Absolutely! Here's a **complete breakdown of the MetaFlow-SDF framework** with:

✅ Detailed **data flow**  
✅ **Input/output** data types per stage  
✅ How components are **connected**  
✅ What setup a team needs to **adopt and implement** it

**🧠 MetaFlow-SDF Workflow with Data Flow, Inputs & Outputs**

**🔁 High-Level Process Overview (Simplified Loop)**

1. **Sprint Planning**
2. **Task Assignment**
3. **Kanban Execution (Progress Monitoring)**
4. **Development & Code Integration**
5. **CI/CD Execution (with Self-Healing)**
6. **Security & Compliance Validation**
7. **Deployment**

All stages are **driven by Agentic AI**, and each stage's **output becomes the next stage’s input.**

**🔄 Stage-by-Stage Flow: Input, Output, and Connection**

**1️⃣ AI Sprint Planning**

📌 **Agent**: Sprint Manager AI  
📥 **Input Data**:

* Backlog Tasks (JSON / DB)
* Project Priority Rules (YAML/JSON)
* Developer Profiles (CSV/DB)
* Historical Sprint Velocity (CSV/JSON)

📤 **Output**:

* Optimized Sprint Backlog
* Estimated Sprint Timeline
* Task Prioritization Map

✅ **Feeds into → Task Allocator AI**

**2️⃣ AI Task Assignment**

📌 **Agent**: Task Allocator AI  
📥 **Input Data**:

* Sprint Backlog (from Step 1)
* Developer Skill Matrix (JSON)
* Real-time Load/Availability (API or Manual Sync)

📤 **Output**:

* Task-to-Developer Map (JSON)
* Dynamic Load Distribution Dashboard

✅ **Feeds into → Kanban Board / Progress Flow AI**

**3️⃣ Kanban Board Automation**

📌 **Agent**: Kanban Flow Optimizer AI  
📥 **Input Data**:

* Task Assignment Logs
* Dev Tools Activity (e.g., GitHub/GitLab commits)
* CI/CD Progress Signals

📤 **Output**:

* Auto-Updated Task Status
* Dependency Alerts
* Performance Heatmap

✅ **Feeds into → Dev Execution & CI/CD**

**4️⃣ Development + Code Integration**

📌 **Human & AI Hybrid**  
📥 **Input Data**:

* Assigned Tasks
* API Specs / Mockups
* Shared Repositories (Git)

📤 **Output**:

* Committed Code
* Pull Requests (PRs)
* Code Change Logs

✅ **Feeds into → CI/CD Pipeline AI**

**5️⃣ CI/CD Execution with Self-Healing**

📌 **Agent**: CI/CD Failure Resolver AI  
📥 **Input Data**:

* Build/Test Logs
* Deployment Configs
* Historical Failure Logs

📤 **Output**:

* Clean Builds
* Error Resolutions (Auto Fix Suggestions or Patches)
* Rollback Logs (if failed)

✅ **Feeds into → Compliance Validation**

**s**

**🔌 How They Are Connected**

Sprint Planning --> Task Assignment

↓ ↓

Kanban Board ← Development

↓ ↓

CI/CD Pipeline ← Code Changes

↓

Blockchain Validator

↓

Deployment

**🛠️ What Setup Does a Team Need to Adopt MetaFlow-SDF?**

**🔧 Team Setup Checklist**

| **Area** | **Tooling Needed** | **Setup** |
| --- | --- | --- |
| **Project Backlog & Planning** | Jira, Notion, CSV, Airtable | Export/Sync tasks |
| **AI Sprint & Task Management** | Custom Agentic AI setup using LangChain / CrewAI / AutoGen | Pre-trained models or connected to LLM APIs |
| **Code Repositories** | GitHub / GitLab | Enable Webhooks for triggers |
| **CI/CD Pipelines** | Jenkins, GitLab CI/CD, GitHub Actions | Add logs API or webhook triggers |
| **Self-Healing Layer** | LangChain Agents + Failure Pattern DB | Train on past CI logs |
| **Kanban Visualization** | Streamlit, React + FastAPI | Optional UI layer |
| **Blockchain Ledger** | Ethereum, Hyperledger, Smart Contract | For compliance history |

**✅ How a Team Can Adopt It (Step-by-Step)**

1. **Start Small**: Connect MetaFlow agents to your GitHub & Jira.
2. **Enable Auto-Sprint Planning**: Feed historical sprint data to Sprint Manager AI.
3. **Build Agentic Layer**: Use LangChain or CrewAI to create modular agents.
4. **Automate CI/CD Fixes**: Use error pattern matching for auto-resolves.
5. **Add Blockchain Ledger** (Optional): Track approvals, logs, releases.
6. **Scale to Full Flow**: Integrate all agents into one continuous loop.

**🧩 Final Note: Input & Output Data Types (At a Glance)**

| **Stage** | **Input Format** | **Output Format** |
| --- | --- | --- |
| Sprint Planning | JSON / CSV | JSON |
| Task Assignment | JSON | JSON |
| Kanban Automation | API / Log Stream | Web UI or JSON |
| CI/CD Logs | TXT / JSON | JSON Fix Logs |
| Blockchain Compliance | YAML / JSON | Signed Hash & Ledger Log |
| Deployment | Build Package | Deployment Log, Status JSON |

Would you like me to generate a **fully illustrated PDF version** of this flow as well? Or a **configurable YAML template** to kick off your team’s adoption?