A Distributed Object Approach to Multidisciplinary Analysis of Aerospace Vehicles (AIAA-2000-4898)

Augustine "Gus" R. Dovi,

Philip Su,

Tham Murthy

Computer Sciences Corporation

Hampton, Virginia

8th AIAA/NASA/USAF/ISSMO Symposium on Multidisciplinary Analysis and Optimization

September 6-8, 2000 / Long Beach, California

A Distributed Object Approach To Multidisciplinary Analysis
Presentation Plan

- Purpose
- Motivation
- Background
- Overview of Processes
- Multidisciplinary Analysis Implementation
- Results
- Conclusion

Purpose

- To bring to the MDO practitioners attention a new framework (CJOpt) based on the CORBA consisting of distributed Java objects for rapid multidisciplinary analysis.

- To present the services provided by the framework for the MDA of a typical High Speed Civil Transport Flight Vehicle.

Motivation

Continuing interest in multidisciplinary aeroScience applications on high performance computers.

- Distribute legacy codes to the platforms most suitable for their resource needs on a heterogeneous network of computers.

- Reduce the vehicle design time by implementing revolutionary new approaches to improve the design by introducing more accurate information earlier in the process, and ultimately better vehicles.

Framework Definition

A framework for multidisciplinary design optimization is defined as a hardware and software architecture that enables integration, execution, and communication among diverse disciplinary processes.

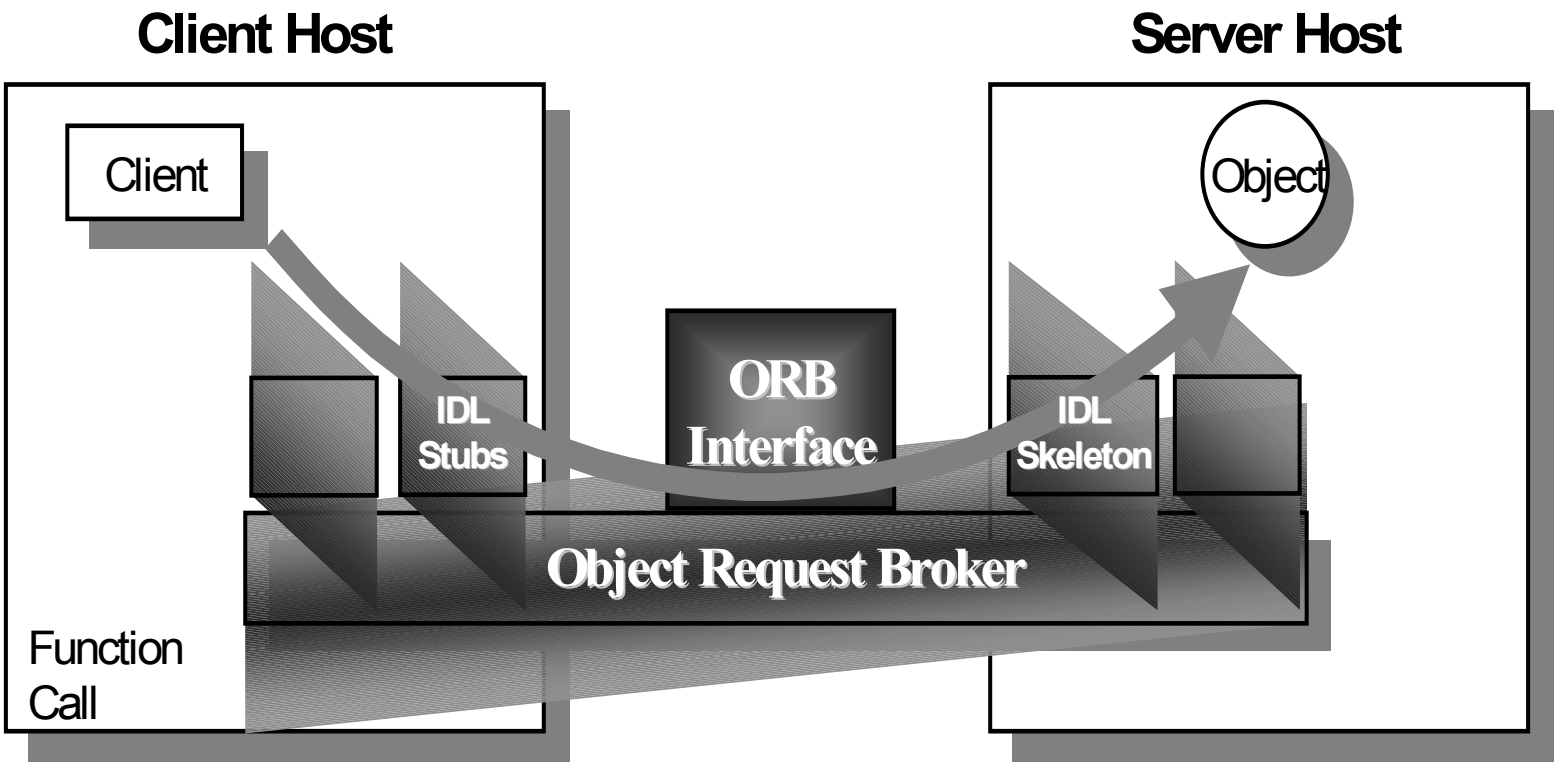
Distributed Object Framework Background

- Suitable general purpose framework for MDO applications
- Communication constructs between the disciplines complex
- Framework tools and analysis tools must be kept separate
- Emerging internet web/client server technologies (CORBA and Java (object oriented) **)**

CORBA Highlights

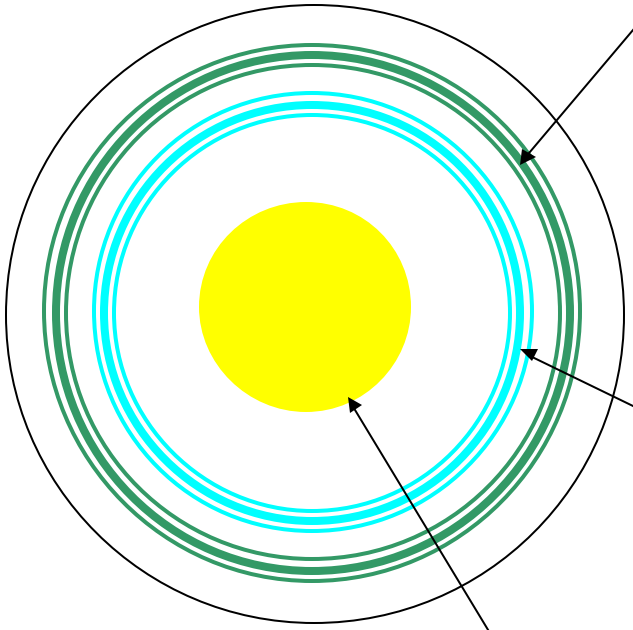
- Object oriented, client-server environment (peer-to-peer).
- Client needs to only know the interface for requesting services of an object.
- Client does not know how the service is provided; language the service is implemented in; what platform the service resides.
- Server does the work of linking up with the services.

Object Request Broker (ORB)



OrbixWeb 3.0

Java Object Content



Client, Server, Interface, Implementation Methods

(with JNI Interface)
(with sharable object interface)

Native Method

C-Wrapper
(with JNI Interface)
↓
Invoke Application Program

Application Program / Legacy Code

- C Language, (Text)/
- FORTRAN, (Text)/
- Executable, (Binary)

Java Object Programming Steps

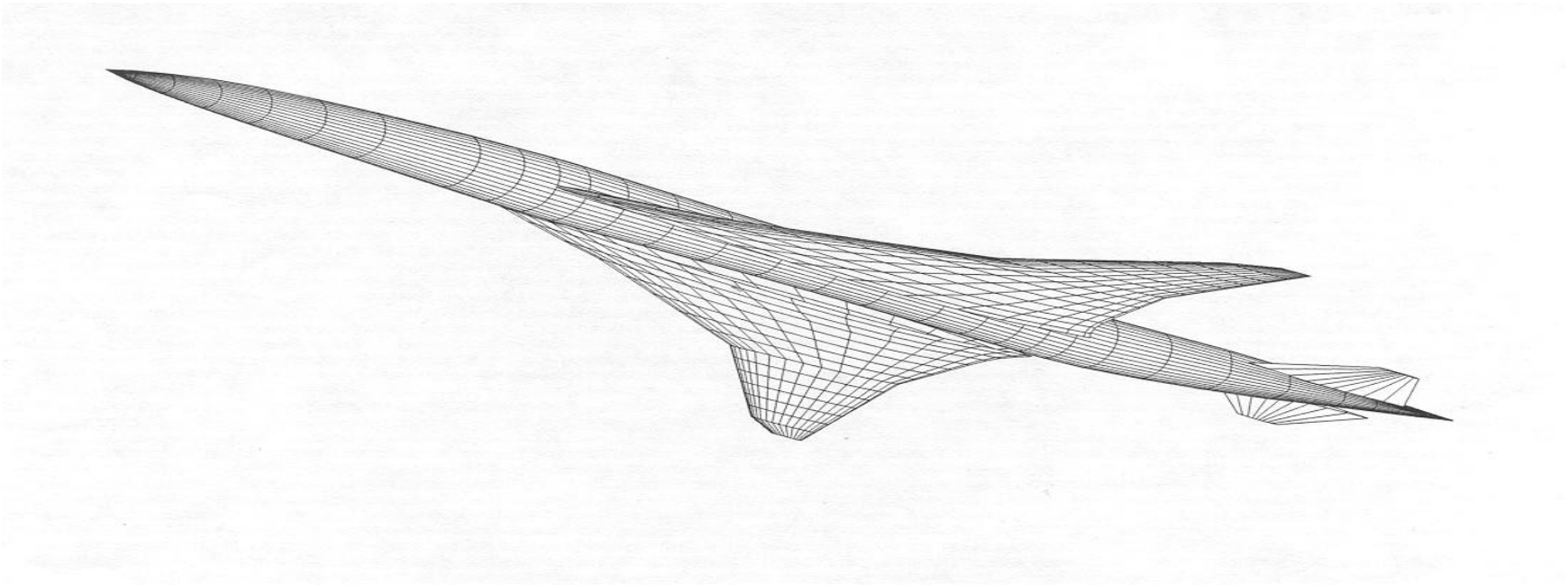
- Write an **interface method** for the object (IDL/Java).
- Write **server method** to register service (ORB/RMI)
- Write **client method** to invoke services (Java).
- Write a **native method**, part of a sharable object.
- Write **implementation method**, which invokes a native method and/or child Object

Data Management

- Central relational database.
- Commercial SQL-compliant.
- Objects use Java's Data Base Connectivity (JDBC).
- User-specific tables for transient data.
- File management information stored in database.

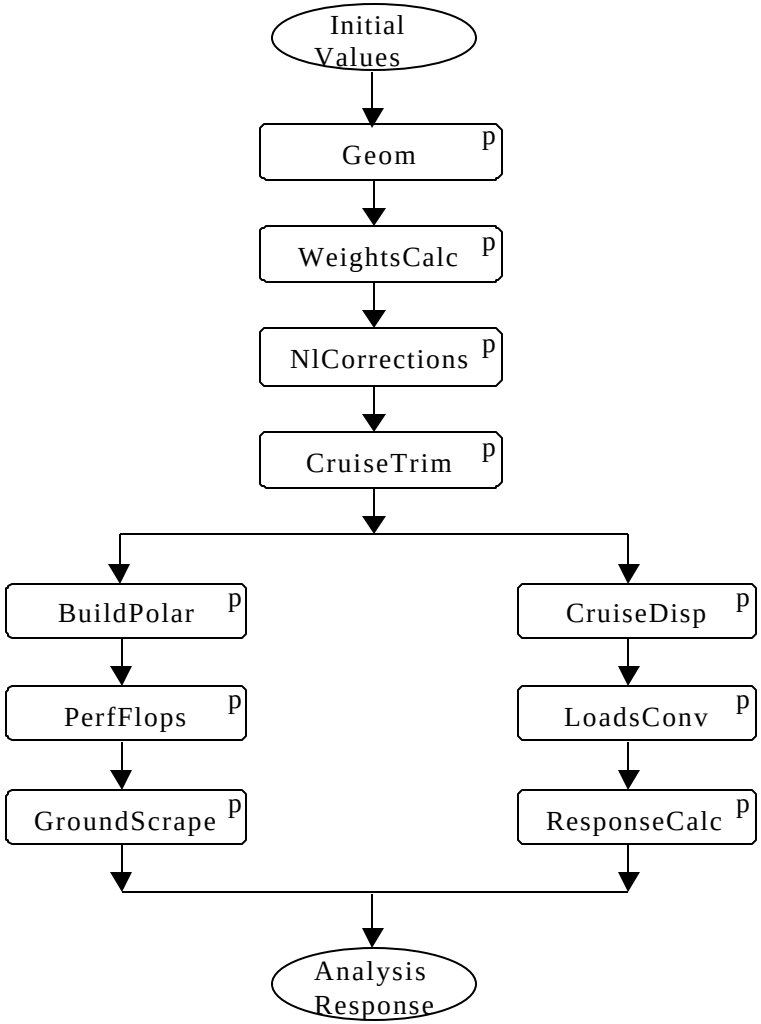
CJOpt/HSCT Multidisciplinary Analysis Implementation

Aerodynamic Surface Grid

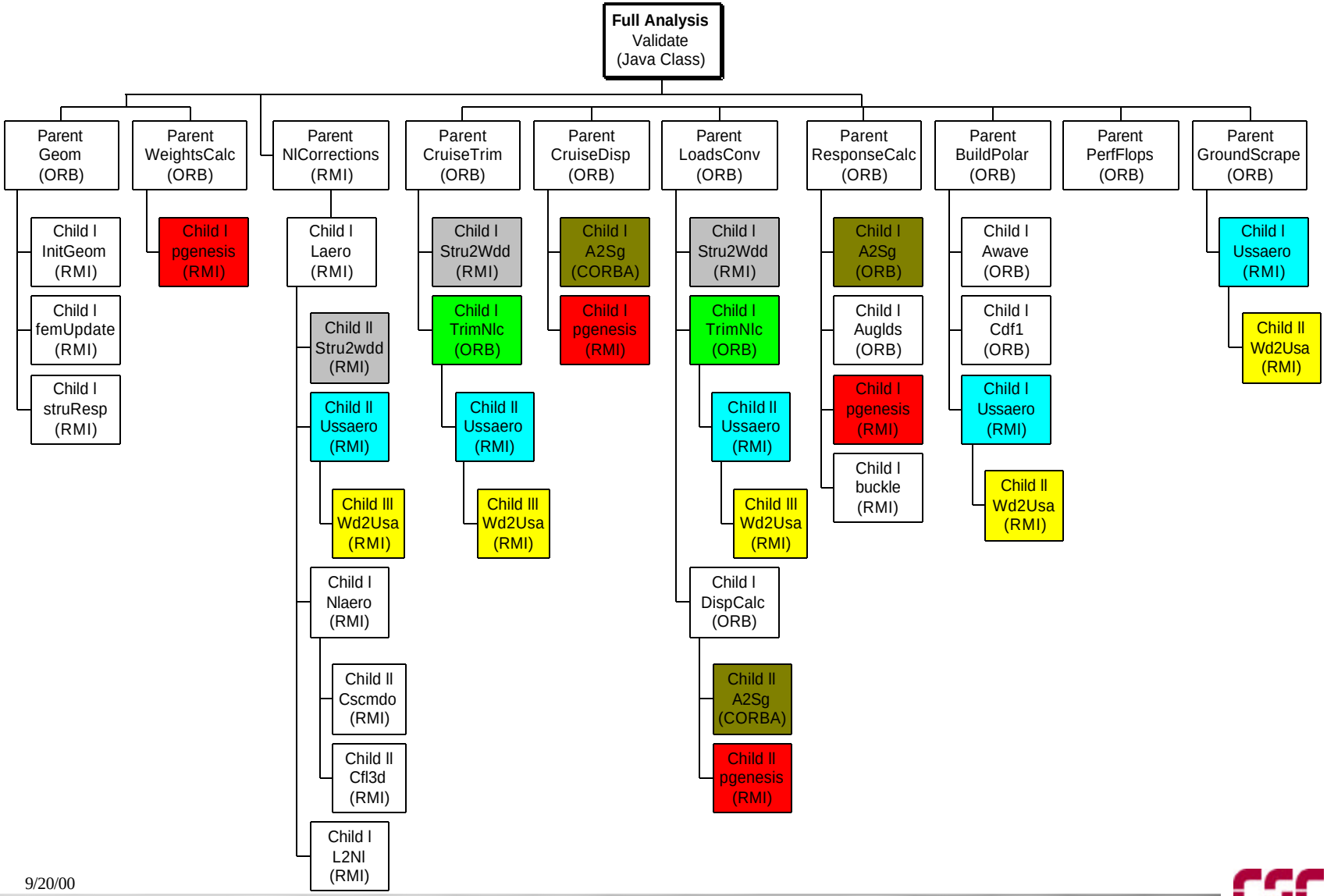


- Independent variables = 271
- Constraints = 31,868
- Analysis processes = 70 without loops ; 1000 with loops
- Data Volume = GBytes
- Major analysis disciplines: Aerodynamics/ Structures/ Performance

Parent Objects for HSCT Analysis



Object Hierarchy and Reusability



Loads Convergence Process

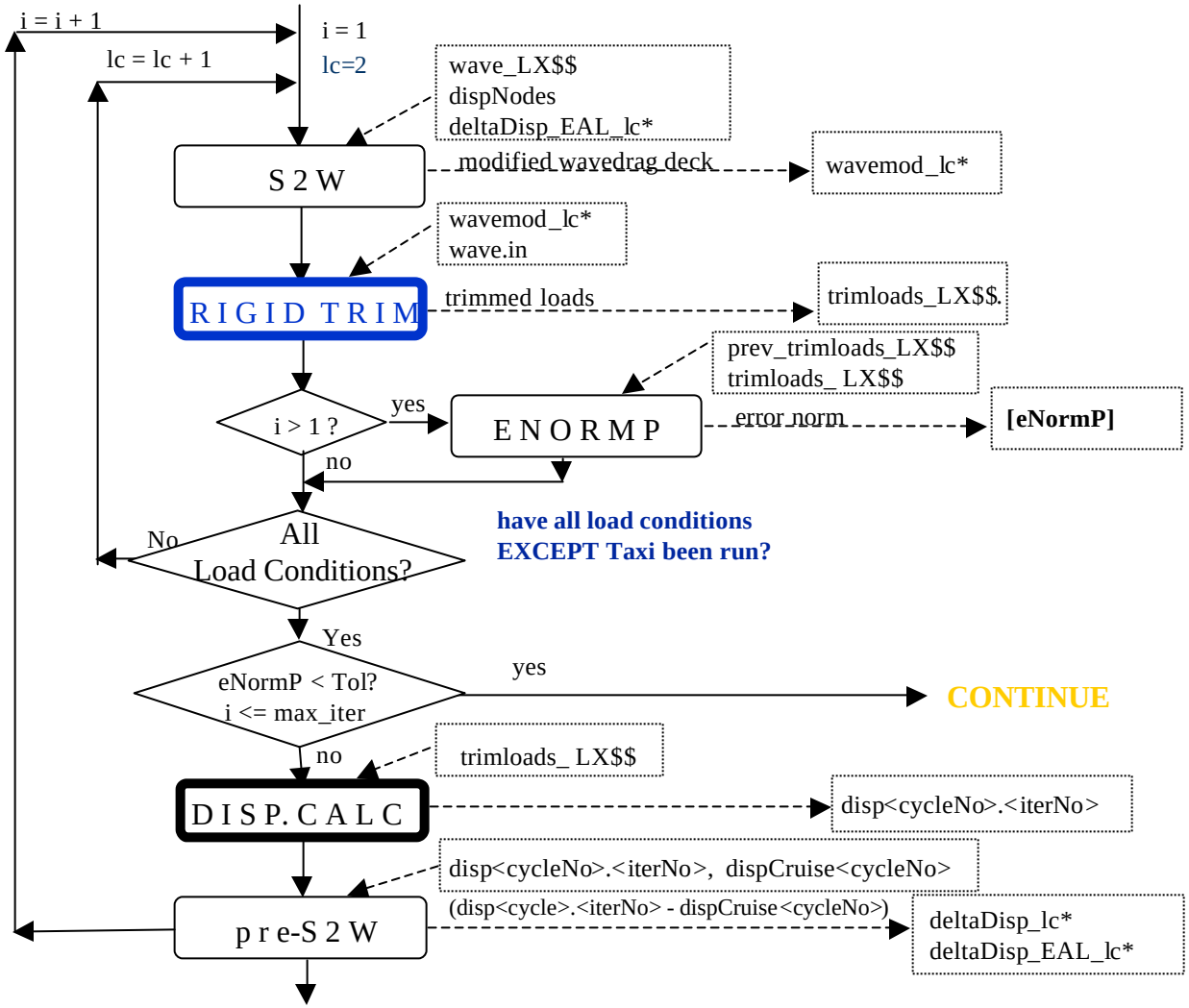
LoadsConv

Provides:

• Consistent aeroelastic structural displacements with the aerodynamic surface loads

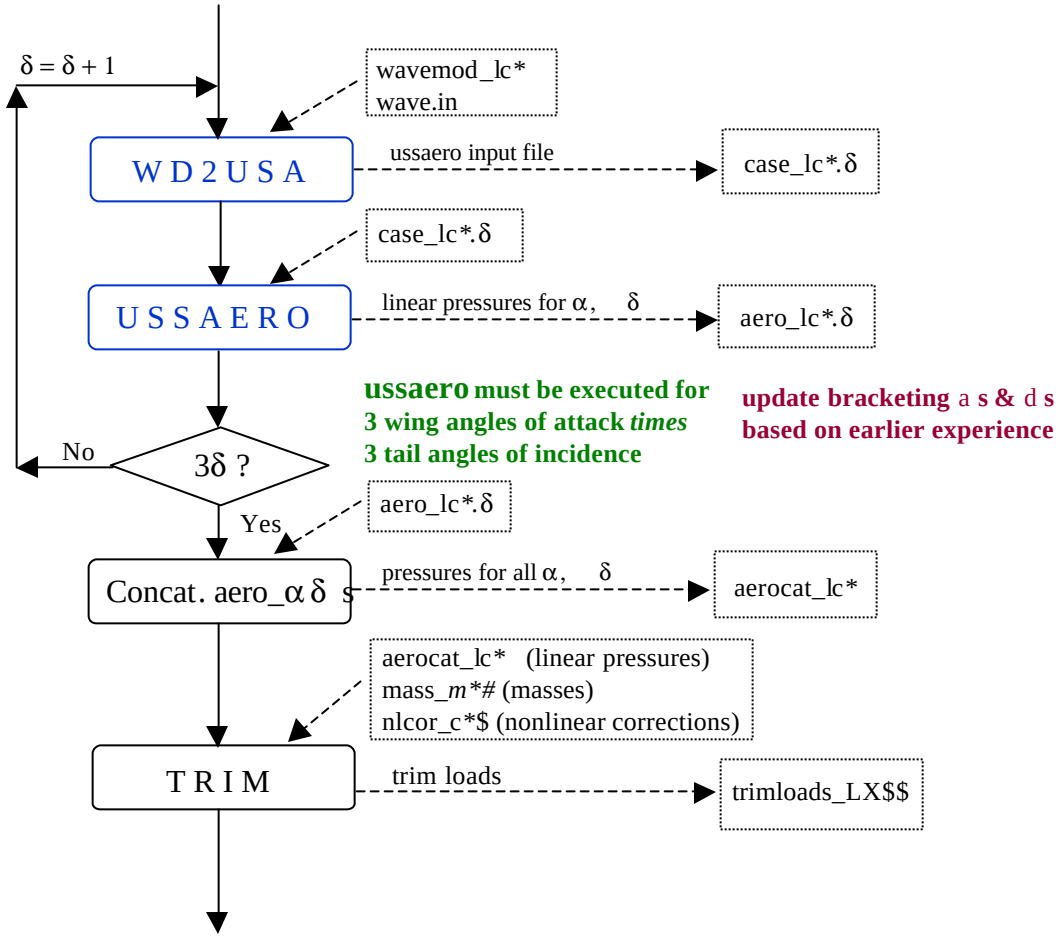
Convergence criteria based on a convergence tolerance or maximum number of iterations

Loads Convergence



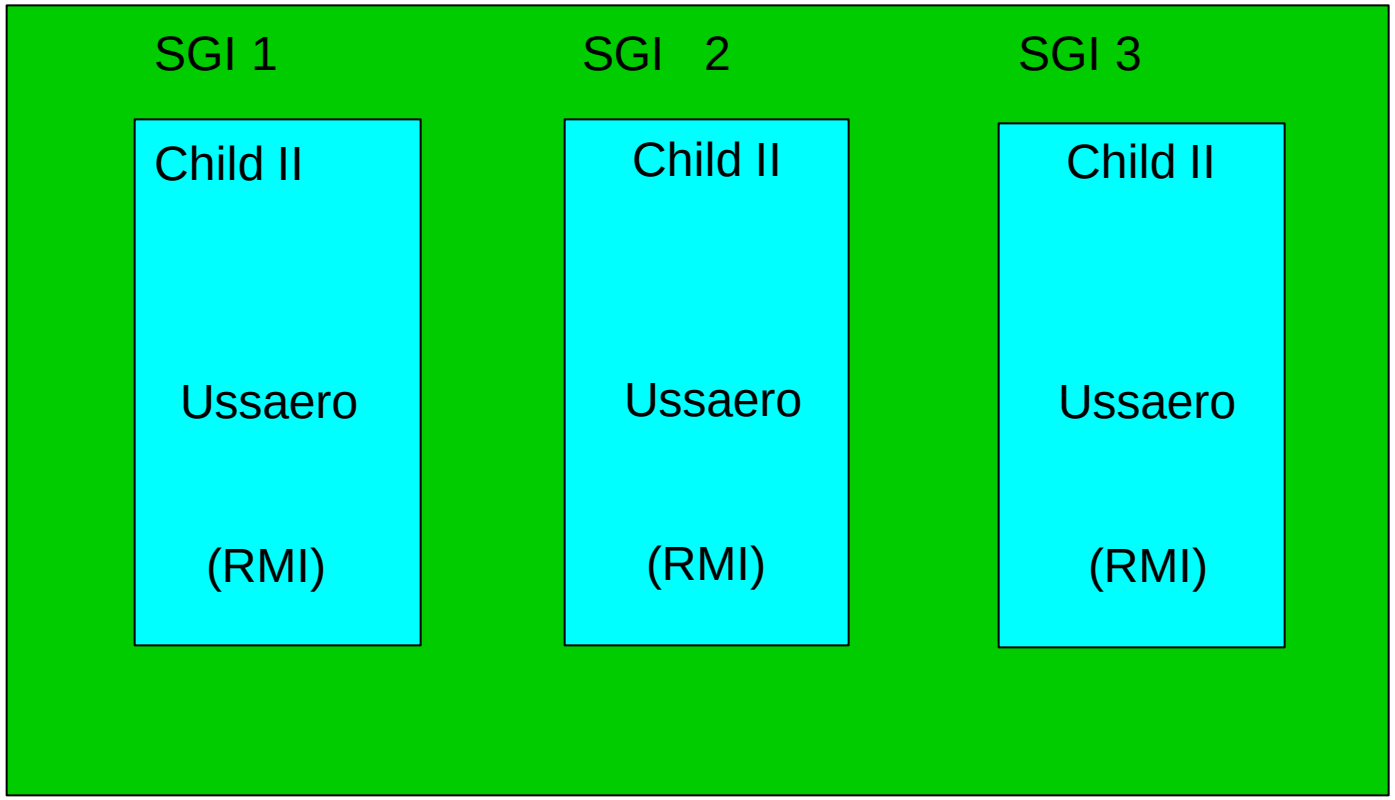
Legend:
lc: load condition
$$: 42, 43, 45, 52, 55, 56
*: 2, 3, 4, 5, 6, 7

TrimNlc



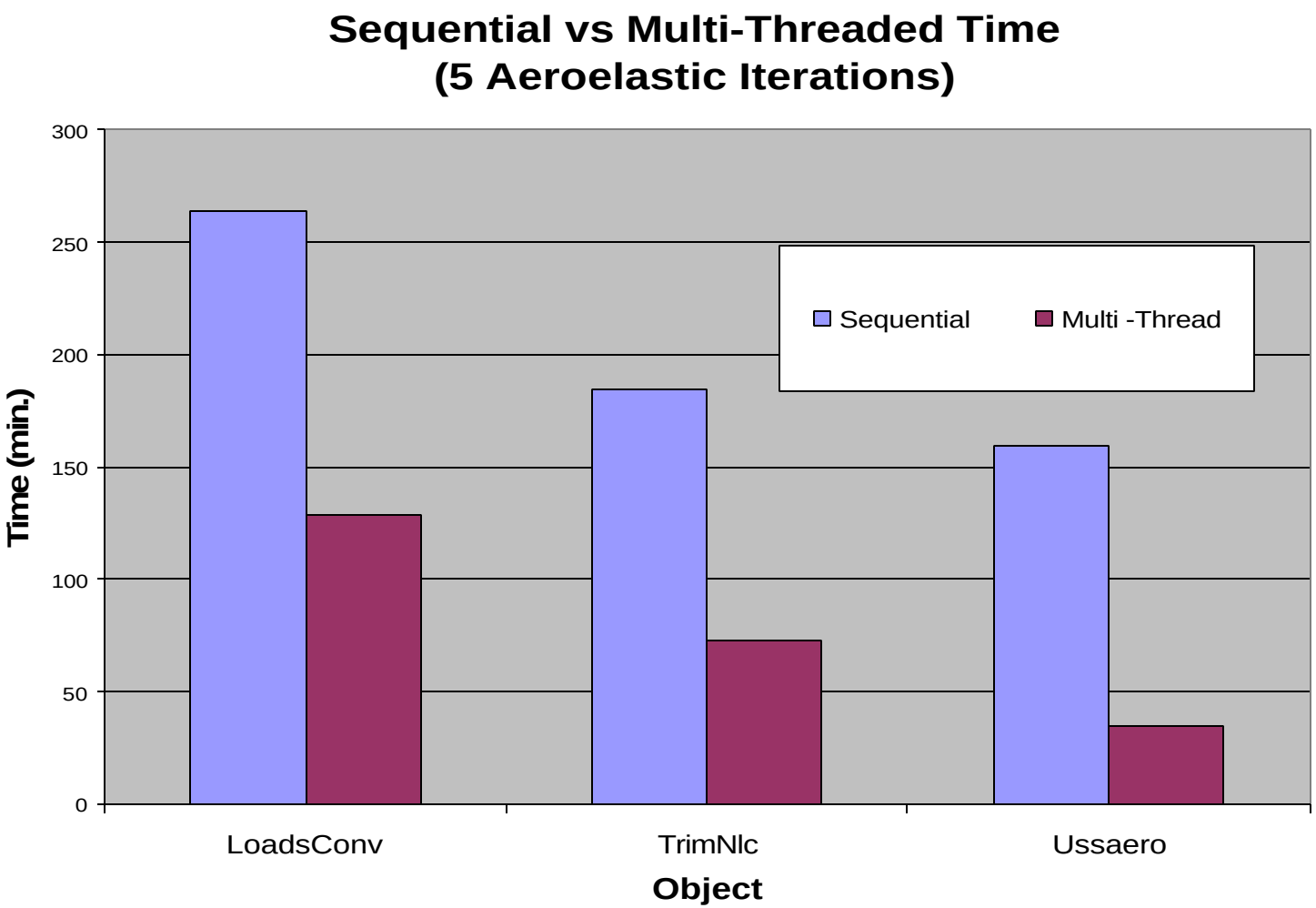
Legend:
*: 1, 2, 3, 4, 5, 6, 7
δ : 1, 2, 3
*#: 03, 05
*$: 01, 02, 03, 04, 05, 06, 07
$$: 79, 42, 43, 45, 52, 55, 56

TrimNlc(ORB) Java Multi-Threading Operations



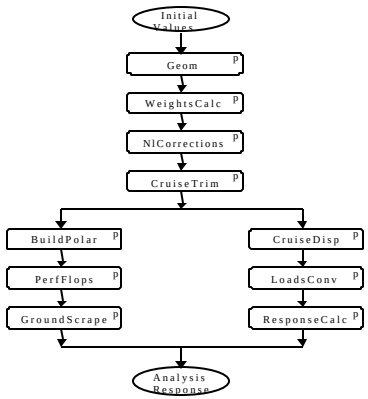
RESULTS

Analysis Clock Times

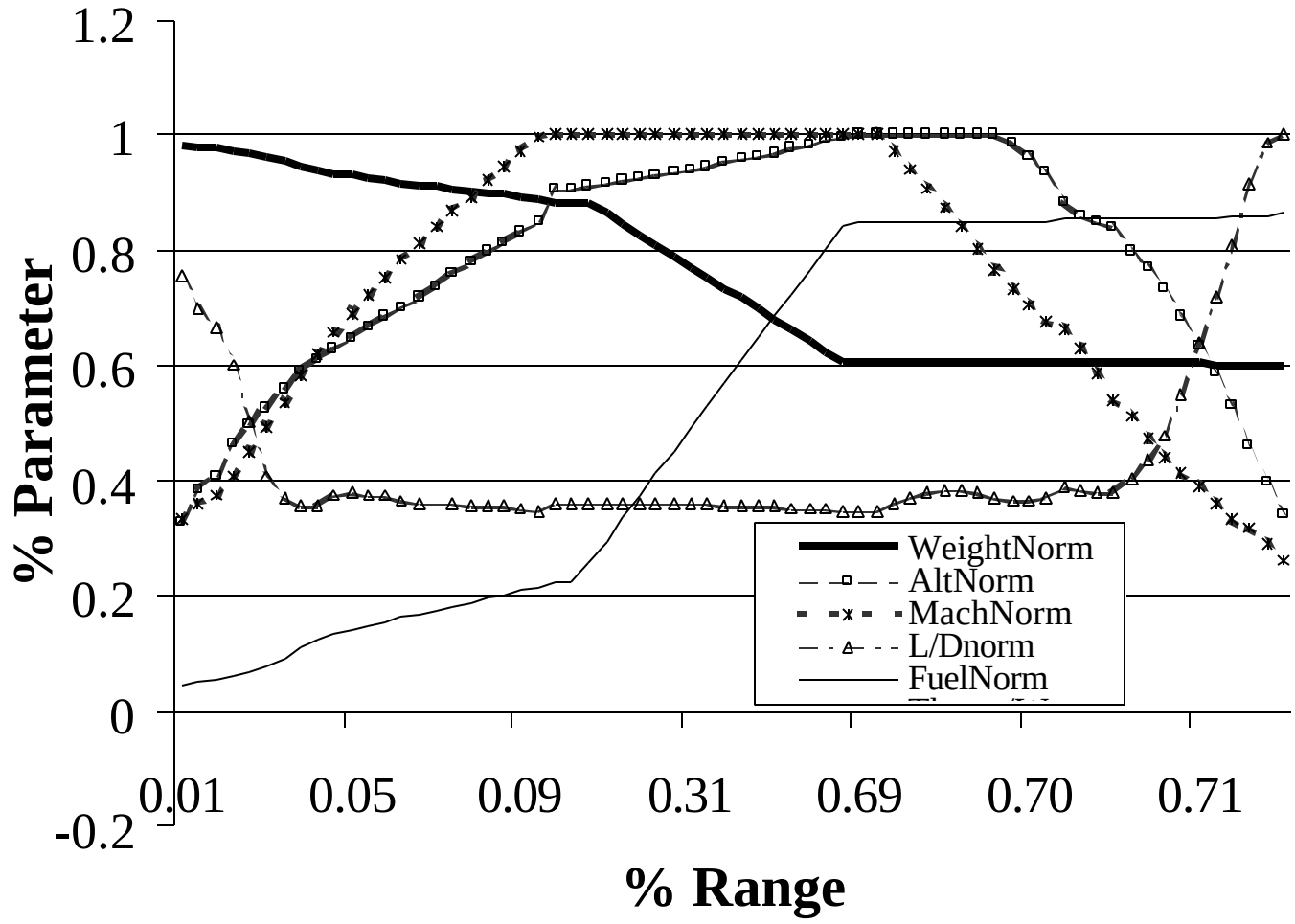


Analysis Clock Times (5 Aeroelastic Iterations)

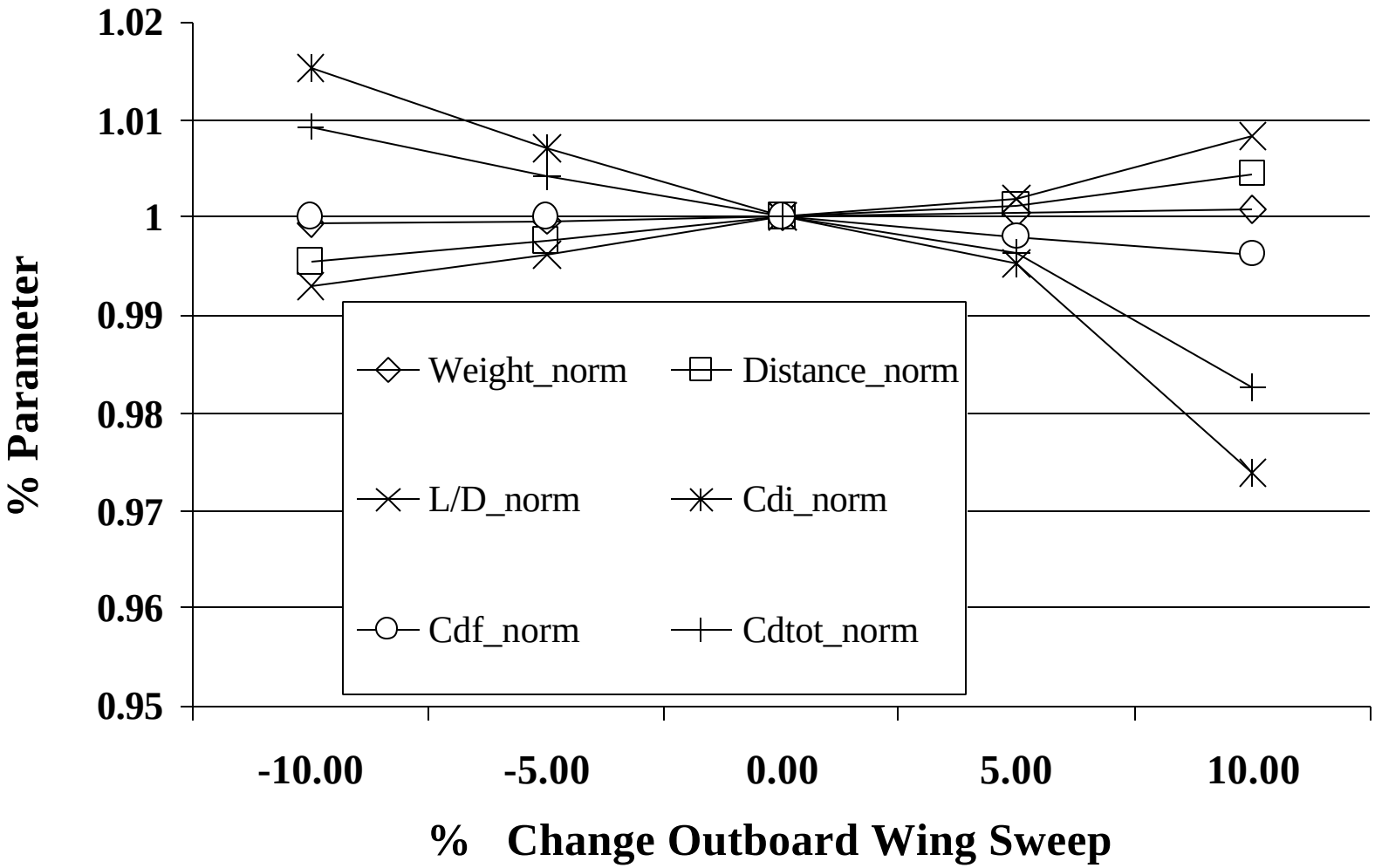
	ClockTime (min.)
Total Analysis Time for top branch operations	14.16
Total Analysis Time for left branch operations	62.00
Total Analysis Time for right branch operations	138.24
Grand Total Analysis Time for sequential operations	(5hr 52min)
Grand Total Analysis Time for multi-threaded operations	2hrs 32min



Performance Metrics



Effect of Outboard Wing Sweep



Conclusions

- HSCT analysis implemented into a new framework (CJOpt) using the Java programming language and is CORBA compliant.
- Framework built up using independent, reusable objects within a heterogeneous computing environment.
- Java's multi-threading used to speed up analysis operations over 50%.
- Uses high fidelity aerodynamic and structural analysis. Performance uses medium fidelity analysis coupled with classic empirical methods
- Parametric study used to verify uncoupled disciplinary analysis trends.
- Mission and performance metrics used to verify interdisciplinary trends.