Component name (official, internal, abbreviation etc.)

Short description

Main stakeholders (Initiator, product mgt, dev, ops, users, support ...)

Context diagram (Scope with users, external systems und main use cases)

Provided capabilities (queryable information, business functionality, delivered events, running jobs, etc.)

_____

_____

_____

_____

_____

_____

_____

_____

_____

Responsibility / accountability (in person)

TOP 3 challenges

1. _____
2. _____
3. _____

TOP 3 quality goals

1. _____
2. _____
3. _____

Technical key facts

Component age: _____ Component size: _____

Version (used): _____ Version (current): _____

Customizing proportion: _____ Development proportion: _____

Licensing/Support: _____

Main tech stack: _____

_____

Integration tech.: _____

_____

Operation platform: _____

_____

Hardware: _____

Collaboration with other components

(Component as a service provider and service consumer)

Inbound ► Component ► Outbound

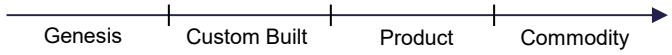
Strategic classification

Importance for the business
- □ Core
- □ Supporting
- □ Generic
- □ Other

Visibility on C-level
- □ Enabler
- □ Asset
- □ Profit Center
- □ Cost Center

Visibility for customers
- □ Differentiator
- □ Money earner
- □ Necessity
- □ None

Evolution of component



Development process, teams and activities

Teams key facts

Name(s): _____ Number: _____ Size: _____ Age: _____ Stability: _____

Process
- □ Random
- □ Waterfall
- □ Agile / Kanban
- □ Continuous delivery

Drive Mode
- □ Problem-driven
- □ Feature-driven
- □ Data-driven
- □ Strategy-driven

Organization
- □ Individual
- □ Hierarchy / silos
- □ Cross-functional
- □ BizDevOps/SRE

Types
- □ Stream-aligned
- □ Enabling
- □ Complicated sub.
- □ Platform

Personalities
- □ Pioneers
- □ Seedlers
- □ Town planners
- □ Mixed

Delivery
- Change/release frequency: _____x per _____
- Duration "Idea to delivery": _____
- Time for bug fixing: _____
- Fail rate: _____

Best Practices
- □ Version control system
- □ Build automation
- □ Continuous integration
- □ Test automation
- □ Code reviews
- □ Static code analysis
- □ Architecture reviews
- □ Retrospectives
- □ Architecture documentation