



---

≡ Experiments > Onboarding and Config Protocol

---

## Experiments

# Onboarding and Config Protocol

Purpose: shared onboarding + config surfaces across CLI, macOS app, and Web UI.

## Components

Wizard engine (shared session + prompts + onboarding state).

CLI onboarding uses the same wizard flow as the UI clients.

Gateway RPC exposes wizard + config schema endpoints.

macOS onboarding uses the wizard step model.

Web UI renders config forms from JSON Schema + UI hints.

## Gateway RPC

```
wizard.start  params:  { mode?: "local"|"remote", workspace?: string }  
wizard.next   params:  { sessionId, answer?: { stepId, value? } }  
wizard.cancel params:  { sessionId }  
wizard.status params:  { sessionId }  
config.schema params:  {}
```

### Responses (shape)

```
Wizard:  { sessionId, done, step?, status?, error? }
```



Config schema: { schema, uiHints, version, generatedAt }

---

## UI Hints >

uiHints keyed by path; optional metadata  
(label/help/group/order/advanced/sensitive/placeholder).  
Sensitive fields render as password inputs; no redaction layer.  
Unsupported schema nodes fall back to the raw JSON editor.

## Notes

This doc is the single place to track protocol refactors for  
onboarding/config.

---

< Tests

Cron Add Hardening >

Powered by [mintlify](#)