# A Hybrid Human-Computer Autonomous Vehicle Architecture

Michelle Bayouth, Illah R. Nourbakhsh, Chuck E. Thorpe
Robotics Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213
{bayouth, illah, cet}@ri.cmu.edu

## Abstract

The principled design of robot architecture is crucial for the development of low-cost and reliable robotics. Recent advances in the study of robot architecture indicate that *layered* architectures are becoming the standard model throughout the robotics community. In this paper, we describe architecture development for the design of automated highway vehicles. These robots are unique in that they combine the fundamental robot challenges of autonomy and reliability with the less traditional issues of human-computer interaction and division of control. We propose a robot architecture, derived from the standard layered model, that enables a system-wide view of the human-machine autonomous system. We then discuss the hybrid nature of the human-computer interaction scheme, describing several possible human-machine hybrid vehicle controllers.

## 1 Introduction

The study of *robot architecture* plays an important role in the development of a new generation of autonomous robots that are required to meet real-time constraints and exceed particular safety minima. A principled approach to design, called *architecting*, is being applied to robot projects in a wide array of domains, from autonomous spacecraft [5] to distributed software robots [3].

This paper is concerned with an autonomous roadway vehicle, an application domain that introduces additional challenges: the system must interface deliberately with the human element, it must accommodate a variety of incremental deployment options, and it must interact with surrounding autonomous and manually-driven vehicles.

Figure 1 pictures Navlabs 6 through 10, four of several vehicles that are being designed to autonomously navigate the roadway system [6]. These vehicles can demonstrate lane-following, speed-keeping, headway-keeping, and obstacle avoidance.

Figure 1: The Navlab 6 through Navlab 10 vehicles

The fundamental challenge of architecting for Navlab is much the same as any complex, real-time robot system. The system must effect sensor fusion to interpret its inputs, and it must control actuators in a temporally continuous and durative manner. All this is done with two constraints in mind: real-time response and long-term goal achievement. Of course, in the case of automated highway vehicles, safety issues are paramount, as the *raison d'être* for an autonomous roadway vehicle is that it can achieve a higher level of safety than a human driver.

Widely accepted robot architectures and development environments such as RAPS and 3T [4,1] provide a means for architecting solutions to the challenges Navlab faces. We began with the 3T architecture as a starting point. However, a number of challenges unique to the automated roadway vehicle problem required further architectural development.

The radical point of departure for automated vehicle systems is that humans will be present in the vehicles, and their relationship (or non-relationship) to vehicle autonomy must be clearly defined. The control system and the system architecture therefore must represent a *hybrid human-machine system*.

Furthermore, this hybrid architecture does not define a static relationship between human and machine. Speed-of-acceptance and long-term deployment demand that the roadway vehicle only gradually transfer autonomy from human to machine. Even in a more temporally fine-grained sense, the everyday user of an autonomous vehicle will see a continuously shifting boundary between human and machine control. Especially at the beginning of the deployment cycle, there are certain to be vehicles that relieve the human of control responsibilities only in some driving regimes, raising crucial issues regarding transfer of control.

In short, the problem of roadway vehicle autonomy brings together the standard problems of robot architecture design—real-time control, goal-based rationality—and less frequent challenges—human-computer interaction, hybrid human-machine control, incremental deploy-ment, and large-scale robot cooperation.

In this paper, we propose an architecture for an autonomous roadway vehicle. The architecture is striking both because of shared characteristics with standard robotic architectures and because the architectural components are designed and interfaced in order to enable either human or computer authority at every level of control. We describe the architecture and its components in Section 2, then discuss a variety of hybrid deployment scenarios in Section 3. Finally, Section 4 offers some conclusions and describes future work.

## 2 Architecture Description

The high-level objectives for vehicle automation are to increase safety and mobility. These objectives lead to four specific requirements for autonomous vehicle operation. *Reactive safety* demands that a vehicle respond in real time to hazards in the environment [2]. *Proactive safety*, or defensive driving, requires the vehicle to choose actions that minimize future danger. *Roadway-oriented deliberation* requires the vehicle to make rational trajectory choices at the roadway level. Finally, *route-oriented deliberation* demands that the vehicle make rational route-level choices to lead from the point of origin to the destination.

These individual requirements lead to a wide variety of sensor, actuator and intelligence needs. Obviously, satisfying all of the requirements with a one-time market introduction that is low-cost, user-friendly and fully-autonomous is not realistic. Issues of technical design are perhaps even superseded by deployment issues concerning the introduction of automation to the public and by liability concerns. As a result, highway autonomy can only proceed via an incremental deployment of the automation.

Figure 2 is a schematic of the architecture we propose to meet these challenges. This architecture is an instance of a *layered architecture*, in which system control is divided between multiple modules, or layers, based upon representational resolution, both geometric and temporal [1].

A layered architecture is useful in this situation for several reasons. At development-time, layers provide natural boundaries for incremental implementation and testing. Functions which must meet similar reactivity and robustness criteria will naturally define a layer, and so careful testing of that single layer will be straightforward, given that the layer's connections to its neighbors are well defined in the architecture specification.

More importantly, layering enables well-defined, mixed human-machine control. The notion of being able to insert a human into any architectural layer has profound implications for the evolutionary deployment of automation. The layers chosen must have clean interfaces which can connect either to a human or to another automated layer. A low-level layer must be capable of safely operating when severed from higher-level layers due to faults or failures. Note that this requirement applies, not simply to the automated component of an architecture, but to the entire human-machine system: if part of the automation is compromised, the *entire system*, which may or may not include the driver, should be functionally capable of continuing operation of the vehicle at a most basic level of safety.

A hybrid human-machine architecture is thus one in which the functions and interfaces of each architectural layer are clearly defined such that either a human or a machine can operate at each level, and one in which safety-related functions are embedded at low levels of control. The 3T architecture is particularly amenable to this functional delineation, and the three layers we propose bear some resemblance to 3T.

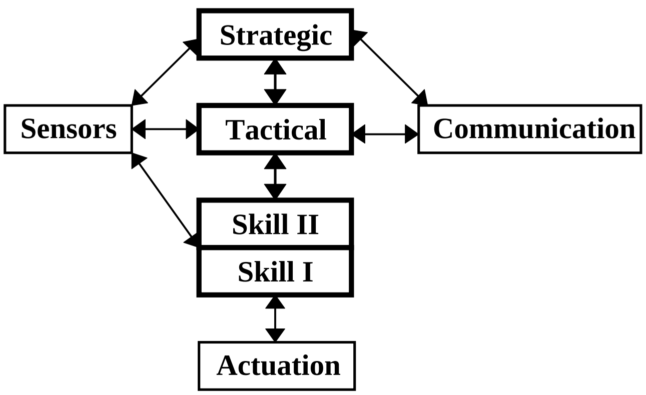


Figure 2: The automated vehicle architecture

We characterize each of these three layers in terms of *function* and *purview*. By *function*, we mean the particular system goals that a layer satisfies through combinations of control techniques. By *purview*, we mean those divisions in spatial and temporal interest that are required in order to satisfy the layer's functional goals.

This functional definition allows us to avoid a common trap in architecture design. Robot architectures commonly make premature representational decisions, frequently imposing an explicit and symbolic form of *knowledge* at the higher layers and an implicit, *reactive* encoding at lower layers. As has been demonstrated by Rosenschein & Kaelbling [8], functional intelligence can be achieved through either implicit or explicit, symbolic or non-symbolic techniques, and so these decisions are premature.

### 2.1 *Skill Layer*

The Skill Layer is responsible for both *reactive safety* and the robot's lowest level of control. The Skill Layer has the following competencies: maintenance of lateral control within a lane, maintenance of headway, speed control, acceleration, deceleration, braking, changing lanes, and informing the driver when a situation beyond its capabilities arises. These competencies, although cognitively limited, form the basis from which the vehicle can react to obstacles and other vehicles in the roadway

In regard to purview, this layer is concerned temporally with actions and reactions on a fine-grained and short time horizon (e.g. within the next three seconds). Spatially, the Skill Layer is interested in the acute location and the activities of vehicles and obstacles in its immediate surroundings (approximately 100 meters to the front and back, and 7 meters to each side).

The Skill Layer does *not* reason about series of actions. Rather, it considers single maneuvers as reactions to hazards in the roadway. Similarly, the Skill Layer does not install or modify goals, such as desired speed or headway distances. Such goals are determined at a higher level of control, be that the Tactical Layer or the highway infrastructure, or by default values originating from the system designers.

A quality of this architecture is that The Skill Layer, being ultimately responsible for the safety of the system, is the only layer capable of generating actuator commands. Indeed, the interfaces to higher levels of control may be viewed simply as channels of *advice*; the Skill Layer is a final arbiter and cannot be subsumed by other layers' goals.

We further subdivide the Skill Layer into two sublayers: Skill I and Skill II. Skill I is responsible for basic longitudinal and lateral control (e.g. acceleration, deceleration, headway control, lane-following). Skill I represents a set of competencies that have been engineered and are available currently on vehicles such as Navlab 9. Skill II adds an extra dimension of lateral control: lane changing and merging. These are technically more challenging functionalities that are currently in development, naturally falling in a different category from the more technically mature functions of Skill I.

### 2.2 *Tactical Layer*

The Tactical Layer is responsible for *proactive safety* and *roadway-oriented deliberation.* The Tactical Layer can project the traffic scene forward in time, predicting the motions and future positions of surrounding vehicles. Furthermore, the Tactical Layer may, via vehicle-to-vehicle and infrastructure-to-vehicle communication, receive information concerning the intentions and future positions of other vehicles on the roadway.

In our prototype implementation, this layer will generate probabilistic descriptions of future world states, allowing it to maximize the likelihood of goal achievement while meeting probabilistic safety constraints. For example, the Tactical Layer can control recommend vehicle speeds and headway separations that maximize the number of extreme swerve options available to the vehicle in the event of an emergency.

Roadway-oriented deliberative planning uses these same tactical reasoning abilities to choose between various paths in order to achieve system goals. For example, if the autonomous vehicle is trailing a slow-moving truck in the right lane, the Tactical Layer can plan a series of maneuvers (*change-lanes-left, accelerate, change-lanes-right*) in order to achieve the desired speed. Of course, if the vehicle's exit is near, the Tactical Layer will remain in the right lane because the lane change may decrease its probability of success below an acceptable value.

The Tactical Layer's ability to take advantage of communication is *opportunistic*. In that surrounding vehicles may or may not have communication links, this capability is not relied upon but rather utilized when available. Negotiations between vehicles enable additional efficiencies and safety guarantees which are particularly useful. For example, when merging into mainline traffic from an entrance ramp, knowledge that a gap will be maintained because of negotiated assurances from other automated vehicles increases the safety of the entire system, and may enable a merge to occur into a smaller gap (and thereby afford greater vehicle throughput).

With respect to purview, the Tactical Layer is concerned temporally with *sequences* and *conditional sets* of actions and reactions within the next tens of seconds. Spatially, it is concerned with the vehicles within several hundred meters meters front and back, and several lanes to each side.

### 2.3 *Strategic Layer*

The Strategic Layer is responsible for the high-level functions of route planning and guidance. It makes use of information from sources in the infrastructure to determine the route which most efficiently satisfies the vehicle's given goals. Indeed, precursors to such technology are already being deployed in several overseas markets. Note that the Strategic Layer's functionality exceeds that of a simple, interactive map. It evaluates the position of the vehicle on the existing roadways on an ongoing basis, informing the Tactical Layer when subgoals must be specified. Furthermore, the Strategic Layer can reason about the global efficiency of alternative, strategic plans, choosing new plans and modifying old plans at run-time in order to achieve greater success as roadway conditions change.

The purview of the Strategic Layer extends throughout the roadway system relevant to the task at hand. By the same token, the Strategic Layer uses a more granular form of representation. For instance, specifics such as the local traffic scene surrounding the vehicle will not be represented at the strategic level of detail.

### 2.4 *Interfaces*

In order for this hybrid human-machine architecture to be successful, the layers should be as independent as possible. The complexities of the system should reside within the layers themselves, and the interfaces should be well-defined and simple [7]. The following discussion provides a framework for the kinds of information that pass between the layers shown in Figure 2.

An important global view concerning the interfaces is that they are meant to communicate *information* as well as subgoals. The interfaces we describe are not just slaved control interfaces;

rather, they serve as a means for neighboring layers to *inform* one-another, providing information gain and thereby aiding in the decision-making process at each layer.

For instance, the Tactical Layer not only presents the Skill Layer with basic goals, such as desired speed and headway distance; it can also inform the Skill Layer about a neighboring vehicle's intention to change lanes, thus modifying the forward projection of that neighboring vehicle, as constructed by the Skill Layer.

Furthermore, the Tactical Layer can achieve its higher-level goals by issuing recommendations to the Skill Layer that cause long-term changes in the vehicle's local scene—changes that cause the vehicle to better match its goals. For instance, using a sequence of granular actions such as *accelerate*, *change lanes left*, *change lanes right*, the Tactical Layer may cause the vehicle to pull ahead of a slow-moving vehicle and thus achieve the desired roadway speed more successfully. By the same token, information can flow from the Skill Layers back up to the Tactical Layer, indicating failures of high-level recommendations as well as reasons for those failures.

The relationship between the Strategic Layer and Tactical Layer is similar, with the difference being primarily one of granularity. The Strategic Layer informs the Tactical Layer with subgoals that allow the long term goals to be achieved. The Tactical Layer, in turn, responds to the Strategic Layer, either communicating success or identifying failures in the achievement of thosesubgoals.

We will leave further detail concerning the representational and translation issues involved with inter-layer communication for a longer publication. Instead, we turn our attention to the most fascinating aspect of this architecture: the hybrid human-machine nature of the automated vehicle.

## 3 Hybrid Human-Machine Scenarios

Autonomous vehicle systems are unique in robot architectures in that the human is formally part of the autonomous system. Because of both technical and non-technical (social, psychological, legal) issues, it is difficult to predict which layers will be machine-controlled first. At the technical level, of course, a highest level strategic system is already in operation on many vehicles. A lowest level Skill layer, at the Skill I level, has been successfully tested [6].

Nevertheless, the non-technical issues as well as the unsolved engineering challenges lying in wait at the Skill II and Tactical Layer demand that the system architecture be capable of incorporating human control at *any* architectural layer. Furthermore, devising an architecture that is amenable to varying levels of human control enables the system designers to implement automation at the level dictated by the design circumstance, leaving the vehicle open to evolution as those circumstances change.

Refer to Table 1, which summarizes a number of viable control schemes with varying degrees of human control. The options shown are surprisingly diverse, proposing human control in the middle layers (scenarios 5 & 6) as well as the opposite (scenario 7).

Table 1: Deployment options indicating human control (h) and machine control (m) at various layers.

| **Layer** | **1** | **2** | **3** | **4** | **5** | **6** | **7** |
|---|---|---|---|---|---|---|---|
| Strategic | h | h | h | m | m | m | h |
| Tactical | h | h | m | h | h | h | m |
| Skill II | h | m | m | h | h | m | h |
| Skill I | m | m | m | h | m | m | h |

Consider scenario 2, in which the Skill Layer, both Skill I and Skill II, is operated under machine control and the human serves at both the Tactical and Strategic Layers. If the driver desires to pass a slow vehicle, he indicates this to the Skill Layer by specifying a subgoal to change lanes left. In turn, the Skill Layer does so when it is safe to execute the maneuver. If no vehicles are in the new lane, the Skill Layer will accelerate to the desired speed as set by the Tactical Layer, and continue to operate in this state until the driver issues a change-lanes-right goal. Thus, the driver is responsible for staying cognizant of the trip plan, and for maneuvering this "push button" vehicle through traffic. This implementation could conceivably improve system safety by executing maneuvers under machine control that meet or exceed specified safety standards.

Next, consider scenario 7, an unusual implementation in which the Tactical Layer is machine-controlled and the Skill and Strategic Layers are human-controlled. In this case, the Tactical Layer would provide the human with information to facilitate vehicle control. For instance, when merging on a freeway, a head-up display could indicate the optimal gap to the driver based on projections for gap openings and formal negotiations with nearby vehicles.

A more traditional approach is captured by scenario 4, in which the machine is responsible for Strategic-level planning. Given a high-level goal specification by the driver, the machine ascertains the trip origin and destination, desired time of departure and arrival, and preferred routes. Using a communication link with the infrastructure, it determines the current traffic conditions and the historic traffic trends for the potential routes, and selects an optimal route and departure time. The Strategic Layer gives road-by-road instructions to the driver, providing congestion-specific instructions such as *begin merging right* in order to ensure that the driver reaches appropriate exits.

The few examples we have provided only begin to shed light on the various operational modes that are possible when the human-machine vehicle is viewed as a single autonomous system. The important lesson is that the architecture must consider the human-machine interface carefully to enable seamless and safe human-machine control.

## 4 Conclusions and Future Directions

We have identified a version of the standard layered architecture that is amenable to the problem of automated vehicle systems. We are fortunate enough to have real-world vehicles on-hand that achieve Skill I and partial Skill II levels of automation. In coming months, we will demonstrate the "push-button car" of scenario 2, then go on to implement basic machine control at the Tactical Layer.

An important issue that will arise is that, initially, the vehicle will only be capable of automatic control at the Tactical Layer in light traffic. Therefore, the issue of run-time transfer of control between human and computer will play an important role in our implementation. A hopeful note is that, in this case, passage of control will only take place at a relatively high level: the Tactical Layer, leaving seamless and continuous low-level control to the machine at the Skill Layer. This facilitates transfer of control immensely by removing hard real-time demands from the transfer process.

Automated vehicle design is a unique problem not only because of the human-computer interaction element but also because safety guarantees are of paramount importance. A formal robot architecture, and in particular a layered architecture with its well-defined control hierarchy between layers, facilitates the process of formally evaluating system safety.

Of course, these issues span further than only automated vehicles. We hope that this paper summarizes the basic problem of architecting automated vehicles clearly so that a productive discourse on this subject can take place in the greater robotics community.

## References


[1] Bonasso, R. Peter, Firby, R. J., Gat, E., Kortenkamp, D., Miller, D. & Slack, M. "Experiences with an Architecture for Intelligent, Reactive Agents." To appear in *Journal of Experimental and Theoretical Artificial Intelligence*, 9(1), 1997.

[2] Drummond, M., Swanson, K., Bresina, J. & Levinson, R. "Reaction-First Search." In *Proceedings, IJCAI-93*. Chambéry, France. 1993.

[3] Etzioni, O., Hanks, S., Jiang, T., Karp, R.M., Madani, O. & Waarts, O. "Efficient information gathering on the Internet." In *Proceedings, 37th Annual Symposium on Foundations of Computer Science*. Los Alamitos, CA. 1996.

[4] Firby, J. "An Investigation into Reactive Planning in Complex Domains." In *Proceedings of the Sixth National Conference on Artificial Intelligence*. AAAI. 1987.

[5] Gat, E. "News from the Trenches: An Overview of Unmanned Spacecraft for AI Researchers." In *Workshop Notes, Planning with Incomplete Information for Robot Problems*. AAAI-96 Spring Symposium. Stanford, CA. 1996.

[6] Jochem, T., Pomerleau, D., Kumar, B. & Armstrong, J. "PANS: A Portable Navigation Platform." *IEEE Symposium on Intelligent Vehicles*. Detroit, MI. 1995.

[7] Rechtin, E. *Systems Architecting: Creating & Building Complex Systems*. Prentice Hall: New Jersey. 1991.

[8] Rosenschein, S. J. & Kaelbling, L.P. "A situated view of representation and control." *Artificial Intelligence*. 73:149-173. 1995.