IETF Hackathon
Agent Communications & Agent Gateway

Kehan Yao, Feiyu Jiang, Jiacheng Zhao, Peng Liu, Zongpeng Du

IETF 125
14–15 March 2026
Shenzhen, China

Hackathon Plan

- What problems were you working on?
  - **Cross-domain** Agent **group communication**s and **skill-based** routing via Agent Gateway.
  - **Dynamic** tools registration and online invocation via Agent Gateway.

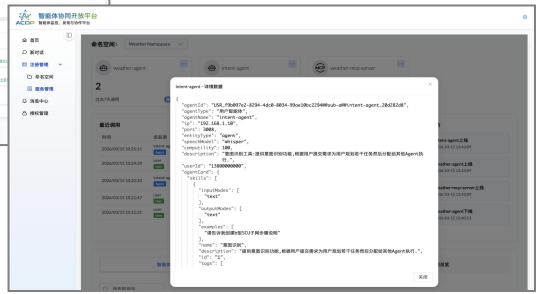
  (Tools quick onboarding without pre-configuration and shutdown/reboot of Agent client services)
  - **In-parallel** tools invocation via Agent Gateway.

  (Simultaneously invoke tools to decrease workflow completion time.)

What got done

- Build a **platform** for **Agents, Tools, and Models** registration and discovery, within an administrative domain.

Platform Portal



Agent Profile

- Build **Agent Gateway(AGW)** prototypes for Agent information synchronization, implementing agent grouping channels, and MCP servers registration and traverse

```
private Message handleDataMessage(Message message, ChannelHandlerContext ctx) {
    String channel = message.getHeader() != null ? message.getHeader().getChannelName() : "N/A";
    String senderId = message.getHeader() != null ? message.getHeader().getAddressLocator() : "unknown";

    Log.info("📨 Data message for channel: {} from: {}", channel, senderId);

    if (channel != null && channel.startsWith("private:")) {
        handlePrivateMessage(message, senderId, channel);
    } else if (channel != null && channel.startsWith("skill:")) {
        handleSkillMessage(message, senderId, channel);
    } else {
        messageRouter.broadcastToChannel(channel, message, senderId);
    }

    return MessageBuilder.createResponse(message, "DATA_ACK");
}
```

Group Communication Basice Primitives

```
switch (message.getMessageType()) {
    case CONNECTION_REQUEST:
        response = handleConnectionRequest(message, ctx);
        break;
    case HEARTBEAT:
        response = handleHeartbeat(message, ctx);
        break;
    case DATA:
        response = handleDataMessage(message, ctx);
        break;
    case CHANNEL_JOIN:
        response = handleChannelJoin(message, ctx);
        break;
    case CHANNEL_LEAVE:
        response = handleChannelLeave(message, ctx);
        break;
    case ROUTE_UPDATE:
        response = handleAgwRouteMessage(message, ctx);
        break;
    case SKILL_REGISTER:
        response = handleSkillRegister(message, ctx);
        break;
    case SKILL_QUERY:
        response = handleSkillQuery(message, ctx);
        break;
    case AGENT_DISCOVER:
        response = handleAgentDiscover(message, ctx);
        break;
    default:
        Log.warn("Unknown message type: {}", message.getMessageType());
        response = MessageBuilder.createErrorResponse(message,
                "Unsupported message type", 400);
}
```

AGW Actions

What we learned and Next Steps

- Lessons learned from this hackathon
  - **Platform-level:** Cross-domain interactions involve information synchronization and trust management across administrative platforms.
  - **Network-level:** Complex skill-based routing across domains require network component like AGW for state handling, policy control, and traffic management, and secure interaction.
  - **Protocol-level:**
    - some protocol extensions can be handled by WGs like OAuth (e.g, token management and **authorization** for in-parallel MCP tools invocation)
    - **Next steps** for implementation:
      - session protocols for better QoS and state management
      - DNS-based discovery to improve scalability.

Wrap Up

Team members:
Kehan Yao,
Feiyu Jiang,
Jiacheng Zhao,
Peng Liu,
Zongpeng Du,

Contact Us:
yaokehan@chinamobile.com,
jiangfeiyu@chinamobile.com,
zhaojiacheng@chinamobile.com,
liupengyjy@chinamobile.com,
duzongpeng@chinamobile.com