Initial Architecture Government Cloud Statens IT

This document is an initial architectur for a project aiming at establishing the first version of a new development and operations model at Statens IT based on cloud technologies and supporting agile development. The architecture describes which decisions that has been made and why, for each element in the reference model. Starting with the most salient for the first known applications (DMI, Myndighedsfortegnelse)

Date	Version	Usages
20180903	0.1	Used for first workshop 20180906
20180911	**0.2**	Updated with input from DMI and SIT used for second workshop
20180927?	1.0	Support for go-decision 20181003

1 Introduction

GovCloud ideas has been around for years, and in the spring of 2018 the planets alligned in a way that allowed a small group of people to conceptualize the idea. Digitaliseringsstyrelsen initiated a coolaboration between the Danish Meteorological Institute and Statens IT that lead to this architecture.

The purpose of Statens IT (SIT, eng. Danish Agency for Governmental IT Services) is to provide better, safer, and more cost-efficient it operation and service to the Danish Government. SIT has focus on accessibility, stability, efficiency and information security in four main areas of operation: personal computers for state employees, it infrastructure, standardisation of generic administrative it and standarised vendor contracts. Currently SIT services 12 of 19 ministeries with around 17.000 employees at 250 physical locations, and is scheduled to expand to more ministeries for a total of 35.000 employes. Ministery of Defense, The Police and The Tax Authorities will then be the only areas with seperate it operations.

Digitaliseringsstyrelsen (DIGST, eng. Agency for Digitisation) was established in 2011 with joint public funding. Its purpose is to create and implement initiatives and policy regarding the digitalisation of the public sector. DIGST supports public sector efficency by providing digital services to citizens, businesses and government organisations. DIGST implements joint public projects and is host for The Danish Council for IT projects and the government it project model.

DIGST and SIT are both part of the Ministery of Finance and employes about 250 people each.

The Digital Strategy 2016-2020 is a joint public strategy for all levels of governement: state, regional and local. Among the many initiatives in strategy to be supported by this architecture are *8.1 Good data and efficient data sharing* and *8.2 Robust operation of the common infrastructure*. During the realisation of the Digital Strategy it has become evident that public authorities on all level of government is struggeling with **time-to-market for new applications and integrations** that does not match the ambitions of integrated public service delivery, the ever-changing political and regulatory landscape and the technological development in general.

A number of barriers to the change of pace has been identified [more about how?].

- Compliance to legal and strategic requirements is a very time consuming aspect and has been extending both large and smaller projects with periods of years. Public procurement, personal data protection and protection of national infrastructure are highly regulated areas, but also characterized by generic and re-usable concrete elements. Strategies are developed on many levels and in many domains, thus a single project or application will be covered by many principles and requirements. Sometimes the strategies align well,

other times not so much.

- Budget scope. Typically data are stored in one application financed by the data reponsible, but can be used by another organisation. Hence the cost usually appear on one budget, whereas the benefits is to be found at another.

- Small professional environments. Public administrative organisations are too small to carry a full professional environment for efficient/agile development. Even that IT plays an ever big role for efficient government, very few organisations has managed to build and maintain their own development.

- Sparse knowledge about application and data landscapes of other organisations. Even when organisation cooperate and depend on each other information about existing application and data with potential reuse is hard to come by.

- Service differentiation. Cloud technology is maturing but product categories, and more important their interfaces, are still plenty. Interoperability and 'moveability' of applications is typically not a priority in an emergent market where service differentation is a driver. This results in a big challenge for the individual organisation and puts them in risk of significant vendor lock-in.

[RAD 'fælles service- og data-platform'! Seperate distribution of data from processes that produce]

2 First Applications

The GovCloud roadmap primarily support identified applications and secondly The Digital Strategy 2016-2020.

Paramount to establishment a GovCloud is the succesfull support of an intial large scale strategic project to proof that the architecture provides a solution to the identified problems.

[DMI as external partner and consumer of platform] [DIGST as consumer to support experience in governance processes] [SIT as consumer of their own platform]

DMI Data
[Describe data to be shared, volume and latency]

[Describe the need for API keys]

PubOrg
[Describe data to be shared, volume and latency]
[Describe few sources and first know usages]

Platform DevOps
[Back Log is prioritized joint public]

Initial

	Providers	**In**	**Comp.**	**Storage**	**Out**	**C**
Observations	DMI	?mb/min stream	-	?GB	?	P c
Forecast	DMI	?GB dump 4 hour	++	?GB	?	P c
PubOrg	KL, SM, DM	1GB/day	-	25GB	1GB/day	D o G D S
Platform	SIT	?GB/day	+	?GB	?GB/day	D D

Future

[Insert graph with guestimates and add application X, Y, Z]

3 Problem

An architecture is a solution to a problem.

SITs customers (ministeries and their agencies) are moving towards modern, cloud based platforms to improve time to market for new applications and integrations. SIT currently has no mature, formalised offering to meet this need. The problem can be describe as:

What service offering will be the most attractive option for public agencies seeking the advantages of 'the cloud'?

This document describes an initial architecture for a government cloud aimed at becoming customers prefered Operational Model at SIT. It consists of a blue print for a technical solution, a new governance model and a perspective for a future where SIT acts as a cloud broker.

SIT has a small number of suggested operational models that today provides the stability and security required by the customers. This new model is believed to be prefered over existing because of the following charactaristics:

- *Continuity of Service.* Existing models allow small and planned windows for maintainance. Cloud technologies typically offers a continuity with rolling updates of both platform and applications without any interuption.

- *Shorter Development and Rapid Changes.* Based on experiences from the proof of concept, customers can expect a rapid deployment cycle. If the suggested goals for deployment and changes are met (see later), the GovCloud will match existing cloud vendors.

To support the first applications SIT most extend existing operational model with new high level use cases.

- *Public Data Distribution.* A private company wants to access data from a public organisation. SIT has a suitable operation model and technology platform to support the intentions in the directive on the re-use of public sector information. The data holding government organisation leaves the operational aspects to SIT and focuses on tailoring the data service applicatin and help the private users to understand the data.

- *Application Development.* A governement organisation decides that GovCloud is the future home for a new application. Early in the process SIT helps to identify reusable services and data on the GovCloud. SIT provides a sandbox environment and a toolchain for developers to produce initial proof of concepts. DIGST supports the project

Furture versions of the GovCloud is expected to focus on *Compliance as a Service.*, *Development procurment framework*, *Sharing of sensistive data*, *Reuse Application/Service*, *Migrate existing application*. [See appendix]

4 Principles

Good architecture is based on agreed upon principles that can guide solutions to future unforseen problems, allow the GovCloud to evolve with less escalation of desicisions, and clarify where substantial diagreement exists. The strategic principle supporting the GovCloud is:

Cloud First - new applications and new integrations between existing applications are designed to take advantage of the characteristics of cloud computing.

Cloud technology is maturing and has proven to support large scale operations and rapid development. For the purpose of this document we focus on the following essential charateristics: Consumer on-demand self-service, Broad network access, Provider resource pooling, Rapid elasticity and Measured services. A detailed describition can be found in the definition from NIST.

The Cloud First principles is further refined by a few principples specific to the use of cloud technologies in a danish governement setting.

Vendor neutral - applications and data shall be movable from one cloud platform to another without unreasonable effort.

Public agencies can not allow there data or application to be locked-in to a specific vendor. From a regulatory view public procurement should support competition and not favour a specific propitary tehcnology or sole vendor. From a strategic perspective public services should be possbile to move between organisation, and to be intergrated into new operational landscapes. This principle supports the provider resource pooling characteristic.

Exposed interfaces are standardized and supported by Open Source. Interfaces towards customers and interfaces exposed internally between platform and applications shall be selected regards to their support in the open source community. Selecting open source supported interfaces ensures a licens-free fall back option, if the benefits of licensed implementions are not balanced with the price. Standardization ensure a clear governance of changes and transparent terms of use.

Applications are containarized. By choosing widely adopted container technologies the binding between applications and operation environment is broken. Containarized applications declare their required ressources and relies on configurations from their environement.

Propriatary technolgies are used with caution. When used, the imidiate benefits needs to be evaluated against the later cost from the binding. A part of the evaluation is to describe a likely exit strategy for replacing propiatary technology with more open.

Continious service availabilty - services run 24/7 even during new service releases and platform upgrades.

It solutions used by public agencies are often critical, not only to the agency itself but to the society in general. Traditionally some services had so called service windows, where service was suspended and new software relases could be rolled out. Digital self-services have changes this and the public is serving it self around the clock at times suitable for the costumer and not for the service provider.

Deployed services run continuously and must be build accordingly. Cloud technology can enforce some of this, but services must be designed for high availabilty, automated testing and detailed monitoring.

Problems must be solved imidiately and in close collaboration between platform consumers, platform providers and software vendors. The GovCloud provides clear seperation of responsibilities, but recognizes that solutions are found in collaboration, and the collaboration must be supported efficiently by it.

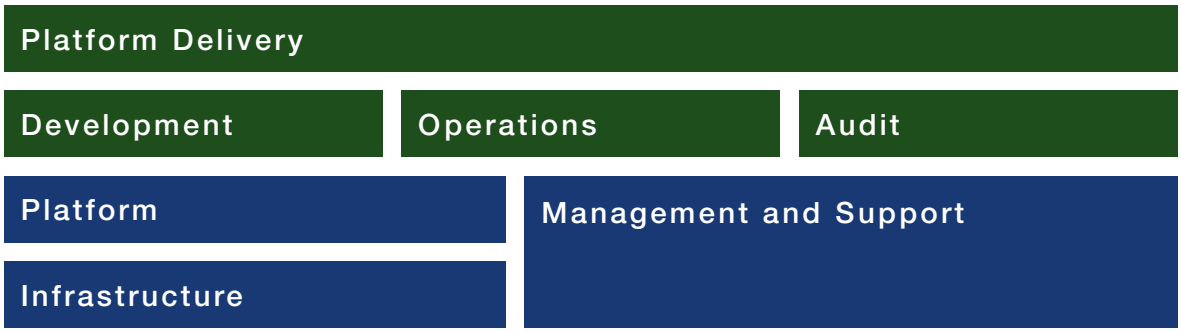
High availability is not only a concern for the platform it self, but is a result of carefull planing across all dependencies from electrical power, bandwidth, incident reaction time and capacity to solve problems when they arise.

Service availability is an extension of the cloud characteristica of measured services. THe health of the platform and it's individual services most be continiously monitored

In future version new principles can be included e.g. *Cloud Broker*, *Compliance As A Service*, *Reuse of data and applications*.[See appendix]

5 Reference model

To identify the different components in the architecture, we use a conceptual model based on a number of excisting frameworks, including
Microsoft Cloud Service Foundation Reference Model, IBM Cloud Garage Method, OpenGroup IT4IT and NIST Cloud Computing Reference Architecture. A total detailed view of all elemements can be found in appendix of this document.



[include roles Cloud Consumer, Cloud Provider, Cloud Broker, Application Responsible, Application Developer, Data Responsible. And correct use in the business area description.]

Four areas of business…[compress the descriptions to a minimum]

Platform Delivery
This area is responsible for translating consumer requirements into platform services. Capabilities and technologies are provided with a specific service level agreement. All changes follow a transparent governance process with consumer involvement,.

Development
This area translate customer requirements into application services. Development is primarily done by third party developers, but can also be done by employees at a consumer. Platform services follow the same processes and are developed by SIT or third parties. Application services can generalised into platform services in a collaboration.

Operations
This area encapsules processes applied to the platform and its applications to meet the requirement in service level agreements. Processes are typically highly automated to minimize human error and labor. [end user support? behaviour?]

Processes in Development and Operations are increasingly considered as overlapping, especially in the case of self service cloud services, where a consumers take a larger part in the operational aspects of service delivery. One definition of DevOps *is a set of practices*

intended to reduce the time between committing a change to a system and the change being placed into normal production, while ensuring high quality Bass, Weber and Limin. 2015.

Audit
Auditing are processes where independent examination through controls with the intent to express an opinion. On opinion can be on the compliance with standards. Audits are based on objective evidence and can evaluate terms of security controls, privacy impact, performance, etc.

The four business areas above are supported by technology components in three areas [consider using IT4IT terms]:

Management and support
The components in this area supports the capabilities defined by the DevOps business area and supports the consumer self-service characteristics of a cloud. Components also provides data to Platform and Service Delivery supporting Meassured Services and fair billing.

Platform
Platform components are aggregated with Infrastructure and Management and Support components to provide platform as a service (PaaS) services. Platform components are mostly provided as services that are consumed by applications, but they may also be consumed directly by end-users.

Infrastructure
Infrastructure components provide the necessary environment to run the platform. Infrastructure can be on-premise or extended with Infrastructure as a Service from an private provider. Infrastructure is heavily-standardized to facilitate both automation in the environment, and to optimize volume purchases of hardware and software.

6 Design

The initial design of the GovCloud consists of a number of design decisions and the rationals behind them. During the establishment decisions can be changed following the established service lifecycle management processes.

6.1 Service Delivery

Statens IT is the strategic appointed provider of IT operation and service to the Danish Government and shall as such provide the requested cloud environment.

[PaaS] Government Cloud is a Platform as a Service offered by SIT within existing license to operate.

NIST defines the capability provied to the consumer in the PaaS operational model as: *to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages, libraries, services, and tools supported by the provider. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly configuration settings for the application-hosting environment.*

Consumers are responsible for the developement and maintainance of applications, data management, awarding and revoking user rights and licensing of software. SIT as a platform provider is responsible for updating platform software, automated testing of services after update, monitoring and reporting on services.

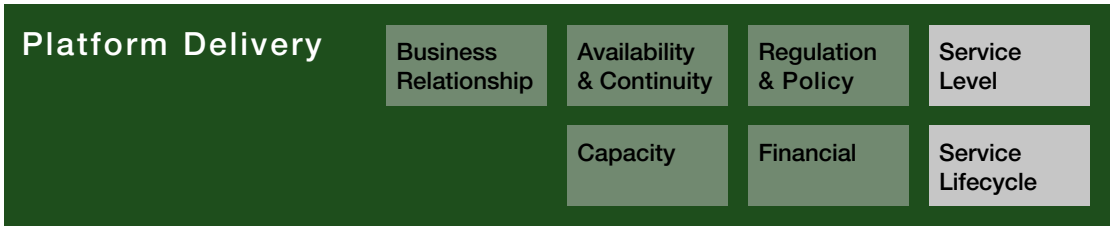
PaaS is choosen over IaaS to limit the ammount of operational knowledge required by government organisations. SIt is setup to lift this requirement from other organisations. [more?] SaaS is an existing offering from SIT and will still be provided. SIT can use the GovCloud as a basis for SaaS offerings to further harmonize the operational environment.

[Operational Model] SIT defines a new Operational Model (da. driftsmodel) in the existing framework.

SIT has established business support functions across a few operational models. Initial PaaS will be a supplement to existing models and will benefit from establihed practices around agreements, reporting and financing.

[do we need to change anything in the SIT regulatory setting?]

For the initial applications processes related the management of Platform Service Lifecycle and Platform Service Level needs to be established.



6.1.1 Service Lifecycle

> - Requirements for many of the processes that are defined in the Service Operations subdomain components in this article.
> - Requirements for many of the technical capabilities that are defined in the Management and Support subdomain components in this article.
> - The plan for how these requirements are to be met. While the processes and technical capability requirements might differ across services, they generally do not dramatically differ, because most organizations define a standard, and then try to adhere to it as much as possible.

[Customer driven] The service life cycle of GovCloud is based on customer needs, in a baglog maintained by SIT, and prioritized by a joint public group lead by DIGST**

> Rational from vendor neutrality, ownership to both 'business' and 'platform' direction

[Support lifetime] Each technology, interface and platform service in GovCloud has a planned support lifetime

> rational from customers need to plan ahead… and from the need to be able to face out

6.1.2 Service Level

> This component is a key enabler for customer satisfaction and results in the service level agreement (SLA) for the service, which is created from the outcomes of many of the components in this subdomain. This component has a close relationship to both the Business Relationship Management component and all Service Operations components.

> Describe what should be in the operational model. Check existing SITs]

> - Uptime for services
> - Internet bandwidth
> - Notfication after platform update
> - Cost model
> - Security notification processes
> - Expected time from Artefact delivery to deployment
> - Automated test requirements and criteria for automated deployment of artefacts.
> - Scale up/down criteria

[24/7 platform] GovCloud includes 24/7 support on operations on infrastructure and platform

[24/7 services] Consumers wanting to host 24/7 application most provide ressources to collaborated on solutions to incidents. [2nd level? Developer?, Check up on NSP]

> SIT can restart, but not change configuration or code without consumers…

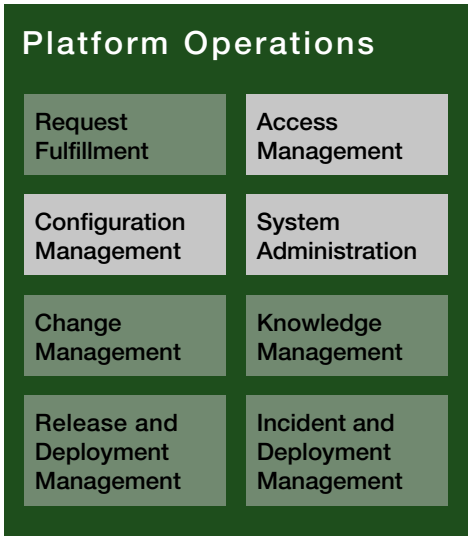
6.2 Platform Operations

> Existing processes are not suitable for DevOps and automation. New variants of existing processes are needed. ITIL and IT4IT are well supported in existing tools

> - terminology of applications, services, pods, image, artefact]

[ITIL] SIT (GovCloud team) establishes processes suitable for cloud platform operations based on the ITIL framework [Check mapping to DevOps].

> Rationale to minimize Employee training, but new because cloud is different so not a direct reuse]



6.2.1 Access Management

[Access] SIT (GovCloud team) establishes new internal and external roles to assigned to users with access to DevOps processes on the platform.

6.2.2 System Administration

[SysAdm] SIT (GovCloud team) establishes new internal and external roles to assigned to users with access to DevOps processes on the platform.

> Initial tools chosen with aim of automation, CLI, rmeote management

6.2.3 Configuration Management

[Config] Configuration of services and platform as different processes.

> Initial tools chosen with to later support self service for application configuration and changes

6.3 Service DevOps

[SharedOps] The platform operator and service operators share dashboards, information and tools to deploy and monitor services [incl online collaboration tool?].

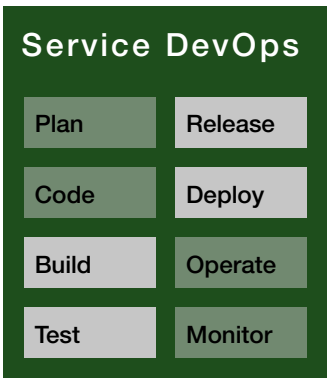
[Toolchain] Consumers can choose between SITs SaaS toolchain or using their own integrating build and test from SITs.

[SIT Tool] SIT offers a Development Toolchain to support agile application development. It is offered as Software as a Service under existing operational model is based on Confluence, Jira, Git, Build?, Test? [Consider license free tools…]

> Rational from efficiency and synergy and not bloating the platform project
>
> - harmonization
> - time-to-market
> - integrated with identity management
> - to allow alternative toolchains at consumers

[Sandbox] The Government Cloud is born with a free-of-charge sand boxe for each existing customer

> rational from time to market and spreading the news…]



6.3.1 Build

[Build Image] Docker images are build from source using custom image maintained by SIT

> Run identical builds on local development

6.3.2 Test

[Test Image] Automated test of services are done using a custom image maintained by SIT

> Run identical builds on local development

[Acceptance test] Service passing the provided automated tests are considered running

> to allow SIT to update platform

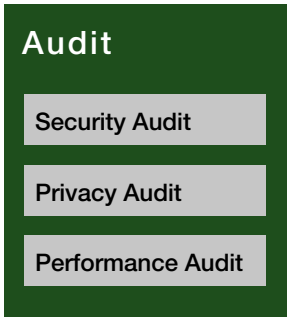
6.3.3 Release

[Release?] follows from PaaS or?

6.3.4 Release

[Elastic?] Every service is deployed with a scaling schedule?

6.4 Audit



6.4.1 Security Audit

[Tænk GovCert]

6.4.2 Privacy Audit

[Tænk Datatilsynet]

6.4.3 Performance Audit

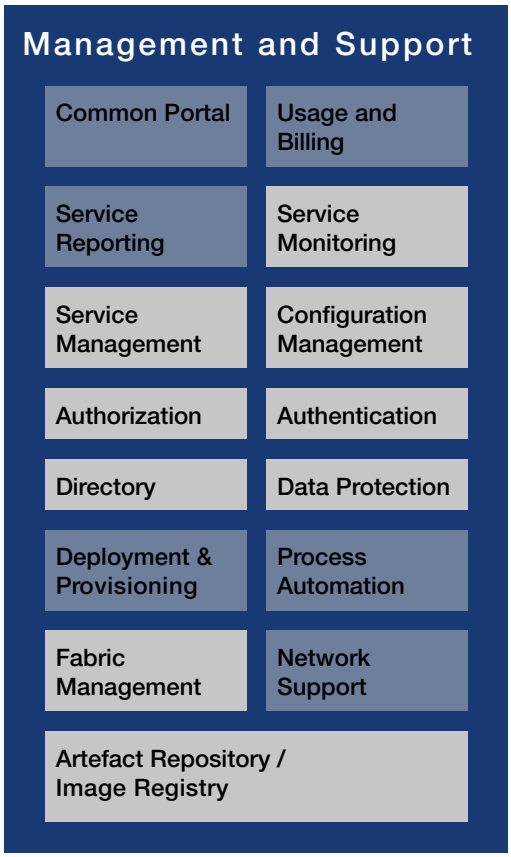
[Tænk SIT og find den der 'sviner']

6.5 Management and support

> Describe intention about DevOps through selfservice, Same views and (mostly) controls as SIT employee.

- No Portal in first version
- No detailed per usage Billing in first version
- Process Automation (left to the team to build ad hoc)
- Authorization (not needed in the first two applications. Future services can use attributes from datasets and roles/rights from AD)
- Data protection (use build in platform, but still processes)???
- Deployment and Provisioning (reuse existing)
- Network Support (use existing)

[Support] Deployment and provising of hardware, network support and data protection…



6.5.1 Service Monitoring

[Monitoring] Applications and services implement a common schema for monitoring, logging… and are monitored and reported in SITs existing tool (Nagios?)|

> Monitoring adskilt fra Reporting
> [Tegning af service i container]
> [Ping and trace]
> [Log content and format]

> Ping, Trace, CallId

6.5.2 Service management

> This component consumes Service Monitoring data and produces reports that describe the actual service level metric values exhibited by a service over regular time intervals. The report data can be compared to SLAs to determine whether the service met its SLAs during the reporting interval that is specified in the SLA. The data from this component is provided to consumers through the Consumer and Provider Portal component. This component is a primary enabler for the Service Level Management and Business Relationship Management components.

> This component monitors service levels of all technical capabilities that are used to provide each cloud service. The Service Reporting component consumes the data from this component. Optimally, the Service Monitoring Component is able to integrate with the Service Management component so that it can auto-generate incidents based on defined criteria.

> This component supports most of the Service Operations components and integrates data from the following Management and Support components:

> - Incident and Problem Management
> - Configuration Management
> - Service Monitoring
> - Service Reporting

> The data integrated by the Service Management component is typically exposed through the Consumer Portal component so that various individuals can view it.

6.5.3 Configuration Monitoring

6.5.4 Authorization

[Rights] All access rights (end-users and other services) are done by service or data responsible

[Authorization] Access policy on service level is enforced in Gateway, Access policy on data level in Service

> consequence that SIT needs to know API keys.

[OAuth2?]

6.5.5 Authentication

[Fed] Authentication of end-users are done in feuderation

> SIT established trust to idp

[Attribute Based Access Policy] only?

6.5.6 Directory

[AD] Users, Applications and Dataset are ressources registered in SITs existing Active

Directory

Only employees and platform devops. Other users in trusted datasets.

6.5.7 Data protection

6.5.8 Fabric Management

6.5.9 Artifact repository

[Registry] The artefact repository is Docker Registry Server**

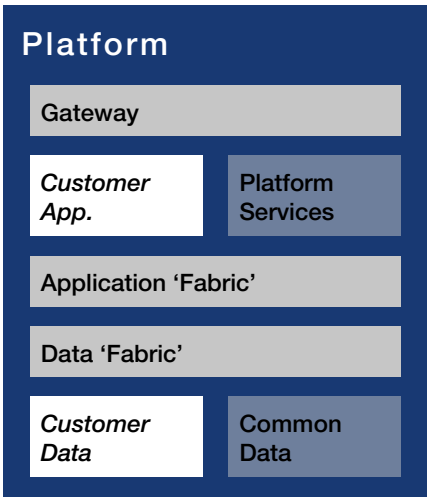
Configuration of environments are done via management.

Artefact are identified in the AD, semantic versioned and taged (production, preprod, latest…)

6.6 Platform

[Layers] Three seperate layers: gateway, application, data

> rationale from compliance, moveability, data management



[Vert?] Either common or Consumer.

6.6.1 Gateway

[KrakenD] The API Gateway is KrakenD

> rational deom scaleability, distributed configuration, OAuth

> SSL termination,Consider ClientMaxRate, Act as circuit braker, Operated by Cloud Platform Team

Sandboxes outside and unprotected.

Test inside but seperate gateway?

6.6.2 Application fabric

[Kubernetes] The application environment is Kubernetes

> rationale: No need for management tools yet, Kubernetes is the core technology, Selfservice and automation is build on top in future versions.

[Service classification] Service layers: data, business, frontend

Relation to gateway

Compute Ressource Quota

Application consist of a few controllers with pods with images.

6.6.3 Data fabric

[MapR] The data fabric is MapR

> rational: quick start, manageability,access control, scale out, clustering

6.7 Infrastructure

[Scale out] Scale out and multiple cloud providers as needed, but not initial**

- on premise because we want to test and prepare SIT to handle sensitive data

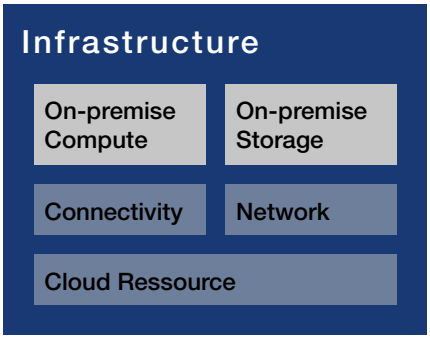
[Consider Hardware certification??]

[On-premise] The on-premise application and data fabric runs on non-virtualized hardware under CentOS**

- no Virtualization to reduce complexity/licensing and performance storage

- same os to minimize learning, but specific to cloud, but seperate for tuning

[decision?]Active/Active Cross..



6.7.1 On-premise Compute

> This component represents physical servers, which include resources such as processors, graphics processing units (GPUs), random access memory (RAM), network interfaces, and the storage necessary to host hypervisor software for the physical server.

> This component represents physical storage that is accessed by Compute devices via some networking technology. Types of data that are often stored on this component are virtual machine hard disk files or consumer data or both. Although the Infrastructure and Management and Support components that are necessary to provide services in the infrastructure as a service (IaaS) service model1 exist in the CSFRM, they are not represented solely to enable IaaS services. They’re represented because they’re necessary to enable any type of cloud service.

6.7.2 On-premise storage

6.7.3 Network

> This component represents the physical network switches, routers, firewalls, and cabling. It also represents logical networking constructs including virtual local area networks (LANs), access control lists, quality of service, and network interfaces defined in converged network architectures.

[Network in Kubernetes]

**# What kind of seperation? **

[What about IP6 internally/eternaly]

7 Appendix

7.1 2019Q2

Platform goals for first version:

- From prototype to production < 1w, Chages in production < 1d
    - supported by toolchain.
    - build and test shared by dev and ops.
- Rebuild services on platform updates
    - automated test and acceptance.
    - multiple versions of same services.
- Performance and near liniar scaling
    - Service monitoring and reporting
    - ressource limiting

7.2 Future releases

7.2.1 Principles

Cloud Broker - cloud services are consumed through a central cloud broker.

Regulatory issues on the use of public cloud services for sensitive personal data in a government settings are still unclear. Even with clearification, some critical data are still to be kept on danish territory and under strict control. But even critical applications need better time to market and cloud technologies can support this. Hence the need for a onpremise infrastructure based on cloud technologies.

At the same time the cost efficiency of public cloud offerings are hard to match. But government organisations will soon find integration of cloud services can be too complexto manage. A cloud consumer may request cloud services from a cloud broker, instead of contacting a cloud provider directly. A cloud broker is an entity that manages the use, performance and delivery of cloud services and negotiates relationships between cloud providers and cloud consumers. NIST definition of Cloud Broker

The cloud broker will take advantage of a competitative market by moving applications to the

most best available platforms. Reason can be applications with no personal data or when needed capabilities are only accessible as SaaS at cloud providers.

Furture versions of the GovCloud is expected to focus on

- *Compliance as a Service.* The GovCloud will differ from existing cloud vendors in regard to establising and assuring compliance with regulatory and strategy requirements. Clarification of compliance can add years to project time and is a major barriere to better time to market. The GovCloud will over time establish more compliance assurances and offer them as shared services. The focus will initial be on data protection, public procurement, and the joint public digitalisation strtegy.

- *Development procurment framework* Since the GovCloud is expected to be used by many public organisations, the general software and development market will be familiarized with the environment and it processes. In a market perspective, the GovCloud can function as a harmonization of services, which can lead to an increased and more transparent competition. This can further be supported by a common framework for procuring application development build to be hosted on the GovCloud.

to support a number of new use cases

- *Sharing of sensistive data*
  An existing customer at SIT wants to access data from another organisation. The data is not available at any existing dataservices. SIT suggests the data owner to replicate the data onto the GovCloud. The data user builds a custom dataservice and reuses existing security controls incl identity management.

- *Reuse Application/Service.* A software vendor delivers an application to a customer of SIT. The vendor wants to deploy on the GovCloud preparing for more customers. The vendor sells a license to another SIT customer and SIT extends the installation and the software product is available to the new customer.

- *Migrate existing application.* SIT has identified a customer application that is sutiable for hosting on the new GovCloud. SIT plans a transition with the customer and reuse a number of exsiting services and datasets in the Cloud and can turn off some application elements.

- Formalized operation model

- self service and DevOps

- Consumer Portal
- Authorisation and Authentication
- Automated Billing

7.3 Processes

> Processes likely to be supported in existing Service Now tool
>
> - Request Fulfillment
> - Change Management
> - Knowledge Management
> - Incident and Problem Management

> Processes likely to be supported by new tools decided by the GovCloud Team
>
> - Access Management
> - Asset and Configuration Management
> - Systems Administration
> - Release and Deployment Management

> and…
>
> - Create identities for Dataset and Application Instance
> - Re-build, Test
> - Deploy
> - Capacity and Scaling
> - Maintain images

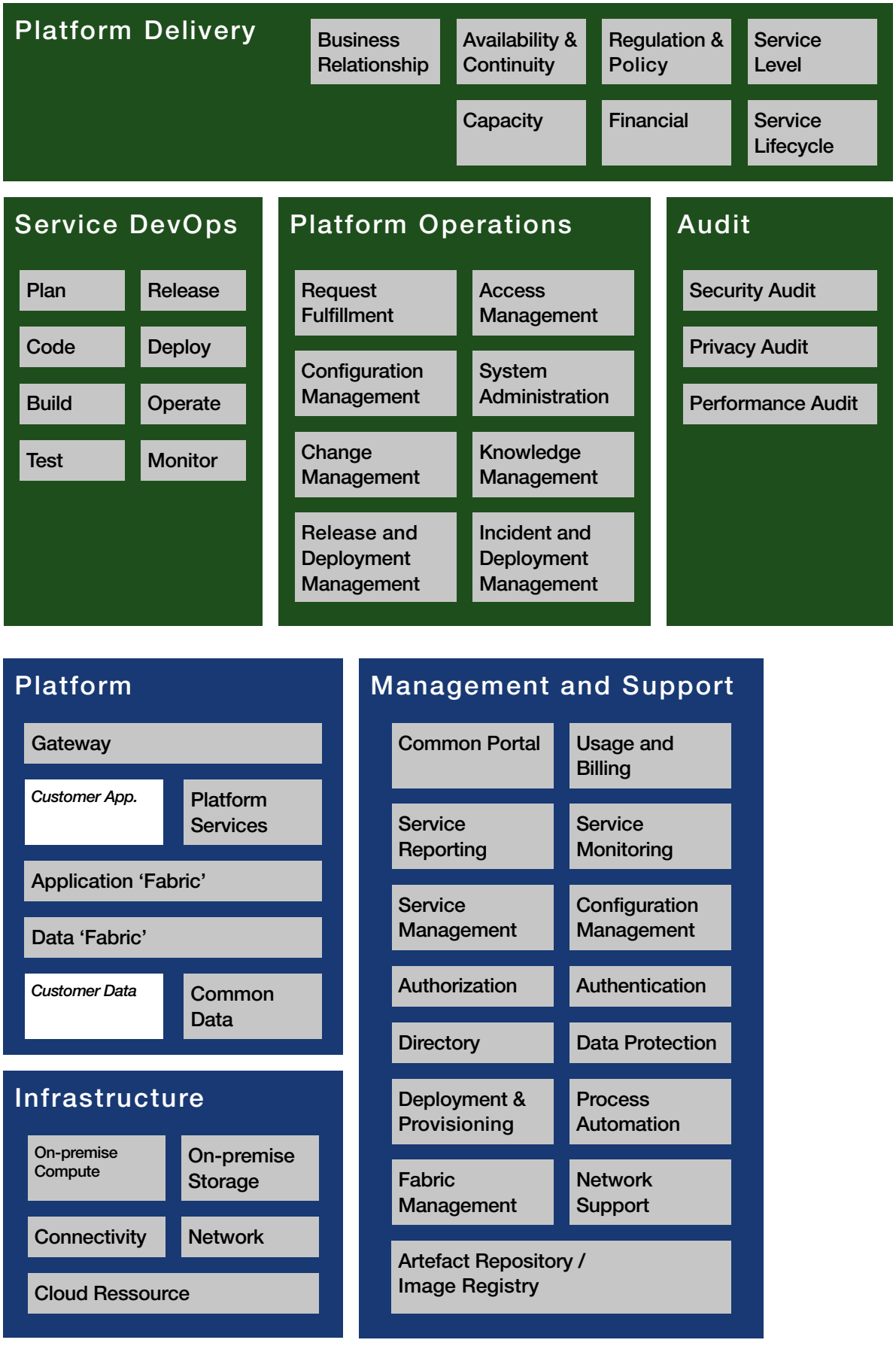
> Sign up as Customer
>
> - Establish electronic identities
> - Create data-domain (Volume in MapR)
> - Assign rights to Customer (SIT User with ACR)
> - Create a Data Collection
> - Describe using template → Publish
> - Decide security controls
> - Data controller agreement with Platform
> - Plan additional controls?
> - Place in data-domain and establish DEV/TEST/PROD

> Sign up as a Developer?

> - Establish electronic identities
> - Platform GIT project
> - Education/Guidelines/Helloworld?
> - Licens/Ownership?

> Request changes on platform? create topics…? create volumes?
>
> - Deploy one or more service instance
> - Identify on GIT
> - Test criteria
> - Data agreements
> - SLA/Monitoring
> - Sign up as Service Consumer
> - Agreement with service owner, data responsible, SIT
> - Firewall?
> - Licensing?

7.4 Referencemodel


Platform Delivery
Business Relationship
Availability & Continuity
Regulation & Policy
Service Level
Capacity
Financial
Service Lifecycle
Service DevOps
Plan
Release
Code
Deploy
Build
Operate
Test
Monitor
Platform Operations
Request Fulfillment
Access Management
Configuration Management
System Administration
Change Management
Knowledge Management
Release and Deployment Management
Incident and Deployment Management
Audit
Security Audit
Privacy Audit
Performance Audit
Platform
Gateway
Customer App.
Platform Services
Application 'Fabric'
Data 'Fabric'
Customer Data
Common Data
Infrastructure
On-premise Compute
On-premise Storage
Connectivity
Network
Cloud Ressource
Management and Support
Common Portal
Usage and Billing
Service Reporting
Service Monitoring
Service Management
Configuration Management
Authorization
Authentication
Directory
Data Protection
Deployment & Provisioning
Process Automation
Fabric Management
Network Support
Artefact Repository / Image Registry