aMP Lyon 2026

17 Novembre 2026

Copilot ne connaît pas votre métier ? Spécialisez-le avec les agents déclaratifs



Benjamin Talmard



Alexis Conia

Présenté par

#aMPLyon

Merci à nos sponsors

LOCAUX

orange Business

Expertime
Digital Success Partner

GOLD

deep

CoreView

SILVER

Synapsys

PowerSyncPro™

A.C.F.

BRONZE

MsCollab

aMP #aMPLyon

Speakers



Benjamin Talmard

Consultant
Indépendant



Alexis Conia

Consultant
Indépendant

aMP #aMPLyon

Agenda

1

ÉPISODE 1

Copilot, un harness généraliste

Puissant, mais pas encore métier

Démo : Agent Builder

2

ÉPISODE 2

L'agent déclaratif

Vous déclarez, Copilot exécute

Démo : Agent Simple

3

ÉPISODE 3

Actions, MCP et plugins

Brancher et distribuer le métier

Démo : Agent avancé

aMP #aMPLyon

Copilot + Agents + Plugins

Copilot



Copilot

Un assistant pour chaque collaborateur

+

Agents & Plugins



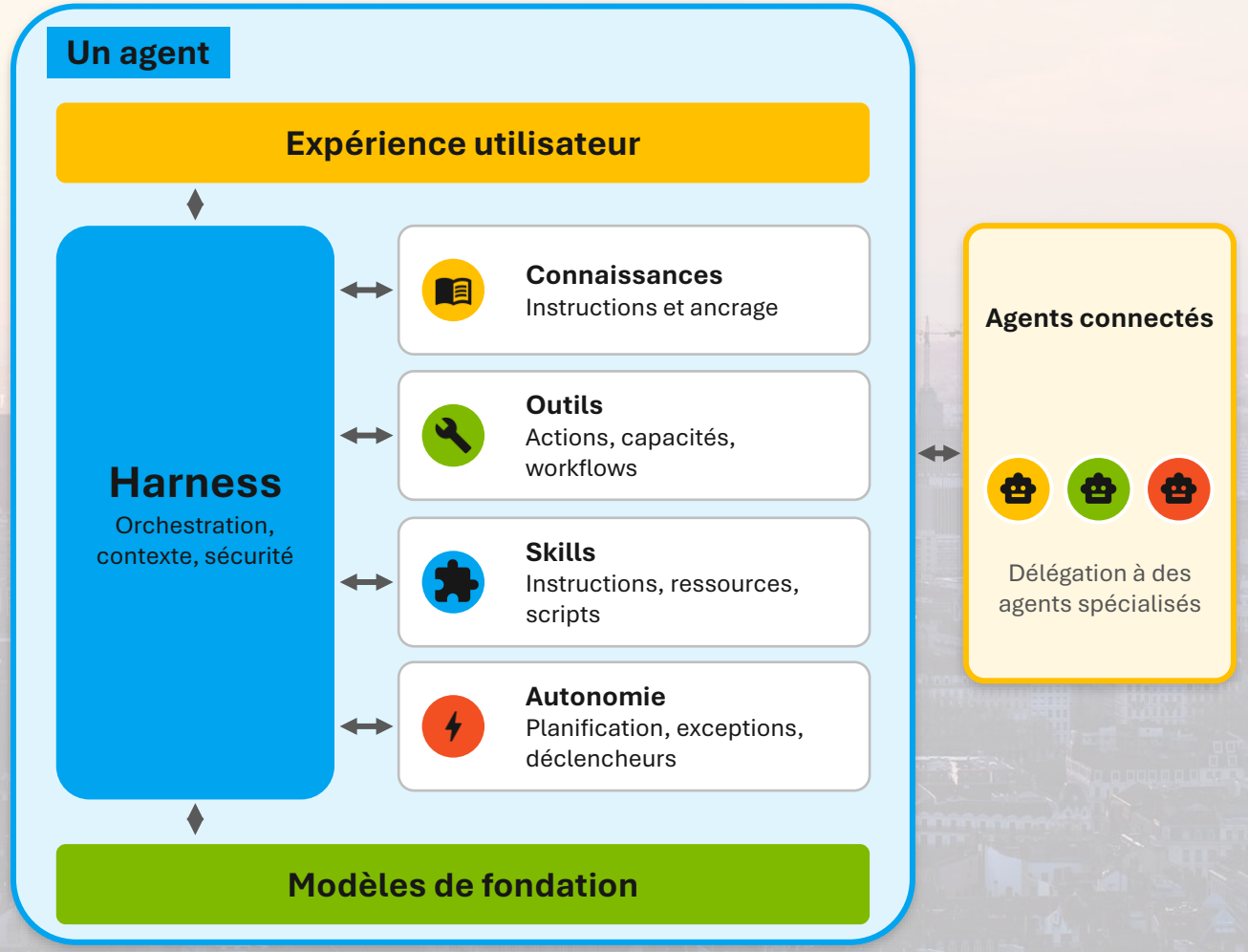
Agents & Plugins

Des agents pour chaque métier, distribués sous forme de plugins

Architecture d'un agent

Le modèle raisonne.

Le harness orchestre : il assemble le contexte, choisit les outils, gère les exceptions et applique la sécurité.



Le harness, c'est quoi ?

Modèle (LLM)
Raisonne et génère du texte

+

Harness
Le runtime autour du modèle : il pilote, outille et sécurise

=

Agent
Un assistant qui agit dans votre contexte

Ce que fait le harness

1. **Orchestration**
   Planifie, appelle le modèle, exécute, recommence
2. **Contexte**
   Instructions, connaissances, mémoire, historique
3. **Outils**
   Actions API, connecteurs, serveurs MCP
4. **Sécurité**
   Identité de l'utilisateur, permissions, approbations
5. **Expérience**
   Copilot Chat, Teams, Word, Excel, PowerPoint...

« Le harness décide quand appeler le modèle, ce qu'il lui envoie, interprète la réponse et appelle les bons outils. » — d'après Microsoft Learn

L'orchestrateur n'est qu'une pièce du harness : sa boucle de décision. Le harness ajoute le contexte, les outils, la sécurité et l'interface.

Un harness par usage

Étendre Copilot Chat

Copilot chat harness

Ajoute vos connaissances, instructions et actions à Microsoft 365 Copilot. C'est celui des agents déclaratifs.

Ex. : un agent d'onboarding sur SharePoint

Cette session

Parcours maîtrisés

Standard harness

Topics, règles et chemins définis par le maker : des réponses prévisibles.

Ex. : un helpdesk interne

Processus multi-étapes

GitHub Copilot harness

Raisonne vers un objectif, découpe en étapes, se corrige. Gère fichiers Office, skills, mémoire et sandbox.

Ex. : rapprochement factures / bons de commande

Côté pro-code, Microsoft Agent Framework fournit aussi son « agent harness » : boucle d'outils, mémoire, compaction du contexte, approbations.

Source : Microsoft Learn, « Harnesses in Copilot Studio » ; blog Agent Framework, Build 2026

Créer un agent ou un plugin

No code → Pro code

Utilisateurs



Agent Builder

Dans Copilot, en langage naturel

Makers



Copilot Studio

Low code, connecteurs, flux, choix du harness

Développeurs



Agents Toolkit · Agents SDK · WIQD

VS Code, CLI, packaging en plugin

Microsoft Foundry

Modèles et agents pro-code hébergés sur Azure



Démonstration

Agent Builder

aMP #aMPLyon

1. Copilot, un harness
2. L'agent déclaratif
3. Actions, MCP et plugins


2
Épisode 2
L'agent déclaratif
Vous déclarez, Copilot exécute
aMP
#aMPLyon
1. Copilot, un harness
2. L'agent déclaratif
3. Actions, MCP et plugins

Agents déclaratifs



Déclaratif

Vous décrivez l'agent en JSON : instructions, connaissances, actions, skills.

Capacités

Recherche web, génération d'images, interpréteur de code

Actions

Vos systèmes via API REST, serveurs MCP, add-ins Office

Sécurité

SSO Entra ID, OAuth ou clé API, avec l'identité de l'utilisateur

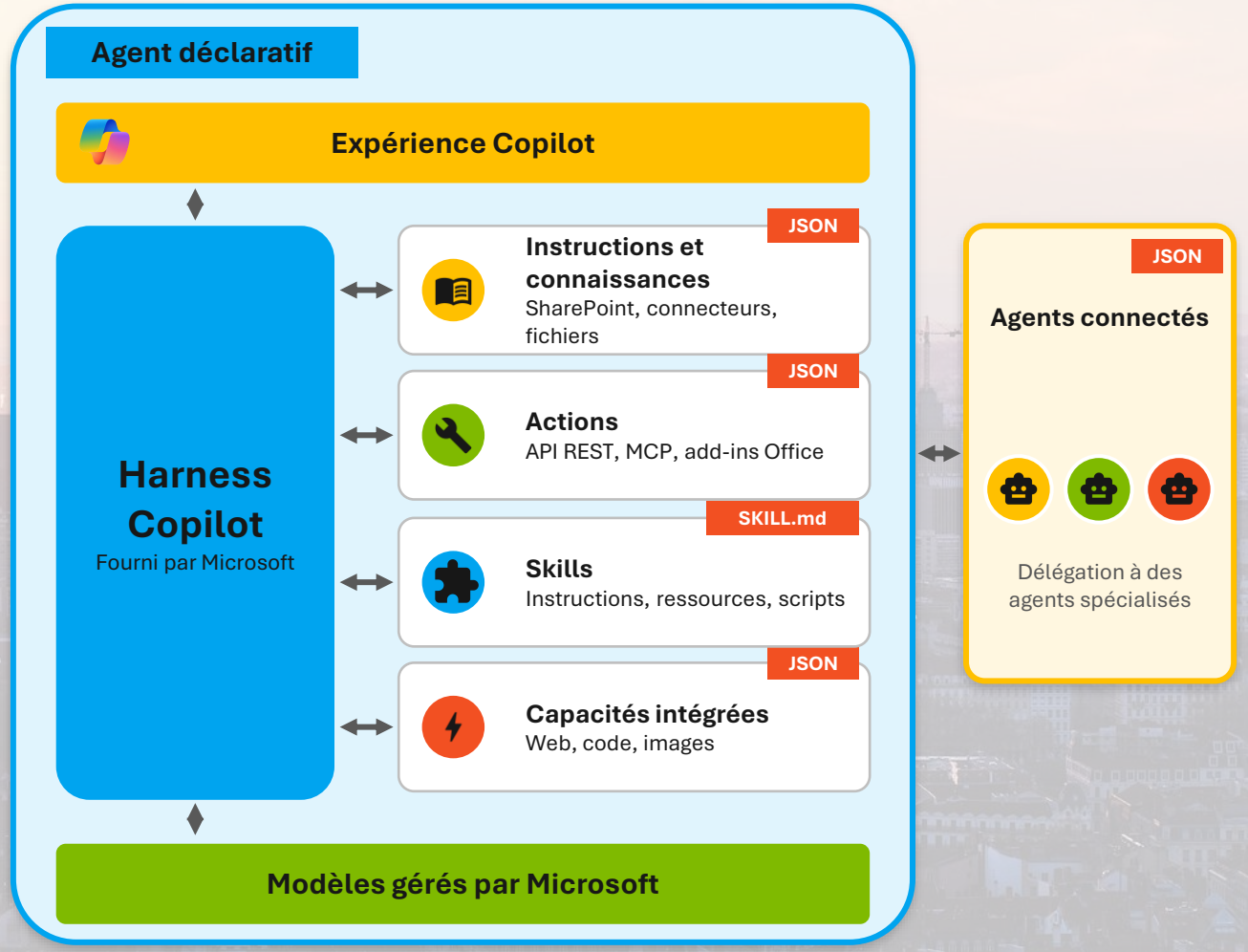
Copilot fournit le harness (modèle, orchestration, sécurité, interface) : vous apportez le métier

Architecture d’un agent déclaratif

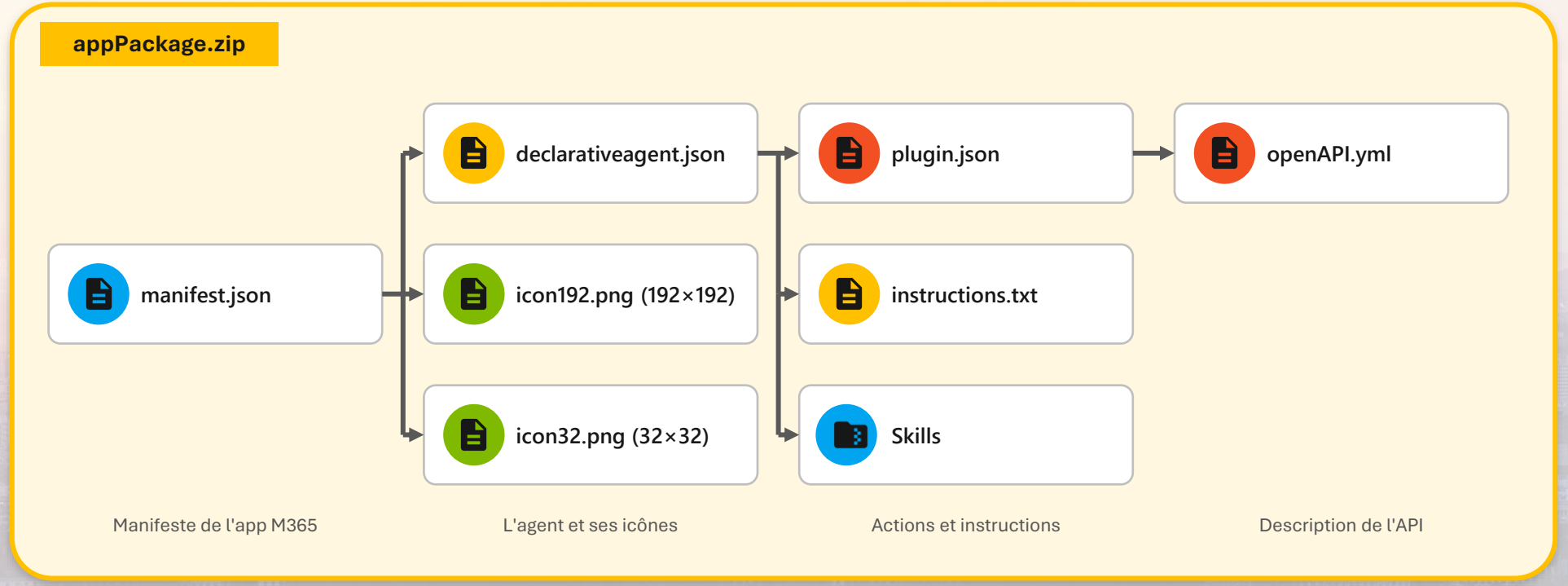
Qui fait quoi ?

JSON Déclaré par vous dans le package

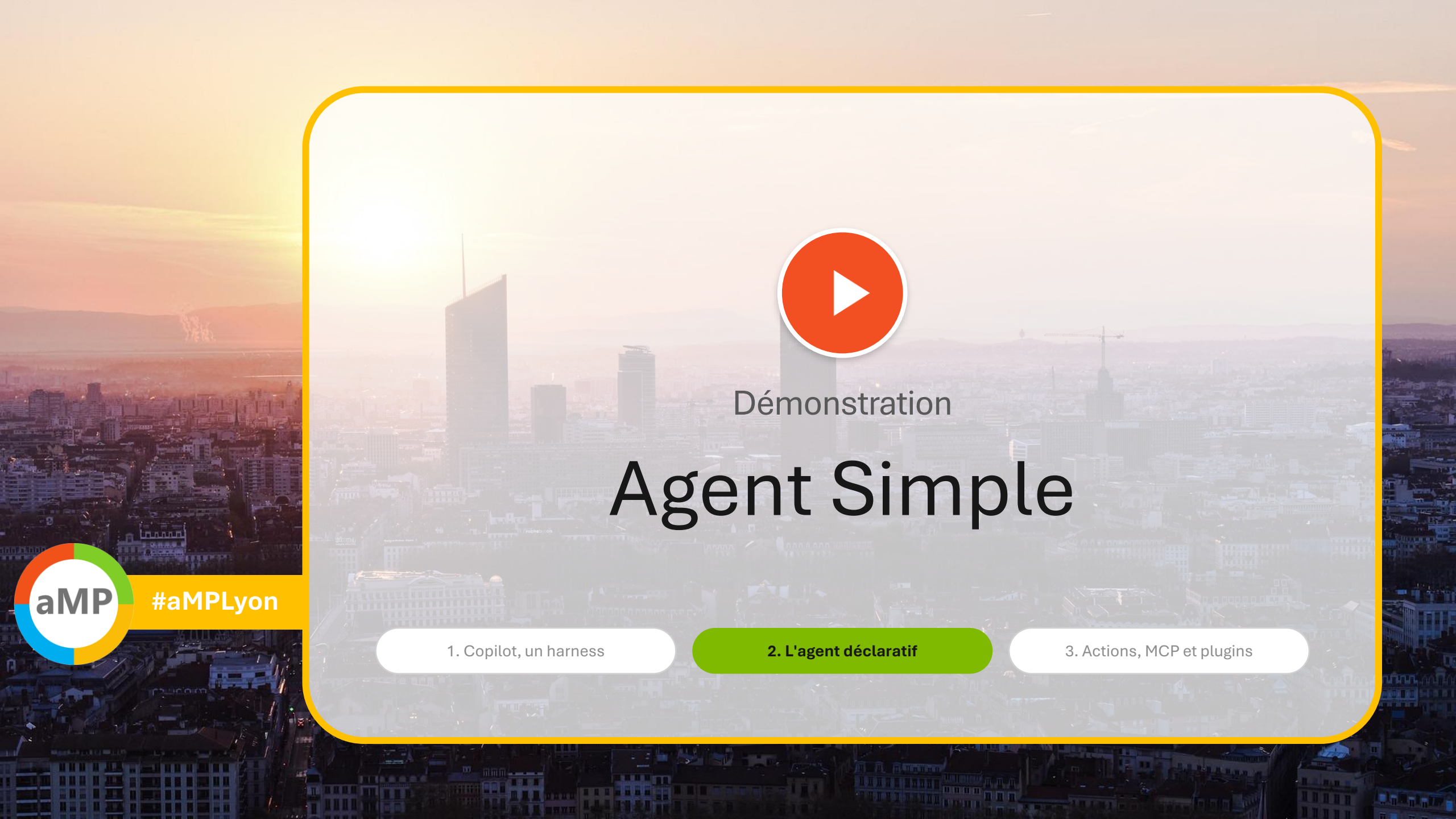
Fourni par Microsoft 365 Copilot



Anatomie d’un agent déclaratif



Attention au vocabulaire : plugin.json décrit une action de l'agent (l'ancien « API plugin »). Le plugin Microsoft Copilot est autre chose : le package qui distribue agents, skills, connecteurs et serveurs MCP (épisode 3).



Démonstration

Agent Simple

aMP #aMPLyon

1. Copilot, un harness
2. **L'agent déclaratif**
3. Actions, MCP et plugins



3

Épisode 3

Actions, MCP et plugins

Brancher et distribuer le métier

aMP #aMPLyon

1. Copilot, un harness
2. L'agent déclaratif
3. **Actions, MCP et plugins**

Plugins Microsoft Copilot

Un package, quatre capacités

Publié dans le registre de plugins, puis disponible dans Copilot Chat, Cowork, Word, Excel, PowerPoint et SharePoint

Agent déclaratif

Un assistant ciblé : instructions, connaissances, outils

Skill

Un savoir-faire réutilisable, décrit dans un SKILL.md

Connecteur Copilot

Vos contenus métier rendus accessibles à Copilot

Serveur MCP

Les outils de vos applications métier

Un format que le marché adopte

Standard ouvert

Agent Skills

Un dossier avec un SKILL.md : instructions, scripts, ressources. Le même fichier fonctionne dans Claude Code, GitHub Copilot, VS Code, Gemini CLI, Cursor… et Microsoft 365 Copilot.

Août 2026

Agent Plugins 1.0

Un format commun pour packager skills et serveurs MCP. Comité technique : Amazon, Cursor, Microsoft, OpenAI, Vercel. Lu par GitHub Copilot, VS Code, Codex, Cursor, Kiro.

Sept. – nov. 2026

Plugins Microsoft Copilot

Registre gouverné dans le centre d'admin Microsoft 365 et Agent 365. Le bouton « Plugins » remplace « Agents » dans Copilot. Import et export de plugins Claude avec WIQD.

Skills et serveurs MCP voyagent d'un harness à l'autre ; chaque harness décide de ce qu'il sait exécuter (« one package does not mean one runtime »).

Sources : MC1484008 (registre de plugins), Microsoft Learn (plugins, Cowork), code.visualstudio.com (agent plugins)

Agents déclaratifs : nouveautés

Manifeste v1.6

Multi-agents

Interagir avec d'autres agents

En préversion

Manifeste v1.6

Fichiers

Fichiers embarqués dans le package comme base de connaissances

Mars 2026

MCP Apps

Support des MCP Apps / OpenAI Apps

Mai 2026

Évaluations

Évaluer l'agent avec la CLI Agent Evaluations et générer un rapport

v1.7 · Mai 2026

Réponses

Questions-réponses prédéfinies et mode de réponse par défaut

v1.8 · Juillet 2026

E-mail et agenda

Trier et envoyer des e-mails, planifier des réunions, sondages de créneaux

v1.9 · Octobre 2026

Skills

Jusqu'à 8 skills (SKILL.md) embarqués dans le package

v1.9 · Octobre 2026

Fabric

Ancrer les réponses dans les données Microsoft Fabric



Démonstration

Skill et MCP

aMP #aMPLyon

1. Copilot, un harness
2. L'agent déclaratif
3. **Actions, MCP et plugins**

À retenir

1

Copilot est un harness

Vous ne le reconstruisez pas : vous le spécialisez avec votre métier.

2

L'agent déclaratif apporte le métier

Instructions, connaissances et actions (API, MCP) décrites en JSON, exécutées par Copilot.

3

Le plugin devient l'unité de distribution

Agents, skills, connecteurs et MCP dans un même package, publié et gouverné par l'IT.

Merci !

Notre session vous a inspiré ?

Merci de nous laisser votre feedback.

Pour être informé des événements à venir, suivez-nous sur Linkedin !

Feedback

LinkedIn

#aMPLyon