Combining Product Line Engineering and Service Oriented Architecture in Health Care Infrastructure Systems: Experience Report

PLE   SOA

Jörg Bartholdt, Bernd Franke, Christa Schwanninger Michael Stal
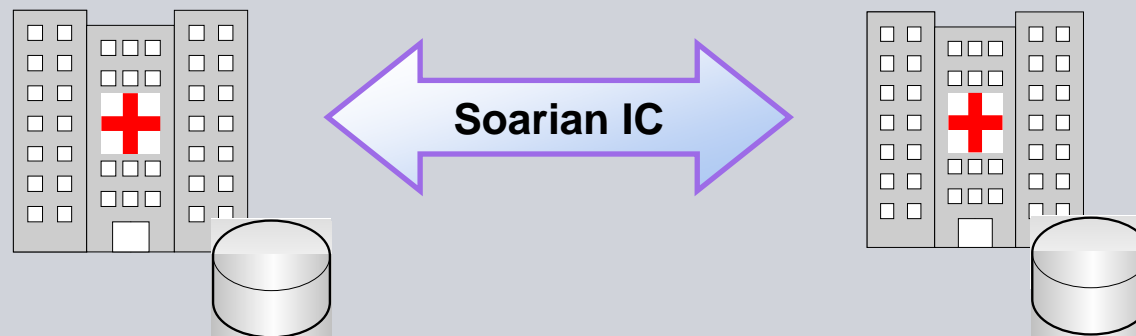Corporate Technology & Health Care
Siemens AG

Business Case

Hospitals have a HIS (Hospital Information System).
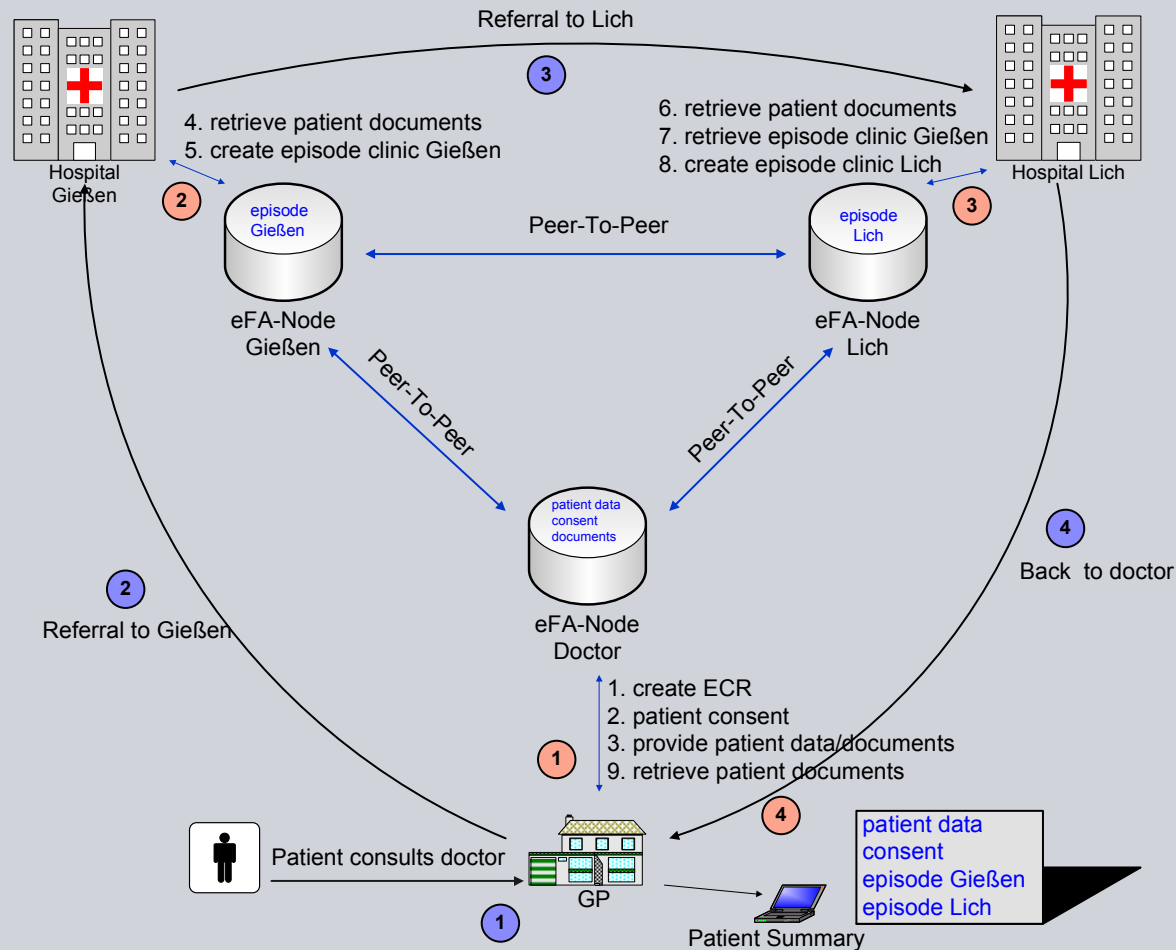Data is shared between departments (intra-hospital)

But what if it comes to transferring a patient to another hospital?
  You carry your X-ray images with you

Soarian IC targets
§   inter-hospital communication
§   Special scenarios of external data integration

Soarian IC

J. Bartholdt, B. Franke, C. Schwanninger M. Stal   © Siemens AG, Corporate Technology

In future: target residential doctors, too

Referral to Lich

3

4. retrieve patient documents
5. create episode clinic Gießen

6. retrieve patient documents
7. retrieve episode clinic Gießen
8. create episode clinic Lich

Hospital Gießen

2

episode Gießen

Peer-To-Peer

episode Lich

3

Hospital Lich

eFA-Node Gießen

eFA-Node Lich

Peer-To-Peer

Peer-To-Peer

patient data consent documents

4

Back to doctor

eFA-Node Doctor

2

Referral to Gießen

1. create ECR
2. patient consent
3. provide patient data/documents
9. retrieve patient documents

1

4

Patient consults doctor

GP

1

Patient Summary

patient data
consent
episode Gießen
episode Lich

J. Bartholdt, B. Franke, C. Schwanninger M. Stal

History

Product development was serialized
Previous version forms the bases for the next version (architecture erosion)

Customer A   Customer B

Results in monolithic application, interwoven dependencies

Assumptions:
Increased customer base (no serialization possible anymore)
Focus on main selling assets
Make system ready for integration

Goal:
Introduce SOA-approach: import/export via interfaces, composition of features via service chaining
Introduce PLE: focus on core assets, allow for customer specific variations, introduce new features in core if proven at one customer
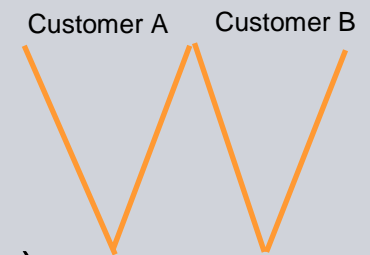
Challenges

1. *Increasing variability*
2. *Configurability/Subset-ability*
3. *Extensibility*
4. *Increased testability*
5. *Outsourcing*
6. *Risk effect mitigation*
7. *Exploitation of COTS (Common-Off-The-Shelf) products*
8. *Prioritization of features to be integrated in the platform*
9. *Positioning in the market (guide the customer)*
10. *Acceleration of tender preparation*
11. *Clinical workflows*
12. *Traceability*

J. Bartholdt, B. Franke, C. Schwanninger M. Stal   © Siemens AG, Corporate Technology

SIEMENS

Approach

1. Scoping (2,8,9,10):
   § Increasing customer base requires focus on most profitable features
   § Starting point: Group current requirements to features
   § Use feature model for reasoning with product mgmt, sales, development, etc („common language")
2. Variability Management (1,3,4,12):
   § Reduce variability points (expensive!)   pre-configurations
3. Building re-use culture (1,2,4,10):
   § Keep clear product portfolio strategy
   § Focus to market commonalities
   § Quick hacks forbidden in the core assets

J. Bartholdt, B. Franke, C. Schwanninger M. Stal   © Siemens AG, Corporate Technology

Approach

SIEMENS

4. Self-containment (2,3,4,5,6,12):
- § Fosters decoupling of components
- § Allows for exchange to third-party components
- § Allows to be used as a system, not only by humans via Web-Interface
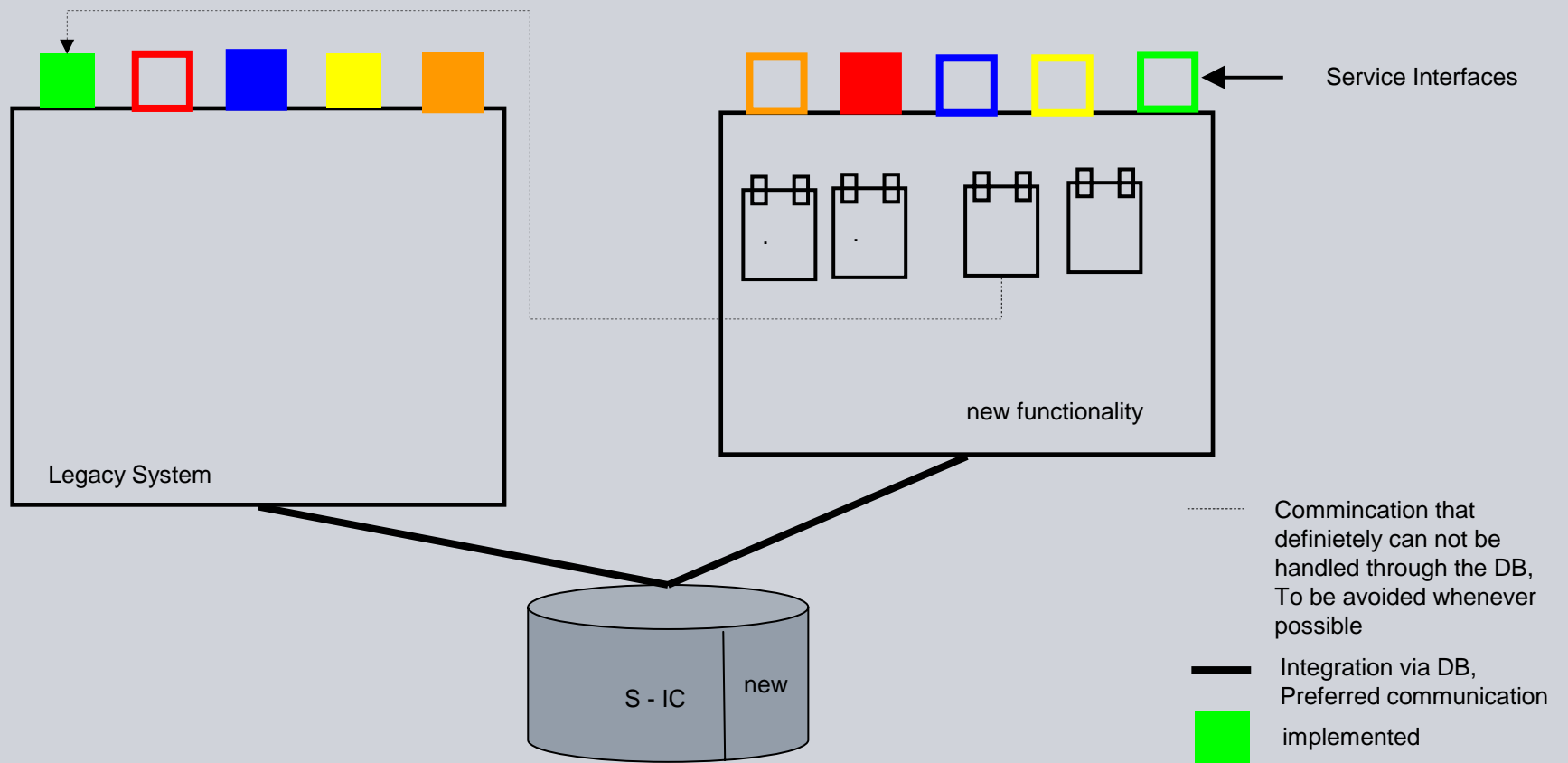- § Improves testability

5. Integration (2,7):
- § More freedom to tailor to customer needs
- § Face the fact that Siemens is not the only supplier

6. Flexibility (5,11):
- § Adding workflow or rule engines
- § support specifics of each customer (ideally by the customer)
- § Late (dynamic) binding

J. Bartholdt, B. Franke, C. Schwanninger M. Stal

Approach

Other projects showed the likelihood of failure in a big-bang approach
We favor a migration strategy

Service Interfaces

Legacy System

new functionality

Commincation that definietely can not be handled through the DB, To be avoided whenever possible

Integration via DB, Preferred communication

implemented

S - IC   new

J. Bartholdt, B. Franke, C. Schwanninger M. Stal © Siemens AG, Corporate Technology

Conclusion

- SOA build a prominent, natural variation point with late (dynamic) binding capabilities
- Services as a variation point means flexible tooling available (Workflow engines, BPEL)
- Self-containment reduces coupling and fosters variation
- We will not follow the total unawareness of the usage context implied by SOA protagonists.

Future challenges
§ Data model can not be changed as long as old application components exist
§ Restructure the organization (nobody wants to loose influence, learning-curve)
§ Wrap legacy system with new service interface without side-effects

J. Bartholdt, B. Franke, C. Schwanninger M. Stal   © Siemens AG, Corporate Technology

Questions & Answers

Now, or later …

Joerg.Bartholdt@Siemens.com

J. Bartholdt, B. Franke, C. Schwanninger M. Stal   © Siemens AG, Corporate Technology