IT Project Guidance

On Onboarding Users

Version:

0.1

## Purpose

This document defines the key patterns, responsibilities, and decision points involved in provisioning users into a system or service environment. It addresses the mechanisms by which individuals are enabled to access digital services, associate with groups, and assume roles. The guidance is designed to support system architects, service designers, and product owners in selecting appropriate provisioning strategies that balance usability, trust, governance, and long-term maintainability.

## Synopsis

Provisioning is the act of creating or enabling user accounts and associating them with the roles, groups, or permissions required to use a system. This document explores the common provisioning approaches available for public sector and education-aligned services, including self-registration, invitations, federated login with Just-In-Time provisioning, SCIM-based synchronisation, and administrator-driven bulk onboarding. It contrasts the strengths and limitations of each approach, offering considerations for trust boundaries, auditability, automation potential, and risk. The document forms part of a wider body of IT Project Guidance designed to improve architectural coherence, operational readiness, and service integrity across evolving digital environments.

## Contents

[Purpose 1](#_Toc195791704)

[Synopsis 1](#_Toc195791705)

[Contents 2](#_Toc195791706)

[Introduction 3](#_Toc195791707)

[Background of this work 3](#_Toc195791708)

[Context 3](#_Toc195791709)

[Background of issue 3](#_Toc195791710)

[Problem Statement 3](#_Toc195791711)

[H3 4](#_Toc195791712)

[H4 4](#_Toc195791713)

[Appendices 5](#_Toc195791714)

[Appendix A - Document Information 5](#_Toc195791715)

[Versions 5](#_Toc195791716)

[Images 5](#_Toc195791717)

[Tables 5](#_Toc195791718)

[References 5](#_Toc195791719)

[Review Distribution 5](#_Toc195791720)

[Audience 5](#_Toc195791721)

[Structure 5](#_Toc195791722)

[Diagrams 6](#_Toc195791723)

[Acronyms 6](#_Toc195791724)

[Terms 6](#_Toc195791725)

# Purpose and Audience

The purpose of this guidance is to establish a clear, technology-independent understanding of provisioning as a foundational capability within digital services. It clarifies the distinction between authentication and provisioning, and emphasises the need for deliberate, accountable processes that govern how users are introduced to systems, assigned roles, and associated with groups. The guidance supports both greenfield and transitional projects, including those integrating federated identity providers, internal directories, or third-party brokers.

The intended audience includes system architects, product owners, service designers, and implementation teams responsible for onboarding users into a system. It is also relevant to governance roles such as information stewards, privacy officers, and audit leads who require assurance that provisioning processes align with policy, accountability, and identity lifecycle expectations. The document assumes moderate familiarity with digital service design and identity concepts, but does not require deep technical knowledge of specific protocols.

# Scope

This guidance covers user provisioning as it applies to digital services that rely on defined roles and group memberships to control access and coordinate activity. It addresses the technical and procedural mechanisms by which individuals are introduced to a system, linked to identities, granted access, and made operationally visible to other users or systems.

The scope includes both interactive user provisioning—such as self-registration, invitations, and delegated group management—and automated provisioning methods such as SCIM integration and Just-In-Time (JIT) account creation through federated login. It also considers transitional states, such as initial bulk onboarding or re-provisioning during system migration.

Excluded from scope are the detailed mechanics of authentication protocols (e.g. OIDC, SAML), role-based access control models, and user deprovisioning or removal. These are addressed in separate guidance documents. However, the document does assume an architectural context where provisioning decisions impact trust boundaries, group visibility, and attribute correlation across systems.

# Background

Digital services are used by individuals in a range of contexts: some act independently, others as part of small informal groups, and many as members of structured organisations. Provisioning models must support all three. The traditional assumption that every user belongs to a single organisation—often modelled as a “tenant”—once underpinned many access control strategies. These tenants were treated as secure, self-contained environments, with isolated directories and dedicated admin models. However, in modern service contexts, this model introduces friction and rigidity. It becomes a barrier when users need to participate across multiple organisations—such as contractors, advisors, or individuals with legitimate roles in different institutions—forcing them to manage multiple identities, passwords, and histories.

A more flexible view recognises that the simplest entry point is the individual end user, signing up to a service and creating a personal account. This is the most widely understood starting point in commercial services: a single user registers, agrees to terms, and begins using the system independently, often linked to billing or subscription. From there, additional layers—such as group creation, role assignment, or association with an existing organisation—are introduced as needed. This model accommodates growth, avoids unnecessary complexity at the outset, and aligns better with modern expectations for usability and interoperability.

Provisioning, then, must support a continuum—from unaffiliated individuals, to emergent group structures, to institutionally governed users—while maintaining trust, clarity, and accountability.

# Provisioning Models and Patterns

Provisioning is not a single mechanism but a set of patterns that suit different contexts, each with trade-offs between ease of access, security, trust, and administrative overhead. The models below represent common approaches to introducing users into a system and associating them with groups and roles. Most systems will use a combination of these patterns over time.

## Direct Provisioning by Service Provider (Manual or Assisted)

In this model, a user is added directly by an administrator or support representative. This may be through a control panel, admin API, or manual entry—commonly triggered by a phone call, helpdesk request, or upstream service agreement.

Advantage:  
Guaranteed accuracy and oversight. Ensures that the system always has an initial, known user to manage or configure access. Useful in trust-bound or regulated environments, or during setup.

Disadvantage:  
Labour-intensive, not scalable. Prone to delay or inconsistency if not well-governed. Often lacks self-service for the user.

Consideration:  
Every system must support at least one form of first-user creation. Whether temporary or permanent, this establishes the minimum trust anchor needed to enable other provisioning models.

## Self-Registration (Individual Sign-Up)

This model allows users to create their own accounts directly, without prior invitation or validation beyond identity proof (e.g. email verification). It is the default model in most public-facing services.

It supports the formation of new groups or organisations later, often with the first user becoming the initial group administrator or billing contact.

**Advantage:**Minimises barriers to entry. Well-suited to organic growth and early adoption.

**Disadvantage:**Requires moderation controls to prevent spam, impersonation, or unverified use. Risk of data fragmentation if individuals create duplicate groups.

**Consideration:**Should be paired with clear escalation paths for linking to existing groups, merging duplicates, or upgrading access.

## Bulk Import

Administrators upload structured data (e.g. CSV or JSON) containing multiple user entries, roles, and affiliations. Commonly used during institutional onboarding or system migration.

Advantage:  
Efficient for provisioning large numbers of users with known structure.

Disadvantage:  
High risk of error, difficult to validate user intent, introduces stale or unclaimed records.

Consideration:  
Must be controlled by a trusted actor—usually a manually provisioned administrator. Always requires at least one user to exist to initiate the process.

## Invitation-Based Provisioning

In this model, an existing user or system administrator invites others to join a group or service. The invitation is typically linked to a role and expires after a set time (e.g. 48 hours).

Advantage:  
Supports secure onboarding and pre-determined role assignment. Ensures accountability and traceability of who brought whom into the system.

Disadvantage:  
Invitation links can expire or be misdelivered. Not suitable for open access models.

Consideration:  
Support re-issue, tracking of invite status, and mechanisms to validate intent on arrival. The first user to establish a new group or account typically follows a self-registration approach and is then authorised to issue invitations to others.

## Bulk Import

Administrators upload structured files (e.g. CSV or JSON) containing lists of users and associated roles, groups, or metadata. This method is commonly used during system rollout or when onboarding known cohorts.

Advantage:  
Efficient for loading large sets of known users with pre-assigned roles. Enables initial configuration of system access at scale.

Disadvantage:  
Error-prone, difficult to verify intent or accuracy at scale, and often disconnected from the user's actual arrival. Privacy risks are elevated, especially if files are misrouted or handled insecurely. Easily leads to stale records, orphaned users, or unintended access if not tightly governed.

Consideration:  
Use only when user identity and role can be confidently asserted from upstream data. Strongly prefer pairing with invitation or activation flows. Require validation, dry-run previews, and secure handling. Bulk import must not be used as a surrogate for trust or identity proof.

## Federated Just-In-Time (JIT) Provisioning

Users sign in via a federated Identity Provider (IdP), such as a school, employer, or government directory. If no account exists, one is created at the moment of login using attributes from the IdP.

Advantage:  
Eliminates duplicate account creation. Simplifies access for users with upstream identity.

Disadvantage:  
Lacks contextual awareness—users may not be assigned to correct groups or roles.

Consideration:  
Should be used with role-mapping, onboarding flows, or continuity resolution services to place users correctly.

## SCIM-Based Provisioning

The service accepts user and group information pushed automatically from an external system via the SCIM protocol.

Advantage:  
Enables lifecycle management from a central authority. Scalable and consistent.

Disadvantage:  
Requires strong governance and technical setup. Prone to silent errors if not monitored.

Consideration:  
SCIM must be configured by an authorised actor, typically a first user or admin. It is never self-starting.

## Delegated Group Management

A trusted user is granted rights to create groups and manage others. Useful for nested team structures, schools, or departmental use.

Advantage:  
Supports decentralised management and peer organisation.

Disadvantage:  
Can result in inconsistent structure, shadow groups, or misassigned roles.

Consideration:  
Always relies on prior provisioning of a trusted user to delegate authority. It is not a primary provisioning path but a secondary governance pattern.

## Broker-Mediated Provisioning

A third-party broker handles identity and role translation between external systems and the local service, provisioning users as they appear.

Advantage:  
Centralises federation logic and reduces local integration complexity.

Disadvantage:  
Can obscure provenance or lose user-level intent. Breaks when brokers change or upstream identity shifts.

Consideration:  
Broker configuration and monitoring requires an initial provisioning step, handled manually or through invitation by an authorised user.

Appendices

Appendix A - Document Information

Authors & Collaborators

* Sky Sigal, Solution Architect

### Versions

* 1. Initial Draft

### Images

[Figure 1: TODO Image 2](#_Toc144995112)

### Tables

[Table 1: TODO Table 3](#_Toc145048484)

[Table 2: TODO Table 2 3](#_Toc145048485)

### References

**There are no sources in the current document.**

### Review Distribution

The document was distributed for review as below:

|  |  |
| --- | --- |
| Identity | Notes |
|  |  |
|  |  |
|  |  |

### Audience

The document is technical in nature, but parts are expected to be read and/or validated by a non-technical audience.

### Structure

Where possible, the document structure is guided by either ISO-\* standards or best practice.

### Diagrams

Diagrams are developed for a wide audience. Unless specifically for a technical audience, where the use of industry standard diagram types (ArchiMate, UML, C4), is appropriate, diagrams are developed as simple “box & line” monochrome diagrams.

### Acronyms

API

: [Application Programming Interface](#Term_ApplicationProgrammingInterface).

DDD

: Domain Driven Design

GUI

: [Graphical User Interface](#Term_ApplicationProgrammingInterface). A form of [UI](#Acronym_UI).

ICT

: acronym for Information & Communication Technology, the domain of defining Information elements and using technology to automate their communication between entities. [IT](#Acronym_IT) is a subset of ICT.

IT

: acronym for Information, using Technology to automate and facilitate its management.

UI

: User Interface. Contrast with [API](#Acronym_API).

### Terms

Refer to the project’s Glossary.

Application Programming Interface

: an Interface provided for other systems to invoke (as opposed to User Interfaces).

Capability

: a capability is what an organisation or system must be able to achieve to meet its goals. Each capability belongs to a domain and is realised through one or more functions that, together, deliver the intended outcome within that area of concern.

Domain

: a domain is a defined area of knowledge, responsibility, or activity within an organisation or system. It groups related capabilities, entities, and functions that collectively serve a common purpose. Each capability belongs to a domain, and each function operates within one.

Entity

: an entity is a core object of interest within a domain, usually representing a person, place, thing, or event that holds information and can change over time, such as a Student, School, or Enrolment.

Function

: a function is a specific task or operation performed by a system, process, or person. Functions work together to enable a capability to be carried out. Each function operates within a domain and supports the delivery of one or more capabilities.

Person

: a physical person, who has one or more Personas. Not necessarily a system User.

Persona

: a facet that a Person presents to a Group of some kind.

Quality

: a quality is a measurable or observable attribute of a system or outcome that indicates how well it meets expectations. Examples include reliability, usability, and performance. Refer to the ISO-25000 SQuaRE series of standards.

User

: a human user of a system via its UIs.

User Interface

: a system interface intended for use by system users. Most computer system UIs are Graphics User Interfaces ([GUI](#Acronym_GUI)) or Text/Console User Interfaces (TUI).