A Blockchain Use Case

The D_Marketplace
Agenda

1. Motivation & Use Case Introduction
2. UI / UX
3. Blockchain Benefits & Concerns
4. Solution Architecture Design
5. Technology Stack
6. Hyperledger Fabric Challenges
7. Q&A

Dr. Michael Kuperberg
Lead Blockchain Architect at Deutsche Bahn

Bertalan Vecsei
External Blockchain Architect at Deutsche Bahn
1. Motivation & Use Case Introduction

- Service procurement within an existing commercial relationship, for pay-as-you-go and fixed-scope services

- **Streamlining the process – disruptive if beneficial**
- End-user first: satisfaction in browser, on mobile devices, in Windows 10 apps

- Standardization, eliminating legacy dependencies and interfaces
- **Trustworthy** contract lifecycle and replicated storage; automation
- Transparency, auditability, improvement of *business performance*
- **Blockchain for trust, out-of-the-box security:** non-repudiation, tampering protection, etc.

- Marketplace approach and a „platform“ with high reuse potential
- Initial release: two *internal* business units (ca. 15 users), simple onboarding
- Vision: Blockchain Smart Contracts to trigger payments once these are due
2. UI / UX (Early Examples)

For business professionals caught between high OEM price and mediocre print and graphic output, there’s a solution: Business Express’s Eclipse line of compatible laser toner cartridges that meet or exceed OEM quality for 20% less than typical OEM price. While even brand name OEMs display “This cartridge may contain recycled components” on packaging.

**Angebot**
- **Preis**: 2SPf
- **Dauer**: 18 Wochen
- **Verfügbarkeit**: KW 21 / JPT p.W

**Bedarf**
- **Budget**: 50k
- **Deadline**: 31. Dez. 2017
- **Priority**: medium
3. Blockchain Benefits & Concerns

- Replication, cryptography out-of-the-box – but without guarantees! Efficiency, speed?
- Initially „Swedish tax records“ approach to privacy, unlike in traditional integration tools
- Early/hyped technology: cryptoassets (Bitcoin, ...) are the major public application → project risk, moving target
- Integration into orchestrators and of third-party systems: major challenge (cf. self-scheduling, endorsement, transactions/rollbacks, interfaces)

---

Monitoring, ease-of-deployment, long-term operations?
Block creation delays (throughput)?
Decryption progress over time?
Chain length growth?
Penetration and resiliency testing
Privacy laws („right to be forgotten“, data mining) vs. write-once-read-only and public blocks

Image source: http://fusion.kinja.com
4. Solution Architecture Design
5. Technology Stack

- Amazon EC2 Cloud
- **Hyperledger Blockchain** (Docker, Java chain code, Node.js SDK, Node.js MarketPlace App Interface)
- Standard **MEAN Web App** Framework (Mongo, Express, Angular.js, Node.js)
- **Web Technologies** (HTML5, CSS, etc.) & Frameworks (Bootstrap, SASS, jQuery, etc.) & UX Design (Sketch)
- Stateful Process Orchestration & Workflow Component (**iBPMS**)
- Integration via Event Driven & Service Oriented **Middleware** Technologies (REST, Swagger, ActiveMQ, JBoss MW, etc.)
- **Document Automation** (XSL-FO, Apache FOP, etc.)
- **Legacy** Application Technologies
- **Microsoft Surface** (C++ / C# / Microsoft Visual Basic / JavaScript & HTML5)

*Note: NO Hyperledger Composer FW at this stage (wait until Fabric has an official Release on v1 and Composer catches up).*
6. Hyperledger Fabric Challenges

- DB & Procedures
- Hype
- IBM
- Plug & Play
- Open Source
- Integration
- Interface (Lib vs. Service)
- Fabric Capabilities & Stability
- Content Objects
- OffChain vs. OnChain Logic
- Security (Privacy)
- Right To Be Forgotten
- "Cool" Capabilities
- Migration, Replay
- Performance
- Know-how
- Consensus & Concurrency
- Chaincode Lifecycle Mgmt.