In my initial post, I stated that the recent rise of agent-based systems across various sectors is attributed to their concept of autonomous agents interacting to achieve common goals. These systems enhance adaptability and agility in response to market changes, allowing organisations to test various decisions and adjust strategies accordingly. By leveraging distributed computing resources, they streamline operations, improve collaboration, and boost productivity. Agent-based systems excel in handling manufacturing-related tasks, from design to supply chain management, while promoting flexibility and efficiency through communication and cooperation among agents. Their benefits include improved responsiveness, cost reduction, enhanced decision-making, and gaining a competitive edge in complex business environments. (Adeniyi, 2023).

Moreover, to correct an error in my initial post, The rise of agent-based systems can be attributed to the increasing solutions to the complexity of modern business environments and the numerous benefits this approach can offer organisations. (Jennings, 2000)

(Feaviour, 2023) examples of agent-based approaches in the supply chain industry include a multi-agent system for real-time make-to-order customer order planning with multiple distributed sites. Agents at each site collaborate with planner agents to ensure on-time delivery and overall customer satisfaction, improving capacity planning, cost savings, cash flow, and customer satisfaction compared to conventional software.

(Kananavelu, 2013) states how AI agents in banking have increased personalised services for customers and employees, reduced costs through automation, improved efficiency, and decreased errors. Also how they have revealed business opportunities by generating insights from data. Banks that have yet to embrace AI have faced competition, lost customers, and fallen behind in the industry.

To conclude this post, (Jennings, 2000) made an excellent case for agent-oriented software engineering, stating that the agent-oriented approach involves encapsulated computer systems called agents that can autonomously act in a specific environment to achieve their objectives. Agents have well-defined boundaries, interfaces, and flexibility in problem-solving behaviour. Multiple agents are often required to represent decentralised problems and interact with each other to manage dependencies and achieve individual objectives. A high-level communication language facilitates agent interactions and requires flexibility to adapt to unforeseen situations in their environment. This approach enables complex problem-solving in software engineering by considering multiple perspectives, competing interests, and interactions dynamically and context-dependently.

References:

Jennings, R. (2002). Building Complex Software Systems: The Case for an Agent-Based Approach. Available from:https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=4c4e39f657e5a8543890e47611e08615018e25e2[Accessed 25 June 2023].

Feaviour, L. (2023). Initial Post. Available https://www.my-course.co.uk/mod/forum/discuss.php?d=156390#p236500 [Accessed 19 July 2023].

Kanakavelu, N. (2023). Initial Post. Available https://www.my-course.co.uk/mod/forum/discuss.php?d=156841#p237922 [Accessed 19 July 2023].