Remote Service

SASG - Big Data

From machine design to IT management & Remote Service

Marcel Boosten
Philips Healthcare

October 7, 2014

innovation you

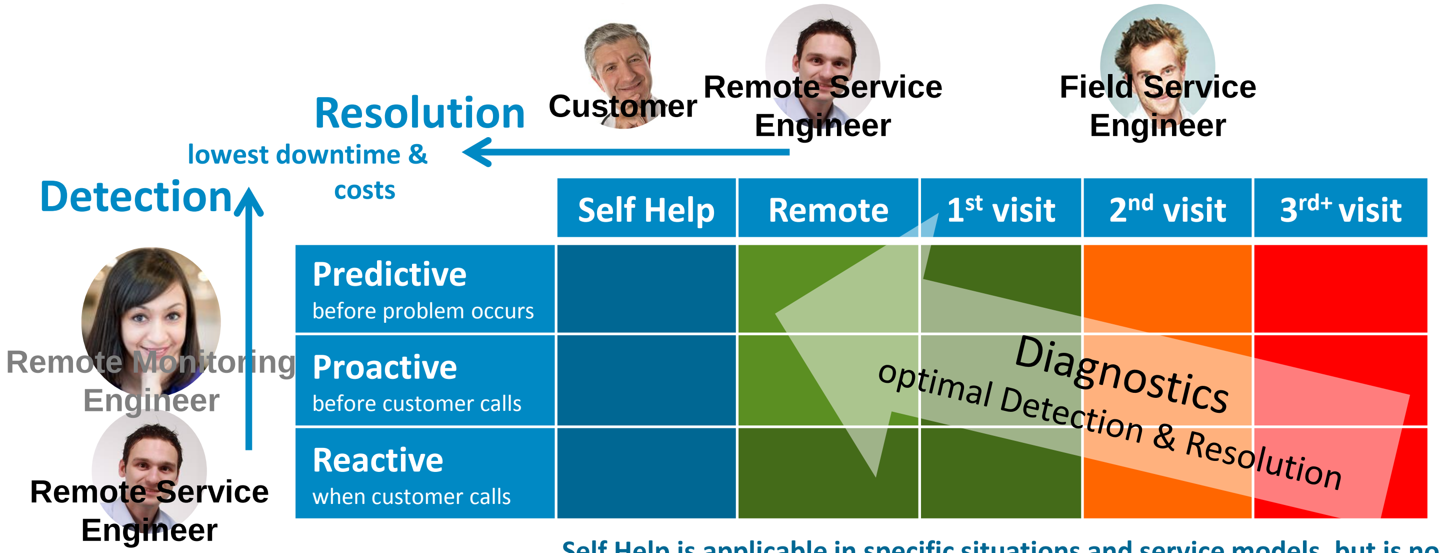
PHILIPS

Marcel Boosten

- Philips Lead – Design for Serviceability
- Solution Architect Diagnostics
- Healthcare – Imaging Systems – Customer Services
- History
  - 1994: ir @ TU/e
  - 1996: mtd @ SAI
  - 1999: phd @ CERN
  - 2000: SW Architect @ CT
  - 2002: System Architect, Innovation Manager @ iXR – 3D
  - 2007: Innovation Architect @ iXR
  - 2010: into Service

Diagnostics

Objective: maximize uptime (=value), minimize internal costs (visits, parts, hours)



Customer

Remote Service Engineer

Field Service Engineer

Resolution

lowest downtime & costs

Detection

Remote Monitoring Engineer

Remote Service Engineer

	Self Help	Remote	1st visit	2nd visit	3rd+ visit
Predictive before problem occurs					
Proactive before customer calls					
Reactive when customer calls					

Diagnostics

optimal Detection & Resolution

Self Help is applicable in specific situations and service models, but is not a general strategy.

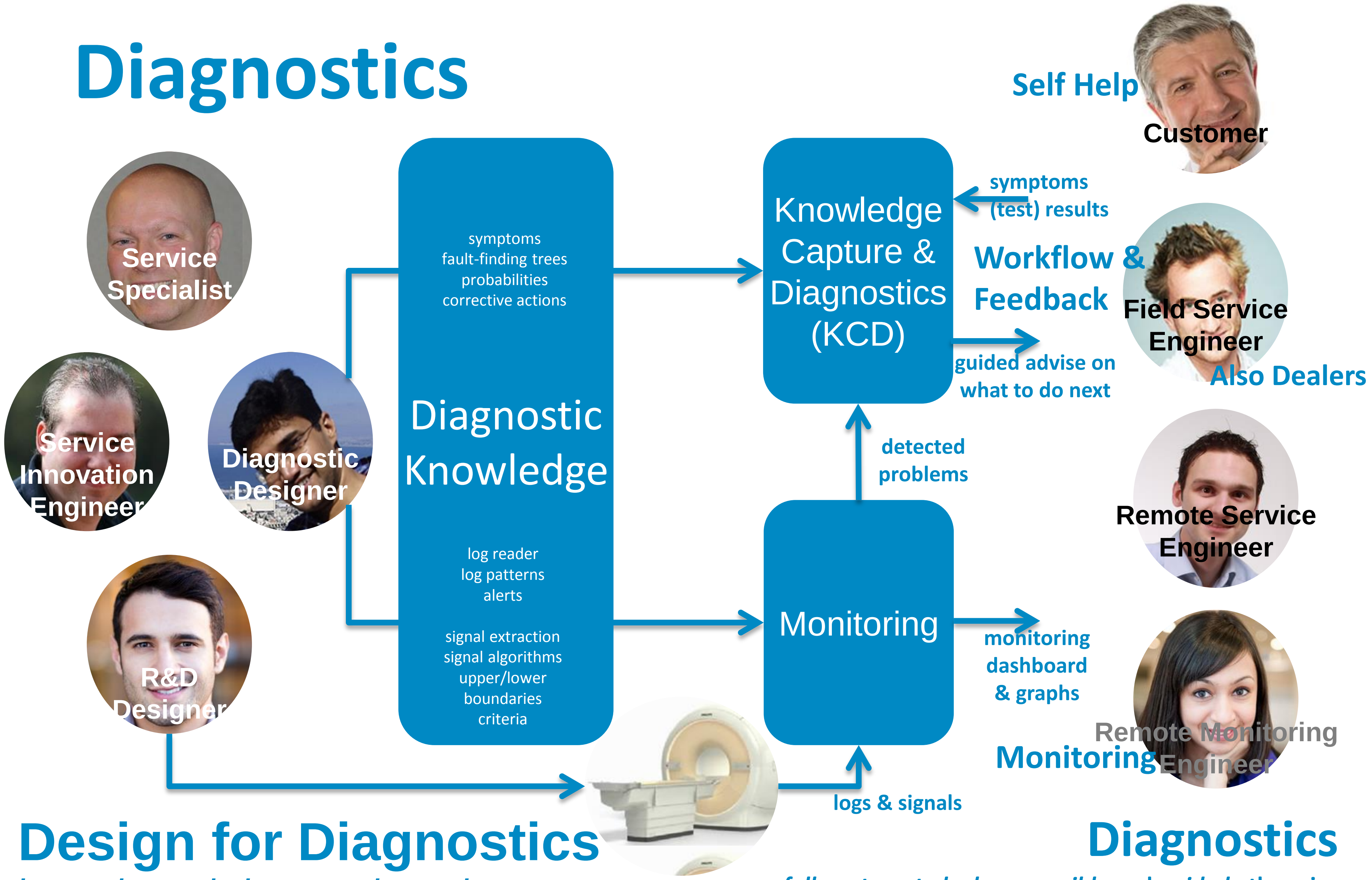
Design state-of-the-art Diagnostics

[1] ***on-to*** devices - no device changes required, only external

[2] ***in-to*** devices - the device changes

while ***continuously improving*** it based on service statistics & feedback.

Diagnostics



Self Help

Customer

symptoms
(test) results

Workflow & Feedback

Field Service Engineer

Also Dealers

guided advise on what to do next

Service Specialist

Service Innovation Engineer

Diagnostic Designer

R&D Designer

Diagnostic Knowledge

symptoms
fault-finding trees
probabilities
corrective actions

log reader
log patterns
alerts

signal extraction
signal algorithms
upper/lower boundaries
criteria

Knowledge Capture & Diagnostics (KCD)

detected problems

Monitoring

monitoring dashboard & graphs

Remote Service Engineer

Remote Monitoring Engineer

Monitoring

logs & signals

Design for Diagnostics

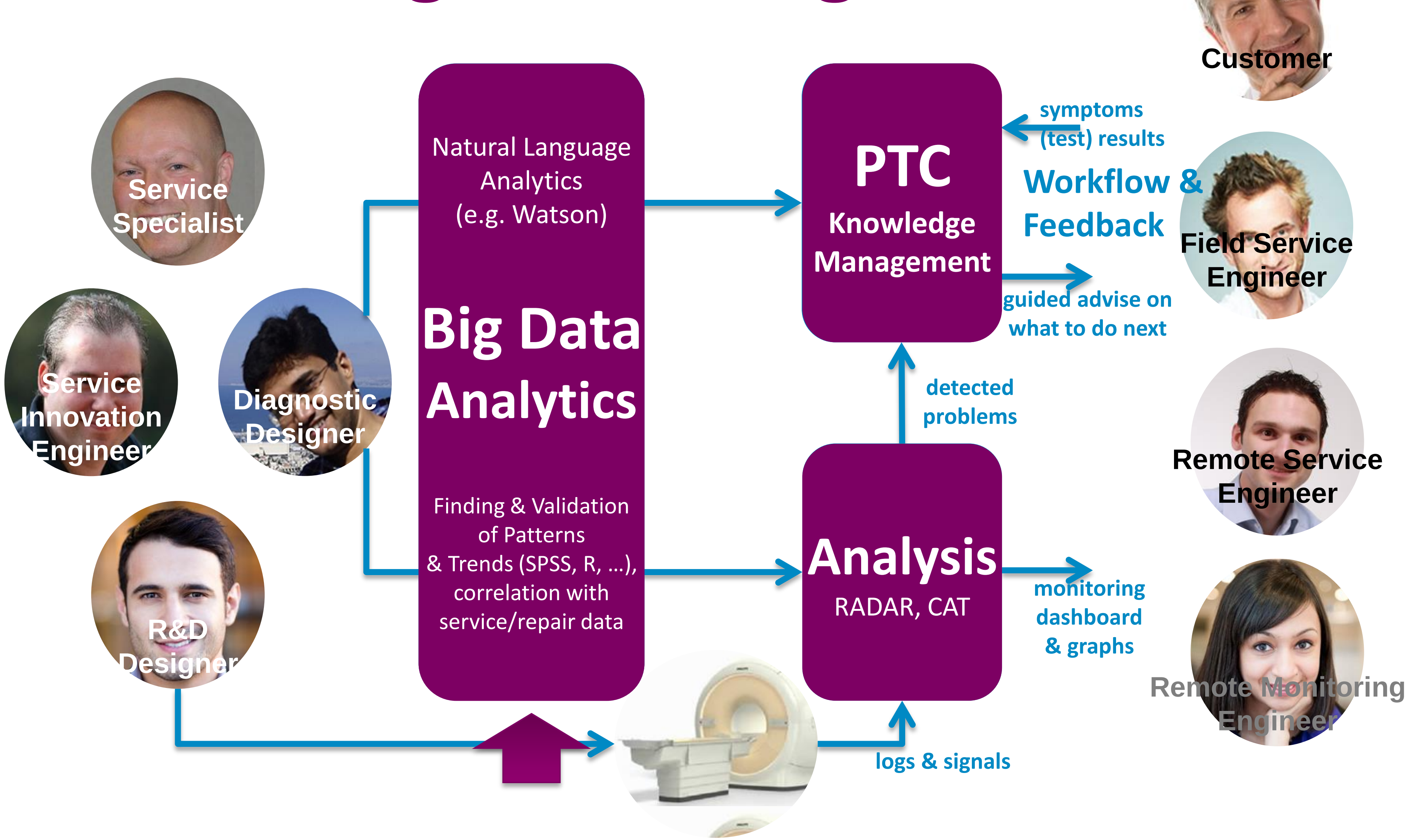
in-to and on-to devices + continuous improvement

Diagnostics

fully automated where possible **and** ***guided*** **otherwise**

Monitoring based on logs & signals to detect/predict problems.
KCD supports human workflow (symptoms, tests, actions) and closes the feedback loop.
Diagnostic Knowledge is created by combining *Expert Knowledge* and *Data Analytics* .

Technologies for Diagnostics



Customer
Service Specialist
Service Innovation Engineer
Diagnostic Designer
R&D Designer
Natural Language Analytics (e.g. Watson)
Big Data Analytics
Finding & Validation of Patterns & Trends (SPSS, R, …), correlation with service/repair data
PTC Knowledge Management
symptoms (test) results
Workflow & Feedback
guided advise on what to do next
Field Service Engineer
detected problems
Analysis
RADAR, CAT
monitoring dashboard & graphs
Remote Service Engineer
Remote Monitoring Engineer
logs & signals

Big Data *Analytics* can be used to *support people* to *efficiently* create & optimize Diagnostic Contents.

Remarks

on Big Data Analytics to create Diagnostic Contents

Big Data Analytics allows *non*-subject-matter-experts to ‘discover’ diagnostic patterns, provided:

1. Data is available, and
2. Data is reliable, and
   E.g. known corrective action & moment in time that actually did solve the problem.
3. Enough data is available to perform statistics for such specific case, and
4. The data contains the useful observations

Expert knowledge allows compensating for 1..4

Look at code & design

Human interpretation

Expert info & judgement

Build new tests

Design for Diagnostics on-to Devices
The combination of Expert knowledge with Data Analytics capabilities
is needed to address Diagnostic Content creation effectively.

Design for Diagnostics in-to Devices
ensures the right monitors & tests & service actions are designed into the system.
This is essential to get good, accurate, and precise remote diagnostics.

Design for Diagnostics

ETTR & Costs grows

Test Method	Remote	Proactive Predictive Intermittent	No clinical downtime	Lead time to Detection
Monitoring	✓	✓	✓	Instant
Self Test	✓	✗	✓	@system available
Remote Test	✓	✗	✗	@scheduled time slot
User Input (a simple question asked via phone)	✓	✗	✓	@tech available
Local Test	✗	✗	✗	@FSE in hospital

Design for Diagnostics

Design a one-to-one relation between observation and the service action required for each failure mode.

Completeness

Component	Failure Mode	Service Action	Monitoring
Harddisk	FM a	Replace	CRCErr
	FM b	Replace	CRCErr
	FM c	Replace	FileReadErr
FileSystem	FM x	Fix	FileReadErr
BIOS	FM y	Config	BlueScreen
OS	FM z	Install	BlueScreen

Diagnosable with lowest costs

The Design Challenge: Fitting it all together

Component	Failure Mode	Service Action	Monitoring	Self Test	Remote Test	User Input	Local Test
Harddisk	FM a	Replace	CRCErr				
	FM b	Replace	CRCErr				
	FM c	Replace	FileReadErr	Post5=1			
FileSystem	FM x	Fix	FileReadErr	Post5=0			
BIOS	FM y	Config	BlueScreen		BiosChk=1		
OS	FM z	Install	BlueScreen		BiosChk=0		

(1) Identify all **Failure Modes**

(2) High-Level Design of **Service Actions**: How-to & Service Agent (€)

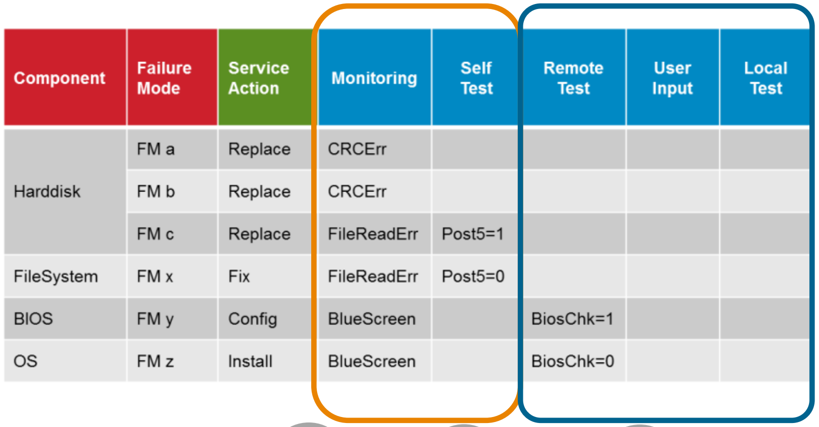
(3) Diagnostic Design

- High Level Design of **Observations**: How-to & Service Agent (€)
- Design the one-to-one relation: **Observations** > **Service Actions**

(4) Detailed Design of **Service Actions**

(5) Detailed Design of **Observations**

Logging

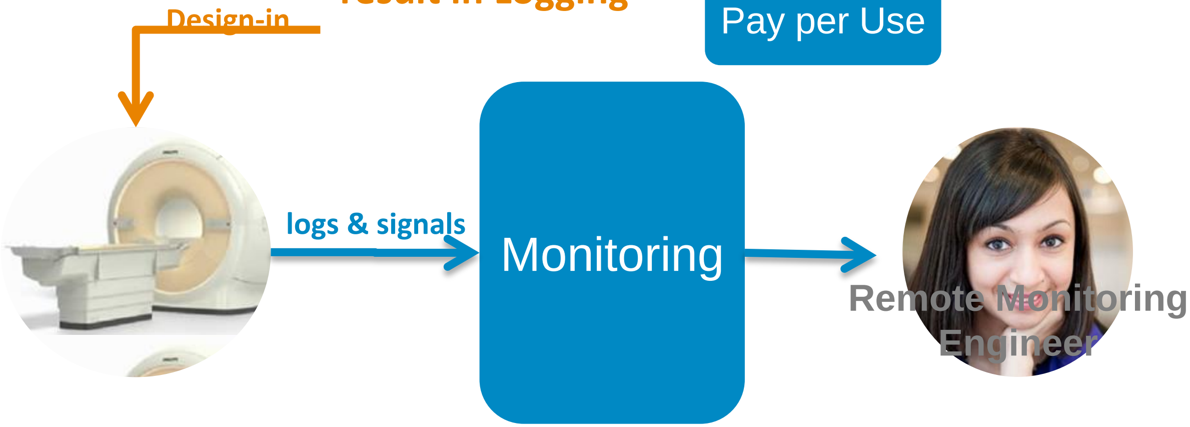


Component	Failure Mode	Service Action	Monitoring	Self Test	Remote Test	User Input	Local Test
Harddisk	FM a	Replace	CRCErr				
	FM b	Replace	CRCErr				
	FM c	Replace	FileReadErr	Post5=1			
FileSystem	FM x	Fix	FileReadErr	Post5=0			
BIOS	FM y	Config	BlueScreen		BiosChk=1		
OS	FM z	Install	BlueScreen		BiosChk=0		

+ and – and V
result in Logging

Utilization

Pay per Use



Each Log is an External Interface of the Device on which Services depend.

(so, sometimes more than 50% of your business)

Logging should be a Managed Interface.

(often, it is not managed in practice!)

PHILIPS

Logging as Managed Interface

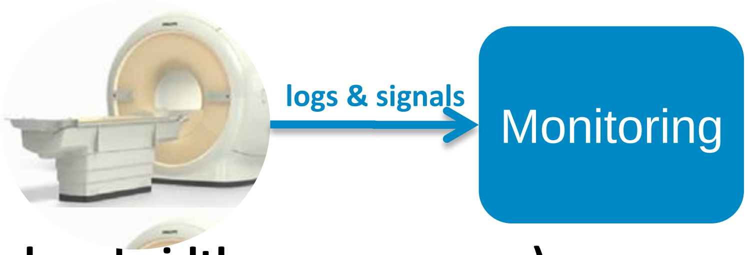
Log => Log Messages

- **Consistency across releases**
  Services, such as Diagnostics, are performed on the installed base – i.e., *across* different SW & HW *versions* of the system.
  The Logging Interface should be managed & kept consistent for a Product Family, consistently across releases.

- **Stable semantics of each Log**
  Each Log has a unique (type) id, and fixed *syntax* and *semantics*.
  - New semantics => NewID
  - Avoid changes

Logging as Managed Interface: it is obviously needed, it is not difficult, but…
hardly ever done properly in practice…

Diagnostics, Logging & Alerting



Challenges

- **Log Size 'Too Big' (for bandwidth, costs of remote solution)**

  usually a lot of garbage is generated. Solution: **garbage filter**, pareto-based reduction
  So not: reducing useful info, or designing for minimal bandwidth. Just avoid excessive bandwidth.

- **Alerting - Immediate attention**

  Solutions: [1] constant upload, or [2] immediately if …, or [3] on-device rules

- **On-Device Diagnostics**
  - Updateable 'with the enterprise' – without FCO (Field Change Order) of Medical Device
  - Running in background – without impacting Medical operations
  - Question to this team: Interested in solutions

- **Big Data Fear, Privacy, Security & Intellectual Property**
  - Service Logging & Utilization Logging 'overlaps'. Privacy laws. 'Site' vs 'Cloud' solutions.
  - Logs are owned by hospital. Hospital can give it to 3rd parties for 3rd party service.
  - Various levels of 'knowledge', Intellectual Property:
    - Log102123
    - Log102123 X-Ray Problem
    - Log102123 X-Ray Generate peak detected
    - Log102123 X-Ray Generate peak detected – replace Maximus board

Summary of Main Messages

- **Design for Diagnostics [1]** ***on-to*** devices, and **[2]** ***in-to*** devices while ***continuously improving*** it based on service statistics & feedback.

- **On-to:** Big Data ***Analytics*** can be used to ***support people*** to ***efficiently*** create & optimize Diagnostic Contents.
  Challenges: creation/optimization efficiency, dealing with uncertainties, access to expert knowledge.

- **In-to**: ***Design*** a ***one-to-one relation*** between ***observation*** and the ***service action*** required for each ***failure mode***.
  Observations via logs resulting from monitors & tests.

- **Each Log** should be a ***Managed*** **External Interface**.

- **Related Challenges**:
  – Independently updateable on-device diagnostics / data preprocessing
  – Big Data Fear, Privacy – ‘Site’ & ‘Cloud’ solutions

PHILIPS