

Open Source Technologies in Safety-critical Medical Device Platforms

Using Open Source to Design Connected Medical Devices to Help Fill EHRs with Clinically Useful Data

Shahid N. Shah, CEO



Netspective
EXTENDING THE ENTERPRISE

O'REILLY*

OSCON

open source convention

Who is Shahid?



- 20+ years of software engineering and multi-site healthcare system deployment experience
- 12+ years of healthcare IT and medical devices experience (blog at <http://healthcareguy.com>)
- 15+ years of technology management experience (government, non-profit, commercial)
- 10+ years as architect, engineer, and implementation manager on various EMR and EHR initiatives (commercial and non-profit)

THE BUSINESS OF MEDICAL PRACTICE

TRANSFORMATIONAL HEALTH 2.0
SKILLS FOR DOCTORS
THIRD EDITION

David Edward Marcinko
Editor-in-Chief

Hope Rachel Heticco
Managing Editor



SPRINGER PUBLISHING COMPANY

*Author of Chapter 13,
“You’re the CIO of your Own
Office”*



Netspective

O'REILLY

OSCON
OPEN SOURCE CONVENTION

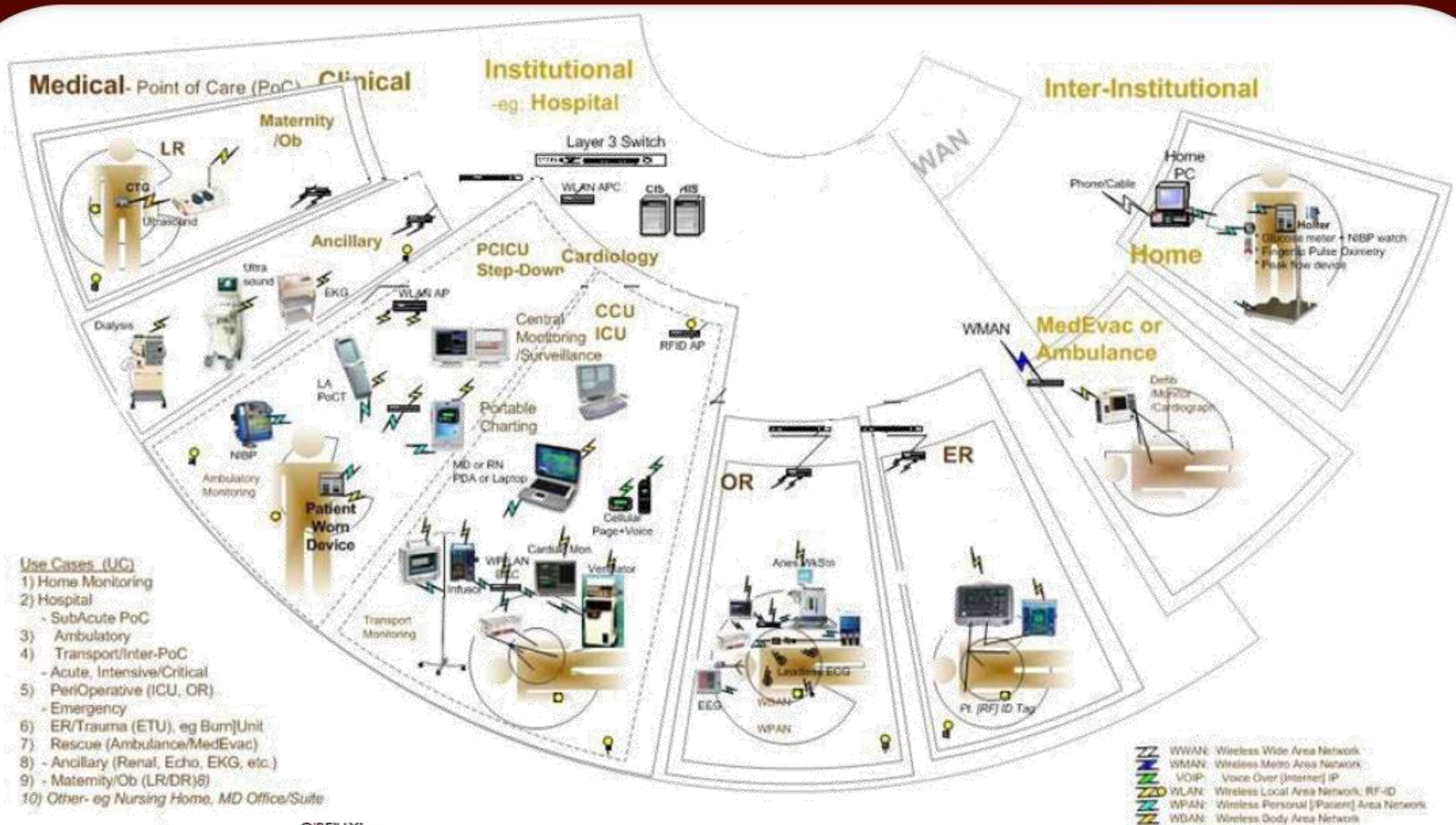
www.netspective.com

Healthcare landscape background

- The government (through Meaningful Use & ACO incentives) is paying for the collection of clinical data.
- Medical devices are the best sources of quantifiable, analyzable, and reportable clinical data.
- Most medical devices today are not connected so you do not have access to the best data.
- New devices are being design and deployed to support connectivity.



What if we had access to all this data?



Where does patient data come from?

	Patient	Health Professional	Labs & Diagnostics	Medical Devices
Source	Self reported by patient	Observations by HCP	Computed from specimens	Computed real-time from patient
Unstructured Data	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
Errors	High	Medium	Low	
Time	Slow	Slow	Medium	
Reliability	Low	Medium	High	
Data size	Small	Small	Large	
Availability	Common	Common	Common	Uncommon



Where does patient data come from?

	Patient	Health Professional	Labs & Diagnostics	Medical Devices
Source	Self reported by patient	Observations by HCP	Computed from specimens	Computed real-time from patient
Structured Data	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Errors	High	Medium	Low	Low
Time	Slow	Slow	Medium	Fast
Reliability	Low	Medium	High	High
Discrete size	Small	Small	Small	Small
Streaming size				Large
Availability	Uncommon	Common	Somewhat Common	Uncommon



Patient data source analysis

- Meaningful Use and CER advocates are promoting (structured) data collection for reduction of medical errors, analysis of treatments and procedures, and research for new methods.
- All the existing MU incentives promote the wrong kinds of collection: unreliable, slow, and error prone.
- Accurate, real-time, data is only available from connected medical devices



Connectivity is a must, OSS is answer

Most obvious benefit



Least attention



Most promising
capability



Netspective

O'REILLY

OSCON
OPEN SOURCE CONVENTION

www.netspective.com

This talk focuses on
connected devices

Key OSS questions

Will the FDA accept open source in safety-critical systems?

Are open source systems safe enough for medical devices?



Simple answer

Yes!

Proof: we did it at American Red Cross in 1996



It's not as hard as we think...

- Modern real-time operating systems (open source and commercial) are reliable for safety-critical medical-grade requirements.
- Open standards such as TCP/IP, DDS, HTTP, and XMPP can pull vendors out of the 1980's and into the 1990's. 😊
- Open source and open standards that promote enterprise IT connectivity can pull vendors into the 2010's and beyond.



But it's not easy either...we need

Risk
Assessments

Hazard Analysis

Design for
Testability

Design for
Simulations

Documentation

Traceability

Mathematical
Proofs

Determinism

Instrumentation

Theoretical
foundations



OSS hazard and risk assessment

- What is the intended use for the device or system?
- How will the OSS product you're planning to use going to be tied to your intended use?
- What is the risk associated with the OSS product for that particular intended use?

$$R = S_h \times P_h$$



Risk is related to severity and harm

$$R = S_h \times P_h$$

R = risk

S_h = severity of harm

P_h = probability of harm

- **Harm** is damage done to a person
- **Severity** is the degree of harm done
- **Probability** is the frequency and duration of exposure



Examples of Severity & Probability

Severity

- multiple fatalities
- fatalities
- severe injury (non-reversible, requires hospitalization)
- moderate injury (reversible, requires hospitalization)
- minor (reversible, requires first aid)
- very minor (no first aid)

Probability

- Constant exposure
- Hourly
- Daily
- Weekly
- Monthly
- Yearly
- Never



Formal risk assessment methods

What-if analysis

Preliminary
hazard analysis
(PHA)

Failure modes
and effects
analysis (FMEA)

Fault tree
analysis (FTA)

Hazard and
operability
studies



OSS Risk analysis steps - FMEA

- Define the function of the OSS product being analyzed.
- Identify potential failures of the OSS.
- Determine the causes of each failure types.
- Determine the effects of potential failures.
- Assign a risk index to each of the failure types.
- Determine the most appropriate corrective/preventive actions.
- Monitor the implementation of the corrective/preventive to ensure that it is having the desired effect.



Good summary of FMEA

- [http://en.wikipedia.org/wiki/
Failure mode and effects analysis](http://en.wikipedia.org/wiki/Failure_mode_and_effects_analysis)



Sampling of OSS / open standards

Project / Standard	Subject area	D	G	Comments
Linux or Android	Operating system	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
OMG DDS (data distribution service)	Publish and subscribe messaging	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Open standard with open source implementations
AppWeb, Apache	Web/app server	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
OpenTSDB	Time series database		<input checked="" type="checkbox"/>	Open source project
Mirth	HL7 messaging engine		<input checked="" type="checkbox"/>	Built on Mule ESB
Alembic Aurion	HIE, message exchange		<input checked="" type="checkbox"/>	Successor to CONNECT
HTML5, XMPP, JSON	Various areas	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Don't reinvent the wheel
SAML, XACML	Security and privacy	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
DynObj, OSGi, JPF	Plugin frameworks	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Build for extensibility



OSS applicability to connectivity

Physical

- Wired, wireless (WiFi, cellular, etc.)

Logical

- Device → Gateway → Data Routers → Systems

Structural

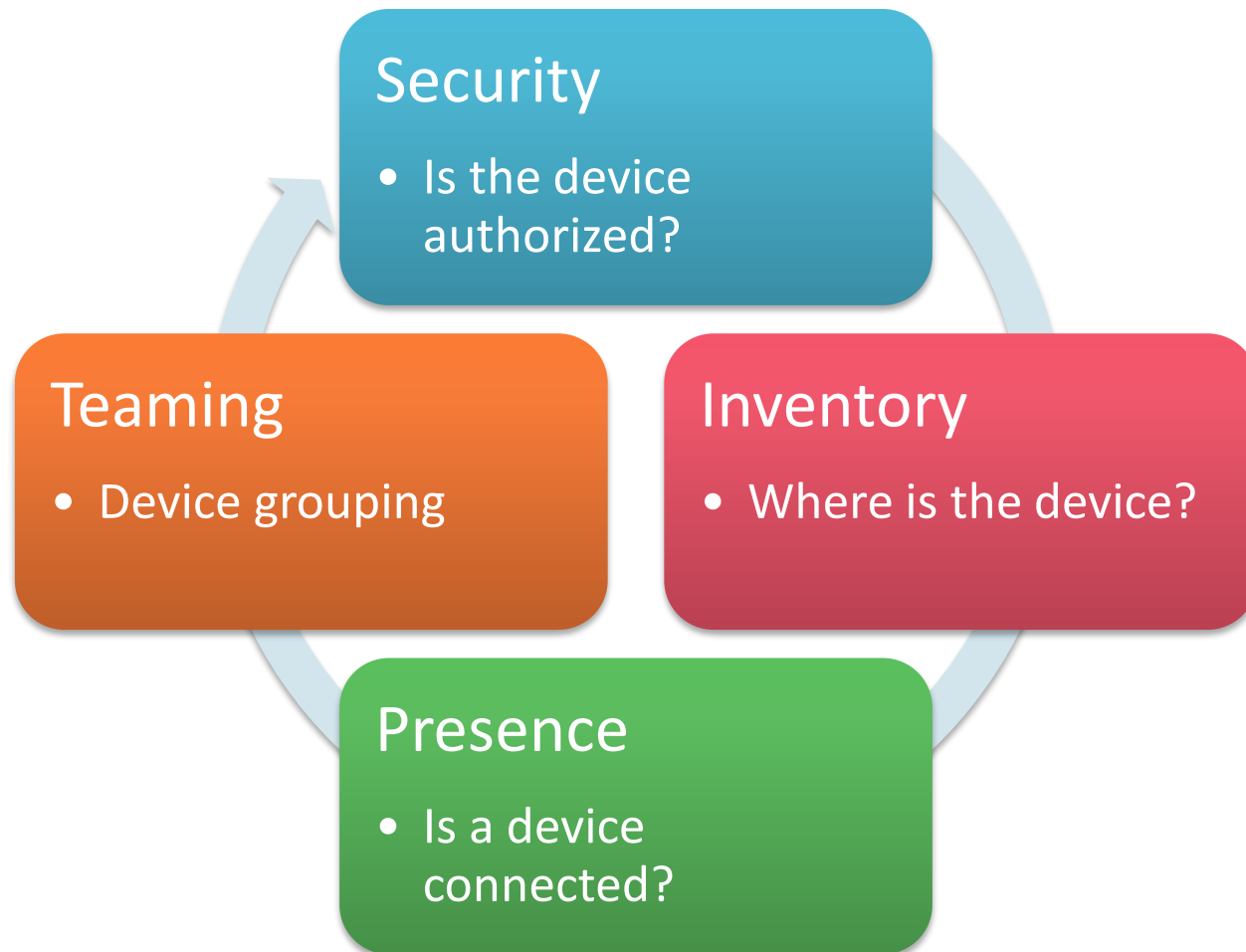
- Security, Numbers, Units of Measure, etc.

Semantic

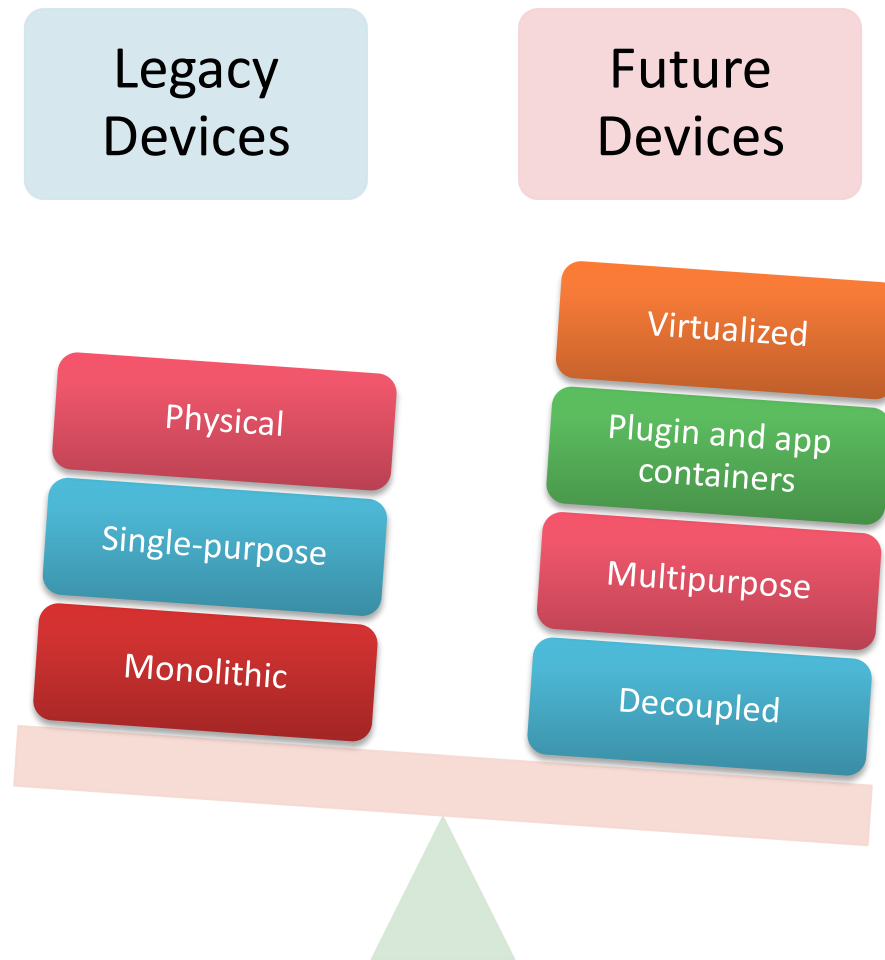
- Presence, Vitals, Glucose, Heartbeats, etc.



OSS applicability to manageability



OSS enables extensible devices



Appreciate tradeoffs

The more connection-friendly a device, the harder it is to validate it



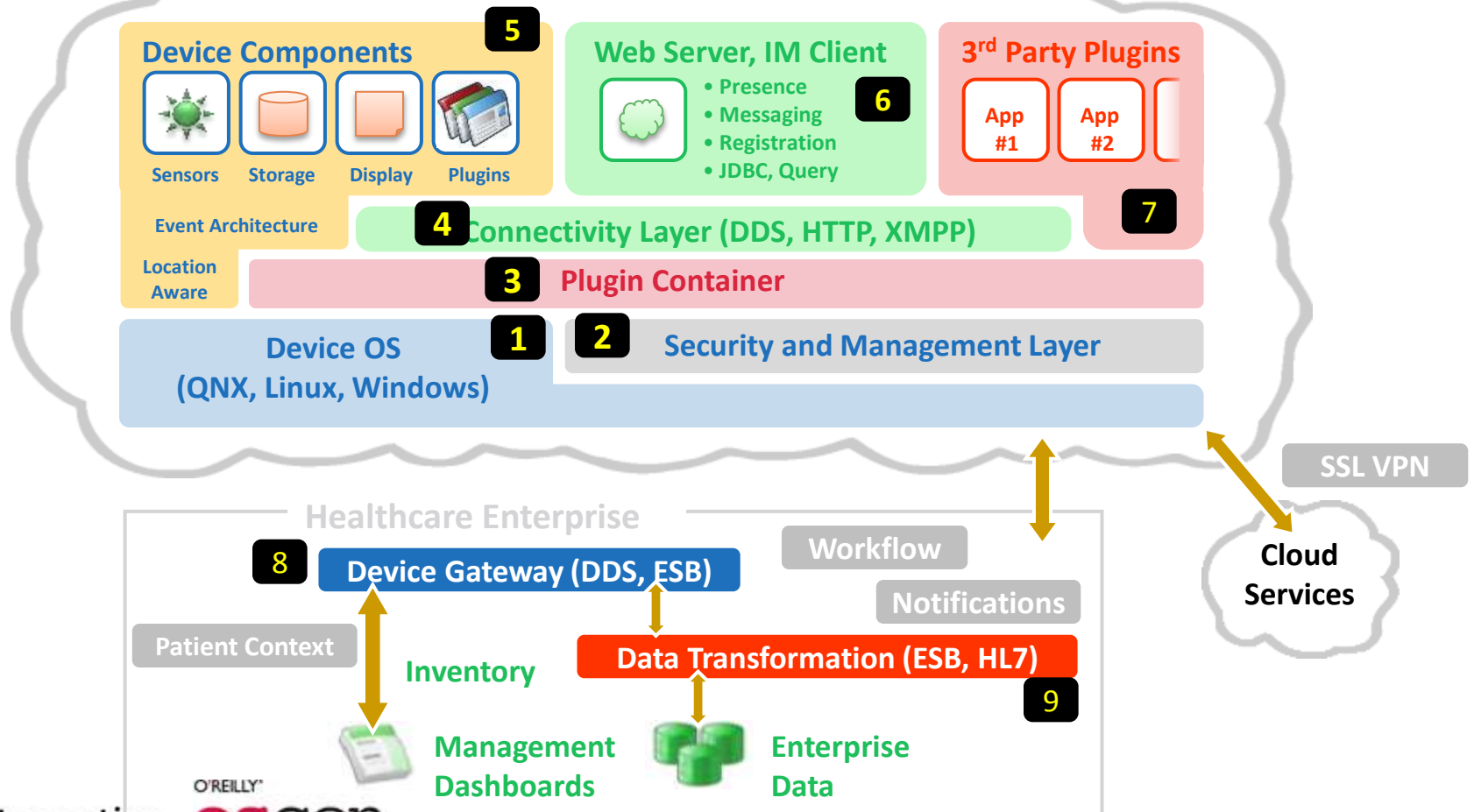
Integration-friendliness

Ease of validation

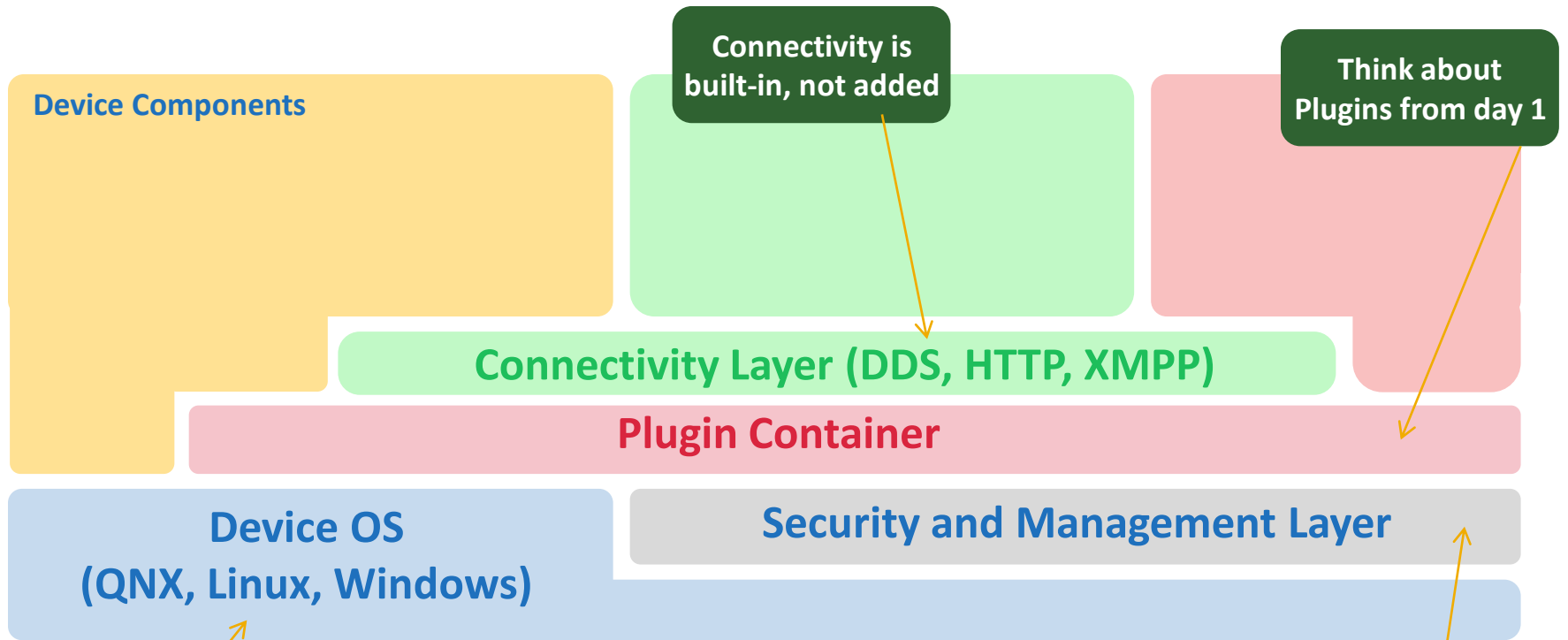
Lesson: Demand Testability



Shahid's "Ultimate Connectivity Architecture"



OSS in Ultimate Architecture Core



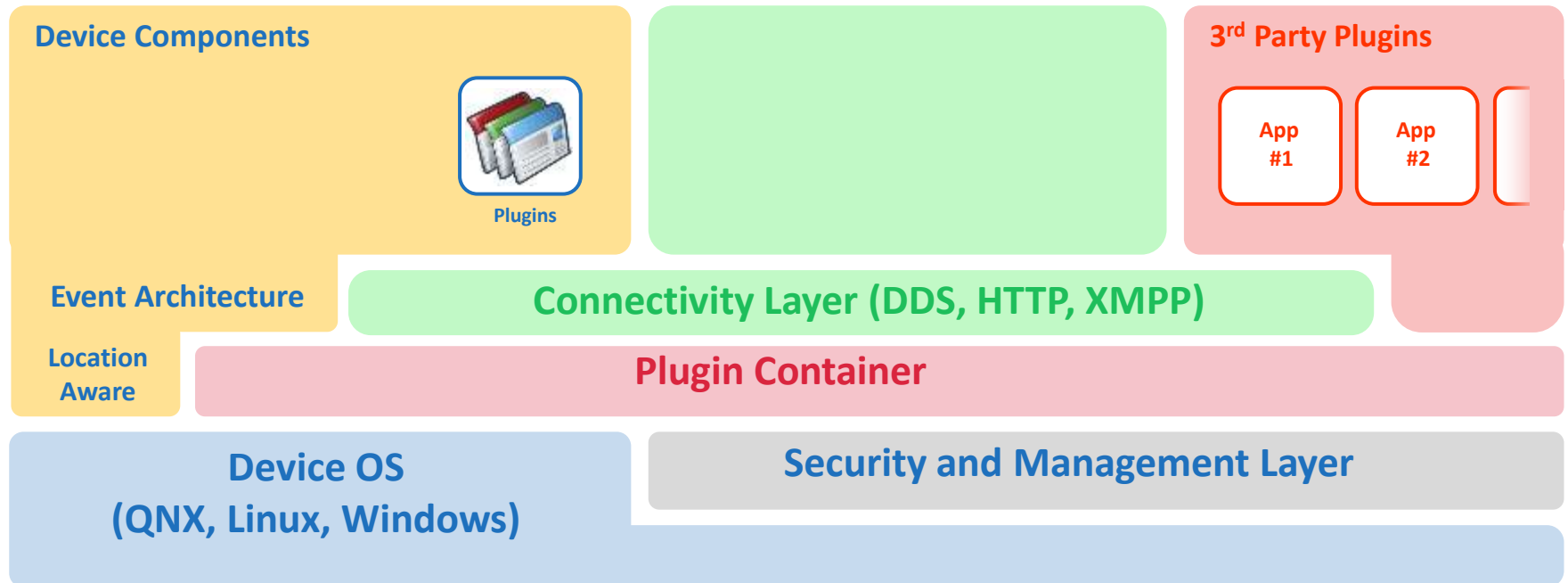
Don't create your own OS!

Build on Open Source

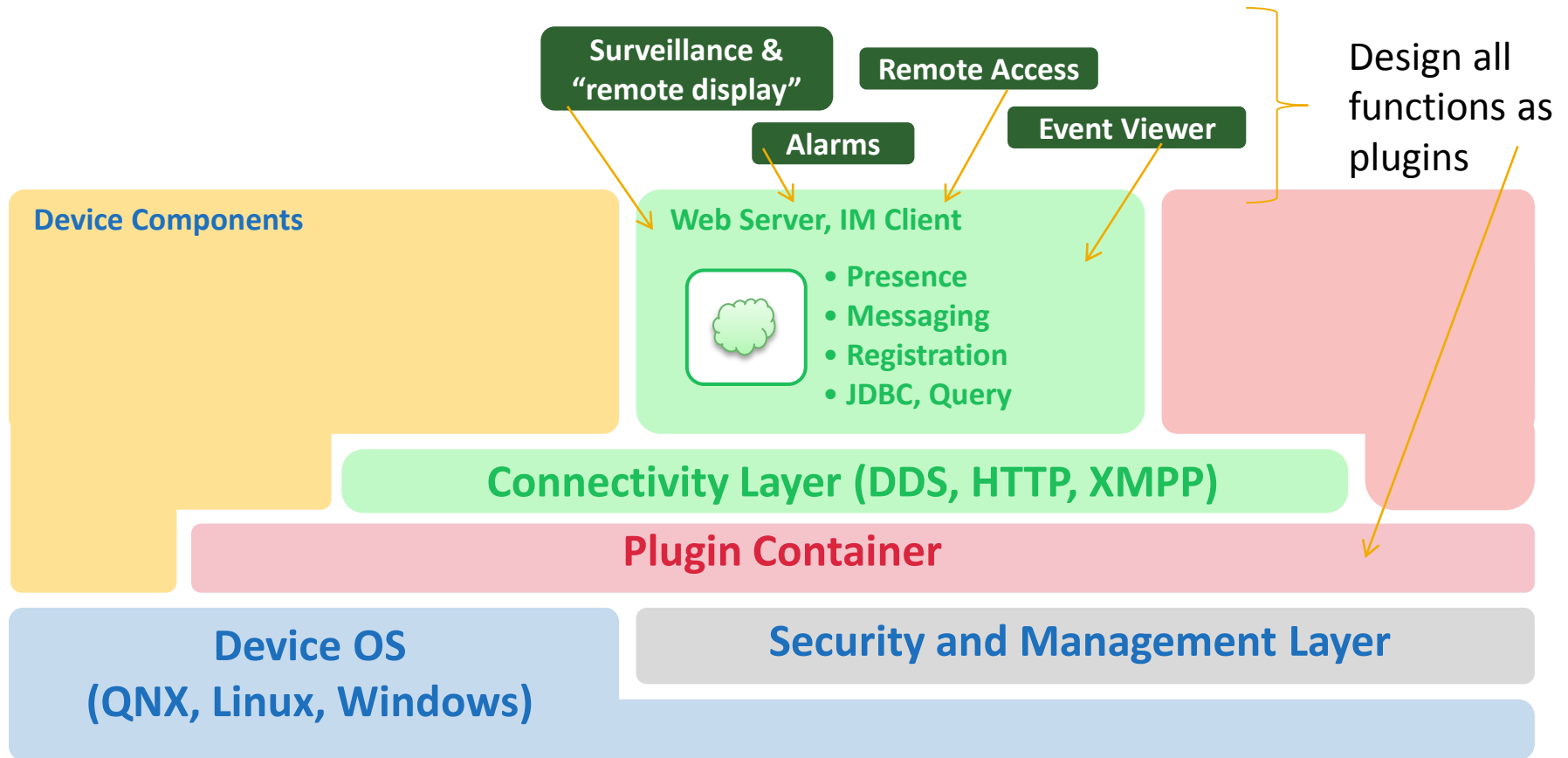
Create code as a last resort

Security isn't added later

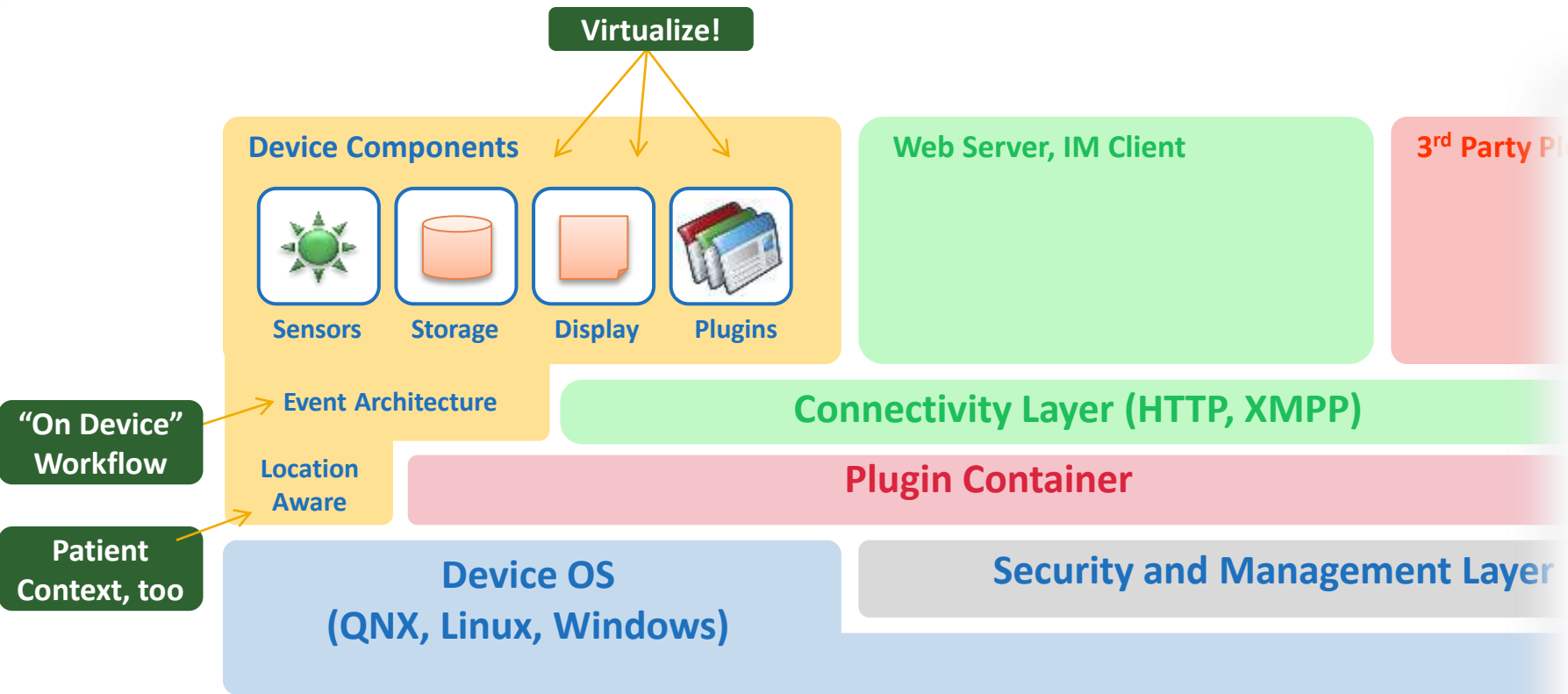
OSS enables plugin architecture



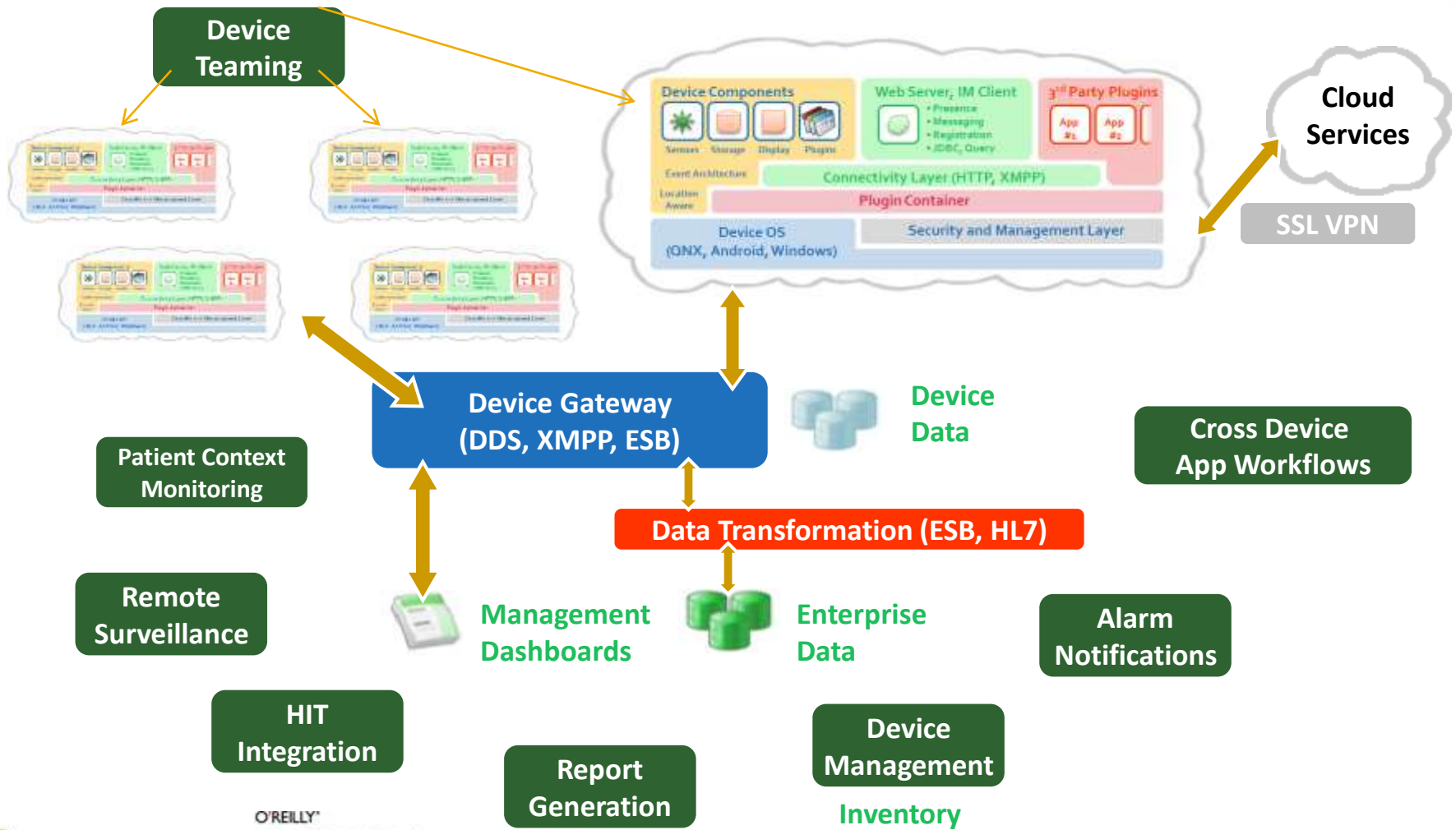
OSS in connectivity components



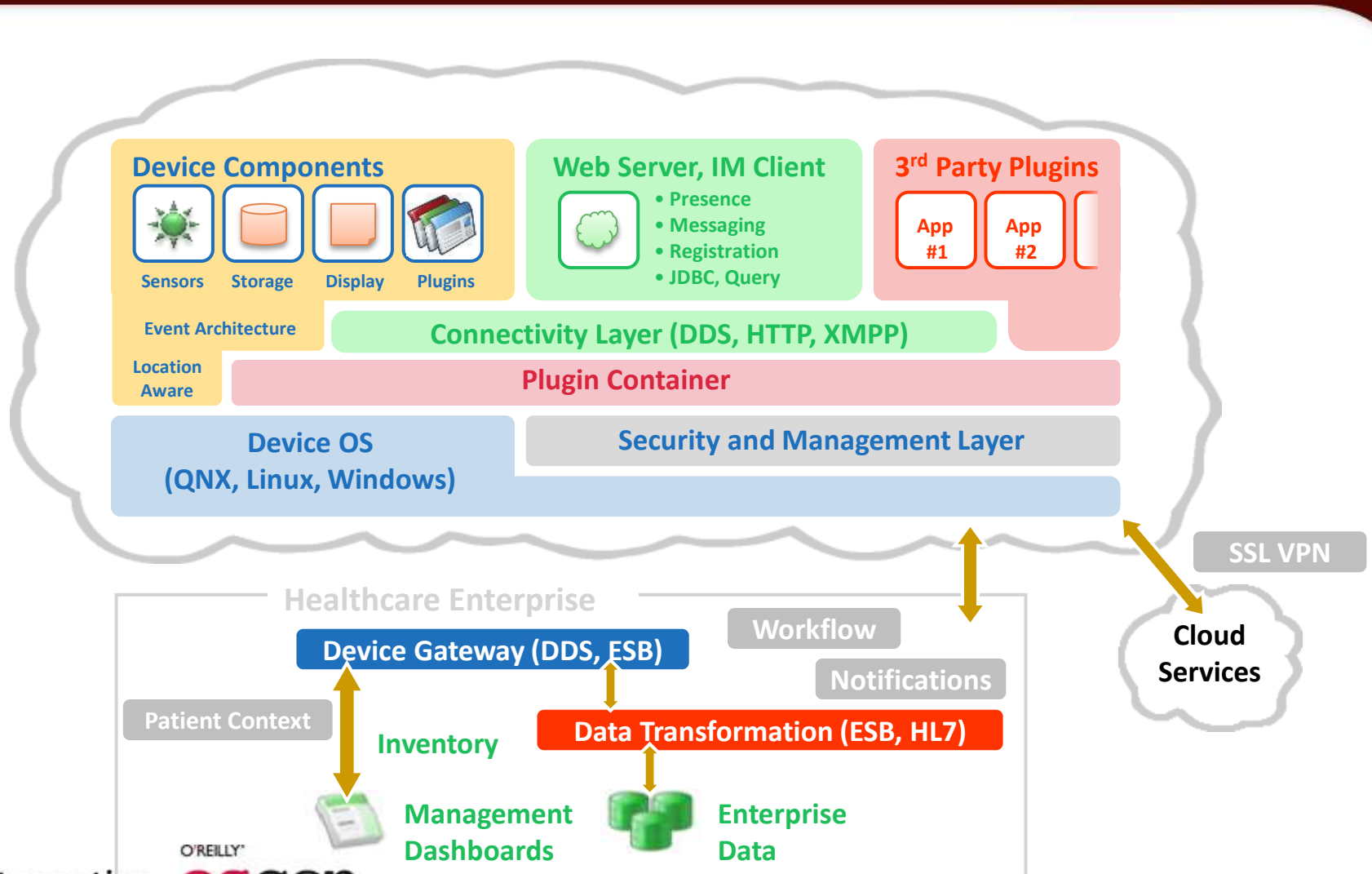
OSS in device components



OSS enables enterprise integration



Ultimate Connectivity Architecture



CONCLUSION AND QUESTIONS