Evolving a Method to Capture Science Stakeholder Inputs to Optimize Instrument, Payload, and Program Design, P.E. Clark[1], M.L. Rilee[2], S.A. Curtis[1], S. Bailin[4], [1]Catholic University of America (Physics Department), Washington DC 20064 [2]Rilee Associates, Herndon, VA; [3]Knowledge Evolution Inc, Washington DC (Correspondence email: clarkp@cua.edu).

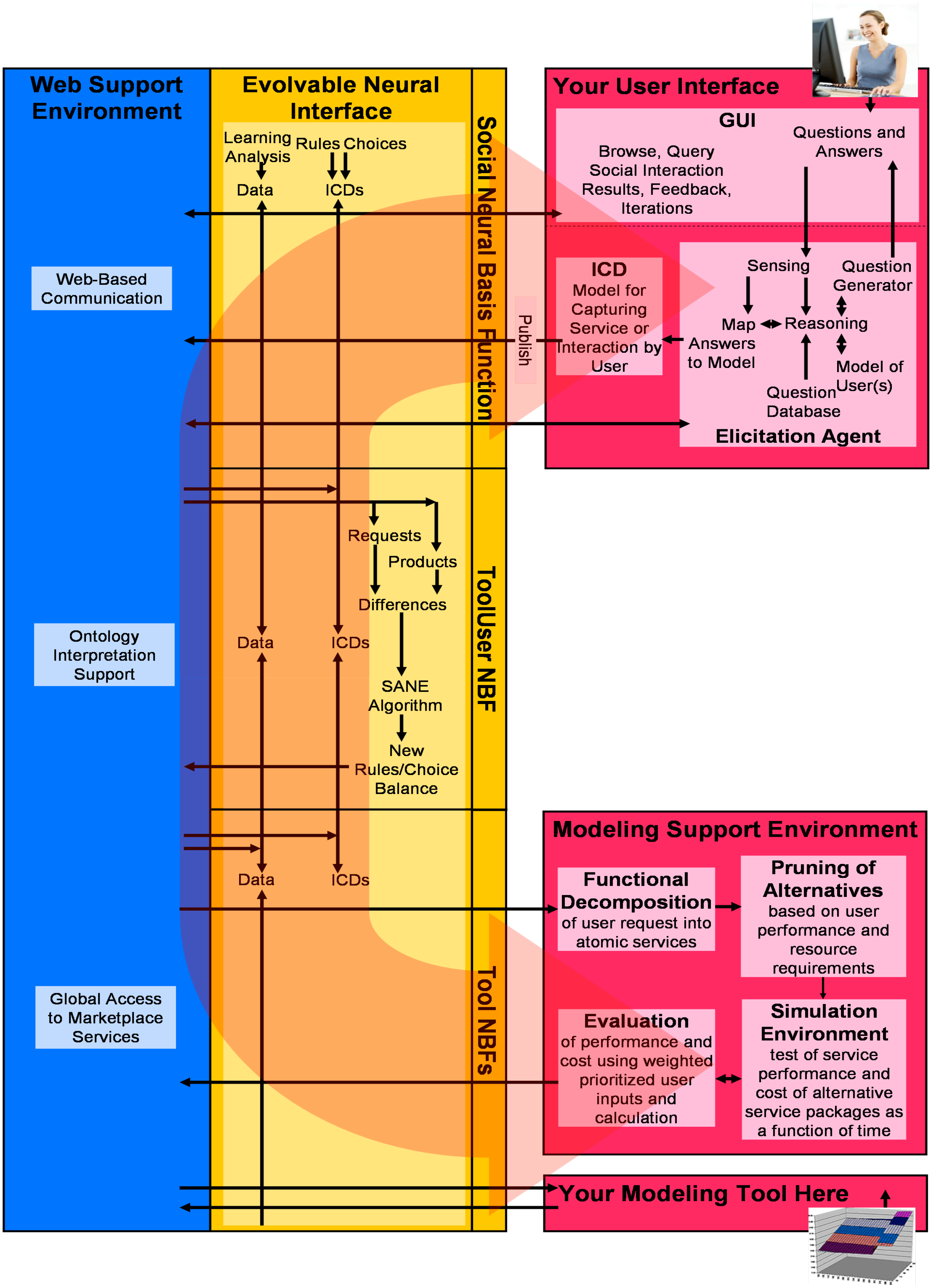
Motivation for Frontier: In support of DARPA System F6 (Future, Fast, Flexible, Fractionated, Free–Flying), Technical Area 1 (Design Toolkit) efforts to understand the requirements for developing a sustainable aerospace environment, we work toward development of Frontier [1, 2], a highly adaptable, stably reconfigurable, web–accessible intelligent decision engine that will be capable of optimizing the design as well as the simulating the operation of complex systems in response to evolving needs and environment.

Adaptable Framework The most innovative aspect of Frontier is what makes it truly unique, capable of absorbing and utilizing lessons learned and thus evolving from a tool to a tool user: an adaptable framework consisting of a decision engine with evolving intelligence based on a genetic algorithm–driven evolving neural interface with an evolving synthetic neural system consisting of neural basis functions for the human (to the Stakeholder GUI) and tool (to the modeling support environment) interfaces and a specially designed stability algorithm to balance rule– and choice–driven inputs originating from either side facilitate the design evaluation and selection process. The adaptable framework will be increasingly capable of dynamic reconfiguration of parameters and rules associated with tools and resources, as well as selection of tools most optimally matched to stakeholder needs through pattern recognition in response to 'lessons learned'. Frontier is built on an open source, web services oriented environment. Through web–based interfaces, it will support distributed, multi–user, concurrent access to resources and tools, including the human and tool interfaces, modeling and development services, databases, simulation, scenario development, analysis, and evaluation.

Human Interface and Inputs Capture: The stakeholder GUI, connected to Frontier via the human interface, although built on NASA heritage, is designed to improve and evolve standard systems engineering practices, by significantly improving 'science' requester/'engineering' developer interaction and encouraging thorough exploration of potential solution space and involvement of the end–user or 'customer' during advanced concept integration, implementation, and operation phases of mission and program lifecycles. This is done by creating a highly adaptable interface guided by and responsive to stakeholder role (e.g., mission or program planner, science data user, component supplier), expertise, time frame (strategic as in program planning, tactical as in mission or instrument development level planning, or operational as in immediate information request). Information capture is systematic and yet tailored for each user. Eventually, this will be done through information capture from natural language. For the time being, this process requires the use of dropdown menus, that can be augmented by users by defining the 'other' categories. Inputs are sought at the level of fidelity appropriate to the users expertise and interest, on 1) activity of interest: type of activity, target, topic, energy region, type of data, sphere and type of operation; 2) performance of critical component or system (spatial, signal, temporal) and subsystem; 3) weight and ranges for a) scope (cost, schedule, risk, technology level); b) performance (number of assets, modularity, scalability, duration and continuity of operation, response time, type and level for external or internal threat, flexibility of use, enviornmental challenges; and c) resource use (mass, volume, bandwidth, power, fuel, time. 4) Inputs on science activity spatial and temporal targeting are also sought. The iterative input process allows multiple assets (e.g., missions for a program, instruments for a payload) to be considered, as well as interaction between request and response sides during evaluation and pruning of potential solutions. The requesting stakeholder is the final arbiter of 'value' during this process.

Modeling Tools: The Modeling Support Environment, connected to Frontier via the tool interface, support the functional decomposition, selection of services (components) for potential design solutions, simulation of performance and pricing (for service requests) for each solution as a function of schedule and time, cost and risk assessment, analysis and evaluation of potential solutions based on stakeholder–driven criteria for value. The interface is designed to accommodate potentially any mission or program development tools from any sources, once function, algorithmic basis, inputs, and outputs are known so the tool can be properly 'wrapped' as a service. We are including several 'external' tools for risk and cost assessment as part of the prototype as well.

References: [1] Clark et al, AIAA Space 2012, Pasadena, CA (in press); [2] Rilee et al, IEEE Aerospace Conference 2012, Big Sky , Montana (in press).



Web Support Environment
Web-Based Communication
Ontology Interpretation Support
Global Access to Marketplace Services
Evolvable Neural Interface
Learning Analysis
Rules Choices
Data
ICDs
Requests
Products
Differences
SANE Algorithm
New Rules/Choice Balance
Social Neural Basis Function
ToolUser NBF
Tool NBFs
Publish
Your User Interface
GUI
Browse, Query Social Interaction Results, Feedback, Iterations
Questions and Answers
ICD Model for Capturing Service or Interaction by User
Sensing
Question Generator
Map Answers to Model
Reasoning
Model of User(s)
Question Database
Elicitation Agent
Modeling Support Environment
Functional Decomposition of user request into atomic services
Pruning of Alternatives based on user performance and resource requirements
Evaluation of performance and cost using weighted prioritized user inputs and calculation
Simulation Environment test of service performance and cost of alternative service packages as a function of time
Your Modeling Tool Here