# Complete Agent Interactions Guide (Excerpt)

## All Agents in the System

1. ManagerAgent (Orchestrator)

* Role: Central coordinator and decision maker
* Responsibilities: Workflow orchestration, task delegation, context management, result aggregation

1. QueryUnderstandingAgent

* Role: Natural language processor
* Responsibilities: Parse user queries, extract intent, entities, and query type classification

1. Data Retrieval Agent

* Role: Data access coordinator
* Responsibilities: Interface with semantic layer, execute SQL queries, fetch API data

1. DataAnalysisAgent

* Role: Analytics processor
* Responsibilities: Perform trend detection, comparisons, correlations, explanations

1. AnomalyDetectionAgent

* Role: Data quality monitor
* Responsibilities: Detect outliers, flag issues, provide anomaly explanations

1. ReportGenerationAgent

* Role: Output formatter
* Responsibilities: Create visualizations, format reports (PDF, HTML, charts)

1. FeedbackAgent

* Role: Learning coordinator
* Responsibilities: Collect user feedback, log interactions, update context/memory

## Agent Interaction Matrix

ManagerAgent → Other Agents (Outbound Calls):

|  |  |  |  |
| --- | --- | --- | --- |
| Target Agent | When Called | Purpose | Data Passed |
| QueryUnderstandingAgent | On every user query | Parse intent, entities, query type | Raw user query, context |
| Data Retrieval Agent | When data is needed (after intent parsing) | Fetch required data | Structured query parameters, filters |
| DataAnalysisAgent | When query requires analytics, explanation | Perform analysis | Data + analysis requirements |
| AnomalyDetectionAgent | When checking for anomalies or after analysis | Health checks, outlier detection | Dataset + detection parameters |
| ReportGenerationAgent | When user requests reports, visualizations | Generate formatted output | Analysis results + format requirements |
| FeedbackAgent | After delivering answer/report | Collect user satisfaction data | Response + user interaction data |

Other Agents → ManagerAgent (Return Calls):

|  |  |  |
| --- | --- | --- |
| Source Agent | When Reports Back | Data Returned |
| QueryUnderstandingAgent | After parsing user query | Intent, entities, query type, confidence scores |
| Data Retrieval Agent | After fetching from BigQuery/APIs | Raw or preprocessed data, metadata |
| DataAnalysisAgent | After performing analytics/comparisons | Analysis results, insights, trends |
| AnomalyDetectionAgent | After checking for anomalies/outliers | Anomaly flags, explanations, severity |
| ReportGenerationAgent | After generating report/visualization | Formatted report, charts, download links |
| FeedbackAgent | After collecting user feedback | Feedback summary, context updates |

## Inter-Agent Communications (Non-Manager)

|  |  |  |  |
| --- | --- | --- | --- |
| Caller | Called | When | Purpose |
| DataAnalysisAgent | AnomalyDetectionAgent | During advanced analytics workflow | Check for anomalies as part of analysis |
| ReportGenerationAgent | DataAnalysisAgent | When creating comprehensive reports | Include additional analysis in reports |

## Detailed Interaction Flows

1. 1. Standard Query Flow

## 4. Example Query Flow: Efficiency Drop in US (Agentic Flow)

Example Query: "Can you please generate a report stating the cause of efficiency drop in country US. Please give supporting plots or tables."

1. 1. User → ManagerAgent: Receives the query.
2. 2. ManagerAgent → QueryUnderstandingAgent: Parses intent (causal analysis, report, plots/tables), extracts "efficiency drop," "country US."
3. 3. ManagerAgent → Data Retrieval Agent: Uses the semantic layer to generate and run SQL for efficiency data (and possibly weather/operational data) for the US.
4. 4. ManagerAgent → DataAnalysisAgent: Performs trend analysis, detects when/where the drop happened, and investigates possible causes (e.g., weather, anomalies). May use AnomalyDetectionAgent for statistical confirmation.
5. 5. DataAnalysisAgent → AnomalyDetectionAgent (if needed): Checks for outliers or significant deviations.
6. 6. ManagerAgent → ReportGenerationAgent: Receives analysis results and supporting data, generates plots (e.g., efficiency over time, weather overlay), tables, and a narrative report.
7. 7. ManagerAgent → User: Delivers the final report and plots/tables.
8. 8. ManagerAgent → FeedbackAgent: Collects user feedback, updates context/memory for future queries.