User Interface - Feature #12190

client device bridge: reach devices attached to the user's workstation

11/16/2026 03:37 PM - Greg Shah

Status:	New	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:			
billable:	No	**reviewer:**	
vendor_id:	GCD	**production:**	No
case_num:		**env_name:**	
version_reported:		**topics:**	
version_resolved:			

Description

History

#1 - 11/16/2026 03:38 PM - Greg Shah

- Project changed from Base Language to User Interface

Problem

Point-of-sale and counter applications write directly to devices attached to the workstation: raw printer output to receipt and label printers (ESC/POS, ZPL, vendor languages), cash drawer kicks through the receipt printer, serial ports, and TCP connections to local payment middleware:

```
OUTPUT TO PRINTER VALUE("ReceiptPrinter").
PUT CONTROL CHR(27) + "p" + CHR(0) + CHR(25) + CHR(250).
OUTPUT CLOSE.

CREATE SOCKET hSocket.
hSocket:CONNECT("-H localhost -S 2011").
```

In FWD's web client the 4GL code runs on the server. Printer output currently becomes a PDF in the browser, and a socket to localhost reaches the server. A browser cannot write raw bytes to a local printer or open arbitrary TCP connections, so these devices are currently unreachable. A small agent on the workstation, paired with the user's FWD session, can relay this traffic.

Work

1. **Workstation agent.** A small Java service installed on the workstation that:
   - writes raw bytes to a named local printer queue (bypassing the driver's rendering) or to a serial port;
   - opens TCP connections to local services on request and relays data in both directions;
   - lists its available devices.
2. **Pairing and identity.** Associate the agent with the user's FWD web client session securely (for example a pairing token issued by the server and passed through the browser, or a per-workstation certificate). Identify the workstation, so that device configuration can be per workstation rather than per user.
3. **Transport.** The agent connects out to the FWD server (no inbound ports on the workstation), with TLS and reconnection. Traffic is only accepted for the paired session.
4. **Runtime routing (configurable per device name):**
   - OUTPUT TO PRINTER VALUE(name) and OUTPUT TO VALUE(port) for configured names are sent raw to the agent instead of producing a PDF;
   - SOCKET:CONNECT to configured hosts and ports (for example localhost:2011) is relayed through the agent, with socket events (READ-RESPONSE) behaving as for a direct connection;
   - names not configured keep the current behaviour.
5. **Configuration.** Device mappings in directory.xml, per workstation, group or user.
6. **Installer and updates.** A signed installer for Windows workstations, automatic start, and version checks against the server.
7. **Security.** Restrict which devices and hosts the agent can reach; log every relayed connection.

8. **Documentation** for administrators: installation, pairing, device configuration and troubleshooting.

The scope (which device types and transports) is to be confirmed against the project's device inventory before implementation.

Testcases

- Raw ESC/POS output (text, cut, drawer kick) to a USB receipt printer through the agent.
- ZPL recall of a stored format on a networked label printer through the agent.
- A socket conversation with a local TCP service, including READ-RESPONSE events, timeouts and a dropped connection.
- Unpaired agent, wrong session, device not in the allowed list.
- Agent restart and network interruption during a print job.