Department for Science, Innovation & Technology

Government Digital Service

AI Insights

Integrated Agents

Automation isn’t just about efficiency - it is also about making better decisions and better use of our time. Integrated agents, which range from artificial intelligence (AI) driven assistants to user-configured automation tools, are increasingly embedded in our daily software.

They are designed to reduce tedious effort and free up time by handling tasks like drafting documents, managing spreadsheets, handling meeting schedules, and summarising video calls. These AI tools can help us to improve accuracy, reduce human error, and support decision-making through responsive, real-time data analysis.

Here we explain some different types of agents, the guidelines for their use, and the measures that should be in place to ensure they operate safely and effectively.

Agent types

A very brief look at three typical agent types for clarification:

AI agents

These are what we consider agentic AI. These are intelligent AI agents that can perform tasks autonomously using registered tools and functions. They are backed by a large language model (LLM) which helps in planning and routing.

Software agents

These are traditional automation tools, or those found in libraries that simulate large models, or models with large numbers of simulated entities. They are not usually AI-based.

Agent interfaces

Also known as agent entities. These are the ones we will discuss here. They are a means of interacting with a backend system or suites of services, and provide a common visual interface in a low-code/no-code environment. They allow non-technical users to configure complex behaviours that might be scheduled, or are called when specific events occur.

In addition to the above agent types, vendors are adding extensive agent interfaces to their products. Users can select and configure pre-built entities to perform everyday tasks.

It is now also possible to create bespoke entities in many vendor systems that will handle repetitive tasks with a very high degree of autonomy. While such user-defined entities are extremely useful and convenient, they also demand careful consideration in their design.

Equally welcome amongst these developments is an increasing tendency to replace or augment traditional application programming interfaces (APIs) with agent interfaces. APIs are a mechanism used to allow systems to communicate with each other. They can become frustratingly complex and difficult to code against.

Providing an agent interface as an intermediary can greatly simplify the interaction between systems, making it easier and faster to achieve your objectives.

Risks and Concerns

As the Department for Science, Innovation and Technology, our highest priority concerns with agent interfaces are:

- Data leakage
  The automation of aggregating and communicating data is certainly high on the list of risks. Our concerns are for our users' data. There are many negative consequences - this is to be protected against at all times.

- Responsible design
  We must ensure that no critical decisions are made by these entities without human expert insight.

- Shadow IT
  We wish to avoid a tranche of unauthorised software entities performing unknown operations with user data. Non-compliance with regulations or uncontrolled proliferation of unsanctioned AI endpoints are undesirable outcomes.

These are reasons enough to clarify why we may wish to audit, govern, and control end-user defined entities. It is not that we expect malicious behaviour, but rather to protect our citizens, our users, and our teams at all times.

Entity Guidelines

The ability to build bespoke AI systems is a tremendous advancement. They are either entirely out of our control, as in the API case where vendors decide their capabilities or, if we are building our own AI systems, almost entirely within our control and governed by software engineering principles in an established and agreed process

End users configuring and creating their own entity processes to consume and potentially disseminate end user data are generally not within our control and require a similar set of agreed and responsible processes.

These systems allow a user to define a process flow which will be handled by the agent process. The user can describe in plain, natural language the behaviours that are expected of the agent. The back-end system interprets those instructions in order to achieve the desired outcome. They are both powerful and flexible, but there are some important matters to consider:

Responsible entity creation

When custom entities are created to automate tasks, care and attention must be given to ensuring that responsibilities are not automated away to the extent that entities are making decisions that should be made by qualified people.

Automating appropriately

It is strongly recommended that any person or team's initial exploration in the creation of user-defined entities are restrained in their ambition to simple, measurable and predictable tasks.

Business justification

It is great to learn something new, and to refine people’s skills with new technologies, but there is a time and place. There should be a valid business justification for creating these entities in a production environment and they should be created through an agreed, understood, and transparent process.

Management

A change in back-end systems can lead to a change in interpretive or predictive capabilities, which in turn can lead to a change in the behaviour of our entities. Simpler entities are less vulnerable to these changes, and can more easily be updated if back-end models change the agent behaviour.

Validation

Prior to running your entity on live or production data, it should be validated as safe to do so. It may be necessary to ensure that anything that communicates data should be validated before going live. For example, it may be justifiably forbidden to send automatically generated reports to an email alias or group, as the user has no way of knowing the full membership at all times.

Confirmation

Having ensured that the entity is complying with the rules, guidelines and guardrails we have established, it should be necessary to get written approval and confirmation from a manager or responsible officer prior to deployment or activation.

Inventory

A signed-off entity should be added to the internal inventory with a detailed description of its behaviours. Administrators should know at a glance the number of user-defined agent processes in their estate, and exactly what they do.

Monitoring

Continuous vigilance is the price of peace-of-mind in this instance. Once your bespoke system begins to interact with your live production data, certain responsibilities must be assumed. Monitoring behaviour is vital, but not always trivial. Audits should be part of this monitoring process and we should assure that entities behave as described and are configured within policy boundaries.

Policies and procedures will vary by team, department and organisation as appropriate. Our intention is to harness these new tools appropriately and as intended - to support users through intelligent automation, to save time and to reduce tedious and often wasteful manual effort. Done properly, they can be a tremendous benefit to people. However, we must do so responsibly, safely and securely.