Framework Requirements for Multidisciplinary Application Development

Andrea Salas and Jim Townsend
NASA Langley Research Center

7th AIAA/USAF/NASA/ISSMO Symposium on
Multidisciplinary Analysis and Optimization
St. Louis, MO, September 2-4, 1998

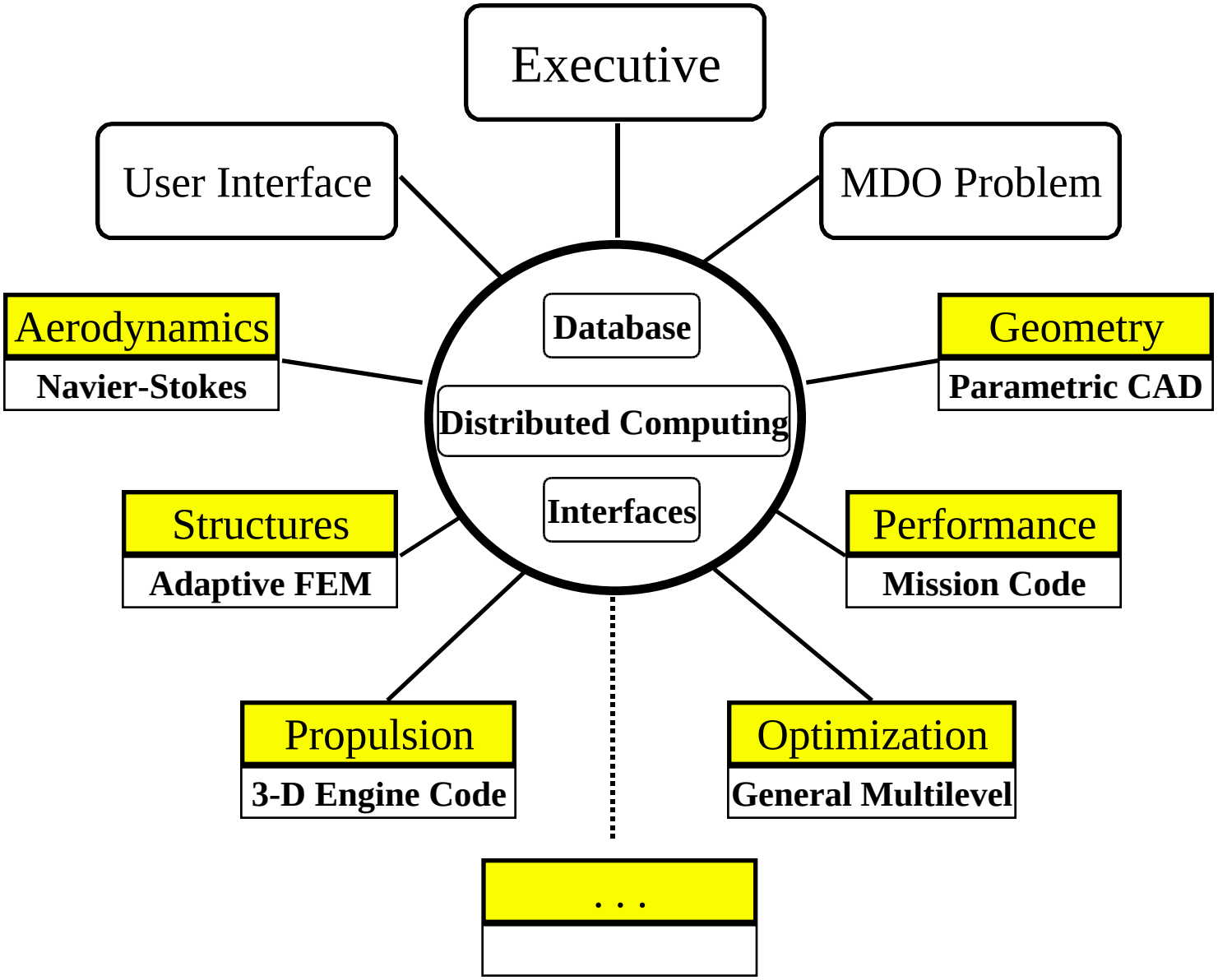
Outline

- Background
- Framework requirements
- Assessment of several frameworks
- Conclusions

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.

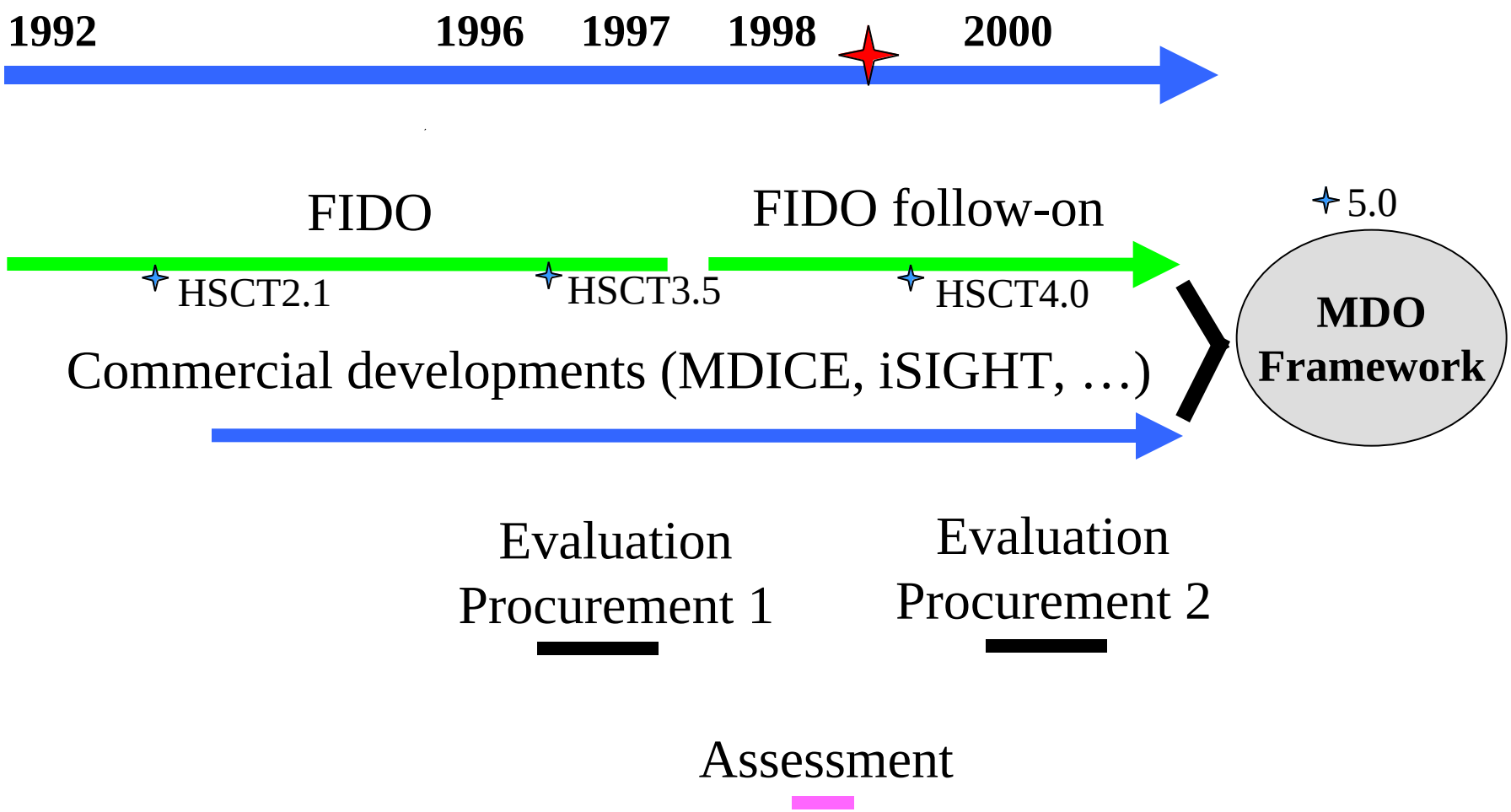
HPCCP HSCT Application Goal



HSCT Applications

Application	**HSCT2.1**	**HSCT3.5**	**HSCT4.0**
Design Variables	5	7	271
Constraints	6	6	31868
Major Codes			
Aerodynamics	Wingdes	ISAAC	CFL3D, USSAERO
Structures	ELAPS	COMET	GENESIS
Performance	Range equation	Range equation	FLOPS
Propulsion	Engine deck	Engine deck	ENG10
Analysis Processes (without looping)	10	20	70
Analysis Control			
Major Loops	Weight Conv., Trim	Weight Conv., Aeroelastic, Trim	Aeroelastic, Trim
Load conditions	2	2	7
Mission conditions	1	1	10
Process (with loops)	O(10)	O(100)	O(1000)
Total time	O(minutes)	O(hours)	O(1 day)
Optimization Cycle			
(ndv+1) #analysis processes	O(100)	O(1000)	O(100,000)
Total time/cyle	O(10 minutes)	O(3 hours)	O(3 days)

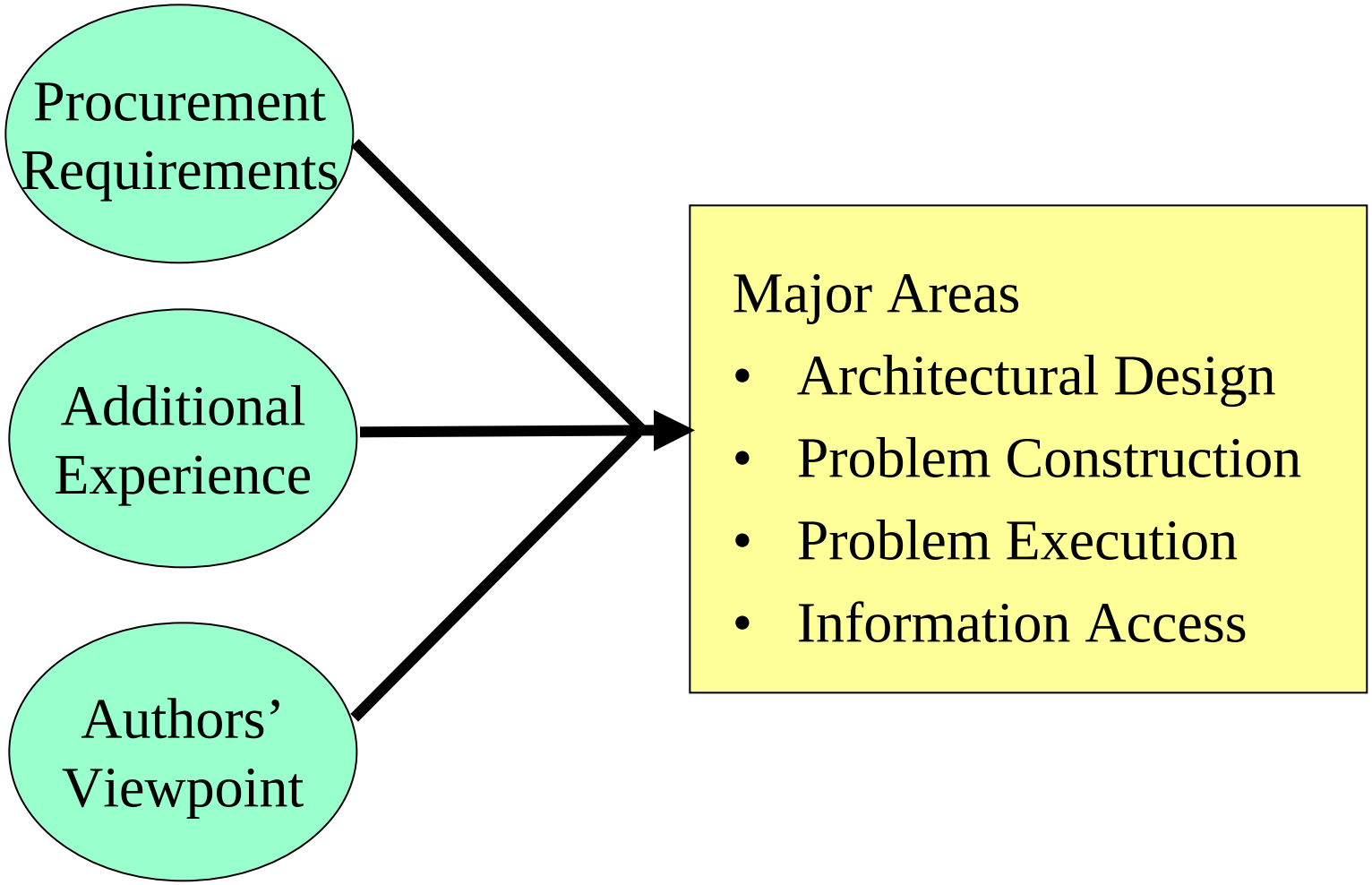
Roadmap to a MDO Framework



First Framework Procurement

- Motivation
  - FIDO limitations
  - resource limitations
  - commercial developments
- Process
  - generated requirements list (WPAFB, NASA LeRC, MDOB, CAS Office, CSC)
  - evaluated frameworks (MDOB, CAS Office, CSC)
  - made decision to select COTS framework
  - posted RFI leading to iSIGHT procurement

MDO Framework Requirements



Requirements - Architectural Design

- Intuitive GUI - ease of use
- Object-Oriented - plug and play
- Extensible - feature addition
- Efficient - low overhead
- Large problems - lots of d.v.’s and constraints
- Collaboration - multiple users
- Standards - preserve investment

Requirements - Problem Construction

- High-level programming - complex branching and iteration
- Reconfigurable - easily change existing problems
- Legacy, proprietary codes - accommodate without change
- Optimization methods - single level, multilevel schemes
- Debugging - multiple distributed processes

Requirements - Problem Execution

- Automation - process execution and data movement
- Parallelism - concurrent processes
- Distribution - workstations, MPPs
- Interactivity - design steering
- Batch mode - multiple problems

Requirements - Information Access

- Database management - data sharing, persistent data
- Visualization - optimization and analysis results
- Monitoring - execution status
- Restart - start from earlier design
- Fault tolerance - computation migration
- Security - access restriction

Current Work in Frameworks

- Industry
  - Engineous Software Inc. (iSIGHT*)
  - LMS Numerical Technologies (LMS Optimus*)
  - Synaps, Inc. (Pointer)
  - Boeing (Access Manager)
  - CFD Research Corporation (MDICE)
  - Phoenix Integration
  - High Technology Corporation (LAWE)
  - TechnoSoft, Inc. (AML)
- Government
  - NASA LaRC (Holist, FIDO*, FIDO Follow-on)
  - NASA Lewis (NPSS)
  - NASA Ames (DARWIN)
  - JPL (MIDAS)
  - Sandia (DAKOTA*, PRE)
  - Argonne (GLOBUS)
- Academia
  - Georgia Tech (IMAGE)
  - UVA (Legion)

* Used for assessment

Assessment Background

- Goal
  - assess usefulness of requirements
  - assess state of current frameworks
- Assessment Accuracy
  - authors' judgement based on literature and private communication
  - FIDO assessment most accurate, followed by iSIGHT

Framework Features - Architectural Design

● Available ◒ Somewhat available ○ Not available – Unknown

Architectural Design	**FIDO**	**ISIGHT V3.1**	**LMS Optimus**	**Dakota**
Intuitive GUI	○	●	●	○
Object-Oriented	○	–	–	●
Extensible	●	●	●	●
Efficient	●	◒	–	–
Large problems	–	–	–	–
Collaboration	◒	○	○	○

Framework Features - Problem Construction

● Available ◒ Somewhat available ○ Not available — Unknown

Problem Construction	**FIDO**	**ISIGHT V3.1**	**LMS Optimus**	**Dakota**
High-level programming	○	◒	◒	○
Reconfigurable	○	◒	◒	◒
Legacy, proprietary codes	◒	●	●	●
Optimization methods	◒	●	●	●
Debugging	◒	○	—	—

Framework Features - Problem Execution

● Available ◒ Somewhat available ○ Not available – Unknown

Problem Execution	**FIDO**	**ISIGHT V3.1**	**LMS Optimus**	**Dakota**
Automation	●	●	●	●
Parallelism	●	○	○	●
Distribution	●	○	○	●
Interactivity	◒	●	○	○
Batch	○	–	–	–

Framework Features - Information Access

● Available ◒ Somewhat available ○ Not available — Unknown

Information Access	**FIDO**	**ISIGHT V3.1**	**LMS Optimus**	**Dakota**
Database management	◒	○	○	○
Visualization	◒	●	●	○
Monitoring	●	◒	○	○
Restart	◒	●	○	◒
Fault tolerance	○	○	○	○
Security	○	○	○	○

Current Work

- iSIGHT
  - distributed computing, debugging
- LMS Optimus
  - parallelism for NLP and DOE
- DAKOTA
  - multilevel parallelism
- FIDO follow-on
  - distributed computation using Java/CORBA
  - Java language: threads, JDBC, Beans, and JNI
  - Mini SQL relational database

Conclusions

- No framework mentioned addresses all requirements
- FIDO follow-on framework tools will be used to implement current HPCCP HSCT application
- Some areas for further research include:
  - problem construction
  - distributed and parallel computing
  - database management
  - debugging
  - interactivity
  - security
- Suggested additions or comments on the requirements are welcome