MP180941
MITRE PRODUCT

Human-Machine Teaming Systems Engineering Guide

Bedford, MA

Patricia McDermott, Cindy Dominguez, Nicholas Kasdaglis, Matthew Ryan, and Isabel Trahan
MITRE

Alexander Nelson
Air Force Research Laboratory

December 11, 2018

Contents

1 Overview: Tailoring Requirements for Your System

1.1 Why and How to Use This Guide

This guide was written to help system developers design autonomy and automation that works in partnership with the human operator. With increased complexity, automation, AI, and autonomy of systems playing out in robotics, unmanned vehicles, cognitive assistance, and more across the MITRE sponsor space, it is critically important to provide research-based guidance on human-machine teaming. Autonomy should act seamlessly within a human operator's workflow, aiding performance by alerting them about behavior that deviates from normal, suggesting alternative solutions that they may not have considered, autonomously reorganizing priorities in response to their changing goals, or other collaborative activities. There is a wealth of published guidance on how to support human-machine teaming (HMT), but that guidance is rarely used to design operational systems. To bridge this gap between researchers and developers, MITRE surveyed and analyzed the existing literature to develop a set of General HMT Requirements. These are evidence-based requirements that address the span of autonomy from autonomous vehicles to cognitive assistants. Because not all General HMT Requirements are relevant to each specific type of system that employs automation and autonomy, and also because each system employs these technologies towards different goals, it is necessary to adapt the requirements for specific systems. This guide describes the system engineering methods for tailoring the General HMT Requirements into specific requirements for your system.

Figure 1. This project makes HMT research accessible to developers in a usable form

Look for boxes with this icon for tips and lessons learned to improve your results.

1.2 HMT Background

Note: We recognize that technology-based assistance can span the spectrum from automated features to full autonomy. In this guide, we use the terms autonomy, automation, and automated partner flexibly. Choose the term that best suits your application.

The General HMT Requirements fall into 10 themes or leverage points for supporting HMT [23], with each theme representing multiple citations from the applied research literature reviewed:

- *Observability*. Observability provides appropriate transparency into what an automated partner is doing relative to task progress. Observability supports shared understanding of the problem to be solved and progress toward goals. Autonomy is observable when it provides the needed level of insight into the information sources and the calculations that contribute to the recommendations and predictions provided to a human partner.

- *Predictability*. Predictability captures the transparency of future intentions, states, and activities of the automation partner, as well as a projection of what the future situation will be if current trajectories are continued. It is an extension of observability that is forward looking.

- *Directing Attention*. Autonomy and automation must be able to direct the attention of human partners to critical problem features, cues, indications, and warnings. It should communicate proactively when information or trends become relevant, such as when obstacles to meeting goals are encountered.

- *Exploring the Solution Space*. Exploring the Solution Space is supported when partners can leverage multiple views, knowledge bases, and solutions to jointly understand effective ways to respond to the problem space. Automation should be able to rapidly generate multiple distinct courses of action and give people ways to rapidly compare those solutions with respect to consequences and outcomes. Both humans and technology should be able to broaden or constrain the solution considerations to shift perspectives.

- *Adaptability*. Adaptability enables partners to recognize and adapt fluidly to unexpected characteristics of a situation. Autonomy should have multiple options to recognize an unexpected situation and change course, or suggest course changes, to address evolving, dynamic situations.

- *Directability*. Directability is supported when humans are able to easily direct and redirect an automated partner's resources, activities, and priorities. Humans will have expertise or insights that the automated system will not. Humans are ultimately accountable for system performance, which means they must be able to stop processes, change course, toggle between levels of autonomy, or override and manually control automation when necessary.

- *Calibrated Trust*. Calibrated Trust is supported when technology provides indications enabling human operators to understand when and how much to trust an automated partner in context. Users need to understand when automation can be relied on, when more oversight is needed, and when performance is degraded or unacceptable. To help users calibrate trust, the autonomy should provide information sources, limits of technology effectiveness, and credibility or uncertainty associated with those sources.

- *Common Ground*. Achieving Common Ground means that pertinent beliefs, assumptions, and intentions are shared among team members. Common Ground needs to be constantly and actively updated and maintained so that human and autonomous team members can maintain a shared picture of what's happening in the world and engage in backup behavior to support each other.

- *Information Presentation*. This category captures recommendations on how to present information to support simplicity and understandability. Human users of technology should be able to view and interact with mission-relevant information and patterns of information in order to understand the implications of the data.

- *Design Process*. This theme captures guidance on using cognitive engineering methods to enable effective HMT. It includes guidance on how to plan and use established methods for effective user engagement (understanding mission-related needs, leveraging expertise to generate design concepts, and validating designs in scenario-based contexts).

The leverage points are further organized into four categories depicted in the Framework for HMT below.

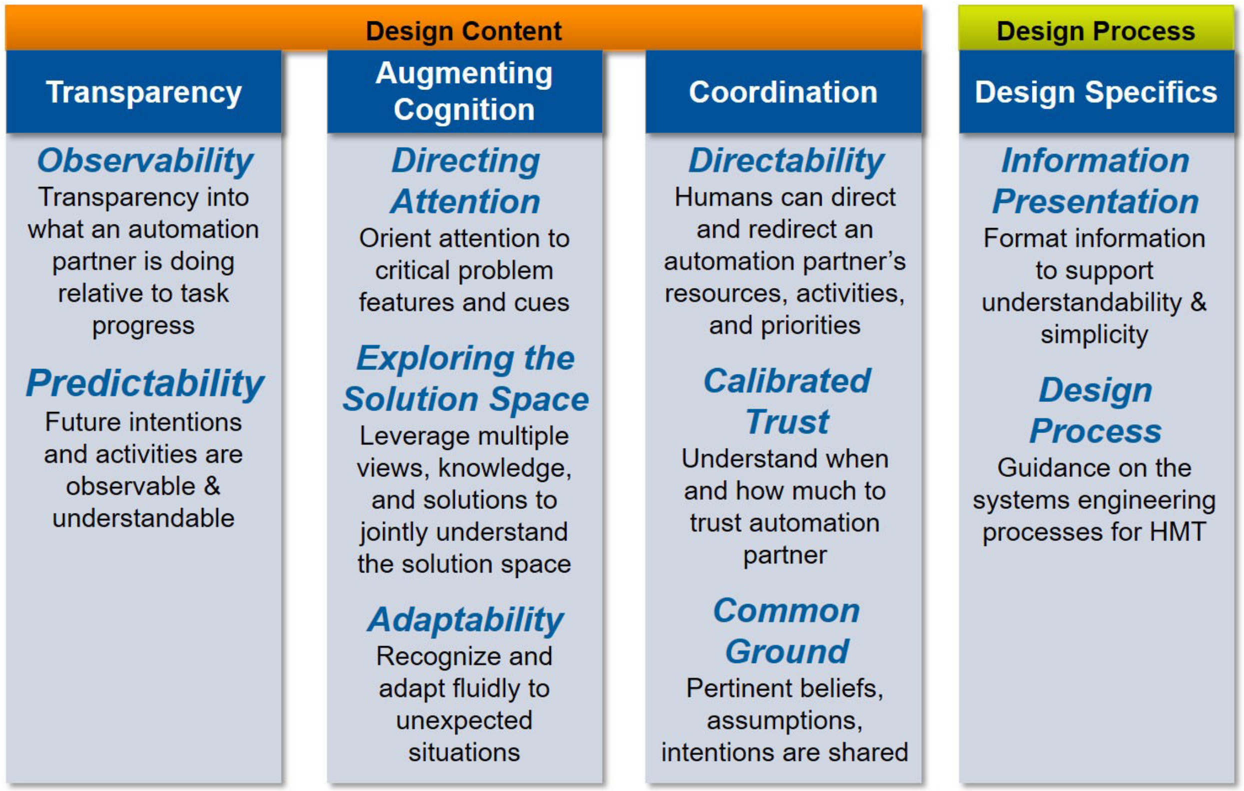


Figure 2. Ten leverage points organized into a Framework for HMT

1.3 Overview of HMT System Engineering Method

Cognitive task analysis is a set of methods to systematically define the decision requirements and cognitive processes used by operators or decision makers as they perform their work. Cognitive task analysis is used to identify, link, and prioritize the critical decisions people make and to understand their associated cues, strategies, goals, challenges, and considerations. A newly developed cognitive task analysis method, the HMT Knowledge Audit, is not intended to elicit or analyze the full decision landscape; rather, it focuses on unpacking the specific knowledge elements related to human-machine teaming. The questions are driven by the leverage points identified in the HMT Framework and the need to collect information to flesh out the associated requirements.

The systems engineering process of tailoring requirements for your specific system has three major steps, as shown in figure 3. A chapter is devoted to each of these steps.

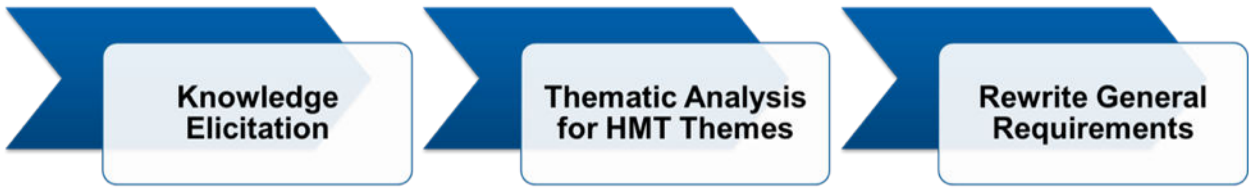


Figure 3. Three major steps, elaborated in chapters 2, 3, and 4

The Interview Guide is designed to be flexible. The questions are widely applicable, from unmanned aerial systems to a commander's digital assistant. Use the HMT Knowledge Audit as a starting point and tailor it to your needs. You are unlikely to ask every question in every interview. Likewise, the questions do not need to be asked in the order listed. Rather, they should follow the flow of the conversation as naturally as possible.

> Tailor the interview guide to your needs.
> Stay flexible during the interview.

1.4 Who Should Use This Guide

This HMT Systems Engineering Guide is written for human factors professionals or cognitive engineers who have experience with knowledge elicitation and user engagements. This guide does not include exhaustive guidance on how to conduct cognitive task analyses. For more information on conducting cognitive task analyses see the Resource List in appendix D.[1] The following sections contain advice on getting started and high-level tips for conducting interviews.

Preparing for interviews:

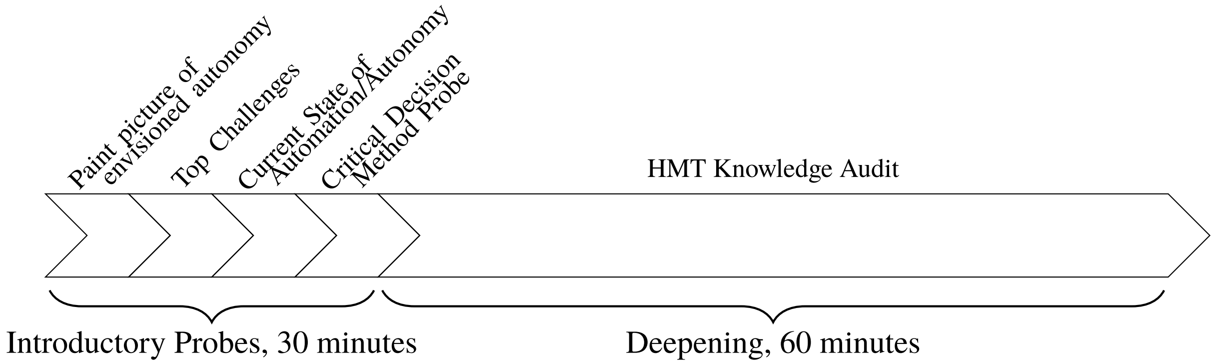
- Develop a general understanding of the domain, including the vocabulary and general job functions. This might be done by talking to program managers, software developers, or former operational people associated with the program, and by analyzing documents (e.g., manuals, technical reports, documented workflows, or use cases).
- Identify Subject Matter Experts (SMEs) who have recent experience in performing the job or who have experience training others to do the job. You may need to talk to SMEs in multiple roles (e.g., operator, crew chief, and maintainer).
- It is important to interview a sufficient number of SMEs so that the data is not skewed by the experience of one or two people. Interview a minimum of 3-5 SMEs per role. For example, if an autonomous system is going to be used by a pilot, co-pilot, and mission manager, you would need to interview at least three SMEs for each role for a total of nine interviews. These are the bare minimum numbers; a larger number of interviews per role is recommended.

Tips for conducting interviews:

[1]The book *Working Minds* by Crandall, Klein and Hoffman provides an excellent overview.

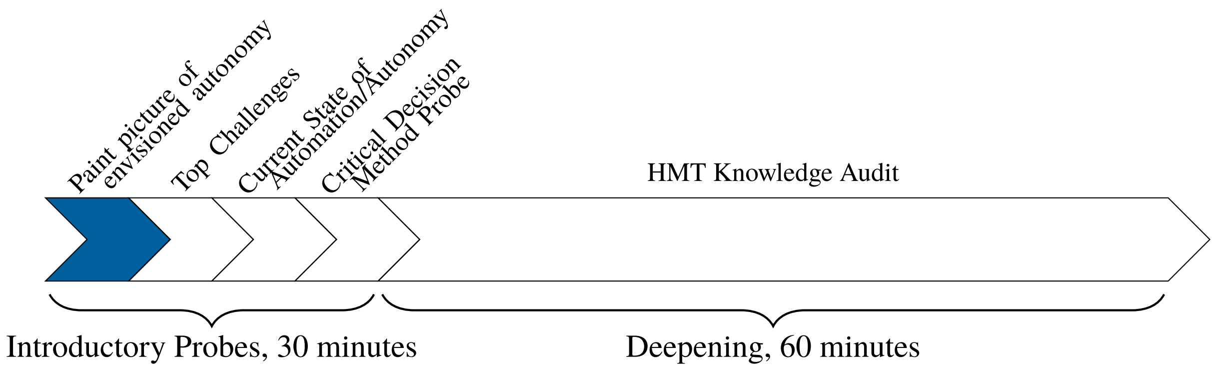
- When capturing notes during the interview, it is helpful to record the questions as well as the answers. This makes it easier to compile notes and to compare responses to the same question across interviewees. You can do this by printing interview guide copies with lots of space to write, or you can type directly into the interview guide during the interview.
- Think twice about audio recording the interviews because transcribing the interview can take a considerable amount of time [13]. A primary note taker who can capture key content from each question in real time is more time efficient.
- Convey that you are excited to be interviewing the SME because they have experiences that you can learn from.
- LISTEN. Focus on what the SME is saying, not your next question.
- Stay in control of the interview. The SME should do most of the talking, but you may need to steer them back on course, for example, by saying that you can come back to that point or story, but you'd like to hear more about your key question.
- SMEs may talk in vague terms, because they don't want to bore you with the details (e.g., my job is to enforce the schedule). Ask follow-up questions, such as:
  - Would you give me an example?
  - Can you elaborate on that?
  - Can you tell me about a time when ...?
  - What makes that part of the job difficult?

2 Interviews with Subject Matter Experts



The general structure of the interview is shown in the figure above. The interview guide assumes 90-minute interviews. The Introductory portion should take approximately 30 minutes; however, in early interviews, more time might be spent here understanding the mission and systems. The remaining 60 minutes should be spent deepening your understanding of the human-automation partnership. The interviews could easily last 2 hours, given the opportunity. If you only have 60 minutes with each SME, we recommend eliminating or truncating the Current State of Autonomy and the Critical Decision Probe. The questions in the HMT Knowledge Audit, or the Deepening portion, can be asked in any order. If the interview time is truncated, you could prioritize the subcategories to focus on the most relevant. Relevance could be determined during pilot interviews or through discussions with the system developers and program managers. For example, before conducting interviews with Army Battalion staff regarding an automated tool for mission planning and execution, we identified six (of 10) leverage points, or HMT themes, that were most applicable to the functionality of the envisioned tool, and focused our interviews on those.

2.1 Paint a Picture of the Envisioned Autonomy



In this portion of the interview, explain to the interviewee that you are working on a new system or an updated system with advanced autonomy or automation. Craft a brief, standard summary of the vision of intended functionality, human interaction, and capabilities of this system to use for this explanation. The SMEs are experienced with an existing system that may not include autonomy or automation. The goal is to describe ideas for the envisioned system in enough detail that the interviewee can understand the concept and imagine how they could use it.

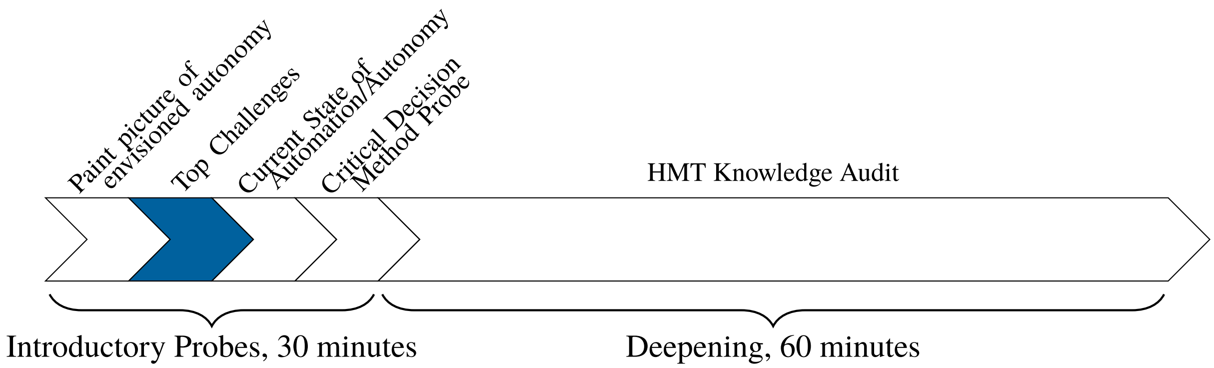
Example description for Army Battalion Tool for Planning and Battle Management:

> *The tool has highly automated features to assist with mission planning and execution. The planner lays out the plan on a digital map and the software can read the plan in order to simulate it and predict outcomes. Planners can use automated features to show how the placement of forces and weapons impact the ability to engage the enemy, taking into account factors like range and terrain. The tool can direct attention to changing information on the Commander's Critical Information Requirements and inform the user how these changes might impact mission goals. Battalion staff can use tools to compare actual progress to the plan. The tool is not designed to replace the human, it is designed to help the human do their job.*

This background helps the SME answer questions about envisioned autonomy such as, "If you had this autonomy as your partner, how would you operate? What would be your top goals and concerns?"

Reassure the interviewee that autonomy will not replace them.

2.2 Demographics and Top Challenges

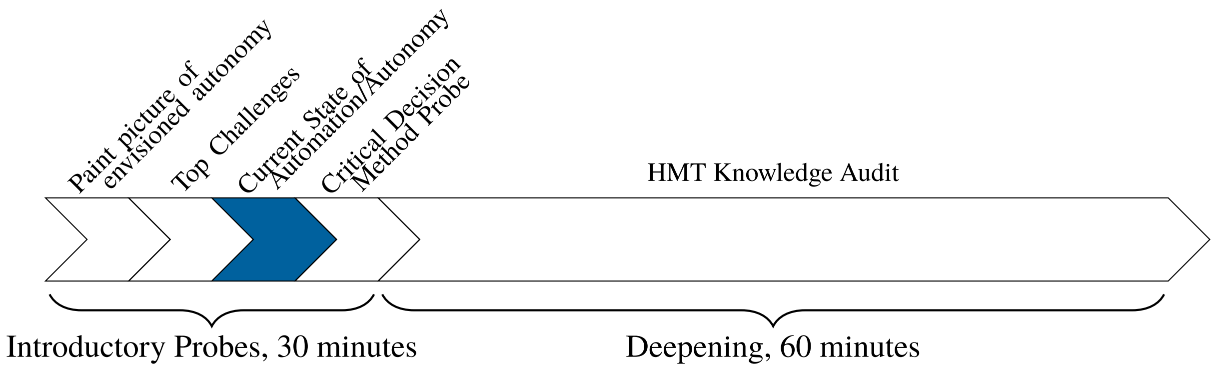


The demographics should be quick and to the point, targeting relevant expertise and not the interviewee's whole career. The Top Challenges questions are similar to the Task Diagram in the Applied Cognitive Task Analysis method [24]. The goal at this point is to understand key responsibilities of the job and which functions are the most cognitively challenging. Knowing these helps frame interview questions later in the interview. For example, if you learn that predicting outages is a challenging aspect of the job, you could later ask a tailored question, "If an automated agent assisted with predicting outages, what do you want the agent to tell you as it's working for you?"

- Formal duty title, rank?
- Years/months experience in role?
- Other relevant experience (training, previous positions, etc.)?
- What are the top 3-5 tasks you're responsible for?
- Which ones are the most difficult, cognitively?

> Put a large sheet of paper in front of the interviewee and ask them to write out the top 3-5 tasks. Invite the interviewee to sketch on the paper at any time during the interview. If tasks are sequential or related, ask the interviewee to sketch the order, dependencies, or relationships.

2.3 Current State of Automation/Autonomy

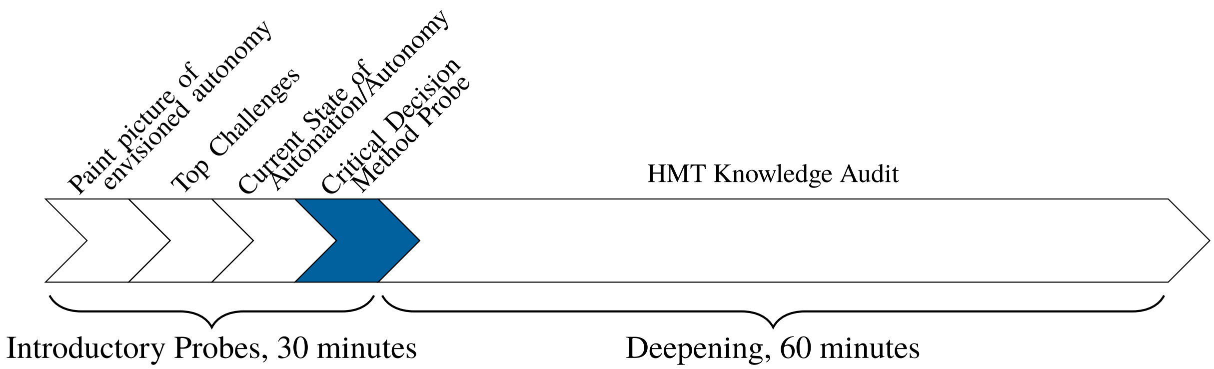


The purpose of this portion is to understand how SMEs are currently using automation/autonomy. Research this before you conduct your initial interview. Even if you know all the automated features, you may be surprised by how they use automation/autonomy to do their job.

1. What is the current state of automation, and how is it presently implemented?
2. How do you use automation to do your job?
3. How does it not support you? In what ways it is unreliable or challenging?

Listen for how the automation was intended versus how it is being used.

2.4 Critical Decision Method Probe



The Critical Decision Method is a technique for unpacking the cognitive elements of an actual lived incident in which the SME used their expertise to deal with a challenging or uncommon incident [14]. In this application, you will not review the incident in detail (which can take an hour). Instead, the goal of this probe is to quickly elicit examples from interviewees' experience that help you understand the challenges of the work and how an autonomous partner could be beneficial. The probe also helps the interviewee understand the types of information you are looking for.

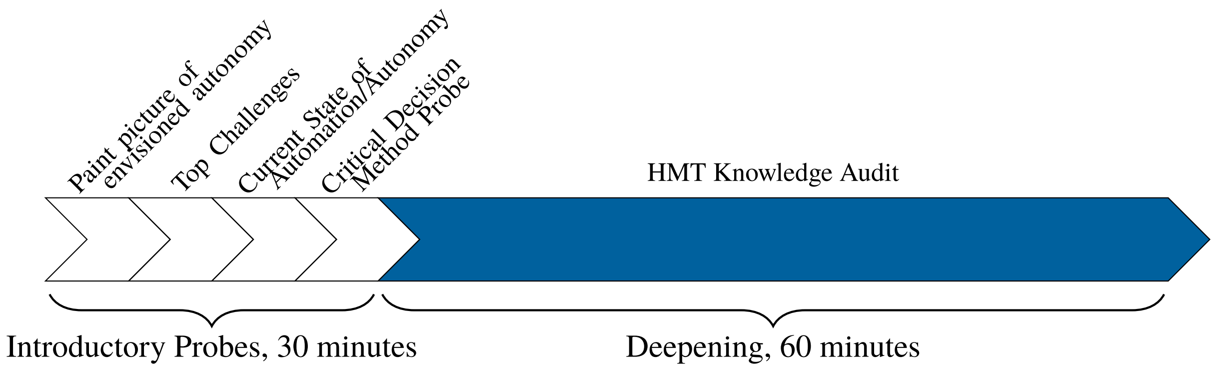
The general format is to ask the SME to "Tell me about a time when..." the SME and their teammates performed tasks in challenging situations (either with or without automation). Tailor the probe to your domain.

Example questions:

- Can you tell me about a particularly challenging situation when...
  - You had degraded communications, and it was a challenge to understand what was going on?
  - You were task saturated and didn't know if you'd be able to get your tasks done or you had to rely on teammates for help?
  - You didn't have the information you needed to make a decision?
  - It was difficult to choose between options?
  - The work you were asked to do felt beyond the expertise of you and your team?

The Critical Decision Method probe can be omitted in a time-constrained interview.

2.5 HMT Knowledge Audit



The Knowledge Audit is a well-known interview technique that focuses on short concrete examples of decision making as opposed to generic knowledge [24]. This version was adapted to address the challenges of human-machine teaming. Figure 4 shows the links between the traditional categories in the Knowledge Audit (Past and Future, Big Picture, etc.) and the HMT Leverage Points that are related.

In addition to offering questions associated with each area, this HMT Systems Engineering Guide describes the purpose behind the questions and types of information that they are intended to elicit. This "relevant information to uncover" should not be confused with the questions. It would not be productive to run through such a laundry list of issues with SMEs. Instead, they are provided to enable you to know what to listen for during the interviews.

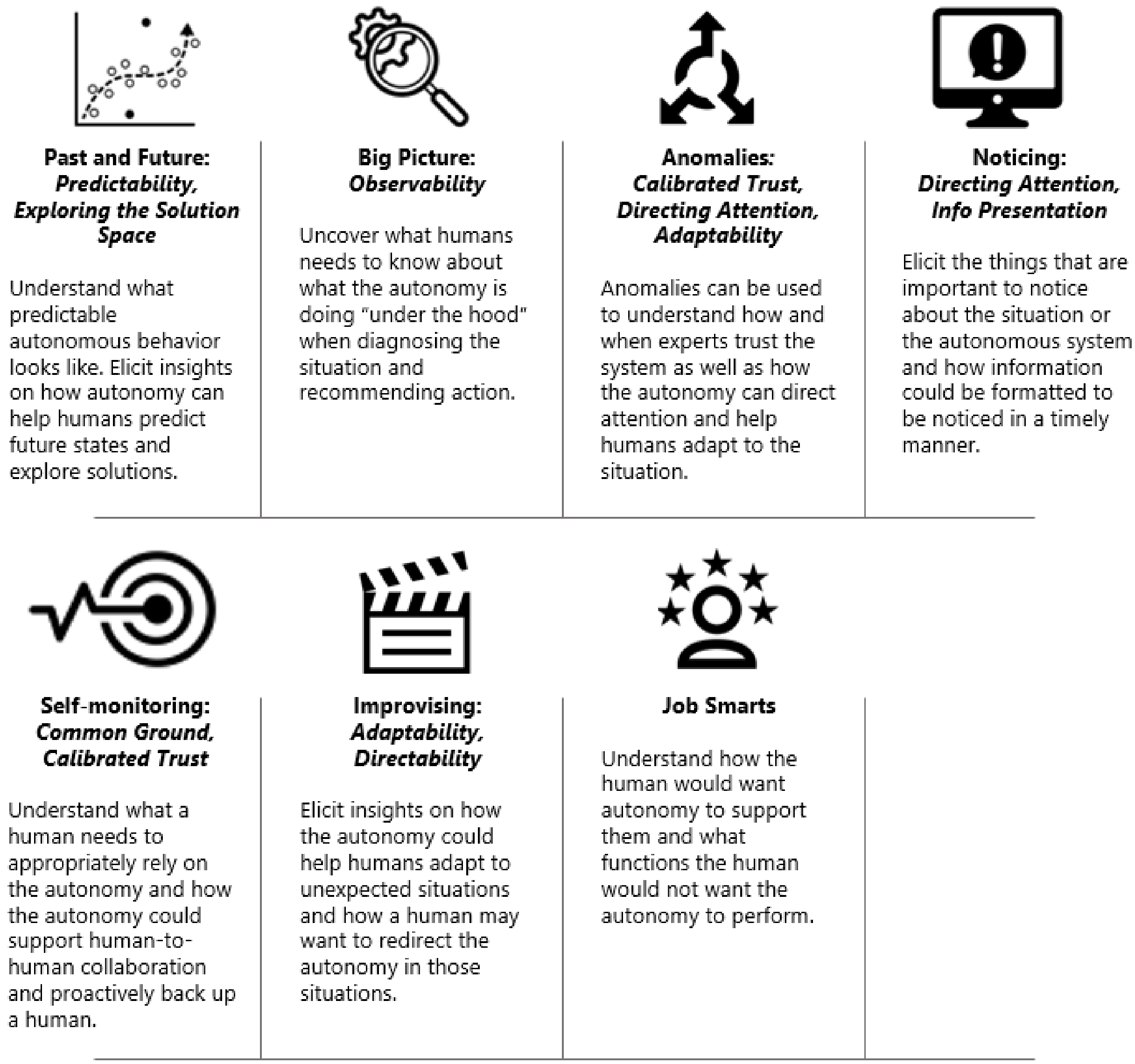
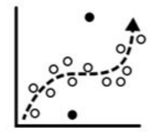


Figure 4. Mapping traditional Knowledge Audit categories to the HMT Leverage Points

2.5.1 Past and Future: Predictability, Exploring the Solution Space



Experts know how a situation might evolve over time. Experts need autonomy that helps them predict future states, and they need autonomy to act in a predictable way so it does not surprise them. Understanding historical information and likely future developments is related to Exploring the Solution Space as experts seek to understand what is happening and how to effectively deal with it. Relevant information to uncover includes:

Predictability Elements	Exploring the Solution Space Elements
How reliability changes over time – under what circumstances does it change	How Courses of Action (COAs) are developed for normal and for atypical days
Automation's goal, ability, limitations	Current ways that COAs are compared
Likely automation faults	The criteria used to compare COAs
Major situation changes and how to anticipate them	How solutions are walked-through to troubleshoot and refine them
How the system tracks a human's goals/actions/needs/priorities	COA considerations that can be missed
The kinds of changes or events that should be anticipated	Dependencies and contingencies
The predictions that would be useful	

Interview questions to understand the Predictability and Exploring the Solution Space components as they relate to the Past and Future:

1. How predictable or variable are the missions you'd accomplish with this system?
2. As you do this work, what is really critical to understand about what might happen next? Are you predicting the next few minutes, or is it hours, days, weeks?
3. Can you think of a time where you needed to understand, or have at your fingertips, historical information to understand what to do in the future? For example, you needed to understand typical satellite movements in order to predict future movements or spot anomalies.
4. In what situations might you need a tripwire or an automated alert that stood watch for you? What would you want it to tell you? For example, you set an airfare alert so you are notified when the airfare from Denver to New York is under $300.

Use the examples if the interviewee has trouble answering.

2.5.2 Big Picture: Observability

The purpose of Observability is to capture what people need to know about their autonomy and what it is doing "under the hood," so that humans can partner effectively with the autonomy and understand the big picture. Elements that a human may need to better understand include:

Observability Elements
Automation goals and intentions
Automation tasks, responsibilities, and priorities
Automation status and state parameters
Automation's progress toward goals
Automation difficulties
The problem being solved
The problem constraints
Ongoing reliability, reliability about what?
Likely automation errors
How the autonomy will adapt to changing situations

Interview questions for Big Picture and Observability:

1. What's the overall battle rhythm, or decision timeline, view of this system? (e.g., does data needed for planning change continually throughout planning cycles?)
2. *What do you want the automation to tell you as it's working for you?
3. *Can you describe a time when you were confused about what your system was doing? For example, what mode of automation the system was in, or how the system calculated a recommendation?
4. What are the key vital signs to know that you are on track to accomplish the mission?
5. How might automation help you coordinate more effectively, if at all?
6. What do you need to know and understand to have situational awareness?

> Keep probing for specific examples, as opposed to generalities.
>
> Questions marked with * are especially good at generating information about observability.

2.5.3 Anomalies: Calibrated Trust, Directing Attention, Adaptability

Anomalies disrupt users from their normal way of working and therefore are important to understand. Eliciting anomalies and understanding how users diagnose and respond to them can be useful in designing autonomy that directs users' attention and adapts to the situation. Analyzing typical anomalies can also help in the design of autonomy that highlights weaknesses/limitations and therefore fosters calibrated trust in the autonomy. Furthermore, anomalies can be used to understand how and when experts trust the system. Relevant information to uncover includes:

Directing Attention Elements	Adaptability Elements
Context	Unexpected situations
Interesting system or situational behavior to look for (changes in behavior, obstacles to goals, hazards, failures)	Cues or patterns used to recognize unexpected situations
Patterns to look for	Cues or patterns used to recognize that the situation is deteriorating
How to judge interpretability	The range of demand, or busyness
Times that are better to interrupt	What sunny day/rainy work periods look like
Alarms and thresholds	Problem elements or things that can be constrained

Interview questions for Calibrated Trust, Directing Attention, and Adaptability:

1. *What are the biggest system anomalies you worry about?
2. *Are there nuances that people pick up on over time to know things are heading south?
3. Can you describe an instance when you spotted a deviation from the norm, or knew something was amiss?
4. Do you know of certain conditions in which your system typically provides unreliable information? For example, communications typically fail when going through a canyon.
5. Can you think of a time that you needed to improvise? Are there things the system could do to help you adapt?
6. Are there anomalies that occur routinely with your system or the situations you encounter?

Questions marked with * are especially good at generating information about anomalies.

2.5.4 Noticing: Directing Attention, Information Presentation

These questions elicit the things that are important to notice about the situation or the autonomous system. The autonomy can help by directing attention to changing elements in the situation or to critical changes in the autonomy itself. The information can be presented in a way that helps users to notice changes in a timely manner.

Interview questions for Directing Attention and Information Presentation:

1. Can you think of a time when you missed something important that didn't pop out at you in a clear way?
2. Can you imagine a role for automation to direct other team members to see what you're looking at, to get a common frame of reference for the team when something important is happening?
3. Have there been times that you wish you were notified of new or changing information? Would that have made a difference in your decision making?

> Asking about other team members can help the interviewee relate to what a digital teammate could do.
>
> Keep probing to elicit specifics, not textbook answers.

2.5.5 Self-Monitoring: Common Ground, Calibrated Trust

The operator should understand what the system is doing and have enough information to appropriately rely on the system. It is important to understand the automation purpose and process (the why behind partner's actions) as well as limitations of the automation. Relevant information to uncover includes:

Common Ground Elements	Calibrated Trust Elements
Signals, cues, or indicators that a coworker needs backup	When automation can be relied on
Situations needing proactive backup	When oversight is needed
Goal prioritization, who decides	Sources of unreliability
How critical information is communicated to teammates	Source of diagnostic information
How your state/workload is represented to teammates	Credibility

Interview questions for Common Ground and Calibrated Trust through Self-Monitoring:

1. Can you describe a situation when you knew you were task saturated/overloaded and had to ask for help?

2. If a new person on your team were to take over your job, what would you be most concerned about? What part of your job would you feel most uneasy about if a new or inexperienced person was doing it?

3. What indicators might you use to know that something's amiss with your automated partner's performance, or its intent, or assumptions? For example, the situation changed, and the system's algorithms aren't calibrated, or applicable.

2.5.6 Improvising: Adaptability, Directability

If an operator needs to improvise, it is likely due to an unexpected situation. This topic can shed light on how well the system helped the operator adapt to unexpected situations. When improvising, an operator may need to take control or redirect the automation. Relevant information to uncover includes:

Directability Elements
Indications of need to retask or change course
Constraints
Task priority considerations
Functions likely to be retasked

Interview questions for Adaptability and Directability through Improvising:

1. What are some example situations in which you have had to rapidly improvise a plan? For example, either to handle a threat or take advantage of an opportunity.

2. Using this example, how might your automated partner and you take advantage of this opportunity/handle this issue?

3. When the situation has changed, what might you need to understand to direct the machine teammate's response to this change?

2.5.7 Job Smarts

This topic is included to capture the tacit knowledge that people gain as they work with systems over time.

These questions deal with the overall capability of an automated/autonomous partner.

1. *If the government were to give you an intelligent robot/digital team member (like C-3PO), what are the top things you'd like it to do for you?
2. *On the flip side, what would you not want your robot/digital team member to do?
3. If you had automation to help, what tribal, or local knowledge would it need to be aware of to be effective?
4. Are there administrative, information seeking, or representation tasks that your automated partner could do to make the team more efficient and effective? For example, rapid lookup of phone numbers for mission partners.

> If you have limited time, be sure to ask the questions marked with *. They have high payoff.
> Listen for clarifying comments about automation prerequisites (e.g., "that would be great, as long as the automation is/can...").

2.6 Printable Interview Guide

For a printable interview guide, see appendix A. You can print the interview guide to use as is or you can tailor it to better match your domain (e.g., description of the autonomous system, Critical Decision Method probe).

3 Data Analysis

After you have conducted a round of interviews, the next step is to analyze the results. This analysis is an important and time-consuming effort which should be included in planning timelines. There are multiple cognitive task analysis techniques for data analysis and representation [7]. This guide focuses on finding, categorizing, and extracting *HMT elements*.

3.1 Prepare Interview Notes

If needed, convert handwritten interview notes to digital format. Format the interview notes in a way that allows you to summarize and categorize the data. If you are using a software tool to aid in analysis (NVivo or Atlas.ti are two commonly used tools) you can leave the answers as text following each question.

Next, consolidate the notes from both interviewers into one electronic document. This provides one source of interview data per interviewee and helps prevent unintended redundancies in coding. At the end of this stage, you will have one document per interviewee. You may want to provide a copy of the interview notes to each of the experts from whom you gathered data so they can suggest changes or clarifications.

3.2 Code the Interview Notes

The purpose of this step is to categorize textual information into the 10 leverage points for HMT (see Section 1.2). This will allow you to see all the nuggets or insights associated with each leverage point.

First, read over the General HMT Requirements so you are familiar with the topics addressed under each leverage point. Concentrate on the leverage points that are most relevant to the domain. Second, read through each interview answer and decide if it is related to any of the leverage points. (Design Process material is not expected to be seen in the data as it represents overall lessons learned and philosophy regarding design and should not vary by domain.)

Tips on recognizing leverage points are shown below.

Information to Look for in Interview Notes	Example from Planning Tool Interviews
Observability	
• Answers to the question, "Observability about What?" For example: • Algorithms • Calculations • What the automation is doing • Automation Checks	*For time distance analysis. . . does it [the automation] take into account whether units are traveling over roads or cross country?*
Predictability	
• Predictions • Historical data • Unexpected behavior of autonomy • Surprises	*Based on terrain and weather, what will the enemy use and where? Accurately predicting is the most difficult.*
Directing Attention	
• Changes in state or status • Something new or out of the ordinary • Alerts • "I want the automation to tell me..."	*... don't alert the entire staff of the FRAGO, but if the FRAGO was only related to an area specific to Fires or Intel, route to the right staff section or officer to evaluate the condition and make the decision/choice how to proceed*
Exploring the Solution Space	
• Problem Solving • COA Development • COA Analysis • "Help me understand..."	*I want a Digital Battlebook that automatically adjusts template for terrain (i.e., units would go quicker if one unit went behind the hill).*
Adaptability	
• Unexpected events • Problem solving constraints	[in response to the question, How might your machine teammate help you take advantage of this opportunity/handle this issue?] *Providing different back up plans*

Directability	
• "I want to control..." • "I don't want the system to..." • Retask, prioritize • Tweak, refine, override	*I want the ability to override the analysis based on the expertise of the S2 (e.g., dry creek bed on route may be a No-Go area when it has rained for the past 3 days).*

Calibrated Trust	
• Reliability, credibility • Information sources	*Let the user view the results and have the ability to see the assumptions and inputs that went into the automation/algorithm result.*

Common Ground	
• Distributed systems (system is remote from human partner) • Common Operational Picture • Team Situation Awareness • Human-human cooperation	*Can work by triggers or phases. By this phase, once enemy reaches this line, pull back the FAS. Support as much as possible until the very last minute. Best way I can support you is to place [resources] here.*

Information Presentation	
• Communicate to the user • See easily	*Can do alerts to notify when certain fuel (or ammo) levels are reached. Colored bars (green, amber, red) under each unit icon indicate status.*

When coding, make sure the context of the statement is captured in the nugget that is coded so that the text is still meaningful when extracted.

Example 1, Original:

Yes, it takes into account the terrain. The human is in control.

Example 2, Expanded with context:

How are movement times calculated? Does it take into account terrain? Response: Yes, it takes into account the terrain. The human is in control.

Example 2, Original:

I want the ability to annotate equipment status

Example 2, Expanded with context:

It would be great to log into a system to see the status of all the communications equipment and be notified of any equipment failures. I want the ability to annotate equipment status so I can change it from mission-ready to non-mission ready if I know of a problem with the equipment that would impact my mission.

Try to limit the number of codes assigned to a chunk of text to 2-3 codes. There may be other applicable codes, but focus on identifying which codes are primarily addressed in the statement. The figure below shows a chunk of text with three associated codes: Calibrated Trust, Directability, and Observability. Figure 5 shows a screenshot of a qualitative coding tool that allows users to highlight blocks of text and assign pre-defined codes. Using such a software tool can streamline the extraction of quotations and the sharing of information across multiple coders.

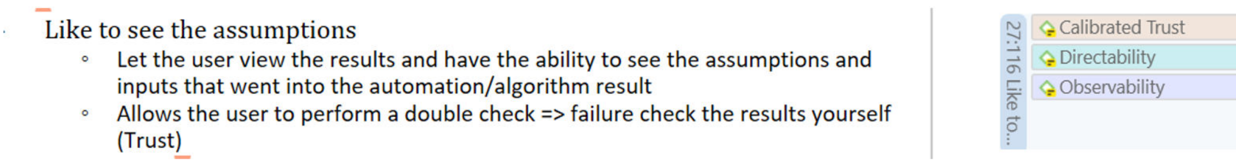


Figure 5. Example of document coding tool

Depending on the timing of your project and the rigor involved, you might distribute the interview coding differently:

- One person codes all interviews.
- Multiple team members code each interview.
- Split the coding among the team so that each interview is coded by one person.

Having multiple people code one document has the advantage of "catching" more quotations, but it is more labor intensive. If you do use multiple coders on the same document (i.e., double coding the same document), it is not typically necessary to review each code for agreement.[2] If double coding the same document, try to stick to 2 codes per statement so the same statement does not end up with more than 3 codes. If splitting the coding between multiple team members (third option above), it is helpful to start by having all coders work on the same document and review it together to discuss discrepancies so all coders do the remaining interviews with the same process and understanding.

Try to limit yourself to no more than 2-3 codes per statement.

[2]When doing formal, reportable research, standard practice is to have more than one coder review each document and calculate inter-rater reliability to report coder agreement. These formal practices are typically beyond the time scales and goals of requirements development processes, but might be needed if reviewers vary widely in interpretation of codes.

4 Writing Requirements

The goal of this step is to produce actionable HMT guidance for developers, based on the data collected. First, decide whether you will format the guidance as requirements or as user stories. To help you decide:

> **User Stories** have the value of capturing elements of the human's workflow and experience with the system. These may be more appropriate for an agile development process, or as an interim step prior to developing specific requirements. User stories have more flexibility in the amount of details or context that can be provided. User stories generally take the form of:
>
> > As a <user role>,
> >
> > I want <what?>
> >
> > so that <why?>.
>
> For more information on User Stories see:
> http://www.romanpichler.com/blog/10-tips-writing-good-user-stories/
>
> **Requirements** have the value of being precise and evaluable. They generally take a form that focuses on system performance, and may, depending on at what point of the system development process they are being delivered into, be written with the appropriate degree of specificity and testability. For example: "The autonomy shall..." For more information on Writing Requirements, see Wiegers (1999) [33].

In the remainder of this section, we will default to the term "requirements" but this could be replaced with "user stories."

Second, choose an HMT leverage point and extract all the interview guide statements/quotations for that topic. Read the General HMT Requirements for that leverage point and keep the list of General HMT Requirements handy. At this point, you have two options. You can take a top-down approach and use the General HMT Requirements as your starting point. Or, you can take a bottom-up approach in which you use the interview quotes as your staring point. In the top-down approach, identify the relevant General HMT Requirements, then use information gathered during the interview to make each one more specific. For example, consider the following General Requirement for Directability:

- The automation/autonomy shall provide the operator the capability to redirect the automation's tasks *in order to achieve operational goals.* [15]

Determine if any of the interview data provided insight on tasks that the human partner should be able to redirect. Write a requirement for each specific task. For example, the General Directability Requirement above could be adapted to the following requirements for an autonomously operated telescope.

- The autonomy shall provide the operator with a capability to change the telescope scan pattern.

- The autonomy shall provide the operator with a capability to redirect the autonomy if it starts to close the dome.

Repeat this process for other relevant General HMT Requirements. Note that a General Requirement may be appropriate as is and not require tailoring.

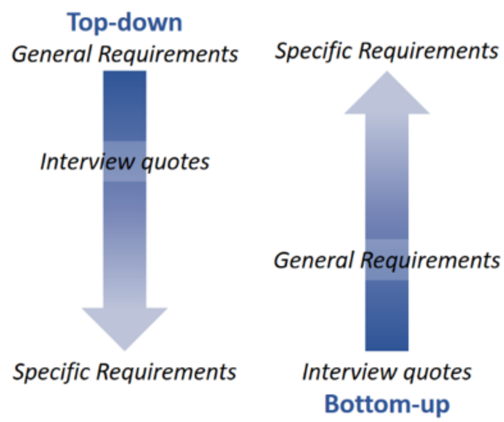


Figure 6. Requirement Synthesis Styles

In the bottom-up approach, the extracted quotations are the jumping off point. For each leverage point, create a table that is divided into the subcategories of that leverage point. For example, the table below has a section (shown in grey) on each of three subcategories in Calibrated Trust. Categorize the interview quotes into these subcategories, grouping quotes that are similar. For example, the two statements under "Shed Light on Information Sources" relate to fact checking and determining the quality of the data source. Write a user story that defines who the user is, what they want, and why they want it. It is possible that not every subcategory will be applicable to your system.

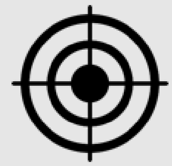
Interview Data	Calibrated Trust User Stories
Shed Light on Information Sources	
Would want the ability to fact check before orders are generated. Automation and receipt of message confirmation, providing the information to the users of the system who made the change, would have ensured, without picking up a phone to call around, the status of the information that their echelon, and HHQ and sister echelons had of the change. This concept ensures that users are aware of the information that they have available to make decision on.	As a planner, I want access to information sources that underlie tasking orders, for both human-entered information and computer-generated information. As a planner, I want access to a history of information updates, including when it was changed, who changed it, and when other users received the change.
Promote Understanding of the Automation	
Like to see the assumptions. Let the user view the results and have the ability to see the assumptions and inputs that went into the automation/algorithm result. Allows the user to perform a double check => failure check the results yourself.	As a user, I want access to assumptions and inputs that are used in automated analyses and recommendations.
Prevent Under and Over Reliance	
System should provide confidence indication for recommendations.	As a planner, I want the system to provide confidence levels in its recommendations, along with an explanation of data used and how those confidence levels were calculated.

The end product is a list of user stories by leverage point. For example, below is an excerpt of user stories related to Observability for an Army Planning tool:

Observability about System Reliability

As a planner, I want to know the source of GPS information (e.g., military grade or commercial grade) to understand the likelihood of error in analyses or calculations [also related to Calibrated Trust].

Observability about What the Automation is Doing



As a planner, I want the ability to drill down to understand how the targeting was derived.

Observability about Intentions

As a planner, I want the automation to communicate projections of fuel use and any uncertainty associated with those predictions so I can plan resupply.

If you are time-constrained, choose a subset of the most relevant HMT leverage points on which to write requirements or user stories. This could be the leverage points with the highest number of codes or it could be the leverage points that you infer will have the highest impact on HMT performance.

> Write requirements one HMT leverage point at a time so you are not overwhelmed with the amount of data to address.

5 HMT Heuristic Evaluation

5.1 How to Use

The HMT Heuristic Evaluation is designed to elicit information during the evaluation stage about how well HMT is supported by an existing system's design. The focus is on specific information about what the autonomy does well (things that should be retained) and what the shortcomings are (things that should be addressed).

The HMT Heuristic Evaluation was originally designed to be administered as a questionnaire. The wording is straightforward for users, and there are examples for each leverage point, which illustrate what that leverage point could look like in context. The HMT Heuristic Evaluation was piloted with a prototype Command and Control system that included highly automated and autonomous features associated with unmanned vehicles as well as cognitive assistance.[3] Key lessons learned included:

- Administering the HMT Heuristic Evaluation as an interview rather than a questionnaire is more effective and timely. An interview format allows the interviewee to expand upon answers without having to write every word. In the pilot, the answers to the first few questions were useful but the length and quality of answers diminished as the questions progressed. Administering it as an interview will help keep the SME engaged and will allow the interviewer to follow-up on interesting comments. For example, if the SME says that they want asset availability to be more pronounced, the interviewer could ask what that could look like.
- The primary respondents are the SMEs using the autonomous systems. However, the HMT Heuristic Evaluation was successfully used with a program team who were involved in the development and the evaluation of the system. These project members answered questions based on their observations of SMEs using the autonomous system.

5.2 HMT Heuristic Evaluation Interview Guide

A printable HMT Heuristic Evaluation interview guide is included in appendix B.

5.3 Example Results

This excerpt of results illustrates the insights gained from applying the HMT Heuristic Evaluation in the Advanced IMPACT pilot effort, with an eye toward identifying gaps and future opportunities for support.

[3] The Advanced IMPACT (Intelligent Multi-UxV Planning with Adaptive Collaborative Control Technology) system prototype has been developed and used in applied experiments by military service research laboratories (AFRL, ARL, SPAWAR) and coalition partners from Five Eyes countries. AFRL partner scientists invited MITRE to participate in a July 2017 evaluation of Advanced IMPACT with SMEs.

Success	Gaps
Observability	
Highly observable aspects: • Upon play call, green progress bar shows the agent's processing status. Includes detailed description of what is currently being reasoned on, e.g., "retrieving routes."	Increase observability about: • The need for or reason behind for auto-initiated plays: they just appear in active play manager and the user must investigate why they were called. • Why the user-selected soft constraints cannot be met by the agent
Directability	
Aspects that offer directability: • The play manager interface allows for quick play state manipulations, e.g., pause/cancel active plays, initiate inactive plays.	Increase the ease and speed of directing the system to: • Reallocate vehicles across active and inactive plays • Cancel and recall plays
Adaptability	
Aspects that offer adaptability: • Reroutes around injected restricted airspace (ROZs).	Aspects that could offer more adaptability: • Use automated agents to address vehicle failures due to sensor or fuel problems, e.g., currently, vehicles with sensor failures will continue until the problem is recognized and adjusted by operator, but in the future, automated agents could intervene and notify the operator of changes.

6 General HMT Requirements

6.1 Overview

6.1.1 Purpose of General HMT Requirements

Well-designed and successfully implemented autonomy acts seamlessly within the human operator's workflow, aiding the operator without interfering with the operator's tasks. The addition of autonomy or automation has the potential for enhanced performance that is better than either the autonomy or the human alone. This chapter compiles evidence-based guidance collected during an extensive literature review of research on human-machine teaming to aid in the design and development process.

The requirements in this document differ from traditional requirements. They are intentionally general, or generic, so that they are widely applicable to a range of autonomous systems, from autonomous vehicles to robotic systems to cognitive assistants. Because the scope is so large, system developers will need to adapt these General HMT Requirements for specific systems. Some of the unique aspects of these General HMT Requirements are:

- Multiple concepts may be combined in a list within a requirement. This contrasts with traditional requirements that contain one statement that can be cleanly evaluated. When tailoring the requirement, a systems engineer would choose which aspects of the list are applicable and write separate requirements for each aspect.
- Statements about the rationale behind a requirement may be included in italics. This information is included to help a system engineer decide if (and when) a requirement is relevant.
- Examples are included to give more details about how a requirement could be applied in context. These examples are hyperlinked within the document (and referenced in the text for print versions).
- Each requirement references the article from which the guidance was extracted. These numbered references are shown in brackets and are hyperlinked to the full citation in the reference list.
- The requirements may contradict each other because they are not all meant to be applicable to a single system.
- The terms autonomy and automation are used interchangeably in the requirements.

The process for tailoring these requirements is documented in Section 1.

6.1.2 HMT Background

The General HMT Requirements fall into 10 themes or leverage points for supporting HMT [23]:

- *Observability*. Observability provides transparency into what an automated partner is doing relative to task progress. Observability supports shared understanding of the problem to be solved and progress toward goals. Autonomy is observable when it provides the right level of information so humans understand how it is doing calculations and arriving at recommendations and predictions.

- *Predictability*. With Predictability, future intentions and activities of the partner are discernible and understandable. It is a subset of Observability, and based on an understanding of the automation's goals, abilities, and limitations. Predictability also means that the autonomy can anticipate changes in the situation to aid the user in projecting future states.

- *Directing Attention*. The autonomy must be able to direct the attention of the human to critical problem features, cues, indications, and warnings. Automation should communicate proactively when information becomes relevant, such as when obstacles to meeting goals are encountered.

- *Exploring the Solution Space*. Exploring the Solution Space helps partners leverage multiple views, knowledge, and solutions to jointly understand the problem space. Automation should be able to rapidly generate multiple distinct courses of action and give people ways to rapidly compare those solutions. Both humans and technology should be able to broaden or constrict the solution considerations to shift perspectives.

- *Adaptability*. Adaptability enables partners to recognize and adapt fluidly to unexpected characteristics of a situation. Autonomy should have multiple options to recognize an unexpected situation and address it.

- *Directability*. Directability is supported when humans are able to direct and redirect an automated partner's resources, activities, and priorities. Humans will have expertise or insights which the automated system will not. The human is ultimately in control, which means they can stop processes, toggle between levels of autonomy, or override and manually control the automation.

- *Calibrated Trust*. Calibrated Trust is supported when human operators have a strong understanding of when and how much to trust an automated partner in context. Users should understand clearly when automation can be relied on, when more oversight is needed, and when performance is unacceptable. To help users calibrate trust, the autonomy can provide information sources, and the credibility of those sources.

- *Common Ground*. Achieving Common Ground means that pertinent beliefs, assumptions, and intentions are shared among team members. Common Ground is constantly and actively updated and maintained so that team members (both human and autonomous) can maintain a shared picture of what's happening in the world and engage in backup behavior to support each other.

- *Information Presentation*. This category captures published recommendations on how to present information to support simplicity and understandability. The user should be able to view and interact with information in order to understand the implications of the data.

- *Design Process*. This category captures elements of the cognitive engineering design process that must be incorporated to enable effective HMT. This includes philosophy of design and guidance on how to structure design activities.

The leverage points are further organized into four categories depicted in Figure 7.

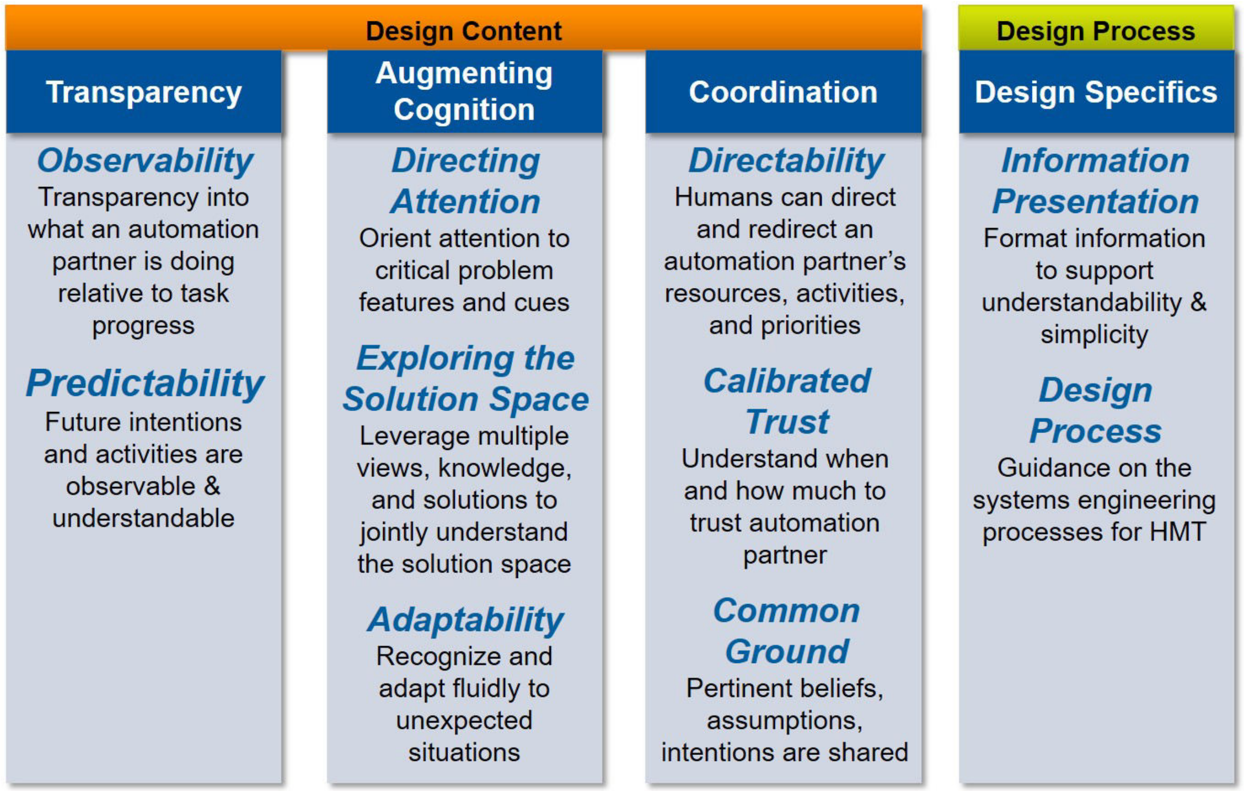


Figure 7. Ten leverage points organized into a Framework for HMT

Many of the leverage points are highly related to other leverage points. Figure 8 captures these relationships as seen in the human-machine teaming interview data from two domains. Some of the noteworthy relationships are as follows:

- *Observability and Common Ground*. In Observability the emphasis is on the human understanding what the autonomy is doing and how it is doing it. Common Ground is the two-way street in which the human and the automation mutually are aware of each other's goals, assumptions, and priorities.

- *Calibrated Trust and Observability; Calibrated Trust and Common Ground*. To understand how and when to trust the autonomy, the actions and goals of the autonomy need to be observable to the human. Likewise, if the autonomy can consider the human's changing goals and priorities, the human can better understand when to trust or rely on the autonomy.

- *Directing Attention and Information Presentation*. Directing attention addresses the type of information that needs attention and when to interrupt the user. Information Presentation addresses how to format the data so it is salient.

- *Directing Attention and Predictability*. It is often beneficial to direct the operator's attention to predictions about how the system may develop.

It is useful to understand these relationships because they can point system engineers and designers to other applicable leverage points to consider. If one leverage point is relevant in a domain, it may be beneficial to research highly related leverage points.

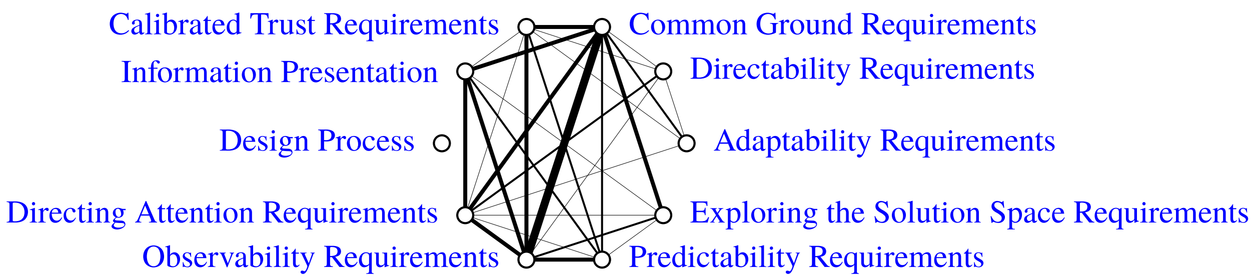


Figure 8. Theme relationships, line weight indicates the strength of the relationship

6.2 Observability Requirements

6.2.1 Observability about System Reliability

- The automation/autonomy shall provide accurate, ongoing feedback on reliability of automation and factors affecting reliability *to encourage more appropriate use.* See §6.9 Calibrated Trust Requirements. [12, 20, 25]
- The automation/autonomy shall provide explanations for automation's errors that occur *in order to discourage automation disuse*. See example §C.1 Distinguishing Targets. [12]
- The automation/autonomy shall provide information about the past performance of the automation in specific contexts. See §6.9 Calibrated Trust Requirements. See example §C.2 Autonomous Accuracy. [19, 20]

6.2.2 Observability about What the Automation is Doing

- The automation/autonomy shall provide information about the automation's goals, status, actions, progress, processes, difficulties, and responsibilities. See §6.8.2 Communicating Goals. [15, 34, 1]
- The automation/autonomy shall communicate the context of its actions *so that the human understands the automation's reasoning*. See example §C.3 Autonomous Telescope. [19]
- The automation/autonomy shall provide the human operator feedback regarding its current performance, *especially if the human and the system are not co-located.* [11]
- The automation/autonomy shall notify the human of changing levels of automation *because the responsibilities of the human and the automation may change.* [15]
- The automation/autonomy shall provide information about how automation activities may evolve in the future. See example §C.4 Auto-GCAS. [5]
- The automation/autonomy shall show intermediate results in a way that is comprehensible to human operators *so the humans have insight into the automation processes and algorithms.* See example §C.5 Advanced Planning System. [19]

6.2.3 Observability about Intentions

- The automation/autonomy shall provide information about status. [18]
- The automation/autonomy shall provide information about intentions. See §6.8 Common Ground Requirements. [18, 20, 34]
- The automation/autonomy shall make changes observable as the automation adapts to evolving situations. [34]
- The automation/autonomy shall communicate its capability in different situations. [19]

- The automation/autonomy shall communicate information regarding the automation's projections of future states and any uncertainty associated with those predictions. [4]

6.3 Predictability Requirements

6.3.1 Supporting Predictable Behavior

- The automation/autonomy shall communicate its goals, abilities, and limitations *so that humans can predict the automation's actions in various contexts.* [15]
- The automation/autonomy shall communicate the actions it can and cannot accomplish in relation to a goal. [15]
- The automation/autonomy shall understand and predict the actions of human team members *so that the human-automation team is mutually predictable.* [18]
- The automation/autonomy shall behave in a manner that is predictable (i.e., matches the expectations of the operator) under all circumstances. [18, 1, 28]
- The automation/autonomy shall behave in a manner that is dependable (i.e. consistent and effective). [28]
- The automation/autonomy shall provide information about its states, actions, and intentions in a way that is salient enough to draw operator attention. [25]

6.4 Directing Attention Requirements

6.4.1 Direct Attention to Changes and their Implications

- The automation/autonomy shall direct attention to plan changes or departures from goals. [15, 1]
- The automation/autonomy shall direct attention to actions or events that may critically affect the goals of the human-automation system. See example §C.6 Autonomous Vehicle Notification. [15, 1]
- The automation/autonomy shall direct attention to relevant changes and events in the monitored process. [5]
- The automation/autonomy shall alert the operator to behavior that deviates from normal. [34]
- The automation/autonomy shall alert the operator to constraints and violations of those constraints. [27]
- The automation/autonomy shall clearly direct attention to automation failures, malfunctions, or changes in automation behavior. [34, 1]
- The automation/autonomy shall inform the user about information that the automation did not consider in solution generation. See example §C.7 Automated Cockpit Assistant. [27]

6.4.2 Anticipating Events

- The automation/autonomy shall direct the operator's attention to the fact that an event is likely to occur, given historical data. See §6.3 Predictability Requirements. See example §C.8 Satellite Deviation. [25]
- The automation/autonomy shall inform the human of situations with high potential for human error. See example §C.9 Equipment Fault. [15]

6.4.3 How and When to Direct Attention

- The automation/autonomy shall direct attention with enough time to allow the operator to intervene effectively in the event. [25]
- The automation/autonomy shall be able to influence where and on what the human partner and other automated systems focus. [34]
- The automation/autonomy shall provide enough information about the developing event so that the operator can intervene effectively. See §6.10 Information Presentation. [25]
- The automation/autonomy shall judge interruptibility of the human operator and other robotic systems and only interrupt when it is appropriate to do so. [34, 31]

- The automation/autonomy shall maintain a model of human work so as to appropriately interrupt that work when necessary. See example §C.10 Interruptability. [31]
- The automation/autonomy shall emphasize information in accordance with its importance. [1]
- The automation/autonomy shall show patterns and relationships in monitored processes. See §6.10 Information Presentation. [34]

6.5 Exploring the Solution Space Requirements

6.5.1 Solution Generation: Under the Hood

- The automation/autonomy's solution generation shall be rapid *to give the human time to iterate and explore options.* [27]
- The automation/autonomy shall generate multiple solutions that are operationally distinct *so the human has meaningfully different options to evaluate.* [27]

6.5.2 Solution Generation: The User in Control

- The automation/autonomy shall allow humans to generate distinct solutions independent of the automation. [27]
- The automation/autonomy shall allow a human to control the scope of the solution by putting constraints on how its solved. See example §C.11 Plan Locking. [27]
- The automation/autonomy shall allow a human to input information that wasn't initially taken into account by the algorithms *so the algorithm can incorporate new information or expertise of the human.* [27]

6.5.3 Solution Evaluation: Under the Hood

- The automation/autonomy shall help broaden the human's view of the situation and trends *to help revise focus and priorities so the human is better able to anticipate and adapt to surprises and errors*. The automation can do this by one or more of the following:
  - Seeding. The automation shall provide an initial analysis to help structure and kick start human thinking.
  - Reminding. The automation shall suggest other possibilities as analysis progresses.
  - Critiquing. The automation shall point out alternatives as analysis matures. [34]
- The automation/autonomy shall provide ways for humans to intuitively and rapidly compare and evaluate solutions. [27]

6.5.4 Solution Evaluation: The User in Control

- The automation/autonomy shall support "what if" exploration by the human *so they can explore the implications to different solutions.* [27, 32]
- The automation/autonomy shall allow the human to evaluate the viability of the automation's proposed changes to a solution. [2]

6.6 Adaptability Requirements

6.6.1 Multiple Paths for Recovery: Avoiding Common Source Failure

- The automation/autonomy shall have multiple options to address and recover from the same problem. [16]
- The automation/autonomy shall have multiple ways to recognize and handle an unexpected situation. [16]
- The automation/autonomy shall allow the human to specify problems at different levels of abstraction. See example §C.12 Plan Constraints. [27]
- The automation/autonomy shall update problem specifications as details emerge. [27]
- The automation/autonomy shall allow the human to update problem specifications as details emerge. [27]
- The automation/autonomy shall allow for incremental allocation of resources *in order to have more flexibility to adapt to changes.* [32]
- The automation/autonomy shall reserve degrees of freedom in an initial plan or course of action *to accommodate changes in requirements or priorities.*
- The automation/autonomy shall enable users to input informal priorities and changes in those priorities. See §6.7.1 Automation Priorities and Attention. [32]

6.6.2 Adjusting Autonomy

- The automation/autonomy shall sense demand on the operator *so the automation/autonomy shall adjust its actions as demand on the operator changes*. See example §C.13 Weather Change. [20]
- The automation/autonomy shall assist operators when performing seldom used functions. [1]
- The automation/autonomy shall be able to adjust behavior or level of automation for different human operators. [10]

6.6.3 System Degradation and Failure

- The automation/autonomy shall be resistant to errors and degrade gracefully *so human operators have sufficient time to recover*. See example §C.14 Degrading Gracefully. [1]
- The automation/autonomy shall be tolerant of errors and recover gracefully. See example §C.15 Recovering Gracefully. [1]
- The automation shall operate in a way so that the human can understand what is going on and take over if automation fails. [1]

- The automation/autonomy shall allow the human to operate manually if the automation fails. [1]
- The automation/autonomy shall maintain its own safety in the event of human operator absence. [10]
- The automation/autonomy shall maintain its own safety in the event of incorrect information. [10]

6.7 Directability Requirements

6.7.1 Automation Priorities and Attention

- The system will offer the operator various means to set, update, modify, or change priorities of the automation. See example §C.16 Sensor Prioritization. [2, 27]
- The automation/autonomy shall provide the means for the operator to redirect the agent's resources *when the situation changes or the human has information/expertise that is beyond the bounds of the agent's algorithms.* [34]
- The automation/autonomy shall provide the operator the capability to redirect the automation's tasks *in order to achieve operational goals.* [15]
- The human partner shall be able to redirect focus of the automated systems. [34]

6.7.2 Human Operator Privilege over System Control

- The automation/autonomy shall provide the operator the capability to override the automation and assume partial or full manual control of the system *to achieve operational goal states.* [15]
- The automation/autonomy shall provide the means for human redirection, stopping, and control of the automation; to include information mediation, processes, task, and goals. This requirement does not preclude automation initiating actions. [15]
- The automation/autonomy shall not remove the human operator from the command role. [1]
- The automation/autonomy shall not prevent or hinder the human operator from exceeding normal system operating constraints when such action is deemed necessary by the human operator. [1]
- The automation/autonomy shall provide the operator the appropriate range of system control and management options. [1]
- The systems shall allow for quick recognition and intervention by the human operator during problem solving. [5, 31]

6.7.3 Specifying Constraints for Collaborative Problem Solving

- The automation/autonomy shall provide the means for the operator to contribute to and revise the problem space. [27]
- The automation/autonomy shall provide the means for the human operator to communicate relevant information to guide the automations work, such as:
  - Providing a way for humans to modify default assumptions;

  - Providing a way for humans to guide problem solution;
  - Providing a way for humans to perform "what if" analysis. [30]

- The automation/autonomy shall allow for a sufficiently rich set of constraints to be input by humans, *so humans can accurately detail their specific planning needs to the system.* See example §C.17 Detailed Constraints. [27, 30]

- The automation/autonomy shall provide the means for the human to focus the automation on subproblems or to specify solution methods *that account for unique aspects of the situation of which the automated agent may be unaware.* [5]

6.7.4 Selection of Level of Automation [particularly relevant to robotics domain]

- The automation/autonomy shall provide a means to toggle between different levels of automation *in order to adjust the division of labor.* [20]

- The automation/autonomy shall allow the human operator to aid it in the selection of the appropriate level of automation for a given situation. [36]

- The automation/autonomy shall provide the means for the operator to ascertain the appropriate level of automation for a given situation and to then deploy it. [31]

- The automation/autonomy shall provide a means to toggle automation on or off that is quick and easy to use. [25]

6.8 Common Ground Requirements

6.8.1 Task Prioritization and Organization

- The automation/autonomy shall allow human operators to make connections between requests from different agents, both human and automated. [2]
- The automation/autonomy shall be able to make connections between requests from different agents, both human and automated. [2]
- The automation/autonomy shall support task prioritization as situations change. [2]
- The automation/autonomy shall allow human and automated team members to negotiate local goals in the context of a global goal or goals. [18]

6.8.2 Communicating Goals

- The automation/autonomy shall understand the goals of the human users. [20]
- The automation/autonomy shall communicate its intent in terms of what goals it is trying to accomplish for a given task. [20]
- The automation/autonomy shall have knowledge of other agents' (human and machine) intent. [1]

6.8.3 Sharing Progress towards Goals

- The automation/autonomy shall enable human operators to keep team members, including automated agents, informed of the status of team goals. [2]
- The system will keep humans and automated agents informed of the status of team goals. [2]
- The automation/autonomy shall allow a team of operators and agents to understand a "big picture" of the current situation. [2, 18]
- The automation/autonomy shall assist in understanding the status of the overall plan. [2, 18]

6.8.4 System Support between Members

- The automation/autonomy shall enable proactive backup between human and automated team members. [15]
- The automation/autonomy shall allow for critique of procedures and activities by both human and automated team members (e.g., identify errors, question improper procedures). [15]
- The automation/autonomy shall monitor metrics of human performance that indicate when human users are becoming overloaded. [20]

- The automation/autonomy shall monitor human work and gauge appropriately when to interrupt. [31]
- The automation/autonomy shall be able to detect limitations in what the operator can do. [10]
- The automation/autonomy shall be able to query the human operator. [10]
- The automation/autonomy shall recognize when to ask for help, and when help cannot be provided. [10]
- The automation/autonomy shall allow for critique of procedures and activities by both human and automated team members. See example §C.18 Errors and Procedures. [15, 20]

6.8.5 Shared Understanding of the Problem

- The automation/autonomy shall utilize visualizations that foster a shared representation of the problem space. See §6.10 Information Presentation. See example §C.19 Shared Representation [27]
- The automation/autonomy shall provide a visual representation of the problem that needs to be solved and the constraints that need to be met. See §6.10 Information Presentation. [27]

6.8.6 Understanding Roles

- The automation/autonomy shall understand its role and the tasks for which it is responsible. [20]
- The automation/autonomy shall understand the human's role and the tasks for which the human is responsible. [20]

6.8.7 System Capability Awareness and Communication

- The automation/autonomy shall have an awareness of its capabilities relative to the task context. [20, 34]
- The automation/autonomy shall communicate to the operator if its abilities do not match the task context. [20, 34]
- The automation/autonomy shall understand and communicate the degree of autonomy it is currently operating under. [20]
- The automation/autonomy shall be able to communicate its understanding of its environment of operation. [20]
- The automation/autonomy shall be able to communicate its physical limitations in a specific environment of operation. See §6.9 Calibrated Trust Requirements. [20]

- The automation/autonomy shall be able to communicate its analytical limitations. [20]
- The automation/autonomy shall be able to communicate awareness of temporal constraints. [20]
- The automation/autonomy shall provide information on history and environment. [9]
- Robotic systems shall provide spatial information about immediate surroundings *in order to determine areas of safe travel.* [9]
- The automation/autonomy shall be able to observe and interpret pertinent signals of human status and intentions. See example §C.20 UAV Controls. [18]

6.9 Calibrated Trust Requirements

6.9.1 Promote Understanding of the Automation

- The automation/autonomy shall be designed so that faults are predictable *to promote calibrated trust.* [19]
- The automation/autonomy shall convey system reliability in different contexts:
  - Situations in which the automation can be relied upon;
  - Situations that require increased oversight by personnel;
  - Situations in which the automation's performance is not acceptable. [15]
- The automation/autonomy shall provide context for past performances *to explain the reason for any performance decrements.* [19]

6.9.2 Prevent Under and Over Reliance

- The automation/autonomy shall minimize false alarms *in order to help operators develop calibrated trust.* [25, 21]
- The automation/autonomy shall adapt to the cognitive need of individuals *in order to reduce task load and prevent over and underreliance.*
- The interface shall provide information about the purpose, process, and performance of automation *to enhance the appropriateness of trust.* [19]
- The automation shall provide insight into how it performed calculations, made predictions, or made recommendations.
- The automation/autonomy shall convey the assumptions used in calculations or algorithms.

6.9.3 Shed Light on Information Sources

- The automation/autonomy shall allow the user to view information sources *to judge source reliability and promote calibrated trust.* [22]
- The automation/autonomy shall provide information about the certainty of information/sources. [22]
- The system shall provide details about when information was last updated.

6.10 Information Presentation

6.10.1 Combining Information Coherently

- The automation/autonomy shall synthesize information from multiple sources into a coherent picture for operators *in order to reduce cognitive load*. See example §C.21 Information Synthesis. [31, 36]
- The automation/autonomy shall communicate the competency and modality of the information source(s). See §6.9 Calibrated Trust Requirements. [34]
- The automation/autonomy shall provide data in context to other data of interest. [34]
- The automation/autonomy shall be designed in a way that reveals patterns and relationships among data. [34]

6.10.2 Reduce Cognitive Burden

- The automation/autonomy shall provide displays that humans can quickly scan to detect possible anomalies *without the operator having to engage in difficult cognitive work, such as calculations and integration, especially during time-pressured situations.* [5]
- The system interface shall present information that has already been integrated and calculated *to reduce the burden on the human.* [5]
- The automation/autonomy shall have a simple mechanism to enter and edit constraints on the problem solution *so that those actions are not an undue burden on the human.* [30]
- The automation/autonomy shall make it easy to visualize constraints on the problem solution. [30]
- The automation/autonomy shall minimize the use of multiple windows or screens. [36]
- The automation/autonomy shall support assessment and classification of situations. [19]

6.10.3 For Autonomy in Motion

- The automation/autonomy shall provide information about past activities. [36]
- The automation/autonomy shall provide information about past locations. [36]
- [for robotics] The automation/autonomy shall provide information about where the robot has been so the human does not need to track the robot's path in their heads. [36]
- The automation/autonomy shall provide information about the environment it is operating in. [36]
- The automation/autonomy shall display spatial information about where the vehicle/robot is in the environment. [36]

6.11 Design Process

6.11.1 Design for Appropriate Trust and Reliance

- The designer should design for appropriate trust rather than greater trust. See example §C.22 Auto Land System. [19]
- The designer should consider how context influences the operator's trust in the automation's authority. [19]
- The designer should consider that the final authority for some time-critical decisions should be allocated to the automation. [19]
- The designer should recognize the threat of overreliance and should understand the situations in which users could become over-reliant and the consequence of overreliance. [25]
- The designer should understand that the operator will rely upon the automation if it is reliable. [1]

6.11.2 Design for Appropriate Automation

- The designer should design constraints upon the automation's authority such that the automation always retains a subordinate position to the operator. [1]
- The designer should ensure that the automation will not foreclose upon the operator's authority to operate outside of the system's normal operating limits if it is required (e.g. safety). [1]
- The designer should ensure that the automation provides the appropriate range of control and management options to the operator. [1]
- The designers should keep human operators involved in an operation by requiring of them meaningful and relevant tasks, regardless of the level of management being utilized by them. [1]
- The designer should understand that human monitoring tends to be poor in work environments that do not conform to well-established ergonomics design principles, in high-workload situations, and in systems in which the automation is highly autonomous and there is little opportunity for manual experience with the automated tasks. [25]
- The designer should not automate because it is possible, instead the designer should give deep consideration to the tradeoffs involved in automating in order to determine what to automate and to what extent. [26]
- The designer should ensure that automation involving modes of control known to be potentially hazardous, contain safeguards against its use under inappropriate conditions. [1]

6.11.3 Design to Support the Operator Awareness, Work, and Goals

- The designer should be cognizant that integration of information does not mean simply adding more elements to a single display. [1]
- The designer should aim to lower cognitive load by providing fused information rather than disjointed data that the operator must mentally fuse. [9]
- The designer should provide the operator with assistance in choosing the appropriate level of automation modality at any given time. [9]

6.11.4 Design Philosophy and Rationale

- The design process should be iterative in learning and improvement so that the implementation and consequence of human teaming with that automation is understood and improved. [26]
- The designer should grasp the interdependencies within the human-agent team. [17]
- The designer should ensure the intended users are made aware of the automation's moral philosophy for its interactions with humans. [20]
- The designer should ensure the human is provided with a means to understand the automation's priorities and the accompanying general taxonomy of behaviors. [20]
- The designer should ensure that design philosophy and rationale is documented and available to users. [25]
- The designer should consider synergies and goal conflicts of the work shared by an automation-operator team. [3]
- The designer should recognize that an operator's practice will be transformed with the introduction of automation such that:
  - The design process should learn from these transformations;
  - The design concept is held as a hypothesis about the relationship between technology and human cognition/collaboration until it is rejected or accepted in the light of empirical evidence;
  - Notions of usefulness of the automation is tentative and open to revision during the design process as more is learned about the mutual shaping that occurs between the operator and automation in a field of practice. [8]
- The designer should not focus on what the technology cannot do and give the left-over functions to the operator; instead the designer should understand how roles adjust during iterative human-in-the-loop evaluations. [25]

7 Conclusion

The HMT Systems Engineering Guide was written to help system developers address HMT in the design of highly automated or autonomous systems. The Guide is organized into three major sections. Chapter 1 describes a framework of HMT and ten leverage points for supporting HMT in design. Chapters 2 through 5 present systems engineering processes for developing tailored HMT requirements for a specific system. The methods describe how to conduct interviews with subject matter experts, analyze the interview data, translate findings into requirements or user stories, and conduct a user survey to assess how well a system supports HMT. The descriptions include tips for use and printable guides. Chapter 6 presents a set of general requirements, derived from the literature, for supporting HMT through design. These general requirements are meant to serve as a starting point and to be tailored or expanded upon through use of the HMT system engineering methods. The end result of using this approach is a system that has constructive teaming between autonomy and humans, a system in which the autonomy and humans partner for enhanced mission results.

MITRE's mission-driven teams are dedicated to solving problems for a safer world. Through our federally funded R&D centers and public-private partnerships, we work across government to tackle challenges to the safety, stability, and well-being of our nation.

Human-Machine Teaming Interview Guide

Introductory Material

Paint Picture of Envisioned Autonomy

The envisioned system has autonomous features to... [tailor description]

Demographics and Top Challenges

1. Formal duty title, rank? Years/months experience in role?
2. Other relevant experience (training, previous positions, etc.)?
3. What are the top 3-5 tasks you're responsible for?
4. Which ones are the most difficult, cognitively?

Current State of Automation/Autonomy

1. What is the current state of automation, how is it presently implemented?
2. How do you use automation to do your job?
3. How does it not support you? In what ways it is unreliable or challenging?

Optional: Critical Decision Method Probe

1. Can you think of particularly challenging time when... [tailor situation]

HMT Knowledge Audit

Past and Future: Predictability, Exploring the Solution Space

1. For missions you'd accomplish with this system, how predictable or variable are they?
2. As you do this work, what is really critical to understand about what might happen next? Are you predicting the next few minutes, or is it hours, days, weeks?
3. Can you think of a time where you needed to understand, or have at your fingertips, historical information to understand what to do in the future? (e.g., you need to understand typical satellite movements to predict future movements or spot anomalies)
4. If you could have a tripwire or an automated alert that stood watch for you, is that a need? What would you want it to tell you? (e.g., you set an airfare alert so you are notified when the airfare from Denver to NY is under $300)

Big Picture: Observability

1. What's the overall battle rhythm, or decision timeline view, of this system? (e.g., does data needed for planning change continually throughout planning cycle?)
2. What do you want the automation to tell you as it's working for you?
3. Can you describe a time when you were confused about what the system was doing? (e.g., how the system calculated a recommendation)
4. What are the key vital signs to know that you are on track to accomplish the mission?
5. How might automation help you coordinate more effectively, if at all?

Anomalies: Calibrated Trust, Directing Attention, Adaptability

1. What are the biggest system anomalies you worry about?
2. Are there nuances that people pick up over time to know things are heading south?

3. Can you describe an instance when you spotted a deviation from the norm, or knew something was amiss?
4. Do you know of certain conditions in which your system typically provides unreliable information? (e.g., communications fail when going through a canyon)
5. Can you think of a time that you needed to improvise? Are there things the system could do to help you adapt?

Noticing: Directing Attention, Information Presentation

1. Can you think of a time when you missed something important that didn't pop out at you in a clear way?
2. Can you imagine a role for automation to direct other team members to see what you're looking at, to get a common frame of reference for the team?
3. Have there been times that you wish you were notified of new or changing information? Would that have made a difference in your decision making?

Self-monitoring: Common Ground, Calibrated Trust

1. Can you think of a time when you knew you were task saturated/overloaded and had to ask for help?
2. If a new person on your team were to take over your job, what would you be most concerned about? What part of your job would you feel most uneasy about if a brand-new person was doing it?
3. What indicators might you use to know that something's amiss with your automated partner's performance, or its intent, or assumptions? (e.g. the situation changed, and the system's algorithms aren't calibrated, or applicable)

Improvising: Adaptability, Directability

1. What are some example situations in which you have had to rapidly improvise a plan? (e.g., either to handle a threat or take advantage of an opportunity)
2. Using this example, how might your automated partner and you take advantage of this opportunity/handle this issue?
3. When the situation has changed, what might you need to understand and to direct the machine teammate's response to this change?

Job Smarts

1. If the government was giving you an intelligent robot/digital team member (like C-3PO), what are the top things you'd like it to do for you?
2. On the flip side, what would you not want your robot/digital team member to do?
3. If you had automation to help, what tribal, or local knowledge would it need to be aware of to be effective?
4. Are there administrative, information seeking, or representation tasks that your automated partner could do to make the team more efficient and effective? (e.g., rapid lookup of phone numbers for mission partners)

Heuristic Evaluation: Human-Machine Teaming

Please provide as much detail as possible in answering the questions below.

Observability

Observability means the system proactively communicates with you to let you know what it's thinking and doing, and tells you how far along it is in accomplishing your joint work. *Example: your GPS/directions app telling you that it's rerouting you because there is an accident ahead.*

1. I rate this system's Observability as: High, Medium, Low (Circle one)
2. What aspects of the system are observable?
3. What aspects of the system should have greater observability?

Predictability

Predictability means the system communicates with you about its intentions, goals, and future actions in various contexts. *Example: The green "armed" light on an auto-pilot control panel informs the pilot that when the approach glide path starts, a descent will automatically begin.*

1. I rate this system's Predictability as: High, Medium, Low (Circle one)
2. What aspects of the system offer good predictability?
3. What are aspects for which you need greater predictability?

Directing Attention

Directing attention is how a system notifies you and presents time-critical or task-critical information in a helpful way, so that you can stay aware of what's important and act proactively. The system should enable users to modify the notification settings. *Example: The auto-GCAS system informs a pilot that it is nearing an automatic ground avoidance maneuver by presenting chevrons on either side of the heads up display, directing a pilot's attention to the imminent situation.*

1. I rate this system's ability to Direct Attention as: High, Medium, Low? (Circle one)
2. What are ways that aspects of this system appropriately direct your attention?
3. What are ways you'd like aspects of this system to improve directing attention?

Exploring the Solution Space

Exploring the solution space means that the system helps problem solve by suggesting things to consider and offering alternative suggestions. *Example: A planning system asks if you have considered repositioning a certain ground asset.*

1. I rate this system's support to Exploring the Solution Space as: High, Medium, Low (Circle one)
2. What aspects of this system help you to explore the solution space?
3. How might aspects of the system improve in enabling you to explore the solution space?

Adaptability

Adaptability means that the system recognizes and can handle unplanned situations. *Example: A digital co-pilot is able to offer replanning based on a change in circumstances.*

1. I rate this system's Adaptability as: High, Medium, Low (Circle one)
2. What are aspects of the system that offer adaptability?
3. How might aspects of this system improve in adaptability?

Calibrated Trust

Supporting calibrated trust means that the system is clear about what it is good at and what its shortcomings are so you can determine how much and when to trust it. *Example: A targeting system notifies a user that it is 99% accurate in identifying targets over land, but misses 22% of targets over water.*

1. I rate this system's assistance in Calibrating Trust as: High, Medium, Low (Circle one)
2. What aspects of the system help you determine how much and when to trust it?
3. How might aspects of this system improve in supporting calibrated trust?

Common Ground

Common ground means that you and the system have a two-way street of communications about each other's intentions, beliefs, and assumptions. *Example: A planning tool has a window in which both the planner and the automated agent nominate the top 3 estimated threats so they are visible to each other.*

1. I rate this system's Common Ground as: High, Medium, Low (Circle one)
2. What aspects of this system enable or provide common ground?
3. What aspects of this system could be improved to enable better common ground?

Information Presentation

Information should be presented so that it is easily understandable and does not require unnecessary work to calculate or interpret. *Example: A weather display of an automated surveillance system shows the trend of humidity over time so that the human can quickly understand the present humidity in relation to past readings, and its rate of change.*

1. I rate this system's ability to Present Information as: High, Medium, Low? (Circle one)
2. What aspects of the system successfully present information?
3. What are ways you'd like this system to be improved to support information presentation?

Which three capabilities are the Most Relevant to the System? (Circle three)

Observability	Predictability	Directability
Directing Attention	Exploring the Solution Space	Adaptability
Calibrated Trust	Information Presentation	Common Ground

C Examples

C.1 Distinguishing Targets

The automation/autonomy has trouble distinguishing targets when the background is blue-grey water. There may be missed targets when searching over water. It is important to share this insight when a human first starts using the automation/autonomy.

Applicable Requirement (pg 34)

C.2 Autonomous Accuracy

The autonomous targeting system is 98% accurate in dry weather and 93% accurate in the rain. The reduced accuracy in the rain is due to a degraded ability to find targets.

Applicable Requirement (pg 34)

C.3 Autonomous Telescope

An autonomous telescope takes action to close its dome. The system notifies the user that the reason for the closure is high wind with a prediction of continued wind.

Applicable Requirement (pg 34)

C.4 Auto-GCAS

The Auto-GCAS (Ground Collision Avoidance System) is designed to autonomously pull up the airplane to prevent ground collision. Auto-GCAS informs the user that an autonomous intervention is pending by the display of chevrons that are moving together. The chevrons move closer as the autonomous intervention point approaches. At the last possible moment that a maneuver can be made, the tips of the chevrons touch and the maneuver is initiated. This is a simple way for the user to know how soon the autonomy will intervene.

Applicable Requirement (pg 34)

C.5 Advanced Planning System

A planning system with advanced algorithms allows operators to drill down into the details of a calculation to see the relative weighting of different factors.

Applicable Requirement (pg 34)

C.6 Autonomous Vehicle Notification

An autonomous ground vehicle notifies the human partner that a missing bridge will slow progress and prevent the vehicle from reaching the target location on time.

Applicable Requirement (pg 37)

C.7 Automated Cockpit Assistant

An automated cockpit assistant delivers a recommended flight path to the pilot and notifies the pilot that it did not consider the impact of wind.

Applicable Requirement (pg 37)

C.8 Satellite Deviation

Automation notifies a space operator that an adversary satellite is deviating from its normal pattern of life. The space operator can then drill down to understand the difference between the satellite's current behavior and normal behavior.

Applicable Requirement (pg 37)

C.9 Equipment Fault

There is a fault in a piece of equipment that only fails every 10 years and the removing the equipment is a complex process. The automated system alerts the human maintainer that this is a rare and complicated equipment failure and prompts the maintainer to review maintenance procedures.

Applicable Requirement (pg 37)

C.10 Interruptability

An automated assistant immediately cues an operator that a critical step in a life-critical process was missed. That same automated assistant waits 2 minutes until the life-critical process is complete to notify the human of new sensor information.

Applicable Requirement (pg 38)

C.11 Plan Locking

The human operator specifies the aspects of a plan that he or she does not want to change, or that practically cannot be changed, such as the location of specific equipment or forces.

Applicable Requirement (pg 39)

C.12 Plan Constraints

The human can specify planning constraints (parts of a plan that should not be changed) at the individual plan level and the global mission level.

Applicable Requirement (pg 40)

C.13 Weather Change

The weather changed rapidly and the human operator must simultaneously replan eight missions involving Unmanned Aerial Systems (UAVs). An autonomous assistant is aware of the load on the human operator and notifies the human that it will begin calculating UAV route changes for the operator to review.

Applicable Requirement (pg 40)

C.14 Degrading Gracefully

An example of not degrading gracefully is the communication link between an operator and an autonomous system abruptly failing without warning. Instead, the autonomous system could degrade gracefully by (1) alerting the operator to any advance indications of communication failure so that the operator can prepare and (2) providing projections of the system location and providing an estimate of when the communication link is likely to resume.

Applicable Requirement (pg 40)

C.15 Recovering Gracefully

An example of not recovering gracefully is for a system to reboot upon regaining communication with an autonomous vehicle. Instead, to recover gracefully, the system could provide an indication that the communication link will soon be restored to the human could prepare to intervene if needed.

Applicable Requirement (pg 40)

C.16 Sensor Prioritization

The system allows a human to prioritize which sensors are monitored. The human may want to alter the prioritization as the Commander's Critical Information Requirements change.

Applicable Requirement (pg 42)

C.17 Detailed Constraints

For the task of reconnaissance and surveillance of an area of interest, the human could specify regions in which stealth was of critical and could also specify the relative importance of persistent surveillance in different regions.

Applicable Requirement (pg 43)

C.18 Errors and Procedures

The automation identifies errors and questions improper procedures.

Applicable Requirement (pg 45)

C.19 Shared Representation

In a command and control environment, an autonomous digital assistant uses plan information to calculate the latest point at which the decision could be made to act and displays that decision point on a mission timeline.

Applicable Requirement (pg 45)

C.20 UAV Controls

The operator puts hands on the unmanned aerial vehicle controls and the autonomous system senses this and infers that the human is likely to take manual control.

Applicable Requirement (pg 46)

C.21 Information Synthesis

An automated system synthesizes information from multiple air and ground sensors to create an integrated picture of a remote location and situation.

Applicable Requirement (pg 48)

C.22 Auto Land System

A pilot using an auto-land system that is only certified for a crosswind of 10 knots should understand from the design of the system that the system should not be relied upon in situations where crosswinds are in excess of 10 knots.

Applicable Requirement (pg 49)

Resources

The following is a short list of additional resources on cognitive task analysis, user stories, and requirements. These resources are intended to serve as a starting point for obtaining more background. There are many other resources available online and in print.

[6] R. E Clark et al. *Cognitive Task Analysis*. 2006. URL: http://www.cogtech.usc.edu/publications/clark_etal_cognitive_task_analysis_chapter.pdf (visited on 09/26/2017).

[7] Beth Crandall, Gary Klein, and Robert R. Hoffman. *Working Minds: A Practitioner's Guide to Cognitive Task Analysis*. 1 edition. Cambridge, Mass: A Bradford Book, July 7, 2006. 332 pp. ISBN: 978-0-262-53281-5.

[13] Robert R Hoffman, Beth Crandall, and Gary Klein. *Protocols for cognitive task analysis*. Pensacola, FL: Florida Institute for Human and Machine Cognition, 2005. URL: http://www.dtic.mil/docs/citations/ADA475456 (visited on 09/26/2017).

[29] J. M Schraagen, S. F Chipman, and V. L Shalin. *Cognitive Task Analysis*. Psychology Press, 2000. URL: http://www.dtic.mil/dtic/tr/fulltext/u2/a387792.pdf.

[33] Karl E. Wiegers. "Writing quality requirements." In: *Software Development* 7.5 (1999), pp. 44–48. URL: http://vincentvanrooijen.com/container/Requirements%20Management/karl%20wiegers%20-%20writing%20quality%20requirements.pdf (visited on 09/28/2017).

[35] World Bank Institute Evaluation Group. *Cognitive Task Analysis: Needs Assessment Knowledge Base*. 2007. URL: http://siteresources.worldbank.org/WBI/Resources/213798-1194538727144/3Final-Cog_Task_Analysis.pdf (visited on 09/20/2017).

[37] Wayne Zachary et al. ""Rapidized" cognitive task analysis." In: *IEEE Intelligent Systems* 27.2 (2012), pp. 61–66. URL: http://ieeexplore.ieee.org/abstract/document/6212527/ (visited on 09/26/2017).

References

[1] Charles E. Billings. *Aviation automation: The search for a human-centered approach*. 1997. URL: https://trid.trb.org/view.aspx?id=572335 (visited on 07/17/2017).

[2] Jeffrey M. Bradshaw et al. "Teamwork-centered autonomy for extended human-agent interaction in space applications." In: *AAAI 2004 Spring Symposium*. 2004, pp. 22–24. URL: http://www.aaai.org/Papers/Symposia/Spring/2004/SS-04-03/SS04-03-026.pdf (visited on 07/14/2017).

[3] Jeffrey M. Bradshaw et al. "The Seven Deadly Myths of 'Autonomous Systems'." In: *IEEE Intelligent Systems* 28.3 (2013), pp. 54–61. URL: http://ieeexplore.ieee.org/abstract/document/6588858/ (visited on 07/17/2017).

[4] Jessie Y. Chen et al. *Situation awareness-based agent transparency*. Army research lab Aberdeen proving ground MD human research and engineering directorate, 2014. URL: http://www.dtic.mil/docs/citations/ADA600351 (visited on 07/26/2017).

[5] Klaus Christoffersen and David D. Woods. "How to make automated systems team players." In: *Advances in human performance and cognitive engineering research*. Emerald Group Publishing Limited, 2002, pp. 1–12. URL: http://www.emeraldinsight.com/doi/abs/10.1016/S1479-3601(02)02003-9 (visited on 07/17/2017).

[6] R. E Clark et al. *Cognitive Task Analysis*. 2006. URL: http://www.cogtech.usc.edu/publications/clark_etal_cognitive_task_analysis_chapter.pdf (visited on 09/26/2017).

[7] Beth Crandall, Gary Klein, and Robert R. Hoffman. *Working Minds: A Practitioner's Guide to Cognitive Task Analysis*. 1 edition. Cambridge, Mass: A Bradford Book, July 7, 2006. 332 pp. ISBN: 978-0-262-53281-5.

[8] Sidney WA Dekker and David D. Woods. "MABA-MABA or abracadabra? Progress on human–automation co-ordination." In: *Cognition, Technology & Work* 4.4 (2002), pp. 240–244. URL: http://www.springerlink.com/index/q2legq6x4cvw4wtw.pdf (visited on 07/17/2017).

[9] Jill L. Drury et al. "Design guidelines for improved human-robot interaction." In: *CHI'04 extended abstracts on Human factors in computing systems*. ACM, 2004, pp. 1540–1540. URL: http://dl.acm.org/citation.cfm?id=986116 (visited on 07/17/2017).

[10] Terrence Fong, Charles Thorpe, and Charles Baur. "Collaboration, dialogue, human-robot interaction." In: *Robotics Research*. Springer, 2003, pp. 255–266. URL: https://link.springer.com/chapter/10.1007/3-540-36460-9_17 (visited on 07/17/2017).

[11] Michael A. Goodrich and Alan C. Schultz. "Human-robot interaction: a survey." In: *Foundations and trends in human-computer interaction* 1.3 (2007), pp. 203–275. URL: http://dl.acm.org/citation.cfm?id=1348100 (visited on 07/17/2017).

[12] Kevin Anthony Hoff and Masooda Bashir. "Trust in automation: Integrating empirical evidence on factors that influence trust." In: *Human Factors* 57.3 (2015), pp. 407–434. URL: http://journals.sagepub.com/doi/abs/10.1177/0018720814547570 (visited on 07/14/2017).

[13] Robert R Hoffman, Beth Crandall, and Gary Klein. *Protocols for cognitive task analysis*. Pensacola, FL: Florida Institute for Human and Machine Cognition, 2005. URL: http://www.dtic.mil/docs/citations/ADA475456 (visited on 09/26/2017).

[14] Robert R. Hoffman et al. "The dynamics of trust in cyberdomains." In: *IEEE Intelligent Systems* 24.6 (2009). URL: http://ieeexplore.ieee.org/abstract/document/5372194/ (visited on 07/17/2017).

[15] J. Joe et al. "Identifying requirements for effective human-automation teamwork." In: *Proceedings of the 12th international conference on probabilistic safety assessment and management (PSAM 12, Paper# 371),(INL/CON-14-31340)*. 2014. URL: http://psam12.org/proceedings/paper/paper_370_1.pdf (visited on 07/14/2017).

[16] Matthew Johnson et al. "Seven cardinal virtues of human-machine teamwork: examples from the DARPA robotic challenge." In: *IEEE Intelligent Systems* 29.6 (2014), pp. 74–80. URL: http://ieeexplore.ieee.org/abstract/document/6982119/ (visited on 07/14/2017).

[17] Matthew Johnson et al. "The fundamental principle of coactive design: Interdependence must shape autonomy." In: *Coordination, organizations, institutions, and norms in agent systems VI* (2011), pp. 172–191. URL: http://link.springer.com/content/pdf/10.1007/978-3-642-21268-0.pdf#page=187 (visited on 07/17/2017).

[18] Gary Klein et al. "Ten challenges for making automation a" team player" in joint human-agent activity." In: *IEEE Intelligent Systems* 19.6 (2004), pp. 91–95. URL: http://ieeexplore.ieee.org/abstract/document/1363742/ (visited on 07/14/2017).

[19] John D. Lee and Katrina A. See. "Trust in automation: Designing for appropriate reliance." In: *Human factors* 46.1 (2004), pp. 50–80. URL: http://journals.sagepub.com/doi/abs/10.1518/hfes.46.1.50_30392 (visited on 07/14/2017).

[20] Joseph B. Lyons. "Being transparent about transparency: A model for human-robot interaction." In: *2013 AAAI Spring Symposium Series*. 2013. URL: http://www.aaai.org/ocs/index.php/SSS/SSS13/paper/viewPaper/5712 (visited on 07/14/2017).

[21] Joseph B. Lyons et al. "Trust-based analysis of an air force collision avoidance system." In: *ergonomics in design* 24.1 (2016), pp. 9–12. URL: http://journals.sagepub.com/doi/abs/10.1177/1064804615611274 (visited on 07/17/2017).

[22] Poornima Madhavan and Douglas A. Wiegmann. "Similarities and differences between human–human and human–automation trust: an integrative review." In: *Theoretical Issues in Ergonomics Science* 8.4 (2007), pp. 277–301. URL: http://www.tandfonline.com/doi/abs/10.1080/14639220500337708 (visited on 07/17/2017).

[23] Patricia McDermott et al. "Quenching the Thirst for Human-Machine Teaming Guidance: Helping Military Systems Acquisition Leverage Cognitive Engineering Research." In: *Proceedings of the Naturalistic Decision Making Conference*. Naturalistic Decision Making Conference. 2017.

[24] L. G. Militello and R. J. Hutton. "Applied cognitive task analysis (ACTA): a practitioner's toolkit for understanding cognitive task demands." In: *Ergonomics* 41.11 (Nov. 1998), pp. 1618–1641. ISSN: 0014-0139. DOI: 10.1080/001401398186108.

[25] Raja Parasuraman and Victor Riley. "Humans and automation: Use, misuse, disuse, abuse." In: *Human factors* 39.2 (1997), pp. 230–253. URL: http://journals.sagepub.com/doi/abs/10.1518/001872097778543886 (visited on 07/14/2017).

[26] Raja Parasuraman, Thomas B. Sheridan, and Christopher D. Wickens. "A model for types and levels of human interaction with automation." In: *IEEE Transactions on systems, man, and cybernetics-Part A: Systems and Humans* 30.3 (2000), pp. 286–297. URL: http://ieeexplore.ieee.org/abstract/document/844354/ (visited on 07/17/2017).

[27] E. M Roth et al. "Designing Collaborative Planning Systems: Putting Joint Cognitive Systems Principles to Practice." In: P Smith and R.R Hoffman. *Cognitive Systems Engineering: The Future for a Changing World*. Ashgate Publishing Ltd, 2016.

[28] Kristin E. Schaefer et al. *A meta-analysis of factors influencing the development of trust in automation: Implications for human-robot interaction*. Army research lab Aberdeen proving ground MD human research and engineering directorate, 2014. URL: http://www.dtic.mil/docs/citations/ADA607926 (visited on 07/17/2017).

[29] J. M Schraagen, S. F Chipman, and V. L Shalin. *Cognitive Task Analysis*. Psychology Press, 2000. URL: http://www.dtic.mil/dtic/tr/fulltext/u2/a387792.pdf.

[30] Ronald Scott et al. "Externalizing planning constraints for more effective joint human-automation planning." In: *conference proceedings of the 12th international naturalistic decision making conference, held June*. 2015, pp. 9–12. URL: http://www2.mitre.org/public/ndm/papers/ScottRossDepassWampler041415.pdf (visited on 07/17/2017).

[31] Aaron Steinfeld et al. "Common Metrics for Human-robot Interaction." In: *Proceedings of the 1st ACM SIGCHI/SIGART Conference on Human-robot Interaction*. HRI '06. New York, NY, USA: ACM, 2006, pp. 33–40. ISBN: 978-1-59593-294-5. DOI: 10.1145/1121241.1121249. URL: http://doi.acm.org/10.1145/1121241.1121249 (visited on 07/17/2017).

[32] Robert Truxler et al. "Designing collaborative automated planners for agile adaptation to dynamic change." In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. Vol. 56. SAGE, 2012, pp. 223–227. URL: http://journals.sagepub.com/doi/abs/10.1177/1071181312561065 (visited on 07/17/2017).

[33] Karl E. Wiegers. "Writing quality requirements." In: *Software Development* 7.5 (1999), pp. 44–48. URL: http://vincentvanrooijen.com/container/Requirements%20Management/karl%20wiegers%20-%20writing%20quality%20requirements.pdf (visited on 09/28/2017).

[34] David Woods. "Generic support requirements for cognitive work: laws that govern cognitive work in action." In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. Vol. 49. SAGE, 2005, pp. 317–321. URL: http://journals.sagepub.com/doi/abs/10.1177/154193120504900322 (visited on 07/17/2017).

[35] World Bank Institute Evaluation Group. *Cognitive Task Analysis: Needs Assessment Knowledge Base*. 2007. URL: http://siteresources.worldbank.org/WBI/Resources/213798-1194538727144/3Final-Cog_Task_Analysis.pdf (visited on 09/20/2017).

[36] Holly A. Yanco, Jill L. Drury, and Jean Scholtz. "Beyond usability evaluation: Analysis of human-robot interaction at a major robotics competition." In: *Human–Computer Interaction* 19.1 (2004), pp. 117–149. URL: http://www.tandfonline.com/doi/abs/10.1080/07370024.2004.9667342 (visited on 07/17/2017).

[37] Wayne Zachary et al. ""Rapidized" cognitive task analysis." In: *IEEE Intelligent Systems* 27.2 (2012), pp. 61–66. URL: http://ieeexplore.ieee.org/abstract/document/6212527/ (visited on 09/26/2017).