An Open Framework for Unstructured Grid Generation

32nd AIAA Fluid Dynamics Conference & Exhibit
St. Louis, Missouri
June 25, 2002

William T. Jones
NASA Langley Research Center
Hampton, VA
w.t.jones@larc.nasa.gov

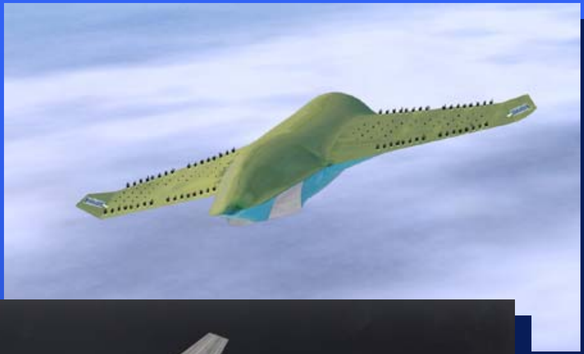
Overview

- Background
- Development of Application Programming Interface
- GridEx a framework based application
- Examples
- Concluding remarks

William T. Jones
w.t.jones@larc.nasa.gov

Fast Adaptive Aero-Space Tools (FAAST)

- **<u>Objective</u>**: Develop fast adaptive methods for the analysis and design of complex aerospace configurations in all speed regimes
  - Automated/unstructured grid generation from CAD adapted to specified global error tolerances
  - Modular software synthesis framework for developing/exploiting improved algorithms and physical models
  - Adaptive grid/order/physics solvers 100x faster than 1999 solvers
  - Shape optimization capabilities for advanced concepts.
- **Elements**
  - Adaptive CAD-Grid Methods
  - High-Energy Flow Solver Synthesis
  - Optimally Convergent Algorithms
  - Efficient Adjoint Design Methods
- **Partners**
  - MIT, ICASE, Weizmann Institute



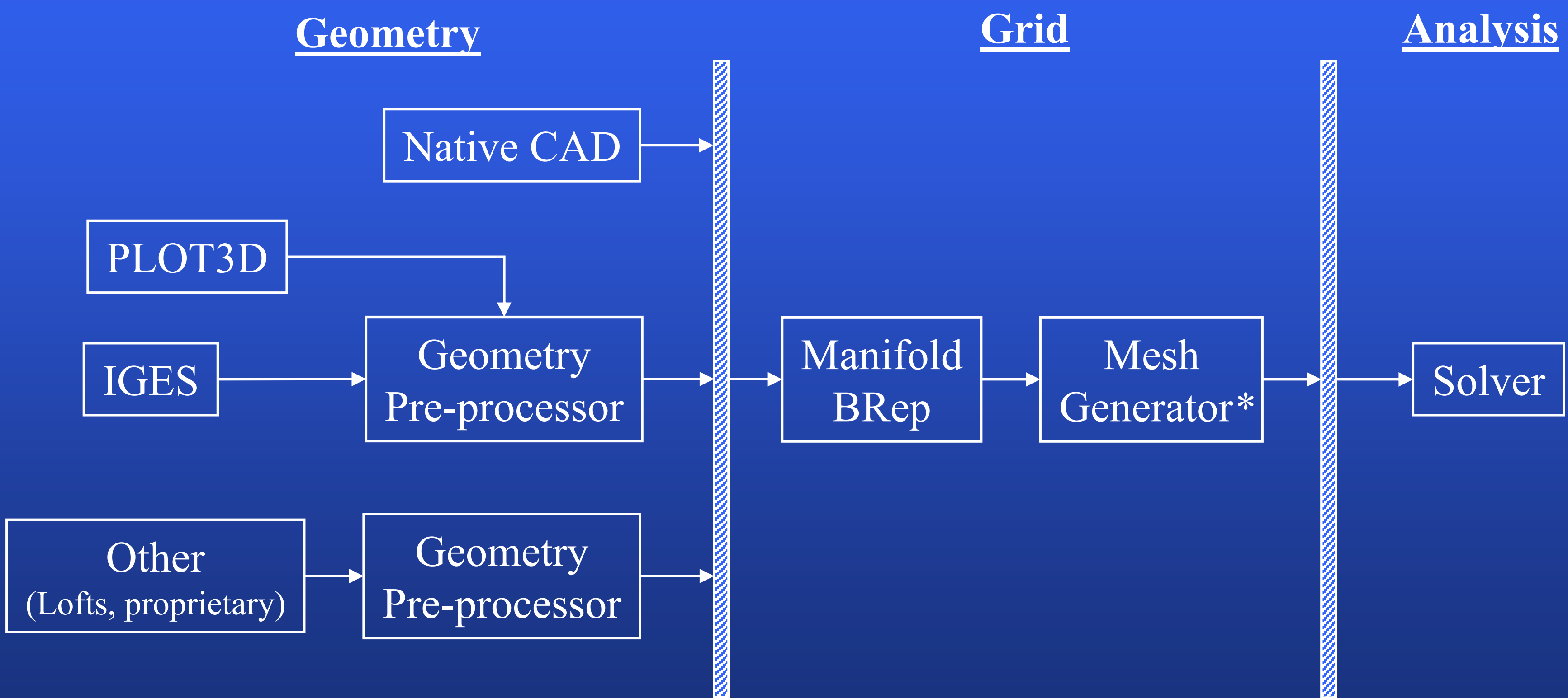
William T. Jones
w.t.jones@larc.nasa.gov

Goals for FAAST Adaptive CAD/Grid Methods

- Automate numerical grid generation for complex geometry
- Develop 3D adaptive capability to generic error estimates
- Rapidly incorporate emerging technologies
- Combine/interchange available techniques
- Collaborative development
- Promote code reuse
- Reduce overall software maintenance

William T. Jones
w.t.jones@larc.nasa.gov

Optimized Grid Generation Process



* Mesh generator takes geometry as “gospel” and encapsulates mesh specific definitions such as element size, boundary conditions, etc.

William T. Jones
w.t.jones@larc.nasa.gov

Traditional Grid Generation Software Development

- Direct incorporation of algorithms and requirements
  - Sufficient for targeted goals
  - Historically produce high quality results
- Closed systems (proprietary?)
  - "Static product"
    - Fixed method of operation
    - Little/No customization
    - Difficult infusion of new technologies limits growth potential
- Small development teams (1-2 persons)
- Monolithic software products
  - Difficult to maintain
  - Little code reuse

William T. Jones
w.t.jones@larc.nasa.gov

Traditional Grid Generation Software

- Direct incorporation of algorithm requirements
  - Application includes available methods of implementation
    - Only methods included are available
  - Functions request results from available methods
  - Functions modified to accommodate additional algorithms

Function A ⟷ *Method*

Function A	*Method*
…	…
Call *Method* †	Implement *Algorithm*
Use results	Return results
…	…
End *Function A*	End *Method*

† Note: *Method* may be inlined

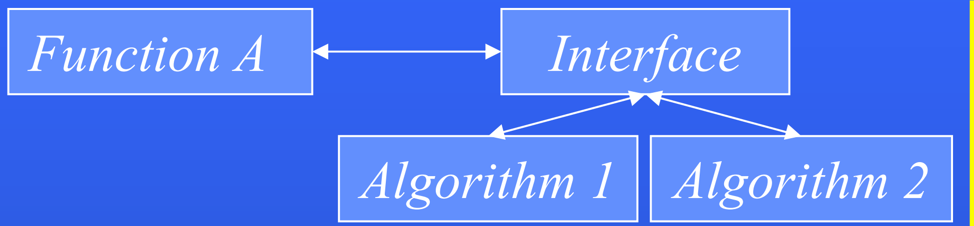
William T. Jones
w.t.jones@larc.nasa.gov

Framework Approach to Software Development

- Indirect incorporation of algorithms
  - Request/results thru Application Programming Interface (API)
  - Each operation defined by a single entry point
  - Standardized API used by all functions
- Open systems
  - "Dynamic product"
    - Customizable method of operation
    - Rapid infusion of new technologies enhances growth potential
- Collaborative development teams (distributed experts)
- Modular software products
  - Easy to maintain (encapsulated chunks)
  - Promotes code reuse

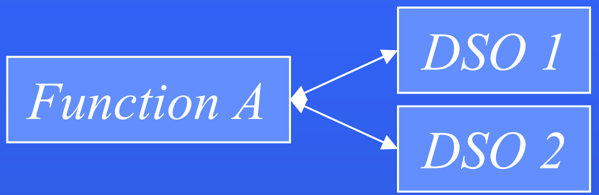
William T. Jones
w.t.jones@larc.nasa.gov

API Approach to Grid Generation



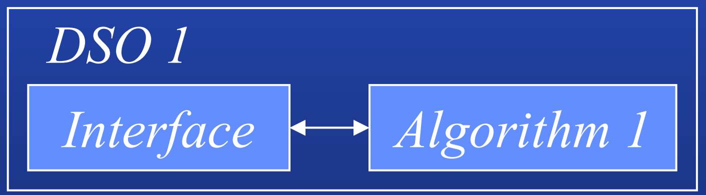

```
Function A
  …
  Call Interface
  Use results
  …
End Function A
```

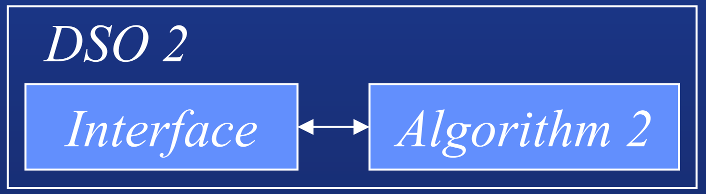
```
Interface
  …
  Switch( param )
    Case Algorithm 1
        Call Algorithm 1
    End Case
    Case Algorithm 2
        Call Algorithm 2
    End Case
  End Switch
  Return results
  …
End Interface
```




```
Function A
  …
  Call Interface
  Use results
  …
End Function A
```

```
Interface
  …
  Algorithm #
  Return results
  …
End Interface
```





National Aeronautics and
Space Administration
Langley Research Center

William T. Jones
w.t.jones@larc.nasa.gov

Grid Generation Framework

- API for Unstructured Grid Generation
- Grid generation defined in terms of component processes
  - Geometry access
  - Generic grid metrics
  - Mesh generation
- Component processes are decoupled
  - Interact solely through a defined interface
    - Implementation is hidden
    - Algorithms may be added/removed/modified/replaced without impact
  - Encapsulation contains error propagation
  - Promotes component testing
  - Reduces maintenance burden/improves productivity of derivative projects

William T. Jones
w.t.jones@larc.nasa.gov

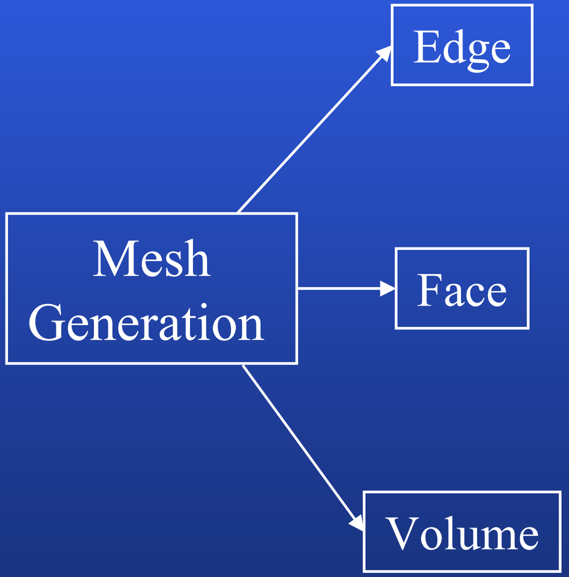
Grid Generation Process Decoupling

- Geometry access
  - Evaluate, snap, local normal, etc.
  - BRep provides geometry and Topology
  - CAPrI provides vendor-neutral API (B. Haimes, MIT)
    - No change to derivative application for a change in modeler
    - In direct alignment with current work
    - Support for major CAD modelers
- Generic grid metrics
  - Local edge lengths along 3 principal directions
  - Anisotropic control
  - Encapsulate algorithmic details (sources, edge seeding, etc.)

William T. Jones
w.t.jones@larc.nasa.gov

Grid Generation Process Decoupling cont.

- Mesh generation
  - Discretization and refinement
  - Separated into phases for flexibility
  - Operational order from hierarchy
  - Algorithms leverage API for geometry and metric information and the meshing of lower level entities
    - Allows for transparent exchange of techniques



National Aeronautics and Space Administration
Langley Research Center

William T. Jones
w.t.jones@larc.nasa.gov

Unstructured Grid Generation API Sample

UGMesh_DiscretizeEdge()

Input: Domain/Edge identifier

Output: Number, coordinates, parameter of computed nodes

GMetric_GetSpacing()

Input: Target coordinates

Output: Spacing, principal directions

CADGeom_LengthOfEdge() †

Input: Domain/Edge identifier, parameter bounds

Output: Physical length of segment

CADGeom_NearestOnEdge() †

Input: Domain/Edge identifier, target coordinates, parameter estimate

Output: Solution coordinates, parameter

CADGeom_PointOnEdge() †

Input: Domain/Edge identifier, parameter, derivative flag

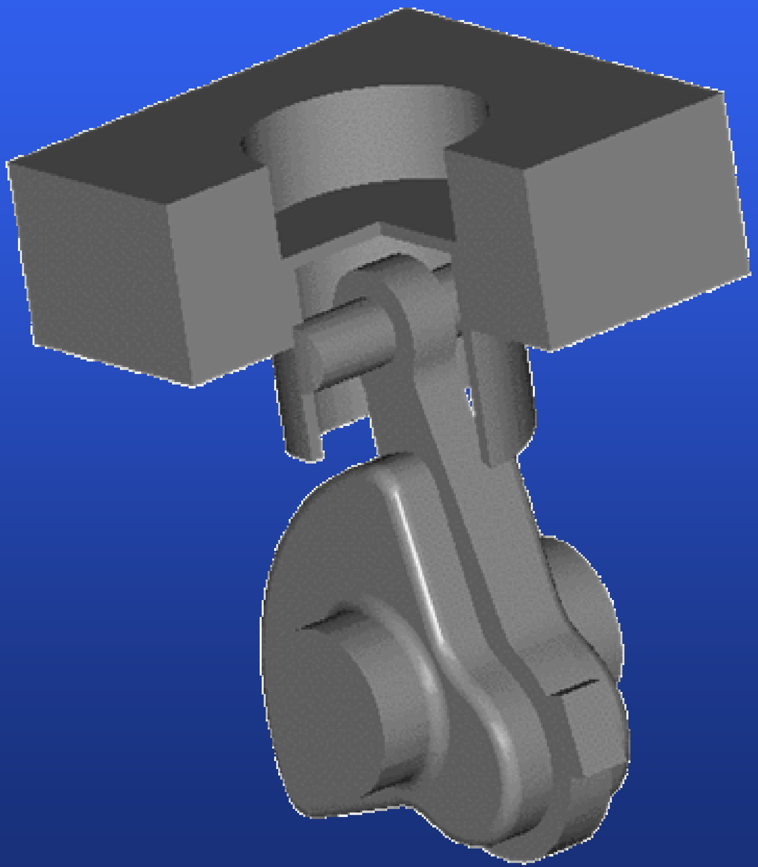
Output: Coordinates, [*derivatives*]

† Direct wrappers of CAPrI methods

William T. Jones
w.t.jones@larc.nasa.gov

Framework Link to CAD

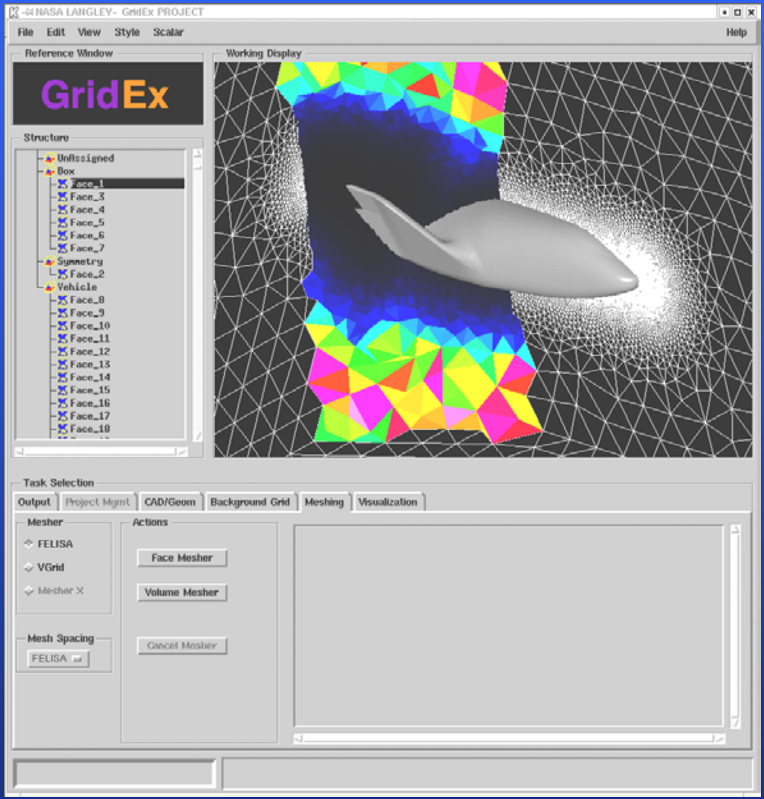
- Indirect access to modeling kernel with CAPrI from MIT
- Manifold solid model
  - Geometry (Points, Curves, and Surfaces)
  - Topology (Hierarchical connections of geometry entities)
- Topology inherent in part
  - Grid generation can be highly automated
- Large setup time reduced
  - Patch definitions preserved in the part
  - Process time reduction upwards of 55% over traditional methods



William T. Jones
w.t.jones@larc.nasa.gov

GridEx Application

- Application built on API Framework
  - Extensible
  - Reduced maintenance burden
- Automated topology extraction (CAPrI) for rapid turnaround
- Independent of Meshing Algorithm
- Ability to customize meshing options
  - Choice of spacing algorithm
  - Side-by-side comparison of algorithms



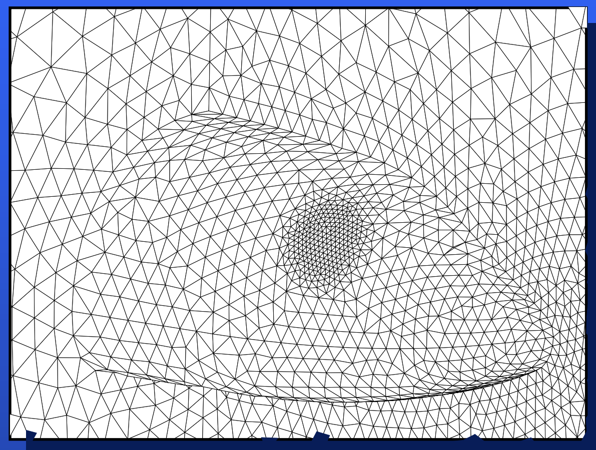
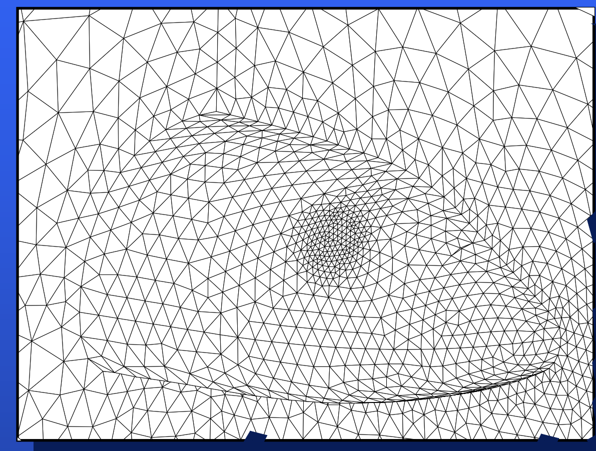
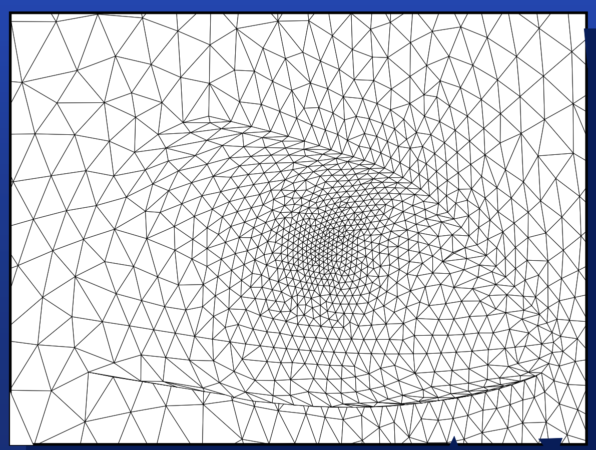
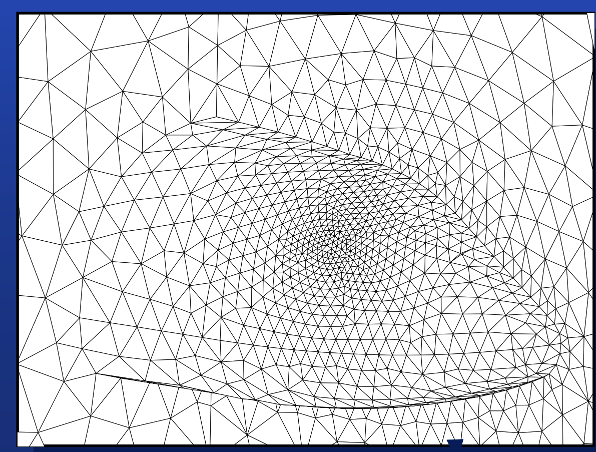
- Initial *Beta* release
  - LaRC FAAST team
  - Lockheed, Aerospace Corp., Cessna*, AFRL*

(* Pending approval)

National Aeronautics and Space Administration
Langley Research Center

William T. Jones
w.t.jones@larc.nasa.gov

Surface Grid Generation Flexibility

	FELISA Mesh	VGRID Mesh
FELISA Spacing		
VGRID Spacing		

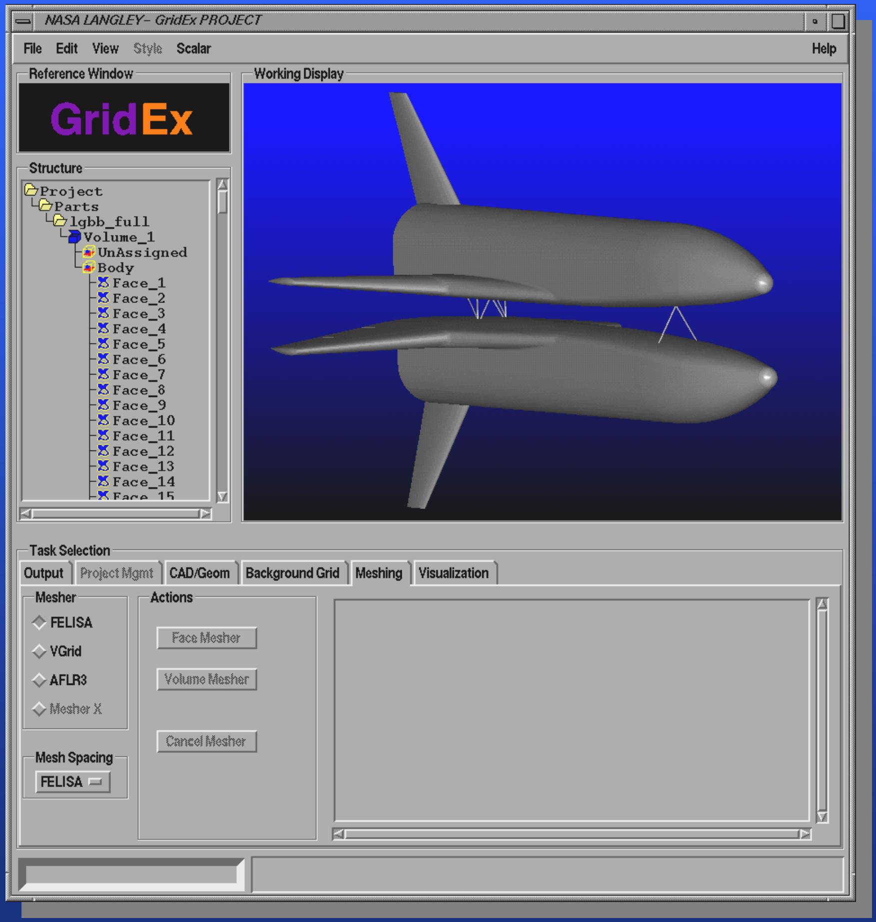
William T. Jones
w.t.jones@larc.nasa.gov

Recent Experiences with Grid Generation API

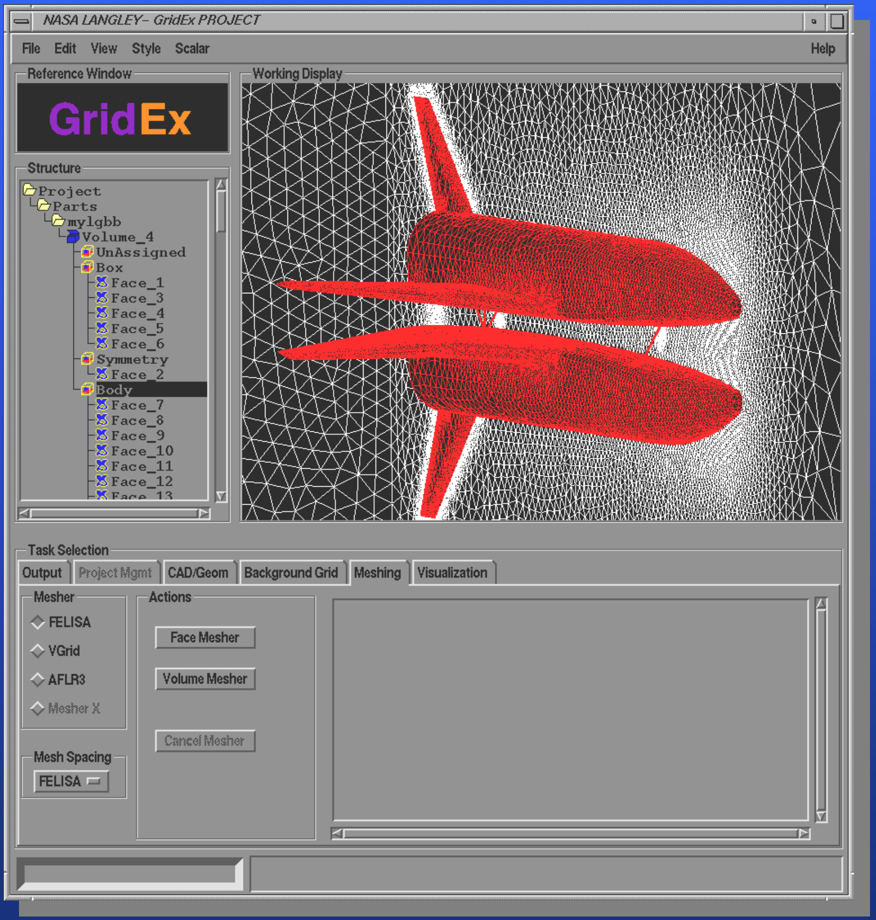
- Facilitated extension of GridEx to include viscous grid support
  - AFLR3 standalone viscous volume grid generator (MSU/ERC)
  - Driver routine written to create/execute script, handle I/O
    - Driver conforms to volume meshing API
  - Integration completed in ***less than 12 hours***
  - Demonstrates support for legacy software
  - Boundary triangulations from FELISA/VGRID techniques
  - No impact on existing GridEx capability
- Links FAAST grid adaptation process to CAD
  - API abstracts geometry/topological access (CAPrI)
  - Maintains grid/geometry association
  - Newly created nodes generated directly on target geometry
  - Grid smoothing utilizes surface parameterization for efficiency

William T. Jones
w.t.jones@larc.nasa.gov

Application of GridEx



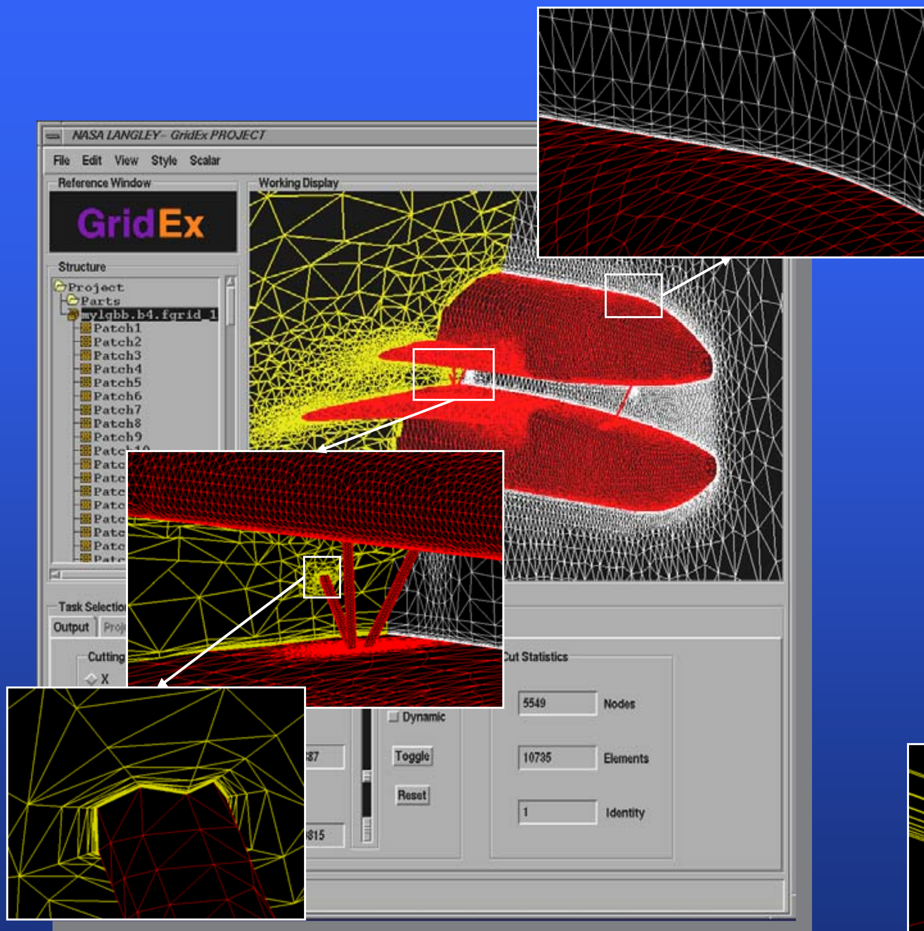
Geometry
(134 topological Faces)



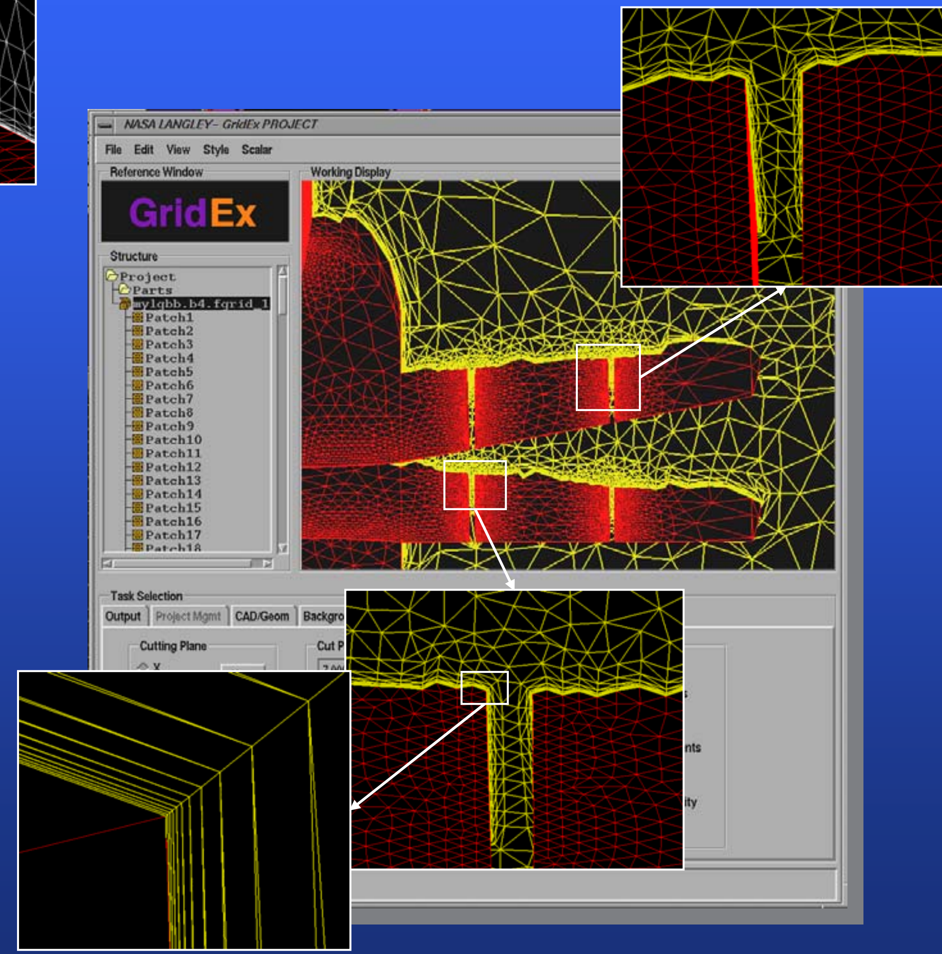
FELISA Surface Mesh
(Topology automatically extracted)

William T. Jones
w.t.jones@larc.nasa.gov

GridEx AFLR3 Extension



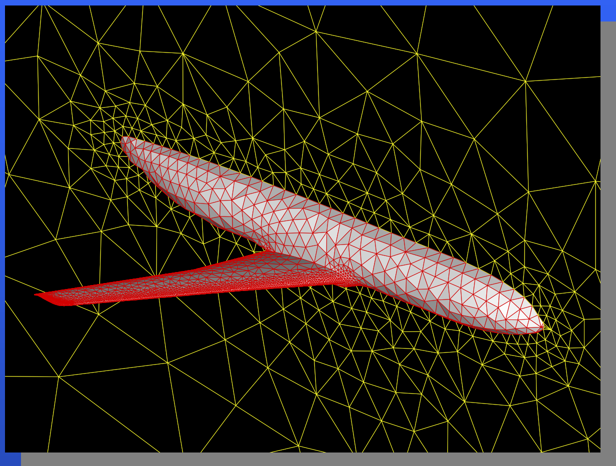
- 107218 surface elements
- 3591675 volume elements
- 3171482 BL elements



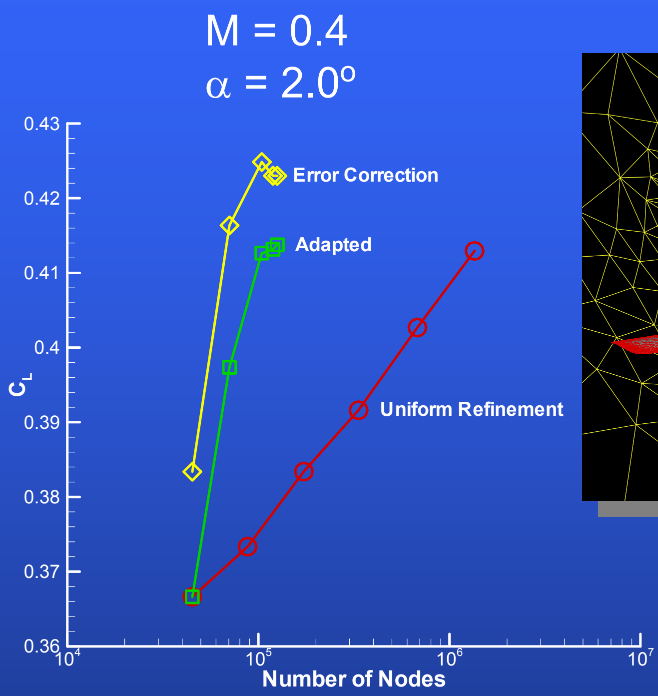
- Under 4 hours from 1st look using FELISA/AFLR3

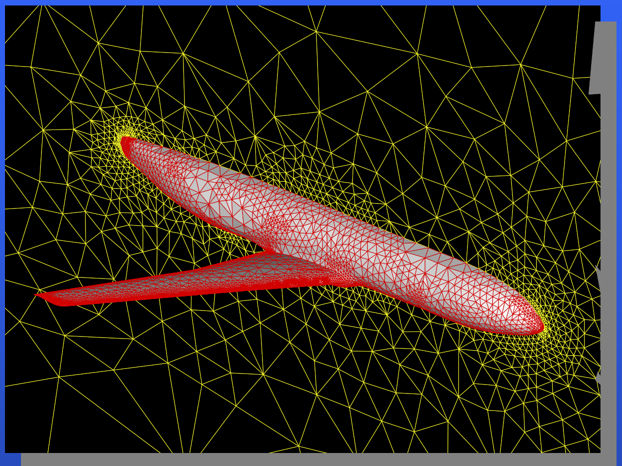
William T. Jones
w.t.jones@larc.nasa.gov

Additional use of Grid Generation API



Original Grid





Adapted Grid

- Used to link FAAST grid adaptation process to CAD
  - 3D inviscid adjoint-based error prediction
  - Adaptation improves accuracy in a given output functional (i.e., lift)
  - An estimate of the numerical uncertainty is provided in the calculation
  - AIAA 2002-3286, M. Park (Session 99-FD-22)

William T. Jones
w.t.jones@larc.nasa.gov

GridEx Advantages

- Significant reduction in process time
  - Integrated, user friendly, comprehensive tool
  - Works directly with original CAD model (no translation)
  - Entities are time stamped to eliminate duplicated effort
  - Viscous grid for complex shape in less than 4 hours
- API based design aids extensions
  - Straightforward addition of new techniques with no impact to existing capabilities
  - User can explore results from a matrix of supported capabilities

William T. Jones
w.t.jones@larc.nasa.gov

Extending the Grid Generation API

- Unstructured Grid Consortium (UGC)
  - AFRL, Boeing (Phantom Works, St. Louis), Lockheed Martin, NavAir, AEDC, NASA LaRC
- Development of a UGC interface standard
  - Initially for unstructured tetrahedral mesh generation
  - 1st draft of Standards Document and Programmer's Reference release imminent.
  - Public comment period to follow release

William T. Jones
w.t.jones@larc.nasa.gov

Concluding Remarks

- An API has been developed to drive the process of Unstructured Grid Generation
  - Provides an abstraction that decouples process components
  - Allows for interchange of algorithms and techniques
  - Facilitates rapid infusion of new technologies
  - Focal point for FAAST grid generation efforts (Adjoint based grid adaptation, etc)
  - Influenced the development of the UGC interface standard
- GridEx application demonstrates API benefits
  - Rapid unstructured grid capability for complex shapes directly from CAD model via CAPrI
  - Framework basis enables extensions to track technology developments

William T. Jones
w.t.jones@larc.nasa.gov