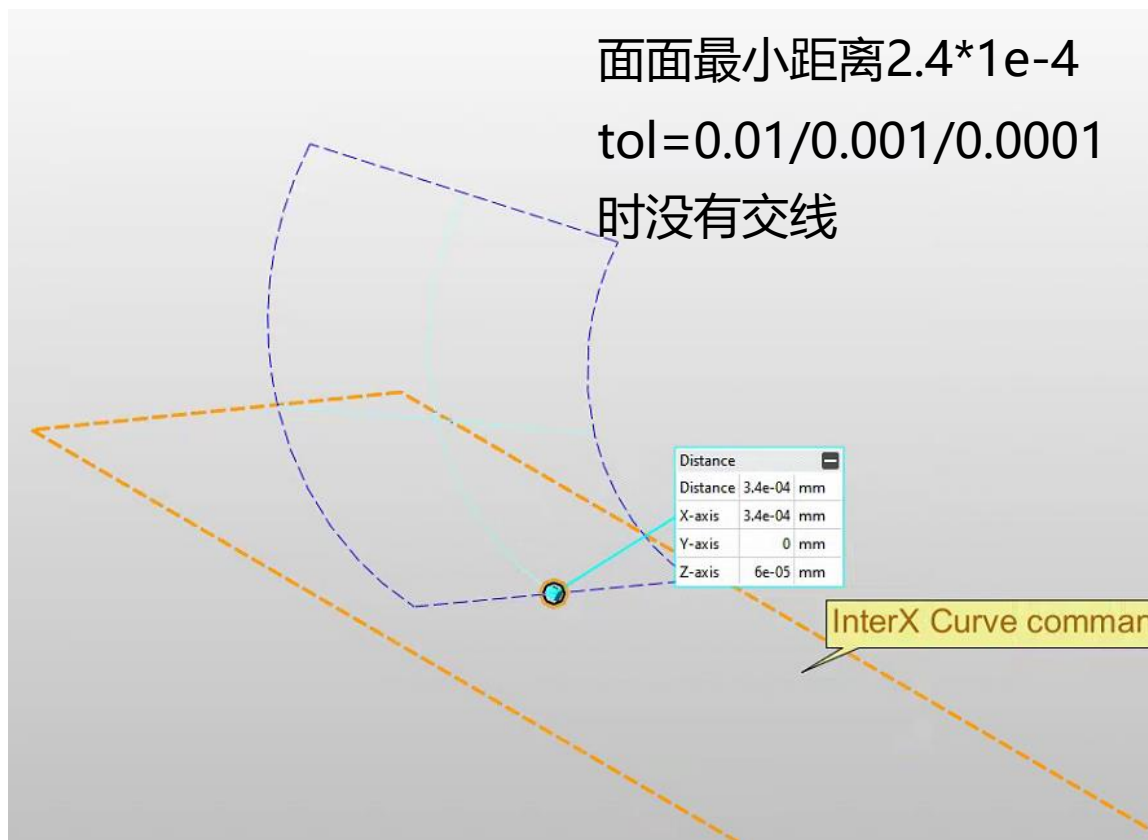
局部容差——处理原则

【企微文档】副本-容差体系

https://doc.weixin.qq.com/doc/w3_ATQAOgbbANQ5qTj2uVjQ1uGQ5BLX0?scode=AGsAwAf3AAo2ebomzQATQAOgbbANQ

案例介绍

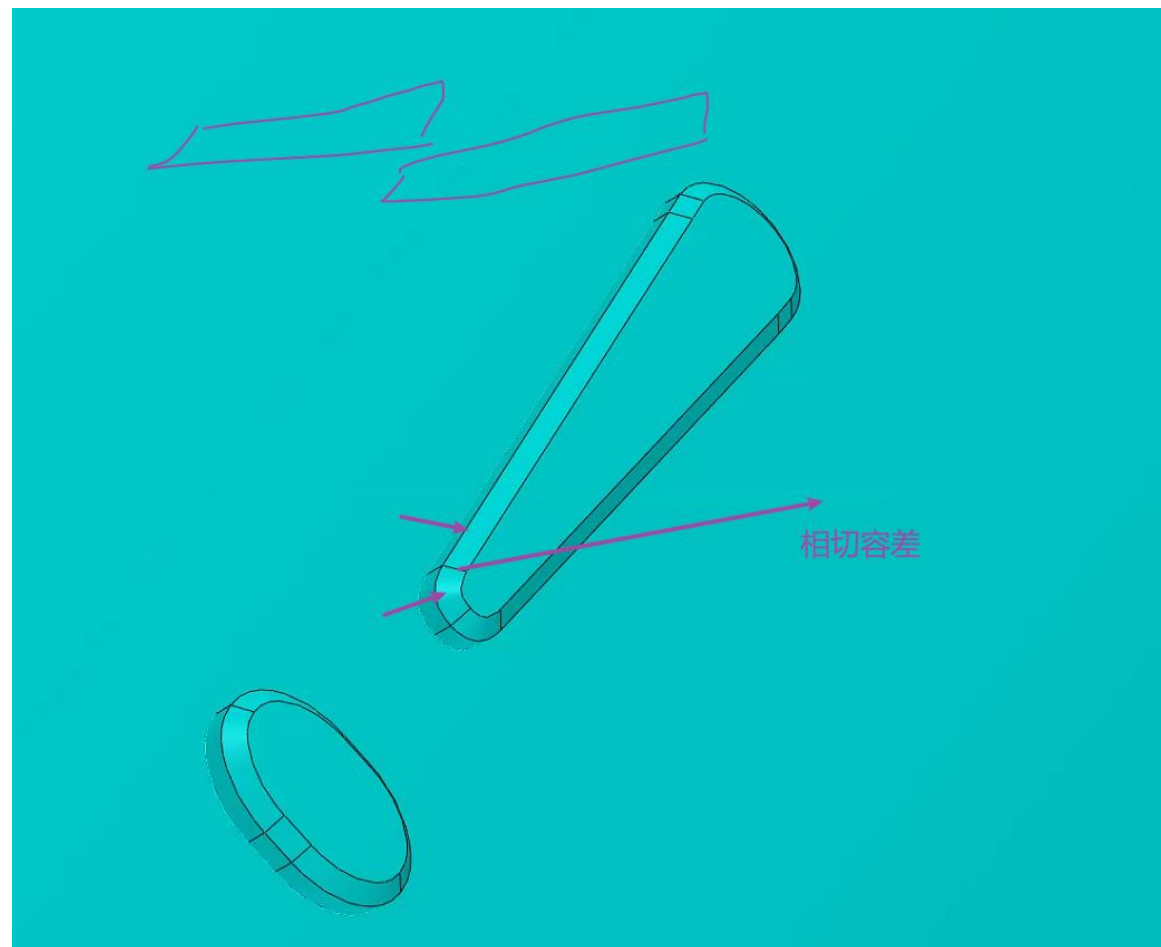
- API容差不明确: <https://jira.zwcad.com/browse/ZW3D-20721>



案例介绍

- 局部容差体系失效

/Shape/Feature/Inlay/Inlay1.mac



总结

- 精度、容差和局部容差具有明确的定义，开发过程应该避免混淆
- 容差体系的构建需要各模块齐心协力解决
- 内核开发是一项长久事业

谢谢聆听