Using Android in Safety-critical Medical Device Platforms

Inherent connectivity creates significant opportunities in medical science

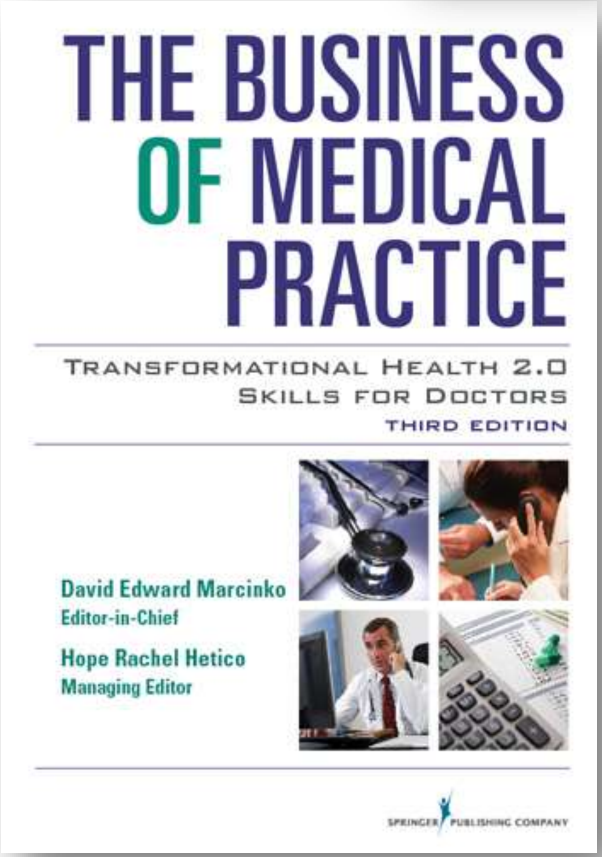
Shahid N. Shah, CEO

Netspective
EXTENDING THE ENTERPRISE

THE LINUX FOUNDATION ANDROID™ BUILDERS SUMMIT

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)



Author of Chapter 13, "You're the CIO of your Own Office"

What’s this talk about?

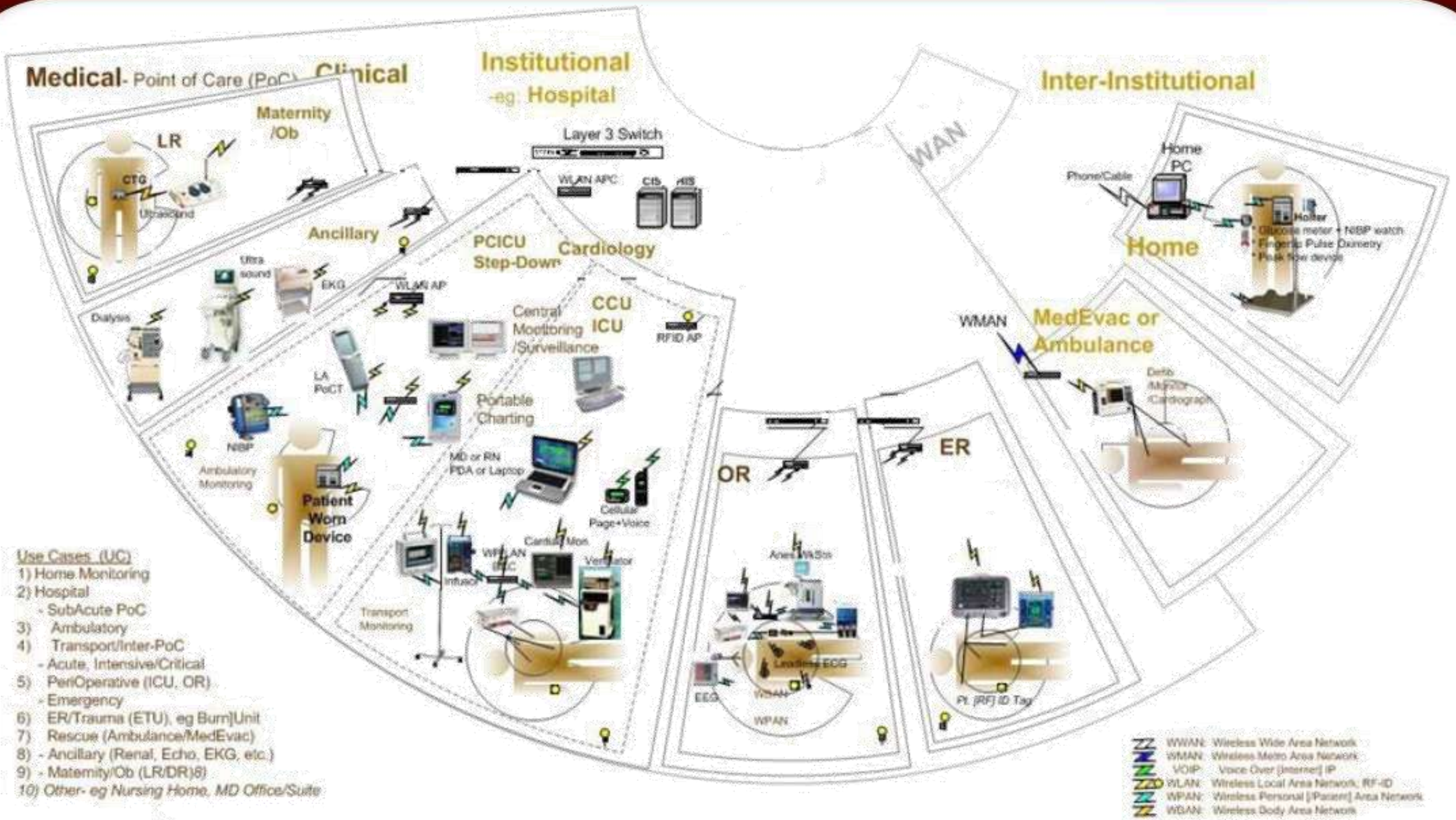
Health IT / MedTech Landscape

- Data has potential to solve some hard healthcare problems and change how medical science is done.
- The government is paying for the collection of clinical data (Meaningful Use or “MU”).
- All the existing MU incentives promote the wrong kinds of data collection: unreliable, slow, and error prone.

Key Takeaways

- Medical devices are the best sources of quantifiable, analyzable, and reportable clinical data.
- New devices must be designed and deployed to support inherent connectivity.
- EL, Android, and OSS are ideal for next generation and innovative medical devices but especially Android due to user experience.

What if we had access to all this data?



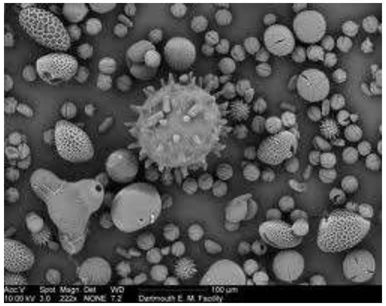
Source: Jan Whittenber, Philips Medical Systems

What problems can data help solve?

- Cost per patient per procedure / treatment going up but without ability to explain why
- Cost for same procedure / treatment plan highly variable across localities
- Unable to compare drug efficacy across patient populations
- Unable to compare health treatment effectiveness across patients
- Variability in fees and treatments promotes fraud
- Lack of visibility of entire patient record causes medical errors

Data changes the questions we ask

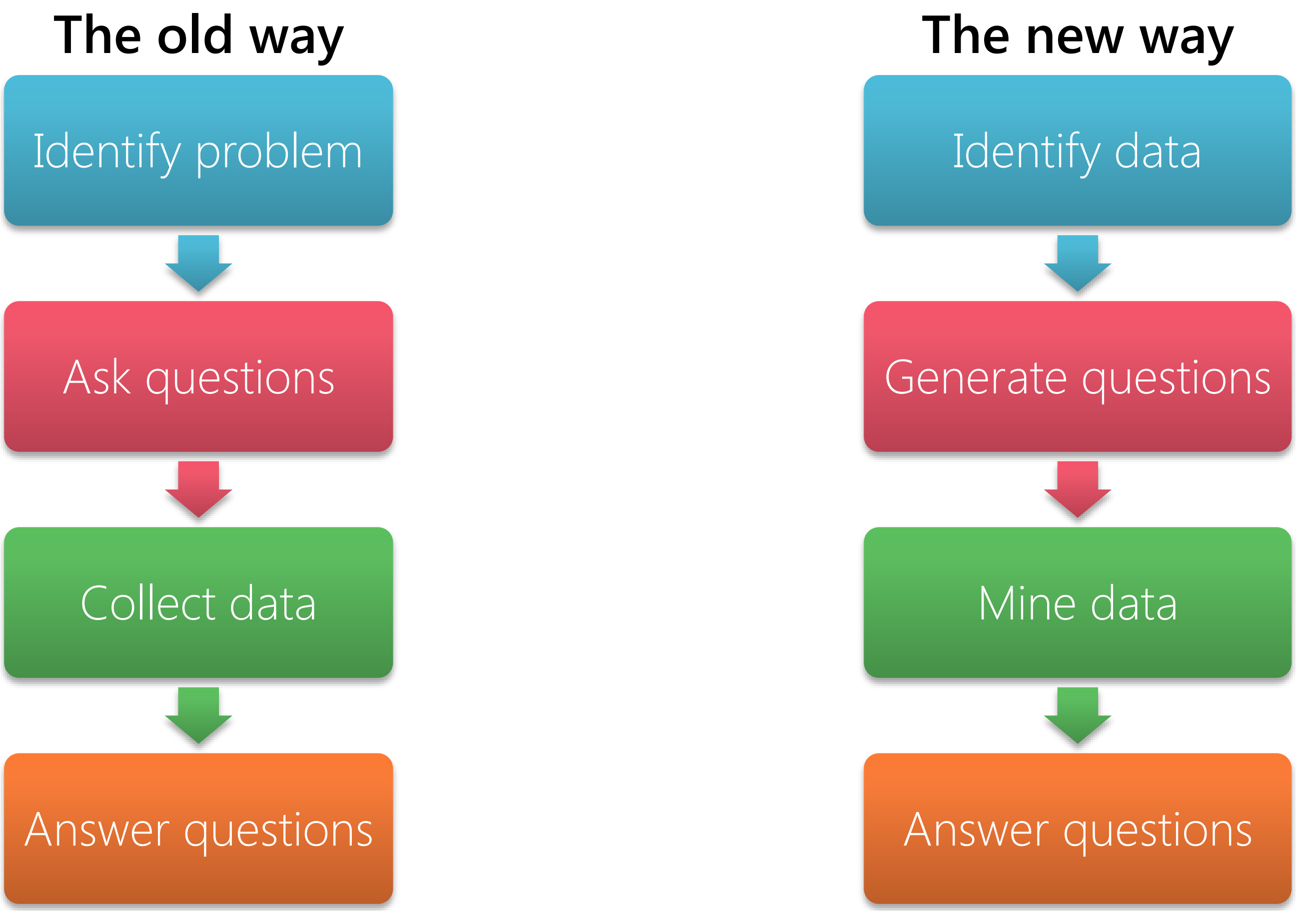
Simple visual facts



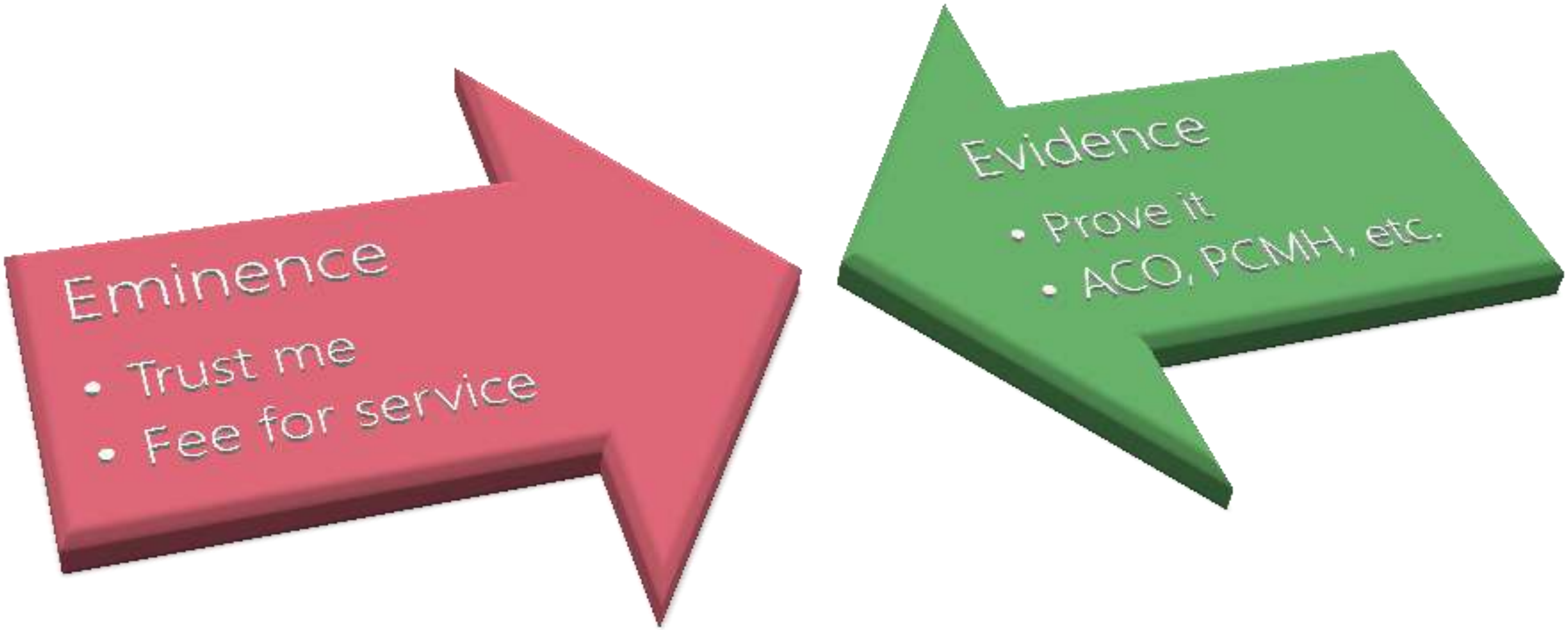
Complex visual facts

Complex computable facts

Data can change medical science



Evidence-based medicine is our goal



Unstructured patient data sources

	Patient	Health Professional	Labs & Diagnostics	Medical Devices	Biomarkers / Genetics
Source	Self reported by patient	Observations by HCP	Computed from specimens	Computed real-time from patient	Computed from specimens
Errors	High	Medium	Low		
Time	Slow	Slow	Medium		
Reliability	Low	Medium	High		
Data size	Megabytes	Megabytes	Megabytes		
Data type	PDFs, images	PDFs, images	PDFs, images		
Availability	Common	Common	Common	Uncommon	Uncommon

Structured patient data sources

	Patient	Health Professional	Labs & Diagnostics	Medical Devices	Biomarkers / Genetics
Source	Self reported by patient	Observations by HCP	Specimens	Real-time from patient	Specimens
Errors	High	Medium	Low	Low	Low
Time	Slow	Slow	Medium	Fast	Slow
Reliability	Low	Medium	High	High	High
Discrete size	Kilobytes	Kilobytes	Kilobytes	Megabytes	Gigabytes
Streaming size				Gigabytes	Gigabytes
Availability	Uncommon	Common	Somewhat Common	Uncommon	Uncommon

The need for connected devices

- 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

Android enables safe connectivity

Most obvious benefit

Data integration

Least attention

Manageability

Enhance functionality

Most promising capability

This talk focuses on connected devices

Key Android questions

Will the FDA accept Android in safety-critical systems?

Is Android safe enough for medical devices?

Simple answers

Will the FDA accept Android in safety-critical systems?

Yes

Is Android safe enough for medical devices?

Yes

but you must prove it

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

Android 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

Android and OSS FMEA risk analysis

- 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

OSS / open standards applicability

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

Android enables connectivity…

But only up to a point, there's more to connectivity than meets the eye

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.

Android improves manageability

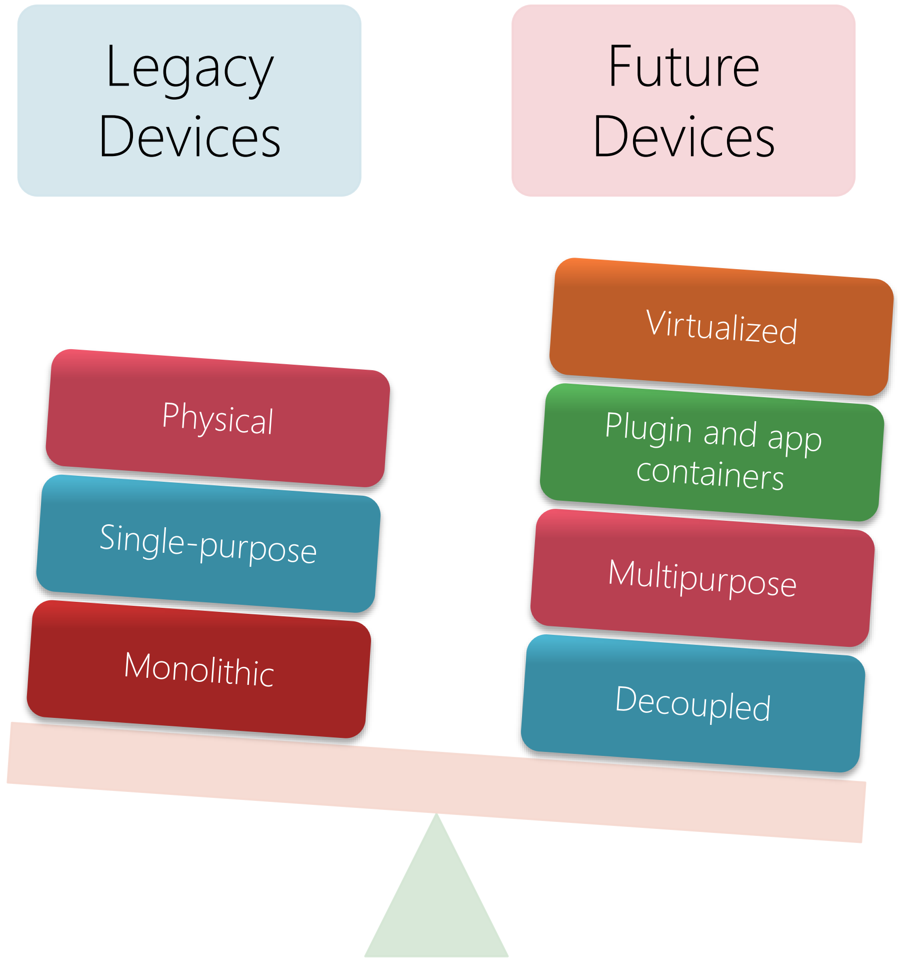
Security
- Is the device authorized?

Teaming
- Device grouping

Inventory
- Where is the device?

Presence
- Is a device connected?

Android enables extensible devices



Appreciate tradeoffs

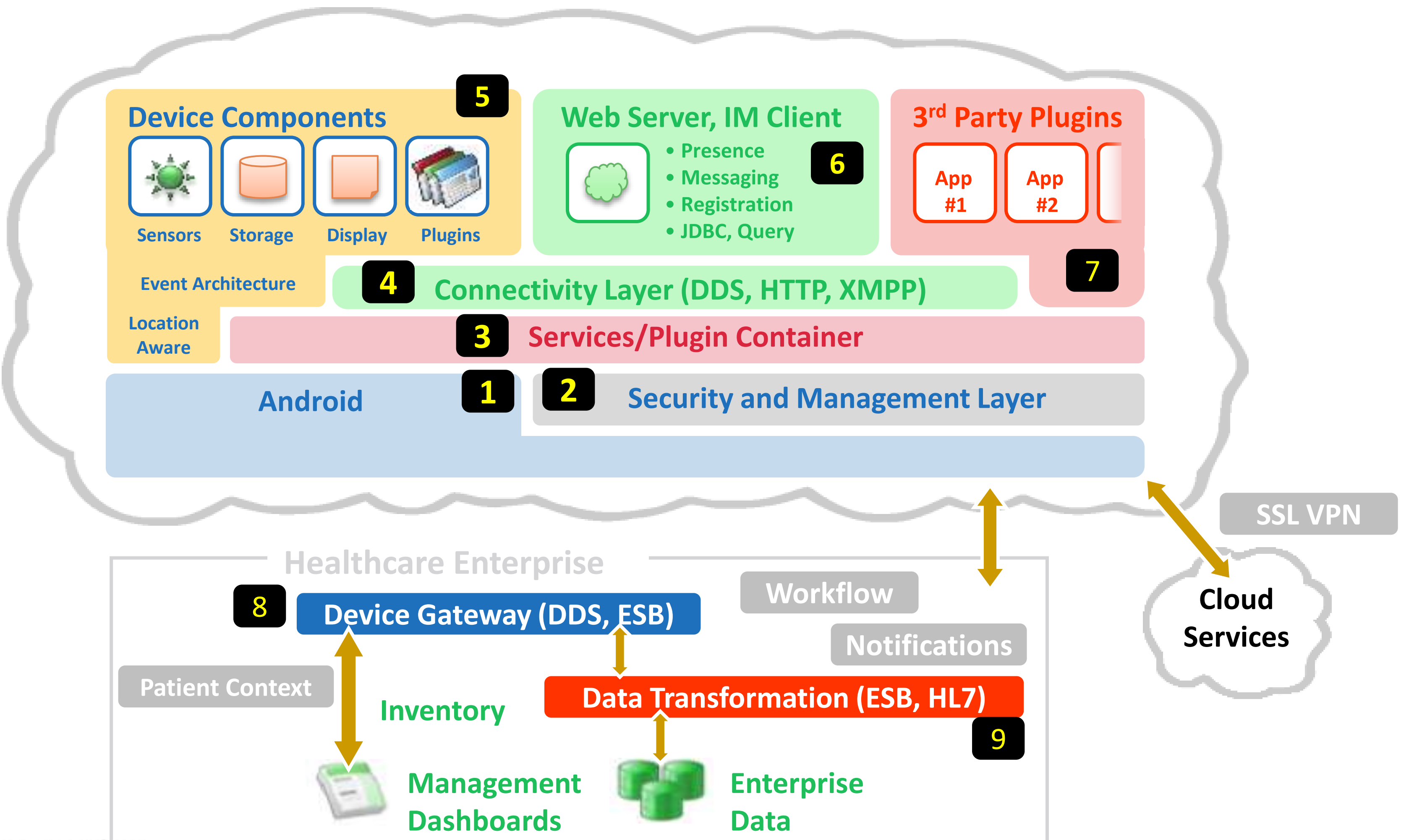
The more connection-friendly a device, the harder it is to validate it

Integration-friendliness

Ease of validation

Lesson: Demand Testability

Ultimate Device Architecture

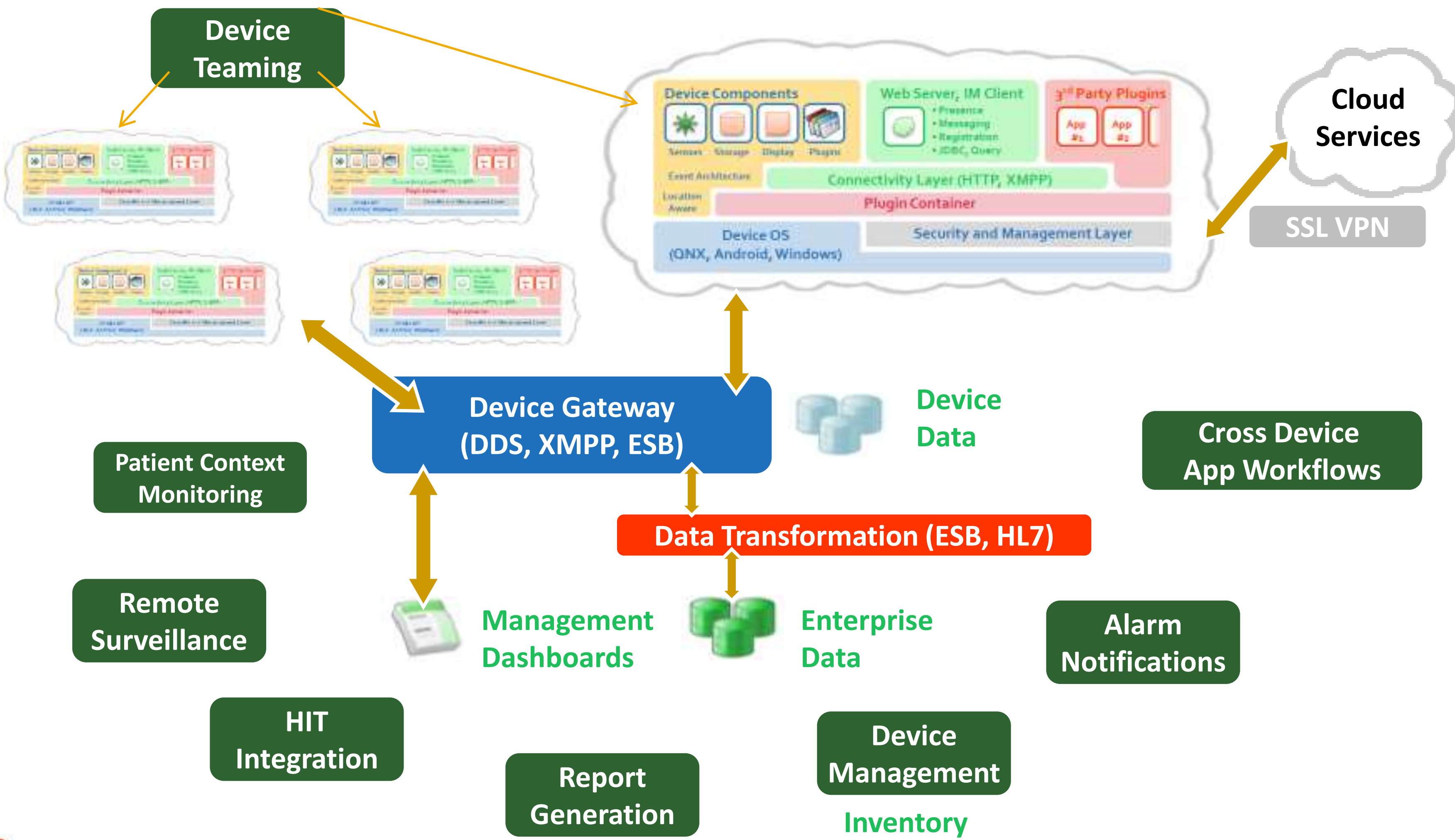


Do's and Dont's for Medical Devices

- Build on not just Android but other open source packages
- Create your own "safety-focused" Android distribution
- Consider device surveillance and remote display as basic requirements
- Virtualize sensors, storage
- Ensure regulated alarm management is built into the infrastructure

- Don't create your own OS
- Don't leave patient association/context to apps
- Don't just assume apps or services will enforce private and security, build-it into the infrastructure
- Don't assume existing M2M tools will be enough for remote servicing and updates in MedTech
- Don't leave cross-device workflows off the table

Promote enterprise integration



shahid.shah@netspective.com
www.HealthcareGuy.com

Thank you for your time

Conclusion and Questions