Habitats – Infrastructure for Knowledge-centric Operations (KO)

John Salasin, Ph.D.
DARPA -- Information Technology Office
jsalasin@darpa.mil

1. New Requirements

The concept of operations is shifting – from large static units with fixed missions to small groups operating independently, with shifting goals. These groups need to coordinate with respect to information, goals and resources. They need to predictably interact with each other and with autonomous, distributed sensors and operational equipment. Unpredictability regarding where, when, how, and with whom we need to interact creates the need for developing open (unbounded) systems that are both flexible and predictable.

We need to provide an infrastructure that lets individuals (and other active objects) in a group, and the groups themselves, dynamically "recruit" each other's capabilities. Since, in warfare, they may control weapons, the recruitment and coordination must be safe and predictable both with respect to "semantics" (e.g., directing the right weapon to fire at the right target) and "non-functional behavior" (e.g., timing, Information Assurance, performance, adherence to doctrine or constraints, freedom from deadlock).

We need to change the assumptions and techniques we use to engineer systems to adapt to the way the systems are changing. Today's architectures and design approaches reflect a tradition of immobile, large processors with relatively static applications. We thus have client-server (rather than peer-to-peer) protocols and static "box and arrow" design tools. Systems are too varied and complex to program all the knowledge that components/agents need for their operation into each component. The context (container) in which they operate must be intelligent enough to help them get the information they need. We term these context-providing containers "habitats". Since habitats are transitory – they are created, destroyed, and change as needed -- they must be dynamic.

For reasons of affordability (and development time) interacting people and objects must be able to (safely and predictably) use commercial infrastructure on a Global Information Grid (GIG). They must have transparent access to services, and the ability to extend services, modify services or servers, and change system configurations when authorized and required (e.g., to improve performance or respond to faults). They must be able to access and use both new and legacy applications. We need to move from hierarchic (e.g., client-server) to peer-to-peer interaction.

The habitats concept moves the locus of control and context sensitivity from the component (or agent) level to the habitat level – for features where the components in a habitat do have common elements of control, function, and accountability. It will provide the ability to easily define and evolve habitats by expressing rules and constraints in system architecture and combining enforcement with adaptability by providing authorized services through controlled extensibility-enabling capabilities.

2. New Technologies

The major challenge posed by this vision is that habitats (or the designated authority (ies) for a habitat) must be able to specify, dynamically modify, and enforce policies regarding access to its own and other services, for both new and legacy components. Since the system and components are dynamic, we need to be able to specify policies in terms of event sequences (rather than hard-wired connections), to monitor events to extract useful information from existing systems, and to represent system status in terms of patterns of events. Habitats (and processes operating within a habitat) need to access, maintain, process, and understand multiple types of information -- automatically.

Proceedings of the Eighth Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems (ECBS '01)
0-7695-1086-8/01 $10.00 © 2001 **IEEE**

Meeting this challenge requires that we evaluate, refine and integrate systems engineering technologies in the following areas:

- Policy/process specification and enforcement – providing the necessary engineering foundation (principles, methods, and technologies) for providing authority and accountability. Habitats need to provide services that delimit the actions that can be conducted by components that are part of the habitat.

- Policy/process coordination/synchronization. Since entities can belong to more than one habitat, they may be subject to the policies of each. Mechanisms need to be put in place to ensure that conflicts are reconciled. Since habitats can (and must) change their policies and behavior over time, the coordination must be dynamic.

- Multiple transaction management schemes, operating concurrently. This is a service that habitats should provide their components, and that is essential to coordination/synchronization.

- Dynamic link management to support knowledge repositories (Uniform Resource Identifier (URI) namespace management for dynamic location transparency including legacy and object components (DCOM, CORBA/Java) and for heterarchical (heterarchical – a structure that combines both hierarchical (e.g., belongs to, is a part of) and set-theoretic (e.g., disjoint, intersecting) concepts) organization of habitat communications)

- Contextualization technologies to enable the content generated by one or more services (or habitats) to influence the customization of other autonomous services, dynamically at run-time as these services operate.

- Constraint-based system representations that efficiently answer the kinds of questions that a habitat (or collection of autonomous cooperating agents) must address during dynamic adaptation, e.g.:
- Are the current configuration choices consistent?
- If the current configuration is inconsistent, what's wrong with it?
- What are the possible options for each system component?
- What conflicts will be produced if a particular option is selected?
- What requirements that will be unsatisfied if an option is selected?

3. Payoffs

This technology will provide a new generation of systems that operate in a network centric world of highly distributed, mobile, computational devices and interact (predictably) with each other and with the real world. These systems will be able to find and predictably recruit the resources/capabilities (e.g., data, programs, computing and communications resources, weapons, and sensors) needed to accomplish computational and operational goals. They will be able to safely use commercial infrastructure (e.g., the WWW) for safety-critical operations, and they will be able to (safely) operate systems where the code isn't stationary (with respect to either location (it moves) or form (it morphs)).

0-7695-1086-8/01 $10.00 © 2001 IEEE